



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

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

(11)



1-0038902

(51)^{2020.01} H04N 19/126; H04N 19/124; H04N
19/60; H04N 19/50; H04N 19/105

(13) B

(21) 1-2021-05319

(22) 21/01/2013

(62) 1-2017-05070

(86) PCT/KR2013/000493 21/01/2013

(87) WO2013/109127 25/07/2013

(30) 10-2012-0006564 20/01/2012 KR; 10-2012-0011672 06/02/2012 KR; 10-2012-0013462 09/02/2012 KR; 10-2012-0013996 10/02/2012 KR; 10-2012-0038971 16/04/2012 KR; 10-2013-0006736 21/01/2013 KR

(45) 26/02/2024 431

(43) 25/11/2021 404

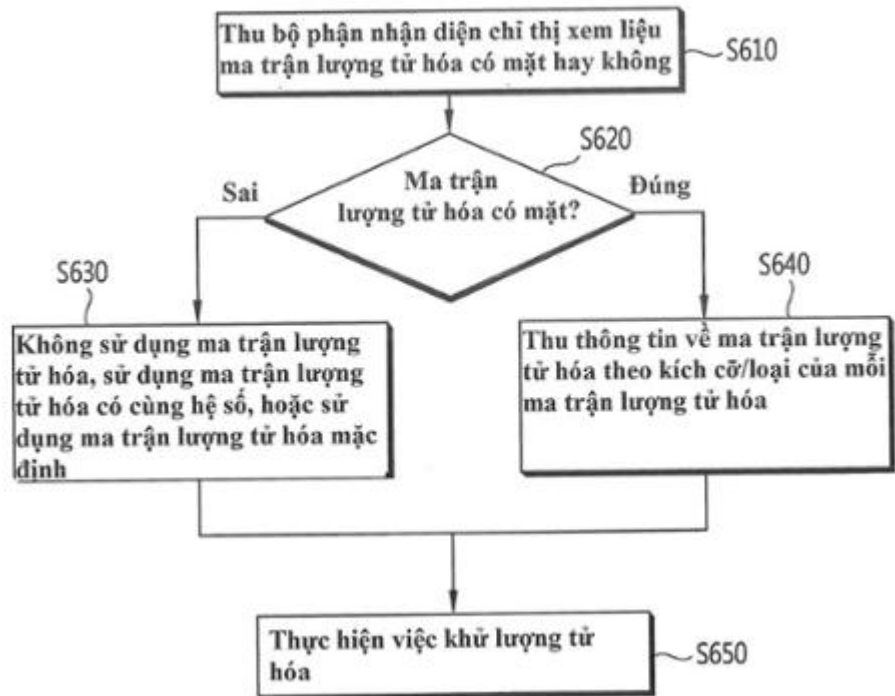
(73) ELECTRONICS AND TELECOMMUNICATIONS RESEARCH INSTITUTE (KR)
161 Gajeong-dong, Yuseong-gu, Daejeon-si 305-700, Republic of Korea

(72) LIM, Sung Chang (KR); KIM, Hui Yong (KR); LEE, Jin Ho (KR); CHOI, Jin Soo (KR); KIM, Jin Woong (KR).

(74) Công ty TNHH Tầm nhìn và Liên danh (VISION & ASSOCIATES CO.LTD.)

(54) PHƯƠNG PHÁP MÃ HÓA VIDEO, PHƯƠNG PHÁP GIẢI MÃ VIDEO VÀ VẬT GHI ĐỌC ĐƯỢC BẰNG MÁY TÍNH

(57) Sáng chế đề cập đến phương pháp giải mã video bao gồm các bước: xác định phương pháp dự đoán ma trận lượng tử hóa được sử dụng trong lượng tử hóa nghịch đảo; và giải mã ma trận lượng tử hóa được sử dụng trong lượng tử hóa nghịch đảo theo phương pháp dự đoán ma trận lượng tử hóa, trong đó phương pháp dự đoán ma trận lượng tử hóa là phương pháp bất kỳ trong số phương pháp dự đoán giữa các hệ số trong ma trận lượng tử hóa và phương pháp sao chép ma trận lượng tử hóa, trong đó trong giải mã ma trận lượng tử hóa, đáp lại phương pháp dự đoán ma trận lượng tử hóa là phương pháp dự đoán giữa các hệ số trong ma trận lượng tử hóa, giá trị hệ số ma trận DC thu được đối với ma trận lượng tử hóa có kích thước và giá trị ban đầu đối với hệ số ma trận lượng tử hóa được suy ra dựa trên giá trị hệ số ma trận DC, trong đó trong giải mã ma trận lượng tử hóa, đáp lại phương pháp dự đoán ma trận lượng tử hóa là phương pháp sao chép ma trận lượng tử hóa, thành phần cú pháp được giải mã, thành phần cú pháp chỉ báo xem ma trận lượng tử hóa tham chiếu được sử dụng hay ma trận lượng tử hóa mặc định được sử dụng làm ma trận lượng tử hóa, trong đó thành phần cú pháp được sử dụng chung cho chức năng thứ nhất để chỉ báo giá trị chênh lệch giữa giá trị để định rõ ma trận lượng tử hóa đích đang giải mã và giá trị để định rõ ma trận lượng tử hóa tham chiếu, và chức năng thứ hai để chỉ báo rằng ma trận lượng tử hóa được giải mã từ ma trận lượng tử hóa mặc định nếu giá trị của thành phần cú pháp là 0. Sáng chế còn đề cập đến phương pháp mã hóa video và vật ghi đọc được bằng máy tính.



Lĩnh vực kỹ thuật được đề cập

Sáng chế liên quan tới công nghệ mã hóa và giải mã hình ảnh và, cụ thể hơn, tới các phương pháp và các thiết bị mã hóa và giải mã ma trận lượng tử hóa.

Tình trạng kỹ thuật của sáng chế

Dịch vụ phát rộng có độ phân giải HD (High Definition – độ phân giải cao) hiện đã mở rộng trên quy mô toàn cầu cũng như ở quy mô địa phương. Do đó, nhiều người sử dụng đã quen với hình ảnh có độ phân giải cao và chất lượng hình ảnh cao, và có nhiều cơ quan đã thúc đẩy sự phát triển của thiết bị hình ảnh thế hệ tiếp theo.

Do ngày càng có nhiều sự quan tâm tới độ phân giải siêu cao (Ultra High Definition - UHD) có độ phân giải cao hơn bốn lần so với HDTV cũng như so với HDTV, nên cần có kỹ thuật nén cho hình ảnh có độ phân giải cao hơn và chất lượng hình ảnh cao hơn.

Để nén hình ảnh, thông tin về các điểm ảnh của hình ảnh hiện tại có thể được mã hóa nhờ việc dự đoán. Ví dụ, kỹ thuật dự đoán liên ảnh trong đó, giá trị điểm ảnh có trong hình ảnh hiện tại được dự đoán từ hình ảnh tạm thời trước đó và/hoặc hình ảnh tạm thời sau đó và kỹ thuật dự đoán nội ảnh trong đó, giá trị điểm ảnh có trong hình ảnh hiện tại được dự đoán sử dụng thông tin về các điểm ảnh nằm trong hình ảnh hiện tại có thể được sử dụng.

Hơn nữa, hiệu quả mã hóa có thể được cải thiện và lượng thông tin truyền tải có thể được giảm bớt sử dụng công nghệ mã hóa entropy mà trong đó, ký hiệu ngắn được gán cho ký hiệu có tần số xuất hiện cao và ký hiệu dài được gán cho ký hiệu có tần số xuất hiện thấp.

Trong trường hợp này, phương pháp thực hiện lượng tử hóa hệ số biến đổi cho khối còn dư được tạo ra bởi việc dự đoán hiệu quả hơn là vấn đề.

Bản chất kỹ thuật của sáng chế

Mục đích của sáng chế là đề xuất phương pháp và thiết bị để hạn chế mã hóa/giải

mã ma trận lượng tử hóa dựa trên kích thước của khối biến đổi có sẵn.

Mục đích khác của sáng chế là đề xuất các phương pháp và các thiết bị mã hóa/giải mã, trong đó, ma trận lượng tử hóa mặc định và ma trận lượng tử hóa không mặc định có thể được trộn và được sử dụng theo kích thước của mỗi khối biến đổi hoặc loại của ma trận lượng tử hóa nằm trong chuỗi, hình ảnh hoặc phiên.

Mục đích khác nữa của sáng chế là đề xuất các phương pháp và các thiết bị mã hóa/giải mã, trong đó, ma trận lượng tử hóa mặc định được sử dụng dựa trên ma trận lượng tử hóa tham chiếu ID.

Mục đích khác nữa của sáng chế là đề xuất các phương pháp và các thiết bị mã hóa/giải mã, tăng hiệu quả mã hóa bằng cách thực hiện việc dự đoán ma trận lượng tử hóa chỉ khi ma trận lượng tử hóa tham chiếu có mặt.

Mục đích khác nữa của sáng chế là đề xuất phương pháp và thiết bị để thực hiện việc dự đoán và mã hóa/giải mã trên hệ số ma trận DC một cách hiệu quả.

Mục đích khác nữa của sáng chế là đề xuất các phương pháp và các thiết bị mã hóa/giải mã để thực hiện việc dự đoán ma trận lượng tử hóa từ ma trận lượng tử hóa có cùng kích thước với ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện.

Mục đích khác nữa của sáng chế là đề xuất phương pháp và thiết bị để thực hiện việc dự đoán và mã hóa/giải mã dựa trên hệ số thứ nhất nằm trong ma trận lượng tử hóa.

Một phương án của sáng chế đề xuất phương pháp mã hóa ma trận lượng tử hóa, bao gồm các bước xác định và lượng tử hóa ma trận lượng tử hóa được sử dụng trong lượng tử hóa, xác định phương pháp dự đoán ma trận lượng tử hóa được sử dụng trong lượng tử hóa, và thông tin mã hóa về ma trận lượng tử hóa theo phương pháp dự đoán được xác định, trong đó, phương pháp dự đoán có thể là một phương pháp bất kỳ trong số phương pháp dự đoán, phương pháp dự đoán liên hệ số nằm trong ma trận lượng tử hóa và sao chép ma trận lượng tử hóa.

Phương án khác theo sáng chế đề xuất phương pháp giải mã ma trận lượng tử hóa, bao gồm bước xác định phương pháp dự đoán ma trận lượng tử hóa được sử dụng trong giải lượng tử hóa và giải mã ma trận lượng tử hóa được sử dụng trong giải lượng tử hóa

theo phương pháp dự đoán được xác định, trong đó, phương pháp dự đoán ma trận lượng tử hóa có thể là một hệ số bất kỳ trong phương pháp dự đoán liên hệ số nằm trong ma trận lượng tử hóa và việc sao chép ma trận lượng tử hóa.

Theo sáng chế, hiệu quả mã hóa có thể được cải thiện và mức độ phức tạp của việc tính toán có thể được giảm bớt bằng cách giới hạn mã hóa ma trận lượng tử hóa dựa trên kích thước của khối biến đổi là khả dụng.

Theo sáng chế, hiệu quả mã hóa có thể được cải thiện và mức độ tự do khi bộ mã hóa chọn ma trận lượng tử hóa có thể được tăng lên bằng cách trộn và sử dụng ma trận lượng tử hóa mặc định và ma trận lượng tử hóa không mặc định theo kích thước của mỗi khối biến đổi hoặc loại ma trận lượng tử hóa nằm trong chuỗi, hình ảnh hoặc phiên.

Theo sáng chế, hiệu quả mã hóa có thể được cải thiện, mức độ phức tạp của việc tính toán có thể được giảm bớt, và mức độ tự do khi bộ mã hóa chọn ma trận lượng tử hóa có thể được tăng lên bằng cách mã hóa/giải mã thông tin về việc liệu ma trận lượng tử hóa mặc định sẽ được sử dụng hay không dựa trên ma trận lượng tử hóa tham chiếu ID hoặc thực hiện việc dự đoán ma trận lượng tử hóa chỉ khi ma trận lượng tử hóa tham chiếu có mặt.

Theo sáng chế, hiệu quả mã hóa có thể được cải thiện, mức độ phức tạp của việc tính toán có thể được giảm bớt, và mức độ tự do khi bộ mã hóa chọn ma trận lượng tử hóa có thể được tăng lên bằng cách dự đoán và mã hóa/giải mã hệ số ma trận DC hoặc thực hiện việc dự đoán ma trận lượng tử hóa từ ma trận lượng tử hóa có cùng kích thước với ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện.

Hơn nữa, theo sáng chế, hiệu quả mã hóa có thể được cải thiện và mức độ phức tạp của việc tính toán có thể được giảm bớt bằng cách mã hóa/giải mã hệ số thứ nhất nằm trong ma trận lượng tử hóa sử dụng giá trị hệ số thường xuyên xuất hiện.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ khối thể hiện cấu trúc theo một phương án của thiết bị mã hóa hình ảnh trong đó, sáng chế được áp dụng.

Fig.2 là sơ đồ khối thể hiện cấu trúc theo một phương án của thiết bị giải mã hình

ảnh trong đó, sáng chế được áp dụng.

Fig.3 là sơ đồ minh họa sơ lược phương án trong đó, một bộ phận được phân mảnh thành nhiều đơn vị cấp thấp hơn.

Fig.4 là sơ đồ minh họa sơ lược phương pháp mã hóa hình ảnh theo sáng chế.

Fig.5 là sơ đồ minh họa sơ lược một ví dụ về việc vận hành của bộ giải mã để giải mã thông tin về ma trận lượng tử hóa và thực hiện giải mã bằng cách sử dụng thông tin được giải mã.

Fig.6 là sơ đồ minh họa sơ lược một ví dụ của phương pháp thực hiện giải lượng tử hóa theo sáng chế.

Fig.7 là sơ đồ minh họa sơ lược một ví dụ của phương pháp thu thông tin về ma trận lượng tử hóa khi ma trận lượng tử hóa có mặt trong bộ tham số và thực hiện giải lượng tử hóa bằng cách sử dụng thông tin này.

Fig.8 là sơ đồ minh họa sơ lược một ví dụ khác về phương pháp thu thông tin về ma trận lượng tử hóa khi ma trận lượng tử hóa có mặt trong bộ tham số và thực hiện giải lượng tử hóa bằng cách sử dụng thông tin này.

Mô tả chi tiết sáng chế

Dưới đây, các phương án của sáng chế được mô tả chi tiết có dựa vào các hình vẽ kèm theo. Hơn nữa, trong quá trình mô tả các phương án của sáng chế, phần mô tả chi tiết về các chức năng và các thành phần đã biết sẽ được bỏ qua nếu các mô tả ấy khiến cho bản chất sáng chế trở nên không rõ ràng.

Trong bản mô tả này, khi nói rằng một thành phần “được nối”, “được kết hợp”, hoặc “gắn kết” với thành phần khác, thì một thành phần có thể được nối hoặc gắn kết trực tiếp với thành phần khác, nhưng cần hiểu rằng thành phần thứ ba cũng có thể “được nối”, “được kết hợp”, hoặc “gắn kết” giữa hai thành phần. Hơn nữa, trong bản mô tả này, các nội dung mô tả rằng một thành phần cụ thể “được bao gồm (hoặc gồm có)” không có nghĩa là các thành phần khác với thành phần cụ thể này sẽ bị loại trừ, mà có nghĩa là các thành phần bổ sung có thể được bao gồm trong phương án áp dụng theo sáng chế hoặc phạm vi bảo hộ của sáng chế.

Các thuật ngữ, như “thứ nhất” và “thứ hai”, có thể được sử dụng để mô tả các thành phần khác nhau nhưng các thành phần không bị giới hạn ở các thuật ngữ này. Các thuật ngữ được sử dụng để phân biệt một thành phần này với thành phần khác. Ví dụ, thành phần thứ nhất có thể được đặt tên là thành phần thứ hai và tương tự thành phần thứ hai có thể được đặt tên là thành phần thứ nhất mà không nằm ngoài phạm vi của sáng chế.

Hơn nữa, các thành phần được mô tả trong các phương án của sáng chế được thể hiện một cách độc lập để chỉ báo các chức năng đặc trưng và khác nhau, và nó không có nghĩa là mỗi thành phần trong các thành phần được tạo thành từ phần cứng hoặc các mảnh môđun phần mềm tách biệt. Nghĩa là, các thành phần được bố trí, để tạo thuận tiện cho việc mô tả, và ít nhất hai thành phần trong các thành phần có thể được kết hợp để tạo thành một thành phần hoặc một thành phần có thể được chia thành nhiều thành phần và nhiều thành phần có thể thực hiện nhiều chức năng. Phương án trong đó, các thành phần được tạo thành từ hoặc mỗi thành phần trong các thành phần được chia thuộc phạm vi bảo hộ của sáng chế mà không tách khỏi bản chất của sáng chế.

Hơn nữa, trong sáng chế, một số thành phần có thể không phải là các thành phần thiết yếu để thực hiện các chức năng thiết yếu, nhưng có thể là các thành phần tùy chọn chỉ để nâng cao hiệu quả hoạt động. Sáng chế có thể được áp dụng chỉ sử dụng các thành phần thiết yếu để áp dụng bản chất của sáng chế hơn là các thành phần được sử dụng chỉ để cải thiện hiệu quả hoạt động, và cấu trúc chỉ bao gồm các thành phần thiết yếu khác với các thành phần tùy chọn được sử dụng chỉ để cải thiện hiệu quả hoạt động thuộc phạm vi bảo hộ của sáng chế.

Đầu tiên, các thuật ngữ được sử dụng trong bản mô tả này được mô tả ngắn gọn để tạo thuận tiện cho việc mô tả và giúp hiểu sáng chế.

Đơn vị có nghĩa là đơn vị mã hóa và giải mã hình ảnh. Theo cách khác, trong việc mã hóa/giải mã hình ảnh, bộ phận mã hóa hoặc đơn vị giải mã để chỉ đơn vị được phân mảnh khi một hình ảnh được chia và được mã hóa hoặc được giải mã. Đơn vị còn được gọi là khối, khối macro, bộ phận mã hóa, bộ phận dự đoán, đơn vị biến đổi, khối mã hóa, khối dự đoán, khối biến đổi hoặc dạng tương tự. Một bộ phận có thể được phân mảnh thành bộ phận nhỏ hơn cấp thấp hơn.

Đơn vị biến đổi là bộ phận cơ bản trong việc thực hiện mã hóa/giải mã của khối còn dư, như biến đổi, biến đổi ngược, việc lượng tử hóa, việc giải lượng tử hóa, và hệ số biến đổi mã hóa/giải mã. Một đơn vị biến đổi có thể được chia thành nhiều đơn vị biến đổi nhỏ hơn. Hơn nữa, đơn vị biến đổi có thể được sử dụng giống như khối biến đổi. Dạng bao gồm thành phần cú pháp có liên quan tới khối biến đổi cho các tín hiệu độ sáng và sắc độ có thể được gọi là đơn vị biến đổi.

Ma trận lượng tử hóa nghĩa là ma trận được sử dụng trong quy trình lượng tử hóa hoặc quy trình lượng tử hóa nghịch đảo để cải thiện chất lượng hình ảnh chủ quan hoặc chất lượng hình ảnh khách quan của hình ảnh. Ma trận lượng tử hóa còn được gọi là danh sách định cỡ và quy trình biến đổi ngược trùng với quy trình lượng tử hóa.

Ma trận lượng tử hóa được sử dụng trong lượng tử hóa/giải lượng tử hóa có thể được truyền dưới dạng dòng bit, và ma trận mặc định đã có sẵn trong bộ mã hóa và/hoặc bộ giải mã có thể được sử dụng làm ma trận lượng tử hóa. Thông tin về ma trận lượng tử hóa có thể được truyền trong cụm theo kích thước của mỗi ma trận lượng tử hóa hoặc kích thước của khối biến đổi trong đó, ma trận lượng tử hóa được áp dụng qua bộ tham số chuỗi (SPS) hoặc bộ tham số hình ảnh (PPS). Ví dụ, các ma trận lượng tử hóa 4x4 cho khối biến đổi 4x4 có thể được truyền, các ma trận 8x8 cho khối biến đổi 8x8 có thể được truyền, các ma trận 16x16 cho khối biến đổi 16x16 có thể được truyền, và các ma trận 32x32 cho khối biến đổi 32x32 có thể được truyền.

Ma trận lượng tử hóa được áp dụng cho khối hiện tại có thể là (1) thu được bằng cách sao chép ma trận lượng tử hóa có cùng kích thước và (2) được tạo ra nhờ việc dự đoán từ ma trận hệ số trước đó nằm trong ma trận lượng tử hóa. Ma trận có cùng kích thước có thể là ma trận lượng tử hóa đã được mã hóa hoặc được giải mã hoặc được chọn từ trước, ma trận lượng tử hóa tham chiếu, hoặc ma trận lượng tử hóa mặc định. Hoặc, ma trận có cùng kích thước có thể được xác định một cách chọn lọc bởi tổ hợp bao gồm ít nhất hai trong số các ma trận lượng tử hóa đã được mã hóa hoặc được giải mã hoặc được sử dụng từ trước, ma trận lượng tử hóa tham chiếu, và ma trận lượng tử hóa mặc định.

Bộ tham số tương ứng với thông tin về phần đầu của cấu trúc trong dòng bit và có nghĩa là nó mô tả chung cho bộ tham số chuỗi, bộ tham số hình ảnh, bộ tham số tương

thích, v.v..

Thông số lượng tử hóa là giá trị được sử dụng trong lượng tử hóa và giải lượng tử hóa và có thể là giá trị được ánh xạ tới kích thước bước lượng tử hóa.

Ma trận mặc định có thể là ma trận lượng tử hóa cụ thể đã được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ma trận lượng tử hóa mặc định sẽ được mô tả sau trong bản mô tả này có thể được sử dụng cùng một ý nghĩa như ma trận mặc định. Ma trận không phải là ma trận mặc định có thể là ma trận lượng tử hóa không được xác định trong bộ mã hóa và/hoặc bộ giải mã và được truyền từ bộ mã hóa tới bộ giải mã, nghĩa là, được truyền/được nhận bởi người sử dụng. Ma trận lượng tử hóa không mặc định sẽ được mô tả sau trong bản mô tả này có thể được sử dụng cùng một ý nghĩa như ma trận không phải là ma trận mặc định.

Fig.1 là sơ đồ khối thể hiện kết cấu theo một phương án của thiết bị mã hóa hình ảnh trong đó, sáng chế được áp dụng.

Đề cập tới Fig.1, thiết bị mã hóa hình ảnh 100 bao gồm môđun dự đoán chuyển động 111, môđun bù chuyển động 112, môđun dự đoán nội ảnh 120, bộ chuyển mạch 115, bộ trừ 125, môđun biến đổi 130, môđun lượng tử hóa 140, môđun mã hóa entropy 150, môđun giải lượng tử hóa (lượng tử hóa nghịch đảo) 160, môđun biến đổi ngược 170, bộ cộng 175, môđun lọc 180, và bộ đệm hình ảnh tham chiếu 190.

Thiết bị mã hóa hình ảnh 100 có thể thực hiện mã hóa trên hình ảnh đưa vào trong chế độ nội ảnh hoặc chế độ liên ảnh và đưa ra dòng bit. Các phương tiện dự đoán nội ảnh việc dự đoán nội ảnh khung, và các phương tiện dự đoán liên ảnh việc dự đoán liên khung. Trong trường hợp của chế độ nội ảnh, chuyển mạch 115 có thể được chuyển mạch sang chế độ nội ảnh. Trong trường hợp của chế độ liên ảnh, chuyển mạch 115 có thể được chuyển mạch sang chế độ liên ảnh. Sau khi tạo ra khối dự đoán cho khối đầu vào của hình ảnh đưa vào, thiết bị mã hóa hình ảnh 100 có thể mã hóa chênh lệch giữa khối đầu vào và khối dự đoán. Ở đây, hình ảnh đưa vào có thể là hình ảnh gốc.

Trong trường hợp của chế độ nội ảnh, môđun dự đoán nội ảnh 120 có thể tạo ra khối dự đoán bằng cách thực hiện việc dự đoán không gian dựa trên giá trị điểm ảnh của hiện

được mã hóa khối lân cận khối hiện tại.

Trong trường hợp của chế độ liên ảnh, môđun dự đoán chuyển động 111 có thể tìm kiếm hình ảnh tham chiếu, được lưu trong bộ đệm hình ảnh tham chiếu 190, cho vùng phù hợp tốt nhất với khối đầu vào trong quy trình dự đoán chuyển động và thu véc tơ chuyển động dựa trên vùng được truy hồi. Môđun bù chuyển động 112 có thể tạo ra khối dự đoán bằng cách thực hiện việc bù chuyển động sử dụng véc tơ chuyển động. Ở đây, véc tơ chuyển động là véc tơ hai chiều được sử dụng trong việc dự đoán liên ảnh, và véc tơ chuyển động có thể chỉ báo dịch chuyển giữa khối hiện tại và khối nằm trong hình ảnh tham chiếu.

Bộ trừ 125 có thể tạo ra khối còn dư dựa trên chênh lệch giữa khối đầu vào và khối dự đoán được tạo ra. Môđun biến đổi 130 có thể kết xuất hệ số biến đổi bằng cách thực hiện biến đổi trên khối còn dư. Tiếp theo, môđun lượng tử hóa 140 có thể kết xuất hệ số được lượng tử hóa bằng cách lượng tử hóa hệ số biến đổi nhận được sử dụng ít nhất một trong số thông số lượng tử hóa và ma trận lượng tử hóa. Ở đây, ma trận lượng tử hóa có thể được nhập tới bộ mã hóa, và nó có thể được xác định rằng ma trận lượng tử hóa được nhập vào được sử dụng trong bộ mã hóa.

Môđun mã hóa entropy 150 có thể đưa ra dòng bit bằng cách thực hiện việc mã hóa entropy dựa trên các giá trị được tính toán bởi môđun lượng tử hóa 140 hoặc mã hóa giá trị thông số, v.v., được tính toán trong quy trình mã hóa. Nếu việc mã hóa entropy được áp dụng, thì các ký hiệu có thể được thể hiện bằng cách gán một số lượng nhỏ bit cho ký hiệu có khả năng xuất hiện cao và một số lượng lớn bit cho ký hiệu có khả năng xuất hiện thấp để làm giảm kích thước của dòng bit cho việc mã hóa các ký hiệu để được mã hóa. Do đó, hiệu suất nén của việc mã hóa hình ảnh có thể được tăng lên nhờ việc mã hóa entropy. Môđun mã hóa entropy 150 có thể sử dụng mã hóa các phương pháp, như mã hóa Golomb hàm mũ, mã hóa chiều dài thay đổi tương thích hoàn cảnh (Context-Adaptive Variable Length Coding - CAVLC), và mã hóa thuật toán nhị phân tương thích hình ảnh (Context-Adaptive Binary Arithmetic Coding - CABAC), cho việc mã hóa entropy.

Thiết bị mã hóa hình ảnh (dưới đây được gọi là bộ mã hóa) theo phương án trên Fig.1 thực hiện việc dự đoán liên ảnh mã hóa, nghĩa là, việc dự đoán liên khung mã hóa, và do đó hình ảnh hiện được mã hóa được giải mã và được lưu được sử dụng làm hình ảnh

tham chiếu. Do đó, hệ số được lượng tử hóa được cho giải lượng tử hóa bởi môđun giải lượng tử hóa 160 và được đưa vào biến đổi ngược bởi môđun biến đổi ngược 170. Hệ số lượng tử hóa nghịch đảo và biến đổi ngược trở thành khối còn dư đã được tái tạo, và khối còn dư đã được tái tạo được thêm vào khối dự đoán qua bộ cộng 175, nhờ đó tạo ra khối được tái tạo.

Khối được tái tạo đi qua môđun lọc 180. Môđun bộ lọc 180 có thể áp dụng một hoặc nhiều thành phần trong số bộ lọc khử khối, dịch chuyển tương thích mẫu (Sample Adaptive Offset - SAO), và bộ lọc vòng lặp tương thích (Adaptive Loop Filter - ALF) tới khối được tái tạo hoặc hình ảnh được tái tạo. Môđun bộ lọc 180 có thể cũng được gọi là bộ lọc trong vòng lặp. Bộ lọc khử khối có thể loại bỏ sự nhiễu của khối xuất hiện tại biên của khối. SAO có thể thêm giá trị dịch chuyển phù hợp tới giá trị điểm ảnh để bù cho lỗi mã hóa. ALF có thể thực hiện việc lọc dựa trên giá trị thu được bằng cách so sánh hình ảnh được tái tạo với hình ảnh gốc. Khối được tái tạo đi qua môđun bộ lọc 180 có thể được lưu trong bộ đệm hình ảnh tham chiếu 190.

Fig.2 là sơ đồ khối thể hiện việc tạo thành theo một phương án của thiết bị giải mã hình ảnh trong đó, sáng chế được áp dụng.

Đề cập tới Fig.2, thiết bị giải mã hình ảnh 200 bao gồm môđun giải mã entropy 210, môđun giải lượng tử hóa (lượng tử hóa nghịch đảo) 220, môđun biến đổi ngược 230, môđun dự đoán nội ảnh 240, môđun bù chuyển động 250, bộ cộng 255, môđun bộ lọc 260, và bộ đệm hình ảnh tham chiếu 270.

Thiết bị giải mã hình ảnh 200 có thể nhận dòng bit được đưa ra từ bộ mã hóa, thực hiện giải mã trên dòng bit trong chế độ nội ảnh hoặc chế độ liên ảnh, và đưa ra hình ảnh được tái tạo. Trong trường hợp của chế độ nội ảnh, chuyển mạch có thể được chuyển mạch sang chế độ nội ảnh. Trong trường hợp của chế độ liên ảnh, chuyển mạch có thể được chuyển mạch sang chế độ liên ảnh. Thiết bị giải mã hình ảnh 200 có thể thu khối còn dư đã được tái tạo từ dòng bit nhận được, tạo ra khối dự đoán, và tạo ra khối được tái tạo bằng cách thêm khối còn dư đã được tái tạo tới khối dự đoán.

Môđun giải mã entropy 210 có thể tạo ra các ký hiệu bao gồm ký hiệu có dạng hệ số được lượng tử hóa bằng cách thực hiện giải mã entropy trên dòng bit nhập vào theo phân

bổ khả năng. Phương pháp giải mã entropy là tương tự như phương pháp mã hóa entropy được mô tả ở trên.

Nếu phương pháp giải mã entropy được áp dụng thì các ký hiệu có thể được thể hiện bằng cách chỉ định các số bit nhỏ cho ký hiệu có khả năng xuất hiện cao và các số bit lớn tới ký hiệu có khả năng xuất hiện thấp để làm giảm kích thước của dòng bit cho mỗi ký hiệu.

Hệ số được lượng tử hóa có thể được đưa vào giải lượng tử hóa bởi môđun giải lượng tử hóa 220 dựa trên thông số lượng tử hóa và có thể được đưa vào biến đổi ngược bởi môđun biến đổi ngược 230. Kết quả là của việc giải lượng tử hóa/biến đổi ngược của hệ số được lượng tử hóa, khối còn dư đã được tái tạo có thể được tạo ra.

Ma trận lượng tử hóa được sử dụng trong giải lượng tử hóa cũng được gọi là danh sách định cỡ. Môđun giải lượng tử hóa 220 có thể tạo ra hệ số được lượng tử hóa nghịch đảo bằng cách áp dụng ma trận lượng tử hóa tới hệ số được lượng tử hóa.

Ở đây, môđun giải lượng tử hóa 220 có thể thực hiện giải lượng tử hóa đáp lại việc lượng tử hóa được áp dụng bởi bộ mã hóa. Ví dụ, môđun giải lượng tử hóa 220 có thể thực hiện giải lượng tử hóa bằng cách áp dụng ma trận lượng tử hóa, được áp dụng bởi bộ mã hóa, tới hệ số được lượng tử hóa nghịch đảo.

Ma trận lượng tử hóa được sử dụng trong giải lượng tử hóa bởi thiết bị giải mã hình ảnh 200 (dưới đây còn được gọi là bộ giải mã) có thể là nhận được từ dòng bit, và ma trận mặc định đã có sẵn trong bộ mã hóa và/hoặc bộ giải mã có thể được sử dụng làm ma trận lượng tử hóa. Thông tin về ma trận lượng tử hóa được truyền có thể được nhận theo khối theo kích thước của mỗi ma trận lượng tử hóa hoặc kích thước của khối biến đổi với nó ma trận lượng tử hóa đi qua bộ tham số chuỗi hoặc bộ tham số hình ảnh. Ví dụ, các ma trận lượng tử hóa 4x4 cho khối 4x4 biến đổi có thể nhận được, các ma trận 8x8 cho khối biến đổi 8x8 có thể nhận được, các ma trận 16x16 cho khối biến đổi 16x16 có thể nhận được, và các ma trận 32x32 cho khối 32x32 biến đổi có thể nhận được.

Trong trường hợp của chế độ nội ảnh, môđun dự đoán nội ảnh 240 có thể tạo ra khối dự đoán bằng cách thực hiện việc dự đoán không gian sử dụng giá trị điểm ảnh của hiện đã

được giải mã khối lân cận khối hiện tại. Trong trường hợp của chế độ liên ảnh, môđun bù chuyển động 250 có thể tạo ra khối dự đoán bằng cách thực hiện việc bù chuyển động bằng cách sử dụng véc tơ chuyển động và hình ảnh tham chiếu được lưu trong bộ đệm hình ảnh tham chiếu 270.

Khối còn dư đã được tái tạo và khối dự đoán được thêm cùng với nhau bởi bộ cộng 255, và khối được thêm có thể được đưa qua môđun bộ lọc 260. Môđun bộ lọc 260 có thể áp dụng một hoặc nhiều trong số bộ lọc khử khối, SAO, và ALF tới khối được tái tạo hoặc hình ảnh được tái tạo. Môđun bộ lọc 260 có thể đưa ra hình ảnh được tái tạo, nghĩa là, hình ảnh được phục hồi. Hình ảnh được tái tạo có thể được lưu trong bộ đệm hình ảnh tham chiếu 270 và được sử dụng trong việc dự đoán liên ảnh.

Trong khi đó, thông tin phân mảnh khối có thể bao gồm thông tin về độ sâu của đơn vị. Thông tin độ sâu có thể chỉ báo số lần mà đơn vị được phân mảnh và/hoặc mức độ mà đơn vị được phân mảnh.

Fig.3 là sơ đồ minh họa sơ lược phương án trong đó, một đơn vị được phân mảnh thành nhiều đơn vị cấp thấp hơn.

Một đơn vị hoặc khối có thể được phân mảnh theo thứ bậc dựa trên cấu trúc cây với thông tin độ sâu. Mỗi đơn vị cấp thấp hơn được phân mảnh có thể có thông tin độ sâu. Thông tin độ sâu có thể bao gồm thông tin về kích thước của đơn vị cấp thấp hơn do nó chỉ báo số lần mà đơn vị được phân mảnh và/hoặc mức độ mà đơn vị được phân mảnh.

Đề cập tới số chỉ dẫn 310 trên Fig.3, nút cao nhất có thể được gọi là nút gốc, và nút cao nhất có thể có giá trị độ sâu nhỏ nhất. Ở đây, nút cao nhất có thể có độ sâu ở mức 0 và có thể chỉ báo đơn vị thứ nhất không được phân mảnh.

Nút cấp thấp hơn có độ sâu ở mức 1 có thể chỉ báo đơn vị được phân mảnh từ đơn vị thứ nhất theo một lần, và nút cấp thấp hơn có độ sâu ở mức 2 có thể chỉ báo đơn vị được phân mảnh từ nút thứ nhất theo hai lần. Ví dụ, trong 320 trên Fig.3, đơn vị 'a' tương ứng với nút 'a' là đơn vị được phân mảnh từ nút thứ nhất theo một lần, và đơn vị 'a' có thể có độ sâu ở mức 1.

Nút lá ở mức 3 có thể chỉ báo đơn vị được phân mảnh từ nút thứ nhất theo ba lần.

Ví dụ, trên 320 trên Fig.3, đơn vị 'd' tương ứng với nút 'd' là đơn vị được phân mảnh từ nút thứ nhất theo ba lần, và nút 'd' có thể có độ sâu ở mức 3. Do đó, nút lá ở mức 3, nghĩa là, nút thấp nhất, có thể có độ sâu sâu nhất.

Các phương pháp mã hóa/giải mã sơ lược đã được mô tả. Các phương pháp mã hóa/giải mã ma trận lượng tử hóa được sử dụng trong lượng tử hóa và các quy trình lượng tử hóa nghịch đảo của các quy trình mã hóa/giải mã có ảnh hưởng lớn đến hiệu quả mã hóa giống như các quy trình mã hóa/giải mã khác. Do đó, cần thiết để cải thiện việc lượng tử hóa/giải lượng tử hóa bằng cách lấy hiệu quả mã hóa để xem xét.

Cụ thể hơn, thông thường, ma trận lượng tử hóa cho tất cả các biến đổi được mã hóa/được giải mã bằng cách không lấy kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi có sẵn để xem xét. Hơn nữa, thông thường, ma trận lượng tử hóa áp dụng trong cụm mà không trộn lẫn và sử dụng ma trận mặc định và ma trận không phải là ma trận mặc định theo kích thước của việc biến đổi hoặc loại ma trận lượng tử hóa nằm trong chuỗi, hình ảnh hoặc phiến. Do đó, có nhược điểm là mức độ tự do khi bộ mã hóa chọn ma trận lượng tử hóa là thấp và hiệu quả mã hóa là thấp do ma trận mặc định không cần thiết để được mã hóa/được giải mã phải được mã hóa và được truyền.

Như được mô tả ở trên, theo các phương pháp thông thường, khi áp dụng ma trận lượng tử hóa, mức độ tự do và hiệu quả mã hóa là thấp và mức độ phức tạp là cao.

Để giải quyết các vấn đề nêu trên, phương pháp sử dụng một cách hiệu quả ma trận lượng tử hóa cần được xem xét để cải thiện hiệu quả mã hóa và làm giảm mức độ phức tạp trong việc lượng tử hóa/giải lượng tử hóa.

Fig.4 là sơ đồ minh họa sơ lược phương pháp mã hóa hình ảnh theo sáng chế.

Đề cập tới Fig.4, bộ mã hóa xác định thông tin về kích thước của đơn vị biến đổi của chuỗi hoặc hình ảnh hiện tại và mã hóa thông tin ở bước S410.

Thông tin về kích thước của đơn vị biến đổi nghĩa là ít nhất một trong số kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi. Bộ mã hóa có thể xác định kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi khi mã hóa hình ảnh.

Ví dụ, bộ mã hóa có thể xác định kích thước tối thiểu của đơn vị biến đổi bình

phương như là khối 4x4 hoặc có thể xác định kích thước tối đa của đơn vị biến đổi bình phương như là khối 32x32. Hơn nữa, bộ mã hóa có thể xác định kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi bình phương như là khối 4x4 khối và khối 32x32, một cách tương ứng. Bộ mã hóa có thể thực hiện mã hóa theo kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi bình phương.

Bộ mã hóa có thể thực hiện việc mã hóa entropy trên được xác định thông tin về kích thước của đơn vị biến đổi cho dòng bit. Ví dụ, bộ mã hóa có thể mã hóa được xác định thông tin về kích thước của đơn vị biến đổi thành bộ tham số trong dòng bit.

Như được mô tả ở trên, thông tin về kích thước của đơn vị biến đổi đề cập tới thông tin về ít nhất một trong số kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi. Do đó, bộ mã hóa có thể thông tin mã hóa về kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi thành dòng bit như là thông tin về kích thước của đơn vị biến đổi. Ở đây, kích thước tối đa của đơn vị biến đổi có thể được định rõ sử dụng giá trị chênh lệch giữa kích thước tối đa của việc biến đổi và kích thước tối thiểu của đơn vị biến đổi.

Bảng 1 thể hiện sơ lược một ví dụ của thông tin về kích thước của đơn vị biến đổi đã được mã hóa entropy thành bộ tham số chuỗi của dòng bit.

Bảng 1

<code>seq_parameter set rbsp() {</code>	Bộ mô tả
<code>...</code>	
<code>log2_min_transform_block_size_minus2</code>	ue(v)
<code>log2_diff_max_min_transform_block_size</code>	ue(v)
<code>...</code>	
<code>}</code>	

Như trong thành phần cú pháp được minh họa trên bảng 1, sau khi tính toán `Log2MinTrafoSize` bằng cách áp dụng hàm `Log2` tới kích thước nằm ngang hoặc thẳng đứng tối thiểu của đơn vị biến đổi bình phương, bộ mã hóa có thể định rõ giá trị thu được bằng cách trừ 2 từ `Log2MinTrafoSize` bằng cách sử dụng `log2_min_transform_block_size_minus2`. Hơn nữa, sau khi tính toán `Log2MaxTrafoSize` bằng cách áp dụng hàm `Log2` tới kích thước nằm ngang hoặc thẳng đứng tối đa của đơn vị biến đổi bình phương, bộ mã hóa có thể xác định giá trị chênh lệch giữa `Log2MaxTrafoSize`

và Log2MinTrafoSize bằng cách sử dụng `log2_diff_max_min_transform_block_size`. Bộ mã hóa có thể mã hóa `log2_min_transform_block_size_minus2` và `log2_diff_max_min_transform_block_size`, nghĩa là, thành phần cú pháp, thành dòng bit và gửi dòng bit kết quả tới bộ giải mã. Theo cách khác, bộ mã hóa có thể mã hóa các giá trị được chỉ báo bởi `log2_min_transform_block_size_minus2` và `log2_diff_max_min_transform_block_size` và gửi được mã hóa các giá trị dưới dạng dòng bit.

Bộ mã hóa có thể thông tin mã hóa về ma trận lượng tử hóa ở bước S420. Bộ mã hóa thông tin mã hóa về ma trận lượng tử hóa, bao gồm một hoặc nhiều thông tin trong số (1) thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa, (2) thông tin về việc ma trận lượng tử hóa có mặt hay không, (3) thông tin về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, (4) phương pháp mã hóa dự đoán và loại của ma trận lượng tử hóa, (5) ma trận lượng tử hóa tham chiếu ID (identifier - mã nhận dạng), và (6) giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa nằm trong ma trận lượng tử hóa.

Ở đây, bộ mã hóa có thể thông tin mã hóa về ma trận lượng tử hóa dựa trên thông tin về kích thước của đơn vị biến đổi.

Ở đây, phương pháp thông tin mã hóa về ma trận lượng tử hóa được mô tả chi tiết có dựa vào các hình vẽ và các bảng sau.

Thông tin về ma trận lượng tử hóa chỉ báo phương pháp được xác định hoặc được sử dụng bởi bộ mã hóa. Ví dụ, bộ mã hóa có thể xác định xem có hay không sử dụng ma trận lượng tử hóa và có thể thông tin mã hóa về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa thành bộ tham số. Do đó, thông tin được mã hóa về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa chỉ báo việc liệu ma trận lượng tử hóa được xác định bởi bộ mã hóa đã được sử dụng hay chưa.

Bảng 2 thể hiện ví dụ trong đó, thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa được mã hóa vào trong thông số chuỗi.

Bảng 2

seq_parameter set rbsp() {	Bộ mô tả
...	
scaling_list_enabled_flag	u(1)
...	
}	

Như trong cú pháp của bảng 2, bộ mã hóa có thể mã hóa `scaling_list_enabled_flag`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa, thành bộ tham số chuỗi và gửi thông tin được mã hóa tới bộ giải mã. Khi giá trị của `scaling_list_enabled_flag` là 1, thì nó có thể chỉ báo rằng ma trận lượng tử hóa được sử dụng trong việc giải lượng tử hóa/định cỡ của hệ số biến đổi cho tất cả các chuỗi. Khi giá trị của `scaling_list_enabled_flag` là 0, thì nó có thể chỉ báo rằng ma trận lượng tử hóa không được sử dụng trong việc giải lượng tử hóa/định cỡ của hệ số biến đổi. Ở đây, cú pháp có thể là thành phần cú pháp.

Sau khi xác định xem ma trận lượng tử hóa có mặt hay không, bộ mã hóa có thể thông tin mã hóa về việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Bảng 3 thể hiện ví dụ trong đó, thông tin về việc ma trận lượng tử hóa có mặt hay không được mã hóa thành bộ tham số.

Bảng 3

aps_rbsp() {	Descriptor
...	
aps_scaling_list_data_present_flag	u(1)
...	
if(aps_scaling_list_data_present_flag)	
scaling_list_param()	
...	
}	

Như trong ví dụ cú pháp của bảng 3, bộ mã hóa có thể mã hóa `aps_scaling_list_data_present_flag`, nghĩa là, thông tin về việc ma trận lượng tử hóa có mặt hay không, thành bộ tham số. Trong bảng 3, ví dụ trong đó, thông tin về việc ma trận lượng tử hóa có mặt hay không được mã hóa thành bộ tham số tương thích đã được minh họa,

nhưng sáng chế không bị giới hạn ở đó. Ví dụ, bộ mã hóa có thể thông tin mã hóa về việc ma trận lượng tử hóa có mặt hay không thành bộ tham số khác.

Theo bảng 3, khi giá trị của `aps_scaling_list_data_present_flag` là 1, điều này chỉ báo rằng ma trận lượng tử hóa có mặt trong bộ tham số tương thích. Khi giá trị của `aps_scaling_list_data_present_flag` là 0, điều này chỉ báo rằng ma trận lượng tử hóa không có mặt trong bộ tham số tương thích. Nếu giá trị của `scaling_list_enabled_flag` là 1 và giá trị của `aps_scaling_list_data_present_flag` là 0, nó có thể đồng nghĩa với việc ma trận mặc định được sử dụng khi giải lượng tử hóa được thực hiện. Hơn nữa, thông tin về việc ma trận lượng tử hóa có mặt hay không có thể được thể hiện trong các bộ tham số khác. Ví dụ, nếu `sps_scaling_list_data_present_flag` chỉ báo việc ma trận lượng tử hóa có mặt hay không trong chuỗi và `pps_scaling_list_data_present_flag` chỉ báo việc ma trận lượng tử hóa có mặt hay không trong hình ảnh được sử dụng, khi giá trị của `sps_scaling_list_data_present_flag` là 1 và giá trị của `pps_scaling_list_data_present_flag` là 0, ma trận lượng tử hóa tương ứng với chuỗi có thể được sử dụng khi việc lượng tử hóa/giải lượng tử hóa được thực hiện. Nghĩa là, nếu ma trận lượng tử hóa được truyền qua nhiều bộ tham số và ma trận lượng tử hóa không có mặt trong một số bộ tham số, ma trận lượng tử hóa có mặt hoặc hiện diện trong bộ tham số chủ động có thể được sử dụng khi việc lượng tử hóa/giải lượng tử hóa được thực hiện. Các nội dung nêu trên cũng có thể được áp dụng cho các nội dung trong đó, thông tin về việc ma trận lượng tử hóa có mặt hay không được mã hóa/được giải mã, từ trong các phương án để được mô tả sau.

Sau khi xác định xem có hay không mã hóa ma trận lượng tử hóa và có hay không sử dụng ma trận mặc định, bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa thành bộ tham số.

Bảng 4 thể hiện ví dụ trong đó, thông tin về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa được mã hóa thành bộ tham số.

Bảng 4

scaling_list param() {	Bộ mô tả
use_default_scaling_list_flag	u(1)
if(!use_default_scaling_list_flag)	
for(SizeID = 0; SizeID < 4; SizeID++)	
for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {	
pred_mode_flag	u(1)
if(!pred_mode_flag)	
pred_matrix_id_delta	ue(v)
else	
scaling_list(QuantMatrix[SizeID][MatrixID],	
(1 << (4 + (SizeID << 1))))	
}	
}	
}	

Như trong ví dụ trên bảng 4, bộ mã hóa có thể mã hóa use_default_scaling_list_flag, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, thành bộ tham số tương thích. Khi giá trị của use_default_scaling_list_flag là 1, ma trận lượng tử hóa không được mã hóa và do đó các giá trị hệ số của tất cả các ma trận lượng tử hóa được xác định là giống với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của use_default_scaling_list_flag là 0, ma trận lượng tử hóa được mã hóa và ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã không được sử dụng.

Bảng 4 minh họa ví dụ trong đó, use_default_scaling_list_flag được mã hóa thành bộ tham số tương thích. Nó chỉ là ví dụ để tạo thuận tiện cho việc mô tả. Theo một số phương án, use_default_scaling_list_flag có thể được mã hóa thành bộ tham số khác.

Thông tin về ma trận lượng tử hóa có thể được xác định bằng cách lấy kích thước của đơn vị biến đổi hoặc kích thước của khối biến đổi, chế độ mã hóa, và thành phần màu để xem xét. Hơn nữa, thông tin về ma trận lượng tử hóa có thể chỉ báo việc liệu thông tin tương ứng là thành phần độ sáng (Y, độ sáng) hoặc thành phần sắc độ (Cb, Cr, sắc độ).

Ví dụ, bộ mã hóa có thể xác định ít nhất một trong số mã hóa ma trận lượng tử hóa, có hay không sử dụng ma trận mặc định, và phương pháp mã hóa dự đoán bằng cách sử dụng SizeID, nghĩa là, thông tin tương ứng với kích thước của ma trận lượng tử hóa, và MatrixID, nghĩa là, thông tin tương ứng với loại ma trận lượng tử hóa. Ở đây, SizeID có thể được giải thích như là thông tin về ma trận lượng tử hóa tương ứng với kích thước của

đơn vị biến đổi hoặc thông tin về ma trận lượng tử hóa tương ứng với kích thước của khối biến đổi. Hơn nữa, SizeID được sử dụng trong bản mô tả này trùng với SizeID và SizeID, và MatrixID trùng với MatrixID và MatrixID.

Ở đây, bộ mã hóa có thể sử dụng bảng được lưu ở đây và/hoặc bảng được lưu trong bộ giải mã.

Bảng 5 thể hiện ví dụ trên bảng được sử dụng để xác định kích thước của khối biến đổi hoặc kích thước của ma trận lượng tử hóa tương ứng với khối biến đổi.

Bảng 5

Giá trị SizeID	Kích thước của đơn vị biến đổi (kích thước của ma trận lượng tử hóa)
0	4x4
1	8x8
2	16x16
3	32x32

Theo ví dụ trên bảng 5, giá trị SizeID xác định kích thước của đơn vị biến đổi, kích thước của khối biến đổi, hoặc kích thước của ma trận lượng tử hóa.

Bảng 6 thể hiện ví dụ trên bảng đối với chế độ mã hóa của khối trong đó, ma trận lượng tử hóa được sử dụng và loại ma trận lượng tử hóa được ánh xạ tới thành phần màu.

Bảng 6

Ý nghĩa của MatrixID (SizeID<3)		
Giá trị MatrixID	Loại chế độ mã hóa	Thành phần
0	Mã hóa nội khung	Y
1	Mã hóa nội khung	Cb
2	Mã hóa nội khung	Cr
3	Mã hóa liên khung	Y
4	Mã hóa liên khung	Cb
5	Mã hóa liên khung	Cr
Ý nghĩa của MatrixID (SizeID==3)		
Giá trị MatrixID	Loại chế độ mã hóa	Thành phần
0	Mã hóa nội khung	Y
1	Mã hóa liên khung	Y

Theo ví dụ trên bảng 6, giá trị MatrixID có thể chỉ báo chế độ mã hóa được sử dụng trong ma trận lượng tử hóa và loại ma trận lượng tử hóa xác định thành phần màu. Ở đây,

chế độ mã hóa có thể là chế độ dự đoán.

Các bảng 7 và 8 là các ví dụ của các bảng ma trận lượng tử hóa mặc định được sử dụng để xác định ma trận lượng tử hóa mặc định dựa trên SizeID và MatrixID được xác định trong các bảng 5 và 6. Ở đây, mỗi giá trị trong các giá trị nằm trong các bảng nghĩa là giá trị được định rõ bởi $\text{ScalingList}[\text{SizeID}][\text{MatrixID}][i]$.

Bảng 7

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\text{ScalingList}[0][0..2][i]$	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
$\text{ScalingList}[0][3..5][i]$	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16

Bảng 8

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\text{ScalingList}[1..2][0..2][i]$ $\text{ScalingList}[3][0][i]$	16	16	16	16	16	16	16	16	16	16	17	16	17	16	17	18
$\text{ScalingList}[1..2][3..5][i]$ $\text{ScalingList}[3][1][i]$	16	16	16	16	16	16	16	16	16	16	17	17	17	17	17	18
i – 16	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\text{ScalingList}[1..2][0..2][i]$ $\text{ScalingList}[3][0][i]$	17	18	18	17	18	21	19	20	21	20	19	21	24	22	22	24
$\text{ScalingList}[1..2][3..5][i]$ $\text{ScalingList}[3][1][i]$	18	18	18	18	18	20	20	20	20	20	20	20	24	24	24	24
i – 32	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\text{ScalingList}[1..2][0..2][i]$ $\text{ScalingList}[3][0][i]$	24	22	22	24	25	25	27	30	27	25	25	29	31	35	35	31
$\text{ScalingList}[1..2][3..5][i]$ $\text{ScalingList}[3][1][i]$	24	24	24	24	25	25	25	25	25	25	25	28	28	28	28	28
i – 48	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\text{ScalingList}[1..2][0..2][i]$ $\text{ScalingList}[3][0][i]$	29	36	41	44	41	36	47	54	54	47	65	70	65	88	88	115
$\text{ScalingList}[1..2][3..5][i]$ $\text{ScalingList}[3][1][i]$	28	33	33	33	33	33	41	41	41	41	54	54	54	71	71	91

Bảng 7 đề cập tới ma trận lượng tử hóa mặc định trong đó, giá trị SizeID là 0 (tức là, khối 4x4), và bảng 8 liên quan tới ma trận lượng tử hóa mặc định trong đó, các giá trị SizeID là 1 (tức là, khối 8x8), 2 (tức là, khối 16x16), và 3 (tức là, khối 32x32). Trong các bảng 7 và 8, các giá trị SizeID và MatrixID là các giá trị được thể hiện trong các bảng 5 và 6.

Trong các bảng 7 và 8, ‘i’ xác định vị trí của mỗi hệ số nằm trong ma trận lượng tử hóa. Trong trường hợp của ma trận lượng tử hóa, ví dụ, trong trường hợp của ma trận lượng

tử hóa cho khối 16×16 hoặc khối 32×32 , các giá trị ma trận lượng tử hóa cho tất cả khối 16×16 và khối 32×32 là không được định rõ, nhưng hệ số ma trận lượng tử hóa chỉ cho khối 8×8 có thể là các hệ số ma trận lượng tử hóa được định rõ và không được định rõ có thể được suy ra dựa trên giá trị khối 8×8 và được sử dụng. Bảng 8 thể hiện ví dụ trong đó, ma trận lượng tử hóa mặc định được định rõ cho mọi khối 8×8 . Hệ số ma trận lượng tử hóa cho khối 16×16 hoặc khối 32×32 có thể được nội suy và được suy ra từ các ma trận lượng tử hóa được lưu cho mọi khối 8×8 hoặc có thể được suy ra theo cách cụ thể. Nếu ma trận lượng tử hóa có kích thước 16×16 hoặc kích thước 32×32 được suy ra từ ma trận lượng tử hóa có kích thước 8×8 thông qua việc nội suy, không phải giá trị được nội suy không được sử dụng, nhưng giá trị bổ sung có thể được sử dụng làm hệ số ma trận lượng tử hóa tại vị trí DC.

Trong khi đó, nếu ma trận lượng tử hóa không được sử dụng bằng cách lấy kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi khả dụng để xem xét, thì ma trận lượng tử hóa cho đơn vị biến đổi có tất cả các kích thước phải được mã hóa. Trong trường hợp này, hiệu quả mã hóa có thể bị giảm sút, và mức độ phức tạp của việc tính toán có thể được tăng.

Để giải quyết các vấn đề này, bộ mã hóa có thể thông tin mã hóa về ma trận lượng tử hóa bằng cách lấy kích thước của đơn vị biến đổi để xem xét. Ví dụ, bộ mã hóa có thể giới hạn SizeID dựa trên kích thước tối thiểu và kích thước tối đa, từ các mẫu thông tin về kích thước của đơn vị biến đổi.

Bộ mã hóa có thể thực hiện một hoặc nhiều trong số mã hóa ma trận lượng tử hóa, thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin mã hóa về loại phương pháp mã hóa dự đoán bằng cách sử dụng SizeID bị giới hạn.

Bảng 9 thể hiện ví dụ của cấu trúc cú pháp được sử dụng khi mã hóa ma trận lượng tử hóa được thực hiện bằng cách giới hạn SizeID.

Bảng 9

scaling_list_param() {	Bộ mô tả
use_default_scaling_list_flag	u(1)
if(!use_default_scaling_list_flag)	
for(SizeID = Log2MinTrafoSize-2, SizeID < Log2MaxTrafoSize-1; SizeID++)	
for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {	
pred_mode_flag	u(1)
if(!pred_mode_flag)	
pred_matrix_id_delta	ue(v)
else	
scaling_list(QuantMatrix[SizeID][MatrixID],	
(1 << (4 + (SizeID << 1))))	
}	
}	
}	

Như trong ví dụ trên bảng 9, bộ mã hóa có thể giới hạn SizeID dựa trên kích thước tối thiểu và kích thước tối đa, từ các mẫu thông tin về kích thước của đơn vị biến đổi, và thực hiện một hoặc nhiều trong số mã hóa ma trận lượng tử hóa, thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin mã hóa về loại phương pháp mã hóa dự đoán trên các kích thước đơn vị biến đổi cụ thể.

Ví dụ, khi giá trị của Log2MinTrafoSize là 3 và giá trị của Log2MaxTrafoSize là 4, thì bộ mã hóa có thể thực hiện một hoặc nhiều trong số mã hóa ma trận lượng tử hóa tương ứng với 8x8 đơn vị biến đổi tới 16x16 đơn vị biến đổi, thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin mã hóa về loại phương pháp mã hóa dự đoán.

Trong khi đó, trong ví dụ trên bảng 9, use_default_scaling_list_flag có thể không được mã hóa.

Hơn nữa, bộ mã hóa có thể giới hạn SizeID dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu, từ các mẫu thông tin về kích thước của đơn vị biến đổi, và thực hiện một hoặc nhiều trong số mã hóa ma trận lượng tử hóa, thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin mã hóa về loại phương pháp mã hóa dự đoán.

Bảng 10 thể hiện sơ lược một ví dụ của cấu trúc cú pháp được sử dụng để mã hóa các mẫu thông tin về ma trận lượng tử hóa bằng cách giới hạn SizeID dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu, từ các mẫu thông tin về kích thước của đơn

vị biến đổi.

Bảng 10

scaling_list_param() {	Bộ mô tả
use default scaling list flag	u(1)
if(!use default scaling list flag)	
for(SizeID = 0; SizeID < Log2MaxTrafoSize-Log2MinTrafoSize+1; SizeID++)	
for(MatrixID = 0; MatrixID < ((SizeID+Log2MinTrafoSize-2) == 3) ? 2:6; MatrixID++) {	
pred mode flag	u(1)
if(!pred mode flag)	
pred matrix id delta	ue(v)
else	
scaling_list(QuantMatrix[SizeID+Log2MinTrafoSize-2 MatrixID], (1 << (4 + ((SizeID+Log2MinTrafoSize-2) << 1))))	
}	
}	

Theo ví dụ cú pháp của bảng 10, SizeID bị giới hạn dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu, từ các mẫu thông tin về kích thước của đơn vị biến đổi. Bộ mã hóa có thể giới hạn SizeID dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu, từ các mẫu thông tin về kích thước của đơn vị biến đổi, và có thể mã hóa một hoặc nhiều thông tin trong số các thông tin về ma trận lượng tử hóa, thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin về loại phương pháp mã hóa dự đoán chỉ trên giới hạn của kích thước của đơn vị biến đổi cụ thể dựa trên SizeID bị giới hạn.

Theo ví dụ trên bảng 10, khi giá trị của Log2MinTrafoSize là 3 và giá trị của Log2MaxTrafoSize là 4, thì một hoặc nhiều trong số mã hóa ma trận lượng tử hóa tương ứng với 8x8 đơn vị biến đổi tới 16x16 đơn vị biến đổi, thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin mã hóa về loại phương pháp mã hóa dự đoán có thể được thực hiện.

Ở đây, giá trị chênh lệch giữa Log2MaxTrafoSize và Log2MinTrafoSize là giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu của đơn vị biến đổi và có thể trùng với log2_diff_max_min_transform_block_size. Hơn nữa, Log2MinTrafoSize-2 có thể trùng với log2_min_transform_block_size_minus2.

Theo ví dụ trên bảng 10, `use_default_scaling_list_flag` có thể không được mã hóa.

Trong khi đó, nếu ma trận mặc định và ma trận không phải là ma trận mặc định không được trộn và được sử dụng theo kích thước của mỗi khối biến đổi (hoặc bộ phận) hoặc loại ma trận lượng tử hóa nằm trong chuỗi, hình ảnh hoặc phiên, thì mức độ tự do khi bộ mã hóa chọn ma trận lượng tử hóa được làm giảm. Ví dụ, để sử dụng ma trận mặc định cho khối biến đổi có kích thước cụ thể nằm trong phiên và ma trận không phải là ma trận mặc định cho khối biến đổi khác có kích thước cụ thể nằm trong phiên, thì các ma trận mặc định phải được mã hóa và được truyền tới bộ giải mã. Kết quả là, hiệu quả mã hóa bị giảm sút.

Để trộn và sử dụng ma trận mặc định và ma trận không phải là ma trận mặc định theo mỗi kích thước biến đổi hoặc loại ma trận lượng tử hóa nằm trong chuỗi, hình ảnh hoặc phiên, thì bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa thành bộ tham số dựa trên `SizeID`.

Bảng 11 thể hiện ví dụ của cấu trúc cú pháp có thể được sử dụng khi thông tin về ma trận lượng tử hóa được mã hóa dựa trên `SizeID`.

Bảng 11

<code>scaling_list_param() {</code>	Bộ mô tả
<code>use_default_scaling_list_flag</code>	<code>u(1)</code>
<code>if(!use_default_scaling_list_flag)</code>	
<code>for(SizeID = 0; SizeID < 4; SizeID++) {</code>	
<code>sid_use_default_scaling_list_flag[SizeID][MatrixID]</code>	<code>u(1)</code>
<code>if(!sid_use_default_scaling_list_flag[SizeID][MatrixID]) {</code>	
<code>for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {</code>	
<code>pred_mode_flag</code>	<code>u(1)</code>
<code>if(!pred_mode_flag)</code>	
<code>pred_matrix_id_delta</code>	<code>uc(v)</code>
<code>else</code>	
<code>scaling_list(QuantMatrix[SizeID][MatrixID],</code>	
<code>(1 << (4 + (SizeID << 1)))</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	

Như trong ví dụ trên bảng 11, bộ mã hóa có thể mã hóa

sid_use_default_scaling_list_flag[SizeID][MatrixID], nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, thành bộ tham số tương thích dựa trên SizeID. MatrixID chỉ báo loại ma trận lượng tử hóa cụ thể như trong bảng 6.

Khi giá trị của sid_use_default_scaling_list_flag[SizeID][MatrixID] là 1, thì ma trận lượng tử hóa tương ứng với SizeID không được mã hóa và các giá trị hệ số của ma trận lượng tử hóa tương ứng với SizeID có thể được xác định là giống với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của sid_use_default_scaling_list_flag[SizeID][MatrixID] là 0, thì ma trận lượng tử hóa tương ứng với SizeID được mã hóa, và ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã có thể không được sử dụng trong ma trận lượng tử hóa theo SizeID.

Theo ví dụ trên bảng 11, use_default_scaling_list_flag có thể không được mã hóa.

Hơn nữa, bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa thành bộ tham số dựa trên MatrixID.

Bảng 12 thể hiện sơ lược một ví dụ của cấu trúc cú pháp có thể được sử dụng khi thông tin về ma trận lượng tử hóa được mã hóa sử dụng MatrixID.

Bảng 12

scaling_list_param() {	Bộ mô tả
use_default_scaling_list_flag	u(1)
if(!use_default_scaling_list_flag)	
for(SizeID = 0; SizeID < 4; SizeID++)	
for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {	
mid_use_default_scaling_list_flag [SizeID][MatrixID]	u(1)
if(!mid_use_default_scaling_list_flag[SizeID][MatrixID]) {	
pred_mode_flag	u(1)
if(!pred_mode_flag)	
pred_matrix_id_delta	ue(v)
else	
scaling_list(QuantMatrix[SizeID][MatrixID], (1 << (4 + (SizeID << 1))))	
}	
}	
}	

Như trong ví dụ trên bảng 12, bộ mã hóa có thể mã hóa `mid_use_default_scaling_list_flag[SizeID][MatrixID]`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, thành bộ tham số tương thích dựa trên `MatrixID`.

Khi giá trị của `mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1, thì ma trận lượng tử hóa tương ứng với `MatrixID` không được mã hóa và các giá trị hệ số của ma trận lượng tử hóa tương ứng với `MatrixID` có thể được xác định là giống với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của `mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0, thì ma trận lượng tử hóa tương ứng với `MatrixID` được mã hóa và ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã không được sử dụng như là ma trận lượng tử hóa tương ứng với `MatrixID`.

Theo ví dụ trên bảng 12, `use_default_scaling_list_flag` có thể không được mã hóa.

Hơn nữa, bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa thành bộ tham số dựa trên `SizeID` và `MatrixID`.

Bảng 13 thể hiện sơ lược một ví dụ của cấu trúc cú pháp có thể được sử dụng khi thông tin về ma trận lượng tử hóa được mã hóa sử dụng `SizeID` và `MatrixID`.

Bảng 13

<code>scaling_list_param()</code> {	Bộ mô tả
<code>use_default_scaling_list_flag</code>	<code>u(1)</code>
<code>if(!use_default_scaling_list_flag)</code>	
<code>for(SizeID = 0; SizeID < 4; SizeID++)</code>	
<code>for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {</code>	
<code>sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]</code>	<code>u(1)</code>
<code>if(!sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]) {</code>	
<code>pred_mode_flag</code>	<code>u(1)</code>
<code>if(!pred_mode_flag)</code>	
<code>pred_matrix_id_delta</code>	<code>ue(v)</code>
<code>else</code>	
<code>scaling_list(QuantMatrix[SizeID][MatrixID],</code> <code>(1 << (4 + (SizeID << 1))))</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	

Như trong ví dụ trên bảng 13, bộ mã hóa có thể mã hóa `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, thành bộ tham số tương thích dựa trên `SizeID` và `MatrixID`.

Khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1, thì ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` không được mã hóa và các giá trị hệ số của ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` có thể được xác định là giống với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã.

Khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0, thì ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` được mã hóa và ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã không được sử dụng làm ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID`.

Theo ví dụ trên bảng 13, `use_default_scaling_list_flag` có thể không được mã hóa.

Trong khi đó, không giống như trong ví dụ trên bảng 13, bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa vào trong thông số dựa trên `SizeID` và `MatrixID`. Bộ mã hóa có thể giới hạn `SizeID` dựa trên kích thước tối thiểu và kích thước tối đa, từ các mẫu thông tin về kích thước của đơn vị biến đổi, và thực hiện một hoặc nhiều trong số mã hóa ma trận lượng tử hóa, thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin mã hóa về loại phương pháp mã hóa dự đoán dựa trên `SizeID` bị giới hạn.

Bảng 14 thể hiện sơ lược một ví dụ của cấu trúc cú pháp khác có thể được sử dụng khi thông tin về ma trận lượng tử hóa được mã hóa bằng cách sử dụng `SizeID` và `MatrixID`.

Bảng 14

scaling_list_param() {	Bộ mô tả
use_default_scaling_list_flag	u(1)
if(!use_default_scaling_list_flag)	
for(SizeID = Log2MinTrafoSize-2; SizeID < Log2MaxTrafoSize-1; SizeID++)	
for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {	
sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]	u(1)
if(!sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]) {	
...	
}	
}	
}	

Theo ví dụ trên bảng 14, use_default_scaling_list_flag có thể không được mã hóa.

Hơn nữa, bộ mã hóa có thể giới hạn SizeID dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu, từ các mẫu thông tin về đơn vị biến đổi, và thực hiện một hoặc nhiều trong số mã hóa ma trận lượng tử hóa, thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin mã hóa về loại phương pháp mã hóa dự đoán.

Bảng 15 thể hiện sơ lược một ví dụ của cấu trúc cú pháp khác có thể được sử dụng khi thông tin về ma trận lượng tử hóa được mã hóa bằng cách sử dụng SizeID và MatrixID.

Bảng 15

scaling_list_param() {	Bộ mô tả
use_default_scaling_list_flag	u(1)
if(!use_default_scaling_list_flag)	
for(SizeID = 0; SizeID < Log2MaxTrafoSize-Log2MinTrafoSize+1; SizeID++)	
for(MatrixID = 0; MatrixID < ((SizeID+Log2MinTrafoSize-2) == 3) ? 2:6; MatrixID++) {	
sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]	u(1)
if(!sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]) {	
...	
}	
}	
}	

Theo ví dụ trên bảng 15, giá trị chênh lệch giữa Log2MaxTrafoSize, xác định kích thước tối đa của đơn vị biến đổi, và Log2MinTrafoSize, xác định kích thước tối thiểu của đơn vị biến đổi, là giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu của đơn vị

biến đổi và trùng với `log2_diff_max_min_transform_block_size`. `Log2MinTrafoSize-2` trùng với `log2_min_transform_block_size_minus2`.

Theo ví dụ trên bảng 15, bộ mã hóa có thể thông tin mã hóa về ma trận lượng tử hóa đã được mã hóa hay không và liệu ma trận mặc định đã được sử dụng hay chưa vào trong thông số dựa trên `SizeID` và `MatrixID` bị giới hạn bởi `Log2MaxTrafoSize-Log2MinTrafoSize+1`.

Theo ví dụ trên bảng 15, `use_default_scaling_list_flag` có thể không được mã hóa.

Trong khi đó, thông tin về việc có hay không thông tin về ma trận lượng tử hóa có mặt trong bộ tham số để được mã hóa hoặc có hay không thông tin về ma trận lượng tử hóa sẽ được cập nhật có thể được mã hóa thành bộ tham số và được sử dụng trong việc mã hóa/giải mã.

Bảng 16 thể hiện sơ lược một ví dụ của cấu trúc cú pháp có thể được sử dụng khi thông tin về ma trận lượng tử hóa có mặt hay không được mã hóa.

Bảng 16

<code>scaling_list_param() {</code>	Bộ mô tả
<code>use_default_scaling_list_flag</code>	<code>u(1)</code>
<code>if(!use_default_scaling_list_flag)</code>	
<code>for(SizeID = 0; SizeID < 4; SizeID++)</code>	
<code>for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {</code>	
<code>scaling_list_update_flag[SizeID][MatrixID]</code>	<code>u(1)</code>
<code>if(scaling_list_update_flag[SizeID][MatrixID]) {</code>	
<code>sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]</code>	<code>u(1)</code>
<code>if(!sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]) {</code>	
<code>...</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	

Theo ví dụ trên bảng 16, bộ mã hóa có thể xác định có hay không thông tin về ma trận lượng tử hóa có mặt trong thông số được mã hóa bằng cách sử dụng thành phần cú pháp `scaling_list_update_flag[SizeID][MatrixID]`.

Ví dụ, khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 1, thì nó có thể chỉ báo rằng thông tin về ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` có mặt.

Khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 1, thì nó có thể chỉ báo rằng thông tin về ma trận lượng tử hóa, tương ứng với `SizeID` và `MatrixID` và được mã hóa từ trước, được cập nhật vào trong thông tin về ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` nằm trong bộ tham số để được mã hóa. Việc cập nhật có thể đồng nghĩa với việc thông tin về ma trận lượng tử hóa đã được mã hóa từ trước được thay thế bằng thông tin về ma trận lượng tử hóa nằm trong bộ tham số để được mã hóa.

Khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì nó có thể chỉ báo rằng thông tin về ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` không có mặt. Khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì nó có thể chỉ báo rằng thông tin về mã hóa ma trận không được cập nhật.

Khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, bộ giải mã không biết là nó phải thực hiện giải lượng tử hóa trên hệ số biến đổi được lượng tử hóa tương ứng với `SizeID` và `MatrixID` bằng cách sử dụng thông tin về ma trận lượng tử hóa nào do thông tin về ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` không có mặt trong bộ tham số và thông tin về ma trận lượng tử hóa không được mã hóa. Do đó, khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` có thể đồng nghĩa với việc ma trận mặc định được sử dụng hoặc ma trận lượng tử hóa không được sử dụng do thông tin về ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` không có mặt.

Theo ví dụ trên bảng 16, khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 1 do `scaling_list_update_flag[SizeID][MatrixID]` được mã hóa dựa trên `SizeID` và `MatrixID`, thì bộ mã hóa có thể mã hóa một hoặc nhiều thông tin trong thông tin về ma trận lượng tử hóa, thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin về loại phương pháp mã hóa dự đoán.

Khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì bộ mã hóa không mã hóa một hoặc nhiều thông tin trong số thông tin về ma trận lượng tử hóa, thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin về loại phương pháp mã hóa dự đoán không được mã hóa. Nghĩa là, bộ mã hóa có thể không mã hóa ma trận lượng tử hóa không cần thiết bằng cách sử dụng

`scaling_list_update_flag[SizeID][MatrixID]`.

Do ma trận mặc định và ma trận không phải là ma trận mặc định không được chọn và được sử dụng cho mỗi kích thước biến đổi hoặc loại mỗi ma trận lượng tử hóa nằm trong chuỗi, nên hình ảnh hoặc phiên bản cách sử dụng chỉ `scaling_list_update_flag[SizeID][MatrixID]`, có nhược điểm là mức độ tự do khi bộ mã hóa chọn ma trận lượng tử hóa là thấp.

Do đó, bộ mã hóa có thể mã hóa `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, thành bộ tham số dựa trên SizeID và MatrixID.

Ví dụ, trong ví dụ trên bảng 16, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1, thì ma trận lượng tử hóa tương ứng với SizeID và MatrixID không được mã hóa và các giá trị hệ số của ma trận lượng tử hóa tương ứng với SizeID và MatrixID có thể được xác định là giống với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0, thì ma trận lượng tử hóa tương ứng với SizeID và MatrixID được mã hóa và ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã không được sử dụng làm ma trận lượng tử hóa tương ứng với SizeID và MatrixID.

Theo ví dụ trên bảng 16, `use_default_scaling_list_flag` có thể không được mã hóa.

Hơn nữa, bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận lượng tử hóa đã được mã hóa hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa thành bộ tham số dựa trên SizeID và MatrixID. Hơn nữa, bộ mã hóa có thể sử dụng `scaling_list_update_flag[SizeID][MatrixID]`, nghĩa là, thông tin chỉ báo việc có hay không thông tin về ma trận lượng tử hóa sẽ được cập nhật vào trong thông tin về ma trận lượng tử hóa, tương ứng với SizeID và MatrixID, trong tham số để được mã hóa.

Bảng 17 thể hiện sơ lược một ví dụ của cấu trúc cú pháp có thể được sử dụng khi thông tin về ma trận lượng tử hóa được mã hóa dựa trên SizeID và MatrixID như được

mô tả ở trên.

Bảng 17

scaling_list_param() {	Bộ mô tả
use_default_scaling_list_flag	u(1)
if(!use_default_scaling_list_flag)	
for(SizeID = 0; SizeID < 4; SizeID++)	
for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {	
sid_mid_use_default_scaling_list_flag [SizeID][MatrixID]	u(1)
if(!sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]) {	
scaling_list_update_flag [SizeID][MatrixID]	u(1)
if(scaling_list_update_flag[SizeID][MatrixID]) {	
...	
}	
}	
}	
}	
}	

Theo ví dụ trên bảng 17, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0 và giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` được mã hóa. Ma trận lượng tử hóa đã được mã hóa từ trước, tương ứng với `SizeID` và `MatrixID`, không được cập nhật vào trong ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` trong tham số để được mã hóa, và ma trận lượng tử hóa đã được mã hóa từ trước, tương ứng với được mã hóa `SizeID` và `MatrixID`, được sử dụng mà không thay đổi.

Hơn nữa, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0 và giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 1, thì ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` được mã hóa và ma trận lượng tử hóa đã được mã hóa từ trước, tương ứng với `SizeID` và `MatrixID`, được cập nhật vào trong ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` trong tham số để được mã hóa.

Hơn nữa, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1 và giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` không được mã hóa và các giá trị hệ số của ma trận lượng tử hóa được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ma trận lượng tử hóa đã được mã hóa từ trước, tương

ứng với SizeID và MatrixID, không được cập nhật vào trong ma trận lượng tử hóa tương ứng với SizeID và MatrixID trong tham số để được mã hóa, và bộ mã hóa và bộ giải mã sử dụng ma trận lượng tử hóa đã được mã hóa từ trước, tương ứng với SizeID và MatrixID.

Hơn nữa, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1 và giá trị của `scaling_list_update_flag[SizeID][MatrixID]` cũng là 1, thì ma trận lượng tử hóa tương ứng với SizeID và MatrixID không được mã hóa và các giá trị hệ số của ma trận lượng tử hóa được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ma trận lượng tử hóa đã được mã hóa từ trước, tương ứng với SizeID và MatrixID, được cập nhật vào trong ma trận lượng tử hóa tương ứng với SizeID và MatrixID trong tham số để được mã hóa.

Theo ví dụ trên bảng 17, `use_default_scaling_list_flag` có thể không được mã hóa.

Hơn nữa, sau khi xác định loại phương pháp dự đoán và mã hóa ma trận lượng tử hóa, bộ mã hóa có thể thông tin mã hóa về phương pháp dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số.

Các bảng 18 và 19 thể hiện một cách sơ lược các ví dụ của các cú pháp có thể được sử dụng khi phương pháp dự đoán và mã hóa ma trận lượng tử hóa được mã hóa thành bộ tham số.

Bảng 18

<code>scaling_list_param() {</code>	Bộ mô tả
<code>use_default_scaling_list_flag</code>	<code>u(1)</code>
<code>if(!use_default_scaling_list_flag)</code>	
<code>for(SizeID = 0; SizeID < 4; SizeID++)</code>	
<code>for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {</code>	
<code>pred_mode_flag</code>	<code>u(1)</code>
<code>if(!pred_mode_flag)</code>	
<code>pred_matrix_id_delta</code>	<code>ue(v)</code>
<code>else</code>	
<code>scaling_list(QuantMatrix[SizeID][MatrixID],</code>	
<code>(1 << (4 + (SizeID << 1)))</code>	
<code>}</code>	
<code>}</code>	

Bảng 19

scaling_list(ScalingList, coefNum) {	Bộ mô tả
nextcoef = 8	
for(i=0; i<coefNum, i++) {	
delta_coef	se(v)
nextcoef = (nextcoef + delta_coef + 256) % 256	
ScalingList[i] = nextcoef	
}	
}	

Như trong ví dụ trên bảng 18, bộ mã hóa có thể mã hóa pred_mode_flag, nghĩa là, thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa, thành bộ tham số tương thích. Scaling_list_pred_mode_flag sẽ mô tả sau trong bản mô tả này có thể được giải thích là có cùng một ý nghĩa với pred_mode_flag.

Ví dụ, trong ví dụ trên bảng 18, khi giá trị của pred_mode_flag là 1, thì bộ mã hóa có thể mã hóa ma trận lượng tử hóa theo phương pháp điều biến mã xung vi phân (Differential Pulse Code Modulation - DPCM) và phương pháp mã hóa Golomb hàm mũ. Khi giá trị của pred_mode_flag là 0, thì bộ mã hóa có thể xác định các giá trị hệ số của ma trận lượng tử hóa được mã hóa sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa được mã hóa từ trước. Ở đây, các giá trị hệ số của các ma trận lượng tử hóa và các giá trị hệ số của các ma trận lượng tử hóa được mã hóa từ trước có thể là giá trị nằm trong các ma trận lượng tử hóa khác.

Nếu giá trị của pred_mode_flag là 0, thì bộ mã hóa có thể thông tin mã hóa về ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số.

Do đó, như trong ví dụ trên bảng 18, bộ mã hóa có thể mã hóa pred_matrix_id_delta, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số tương thích.

scaling_list_pred_matrix_id_delta được mô tả trong bản mô tả này có thể được giải thích là có cùng một ý nghĩa với pred_matrix_id_delta. Ở đây, bộ mã hóa và bộ giải mã có thể xác định giá trị của RefMatrixID, chỉ báo ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa, bằng cách sử dụng pred_matrix_id_delta và Phương trình 1.

Phương trình 1

$$\text{RefMatrixID} = \text{MatrixID} - (1 + \text{pred_matrix_id_delta})$$

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, giá trị hệ số của ma trận lượng tử hóa được mã hóa từ trước có thể là giá trị hệ số nằm trong ma trận lượng tử hóa được mã hóa. Nghĩa là, giá trị hệ số của ma trận lượng tử hóa được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa có thể là các giá trị nằm trong cùng một ma trận lượng tử hóa.

Do đó, như trong ví dụ trên bảng 19, bộ mã hóa có thể mã hóa `delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa, thành bộ tham số tương thích. `delta_coef` sẽ mô tả sau trong bản mô tả này có thể được giải thích là có cùng một ý nghĩa với `scaling_list_delta_coef`.

Bộ mã hóa có thể trộn lẫn và sử dụng ma trận lượng tử hóa mặc định và ma trận lượng tử hóa không mặc định nằm trong chuỗi, hình ảnh hoặc phiên bản cách sử dụng phương pháp dưới đây và có thể ngăn không cho ma trận lượng tử hóa không cần thiết bị truyền.

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp xác định ma trận mặc định sao cho ma trận mặc định trùng với ma trận lượng tử hóa hiện đã được chứa trong bộ mã hóa và được mã hóa từ trước (`pred_mode_flag=0`), thì bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa thành bộ tham số bằng cách sử dụng ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa.

Ví dụ, như trong ví dụ trên bảng 18, bộ mã hóa có thể mã hóa `pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ mã hóa và bộ giải mã có thể xác định `RefMatrixID` chỉ báo của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa hoặc ma trận mặc định bằng cách sử dụng `pred_matrix_id_delta` và Phương trình 2 và xác định xem có hay không sử dụng ma trận mặc định.

Phương trình 2

$$\text{RefMatrixID} = \text{MatrixID} - \text{pred_matrix_id_delta}$$

Theo phương trình 2, nếu giá trị của RefMatrixID trùng với giá trị MatrixID, thì các giá trị hệ số của ma trận lượng tử hóa tương ứng với SizeID và RefMatrixID được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ở đây, ma trận mặc định nghĩa là ma trận mặc định được định rõ bởi SizeID và RefMatrixID.

Hơn nữa, khi giá trị của pred_matrix_id_delta là 0, thì giá trị của RefMatrixID trở nên trùng với giá trị MatrixID. Nếu giá trị của RefMatrixID không trùng với giá trị MatrixID, thì bộ mã hóa có thể xác định ma trận lượng tử hóa tương ứng với RefMatrixID làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa. Trong trường hợp này, bộ mã hóa có thể xác định các giá trị hệ số của ma trận lượng tử hóa được mã hóa sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu.

Nếu kích thước của ma trận lượng tử hóa tương ứng với SizeID bao gồm trong kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi có sẵn bởi bộ giải mã, thì ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa và có hay không sử dụng ma trận mặc định có thể được xác định bằng cách sử dụng phương pháp nêu trên. Nếu kích thước của ma trận lượng tử hóa tương ứng với SizeID không bao gồm trong kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi khả dụng bởi bộ giải mã, thì phương pháp mã hóa ma trận lượng tử hóa tương ứng với SizeID có thể không được xác định là giống như ma trận mặc định. Việc xác định các quy trình có thể được thực hiện khi mã hóa một hoặc nhiều thông tin về ma trận lượng tử hóa, thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và thông tin về loại phương pháp mã hóa dự đoán dựa trên SizeID và giá trị chênh lệch giữa kích thước tối đa của đơn vị biến đổi và kích thước tối thiểu của đơn vị biến đổi.

Sau khi xác định xem có hay không sử dụng ma trận lượng tử hóa, bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa thành bộ tham số.

Bảng 20 thể hiện sơ lược ví dụ trong đó, thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa được mã hóa thành bộ tham số.

Bảng 20

scaling_list_param() {	Bộ mô tả
scaling_list_enable_flag	u(1)
if(scaling_list_enable_flag)	
for(SizeID = 0; SizeID < 4; SizeID++)	
for(MatrixID = 0; MatrixID < (SizeID == 3) ? 2:6; MatrixID++) {	
pred_mode_flag	u(1)
if(!pred_mode_flag)	
pred_matrix_id_delta	ue(v)
else	
scaling_list(QuantMatrix[SizeID][MatrixID],	
(1 << (4 + (SizeID << 1)))	
}	
}	

Như trong ví dụ trên bảng 20, bộ mã hóa có thể mã hóa `scaling_list_enable_flag`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa, thành bộ tham số.

Theo ví dụ trên bảng 20, khi giá trị của `scaling_list_enable_flag` là 1, thì ma trận lượng tử hóa, như ma trận mặc định hoặc ma trận không phải là ma trận mặc định, có thể được sử dụng trong lượng tử hóa/giải lượng tử hóa. Khi giá trị của `scaling_list_enable_flag` là 0, thì ma trận lượng tử hóa không được sử dụng trong lượng tử hóa/giải lượng tử hóa hoặc tất cả các giá trị hệ số có thể sử dụng cùng một ma trận lượng tử hóa. Ở đây, tất cả các giá trị hệ số có thể là 16.

Hơn nữa, nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, giá trị hệ số của ma trận lượng tử hóa được mã hóa từ trước có thể là giá trị hệ số nằm trong ma trận lượng tử hóa được mã hóa. Nghĩa là, giá trị hệ số của ma trận lượng tử hóa được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa có thể là các giá trị nằm trong cùng một ma trận lượng tử hóa.

Bảng 21 thể hiện sơ lược một ví dụ của cấu trúc cú pháp có thể được sử dụng khi thông tin về ma trận lượng tử hóa được mã hóa bằng cách sử dụng giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa.

Bảng 21

<code>scaling_list(ScalingList, coefNum) {</code>	Bộ mô tả
<code>nextcoef = 8</code>	<code>u(1)</code>
<code>for(i=0; i < coefNum, i++) {</code>	
<code> scaling_list_delta_coef</code>	<code>se(v)</code>
<code> nextcoef = (nextcoef + delta_coef + 256) % 256</code>	
<code> if (useDefaultScalingMatrixFlag = (i == 0 && nextcoef == 0))</code>	
<code> break;</code>	
<code> ScalingList[i] = nextcoef</code>	
<code>}</code>	
<code>}</code>	

Theo ví dụ trên bảng 21, `delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa, được mã hóa thành bộ tham số.

Theo ví dụ trên bảng 21, giá trị hệ số của ma trận lượng tử hóa có thể được tính toán sử dụng Phương trình 3.

Phương trình 3

$$\text{Nextcoef} = (\text{nextcoef} + \text{delta_coef} + 256) \% 256$$

Nếu `nextcoef`, nghĩa là, giá trị hệ số của ma trận lượng tử hóa được tính toán sử dụng Phương trình 3, trùng với (1) giá trị cụ thể và là (2) giá trị thứ nhất của ma trận lượng tử hóa, thì các giá trị hệ số của ma trận lượng tử hóa có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã.

Nghĩa là, nếu giá trị của `nextcoef` trùng với (1) giá trị cụ thể và là (2) giá trị thứ nhất của ma trận lượng tử hóa, thì ma trận lượng tử hóa tương ứng có thể được sử dụng làm ma trận mặc định.

Ở đây, giá trị cụ thể có thể là 0. Hơn nữa, ma trận mặc định có thể là ma trận mặc định tương ứng với `SizeID` và `MatrixID`. Do đó, nếu `nextcoef`, nghĩa là, giá trị hệ số của ma

trận lượng tử hóa, trùng với 0 và giá trị hệ số của ma trận lượng tử hóa tương ứng với giá trị thứ nhất của ma trận lượng tử hóa, thì mã hóa của giá trị chênh lệch giữa ma trận lượng tử hóa tương ứng và ma trận lượng tử hóa có thể được dùng.

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước có thể là giá trị hệ số nằm trong ma trận lượng tử hóa được mã hóa. Nghĩa là, giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa có thể là các giá trị nằm trong cùng một ma trận lượng tử hóa.

Bảng 22 thể hiện sơ lược một ví dụ của cấu trúc cú pháp khi chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa được sử dụng.

Bảng 22

scaling_list(scalingList, sizeID , matrixID) {	Bộ mô tả
nextcoef = 8	u(1)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
UseDefaultScalingMatrix = 0	
if(sizeID > 1) {	
scaling_list dc coef minus8 [sizeID - 2][matrixID]	se(v)
if(scaling_list dc coef minus8[sizeID - 2][matrixID] + 8 == 0)	
UseDefaultScalingMatrixFlag = 1	
}	
if(UseDefaultScalingMatrixFlag == 0) {	
for(i=0; i < coefNum, i++) {	
delta coef	se(v)

<code>nextcoef = (nextcoef + delta_coef + 256) % 256</code>	
<code>if(sizeID < 2) {</code>	
<code> useDefaultScalingMatrixFlag = (i == 0 && nextcoef == 0)</code>	
<code> if(useDefaultScalingMatrixFlag)</code>	
<code> Break</code>	
<code> }</code>	
<code> scalingList[i] = nextcoef</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	

Theo ví dụ trên bảng 22, `delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước và giá trị hệ số của ma trận lượng tử hóa được mã hóa, được mã hóa thành bộ tham số. Trong ví dụ trên bảng 22, nếu `nextcoef`, nghĩa là, giá trị hệ số của ma trận lượng tử hóa được tính toán sử dụng Phương trình 3, trùng với một giá trị cụ thể và là giá trị thứ nhất của ma trận lượng tử hóa, thì các giá trị hệ số của ma trận lượng tử hóa có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã.

Nghĩa là, nếu giá trị của `nextcoef` trùng với (1) giá trị cụ thể và là (2) giá trị thứ nhất của ma trận lượng tử hóa, thì ma trận lượng tử hóa có thể được sử dụng làm ma trận mặc định. Ở đây, giá trị cụ thể có thể là 0, và giá trị hệ số của ma trận lượng tử hóa được tính toán sử dụng Phương trình 3 có thể là giá trị hệ số của ma trận lượng tử hóa có ma trận lượng tử hóa có kích thước hoặc kích thước biến đổi là 4x4 và 8x8.

Hơn nữa, giá trị thứ nhất của ma trận lượng tử hóa có thể là giá trị sử dụng `scaling_list_dc_coef_minus8`, và giá trị cụ thể có thể là giá trị tương ứng với `scaling_list_dc_coef_minus8+8`.

Ở đây, `scaling_list_dc_coef_minus8` có thể đồng nghĩa với việc giá trị thứ nhất của ma trận lượng tử hóa có kích thước 16x16 hoặc ma trận lượng tử hóa có kích thước 32x32. Nghĩa là, `scaling_list_dc_coef_minus8` có thể có nghĩa là giá trị hệ số của ma trận lượng tử hóa cho hệ số ma trận DC và có thể có nghĩa là hệ số ma trận DC. Trong bản mô tả này, hệ số ma trận DC có mặt nằm trong ma trận lượng tử hóa được sử dụng khi giải lượng tử hóa được thực hiện, và hệ số ma trận DC có thể là hệ số ma trận lượng tử hóa cho hệ số biến đổi DC nằm trong khối biến đổi.

Ví dụ, khi SizeID là 2, thì `scaling_list_dc_coef_minus8[SizeID-2][MatrixID]` có thể đồng nghĩa với việc giá trị hệ số của ma trận lượng tử hóa cho hệ số ma trận DC trong ma trận lượng tử hóa có kích thước 16x16 hoặc kích thước biến đổi. Khi SizeID là 3, `scaling_list_dc_coef_minus8[SizeID-2][MatrixID]` có thể đồng nghĩa với việc giá trị hệ số của ma trận lượng tử hóa cho hệ số ma trận DC trong ma trận lượng tử hóa có kích thước 32x32 hoặc kích thước biến đổi.

Hơn nữa, ma trận mặc định nêu trên có thể đồng nghĩa với việc ma trận mặc định tương ứng với SizeID và MatrixID. Như được mô tả ở trên, nếu giá trị hệ số của ma trận lượng tử hóa, `nextcoef`, trùng với 0 và là giá trị thứ nhất của ma trận lượng tử hóa, thì mã hóa của giá trị chênh lệch giữa ma trận lượng tử hóa tương ứng và ma trận lượng tử hóa có thể được dừng.

Nếu phương pháp này được sử dụng, dù có hay không mã hóa `scaling_list_dc_coef_minus8`, nghĩa là, giá trị hệ số của ma trận lượng tử hóa cho hệ số ma trận DC, và thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa có thể được thực hiện theo cách khác theo kích thước của ma trận lượng tử hóa hoặc kích thước biến đổi.

Trong khi đó, các phương pháp mã hóa/giải mã ma trận lượng tử hóa để xác định xem có hay không sử dụng ma trận mặc định bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa có nhược điểm ở chỗ mức độ phức tạp trong quy trình mã hóa/giải mã giá trị hệ số của ma trận lượng tử hóa tăng lên.

Phương pháp dưới đây đề xuất phương pháp xác định xem có hay không sử dụng ma trận mặc định bằng cách sử dụng ma trận lượng tử hóa tham chiếu ID trong mã hóa/giải mã hình ảnh. Do đó, nếu phương pháp sau được sử dụng, mức độ phức tạp của việc tính toán có thể được giảm bớt khi mã hóa/giải mã ma trận lượng tử hóa.

Các bảng 23 và 24 thể hiện sơ lược các ví dụ của các cấu trúc cú pháp có thể được sử dụng khi ma trận lượng tử hóa tham chiếu ID được sử dụng.

Bảng 23

scaling_list_param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag)	
scaling_list_pred_matrix_id_delta	uc(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	
}	

Bảng 24

scaling_list(scalingList, sizeID, matrixID) {	Bộ mô tả
nextCoef = 8	u(1)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
if(sizeID > 1)	
scaling_list_dc_coef_minus8 [sizeID - 2][matrixID]	se(v)
for(i=0; i < coefNum; i++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
scalingList[i] = nextCoef	
}	
}	

Đầu tiên, bộ mã hóa có thể thông tin mã hóa về việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Như trong ví dụ trên bảng 23, bộ mã hóa có thể mã hóa `scaling_list_present_flag`, nghĩa là, thông tin về việc ma trận lượng tử hóa có mặt hay không trong dòng bit, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa không có mặt và các ma trận lượng tử hóa được xác định là ma trận lượng tử hóa mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 0. Nếu ma trận lượng tử hóa được mã hóa có mặt, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 1.

Bộ mã hóa có thể xác định loại phương pháp mã hóa dự đoán cho ma trận lượng tử hóa và thông tin mã hóa về phương pháp đã được xác định dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa được mã hóa, có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 23, bộ mã hóa có thể mã hóa `scaling_list_pred_mode_flag`, nghĩa là, thông tin xác định phương pháp dự đoán và mã hóa ma trận lượng tử hóa, thành bộ tham số. Ví dụ, nếu giá trị hệ số của ma trận lượng tử hóa được mã hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa để dự đoán và mã hóa ma trận lượng tử hóa, thì bộ mã hóa mã hóa giá trị của `scaling_list_pred_mode_flag` là 1. Hơn nữa, nếu ma trận lượng tử hóa tham chiếu và ma trận lượng tử hóa được mã hóa được xác định để có các giá trị giống nhau hoặc các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 0.

Ở đây, để xác định các ma trận hoặc các giá trị hệ số sao cho chúng có các giá trị giống nhau có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa cụ thể sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa thành bộ tham số. Ở đây, bộ tham số có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 23, nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số.

Ở đây, bộ mã hóa có thể xác định giá trị của `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa ID, bằng cách sử dụng MatrixID chỉ báo của ma trận lượng tử hóa được mã hóa, RefMatrixID chỉ báo của ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định, và Phương trình 4.

Phương trình 4

$$\text{scaling_list_pred_matrix_id_delta} = \text{MatrixID} - \text{RefMatrixID}$$

Nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với

các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã, thì bộ mã hóa tạo ra giá trị của RefMatrixID trùng với giá trị của MatrixID bằng cách mã hóa giá trị của `scaling_list_pred_matrix_id_delta` là 0. Ở đây, ma trận mặc định có thể đồng nghĩa với việc ma trận mặc định tương ứng với SizeID và MatrixID.

Nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu (tức là, được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước), thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_matrix_id_delta` làm giá trị không phải là 0 sao cho giá trị của RefMatrixID không trùng với giá trị của MatrixID. Ở đây, giá trị của `scaling_list_pred_matrix_id_delta` có thể là số nguyên dương.

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 24, nếu kích thước của ma trận lượng tử hóa được mã hóa là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ mã hóa có thể mã hóa `scaling_list_dc_coef_minus8`, nghĩa là, hệ số ma trận DC, thành bộ tham số.

Như trong ví dụ trên bảng 24, bộ mã hóa có thể mã hóa `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Trong các bảng 23 và 24, các ví dụ trong đó, thông tin về ma trận lượng tử hóa được mã hóa thành bộ tham số tương thích đã được minh họa, nhưng sáng chế không bị giới hạn ở đó. Ví dụ, bộ mã hóa có thể thông tin mã hóa về ma trận lượng tử hóa thành bộ tham số khác (tức là, bộ tham số bao gồm ít nhất một trong số bộ tham số chuỗi và bộ tham số hình ảnh).

Như được mô tả ở trên, các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống có nhược điểm là hiệu quả mã hóa bị giảm sút do thông tin không cần thiết được mã

hóa/được giải mã khi dự đoán ma trận lượng tử hóa.

Tuy nhiên, theo sáng chế, hiệu quả mã hóa có thể được cải thiện khi mã hóa/giải mã ma trận lượng tử hóa bằng cách thực hiện việc mã hóa/giải mã khác biệt của ma trận lượng tử hóa phụ thuộc vào việc liệu ma trận lượng tử hóa tham chiếu có mặt hay không.

Các bảng 25 và 26 thể hiện một cách sơ lược các ví dụ của các cấu trúc cú pháp có thể được sử dụng khi mã hóa/giải mã ma trận lượng tử hóa được thực hiện theo cách khác phụ thuộc vào việc liệu ma trận lượng tử hóa tham chiếu có mặt hay không.

Bảng 25

scaling_list_param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag = TRUE;	
if (matrixID > 0)	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag && matrixID > 0)	
scaling_list_pred_matrix_id_delta	ue(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID , matrixID)	
}	
}	

Bảng 26

scaling_list(scalingList, sizeID , matrixID) {	Bộ mô tả
nextCoef = 8	u(1)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
UseDefaultScalingMatrix = 0	
if(sizeID > 1) {	
scaling_list_dc_coef_minus8[sizeID - 2][matrixID]	se(v)
if(scaling_list_dc_coef_minus8[sizeID - 2][matrixID] == -8)	
UseDefaultScalingMatrixFlag = 1	
}	
if(UseDefaultScalingMatrixFlag == 0) {	
stopNow = 0	
for(i=0; i < coefNum && !stopNow; i++) {	
scaling_list_delta_coef	se(v)

<code>nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256</code>	
<code>if(sizeID < 2) {</code>	
<code> useDefaultScalingMatrixFlag = (i == 0 && nextCoef == 0)</code>	
<code> if(useDefaultScalingMatrixFlag)</code>	
<code> stopNow = 1</code>	
<code> }</code>	
<code> if(!stopNow)</code>	
<code> scalingList[i] = nextCoef</code>	
<code> }</code>	
<code>}</code>	
<code>}</code>	

Đầu tiên, bộ mã hóa có thể mã hóa thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Như trong ví dụ trên bảng 25, bộ mã hóa có thể mã hóa `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa không có mặt và tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 0. Nếu ma trận lượng tử hóa được mã hóa có mặt, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 1.

Sau khi xác định loại phương pháp dự đoán và mã hóa ma trận lượng tử hóa, nếu giá trị của `MatrixID` lớn hơn 0, thì bộ mã hóa có thể thông tin mã hóa về phương pháp dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 25, bộ mã hóa có thể mã hóa `scaling_list_pred_mode_flag`, nghĩa là, thông tin xác định phương pháp dự đoán và mã hóa ma trận lượng tử hóa, thành bộ tham số chỉ khi giá trị của `MatrixID` lớn hơn 0. Nếu ma trận lượng tử hóa được mã hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa mã hóa giá trị của `scaling_list_pred_mode_flag` là 1. Hơn nữa, nếu ma trận lượng tử hóa tham chiếu được xác định có cùng các giá trị làm ma trận lượng tử hóa được mã hóa, thì bộ mã hóa mã hóa giá trị của `scaling_list_pred_mode_flag` là 0. Ở đây, để xác định các ma trận sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc sử dụng phương pháp dự đoán ma trận

lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa.

Nếu giá trị của MatrixID là 0, thì giá trị của `scaling_list_pred_mode_flag` trở thành TRUE như trong ví dụ trên bảng 23. Do đó, bộ mã hóa không mã hóa giá trị của `scaling_list_pred_mode_flag` và có thể mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0 và giá trị của MatrixID lớn hơn 0, thì bộ mã hóa có thể mã hóa ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà ma trận lượng tử hóa tham chiếu ID được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 25, nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số chỉ khi giá trị của MatrixID lớn hơn 0. Ở đây, bộ mã hóa và bộ giải mã có thể xác định giá trị của `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa ID, bằng cách sử dụng MatrixID chỉ báo của ma trận lượng tử hóa được mã hóa, RefMatrixID chỉ báo của ma trận lượng tử hóa tham chiếu, và Phương trình 5.

Phương trình 5

$$\text{scaling_list_pred_matrix_id_delta} = \text{MatrixID} - (\text{RefMatrixID} + 1)$$

Khi giá trị của MatrixID là 0, thì ma trận lượng tử hóa thứ nhất được chỉ báo cho mọi SizeID. Do ma trận lượng tử hóa có thể được dự đoán chỉ từ ma trận lượng tử hóa có cùng SizeID, đã được mã hóa từ trước, nên ma trận lượng tử hóa tham chiếu có cùng giá trị SizeID không có mặt cho ma trận lượng tử hóa thứ nhất cho mọi SizeID. Do đó, việc dự đoán ma trận lượng tử hóa, như phương pháp sao chép ma trận, không thể được thực hiện. Kết quả là, chỉ khi giá trị của MatrixID lớn hơn 0, thì bộ mã hóa có thể mã hóa `scaling_list_pred_matrix_id_delta` là ma trận lượng tử hóa tham chiếu ID, xác định RefMatrixID dựa trên ma trận lượng tử hóa được mã hóa tham chiếu ID, và xác định các giá trị hệ số của ma trận lượng tử hóa được mã hóa sao cho chúng có cùng các giá trị với

các giá trị hệ số của ma trận lượng tử hóa tham chiếu.

Để xác định các giá trị hệ số của ma trận lượng tử hóa được mã hóa sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, RefMatrixID tương ứng, là ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa.

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ qua việc quét để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 26, nếu kích thước của ma trận lượng tử hóa được mã hóa là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ mã hóa có thể mã hóa scaling_list_dc_coef_minus8, nghĩa là, hệ số ma trận DC, thành bộ tham số.

Như trong ví dụ trên bảng 26, bộ mã hóa có thể mã hóa scaling_list_delta_coef, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số.

Hơn nữa, bộ mã hóa có thể thông tin mã hóa về việc liệu ma trận mặc định đã được sử dụng hay chưa bằng cách sử dụng scaling_list_delta_coef được sử dụng để tính toán scaling_list_dc_coef_minus8 hoặc nextcoef. Nghĩa là, bộ mã hóa có thể mã hóa giá trị của scaling_list_dc_coef_minus8 sao cho nó trở thành -8 để ra lệnh cho bộ giải mã sử dụng ma trận mặc định và có thể mã hóa scaling_list_delta_coef sao cho giá trị nextcoef thứ nhất trở thành 0 để ra lệnh cho bộ giải mã sử dụng ma trận mặc định.

Trong khi đó, phương pháp mã hóa/giải mã ma trận lượng tử hóa trong đó, có hay

không sử dụng ma trận mặc định bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa có nhược điểm là nó tăng độ phức tạp trong quy trình mã hóa/giải mã giá trị hệ số của ma trận lượng tử hóa. Hơn nữa, có nhược điểm là hiệu quả mã hóa bị giảm sút do thông tin không cần thiết được mã hóa/được giải mã khi dự đoán ma trận lượng tử hóa.

Tuy nhiên, sáng chế có thể làm giảm mức độ phức tạp của việc tính toán khi mã hóa/giải mã ma trận lượng tử hóa do có hay không sử dụng ma trận mặc định được xác định bằng cách sử dụng ID của ma trận lượng tử hóa tham chiếu trong mã hóa/giải mã hình ảnh. Hơn nữa, sáng chế có thể cải thiện hiệu quả mã hóa khi mã hóa/giải mã ma trận lượng tử hóa bằng cách mã hóa/giải mã khác biệt ma trận lượng tử hóa phụ thuộc vào việc liệu ma trận lượng tử hóa tham chiếu có mặt hay không.

Các bảng 27 và 28 thể hiện một cách sơ lược các ví dụ của các cấu trúc cú pháp có thể được sử dụng khi ID của ma trận lượng tử hóa tham chiếu được sử dụng.

Bảng 27

scaling_list_param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag && matrixID > 0)	
scaling_list_pred_matrix_id_delta	ue(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	

Bảng 28

scaling_list(scalingList, sizeID , matrixID) {	Bộ mô tả
nextCoef = 8	u(1)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
if(sizeID > 1)	
scaling_list_dc_coef_minus8[sizeID - 2][matrixID]	sc(v)
for(i=0; i < coefNum; i++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
scalingList[i] = nextCoef	
}	
}	

Đầu tiên, bộ mã hóa có thể mã hóa thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Như trong ví dụ trên bảng 27, bộ mã hóa có thể mã hóa `scaling_list_present_flag`, nghĩa là, thông tin xác định liệu ma trận lượng tử hóa có mặt hay không trong dòng bit, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa không có mặt và tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 0. Nếu ma trận lượng tử hóa được mã hóa có mặt, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 1.

Sau khi xác định loại phương pháp mã hóa dự đoán cho ma trận lượng tử hóa, bộ mã hóa có thể thông tin mã hóa về phương pháp dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 27, bộ mã hóa có thể mã hóa `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp mã hóa dự đoán của ma trận lượng tử hóa, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa được mã hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 1. Nếu ma trận lượng tử hóa tham chiếu và ma trận lượng tử hóa được mã hóa được xác định có cùng các giá trị hoặc các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 0. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp

dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa cụ thể sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0 và giá trị của `MatrixID` lớn hơn 0, thì bộ mã hóa có thể thông tin mã hóa về ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa và có hay không sử dụng ma trận mặc định thành bộ tham số. Ở đây, bộ tham số mà thông tin về ma trận lượng tử hóa tham chiếu ID và có hay không sử dụng ma trận mặc định được mã hóa vào trong đó có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 27, nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số chỉ khi giá trị của `MatrixID` lớn hơn 0.

Ở đây, bộ mã hóa có thể xác định `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa ID, bằng cách sử dụng `MatrixID` chỉ báo ma trận lượng tử hóa được mã hóa, `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định, và Phương trình 6.

Phương trình 6

$$\text{scaling_list_pred_matrix_id_delta} = \text{MatrixID} - \text{RefMatrixID}$$

Nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa đã được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_matrix_id_delta` là 0 sao cho giá trị của `RefMatrixID` trùng với giá trị của `MatrixID`. Ở đây, ma trận mặc định nghĩa là ma trận mặc định tương ứng với `SizeID` và `MatrixID`.

Nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_matrix_id_delta` như là giá trị không phải là 0 sao cho giá trị của `RefMatrixID` và giá trị của `MatrixID` không trùng với nhau.

Hơn nữa, khi giá trị của `scaling_list_pred_mode_flag` là 0, thì điều này chỉ báo rằng phương pháp dự đoán và mã hóa ma trận lượng tử hóa được thực hiện để xác định ma trận

lượng tử hóa là cùng một ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định. Trong trường hợp này, ma trận lượng tử hóa có thể được dự đoán từ ma trận lượng tử hóa hoặc ma trận mặc định có cùng SizeID, đã được mã hóa từ trước.

Khi giá trị của MatrixID là 0, có nghĩa là ma trận lượng tử hóa thứ nhất cho mọi SizeID. Do đó, khi giá trị của `scaling_list_pred_mode_flag` là 0 và giá trị của MatrixID là 0, thì ma trận lượng tử hóa tham chiếu có cùng giá trị SizeID cho ma trận lượng tử hóa thứ nhất cho mọi SizeID không có mặt. Do đó, việc dự đoán ma trận lượng tử hóa, như phương pháp sao chép ma trận, không thể được thực hiện trên ma trận lượng tử hóa thứ nhất cho mọi SizeID.

Trong trường hợp này, bộ mã hóa có thể suy ra giá trị của `scaling_list_pred_matrix_id_delta` là 0 mà không mã hóa `scaling_list_pred_matrix_id_delta`. Khi giá trị của `scaling_list_pred_matrix_id_delta` là 0, thì các giá trị hệ số của ma trận lượng tử hóa được mã hóa tương ứng với SizeID và MatrixID có thể được xác định trùng với các giá trị hệ số của ma trận mặc định đã được xác định trong bộ mã hóa và/hoặc bộ giải mã do giá trị của RefMatrixID trùng với giá trị của MatrixID. Ở đây, ma trận mặc định có thể đồng nghĩa với việc ma trận mặc định tương ứng với SizeID và MatrixID.

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ thông qua việc quét để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 28, nếu kích thước của ma trận lượng tử hóa được mã hóa là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ mã hóa có thể mã hóa `scaling_list_dc_coef_minus8`, nghĩa là, hệ số ma trận DC, thành bộ tham số.

`scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của

ma trận lượng tử hóa được mã hóa, có thể được mã hóa thành bộ tham số như trong ví dụ trên bảng 28.

Trong các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống, khi gửi ma trận lượng tử hóa, tất cả các hệ số và hệ số ma trận DC nằm trong ma trận được mã hóa/được giải mã. Trong trường hợp này, sự cải thiện hiệu quả mã hóa bị giới hạn do việc dự đoán và mã hóa/giải mã không được thực hiện trên hệ số ma trận DC.

Trái lại, nếu việc dự đoán và mã hóa/giải mã được thực hiện trên hệ số ma trận DC nằm trong ma trận lượng tử hóa, thì hiệu quả mã hóa có thể được cải thiện.

Ở đây, sáng chế đề xuất phương pháp dự đoán hệ số ma trận DC từ các hệ số AC lân cận bằng cách sử dụng mối tương quan cao giữa các hệ số lân cận mà không dự đoán hệ số ma trận DC từ hằng số 8. Sáng chế có thể cải thiện hiệu quả mã hóa qua phương pháp này.

Các bảng 29 và 30 thể hiện một cách sơ lược các ví dụ của các cấu trúc cú pháp có thể được sử dụng khi dự đoán hệ số ma trận DC bằng cách sử dụng tương quan giữa các hệ số lân cận.

Bảng 29

scaling_list_param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag)	
scaling_list_pred_matrix_id_delta	ue(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID , matrixID)	
}	
}	
}	

Bảng 30

scaling_list(scalingList, sizeID , matrixID) {	Bộ mô tả
nextCoef = 8	u(1)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
UseDefaultScalingMatrix = 0	
stopNow = 0	
for(i=0; i < coefNum && !stopNow; i++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
if(sizeID < 2) {	
useDefaultScalingMatrixFlag = (i == 0 && nextCoef == 0)	
if(useDefaultScalingMatrixFlag)	
stopNow = 1	
}	
if(!stopNow)	
scalingList[i] = nextCoef	
}	
if(sizeID > 1 && !useDefaultScalingMatrixFlag)	
scaling_list_dc_coef_res[sizeID - 2][matrixID]	se(v)
}	

Theo các ví dụ trên các bảng 29 và 30, chuỗi mã hóa/giải mã ma trận lượng tử hóa được tạo ra trùng với chuỗi phục hồi của ma trận lượng tử hóa. Trong trường hợp này, không gian nhớ để lưu trữ các hệ số ma trận DC có thể được giảm bớt. Hơn nữa, trong các ví dụ trên các bảng 29 và 30, khi chỉ báo có hay không sử dụng ma trận mặc định, chỉ có thể sử dụng thành phần cú pháp ‘scaling_list_delta_coef’ mà không sử dụng nhiều thành phần cú pháp.

Đầu tiên, bộ mã hóa có thể mã hóa thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Như trong ví dụ trên bảng 28, bộ mã hóa có thể mã hóa scaling_list_present_flag, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa không có mặt và tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định, thì bộ mã hóa có thể mã hóa giá trị của scaling_list_present_flag là 0. Nếu ma trận lượng tử hóa được mã hóa có mặt, thì bộ mã hóa có thể mã hóa giá trị của scaling_list_present_flag là 1.

Sau khi xác định loại phương pháp dự đoán và mã hóa ma trận lượng tử hóa, bộ mã hóa có thể thông tin mã hóa về phương pháp dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và mã hóa ma trận

lượng tử hóa được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 29, bộ mã hóa có thể mã hóa `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa được mã hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 1. Nếu ma trận lượng tử hóa tham chiếu được xác định có cùng các giá trị với ma trận lượng tử hóa được mã hóa, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 0. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc sử dụng phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà ma trận lượng tử hóa tham chiếu ID được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 29, nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ mã hóa có thể xác định `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa ID, bằng cách sử dụng `MatrixID` chỉ báo ma trận lượng tử hóa được mã hóa, `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu, và Phương trình 7 dưới đây.

Phương trình 7

$$\text{scaling_list_pred_matrix_id_delta} = \text{MatrixID} - (\text{RefMatrixID} + 1)$$

Để xác định các giá trị hệ số của ma trận lượng tử hóa được mã hóa sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với `RefMatrixID`, làm ma trận lượng tử hóa tham chiếu của ma trận

lượng tử hóa được mã hóa và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa được sử dụng.

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ thông qua việc quét để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 30, bộ mã hóa có thể mã hóa `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số.

Ở đây, bộ mã hóa có thể thông tin mã hóa về việc có hay không sử dụng ma trận mặc định bằng cách sử dụng `scaling_list_delta_coef` được sử dụng để tính toán `nextcoef`. Ví dụ, bộ mã hóa có thể mã hóa `scaling_list_delta_coef` sao cho giá trị `nextcoef` thứ nhất trở thành 0 để ra lệnh cho bộ giải mã sử dụng ma trận mặc định.

Như trong ví dụ trên bảng 30, bộ mã hóa có thể mã hóa `scaling_list_dc_coef_res`, nghĩa là, giá trị chênh lệch giữa các giá trị hệ số của các ma trận lượng tử hóa tương ứng với hệ số ma trận DC, thành bộ tham số. Ở đây, `scaling_list_dc_coef_res` có thể được mã hóa thành (`useDefaultScalingMatrixFlag=0`) nếu kích thước của ma trận lượng tử hóa được mã hóa là ma trận lượng tử hóa 16x16 (`SizeID=2`) hoặc 32x32 (`SizeID=3`) và ma trận mặc định không được sử dụng.

Theo ví dụ trên bảng 30, giá trị của `scaling_list_dc_coef_res` cho 16x16-size hoặc ma trận lượng tử hóa có kích thước 32x32 để mã hóa riêng hệ số ma trận DC có thể được tính toán bằng cách sử dụng hiệu số giữa giá trị của hệ số ma trận DC và ma trận hệ số có mặt tại vị trí DC.

Trong khi đó, phương pháp mã hóa/giải mã ma trận lượng tử hóa để xác định xem

có hay không sử dụng ma trận mặc định bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa có nhược điểm là nó tăng độ phức tạp trong quy trình mã hóa/giải mã giá trị hệ số của ma trận lượng tử hóa. Hơn nữa, thông thường, khi gửi ma trận lượng tử hóa, tất cả các hệ số và hệ số ma trận DC nằm trong ma trận được mã hóa/được giải mã. Trong trường hợp này, sự cải thiện hiệu quả mã hóa bị giới hạn do việc dự đoán và mã hóa/giải mã không được thực hiện trên hệ số ma trận DC.

Theo sáng chế, trong việc mã hóa/giải mã hình ảnh, việc có hay không sử dụng ma trận mặc định được xác định bằng cách sử dụng ID của ma trận lượng tử hóa tham chiếu. Do đó, mức độ phức tạp của việc tính toán khi mã hóa/giải mã ma trận lượng tử hóa có thể được giảm bớt. Hơn nữa, trong sáng chế, do việc dự đoán và mã hóa/giải mã được thực hiện trên hệ số ma trận DC nằm trong ma trận lượng tử hóa, nên hiệu quả mã hóa có thể được cải thiện. Hơn nữa, hiệu quả mã hóa có thể được cải thiện do hệ số ma trận DC được dự đoán từ các hệ số AC lân cận bằng cách sử dụng mối tương quan cao giữa các hệ số AC lân cận mà không dự đoán hằng số 8.

Các bảng 31 và 32 thể hiện một cách sơ lược một ví dụ của các cấu trúc cú pháp có thể được sử dụng khi tất cả các đặc điểm trên được áp dụng.

Bảng 31

<code>scaling_list_param() {</code>	Bộ mô tả
<code> scaling_list_present_flag</code>	<code>u(1)</code>
<code> if(scaling_list_present_flag)</code>	
<code> for(sizeID = 0; sizeID < 4; sizeID++)</code>	
<code> for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {</code>	
<code> scaling_list_pred_mode_flag</code>	<code>u(1)</code>
<code> if(!scaling_list_pred_mode_flag)</code>	
<code> scaling_list_pred_matrix_id_delta</code>	<code>ue(v)</code>
<code> else</code>	
<code> scaling_list(ScalingList[sizeID][matrixID], sizeID , matrixID)</code>	
<code> }</code>	
<code>}</code>	

Bảng 32

scaling_list(scalingList, sizeID , matrixID) {	Bộ mô tả
nextCoef = 8	u(1)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
for(i=0; i < coefNum; i++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
scalingList[i] = nextCoef	
}	
if(sizeID > 1)	
scaling_list_dc_coef_res[sizeID - 2][matrixID]	se(v)
}	

Theo các ví dụ trên các bảng 31 và 32, chuỗi mã hóa/giải mã ma trận lượng tử hóa được tạo ra trùng với chuỗi phục hồi của ma trận lượng tử hóa. Do đó, không gian nhớ để lưu trữ các hệ số ma trận DC có thể được giảm bớt. Hơn nữa, trong các ví dụ trên các bảng 31 và 32, khi chỉ báo có hay không sử dụng ma trận mặc định, nhiều thành phần cú pháp không được sử dụng, mà chỉ thành phần cú pháp `scaling_list_delta_coef` có thể được sử dụng.

Đầu tiên, bộ mã hóa có thể mã hóa thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Như trong ví dụ trên bảng 31, bộ mã hóa có thể mã hóa `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa không có mặt và tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 0. Nếu ma trận lượng tử hóa được mã hóa có mặt, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 1.

Sau khi xác định loại phương pháp mã hóa dự đoán cho ma trận lượng tử hóa, bộ mã hóa có thể thông tin mã hóa về phương pháp dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 31, bộ mã hóa có thể mã hóa `scaling_list_pred_mode_flag`, nghĩa là, thông tin xác định phương pháp dự đoán và mã hóa ma trận lượng tử hóa, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa được mã hóa theo các phương pháp DPCM và

mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa mã hóa giá trị của `scaling_list_pred_mode_flag` là 1. Hơn nữa, nếu ma trận lượng tử hóa tham chiếu và ma trận lượng tử hóa được mã hóa được xác định có cùng các giá trị hoặc nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận mặc định, thì bộ mã hóa mã hóa giá trị của `scaling_list_pred_mode_flag` là 0. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa cụ thể sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa và thông tin về có hay không sử dụng ma trận mặc định thành bộ tham số. Ở đây, bộ tham số mà việc mã hóa được thực hiện, có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 31, nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số.

Ở đây, bộ mã hóa và bộ giải mã có thể xác định giá trị của `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa ID, bằng cách sử dụng `MatrixID` chỉ báo ma trận lượng tử hóa được mã hóa, `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định, và Phương trình 8.

Phương trình 8

$$\text{scaling_list_pred_matrix_id_delta} = \text{MatrixID} - \text{RefMatrixID}$$

Nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_matrix_id_delta` là 0 sao cho giá trị của `RefMatrixID` trùng với giá trị của `MatrixID`. Ở đây, ma trận mặc định nghĩa là ma trận mặc định tương ứng với `SizeID` và `MatrixID`.

Nếu giá trị của `scaling_list_pred_matrix_id_delta` là 0, thì bộ mã hóa có thể mã hóa

giá trị của `scaling_list_pred_matrix_id_delta` là giá trị không phải là 0 sao cho giá trị của `RefMatrixID` không trùng với giá trị của `MatrixID`.

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ thông qua việc quét để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 32, bộ mã hóa có thể mã hóa `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số.

Như trong ví dụ trên bảng 32, bộ mã hóa có thể mã hóa `scaling_list_dc_coef_res`, nghĩa là, giá trị chênh lệch giữa các giá trị hệ số của các ma trận lượng tử hóa tương ứng với hệ số ma trận DC, thành bộ tham số. Ở đây, `scaling_list_dc_coef_res` có thể được mã hóa thành (`useDefaultScalingMatrixFlag=0`) nếu kích thước của ma trận lượng tử hóa được mã hóa là 16x16 (`SizeID=2`) hoặc 32x32 (`SizeID=3`) và ma trận mặc định không được sử dụng.

Giá trị của `scaling_list_dc_coef_res` cho ma trận lượng tử hóa có kích thước 16x16 hoặc ma trận lượng tử hóa có kích thước 32x32 để mã hóa riêng hệ số ma trận DC có thể được tính toán bằng cách sử dụng hiệu số giữa giá trị của hệ số ma trận DC và ma trận hệ số có mặt tại vị trí DC.

Trong khi đó, trong các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống, ma trận lượng tử hóa được sao chép bằng cách sử dụng kích thước của ma trận lượng tử hóa khi việc lượng tử hóa và giải lượng tử hóa được thực hiện không phải là kích thước của ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện. Do đó, ma trận lượng tử hóa được sao chép từ số lượng giới hạn các ma trận lượng tử hóa. Kết quả là, sự cải thiện hiệu quả mã hóa bị giới hạn khi ma trận lượng tử hóa được mã hóa/được giải mã.

Theo sáng chế, tuy nhiên, hiệu quả mã hóa có thể được cải thiện và mức độ tự do trong việc dự đoán ma trận lượng tử hóa có thể được tăng do ma trận lượng tử hóa được dự đoán từ ma trận lượng tử hóa có cùng kích thước với ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện.

Các bảng 33 và 34 thể hiện một cách sơ lược các ví dụ của các cấu trúc cú pháp có thể được sử dụng khi ma trận lượng tử hóa được dự đoán từ ma trận lượng tử hóa có cùng kích thước với ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện.

Bảng 33

scaling_list_param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag) {	
scaling_list_pred_size_matrix_id_delta	ue(v)
}	
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	

Bảng 34

scaling_list_param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag) {	
scaling_list_pred_size_id_delta	ue(v)
scaling_list_pred_matrix_id_delta	se(v)
}	
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	

Đầu tiên, bộ mã hóa có thể mã hóa thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Như trong ví dụ trên bảng 33 hoặc 34, bộ mã hóa có thể mã hóa `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa không có mặt và tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 0. Nếu ma trận lượng tử hóa được mã hóa có mặt, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 1.

Sau khi xác định loại phương pháp mã hóa dự đoán cho ma trận lượng tử hóa, bộ mã hóa có thể thông tin mã hóa về phương pháp dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa được mã hóa có thể là bộ tham số tương thích. Ví dụ, như trong ví dụ trên bảng 33 hoặc bảng 34, bộ mã hóa có thể mã hóa `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa, thành bộ tham số. Ví dụ chi tiết, nếu ma trận lượng tử hóa được mã hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 1. Nếu ma trận lượng tử hóa tham chiếu và ma trận lượng tử hóa được mã hóa được xác định có cùng các giá trị, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 0. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc sử dụng phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa.

Hơn nữa, nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, ít nhất một thông tin, quy định kích thước của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa, và thông tin, quy định ma trận lượng tử hóa tham chiếu, có thể được mã hóa thành bộ tham số làm thông tin ID (identifier - mã nhận dạng) về ma trận lượng tử hóa tham chiếu. Bộ tham số được mã hóa có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 33, nếu các giá trị hệ số của ma trận lượng tử hóa

được mã hóa được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu, thì bộ mã hóa có thể mã hóa `scaling_list_pred_size_matrix_id_delta`, nghĩa là, thông tin ID về ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, giá trị của `scaling_list_pred_size_matrix_id_delta`, nghĩa là, thông tin ID về ma trận lượng tử hóa, có thể được xác định bằng cách sử dụng `RefSizeID`, nghĩa là, kích thước của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa, `RefMatrixID` chỉ báo của ma trận lượng tử hóa tham chiếu, và Phương trình 9.

Phương trình 9

$$\text{scaling_list_pred_size_matrix_id_delta} = 6 * (\text{RefSizeID} - \text{SizeID}) + (\text{RefMatrixID} \% 6)$$

6)

Với ví dụ khác, như trong ví dụ trên bảng 34, nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu, thì bộ mã hóa có thể mã hóa `scaling_list_pred_size_id_delta` và `scaling_list_pred_size_matrix_id_delta`, nghĩa là, thông tin ID về ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, giá trị của `scaling_list_pred_size_id_delta` có thể được xác định bằng cách sử dụng `RefSizeID` và Phương trình 10, và giá trị của `scaling_list_pred_matrix_id_delta` có thể được xác định bằng cách sử dụng `RefMatrixID` chỉ báo của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa và Phương trình 11.

Phương trình 10

$$\text{scaling_list_pred_size_id_delta} = \text{SizeID} - \text{RefSizeID}$$

Phương trình 11

$$\text{scaling_list_pred_matrix_id_delta} = \text{MatrixID} - \text{RefMatrixID}$$

Để xác định các giá trị hệ số của ma trận lượng tử hóa được mã hóa trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa được sử dụng.

Do đó, không cho phép dự đoán ma trận lượng tử hóa từ ma trận lượng tử hóa có

cùng SizeID, nhưng có thể dự đoán ma trận lượng tử hóa từ ma trận lượng tử hóa có cùng kích thước ma trận, nhưng SizeID khác khi mã hóa/giải mã được thực hiện qua ví dụ trên bảng 33 hoặc bảng 34.

Hơn nữa, trong ví dụ trên bảng 33 hoặc bảng 34, giới hạn của các giá trị của các thành phần cú pháp có thể được giới hạn ở giá trị cụ thể. Ví dụ, trong ví dụ trên bảng 33 hoặc bảng 34, `scaling_list_pred_size_matrix_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 17, `scaling_list_pred_size_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 2, và `scaling_list_pred_matrix_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 5.

Hơn nữa, trong ví dụ trên bảng 33 hoặc bảng 34, bộ mã hóa có thể không dự đoán ma trận lượng tử hóa từ ma trận lượng tử hóa có kích thước lớn hơn ma trận lượng tử hóa được mã hóa.

Hơn nữa, bộ mã hóa có thể phân loại ma trận lượng tử hóa thành hệ số ma trận DC và hệ số ma trận AC theo kích thước của mỗi ma trận lượng tử hóa và dự đoán ma trận lượng tử hóa cho hệ số ma trận DC và hệ số ma trận AC. Ví dụ, ma trận lượng tử hóa có kích thước 8x8 có thể được phân loại thành hệ số ma trận DC và hệ số ma trận AC và sau đó được dự đoán, và cả hệ số ma trận DC và hệ số ma trận AC có thể được dự đoán liên quan tới các ma trận lượng tử hóa có các kích thước còn lại. Theo cách khác, khi thực hiện việc dự đoán từ ma trận lượng tử hóa có kích thước 8x8, bộ mã hóa có thể dự đoán giá trị tại vị trí tương ứng bằng cách xác định một giá trị, tương ứng với vị trí hệ số ma trận DC nằm trong ma trận lượng tử hóa có kích thước 8x8, như hệ số ma trận DC của ma trận lượng tử hóa được mã hóa. Nếu việc dự đoán được thực hiện từ ma trận lượng tử hóa có kích thước 16x16 hoặc 32x32, thì bộ mã hóa cũng có thể dự đoán hệ số ma trận DC của ma trận lượng tử hóa.

Trong khi đó, nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ thông qua việc quét để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa

được mã hóa thành bộ tham số. Ở đây, bộ tham số có thể là bộ tham số tương thích.

Bảng 35 thể hiện ví dụ của cấu trúc cú pháp có thể được sử dụng khi hệ số nằm trong ma trận lượng tử hóa được dự đoán bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa.

Bảng 35

<code>scaling_list(scalingList, sizeID , matrixID) {</code>	Bộ mô tả
<code>nextCoef = 8</code>	<code>u(1)</code>
<code>coefNum = Min(64, (1 << (4 + (sizeID << 1))))</code>	
<code>UseDefaultScalingMatrix = 0</code>	
<code>if(sizeID > 1) {</code>	
<code> scaling_list_dc_coef_minus8[sizeID - 2][matrixID]</code>	<code>se(v)</code>
<code> if(scaling_list_dc_coef_minus8[sizeID - 2][matrixID] == -8)</code>	
<code> UseDefaultScalingMatrixFlag = 1</code>	
<code> }</code>	
<code>if(UseDefaultScalingMatrixFlag == 0) {</code>	
<code> stopNow = 0</code>	
<code> for(i=0; i < coefNum && !stopNow; i++) {</code>	
<code> scaling_list_delta_coef</code>	<code>se(v)</code>
<code> }</code>	
<code>nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256</code>	
<code>if(sizeID < 2) {</code>	
<code> useDefaultScalingMatrixFlag = (i == 0 && nextCoef == 0)</code>	
<code> if(useDefaultScalingMatrixFlag)</code>	
<code> stopNow = 1</code>	
<code> }</code>	
<code>if(!stopNow)</code>	
<code> scalingList[i] = nextCoef</code>	
<code> }</code>	
<code>}</code>	

Như trong ví dụ trên bảng 35, nếu kích thước của ma trận lượng tử hóa được mã hóa là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ mã hóa có thể mã hóa `scaling_list_dc_coef_minus8`, nghĩa là, thông tin về hệ số ma trận DC, thành bộ tham số.

Như trong ví dụ trên bảng 35, bộ mã hóa có thể mã hóa `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số.

Hơn nữa, bộ mã hóa có thể thông tin mã hóa về việc có hay không sử dụng ma trận

mặc định bằng cách sử dụng `scaling_list_delta_coef` được sử dụng để suy ra giá trị của `scaling_list_dc_coef_minus8` hoặc `nextcoef`. Ví dụ, bộ mã hóa có thể mã hóa giá trị của `scaling_list_dc_coef_minus8` là -8 để ra lệnh cho bộ giải mã sử dụng ma trận mặc định và có thể mã hóa `scaling_list_delta_coef` sao cho giá trị `nextcoef` thứ nhất là 0 để ra lệnh cho bộ giải mã sử dụng ma trận mặc định.

Theo cách khác, phương pháp mã hóa/giải mã ma trận lượng tử hóa để xác định xem có hay không sử dụng ma trận mặc định bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa có nhược điểm là nó tăng mức độ phức tạp trong quy trình mã hóa/giải mã giá trị hệ số của ma trận lượng tử hóa. Hơn nữa, có nhược điểm là hiệu quả mã hóa bị giảm sút do thông tin không cần thiết được mã hóa/được giải mã khi ma trận lượng tử hóa được dự đoán. Hơn nữa, ma trận lượng tử hóa được sao chép bằng cách sử dụng kích thước của ma trận lượng tử hóa khi việc lượng tử hóa và giải lượng tử hóa được thực hiện không phải là kích thước của ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện. Do ma trận lượng tử hóa được sao chép từ số lượng giới hạn các ma trận lượng tử hóa, nên sự cải thiện hiệu quả mã hóa bị giới hạn khi mã hóa/giải mã ma trận lượng tử hóa.

Theo sáng chế, trong việc mã hóa/giải mã hình ảnh, việc có hay không sử dụng ma trận mặc định có thể được xác định bằng cách sử dụng ID của ma trận lượng tử hóa tham chiếu. Do đó, mức độ phức tạp của việc tính toán khi mã hóa/giải mã ma trận lượng tử hóa có thể được giảm bớt. Hơn nữa, hiệu quả mã hóa có thể được cải thiện và mức độ tự do trong dự đoán ma trận lượng tử hóa có thể được tăng do ma trận lượng tử hóa được dự đoán từ ma trận lượng tử hóa có cùng kích thước với ma trận lượng tử hóa khi mã hóa/giải mã ma trận lượng tử hóa.

Các bảng 36 và 37 thể hiện một cách sơ lược các ví dụ của các cấu trúc cú pháp có thể được sử dụng khi dự đoán ma trận lượng tử hóa bằng cách sử dụng ID của ma trận lượng tử hóa tham chiếu và sử dụng ma trận lượng tử hóa có cùng kích thước khi mã hóa/giải mã ma trận lượng tử hóa.

Bảng 36

scaling_list param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag) {	
scaling_list_pred_size_matrix_id_delta	uc(v)
}	
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID , matrixID)	
}	
}	

Bảng 37

scaling_list param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag) {	
scaling_list_pred_size_id_delta	ue(v)
scaling_list_pred_matrix_id_delta	sc(v)
}	
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID , matrixID)	
}	
}	

Đầu tiên, bộ mã hóa có thể mã hóa thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Như trong ví dụ trên bảng 36 hoặc 37, bộ mã hóa có thể mã hóa scaling_list_present_flag, nghĩa là, thông tin về việc ma trận lượng tử hóa có mặt hay không trong dòng bit, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa không có mặt và tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định, thì bộ mã hóa mã hóa giá trị của scaling_list_present_flag là 0. Nếu ma trận lượng tử hóa được mã hóa có mặt, thì bộ mã hóa mã hóa giá trị của scaling_list_present_flag là 1.

Sau khi xác định loại phương pháp mã hóa dự đoán cho ma trận lượng tử hóa, bộ mã hóa có thể thông tin mã hóa về phương pháp dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 36 hoặc bảng 37, bộ mã hóa có thể mã hóa `scaling_list_pred_mode_flag`, nghĩa là, thông tin xác định phương pháp dự đoán và mã hóa ma trận lượng tử hóa, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa được mã hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 1. Hơn nữa, nếu ma trận lượng tử hóa tham chiếu và ma trận lượng tử hóa được mã hóa được xác định có cùng các giá trị hoặc nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 0. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa cụ thể sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, ít nhất một thông tin, quy định kích thước của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa, và thông tin, quy định ma trận lượng tử hóa tham chiếu, có thể được mã hóa thành bộ tham số làm thông tin ID về ma trận lượng tử hóa tham chiếu. Bộ tham số được mã hóa có thể là bộ tham số tương thích. Hơn nữa, bộ tham số mà ID hoặc thông tin ID được mã hóa có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 36, nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa `scaling_list_pred_size_matrix_id_delta`, nghĩa là, thông tin ID về ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ mã hóa có thể xác định `scaling_list_pred_size_matrix_id_delta`, nghĩa là, thông tin ID về ma trận lượng tử hóa, bằng cách sử dụng `RefSizeID` và `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định và Phương trình 12.

Phương trình 12

$$\text{scaling_list_pred_size_matrix_id_delta} = 6 * (\text{RefSizeID} - \text{SizeID}) + (\text{RefMatrixID} \% 6)$$

6)

Theo ví dụ khác, như trong bảng 37, nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu hoặc nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận mặc định, thì bộ mã hóa có thể mã hóa $\text{scaling_list_pred_size_id_delta}$ và $\text{scaling_list_pred_size_matrix_id_delta}$, nghĩa là, thông tin ID về ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ mã hóa có thể suy ra giá trị của $\text{scaling_list_pred_size_id_delta}$ bằng cách sử dụng RefSizeID và Phương trình 13 và có thể suy ra giá trị của $\text{scaling_list_pred_matrix_id_delta}$ bằng cách sử dụng RefMatrixID chỉ báo của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được mã hóa hoặc ma trận mặc định và Phương trình 14.

Phương trình 13

$$\text{scaling_list_pred_size_id_delta} = \text{SizeID} - \text{RefSizeID}$$

Phương trình 14

$$\text{scaling_list_pred_matrix_id_delta} = \text{MatrixID} - \text{RefMatrixID}$$

Nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã, thì bộ mã hóa có thể mã hóa giá trị của $\text{scaling_list_pred_matrix_id_delta}$ là 0 sao cho giá trị của RefMatrixID và giá trị của MatrixID trùng với nhau. Ở đây, ma trận mặc định nghĩa là ma trận mặc định tương ứng với SizeID và MatrixID .

Nếu các giá trị hệ số của ma trận lượng tử hóa được mã hóa được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu, thì bộ mã hóa có thể mã hóa giá trị của $\text{scaling_list_pred_matrix_id_delta}$ là giá trị không phải là 0 sao cho giá trị của RefMatrixID và giá trị của MatrixID không trùng với nhau.

Do đó, theo cách làm ví dụ trên bảng 36 hoặc bảng 37, ma trận lượng tử hóa có thể

được dự đoán từ ma trận lượng tử hóa có cùng SizeID, và ma trận lượng tử hóa có thể được dự đoán từ ma trận lượng tử hóa có cùng kích thước, nhưng SizeID khác khi mã hóa/giải mã được thực hiện có thể được dự đoán.

Hơn nữa, trong ví dụ trên bảng 36 hoặc bảng 37, giới hạn của các giá trị của `scaling_list_pred_size_matrix_id_delta` và `caling_list_pred_size_id_delta`, `scaling_list_pred_matrix_id_delta` có thể bị giới hạn. Ví dụ, `scaling_list_pred_size_matrix_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 17, `scaling_list_pred_size_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 2, và `scaling_list_pred_matrix_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 5.

Hơn nữa, việc dự đoán ma trận lượng tử hóa có thể không được thực hiện từ ma trận lượng tử hóa có kích thước lớn hơn ma trận lượng tử hóa được mã hóa.

Trong việc dự đoán ma trận lượng tử hóa, phương pháp dự đoán ma trận lượng tử hóa có thể được thực hiện theo cách khác bằng cách lấy kích thước của ma trận để xem xét. Ví dụ, khi thực hiện trước việc dự đoán từ ma trận lượng tử hóa có kích thước 8x8, thì bộ mã hóa có thể dự đoán giá trị tại vị trí tương ứng bằng cách xác định giá trị, tương ứng với vị trí của hệ số ma trận DC nằm trong ma trận lượng tử hóa có kích thước 8x8, làm hệ số ma trận DC. Nếu việc dự đoán được thực hiện từ ma trận lượng tử hóa có kích thước 16x16 hoặc 32x32, thì bộ mã hóa có thể cũng dự đoán hệ số ma trận DC.

Trong khi đó, nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ thông qua việc quét để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được mã hóa có thể là bộ tham số tương thích.

Bảng 38 thể hiện ví dụ của cấu trúc cú pháp có thể được sử dụng khi dự đoán hệ số nằm trong ma trận lượng tử hóa bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa.

Bảng 38

scaling_list(scalingList, sizeID , matrixID) {	Bộ mô tả
nextCoef = 8	u(l)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
if(sizeID > 1)	
scaling_list_dc_coef_minus8[sizeID - 2][matrixID]	se(v)
for(i=0; i < coefNum; i++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
scalingList[i] = nextCoef	
}	
}	

Như trong ví dụ trên bảng 38, nếu kích thước của ma trận lượng tử hóa được mã hóa là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ mã hóa có thể mã hóa scaling_list_dc_coef_minus8, nghĩa là, thông tin xác định hệ số ma trận DC, thành bộ tham số.

Như trong ví dụ trên bảng 38, bộ mã hóa có thể mã hóa scaling_list_delta_coef, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số.

Trong các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống, giá trị hệ số của ma trận lượng tử hóa được mã hóa bằng cách không lấy giá trị hệ số, sự xuất hiện thường xuyên khi mã hóa/giải mã hệ số thứ nhất nằm trong ma trận lượng tử hóa, để xem xét. Do đó, trong các phương pháp mã hóa/giải mã ma trận lượng tử hóa thông thường, sự cải thiện hiệu quả mã hóa bị giới hạn.

Theo sáng chế, hệ số thứ nhất nằm trong ma trận lượng tử hóa có thể được dự đoán và được mã hóa/được giải mã bằng cách sử dụng giá trị hệ số thường xuyên xuất hiện. Hơn nữa, trong sáng chế, nếu giá trị hệ số thứ nhất hoặc giá trị hệ số ma trận DC của ma trận mặc định được xác định là 16 hoặc giá trị hệ số thứ nhất hoặc giá trị hệ số ma trận DC của ma trận không phải là ma trận mặc định được phân bố xung quanh 16, thì giá trị hệ số thứ nhất hoặc giá trị hệ số ma trận DC nằm trong ma trận lượng tử hóa được mã hóa/được giải mã có thể được dự đoán từ hằng số 16 và sau đó sẽ được mã hóa/được giải mã. Trong

trường hợp này, sáng chế có thể cải thiện hiệu quả mã hóa.

Các bảng 39 và 40 thể hiện một cách sơ lược các ví dụ của các cấu trúc cú pháp có thể được sử dụng khi hệ số thứ nhất nằm trong ma trận lượng tử hóa được xem xét.

Bảng 39

scaling_list_param() {	Bộ mô tả
scaling_list_present_flag	u(1)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if(!scaling_list_pred_mode_flag)	
scaling_list_pred_matrix_id_delta	ue(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	
}	

Bảng 40

scaling_list(scalingList, sizeID, matrixID) {	Bộ mô tả
nextCoef = 16	u(1)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
UseDefaultScalingMatrix = 0	
if(sizeID > 1) {	
scaling_list_dc_coef_minus16 [sizeID - 2][matrixID]	se(v)
if(scaling_list_dc_coef_minus16[sizeID - 2][matrixID] == -16)	
UseDefaultScalingMatrixFlag = 1	
}	
if(UseDefaultScalingMatrixFlag == 0) {	
stopNow = 0	
for(i=0; i < coefNum && !stopNow; i++) {	
scaling_list_delta_coef	se(v)
}	
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
if(sizeID < 2) {	
useDefaultScalingMatrixFlag = (i == 0 && nextCoef == 0)	
if(useDefaultScalingMatrixFlag)	
stopNow = 1	
}	
if(!stopNow)	
scalingList[i] = nextCoef	
}	
}	
}	

Đầu tiên, bộ mã hóa có thể mã hóa thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không thành bộ tham số.

Như trong ví dụ trên bảng 39, bộ mã hóa có thể mã hóa `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa không có mặt và tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 0. Nếu ma trận lượng tử hóa được mã hóa có mặt, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_present_flag` là 1.

Sau khi xác định loại phương pháp mã hóa dự đoán cho ma trận lượng tử hóa, bộ mã hóa có thể thông tin mã hóa về phương pháp dự đoán và mã hóa ma trận lượng tử hóa thành bộ tham số. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 39, bộ mã hóa có thể mã hóa `scaling_list_pred_mode_flag`, nghĩa là, thông tin thông tin về phương pháp dự đoán và mã hóa ma trận lượng tử hóa, thành bộ tham số. Ví dụ, nếu ma trận lượng tử hóa được mã hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ bằng cách quét ma trận lượng tử hóa để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 1. Hơn nữa, nếu ma trận lượng tử hóa tham chiếu và ma trận lượng tử hóa được mã hóa được xác định có cùng các giá trị, thì bộ mã hóa có thể mã hóa giá trị của `scaling_list_pred_mode_flag` là 0. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc sử dụng phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà ma trận lượng tử hóa tham chiếu ID được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 39, nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ mã hóa có thể mã hóa `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa

tham chiếu ID của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ mã hóa có thể xác định `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa ID, bằng cách sử dụng `MatrixID` chỉ báo ma trận lượng tử hóa được mã hóa, `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu, và Phương trình 15.

Phương trình 15

$$\text{scaling_list_pred_matrix_id_delta} = \text{MatrixID} - (\text{RefMatrixID} + 1)$$

Để xác định các giá trị hệ số của ma trận lượng tử hóa được mã hóa sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được mã hóa được sử dụng.

Nếu phương pháp dự đoán và mã hóa ma trận lượng tử hóa là phương pháp mã hóa ma trận lượng tử hóa theo các phương pháp DPCM và mã hóa Golomb hàm mũ thông qua việc quét để dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa, thì bộ mã hóa có thể mã hóa giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được mã hóa có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 40, nếu kích thước của ma trận lượng tử hóa được mã hóa là 16×16 (`SizeID=2`) hoặc 32×32 (`SizeID=3`), thì bộ mã hóa có thể mã hóa `scaling_list_dc_coef_minus16`, nghĩa là, giá trị hệ số của ma trận lượng tử hóa tương ứng với hệ số ma trận DC, thành bộ tham số. Ở đây, giá trị của `scaling_list_dc_coef_minus16` chỉ báo hệ số ma trận DC được tính toán giả sử rằng giá trị dự đoán là 16.

Như trong ví dụ trên bảng 40, bộ mã hóa có thể mã hóa `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được mã hóa từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được mã hóa thành bộ tham số. Như trong ví dụ trên bảng 40, bộ mã hóa có thể đặt '`nextcoef=16`', nghĩa là, đặt giá trị dự đoán cho giá trị hệ số thứ nhất là 16.

Hơn nữa, bộ mã hóa có thể xác định xem có hay không sử dụng ma trận mặc định

bằng cách sử dụng `scaling_list_delta_coef` được sử dụng để được tính toán `scaling_list_dc_coef_minus16` hoặc `nextcoef`. Nghĩa là, bộ mã hóa có thể mã hóa giá trị của `scaling_list_dc_coef_minus16` là -16 để ra lệnh cho bộ giải mã sử dụng ma trận mặc định và có thể mã hóa `scaling_list_delta_coef` sao cho giá trị của `nextcoef` thứ nhất trở thành 0 để ra lệnh cho bộ giải mã sử dụng ma trận mặc định.

Theo các phương án của các bảng ở trên và các bảng dưới đây cũng như các ví dụ trên các bảng 39 và 40, bộ mã hóa có thể đặt `nextcoef` thành 16, và giá trị của `scaling_list_dc_coef_minus16` có thể đồng nghĩa với việc hệ số ma trận DC được tính toán giả sử rằng giá trị dự đoán là 16. Hơn nữa, trong các phương án của các bảng nêu trên và các bảng dưới đây, bộ mã hóa có thể mã hóa giá trị của `scaling_list_dc_coef_minus16` là -16 để ra lệnh cho bộ giải mã sử dụng ma trận mặc định.

Các ví dụ liên quan tới hoạt động của bộ mã hóa để thông tin mã hóa về ma trận lượng tử hóa và tạo tín hiệu thông tin được mã hóa đã được mô tả. Dưới đây, các ví dụ trong đó, bộ giải mã giải mã thông tin về ma trận lượng tử hóa và thu được ma trận lượng tử hóa được mô tả bằng cách sử dụng các ví dụ của các bảng nêu trên.

Fig.5 là sơ đồ minh họa sơ lược một ví dụ về việc vận hành của bộ giải mã để giải mã thông tin về ma trận lượng tử hóa và thực hiện giải mã bằng cách sử dụng thông tin được giải mã.

Đề cập tới Fig.5, bộ giải mã giải mã thông tin về kích thước của đơn vị biến đổi và xác định kích thước của đơn vị biến đổi dựa trên thông tin được giải mã ở bước S510.

Bộ giải mã thực hiện việc giải mã entropy trên thông tin về kích thước của đơn vị biến đổi từ dòng bit nhận được. Bộ giải mã có thể giải mã thông tin về kích thước của đơn vị biến đổi từ bộ tham số trong dòng bit.

Ví dụ, bộ giải mã có thể giải mã thông tin về kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi từ dòng bit.

Như trong ví dụ trên bảng 1, bộ giải mã có thể giải mã kích thước tối thiểu theo hướng nằm ngang hoặc thẳng đứng của đơn vị biến đổi bình phương mà hàm `Log2` được áp dụng bằng cách sử dụng thành phần cú pháp `'log2_min_transform_block_size_minus2'`

trong dòng bit. Hơn nữa, bộ giải mã có thể giải mã giá trị chênh lệch giữa kích thước tối đa theo hướng nằm ngang hoặc thẳng đứng của đơn vị biến đổi bình phương mà hàm Log2 được áp dụng và kích thước tối thiểu theo hướng nằm ngang hoặc thẳng đứng của đơn vị biến đổi bình phương bằng cách sử dụng thành phần cú pháp ‘log2_diff_max_min_transform_block_size’ trong dòng bit.

Bộ giải mã có thể xác định kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi được giải mã. Ở đây, kích thước tối đa của đơn vị biến đổi có thể được xác định bởi giá trị chênh lệch được giải mã giữa kích thước tối đa và giá trị tối thiểu và kích thước tối thiểu được giải mã.

Ví dụ, bộ giải mã có thể tính toán Log2MinTrafoSize bằng cách thêm 2 vào log2_min_transform_block_size_minus2 được giải mã và xác định giá trị, được tính toán sử dụng $1 < \text{Log2MinTrafoSize}$, làm kích thước tối thiểu theo hướng nằm ngang hoặc thẳng đứng của đơn vị biến đổi bình phương. Bộ giải mã có thể tính toán Log2MaxTrafoSize dựa trên giá trị của log2_diff_max_min_transform_block_size được giải mã và giá trị thu được bằng cách thêm 2 vào log2_min_transform_block_size_minus2 được giải mã và xác định giá trị, được tính toán sử dụng $1 < \text{Log2MaxTrafoSize}$, làm kích thước tối đa theo hướng nằm ngang hoặc thẳng đứng của đơn vị biến đổi bình phương.

Ở đây, kích thước tối thiểu của đơn vị biến đổi có thể đồng nghĩa với việc giá trị được tính toán sử dụng Log2MinTrafoSize hoặc $1 < \text{Log2MinTrafoSize}$, và kích thước tối đa của đơn vị biến đổi có thể đồng nghĩa với việc giá trị được tính toán sử dụng Log2MaxTrafoSize hoặc $1 < \text{Log2MaxTrafoSize}$.

Bộ giải mã giải mã thông tin về ma trận lượng tử hóa ở bước S520. Bộ giải mã có thể giải mã thông tin về ma trận lượng tử hóa, bao gồm một hoặc nhiều thành phần trong số (1) thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa, (2) thông tin về việc ma trận lượng tử hóa có mặt hay không, (3) thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, (4) thông tin về loại phương pháp dự đoán và giải mã ma trận lượng tử hóa, (5) thông tin về ma trận lượng tử hóa tham chiếu ID, và (6) thông tin về giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá

trị hệ số của ma trận lượng tử hóa được giải mã. Ở đây, thông tin về ma trận lượng tử hóa có thể phụ thuộc vào kích thước của đơn vị biến đổi.

Đầu tiên, bộ giải mã có thể xác định xem ma trận lượng tử hóa có được sử dụng hay chưa bằng cách giải mã thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa từ bộ tham số. Như trong ví dụ trên bảng 2, bộ giải mã có thể giải mã `scaling_list_enabled_flag`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa, từ bộ tham số chuỗi. Ở đây, khi giá trị của `scaling_list_enabled_flag` là 1, thì bộ giải mã có thể sử dụng ma trận lượng tử hóa trong việc giải lượng tử hóa/việc định cỡ của hệ số biến đổi cho toàn bộ chuỗi. Khi giá trị của `scaling_list_enabled_flag` là 0, thì bộ giải mã có thể không sử dụng ma trận lượng tử hóa trong việc giải lượng tử hóa/định cỡ của hệ số biến đổi.

Bộ giải mã có thể xác định xem ma trận lượng tử hóa có mặt hay không bằng cách giải mã thông tin về việc ma trận lượng tử hóa có mặt hay không từ bộ tham số. Như trong ví dụ trên bảng 3, bộ giải mã có thể giải mã `aps_scaling_list_data_present_flag`, nghĩa là, thông tin về việc ma trận lượng tử hóa có mặt hay không, từ bộ tham số tương thích. Ví dụ, khi giá trị của `aps_scaling_list_data_present_flag` là 1, điều này có nghĩa là ma trận lượng tử hóa có mặt trong bộ tham số tương thích. Khi giá trị của `aps_scaling_list_data_present_flag` là 0, điều này có nghĩa là ma trận lượng tử hóa không có mặt trong bộ tham số tương thích. Nếu `scaling_list_enabled_flag` là 1 và `aps_scaling_list_data_present_flag` là 0, nó có thể đồng nghĩa với việc ma trận mặc định được sử dụng khi giải lượng tử hóa được thực hiện. Hơn nữa, các ví dụ trong đó, thông tin về việc ma trận lượng tử hóa có mặt hay không được giải mã từ bộ tham số tương thích đã được minh họa, nhưng sáng chế không bị giới hạn ở đó. Bộ giải mã có thể giải mã thông tin về việc ma trận lượng tử hóa có mặt hay không từ bộ tham số khác.

Bộ giải mã có thể xác định xem ma trận lượng tử hóa đã được giải mã hay chưa và xem ma trận mặc định đã được sử dụng hay chưa bằng cách giải mã thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số. Như trong ví dụ trên bảng 4, bộ giải mã có thể giải mã `use_default_scaling_list_flag`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được

giải mã hay chưa và việc liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số tương thích. Ví dụ, khi giá trị của `use_default_scaling_list_flag` là 1, ma trận lượng tử hóa không được giải mã và các giá trị hệ số của tất cả ma trận lượng tử hóa có thể được xác định là giống với các giá trị hệ số của ma trận lượng tử hóa mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của `use_default_scaling_list_flag` là 0, thì ma trận lượng tử hóa được giải mã và ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã có thể không được sử dụng.

Bộ giải mã có thể xác định một hoặc nhiều việc như có hay không giải mã ma trận lượng tử hóa, có hay không sử dụng ma trận mặc định, và có hay không thực hiện việc giải mã dự đoán bằng cách sử dụng `SizeID` và `MatrixID`.

Theo các ví dụ trên các bảng 5 và 6, giá trị của `SizeID` có thể xác định ma trận lượng tử hóa theo kích thước của đơn vị biến đổi hoặc kích thước của ma trận lượng tử hóa bằng cách sử dụng các bảng, và giá trị của `MatrixID` có thể xác định chế độ mã hóa trong đó, ma trận lượng tử hóa được sử dụng và loại ma trận lượng tử hóa tương ứng với thành phần màu.

Trong khi đó, ma trận lượng tử hóa mặc định có thể được chỉ báo sử dụng các bảng 7 và 8.

Nếu kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi khả dụng không được xem xét, thì ma trận lượng tử hóa cho đơn vị biến đổi có tất cả các kích thước phải được giải mã. Trong trường hợp này, hiệu quả mã hóa bị giảm sút, và mức độ phức tạp của việc tính toán gia tăng.

Theo sáng chế, thông tin về ma trận lượng tử hóa có thể được giải mã bằng cách lấy kích thước của đơn vị biến đổi để xem xét.

Theo ví dụ trên bảng 9, `SizeID` tương ứng với kích thước của mỗi đơn vị biến đổi bị giới hạn dựa trên kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi, từ các mẫu thông tin về đơn vị biến đổi, và một hoặc nhiều trong số giải mã ma trận lượng tử hóa, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và giải mã thông tin về loại dự đoán và phương pháp giải mã được thực hiện dựa trên `SizeID` bị giới hạn.

Như trong ví dụ trên bảng 9, SizeID có thể bị giới hạn dựa trên kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi, từ các mẫu thông tin về đơn vị biến đổi, và một hoặc nhiều thành phần trong số thông tin về việc liệu ma trận lượng tử hóa hoặc ma trận mặc định đã được sử dụng hay chưa và thông tin về loại phương pháp giải mã dự đoán có thể được giải mã liên quan tới chỉ các đơn vị biến đổi có kích thước cụ thể.

Ví dụ, nếu giá trị của Log2MinTrafoSize xác định kích thước tối thiểu của đơn vị biến đổi là 3 và giá trị của Log2MaxTrafoSize xác định kích thước tối đa của đơn vị biến đổi là 4, thì bộ giải mã có thể thực hiện một hoặc nhiều trong số giải mã ma trận lượng tử hóa tương ứng với các đơn vị biến đổi từ kích thước 8x8 tới kích thước 16x16, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và việc giải mã thông tin về loại phương pháp giải mã dự đoán.

Theo ví dụ trên bảng 9, use_default_scaling_list_flag có thể không được giải mã.

Hơn nữa, không giống như trong ví dụ trên, bộ giải mã có thể giới hạn SizeID dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu của đơn vị biến đổi và thực hiện một hoặc nhiều trong số giải mã ma trận lượng tử hóa, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và giải mã thông tin về loại phương pháp giải mã dự đoán dựa trên SizeID giới hạn.

Như trong ví dụ trên bảng 10, bộ giải mã có thể giới hạn SizeID dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu của đơn vị biến đổi và thực hiện một hoặc nhiều trong số giải mã ma trận lượng tử hóa, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và giải mã thông tin về loại phương pháp giải mã dự đoán trên chỉ các đơn vị biến đổi có kích thước cụ thể (tức là, chỉ các kích thước cụ thể của đơn vị biến đổi).

Ví dụ, nếu giá trị của Log2MinTrafoSize xác định kích thước tối thiểu của đơn vị biến đổi là 3 và giá trị của Log2MaxTrafoSize xác định kích thước tối đa của đơn vị biến đổi là 4, thì bộ giải mã có thể thực hiện một hoặc nhiều trong số giải mã ma trận lượng tử hóa tương ứng với các đơn vị biến đổi từ kích thước 8x8 tới kích thước 16x16, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và việc giải mã thông tin về loại phương pháp giải mã dự đoán.

Ở đây, giá trị chênh lệch giữa `Log2MaxTrafoSize` và `Log2MinTrafoSize` là giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu của đơn vị biến đổi và có thể được định rõ bởi `log2_diff_max_min_transform_block_size`. Hơn nữa, `Log2MinTrafoSize-2` là giống như `log2_min_transform_block_size_minus2`.

Theo ví dụ trên bảng 10, `use_default_scaling_list_flag` có thể không được giải mã.

Trong khi đó, nếu ma trận mặc định và ma trận không phải là ma trận mặc định không được trộn và được sử dụng theo kích thước của mỗi khối biến đổi hoặc loại ma trận lượng tử hóa nằm trong chuỗi, hình ảnh hoặc phiên, mức độ tự do bị giảm sút khi bộ mã hóa chọn ma trận lượng tử hóa. Ví dụ, để sử dụng ma trận mặc định cho kích thước biến đổi cụ thể nằm trong các phiên và để sử dụng ma trận không phải là ma trận mặc định cho kích thước biến đổi cụ thể khác nằm trong phiên, hiệu quả mã hóa có thể bị giảm sút do ma trận mặc định phải được mã hóa và được truyền.

Theo sáng chế, ma trận mặc định và ma trận không phải là ma trận mặc định có thể được trộn và được sử dụng theo kích thước của mỗi khối biến đổi hoặc loại ma trận lượng tử hóa nằm trong chuỗi, hình ảnh hoặc phiên.

Ví dụ, bộ giải mã có thể giải mã thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số dựa trên `SizeID`. Như trong ví dụ trên bảng 11, bộ giải mã có thể giải mã `sid_use_default_scaling_list_flag[SizeID][MatrixID]`, nghĩa là, thông tin xác định xem ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số tương thích dựa trên `SizeID`. Ví dụ, khi giá trị của `sid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1, thì bộ giải mã không giải mã ma trận lượng tử hóa tương ứng với `SizeID`, và các giá trị hệ số của ma trận lượng tử hóa tương ứng với `SizeID` có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của `sid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0, thì bộ giải mã giải mã ma trận lượng tử hóa tương ứng với `SizeID`, nhưng không sử dụng ma trận mặc định, được xác định trong bộ mã hóa và/hoặc bộ giải mã, là ma trận lượng tử hóa tương ứng với `SizeID`.

Trong khi đó, trong ví dụ trên bảng 11, `use_default_scaling_list_flag` có thể không

được giải mã.

Hơn nữa, bộ giải mã có thể giải mã thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và việc liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số dựa trên MatrixID thay cho SizeID.

Như trong ví dụ trên bảng 12, bộ giải mã có thể giải mã `mid_use_default_scaling_list_flag[SizeID][MatrixID]`, nghĩa là, thông tin xác định liệu ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số tương thích dựa trên dựa trên MatrixID. Ví dụ, khi giá trị của `mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1, thì bộ giải mã không giải mã ma trận lượng tử hóa tương ứng với MatrixID, và các giá trị hệ số của ma trận lượng tử hóa tương ứng với MatrixID có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của `mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0, thì bộ giải mã giải mã ma trận lượng tử hóa tương ứng với MatrixID, nhưng không sử dụng ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã là ma trận lượng tử hóa tương ứng với MatrixID.

Theo ví dụ trên bảng 12, `use_default_scaling_list_flag` có thể không được giải mã.

Hơn nữa, bộ giải mã có thể giải mã thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số dựa trên SizeID và MatrixID bằng cách lấy cả SizeID và MatrixID để xem xét, thay cho chỉ lấy SizeID hoặc chỉ MatrixID để xem xét.

Như trong ví dụ trên bảng 13, bộ giải mã có thể giải mã `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]`, nghĩa là, thông tin xác định xem ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số tương thích dựa trên SizeID và MatrixID. Ví dụ, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1, thì bộ giải mã không giải mã ma trận lượng tử hóa tương ứng với SizeID và MatrixID, và các giá trị hệ số của ma trận lượng tử hóa tương ứng với SizeID và MatrixID có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0, thì bộ giải mã

giải mã ma trận lượng tử hóa tương ứng với SizeID và MatrixID và không sử dụng ma trận mặc định, được xác định trong bộ mã hóa và/hoặc bộ giải mã, là ma trận lượng tử hóa tương ứng với SizeID và MatrixID.

Theo ví dụ trên bảng 13, `use_default_scaling_list_flag` có thể không được giải mã.

Trong khi đó, như trong ví dụ trên bảng 14, bộ giải mã giới hạn SizeID theo kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi và thực hiện một hoặc nhiều trong số giải mã ma trận lượng tử hóa, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và giải mã thông tin về loại phương pháp giải mã dự đoán dựa trên SizeID bị giới hạn. Hơn nữa, bộ giải mã có thể giải mã thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số dựa trên SizeID và MatrixID.

Theo ví dụ trên bảng 14, `use_default_scaling_list_flag` có thể không được giải mã.

Như trong ví dụ trên bảng 15, bộ giải mã có thể giới hạn SizeID dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu của đơn vị biến đổi, từ các mẫu thông tin về đơn vị biến đổi, và thực hiện một hoặc nhiều trong số giải mã ma trận lượng tử hóa, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và giải mã thông tin về loại phương pháp giải mã dự đoán dựa trên SizeID bị giới hạn. Ví dụ, giá trị chênh lệch giữa `Log2MaxTrafoSize` và `Log2MinTrafoSize` là giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu của đơn vị biến đổi và trùng với `log2_diff_max_min_transform_block_size`, và `Log2MinTrafoSize-2` trùng với `log2_min_transform_block_size_minus2`. Hơn nữa, bộ giải mã có thể giải mã thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số dựa trên SizeID và MatrixID.

Theo ví dụ trên bảng 15, `use_default_scaling_list_flag` có thể không được giải mã.

Hơn nữa, như trong ví dụ trên bảng 16, bộ giải mã có thể xác định xem thông tin về ma trận lượng tử hóa có mặt hay không trong bộ tham số được giải mã hoặc có hay không cập nhật ma trận lượng tử hóa dựa trên `scaling_list_update_flag[SizeID][MatrixID]` nằm trong bộ tham số. Ví dụ, `scaling_list_update_flag[SizeID][MatrixID]` có giá trị là 1 chỉ báo

rằng thông tin về ma trận lượng tử hóa được định rõ bởi SizeID và MatrixID có mặt trong bộ tham số được giải mã hoặc chỉ báo rằng thông tin về ma trận lượng tử hóa, tương ứng với SizeID và MatrixID và được giải mã từ trước, nên được cập nhật vào thông tin về ma trận lượng tử hóa tương ứng với SizeID và MatrixID nằm trong bộ tham số được giải mã. Ở đây, để cập nhật thông tin về ma trận lượng tử hóa có thể đồng nghĩa với việc thông tin về ma trận lượng tử hóa đã được giải mã từ trước được thay thế bằng thông tin về ma trận lượng tử hóa nằm trong bộ tham số được giải mã. Hơn nữa, `scaling_list_update_flag[SizeID][MatrixID]` có giá trị là 0 chỉ báo rằng ma trận lượng tử hóa tương ứng với SizeID và MatrixID không có mặt trong bộ tham số được giải mã hoặc thông tin về ma trận lượng tử hóa đã được giải mã từ trước là không được cập nhật. Ví dụ, khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì bộ giải mã không biết rằng thông tin về ma trận lượng tử hóa nào nên được sử dụng trong giải lượng tử hóa cho hệ số của ma trận lượng tử hóa được định rõ bởi SizeID và MatrixID do ma trận lượng tử hóa tương ứng với SizeID và MatrixID không có mặt trong bộ tham số và thông tin về ma trận lượng tử hóa không được giải mã. Do đó, khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì nó có thể được lệnh là ma trận mặc định được sử dụng làm ma trận lượng tử hóa tương ứng với SizeID và MatrixID hoặc ma trận lượng tử hóa không được sử dụng do thông tin về ma trận lượng tử hóa tương ứng với SizeID và MatrixID không có mặt trong bộ tham số được giải mã.

Ở đây, `scaling_list_update_flag[SizeID][MatrixID]` được giải mã dựa trên SizeID và MatrixID tương ứng với kích thước của mỗi đơn vị biến đổi. Do đó, khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 1, thì bộ giải mã có thể thực hiện một hoặc nhiều trong số giải mã ma trận lượng tử hóa, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và giải mã thông tin về loại phương pháp giải mã dự đoán. Khi giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì bộ giải mã không thực hiện một hoặc nhiều việc giải mã trong số giải mã ma trận lượng tử hóa, giải mã thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, và giải mã thông tin về loại phương pháp giải mã dự đoán. Nghĩa là, bộ giải mã có thể không giải mã ma trận lượng tử hóa không cần thiết dựa trên chỉ báo của `scaling_list_update_flag[SizeID][MatrixID]`.

Trong khi đó, có nhược điểm là mức độ tự do trong việc chọn ma trận lượng tử hóa

là thấp do ma trận mặc định và ma trận không phải là ma trận mặc định không được trộn và được sử dụng theo kích thước của mỗi khối biến đổi hoặc loại ma trận lượng tử hóa nằm trong chuỗi, hình ảnh hoặc phiên bản cách chỉ sử dụng `scaling_list_update_flag[SizeID][MatrixID]`. Do đó, bộ giải mã có thể giải mã `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số dựa trên `SizeID` và `MatrixID`. Ví dụ, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1, thì bộ giải mã không giải mã ma trận lượng tử hóa được định rõ bởi `SizeID` và `MatrixID`, và các giá trị hệ số của ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0, thì bộ giải mã giải mã ma trận lượng tử hóa được định rõ bởi `SizeID` và `MatrixID` và không sử dụng ma trận mặc định, được xác định trong bộ mã hóa và/hoặc bộ giải mã, là ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID`.

Theo ví dụ trên bảng 16, `use_default_scaling_list_flag` có thể không được giải mã.

Hơn nữa, như trong ví dụ trên bảng 17, bộ giải mã có thể giải mã thông tin về việc liệu ma trận lượng tử hóa đã được giải mã hay chưa và việc liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số dựa trên `SizeID` và `MatrixID`. Hơn nữa, `scaling_list_update_flag[SizeID][MatrixID]` liên quan tới việc cập nhật vào trong thông tin về ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` có thể được sử dụng trong tham số được giải mã.

Ví dụ, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 0 và giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì bộ giải mã giải mã ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID`, không cập nhật ma trận lượng tử hóa, tương ứng với `SizeID` và `MatrixID` được giải mã từ trước, vào ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` trong tham số được giải mã, và sử dụng ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` đã được giải mã từ trước.

Hơn nữa, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]`

là 0 và giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 1, thì bộ giải mã giải mã ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` và cập nhật ma trận lượng tử hóa, tương ứng với `SizeID` và `MatrixID` và được giải mã từ trước, vào ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` trong tham số được giải mã.

Hơn nữa, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1 và giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 0, thì bộ giải mã không giải mã ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID`, xác định các giá trị hệ số của ma trận lượng tử hóa là giống nhau làm các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã, không cập nhật ma trận lượng tử hóa, tương ứng với `SizeID` và `MatrixID` và được giải mã từ trước, vào ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` trong tham số được giải mã, và sử dụng ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` đã được giải mã từ trước.

Hơn nữa, khi giá trị của `sid_mid_use_default_scaling_list_flag[SizeID][MatrixID]` là 1 và giá trị của `scaling_list_update_flag[SizeID][MatrixID]` là 1, thì bộ giải mã không giải mã ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID`, xác định các giá trị hệ số của ma trận lượng tử hóa là giống nhau làm các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã, và cập nhật ma trận lượng tử hóa, tương ứng với `SizeID` và `MatrixID` và được giải mã từ trước, vào ma trận lượng tử hóa tương ứng với `SizeID` và `MatrixID` trong tham số được giải mã.

Theo ví dụ trên bảng 17, `use_default_scaling_list_flag` có thể không được giải mã.

Hơn nữa, bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa từ bộ tham số và có thể xác định loại phương pháp giải mã dự đoán cho ma trận lượng tử hóa dựa trên thông tin được giải mã. Như trong ví dụ trên bảng 18, bộ giải mã có thể giải mã `pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số tương thích. Ví dụ, khi giá trị của `pred_mode_flag` là 1, thì bộ giải mã có thể giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và điều biến mã xung vi phân ngược (Inverse-Differential Pulse Code Modulation - DPCM). Khi giá trị của `pred_mode_flag` là 0, thì bộ giải mã xác định các giá trị hệ số của ma trận lượng tử hóa là giống nhau làm các giá trị hệ số của ma trận lượng tử

hóa đã được giải mã từ trước. Ở đây, các giá trị hệ số của ma trận lượng tử hóa và các giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước có thể là các giá trị nằm trong các ma trận lượng tử hóa khác, và ma trận lượng tử hóa được giải mã từ trước có thể là ma trận lượng tử hóa tham chiếu.

Nếu giá trị của `pred_mode_flag` là 0, thì bộ giải mã có thể giải mã ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã từ bộ tham số. Như trong ví dụ trên bảng 18, bộ giải mã có thể giải mã `pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã, từ bộ tham số tương thích. Ở đây, bộ giải mã có thể xác định `RefMatrixID` chỉ báo của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã bằng cách sử dụng `pred_matrix_id_delta` và Phương trình 16.

Phương trình 16

$$\text{RefMatrixID} = \text{MatrixID} - (1 + \text{pred_matrix_id_delta})$$

Nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa là phương pháp giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước có thể là giá trị hệ số nằm trong ma trận lượng tử hóa được giải mã. Như trong ví dụ trên bảng 19, bộ giải mã có thể giải mã `delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số tương thích.

Trong khi đó, ma trận lượng tử hóa mặc định và ma trận lượng tử hóa không mặc định có thể được trộn và được sử dụng nằm trong chuỗi, hình ảnh hoặc phiên bản bằng cách sử dụng phương pháp sau, và việc nhận ma trận lượng tử hóa không cần thiết có thể bị cản lại.

Ví dụ, nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa là phương pháp xác định ma trận lượng tử hóa giống với ma trận lượng tử hóa đã có sẵn trong bộ giải mã và đã được giải mã từ trước (`pred_mode_flag=0`), thì bộ giải mã có thể giải mã thông tin về

việc liệu ma trận mặc định đã được sử dụng hay chưa bằng cách sử dụng ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã từ bộ tham số.

Như trong ví dụ trên bảng 18, bộ giải mã có thể giải mã `pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã, từ bộ tham số. Ở đây, bộ giải mã có thể xác định `RefMatrixID`, nghĩa là, thông tin xác định ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, bằng cách sử dụng `pred_matrix_id_delta` và Phương trình 17.

Phương trình 17

$$\text{RefMatrixID} = \text{MatrixID} - \text{pred_matrix_id_delta}$$

Ví dụ, khi giá trị của `RefMatrixID` trùng với giá trị của `MatrixID`, các giá trị hệ số của ma trận lượng tử hóa tương ứng với `SizeID` và `RefMatrixID` được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ở đây, ma trận mặc định nghĩa là ma trận mặc định tương ứng với `SizeID` và `RefMatrixID`. Hơn nữa, khi giá trị của `pred_matrix_id_delta` là 0, thì giá trị của `RefMatrixID` trở nên trùng với giá trị của `MatrixID`. Nếu giá trị của `RefMatrixID` không trùng với giá trị của `MatrixID`, thì ma trận lượng tử hóa tương ứng với `RefMatrixID` được xác định là ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và các giá trị hệ số của ma trận lượng tử hóa được giải mã được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu.

Nếu kích thước của ma trận lượng tử hóa tương ứng với `SizeID` bao gồm trong kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi khả dụng cho bộ giải mã, thì ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và liệu ma trận mặc định đã được sử dụng hay chưa có thể được xác định bằng cách sử dụng phương pháp nêu trên. Nếu kích thước của ma trận lượng tử hóa tương ứng với `SizeID` không bao gồm trong kích thước tối thiểu và kích thước tối đa của đơn vị biến đổi khả dụng cho bộ giải mã, thì ma trận lượng tử hóa tương ứng với `SizeID` có thể không được xác định là giống như ma trận mặc định. Quy trình xác định nêu trên cho `SizeID` có thể được thực hiện nếu một hoặc nhiều trong số giải mã ma trận lượng tử hóa, giải mã thông tin về việc liệu ma trận mặc

định đã được sử dụng hay chưa, và giải mã thông tin về loại phương pháp giải mã dự đoán được thực hiện dựa trên giá trị chênh lệch giữa kích thước tối đa và giá trị tối thiểu của đơn vị biến đổi, từ các mẫu thông tin về đơn vị biến đổi.

Sau khi xác định xem ma trận lượng tử hóa đã được sử dụng hay chưa, bộ giải mã có thể giải mã thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa từ bộ tham số.

Như trong ví dụ trên bảng 20, bộ giải mã có thể giải mã `scaling_list_enable_flag`, nghĩa là, thông tin về việc liệu ma trận lượng tử hóa đã được sử dụng hay chưa, từ bộ tham số. Ở đây, khi giá trị của `scaling_list_enable_flag` là 1, thì bộ giải mã có thể sử dụng ma trận lượng tử hóa, như ma trận mặc định hoặc ma trận không phải là ma trận mặc định, trong giải lượng tử hóa. Khi giá trị của `scaling_list_enable_flag` là 0, thì bộ giải mã không sử dụng ma trận lượng tử hóa hoặc có thể sử dụng ma trận lượng tử hóa có tất cả các giá trị hệ số giống nhau trong giải lượng tử hóa. Ở đây, tất cả các giá trị hệ số có thể là 16.

Hơn nữa, nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa là phương pháp giải mã ma trận lượng tử hóa theo DPCM ngược và các phương pháp mã hóa Golomb hàm mũ, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước có thể là giá trị hệ số nằm trong ma trận lượng tử hóa được giải mã. Nghĩa là, giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước và giá trị hệ số của ma trận lượng tử hóa được giải mã có thể là các giá trị nằm trong cùng một ma trận lượng tử hóa.

Như trong ví dụ trên bảng 21, bộ giải mã có thể giải mã `delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số. Như trong ví dụ trên bảng 21, nếu giá trị hệ số ‘`nextcoef`’ của ma trận lượng tử hóa là (1) trùng với một giá trị cụ thể và là (2) giá trị thứ nhất của ma trận lượng tử hóa, thì các giá trị hệ số của ma trận lượng tử hóa tương ứng có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã bằng cách sử dụng Phương trình 18.

Phương trình 18

$$\text{nextcoef} = (\text{nextcoef} + \text{delta_coef} + 256) \% 256$$

Nghĩa là, nếu giá trị hệ số ‘nextcoef’ của ma trận lượng tử hóa là (1) trùng với một giá trị cụ thể và là (2) giá trị thứ nhất của ma trận lượng tử hóa, thì bộ giải mã có thể sử dụng ma trận lượng tử hóa tương ứng làm ma trận mặc định. Ở đây, giá trị cụ thể có thể là 0. Hơn nữa, ma trận mặc định có thể đồng nghĩa với việc ma trận mặc định được định rõ bởi SizeID và MatrixID. Do đó, nếu giá trị hệ số ‘nextcoef’ của ma trận lượng tử hóa trùng với 0 và là giá trị thứ nhất của ma trận lượng tử hóa, thì bộ giải mã có thể dùng giá trị của giá trị chênh lệch giữa ma trận lượng tử hóa tương ứng và ma trận lượng tử hóa.

Trong khi đó, nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa là phương pháp giải mã ma trận lượng tử hóa theo DPCM ngược và các phương pháp mã hóa Golomb hàm mũ, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước có thể là giá trị hệ số nằm trong ma trận lượng tử hóa được giải mã. Nghĩa là, giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước và giá trị hệ số của ma trận lượng tử hóa được giải mã có thể là các giá trị nằm trong cùng một ma trận lượng tử hóa.

Như trong ví dụ trên bảng 22, bộ giải mã có thể giải mã delta_coef, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số. Ví dụ, như trong ví dụ trên bảng 22, nếu giá trị hệ số ‘nextcoef’ của ma trận lượng tử hóa được tính toán sử dụng Phương trình 18, nghĩa là, $(\text{nextcoef} + \text{delta_coef} + 256) \% 256$, trùng với một giá trị cụ thể và là giá trị thứ nhất của ma trận lượng tử hóa, thì bộ giải mã có thể xác định các giá trị hệ số của ma trận lượng tử hóa tương ứng là giống nhau làm các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Nghĩa là, nếu nextcoef là (1) trùng với một giá trị cụ thể và là (2) giá trị thứ nhất của ma trận lượng tử hóa, thì bộ giải mã có thể sử dụng ma trận lượng tử hóa tương ứng làm ma trận mặc định. Ở đây, giá trị cụ thể có thể là 0, và giá trị hệ số của ma trận lượng tử hóa được tính toán sử dụng $(\text{nextcoef} + \text{delta_coef} + 256) \% 256$ có thể là giá trị hệ số của ma trận lượng tử hóa có kích thước 4x4 hoặc 8x8.

Hơn nữa, giá trị thứ nhất của ma trận lượng tử hóa có thể là giá trị sử dụng `scaling_list_dc_coef_minus8`, và giá trị cụ thể có thể là giá trị tương ứng với `scaling_list_dc_coef_minus8 + 8`.

`scaling_list_dc_coef_minus8` có thể đồng nghĩa với việc giá trị thứ nhất của ma trận lượng tử hóa có kích thước 16x16 hoặc 32x32, có thể đồng nghĩa với việc giá trị hệ số của ma trận DC.

Khi giá trị của `SizeID` là 2, thì `scaling_list_dc_coef_minus8[SizeID-2][MatrixID]` có thể tương ứng với giá trị hệ số ma trận DC có kích thước 16x16. Ở đây, 16x16 có thể là kích thước của khối biến đổi tương ứng với ma trận lượng tử hóa. Khi giá trị của `SizeID` là 3, thì `scaling_list_dc_coef_minus8[SizeID-2][MatrixID]` có thể tương ứng với giá trị hệ số ma trận DC trong ma trận lượng tử hóa có kích thước 32x32. Ở đây, kích thước 32x32 có thể là kích thước của khối biến đổi tương ứng với ma trận lượng tử hóa. Trong mỗi trường hợp, ma trận mặc định có thể đồng nghĩa với việc ma trận mặc định tương ứng với `SizeID` và `MatrixID`. Do đó, nếu giá trị hệ số 'nextcoef' của ma trận lượng tử hóa trùng với 0 và là giá trị thứ nhất của ma trận lượng tử hóa, thì bộ giải mã có thể dừng giải mã của giá trị chênh lệch giữa (tức là, chênh lệch giữa các giá trị hệ số) giữa ma trận lượng tử hóa tương ứng và ma trận lượng tử hóa.

Bộ giải mã theo cách khác có thể thực hiện để xem có giải mã `scaling_list_dc_coef_minus8` hay không, nghĩa là, giá trị hệ số của ma trận DC, và giải mã của thông tin về việc liệu ma trận mặc định đã được sử dụng theo kích thước của ma trận lượng tử hóa hay chưa hoặc kích thước biến đổi bằng cách sử dụng phương pháp.

Trong khi đó, phương pháp mã hóa/giải mã ma trận lượng tử hóa để xác định xem có hay không sử dụng ma trận mặc định bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa là có nhược điểm ở chỗ mức độ phức tạp trong quy trình mã hóa/giải mã giá trị hệ số của ma trận lượng tử hóa tăng lên. Trái lại, trong sáng chế, trong việc mã hóa/giải mã hình ảnh, việc có hay không sử dụng ma trận mặc định có thể được xác định bằng cách sử dụng ma trận lượng tử hóa tham chiếu ID. Do đó, mức độ phức tạp của việc tính toán trong quy trình mã hóa/giải mã ma trận lượng tử hóa có thể được giảm bớt.

Đầu tiên, bộ giải mã có thể giải mã thông tin chỉ báo liệu ma trận lượng tử hóa có

mặt hay không từ bộ tham số. Như trong ví dụ trên bảng 23, bộ giải mã có thể giải mã `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo liệu ma trận lượng tử hóa có mặt hoặc không trong dòng bit, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_present_flag` là 0, điều này có nghĩa là ma trận lượng tử hóa không có mặt và ma trận lượng tử hóa được xác định là ma trận lượng tử hóa mặc định. Khi giá trị của `scaling_list_present_flag` là 1, điều này có nghĩa ma trận lượng tử hóa được mã hóa có mặt.

Bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa từ bộ tham số và có thể xác định loại phương pháp giải mã dự đoán cho ma trận lượng tử hóa dựa trên thông tin được giải mã. Ở đây, bộ tham số mà từ đó thông tin về phương pháp giải mã dự đoán được giải mã có thể là bộ tham số tương thích. Như trong ví dụ trên bảng 23, bộ giải mã có thể giải mã `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_pred_mode_flag` là 1, bộ giải mã dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa bằng cách giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét. Khi giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu hoặc xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận mặc định. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa cụ thể sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể giải mã ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số. Ở đây, bộ tham số mà ma trận lượng tử hóa tham chiếu ID và thông tin được giải mã có thể là bộ tham số tương thích.

Nghĩa là, như trong ví dụ trên bảng 23, bộ giải mã có thể giải mã `scaling_list_pred_matrix_id_delta`, nghĩa là, thông tin xác định ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định

đã được sử dụng hay chưa, từ bộ tham số. Ở đây, bộ giải mã có thể xác định RefMatrixID chỉ báo của ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định của ma trận lượng tử hóa được giải mã bằng cách sử dụng scaling_list_pred_matrix_id_delta và Phương trình 19.

Phương trình 19

$$\text{RefMatrixID} = \text{MatrixID} - \text{scaling_list_pred_matrix_id_delta}$$

Nếu giá trị của RefMatrixID trùng với giá trị của MatrixID, thì các giá trị hệ số của ma trận lượng tử hóa được giải mã, tương ứng với SizeID và MatrixID, có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ở đây, ma trận mặc định có thể là ma trận mặc định tương ứng với SizeID và MatrixID. Đề cập tới Phương trình 19, khi giá trị của scaling_list_pred_matrix_id_delta là 0, điều này có nghĩa là giá trị của RefMatrixID trùng với giá trị của MatrixID.

Nếu giá trị của RefMatrixID không trùng với giá trị của MatrixID, thì bộ giải mã có thể xác định ma trận lượng tử hóa tương ứng với RefMatrixID làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Để xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với RefMatrixID, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và việc sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng. Ở đây, giá trị của scaling_list_pred_matrix_id_delta có thể là giá trị nguyên dương.

Nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa là phương pháp dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây,

bộ tham số mà giá trị chênh lệch được giải mã bởi bộ giải mã có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 24, nếu kích thước của ma trận lượng tử hóa được giải mã là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ giải mã có thể giải mã `scaling_list_dc_coef_minus8`, nghĩa là, hệ số ma trận DC, từ bộ tham số. Hơn nữa, như trong ví dụ trên bảng 24, bộ giải mã có thể giải mã `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số. Trong các ví dụ trên các bảng 23 và 24, các ví dụ trong đó, thông tin về ma trận lượng tử hóa được giải mã từ bộ tham số tương thích được minh họa, nhưng sáng chế không bị giới hạn ở đó. Bộ giải mã có thể giải mã thông tin về ma trận lượng tử hóa từ bộ tham số khác (tức là, bộ tham số bao gồm ít nhất một trong số bộ tham số chuỗi và bộ tham số hình ảnh).

Như được mô tả ở trên trong ví dụ của bộ mã hóa, theo các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống, hiệu quả mã hóa bị giảm sút do thông tin không cần thiết được mã hóa/được giải mã khi dự đoán ma trận lượng tử hóa. Trong sáng chế, tuy nhiên, hiệu quả mã hóa có thể được cải thiện khi mã hóa/giải mã ma trận lượng tử hóa do mã hóa/giải mã ma trận lượng tử hóa có thể được thực hiện theo cách khác phụ thuộc vào việc liệu ma trận lượng tử hóa tham chiếu có mặt hay không.

Cụ thể hơn, bộ giải mã có thể giải mã thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không từ bộ tham số. Như trong ví dụ trên bảng 25, bộ giải mã có thể giải mã `scaling_list_present_flag`, nghĩa là, thông tin về việc ma trận lượng tử hóa có mặt hay không trong dòng bit, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_present_flag` là 0, điều này có nghĩa là ma trận lượng tử hóa không có mặt và tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định. Khi giá trị của `scaling_list_present_flag` là 1, điều này có nghĩa ma trận lượng tử hóa được mã hóa có mặt.

Hơn nữa, khi giá trị của MatrixID lớn hơn 0, thì bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa từ bộ tham số và xác định loại phương pháp giải mã dự đoán cho ma trận lượng tử hóa dựa trên thông tin được giải mã. Ở đây, bộ tham số mà thông tin về phương pháp giải mã dự đoán được giải mã, có thể là bộ

tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 25, khi giá trị của MatrixID lớn hơn 0, thì bộ giải mã có thể giải mã `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số. Trong ví dụ trên bảng 25, khi giá trị của `scaling_list_pred_mode_flag` là 1, thì bộ giải mã có thể dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa bằng cách giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét. Khi giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Hơn nữa, trong ví dụ trên bảng 25, khi giá trị của MatrixID là 0, thì giá trị của `scaling_list_pred_mode_flag` trở thành TRUE. Do đó, bộ giải mã không giải mã `scaling_list_pred_mode_flag` và có thể giải mã ma trận lượng tử hóa bằng cách quét ma trận lượng tử hóa theo các phương pháp DPCM ngược và mã hóa Golomb hàm mũ.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0 và giá trị của MatrixID lớn hơn 0, thì bộ giải mã có thể giải mã ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà ma trận lượng tử hóa tham chiếu ID được giải mã có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 25, bộ giải mã có thể giải mã `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã, từ bộ tham số khi giá trị của MatrixID lớn hơn 0. Ở đây, RefMatrixID chỉ báo ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã có thể được xác định bằng cách sử dụng `scaling_list_pred_matrix_id_delta` và Phương trình 20.

Phương trình 20

$$\text{RefMatrixID} = \text{MatrixID} - (1 + \text{scaling_list_pred_matrix_id_delta})$$

Trong ví dụ trên bảng 25, MatrixID có giá trị là 0 chỉ báo ma trận lượng tử hóa thứ nhất cho mỗi SizeID. Ma trận lượng tử hóa có thể được dự đoán từ chỉ ma trận lượng tử hóa được giải mã từ trước có cùng SizeID, và việc dự đoán ma trận lượng tử hóa sử dụng phương pháp, như sao chép ma trận, không thể được thực hiện trên ma trận lượng tử hóa thứ nhất cho mỗi SizeID do ma trận lượng tử hóa tham chiếu có cùng giá trị SizeID không có mặt. Do đó, khi giá trị của MatrixID lớn hơn 0, thì bộ giải mã có thể giải mã scaling_list_pred_matrix_id_delta, nghĩa là, ma trận lượng tử hóa tham chiếu ID, xác định ma trận lượng tử hóa tương ứng với RefMatrixID làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã dựa trên ma trận lượng tử hóa được giải mã tham chiếu ID, và xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Để xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với RefMatrixID, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa được giải mã là phương pháp dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được giải mã có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 26, khi kích thước của ma trận lượng tử hóa được giải mã là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ giải mã có thể giải mã scaling_list_dc_coef_minus8, nghĩa là, hệ số ma trận DC, từ bộ tham số. Hơn nữa, như trong ví dụ trên bảng 26, bộ giải mã có thể giải mã scaling_list_delta_coef, nghĩa là, giá trị

chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số.

Hơn nữa, bộ giải mã có thể xác định xem ma trận mặc định đã được sử dụng hay chưa bằng cách sử dụng `scaling_list_delta_coef` được sử dụng để tính toán `scaling_list_dc_coef_minus8` hoặc `nextcoef`. Ví dụ, giá trị của `scaling_list_dc_coef_minus8` được giải mã là -8, thì bộ giải mã có thể xác định ma trận lượng tử hóa tương ứng làm ma trận mặc định. Khi giá trị của giá trị `nextcoef` thứ nhất được tính toán bằng cách giải mã `scaling_list_delta_coef` là 0, thì bộ giải mã có thể xác định ma trận lượng tử hóa tương ứng làm ma trận mặc định.

Như được mô tả ở trên trong ví dụ của bộ mã hóa, phương pháp mã hóa/giải mã ma trận lượng tử hóa để xác định xem ma trận mặc định đã được sử dụng hay chưa bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa có nhược điểm là nó tăng độ phức tạp trong quy trình mã hóa/giải mã giá trị hệ số của ma trận lượng tử hóa. Hơn nữa, hiệu quả mã hóa bị giảm sút do thông tin không cần thiết được mã hóa/được giải mã khi dự đoán ma trận lượng tử hóa. Tuy nhiên, trong sáng chế, trong mã hóa/giải mã hình ảnh, mức độ phức tạp của việc tính toán trong mã hóa/giải mã ma trận lượng tử hóa có thể được giảm bớt do việc liệu ma trận mặc định đã được sử dụng hay chưa có thể được xác định dựa trên ma trận lượng tử hóa tham chiếu ID, và hiệu quả mã hóa in mã hóa/giải mã ma trận lượng tử hóa có thể được cải thiện do ma trận lượng tử hóa được mã hóa/được giải mã theo cách khác phụ thuộc vào việc liệu ma trận lượng tử hóa tham chiếu có mặt hay không.

Cụ thể hơn, đầu tiên, bộ giải mã có thể giải mã thông tin chỉ báo việc liệu ma trận lượng tử hóa có mặt từ bộ tham số hay không. Như trong ví dụ trên bảng 27, bộ giải mã có thể giải mã `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc liệu ma trận lượng tử hóa có mặt trong dòng bit hay không, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_present_flag` là 0, nó chỉ báo rằng ma trận lượng tử hóa không có mặt và do đó tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định. Khi giá trị của `scaling_list_present_flag` là 1, nó chỉ báo rằng ma trận lượng tử hóa được mã hóa.

Hơn nữa, bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã

ma trận lượng tử hóa từ bộ tham số và có thể xác định loại phương pháp giải mã dự đoán cho ma trận lượng tử hóa dựa trên thông tin được giải mã. Ở đây, bộ tham số mà thông tin về phương pháp giải mã dự đoán được giải mã có thể là bộ tham số tương thích.

Cụ thể hơn, như trong ví dụ trên bảng 27, bộ giải mã có thể giải mã `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_pred_mode_flag` là 1, thì bộ giải mã có thể dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa bằng cách giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét. Khi giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu hoặc xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận mặc định. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa cụ thể sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0 và giá trị của `MatrixID` lớn hơn 0, thì bộ giải mã có thể giải mã ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số. Ở đây, bộ tham số mà ma trận lượng tử hóa tham chiếu ID và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa được giải mã có thể là bộ tham số tương thích.

Trong trường hợp này, như trong ví dụ trên bảng 27, khi giá trị của `scaling_list_pred_mode_flag` là 0 và giá trị của `MatrixID` lớn hơn 0, thì bộ giải mã có thể giải mã `scaling_list_pred_matrix_id_delta`, nghĩa là, chỉ báo ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số. Ở đây, `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định của ma trận lượng tử hóa được giải mã có thể được xác định bằng cách sử dụng `scaling_list_pred_matrix_id_delta` và Phương trình 21.

Phương trình 21

$$\text{RefMatrixID} = \text{MatrixID} - \text{scaling_list_pred_matrix_id_delta}$$

Nếu giá trị của RefMatrixID trùng với giá trị của MatrixID, thì các giá trị hệ số của ma trận lượng tử hóa được giải mã, tương ứng với SizeID và MatrixID, được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ở đây, ma trận mặc định nghĩa là ma trận mặc định tương ứng với SizeID và MatrixID. Theo Phương trình 21, khi giá trị của scaling_list_pred_matrix_id_delta là 0, điều này có nghĩa là giá trị của RefMatrixID trùng với giá trị của MatrixID.

Nếu giá trị của RefMatrixID không trùng với giá trị của MatrixID, thì ma trận lượng tử hóa tương ứng với RefMatrixID được xác định là ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và các giá trị hệ số của ma trận lượng tử hóa được giải mã được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Để xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với RefMatrixID, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Ở đây, khi giá trị của scaling_list_pred_mode_flag là 0, điều này chỉ báo rằng phương pháp dự đoán và giải mã ma trận lượng tử hóa là phương pháp xác định ma trận lượng tử hóa trùng với ma trận lượng tử hóa được giải mã từ trước. Trong trường hợp này, ma trận lượng tử hóa có thể được dự đoán từ ma trận lượng tử hóa được giải mã từ trước có cùng SizeID.

Khi giá trị của MatrixID là 0, thì ma trận lượng tử hóa thứ nhất được chỉ báo cho mỗi SizeID. Nếu giá trị của scaling_list_pred_mode_flag là 0 và giá trị của MatrixID là 0, thì việc dự đoán ma trận lượng tử hóa sử dụng phương pháp, như sao chép ma trận, không thể được thực hiện trên ma trận lượng tử hóa thứ nhất cho mỗi SizeID do ma trận lượng tử hóa tham chiếu có cùng giá trị SizeID không có mặt. Trong trường hợp này, bộ giải mã không giải mã scaling_list_pred_matrix_id_delta và suy ra giá trị của

scaling_list_pred_matrix_id_delta là 0. Khi giá trị của scaling_list_pred_matrix_id_delta là 0, thì các giá trị hệ số của ma trận lượng tử hóa được giải mã, tương ứng với SizeID và MatrixID, được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã do giá trị của RefMatrixID trùng với giá trị của MatrixID. Ở đây, ma trận mặc định nghĩa là ma trận mặc định tương ứng với SizeID và MatrixID.

Nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa được giải mã là phương pháp dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được giải mã có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 28, nếu kích thước của ma trận lượng tử hóa được giải mã là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ giải mã có thể giải mã scaling_list_dc_coef_minus8, nghĩa là, hệ số ma trận DC, từ bộ tham số. Như trong ví dụ trên bảng 28, bộ giải mã có thể giải mã scaling_list_delta_coef, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số.

Như được mô tả ở trên trong ví dụ của bộ mã hóa, trong các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống, khi gửi ma trận lượng tử hóa, tổng số các hệ số và hệ số ma trận DC nằm trong ma trận được mã hóa/được giải mã. Trong trường hợp này, sự cải thiện hiệu quả mã hóa bị giới hạn do hệ số ma trận DC không được dự đoán và được mã hóa/được giải mã. Trong sáng chế, hệ số ma trận DC nằm trong ma trận lượng tử hóa có thể được dự đoán và được mã hóa/được giải mã, và do đó hiệu quả mã hóa có thể được cải thiện. Ví dụ, trong các ví dụ trên các bảng 29 và 30, hiệu quả mã hóa có thể được cải thiện do hệ số ma trận DC không được dự đoán từ hàng số 8, nhưng các hệ số AC lân cận được dự đoán bằng cách sử dụng mối tương quan cao giữa các hệ số lân cận. Hơn nữa, trong các ví dụ trên các bảng 29 và 30, không gian nhớ để lưu trữ các hệ số ma trận DC có

thể được giảm bớt do chuỗi mã hóa/giải mã ma trận lượng tử hóa được tạo trùng với chuỗi phục hồi của ma trận lượng tử hóa. Hơn nữa, trong các ví dụ trên các bảng 29 và 30, liệu ma trận mặc định đã được sử dụng hay chưa có thể được chỉ báo bởi thành phần cú pháp ‘scaling_list_delta_coef’ mà không sử dụng nhiều thành phần cú pháp.

Cụ thể hơn, bộ giải mã có thể giải mã thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không từ bộ tham số. Ví dụ, như trong ví dụ trên bảng 29, bộ giải mã có thể giải mã scaling_list_present_flag, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, từ bộ tham số. Ở đây, khi giá trị của scaling_list_present_flag là 0, điều này chỉ báo rằng ma trận lượng tử hóa không có mặt và do đó tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định. Khi giá trị của scaling_list_present_flag là 1, điều này chỉ báo rằng ma trận lượng tử hóa được mã hóa có mặt.

Bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa từ bộ tham số và có thể xác định loại phương pháp giải mã dự đoán cho ma trận lượng tử hóa dựa trên thông tin được giải mã. Ở đây, bộ tham số mà thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa được giải mã có thể là bộ tham số tương thích.

Cụ thể hơn, như trong ví dụ trên bảng 29, bộ giải mã có thể giải mã scaling_list_pred_mode_flag, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số. Ví dụ, khi giá trị của scaling_list_pred_mode_flag là 1, thì bộ giải mã có thể dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét. Khi giá trị của scaling_list_pred_mode_flag là 0, thì bộ giải mã có thể xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu giá trị của scaling_list_pred_mode_flag là 0, thì bộ giải mã có thể giải mã ma

trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà ma trận lượng tử hóa tham chiếu ID được giải mã có thể là bộ tham số tương thích.

Cụ thể hơn, như trong ví dụ trên bảng 29, bộ giải mã có thể giải mã `scaling_list_pred_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã, từ bộ tham số. Ở đây, `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã có thể được xác định bằng cách sử dụng `scaling_list_pred_matrix_id_delta` và Phương trình 22.

Phương trình 22

$$\text{RefMatrixID} = \text{MatrixID} - (1 + \text{scaling_list_pred_matrix_id_delta})$$

Bộ giải mã có thể xác định ma trận lượng tử hóa, tương ứng với `RefMatrixID`, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Để xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với `RefMatrixID`, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa được giải mã là phương pháp dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được giải mã có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 30, bộ giải mã có thể giải mã

scaling_list_delta_coef, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số. Ở đây, bộ giải mã có thể xác định xem ma trận mặc định đã được sử dụng hay chưa bằng cách sử dụng scaling_list_delta_coef được sử dụng để tính toán nextcoef. Nghĩa là, khi giá trị của nextcoef thứ nhất được tính toán bằng cách giải mã scaling_list_delta_coef là 0, thì bộ giải mã có thể xác định ma trận lượng tử hóa tương ứng làm ma trận mặc định.

Hơn nữa, như trong ví dụ trên bảng 30, bộ giải mã có thể giải mã scaling_list_dc_coef_res, nghĩa là, giá trị chênh lệch giữa các giá trị hệ số của các ma trận lượng tử hóa tương ứng với hệ số ma trận DC, từ bộ tham số. Ở đây, scaling_list_dc_coef_res có thể được giải mã từ (useDefaultScalingMatrixFlag=0) khi kích thước của ma trận lượng tử hóa được giải mã là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3) và ma trận mặc định không được sử dụng.

Liên quan tới ma trận lượng tử hóa có kích thước 16x16 hoặc 32x32 mà hệ số ma trận DC của nó được giải mã riêng, hệ số ma trận DC có thể được tính toán bằng cách sử dụng tổng của giá trị của scaling_list_dc_coef_res và ma trận hệ số có mặt tại vị trí DC theo Phương trình 23.

Phương trình 23

$$\text{ScalingFactor}[2][\text{MatrixID}][0][0] = \text{scaling_list_dc_coef_res}[0][\text{MatrixID}] + \text{ScalingFactor}[2][\text{MatrixID}][0][0] \text{ với } \text{MatrixID}=0..5$$

$$\text{ScalingFactor}[3][\text{MatrixID}][0][0] = \text{scaling_list_dc_coef_res}[1][\text{MatrixID}] + \text{ScalingFactor}[3][\text{MatrixID}][0][0] \text{ với } \text{MatrixID}=0..1$$

Trong phương trình 23, ScalingFactor[2] đề cập tới ma trận lượng tử hóa có kích thước 16x16, và ScalingFactor[3] đề cập tới ma trận lượng tử hóa có kích thước 32x32. Hơn nữa, ScalingFactor[2][MatrixID][0][0] đề cập tới hệ số ma trận DC nằm trong ma trận lượng tử hóa có kích thước 16x16, tương ứng với MatrixID. ScalingFactor[3][MatrixID][0][0] đề cập tới hệ số ma trận DC nằm trong ma trận lượng tử hóa có kích thước 32x32, tương ứng với MatrixID.

Trong khi đó, phương pháp mã hóa/giải mã ma trận lượng tử hóa để xác định xem ma trận mặc định đã được sử dụng hay chưa bằng cách sử dụng giá trị hệ số của ma trận lượng tử hóa có nhược điểm là nó tăng độ phức tạp trong quy trình mã hóa/giải mã giá trị hệ số của ma trận lượng tử hóa. Hơn nữa, tất cả các hệ số và hệ số ma trận DC nằm trong ma trận lượng tử hóa được mã hóa/được giải mã khi gửi ma trận lượng tử hóa. Trong trường hợp này, sự cải thiện hiệu quả mã hóa bị giới hạn do hệ số ma trận DC không được dự đoán và được mã hóa/được giải mã.

Theo sáng chế, trong khi mã hóa/giải mã hình ảnh, mức độ phức tạp khi mã hóa/giải mã ma trận lượng tử hóa có thể được giảm bớt bằng cách xác định xem ma trận mặc định đã được sử dụng hay chưa dựa trên ma trận lượng tử hóa tham chiếu ID. Hơn nữa, hiệu quả mã hóa có thể được cải thiện bằng cách dự đoán và mã hóa/giải mã hệ số ma trận DC nằm trong ma trận lượng tử hóa. Ví dụ, trong các ví dụ trên các bảng 31 và 32, hiệu quả mã hóa có thể được cải thiện bằng cách dự đoán hệ số ma trận DC nằm trong ma trận lượng tử hóa từ các hệ số AC lân cận dựa trên tương quan cao giữa các hệ số lân cận mà không dự đoán hệ số ma trận DC từ hằng số 8. Hơn nữa, trong các ví dụ trên các bảng 31 và 32, không gian nhớ để lưu trữ các hệ số ma trận DC có thể được giảm bớt do chuỗi mã hóa/giải mã ma trận lượng tử hóa có thể được tạo ra trùng với chuỗi phục hồi của ma trận lượng tử hóa. Hơn nữa, trong ví dụ của các bảng 31 và 32, liệu ma trận mặc định đã được sử dụng hay chưa có thể được chỉ báo bởi thành phần cú pháp ‘`scaling_list_delta_coef`’ mà không sử dụng nhiều thành phần cú pháp.

Cụ thể hơn, bộ giải mã có thể giải mã thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không từ bộ tham số. Như trong ví dụ trên bảng 31, bộ giải mã có thể giải mã `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, từ bộ tham số. Ở đây, khi giá trị của `scaling_list_present_flag` là 0, điều này chỉ báo rằng ma trận lượng tử hóa không có mặt và do đó tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định. Khi giá trị của `scaling_list_present_flag` là 1, điều này chỉ báo rằng ma trận lượng tử hóa được mã hóa có mặt.

Bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã ma trận

lượng tử hóa từ bộ tham số và có thể xác định loại phương pháp giải mã dự đoán cho ma trận lượng tử hóa dựa trên thông tin được giải mã. Ở đây, bộ tham số mà thông tin về phương pháp giải mã dự đoán được giải mã có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 31, bộ giải mã có thể giải mã `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số. Trong ví dụ trên bảng 31, khi giá trị của `scaling_list_pred_mode_flag` là 1, thì bộ giải mã có thể dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa bằng cách giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét. Khi giá trị của `scaling_list_pred_mode_flag` là 0, thì các giá trị hệ số của ma trận lượng tử hóa được giải mã có thể được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu hoặc các giá trị hệ số của ma trận lượng tử hóa được giải mã có thể được xác định trùng với các giá trị hệ số của ma trận mặc định. Ở đây, để xác định các giá trị sao cho chúng có các giá trị giống nhau có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa cụ thể sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể giải mã ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số. Ở đây, bộ tham số được giải mã có thể là bộ tham số tương thích.

Cụ thể hơn, như trong ví dụ trên bảng 31, bộ giải mã có thể giải mã `scaling_list_pred_matrix_id_delta`, nghĩa là, thông tin về ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số. Ở đây, `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định của ma trận lượng tử hóa được giải mã được xác định bằng cách sử dụng `scaling_list_pred_matrix_id_delta` và Phương trình 24.

Phương trình 24

$$\text{RefMatrixID} = \text{MatrixID} - \text{scaling_list_pred_matrix_id_delta}$$

Nếu giá trị của RefMatrixID trùng với giá trị của MatrixID, thì các giá trị hệ số của ma trận lượng tử hóa được giải mã, tương ứng với SizeID và MatrixID, có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ở đây, ma trận mặc định đề cập tới ma trận mặc định tương ứng với SizeID và MatrixID. Hơn nữa, theo phương trình 24, khi giá trị của `scaling_list_pred_matrix_id_delta` là 0, điều này có nghĩa là giá trị của RefMatrixID trùng với giá trị của MatrixID. Khi giá trị của RefMatrixID không trùng với giá trị của MatrixID, ma trận lượng tử hóa tương ứng với RefMatrixID có thể được xác định là ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và các giá trị hệ số của ma trận lượng tử hóa được giải mã có thể được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Để xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với RefMatrixID, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa được giải mã là phương pháp giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược để dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa thông qua việc quét, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được giải mã có thể là bộ tham số tương thích.

Hơn nữa, như trong ví dụ trên bảng 32, bộ giải mã có thể giải mã `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số. Như trong ví dụ trên bảng 32, bộ giải mã có thể giải mã `scaling_list_dc_coef_res`, nghĩa là, giá trị chênh lệch giữa các giá trị hệ số của các ma trận lượng tử hóa tương ứng với hệ số ma trận DC, từ bộ tham số. Ở đây, `scaling_list_dc_coef_res` có thể được giải mã từ (`useDefaultScalingMatrixFlag=0`) nếu kích

thước của ma trận lượng tử hóa được giải mã là ma trận lượng tử hóa 16x16 (SizeID=2) hoặc 32x32 (SizeID=3) và ma trận mặc định không được sử dụng.

Liên quan tới ma trận lượng tử hóa có kích thước 16x16 hoặc ma trận lượng tử hóa có kích thước 32x32 mà hệ số ma trận DC của nó được giải mã riêng, hệ số ma trận DC có thể được tính toán bằng cách sử dụng tổng của giá trị của `scaling_list_dc_coef_res` và hệ số của ma trận có mặt tại vị trí DC như theo phương trình 25.

Phương trình 25

$$\text{ScalingFactor}[2][\text{MatrixID}][0][0] = \text{scaling_list_dc_coef_res}[0][\text{MatrixID}] + \text{ScalingFactor}[2][\text{MatrixID}][0][0] \text{ với } \text{MatrixID}=0..5$$

$$\text{ScalingFactor}[3][\text{MatrixID}][0][0] = \text{scaling_list_dc_coef_res}[1][\text{MatrixID}] + \text{ScalingFactor}[3][\text{MatrixID}][0][0] \text{ với } \text{MatrixID}=0..1$$

Theo phương trình 25, `ScalingFactor[2]` đề cập tới ma trận lượng tử hóa có kích thước 16x16, và `ScalingFactor[3]` đề cập tới ma trận lượng tử hóa có kích thước 32x32. Hơn nữa, `ScalingFactor[2][MatrixID][0][0]` đề cập tới hệ số ma trận DC nằm trong ma trận lượng tử hóa có kích thước 16x16 tương ứng với `MatrixID` tương ứng, và `ScalingFactor[3][MatrixID][0][0]` đề cập tới hệ số ma trận DC nằm trong ma trận lượng tử hóa có kích thước 32x32 tương ứng với `MatrixID` tương ứng.

Như được mô tả ở trên trong ví dụ có liên quan tới bộ mã hóa, trong các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống, ma trận lượng tử hóa được sao chép bằng cách sử dụng kích thước của ma trận lượng tử hóa khi việc lượng tử hóa và giải lượng tử hóa được thực hiện không sử dụng kích thước của ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện. Do đó, hiệu quả mã hóa ma trận lượng tử hóa bị giới hạn do ma trận lượng tử hóa phải được sao chép từ số các ma trận lượng tử hóa bị giới hạn. Trong sáng chế, tuy nhiên, hiệu quả mã hóa có thể được cải thiện và mức độ tự do trong khi dự đoán ma trận lượng tử hóa có thể được tăng do ma trận lượng tử hóa có thể được dự đoán từ ma trận lượng tử hóa có cùng kích thước với ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện.

Cụ thể hơn, bộ giải mã có thể giải mã thông tin chỉ báo việc ma trận lượng tử hóa có

mặt hay không từ bộ tham số. Như trong các ví dụ trên các bảng 33 và 34, bộ giải mã có thể giải mã `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_present_flag` là 0, điều này chỉ báo rằng ma trận lượng tử hóa không có mặt và do đó tất cả các ma trận lượng tử hóa được xác định là các ma trận lượng tử hóa mặc định. Khi giá trị của `scaling_list_present_flag` là 1, điều này chỉ báo rằng ma trận lượng tử hóa được mã hóa có mặt.

Hơn nữa, bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa từ bộ tham số và có thể xác định loại phương pháp giải mã dự đoán cho ma trận lượng tử hóa dựa trên thông tin được giải mã. Ở đây, bộ tham số mà thông tin về phương pháp giải mã dự đoán được giải mã có thể là bộ tham số tương thích.

Cụ thể hơn, như trong các ví dụ trên các bảng 33 và 34, bộ giải mã có thể giải mã `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_pred_mode_flag` là 1, thì bộ giải mã có thể dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa bằng cách giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét. Khi giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể giải mã ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, thông tin ID (identifier - mã nhận dạng) về ma trận lượng tử hóa tham chiếu có thể bao gồm một hoặc nhiều trong số kích thước của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và ma trận lượng tử hóa tham chiếu. Hơn nữa, bộ tham số mà ma trận lượng tử hóa tham chiếu ID được giải mã có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 33, bộ giải mã có thể giải mã

scaling_list_pred_size_matrix_id_delta, nghĩa là, thông tin ID về ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã, từ bộ tham số. Ở đây, RefSizeID, nghĩa là, kích thước của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã, và RefMatrixID chỉ báo ma trận lượng tử hóa tham chiếu có thể được xác định bằng cách sử dụng scaling_list_pred_size_matrix_id_delta và Phương trình 26.

Phương trình 26

$$\text{RefSizeID} = \text{SizeID} - (\text{scaling_list_pred_size_matrix_id_delta} / 6)$$

$$\text{RefMatrixID} = \text{scaling_list_pred_size_matrix_id_delta} \% 6$$

Hơn nữa, như trong ví dụ trên bảng 34, bộ giải mã có thể giải mã scaling_list_pred_size_id_delta và scaling_list_pred_size_matrix_id_delta, nghĩa là, thông tin ID về ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã, từ bộ tham số. Ở đây, RefSizeID có thể được xác định bằng cách sử dụng giá trị của scaling_list_pred_size_id_delta và Phương trình 27, và RefMatrixID chỉ báo ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã có thể được xác định bằng cách sử dụng scaling_list_pred_matrix_id_delta và Phương trình 28.

Phương trình 27

$$\text{RefSizeID} = \text{SizeID} - \text{scaling_list_pred_size_id_delta}$$

Phương trình 28

$$\text{RefMatrixID} = \text{MatrixID} - \text{scaling_list_pred_matrix_id_delta}$$

Tương tự, các giá trị hệ số của ma trận lượng tử hóa được giải mã có thể được xác định từ các giá trị hệ số của ma trận lượng tử hóa tham chiếu mà SizeID của nó là bằng với RefSizeID và MatrixID là bằng với RefMatrixID. Nghĩa là, bộ giải mã có thể sao chép ma trận lượng tử hóa tham chiếu sang ma trận lượng tử hóa được giải mã. Để xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với RefSizeID và RefMatrixID, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các

giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Qua các ví dụ trên các bảng 33 và 34, ma trận lượng tử hóa có thể được dự đoán từ ma trận lượng tử hóa có cùng SizeID, và ma trận lượng tử hóa có thể cũng được dự đoán từ ma trận lượng tử hóa có cùng một kích thước ma trận khi mã hóa/giải mã được thực hiện, nhưng có SizeID khác.

Hơn nữa, trong các ví dụ trên các bảng 33 và 34, giá trị của `scaling_list_pred_size_matrix_id_delta`, giá trị của `scaling_list_pred_size_id_delta`, và giá trị của `scaling_list_pred_matrix_id_delta` có thể được giới hạn ở giá trị có giới hạn cụ thể. Ví dụ, `scaling_list_pred_size_matrix_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 17, `scaling_list_pred_size_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 2, và `scaling_list_pred_matrix_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 5.

Hơn nữa, trong các ví dụ trên các bảng 33 và 34, bộ giải mã có thể không dự đoán ma trận lượng tử hóa được giải mã từ ma trận lượng tử hóa có kích thước lớn hơn ma trận lượng tử hóa được giải mã.

Hơn nữa, trong các ví dụ trên các bảng 33 và 34, khi dự đoán ma trận lượng tử hóa được giải mã từ ma trận lượng tử hóa có kích thước 8x8, bộ giải mã có thể dự đoán giá trị tại vị trí tương ứng bằng cách xác định giá trị, tương ứng với vị trí của hệ số ma trận DC nằm trong ma trận lượng tử hóa có kích thước 8x8, làm hệ số ma trận DC. Hơn nữa, khi dự đoán ma trận lượng tử hóa được giải mã từ ma trận lượng tử hóa có kích thước 16x16 hoặc 32x32, thì bộ giải mã có thể cũng dự đoán hệ số ma trận DC.

Trong khi đó, nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa là phương pháp giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét để dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được giải mã có thể là bộ tham số tương thích.

Cụ thể hơn, như trong ví dụ trên bảng 35, nếu kích thước của ma trận lượng tử hóa được giải mã là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ giải mã có thể giải mã `scaling_list_dc_coef_minus8`, nghĩa là, hệ số ma trận DC, từ bộ tham số.

Hơn nữa, như trong ví dụ trên bảng 35, bộ giải mã có thể giải mã `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số.

Hơn nữa, bộ giải mã có thể xác định xem ma trận mặc định đã được sử dụng hay chưa bằng cách sử dụng `scaling_list_delta_coef` được sử dụng để tính toán `scaling_list_dc_coef_minus8` hoặc `nextcoef`. Nghĩa là, nếu giá trị của `scaling_list_dc_coef_minus8` được giải mã là -8, thì ma trận lượng tử hóa tương ứng có thể được xác định làm ma trận mặc định. Nếu giá trị của `nextcoef` thứ nhất được tính toán bằng cách giải mã `scaling_list_delta_coef` là 0, thì ma trận lượng tử hóa tương ứng có thể được xác định làm ma trận mặc định.

Như được mô tả ở trên trong ví dụ về bộ mã hóa, trong các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống, mức độ phức tạp trong quy trình mã hóa/giải mã giá trị hệ số của ma trận lượng tử hóa tăng lên do việc liệu ma trận mặc định đã được sử dụng hay chưa được xác định sử dụng giá trị hệ số của ma trận lượng tử hóa. Hơn nữa, ma trận lượng tử hóa được sao chép từ số lượng giới hạn các ma trận lượng tử hóa do ma trận lượng tử hóa được sao chép sử dụng kích thước của ma trận lượng tử hóa khi việc lượng tử hóa và giải lượng tử hóa được thực hiện chứ không phải kích thước của ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện. Tuy nhiên, trong sáng chế, mức độ phức tạp của việc tính toán khi mã hóa/giải mã ma trận lượng tử hóa có thể được giảm bớt bằng cách xác định xem ma trận mặc định đã được sử dụng hay chưa dựa trên ma trận lượng tử hóa tham chiếu ID. Hơn nữa, trong sáng chế, hiệu quả mã hóa có thể được cải thiện và mức độ tự do trong dự đoán ma trận lượng tử hóa có thể được tăng lên bằng cách dự đoán ma trận lượng tử hóa từ ma trận lượng tử hóa có cùng kích thước với ma trận lượng tử hóa khi mã hóa/giải mã được thực hiện.

Cụ thể hơn, bộ giải mã có thể giải mã thông tin chỉ báo việc ma trận lượng tử hóa có

mặt hay không từ bộ tham số. Như trong các ví dụ trong các bảng 36 và 37, bộ giải mã có thể giải mã `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_present_flag` là 0, điều này chỉ báo rằng ma trận lượng tử hóa không có mặt và do đó tất cả các ma trận lượng tử hóa có thể được xác định là các ma trận lượng tử hóa mặc định. Khi giá trị của `scaling_list_present_flag` là 1, điều này chỉ báo rằng ma trận lượng tử hóa được mã hóa có mặt.

Bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa từ bộ tham số và có thể xác định loại phương pháp dự đoán và giải mã ma trận lượng tử hóa dựa trên thông tin được giải mã. Ở đây, bộ tham số được giải mã có thể là bộ tham số tương thích.

Như trong các ví dụ trên các bảng 36 và 37, bộ giải mã có thể giải mã `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số. Ở đây, khi giá trị của `scaling_list_pred_mode_flag` là 1, thì bộ giải mã có thể giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét để dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa. Khi giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu hoặc xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận mặc định. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa cụ thể sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể giải mã ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa từ bộ tham số. Ở đây, thông tin ID (identifier - mã nhận dạng) về ma trận lượng tử hóa tham chiếu có thể bao gồm một hoặc nhiều trong số kích thước của ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa

được giải mã và ma trận lượng tử hóa tham chiếu. Hơn nữa, bộ tham số được giải mã có thể là bộ tham số tương thích.

Ví dụ, như trong ví dụ trên bảng 36, bộ giải mã có thể giải mã `scaling_list_pred_size_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số. Ở đây, `RefSizeID` và `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu có thể được xác định bằng cách sử dụng `scaling_list_pred_size_matrix_id_delta` và Phương trình 29.

Phương trình 29

$$\text{RefSizeID} = \text{SizeID} - (\text{scaling_list_pred_size_matrix_id_delta} / 6)$$

$$\text{RefMatrixID} = \text{scaling_list_pred_size_matrix_id_delta} \% 6$$

Hơn nữa, như trong ví dụ trên bảng 37, bộ giải mã có thể giải mã `scaling_list_pred_size_id_delta` và `scaling_list_pred_size_matrix_id_delta`, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã và thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, từ bộ tham số. Ở đây, `RefSizeID` có thể được xác định sử dụng giá trị của `scaling_list_pred_size_id_delta` và Phương trình 30, và `RefMatrixID` chỉ báo ma trận lượng tử hóa tham chiếu hoặc ma trận mặc định của ma trận lượng tử hóa được giải mã có thể được xác định sử dụng giá trị của `scaling_list_pred_matrix_id_delta` và Phương trình 31.

Phương trình 30

$$\text{RefSizeID} = \text{SizeID} - \text{scaling_list_pred_size_id_delta}$$

Phương trình 31

$$\text{RefMatrixID} = \text{MatrixID} - \text{scaling_list_pred_matrix_id_delta}$$

Nếu giá trị của `RefMatrixID` trùng với giá trị của `MatrixID`, thì các giá trị hệ số của ma trận lượng tử hóa được giải mã, tương ứng với `SizeID` và `MatrixID`, có thể được xác định trùng với các giá trị hệ số của ma trận mặc định được xác định trong bộ mã hóa và/hoặc bộ giải mã. Ở đây, ma trận mặc định đề cập tới ma trận mặc định tương ứng với `SizeID` và

MatrixID. Hơn nữa, theo phương trình 31, khi giá trị của `scaling_list_pred_matrix_id_delta` là 0, điều này có nghĩa là giá trị của RefMatrixID là bằng với giá trị của MatrixID.

Nếu giá trị của RefMatrixID không trùng với giá trị của MatrixID, ma trận lượng tử hóa tương ứng với RefSizeID và RefMatrixID được xác định là ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và các giá trị hệ số của ma trận lượng tử hóa được giải mã có thể được xác định trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Để xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã trùng với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với RefSizeID và RefMatrixID, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Theo ví dụ trên bảng 36 hoặc 37, ma trận lượng tử hóa có thể được dự đoán từ ma trận lượng tử hóa có cùng SizeID, và ma trận lượng tử hóa cùng có thể được dự đoán từ ma trận lượng tử hóa có cùng một kích thước ma trận khi mã hóa/giải mã được thực hiện, nhưng có SizeID khác.

Hơn nữa, trong ví dụ trên bảng 36 hoặc 37, giá trị của `scaling_list_pred_size_matrix_id_delta`, giá trị của `scaling_list_pred_size_id_delta`, và giá trị của `scaling_list_pred_matrix_id_delta` có thể bị giới hạn tới giới hạn có thể đo được. Ví dụ, `scaling_list_pred_size_matrix_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 17, `scaling_list_pred_size_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 2, và `scaling_list_pred_matrix_id_delta` có thể được giới hạn ở giá trị nằm trong giới hạn từ 0 tới 5.

Hơn nữa, trong ví dụ trên bảng 36 hoặc 37, bộ giải mã có thể không dự đoán ma trận lượng tử hóa từ ma trận lượng tử hóa có kích thước lớn hơn ma trận lượng tử hóa được giải mã.

Hơn nữa, khi thực hiện việc dự đoán từ ma trận lượng tử hóa có kích thước 8x8, bộ giải mã có thể dự đoán giá trị tại vị trí tương ứng bằng cách xác định giá trị, tương ứng với vị trí của hệ số ma trận DC nằm trong ma trận lượng tử hóa có kích thước 8x8, làm hệ số

ma trận DC. Hơn nữa, khi thực hiện việc dự đoán từ ma trận lượng tử hóa có kích thước 16x16 hoặc ma trận lượng tử hóa có kích thước 32x32, bộ giải mã cũng có thể dự đoán hệ số ma trận DC.

Trong khi đó, nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa là phương pháp giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét để dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được giải mã có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 38, nếu kích thước của ma trận lượng tử hóa được giải mã là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), bộ giải mã có thể giải mã `scaling_list_dc_coef_minus8`, nghĩa là, hệ số ma trận DC, từ bộ tham số.

Như trong ví dụ trên bảng 38, bộ giải mã có thể giải mã `scaling_list_delta_coef`, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số.

Như được mô tả ở trên trong ví dụ có liên quan tới bộ mã hóa, trong các phương pháp mã hóa/giải mã ma trận lượng tử hóa truyền thống, sự cải thiện hiệu quả mã hóa bị giới hạn do giá trị hệ số của ma trận lượng tử hóa được mã hóa bằng cách không lấy giá trị hệ số thường xuyên xuất hiện để xem xét khi mã hóa/giải mã hệ số thứ nhất nằm trong ma trận lượng tử hóa. Trong sáng chế, tuy nhiên, hiệu quả mã hóa có thể được cải thiện do hệ số thứ nhất nằm trong ma trận lượng tử hóa có thể được dự đoán và được mã hóa/được giải mã sử dụng giá trị hệ số thường xuyên xuất hiện. Nếu giá trị hệ số thứ nhất hoặc giá trị hệ số ma trận DC của ma trận mặc định được xác định là 16 và giá trị hệ số thứ nhất hoặc giá trị hệ số ma trận DC của ma trận không phải là ma trận mặc định được phân bố trên cơ sở 16, thì hiệu quả mã hóa có thể được cải thiện nếu giá trị hệ số thứ nhất hoặc giá trị hệ số ma trận DC nằm trong ma trận lượng tử hóa được mã hóa/được giải mã được mã hóa/được giải mã bằng cách dự đoán giá trị hệ số thứ nhất hoặc giá trị hệ số ma trận DC từ hằng số

16.

Cụ thể hơn, bộ giải mã có thể giải mã thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không từ bộ tham số. Như trong ví dụ trên bảng 39, bộ giải mã có thể giải mã `scaling_list_present_flag`, nghĩa là, thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong dòng bit, từ bộ tham số. Ví dụ, khi giá trị của `scaling_list_present_flag` là 0, điều này chỉ báo rằng ma trận lượng tử hóa không có mặt và do đó tất cả các ma trận lượng tử hóa có thể được xác định là các ma trận lượng tử hóa mặc định. Khi giá trị của `scaling_list_present_flag` là 1, điều này chỉ báo rằng ma trận lượng tử hóa được mã hóa có mặt.

Hơn nữa, bộ giải mã có thể giải mã thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa từ bộ tham số và có thể xác định loại phương pháp giải mã dự đoán cho ma trận lượng tử hóa dựa trên thông tin được giải mã. Ở đây, bộ tham số mà thông tin về phương pháp mã hóa dự đoán được giải mã có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 39, bộ giải mã có thể giải mã `scaling_list_pred_mode_flag`, nghĩa là, thông tin về phương pháp dự đoán và giải mã ma trận lượng tử hóa, từ bộ tham số. Ở đây, khi giá trị của `scaling_list_pred_mode_flag` là 1, thì bộ giải mã giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét để dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa. Khi giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Ở đây, để xác định các giá trị sao cho chúng có cùng các giá trị có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng.

Nếu giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể giải mã ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà ma trận lượng tử hóa tham chiếu ID được giải mã có thể là bộ tham số tương thích.

Hơn nữa, như trong ví dụ trên bảng 39, bộ giải mã có thể giải mã

scaling_list_pred_matrix_id_delta, nghĩa là, ma trận lượng tử hóa tham chiếu ID của ma trận lượng tử hóa được giải mã, từ bộ tham số. Ở đây, RefMatrixID chỉ báo ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã có thể được xác định bằng cách sử dụng scaling_list_pred_matrix_id_delta và Phương trình 32.

Phương trình 32

$$\text{RefMatrixID} = \text{MatrixID} - (1 + \text{scaling_list_pred_matrix_id_delta})$$

Bộ giải mã có thể xác định ma trận lượng tử hóa, tương ứng với RefMatrixID, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu. Để xác định các giá trị hệ số của ma trận lượng tử hóa được giải mã sao cho chúng có cùng các giá trị với các giá trị hệ số của ma trận lượng tử hóa tham chiếu có thể đồng nghĩa với việc phương pháp dự đoán ma trận lượng tử hóa xác định ma trận lượng tử hóa tham chiếu, tương ứng với RefMatrixID, làm ma trận lượng tử hóa tham chiếu của ma trận lượng tử hóa được giải mã và sao chép các giá trị hệ số của ma trận lượng tử hóa tham chiếu sang các giá trị hệ số của ma trận lượng tử hóa được giải mã được sử dụng. Ở đây, giá trị của scaling_list_pred_matrix_id_delta có thể là giá trị nguyên dương.

Nếu phương pháp dự đoán và giải mã ma trận lượng tử hóa được giải mã là phương pháp giải mã ma trận lượng tử hóa theo các phương pháp mã hóa Golomb hàm mũ và DPCM ngược thông qua việc quét để dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa, thì bộ giải mã có thể giải mã giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã từ bộ tham số. Ở đây, bộ tham số mà giá trị chênh lệch được giải mã có thể là bộ tham số tương thích.

Như trong ví dụ trên bảng 40, bộ giải mã có thể giải mã scaling_list_delta_coef, nghĩa là, giá trị chênh lệch giữa giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước nằm trong ma trận lượng tử hóa và giá trị hệ số của ma trận lượng tử hóa được giải mã, từ bộ tham số. Ở đây, bộ giải mã có thể đặt giá trị dự đoán cho giá trị hệ số thứ nhất thành 16 như trong nextcoef=16.

Hơn nữa, như trong ví dụ trên bảng 40, nếu kích thước của ma trận lượng tử hóa được giải mã là 16x16 (SizeID=2) hoặc 32x32 (SizeID=3), thì bộ giải mã có thể giải mã `scaling_list_dc_coef_minus16`, nghĩa là, giá trị hệ số của ma trận lượng tử hóa tương ứng với hệ số ma trận DC, từ bộ tham số. Giá trị của `scaling_list_dc_coef_minus16` nghĩa là hệ số ma trận DC được tính toán giả sử rằng giá trị dự đoán là 16.

Hơn nữa, bộ giải mã có thể xác định xem ma trận mặc định đã được sử dụng hay chưa bằng cách sử dụng `scaling_list_delta_coef` được sử dụng để tính toán `scaling_list_dc_coef_minus16` hoặc `nextcoef`. Ví dụ, nếu giá trị của `scaling_list_dc_coef_minus16` được giải mã làm -16, thì ma trận lượng tử hóa tương ứng có thể được xác định làm ma trận mặc định. Nếu giá trị của giá trị `nextcoef` thứ nhất được tính toán bằng cách giải mã `scaling_list_delta_coef` là 0, thì ma trận lượng tử hóa tương ứng có thể được xác định làm ma trận mặc định.

Trong các ví dụ trên các bảng cũng như các ví dụ trên các bảng 39 và 40, `nextcoef` có thể được đặt thành 16 và giá trị của `scaling_list_dc_coef_minus16` có thể đồng nghĩa với việc hệ số ma trận DC được tính toán giả sử rằng giá trị dự đoán là 16. Nếu giá trị của `scaling_list_dc_coef_minus16` được giải mã làm -16, thì bộ giải mã có thể xác định ma trận lượng tử hóa tương ứng làm ma trận mặc định.

Các phương án liên quan tới mã hóa/giải mã và việc truyền/nhận của thông tin về ma trận lượng tử hóa theo sáng chế đã được mô tả tham khảo tới các bảng và các hình vẽ.

Theo các ví dụ trên các bảng 18 và 19, các ví dụ trên các bảng 20 và 21, các ví dụ trên các bảng 23 và 24, các ví dụ trên các bảng 25 và 26, các ví dụ trên các bảng 27 và 28, các ví dụ trên các bảng 29 và 30, các ví dụ trên các bảng 31 và 32, các ví dụ trên các bảng 33 và 34, các ví dụ trên các bảng 36 và 38, các ví dụ trên các bảng 37 và 38, và các ví dụ trên các bảng 39 và 40, các ví dụ về các cấu trúc cú pháp theo sáng chế đã được mô tả tham khảo tới hai bảng, nhưng nó chỉ tạo thuận tiện cho việc mô tả và sáng chế là không bị giới hạn ở các ví dụ này.

Ví dụ, trong các ví dụ của các cấu trúc cú pháp, `scaling_list_pred_mode_flag` được minh họa để chỉ báo phương pháp dự đoán ma trận lượng tử hóa. Khi giá trị của `scaling_list_pred_mode_flag` là 0, ma trận lượng tử hóa thu được thông qua sao chép ma

trận và khi giá trị của `scaling_list_pred_mode_flag` là 1, ma trận lượng tử hóa thu được bằng cách dự đoán ma trận hệ số từ ma trận hệ số trước đó nằm trong ma trận lượng tử hóa.

Ví dụ, trong các ví dụ trên các bảng 23 và 24, khi giá trị của `scaling_list_pred_mode_flag` là 1, ma trận lượng tử hóa thu được bằng cách thu về cú pháp của ma trận lượng tử hóa, nghĩa là, cú pháp của danh sách định cỡ ‘`scaling_list`’, nhưng nó có thể được giải quyết trong cấu trúc cú pháp đơn. Cần lưu ý rằng để tạo ra hai hoặc hơn hai cấu trúc cú pháp bằng cách sử dụng một cấu trúc cú pháp có cùng một ý nghĩa không làm thay đổi các nội dung của sáng chế, mà thuộc phạm vi theo sáng chế.

Bảng 41 thể hiện ví dụ trong đó, hai cấu trúc cú pháp của các bảng 23 và 24 được tạo thành vào trong một cấu trúc cú pháp. Giống như trong các ví dụ trên các bảng 23 và 24, bộ mã hóa có thể thông tin mã hóa về ma trận lượng tử hóa của bảng 41 thành bộ tham số bao gồm ít nhất một trong số bộ tham số chuỗi và bộ tham số hình ảnh, và bộ giải mã có thể giải mã thông tin về ma trận lượng tử hóa của bảng 41 từ bộ tham số bao gồm ít nhất một trong số bộ tham số chuỗi và bộ tham số hình ảnh.

Bảng 41

<code>scaling_list_param()</code> {	Bộ mô tả
<code>for(sizeId = 0; sizeId < 4; sizeId++)</code>	
<code>for(matrixId = 0; matrixId < (sizeId == 3) ? 2 : 6; matrixId++) {</code>	
<code>scaling_list_pred_mode_flag</code> <code>[sizeId][matrixId]</code>	<code>u(1)</code>
<code>if(!<code>scaling_list_pred_mode_flag</code>[sizeId][matrixId])</code>	
<code>scaling_list_pred_matrix_id_delta</code> <code>[sizeId][matrixId]</code>	<code>ue(v)</code>
<code>else {</code>	
<code>nextCoef = 8</code>	
<code>coefNum = Min(64, (1 << (4 + (sizeId << 1))))</code>	
<code>if(sizeId > 1) {</code>	
<code>scaling_list_dc_coef_minus8</code> <code>[sizeId - 2][matrixId]</code>	<code>se(v)</code>

nextCoef = scaling_list_dc_coef_minus8[sizeId - 2][matrixId] + 8	
}	
for(i = 0; i < coefNum; i++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
ScalingList[sizeId][matrixId][i] = nextCoef	
}	
}	
}	
}	

Như được mô tả ở trên, các ví dụ trên các bảng 23 và 24 và ví dụ trên bảng 41 là giống nhau ngoại trừ số lượng các cấu trúc cú pháp là hai hoặc một.

Như trong ví dụ trên bảng 41, bộ mã hóa có thể chỉ báo chế độ dự đoán của ma trận thông qua `scaling_list_pred_mode_flag`. Ví dụ, khi sao chép giữa các ma trận lượng tử hóa được thực hiện, giá trị của `scaling_list_pred_mode_flag` được xác định là 0 và được mã hóa. Khi hệ số của ma trận được dự đoán và được mã hóa nằm trong ma trận lượng tử hóa, thì giá trị của `scaling_list_pred_mode_flag` được xác định là 1 và được mã hóa. Việc sao chép ma trận lượng tử hóa, như được mô tả ở trên, nghĩa là ma trận lượng tử hóa mặc định được sử dụng làm ma trận lượng tử hóa được mã hóa hoặc ma trận lượng tử hóa tham chiếu được sử dụng làm ma trận lượng tử hóa được mã hóa. Phương pháp dự đoán hệ số của ma trận, như được mô tả ở trên, nghĩa là phương pháp dự đoán và mã hóa hệ số nằm trong ma trận lượng tử hóa.

Nếu việc sao chép ma trận lượng tử hóa được thực hiện (`scaling_list_pred_mode_flag==0`), `scaling_list_pred_matrix_id_delta` được truyền. Như được mô tả ở trên, `scaling_list_pred_matrix_id_delta` xác định ma trận lượng tử hóa tham chiếu hoặc ma trận lượng tử hóa mặc định được sử dụng để suy ra ma trận lượng tử hóa hiện đang được mã hóa.

Ví dụ, nếu ma trận lượng tử hóa được mã hóa được xác định là ma trận lượng tử hóa mặc định, thì giá trị của `scaling_list_pred_matrix_id_delta` có thể được xác định là 0 và được mã hóa. Nghĩa là, điều này tương ứng với trường hợp mà trong đó, ma trận lượng tử hóa hiện được mã hóa thu được từ trong các ma trận lượng tử hóa mặc định. Ma trận lượng

tử hóa mặc định có thể được xác định rõ bởi các bảng 7 và 8.

Nếu ma trận lượng tử hóa hiện đang được mã hóa được xác định từ ma trận lượng tử hóa tham chiếu, giá trị của `scaling_list_pred_matrix_id_delta` có thể được xác định khi giá trị không phải là 0 và được mã hóa. Nghĩa là, điều này tương ứng với trường hợp mà trong đó, ma trận lượng tử hóa hiện đang được mã hóa, nghĩa là, `ScalingList`, được xác định từ ma trận lượng tử hóa tham chiếu như theo phương trình 33.

Phương trình 33

$$\begin{aligned} \text{RefMatrixID} &= \text{MatrixID} - \text{scaling_list_pred_matrix_id_delta}[\text{SizeID}][\text{MatrixID}] \\ \text{ScalingList}[\text{SizeID}][\text{MatrixID}][i] &= \text{ScalingList}[\text{SizeID}][\text{RefMatrixID}][i] \end{aligned}$$

Theo phương trình 33, giá trị của `scaling_list_pred_matrix_id_delta` được định rõ bởi `SizeID` và `MatrixID`, và chỉ số 'i' xác định vị trí của hệ số nằm trong ma trận lượng tử hóa.

Nếu việc dự đoán và mã hóa được thực hiện nằm trong ma trận lượng tử hóa (`scaling_list_pred_mode_flag==1`), thì các hệ số ma trận cho ma trận lượng tử hóa có kích thước 4x4, các hệ số ma trận cho ma trận lượng tử hóa có kích thước 8x8, các hệ số ma trận cho ma trận lượng tử hóa có kích thước 16x16 bao gồm hệ số ma trận DC, và các hệ số ma trận cho ma trận lượng tử hóa có kích thước 32x32 bao gồm hệ số ma trận DC có thể được mã hóa. Ở đây, tổng số các hệ số ma trận được mã hóa có thể được tính toán sử dụng $\text{coefNum} = \text{Min}(64, (1 \ll (4 + (\text{SizeID} \ll 1))))$. Hơn nữa, từng ma trận lượng tử hóa có kích thước 16x16 và ma trận lượng tử hóa có kích thước 32x32 có thể được lấy mẫu xuống tới ma trận lượng tử hóa có kích thước 8x8 và được mã hóa.

Bộ giải mã có thể giải mã các thành phần cú pháp của bảng 41 từ dòng bit, thực hiện giải lượng tử hóa trên được giải mã các thành phần cú pháp, và tái tạo hình ảnh sử dụng các kết quả này.

Như trong ví dụ trên bảng 41, bộ giải mã có thể xác định chế độ dự đoán của ma trận như được chỉ báo bởi `scaling_list_pred_mode_flag` nhận được. Ví dụ, khi giá trị của `scaling_list_pred_mode_flag` là 0, thì bộ giải mã có thể thực hiện việc sao chép giữa các ma trận lượng tử hóa. Khi giá trị của `scaling_list_pred_mode_flag` là 1, thì bộ giải mã có

thể dự đoán ma trận hệ số nằm trong ma trận lượng tử hóa. Do đó, bộ giải mã có thể thu ma trận lượng tử hóa (ngược) được sử dụng trong giải lượng tử hóa. Cần lưu ý rằng trong bản mô tả này, cả ma trận lượng tử hóa được mã hóa được áp dụng trong trường hợp lượng tử hóa lẫn ma trận lượng tử hóa được giải mã được áp dụng trong trường hợp giải lượng tử hóa được gọi là các ma trận lượng tử hóa, nhưng nó chỉ tạo thuận tiện cho việc mô tả. Ma trận được áp dụng cho lượng tử hóa và ma trận được áp dụng cho giải lượng tử hóa có thể có mối liên quan ngược, và ma trận lượng tử hóa được sử dụng trong giải lượng tử hóa có thể được gọi là danh sách định cỡ.

Việc sao chép ma trận lượng tử hóa, như được mô tả ở trên, nghĩa là ma trận lượng tử hóa mặc định được xác định là ma trận lượng tử hóa được giải mã hoặc ma trận lượng tử hóa tham chiếu được xác định là ma trận lượng tử hóa được giải mã. Phương pháp dự đoán hệ số của ma trận, như được mô tả ở trên, nghĩa là phương pháp dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa.

Nếu việc sao chép ma trận lượng tử hóa được thực hiện (`scaling_list_pred_mode_flag==0`), thì bộ giải mã xác định ma trận lượng tử hóa tham chiếu hoặc ma trận lượng tử hóa mặc định được sử dụng để suy ra ma trận lượng tử hóa hiện đang được giải mã.

Ví dụ, khi giá trị của `scaling_list_pred_matrix_id_delta` là 0, bộ giải mã có thể xác định ma trận lượng tử hóa được giải mã như là ma trận lượng tử hóa mặc định. Nghĩa là, ma trận lượng tử hóa hiện đang được giải mã có thể thu được từ trong các ma trận lượng tử hóa mặc định. Ma trận lượng tử hóa mặc định có thể được định rõ bởi các bảng 7 và 8.

Khi giá trị của `scaling_list_pred_matrix_id_delta` là không 0, thì bộ giải mã có thể xác định ma trận lượng tử hóa hiện đang được giải mã từ ma trận lượng tử hóa tham chiếu. Nghĩa là, ma trận lượng tử hóa hiện đang được giải mã, nghĩa là, `ScalingList`, có thể được xác định từ ma trận lượng tử hóa tham chiếu như theo phương trình 34.

Phương trình 34

$$\text{RefMatrixID} = \text{MatrixID} - \text{scaling_list_pred_matrix_id_delta}[\text{SizeID}][\text{MatrixID}]$$

$$\text{ScalingList}[\text{SizeID}][\text{MatrixID}][i] = \text{ScalingList}[\text{SizeID}][\text{RefMatrixID}][i]$$

Theo phương trình 34, giá trị của `scaling_list_pred_matrix_id_delta` được định rõ bởi `SizeID` và `MatrixID`, và chỉ số 'i' xác định vị trí của hệ số nằm trong ma trận lượng tử hóa.

Nếu việc dự đoán và giải mã nằm trong ma trận lượng tử hóa được thực hiện (`scaling_list_pred_mode_flag==1`), thì bộ giải mã có thể giải mã các hệ số ma trận cho ma trận lượng tử hóa có kích thước 4x4, các hệ số ma trận cho ma trận lượng tử hóa có kích thước 8x8, các hệ số ma trận cho ma trận lượng tử hóa có kích thước 16x16 bao gồm hệ số ma trận DC, và các hệ số ma trận cho ma trận lượng tử hóa có kích thước 32x32 bao gồm hệ số ma trận DC. Ở đây, tổng số các hệ số ma trận được giải mã có thể được tính toán sử dụng $\text{coefNum} = \text{Min}(64, (1 \ll (4 + (\text{SizeID} \ll 1))))$. Ở đây, do từng ma trận lượng tử hóa có kích thước 16x16 và ma trận lượng tử hóa có kích thước 32x32 được lấy mẫu xuống tới ma trận lượng tử hóa có kích thước 8x8 khi ma trận lượng tử hóa có kích thước 16x16 và ma trận lượng tử hóa có kích thước 32x32 được mã hóa, nên ma trận lượng tử hóa có kích thước 8x8 có thể được phục hồi về ma trận lượng tử hóa có kích thước 16x16 và ma trận lượng tử hóa có kích thước 32x32 bằng cách lấy mẫu lên hoặc nội suy ma trận lượng tử hóa có kích thước 8x8. Hơn nữa, nếu việc lấy mẫu lên hoặc nội suy được sử dụng, thì hệ số ma trận DC không sử dụng giá trị nội suy, nhưng có thể được thay thế bằng giá trị được suy ra từ giá trị được tạo tín hiệu bổ sung, như `scaling_list_dc_coef_minus8`.

Các ví dụ của các cấu trúc cú pháp và các ví dụ của việc mã hóa và giải mã sử dụng các cấu trúc cú pháp đã được mô tả tham khảo tới các bảng. Trong các ví dụ nêu trên, các ví dụ về mã hóa và các ví dụ về giải mã đã được mô tả sử dụng các cấu trúc cú pháp, nhưng sáng chế không bị giới hạn ở đó. Ví dụ, các bảng của các cấu trúc cú pháp chỉ có thể được sử dụng trong mã hóa hoặc giải mã.

Fig.6 là sơ đồ minh họa sơ lược một ví dụ của phương pháp thực hiện giải lượng tử hóa theo sáng chế.

Đề cập tới Fig.6, bộ giải mã có thể thu mã nhận dạng chỉ báo việc ma trận lượng tử hóa có mặt hay không trong bộ tham số ở bước S610. Thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không trong bộ tham số có thể là `scaling_list_present_flag` trong các ví dụ của các bảng nêu trên.

Ở đây, sự hiện diện của ma trận lượng tử hóa trong bộ tham số bao gồm thông tin (ví dụ, `scaling_list_pred_mode_flag`, `scaling_list_pred_matrix_id_delta`, `scaling_list_dc_coef_minus8`, và `scaling_list_delta_coef`) về ma trận lượng tử hóa có mặt trong bộ tham số. Hơn nữa, trước khi thu mã nhận dạng chỉ báo việc ma trận lượng tử hóa có mặt hay không, `scaling_list_enable_flag`, nghĩa là, mã nhận dạng chỉ báo việc ma trận lượng tử hóa được sử dụng hay chưa, có thể được thu. Nếu ma trận lượng tử hóa được sử dụng bằng cách thu `scaling_list_enable_flag`, thì mã nhận dạng chỉ báo việc ma trận lượng tử hóa có mặt hay không có thể được thu.

Ở đây, bộ tham số có thể là bộ tham số chuỗi hoặc bộ tham số hình ảnh mà thông tin về ma trận lượng tử hóa được truyền qua đó.

Bộ giải mã có thể xác định xem ma trận lượng tử hóa có mặt hay không trong bộ tham số dựa trên mã nhận dạng ở bước S620. Ví dụ, khi giá trị của `scaling_list_present_flag` là 1, thì bộ giải mã có thể xác định rằng ma trận lượng tử hóa có mặt trong bộ tham số. Khi giá trị của `scaling_list_present_flag` là 0, thì bộ giải mã có thể xác định rằng ma trận lượng tử hóa không có mặt trong bộ tham số.

Nếu, theo kết quả của việc xác định, ma trận lượng tử hóa không có mặt trong bộ tham số (tức là, nếu ma trận lượng tử hóa được xác định không có mặt trong bộ tham số), thì bộ giải mã có thể không sử dụng ma trận lượng tử hóa trong giải lượng tử hóa, có thể sử dụng ma trận lượng tử hóa (tức là, ma trận phẳng) có cùng các hệ số ma trận là 16 trong giải lượng tử hóa, hoặc có thể xác định tất cả các ma trận lượng tử hóa khi các ma trận lượng tử hóa mặc định trong giải lượng tử hóa ở bước S630.

Nếu, theo kết quả của việc xác định, ma trận lượng tử hóa có mặt trong bộ tham số (tức là, nếu ma trận lượng tử hóa được xác định có mặt trong bộ tham số), thì bộ giải mã có thể thu thông tin về ma trận lượng tử hóa theo kích thước của mỗi ma trận lượng tử hóa hoặc loại ma trận lượng tử hóa ở bước S640. Ở đây, loại ma trận lượng tử hóa có thể bao gồm ít nhất một trong số ma trận lượng tử hóa cho giải lượng tử hóa của hệ số biến đổi cho khối còn dư nội ảnh, ma trận lượng tử hóa cho giải lượng tử hóa của hệ số biến đổi cho khối còn dư liên ảnh, ma trận lượng tử hóa cho giải lượng tử hóa của hệ số biến đổi cho độ sáng khối còn dư, và ma trận lượng tử hóa cho giải lượng tử hóa của hệ số biến đổi cho

khối còn dư sắc độ hoặc tổ hợp của chúng.

Bộ giải mã có thể thực hiện giải lượng tử hóa bằng cách sử dụng ma trận lượng tử hóa thu được ở bước S650. Nếu ma trận lượng tử hóa không có mặt trong bộ tham số, thì bộ giải mã có thể thực hiện giải lượng tử hóa mà không sử dụng ma trận lượng tử hóa khi được xác định ở bước S630 hoặc có thể thực hiện giải lượng tử hóa sao cho ma trận lượng tử hóa (tức là, ma trận phẳng) có cùng các hệ số ma trận là 16 được thực hiện trong giải lượng tử hóa hoặc tất cả các ma trận lượng tử hóa được sử dụng làm các ma trận lượng tử hóa mặc định trong giải lượng tử hóa. Việc dự đoán hệ số của ma trận lượng tử hóa và sử dụng ma trận lượng tử hóa mặc định đã được mô tả ở trên liên quan tới các phương án nêu trên. Nếu ma trận lượng tử hóa có mặt trong bộ tham số, thì bộ giải mã có thể thu ma trận lượng tử hóa tương ứng và sử dụng ma trận lượng tử hóa thu được trong giải lượng tử hóa. Bộ giải mã có thể tái tạo hình ảnh dựa trên tín hiệu được lượng tử hóa nghịch đảo như được mô tả ở trên tham khảo tới Fig.2.

Trong khi đó, trên Fig.6, ví dụ trong đó, mã nhận dạng chỉ báo việc ma trận lượng tử hóa có mặt hay không được truyền bởi bộ mã hóa đã được minh họa, nhưng nó chỉ là một ví dụ của sáng chế. Như được mô tả ở trên, nếu thông tin về ma trận lượng tử hóa được truyền, thì thông tin chỉ báo việc ma trận lượng tử hóa có mặt hay không có thể không được truyền bổ sung.

Fig.7 là sơ đồ minh họa sơ lược một ví dụ của phương pháp thu thông tin về ma trận lượng tử hóa khi ma trận lượng tử hóa có mặt trong bộ tham số và thực hiện giải lượng tử hóa bằng cách sử dụng thông tin. Phương pháp Fig.7 có thể tương ứng với (1) phương pháp bao gồm các bước S640 và S650 trên Fig.6 và (2) phương pháp thu ma trận lượng tử hóa và thực hiện giải lượng tử hóa sử dụng ma trận lượng tử hóa thu được mặc dù mã nhận dạng bổ sung chỉ báo việc ma trận lượng tử hóa có mặt hay không không được truyền, nhưng ma trận lượng tử hóa có mặt trong bộ tham số và thông tin về ma trận lượng tử hóa được truyền bởi bộ mã hóa. Phương pháp (1) là giống như phương pháp (2) trừ thông tin về việc ma trận lượng tử hóa có mặt hay không được truyền qua mã nhận dạng bổ sung hay không. Trên Fig.7, bộ giải mã có thể giải mã thông tin về ma trận lượng tử hóa từ bộ tham số bao gồm ít nhất một trong số bộ tham số chuỗi và bộ tham số hình ảnh.

Trong ví dụ trên Fig.7, để làm giảm mức độ phức tạp khi mã hóa/giải mã ma trận lượng tử hóa, nếu giá trị ID của ma trận lượng tử hóa tham chiếu để dự đoán ma trận lượng tử hóa được giải mã trùng với giá trị ID của ma trận lượng tử hóa được giải mã, thì bộ giải mã không giải mã ma trận lượng tử hóa được giải mã, nhưng có thể sử dụng ma trận lượng tử hóa đã có sẵn trong bộ giải mã khi thực hiện giải lượng tử hóa. Ở đây, ma trận lượng tử hóa đã có sẵn trong bộ giải mã có thể là ma trận lượng tử hóa mặc định.

Cụ thể hơn, đề cập tới Fig.7, nếu ma trận lượng tử hóa có mặt trong bộ tham số, thì bộ giải mã có thể xác định phương pháp dự đoán ma trận lượng tử hóa ở bước S710. Sự hiện diện của ma trận lượng tử hóa trong bộ tham số bao gồm sự hiện diện của thông tin (ví dụ, `scaling_list_pred_mode_flag`, `scaling_list_pred_matrix_id_delta`, `scaling_list_dc_coef_minus8`, và `scaling_list_delta_coef`) về ma trận lượng tử hóa trong bộ tham số.

Đề cập tới các ví dụ nêu trên, phương pháp dự đoán ma trận lượng tử hóa có thể được xác định dựa trên giá trị của thành phần cú pháp ‘`pred_mode_flag`’ nhận được từ bộ mã hóa. Để làm rõ phương pháp dự đoán là cho ma trận lượng tử hóa, `pred_mode_flag` có thể được thể hiện như là `scaling_list_pred_mode_flag` giống như trong các ví dụ nêu trên. Phương pháp dự đoán ma trận lượng tử hóa, được chỉ báo bởi `pred_mode_flag`, có thể là một phương pháp bất kỳ trong số (1) phương pháp sử dụng ma trận lượng tử hóa đã có sẵn trong bộ giải mã mà không thay đổi và (2) phương pháp nhận giá trị của ma trận lượng tử hóa và thực hiện DPCM ngược giữa các giá trị hệ số của ma trận lượng tử hóa nằm trong ma trận lượng tử hóa.

Ví dụ, khi giá trị của `pred_mode_flag` là 0, thì bộ giải mã có thể sử dụng ma trận lượng tử hóa (tức là, ma trận lượng tử hóa (hoặc ma trận tham chiếu) hoặc ma trận mặc định hiện đang được giải mã) đã có sẵn trong đó. Khi giá trị của `pred_mode_flag` là 1, thì bộ giải mã có thể dự đoán và giải mã hệ số nằm trong ma trận lượng tử hóa dựa trên thông tin nhận được.

Nếu phương pháp dự đoán ma trận lượng tử hóa là phương pháp sử dụng ma trận lượng tử hóa đã có sẵn trong bộ giải mã mà không thay đổi, thì bộ giải mã có thể thu thông tin ID về ma trận lượng tử hóa ở bước S720. Thông tin ID của ma trận lượng tử hóa là thông

tin mà ma trận lượng tử hóa đã có sẵn trong bộ giải mã trên đó có thể được nhận diện, và thông tin ID có thể tương ứng với `scaling_list_pred_matrix_id_delta`, `pred_matrix_id_delta`, v.v., đã được mô tả trong các ví dụ nêu trên.

Bộ giải mã có thể xác định xem ma trận lượng tử hóa được định rõ bởi thông tin ID trùng với ma trận lượng tử hóa hiện đang được giải mã hay không ở bước S730. `Pred_matrix_id_delta` chỉ báo ma trận lượng tử hóa tham chiếu được sử dụng để suy ra ma trận lượng tử hóa. Để làm rõ rằng `pred_matrix_id_delta` là cho ma trận lượng tử hóa, `pred_matrix_id_delta` có thể được thể hiện như là `scaling_list_pred_matrix_id_delta` như trong các ví dụ nêu trên.

Khi giá trị của `scaling_list_pred_matrix_id_delta` là 0, thì bộ giải mã có thể sử dụng ma trận lượng tử hóa mặc định, được định rõ bởi thông tin (`SizeID` và/hoặc `MatrixID`) về ma trận lượng tử hóa, làm ma trận lượng tử hóa được giải mã.

Khi giá trị của `scaling_list_pred_matrix_id_delta` là không 0, thì bộ giải mã có thể suy ra ma trận lượng tử hóa được giải mã từ ma trận lượng tử hóa tham chiếu đã được giải mã do `MatrixID` xác định ma trận lượng tử hóa được giải mã (hoặc loại ma trận lượng tử hóa) và `RefMatrixID` xác định ma trận lượng tử hóa tham chiếu có quan hệ '`scaling_list_pred_matrix_id_delta=MatrixID-RefMatrixID`' với tham khảo tới các ví dụ nêu trên.

Nếu, theo kết quả của việc xác định ở bước S730, ma trận lượng tử hóa được xác định bởi thông tin ID của ma trận lượng tử hóa được xác định không trùng với ma trận lượng tử hóa được giải mã (tức là, `!scaling_list_pred_matrix_id_delta==0` hoặc `MatrixID!=RefMatrixID`), thì bộ giải mã có thể xác định ma trận lượng tử hóa được sử dụng khi giải lượng tử hóa được thực hiện bằng cách sử dụng thông tin ID của ma trận lượng tử hóa ở bước S740. Trong trường hợp này, bộ giải mã có thể xác định ma trận lượng tử hóa được sử dụng khi giải lượng tử hóa được thực hiện như trong các bảng nêu trên dựa trên thông tin ID '`scaling_list_pred_matrix_id_delta`' của ma trận lượng tử hóa.

Nếu, theo kết quả của việc xác định ở bước S730, ma trận lượng tử hóa được xác định bởi thông tin ID của ma trận lượng tử hóa được xác định trùng với ma trận lượng tử hóa được giải mã (tức là, `scaling_list_pred_matrix_id_delta==0` hoặc

MatrixID==RefMatrixID), thì bộ giải mã có thể sử dụng ma trận lượng tử hóa mặc định đã có trong đó ở bước S750. Ở đây, ma trận lượng tử hóa mặc định có thể được xác định bằng cách sử dụng các bảng 7 và 8.

Nếu phương pháp dự đoán ma trận lượng tử hóa là phương pháp sử dụng phương pháp DPCM ngược giữa các giá trị hệ số của ma trận lượng tử hóa ở bước S710, thì bộ giải mã có thể khởi tạo giá trị hệ số của ma trận lượng tử hóa ở bước S760. Ví dụ, bộ giải mã có thể khởi tạo giá trị hệ số 'nextcoef' của ma trận lượng tử hóa bằng cách thiết lập giá trị hệ số bằng một hằng số. Hằng số thiết lập khi khởi tạo giá trị hệ số 'nextcoef' có thể là một số bất kỳ trong số 8 và 16 như trong các ví dụ của các bảng nêu trên.

Bộ giải mã có thể giải mã giá trị chênh lệch giữa các hệ số nằm trong ma trận lượng tử hóa nhận được từ bộ mã hóa ở bước S770. Giá trị chênh lệch giữa các hệ số nằm trong ma trận lượng tử hóa có thể được định rõ bởi thành phần cú pháp, như delta_coef. Để làm rõ rằng thành phần cú pháp là cho ma trận lượng tử hóa, delta_coef có thể được gọi là scaling_list_delta_coef như trong các ví dụ của các bảng nêu trên.

Bộ giải mã có thể suy ra giá trị hệ số của ma trận lượng tử hóa ở bước S780. Bộ giải mã có thể suy ra giá trị hệ số của ma trận lượng tử hóa được giải mã (tức là, ma trận lượng tử hóa hiện tại) bằng cách thêm giá trị chênh lệch giữa các hệ số nằm trong ma trận lượng tử hóa vào giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước. Ví dụ, bộ giải mã có thể suy ra giá trị hệ số của ma trận lượng tử hóa hiện tại bằng cách sử dụng quan hệ 'nextcoef=(nextcoef+delta_coef+256)%256' như trong các phương trình nêu trên.

Bộ giải mã có thể xác định xem ma trận lượng tử hóa đã được suy ra hay chưa ở bước S790. Nếu, theo kết quả của việc xác định, ma trận lượng tử hóa được xác định là chưa được suy ra (tức là, nếu tất cả các giá trị hệ số nằm trong ma trận lượng tử hóa không được giải mã), thì bộ giải mã quay lại bước S770 và thực hiện các bước tiếp theo.

Bộ giải mã có thể thực hiện giải lượng tử hóa bằng cách sử dụng ma trận lượng tử hóa thu được ở bước S650.

Trong khi đó, mặc dù không được mô tả rõ ràng trong ví dụ trên Fig.7 nhưng khi suy ra giá trị hệ số của ma trận lượng tử hóa, thì hệ số ma trận DC cho ma trận lượng tử

hóa có kích thước cụ thể trước tiên có thể được suy ra như trong các phương án nêu trên.

Fig.8 là sơ đồ minh họa sơ lược một ví dụ khác về phương pháp thu thông tin về ma trận lượng tử hóa khi ma trận lượng tử hóa có mặt trong bộ tham số và thực hiện giải lượng tử hóa bằng cách sử dụng thông tin. Phương pháp theo Fig.8 có thể tương ứng với (1) phương pháp bao gồm các bước S640 và S650 trên Fig.6 và (2) phương pháp thu ma trận lượng tử hóa và thực hiện giải lượng tử hóa sử dụng ma trận lượng tử hóa thu được mặc dù mã nhận dạng bổ sung chỉ báo việc ma trận lượng tử hóa có mặt hay không không được truyền, nhưng ma trận lượng tử hóa có mặt trong bộ tham số và thông tin về ma trận lượng tử hóa được truyền bởi bộ mã hóa. Phương pháp (1) giống như phương pháp (2) trừ thông tin về việc ma trận lượng tử hóa có mặt hay không được truyền qua mã nhận dạng bổ sung hay không.

Theo ví dụ trên Fig.8, để ngăn không cho ma trận lượng tử hóa không cần thiết bị truyền đi, nếu giá trị thứ nhất của ma trận lượng tử hóa được giải mã là giá trị cụ thể thì bộ giải mã không giải mã ma trận lượng tử hóa được giải mã và có thể sử dụng ma trận lượng tử hóa mặc định đã có sẵn trong đó khi thực hiện giải lượng tử hóa.

Cụ thể hơn, đề cập tới Fig.8, nếu ma trận lượng tử hóa có mặt trong bộ tham số, thì bộ giải mã có thể xác định phương pháp dự đoán ma trận lượng tử hóa ở bước S810. Sự hiện diện của ma trận lượng tử hóa trong bộ tham số bao gồm sự hiện diện của thông tin về ma trận lượng tử hóa trong bộ tham số.

Nếu phương pháp dự đoán ma trận lượng tử hóa là phương pháp sử dụng ma trận lượng tử hóa đã có sẵn trong bộ giải mã mà không thay đổi, thì bộ giải mã có thể thu thông tin ID về ma trận lượng tử hóa ở bước S820.

Thông tin ID của ma trận lượng tử hóa là thông tin mà ma trận lượng tử hóa đã có sẵn trong bộ giải mã có thể được nhận diện, và thông tin ID có thể tương ứng với `scaling_list_pred_matrix_id_delta`, `pred_matrix_id_delta`, v.v., đã được mô tả trong các ví dụ nêu trên.

Bộ giải mã có thể xác định ma trận lượng tử hóa được sử dụng khi thực hiện giải lượng tử hóa bằng cách sử dụng thông tin ID của ma trận lượng tử hóa ở bước S830. Bộ

giải mã có thể xác định ma trận lượng tử hóa như được chỉ báo bởi thông tin ID, như `scaling_list_pred_matrix_id_delta` hoặc `pred_matrix_id_delta`.

Nếu phương pháp dự đoán ma trận lượng tử hóa là phương pháp sử dụng phương pháp DPCM ngược giữa các giá trị hệ số của ma trận lượng tử hóa ở bước S810, bộ giải mã có thể khởi tạo giá trị hệ số của ma trận lượng tử hóa ở bước S840. Ví dụ, bộ giải mã có thể khởi tạo giá trị hệ số 'nextcoef' của ma trận lượng tử hóa bằng cách thiết lập giá trị hệ số bằng một hằng số. Hằng số đặt khi khởi tạo giá trị hệ số 'nextcoef' có thể là một số bất kỳ trong 8 và 16 như trong các ví dụ của các bảng nêu trên.

Bộ giải mã có thể giải mã giá trị chênh lệch giữa các hệ số nằm trong ma trận lượng tử hóa nhận được từ bộ mã hóa ở bước S850. Giá trị chênh lệch giữa các hệ số nằm trong ma trận lượng tử hóa có thể được định rõ bởi thành phần cú pháp, như `delta_coef`.

Bộ giải mã có thể suy ra giá trị hệ số của ma trận lượng tử hóa ở bước S860. Bộ giải mã có thể suy ra giá trị hệ số của ma trận lượng tử hóa được giải mã (tức là, ma trận lượng tử hóa hiện tại) bằng cách thêm giá trị chênh lệch giữa các hệ số nằm trong ma trận lượng tử hóa tới giá trị hệ số của ma trận lượng tử hóa đã được giải mã từ trước. Ví dụ, bộ giải mã có thể suy ra giá trị hệ số của ma trận lượng tử hóa hiện tại bằng cách sử dụng quan hệ ' $\text{nextcoef} = (\text{nextcoef} + \text{delta_coef}) \% 256$ ' như trong các phương trình nêu trên.

Bộ giải mã có thể xác định xem hệ số được suy ra là giá trị hệ số thứ nhất của ma trận lượng tử hóa hay không hoặc hệ số được suy ra có trùng với một giá trị cụ thể hay không ở bước S870. Ở đây, giá trị cụ thể có thể là 0.

Nếu, theo kết quả của việc xác định ở bước S870, hệ số được suy ra được xác định là giá trị hệ số thứ nhất của ma trận lượng tử hóa và trùng với một giá trị cụ thể, thì bộ giải mã có thể xác định để sử dụng ma trận lượng tử hóa mặc định, đã có sẵn ở trong đó, trong giải lượng tử hóa ở bước S880.

Nếu, theo kết quả của việc xác định ở bước S870, hệ số được suy ra được xác định không phải là giá trị hệ số thứ nhất của ma trận lượng tử hóa và không trùng với một giá trị cụ thể, thì bộ giải mã kiểm tra xem tất cả các giá trị chênh lệch cho các hệ số nằm trong ma trận lượng tử hóa đã được giải mã hay chưa ở bước S890. Nếu, theo kết quả của việc kiểm

tra, tất cả các giá trị chênh lệch được xác định chưa được giải mã, thì bộ giải mã có thể quay lại bước S850 và thực hiện các bước tiếp theo.

Bộ giải mã có thể thực hiện giải lượng tử hóa bằng cách sử dụng ma trận lượng tử hóa thu được ở bước S650.

Trong các phương án nêu trên, thông tin về ma trận lượng tử hóa nghĩa là ma trận lượng tử hóa hoặc thông tin mà ma trận lượng tử hóa có thể được suy ra trên đó. Ở đây, thông tin mà ma trận lượng tử hóa có thể được suy ra trên đó có thể đồng nghĩa với việc thông tin về việc liệu ma trận mặc định đã được sử dụng hay chưa, loại phương pháp mã hóa/giải mã dự đoán, ma trận lượng tử hóa tham chiếu ID, ma trận lượng tử hóa tham chiếu, v.v..

Trong hệ thống làm ví dụ nêu trên, mặc dù các phương pháp đã được mô tả dựa trên các sơ đồ dưới dạng một chuỗi các bước hoặc các khối, nhưng sáng chế không bị giới hạn ở thứ tự của các bước và một số bước có thể được thực hiện theo thứ tự khác với thứ tự của các bước khác hoặc có thể được thực hiện đồng thời với các bước khác. Hơn nữa, người có hiểu biết trung bình trong lĩnh vực sẽ hiểu rằng các bước được thể hiện trong sơ đồ không phải là duy nhất và các bước có thể bao gồm các bước bổ sung hoặc một hoặc nhiều bước trong sơ đồ có thể được loại bỏ mà không nằm ngoài phạm vi của sáng chế.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã video, phương pháp này bao gồm các bước:

xác định phương pháp dự đoán ma trận lượng tử hóa được sử dụng trong lượng tử hóa nghịch đảo; và

giải mã ma trận lượng tử hóa được sử dụng trong lượng tử hóa nghịch đảo theo phương pháp dự đoán ma trận lượng tử hóa,

trong đó phương pháp dự đoán ma trận lượng tử hóa là phương pháp bất kỳ trong số phương pháp dự đoán giữa các hệ số trong ma trận lượng tử hóa và phương pháp sao chép ma trận lượng tử hóa,

trong đó trong giải mã ma trận lượng tử hóa, đáp lại phương pháp dự đoán ma trận lượng tử hóa là phương pháp dự đoán giữa các hệ số trong ma trận lượng tử hóa, giá trị hệ số ma trận DC thu được đối với ma trận lượng tử hóa có kích thước và giá trị ban đầu đối với hệ số ma trận lượng tử hóa được suy ra dựa trên giá trị hệ số ma trận DC,

trong đó trong giải mã ma trận lượng tử hóa, đáp lại phương pháp dự đoán ma trận lượng tử hóa là phương pháp sao chép ma trận lượng tử hóa, thành phần cú pháp được giải mã, thành phần cú pháp chỉ báo xem ma trận lượng tử hóa tham chiếu được sử dụng hay ma trận lượng tử hóa mặc định được sử dụng làm ma trận lượng tử hóa,

trong đó thành phần cú pháp được sử dụng chung cho

chức năng thứ nhất để chỉ báo giá trị chênh lệch giữa giá trị để định rõ ma trận lượng tử hóa đích đang giải mã và giá trị để định rõ ma trận lượng tử hóa tham chiếu, và

chức năng thứ hai để chỉ báo rằng ma trận lượng tử hóa được giải mã từ ma trận lượng tử hóa mặc định nếu giá trị của thành phần cú pháp là 0.

2. Phương pháp giải mã video theo điểm 1, trong đó trong bước xác định phương pháp dự đoán ma trận lượng tử hóa, phương pháp dự đoán ma trận lượng tử hóa được xác định dựa trên thông tin cờ nhận được từ bộ mã hóa.

3. Phương pháp mã hóa video, phương pháp này bao gồm các bước:

xác định ma trận lượng tử hóa cần được sử dụng trong lượng tử hóa và thực hiện lượng tử hóa;

xác định phương pháp dự đoán ma trận lượng tử hóa; và

thông tin mã hóa về ma trận lượng tử hóa theo phương pháp dự đoán ma trận lượng tử hóa,

trong đó phương pháp dự đoán ma trận lượng tử hóa là phương pháp bất kỳ trong số phương pháp dự đoán giữa các hệ số trong ma trận lượng tử hóa và phương pháp sao chép ma trận lượng tử hóa,

trong đó thông tin mã hóa về ma trận lượng tử hóa mã hóa giá trị hệ số ma trận DC đối với ma trận lượng tử hóa có kích thước đáp lại phương pháp dự đoán ma trận lượng tử hóa là phương pháp dự đoán giữa các hệ số trong ma trận lượng tử hóa,

trong đó thông tin mã hóa về ma trận lượng tử hóa mã hóa thành phần cú pháp chỉ báo xem ma trận lượng tử hóa tham chiếu được sử dụng hay ma trận lượng tử hóa mặc định được sử dụng làm ma trận lượng tử hóa đáp lại phương pháp dự đoán ma trận lượng tử hóa là phương pháp sao chép ma trận lượng tử hóa,

trong đó thành phần cú pháp được sử dụng chung cho

chức năng thứ nhất để chỉ báo giá trị chênh lệch giữa giá trị để định rõ ma trận lượng tử hóa đích đang giải mã và giá trị để định rõ ma trận lượng tử hóa tham chiếu, và

chức năng thứ hai để chỉ báo rằng ma trận lượng tử hóa được giải mã từ ma trận lượng tử hóa mặc định nếu giá trị của thành phần cú pháp là 0.

4. Vật ghi đọc được bằng máy tính lâu dài lưu trữ dòng بیت, dòng بیت này bao gồm:

thông tin về ma trận lượng tử hóa,

trong đó thông tin về ma trận lượng tử hóa được mã hóa theo phương pháp dự đoán ma trận lượng tử hóa,

trong đó phương pháp dự đoán ma trận lượng tử hóa là phương pháp bất kỳ trong số phương pháp dự đoán giữa các hệ số trong ma trận lượng tử hóa và phương pháp sao chép ma trận lượng tử hóa,

trong đó thông tin về ma trận lượng tử hóa bao gồm giá trị hệ số ma trận DC đối với ma trận lượng tử hóa có kích thước đáp lại phương pháp dự đoán ma trận lượng tử hóa là phương pháp dự đoán giữa các hệ số trong ma trận lượng tử hóa,

trong đó thông tin về ma trận lượng tử hóa bao gồm thành phần cú pháp chỉ báo xem ma trận lượng tử hóa tham chiếu được sử dụng hay ma trận lượng tử hóa mặc định được sử dụng làm ma trận lượng tử hóa đáp lại phương pháp dự đoán ma trận lượng tử hóa là phương pháp sao chép ma trận lượng tử hóa,

trong đó thành phần cú pháp được sử dụng chung cho

chức năng thứ nhất để chỉ báo giá trị chênh lệch giữa giá trị để định rõ ma trận lượng tử hóa đích đang giải mã và giá trị để định rõ ma trận lượng tử hóa tham chiếu, và

chức năng thứ hai để chỉ báo rằng ma trận lượng tử hóa được giải mã từ ma trận lượng tử hóa mặc định nếu giá trị của thành phần cú pháp là 0.

FIG. 2

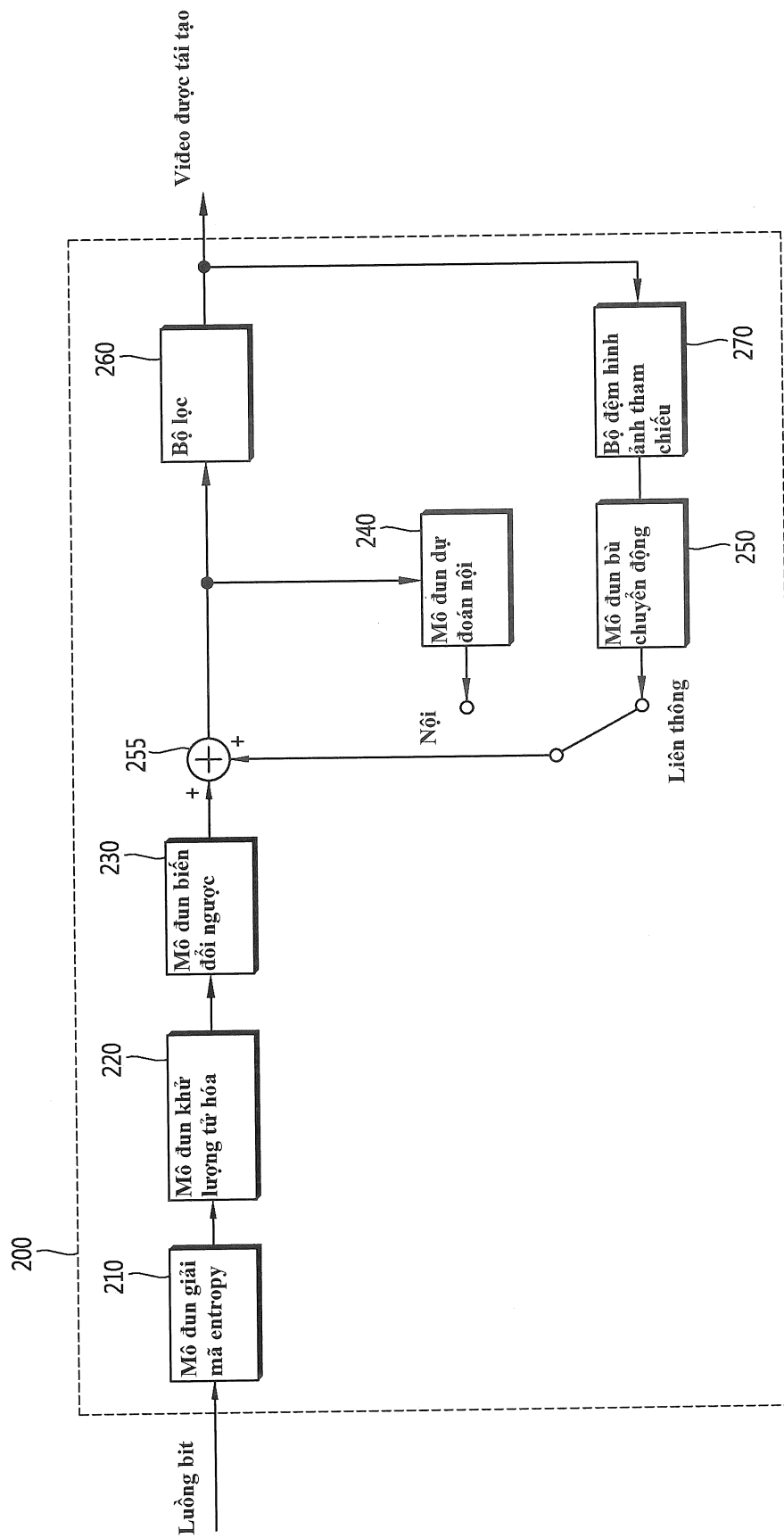


FIG. 3

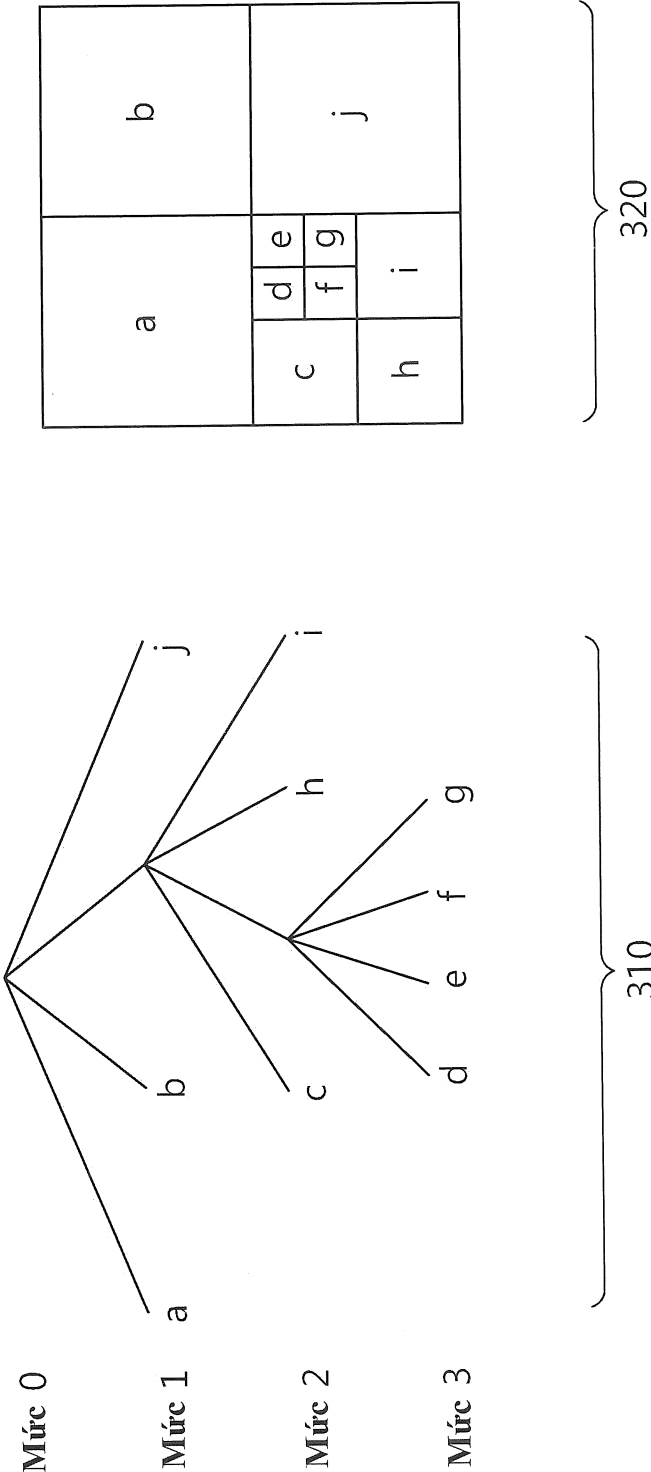


FIG. 4

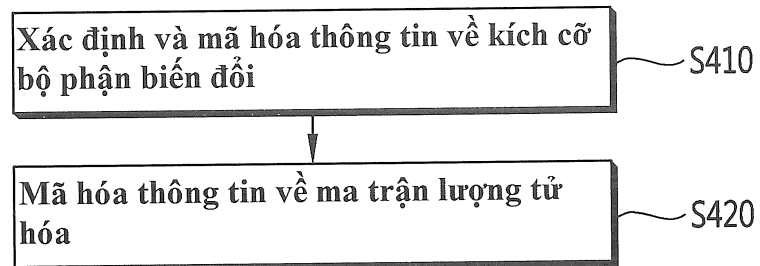


FIG. 5

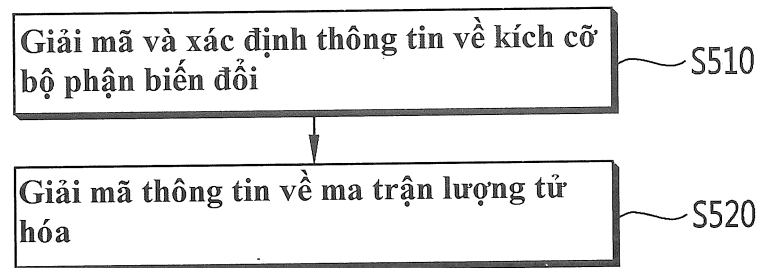


FIG. 6

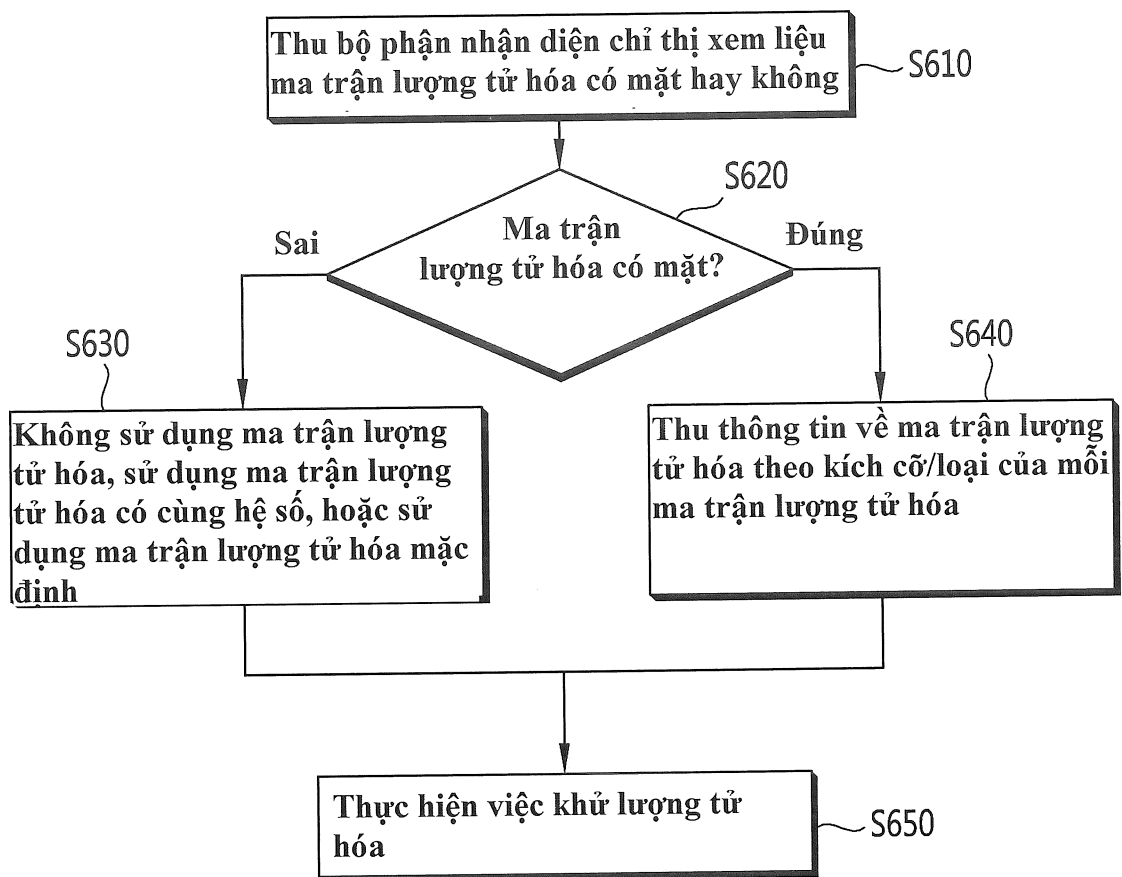


FIG. 7

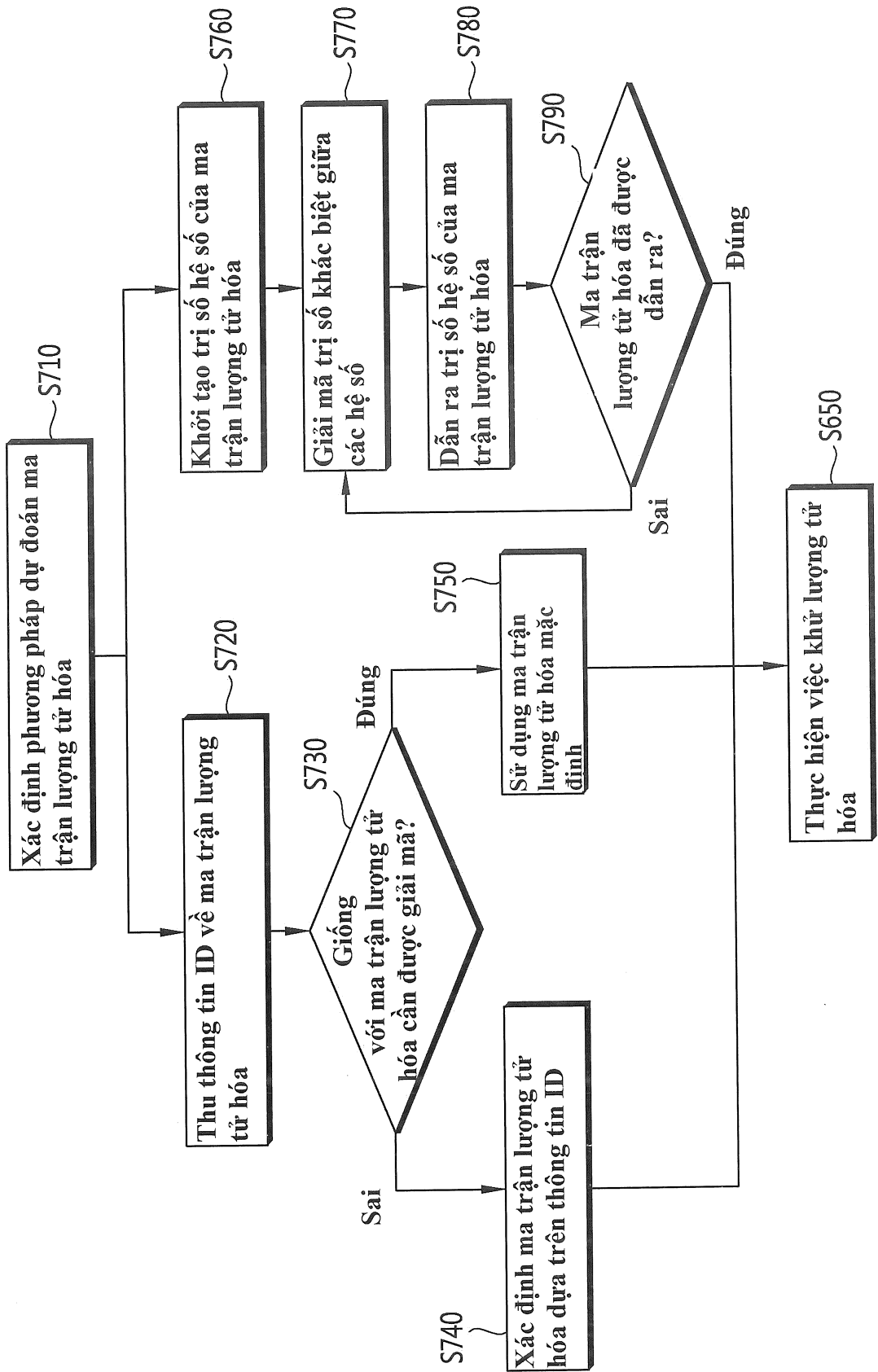


FIG. 8

