



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

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

(11)



1-0049308

(51)^{2020.01} H04N 19/157; H04N 19/119; H04N
19/137; H04N 19/70; H04N 19/573;
H04N 19/593; H04N 19/11

(13) B

(21) 1-2021-05104

(22) 20/01/2020

(86) PCT/KR2020/000964 20/01/2020

(87) WO 2020/149725 23/07/2020

(30) 10-2019-0006511 18/01/2019 KR; 10-2019-0037000 29/03/2019 KR; 10-2019-0040400 05/04/2019 KR; 10-2019-0064959 31/05/2019 KR; 10-2019-0075316 24/06/2019 KR; 10-2019-0081764 07/07/2019 KR; 10-2019-0125746 11/10/2019 KR

(45) 25/07/2025 448

(43) 25/01/2022 406A

(73) WILUS INSTITUTE OF STANDARDS AND TECHNOLOGY INC. (KR)
5F 216 Hwangsaekul-ro Bundang-gu Seongnam-si Gyeonggi-do 13595, Republic of Korea

(72) KO, Geonjung (KR); KIM, Dongcheol (KR); SON, Juhyung (KR); JUNG, Jaehong (KR); KWAK, Jinsam (KR).

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

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ XỬ LÝ TÍN HIỆU VIDEO, VÀ VẬT GHI KHÔNG KHẢ BIẾN ĐỌC ĐƯỢC BỞI MÁY TÍNH LƯU TRỮ DÒNG BIT

(21) 1-2021-05104

(57) Sáng chế đề cập đến phương pháp và thiết bị xử lý tín hiệu video để mã hóa hoặc giải mã tín hiệu video. Cụ thể là, phương pháp xử lý tín hiệu video có thể bao gồm các bước: phân tích cú pháp phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp nhất có được áp dụng cho khối hiện thời hay không; khi chế độ hợp nhất được áp dụng cho khối hiện thời, xác định liệu có phân tích cú pháp phần tử cú pháp thứ hai hay không dựa vào điều kiện định trước thứ nhất, trong đó phần tử cú pháp thứ hai biểu thị liệu chế độ thứ nhất hay chế độ thứ hai được áp dụng cho khối hiện thời; khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, xác định liệu có phân tích cú pháp phần tử cú pháp thứ ba hay không dựa vào điều kiện định trước thứ hai, trong đó phần tử cú pháp thứ ba biểu thị liệu chế độ được áp dụng cho khối hiện thời giữa chế độ thứ ba và chế độ thứ tư; xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba; thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định; và tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời.

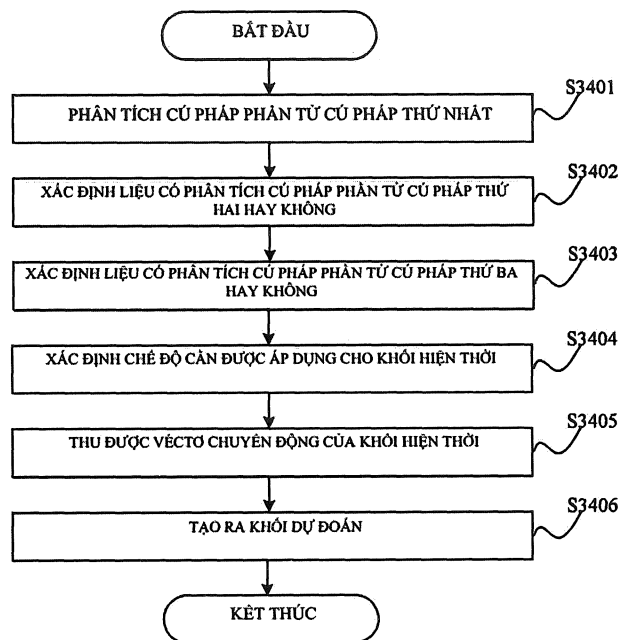


FIG.34

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến phương pháp và thiết bị xử lý tín hiệu video và, cụ thể hơn là, phương pháp và thiết bị xử lý tín hiệu video, trong đó tín hiệu video này được mã hóa hoặc giải mã bằng cách sử dụng phép bù chuyển động.

Tình trạng kỹ thuật của sáng chế

Kỹ thuật mã hóa nén đề cập đến một loạt công nghệ xử lý tín hiệu để truyền thông tin được số hóa thông qua đường truyền thông hoặc lưu trữ thông tin này theo kiểu thích hợp đối với vật ghi. Các đối tượng của kỹ thuật mã hóa nén bao gồm các âm thanh, hình ảnh, ký hiệu và dạng tương tự, và cụ thể, công nghệ để thực hiện bước mã hóa nén liên quan đến các hình ảnh được gọi là phép nén hình ảnh video. Kỹ thuật mã hóa nén liên quan đến các tín hiệu video được thực hiện bằng cách loại bỏ thông tin dư thừa khi xem xét sự tương quan theo không gian, tương quan theo thời gian, tương quan theo xác suất và dạng tương tự. Tuy nhiên, sự phát triển gần đây của các loại phương tiện và công cụ truyền dữ liệu khác nhau đã tạo ra yêu cầu đối với các phương pháp và thiết bị để xử lý các tín hiệu video với các hiệu quả cao hơn.

Bản chất kỹ thuật của sáng chế

Vấn đề kỹ thuật

Một khía cạnh của sáng chế là để cải thiện hiệu quả mã hóa tín hiệu video. Một khía cạnh khác của sáng chế là để đề xuất phương pháp báo hiệu chế độ hợp nhất hiệu quả.

Giải pháp cho vấn đề

Để giải quyết các vấn đề nêu trên, sáng chế đề xuất thiết bị xử lý tín hiệu video và phương pháp xử lý tín hiệu video như sau.

Theo một phương án, sáng chế đề xuất phương pháp xử lý tín hiệu video, phương pháp này bao gồm các bước: phân tích cú pháp phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp nhất có được áp dụng cho khối hiện thời hay không; xác định liệu có phân tích cú pháp phần tử cú pháp thứ hai hay không dựa vào điều kiện định trước thứ nhất khi chế độ hợp nhất được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ hai này biểu thị liệu chế độ thứ nhất hay chế độ thứ hai được áp dụng cho khối hiện thời; xác định liệu có

phân tích cú pháp phần tử cú pháp thứ ba hay không dựa vào điều kiện định trước thứ hai khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ ba này biểu thị chế độ trong số chế độ thứ ba và chế độ thứ tư được áp dụng cho khối hiện thời; xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba; thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định; và tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời, trong đó điều kiện thứ nhất bao gồm ít nhất một điều kiện trong số điều kiện nhờ đó chế độ thứ ba sử dụng được và điều kiện nhờ đó chế độ thứ tư sử dụng được.

Theo một phương án, chế độ thứ ba và chế độ thứ tư có thể được đặt muộn hơn so với chế độ thứ nhất trong trình tự giải mã theo cú pháp dữ liệu hợp nhất.

Theo một phương án, phương pháp này có thể còn bao gồm bước phân tích cú pháp phần tử cú pháp thứ hai khi điều kiện thứ nhất được thỏa mãn, và khi điều kiện thứ nhất không được thỏa mãn, thì giá trị của phần tử cú pháp thứ hai có thể được suy ra là 1.

Theo một phương án, khi điều kiện thứ nhất không được thỏa mãn, thì phần tử cú pháp thứ hai có thể được suy ra dựa vào phần tử cú pháp thứ tư biểu thị liệu chế độ hợp nhất dựa trên khối phụ có được áp dụng cho khối hiện thời hay không.

Theo một phương án, điều kiện thứ hai có thể bao gồm điều kiện nhờ đó chế độ thứ tư sử dụng được.

Theo một phương án, điều kiện thứ hai có thể bao gồm ít nhất một điều kiện trong số các điều kiện liên quan đến việc liệu chế độ thứ ba sử dụng được theo trình tự hiện thời hay không, liệu chế độ thứ tư sử dụng được theo trình tự hiện thời hay không, liệu số lượng ứng viên tối đa dùng cho chế độ thứ tư có lớn hơn 1 hay không, liệu chiều rộng của khối hiện thời có nhỏ hơn kích thước định trước thứ nhất hay không, và liệu chiều cao của khối hiện thời có nhỏ hơn kích thước định trước thứ hai hay không.

Theo một phương án, phương pháp này có thể còn bao gồm bước: khi phần tử cú pháp thứ hai có giá trị bằng 1, thì thu được phần tử cú pháp thứ năm biểu thị liệu chế độ được áp dụng cho khối hiện thời là chế độ thứ nhất hay chế độ thứ hai.

Theo một phương án, sáng chế đề xuất thiết bị xử lý tín hiệu video bao gồm bộ xử lý, trong đó bộ xử lý này được tạo cấu hình để: phân tích cú pháp phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp nhất có được áp dụng cho khối hiện thời hay không; xác định liệu có phân tích cú pháp phần tử cú pháp thứ hai hay không dựa vào điều kiện định

trước thứ nhất khi chế độ hợp nhất được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ hai biểu thị liệu chế độ thứ nhất hay chế độ thứ hai được áp dụng cho khối hiện thời; xác định liệu có phân tích cú pháp phần tử cú pháp thứ ba hay không dựa vào điều kiện định trước thứ hai khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ ba biểu thị chế độ trong số chế độ thứ ba và chế độ thứ tư được áp dụng cho khối hiện thời; xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba; thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định; và tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời, trong đó điều kiện thứ nhất bao gồm ít nhất một điều kiện trong số điều kiện nhờ đó chế độ thứ ba sử dụng được và điều kiện nhờ đó chế độ thứ tư sử dụng được.

Theo một phương án, chế độ thứ ba và chế độ thứ tư có thể được đặt muợn hơn so với chế độ thứ nhất trong trình tự giải mã theo cú pháp dữ liệu hợp nhất.

Theo một phương án, bộ xử lý có thể phân tích cú pháp phần tử cú pháp thứ hai khi điều kiện thứ nhất được thỏa mãn; và khi điều kiện thứ nhất không được thỏa mãn, thì giá trị của phần tử cú pháp thứ hai có thể được suy ra là 1.

Theo một phương án, khi điều kiện thứ nhất không được thỏa mãn, thì phần tử cú pháp thứ hai có thể được suy ra dựa vào phần tử cú pháp thứ tư biểu thị liệu chế độ hợp nhất dựa trên khối phụ có được áp dụng cho khối hiện thời hay không.

Theo một phương án, điều kiện thứ hai có thể bao gồm điều kiện nhờ đó chế độ thứ tư sử dụng được.

Theo một phương án, điều kiện thứ hai có thể bao gồm ít nhất một điều kiện trong số các điều kiện liên quan đến việc liệu chế độ thứ ba sử dụng được theo trình tự hiện thời hay không, liệu chế độ thứ tư sử dụng được theo trình tự hiện thời hay không, liệu số lượng ứng viên tối đa dùng cho chế độ thứ tư có lớn hơn 1 hay không, liệu chiều rộng của khối hiện thời có nhỏ hơn kích thước định trước thứ nhất hay không và liệu chiều cao của khối hiện thời có nhỏ hơn kích thước định trước thứ hai hay không.

Theo một phương án, khi phần tử cú pháp thứ hai có giá trị bằng 1, thì bộ xử lý có thể thu được phần tử cú pháp thứ năm biểu thị liệu chế độ được áp dụng cho khối hiện thời là chế độ thứ nhất hay chế độ thứ hai.

Theo một phương án, sáng chế đề xuất phương pháp xử lý tín hiệu video, phương pháp này bao gồm các bước: mã hóa phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp

nhất có được áp dụng cho khối hiện thời hay không; xác định liệu có mã hóa phần tử cú pháp thứ hai hay không dựa vào điều kiện định trước thứ nhất khi chế độ hợp nhất được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ hai biểu thị liệu chế độ được áp dụng cho khối hiện thời là chế độ thứ nhất hay chế độ thứ hai; xác định liệu có mã hóa phần tử cú pháp thứ ba hay không dựa vào điều kiện định trước thứ hai khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ ba biểu thị chế độ được áp dụng cho khối hiện thời trong số chế độ thứ ba hoặc chế độ thứ tư; xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba; thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định; và tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời, trong đó điều kiện thứ nhất bao gồm ít nhất một điều kiện trong số điều kiện nhờ đó chế độ thứ ba sử dụng được và điều kiện nhờ đó chế độ thứ tư sử dụng được.

Hiệu quả đạt được bởi sáng chế

Theo một phương án của sáng chế, có thể làm tăng hiệu quả mã hóa của tín hiệu video. Cụ thể, theo một phương án của sáng chế, có thể làm giảm tổng phí báo hiệu và làm tăng hiệu quả nén bằng cách điều khiển thứ tự phân tích cú pháp.

Mô tả vắn tắt các hình vẽ

FIG.1 là sơ đồ khối sơ lược minh họa thiết bị để mã hóa tín hiệu video theo một phương án của sáng chế;

FIG.2 là sơ đồ khối sơ lược minh họa thiết bị để giải mã tín hiệu video theo một phương án của sáng chế;

FIG.3 minh họa một phương án trong đó đơn vị cây mã hóa được chia thành các đơn vị mã hóa nằm trong ảnh;

FIG.4 minh họa một phương án về phương pháp phân chia tín hiệu của cây tứ phân và cây đa kiểu;

FIG.5 minh họa phép dự đoán liên ảnh theo một phương án của sáng chế;

FIG.6 minh họa phương pháp báo hiệu vectơ chuyển động theo một phương án của sáng chế;

FIG.7 minh họa phương pháp báo hiệu thông tin phân giải vectơ chuyển động thích ứng theo một phương án của sáng chế;

FIG.8 minh họa một ví dụ về cú pháp của đơn vị mã hóa theo một phương án của sáng chế;

FIG.9 minh họa một ví dụ về cú pháp của đơn vị mã hóa theo một phương án của sáng chế;

FIG.10 minh họa một ví dụ về phương pháp báo hiệu chế độ hợp nhất theo một phương án của sáng chế;

FIG.11 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.12 minh họa cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.13 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.14 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.15 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.16 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.17 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.18 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.19 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.20 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.21 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.22 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.23 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương

án của sáng chế;

FIG.24 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.25 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.26 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.27 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.28 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.29 minh họa một ví dụ về phương pháp báo hiệu chế độ hợp nhất theo một phương án của sáng chế;

FIG.30 minh họa một ví dụ về cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.31 minh họa một ví dụ về cú pháp dữ liệu hợp nhất theo một phương án của sáng chế;

FIG.32 minh họa chế độ hợp nhất hình học theo một phương án của sáng chế;

FIG.33 minh họa cú pháp dữ liệu hợp nhất theo một phương án của sáng chế; và

FIG.34 minh họa một ví dụ về phương pháp xử lý tín hiệu video theo một phương án mà sáng chế được áp dụng.

Mô tả chi tiết sáng chế

Các thuật ngữ được sử dụng trong bản mô tả này có thể là các thuật ngữ chung đang được sử dụng rộng rãi khi xem xét các chức năng theo sáng chế nhưng có thể thay đổi theo dự định của người có hiểu biết trung bình trong lĩnh vực kỹ thuật, thói quen hoặc sự ra đời của công nghệ mới. Ngoài ra, trong các trường hợp nhất định, có thể có các thuật ngữ mà chủ đơn chọn tùy ý và trong trường hợp này, các ý nghĩa của các thuật ngữ này được mô tả trong phần mô tả tương ứng theo sáng chế này. Theo đó, các thuật ngữ được sử dụng trong bản mô tả này cần được hiểu dựa vào các ý nghĩa cơ bản của các thuật ngữ và nội dung qua toàn bộ bản mô tả này.

Trong bản mô tả này, một số thuật ngữ có thể được hiểu như sau. Mã hóa có thể được hiểu là mã hóa hoặc giải mã trong một số trường hợp. Trong bản mô tả này, thiết bị để tạo ra dòng bit tín hiệu video bằng cách thực hiện bước mã hóa (coding) tín hiệu video được gọi là thiết bị mã hóa hoặc bộ mã hóa, và thiết bị mà thực hiện bước giải mã (decoding) dòng bit tín hiệu video để khôi phục tín hiệu video được gọi là thiết bị giải mã hoặc bộ giải mã. Ngoài ra, trong bản mô tả này, thiết bị xử lý tín hiệu video được sử dụng làm thuật ngữ về khái niệm bao gồm cả bộ mã hóa và bộ giải mã. Thông tin là thuật ngữ bao gồm tất cả các giá trị, tham số, hệ số, phân tử, v.v. Trong một số trường hợp, ý nghĩa được hiểu theo cách khác nhau, do đó sáng chế không bị giới hạn ở đó. 'Đơn vị' được sử dụng làm ý nghĩa để đề cập đến đơn vị cơ bản của quá trình xử lý hình ảnh hoặc vị trí cụ thể của ảnh, và đề cập đến vùng hình ảnh bao gồm cả thành phần độ chói và thành phần sắc độ. Ngoài ra, 'khối' đề cập đến vùng hình ảnh bao gồm một thành phần cụ thể trong số các thành phần độ chói và các thành phần sắc độ (nghĩa là, Cb và Cr). Tuy nhiên, tùy thuộc vào phương án, các thuật ngữ chẳng hạn như 'đơn vị', 'khối', 'phân vùng' và 'vùng' có thể được sử dụng thay thế lẫn nhau. Ngoài ra, trong bản mô tả này, đơn vị có thể được sử dụng làm khái niệm bao gồm tất cả trong số đơn vị mã hóa, đơn vị dự đoán và đơn vị biến đổi. Ảnh biểu thị trường hoặc khung, và theo một phương án, các thuật ngữ có thể được sử dụng thay thế lẫn nhau.

FIG.1 là sơ đồ khối sơ lược của thiết bị mã hóa tín hiệu video theo một phương án của sáng chế. Dựa vào FIG.1, thiết bị mã hóa 100 theo sáng chế này bao gồm bộ phận biến đổi 110, bộ phận lượng tử hóa 115, bộ phận lượng tử hóa ngược 120, bộ phận biến đổi ngược 125, bộ phận lọc 130, bộ phận dự đoán 150 và bộ phận mã hóa entropy 160.

Bộ phận biến đổi 110 thu giá trị của hệ số biến đổi bằng cách biến đổi tín hiệu dư, là chênh lệch giữa tín hiệu video được nhập vào và tín hiệu được dự đoán được tạo ra bởi bộ phận dự đoán 150. Ví dụ, phép biến đổi cosin rời rạc (Discrete Cosine Transform, DCT), phép biến đổi sin rời rạc (Discrete Sine Transform, DST) hoặc phép biến đổi Wavelet có thể được sử dụng. DCT và DST thực hiện bước biến đổi bằng cách phân chia tín hiệu hình ảnh đầu vào thành các khối. Trong phép biến đổi, hiệu quả mã hóa có thể thay đổi theo sự phân bố và các đặc điểm của các giá trị ở vùng biến đổi. Bộ phận lượng tử hóa 115 lượng tử hóa giá trị của hệ số biến đổi được kết xuất từ bộ phận biến đổi 110.

Để cải thiện hiệu quả mã hóa, thay vì mã hóa nguyên trạng tín hiệu hình ảnh, phương pháp dự đoán ảnh bằng cách sử dụng vùng đã được mã hóa thông qua bộ phận dự đoán 150 và thu ảnh được khôi phục bằng cách thêm giá trị dư giữa ảnh gốc và ảnh được dự đoán vào ảnh được dự đoán này được sử dụng. Để ngăn chặn sự không khớp trong

bộ mã hóa và bộ giải mã, thông tin mà có thể được sử dụng trong bộ giải mã cần được sử dụng khi thực hiện phép dự đoán trong bộ mã hóa. Để thực hiện điều này, bộ mã hóa thực hiện lại quy trình khôi phục khôi hiện thời được mã hóa. Bộ phận lượng tử hóa ngược 120 lượng tử hóa ngược giá trị của hệ số biến đổi, và bộ phận biến đổi ngược 125 khôi phục giá trị dư sử dụng giá trị hệ số biến đổi được lượng tử hóa ngược. Trong khi đó, bộ phận lọc 130 thực hiện các hoạt động lọc để cải thiện chất lượng của ảnh được khôi phục và cải thiện hiệu quả mã hóa. Ví dụ, có thể bao gồm bộ lọc khử khối, độ lệch thích ứng mẫu (sample adaptive offset, SAO), và bộ lọc vòng lặp thích ứng. Ảnh được lọc được kết xuất hoặc lưu trữ trong bộ đệm hình ảnh được giải mã (decoded picture buffer, DPB) 156 để sử dụng làm hình ảnh tham chiếu.

Để cải thiện hiệu quả mã hóa, tín hiệu ảnh không được mã hóa, mà phương pháp dự đoán hình ảnh thông qua bộ phận dự đoán 150 bằng cách sử dụng vùng đã được mã hóa, và thêm, vào ảnh được dự đoán, giá trị dư giữa ảnh gốc và ảnh được dự đoán, nhờ đó thu hình ảnh được khôi phục. Bộ phận dự đoán nội ảnh 152 thực hiện bước dự đoán nội ảnh trong ảnh hiện thời, và bộ phận dự đoán liên ảnh 154 dự đoán ảnh hiện thời bằng cách sử dụng ảnh tham chiếu được lưu trữ trong bộ đệm hình ảnh giải mã 156. Bộ phận dự đoán nội ảnh 152 thực hiện bước dự đoán nội ảnh từ các vùng được khôi phục trong ảnh hiện thời, và truyền thông tin mã hóa nội ảnh đến bộ phận mã hóa entropy 160. Bộ phận dự đoán liên ảnh 154 có thể bao gồm bộ phận đánh giá chuyển động 154a và bộ phận bù chuyển động 154b. Bộ phận đánh giá chuyển động 154a thu giá trị vectơ chuyển động của vùng hiện thời nhờ dựa vào vùng được khôi phục cụ thể. Bộ phận đánh giá chuyển động 154a truyền thông tin vị trí (khung tham chiếu, vectơ chuyển động, v.v.) của vùng tham chiếu đến bộ phận mã hóa entropy 160 để cho phép thông tin vị trí có trong dòng bit. Bộ phận bù chuyển động 154b thực hiện phép bù chuyển động liên ảnh bằng cách sử dụng giá trị vectơ chuyển động được truyền từ bộ phận đánh giá chuyển động 154a.

Bộ phận dự đoán 150 bao gồm bộ phận dự đoán nội ảnh 152 và bộ phận dự đoán liên ảnh 154. Bộ phận dự đoán nội ảnh 152 thực hiện bước dự đoán nội ảnh trong ảnh hiện thời, và bộ phận dự đoán liên ảnh 154 thực hiện bước dự đoán liên ảnh để dự đoán ảnh hiện thời bằng cách sử dụng ảnh tham chiếu được lưu trữ trong DPB 156. Bộ phận dự đoán nội ảnh 152 thực hiện bước dự đoán nội ảnh từ các mẫu được khôi phục trong ảnh hiện thời, và truyền thông tin mã hóa nội ảnh đến bộ phận mã hóa entropy 160. Thông tin mã hóa nội ảnh này có thể gồm ít nhất một trong số chế độ dự đoán nội ảnh, cờ chế độ dự đoán đúng nhất (most probable mode, MPM), và chỉ số MPM. Thông tin mã hóa nội ảnh có thể bao gồm thông tin về mẫu tham chiếu. Bộ phận dự đoán liên ảnh 154 có thể bao

gồm bộ phận đánh giá chuyển động 154a và bộ phận bù chuyển động 154b. Bộ phận đánh giá chuyển động 154a thu giá trị vectơ chuyển động của vùng hiện thời bằng cách dựa vào vùng cụ thể của ảnh tham chiếu được khôi phục. Bộ phận đánh giá chuyển động 154a truyền tập thông tin chuyển động (chỉ số ảnh tham chiếu, thông tin vectơ chuyển động, v.v.) đối với vùng tham chiếu đến bộ phận mã hóa entropy 160. Bộ phận bù chuyển động 154b thực hiện phép bù chuyển động bằng cách sử dụng giá trị vectơ chuyển động được truyền từ bộ phận đánh giá chuyển động 154a. Bộ phận dự đoán liên ảnh 154 truyền thông tin mã hóa liên ảnh bao gồm thông tin chuyển động ở vùng tham chiếu đến bộ phận mã hóa entropy 160.

Theo một phương án bổ sung, bộ phận dự đoán 150 có thể bao gồm đơn vị dự đoán sao chép nội khối (block copy, BC) (không được thể hiện trên hình vẽ). Đơn vị dự đoán BC nội bộ thực hiện phép dự đoán BC nội bộ dựa vào các mẫu được khôi phục trong ảnh hiện thời, và truyền thông tin mã hóa BC nội bộ đến bộ phận mã hóa entropy 160. Đơn vị dự đoán BC nội bộ thu giá trị vectơ khối biểu thị khu vực tham chiếu được sử dụng để dự đoán khu vực hiện thời dựa vào khu vực cụ thể trong ảnh hiện thời. Đơn vị dự đoán BC nội bộ có thể thực hiện phép dự đoán BC nội bộ bằng cách sử dụng giá trị vectơ khối thu được. Đơn vị dự đoán BC nội bộ này truyền thông tin mã hóa BC nội bộ đến bộ phận mã hóa entropy 160. Thông tin mã hóa BC nội bộ có thể bao gồm thông tin vectơ khối.

Khi phép dự đoán ảnh được mô tả ở trên được thực hiện, thì bộ phận biến đổi 110 biến đổi giá trị dư giữa ảnh gốc và ảnh được dự đoán để thu được giá trị hệ số biến đổi. Trong trường hợp này, phép biến đổi có thể được thực hiện ở đơn vị khối cụ thể nằm trong ảnh, và kích thước của khối cụ thể này có thể được thay đổi nằm trong khoảng được thiết lập trước. Bộ phận lượng tử hóa 115 lượng tử hóa giá trị hệ số biến đổi được tạo ra trong bộ phận biến đổi 110 và truyền giá trị này đến bộ phận mã hóa entropy 160.

Bộ phận mã hóa entropy 160 mã hóa entropy thông tin biểu thị hệ số biến đổi được lượng tử hóa, thông tin mã hóa nội ảnh, thông tin mã hóa liên ảnh và thông tin tương tự để tạo ra dòng bit tín hiệu video. Trong bộ phận mã hóa entropy 160, sơ đồ mã hóa theo chiều dài biến đổi (variable length coding, VLC), sơ đồ mã hóa số học, v.v. có thể được sử dụng. Sơ đồ mã hóa theo chiều dài biến đổi (VLC) bao gồm các ký hiệu đầu vào biến đổi thành các từ mã liên tiếp, và chiều dài của từ mã có thể biến đổi. Ví dụ, các ký hiệu xảy ra thường xuyên được biểu diễn bởi từ mã ngắn, và các ký hiệu xảy ra không thường xuyên được biểu diễn bởi từ mã dài. Sơ đồ mã hóa theo chiều dài biến đổi thích ứng dựa trên ngữ cảnh (context-based adaptive variable length coding, CAVLC) có thể được sử dụng làm sơ đồ mã hóa theo chiều dài biến đổi. Bước mã hóa số học có thể biến đổi các ký hiệu

dữ liệu liên tục thành số nguyên tố duy nhất, trong đó bước mã hóa số học có thể thu bit tối ưu được yêu cầu để biểu diễn mỗi ký hiệu. Mã hóa số học nhị phân thích ứng dựa trên ngữ cảnh (context-based adaptive binary arithmetic code, CABAC) có thể được sử dụng làm mã hóa số học. Ví dụ, bộ phận mã hóa entropy 160 có thể nhị phân hóa thông tin biểu thị hệ số biến đổi được lượng tử hóa. Bộ phận mã hóa entropy 160 này có thể tạo ra dòng bit bằng cách mã hóa số học thông tin nhị phân.

Dòng bit được tạo ra được đóng gói bằng cách sử dụng đơn vị lớp trừu tượng mạng (network abstraction layer, NAL) làm đơn vị cơ bản. Đơn vị NAL bao gồm số đơn vị cây mã hóa được mã hóa nguyên. Để giải mã dòng bit trong bộ giải mã video, trước tiên, dòng bit phải được tách rời trong các đơn vị NAL, và sau đó mỗi đơn vị NAL được tách rời này phải được giải mã. Trong khi đó, thông tin cần thiết để giải mã dòng bit tín hiệu video có thể được truyền thông qua tập hợp mức phía trên của tải tin trình tự byte thô (Raw Byte Sequence Payload, RBSP) chẳng hạn như tập tham số hình ảnh (Picture Parameter Set, PPS), tập tham số chuỗi (Sequence Parameter Set, SPS), tập tham số video (Video Parameter Set, VPS) và dạng tương tự.

Trong khi đó, sơ đồ khối trên FIG.1 thể hiện thiết bị mã hóa 100 theo một phương án của sáng chế, và các khối được hiển thị riêng biệt được phân biệt và thể hiện hợp lý các phần tử của thiết bị mã hóa 100. Theo đó, các phần tử của thiết bị mã hóa 100 được mô tả ở trên có thể được lắp dưới dạng một chip hoặc nhiều chip tùy thuộc vào thiết kế của thiết bị. Theo một phương án, hoạt động của mỗi phần tử của thiết bị mã hóa 100 được mô tả ở trên có thể được thực hiện bởi bộ xử lý (không được thể hiện trên hình vẽ).

FIG.2 là sơ đồ khối sơ lược của thiết bị giải mã tín hiệu video 200 theo một phương án của sáng chế. Dựa vào FIG.2, thiết bị giải mã 200 theo sáng chế này bao gồm bộ phận giải mã entropy 210, bộ phận lượng tử hóa ngược 220, bộ phận biến đổi ngược 225, bộ phận lọc 230 và bộ phận dự đoán 250.

Bộ phận giải mã entropy 210 giải mã entropy dòng bit tín hiệu video để trích xuất thông tin hệ số biến đổi, thông tin mã hóa nội ảnh, thông tin mã hóa liên ảnh và thông tin tương tự cho mỗi vùng. Ví dụ, bộ phận giải mã entropy 210 có thể thu mã nhị phân hóa cho thông tin hệ số biến đổi của vùng cụ thể từ dòng bit tín hiệu video. Bộ phận giải mã entropy 210 thu hệ số biến đổi được lượng tử hóa bằng cách nhị phân hóa ngược mã nhị phân. Bộ phận lượng tử hóa ngược 220 lượng tử hóa ngược hệ số biến đổi được lượng tử hóa, và bộ phận biến đổi ngược 225 khôi phục giá trị dư bằng cách sử dụng hệ số biến đổi ngược được lượng tử hóa. Thiết bị xử lý tín hiệu video 200 khôi phục giá trị điểm ảnh gốc

bằng cách tính tổng giá trị dư thu được bởi bộ phận biến đổi ngược 225 bằng giá trị dự đoán thu được bởi bộ phận dự đoán 250.

Trong khi đó, bộ phận lọc 230 thực hiện bước lọc đối với ảnh để cải thiện chất lượng hình ảnh. Bộ phận lọc này có thể bao gồm bộ lọc khử khối để giảm sự biến dạng khối và/hoặc bộ lọc vòng lặp thích ứng để loại bỏ sự biến dạng của toàn bộ ảnh. Ảnh đã lọc được kết xuất hoặc lưu trữ trong DPB 256 để sử dụng làm ảnh tham chiếu cho ảnh tiếp theo.

Bộ phận dự đoán 250 bao gồm bộ phận dự đoán nội ảnh 252 và bộ phận dự đoán liên ảnh 254. Bộ phận dự đoán 250 tạo ra ảnh dự đoán bằng cách sử dụng kiểu mã hóa được giải mã thông bộ phận giải mã entropy 210 được mô tả ở trên, các hệ số biến đổi cho mỗi vùng, và thông tin mã hóa nội ảnh/liên ảnh. Để khôi phục khối hiện thời trong đó bước giải mã được thực hiện, thì vùng được giải mã của ảnh hiện thời hoặc các ảnh khác bao gồm khối hiện thời có thể được sử dụng. Trong quá trình khôi phục, chỉ ảnh hiện thời, nghĩa là, ảnh (hoặc ô/lát) mà thực hiện bước dự đoán nội ảnh hoặc dự đoán BC nội bộ, được gọi là nội bộ ảnh hoặc ảnh I (hoặc ô/lát), và ảnh (hoặc ô/lát) mà có thể thực hiện tất cả trong số phép dự đoán nội ảnh, dự đoán liên ảnh và dự đoán BC nội bộ được gọi là liên ảnh (hoặc ô/lát). Để dự đoán các giá trị mẫu của mỗi khối trong số các nội bộ ảnh (hoặc các ô/các lát), ảnh (hoặc ô/lát) sử dụng tối đa một vectơ chuyển động và chỉ số ảnh tham chiếu được gọi là ảnh dự đoán hoặc ảnh P (hoặc ô/lát), và ảnh (hoặc ô/lát) sử dụng tối đa hai vectơ chuyển động và chỉ số ảnh tham chiếu được gọi là ảnh dự đoán kép hoặc ảnh B (hoặc ô/lát). Nói cách khác, ảnh P (hoặc ô/lát) sử dụng tối đa một tập thông tin chuyển động để dự đoán mỗi khối, và ảnh B (hoặc ô/lát) sử dụng tối đa hai tập thông tin chuyển động để dự đoán mỗi khối. Ở đây, tập thông tin chuyển động bao gồm một hoặc nhiều vectơ chuyển động và một chỉ số ảnh tham chiếu.

Bộ phận dự đoán nội ảnh 252 tạo ra khối dự đoán bằng cách sử dụng thông tin mã hóa nội ảnh và các mẫu được khôi phục trong ảnh hiện thời. Như được mô tả ở trên, thông tin mã hóa nội ảnh có thể gồm ít nhất một trong số chế độ dự đoán nội ảnh, cờ chế độ dự đoán đúng nhất (MPM) và chỉ số MPM. Bộ phận dự đoán nội ảnh 252 dự đoán các giá trị mẫu của khối hiện thời bằng cách sử dụng các mẫu được khôi phục nằm ở phía bên trái và/hoặc phía trên của khối hiện thời làm các mẫu tham chiếu. Theo sáng chế này, các mẫu được khôi phục, các mẫu tham chiếu, và các mẫu của khối hiện thời có thể là các điểm ảnh. Ngoài ra, các giá trị mẫu có thể thể hiện các giá trị điểm ảnh.

Theo một phương án, các mẫu tham chiếu có thể là các mẫu có trong khối lân cận

của khối hiện thời. Ví dụ, các mẫu tham chiếu có thể là các mẫu liền kề với biên bên trái của khối hiện thời và/hoặc các mẫu có thể là các mẫu liền kề với biên phía trên. Ngoài ra, các mẫu tham chiếu có thể là các mẫu nằm ở một đường nằm trong khoảng cách định trước từ biên bên trái của khối hiện thời và/hoặc các mẫu nằm ở đường trong khoảng cách định trước từ biên phía trên của khối hiện thời trong số các mẫu của các khối lân cận của khối hiện thời. Trong trường hợp này, khối lân cận của khối hiện thời có thể bao gồm khối bên trái (L), khối phía trên (A), khối bên trái phía dưới (BL), khối bên phải phía trên (AR) hoặc khối bên trái phía trên (AL).

Bộ phận dự đoán liên ảnh 254 tạo ra khối dự đoán bằng cách sử dụng các ảnh tham chiếu và thông tin mã hóa liên ảnh được lưu trữ trong DPB 256. Thông tin mã hóa liên ảnh có thể bao gồm tập thông tin chuyển động (chỉ số ảnh tham chiếu, thông tin vectơ chuyển động, v.v.) của khối hiện thời cho khối tham chiếu này. Phép dự đoán liên ảnh có thể bao gồm phép dự đoán L0, phép dự đoán L1 và phép dự đoán kép. Phép dự đoán L0 nghĩa là phép dự đoán sử dụng một ảnh tham chiếu có trong danh sách ảnh L0, và phép dự đoán L1 nghĩa là phép dự đoán sử dụng một ảnh tham chiếu có trong danh sách ảnh L1. Đối với việc này, có thể yêu cầu một tập thông tin chuyển động (ví dụ, vectơ chuyển động và chỉ số ảnh tham chiếu). Theo phương pháp dự đoán kép, tối đa hai vùng tham chiếu có thể được sử dụng, và hai vùng tham chiếu này có thể có mặt trong cùng ảnh tham chiếu hoặc có thể có mặt trong các ảnh khác nhau. Nghĩa là, theo phương pháp dự đoán kép, tối đa hai tập thông tin chuyển động (ví dụ, vectơ chuyển động và chỉ số ảnh tham chiếu) có thể được sử dụng và hai vectơ chuyển động có thể tương ứng với cùng chỉ số ảnh tham chiếu hoặc các chỉ số ảnh tham chiếu khác nhau. Trong trường hợp này, các ảnh tham chiếu đều có thể được hiển thị (hoặc được kết xuất) trước và sau ảnh hiện thời theo khía cạnh thời gian. Theo một phương án, hai vùng tham chiếu được sử dụng trong sơ đồ dự đoán kép có thể là các vùng lần lượt được chọn từ danh sách ảnh L0 và danh sách ảnh L1.

Bộ phận dự đoán liên ảnh 254 có thể thu khối tham chiếu của khối hiện thời bằng cách sử dụng vectơ chuyển động và chỉ số ảnh tham chiếu. Khối tham chiếu trong ảnh tham chiếu tương ứng với chỉ số ảnh tham chiếu. Ngoài ra, giá trị mẫu của khối được chỉ định bởi vectơ chuyển động hoặc giá trị được nội suy của nó có thể được sử dụng làm bộ dự đoán của khối hiện thời. Đối với phép dự đoán chuyển động với độ chính xác đơn vị điểm ảnh phụ, ví dụ, bộ lọc nội suy 8-tap đối với tín hiệu độ chói và bộ lọc nội suy 4-tap cho tín hiệu sắc độ có thể được sử dụng. Tuy nhiên, bộ lọc nội suy dự đoán chuyển động trong các đơn vị điểm ảnh phụ không bị giới hạn ở đó. Bằng cách này, bộ phận dự đoán

liên ảnh 254 thực hiện phép bù chuyển động để dự đoán kết cấu của đơn vị hiện thời từ các ảnh chuyển động được khôi phục trước đó. Trong trường hợp này, đơn vị dự đoán liên ảnh có thể sử dụng tập thông tin chuyển động.

Theo một phương án bổ sung, bộ phận dự đoán 250 có thể bao gồm đơn vị dự đoán BC nội bộ (không được thể hiện trên hình vẽ). Đơn vị dự đoán BC nội bộ có thể khôi phục vùng hiện thời nhờ tham chiếu đến vùng cụ thể bao gồm các mẫu được khôi phục trong ảnh hiện thời. Đơn vị dự đoán BC nội bộ thu thông tin mã hóa BC nội bộ đối với vùng hiện thời từ bộ phận giải mã entropy 210. Đơn vị dự đoán BC nội bộ thu giá trị vectơ khối của vùng hiện thời biểu thị vùng cụ thể trong ảnh hiện thời. Đơn vị dự đoán BC nội bộ này có thể thực hiện phép dự đoán BC nội bộ bằng cách sử dụng giá trị vectơ khối thu được. Thông tin mã hóa BC nội bộ có thể bao gồm thông tin vectơ khối.

Ảnh video được khôi phục được tạo ra bằng cách thêm giá trị dự đoán được kết xuất từ bộ phận dự đoán nội ảnh 252 hoặc bộ phận dự đoán liên ảnh 254 và giá trị dư được kết xuất từ bộ phận biến đổi ngược 225. Nghĩa là, thiết bị giải mã tín hiệu video 200 khôi phục khối hiện thời bằng cách sử dụng khối dự đoán được tạo ra bởi bộ phận dự đoán 250 và giá trị dư thu được từ bộ phận biến đổi ngược 225.

Trong khi đó, sơ đồ khối trên Fig.2 thể hiện thiết bị giải mã 200 Theo một phương án của sáng chế, và các khối được hiển thị riêng biệt được phân biệt và thể hiện hợp lý các phần tử của thiết bị giải mã 200. Theo đó, các phần tử của thiết bị giải mã 200 được mô tả ở trên có thể được lắp dưới dạng một chip hoặc nhiều chip tùy thuộc vào thiết kế của thiết bị. Theo một phương án, hoạt động của mỗi phần tử của thiết bị giải mã 200 được mô tả ở trên có thể được thực hiện bởi bộ xử lý (không được thể hiện trên hình vẽ).

FIG.3 minh họa một phương án trong đó đơn vị cây mã hóa (coding tree unit, CTU) được phân chia thành các đơn vị mã hóa (coding unit, CU) trong ảnh. Trong quy trình mã hóa tín hiệu video, ảnh có thể được phân chia thành chuỗi các đơn vị cây mã hóa (CTU). Đơn vị cây mã hóa bao gồm khối NXN của các mẫu độ chói và hai khối của các mẫu độ sắc độ tương ứng với nó. Đơn vị cây mã hóa có thể được phân chia thành nhiều đơn vị mã hóa. Đơn vị cây mã hóa không được phân chia và có thể là nút lá. Trong trường hợp này, chính đơn vị cây mã hóa có thể là đơn vị mã hóa. Đơn vị mã hóa này đề cập đến đơn vị cơ bản để xử lý ảnh trong quy trình xử lý tín hiệu video được mô tả ở trên, nghĩa là, dự đoán nội ảnh/liên ảnh, biến đổi, lượng tử hóa, và/hoặc mã hóa entropy. Kích thước và hình dạng của đơn vị mã hóa trong một ảnh có thể không phải là hằng số. Đơn vị mã hóa có thể có dạng hình vuông hoặc hình chữ nhật. Đơn vị mã hóa hình chữ nhật (hoặc

khối hình chữ nhật) bao gồm đơn vị mã hóa dọc (hoặc khối dọc) và đơn vị mã hóa ngang (hoặc khối ngang). Trong bản mô tả này, khối dọc là khối có chiều cao lớn hơn chiều rộng, và khối ngang là khối có chiều rộng lớn hơn chiều cao. Ngoài ra, trong bản mô tả này, khối không vuông có thể đề cập đến khối hình chữ nhật, nhưng sáng chế không bị giới hạn ở đó.

Dựa vào FIG.3, trước tiên đơn vị cây mã hóa được phân chia thành cấu trúc cây tứ phân (quad tree, QT). Nghĩa là, một nút có kích thước $2N \times 2N$ trong cấu trúc cây tứ phân có thể được phân chia thành bốn nút có kích thước $N \times N$. Trong bản mô tả này, cây tứ phân cũng có thể được gọi là cây bậc bốn. Việc phân chia cây tứ phân có thể được thực hiện theo cách đệ quy, và không phải tất cả các nút đều cần được phân chia với cùng độ sâu.

Trong khi đó, nút lá của cây tứ phân được mô tả ở trên có thể còn được phân chia thành cấu trúc cây đa kiểu (multi-type tree, MTT). Theo một phương án của sáng chế, theo cấu trúc cây đa kiểu, một nút có thể được phân chia thành cấu trúc cây nhị phân hoặc tam phân theo sự phân chia ngang hoặc dọc. Nghĩa là, theo cấu trúc cây đa kiểu, có bốn cấu trúc phân chia chẳng hạn như phân chia nhị phân dọc, phân chia nhị phân ngang, phân chia tam phân dọc và phân chia tam phân ngang. Theo một phương án của sáng chế, trong mỗi trong số các cấu trúc cây, chiều rộng và chiều cao của các nút đều có thể là lũy thừa cơ số 2. Ví dụ, trong cấu trúc cây nhị phân (binary tree, BT), nút có kích thước $2N \times 2N$ có thể được phân chia thành hai nút $N \times 2N$ bằng cách phân chia nhị phân dọc, và được phân chia thành hai nút $2N \times N$ bằng cách phân chia nhị phân ngang. Ngoài ra, trong cấu trúc cây tam phân (TT), nút có kích thước $2N \times 2N$ được phân chia thành các nút $(N/2) \times 2N$, $N \times 2N$ và $(N/2) \times N$ bằng cách phân chia tam phân dọc, và được phân chia thành các nút $2N \times (N/2)$, $2N \times N$ và $2N \times (N/2)$ bằng cách phân chia tam phân ngang. Việc phân chia cây đa kiểu này có thể được thực hiện theo cách đệ quy.

Nút lá của cây đa kiểu có thể là đơn vị mã hóa. Nếu việc phân chia đối với đơn vị mã hóa này không được biểu thị hoặc đơn vị mã hóa không lớn đối với chiều dài biến đổi tối đa, thì đơn vị mã hóa được sử dụng làm đơn vị dự đoán và biến đổi mà không cần phân chia thêm. Mặt khác, ít nhất một tham số trong số các tham số sau trong cây tứ phân và cây đa kiểu được mô tả ở trên có thể được xác định hoặc truyền thông qua tập hợp mức cao của các RBSP chẳng hạn như PPS, SPS, VPS và dạng tương tự. 1) kích thước CTU: kích thước nút gốc của cây tứ phân, 2) kích thước QT tối thiểu MinQtSize: kích thước nút lá QT được cho phép tối thiểu, 3) kích thước BT tối đa MaxBtSize: kích thước nút gốc BT được cho phép tối đa, 4) kích thước TT tối đa MaxTtSize: kích thước nút gốc TT được

cho phép tối đa, 5) Độ sâu MTT tối đa `MaxMttDepth`: độ sâu được cho phép tối đa của MTT được phân chia từ nút lá của QT, 6) Kích thước BT tối thiểu `MinBtSize`: kích thước nút lá BT được cho phép tối thiểu, 7) Kích thước TT tối thiểu `MinTtSize`: kích thước nút lá TT được cho phép tối thiểu.

FIG.4 thể hiện một phương án về phương pháp báo hiệu việc phân chia cây tứ phân và cây đa kiểu. Các cờ được thiết lập trước có thể được sử dụng để báo hiệu việc phân chia cây tứ phân và cây đa kiểu được mô tả ở trên. Dựa vào FIG.4, ít nhất một cờ trong số cờ '`qt_split_flag`' biểu thị liệu có phân chia nút cây tứ phân hay không, cờ '`mtt_split_flag`' biểu thị liệu có phân chia nút cây đa kiểu hay không, cờ '`mtt_split_vertical_flag`' biểu thị hướng phân chia của nút cây đa kiểu hoặc cờ '`mtt_split_binary_flag`' biểu thị hình dạng phân chia của nút cây đa kiểu có thể được sử dụng.

Theo một phương án của sáng chế, đơn vị cây mã hóa là nút gốc của cây tứ phân, và trước tiên có thể được phân chia thành cấu trúc cây tứ phân. Trong cấu trúc cây tứ phân này, '`qt_split_flag`' được báo hiệu cho mỗi nút '`QT_node`'. Nếu giá trị của '`qt_split_flag`' là 1, thì nút được phân chia thành 4 nút hình vuông, và nếu giá trị của '`qt_split_flag`' là 0, thì nút tương ứng trở thành nút lá '`QT_leaf_node`' của cây tứ phân.

Mỗi nút lá cây tứ phân '`QT_leaf_node`' có thể còn được phân chia thành cấu trúc cây đa kiểu. Trong cấu trúc cây đa kiểu này, '`mtt_split_flag`' được báo hiệu cho mỗi nút '`MTT_node`'. Khi giá trị của '`mtt_split_flag`' là 1, thì nút tương ứng được phân chia thành nhiều nút hình chữ nhật, và khi giá trị của '`mtt_split_flag`' là 0, thì nút tương ứng là nút lá '`MTT_leaf_node`' của cây đa kiểu. Khi nút cây đa kiểu '`MTT_node`' được phân chia thành nhiều nút hình chữ nhật (nghĩa là, khi giá trị của '`mtt_split_flag`' là 1), '`mtt_split_vertical_flag`' và '`mtt_split_binary_flag`' đối với nút '`MTT_node`' có thể được báo hiệu bổ sung. Khi giá trị của '`mtt_split_vertical_flag`' là 1, việc phân chia dọc của nút '`MTT_node`' được biểu thị, và khi giá trị của '`mtt_split_vertical_flag`' là 0, việc phân chia ngang của nút '`MTT_node`' được biểu thị. Ngoài ra, khi giá trị của '`mtt_split_binary_flag`' là 1, thì nút '`MTT_node`' được phân chia thành 2 nút hình chữ nhật, và khi giá trị của '`mtt_split_binary_flag`' là 0, thì nút '`MTT_node`' được phân chia thành 3 nút hình chữ nhật.

Phép dự đoán ảnh (bù chuyển động) để mã hóa được thực hiện đối với đơn vị mã hóa mà không còn được phân chia (nghĩa là, nút lá của cây đơn vị mã hóa). Sau đây, đơn vị cơ bản để thực hiện bước dự đoán này sẽ được gọi là "đơn vị dự đoán" hoặc "khối dự đoán".

Sau đây, thuật ngữ "đơn vị" được sử dụng ở đây có thể thay thế cho đơn vị dự đoán, là đơn vị cơ bản để thực hiện bước dự đoán. Tuy nhiên, sáng chế không bị giới hạn ở đó, và "đơn vị" có thể được hiểu là khái niệm rộng bao gồm đơn vị mã hóa.

FIG.5 minh họa phép dự đoán liên ảnh theo một phương án của sáng chế. Như được mô tả ở trên, bộ giải mã có thể dự đoán khối hiện thời dựa vào các mẫu được khôi phục của ảnh được giải mã khác. Dựa vào FIG.5, bộ giải mã thu được khối tham chiếu 42 trong ảnh tham chiếu dựa vào thông tin chuyển động của khối hiện thời 32. Trong trường hợp này, thông tin chuyển động có thể bao gồm chỉ số ảnh tham chiếu và vectơ chuyển động 50. Chỉ số ảnh tham chiếu này biểu thị ảnh tham chiếu của khối hiện thời trong danh sách ảnh tham chiếu. Ngoài ra, vectơ chuyển động 50 biểu thị độ lệch giữa giá trị tọa độ của khối hiện thời 32 trong ảnh hiện thời và giá trị tọa độ của khối tham chiếu 42 trong ảnh tham chiếu. Bộ giải mã thu được bộ dự đoán của khối hiện thời 32 dựa vào các giá trị mẫu của khối tham chiếu 42 và khôi phục khối hiện thời 32 bằng cách sử dụng bộ dự đoán này.

Trong khi đó, theo một phương án của sáng chế, phép bù chuyển động dựa trên khối phụ có thể được sử dụng. Nghĩa là, khối hiện thời 32 có thể được chia thành nhiều khối phụ, và vectơ chuyển động độc lập có thể được sử dụng cho mỗi khối phụ này. Theo đó, các khối phụ tương ứng có thể được dự đoán bằng cách sử dụng các khối tham chiếu khác nhau nằm trong khối hiện thời 32. Theo một phương án, khối phụ có thể có kích thước được thiết lập trước chẳng hạn như 4X4 hoặc 8X8. Bộ giải mã thu được bộ dự đoán của mỗi khối phụ của khối hiện thời 32 thông qua vectơ chuyển động của mỗi khối phụ. Thông qua sự kết hợp của các bộ dự đoán của các khối phụ tương ứng, có thể thu được bộ dự đoán của khối hiện thời 32, và bộ giải mã có thể khôi phục khối hiện thời 32 bằng cách sử dụng bộ dự đoán thu được của khối hiện thời 32.

Theo một phương án của sáng chế, phép bù chuyển động dựa trên khối phụ thông qua các phương pháp khác nhau có thể được thực hiện. Phép bù chuyển động dựa trên khối phụ này có thể bao gồm phép bù chuyển động dựa trên mô hình afin (sau đây, được gọi là phép bù chuyển động afin hoặc phép dự đoán chuyển động afin) và phép dự đoán vectơ chuyển động theo thời gian dựa trên khối phụ (Subblock-based Temporal Motion Vector Prediction, SbTMVP). Sau đây, các phương án khác nhau về phép bù chuyển động afin và SbTMVP được mô tả dựa vào các hình vẽ.

FIG.6 minh họa phương pháp báo hiệu vectơ chuyển động theo một phương án của sáng chế. Theo một phương án của sáng chế, vectơ chuyển động (Motion Vector, MV) có

thể được tạo ra dựa vào phép dự đoán vectơ chuyển động (Motion Vector Prediction, MVP) (hoặc Bộ dự đoán). Ví dụ, MV có thể được xác định bởi MVP như được thể hiện trên phương trình 1 dưới đây. Nói cách khác, MV có thể được xác định (được tạo cấu hình hoặc tạo ra) dưới dạng giá trị giống với MVP.

Phương trình 1

$$MV = MVP$$

Trong một ví dụ khác, MV có thể được xác định dựa vào MVP và chênh lệch vectơ chuyển động (Motion Vector Difference, MVD) như được thể hiện trên phương trình 2 dưới đây. Bộ mã hóa có thể báo hiệu thông tin MVD cho bộ giải mã để biểu thị MV một cách chính xác hơn, và bộ giải mã có thể tạo ra MV bằng cách thêm MVD thu được vào MVP.

Phương trình 2

$$MV = MVP + MVD$$

Theo một phương án của sáng chế, bộ mã hóa có thể truyền thông tin chuyển động được xác định đến bộ giải mã, và bộ giải mã này có thể tạo ra MV từ thông tin chuyển động nhận được và tạo ra (hoặc sinh ra) khối được dự đoán trên cơ sở này. Ví dụ, thông tin chuyển động có thể bao gồm thông tin MVP và thông tin MVD. Trong trường hợp này, các phần tử của thông tin chuyển động có thể thay đổi tùy thuộc vào chế độ dự đoán liên ảnh. Ví dụ, thông tin chuyển động có thể có hoặc có thể không bao gồm thông tin MVP trong chế độ hợp nhất. Theo một ví dụ khác, thông tin chuyển động có thể bao gồm thông tin MVP và thông tin MVD trong chế độ dự đoán vectơ chuyển động nâng cao (Advanced Motion vector prediction, AMVP).

Để xác định, truyền, và nhận thông tin về MVP, bộ mã hóa và bộ giải mã có thể tạo ra các ứng viên MVP (hoặc danh sách ứng viên MVP) theo cùng cách. Ví dụ, bộ mã hóa và bộ giải mã có thể tạo ra các ứng viên MVP giống nhau theo cùng thứ tự. Ngoài ra, bộ mã hóa có thể truyền chỉ số biểu thị (hoặc biểu diễn) MVP đã xác định (hoặc đã chọn) trong số các ứng viên MVP đã tạo ra đến bộ giải mã, và bộ giải mã này có thể tạo ra MVP và/hoặc MV đã xác định dựa vào chỉ số nhận được.

Theo một phương án của sáng chế, các ứng viên MVP có thể bao gồm ứng viên theo không gian, ứng viên theo thời gian và ứng viên tương tự. Các ứng viên MVP có thể được gọi là các ứng viên hợp nhất khi chế độ hợp nhất được áp dụng, và có thể được gọi

là các ứng viên AMVP khi chế độ AMVP được áp dụng. Ứng viên theo không gian có thể là MV (hoặc thông tin chuyển động) đối với khối được bố trí ở một vị trí cụ thể từ khối hiện thời. Ví dụ, ứng viên theo không gian có thể là MV của khối liền kề hoặc không liền kề với khối hiện thời. Ứng viên theo thời gian có thể là MV tương ứng với khối nằm trong ảnh khác với ảnh hiện thời. Ngoài ra, ví dụ, các ứng viên MVP có thể bao gồm MV afin, ATMVP, STMVP, sự kết hợp của các MV (hoặc các ứng viên), MV trung bình của các MV (hoặc các ứng viên), MV bằng không và dạng tương tự.

Theo một phương án, bộ mã hóa có thể báo hiệu thông tin biểu thị ảnh tham chiếu cho bộ giải mã. Theo một phương án, khi ảnh tham chiếu của các ứng viên MVP khác với ảnh tham chiếu của khối hiện thời (hoặc khối được xử lý hiện thời), thì bộ mã hóa/bộ giải mã có thể thay đổi tỷ lệ MV (thay đổi tỷ lệ vectơ chuyển động) của các ứng viên MVP. Trong trường hợp này, việc thay đổi tỷ lệ MV có thể được thực hiện dựa vào số đếm thứ tự hình ảnh (Picture Order Count, POC) của ảnh hiện thời, POC của ảnh tham chiếu của khối hiện thời, và POC của ảnh tham chiếu của MVP ứng viên.

Một phương án chi tiết về phương pháp báo hiệu MVD được mô tả sau đây. Bảng 1 dưới đây thể hiện cấu trúc cú pháp đối với báo hiệu MVD.

Bảng 1

	Từ khóa mô tả
mvd_coding(x0, y0, refList) {	
abs_mvd_greater0_flag[0]	ae(v)
abs_mvd_greater0_flag[1]	ae(v)
if(abs_mvd_greater0_flag[0])	
abs_mvd_greater1_flag[0]	ae(v)
if(abs_mvd_greater0_flag[1])	
abs_mvd_greater1_flag[1]	ae(v)
if(abs_mvd_greater0_flag[0]) {	
if(abs_mvd_greater1_flag[0])	
abs_mvd_minus2[0]	ae(v)
mvd_sign_flag[0]	ae(v)
}	
if(abs_mvd_greater0_flag[1]) {	
if(abs_mvd_greater1_flag[1])	
abs_mvd_minus2[1]	ae(v)
mvd_sign_flag[1]	ae(v)
}	
}	

Dựa vào bảng 1, dấu và giá trị tuyệt đối của MVD có thể được mã hóa riêng biệt theo một phương án của sáng chế. Nghĩa là, dấu và giá trị tuyệt đối của MVD có thể có cú

pháp (hoặc các phần tử cú pháp) khác nhau. Ngoài ra, giá trị tuyệt đối của MVD có thể được mã hóa trực tiếp hoặc có thể được mã hóa dựa vào một cờ biểu thị liệu giá trị tuyệt đối có lớn hơn N hay không như được thể hiện trên bảng 1. Nếu giá trị tuyệt đối này lớn hơn N, thì giá trị của (giá trị tuyệt đối - N) cũng có thể được báo hiệu. Cụ thể, trong ví dụ về bảng 1, `abs_mvd_greater0_flag` biểu thị liệu giá trị tuyệt đối có lớn hơn 0 hay không có thể được truyền. Nếu `abs_mvd_greater0_flag` biểu thị (biểu diễn) rằng giá trị tuyệt đối không lớn hơn 0, thì giá trị tuyệt đối của MVD có thể được xác định là 0. Ngoài ra, nếu `abs_mvd_greater0_flag` biểu thị rằng giá trị tuyệt đối lớn hơn 0, thì có thể có mặt cú pháp bổ sung (hoặc phần tử cú pháp).

Ví dụ, `abs_mvd_greater1_flag` biểu thị rằng giá trị tuyệt đối lớn hơn 1 có thể được truyền. Nếu `abs_mvd_greater1_flag` biểu thị (biểu diễn) rằng giá trị tuyệt đối không lớn hơn 1, thì giá trị tuyệt đối của MVD có thể được xác định là 1. Nếu `abs_mvd_greater1_flag` biểu thị rằng giá trị tuyệt đối lớn hơn 1, thì có thể có mặt cú pháp bổ sung. Ví dụ, có thể có mặt `abs_mvd_minus2`. `abs_mvd_minus2` có thể là giá trị của (giá trị tuyệt đối - 2). Vì xác định được rằng giá trị tuyệt đối lớn hơn 1 (nghĩa là, lớn hơn hoặc bằng 2) theo các giá trị của `abs_mvd_greater0_flag` và `abs_mvd_greater1_flag`, nên giá trị của (giá trị tuyệt đối - 2) có thể được báo hiệu. Việc báo hiệu cú pháp thực hiện theo cách phân cấp đối với thông tin về giá trị tuyệt đối có thể sử dụng số lượng bit nhỏ hơn so với việc nhị phân hóa và báo hiệu giá trị tuyệt đối này.

Theo một phương án, cú pháp liên quan đến giá trị tuyệt đối có thể được mã hóa bằng áp dụng phương pháp nhị phân hóa theo chiều dài biến đổi chẳng hạn như phương pháp Golomb-hàm mũ, phương pháp đơn phân được lược bỏ, phương pháp rice được lược bỏ hoặc dạng tương tự. Ngoài ra, cờ biểu thị dấu của MVD có thể được báo hiệu thông qua `mvd_sign_flag`.

Mặc dù phương pháp mã hóa MVD đã được mô tả theo phương án được mô tả ở trên, nhưng dấu và giá trị tuyệt đối của thông tin cũng như MVD có thể được báo hiệu riêng biệt. Giá trị tuyệt đối có thể được mã hóa đối với cờ biểu thị liệu giá trị tuyệt đối này lớn hơn giá trị cụ thể định trước hay giá trị thu được bằng cách trừ giá trị cụ thể cho giá trị tuyệt đối. Trên bảng 1, [0] và [1] có thể biểu thị các chỉ số thành phần. Ví dụ, [0] và [1] có thể là thành phần x (nghĩa là, thành phần ngang) và thành phần y (nghĩa là thành phần dọc).

FIG.7 minh họa phương pháp báo hiệu thông tin độ phân giải vectơ chuyển động thích ứng theo một phương án của sáng chế. Theo một phương án của sáng chế, độ phân

giải biểu thị MV hoặc MVD có thể thay đổi. Ví dụ, độ phân giải có thể được biểu thị dựa vào điểm ảnh (hoặc pel). Ví dụ, MV hoặc MVD có thể được báo hiệu trong các đơn vị của $1/4$ (một phần tư), $1/2$ (một nửa), 1 (số nguyên), 2 hoặc 4 điểm ảnh. Bộ mã hóa có thể báo hiệu thông tin độ phân giải của MV hoặc MVD cho bộ giải mã. Ngoài ra, ví dụ, 16 có thể được mã hóa thành 64 trong các đơn vị của $1/4$ ($1/4 * 64 = 16$), thành 16 trong các đơn vị của 1 ($1 * 16 = 16$) và thành 4 trong các đơn vị của 4 ($4 * 4 = 16$). Nghĩa là, giá trị MV hoặc MVD có thể được xác định bằng cách sử dụng phương trình 3 dưới đây.

Phương trình 3

$$\text{valueDetermined} = \text{resolution} * \text{valuePerResolution}$$

Trong phương trình 3, valueDetermined biểu thị giá trị MV hoặc MVD. Ngoài ra, valuePerResolution là giá trị được báo hiệu dựa vào độ phân giải được xác định. Trong trường hợp này, khi giá trị được báo hiệu bởi MV hoặc MVD không thể được chia bởi độ phân giải được xác định, thì quy trình làm tròn và quy trình tương tự có thể được áp dụng. Khi độ phân giải cao được sử dụng, thì độ chính xác có thể tăng lên nhưng giá trị được mã hóa là lớn và do đó số lượng bit lớn có thể được sử dụng. Khi độ phân giải thấp được sử dụng, thì độ chính xác có thể giảm xuống nhưng giá trị được mã hóa là nhỏ và do đó số lượng bit nhỏ có thể được sử dụng. Theo một phương án, độ phân giải có thể được tạo cấu hình theo cách khác nhau trong các đơn vị của các chuỗi, ảnh, lát, đơn vị cây mã hóa (CTU), đơn vị mã hóa (CU) và dạng tương tự. Nghĩa là, bộ mã hóa/bộ giải mã có thể xác định/áp dụng một cách thích ứng độ phân giải theo đơn vị được xác định trước trong số các đơn vị được mô tả ở trên.

Theo một phương án của sáng chế, thông tin độ phân giải có thể được báo hiệu từ bộ mã hóa đến bộ giải mã. Trong trường hợp này, thông tin độ phân giải này có thể được nhị phân hóa và báo hiệu dựa vào chiều dài biến đổi. Trong trường hợp này, tổng phí báo hiệu có thể được giảm bớt khi việc báo hiệu được thực hiện dựa vào chỉ số tương ứng với giá trị nhỏ nhất (nghĩa là giá trị quan trọng nhất). Theo một phương án, độ phân giải có thể được ánh xạ để báo hiệu các chỉ số theo thứ tự từ độ phân giải cao đến độ phân giải thấp.

Theo một phương án của sáng chế, FIG.7 minh họa phương pháp báo hiệu khi giả sử rằng ba độ phân giải được sử dụng trong số các độ phân giải khác nhau. Trong trường hợp này, ba bit báo hiệu có thể là 0, 10 và 11, và ba chỉ số báo hiệu có thể được gọi là độ phân giải thứ nhất, độ phân giải thứ hai và độ phân giải thứ ba. Vì 1 bit cần báo hiệu độ

phân giải thứ nhất và 2 bit cần báo hiệu các độ phân giải còn lại, nên tổng phí báo hiệu có thể được giảm bớt tương đối khi độ phân giải thứ nhất được báo hiệu. Trong một ví dụ trên Fig.7, độ phân giải thứ nhất, độ phân giải thứ hai và độ phân giải thứ ba có thể lần lượt được xác định là độ phân giải 1/4, 1 và 4 điểm ảnh. Theo các phương án sau, độ phân giải MV có thể là độ phân giải của MVD.

Sau đây, phương pháp bù chuyển động dựa vào chế độ hợp nhất sử dụng chênh lệch về vectơ chuyển động (chế độ hợp nhất với MVD (MMVD)) (hoặc MVD hợp nhất) sẽ được mô tả.

FIG.8 minh họa một ví dụ về cú pháp của đơn vị mã hóa theo một phương án của sáng chế. Theo một phương án của sáng chế, phần tử cú pháp biểu thị liệu MMVD có được áp dụng hay không có thể được báo hiệu dựa vào phần tử cú pháp biểu thị liệu chế độ hợp nhất có được áp dụng hay không. Dựa vào FIG.8, ở bước S801, cờ MMVD (`mmvd_flag`) có thể được báo hiệu khi cờ hợp nhất (`merge_flag`) bằng 0 (nghĩa là, trong trường hợp trong đó chế độ hợp nhất không được sử dụng). Trên FIG.8, cờ MMVD biểu thị phần tử cú pháp (hoặc cờ) biểu thị liệu MMVD có được áp dụng hay không. Ngoài ra, cờ hợp nhất biểu thị phần tử cú pháp (hoặc cờ) biểu thị liệu chế độ hợp nhất có được áp dụng hay không.

Theo một phương án của sáng chế, bộ mã hóa/bộ giải mã có thể xác định vectơ chuyển động (motion vector, MV) dựa vào bộ dự đoán vectơ chuyển động (motion vector predictor, MVP) và chênh lệch về vectơ chuyển động (motion vector difference, MVD) khi chế độ hợp nhất được áp dụng. Trong bản mô tả này, MVP có thể được gọi là vectơ chuyển động cơ sở (base motion vector, baseMV). Nghĩa là, bộ mã hóa/bộ giải mã có thể thu được vectơ chuyển động (nghĩa là, vectơ chuyển động cuối cùng) bằng cách tính tổng vectơ chuyển động cơ sở với chênh lệch về vectơ chuyển động. Tuy nhiên, sáng chế này không bị giới hạn ở các tên gọi này, và MVP có thể được gọi là vectơ chuyển động cơ sở, vectơ chuyển động theo thời gian, vectơ chuyển động ban đầu, vectơ chuyển động ứng viên MMVD và dạng tương tự. MVD có thể được biểu diễn dưới dạng giá trị để lọc MVP, và có thể được gọi là vectơ chuyển động được lọc (`refineMV`) hoặc chênh lệch về vectơ chuyển động hợp nhất.

Theo một phương án của sáng chế, khi MMVD được áp dụng, nghĩa là, ở chế độ MMVD, MV có thể được xác định dựa vào vectơ chuyển động cơ sở, tham số khoảng cách (hoặc biến số) hoặc tham số hướng (hoặc biến số). Hơn nữa, theo một phương án của sáng chế, vectơ chuyển động cơ sở có thể được xác định từ danh sách ứng viên. Ví

dụ, vectơ chuyển động cơ sở có thể được xác định từ danh sách ứng viên hợp nhất. Bộ mã hóa/bộ giải mã có thể xác định vectơ chuyển động cơ sở từ một số trong số danh sách ứng viên khác. Một số trong số danh sách ứng viên có thể là phần trước của danh sách ứng viên (phần có chỉ số nhỏ hơn). Ví dụ, bộ mã hóa/bộ giải mã có thể xác định vectơ chuyển động cơ sở bằng cách sử dụng các ứng viên thứ nhất và thứ hai trong số các ứng viên của danh sách ứng viên hợp nhất. Cuối cùng, chỉ số ứng viên biểu thị ứng viên cụ thể trong số hai ứng viên có thể được báo hiệu từ bộ mã hóa đến bộ giải mã. Dựa vào FIG.21, chỉ số ứng viên cơ sở là chỉ số để báo hiệu vectơ chuyển động cơ sở có thể được xác định. Bộ mã hóa/bộ giải mã này có thể xác định ứng viên được áp dụng cho khối hiện thời trong số các ứng viên của danh sách ứng viên theo chỉ số ứng viên cơ sở và xác định vectơ chuyển động của ứng viên đã xác định làm vectơ chuyển động cơ sở. Theo sáng chế này, chỉ số ứng viên cơ sở không bị giới hạn ở tên gọi này, và có thể được gọi là cờ ứng viên cơ sở, chỉ số ứng viên, cờ ứng viên, chỉ số MMVD, chỉ số ứng viên MMVD, cờ ứng viên MMVD hoặc dạng tương tự.

Theo một phương án của sáng chế, có thể có MVD khác với MVD được mô tả trên FIG.6 và FIG.7. Ví dụ, MVD trong MMVD có thể được xác định theo cách khác nhau từ MVD được mô tả trên FIG.6 và FIG.7. Trong bản mô tả này, MMVD có thể biểu thị chế độ hợp nhất bằng cách sử dụng chênh lệch về vectơ chuyển động (nghĩa là, chế độ hoặc phương pháp bù chuyển động) hoặc biểu thị chênh lệch về vectơ chuyển động trong trường hợp trong đó MMVD được áp dụng. Ví dụ, bộ mã hóa/bộ giải mã có thể xác định liệu có áp dụng (hoặc sử dụng) MMVD hay không. Nếu MMVD được áp dụng, thì bộ mã hóa/bộ giải mã có thể tạo ra ứng viên hợp nhất được sử dụng để dự đoán liên ảnh khối hiện thời từ danh sách ứng viên hợp nhất và xác định vectơ chuyển động của khối hiện thời bằng cách tạo ra MMVD và áp dụng (hoặc thêm) MMVD vào vectơ chuyển động của ứng viên hợp nhất.

Theo một phương án, MVD khác nhau có thể là MVD được đơn giản hóa, MVD có độ phân giải khác nhau (hoặc nhỏ), các MVD có số lượng sử dụng được nhỏ, các MVD có các phương pháp báo hiệu khác nhau hoặc dạng tương tự. Ví dụ, trong khi MVD được sử dụng trong AMVP thông thường, chế độ liên afin hoặc dạng tương tự được mô tả trên FIG.6 và FIG.7 có thể biểu thị tất cả các vùng theo các trục x và y (nghĩa là, các hướng nằm ngang và thẳng đứng), ví dụ, các vùng dựa vào các ảnh (ví dụ, các vùng ảnh hoặc các vùng bao gồm các ảnh và các vùng bao quanh) ở các khoảng đồng đều đối với đơn vị tín hiệu cụ thể (ví dụ, x-pel), MMVD có thể có đơn vị tương đối giới hạn của báo hiệu cụ thể. Hơn nữa, các vùng (hoặc các đơn vị) để báo hiệu MMVD có thể không có các

khoảng đồng đều. MMVD chỉ có thể biểu thị hướng cụ thể đối với đơn vị báo hiệu cụ thể.

Theo một phương án của sáng chế, MMVD có thể được xác định dựa vào khoảng cách và hướng. Khoảng cách và hướng của MMVD theo chỉ số khoảng cách biểu thị khoảng cách của MMVD và chỉ số hướng biểu thị hướng của MMVD có thể được thiết lập trước. Theo một phương án, khoảng cách có thể biểu thị kích thước MMVD (ví dụ, giá trị tuyệt đối) trong các đơn vị của các điểm ảnh cụ thể, và hướng có thể biểu thị hướng của MMVD. Bộ mã hóa/bộ giải mã có thể báo hiệu khoảng cách tương đối nhỏ thông qua chỉ số tương đối nhỏ. Nghĩa là, nếu báo hiệu không phải là báo hiệu thông qua phép nhị phân hóa theo chiều dài cố định, thì bộ mã hóa/bộ giải mã có thể báo hiệu khoảng cách tương đối nhỏ thông qua các bit tương đối nhỏ.

Theo một phương án của sáng chế, các phần tử cú pháp liên quan đến MMVD có thể được báo hiệu khi cờ hợp nhất (nghĩa là, `merge_flag`) bằng 0 (nghĩa là, trong trường hợp trong đó chế độ hợp nhất không được sử dụng). Như được mô tả ở trên, MMVD có thể là phương pháp báo hiệu MVD đối với ứng viên cơ sở. Về khía cạnh này, chế độ MMVD có thể có các điểm tương đồng với các chế độ chẳng hạn như AMVP báo hiệu MVD và AMVP afin (hoặc liên afin). Theo đó, khi cờ hợp nhất là 0, thì báo hiệu có thể được thực hiện. Ở bước S802, khi MMVD được áp dụng cho khối hiện thời, nghĩa là, khi cờ MMVD là 1, thì bộ giải mã có thể phân tích cú pháp các phần tử cú pháp liên quan đến MMVD. Theo một phương án, phần tử cú pháp liên quan đến MMVD có thể bao gồm ít nhất một trong số `mmvd_merge_flag`, `mmvd_distance_idx` và `mmvd_direction_idx`. Ở đây, `mmvd_merge_flag` biểu thị cờ (hoặc phần tử cú pháp) biểu thị ứng viên cơ sở của MMVD, `mmvd_distance_idx` biểu thị chỉ số (hoặc phần tử cú pháp) biểu thị khoảng cách giá trị của MVD và `mmvd_direction_idx` biểu thị chỉ số (hoặc phần tử cú pháp) biểu thị hướng của MVD.

Hơn nữa, dựa vào FIG.8, `CuPredMode` biểu thị biến số (hoặc giá trị) biểu thị chế độ dự đoán của khối hiện thời. Theo cách khác, chế độ dự đoán của khối hiện thời có thể là giá trị biểu thị liệu khối hiện thời được dự đoán nội ảnh hay dự đoán liên ảnh. Theo cách khác, chế độ dự đoán của khối hiện thời có thể được xác định dựa vào `pred_mode_flag`. Ở đây, `pred_mode_flag` biểu thị phần tử cú pháp biểu thị liệu khối hiện thời được mã hóa ở chế độ dự đoán liên ảnh hay chế độ dự đoán nội ảnh. Nếu `pred_mode_flag` bằng 0, thì chế độ dự đoán của khối hiện thời có thể được thiết lập thành giá trị biểu thị rằng phép dự đoán liên ảnh được sử dụng. Giá trị của chế độ dự đoán biểu thị rằng phép dự đoán liên ảnh được sử dụng có thể là `MODE_INTER`. Nếu `pred_mode_flag` bằng 1, thì chế độ dự đoán của khối hiện thời có thể được thiết lập thành

giá trị biểu thị rằng phép dự đoán nội ảnh được sử dụng. Giá trị của chế độ dự đoán biểu thị rằng phép dự đoán nội ảnh được sử dụng có thể là `MODE_INTRA`. Nếu `pred_mode_flag` không có mặt, thì `CuPredMode` có thể được thiết lập thành giá trị được thiết lập trước. Hơn nữa, một ví dụ là, giá trị được thiết lập trước có thể là `MODE_INTRA`.

Hơn nữa, dựa vào FIG.8, `cu_cbf` có thể là giá trị biểu thị liệu cú pháp liên quan đến phép biến đổi có có mặt hay không. Cú pháp liên quan đến phép biến đổi có thể là cấu trúc cú pháp cây biến đổi. Hơn nữa, cú pháp liên quan đến phép biến đổi có thể là cú pháp được báo hiệu thông qua cây biến đổi (`transform_tree`) trên FIG.28. Hơn nữa, khi `cu_cbf` bằng 0, thì cú pháp liên quan đến phép biến đổi có thể không có mặt. Khi `cu_cbf` bằng 1, thì cú pháp liên quan đến phép biến đổi có thể có mặt. Dựa vào FIG.28, ở bước S803, khi `cu_cbf` bằng 1, thì bộ giải mã có thể gọi ra cú pháp cây biến đổi. Nếu `cu_cbf` không có mặt, thì giá trị `cu_cbf` có thể được xác định dựa vào `cu_skip_flag`. Ví dụ, khi `cu_skip_flag` là 1, thì `cu_cbf` có thể bằng 0. Hơn nữa, khi `cu_skip_flag` là 0, thì `cu_cbf` có thể bằng 1. Như được mô tả ở trên, `cu_skip_flag` biểu thị phần tử cú pháp biểu thị liệu chế độ bỏ qua có được sử dụng hay không. Ngoài ra, khi chế độ bỏ qua được áp dụng, thì tín hiệu dư có thể không được sử dụng. Nghĩa là, chế độ bỏ qua có thể là chế độ trong đó việc khôi phục xảy ra mà không thêm phần dư vào tín hiệu dự đoán. Theo đó, chỉ báo rằng `cu_skip_flag` là 1 có thể biểu thị rằng cú pháp liên quan đến phép biến đổi không có mặt.

Theo một phương án của sáng chế, ở bước S802, bộ giải mã có thể phân tích cú pháp `cu_cbf` khi phép dự đoán nội ảnh không được sử dụng. Hơn nữa, khi `cu_skip_flag` là 0, thì `cu_cbf` có thể được phân tích cú pháp. Hơn nữa, bộ giải mã có thể phân tích cú pháp `cu_cbf` khi cờ hợp nhất là 0. Hơn nữa, các điều kiện này có thể được kết hợp và áp dụng. Ví dụ, bộ giải mã có thể phân tích cú pháp `cu_cbf` khi chế độ dự đoán của khối hiện thời không phải là chế độ dự đoán nội ảnh và cờ hợp nhất là 0. Theo cách khác, bộ giải mã có thể phân tích cú pháp `cu_cbf` khi chế độ dự đoán của khối hiện thời là chế độ dự đoán liên ảnh và cờ hợp nhất là 0. Điều này có thể xảy ra vì chế độ bỏ qua có thể được sử dụng hoặc không phải trong trường hợp đối với phép dự đoán liên chế độ không hợp nhất.

FIG.9 minh họa một ví dụ về cú pháp của đơn vị mã hóa theo một phương án của sáng chế. Theo một phương án của sáng chế, `cu_cbf` và cú pháp liên quan đến phép biến đổi trên FIG.8 được mô tả ở trên có thể được thay đổi thành cú pháp được thể hiện trên FIG.9. Nghĩa là, theo một phương án của sáng chế, khi chế độ cụ thể được áp dụng, thì việc liệu có sử dụng chế độ bỏ qua hay không có thể được xác định. Ví dụ, khi MMVD được áp dụng, thì việc liệu có sử dụng chế độ bỏ qua hay không có thể được xác định.

Theo một phương án, dựa vào FIG.9, ở bước S901, bộ giải mã có thể xác định liệu có phân tích cú pháp `cu_cbf` hay không dựa vào việc liệu MMVD có được áp dụng hay không. Nghĩa là, vì việc sử dụng chế độ bỏ qua được xác định theo việc liệu MMVD có được áp dụng hay không, nên việc xác định liệu có phân tích cú pháp `cu_cbf` hay không có thể được thực hiện phù hợp. Khi rõ ràng việc liệu có sử dụng chế độ bỏ qua hay không, thì bộ giải mã có thể không phân tích cú pháp `cu_cbf`.

Theo một phương án, trong trường hợp sử dụng MMVD, bộ mã hóa/bộ giải mã có thể không sử dụng chế độ bỏ qua. Vì MVD không thể được chỉ ra một cách chính xác giống như AMVP và có thể chỉ được chỉ ra ở khoảng giới hạn như được mô tả ở trên, nên MMVD có thể được khôi phục một cách chính xác bằng cách sử dụng phần dư. Theo đó, bằng cách xác định liệu có phân tích cú pháp `cu_cbf` hay không dựa vào việc liệu MMVD có được sử dụng hay không, độ chính xác dự đoán và hiệu quả nén có thể được tăng lên. Ví dụ, trong trường hợp sử dụng MMVD, bộ giải mã có thể không phân tích cú pháp `cu_cbf`. Nếu MMVD không được sử dụng, thì bộ giải mã có thể phân tích cú pháp `cu_cbf`. Ở bước S901, bộ giải mã có thể phân tích cú pháp `cu_cbf` khi cờ MMVD là 0, và có thể không phân tích cú pháp `cu_cbf` khi cờ MMVD là 1.

Theo một phương án của sáng chế, khi `cu_cbf` không có mặt, thì bộ giải mã có thể suy ra giá trị `cu_cbf`. Theo phương pháp được mô tả dựa vào FIG.8, bộ giải mã có thể suy ra `cu_cbf` dựa vào giá trị `cu_skip_flag`. Theo một phương án của sáng chế, giá trị `cu_cbf` có thể được suy ra dựa vào cờ hợp nhất. Nếu cờ hợp nhất là 0, thì bộ giải mã có thể suy ra giá trị `cu_cbf` là 1. Một ví dụ là, khi chế độ hợp nhất không được sử dụng, thì có thể biểu thị rằng cú pháp liên quan đến phép biến đổi có mặt. Theo đó, trong trường hợp sử dụng MMVD theo các phương án trên các hình vẽ từ FIG.8 đến FIG.9, bộ giải mã có thể suy ra giá trị `cu_cbf` là 1. Theo một phương án, bộ giải mã có thể suy ra 1) giá trị `cu_cbf` là 0 khi cờ hợp nhất là 1 và `cu_skip_flag` là 1, 2) giá trị `cu_cbf` là 1 khi cờ hợp nhất là 1 và `cu_skip_flag` là 0 và 3) giá trị `cu_cbf` là 1 khi cờ hợp nhất là 0. Theo cách khác, theo một phương án như được thể hiện trên FIG.8, bộ giải mã có thể suy ra 1) giá trị `cu_cbf` là 0 khi `cu_skip_flag` là 1 và 2) giá trị `cu_cbf` là 1 khi `cu_skip_flag` là 0.

Theo một phương án khác của sáng chế, giá trị `cu_cbf` có thể được suy ra dựa vào cờ MMVD. Khi cờ MMVD là 1, thì `cu_cbf` có thể được suy ra là 1. Hơn nữa, khi cờ MMVD là 0, thì `cu_cbf` có thể được suy ra là 0 hoặc 1. Khi được kết hợp với phương pháp suy luận được mô tả trên FIG.28, 1) khi cờ MMVD là 1, thì giá trị `cu_cbf` có thể được suy ra là 1, 2) khi cờ MMVD là 0 và `cu_skip_flag` là 1, thì giá trị `cu_cbf` có thể được suy ra là 0 và 3) khi cờ MMVD là 0 và `cu_skip_flag` là 0, thì giá trị `cu_cbf` có thể được

suy ra là 1.

Hơn nữa, theo một phương án, theo các phương án trên các hình vẽ từ FIG.8 đến FIG.9, cờ MMVD có thể không có mặt ở cú pháp dữ liệu hợp nhất (merge_data). Cú pháp dữ liệu hợp nhất này có thể là cú pháp dữ liệu hợp nhất được thể hiện trên FIG.8 và FIG.9.

FIG.10 minh họa một ví dụ về phương pháp báo hiệu chế độ hợp nhất theo một phương án của sáng chế. Theo một phương án của sáng chế, chế độ hợp nhất có thể được báo hiệu dựa vào phần tử cú pháp như được thể hiện trên FIG.10. Dựa vào FIG.10, chế độ hợp nhất được báo hiệu dựa vào ít nhất một cờ trong số cờ chính quy, cờ MMVD, cờ khối phụ, và/hoặc cờ hợp nhất liên ảnh và dự đoán nội ảnh được kết hợp (combined inter-picture merge and intra-picture prediction, CIIP). Theo sáng chế này, CIIP biểu thị phương pháp dự đoán kết hợp phép dự đoán liên ảnh (ví dụ, phép dự đoán liên chế độ hợp nhất) và phép dự đoán nội ảnh, và có thể được gọi là phép dự đoán đa giả thuyết.

Dựa vào FIG.10, các bảng (a) và (b) trên FIG.10 có thể biểu thị các trường hợp lần lượt tương ứng với chế độ hợp nhất không bỏ qua và chế độ hợp nhất bỏ qua. Không giống như cú pháp dữ liệu hợp nhất trên FIG.8 và FIG.9 được mô tả ở trên, cờ chính quy có thể có mặt theo một phương án trên FIG.10. Ví dụ, cờ hình tam giác có thể không có mặt. Cờ chính quy có thể là phần tử cú pháp biểu thị có sử dụng chế độ hợp nhất thông thường, và theo sáng chế này, cờ chính quy có thể được gọi là cờ hợp nhất chính quy. Chế độ hợp nhất thông thường có thể là cùng chế độ hợp nhất với chế độ được sử dụng trong HEVC. Hơn nữa, chế độ hợp nhất thông thường có thể là chế độ hợp nhất trong đó ứng viên được biểu thị bởi chỉ số hợp nhất được sử dụng và phép bù chuyển động được thực hiện mà không sử dụng MVD. Theo một phương án, cờ chính quy, cờ MMVD, cờ khối phụ và cờ CIIP có thể được báo hiệu theo trình tự được thiết lập trước. Cờ MMVD biểu thị phần tử cú pháp biểu thị liệu MMVD có được sử dụng hay không. Cờ khối phụ biểu thị phần tử cú pháp biểu thị liệu chế độ khối phụ, trong đó phép dự đoán dựa trên khối phụ được thực hiện, có được sử dụng hay không. Cờ CIIP biểu thị phần tử cú pháp biểu thị liệu chế độ CIIP có được áp dụng hay không.

Theo một phương án của sáng chế, giá trị của báo hiệu để biểu thị liệu có sử dụng chế độ tương ứng hay không trong số cờ chính quy, cờ MMVD, cờ khối phụ và cờ CIIP có thể nhỏ hơn hoặc bằng 1. Theo đó, khi một cờ trong số cờ chính quy, cờ MMVD, cờ khối phụ và cờ CIIP có giá trị bằng 1, thì bộ mã hóa/bộ giải mã có thể xác định rằng các giá trị của các cờ có được sau đó theo trình tự giải mã bằng 0. Hơn nữa, khi cờ chính quy,

cờ MMVD, cờ khối phụ và cờ CIIP đều bằng 0, thì chế độ không được biểu thị bởi cờ chính quy, cờ MMVD, cờ khối phụ hoặc cờ CIIP có thể được sử dụng. Chế độ không được biểu thị bởi cờ chính quy, cờ MMVD, cờ khối phụ và cờ CIIP có thể là phép dự đoán hình tam giác. Nghĩa là, theo một phương án, khi cờ chính quy, cờ MMVD, cờ khối phụ và cờ CIIP đều bằng 0, thì có thể xác định được rằng chế độ dự đoán hình tam giác được áp dụng.

FIG.11 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. FIG.11 minh họa một ví dụ về cấu trúc cú pháp trong đó các phần tử cú pháp được mô tả dựa vào FIG.10 được sử dụng. Trên FIG.11, “cờ hợp nhất chính quy” đề cập đến cờ hợp nhất chính quy. Một ví dụ là, cờ hợp nhất chính quy có thể là cờ chính quy được mô tả ở trên dựa vào FIG.10.

Theo một phương án của sáng chế, cờ hợp nhất chính quy có thể đứng trước trong cú pháp dữ liệu hợp nhất. Ở bước S1101, bộ giải mã trước tiên có thể phân tích cú pháp cờ hợp nhất chính quy theo cú pháp dữ liệu hợp nhất. Nghĩa là, cờ hợp nhất chính quy có thể là phần tử cú pháp thứ nhất mà được phân tích cú pháp sau khi xác định rằng cờ hợp nhất là 1. Ở bước S1102, bộ giải mã có thể phân tích cú pháp cờ MMVD khi cờ hợp nhất chính quy là 0. Ở các bước S1103, S1106 và S1105, khi cờ hợp nhất chính quy là 0, thì bộ giải mã có thể phân tích cú pháp ít nhất một cờ trong số cờ hợp nhất khối phụ, cờ đa giả thuyết và/hoặc cờ hợp nhất hình tam giác. Trên FIG.11, merge_subblock_flag biểu thị cờ hợp nhất khối phụ biểu thị liệu chế độ hợp nhất khối phụ có được áp dụng hay không, mh_intra_flag biểu thị cờ dự đoán đa giả thuyết biểu thị liệu chế độ dự đoán đa giả thuyết có được áp dụng hay không và merge_triangle_flag biểu thị cờ hợp nhất hình tam giác biểu thị liệu chế độ hợp nhất hình chữ nhật có được áp dụng hay không.

Dựa vào FIG.11, khi khối hiện thời thỏa mãn điều kiện về kích thước khối cụ thể được xác định trước, thì bộ giải mã có thể phân tích cú pháp cờ MMVD. Hơn nữa, theo một phương án, cờ hợp nhất hình tam giác có thể được xác định dưới dạng (!regular_merge_flag && !MMVD_flag && !merge_subblock_flag && !mh_intra_flag). Nghĩa là, cờ hợp nhất hình tam giác là 1 khi tất cả trong số cờ hợp nhất chính quy, cờ MMVD, cờ hợp nhất khối phụ và cờ dự đoán đa giả thuyết là 0, và khi ít nhất một cờ trong số cờ hợp nhất chính quy, cờ MMVD, cờ hợp nhất khối phụ và mh_intra_flag là 1, thì cờ hợp nhất hình tam giác có thể là 0.

FIG.12 minh họa cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. FIG.12 minh họa cấu trúc cú pháp trong đó các phần tử cú pháp được mô tả ở trên dựa

vào FIG.10 được sử dụng. Dựa vào FIG.12, theo một phương án của sáng chế, bộ giải mã trước tiên có thể phân tích cú pháp cờ hợp nhất chính quy, và có thể phân tích cú pháp chỉ số hợp nhất khi cờ hợp nhất chính quy đã phân tích cú pháp là 1 (S1201). Hơn nữa, bộ giải mã có thể phân tích cú pháp chỉ số hợp nhất khi MaxNumMergeCand lớn hơn 1. Ở đây, MaxNumMergeCand là biến số biểu thị số lượng ứng viên hợp nhất tối đa. Hơn nữa, khi cờ hợp nhất chính quy là 0, thì bộ giải mã có thể phân tích cú pháp ít nhất một cờ trong số cờ MMVD, cờ hợp nhất khối phụ, cờ đa giả thuyết và/hoặc cờ hợp nhất hình tam giác. Theo một phương án, giá trị của cờ hợp nhất hình tam giác có thể được xác định bằng phương pháp được mô tả ở trên dựa vào FIG.10 và FIG.11. Nghĩa là, cờ hợp nhất hình tam giác có thể được xác định dựa vào giá trị cờ biểu thị liệu chế độ khác có được áp dụng hay không. Nếu cờ hợp nhất hình tam giác là 1, thì bộ giải mã có thể phân tích cú pháp cú pháp (hoặc phần tử cú pháp) liên quan đến phép dự đoán hình tam giác. Ví dụ, khi cờ hợp nhất hình tam giác là 1, thì bộ giải mã có thể phân tích cú pháp chỉ số hợp nhất hình tam giác (nghĩa là, merge_triangle_idx).

Trong trường hợp theo một phương án trên FIG.11, chỉ số hợp nhất được yêu cầu cho chế độ hợp nhất chính quy có mặt ở phía sau của cú pháp dữ liệu hợp nhất, và vì báo hiệu cờ MMVD, cờ hợp nhất khối phụ, cờ dự đoán đa giả thuyết, ví dụ, có thể có mặt giữa cờ hợp nhất chính quy và chỉ số hợp nhất, nên báo hiệu có thể vô hiệu khi chế độ hợp nhất chính quy được sử dụng. Tuy nhiên, theo một phương án trên FIG.12, khi cờ hợp nhất chính quy là 1, thì chỉ số hợp nhất có thể được phân tích cú pháp ngay sau cờ hợp nhất chính quy, và do đó không cần phải phân tích cú pháp báo hiệu khác liên quan đến chế độ hợp nhất chính quy. Theo đó, hiệu quả nén có thể được cải thiện.

Hơn nữa, theo một phương án của sáng chế, khi nhiều chế độ dự đoán khác nhau được sử dụng, thì việc liệu có phân tích cú pháp chế độ dự đoán cụ thể hay không có thể được xác định dựa vào điều kiện áp dụng chế độ dự đoán. Điều này sẽ được mô tả dựa vào bảng 2 dưới đây.

Bảng 2

```
If(A1 && A2 && A3 )
    mode_A_flag
    If(mode_A_flag) {
        // mode A related syntax elements
    } else {
        if(B1 && B2 && B3 )
```

```

mode_B_flag
if(mode_B_flag) {
    // mode B related syntax elements
} else {
    // mode C related syntax elements
}
}

```

Dựa vào bảng 2, chế độ A, chế độ B và chế độ C được giả sử là có mặt dưới dạng các chế độ dự đoán. Hơn nữa, giả sử rằng chỉ một chế độ dự đoán trong số chế độ A, chế độ B và chế độ C được sử dụng để dự đoán. Ngoài ra, các điều kiện để sử dụng chế độ A có thể được xác định, và các điều kiện để sử dụng chế độ A có thể được giả sử là A1, A2 và A3. Theo phương án này, khi tất cả các điều kiện A1, A2 và A3 được thỏa mãn, thì bộ mã hóa/bộ giải mã có thể áp dụng chế độ A. Ngoài ra, giả sử rằng các điều kiện để sử dụng chế độ B có thể là B1, B2 và B3. Theo phương án này, khi tất cả các điều kiện B1, B2 và B3 được thỏa mãn, thì bộ mã hóa/bộ giải mã có thể áp dụng chế độ B. Ngoài ra, các điều kiện để sử dụng chế độ C có thể là C1, C2 và C3, và khi tất cả các điều kiện C1, C2 và C3 được thỏa mãn, thì bộ mã hóa/bộ giải mã có thể áp dụng chế độ C. Báo hiệu (hoặc phần tử cú pháp) biểu thị liệu có sử dụng chế độ dự đoán định trước X (Chế độ X) hay không có thể là `MODE_X_flag`.

Dựa vào bảng 2, bộ giải mã có thể phân tích cú pháp cú pháp có liên quan để xác định chế độ dự đoán cần được áp dụng đối với khối hiện thời theo trình tự của chế độ A, chế độ B và chế độ C. Theo cách khác, bộ mã hóa có thể thực hiện việc báo hiệu theo trình tự của `mode_A_flag`, `mode_B_flag` và `mode_C_flag`, như được thể hiện ở bảng 2. Nếu điều kiện để sử dụng chế độ A được thỏa mãn, thì bộ giải mã có thể phân tích cú pháp `mode_A_flag`. Nếu `mode_A_flag` là 1, thì bộ giải mã có thể phân tích cú pháp cú pháp liên quan đến chế độ A, và có thể không phân tích cú pháp các cờ liên quan đến các chế độ còn lại và cú pháp liên quan đến chế độ này. Nếu `mode_A_flag` là 0, thì có thể có khả năng sử dụng chế độ B hoặc chế độ C. Theo đó, nếu điều kiện để sử dụng chế độ B được thỏa mãn, thì bộ giải mã có thể phân tích cú pháp `mode_B_flag`. Nếu `mode_B_flag` là 1, thì bộ giải mã có thể phân tích cú pháp cú pháp liên quan đến chế độ B, và có thể không phân tích cú pháp `mode_X_flag` liên quan đến các chế độ còn lại (nghĩa là, chế độ C) và cú pháp liên quan đến chế độ này. Nếu `mode_B_flag` là 0, thì bộ giải mã có thể xác định có sử dụng chế độ C. Nghĩa là, nếu tất cả các `mode_X_flag` mà không tương ứng với chế độ C là 0, thì bộ giải mã có thể xác định có sử dụng chế độ C. Ngoài ra, bộ giải mã có

thể phân tích cú pháp cú pháp liên quan đến chế độ C.

Ngoài ra, theo một phương án của sáng chế, nếu nhiều chế độ dự đoán khác nhau được sử dụng, thì việc liệu có phân tích cú pháp chế độ dự đoán cụ thể hay không có thể được xác định dựa vào điều kiện áp dụng chế độ dự đoán. Điều này sẽ được mô tả dựa vào bảng 3 dưới đây.

Bảng 3

```

If((A1 && A2 && A3) && !((!B1 || !B2 || !B3) && (!C1 || !C2 || !C3)))
    mode_A_flag
    If(mode_A_flag) {
        // mode A related syntax elements
    } else {
        if(B1 && B2 && B3 )
            mode_B_flag
            if(mode_B_flag) {
                // mode B related syntax elements
            } else {
                // mode C related syntax elements
            }
        }
    }

```

Dựa vào bảng 3, như được thể hiện ở bảng 2 được mô tả ở trên, chế độ A, chế độ B và chế độ C có thể được xác định dưới dạng các chế độ dự đoán, và phần tử cú pháp (nghĩa là, mode_X_flag) biểu thị liệu có sử dụng chế độ dự đoán và/hoặc phần tử cú pháp hay không biểu thị thông tin chế độ dự đoán có liên quan có thể được xác định. Ngoài ra, các điều kiện X1, X2, X3 và điều kiện tương tự để sử dụng chế độ định trước X có thể được xác định. Như được thể hiện ở bảng 2 được mô tả ở trên, việc liệu có áp dụng tuân tự chế độ A, chế độ B và chế độ C hay không được xác định, và nếu chế độ đã xác định được áp dụng, thì phần tử cú pháp liên quan đến chế độ dự đoán tương ứng có thể được phân tích cú pháp.

Ở đây, theo một phương án của sáng chế, khi các chế độ dự đoán, việc sử dụng các chế độ này được xác định muộn hơn so với sử dụng chế độ dự đoán cụ thể, đều không sử dụng được, thì bộ mã hóa/bộ giải mã có thể xác định có sử dụng chế độ dự đoán cụ thể nêu trên. Ở đây, bộ giải mã có thể không phân tích cú pháp cờ biểu thị liệu có áp dụng chế độ dự đoán cụ thể hay không (nghĩa là, mode_X_flag khi chế độ dự đoán cụ thể là chế độ

X). Theo một phương án, tính không sử dụng được của chế độ dự đoán có thể được xác định theo việc liệu điều kiện nêu trên để sử dụng chế độ dự đoán có được thỏa mãn hay không. Ví dụ, nếu chế độ B và chế độ C, được xác định là được sử dụng ở mức độ ưu tiên tương đối thấp hơn, đều không sử dụng được, thì bộ giải mã có thể không phân tích cú pháp `mode_A_flag` và có thể xác định (hoặc quyết định hoặc suy ra) có sử dụng chế độ A.

Ở bảng 2 và bảng 3 được mô tả ở trên, giả sử trường hợp trong đó ba chế độ dự đoán của chế độ A, chế độ B và chế độ C được áp dụng. Tuy nhiên, sáng chế này không bị giới hạn ở số lượng chế độ dự đoán, và ngay cả khi nhiều chế độ dự đoán hơn tồn tại, thì chế độ này có thể được xác định bằng cách sử dụng phương pháp đã đề xuất. Ví dụ, khi chế độ A, chế độ B, chế độ C và chế độ D được giả sử là sử dụng được, nếu chế độ B, chế độ C và chế độ D đều không sử dụng được, thì bộ giải mã có thể xác định có sử dụng chế độ A mà không thực hiện báo hiệu (hoặc phân tích cú pháp) riêng biệt. Ngoài ra, sau khi xác định không sử dụng chế độ A, khi chế độ C và chế độ D đều không sử dụng được, thì bộ giải mã có thể xác định có sử dụng chế độ B.

Dựa vào bảng 3, điều kiện mà chế độ dự đoán định trước X (nghĩa là, chế độ X) là không sử dụng được có thể là trường hợp trong đó ít nhất một điều kiện trong số các điều kiện X1, X2 và X3 không được thỏa mãn. Nghĩa là, trong trường hợp đối với $!X1 \parallel !X2 \parallel !X3$, chế độ X có thể không sử dụng được. Do đó, trường hợp trong đó chế độ B và chế độ C đều không sử dụng được có thể biểu thị trường hợp thỏa mãn điều kiện $((!B1 \parallel !B2 \parallel !B3) \&\& (!C1 \parallel !C2 \parallel !C3))$. Khi điều kiện này được thỏa mãn, thì bộ giải mã có thể không phân tích cú pháp `mode_A_flag`, và có thể suy ra giá trị của `mode_A_flag` là 1. Nghĩa là, bộ giải mã có thể xác định có sử dụng chế độ A. Khi điều kiện $((!B1 \parallel !B2 \parallel !B3) \&\& (!C1 \parallel !C2 \parallel !C3))$ không được thỏa mãn, thì bộ giải mã có thể phân tích cú pháp `mode_A_flag`. Ở đây, bộ giải mã có thể còn xem xét điều kiện để sử dụng chế độ A. Nghĩa là, bộ giải mã có thể phân tích cú pháp `mode_A_flag` nếu điều kiện $!((!B1 \parallel !B2 \parallel !B3) \&\& (!C1 \parallel !C2 \parallel !C3))$ được thỏa mãn và trong trường hợp đối với $(A1 \&\& A2 \&\& A3)$. Nói cách khác, khi ít nhất một điều kiện trong số điều kiện để sử dụng chế độ B và điều kiện để sử dụng chế độ C được thỏa mãn, thì bộ giải mã có thể phân tích cú pháp `mode_A_flag`. Trong trường hợp đối với $(B1 \&\& B2 \&\& B3)$ hoặc $(C1 \&\& C2 \&\& C3)$, bộ giải mã có thể phân tích cú pháp `mode_A_flag`.

Hơn nữa, khi `mode_A_flag` không tồn tại, trong trường hợp đối với $(B1 \&\& B2 \&\& B3)$ hoặc $(C1 \&\& C2 \&\& C3)$, thì bộ giải mã có thể suy ra giá trị `mode_A_flag` là 0. Ngược lại, bộ giải mã có thể suy ra giá trị `mode_A_flag` là 1. Nghĩa là, nếu không có chế độ B và chế độ C sử dụng được, khi `mode_A_flag` không tồn tại, thì bộ giải mã có thể suy

ra giá trị mode_A_flag là 1 (nghĩa là, chế độ A cần được áp dụng).

Ở bảng 2 và bảng 3 nêu trên, các bảng này đã được mô tả theo giả thuyết rằng chế độ A, chế độ B và chế độ C, là các chế độ dự đoán, được áp dụng theo cách chọn lọc, và chế độ A, chế độ B và chế độ C có thể được xác định dưới dạng các chế độ dự đoán cụ thể trong số các chế độ dự đoán khác nhau được đề xuất theo sáng chế này. Ví dụ, mỗi trong số chế độ A, chế độ B và chế độ C có thể được xác định dưới dạng một chế độ trong số chế độ hợp nhất chính quy, chế độ CIIP và chế độ hợp nhất hình tam giác. Theo cách khác, như được mô tả ở trên, bảng 2 và bảng 3 được mô tả ở trên có thể được áp dụng ngay cả khi chế độ A, chế độ B, chế độ C và chế độ D được xác định. Ví dụ, mỗi trong số chế độ A, chế độ B, chế độ C và chế độ D có thể được xác định dưới dạng một chế độ trong số chế độ hợp nhất chính quy, chế độ MMVD, chế độ CIIP và chế độ hợp nhất hình tam giác.

FIG.13 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Theo một phương án của sáng chế, phương pháp được mô tả ở bảng 2 và/hoặc bảng 3 có thể được áp dụng trên FIG.13, và phần mô tả lặp lại có liên quan sẽ được lược bỏ. Hơn nữa, FIG.13 có thể là một phương án liên quan đến cờ hợp nhất chính quy như được mô tả trên FIG.10 và FIG.11.

Như được mô tả ở trên, theo một phương án của sáng chế, khi các chế độ, việc sử dụng các chế độ này được xác định tương đối muộn hơn so với việc sử dụng chế độ cụ thể theo trình tự của quy trình giải mã, đều không sử dụng được, thì bộ giải mã có thể xác định (hoặc quyết định hoặc suy ra) có sử dụng chế độ cụ thể mà không báo hiệu việc phân tích cú pháp biểu thị liệu có sử dụng chế độ cụ thể hay không. Ví dụ, khi các chế độ, sử dụng các chế độ này được xác định tương đối muộn hơn so với việc sử dụng chế độ hợp nhất khối phụ, đều không sử dụng được, thì bộ giải mã có thể không báo hiệu việc phân tích cú pháp (hoặc phần tử cú pháp) biểu thị liệu có sử dụng chế độ hợp nhất khối phụ hay không. Ở đây, bộ giải mã có thể xác định có sử dụng chế độ hợp nhất khối phụ mà không thực hiện việc phân tích cú pháp. Ví dụ, các chế độ, sử dụng các chế độ này được xác định tương đối muộn hơn, có thể bao gồm phép dự đoán đa giả thuyết và phép dự đoán hình tam giác.

Theo một phương án, ở bước S1301, nếu các chế độ, việc sử dụng các chế độ này được xác định muộn hơn việc sử dụng MMVD, đều không sử dụng được, thì bộ giải mã có thể không báo hiệu việc phân tích cú pháp biểu thị liệu có sử dụng MMVD hay không, nhưng có thể xác định có sử dụng MMVD. Ví dụ, các chế độ, việc sử dụng các chế độ

này được xác định muộn hơn, có thể bao gồm chế độ hợp nhất khối phụ, phép dự đoán đa giả thuyết và phép dự đoán hình tam giác.

Ngoài ra, theo phương án được mô tả ở trên, các điều kiện (nghĩa là, `mh_intra_conditions` trên FIG.13) mà phép dự đoán đa giả thuyết sử dụng được có thể bao gồm ít nhất một trong số 1) `sps_mh_intra_enabled_flag`, 2) `cu_skip_flag[x0][y0] == 0`, và 3) điều kiện kích thước khối. Một ví dụ là, điều kiện kích thước khối có thể được xác định dưới dạng $((cbWidth * cbHeight) \geq 64 \ \&\& \ cbWidth < 128 \ \&\& \ cbHeight < 128)$. Ở đây, `sps_mh_intra_enabled_flag` thể hiện phần tử cú pháp biểu thị liệu phép dự đoán đa giả thuyết có sử dụng được theo trình tự hiện thời hay không, ví dụ, phần tử cú pháp có thể được báo hiệu thông qua tập tham số chuỗi (sequence parameter set, SPS). Ngoài ra, `cbWidth` và `cbHeight` là các biến số lần lượt thể hiện chiều rộng và chiều cao của khối hiện thời (khối mã hóa hiện thời).

Ngoài ra, theo phương án được mô tả ở trên, các điều kiện mà phép dự đoán hình tam giác (`merge_triangle_conditions` trên FIG.13) sử dụng được có thể bao gồm ít nhất một trong số 1) `sps_triangle_enabled_flag`, 2) `tile_group_type` (hoặc `slice_type`) $== B$, và 3) điều kiện kích thước khối. Một ví dụ là, điều kiện kích thước khối có thể được xác định dưới dạng $(cbWidth * cbHeight \geq 64)$. Ở đây, `sps_triangle_enabled_flag` biểu thị phần tử cú pháp biểu thị liệu phép dự đoán hình tam giác có sử dụng được theo trình tự hiện thời hay không. Ví dụ, phần tử cú pháp có thể được báo hiệu thông qua SPS.

Ngoài ra, theo phương án được mô tả ở trên, các điều kiện (`merge_subblock_conditions` trên FIG.13) mà sự hợp nhất khối phụ sử dụng được có thể bao gồm ít nhất một trong số 1) `MaxNumSubblockMergeCand > 0`, và 2) điều kiện kích thước khối. Một ví dụ là, điều kiện kích thước khối có thể được xác định dưới dạng $(cbWidth \geq 8 \ \&\& \ cbHeight \geq 8)$. Ở đây, `MaxNumSubblockMergeCand` là biến số biểu thị số lượng ứng viên hợp nhất khối phụ tối đa.

Theo đó, theo một phương án, bộ giải mã có thể không phân tích cú pháp cờ hợp nhất khối phụ trong trường hợp đối với $(!mh_intra_conditions \ \&\& \ !merge_triangle_conditions)$. Ngoài ra, khi cờ hợp nhất khối phụ không tồn tại, thì bộ giải mã có thể suy ra cờ hợp nhất khối phụ là 1 trong trường hợp đối với $(!mh_intra_conditions \ \&\& \ !merge_triangle_conditions)$. Ngược lại, bộ giải mã có thể suy ra cờ hợp nhất khối phụ là 0.

Hơn nữa, theo một phương án, bộ giải mã có thể không phân tích cú pháp cờ MMVD trong trường hợp đối với $(!merge_subblock_conditions$

&& !mh_intra_conditions && !merge_triangle_conditions). Hơn nữa, nếu cờ MMVD không tồn tại (!merge_subblock_conditions && !mh_intra_conditions && !merge_triangle_conditions), thì bộ giải mã có thể suy ra cờ MMVD là 1. Ngược lại, bộ giải mã có thể suy ra cờ MMVD là 0.

Hơn nữa, theo một phương án, bộ giải mã có thể không phân tích cú pháp cờ hợp nhất khối phụ trong trường hợp đối với (!sps_mh_intra_enabled_flag && !sps_triangle_enabled_flag), và có thể suy ra giá trị của cờ hợp nhất khối phụ là 1. Theo cách khác, khi cu_skip_flag là 1 và tile_group_type (slice_type) không phải là B, thì bộ giải mã có thể không phân tích cú pháp cờ hợp nhất khối phụ và có thể suy ra giá trị của chúng là 1. Theo cách khác, nếu cả chiều rộng và chiều cao bằng 128, và tile_group_type không phải là B, thì bộ giải mã có thể không phân tích cú pháp cờ hợp nhất khối phụ, nhưng có thể suy ra giá trị của chúng là 1.

FIG.14 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Đối với phương án được mô tả dựa vào FIG.14, các chi tiết được mô tả ở trên dựa vào các hình vẽ từ FIG.10 đến FIG.13 có thể được áp dụng, và các phần mô tả lặp lại sẽ được lược bỏ để thuận tiện cho việc mô tả. Theo một phương án của sáng chế, chế độ hợp nhất có thể bao gồm chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ, chế độ CIIP, chế độ hợp nhất hình tam giác (hoặc chế độ phân vùng hình tam giác (triangle partitioning mode, TPM)) và dạng tương tự. Hơn nữa, có thể có cờ hợp nhất chính quy, cờ MMVD, cờ hợp nhất khối phụ, cờ CIIP và cờ hợp nhất hình tam giác, là báo hiệu (hoặc các phần tử cú pháp) biểu thị liệu chế độ có được sử dụng (hoặc được áp dụng) một cách tương ứng hay không. Như được mô tả ở trên, chế độ dự đoán có thể bao gồm MODE_INTRA, MODE_IBC và MODE_INTER. MODE_INTRA và MODE_IBC có thể là các chế độ dự đoán sử dụng ảnh hiện thời bao gồm khối hiện thời. Hơn nữa, MODE_INTRA có thể là phép dự đoán nội ảnh được mô tả ở trên. MODE_IBC có thể là phương pháp dự đoán sử dụng vectơ chuyển động hoặc vectơ khối. MODE_INTER có thể là phương pháp dự đoán sử dụng ảnh khác, hoặc có thể là phép dự đoán liên ảnh được mô tả ở trên.

Khi chế độ dự đoán hiện thời là MODE_IBC và cờ hợp nhất là 1, thì bộ giải mã có thể chỉ sử dụng chế độ hợp nhất chính quy (S1401). Hơn nữa, ở đây, cờ hợp nhất chính quy có thể không được phân tích cú pháp. Bộ giải mã có thể suy ra cờ hợp nhất chính quy là 1.

Hơn nữa, theo một phương án của sáng chế, việc liệu có phân tích cú pháp phần tử

cú pháp hay không có thể được xác định dựa vào kích thước khối. Ví dụ, việc liệu có phân tích cú pháp phần tử cú pháp hay không có thể được xác định dựa vào kích thước khối. Ví dụ, khi các phần tử cú pháp được báo hiệu theo trình tự của chế độ thứ nhất, chế độ thứ hai, chế độ thứ ba, chế độ thứ tư và chế độ thứ năm, thì các điều kiện kích thước khối nhờ đó chế độ thứ ba, chế độ thứ tư và chế độ thứ năm sử dụng được có thể bao gồm điều kiện thứ ba, điều kiện thứ tư và điều kiện thứ năm. Nếu điều kiện A, là điều kiện không thỏa mãn một điều kiện trong số các điều kiện thứ ba, thứ tư và thứ năm, được thỏa mãn, thì bộ giải mã có thể suy ra không sử dụng hoặc phân tích cú pháp các phần tử cú pháp đối với các chế độ thứ ba, thứ tư và thứ năm. Hơn nữa, khi điều kiện A được thỏa mãn, thì bộ giải mã có thể thực hiện việc xác định dựa vào phần tử cú pháp đối với chế độ thứ nhất mà không phân tích cú pháp phần tử cú pháp liên quan đến chế độ thứ hai. Ở đây, nếu xác định và suy ra được là không sử dụng chế độ thứ nhất, thì bộ giải mã có thể xác định và suy ra có sử dụng chế độ thứ hai. Ngoài ra, dựa vào việc xác định, bộ giải mã có thể phân tích cú pháp phần tử cú pháp được yêu cầu có sử dụng chế độ thứ hai.

Theo một phương án của sáng chế, các điều kiện kích thước khối, nhờ đó chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình chữ nhật sử dụng được, có thể có mặt. Ví dụ, một phương án có thể giống với phương án được mô tả theo một phương án trên FIG.13. Do đó, các khối kích thước 4x4, 8x4 và 4x8 có thể không sử dụng chế độ hợp nhất khối phụ, CIIP hoặc chế độ hợp nhất hình chữ nhật. Do đó, trong trường hợp đối với các khối có các kích thước 4x4, 8x4 và 4x8, khi cờ hợp nhất là 1, thì chỉ chế độ hợp nhất chính quy và MMVD sử dụng được. Do đó, ở đây, bộ giải mã có thể không phân tích cú pháp cờ MMVD. Hơn nữa, ở đây, bộ giải mã có thể xác định hoặc suy ra giá trị cờ MMVD dựa vào cờ hợp nhất chính quy.

Theo một phương án, bộ giải mã có thể không thực hiện phép dự đoán liên ảnh đối với khối 4x4. Do đó, theo các phương án sau, phần mô tả có thể được thực hiện mà không bao gồm các điều kiện liên quan đến khối 4x4, nhưng có thể có một phương án của sáng chế có thể được áp dụng ngay cả khi phép dự đoán liên ảnh 4x4.

Dựa vào FIG.14, khi cbWidth và cbHeight lần lượt bằng 8 và 4 hoặc 4 và 8, thì bộ giải mã có thể không phân tích cú pháp cờ MMVD, cờ hợp nhất khối phụ hoặc cờ dự đoán đa giả thuyết (S1402, S1403 và S1404). Hơn nữa, mặc dù không được thể hiện trên FIG.14, ngay cả khi cả cbWidth và cbHeight bằng 4, bộ giải mã có thể không phân tích cú pháp cờ MMVD, cờ hợp nhất khối phụ hoặc cờ dự đoán đa giả thuyết. Ngoài ra, ở đây, các MMVD khác, chế độ hợp nhất khối phụ, CIIP và các phần tử cú pháp liên quan đến hình tam giác có thể không được phân tích cú pháp.

Hơn nữa, theo sáng chế này, cbWidth và cbHeight có thể lần lượt bằng 4 và 8, hoặc 8 và 4, và do đó tổng của cbWidth và cbHeight bằng 12. Nghĩa là, khi tổng của cbWidth và cbHeight bằng hoặc nhỏ hơn 12, thì cờ MMVD, cờ hợp nhất khối phụ và mh_intra_flag có thể không được phân tích cú pháp. Hơn nữa, sáng chế này có thể được áp dụng khi chế độ dự đoán là MODE_INTER.

Theo một phương án của sáng chế, báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không có thể có mặt. Báo hiệu mức cao hơn có thể là báo hiệu trong đơn vị bao gồm khối hiện thời. Ví dụ, mức cao hơn của khối hiện thời có thể là CTU, chuỗi, ảnh, lát, ô, nhóm ô và dạng tương tự. Ví dụ, báo hiệu mức cao hơn (hoặc phần tử cú pháp) biểu thị liệu MMVD có sử dụng được hay không có thể là báo hiệu mức SPS. Ví dụ, báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không có thể là sps_mmvd_enabled_flag. Báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không có thể biểu thị liệu có sử dụng MMVD hay không. Nếu báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không là 0, thì bộ giải mã có thể không phân tích cú pháp phần tử cú pháp liên quan đến MMVD. Ngoài ra, khi báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không là 0, thì bộ giải mã có thể suy ra cờ MMVD là 0. Khi báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không là 1, thì cờ MMVD có thể là 1 hoặc 0 tùy thuộc vào khối.

Hơn nữa, theo một phương án, phần tử cú pháp liên quan đến chế độ hợp nhất khối phụ có thể bao gồm cờ hợp nhất khối phụ và chỉ số hợp nhất khối phụ. Chế độ hợp nhất khối phụ có thể bao gồm vectơ chuyển động theo thời gian dựa trên khối phụ (subblock-based temporal motion vector, SbTMVP) và chế độ bù chuyển động afin. Ngoài ra, phần tử cú pháp liên quan đến CIIP có thể bao gồm mh_intra_flag (cờ CIIP), và chỉ số biểu thị ứng viên của phần tử đoán liên ảnh của CIIP. Chỉ số biểu thị ứng viên của phần tử đoán liên ảnh của CIIP có thể là chỉ số hợp nhất. Như được mô tả ở trên, CIIP có thể là phương pháp thực hiện phép dự đoán dựa vào tín hiệu dự đoán được tạo ra từ ảnh hiện thời và tín hiệu dự đoán được tạo ra từ ảnh tham chiếu khác, và có thể được gọi là phép dự đoán đa giả thuyết.

Hơn nữa, theo một phương án, phần tử cú pháp liên quan đến chế độ hợp nhất hình chữ nhật có thể bao gồm merge_triangle_split_dir, merge_triangle_idx0 và merge_triangle_idx1. Chế độ hợp nhất hình chữ nhật có thể là phương pháp dự đoán (hoặc chế độ dự đoán) trong đó khối hiện thời được chia thành hai phần và hai phần này sử dụng các thông tin chuyển động khác nhau. Ngoài ra, mỗi trong số hai phần này có thể có dạng hình đa giác định trước thay vì dạng hình chữ nhật. Sáng chế này không bị giới

hạn ở các tên gọi này, và chế độ hợp nhất hình chữ nhật có thể có các tên gọi khác nhau. Hơn nữa, `merge_triangle_split_dir` có thể là phần tử cú pháp biểu thị việc phân chia hai phần. Hơn nữa, `merge_triangle_idx0` và `merge_triangle_idx1` có thể là các phần tử cú pháp biểu thị thông tin chuyển động lần lượt được sử dụng bởi hai phần này.

Theo một phương án của sáng chế, có thể có trường hợp trong đó cờ MMVD không tồn tại. Ví dụ, như được mô tả dựa vào FIG.14, có thể có trường hợp trong đó cờ MMVD không tồn tại theo báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không, điều kiện kích thước khối và dạng tương tự. Theo một phương án bên dưới, phương pháp để thực hiện việc suy luận khi cờ MMVD không tồn tại có thể được mô tả. Theo một phương án của sáng chế, khi các điều kiện cụ thể được thỏa mãn, thì bộ giải mã có thể suy ra cờ MMVD là 1. Hơn nữa, khi ít nhất một điều kiện trong số các điều kiện cụ thể không được thỏa mãn, thì bộ giải mã có thể suy ra cờ MMVD là 0.

Theo một phương án, điều kiện cụ thể có thể bao gồm trường hợp trong đó báo hiệu mức cao hơn (hoặc phần tử cú pháp) biểu thị liệu MMVD có sử dụng được hay không có giá trị bằng 1. Như được mô tả ở trên, báo hiệu mức cao hơn có thể có trong một trong số SPS, PPS, tiêu đề lát, tiêu đề nhóm ô và CTU. Hơn nữa, điều kiện cụ thể có thể bao gồm điều kiện kích thước khối. Ví dụ, điều kiện cụ thể có thể bao gồm trường hợp trong đó kích thước khối là 4x8, 8x4 hoặc 4x4. Nghĩa là, trường hợp cụ thể có thể bao gồm trường hợp trong đó tổng của `cbWidth` và `cbHeight` bằng hoặc nhỏ hơn 12. Nếu phép dự đoán liên ảnh 4x4 không được cho phép, thì trường hợp trong đó kích thước khối là 4x4 có thể được loại trừ. Hơn nữa, điều kiện cụ thể có thể bao gồm trường hợp trong đó cờ hợp nhất chính quy là 0. Hơn nữa, điều kiện cụ thể có thể bao gồm trường hợp trong đó cờ hợp nhất là 1.

Hơn nữa, theo một phương án, trong trường hợp trong đó cờ MMVD không tồn tại, bộ mã hóa/bộ giải mã có thể suy ra cờ MMVD là 1, nếu 1) `sps_mmvd_enabled_flag` là 1, 2) tổng của `cbWidth` và `cbHeight` bằng 12, và 3) cờ hợp nhất chính quy là 0. Hơn nữa, nếu ít nhất một trong số 1), 2) và 3) không được thỏa mãn, thì bộ mã hóa/bộ giải mã có thể suy ra cờ MMVD là 0.

Hơn nữa, theo một phương án của sáng chế, khi cờ hợp nhất chính quy không tồn tại, thì bộ giải mã có thể suy ra giá trị của cờ hợp nhất chính quy theo điều kiện được xác định trước. Theo một phương án, bộ giải mã có thể suy ra cờ hợp nhất chính quy dựa vào chế độ dự đoán của khối hiện thời. Ví dụ, bộ giải mã có thể suy ra cờ hợp nhất chính quy dựa vào giá trị `CuPredMode`. Ví dụ, bộ giải mã có thể suy ra cờ hợp nhất chính quy là 1

khi giá trị CuPredMode là MODE_IBC. Hơn nữa, bộ giải mã có thể suy ra cờ hợp nhất chính quy là 0 khi giá trị CuPredMode là MODE_INTER.

Theo một phương án bổ sung, bộ giải mã có thể suy ra giá trị của cờ hợp nhất chính quy dựa vào cờ hợp nhất. Ví dụ, khi cờ hợp nhất là 1 và CuPredMode là MODE_IBC, thì bộ giải mã có thể suy ra giá trị cờ hợp nhất chính quy là 1. Hơn nữa, khi cờ hợp nhất là 0, thì giá trị cờ hợp nhất chính quy có thể được suy ra là 0.

FIG.15 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.15 có thể là phương án khác liên quan đến các phương án được mô tả dựa vào các hình vẽ từ FIG.10 đến FIG.13. Như được mô tả ở trên, theo một phương án của sáng chế, các chế độ có thể được xác định dưới dạng chế độ hợp nhất. Hơn nữa, trong trường hợp báo hiệu chế độ định trước cần được sử dụng, thì việc liệu có sử dụng chế độ định trước hay không, liệu có phân tích cú pháp báo hiệu hay không biểu thị liệu có sử dụng chế độ định trước hay không hoặc phương pháp suy luận báo hiệu biểu thị liệu có sử dụng chế độ định trước hay không có thể được xác định dựa vào trình tự báo hiệu đối với nhiều chế độ và các điều kiện mà nhiều chế độ này sử dụng được.

Theo một phương án của sáng chế, bộ giải mã có thể xác định liệu có phân tích cú pháp báo hiệu hay không biểu thị liệu có sử dụng chế độ thứ nhất hay không, dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ thứ hai có sử dụng được hay không. Hơn nữa, bộ giải mã có thể suy ra giá trị báo hiệu biểu thị liệu có sử dụng chế độ thứ nhất hay không, dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ thứ hai có sử dụng được hay không. Ở đây, chế độ thứ hai có thể là chế độ, việc sử dụng chế độ này được xác định muộn hơn việc sử dụng chế độ thứ nhất.

Theo phương án cụ thể hơn, bộ giải mã có thể xác định liệu có phân tích cú pháp cờ hợp nhất chính quy dựa vào báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không. Hơn nữa, bộ giải mã có thể suy ra (hoặc xác định) giá trị cờ hợp nhất chính quy dựa vào báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không. Dựa vào FIG.15, một ví dụ là, khi `sps_mmvd_enabled_flag` là 1, thì bộ giải mã có thể phân tích cú pháp cờ hợp nhất chính quy (S1501).

Hơn nữa, theo một phương án, có thể xác định liệu có phân tích cú pháp báo hiệu hay không biểu thị liệu có sử dụng chế độ cụ thể hay không, dựa vào kích thước của khối hiện thời. Hơn nữa, giá trị báo hiệu biểu thị liệu chế độ cụ thể có sử dụng được hay không có thể được suy ra dựa vào kích thước của khối hiện thời. Theo một phương án, ngay cả khi báo hiệu biểu thị liệu có sử dụng chế độ cụ thể hay không không được phân tích cú

pháp dựa vào kích thước của khối hiện thời, thì có thể có trường hợp trong đó chế độ cụ thể được sử dụng. Nghĩa là, giá trị báo hiệu biểu thị liệu có sử dụng chế độ cụ thể hay không có thể được suy ra là 1.

Theo phương án cụ thể hơn, bộ giải mã có thể xác định liệu có phân tích cú pháp cờ hợp nhất chính quy dựa vào kích thước của khối hiện thời. Ví dụ, bộ giải mã có thể xác định liệu có phân tích cú pháp cờ hợp nhất chính quy dựa vào thông tin về việc liệu kích thước của khối hiện thời lớn hơn 4x8 hay 8x4. Kích thước khối lớn hơn 4x8 hoặc 8x4 có thể có tổng của chiều rộng và chiều cao lớn hơn 12. Dựa vào FIG.15, khi tổng của cbWidth và cbHeight lớn hơn 12, thì cờ hợp nhất chính quy có thể được phân tích cú pháp (S1501). Ngoài ra, có thể có chế độ, việc sử dụng chế độ này bị giới hạn theo các kích thước khối nhỏ hơn hoặc bằng 4x8 và 8x4.

Theo một phương án của sáng chế, khi tất cả các điều kiện được thỏa mãn, thì báo hiệu biểu thị liệu có sử dụng chế độ cụ thể hay không có thể không được phân tích cú pháp. Hơn nữa, ở đây, báo hiệu biểu thị liệu có sử dụng chế độ cụ thể hay không có thể được suy ra là 1. Báo hiệu biểu thị liệu có sử dụng chế độ cụ thể hay không bằng 1 có thể biểu thị rằng chế độ cụ thể được sử dụng. Theo một phương án, các điều kiện có thể bao gồm điều kiện liên quan đến báo hiệu mức cao hơn biểu thị liệu chế độ thứ hai khác với chế độ thứ nhất có sử dụng được hay không. Ví dụ, các điều kiện có thể bao gồm điều kiện trong đó báo hiệu mức cao hơn biểu thị liệu chế độ thứ hai khác với chế độ thứ nhất có sử dụng được hay không có giá trị bằng 0. Ở đây, chế độ thứ hai có thể là chế độ, việc sử dụng chế độ này được xác định muộn hơn việc sử dụng chế độ thứ nhất, hoặc có thể là chế độ, phần tử cú pháp có liên quan mà có mặt muộn hơn phần tử cú pháp có liên quan của chế độ thứ nhất.

Theo phương án cụ thể hơn, báo hiệu biểu thị liệu bộ giải mã sử dụng chế độ cụ thể có thể là cờ hợp nhất chính quy hay không. Ngoài ra, các điều kiện có thể là trường hợp trong đó giá trị báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không là 0. Ngoài ra, các điều kiện có thể bao gồm điều kiện liên quan đến kích thước khối. Ví dụ, các điều kiện có thể bao gồm điều kiện trong đó kích thước khối bằng hoặc nhỏ hơn ngưỡng. Theo điều kiện trong đó kích thước khối bằng hoặc nhỏ hơn ngưỡng, một hoặc nhiều chế độ khác, việc sử dụng chế độ này được xác định muộn hơn việc sử dụng chế độ cụ thể hoặc các phần tử cú pháp có liên quan mà tồn tại ở phía sau của chúng, có thể không sử dụng được.

Cụ thể hơn là, báo hiệu biểu thị liệu có sử dụng chế độ định trước hay không có thể

là cờ hợp nhất chính quy. Ngoài ra, các điều kiện có thể bao gồm trường hợp trong đó tổng của chiều rộng và chiều cao của khối hiện thời nhỏ hơn hoặc bằng 12. Theo cách khác, các điều kiện có thể bao gồm trường hợp trong đó kích thước của khối hiện thời là 4×8 hoặc 8×4 . Hơn nữa, khi có thể có phép dự đoán liên ảnh 4×4 , thì các điều kiện có thể bao gồm trường hợp trong đó kích thước khối hiện thời là 4×8 , 8×4 hoặc 4×4 .

Do đó, theo một phương án, khi giá trị báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không là 0 và kích thước khối hiện thời là 4×8 hoặc 8×4 , thì cờ hợp nhất chính quy có thể không được phân tích cú pháp. Hơn nữa, ở đây, giá trị cờ hợp nhất chính quy có thể được suy ra là 1. Ngoài ra, khi giá trị báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không là 1 hoặc kích thước khối hiện thời lớn hơn 4×8 hoặc 8×4 , thì cờ hợp nhất chính quy có thể được phân tích cú pháp.

Ở bước S1501, nếu `sps_mmvd_enabled_flag` là 1 hoặc $cbWidth + cbHeight > 12$, thì bộ giải mã có thể phân tích cú pháp cờ hợp nhất chính quy. Hơn nữa, nếu không, nghĩa là, nếu `sps_mmvd_enabled_flag` là 0 và $cbWidth + cbHeight \leq 12$, thì cờ hợp nhất chính quy có thể không được phân tích cú pháp.

Các phương án như được mô tả ở trên dựa vào các hình vẽ từ FIG.10 đến FIG.13 có thể là liên quan đến các điều kiện sử dụng được của các chế độ liên quan đến các phần tử cú pháp tồn tại sau cờ hợp nhất chính quy. Ví dụ, khi báo hiệu được thực hiện hoặc việc sử dụng chế độ được xác định theo trình tự của chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình tam giác, thì báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không theo phương án được mô tả ở trên là 0, và do đó bộ giải mã có thể không sử dụng MMVD. Hơn nữa, bộ giải mã có thể không sử dụng chế độ hợp nhất khối phụ, CIIP hoặc chế độ hợp nhất hình chữ nhật khi kích thước khối bằng hoặc nhỏ hơn ngưỡng. Do đó, khi tất cả các điều kiện này được thỏa mãn, thì bộ giải mã có thể xác định có sử dụng chế độ hợp nhất chính quy mà không báo hiệu bổ sung. Ngoài ra, phương án này có thể được áp dụng cho trường hợp `MODE_INTER`.

Theo một phương án của sáng chế, khi các điều kiện cụ thể được xác định trước được thỏa mãn như được thể hiện trên FIG.15, thì có thể có trường hợp trong đó cờ hợp nhất chính quy không được phân tích cú pháp, và trong trường hợp này, bộ giải mã có thể suy ra giá trị của cờ hợp nhất chính quy là 1. Ví dụ, khi giá trị báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không là 0 và kích thước khối là 4×8 hoặc 8×4 , thì bộ giải mã có thể suy ra giá trị của cờ hợp nhất chính quy là 1. Hơn nữa, điều này có thể

được thực hiện khi cờ hợp nhất là 1. Hơn nữa, điều này có thể được thực hiện khi CuPredMode là MODE_INTER. Nếu giá trị báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không là 1 hoặc kích thước khối lớn hơn 4x8 hoặc 8x4, thì bộ giải mã có thể suy ra giá trị cờ hợp nhất chính quy là 0.

Một ví dụ là, khi cờ hợp nhất chính quy không tồn tại, bộ giải mã có thể suy ra cờ hợp nhất chính quy theo các điều kiện sau. Cụ thể, khi sps_mmvd_enabled_flag là 0 và cbWidth+cbHeight == 12, thì bộ giải mã có thể suy ra cờ hợp nhất chính quy là 1. Ở đây, nếu phép dự đoán liên ảnh 4x4 được cho phép, thì điều kiện của cbWidth+cbHeight == 12 có thể được thay đổi thành cbWidth+cbHeight <= 12. Ngược lại, bộ giải mã có thể suy ra cờ hợp nhất chính quy là 0.

Theo một phương án của sáng chế, khi cờ hợp nhất hình tam giác, cờ liên afin và cờ hợp nhất khối phụ đều bằng 0, thì cùng thông tin chuyển động có thể được sử dụng cho toàn bộ khối hiện thời. Đối với trường hợp này, quy trình thu được thông tin chuyển động sau có thể được thực hiện. Hơn nữa, ở đây, nếu một hoặc nhiều điều kiện được thỏa mãn, thì bộ giải mã có thể thiết lập dmvrFlag thành 1.

- nếu sps_dmvr_enabled_flag là 1
- nếu merge_flag[xCb][yCb] là 1
- nếu predFlagL0[0][0] và predFlagL1[0][0] là 1
- Khi mmvd_flag[xCb][yCb] là 1
- Khi DiffPicOrderCnt(currPic, RefPicList[0][refIdxL0]) và DiffPicOrderCnt(RefPicList[1][refIdxL1], currPic) giống nhau
- nếu cbHeight bằng hoặc lớn hơn 8
- nếu cbHeight*cbWidth bằng hoặc lớn hơn 64

Hơn nữa, quy trình thu được thông tin chuyển động có thể được thực hiện đối với khối có kích thước bằng 4x8 hoặc 8x4. Nếu phép dự đoán kép được sử dụng đối với khối 4x8 hoặc khối 8x4, thì bộ giải mã có thể chuyển đổi phép dự đoán kép sang phép dự đoán đơn.

Hơn nữa, theo một phương án của sáng chế, khi cờ hợp nhất là 1 và cờ hợp nhất chính quy là 1, thì cùng thông tin chuyển động có thể được sử dụng cho toàn bộ khối hiện thời. Theo cách khác, khi cờ hợp nhất là 1 và cờ MMVD là 1, thì cùng thông tin chuyển động có thể được sử dụng cho toàn bộ khối hiện thời. Theo cách khác, khi cờ hợp nhất là

1 và cờ CIIP là 1, thì cùng thông tin chuyển động có thể được sử dụng cho toàn bộ khối hiện thời. Theo cách khác, khi cờ hợp nhất là 0 và `inter_affine_flag` là 0, thì cùng thông tin chuyển động có thể được sử dụng cho toàn bộ khối hiện thời. Ở đây, quy trình thu được thông tin chuyển động đối với trường hợp này có thể được thực hiện. Hơn nữa, ở đây, nếu một hoặc nhiều trong số các điều kiện được xác định trước được thỏa mãn, thì bộ giải mã có thể thiết lập `dmvrFlag` thành 1. Ở đây, các điều kiện theo phương án được mô tả ở trên có thể được áp dụng. Ngoài ra, quy trình thu được thông tin chuyển động có thể được thực hiện đối với khối 4x8 hoặc 8x4. Nếu phép dự đoán kép được sử dụng đối với khối 4x8 hoặc 8x4, thì bộ giải mã có thể chuyển đổi phép dự đoán kép sang phép dự đoán đơn.

Theo một phương án của sáng chế, CIIP có thể là chế độ được xác định hoặc báo hiệu cuối cùng trong số các chế độ hợp nhất. Ví dụ, việc xác định có được thực hiện theo trình tự của chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ, chế độ hợp nhất hình tam giác và CIIP. Ở đây, khi điều kiện mà CIIP sử dụng được không được thỏa mãn, thì bộ giải mã có thể xác định chế độ tương ứng mà không báo hiệu việc phân tích cú pháp biểu thị liệu có sử dụng chế độ được xác định trước tiên theo trình tự giải mã (hoặc trình tự phân tích cú pháp) hay không. Ví dụ, ở đây, bộ giải mã có thể không báo hiệu việc phân tích cú pháp biểu thị liệu có sử dụng chế độ ngay trước CIIP hay không. Hơn nữa, ở đây, bộ giải mã có thể xác định có sử dụng chế độ ngay trước CIIP. Ví dụ, trường hợp này có thể bao gồm trường hợp trong đó `cu_skip_flag` là 1. Theo cách khác, trường hợp này có thể là trường hợp trong đó `cbWidth` có giá trị lớn hơn hoặc bằng 128 hoặc trường hợp trong đó `cbHeight` có giá trị lớn hơn hoặc bằng 128. Theo cách khác, trường hợp này có thể bao gồm trường hợp trong đó báo hiệu mức cao hơn biểu thị liệu CIIP có sử dụng được hay không, ví dụ, `sps_ciip_enabled_flag` là 0.

FIG.16 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Liên quan đến các phương án trên các hình vẽ từ FIG.16 đến FIG.19, các phương án được mô tả dựa vào các hình vẽ từ FIG.10 đến FIG.13 có thể được áp dụng, và phần mô tả lặp lại có liên quan sẽ được lược bỏ. Như được mô tả ở trên, trong số các chế độ hợp nhất, CIIP có thể là chế độ được xác định hoặc báo hiệu cuối cùng. Theo đó, bộ giải mã có thể xác định xem có sử dụng CIIP hay không mà không phân tích cú pháp cờ CIIP. Ví dụ, nếu không có các chế độ được báo hiệu trước CIIP được sử dụng, thì bộ giải mã có thể xác định có sử dụng CIIP. Hơn nữa, cờ CIIP có thể là giá trị thu được từ báo hiệu khác (hoặc phần tử cú pháp).

Theo một phương án của sáng chế, nhiều báo hiệu biểu thị liệu có sử dụng chế độ

hay không có thể có mặt. Dựa vào FIG.53, báo hiệu biểu thị liệu có sử dụng chế độ hay không có thể bao gồm cờ hợp nhất chính quy, cờ MMVD, cờ hợp nhất khối phụ và cờ hợp nhất hình tam giác. Hơn nữa, bước phân tích cú pháp cờ hợp nhất chính quy, cờ MMVD, cờ hợp nhất khối phụ và cờ hợp nhất hình tam giác có thể có mặt. Ví dụ, khi giá trị cờ hợp nhất là 1, thì báo hiệu biểu thị liệu có sử dụng chế độ hay không có thể được phân tích cú pháp. Theo cách khác, khi CuPredMode là MODE_INTER, thì báo hiệu biểu thị liệu có sử dụng chế độ hay không có thể được phân tích cú pháp. Hơn nữa, khi giá trị cờ hợp nhất là 1, thì bộ giải mã có thể phân tích cú pháp cờ hợp nhất chính quy.

Hơn nữa, khi giá trị cờ hợp nhất chính quy là 0, thì bộ giải mã có thể phân tích cú pháp cờ MMVD. Hơn nữa, khi giá trị của `sps_mmvd_enabled_flag` là 1, thì bộ giải mã có thể phân tích cú pháp cờ MMVD. Hơn nữa, bộ giải mã có thể phân tích cú pháp cờ MMVD khi điều kiện kích thước khối được thỏa mãn. Ví dụ, nếu kích thước khối không phải là 4x8, 8x4 và 4x4, thì cờ MMVD có thể được phân tích cú pháp.

Hơn nữa, khi giá trị cờ hợp nhất chính quy là 0, thì cờ hợp nhất khối phụ có thể được phân tích cú pháp. Hơn nữa, khi giá trị cờ MMVD là 0, thì cờ hợp nhất khối phụ có thể được phân tích cú pháp. Hơn nữa, khi điều kiện kích thước khối được thỏa mãn, thì cờ hợp nhất khối phụ có thể được phân tích cú pháp. Ví dụ, khi kích thước khối lớn hơn hoặc bằng 8x8, thì cờ hợp nhất khối phụ có thể được phân tích cú pháp. Hơn nữa, khi số lượng ứng viên hợp nhất khối phụ tối đa lớn hơn 0, thì cờ hợp nhất khối phụ có thể được phân tích cú pháp. Ví dụ, trường hợp trong đó số lượng ứng viên hợp nhất khối phụ tối đa lớn hơn 0 có thể biểu thị rằng ít nhất một báo hiệu trong số các báo hiệu mức cao hơn liên quan đến việc liệu có sử dụng ứng viên hay không mà có thể có trong danh sách ứng viên hợp nhất khối phụ sử dụng được. Ví dụ, khi `sps_affine_enabled_flag` hoặc `sps_sbtmvp_enabled_flag` là 1, thì số lượng ứng viên hợp nhất khối phụ tối đa có thể lớn hơn 0.

Hơn nữa, khi giá trị cờ hợp nhất chính quy là 0, thì cờ hợp nhất hình tam giác có thể được phân tích cú pháp. Hơn nữa, khi giá trị cờ MMVD là 0, thì cờ hợp nhất hình tam giác có thể được phân tích cú pháp. Hơn nữa, khi giá trị cờ hợp nhất khối phụ là 0, thì cờ hợp nhất hình tam giác có thể được phân tích cú pháp. Hơn nữa, khi điều kiện kích thước khối được thỏa mãn, thì cờ hợp nhất hình tam giác có thể được phân tích cú pháp. Ví dụ, cờ hợp nhất hình tam giác có thể được phân tích cú pháp khi kích thước khối thỏa mãn điều kiện ($\text{chiều rộng} \times \text{chiều cao} \geq 64$). Hơn nữa, khi kiểu lát là B, thì cờ hợp nhất hình tam giác có thể được phân tích cú pháp. Ví dụ, kiểu lát của B có thể biểu thị rằng hai hoặc nhiều hơn hai thông tin chuyển động sử dụng được khi dự đoán một mẫu. Hơn nữa, khi

giá trị `sps_triangle_enabled_flag` là 1, thì bộ giải mã có thể phân tích cú pháp cờ hợp nhất hình tam giác. Hơn nữa, nếu điều kiện dựa vào số lượng ứng viên hợp nhất hình tam giác tối đa (`MaxNumTriangleMergeCand`) được thỏa mãn, thì cờ hợp nhất hình tam giác có thể được phân tích cú pháp. Ví dụ, khi số lượng ứng viên hợp nhất hình tam giác tối đa lớn hơn hoặc bằng 2, thì bộ giải mã có thể phân tích cú pháp cờ hợp nhất hình tam giác. Số lượng ứng viên hợp nhất hình tam giác tối đa có thể là số lượng ứng viên tối đa (hoặc chiều dài) của danh sách ứng viên sử dụng được ở chế độ hợp nhất hình chữ nhật.

Khi các điều kiện phân tích cú pháp được mô tả ở trên được thỏa mãn, thì bộ giải mã có thể phân tích cú pháp báo hiệu. Nghĩa là, báo hiệu có thể không được phân tích cú pháp trừ khi một điều kiện trong số các điều kiện nêu trên mà cho phép phân tích cú pháp được thỏa mãn. Hơn nữa, bộ giải mã có thể thực hiện việc suy luận khi báo hiệu không được phân tích cú pháp. Ví dụ, nếu một điều kiện trong số các điều kiện nêu trên mà cho phép phân tích cú pháp không được thỏa mãn, thì giá trị báo hiệu có thể được suy ra là 0. Một ví dụ khác là, nếu một điều kiện trong số các điều kiện nêu trên mà cho phép phân tích cú pháp không được thỏa mãn, khi báo hiệu liên quan đến việc liệu có sử dụng chế độ thứ nhất là 0 hay không, thì giá trị của báo hiệu liên quan đến việc liệu có sử dụng chế độ thứ nhất hay không có thể được suy ra là 1. Một ví dụ khác là, nếu một điều kiện trong số các điều kiện nêu trên mà cho phép phân tích cú pháp không được thỏa mãn, khi báo hiệu liên quan đến việc liệu có sử dụng chế độ thứ nhất là 1 hay không, thì giá trị của báo hiệu liên quan đến việc liệu có sử dụng chế độ thứ nhất hay không có thể được suy ra là 0.

Hơn nữa, theo một phương án của sáng chế, khi cờ CIIP không tồn tại, thì bộ giải mã có thể suy ra giá trị của chúng. Ví dụ, giá trị cần được suy ra có thể được xác định dựa vào giá trị của báo hiệu liên quan đến việc liệu có sử dụng một hoặc nhiều chế độ hay không. Báo hiệu liên quan đến việc liệu có sử dụng chế độ hay không có thể bao gồm báo hiệu liên quan đến việc liệu có sử dụng các chế độ được xác định hay không trước khi xác định liệu có sử dụng CIIP hay không. Ví dụ, báo hiệu liên quan đến việc liệu có sử dụng chế độ hay không có thể bao gồm báo hiệu liên quan đến việc liệu có sử dụng chế độ hợp nhất chính quy hay không, báo hiệu liên quan đến việc liệu có sử dụng MMVD hay không, báo hiệu liên quan đến việc liệu có sử dụng chế độ hợp nhất khối phụ hay không và báo hiệu liên quan đến việc liệu có sử dụng chế độ hợp nhất hình chữ nhật hay không. Ngoài ra, báo hiệu liên quan đến việc liệu có sử dụng chế độ hay không có thể bao gồm báo hiệu liên quan đến việc liệu có sử dụng chế độ hợp nhất hay không.

Theo một phương án, khi tất cả các giá trị của báo hiệu liên quan đến việc liệu có sử dụng một hoặc nhiều chế độ là 0 hay không, thì bộ giải mã có thể suy ra giá trị cờ CIIP

là 1. Báo hiệu liên quan đến việc liệu có sử dụng một hoặc nhiều chế độ hay không có thể bao gồm cờ hợp nhất chính quy, cờ MMVD, cờ hợp nhất khối phụ và cờ hợp nhất hình tam giác. Do đó, nếu cờ hợp nhất chính quy $= 0$ && cờ MMVD $= 0$ && cờ hợp nhất khối phụ $= 0$ && cờ hợp nhất hình tam giác $= 0$, thì giá trị cờ CIIP có thể được suy ra là 1. Hơn nữa, nếu không, thì giá trị cờ CIIP có thể được suy ra là 0.

Theo một phương án, khi tất cả các giá trị của báo hiệu liên quan đến việc liệu có sử dụng một hoặc nhiều chế độ là 0 và cờ hợp nhất là 1 hay không, thì bộ giải mã có thể suy ra giá trị cờ CIIP là 1. Báo hiệu liên quan đến việc liệu có sử dụng một hoặc nhiều chế độ hay không có thể bao gồm cờ hợp nhất chính quy, cờ MMVD, cờ hợp nhất khối phụ và cờ hợp nhất hình tam giác. Do đó, nếu cờ hợp nhất chính quy $= 0$ && cờ MMVD $= 0$ && cờ hợp nhất khối phụ $= 0$ && cờ hợp nhất hình tam giác $= 0$ && cờ hợp nhất $= 1$, thì bộ giải mã có thể suy ra giá trị cờ CIIP là 1. Hơn nữa, nếu không, bộ giải mã có thể suy ra giá trị cờ CIIP là 0. Ngoài ra, chỉ báo rằng giá trị của báo hiệu liên quan đến việc liệu có sử dụng chế độ là 1 hay không có thể biểu thị rằng chế độ được sử dụng, và chỉ báo rằng giá trị của báo hiệu liên quan đến việc liệu có sử dụng chế độ là 0 hay không có thể biểu thị rằng chế độ không được sử dụng.

FIG.17 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.17 có thể là phương pháp báo hiệu hiệu quả dựa vào một phương án trên FIG.16, và phân mô tả lặp lại của chúng sẽ được lược bỏ. Như được mô tả ở trên, CIIP có thể là chế độ được xác định hoặc báo hiệu cuối cùng trong số các chế độ hợp nhất. Theo một phương án, ở đây, phương pháp báo hiệu được mô tả dựa vào các hình vẽ từ FIG.10 đến FIG.13 có thể được sử dụng. Một phương án trên các hình vẽ từ FIG.17 đến FIG.19 có thể là một phương án cụ thể về phương pháp được mô tả dựa vào các hình vẽ từ FIG.10 đến FIG.13.

Theo một phương án của sáng chế, khi việc sử dụng chế độ được xác định hoặc báo hiệu được thực hiện theo trình tự của chế độ A, chế độ B, chế độ C và chế độ D, thì các điều kiện mà chế độ D là không sử dụng được có thể có mặt. Nếu ít nhất một điều kiện trong số các điều kiện mà chế độ D là không sử dụng được được thỏa mãn, thì bộ giải mã có thể không báo hiệu việc phân tích cú pháp liên quan đến việc liệu có sử dụng chế độ C hay không. Ngoài ra, nếu báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không không tồn tại, thì bộ giải mã có thể suy ra giá trị của chúng. Ở đây, giá trị cần được suy ra có thể dựa vào điều kiện mà chế độ D là không sử dụng được, báo hiệu liên quan đến việc liệu có sử dụng chế độ A hay không và báo hiệu liên quan đến việc liệu có sử dụng chế độ B hay không. Ngoài ra, khi một điều kiện trong số các điều kiện mà chế

độ D là không sử dụng được được thỏa mãn, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không có thể được phân tích cú pháp. Theo cách khác, khi một số trong số các điều kiện mà chế độ D là không sử dụng được không thỏa mãn, thì bộ giải mã có thể phân tích cú pháp báo hiệu (hoặc phần tử cú pháp) liên quan đến việc liệu có sử dụng chế độ C hay không. Khi có nhiều điều kiện mà chế độ D là không sử dụng được, chỉ một số trong số các điều kiện nêu trên có thể sử dụng được theo phương pháp báo hiệu của sáng chế. Ví dụ, chỉ một số điều kiện có thể được sử dụng để giảm các điều kiện cần được kiểm tra khi xác định liệu có phân tích cú pháp báo hiệu hay không liên quan đến việc liệu có sử dụng chế độ C hay không.

Theo một phương án, chế độ D có thể là CIIP. Ngoài ra, chế độ A, chế độ B và chế độ C có thể lần lượt là MMVD, chế độ hợp nhất khối phụ và chế độ hợp nhất hình chữ nhật, và ở đây, có thể còn để tạo cấu hình chế độ A, chế độ B và chế độ C theo trình tự khác nhau. Các hình vẽ từ FIG.17 đến FIG.19 giả sử rằng chế độ A, chế độ B và chế độ C lần lượt là MMVD, chế độ hợp nhất khối phụ và chế độ hợp nhất hình chữ nhật. Theo một phương án, điều kiện mà chế độ D là không sử dụng được có thể dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ D có sử dụng được hay không. Hơn nữa, điều kiện mà chế độ D là không sử dụng được có thể dựa vào kích thước khối. Hơn nữa, điều kiện mà chế độ D là không sử dụng được có thể dựa vào `cu_skip_flag`. Ngoài ra, điều kiện mà chế độ D là không sử dụng được có thể dựa vào kiểu nhóm ô (hoặc lát). Ngoài ra, điều kiện mà chế độ D là không sử dụng được có thể dựa vào số lượng ứng viên tối đa mà sử dụng được ở chế độ D.

Dựa vào FIG.17, các điều kiện mà CIIP là không sử dụng được có thể bao gồm trường hợp trong đó `sps_ciip_enabled_flag` là 0, trường hợp trong đó `cu_skip_flag` là 1, trường hợp trong đó `cbWidth` có giá trị lớn hơn hoặc bằng 128, và trường hợp trong đó `cbHeight` có giá trị lớn hơn hoặc bằng 128. Do đó, theo một phương án của sáng chế, khi `sps_ciip_enabled_flag` là 0, `cu_skip_flag` là 1, `cbWidth` có giá trị lớn hơn hoặc bằng 128, hoặc `cbHeight` có giá trị lớn hơn hoặc bằng 128, báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không có thể không được phân tích cú pháp. Nghĩa là, theo một phương án trên FIG.17, khi `sps_ciip_enabled_flag` là 0, `cu_skip_flag` là 1, `cbWidth` có giá trị lớn hơn hoặc bằng 128 hoặc `cbHeight` có giá trị lớn hơn hoặc bằng 128, thì cờ hợp nhất hình tam giác có thể không được phân tích cú pháp. Ngoài ra, khi `sps_ciip_enabled_flag` là 1, `cu_skip_flag` là 0, `cbWidth` có giá trị nhỏ hơn 128 và `cbHeight` có giá trị nhỏ hơn 128, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không có thể được phân tích cú pháp. Nghĩa là, theo một phương án trên FIG.54, khi `sps_ciip_enabled_flag` là 1,

cu_skip_flag là 0, cbWidth có giá trị nhỏ hơn 128 và cbHeight có giá trị nhỏ hơn 128, thì cờ hợp nhất hình tam giác có thể được phân tích cú pháp.

Ngoài ra, khi xác định liệu có phân tích cú pháp báo hiệu hay không liên quan đến việc liệu có sử dụng chế độ C hay không, điều kiện nhờ đó chế độ C sử dụng được có thể được xem xét bổ sung. Ví dụ, khi điều kiện nhờ đó chế độ C sử dụng được được thỏa mãn, thì báo hiệu (hoặc phần tử cú pháp) liên quan đến việc liệu có sử dụng chế độ C hay không có thể được phân tích cú pháp. Dựa vào FIG.16, các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được có thể bao gồm điều kiện trong đó giá trị của sps_triangle_enabled_flag là 1, điều kiện trong đó tile_group_type là B, và điều kiện trong đó $cbWidth * cbHeight \geq 64$.

Theo một phương án của sáng chế, một ví dụ về phương pháp suy luận liên quan đến một phương án được mô tả dựa vào FIG.17 sẽ được mô tả. Phương án này có thể là phương pháp suy luận báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không được mô tả dựa vào FIG.17. Hơn nữa, khi báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không không tồn tại, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không có thể được suy ra.

Theo một phương án trên FIG.17, khi ít nhất một điều kiện trong số các điều kiện mà chế độ D là không sử dụng được không được thỏa mãn, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không có thể không được phân tích cú pháp. Theo một phương án của sáng chế, khi các điều kiện được thỏa mãn, thì giá trị của báo hiệu liên quan đến việc liệu có sử dụng chế độ C hay không có thể được suy ra là 1. Ví dụ, giá trị 1 có thể biểu thị rằng chế độ C được sử dụng, và giá trị 0 có thể biểu thị rằng chế độ C không được sử dụng. Ngoài ra, các điều kiện có thể bao gồm điều kiện thỏa mãn ít nhất một điều kiện trong số các điều kiện mà chế độ D là không sử dụng được. Ngoài ra, các điều kiện có thể bao gồm điều kiện nhờ đó chế độ C sử dụng được. Ngoài ra, các điều kiện có thể bao gồm các điều kiện dựa vào báo hiệu liên quan đến việc liệu có sử dụng chế độ A và chế độ B hay không. Ví dụ, các điều kiện có thể bao gồm trường hợp trong đó báo hiệu liên quan đến việc liệu có sử dụng chế độ A và chế độ B hay không biểu thị rằng không có chế độ A và chế độ B được sử dụng. Ngoài ra, khi ít nhất một điều kiện trong số các điều kiện không được thỏa mãn, thì bộ giải mã có thể suy ra giá trị của báo hiệu liên quan đến việc liệu có sử dụng chế độ C là 0 hay không.

Theo một phương án của sáng chế, bộ giải mã có thể suy ra giá trị cờ hợp nhất hình tam giác dựa vào điều kiện được xác định trước. Một ví dụ là, bộ giải mã có thể suy ra giá

trị cờ hợp nhất hình tam giác là 1 khi `sps_ciip_enabled_flag` là 0, `cu_skip_flag` là 1, `cbWidth` có giá trị lớn hơn hoặc bằng 128 hoặc `cbHeight` có giá trị lớn hơn hoặc bằng 128. Ví dụ, bộ giải mã có thể suy ra giá trị cờ hợp nhất hình tam giác là 1 chỉ khi `sps_ciip_enabled_flag` là 0, `cu_skip_flag` là 1, `cbWidth` có giá trị lớn hơn hoặc bằng 128 hoặc `cbHeight` có giá trị lớn hơn hoặc bằng 128. Ngoài ra, để suy ra giá trị cờ hợp nhất hình tam giác là 1, thì điều kiện bổ sung có thể cần được thỏa mãn. Ví dụ, điều kiện bổ sung có thể bao gồm điều kiện mà cờ hợp nhất chính quy là 0, điều kiện trong đó cờ MMVD là 0, và điều kiện trong đó cờ hợp nhất khối phụ là 0. Hơn nữa, điều kiện bổ sung có thể bao gồm điều kiện trong đó cờ hợp nhất là 1. Hơn nữa, điều kiện bổ sung có thể bao gồm điều kiện trong đó `sps_triangle_enabled_flag` là 1, điều kiện trong đó `tile_group_type` là B và điều kiện trong đó `cbWidth*cbHeight >= 64`. Hơn nữa, khi tất cả các điều kiện bổ sung được thỏa mãn, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 1.

Theo một phương án, giá trị cờ hợp nhất hình tam giác có thể được suy ra là 1 khi tất cả các điều kiện sau được thỏa mãn.

- 1) cờ hợp nhất chính quy == 0
- 2) cờ MMVD == 0
- 3) cờ hợp nhất khối phụ == 0
- 4) `sps_ciip_enabled_flag == 0 || cu_skip_flag == 1 || cbWidth >= 128 || cbHeight >= 128`
- 5) `sps_triangle_enabled_flag == 1 && tile_group_type == B && cbWidth*cbHeight >= 64`

Theo cách khác, theo một phương án khác, khi tất cả các điều kiện sau được thỏa mãn, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 1.

- 1) cờ hợp nhất chính quy == 0
- 2) cờ MMVD == 0
- 3) cờ hợp nhất khối phụ == 0
- 4) `sps_ciip_enabled_flag == 0 || cu_skip_flag == 1 || cbWidth >= 128 || cbHeight >= 128`
- 5) `sps_triangle_enabled_flag == 1 && tile_group_type == B && cbWidth*cbHeight >= 64`

6) cờ hợp nhất == 1

Hơn nữa, theo một phương án, khi một điều kiện trong số các điều kiện được mô tả ở trên không được thỏa mãn, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 0. Ví dụ, khi `sps_ciip_enabled_flag` là 1, `cu_skip_flag` là 0, `cbWidth` < 128, và `cbHeight` < 128, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 0. Theo cách khác, khi cờ hợp nhất chính quy là 1, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 0. Theo cách khác, khi cờ MMVD là 1, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 0. Theo cách khác, khi cờ hợp nhất khối phụ là 1, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 0. Theo cách khác, khi `sps_triangle_enabled_flag` là 0, `tile_group_type` không phải là B hoặc `cbWidth*cbHeight`<64, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 0. Theo cách khác, khi cờ hợp nhất là 0, thì giá trị cờ hợp nhất hình tam giác có thể được suy ra là 0.

FIG.18 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.18 có thể là một phương án cụ thể về phương pháp được mô tả dựa vào FIG.17. Trong phần mô tả trên FIG.17, khi xác định liệu có phân tích cú pháp báo hiệu hay không liên quan đến việc liệu có sử dụng chế độ C hay không, thì chỉ một số điều kiện có thể được sử dụng để giảm các điều kiện cần được kiểm tra như được mô tả ở trên. Ví dụ, một phương án trên FIG.18 có thể là phương pháp không kiểm tra `sps_ciip_enabled_flag` trên FIG.17.

Ví dụ, khi `cu_skip_flag` là 1, `cbWidth` >= 128 hoặc `cbHeight` >= 128, thì bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác. Hơn nữa, ở đây, bộ giải mã có thể suy ra giá trị cờ hợp nhất hình tam giác là 1. Theo cách khác, bộ giải mã có thể suy ra giá trị cờ hợp nhất hình tam giác là 1 chỉ khi điều kiện này được thỏa mãn. Hơn nữa, như được mô tả ở trên, khi điều kiện bổ sung được thỏa mãn, thì bộ giải mã có thể suy ra giá trị cờ hợp nhất hình tam giác là 1. Hơn nữa, khi `cu_skip_flag` là 0, `cbWidth` < 128 và `cbHeight` < 128, thì bộ giải mã có thể phân tích cú pháp cờ hợp nhất hình tam giác. Hơn nữa, khi `cu_skip_flag` là 0, `cbWidth` < 128 và `cbHeight` < 128, nếu cờ hợp nhất hình tam giác không tồn tại, thì bộ giải mã có thể suy ra giá trị là 0.

Phương án này có thể có ưu điểm làm giảm các thao tác kiểm tra các điều kiện trong quá trình phân tích cú pháp phần tử cú pháp so với một phương án trên FIG.17 được mô tả ở trên. Như nêu trên, khi trình tự báo hiệu chế độ được tạo cấu hình theo cách khác nhau, thì sáng chế này có thể được áp dụng cho báo hiệu khác thay vì được áp dụng cho cờ hợp nhất hình tam giác.

FIG.19 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.19 có thể là một phương án cụ thể về phương pháp được mô tả dựa vào FIG.17. Trong phần mô tả dựa vào FIG.17, phần mô tả này đã đề cập rằng chỉ một số điều kiện được sử dụng để giảm các điều kiện cần được kiểm tra khi xác định liệu có phân tích cú pháp báo hiệu hay không liên quan đến việc liệu có sử dụng chế độ C hay không, và FIG.19 có thể minh họa phần mô tả liên quan đến FIG.17. Ví dụ, một phương án trên FIG.19 có thể là phương pháp không kiểm tra `sps_ciip_enabled_flag` trên FIG.17, và có thể là phương pháp không kiểm tra liệu `cbWidth` có giá trị nhỏ hơn 128 hay không và liệu `cbHeight` có giá trị nhỏ hơn 128 hay không.

Ví dụ, khi `cu_skip_flag` là 1, thì bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác. Hơn nữa, ở đây, bộ giải mã có thể suy ra giá trị cờ hợp nhất hình tam giác là 1. Theo cách khác, bộ giải mã có thể suy ra giá trị cờ hợp nhất hình tam giác là 1 chỉ khi điều kiện này được thỏa mãn. Ngoài ra, bộ giải mã có thể suy ra giá trị cờ hợp nhất hình tam giác là 1 khi điều kiện bổ sung được thỏa mãn như được mô tả ở trên. Hơn nữa, khi `cu_skip_flag` là 0, thì cờ hợp nhất hình tam giác có thể được phân tích cú pháp. Hơn nữa, khi `cu_skip_flag` là 0, thì bộ giải mã có thể suy ra giá trị bằng 0 nếu cờ hợp nhất hình tam giác không tồn tại.

Phương án này có thể có ưu điểm làm giảm các thao tác kiểm tra các điều kiện trong quá trình phân tích cú pháp phần tử cú pháp so với một phương án trên FIG.17 được mô tả ở trên. Như nêu trên, khi trình tự báo hiệu chế độ được tạo cấu hình theo cách khác nhau, thì sáng chế này có thể được áp dụng cho báo hiệu khác thay vì được áp dụng cho cờ hợp nhất hình tam giác.

FIG.20 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.20 có thể là một phương án cụ thể về phương pháp được mô tả dựa vào FIG.17. Các hình vẽ từ FIG.20 đến FIG.24 có thể là các phương án cụ thể của sáng chế được mô tả ở trên. Ví dụ, các phương án trên các hình vẽ từ FIG.20 đến FIG.24 có thể là liên quan đến các phương án được mô tả dựa vào các hình vẽ từ FIG.10 đến FIG.13, và phần mô tả lặp lại có liên quan sẽ được lược bỏ.

Dựa vào FIG.20, báo hiệu biểu thị liệu có sử dụng MMVD hay không theo sáng chế này được mô tả dựa vào FIG.20 có thể là `mmvd_merge_flag`. Nghĩa là, theo một phương án được mô tả ở trên, cờ MMVD có thể được gọi là `mmvd_merge_flag` trong phần mô tả sau. Hơn nữa, theo sáng chế này được mô tả dựa vào FIG.20, báo hiệu biểu thị các ứng viên cơ bản của MMVD có thể là `mmvd_cand_flag`. Nghĩa là, `mmvd_merge_flag`

được nêu trước đó có thể tương ứng với `mmvd_cand_flag` dựa trên FIG.20. Hơn nữa, các dấu hiệu liên quan để kiểu lát có thể được áp dụng cho kiểu nhóm ô, và ngược lại. Hơn nữa, các giá trị biểu thị kiểu lát và kiểu nhóm ô có thể lần lượt là `slice_type` và `tile_group_type`. Hơn nữa, báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất được mô tả ở trên hay không có thể là `general_merge_flag`. Nghĩa là, phần mô tả, mà đã được thực hiện liên quan đến cờ hợp nhất, có thể là liên quan đến `general_merge_flag`, và phần mô tả liên quan đến `general_merge_flag` có thể được áp dụng cho cờ hợp nhất.

Theo một phương án của sáng chế, chế độ được báo hiệu cuối cùng trong số các chế độ hợp nhất khác nhau có thể là chế độ hợp nhất khối phụ. Như được mô tả ở trên, các chế độ hợp nhất khác nhau có thể bao gồm chế độ hợp nhất chính quy, MMVD, CIIP, chế độ hợp nhất hình tam giác, chế độ hợp nhất khối phụ và dạng tương tự. Ngoài ra, chế độ hợp nhất hình chữ nhật có thể được báo hiệu ngay trước chế độ hợp nhất khối phụ trong số các chế độ hợp nhất khác nhau.

Dựa vào FIG.20, bộ mã hóa/bộ giải mã có thể thực hiện việc báo hiệu/phân tích cú pháp theo trình tự của chế độ hợp nhất chính quy, MMVD, CIIP, chế độ hợp nhất hình tam giác và chế độ hợp nhất khối phụ theo cú pháp dữ liệu hợp nhất. Theo một phương án của sáng chế, việc liệu có phân tích cú pháp cờ hợp nhất chính quy hay không có thể được xác định dựa vào `general_merge_flag`. Trong bản mô tả này, `general_merge_flag` có thể được gọi là cờ hợp nhất chung. Theo một phương án của sáng chế, khi `general_merge_flag` là 1, thì cờ hợp nhất chính quy có thể được phân tích cú pháp. Ở đây, các điều kiện bổ sung để phân tích cú pháp có thể được yêu cầu. Ngoài ra, khi `general_merge_flag` là 0, thì cờ hợp nhất chính quy có thể không được phân tích cú pháp. Ở đây, nếu `general_merge_flag` là 0, thì cờ hợp nhất chính quy có thể không được phân tích cú pháp bất kể các điều kiện khác. Theo một phương án của sáng chế, khi `general_merge_flag` là 1, thì một phần của cấu trúc `merge_data` trên FIG.20 có thể được phân tích cú pháp.

Theo một phương án của sáng chế, bộ giải mã có thể xác định liệu có phân tích cú pháp `mmvd_merge_flag` dựa vào `general_merge_flag` hay không. Theo một phương án của sáng chế, khi `general_merge_flag` là 1, thì `mmvd_merge_flag` có thể được phân tích cú pháp. Ở đây, các điều kiện bổ sung để phân tích cú pháp có thể được yêu cầu. Hơn nữa, khi `general_merge_flag` là 0, thì `mmvd_merge_flag` có thể không được phân tích cú pháp. Ở đây, nếu `general_merge_flag` là 0, thì `MMVD_merge_flag` có thể không được phân tích cú pháp bất kể các điều kiện khác.

Theo một phương án của sáng chế, việc liệu có phân tích cờ CIIP hay không có thể được xác định dựa vào `general_merge_flag`. Theo một phương án của sáng chế, khi `general_merge_flag` là 1, thì có thể có bước phân tích cú pháp cờ CIIP. Ở đây, các điều kiện bổ sung để phân tích cú pháp có thể được yêu cầu. Hơn nữa, khi `general_merge_flag` là 0, thì cờ CIIP có thể không được phân tích cú pháp. Ở đây, nếu `general_merge_flag` là 0, thì cờ CIIP có thể không được phân tích cú pháp bất kể các điều kiện khác.

Theo một phương án của sáng chế, việc liệu có phân tích cú pháp cờ hợp nhất hình tam giác hay không có thể được xác định dựa vào `general_merge_flag`. Theo một phương án của sáng chế, khi `general_merge_flag` là 1, thì có thể có bước phân tích cú pháp cờ hợp nhất hình tam giác. Ở đây, các điều kiện bổ sung để phân tích cú pháp có thể được yêu cầu. Hơn nữa, khi `general_merge_flag` là 0, thì cờ hợp nhất hình tam giác có thể không được phân tích cú pháp. Ở đây, nếu `general_merge_flag` là 0, thì cờ hợp nhất hình tam giác có thể không được phân tích cú pháp bất kể các điều kiện khác.

Theo một phương án của sáng chế, bộ giải mã có thể xác định liệu có phân tích cú pháp `mmvd_merge_flag` dựa vào cờ hợp nhất chính quy hay không. Theo một phương án của sáng chế, có thể có bước phân tích cú pháp `MMVD_merge_flag` khi cờ hợp nhất chính quy là 0. Ở đây, các điều kiện bổ sung để phân tích cú pháp có thể được yêu cầu. Hơn nữa, khi cờ hợp nhất chính quy là 1, thì `mmvd_merge_flag` có thể không được phân tích cú pháp. Ở đây, nếu cờ hợp nhất chính quy là 1, thì `mmvd_merge_flag` có thể không được phân tích cú pháp bất kể các điều kiện khác.

Theo một phương án của sáng chế, bộ giải mã có thể xác định liệu có phân tích cờ CIIP dựa vào `mmvd_merge_flag` hay không. Theo một phương án của sáng chế, khi `mmvd_merge_flag` là 0, thì có thể có bước phân tích cú pháp cờ CIIP. Ở đây, các điều kiện bổ sung để phân tích cú pháp có thể được yêu cầu. Hơn nữa, khi `mmvd_merge_flag` là 1, thì cờ CIIP có thể không được phân tích cú pháp. Ở đây, khi `mmvd_merge_flag` là 1, thì cờ CIIP có thể không được phân tích cú pháp bất kể các điều kiện khác.

Theo một phương án của sáng chế, có thể xác định được liệu có phân tích cú pháp cờ hợp nhất hình tam giác dựa vào cờ CII hay không. Theo một phương án của sáng chế, khi cờ CII là 0, thì có thể có bước phân tích cú pháp cờ hợp nhất hình tam giác. Ở đây, các điều kiện bổ sung để phân tích cú pháp có thể được yêu cầu. Hơn nữa, khi cờ CII là 1, thì cờ hợp nhất hình tam giác có thể không được phân tích cú pháp. Ở đây, khi cờ CII là 1, thì cờ hợp nhất hình tam giác có thể không được phân tích cú pháp bất kể các điều kiện khác.

Theo một phương án của sáng chế, việc liệu có phân tích cú pháp cờ hợp nhất khối

phụ hay không có thể được xác định dựa vào cờ hợp nhất hình tam giác. Theo một phương án của sáng chế, khi cờ hợp nhất hình tam giác là 0, thì có thể có bước phân tích cú pháp cờ hợp nhất khối phụ. Ở đây, các điều kiện bổ sung để phân tích cú pháp có thể được yêu cầu. Hơn nữa, khi cờ hợp nhất hình tam giác là 1, thì cờ hợp nhất khối phụ có thể không được phân tích cú pháp. Ở đây, khi cờ hợp nhất hình tam giác là 1, thì cờ hợp nhất khối phụ có thể không được phân tích cú pháp bất kể các điều kiện khác.

Theo một phương án khác của sáng chế, chế độ được báo hiệu cuối cùng trong số các chế độ hợp nhất khác nhau có thể xác định giá trị biểu thị liệu có sử dụng mà không phân tích cú pháp hay không. Ví dụ, dựa vào FIG.20, cờ hợp nhất khối phụ có thể được xác định mà không phân tích cú pháp. Ví dụ, khi tất cả các điều kiện sau được thỏa mãn, thì cờ hợp nhất khối phụ có thể được xác định là 1.

1) `general_merge_flag == 1`

2) Khi không có các chế độ được báo hiệu trước chế độ hợp nhất khối phụ trong số các chế độ hợp nhất khác nhau được sử dụng

3) Khi các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được được thỏa mãn

Hơn nữa, nếu không (nghĩa là, khi ít nhất một điều kiện trong số các điều kiện nêu trên không được thỏa mãn), thì cờ hợp nhất khối phụ có thể được xác định là 0.

Ví dụ, trong số các điều kiện nêu trên, điều kiện “2) khi không có các chế độ được báo hiệu trước chế độ hợp nhất khối phụ trong số các chế độ hợp nhất khác nhau được sử dụng” có thể được xác định dưới dạng điều kiện sau (hoặc có thể bao gồm điều kiện sau).

`(regular_merge_flag == 0 && mmvd_merge_flag == 0 && ciip_flag == 0 && merge_triangle_flag == 0)`

Hơn nữa, trong số các điều kiện nêu trên, điều kiện “3) khi các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được được thỏa mãn” có thể như sau (hoặc có thể bao gồm điều kiện sau).

`(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)`

Theo cách khác, “3)” của các điều kiện nêu trên có thể giống với các điều kiện sau (hoặc có thể bao gồm các điều kiện sau).

(ít nhất một trong số các phương pháp mà có thể có trong chế độ hợp nhất khối phụ được kích hoạt `&& cbWidth >= 8 && cbHeight >= 8`)

Ngoài ra, phương pháp mà có thể có trong chế độ hợp nhất khối phụ có thể bao gồm phép bù chuyển động afin và phép dự đoán vectơ chuyển động theo thời gian dựa trên khối phụ. Ngoài ra, báo hiệu mức cao hơn biểu thị liệu phép bù chuyển động afin và các phép dự đoán vectơ chuyển động theo thời gian dựa trên khối phụ có sử dụng được hay không có thể lần lượt được xác định dưới dạng `sps_affine_enabled_flag` và `sps_sbtmvp_enabled_flag`. Theo phương án này, các giá trị cụ thể đối với chiều rộng và chiều cao trong các điều kiện nêu trên đã được mô tả làm các ví dụ, nhưng sáng chế không bị giới hạn ở đó, và có thể bao gồm điều kiện dựa vào kích thước khối chung.

FIG.21 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. FIG.21 có thể thể hiện một phương án trong đó phương pháp báo hiệu hiệu quả hơn được thêm vào một phương án trên FIG.20 được mô tả ở trên. Theo một phương án của sáng chế, nếu các điều kiện sử dụng được cho tất cả hoặc một hoặc nhiều chế độ được báo hiệu muộn hơn chế độ định trước trong số các chế độ khác nhau không thỏa mãn, thì việc liệu có sử dụng chế độ định trước hay không có thể được xác định mà không báo hiệu một cách rõ ràng giống nhau. Ví dụ, báo hiệu biểu thị liệu có sử dụng chế độ định trước hay không có thể không được phân tích cú pháp.

Ví dụ, nếu các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được không thỏa mãn trong cấu trúc cú pháp được báo hiệu như được thể hiện trên FIG.21, thì việc liệu chế độ hợp nhất hình chữ nhật có sử dụng được hay không có thể được xác định mà không báo hiệu rõ ràng. Ví dụ, cờ hợp nhất hình tam giác có thể không được phân tích cú pháp nếu các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được không thỏa mãn. Theo một phương án, các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được có thể giống với trường hợp trong đó các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được được thỏa mãn trên FIG.20 (hoặc các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được có thể bao gồm điều kiện sau).

Do đó, dựa vào FIG.21, nếu `MaxNumSubblockMergeCand` là 0, thì bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác. Ở đây, `MaxNumSubblockMergeCand` là biến số biểu thị số lượng ứng viên hợp nhất khối phụ tối đa. Theo cách khác, bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác dựa vào kích thước khối. Theo cách khác, bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác khi `cbWidth` có giá trị nhỏ hơn 8. Nghĩa là, khi `cbWidth` có giá trị bằng 4 (hoặc nhỏ hơn hoặc bằng 4), thì bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác. Theo cách khác, bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác khi `cbHeight` có giá trị nhỏ hơn 8. Nghĩa là, khi `cbHeight` có giá trị

bằng 4 (hoặc nhỏ hơn hoặc bằng 4), thì bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác.

Do đó, theo một phương án của sáng chế, liên quan đến khối $4 \times X$ hoặc khối $X \times 4$ (liên quan đến khối có chiều rộng hoặc chiều cao bằng 4), bộ giải mã có thể không phân tích cú pháp cờ hợp nhất hình tam giác. Sau đây, phương pháp suy luận cờ hợp nhất hình tam giác sẽ được mô tả. Theo một phương án của sáng chế, các giá trị tối thiểu của $cbWidth$ và $cbHeight$ có thể là 4. Ví dụ, giá trị tối thiểu của $cbWidth$ và $cbHeight$ đối với khối độ sáng có thể là 4. Hơn nữa, $cbWidth$ và $cbHeight$ có thể được biểu diễn dưới dạng lũy thừa cơ số 2. Do đó, ví dụ, $cbWidth$ có giá trị lớn hơn hoặc bằng 8 có thể có cùng ý nghĩa với $cbWidth$ có giá trị khác 4. Theo một phương án bổ sung, các giá trị tối đa của $cbWidth$ và $cbHeight$ có thể là 128.

Ngoài ra, theo sáng chế trên FIG.21, sáng chế này đã mô tả được rằng các chế độ hợp nhất khác nhau được báo hiệu theo trình tự của chế độ hợp nhất hình tam giác và chế độ hợp nhất khối phụ. Tuy nhiên, sáng chế không bị giới hạn ở đó, và có thể được áp dụng cho trường hợp trong đó báo hiệu được thực hiện theo trình tự của CIIP và chế độ hợp nhất khối phụ. Nghĩa là, theo các phương án được mô tả ở trên, chế độ hợp nhất hình chữ nhật và cờ hợp nhất hình tam giác có thể được thay thế bằng CIIP và cờ CIIP.

FIG.21 minh họa phương pháp xác định báo hiệu biểu thị liệu có sử dụng chế độ hay không theo một phương án của sáng chế. Phương pháp được mô tả theo một phương án trên FIG.20 có thể được áp dụng cho một phương án trên FIG.21, và phần mô tả lặp lại có liên quan sẽ được lược bỏ. Dựa vào FIG.21, khi cờ hợp nhất hình tam giác không tồn tại, thì bộ giải mã có thể suy ra giá trị của chúng.

Theo một phương án của sáng chế, khi báo hiệu liên quan đến việc liệu có sử dụng chế độ định trước trong số các chế độ hợp nhất khác nhau hay không không tồn tại, thì giá trị của chúng có thể được suy ra. Theo một phương án, bộ giải mã có thể suy ra giá trị là “1” nếu 1) không có các chế độ được báo hiệu trước chế độ định trước trong số các chế độ hợp nhất khác nhau được sử dụng, 2) các điều kiện sử dụng được đối với tất cả các chế độ mà được báo hiệu muộn hơn chế độ định trước trong số các chế độ hợp nhất khác nhau không thỏa mãn, và 3) các điều kiện nhờ đó chế độ định trước sử dụng được được thỏa mãn. Hơn nữa, nếu không (nghĩa là, một điều kiện trong số các điều kiện của 1), 2) hoặc 3) không được thỏa mãn), thì bộ giải mã có thể suy ra giá trị là 0. Hơn nữa, ở đây, việc không thỏa mãn các điều kiện sử dụng được đối với tất cả các chế độ ở điều kiện “2)” có thể biểu thị rằng ít nhất một điều kiện trong số các điều kiện sử dụng được đối với mỗi

trong số tất cả các chế độ không được thỏa mãn.

Ngoài ra, ở đây, 4) điều kiện để sử dụng một chế độ trong số các chế độ hợp nhất khác nhau có thể được thêm vào các điều kiện để suy ra báo hiệu liên quan đến việc liệu có sử dụng chế độ định trước là 1 hay không. Ví dụ, 4) nếu `general_merge_mode` là 1 có thể được thêm vào đó.

Ví dụ, dựa vào một phương án trên FIG.20, nếu 1) không có các chế độ được báo hiệu trước chế độ hợp nhất hình chữ nhật trong số các chế độ hợp nhất khác nhau được sử dụng, 2) các điều kiện sử dụng được đối với tất cả các chế độ mà được báo hiệu muộn hơn chế độ hợp nhất hình chữ nhật trong số các chế độ hợp nhất khác nhau không thỏa mãn, 3) các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được được thỏa mãn, và 4) nếu `general_merge_mode` là 1, thì cờ hợp nhất hình tam giác có thể được suy ra là 1. Hơn nữa, nếu không (nghĩa là, một điều kiện trong số các điều kiện của 1), 2), 3) hoặc 4) không được thỏa mãn), thì bộ giải mã có thể suy ra cờ hợp nhất hình tam giác là 0.

Ở đây, dựa vào FIG.21, “1) không có các chế độ được báo hiệu trước chế độ hợp nhất hình chữ nhật trong số các chế độ hợp nhất khác nhau được sử dụng” có thể bao gồm điều kiện sau.

$$(\text{regular_merge_flag} == 0 \ \&\& \ \text{mmvd_merge_flag} == 0 \ \&\& \ \text{ciip_flag} == 0)$$

Hơn nữa, dựa vào FIG.21, “2) việc không thỏa mãn các điều kiện sử dụng được đối với tất cả các chế độ mà được báo hiệu muộn hơn chế độ hợp nhất hình chữ nhật trong số các chế độ hợp nhất khác nhau” có thể tương ứng với trường hợp trong đó các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được không thỏa mãn, và có thể bao gồm các điều kiện sau. Ví dụ, điều kiện 2) nêu trên có thể bao gồm điều kiện của kích thước khối.

$$(\text{MaxNumSubblockMergeCand} == 0 \ || \ \text{cbWidth} == 4 \ || \ \text{cbHeight} == 4)$$

Hơn nữa, dựa vào FIG.60, “3) thỏa mãn các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được” có thể bao gồm điều kiện sau.

$$(\text{MaxNumTriangleMergeCand} \geq 2 \ \&\& \ \text{sps_triangle_enabled_flag} \ \&\& \ \text{slice_type} == \text{B} \ \&\& \ \text{cbWidth} * \text{cbHeight} \geq 64)$$

Theo một phương án bổ sung, một số trong số các điều kiện có thể không được yêu cầu cần được kiểm tra để giảm các thao tác được yêu cầu cho việc kiểm tra điều kiện theo một phương án trên các hình vẽ từ FIG.20 đến FIG.21. Ví dụ, bộ giải mã có thể không sử dụng một số trong số các điều kiện không thỏa mãn các điều kiện nhờ đó chế độ hợp nhất

khối phụ sử dụng được hoặc các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được khi phân tích cú pháp hoặc suy luận cờ hợp nhất hình tam giác. Ở đây, các điều kiện được sử dụng trong thao tác phân tích cú pháp và điều kiện suy luận có thể là các điều kiện giống nhau. Ví dụ, khi phân tích cú pháp hoặc suy luận cờ hợp nhất hình tam giác, bộ giải mã có thể không kiểm tra điều kiện đối với MaxNumSubblockMergeCand. Nghĩa là, cờ hợp nhất hình tam giác có thể được phân tích cú pháp ngay cả khi MaxNumSubblockMergeCand là 0, và nếu cờ hợp nhất hình tam giác không tồn tại, thì bộ giải mã có thể không kiểm tra điều kiện liên quan đến việc liệu MaxNumSubblockMergeCand có bằng 0 hay không khi suy luận giá trị của chúng.

FIG.22 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.22 có thể là một phương án chi tiết trong đó các phương án trên các hình vẽ từ FIG.16 đến FIG.19 và các phương án trên FIG.20 và FIG.21 được thực hiện cùng nhau. Theo một phương án của sáng chế, tổng phí báo hiệu có thể được giảm bớt. Theo một phương án của sáng chế, báo hiệu có thể được thực hiện theo trình tự của chế độ A, chế độ B, chế độ C, chế độ D và chế độ E. Ở đây, khi chế độ D hoặc chế độ E sử dụng được, thì báo hiệu biểu thị liệu có sử dụng chế độ C hay không có thể được phân tích cú pháp. Nếu không có chế độ D và chế độ E có thể được sử dụng, thì báo hiệu biểu thị liệu có sử dụng chế độ C hay không có thể không được phân tích cú pháp. Hơn nữa, nếu không có chế độ D và chế độ E có thể được sử dụng, và nếu không có chế độ A và chế độ B được sử dụng, thì có thể xác định được có sử dụng báo hiệu biểu thị liệu có sử dụng chế độ C hay không.

Dựa vào FIG.22, các chế độ hợp nhất khác nhau có thể được báo hiệu theo trình tự của chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất hình tam giác, chế độ hợp nhất khối phụ và CIIP. Ở đây, theo một phương án của sáng chế, khi CIIP là không sử dụng được, thì bộ giải mã có thể không báo hiệu việc phân tích cú pháp biểu thị liệu có sử dụng chế độ hợp nhất khối phụ hay không. Ngoài ra, khi CIIP sử dụng được, thì bộ giải mã có thể phân tích cú pháp báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất khối phụ hay không. Ngoài ra, nếu một trong số CIIP, hợp nhất chính quy, MMVD và chế độ hợp nhất hình chữ nhật được sử dụng, và general_merge_flag là 1, thì bộ giải mã có thể suy ra có sử dụng báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất khối phụ hay không. Ngược lại, bộ giải mã có thể suy ra không sử dụng báo hiệu này.

Ví dụ, các điều kiện mà CIIP sử dụng được có thể bao gồm, dưới dạng điều kiện &&(và), ít nhất một trong số 1) điều kiện dựa vào báo hiệu mức cao hơn biểu thị liệu CIIP có sử dụng được hay không, 2) điều kiện dựa vào cu_skip_flag, và 3) điều kiện dựa

vào kích thước khối (chiều rộng hoặc chiều cao). Dựa vào FIG.61, các điều kiện mà CIIP sử dụng được có thể bao gồm, dưới dạng điều kiện &&(và), ít nhất một điều kiện trong số các điều kiện 1) `sps_ciip_enabled_flag`, 2) `cu_skip_flag == 0` và 3) `cbWidth*cbHeight >= 64 && cbWidth < 128 && cbHeight < 128`. Dựa vào FIG.22, CIIP có thể được sử dụng trong trường hợp trong đó (`sps_ciip_enabled_flag && cu_skip_flag == 0 && cbWidth*cbHeight >= 64 && cbWidth < 128 && cbHeight < 128`).

Ngoài ra, theo một phương án của sáng chế, nếu không có chế độ hợp nhất khối phụ và CIIP có thể được sử dụng, thì báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất hình chữ nhật hay không có thể không được phân tích cú pháp. Ngoài ra, khi chế độ hợp nhất khối phụ hoặc CIIP sử dụng được, thì báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất hình chữ nhật hay không có thể được phân tích cú pháp. Ngoài ra, nếu không có chế độ hợp nhất khối phụ và CIIP có thể được sử dụng, thì hợp nhất chính quy và MMVD không được sử dụng, và `general_merge_flag` là 1, có thể suy ra được rằng báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất hình chữ nhật hay không được sử dụng. Ngược lại, có thể suy ra được rằng báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất hình chữ nhật hay không không được sử dụng. Ví dụ, các điều kiện mà CIIP sử dụng được hoặc trường hợp trong đó CIIP là không sử dụng được có thể là được mô tả dựa vào phần mô tả trước đó. Tuy nhiên, ở đây, điều kiện trùng lặp (ví dụ, `cbWidth*cbHeight >= 64` trên FIG.22) trong số các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được và các điều kiện mà CIIP sử dụng được có thể được lược bỏ khỏi các điều kiện mà CIIP sử dụng được. Ngoài ra, các điều kiện nhờ đó chế độ `mergeonjungko` khối phụ sử dụng được có thể bao gồm, dưới dạng các điều kiện &&(và), một hoặc nhiều điều kiện trong số 1) các điều kiện dựa vào `MaxNumSubblockMergeonjungkoCand` và 2) các điều kiện dựa vào kích thước khối.

Dựa vào FIG.22, các điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được có thể bao gồm, dưới dạng các điều kiện && (và), một hoặc nhiều điều kiện 1) `MaxNumSubblockMergeCand > 0`, và 2) `cbWidth >= 8 && cbHeight >= 8`. Dựa vào FIG.22, điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được có thể là (`MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8`). Trường hợp trong đó chế độ hợp nhất khối phụ là không sử dụng được có thể là điều kiện KHÔNG đối với trường hợp trong đó chế độ hợp nhất khối phụ sử dụng được.

FIG.23 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.23 có thể là phương án khác tương tự với FIG.22. Theo một phương án của sáng chế, báo hiệu có thể được thực hiện theo trình tự của chế độ A, chế độ B, chế độ C, chế độ D và chế độ E. Ở đây, khi chế độ D hoặc chế độ

E sử dụng được, thì báo hiệu biểu thị liệu có sử dụng chế độ C hay không có thể được phân tích cú pháp. Nếu không có chế độ D và chế độ E có thể được sử dụng, thì báo hiệu biểu thị liệu có sử dụng chế độ C hay không có thể không được phân tích cú pháp. Hơn nữa, nếu không có chế độ D và chế độ E có thể được sử dụng, và khi không có chế độ A và chế độ B được sử dụng, thì có thể xác định được có sử dụng báo hiệu biểu thị liệu có sử dụng chế độ C hay không.

Dựa vào FIG.23, các chế độ hợp nhất khác nhau có thể được báo hiệu theo trình tự của chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất hình tam giác, CIIP và chế độ hợp nhất khối phụ. Ở đây, theo một phương án của sáng chế, khi chế độ hợp nhất khối phụ là không sử dụng được, thì báo hiệu biểu thị liệu có sử dụng CIIP hay không có thể không được phân tích cú pháp. Ngoài ra, khi chế độ hợp nhất khối phụ sử dụng được, thì báo hiệu biểu thị liệu có sử dụng CIIP hay không có thể được phân tích cú pháp. Ngoài ra, khi chế độ hợp nhất khối phụ là không sử dụng được, nếu hợp nhất chính quy, MMVD và chế độ hợp nhất hình tam giác không được sử dụng, và `general_merge_flag` là 1, thì có thể suy ra được rằng báo hiệu biểu thị liệu có sử dụng CIIP hay không được sử dụng. Ngược lại, có thể suy ra được rằng báo hiệu biểu thị liệu có sử dụng CIIP hay không không được sử dụng. Về vấn đề này, phần mô tả trên FIG.21 sẽ được đề cập đến. Ngoài ra, theo một phương án của sáng chế, khi không có CIIP và chế độ hợp nhất khối phụ có thể được sử dụng, và nếu CIIP hoặc chế độ hợp nhất khối phụ sử dụng được, thì báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất hình chữ nhật hay không sẽ được mô tả dựa vào phần mô tả trên FIG.22.

FIG.24 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Các hình vẽ từ FIG.24 đến FIG.25 thể hiện một phương án trong đó các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được được thêm vào một phương án trên FIG.17. Theo một phương án của sáng chế, các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được có thể bao gồm số lượng ứng viên chế độ hợp nhất hình tam giác tối đa. Ví dụ, giá trị biểu thị số lượng ứng viên chế độ hợp nhất hình tam giác tối đa có thể là `MaxNumTriangleMergeCand`. Ví dụ, để kích hoạt chế độ hợp nhất hình chữ nhật được sử dụng, thì có thể cần điều kiện (`MaxNumTriangleMergeCand` $\geq$ 2).

Do đó, dựa vào FIG.24, nếu (`MaxNumTriangleMergeCand` $\geq$ 2) được thỏa mãn, thì có thể phân tích cú pháp cờ hợp nhất hình tam giác, và nếu (`MaxNumTriangleMergeCand` $\geq$ 2) không được thỏa mãn, cờ hợp nhất hình tam giác có thể không được phân tích cú pháp. Ngoài ra, phần mô tả trùng lặp với phần mô tả trên FIG.17 sẽ được lược bỏ.

Theo đó, bộ giải mã có thể phân tích cú pháp cờ hợp nhất hình tam giác khi tất cả các điều kiện sau được thỏa mãn. Nếu ít nhất một điều kiện trong số các điều kiện sau không được thỏa mãn, thì cờ hợp nhất hình tam giác có thể không được phân tích cú pháp.

- 1) $\text{MaxNumTriangleMergeCand} \geq 2$
- 2) $\text{sps_triangle_enabled_flag}$
- 3) $\text{slice_type} == B$
- 4) $\text{cbWidth} * \text{cbHeight} \geq 64$
- 5) $\text{sps_ciip_enabled_flag}$
- 6) $\text{cu_skip_flag} == 0$
- 7) $\text{cbWidth} < 128$
- 8) $\text{cbHeight} < 128$

Theo một phương án khác, một số trong số các điều kiện nêu trên có thể được lược bỏ. Điều này có thể làm giảm thao tác được yêu cầu để kiểm tra các điều kiện. Ví dụ, điều kiện được lược bỏ có thể là ít nhất một trong số 5), 6), 7) và 8).

Hơn nữa, theo một phương án của sáng chế, cờ CIIP có thể được xác định như sau. Khi tất cả các điều kiện sau được thỏa mãn, thì cờ CIIP có thể được thiết lập thành 1.

- a) $\text{general_merge_flag} == 1$
- b) $\text{regular_merge_flag} == 0$
- c) $\text{mmvd_merge_flag} == 0$
- d) $\text{merge_subblock_flag} == 0$
- e) $\text{merge_triangle_flag} == 0$
- f) $\text{sps_ciip_enabled_flag} == 1$
- g) $\text{cu_skip_flag} == 0$
- h) $\text{cbWidth} * \text{cbHeight} \geq 64$
- i) $\text{cbWidth} < 128$
- j) $\text{cbHeight} < 128$

Nếu ít nhất một điều kiện trong số các điều kiện nêu trên không được thỏa mãn, thì

cờ CIIP có thể được thiết lập thành 0. Ví dụ, các điều kiện h), i) và j) có thể được thay thế bằng các điều kiện khác liên quan đến kích thước khối.

Theo một phương án của sáng chế, khi cờ hợp nhất hình tam giác không tồn tại, thì cờ hợp nhất hình tam giác có thể được suy ra theo quy trình sau. Nếu tất cả các điều kiện sau được thỏa mãn, thì cờ hợp nhất hình tam giác có thể được suy ra là 1.

- 1) `regular_merge_flag == 0`
- 2) `mmvd_merge_flag == 0`
- 3) `merge_subblock_flag == 0`
- 4) `sps_ciip_enabled_flag == 0 || cu_skip_flag == 1 || cbWidth >= 128 || cbHeight >= 128`
- 5) `MaxNumTriangleMergeCand >= 2 && sps_triangle_enabled_flag == 1 && tile_group_type == B && cbWidth*cbHeight >= 64`
- 6) `general_merge_flag == 1`

Ngược lại, giá trị cờ hợp nhất hình tam giác có thể được suy ra là 0. Trong số các điều kiện nêu trên, các điều kiện này được kết nối bằng cách sử dụng `||` (nghĩa là, HOẶC) ở điều kiện 4) tương ứng với các điều kiện 5), 6), 7) và 8) được mô tả trên FIG.63, và nếu có thiếu sót bất kỳ trong số các điều kiện 5), 6), 7) và 8), điều tương tự có thể xảy ra ở điều kiện 4) trên FIG.64. Như được mô tả ở trên dựa vào FIG.24, các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được có thể bao gồm số lượng ứng viên chế độ hợp nhất hình tam giác tối đa. Về vấn đề này, phần mô tả lặp lại sẽ được lược bỏ.

Hơn nữa, theo một phương án của sáng chế, ít nhất một chế độ trong số các chế độ hợp nhất khác nhau có thể được sử dụng để báo hiệu liên quan đến việc có sử dụng chế độ này hay không. Ví dụ, khi chế độ hợp nhất được sử dụng (khi `general_merge_flag` là “1”), thì ít nhất một chế độ trong số các chế độ hợp nhất khác nhau có thể được biểu thị có sử dụng báo hiệu liên quan đến việc liệu có sử dụng chế độ này hay không. Theo một phương án, ít nhất một chế độ có thể là chế độ được thiết lập trước. Ví dụ, ít nhất một chế độ có thể là một chế độ. Ví dụ, ít nhất một chế độ có thể là chế độ hợp nhất chính quy.

Theo một phương án, khi chế độ hợp nhất được sử dụng, nếu không có các báo hiệu liên quan đến việc liệu có sử dụng các chế độ hợp nhất khác nhau hay không được sử dụng, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ định trước được tạo cấu hình hay không được sử dụng. Theo một phương án khác, nếu không có các báo hiệu liên quan đến việc liệu có sử dụng các chế độ hợp nhất khác nhau hay không ngoại trừ đối với báo

hiệu liên quan đến chế độ định trước được sử dụng, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ định trước được tạo cấu hình hay không được sử dụng. Cấu hình này có thể ngăn chặn báo hiệu có lỗi và các hoạt động theo đó xảy ra.

Theo một phương án của sáng chế, cờ hợp nhất chính quy có thể được thiết lập thành 1 khi tất cả các điều kiện sau được thỏa mãn.

- 1) `regular_merge_flag == 0`
- 2) `mmvd_merge_flag == 0`
- 3) `merge_subblock_flag == 0`
- 4) `ciip_flag == 0`
- 5) `merge_triangle_flag == 0`
- 6) `general_merge_flag == 1`

Theo một phương án khác, một số trong số các điều kiện có thể được lược bỏ. Ví dụ, điều kiện 1) trong số các điều kiện nêu trên có thể được lược bỏ.

Theo một phương án khác của sáng chế, khi các điều kiện sử dụng được đối với tất cả các chế độ ngoại trừ đối với một chế độ cụ thể trong số các chế độ hợp nhất khác nhau không thỏa mãn, thì bộ giải mã có thể suy ra giá trị của các chế độ này là 1 mà không báo hiệu việc phân tích cú pháp liên quan đến việc liệu có sử dụng chế độ cụ thể hay không. Theo cách khác, khi các điều kiện khả dụng đối với ít nhất một chế độ trong số các chế độ ngoại trừ đối với một chế độ cụ thể trong số các chế độ hợp nhất khác nhau được thỏa mãn, thì báo hiệu liên quan đến việc liệu có sử dụng một chế độ cụ thể hay không có thể được phân tích cú pháp. Theo một phương án, điều này có thể tương ứng với trường hợp sử dụng chế độ hợp nhất. Hơn nữa, một chế độ định trước có thể là chế độ hợp nhất chính quy.

Cụ thể hơn là, nếu ít nhất một điều kiện trong số các điều kiện từ 1) đến 4) sau được thỏa mãn, thì bộ giải mã có thể phân tích cú pháp cờ hợp nhất chính quy. Theo một phương án, điều này có thể tương ứng với trường hợp sử dụng chế độ hợp nhất.

- 1) `sps_mmvd_enabled_flag && cbWidth*cbHeight != 32`
- 2) `MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8`
- 3) `sps_ciip_enabled_flag && cu_skip_flag == 0 && cbWidth*cbHeight >= 64 && cbWidth < 128 && cbHeight < 128`

4) `MaxNumTriangleMergeCand >= 2 && sps_triangle_enabled_flag && slice_type == B && cbWidth*cbHeight >= 64`

Hơn nữa, nếu không có các điều kiện từ 1) đến 4) nêu trên được thỏa mãn, thì cờ hợp nhất chính quy có thể không được phân tích cú pháp, và giá trị của cờ này có thể được suy ra là 1. Điều này có thể tương ứng với trường hợp sử dụng chế độ hợp nhất. Ở đây, một số trong số các điều kiện nêu trên có thể được lược bỏ để giảm lượng thao tác.

Ngoài ra, theo một phương án khác của sáng chế, khi chỉ báo của báo hiệu liên quan đến việc liệu hai hay nhiều hơn hai chế độ trong số các chế độ hợp nhất khác nhau được sử dụng xảy ra, thì báo hiệu liên quan đến việc liệu có sử dụng tất cả các chế độ ngoại trừ chế độ được thiết lập trước trong số các chế độ hợp nhất khác nhau được tạo cấu hình hay không không được sử dụng, và báo hiệu liên quan đến việc liệu có sử dụng chế độ được thiết lập trước có thể được tạo cấu hình hay không được sử dụng. Ví dụ, chế độ được thiết lập trước có thể là chế độ hợp nhất chính quy. Một ví dụ khác là, chế độ được thiết lập trước có thể là một chế độ trong số hai hoặc nhiều hơn hai chế độ mà báo hiệu liên quan đến việc liệu có sử dụng chế độ này hay không được biểu thị là cần được sử dụng. Ở đây, phương pháp được thiết lập trước để xác định một chế độ có thể có mặt. Ví dụ, chế độ thứ nhất theo trình tự được thiết lập trước đối với các chế độ hợp nhất khác nhau được xác định. Ví dụ, phương án này có thể tương ứng với trường hợp sử dụng chế độ hợp nhất.

Ví dụ, nếu `regular_merge_flag == 1` và `merge_subblock_flag == 1`, thì `merge_subblock_flag` có thể được thiết lập thành 0. Theo cách khác, nếu `ciip_flag == 1` và `merge_subblock_flag == 1`, thì `ciip_flag` và `merge_subblock_flag` có thể được thiết lập thành 0 và `regular_merge_flag` có thể được thiết lập thành 1. Một ví dụ khác là, nếu `ciip_flag == 1` và `merge_subblock_flag == 1`, thì `merge_subblock_flag`, xuất hiện trước tiên theo trình tự được thiết lập trước của chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình tam giác, có thể được thiết lập thành 1, và cờ CIIP có thể được thiết lập thành 1.

FIG.25 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.25 có thể là một phương án cụ thể về sáng chế được mô tả ở trên. Ví dụ, các phương pháp được mô tả theo một phương án trên các hình vẽ từ FIG.10 đến FIG.13 có thể được áp dụng cho một phương án trên FIG.25, và phần mô tả lặp lại sẽ được lược bỏ.

Như được mô tả ở trên, việc liệu có sử dụng nhiều chế độ hợp nhất hay không có

thể được báo hiệu hoặc xác định theo trình tự được thiết lập trước. Dựa vào FIG.66, nhiều chế độ hợp nhất có thể bao gồm chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình tam giác. Hơn nữa, dựa vào FIG.25, việc liệu có sử dụng nhiều chế độ hợp nhất hay không có thể được báo hiệu hoặc xác định theo trình tự của chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình tam giác 1. Hơn nữa, dựa vào FIG.25, báo hiệu biểu thị liệu có sử dụng chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình tam giác hay không có thể lần lượt là `regular_merge_flag`, `mmvd_merge_flag`, `merge_subblock_flag`, `ciip_flag` và `MergeTriangleFlag`. Hơn nữa, `MergeTriangleFlag` có thể có cùng ý nghĩa với `merge_triangle_flag` được mô tả ở trên.

Ngoài ra, các điều kiện cần được thỏa mãn có thể có mặt để kích hoạt việc thực thi mỗi chế độ. Ví dụ, khi các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ định trước không thỏa mãn, thì chế độ định trước có thể không được thực thi. Hơn nữa, ở đây, chế độ khác với chế độ định trước có thể được thực hiện. Theo cách khác, nếu các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ định trước được thỏa mãn, thì chế độ định trước có thể được thực hiện hay không. Ở đây, báo hiệu bổ sung để xác định liệu có thực hiện chế độ định trước hay không có thể có mặt.

Ví dụ, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ định trước có thể dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ định trước có sử dụng được hay không. Mức cao hơn có thể bao gồm mức độ chuỗi, mức tập tham số chuỗi (SPS), mức độ lát, mức độ ô, mức độ nhóm ô, mức độ khối gạch, mức độ CTU và dạng tương tự. Ngoài ra, `sps_mode_enabled_flag` được mô tả ở trên có thể được bao gồm trong đó. Ở đây, chế độ có thể được thay thế bằng các chế độ định trước.

Ngoài ra, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ định trước có thể bao gồm điều kiện liên quan đến kích thước khối. Ví dụ, các điều kiện dựa vào chiều rộng hoặc chiều cao của khối hiện thời có thể được bao gồm trong đó. Ví dụ, có thể có giới hạn trên hoặc giới hạn dưới đối với chiều rộng. Theo cách khác, có thể có giới hạn trên hoặc giới hạn dưới đối với chiều cao. Theo cách khác, có thể có giới hạn trên hoặc giới hạn dưới của vùng (chiều rộng * chiều cao). Hơn nữa, khối hiện thời có thể là CU hoặc PU. Hơn nữa, chiều rộng và chiều cao của khối hiện thời có thể lần lượt là `cbWidth` và `cbHeight`. Theo sáng chế này, chiều rộng và chiều cao có thể lần lượt được sử dụng thay thế được với `cbWidth` và `cbHeight`.

Hơn nữa, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ định

trước có thể dựa vào kiểu lát hoặc kiểu nhóm ô. Kiểu lát và kiểu nhóm ô có thể có cùng ý nghĩa.

Hơn nữa, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ định trước có thể dựa vào việc liệu chế độ định trước khác có được sử dụng hay không. Chế độ định trước khác có thể bao gồm chế độ bỏ qua. Hơn nữa, việc liệu có sử dụng chế độ bỏ qua hay không có thể được xác định dựa vào `cu_skip_flag`. Hơn nữa, chế độ định trước khác có thể bao gồm chế độ được báo hiệu hoặc xác định trước chế độ định trước. Ví dụ, có thể thực thi chế độ định trước khi chế độ định trước khác không được sử dụng.

Hơn nữa, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ định trước có thể dựa vào số lượng ứng viên tối đa. Ví dụ, các ứng viên có thể là các ứng viên liên quan đến chế độ định trước. Ví dụ, các ứng viên có thể là các ứng viên được sử dụng ở chế độ định trước. Ví dụ, có thể thực hiện chế độ định trước khi có nhiều hơn số lượng ứng viên đủ. Ví dụ, có thể thực hiện chế độ định trước khi số lượng ứng viên tối đa bằng hoặc lớn hơn giá trị được thiết lập trước. Ví dụ, số lượng ứng viên tối đa có thể được thể hiện bởi tham số được gọi là `MaxNumModeCand`, và chế độ trong `MaxNumModeCand` có thể được thay thế bằng chế độ cần được thể hiện. Ví dụ, giá trị `MaxNumMergeCand` đối với chế độ hợp nhất có thể có mặt. Ví dụ, có thể có giá trị `MaxNumTriangleMergeCand` đối với chế độ hợp nhất hình chữ nhật. Ví dụ, có thể có giá trị `MaxNumSubblockMergeCand` đối với chế độ hợp nhất khối phụ. Ngoài ra, số lượng ứng viên tối đa có thể dựa vào báo hiệu mức cao hơn biểu thị chế độ cần được kích hoạt. Ví dụ, `MaxNumSubblockMergeCand` có thể dựa vào `sps_affine_enabled_flag` hoặc `sps_sbtmvp_enabled_flag`. `Sps_sbtmvp_enabled_flag` có thể là báo hiệu mức cao hơn biểu thị liệu phép dự đoán (véctor) chuyển động theo thời gian dựa trên khối phụ có sử dụng được hay không.

Theo một phương án của sáng chế, các điều kiện cần được thỏa mãn để thực hiện chế độ hợp nhất chính quy có thể có mặt. Ví dụ, báo hiệu biểu thị việc sử dụng chế độ hợp nhất cần phải đúng để kích hoạt việc thực thi chế độ hợp nhất chính quy. Báo hiệu biểu thị việc sử dụng chế độ hợp nhất có thể là `merge_flag` hoặc `general_merge_flag`. Ngoài ra, liên quan đến các chế độ khác được mô tả dưới đây, có thể thực hiện các chế độ khác chỉ khi báo hiệu biểu thị việc sử dụng chế độ hợp nhất là đúng.

Theo cách khác, không giống với các chế độ khác, các điều kiện cần được thỏa mãn để thực hiện chế độ hợp nhất chính quy có thể không có mặt. Điều này có thể xảy ra vì chế độ hợp nhất chính quy có thể là chế độ cơ bản nhất. Nếu chế độ hợp nhất được mô

tả ở trên được sử dụng, thì các điều kiện bổ sung để sử dụng chế độ hợp nhất chính quy có thể không được yêu cầu.

Dựa vào FIG.25, khi điều kiện thứ nhất 2501 được thỏa mãn, thì bộ giải mã có thể phân tích cú pháp cờ hợp nhất chính quy (nghĩa là, `regular_merge_flag`). Điều kiện thứ nhất 2501 có thể là trường hợp trong đó `sps_mmvd_enabled_flag` là 1 hoặc giá trị thu được bởi `chiều rộng*chiều cao` không phải là 32. Hơn nữa, khi điều kiện thứ nhất 2501 không được thỏa mãn, thì `regular_merge_flag` có thể không được phân tích cú pháp. Ở đây, bộ giải mã có thể suy ra giá trị của chúng là 1. Ví dụ, nếu `sps_mmvd_enabled_flag` là 0 && `chiều rộng*chiều cao == 32`, thì bộ giải mã có thể suy ra `regular_merge_flag` là 1. Theo cách khác, nếu điều kiện mà `general_merge_flag` là 1 có thể có trong trường hợp suy ra giá trị của chúng là 1. Điều này có thể là vì, khi điều kiện thứ nhất 2501 không được thỏa mãn, thì không có các điều kiện cần được thỏa mãn để thực hiện các chế độ khác thuộc về chế độ hợp nhất được thỏa mãn. Ngoài ra, khi `regular_merge_flag` không tồn tại, thì bộ giải mã có thể suy ra giá trị là 0 nếu điều kiện nêu trên suy ra giá trị là 1 không được thỏa mãn. Hơn nữa, `chiều rộng` hoặc `chiều cao` có thể được biểu diễn dưới dạng lũy thừa cơ số 2. Hơn nữa, `chiều rộng` hoặc `chiều cao` có thể là số dương. Do đó, giá trị 32, thu được bằng cách tính toán `chiều rộng*chiều cao`, có thể lần lượt biểu thị `chiều rộng` và `chiều cao` bằng 4 và 8 hoặc 8 và 4. Hơn nữa, nếu giá trị thu được bởi `chiều rộng*chiều cao` không phải là 32, thì `chiều rộng` và `chiều cao` có thể lần lượt không phải 4 và 8 cũng không phải 8 và 4. Hơn nữa, nếu giá trị thu được bởi `chiều rộng*chiều cao` không phải là 32, thì `chiều rộng` hoặc `chiều cao` có thể có giá trị lớn hơn hoặc bằng 8. Trường hợp này có thể tương ứng với phép dự đoán liên ảnh, và điều này có thể xảy ra vì phép dự đoán liên ảnh có thể không được cho phép đối với khối 4x4 chẳng hạn.

Theo một phương án của sáng chế, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi MMVD có thể có mặt. Ví dụ, các điều kiện có thể dựa vào báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không. Ví dụ, báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không có thể là `sps_mmvd_enabled_flag`. Dựa vào FIG.25, khi điều kiện thứ hai 2502 được thỏa mãn, thì `mmvd_merge_flag` có thể được phân tích cú pháp. Ngoài ra, nếu điều kiện thứ hai 2502 không được thỏa mãn, thì `MMVD_merge_flag` có thể không được phân tích cú pháp và giá trị của chúng có thể được suy ra. Điều kiện thứ hai 2502 có thể là $(\text{sps_mmvd_enabled_flag} \ \&\& \ \text{cbWidth*cbHeight} \neq 32)$. Nghĩa là, khi `sps_mmvd_enabled_flag` là 1 và điều kiện kích thước khối được thỏa mãn, thì `mmvd_merge_flag` có thể được phân tích cú pháp, và khi `sps_mmvd_enabled_flag` là 0 hoặc điều kiện kích thước khối không được thỏa mãn, thì

`mmvd_merge_flag` có thể không được phân tích cú pháp. Hơn nữa, khi `sps_mmvd_enabled_flag` là 1 và kích thước khối không được thỏa mãn, thì `mmvd_merge_flag` có thể được suy ra là 1. Ví dụ, khi `sps_mmvd_enabled_flag` là 1 và chiều rộng*chiều cao là 32, nếu `regular_merge_flag` là 0, và khi `general_merge_flag` là 1, thì `mmvd_merge_flag` có thể được suy ra là 1. Điều kiện kích thước khối có thể là liên quan đến điều kiện mà các chế độ được báo hiệu sau MMVD không sử dụng được.

Theo một phương án của sáng chế, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ hợp nhất khối phụ có thể có mặt. Ví dụ, các điều kiện có thể dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ hợp nhất khối phụ có sử dụng được hay không. Theo cách khác, các điều kiện có thể dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ thuộc về chế độ hợp nhất khối phụ có sử dụng được hay không. Ví dụ, chế độ hợp nhất khối phụ có thể bao gồm phép dự đoán chuyển động afin, phép dự đoán vectơ chuyển động theo thời gian dựa trên khối phụ và dạng tương tự. Theo đó, có thể xác định được liệu việc thực thi chế độ hợp nhất khối phụ có được kích hoạt hay không, dựa vào báo hiệu mức cao hơn (ví dụ, `sps_affine_enabled_flag`) biểu thị liệu phép dự đoán chuyển động afin có sử dụng được hay không.

Theo cách khác, có thể xác định được liệu việc thực thi chế độ hợp nhất khối phụ có được kích hoạt hay không, dựa vào báo hiệu mức cao hơn (ví dụ, `sps_sbtmvp_enabled_flag`) biểu thị liệu phép dự đoán vectơ chuyển động theo thời gian dựa trên khối phụ có sử dụng được hay không. Theo cách khác, để kích hoạt việc thực thi chế độ hợp nhất khối phụ, điều kiện dựa vào số lượng ứng viên tối đa đối với chế độ hợp nhất khối phụ có thể cần được thỏa mãn. Ví dụ, khi giá trị của số lượng ứng viên tối đa đối với chế độ hợp nhất khối phụ lớn hơn 0, thì chế độ hợp nhất khối phụ có thể được sử dụng. Ngoài ra, số lượng ứng viên tối đa đối với chế độ hợp nhất khối phụ có thể dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ thuộc về chế độ hợp nhất khối phụ có sử dụng được hay không. Ví dụ, có thể biết rằng số lượng ứng viên tối đa đối với chế độ hợp nhất khối phụ lớn hơn 0 chỉ khi ít nhất một báo hiệu trong số báo hiệu mức cao hơn biểu thị liệu chế độ thuộc về các chế độ hợp nhất khối phụ có sử dụng được hay không là 1. Ngoài ra, để kích hoạt việc thực thi chế độ hợp nhất khối phụ, điều kiện dựa vào kích thước khối có thể cần được thỏa mãn. Ví dụ, có thể có giới hạn dưới đối với chiều rộng và chiều cao. Ví dụ, có thể sử dụng chế độ hợp nhất khối phụ khi chiều rộng lớn hơn hoặc bằng 8 và chiều cao lớn hơn hoặc bằng 8.

Dựa vào FIG.25, nếu điều kiện thứ ba 2503 được thỏa mãn, thì chế độ hợp nhất khối phụ có thể được phân tích cú pháp. Hơn nữa, nếu điều kiện thứ ba 2503 không được thỏa

mãn, thì `merge_subblock_flag` có thể không được phân tích cú pháp, và giá trị của chúng có thể được suy ra là 0. Điều kiện thứ ba 2503 có thể là $(\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8)$.

Theo một phương án của sáng chế, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi CIIP có thể mặt. Ví dụ, có thể xác định được liệu việc thực thi CIIP có được kích hoạt dựa vào báo hiệu mức cao hơn hay không (ví dụ, `spsXBT_ciip_enabled_flag`) biểu thị liệu CIIP có sử dụng được hay không. Ngoài ra, có thể xác định được liệu việc thực thi CIIP có được kích hoạt hay không dựa vào việc liệu chế độ bỏ qua có được sử dụng hay không. Ví dụ, khi chế độ bỏ qua được sử dụng, thì CIIP có thể không được thực hiện. Hơn nữa, có thể xác định được liệu việc thực thi CIIP có được kích hoạt dựa vào kích thước khối hay không. Ví dụ, có thể xác định được liệu việc thực thi CIIP có được kích hoạt hay không dựa vào việc liệu kích thước khối bằng hay lớn hơn giới hạn dưới và bằng hay nhỏ hơn giới hạn trên. Ví dụ, việc thực thi CIIP được kích hoạt khi giá trị thu được bởi $\text{chiều rộng} \times \text{chiều cao}$ bằng hoặc lớn hơn giới hạn dưới, giá trị của chiều rộng bằng hoặc thấp hơn giới hạn trên, và giá trị của chiều cao bằng hoặc thấp hơn giới hạn trên. Ví dụ, khi giá trị thu được bởi $\text{chiều rộng} \times \text{chiều cao}$ bằng hoặc lớn hơn 64, giá trị của chiều rộng nhỏ hơn 128 và giá trị của chiều cao nhỏ hơn 128, có thể thực thi CIIP.

Dựa vào FIG.25, cờ CIIP có thể được phân tích cú pháp khi điều kiện thứ tư 2504 được thỏa mãn. Hơn nữa, khi điều kiện thứ tư 2504 không được thỏa mãn, thì cờ CIIP có thể không được phân tích cú pháp, và giá trị của chúng có thể được suy ra là 0. Điều kiện thứ tư 2504 có thể là $(\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} \times \text{chiều cao} \geq 64 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128)$.

Theo một phương án của sáng chế, các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ hợp nhất hình chữ nhật có thể có mặt. Ví dụ, có thể xác định được liệu việc thực thi chế độ hợp nhất hình chữ nhật có được kích hoạt dựa vào báo hiệu mức cao hơn hay không (ví dụ, `sps_triangle_enabled_flag`) biểu thị liệu chế độ hợp nhất hình chữ nhật có sử dụng được hay không. Hơn nữa, có thể xác định được liệu việc thực thi chế độ hợp nhất hình chữ nhật có được kích hoạt dựa vào kiểu lát hay không. Ví dụ, khi kiểu lát là B, thì có thể thực hiện chế độ hợp nhất hình chữ nhật. Điều này có thể là vì hai hoặc nhiều hơn hai thông tin chuyển động hoặc hai hoặc nhiều hơn hai ảnh tham chiếu được yêu cầu để thực hiện chế độ hợp nhất hình chữ nhật. Hơn nữa, có thể xác định được liệu việc thực thi chế độ hợp nhất hình chữ nhật có được kích hoạt hay không dựa vào số lượng ứng viên tối đa đối với chế độ hợp nhất hình chữ nhật. Số lượng ứng viên tối đa đối với chế độ hợp nhất hình chữ nhật có thể được biểu diễn dưới dạng giá trị

MaxNumTriangleMergeCand. Ví dụ, khi số lượng ứng viên tối đa đối với chế độ hợp nhất hình chữ nhật lớn hơn hoặc bằng 2, thì có thể thực hiện chế độ hợp nhất hình chữ nhật. Điều này có thể là vì hai hoặc nhiều hơn hai thông tin chuyển động hoặc hai hoặc nhiều hơn hai ảnh tham chiếu được yêu cầu để thực hiện chế độ hợp nhất hình chữ nhật. Ngoài ra, theo một phương án của sáng chế, khi báo hiệu mức cao hơn, biểu thị liệu chế độ hợp nhất hình chữ nhật có sử dụng được hay không, biểu thị tính sử dụng được của chế độ này, thì số lượng ứng viên tối đa đối với chế độ hợp nhất hình chữ nhật luôn lớn hơn hoặc bằng 2, và khi báo hiệu mức cao hơn, biểu thị liệu chế độ hợp nhất hình chữ nhật có sử dụng được hay không, biểu thị tính không sử dụng được của chế độ này, thì số lượng ứng viên tối đa đối với chế độ hợp nhất hình chữ nhật có thể luôn nhỏ hơn 2 hoặc 0.

Theo đó, ở đây, có thể xác định được liệu việc thực thi chế độ hợp nhất hình chữ nhật có được kích hoạt hay không dựa vào số lượng ứng viên tối đa đối với chế độ hợp nhất hình chữ nhật thay vì dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ hợp nhất hình chữ nhật sử dụng được. Theo đó, có thể giảm các thao tác kiểm tra các điều kiện. Hơn nữa, có thể xác định được liệu việc thực thi chế độ hợp nhất hình chữ nhật có được kích hoạt dựa vào kích thước khối hay không. Ví dụ, có thể xác định được liệu việc thực thi chế độ hợp nhất hình chữ nhật có được kích hoạt hay không dựa vào việc liệu kích thước khối bằng hay lớn hơn giới hạn dưới và bằng hay nhỏ hơn giới hạn trên. Ví dụ, có thể thực hiện chế độ hợp nhất hình chữ nhật nếu giá trị thu được bởi chiều rộng*chiều cao bằng hoặc lớn hơn giới hạn dưới, giá trị của chiều rộng bằng hoặc thấp hơn giới hạn trên và giá trị của chiều cao bằng hoặc thấp hơn giới hạn trên. Ví dụ, có thể thực hiện chế độ hợp nhất hình tam giác nếu giá trị thu được bởi chiều rộng*chiều cao lớn hơn hoặc bằng 64. Có thể thực hiện chế độ hợp nhất hình tam giác nếu giá trị của chiều rộng nhỏ hơn 128 và giá trị của chiều cao nhỏ hơn 128.

Dựa vào FIG.25, khi điều kiện thứ năm 2505 được thỏa mãn, thì chế độ hợp nhất hình chữ nhật có thể được sử dụng. Hơn nữa, khi điều kiện thứ năm 2505 không được thỏa mãn, thì chế độ hợp nhất hình chữ nhật có thể được sử dụng. Điều kiện thứ năm 2505 có thể tương ứng với trường hợp trong đó MergeTriangleFlag là 1. Ngoài ra, các điều kiện cần được thỏa mãn để thỏa mãn điều kiện thứ năm 2505 có thể có mặt. Ví dụ, điều kiện có thể bao gồm (`sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2 && chiều rộng*chiều cao >= 64`). Nếu (`sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2 && chiều rộng*chiều cao >= 64`) là đúng, thì MergeTriangleFlag có thể là 1 hoặc 0. Ở đây, việc liệu MergeTriangleFlag là 1 hay 0 có thể được xác định theo điều kiện bổ sung.

Điều kiện bổ sung có thể bao gồm trường hợp trong đó không có các chế độ (ví dụ, chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ hoặc CIIP) mà được báo hiệu hoặc xác định trước chế độ hợp nhất hình chữ nhật được sử dụng và trường hợp trong đó chế độ hợp nhất được sử dụng (ví dụ, `general_merge_flag == 1`). `MergeTriangleFlag` có thể là 1 nếu điều kiện bổ sung được thỏa mãn, và `MergeTriangleFlag` có thể là 0 nếu điều kiện bổ sung không được thỏa mãn. Hơn nữa, khi (`sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2 && chiều rộng*chiều cao >= 64`) là sai, thì `MergeTriangleFlag` có thể là 0.

FIG.26 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. FIG.26 có thể là một phương án cụ thể về phương pháp báo hiệu được mô tả dựa vào các hình vẽ từ FIG.12 đến FIG.13.

Theo một phương án của sáng chế, khi ít nhất một chế độ trong số các chế độ được báo hiệu hoặc xác định muộn hơn chế độ định trước có thể được thực hiện, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ định trước hay không có thể được phân tích cú pháp. Ngoài ra, khi không có các chế độ được báo hiệu hoặc xác định muộn hơn chế độ định trước có thể được thực hiện, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ định trước hay không có thể không được phân tích cú pháp. Ngoài ra, khi không có các chế độ được báo hiệu hoặc xác định muộn hơn chế độ định trước có thể được thực hiện, thì có thể được xác định có sử dụng giá trị của báo hiệu liên quan đến việc liệu có sử dụng chế độ định trước hay không.

Ngoài ra, liệu việc thực thi chế độ được báo hiệu hoặc xác định muộn hơn chế độ định trước được kích hoạt hay hủy kích hoạt có thể tùy thuộc vào việc liệu các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ được mô tả trên FIG.26 có được thỏa mãn hay không. Theo cách khác, liệu việc thực thi chế độ được báo hiệu hoặc xác định muộn hơn chế độ định trước được kích hoạt hay hủy kích hoạt có thể tùy thuộc vào việc liệu một số trong số các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ được mô tả trên FIG.26 có được thỏa mãn hay không. Ví dụ, một số trong số các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ có thể được lược bỏ khi xác định báo hiệu đối với chế độ khác. Điều này có thể dẫn đến việc giảm các thao tác kiểm tra các điều kiện, ví dụ, có thể lược bỏ các điều kiện mà thường đúng.

Cụ thể hơn là, để sử dụng chế độ định trước, báo hiệu mức cao hơn biểu thị chế độ sử dụng được cần phải đúng, và kiểu lát cần có giá trị cụ thể. Tuy nhiên, khi kiểm tra điều kiện sử dụng chế độ định trước để xác định việc phân tích báo hiệu liên quan đến việc liệu

có sử dụng chế độ khác với chế độ định trước hay không, có thể xác định việc phân tích báo hiệu liên quan đến việc liệu có sử dụng chế độ khác hay không dựa vào báo hiệu mức cao hơn thay vì dựa vào kiểu lát. Điều này có thể là vì kiểu lát thường có giá trị cụ thể. Theo đó, ngay cả khi kiểu lát không có giá trị cụ thể, nếu báo hiệu mức cao hơn biểu thị chế độ sử dụng được là đúng, thì báo hiệu liên quan đến việc liệu có sử dụng chế độ khác với chế độ định trước hay không có thể được phân tích cú pháp.

Dựa vào FIG.26, `mmvd_condition`, `subblock_merge_condition`, `ciip_condition` và `triangle_merge_condition` có thể có mặt. Ví dụ, `mmvd_condition`, `subblock_merge_condition`, `ciip_condition` và `triangle_merge_condition` có thể là các điều kiện cần được thỏa mãn để lần lượt thực hiện MMVD, chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình tam giác được mô tả dựa vào FIG.25. Theo cách khác, `mmvd_condition`, `subblock_merge_condition`, `ciip_condition` và `triangle_merge_condition` có thể là một phần của các điều kiện cần được thỏa mãn để lần lượt thực hiện MMVD, chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình tam giác được mô tả dựa vào FIG.25. Ví dụ, `mmvd_condition`, `subblock_merge_condition` và `ciip_condition` có thể lần lượt là điều kiện thứ hai 2502, điều kiện thứ ba 2503 và điều kiện thứ tư 2504 được mô tả dựa vào FIG.25, hoặc một phần của các điều kiện này.

Hơn nữa, nếu các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ được báo hiệu hoặc xác định muộn hơn chế độ định trước và các điều kiện cần được thỏa mãn để kích hoạt việc thực thi trùng lặp chế độ định trước, thì các điều kiện cần được thỏa mãn để kích hoạt việc thực thi chế độ định trước cần được thỏa mãn để kích hoạt chế độ định trước cần được sử dụng. Do đó, các điều kiện trùng lặp có thể được loại trừ khỏi `mmvd_condition`, `subblock_merge_condition`, `ciip_condition`, `triangle_merge_condition` và dạng tương tự trên FIG.26.

Dựa vào FIG.26, khi `triangle_merge_condition` được thỏa mãn, thì `ciip_flag` có thể được phân tích cú pháp. Hơn nữa, `ciip_flag` có thể không được phân tích cú pháp nếu `triangle_merge_condition` không được thỏa mãn. Hơn nữa, nếu `triangle_merge_condition` không được thỏa mãn, thì `ciip_flag` có thể được suy ra là 1. Ở đây, để suy ra giá trị của chúng là 1, thì điều kiện để thực hiện CIIP cần được thỏa mãn, không có các chế độ được báo hiệu hoặc xác định trước CIIP được sử dụng (ví dụ, `regular_merge_flag == 0 && mmvd_merge_flag == 0 && merge_subblock_flag == 0`), và nếu chế độ hợp nhất được sử dụng, thì điều kiện của (`general_merge_flag == 1`) cần được thỏa mãn. Ngược lại, nếu `ciip_flag` không tồn tại, thì giá trị của chúng có thể được suy ra là 0.

Dựa vào FIG.26, nếu `ciip_condition` được thỏa mãn hoặc `triangle_merge_condition` được thỏa mãn, thì `merge_subblock_flag` có thể được phân tích cú pháp. Hơn nữa, nếu `ciip_condition` không được thỏa mãn và `triangle_merge_coondition` không được thỏa mãn, thì `merge_subblock_flag` có thể không được phân tích cú pháp. Hơn nữa, nếu `ciip_condition` không được thỏa mãn, và `triangle_merge_coondition` không được thỏa mãn, thì `merge_subblock_flag` có thể được suy ra là 1. Ở đây, để suy ra giá trị của chúng là 1, thì điều kiện để thực hiện chế độ hợp nhất khối phụ cần được thỏa mãn, không có các chế độ được báo hiệu hoặc xác định trước chế độ hợp nhất khối phụ được sử dụng (ví dụ, `regular_merge_flag == 0 && mmvd_merge_flag == 0`), và nếu chế độ hợp nhất được sử dụng, thì điều kiện của (`general_merge_flag == 1`) cần được thỏa mãn. Ngược lại, nếu `merge_subblock_flag` không tồn tại, thì giá trị của chúng có thể được suy ra là 0.

Dựa vào FIG.26, khi `subblock_merge_condition` được thỏa mãn, `ciip_condition` được thỏa mãn hoặc `triangle_merge_condition` được thỏa mãn, thì `mmvd_merge_flag` có thể được phân tích cú pháp. Hơn nữa, nếu `subblock_merge_condition` không được thỏa mãn, `ciip_condition` không được thỏa mãn và `triangle_merge_coondition` không được thỏa mãn, thì `mmvd_merge_flag` có thể không được phân tích cú pháp. Hơn nữa, nếu `subblock_merge_condition` không được thỏa mãn, `ciip_condition` không được thỏa mãn và `triangle_merge_coondition` không được thỏa mãn, thì `mmvd_merge_flag` có thể được suy ra là 1. Ở đây, để suy ra giá trị của `mmvd_merge_flag` là 1, thì các điều kiện để thực thi MMVD cần được thỏa mãn, không có các chế độ được báo hiệu hoặc xác định trước MMVD được sử dụng (ví dụ, `regular_merge_flag == 0`), và nếu chế độ hợp nhất được sử dụng, thì điều kiện của (`general_merge_flag == 1`) có thể cần được thỏa mãn. Hơn nữa, có thể có trường hợp trong đó `MMVD_merge_flag` được mô tả dựa vào FIG.66 được suy ra là 1. Ngược lại, nếu `mmvd_merge_flag` không tồn tại, thì giá trị của chúng có thể được suy ra là 0.

Dựa vào FIG.26, khi `mmvd_condition` được thỏa mãn, `subblock_merge_condition` được thỏa mãn, `ciip_condition` được thỏa mãn hoặc `triangle_merge_condition` được thỏa mãn, thì `regular_merge_flag` có thể được phân tích cú pháp. Hơn nữa, nếu `mmvd_condition` không được thỏa mãn, `subblock_merge_condition` không được thỏa mãn, `ciip_condition` không được thỏa mãn và `triangle_merge_coondition` không được thỏa mãn, thì `regular_merge_flag` có thể không được phân tích cú pháp. Hơn nữa, nếu `mmvd_condition` không được thỏa mãn, `subblock_merge_condition` không được thỏa mãn, `ciip_condition` không được thỏa mãn và `triangle_merge_coondition` không được thỏa mãn, thì `regular_merge_flag` có thể được suy ra là 1. Ở đây, để suy ra giá trị của

chúng là 1, các điều kiện để thực thi chế độ hợp nhất chính quy cần được thỏa mãn (điều kiện này có thể không có mặt đối với chế độ hợp nhất chính quy), không có các chế độ được báo hiệu hoặc xác định trước chế độ hợp nhất chính quy được sử dụng (điều kiện này có thể không có mặt đối với chế độ hợp nhất chính quy), và nếu chế độ hợp nhất được sử dụng, thì điều kiện của (`general_merge_flag == 1`) cần được thỏa mãn. Hơn nữa, có thể có trường hợp trong đó `regular_merge_flag` được mô tả dựa vào FIG.25 được suy ra là 1. Ngược lại, nếu `regular_merge_flag` không tồn tại, thì giá trị của chúng có thể được suy ra là 0.

Ngoài ra, FIG.26 minh họa báo hiệu theo việc liệu bước thực thi chế độ được báo hiệu hoặc xác định muộn hơn các chế độ có được kích hoạt hay không. Phương pháp báo hiệu sử dụng được chỉ đối với một số trong số các chế độ. Nghĩa là, ít nhất một điều kiện trong số điều kiện thứ nhất 2501, điều kiện thứ hai 2502, điều kiện thứ ba 2503 và điều kiện thứ tư 2504 trên FIG.25 có thể được sử dụng, và phương pháp trên FIG.26 có thể được sử dụng cho phần còn lại. Nghĩa là, điều kiện thứ nhất 2501 trên FIG.25, và điều kiện thứ hai 2602, điều kiện thứ ba 2603 và điều kiện thứ tư 2604 trên FIG.26 có thể được sử dụng.

FIG.27 minh họa cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. FIG.27 có thể là một phương án cụ thể về phương pháp báo hiệu được mô tả dựa vào FIG.12, FIG.13 và FIG.26. Dựa vào FIG.27, `ciip_flag` có thể được phân tích cú pháp khi (`sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2`). Ở đây, chỉ có thể phân tích cú pháp `CIIP_flag` khi điều kiện để thực hiện CIIP được thỏa mãn. Hơn nữa, nếu điều kiện của (`sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2`) không được thỏa mãn, thì `ciip_flag` có thể không được phân tích cú pháp. Hơn nữa, nếu điều kiện của (`sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2`) không được thỏa mãn, thì `ciip_flag` có thể được suy ra là 1. Ở đây, `ciip_flag` có thể được suy ra là 1 khi chế độ hợp nhất được sử dụng, các điều kiện để thực thi CIIP được thỏa mãn, và không có các chế độ được báo hiệu hoặc xác định trước CIIP được sử dụng.

Ví dụ, `ciip_flag` có thể được suy ra là 1 trong trường hợp đối với (`general_merge_flag == 1 && sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng*chiều cao >= 64 && chiều rộng < 128 && chiều cao < 128 && regular_merge_flag == 0 && mmvd_triangle_flag == 0 && merge_subblock_flag == 0`), và nếu điều kiện của (`sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2`) không được thỏa mãn. Ở đây, chỉ một số điều kiện trong số (`sps_triangle_enabled_flag`

$\&\& \text{slice_type} == B \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) có thể được sử dụng. Nếu chỉ một số điều kiện được sử dụng, thì một số điều kiện để xác định liệu có phân tích cú pháp hay không và một số điều kiện được sử dụng khi suy ra có thể cần phải phù hợp. Ví dụ, là một ví dụ về việc không sử dụng slice_type , ciip_flag có thể được phân tích cú pháp khi thỏa mãn ($\text{sps_triangle_enabled_flag} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$), và ciip_flag có thể không được phân tích cú pháp nếu ($\text{sps_triangle_enabled_flag} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) không được thỏa mãn. Hơn nữa, chế độ hợp nhất được sử dụng nếu ($\text{sps_triangle_enabled_flag} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) không được thỏa mãn, và ciip_flag có thể được suy ra là 1 khi các điều kiện để thực thi CIIP được thỏa mãn và không có các chế độ được báo hiệu hoặc xác định trước CIIP được sử dụng.

Dựa vào FIG.27, trong trường hợp đối với ($\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == B \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) $\parallel$ ($\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128$)), $\text{merge_subblock_flag}$ có thể được phân tích cú pháp. Ở đây, chỉ có thể phân tích cú pháp $\text{merge_subblock_flag}$ khi các điều kiện để thực thi chế độ hợp nhất khối phụ được thỏa mãn. Hơn nữa, nếu ($\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == B \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) $\parallel$ ($\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128$)) không được thỏa mãn, thì $\text{merge_subblock_flag}$ có thể không được phân tích cú pháp. Hơn nữa, nếu ($\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == B \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) $\parallel$ ($\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128$)) không được thỏa mãn, thì $\text{merge_subblock_flag}$ có thể được suy ra là 1. Ở đây, $\text{merge_subblock_flag}$ có thể được suy ra là 1 khi chế độ hợp nhất được sử dụng, các điều kiện để thực thi chế độ hợp nhất khối phụ được thỏa mãn, và không có các chế độ được báo hiệu hoặc xác định trước chế độ hợp nhất khối phụ được sử dụng.

Ví dụ, $\text{merge_subblock_flag}$ có thể được suy ra là 1 trong trường hợp đối với ($\text{general_merge_flag} == 1 \ \&\& \text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8 \ \&\& \text{regular_merge_flag} == 0 \ \&\& \text{mmvd_merge_flag} == 0$), và nếu (($\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == B \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) $\parallel$ ($\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128$)) không được thỏa mãn. Ở đây, chỉ một số điều kiện của ($\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == B \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) có thể được sử dụng và chỉ một số điều kiện của ($\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128$

&& chiều cao < 128) có thể được sử dụng. Khi sử dụng chỉ một số điều kiện, một số điều kiện để xác định liệu có phân tích cú pháp hay không và một số điều kiện được sử dụng khi suy luận có thể cần phải phù hợp. Hơn nữa, trường hợp trong đó (sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128)) không được thỏa mãn, đã được mô tả ở trên, có thể giống với trường hợp trong đó (sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) không được thỏa mãn (&&), và trường hợp trong đó (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) không được thỏa mãn.

Dựa vào FIG.27, trong trường hợp đối với ((sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều cao >= 8 && chiều cao >= 8)), mmvd_merge_flag có thể được phân tích cú pháp. Ở đây, chỉ có thể phân tích cú pháp MMVD_merge_flag khi các điều kiện để thực thi MMVD được thỏa mãn. Ngoài ra, mmvd_merge_flag có thể không được phân tích cú pháp nếu ((sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều rộng >= 8)) không được thỏa mãn. Ngoài ra, nếu ((sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều rộng >= 8)) không được thỏa mãn, thì mmvd_merge_flag có thể được suy ra là 1.

Ở đây, mmvd_merge_flag có thể được suy ra là 1 khi chế độ hợp nhất được sử dụng, các điều kiện để thực thi MMVD được thỏa mãn, và không có các chế độ được báo hiệu hoặc xác định trước MMVD được sử dụng. Ví dụ, trong trường hợp đối với (general_merge_flag == 1 && sps_mmvd_enabled_flag && regular_merge_flag == 0), và nếu ((sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều rộng >= 8 && chiều cao >= 8)) không được thỏa mãn, thì mmvd_merge_flag có thể được suy ra là 1. Ở đây, chỉ một số điều kiện (sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) sử dụng được, chỉ một số điều kiện (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) không được thỏa mãn.

128) sử dụng được, và chỉ một số điều kiện ($\text{MaxNumSubblockMergeCand} \geq 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8$) sử dụng được. Khi sử dụng chỉ một số điều kiện, một số điều kiện để xác định liệu có phân tích cú pháp hay không và một số điều kiện được sử dụng khi suy luận có thể cần phải phù hợp. Ngoài ra, trường hợp trong đó ($(\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2) \parallel (\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128) \parallel (\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8)$) không thỏa mãn, đã được mô tả ở trên, giống với trường hợp trong đó ($\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2$) không được thỏa mãn ($\&\&$), trường hợp trong đó ($\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128$) không được thỏa mãn ($\&\&$), và trường hợp trong đó ($\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8$) không được thỏa mãn.

Như được mô tả dựa vào FIG.25, có thể có trường hợp trong đó mmvd_merge_flag được suy ra là 1. Ví dụ, trong trường hợp đối với ($\text{sps_mmvd_enabled_flag} == 1 \ \&\& \text{general_merge_flag} == 1 \ \&\& \text{chiều rộng} * \text{chiều cao} == 32 \ \&\& \text{regular_merge_flag} == 0$), mmvd_merge_flag có thể được suy ra là 1. Ngoài ra, như được mô tả trên FIG.26, trong trường hợp đối với ($\text{general_merge_flag} == 1 \ \&\& \text{sps_mmvd_enabled_flag} \ \&\& \text{regular_merge_flag} == 0$), và nếu ($(\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2) \parallel (\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128) \parallel (\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8)$) không được thỏa mãn, thì mmvd_merge_flag có thể được suy ra là 1. Ngược lại, mmvd_merge_flag có thể được suy ra là 0.

Dựa vào FIG.27, trong trường hợp đối với ($(\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2) \parallel (\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128) \parallel (\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8) \parallel \text{sps_mmvd_enabled_flag}$), $\text{regular_merge_flag}$ có thể được phân tích cú pháp. Ở đây, chỉ có thể phân tích cú pháp $\text{regular_merge_flag}$ khi không chỉ tồn tại chế độ hợp nhất chính quy mà còn tồn tại chế độ hợp nhất khả thi. Ngoài ra, nếu ($(\text{sps_triangle_enabled_flag} \ \&\& \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2) \parallel (\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128) \parallel (\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8) \parallel \text{sps_mmvd_enabled_flag}$) không được thỏa mãn, thì $\text{regular_merge_flag}$ có thể không

được phân tích cú pháp.

Ngoài ra, nếu $((\text{sps_triangle_enabled_flag} \ \&\& \ \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2) \parallel (\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128) \parallel (\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8) \parallel \text{sps_mmvd_enabled_flag})$ không được thỏa mãn, thì $\text{regular_merge_flag}$ có thể được suy ra là 1. Ở đây, khi chế độ hợp nhất được sử dụng, thì $\text{regular_merge_flag}$ có thể được suy ra là 1. Ví dụ, trong trường hợp đối với $(\text{general_merge_flag} == 1)$, và nếu $((\text{sps_triangle_enabled_flag} \ \&\& \ \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2) \parallel (\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128) \parallel (\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8) \parallel \text{sps_mmvd_enabled_flag})$ không được thỏa mãn, thì $\text{regular_merge_flag}$ có thể được suy ra là 1. Ở đây, chỉ một số điều kiện $(\text{sps_triangle_enabled_flag} \ \&\& \ \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2)$ có thể được sử dụng, chỉ một số điều kiện $(\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128)$ có thể được sử dụng, và chỉ một số điều kiện $(\text{MaxNumSubblockMergeCand} \geq 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8)$ có thể được sử dụng. Khi sử dụng chỉ một số điều kiện, một số điều kiện để xác định liệu có phân tích cú pháp hay không và một số điều kiện được sử dụng khi suy luận có thể cần phải phù hợp.

Cụ thể hơn là, bộ mã hóa có độ phức tạp thấp có thể không sử dụng các công cụ hợp nhất khác nhau, và đối với bộ mã hóa, nếu $(\text{sps_triangle_enabled_flag} \parallel \text{sps_affine_enabled_flag} \parallel \text{sps_sbtmvp_enabled_flag} \parallel \text{sps_ciip_enabled_flag} \parallel \text{sps_mmvd_enabled_flag})$ không được thỏa mãn, thì $\text{regular_merge_flag}$ có thể không được phân tích cú pháp và giá trị của chúng có thể được suy ra là 1. Ngoài ra, trường hợp trong đó $((\text{sps_triangle_enabled_flag} \ \&\& \ \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2) \parallel (\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128) \parallel (\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8) \parallel \text{sps_mmvd_enabled_flag})$ không được thỏa mãn, đã được mô tả ở trên, có thể giống với trường hợp trong đó $(\text{sps_triangle_enabled_flag} \ \&\& \ \text{slice_type} == \text{B} \ \&\& \text{MaxNumTriangleMergeCand} \geq 2)$ không được thỏa mãn ($\&\&$), trường hợp trong đó $(\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{chiều rộng} < 128 \ \&\& \text{chiều cao} < 128)$ không được thỏa mãn ($\&\&$), trường hợp trong đó $(\text{MaxNumSubblockMergeCand} > 0 \ \&\& \text{chiều rộng} \geq 8 \ \&\& \text{chiều cao} \geq 8)$ không được thỏa mãn ($\&\&$) và trường hợp trong đó $\text{sps_mmvd_enabled_flag}$ không được thỏa mãn.

Như được mô tả dựa vào FIG.25, có thể có trường hợp trong đó `regular_merge_flag` được suy ra là 1. Ví dụ, trong trường hợp đối với (`sps_mmvd_enabled_flag == 0 && general_merge_flag == 1 && chiều rộng*chiều cao == 32`), `regular_merge_flag` có thể được suy ra là 1. Hơn nữa, như được mô tả trên FIG.68, trong trường hợp đối với (`general_merge_flag == 1`), và nếu (`(sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều rộng >= 8 && chiều cao >= 8) || sps_mmvd_enabled_flag`) không được thỏa mãn, thì `regular_merge_flag` có thể được suy ra là 1. Ngược lại, `regular_merge_flag` có thể được suy ra là 0.

Hơn nữa, vì điều kiện thứ nhất 2701 và điều kiện thứ hai 2702 trên FIG.27 có thể bao gồm các điều kiện (1) `chiều rộng*chiều cao >= 64`, và (2) `chiều rộng >= 8` và `chiều cao >= 8`, liên quan đến các điều kiện (1) hoặc (2), `chiều rộng` và `chiều cao` có thể lần lượt không phải là 4 và 8 cũng không phải là 8 và 4. Do đó, không cần phải kiểm tra liệu `chiều rộng*chiều cao` có phải là 32 ở điều kiện thứ nhất 2701 và điều kiện thứ hai 2702 hay không. Do đó, điều kiện thứ hai 2702 có thể chỉ bao gồm (`(sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều rộng >= 8 && chiều cao >= 8) || sps_mmvd_enabled_flag`). Ở đây, nếu `general_merge_flag` là 1, và nếu (`(sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều rộng >= 8 && chiều cao >= 8) || sps_mmvd_enabled_flag`) không được thỏa mãn, thì `regular_merge_flag` có thể được suy ra là 1. Ngược lại, `regular_merge_flag` có thể được suy ra là 0.

Hơn nữa, điều kiện thứ hai 2702 có thể chỉ bao gồm `sps_mmvd_enabled_flag && ((sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều rộng >= 8 && chiều cao >= 8))`. Ở đây, trong trường hợp đối với (`general_merge_flag == 1 && sps_mmvd_enabled_flag && regular_merge_flag == 0`), và nếu (`(sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) || (sps_ciip_enabled_flag && cu_skip_flag == 0 && chiều rộng < 128 && chiều cao < 128) || (MaxNumSubblockMergeCand > 0 && chiều rộng >= 8 && chiều cao >= 8)`) không được thỏa mãn, thì `mmvd_merge_flag` có

thể được suy ra là 1. Ngược lại, `mmvd_merge_flag` có thể được suy ra là 0.

Hơn nữa, như được mô tả dựa vào FIG.26, phương pháp báo hiệu có thể được sử dụng chỉ đối với một số trong số các chế độ trên FIG.27. Nghĩa là, một số trong số điều kiện thứ nhất 2501, điều kiện thứ hai 2502, điều kiện thứ ba 2503 và điều kiện thứ tư 2504 trên FIG.25 có thể được sử dụng, và phương pháp trên FIG.27 có thể được sử dụng cho phần còn lại. Nghĩa là, điều kiện thứ nhất 2501 trên FIG.25, điều kiện thứ hai 2702, điều kiện thứ ba 2703 và điều kiện thứ tư 2704 trên FIG.27 có thể được sử dụng.

FIG.28 minh họa một ví dụ về cấu trúc cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. FIG.28 có thể là một phương án cụ thể về phương pháp báo hiệu được mô tả dựa vào FIG.12, FIG.13 và FIG.26. Ngoài ra, theo một phương án trên FIG.28, các điều kiện trùng lặp với các điều kiện theo phương án được mô tả trên FIG.27 có thể được loại trừ.

Dựa vào FIG.28, so với FIG.27, điều kiện thứ hai 2802 có thể không bao gồm điều kiện chiều rộng*chiều cao ≥ 64 . Điều này có thể xảy ra vì, nếu điều kiện (chiều rộng ≥ 8 & chiều cao ≥ 8) không được thỏa mãn, thì điều kiện chiều rộng*chiều cao ≥ 64 không được thỏa mãn và do đó điều kiện chiều rộng*chiều cao $\neq 32$ đã tồn tại. Nghĩa là, điều này có thể xảy ra vì chiều rộng*chiều cao ≥ 64 có thể luôn được thỏa mãn theo các điều kiện khác.

Hơn nữa, như được mô tả dựa vào FIG.26, phương pháp báo hiệu có thể được sử dụng chỉ đối với một số trong số các chế độ trên FIG.28. Nghĩa là, một số trong số điều kiện thứ nhất 2501, điều kiện thứ hai 2502, điều kiện thứ ba 2503 và điều kiện thứ tư 2504 trên FIG.25 có thể được sử dụng, và phương pháp trên FIG.28 có thể được sử dụng cho phần còn lại. Nghĩa là, điều kiện thứ nhất 2501 trên FIG.25, và điều kiện thứ hai 2802, điều kiện thứ ba 2803 và điều kiện thứ tư 2804 trên FIG.28 có thể được sử dụng. Theo cách khác, các điều kiện từ thứ nhất đến thứ tư trên các hình vẽ từ FIG.25 đến FIG.28 có thể được sử dụng thay thế cho nhau. Nghĩa là, điều kiện thứ nhất 2701 trên FIG.27, và điều kiện thứ hai 2802, điều kiện thứ ba 2801 và điều kiện thứ tư 2801 trên FIG.28 có thể được sử dụng.

FIG.29 minh họa một ví dụ về phương pháp báo hiệu chế độ hợp nhất theo một phương án của sáng chế. Trong báo hiệu chế độ hợp nhất được mô tả trước đó, phương pháp báo hiệu tuần tự đã được mô tả. Ví dụ, phương pháp báo hiệu tuần tự như được thể hiện trên FIG.25 có thể được sử dụng. FIG.29 (a) thể hiện phương pháp báo hiệu tuần tự. Trên FIG.29, phần in đậm biểu thị chế độ cần được xác định, và phần in nghiêng biểu thị

báo hiệu. Báo hiệu này có thể là cờ và có thể có giá trị 0 hoặc 1.

Ngoài ra, báo hiệu rõ ràng hoặc báo hiệu ngầm có thể được thực hiện tùy thuộc vào các trường hợp. Ví dụ, trên FIG.29(a), `regular_merge_flag` có thể được báo hiệu, và dựa vào giá trị `regular_merge_flag`, liệu có thể xác định được đó có phải là chế độ hợp nhất chính quy hay không. Nếu xác định được đây không phải là chế độ hợp nhất chính quy theo `regular_merge_flag`, thì `mmvd_merge_flag` có thể được báo hiệu, và dựa vào `mmvd_merge_flag`, liệu có thể xác định được đây có phải là MMVD hay không. Nếu xác định được đây không phải là MMVD, thì `merge_subblock_flag` có thể được báo hiệu, và dựa vào `merge_subblock_flag`, liệu có thể xác định được đây có phải là chế độ hợp nhất khối phụ hay không. Nếu xác định được đây không phải là chế độ hợp nhất khối phụ, thì `ciip_flag` có thể được báo hiệu, và liệu có thể xác định được đây có phải là CIIP dựa vào `ciip_flag` hay không. Ngoài ra, có thể xác định được liệu đây là chế độ hợp nhất hình tam giác hay chế độ phân vùng hình tam giác (triangular partitioning mode, TPM) dựa vào `ciip_flag`. FIG.29 (a) thể hiện một phương án trong đó báo hiệu được thực hiện theo trình tự của chế độ hợp nhất chính quy, MMVD, chế độ hợp nhất khối phụ, CIIP và chế độ hợp nhất hình tam giác. Tuy nhiên, sáng chế không bị giới hạn ở đó, và báo hiệu có thể được tạo cấu hình theo trình tự khác nhau. Các hình vẽ trước đó minh họa các phương án trong đó báo hiệu được thực hiện theo trình tự khác nhau

Là phương pháp báo hiệu chế độ hợp nhất khác, phương pháp tạo nhóm có thể được sử dụng. FIG.29(b) thể hiện một phương án về phương pháp tạo nhóm. Ví dụ, `group_1_flag` có thể được báo hiệu, và có thể xác định được liệu chế độ có được chọn dựa vào `group_1_flag` thuộc về nhóm 1 hay không. Nếu xác định không phải là nhóm 1 theo `group_1_flag`, thì `group_2_flag` có thể được báo hiệu. Ngoài ra, có thể xác định được liệu chế độ có được chọn dựa vào `group_2_flag` thuộc về nhóm 2 hay không. Thao tác này có thể được thực hiện ngay cả khi các nhóm tồn tại. Hơn nữa, báo hiệu biểu thị chế độ trong nhóm có thể có mặt. Phương pháp tạo nhóm có thể làm giảm độ sâu báo hiệu so với báo hiệu tuần tự. Hơn nữa, phương pháp tạo nhóm có thể làm giảm chiều dài báo hiệu tối đa (chiều dài tối đa của từ mã).

Theo một phương án của sáng chế, ba nhóm có thể có mặt. Hơn nữa, có thể có một chế độ thuộc về nhóm định trước. Ví dụ, một chế độ thuộc về nhóm 1 có thể có mặt. Ngoài ra, hai chế độ thuộc về mỗi trong số nhóm 2 và nhóm 3 có thể có mặt. Dựa vào FIG.29(b), nhóm 1 có thể bao gồm chế độ hợp nhất khối phụ, nhóm 2 có thể bao gồm chế độ hợp nhất chính quy và MMVD và nhóm 3 có thể bao gồm CIIP và chế độ hợp nhất hình tam giác. Ngoài ra, `group_1_flag` có thể là `merge_subblock_flag`, và `group_2_flag` có

thể là `regular_merge_flag`. Ngoài ra, `ciip_flag` và `mmvd_merge_flag` có thể có mặt dưới dạng báo hiệu biểu thị chế độ trong nhóm. Ví dụ, `merge_subblock_flag` được báo hiệu, và dựa vào `merge_subblock_flag`, liệu có thể xác định được đây có phải là chế độ hợp nhất khối phụ hay không. Nếu xác định được đây không phải là chế độ hợp nhất khối phụ, thì `regular_merge_flag` có thể được báo hiệu. Việc xác định liệu đây là nhóm 2 (chế độ hợp nhất chính quy hoặc MMVD) hay nhóm 3 (CIIP hoặc chế độ hợp nhất hình tam giác) có thể được thực hiện dựa vào `regular_merge_flag`. Ngoài ra, khi xác định được đây là nhóm 2, thì liệu xác định được đây là chế độ hợp nhất chính quy hay MMVD dựa vào `mmvd_merge_flag`. Ngoài ra, khi xác định được đây là nhóm 3, thì việc xác định liệu đây là CIIP hay chế độ hợp nhất hình chữ nhật có thể được thực hiện dựa vào `ciip_flag`. Nghĩa là, `merge_subblock_flag`, `regular_merge_flag`, `mmvd_merge_flag`, và `ciip_flag` trên FIG.29(a) và FIG.29(b) có thể có các ý nghĩa khác nhau một chút.

FIG.30 minh họa một ví dụ về cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Một phương án trên FIG.30 có thể sử dụng phương pháp tạo nhóm được mô tả dựa vào FIG.29(b). Theo phương án này, phần mô tả mà trùng lặp với phần mô tả được mô tả ở trên có thể được lược bỏ.

Theo một phương án của sáng chế, `merge_subblock_flag` có thể được báo hiệu khi chế độ hợp nhất được sử dụng. Trường hợp sử dụng chế độ hợp nhất có thể giống với trường hợp được mô tả ở trên, và có thể tương ứng với trường hợp trong đó `general_merge_flag` là 1. Hơn nữa, sáng chế này có thể tương ứng với trường hợp trong đó `CuPredMode` không phải là `MODE_IBC` hoặc trường hợp trong đó `CuPredMode` là `MODE_INTER`. Ngoài ra, có thể xác định được liệu có phân tích cú pháp `merge_subblock_flag` dựa vào `MaxNumSubblockMergeCand` và kích thước khối hay không, và việc xác định này có thể được thực hiện dựa vào điều kiện nhờ đó chế độ hợp nhất khối phụ sử dụng được như được mô tả ở trên. Nếu `merge_subblock_flag` là 1, thì có thể xác định được là có sử dụng chế độ hợp nhất khối phụ, và chỉ số ứng viên có thể được xác định bổ sung dựa vào `merge_subblock_idx`.

Hơn nữa, nếu `merge_subblock_flag` là 0, thì `regular_merge_flag` có thể được phân tích cú pháp. Ở đây, điều kiện để phân tích cú pháp `regular_merge_flag` có thể có mặt. Ví dụ, điều kiện dựa vào kích thước khối có thể được bao gồm trong đó. Ngoài ra, điều kiện dựa vào báo hiệu mức cao hơn biểu thị liệu chế độ có sử dụng được hay không có thể được bao gồm trong đó. Báo hiệu mức cao hơn biểu thị liệu chế độ có sử dụng được hay không có thể bao gồm `sps_ciip_enabled_flag` và `sps_triangle_enabled_flag`. Báo hiệu mức cao hơn có thể bao gồm điều kiện dựa vào kiểu lát. Hơn nữa, điều kiện dựa vào

cu_skip_flag có thể được bao gồm trong đó. Dựa vào FIG.71, regular_merge_flag có thể được phân tích cú pháp chỉ khi (chiều rộng*chiều cao ≥ 64 && chiều rộng < 128 && chiều cao < 128) được thỏa mãn. Hơn nữa, nếu (chiều rộng*chiều cao ≥ 64 && chiều rộng < 128 && chiều cao < 128) không được thỏa mãn, thì regular_merge_flag có thể không được phân tích cú pháp.

Ngoài ra, các điều kiện mà CIIP sử dụng được có thể bao gồm (sps_ciip_enabled_flag && cu_skip_flag == 0). Hơn nữa, điều kiện kích thước khối mà CIIP sử dụng được có thể bao gồm (chiều rộng*chiều cao ≥ 64 && chiều rộng < 128 && chiều cao < 128). Hơn nữa, các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được có thể bao gồm (sps_triangle_enabled_flag && slice_type == B). Hơn nữa, điều kiện kích thước khối nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được có thể bao gồm (chiều rộng*chiều cao ≥ 64 && chiều rộng < 128 && chiều cao < 128). Nếu các điều kiện mà CIIP sử dụng được hoặc các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được được thỏa mãn, thì regular_merge_flag có thể được phân tích cú pháp. Hơn nữa, khi không có các điều kiện mà CIIP sử dụng được và các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được được thỏa mãn, thì regular_merge_flag có thể không được phân tích cú pháp.

Theo một phương án của sáng chế, khi regular_merge_flag không tồn tại, thì giá trị của chúng có thể được suy ra là 1. Ví dụ, giá trị có thể luôn được suy ra là 1. Điều này có thể tương ứng với trường hợp trong đó chế độ hợp nhất chính quy hoặc MMVD được sử dụng nếu regular_merge_flag là 1 theo sáng chế này. Do đó, nếu không có điều kiện kích thước khối mà CIIP sử dụng được và điều kiện kích thước khối nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được được thỏa mãn, thì chế độ hợp nhất chính quy và MMVD có thể sử dụng được, và giá trị của chúng có thể được xác định mà không phân tích cú pháp regular_merge_flag. Theo một phương án được thể hiện trên FIG.71, điều kiện kích thước khối mà CIIP sử dụng được và điều kiện kích thước khối nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được là giống nhau. Nghĩa là, không có CIIP và chế độ hợp nhất hình chữ nhật có thể được sử dụng đối với khối có chiều rộng hoặc chiều cao bằng 128.

Hơn nữa, ngay cả khi không có các điều kiện mà CIIP sử dụng được và các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được được thỏa mãn, thì chế độ hợp nhất chính quy hoặc MMVD có thể sử dụng được như được mô tả ở trên, và do đó giá trị của chúng có thể được suy ra là 1 mà không phân tích cú pháp regular_merge_flag.

Dựa vào FIG.30, nếu regular_merge_flag là 1, thì phần tử cú pháp có thể được

phân tích cú pháp dựa vào giá trị `sps_mmvd_enabled_flag`. `sps_mmvd_enabled_flag` có thể là báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không như được mô tả ở trên. Khi `sps_mmvd_enabled_flag` là 0, thì MMVD có thể không được sử dụng. Dựa vào FIG.71, nếu `sps_mmvd_enabled_flag` là 0, thì `mmvd_merge_flag`, `mmvd_cand_flag`, `mmvd_distance_idx`, `mmvd_direction_idx` và `merge_idx` có thể không được phân tích cú pháp. Ngoài ra, nếu `mmvd_merge_flag` không tồn tại, thì giá trị của chúng có thể được suy ra là 0. Hơn nữa, nếu `merge_idx` không tồn tại, thì giá trị của chúng có thể được suy ra theo phương pháp được tạo cấu hình trước. Ví dụ, nếu `merge_idx` không tồn tại và `mmvd_merge_flag` là 1, thì có thể được suy ra là `mmvd_cand_flag`, và nếu `mmvd_merge_flag` là 0, thì giá trị của chúng có thể được suy ra là 0. Do đó, theo một phương án trên FIG.30, trong trường hợp trong đó `sps_mmvd_enabled_flag` là 0, nếu `regular_merge_flag` là 1, thì giá trị của `merge_idx` có thể luôn là 0, và phép dự đoán chế độ hợp nhất chính quy có thể được thực hiện bằng cách sử dụng các ứng viên có chỉ số 0 của danh sách ứng viên hợp nhất. Do đó, có thể không có mức độ tự do cho việc chọn lựa ứng viên, và do đó hiệu quả mã hóa có thể giảm. Hơn nữa, nếu `sps_mmvd_enabled_flag` là 1, thì `mmvd_merge_flag` có thể được phân tích cú pháp, và nếu `mmvd_merge_flag` là 0, thì `merge_idx` có thể được phân tích cú pháp dựa vào `MaxNumMergeCand`.

Hơn nữa, dựa vào FIG.30, nếu `regular_merge_flag` là 0, thì `ciip_flag` có thể được phân tích cú pháp khi tất cả các điều kiện mà CIIP sử dụng được và các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được được thỏa mãn. Nếu `ciip_flag` là 1, thì CIIP có thể được sử dụng, và nếu `ciip_flag` là 0, thì chế độ hợp nhất hình chữ nhật có thể được sử dụng. Nếu các điều kiện mà CIIP sử dụng được hoặc các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được được thỏa mãn, thì `CIIP_flag` có thể không được phân tích cú pháp. Nếu `ciip_flag` không tồn tại và `regular_merge_flag` là 1, thì `ciip_flag` có thể được suy ra là 0. Nếu `ciip_flag` không tồn tại và `regular_merge_flag` là 0, thì `ciip_flag` có thể được suy ra là (`sps_ciip_enabled_flag && cu_skip_flag == 0`). Ngoài ra, `MergeTriangleFlag` có thể được thiết lập thành `!ciip_flag` trong trường hợp đối với lát B. Hơn nữa, `MergeTriangleFlag` có thể được thiết lập thành 0 trong trường hợp đối với lát P.

FIG.31 minh họa một ví dụ về cú pháp dữ liệu hợp nhất theo một phương án của sáng chế. Theo phương án này, các chi tiết mà trùng lặp với các chi tiết được mô tả ở trên có thể được lược bỏ. Như được mô tả trên FIG.30, nếu `regular_merge_flag` là 1 và báo hiệu mức cao hơn biểu thị liệu MMVD có sử dụng được hay không biểu thị rằng MMVD là không sử dụng được, thì mức độ tự do để lựa chọn ứng viên được giảm bớt. Theo một

phương án trên FIG.31, vấn đề này có thể được giải quyết.

Dựa vào FIG.31, liệu việc phân tích cú pháp merge_idx có thể độc lập với sps_mmvd_enabled_flag hay không. Nghĩa là, liệu merge_idx có thể được phân tích cú pháp hay không có thể được xác định bất kể giá trị sps_mmvd_enabled_flag. Theo một phương án của sáng chế, nếu regular_merge_flag là 1, mmvd_merge_flag là 0, và MaxNumMergeCand > 1, thì merge_idx có thể được phân tích cú pháp. Hơn nữa, nếu regular_merge_flag là 1 và mmvd_merge_flag là 1, thì merge_idx có thể không được phân tích cú pháp. Hơn nữa, nếu regular_merge_flag là 1 và MaxNumMergeCand là 1, thì merge_idx có thể không được phân tích cú pháp. Ví dụ, nếu sps_mmvd_enabled_flag là 1, regular_merge_flag là 1, mmvd_merge_flag là 0 và MaxNumMergeCand > 1, thì merge_idx có thể được phân tích cú pháp. Tương tự, nếu sps_mmvd_enabled_flag là 0, regular_merge_flag là 1, mmvd_merge_flag là 0 và MaxNumMergeCand > 1, thì merge_idx có thể được phân tích cú pháp.

Ngoài ra, điều kiện kích thước khối nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được theo một phương án trên FIG.30 có thể là (chiều rộng*chiều cao ≥ 64 && chiều rộng < 128 && chiều cao < 128). Tuy nhiên, nếu chiều rộng hoặc chiều cao có giá trị bằng 128, thì chế độ hợp nhất hình chữ nhật sử dụng được. Ví dụ, nếu chiều rộng hoặc chiều cao có giá trị bằng 128, thì phép dự đoán chế độ hợp nhất hình chữ nhật có thể giúp cải thiện hiệu quả mã hóa. Trên FIG.72, chế độ hợp nhất hình chữ nhật có thể được sử dụng ngay cả khi chiều rộng hoặc chiều cao có giá trị bằng 128.

Trên FIG.31, điều kiện kích thước khối mà CIIP sử dụng được có thể là (chiều rộng*chiều cao ≥ 64 && chiều rộng < 128 && chiều cao < 128). Hơn nữa, điều kiện kích thước khối nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được có thể là (chiều rộng*chiều cao ≥ 64). Do đó, dựa vào FIG.72, nếu (chiều rộng*chiều cao ≥ 64) không được thỏa mãn, thì regular_merge_flag có thể không được phân tích cú pháp. Hơn nữa, regular_merge_flag có thể được phân tích cú pháp khi chiều rộng có giá trị bằng 128 (hoặc chiều cao có giá trị lớn hơn hoặc bằng 128) hoặc chiều cao có giá trị bằng 128 (hoặc chiều cao có giá trị lớn hơn hoặc bằng 128). Ví dụ, trong trường hợp đối với (sps_ciip_enabled_flag && cu_skip_flag==0 && chiều rộng < 128 && chiều cao < 128), và nếu (chiều rộng*chiều cao ≥ 64) được thỏa mãn, thì regular_merge_flag có thể được phân tích cú pháp. Hơn nữa, trong trường hợp đối với (sps_triangle_enabled_flag && slice_type == B), nếu (chiều rộng*chiều cao ≥ 64) được thỏa mãn, thì regular_merge_flag có thể được phân tích cú pháp. Hơn nữa, nếu (sps_ciip_enabled_flag && cu_skip_flag==0 && chiều rộng < 128 && chiều cao < 128) không được thỏa mãn

và (`sps_triangle_enabled_flag && slice_type == B`) không được thỏa mãn, thì `regular_merge_flag` có thể không được phân tích cú pháp.

Hơn nữa, dựa vào FIG.31, khi xác định liệu có phân tích cú pháp `ciip_flag` hay không, thì điều kiện dựa vào kích thước khối có thể được yêu cầu. Ví dụ, nếu chiều rộng < 128 và chiều cao < 128, thì bộ giải mã có thể phân tích cú pháp `ciip_flag`. Nếu chiều rộng có giá trị bằng 128 (hoặc lớn hơn 128) hoặc chiều cao có giá trị bằng 128 (hoặc lớn hơn 128), thì `ciip_flag` có thể không được phân tích cú pháp. Điều này có thể xảy ra vì nếu chiều rộng hoặc chiều cao có giá trị bằng 128 (hoặc lớn hơn 128), thì một chế độ trong số chế độ hợp nhất CIIP và hình tam giác có thể không sử dụng được, nhưng chế độ còn lại có thể sử dụng được. Điều này có thể xảy ra vì nếu chiều rộng hoặc chiều cao có giá trị bằng 128 (hoặc lớn hơn 128), thì CIIP không sử dụng được nhưng chế độ hợp nhất hình tam giác có thể sử dụng được. Theo một phương án trên FIG.30, nếu chiều rộng hoặc chiều cao có giá trị bằng 128 (hoặc lớn hơn hoặc bằng 128), thì cả CIIP và chế độ hợp nhất hình tam giác đều không sử dụng được, và theo đó, `regular_merge_flag` không được phân tích cú pháp và giá trị của chúng được suy ra là 1. Do đó, có sự khác biệt giữa các phương án trên FIG.30 và FIG.31.

Theo một phương án của sáng chế, giá trị `regular_merge_flag` có thể được suy ra dựa vào `merge_subblock_flag`. Theo phương án này, `regular_merge_flag`, `merge_subblock_flag` và `ciip_flag` có thể tương ứng với `regular_merge_flag`, `merge_subblock_flag` và `ciip_flag` được mô tả dựa vào FIG.29(b), FIG.30 và FIG.31.

Trong phần mô tả trên FIG.30, nếu `regular_merge_flag` không tồn tại, thì giá trị của chúng luôn được suy ra là 1, nhưng có thể có trường hợp trong đó `merge_subblock_flag` được báo hiệu trước `regular_merge_flag` là 1. Ở đây, khi phép dự đoán được thực hiện dựa vào `merge_subblock_flag` hoặc `regular_merge_flag`, vì cả giá trị `merge_subblock_flag` và `regular_merge_flag` đều là 1, nên tính không rõ ràng có thể xảy ra đối với phép dự đoán cần được thực hiện. Do đó, theo phương án này, `regular_merge_flag` có thể được suy ra dựa vào `merge_subblock_flag`. Ví dụ, nếu `merge_subblock_flag` là 1, thì `regular_merge_flag` có thể được suy ra là 0. Hơn nữa, nếu `merge_subblock_flag` là 0, thì `regular_merge_flag` có thể được suy ra là 1. Theo cách khác, điều kiện của `general_merge_flag` có thể được thêm vào đó. Ví dụ, khi `merge_subblock_flag` là 0 và `general_merge_flag` là 1, thì `regular_merge_flag` có thể được suy ra là 1.

Ngoài ra, trên FIG.30, phương pháp suy luận giá trị `ciip_flag` khi CIIP_flag không

tồn tại được thể hiện. Trong trường hợp trong đó điều kiện kích thước khối mà CIIP sử dụng được và điều kiện kích thước khối nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được khác nhau, nếu phương pháp suy luận `ciip_flag` được mô tả trên FIG.30 được sử dụng, thì báo hiệu được rằng chế độ cụ thể được sử dụng liên quan đến kích thước khối mà chế độ cụ thể không thể được sử dụng. Nghĩa là, ví dụ, nếu chiều rộng hoặc chiều cao có giá trị bằng 128, thì CIIP không sử dụng được, nhưng `ciip_flag` có thể được thiết lập thành 1. Phương án này có thể giải quyết vấn đề này.

Theo một phương án của sáng chế, nếu `ciip_flag` không tồn tại, thì giá trị của chúng có thể được suy ra dựa vào kích thước khối. Hơn nữa, nếu `ciip_flag` không tồn tại, thì giá trị của chúng có thể được suy ra dựa vào `regular_merge_flag`. Ví dụ, nếu `regular_merge_flag` là 1, thì `ciip_flag` có thể được suy ra là 0. Hơn nữa, nếu `regular_merge_flag` là 0, thì `ciip_flag` có thể được suy ra dựa vào kích thước khối. Ví dụ, nếu `regular_merge_flag` là 0, thì `ciip_flag` có thể được suy ra dựa vào kích thước khối, `sps_ciip_enabled_flag` và `cu_skip_flag`. Nếu `regular_merge_flag` là 0, thì `ciip_flag` có thể được suy ra là $(\text{sps_ciip_enabled_flag} \ \&\& \ \text{cu_skip_flag} == 0 \ \&\& \ \text{chiều rộng} < 128 \ \&\& \ \text{chiều cao} < 128)$. Do đó, nếu `regular_merge_flag` là 0 và chiều rộng hoặc chiều cao có giá trị bằng 128, thì `ciip_flag` có thể được suy ra là 0. Hơn nữa, để suy ra giá trị `ciip_flag` là 1, thì có thể có điều kiện mà `general_merge_flag` là 1. Nếu `general_merge_flag` là 0, thì giá trị của `ciip_flag` có thể được suy ra là 0. Nghĩa là, nếu `regular_merge_flag` là 0, thì `ciip_flag` có thể được suy ra là $(\text{sps_ciip_enabled_flag} \ \&\& \ \text{cu_skip_flag} == 0 \ \&\& \ \text{chiều rộng} < 128 \ \&\& \ \text{chiều cao} < 128 \ \&\& \ \text{general_merge_flag} == 1)$. Ngược lại, giá trị của `ciip_flag` có thể được suy ra là 0.

Theo cách khác, nếu `regular_merge_flag` là 0 và `general_merge_flag` là 1, thì `ciip_flag` có thể được suy ra là $(\text{sps_ciip_enabled_flag} \ \&\& \ \text{cu_skip_flag} == 0 \ \&\& \ \text{chiều rộng} < 128 \ \&\& \ \text{chiều cao} < 128)$. Nếu `general_merge_flag` là 0, thì giá trị của `ciip_flag` có thể được suy ra là 0.

Ngoài ra, phương pháp thiết lập `MergeTriangleFlag` bất kể giá trị của `regular_merge_flag` đã được mô tả theo một phương án trên FIG.30. Do đó, trường hợp trong đó `regular_merge_flag` là 1 và `MergeTriangleFlag` là 1 có thể xảy ra, và điều này gây ra tính không rõ ràng đối với phương pháp dự đoán. Do đó, theo phương án này, `MergeTriangleFlag` có thể được thiết lập dựa vào `regular_merge_flag`. Ví dụ, nếu `regular_merge_flag` là 1, thì `MergeTriangleFlag` có thể được thiết lập thành 0. Hơn nữa, nếu `regular_merge_flag` là 0, thì `MergeTriangleFlag` có thể được thiết lập thành `!ciip_flag`. Ngoài ra, nếu `regular_merge_flag` là 0, thì `MergeTriangleFlag` có thể được thiết lập khi

xem xét các điều kiện nhờ đó chế độ hợp nhất hình chữ nhật sử dụng được. Ví dụ, nếu `regular_merge_flag` là 0, thì `MergeTriangleFlag` có thể được thiết lập thành `(!ciip_flag && sps_triangle_enabled_flag && slice_type == B)`.

Do đó, khi `sps_triangle_enabled_flag` là 0 hoặc `slice_type` không phải là B, thì tình huống trong đó `MergeTriangleFlag` được thiết lập thành 1 có thể được ngăn chặn. Ngoài ra, điều kiện mà `general_merge_flag` là 1 có thể được bao gồm để xác định `MergeTriangleFlag` là 1. Nếu `general_merge_flag` là 0, thì `MergeTriangleFlag` có thể được thiết lập thành 0. Nghĩa là, nếu `regular_merge_flag` là 0, thì `MergeTriangleFlag` có thể được thiết lập thành `(!ciip_flag && sps_triangle_enabled_flag && slice_type == B && general_merge_flag == 1)`. Ngược lại, `MergeTriangleFlag` có thể được thiết lập thành 0.

Theo cách khác, nếu `regular_merge_flag` là 0 và `general_merge_flag` là 1, thì `MergeTriangleFlag` có thể được thiết lập thành `(!ciip_flag && sps_triangle_enabled_flag && slice_type == B)`. Nếu `general_merge_flag` là 0, thì `MergeTriangleFlag` có thể được thiết lập thành 0.

FIG.32 minh họa chế độ hợp nhất hình học theo một phương án của sáng chế. Theo một phương án của sáng chế, chế độ hợp nhất hình học có thể được gọi là các tên gọi khác nhau, chẳng hạn như chế độ phân vùng hình học, chế độ GEO, chế độ hợp nhất GEO và phân vùng GEO. Theo một phương án của sáng chế, chế độ hợp nhất hình học có thể là phương pháp chia đơn vị mã hóa (coding unit, CU) và khối mã hóa (coding block, CB). Ví dụ, chế độ hợp nhất hình học có thể là phương pháp chia CU hoặc CB có dạng hình vuông hoặc hình chữ nhật bằng cách sử dụng các phân vùng không vuông hoặc không phải hình chữ nhật. Dựa vào FIG.32, một ví dụ về phân vùng hình học được thể hiện. Như được thể hiện trên FIG.32, CU hình chữ nhật có thể được chia thành các phân vùng hình tam giác hoặc hình thang (hoặc các đa giác) bằng cách phân vùng hình học. Hơn nữa, báo hiệu dùng cho phương pháp thực hiện chế độ hợp nhất hình học có thể được báo hiệu cho CU. Hơn nữa, liên quan đến chế độ hợp nhất hình học, phép nén chuyển động và phép dự đoán có thể được thực hiện dựa vào hai thông tin chuyển động. Hơn nữa, hai thông tin chuyển động này có thể thu được từ ứng viên hợp nhất. Theo một phương án của sáng chế, báo hiệu biểu thị hai thông tin chuyển động được sử dụng ở chế độ hợp nhất hình học có thể có mặt. Ví dụ, hai chỉ số có thể được báo hiệu để biểu thị hai thông tin chuyển động được sử dụng ở chế độ hợp nhất hình học. Cụ thể hơn là, ví dụ, hai chỉ số ứng viên hợp nhất có thể được báo hiệu để biểu thị hai thông tin chuyển động được sử dụng ở chế độ hợp nhất hình học. Hơn nữa, hai phép dự đoán có thể được tổng hợp ở chế độ hợp nhất hình học. Ví dụ, ở chế độ hợp nhất hình học, hai phép dự đoán có thể được

tổng hợp gần biên bên trong trong CU. Việc tổng hợp hai phép dự đoán có thể biểu thị rằng tổng hai phép dự đoán được lấy trọng số được thực hiện.

Theo một phương án, các phần tử cú pháp để biểu thị hai thông tin chuyển động được sử dụng ở chế độ hợp nhất hình học có thể là `merge_triangle_idx0` và `merge_triangle_idx1`. Ở đây, hai chỉ số `m` và `n` có thể được suy ra từ phần tử cú pháp. Ví dụ, phương trình sau có thể được suy ra.

$$m = \text{merge_triangle_idx0}$$

$$n = \text{merge_triangle_idx1} + ((\text{merge_triangle_idx1} \geq m) ? 1 : 0)$$

Nghĩa là, chỉ số `m` có thể bằng `merge_triangle_idx0`. Hơn nữa, chỉ số `n` có thể là `merge_triangle_idx1 + 1` nếu `merge_triangle_idx1` bằng hoặc lớn hơn `merge_triangle_idx0`, và chỉ số `n` có thể là `merge_triangle_idx1` nếu `merge_triangle_idx1` nhỏ hơn `merge_triangle_idx0`.

Hơn nữa, dựa vào FIG.32, biên phân chia của chế độ hợp nhất hình học có thể được thể hiện theo góc phi và khoảng cách độ dịch rho. Góc phi có thể thể hiện góc được lượng tử hóa, và khoảng cách độ dịch rho có thể thể hiện độ dịch được lượng tử hóa. Góc và khoảng cách độ dịch có thể được báo hiệu bởi `merge_geo_idx`. Ví dụ, góc và khoảng cách độ dịch có thể được xác định dựa vào bảng tra cứu. Chế độ hợp nhất hình học có thể tạo ra tín hiệu dự đoán từ hai phép dự đoán dựa vào hai thông tin chuyển động. Ở đây, khi thực hiện tính tổng hai phép dự đoán được lấy trọng số, trọng số có thể dựa vào góc và độ dịch khoảng cách. Theo cách khác, khi thực hiện tính tổng hai phép dự đoán được lấy trọng số, thì trọng số có thể dựa vào vị trí (tọa độ) nằm trong khối. Theo cách khác, khi thực hiện tính tổng hai phép dự đoán được lấy trọng số, thì trọng số có thể dựa vào chiều rộng và chiều cao khối.

Chế độ hợp nhất hình học có thể có nhiều kiểu phân chia khả thi hơn so với chế độ của TPM. Ví dụ, có thể có hai hoặc nhiều hơn hai kiểu phân chia ở chế độ hợp nhất hình học. Ví dụ, có thể có 80 kiểu phân chia. Chế độ hợp nhất hình học có thể là kiểu chế độ hợp nhất. Nghĩa là, `general_merge_flag` có thể có giá trị bằng 1 ở chế độ hợp nhất hình học.

FIG.33 minh họa cú pháp dữ liệu hợp nhất theo một phương án của sáng chế.

Liên quan đến một phương án trên FIG.33, các chi tiết được mô tả dựa vào các hình vẽ từ FIG.29 đến FIG.32 hoặc các chi tiết được mô tả trước đó có thể được lược bỏ.

Như được mô tả ở trên, phương pháp báo hiệu nhiều chế độ hợp nhất có thể có

mặt. Hơn nữa, nhiều chế độ hợp nhất có thể bao gồm chế độ hợp nhất khối phụ, chế độ hợp nhất chính quy, MMVD, CIIP, chế độ hợp nhất hình học và dạng tương tự. Hơn nữa, nhiều chế độ hợp nhất có thể không bao gồm chế độ phân vùng hình tam giác. Theo cách khác, chế độ phân vùng hình tam giác có thể ở dạng có trong chế độ hợp nhất hình học. Ngoài ra, khi các chế độ hợp nhất báo hiệu bằng cách sử dụng phương pháp báo hiệu theo phương án này, thì các từ mã có các chiều dài khác nhau có thể được sử dụng và từ mã có chiều dài ngắn có thể được sử dụng đối với chế độ cụ thể, và do đó hiệu quả mã hóa có thể được cải thiện. Hơn nữa, phương pháp báo hiệu theo phương án này có thể loại bỏ báo hiệu dư và cải thiện hiệu quả mã hóa. Ngoài ra, độ phức tạp phân tích cú pháp có thể được giảm xuống bằng cách lược bỏ việc kiểm tra điều kiện không cần thiết khi báo hiệu theo phương án này.

Theo một phương án của sáng chế, có thể có các điều kiện mà CIIP sử dụng được. Các điều kiện mà CIIP sử dụng được có thể được gọi là CIIP_conditions. CIIP_conditions này có thể đúng khi tất cả các điều kiện sau được thỏa mãn.

Điều kiện 1. `sps_ciip_enabled_flag`

Điều kiện 2. `cu_skip_flag == 0`

Điều kiện 3. `cbWidth * cbHeight >= 64`

Điều kiện 4. `cbWidth < 128`

Điều kiện 5. `cbHeight < 128`

Hơn nữa, CIIP_conditions có thể sai khi ít nhất một điều kiện trong số các điều kiện nêu trên không được thỏa mãn. Vì các điều kiện nêu trên đã được mô tả theo phương án nêu trên, nên các chi tiết sẽ được lược bỏ ở đây.

Theo một phương án của sáng chế, các điều kiện nhờ đó chế độ hợp nhất hình học sử dụng được có thể có mặt. Các điều kiện nhờ đó chế độ hợp nhất hình học sử dụng được có thể được gọi là GEO_conditions. GEO_conditions có thể đúng khi tất cả các điều kiện sau được thỏa mãn.

Điều kiện 1. `sps_triangle_enabled_flag`

Điều kiện 2. `MaxNumTriangleMergeCand > 1`

Điều kiện 3. `slice_type == B`

Điều kiện 4. `cbWidth >= 8`

Điều kiện 5. `cbHeight >= 8`

Hơn nữa, GEO_conditions có thể sai khi ít nhất một điều kiện trong số các điều kiện nêu trên không được thỏa mãn.

Theo một phương án khác, điều kiện slice_type có thể không được yêu cầu. Điều này có thể thực hiện được vì điều kiện dựa vào slice_type được thỏa mãn khi điều kiện còn lại, ví dụ, điều kiện dựa vào MaxNumTriangleMergeCand, được thỏa mãn. Theo một phương án của sáng chế, các điều kiện nhờ đó chế độ hợp nhất hình học sử dụng được có thể có mặt. Các điều kiện nhờ đó chế độ hợp nhất hình học sử dụng được có thể được gọi là GEO_conditions. GEO_conditions có thể đúng khi tất cả các điều kiện sau được thỏa mãn.

Điều kiện 1. sps_triangle_enabled_flag

Điều kiện 2. MaxNumTriangleMergeCand > 1

Điều kiện 3. cbWidth >= 8

Điều kiện 4. cbHeight >= 8

Hơn nữa, GEO_conditions có thể sai khi ít nhất một điều kiện trong số các điều kiện nêu trên không được thỏa mãn.

Vì các điều kiện nêu trên đã được mô tả theo phương án nêu trên, nên các chi tiết sẽ được lược bỏ ở đây. Tuy nhiên, liên quan đến sps_triangle_enabled_flag và MaxNumTriangleMergeCand, đã được mô tả trước đó liên quan đến giá trị liên quan đến TPM, nhưng theo phương án này, giá trị liên quan đến chế độ hợp nhất hình học có thể được xem xét. Nghĩa là, sps_triangle_enabled_flag có thể là báo hiệu mức cao hơn biểu thị việc liệu chế độ hợp nhất hình học có sử dụng được hay không. Hơn nữa, MaxNumTriangleMergeCand có thể là số lượng danh sách ứng viên tối đa được sử dụng ở chế độ hợp nhất hình học.

Theo một phương án của sáng chế, regular_merge_flag có thể được phân tích cú pháp khi CIIP_conditions hoặc GEO_conditions được thỏa mãn. Hơn nữa, nếu CIIP_conditions hoặc GEO_conditions được thỏa mãn, thì regular_merge_flag có thể không được phân tích cú pháp. Dựa vào FIG.75, điều kiện 2 biểu thị (CIIP_conditions || GEO_conditions). Nghĩa là, regular_merge_flag có thể được phân tích cú pháp nếu ít nhất một điều kiện trong số các điều kiện sau được thỏa mãn.

Điều kiện 1 (CIIP_conditions). sps_ciip_enabled_flag && cu_skip_flag == 0 && cbWidth * cbHeight >= 64 && cbWidth < 128 && cbHeight < 128

Điều kiện 2 (GEO_conditions). sps_triangle_enabled_flag &&

$\text{MaxNumTriangleMergeCand} > 1 \ \&\& \text{cbWidth} \geq 8 \ \&\& \text{cbHeight} \geq 8$

Hơn nữa, nếu không có điều kiện nào trong số các điều kiện nêu trên được thỏa mãn, thì `regular_merge_flag` có thể không được phân tích cú pháp. Ngoài ra, khi `regular_merge_flag` không tồn tại, thì giá trị của chúng có thể được suy ra là `general_merge_flag && !merge_subblock_flag`.

Theo một phương án khác, như được mô tả ở trên, điều kiện 2 (`GEO_conditions`) có thể bao gồm điều kiện `slice_type` và được thể hiện như sau.

Điều kiện 2 (`GEO_conditions`). $\text{sps_triangle_enabled_flag} \ \&\& \text{MaxNumTriangleMergeCand} > 1 \ \&\& \text{slice_type} == B \ \&\& \text{cbWidth} \geq 8 \ \&\& \text{cbHeight} \geq 8$

Tuy nhiên, nếu điều kiện `slice_type` luôn luôn được thỏa mãn khi các điều kiện khác được thỏa mãn, thì việc kiểm tra bổ sung điều kiện kiểu lát có thể không được yêu cầu để làm giảm độ phức tạp của việc kiểm tra điều kiện phân tích cú pháp.

Theo một phương án của sáng chế, `ciip_flag` có thể được phân tích cú pháp nếu cả `CIIP_conditions` và `GEO_conditions` đều được thỏa mãn. Hơn nữa, nếu `CIIP_conditions` hoặc `GEO_conditions` không được thỏa mãn, thì `ciip_flag` có thể không được phân tích cú pháp. Nghĩa là, nếu tất cả các điều kiện sau được thỏa mãn, thì `ciip_flag` có thể được phân tích cú pháp, và nếu ít nhất một điều kiện trong số các điều kiện sau không được thỏa mãn, thì `ciip_flag` có thể không được phân tích cú pháp.

Điều kiện 1 (`CIIP_conditions`). $\text{sps_ciip_enabled_flag} \ \&\& \text{cu_skip_flag} == 0 \ \&\& \text{cbWidth} * \text{cbHeight} \geq 64 \ \&\& \text{cbWidth} < 128 \ \&\& \text{cbHeight} < 128$

Điều kiện 2 (`GEO_conditions`). $\text{sps_triangle_enabled_flag} \ \&\& \text{MaxNumTriangleMergeCand} > 1 \ \&\& \text{cbWidth} \geq 8 \ \&\& \text{cbHeight} \geq 8$

Hơn nữa, như được mô tả ở trên, có thể bao gồm các điều kiện dựa vào `slice_type` ở điều kiện 2 (`GEO_conditions`). Điều kiện có thể như sau.

Điều kiện 2 (`GEO_conditions`). $\text{sps_triangle_enabled_flag} \ \&\& \text{MaxNumTriangleMergeCand} > 1 \ \&\& \text{slice_type} == B \ \&\& \text{cbWidth} \geq 8 \ \&\& \text{cbHeight} \geq 8$

Theo một phương án của sáng chế, các điều kiện để phân tích cú pháp `ciip_flag` có thể khác nhau để giảm độ phức tạp khi phân tích cú pháp. Ví dụ, một số trong số các điều kiện kích thước khối có thể được lược bỏ. Theo sáng chế này, khi các điều kiện kích thước khối để sử dụng chế độ hợp nhất hình học được thỏa mãn, thì một số trong số các

điều kiện kích thước khối để sử dụng CIIP có thể được thỏa mãn. Do đó, dựa vào điều kiện 3 trên FIG.75 và theo một phương án của sáng chế, ciip_flag có thể được phân tích cú pháp nếu tất cả các điều kiện sau được thỏa mãn, và ciip_flag có thể không được phân tích cú pháp nếu ít nhất một điều kiện trong số các điều kiện sau không được thỏa mãn.

Điều kiện 1 (CIIP_conditions). $\text{sps_ciip_enabled_flag} \&\& \text{cu_skip_flag} == 0 \&\& \text{cbWidth} < 128 \&\& \text{cbHeight} < 128$

Điều kiện 2 (GEO_conditions). $\text{sps_triangle_enabled_flag} \&\& \text{MaxNumTriangleMergeCand} > 1 \&\& \text{cbWidth} \geq 8 \&\& \text{cbHeight} \geq 8$

Hơn nữa, nếu ciip_flag không tồn tại, thì giá trị của chúng có thể được suy ra là 1 khi tất cả các điều kiện sau được thỏa mãn, và giá trị của chúng có thể được suy ra là 0 khi ít nhất một điều kiện trong số các điều kiện sau không được thỏa mãn.

Điều kiện 1. $\text{sps_ciip_enabled_flag} == 1$

Điều kiện 2. $\text{general_merge_flag} == 1$

Điều kiện 3. $\text{merge_subblock_flag} == 0$

Điều kiện 4. $\text{regular_merge_flag} == 0$

Điều kiện 5. $\text{cu_skip_flag} == 0$

Điều kiện 6. $\text{cbWidth} < 128$

Điều kiện 7. $\text{cbHeight} < 128$

Điều kiện 8. $\text{cbWidth} * \text{cbHeight} \geq 64$

Nghĩa là, các điều kiện trong quá trình phân tích cú pháp và các điều kiện trong quá trình suy luận có thể khác nhau. Ngoài ra, các điều kiện được lược bỏ trong quá trình phân tích cú pháp (ví dụ, các điều kiện dựa vào kích thước khối) có thể có trong điều kiện suy luận.

Ngoài ra, merge_geo_flag, là giá trị biểu thị liệu có sử dụng chế độ hợp nhất hình học hay không, có thể được xác định là 1 nếu tất cả các điều kiện sau được thỏa mãn, và có thể được xác định là 0 nếu ít nhất một điều kiện trong số các điều kiện sau không được thỏa mãn.

Điều kiện 1. $\text{sps_triangle_enabled_flag} == 1$

Điều kiện 2. $\text{general_merge_flag} == 1$

Điều kiện 3. $\text{merge_subblock_flag} == 0$

Điều kiện 4. `regular_merge_flag == 0`

Điều kiện 5. `ciip_flag == 0`

Điều kiện 6. `MaxNumTriangleMergeCand >= 2`

Điều kiện 7. `cbWidth >= 8`

Điều kiện 8. `cbHeight >= 8`

Theo một phương án bổ sung, `slice_type == B` của điều kiện 9 có thể được thêm vào.

Do đó, dựa vào FIG.33, cấu trúc báo hiệu sau có thể được đề xuất. Khi điều kiện thứ nhất 3301 được thỏa mãn, thì `merge_subblock_flag` có thể được phân tích cú pháp. Nếu `merge_subblock_flag` là 1, thì chế độ hợp nhất khối phụ có thể được sử dụng, `merge_subblock_idx` có thể được phân tích cú pháp và `regular_merge_flag`, `mmvd_merge_flag`, và `ciip_flag` có thể không được phân tích cú pháp. Nếu `merge_subblock_flag` là 0 và điều kiện thứ hai 3302 được thỏa mãn, thì `regular_merge_flag` có thể được phân tích cú pháp. Nếu `regular_merge_flag` là 1, thì chế độ hợp nhất chính quy hoặc MMVD có thể được sử dụng, và `mmvd_merge_flag` bổ sung có thể được phân tích cú pháp. Về vấn đề này, các chi tiết được mô tả dựa vào các hình vẽ từ FIG.29 đến FIG.32 có thể được áp dụng. Nếu `regular_merge_flag` là 0 và điều kiện thứ hai 3302 được thỏa mãn, thì `ciip_flag` có thể được phân tích cú pháp. Nếu `ciip_flag` là 1, thì CIIP có thể được sử dụng. Khi sử dụng CIIP, thì `merge_idx` có thể được phân tích cú pháp nếu `MaxNumMergeCand` lớn hơn 1. Nếu `ciip_flag` là 0, thì `merge_geo_flag` có thể được xác định là 1. Hơn nữa, nếu `ciip_flag` là 0, thì chế độ hợp nhất hình học có thể được sử dụng. Khi sử dụng chế độ hợp nhất hình học, thì `merge_geo_idx`, `merge_triangle_idx0` và `merge_triangle_idx1` có thể được phân tích cú pháp. Theo cách khác, khi sử dụng chế độ hợp nhất hình học, nếu `MaxNumTriangleMergeCand` lớn hơn 1, thì `merge_geo_idx`, `merge_triangle_idx0` và `merge_triangle_idx1` có thể được phân tích cú pháp.

Do đó, theo một phương án của sáng chế, khi khối có chiều rộng hoặc chiều cao bằng 4, nghĩa là, khối có kích thước $4 \times N$ hoặc $N \times 4$ sử dụng CIIP, thì báo hiệu có thể được thực hiện như sau. `Merge_subblock_flag` có thể là 0, vì điều kiện thứ hai 3302 được thỏa mãn, nên `regular_merge_flag` có thể được phân tích cú pháp và giá trị của chúng có thể là 0, và vì điều kiện thứ ba 3303 không được thỏa mãn, nên `ciip_flag` có thể không được phân tích cú pháp và giá trị của chúng có thể được suy ra là 1 theo các chi tiết được mô tả ở trên. Ngoài ra, khi sử dụng chế độ hợp nhất hình học, thì báo hiệu có thể được thực hiện

như sau. Merge_subblock_flag có thể là 0, regular_merge_flag có thể là 0 và ciip_flag có thể là 0.

Theo một phương án được mô tả dựa vào FIG.33, chỉ ciip_flag có thể có mặt dưới dạng phần tử cú pháp biểu thị liệu có sử dụng chế độ hợp nhất hình học hay không. Ngoài ra, một phương án trên FIG.33 đã được mô tả dựa vào chế độ hợp nhất hình học không thể thực hiện đối với khối có chiều rộng hoặc chiều cao nhỏ hơn 8. Tuy nhiên, sáng chế không bị giới hạn ở đó, và một phương án có thể được áp dụng ngay cả khi chế độ hợp nhất hình học không sử dụng được cho kích thước khối khác (ví dụ, kích thước khối nhỏ hơn ngưỡng).

FIG.34 minh họa một ví dụ về phương pháp xử lý tín hiệu video theo một phương án mà sáng chế được áp dụng. Dựa vào FIG.34, bộ giải mã chủ yếu được mô tả để thuận tiện cho việc mô tả, nhưng sáng chế không bị giới hạn ở đó. Hơn nữa, phương pháp để xử lý tín hiệu video dựa vào phép dự đoán đa giả thuyết theo sáng chế này có thể được áp dụng cho bộ mã hóa gần như theo cùng cách.

Bộ giải mã phân tích cú pháp phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp nhất có được áp dụng cho khối hiện thời hay không (S3401).

Khi chế độ hợp nhất được áp dụng cho khối hiện thời, thì bộ giải mã xác định liệu có phân tích cú pháp phần tử cú pháp thứ hai dựa vào điều kiện định trước thứ nhất (S3402) hay không. Theo một phương án, phần tử cú pháp thứ hai có thể biểu thị liệu chế độ thứ nhất hay chế độ thứ hai được áp dụng cho khối hiện thời.

Khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, thì bộ giải mã xác định liệu có phân tích cú pháp phần tử cú pháp thứ ba dựa vào điều kiện định trước thứ hai (S3403) hay không. Theo một phương án, phần tử cú pháp thứ ba có thể biểu thị chế độ được áp dụng cho khối hiện thời trong số chế độ thứ ba hoặc chế độ thứ tư.

Bộ giải mã xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba (S3404).

Bộ giải mã thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định (S3405).

Bộ giải mã tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời (S3406).

Sáng chế đề cập đến phương pháp xử lý tín hiệu video trong đó điều kiện thứ nhất

bao gồm ít nhất một điều kiện trong số điều kiện nhờ đó chế độ thứ ba sử dụng được và điều kiện nhờ đó chế độ thứ tư sử dụng được.

Như được mô tả ở trên, theo một phương án, chế độ thứ ba và chế độ thứ tư có thể được đặt muện hơn so với chế độ thứ nhất trong trình tự giải mã theo cú pháp dữ liệu hợp nhất.

Như được mô tả ở trên, theo một phương án, phương pháp xử lý tín hiệu video theo sáng chế này bao gồm bước phân tích cú pháp phần tử cú pháp thứ hai khi điều kiện thứ nhất được thỏa mãn; và khi điều kiện thứ nhất không được thỏa mãn, thì giá trị của phần tử cú pháp thứ hai có thể được suy ra là 1.

Như được mô tả ở trên, theo một phương án, khi điều kiện thứ nhất không được thỏa mãn, thì phần tử cú pháp thứ hai có thể được suy ra dựa vào phần tử cú pháp thứ tư biểu thị liệu chế độ hợp nhất dựa trên khối phụ có được áp dụng cho khối hiện thời hay không.

Như được mô tả ở trên, theo một phương án, điều kiện thứ hai có thể bao gồm điều kiện nhờ đó chế độ thứ tư sử dụng được.

Như được mô tả ở trên, theo một phương án, điều kiện thứ hai có thể bao gồm ít nhất một điều kiện trong số các điều kiện liên quan đến việc liệu chế độ thứ ba sử dụng được theo trình tự hiện thời hay không, liệu chế độ thứ tư sử dụng được theo trình tự hiện thời hay không, liệu số lượng ứng viên tối đa dùng cho chế độ thứ tư có lớn hơn 1 hay không, liệu chiều rộng của khối hiện thời có nhỏ hơn kích thước định trước thứ nhất hay không và liệu chiều cao của khối hiện thời có nhỏ hơn kích thước định trước thứ hai hay không.

Như được mô tả ở trên, theo một phương án, phương pháp xử lý tín hiệu video theo sáng chế này bao gồm bước: khi phần tử cú pháp thứ hai có giá trị của 1, thì thu được phần tử cú pháp thứ năm biểu thị liệu chế độ được áp dụng cho khối hiện thời là chế độ thứ nhất hay chế độ thứ hai.

Các phương án của sáng chế có thể được thực hiện thông qua các phương tiện khác nhau. Ví dụ, các phương án của sáng chế có thể được thực hiện bởi phần cứng, phần sụn, phần mềm hoặc sự kết hợp của các phần này.

Theo phương án thực hiện bởi phần cứng, các phương án của sáng chế có thể được thực hiện bởi một hoặc nhiều mạch tích hợp cho ứng dụng cụ thể (Application Specific Integrated Circuit, ASIC), bộ xử lý tín hiệu số (Digital Signal Processor, DSP), thiết bị xử

lý tín hiệu số (Digital Signal Processing Device, DSPD), thiết bị logic lập trình được (Programmable Logic Device, PLD), mảng cổng lập trình được dạng trường (Field Programmable Gate Array, FPGA), bộ xử lý, bộ điều khiển, bộ vi điều khiển, bộ vi xử lý và dạng tương tự.

Theo phương án thực hiện bởi phần sụn hoặc phần mềm, phương pháp theo các phương án của sáng chế có thể được thực hiện dưới dạng mô đun, thủ tục, hàm số hoặc dạng tương tự để thực hiện các chức năng hoặc hoạt động được mô tả ở trên. Mã phần mềm có thể được lưu trữ trong bộ nhớ và được thực thi bởi bộ xử lý. Bộ nhớ có thể được bố trí bên trong hoặc bên ngoài bộ xử lý, và có thể trao đổi dữ liệu với bộ xử lý thông qua các phương tiện khác nhau đã biết đến.

Một số phương án có thể được thực hiện dưới dạng vật ghi gồm các lệnh thực thi được bởi máy tính chẳng hạn như mô đun chương trình được thực thi bởi máy tính. Vật ghi đọc được bởi máy tính có thể là vật ghi khả dụng ngẫu nhiên mà có thể truy cập được bởi máy tính, và có thể bao gồm tất cả các vật ghi khả biến, không khả biến, tháo được và không tháo được. Ngoài ra, vật ghi đọc được bởi máy tính này có thể bao gồm tất cả các vật ghi lưu trữ bởi máy tính và vật ghi truyền thông. Vật ghi lưu trữ bởi máy tính bao gồm tất cả các vật ghi khả biến, không khả biến, tháo được và không tháo được được thực hiện theo phương pháp hoặc công nghệ ngẫu nhiên để lưu trữ thông tin chẳng hạn như các lệnh đọc được bởi máy tính, cấu trúc dữ liệu, mô đun chương trình hoặc dữ liệu khác. Các vật ghi truyền thông bao gồm các lệnh đọc được bởi máy tính, cấu trúc dữ liệu, dữ liệu khác của các tín hiệu dữ liệu được điều biến chẳng hạn như các mô đun chương trình, hoặc các cơ chế truyền dẫn khác, và bao gồm vật ghi truyền thông tin ngẫu nhiên.

Phần mô tả theo sáng chế này chỉ là một ví dụ, và người có hiểu biết trung bình trong lĩnh vực kỹ thuật có thể hiểu rằng các phương án của sáng chế có thể được thay đổi dễ dàng thành các dạng chi tiết khác mà không nằm ngoài ý tưởng kỹ thuật hoặc các dấu hiệu cần thiết của sáng chế. Do đó, các phương án được mô tả ở trên cần được hiểu là các ví dụ trong tất cả các khía cạnh và không nhằm giới hạn. Ví dụ, mỗi phần tử được đề cập dưới dạng số ít có thể được thực hiện theo cách được phân bố, và các phần tử được đề cập dưới dạng được phân tán có thể được thực hiện dưới dạng kết hợp.

Phạm vi của sáng chế được xác định bởi các điểm yêu cầu bảo hộ kèm thay vì được xác định bởi phần mô tả chi tiết. Ý nghĩa và phạm vi của các điểm yêu cầu bảo hộ và tất cả các cải biến hoặc dạng được cải biến có nguồn gốc từ khái niệm của các dấu hiệu tương đương với nó cần được hiểu là nằm trong phạm vi của sáng chế.

Khả năng áp dụng trong công nghiệp

Các phương án được ưu tiên theo sáng chế được mô tả ở trên được đề xuất nhằm mục đích minh họa, và người có hiểu biết trung bình trong lĩnh vực kỹ thuật có thể cải thiện, cải biến, thay thế hoặc thêm các phương án khác nhau nằm trong ý tưởng kỹ thuật và phạm vi kỹ thuật của sáng chế được bộc lộ trong các điểm yêu cầu bảo hộ kèm theo.

YÊU CẦU BẢO HỘ

1. Phương pháp xử lý tín hiệu video, phương pháp này bao gồm các bước:

phân tích cú pháp phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp nhất có được áp dụng cho khối hiện thời hay không;

xác định liệu có phân tích cú pháp phần tử cú pháp thứ hai hay không dựa vào điều kiện định trước thứ nhất khi chế độ hợp nhất được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ hai này biểu thị liệu chế độ thứ nhất hay chế độ thứ hai được áp dụng cho khối hiện thời;

xác định liệu có phân tích cú pháp phần tử cú pháp thứ ba hay không dựa vào điều kiện định trước thứ hai khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ ba này biểu thị chế độ trong số chế độ thứ ba và chế độ thứ tư được áp dụng cho khối hiện thời;

xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba;

thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định; và
tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời,

trong đó điều kiện định trước thứ nhất bao gồm ít nhất một điều kiện trong số điều kiện nhờ đó chế độ thứ ba sử dụng được và điều kiện nhờ đó chế độ thứ tư sử dụng được.

2. Phương pháp xử lý tín hiệu video theo điểm 1, trong đó chế độ thứ ba và chế độ thứ tư được đặt muộn hơn so với chế độ thứ nhất trong trình tự giải mã theo cú pháp dữ liệu hợp nhất.

3. Phương pháp xử lý tín hiệu video theo điểm 1, phương pháp này còn bao gồm bước phân tích cú pháp phần tử cú pháp thứ hai khi điều kiện định trước thứ nhất được thỏa mãn,

trong đó khi điều kiện định trước thứ nhất không được thỏa mãn, thì giá trị của phần tử cú pháp thứ hai được suy ra là 1.

4. Phương pháp xử lý tín hiệu video theo điểm 1, trong đó khi điều kiện định trước thứ nhất không được thỏa mãn, thì phần tử cú pháp thứ hai được suy ra dựa vào phần tử cú pháp thứ tư biểu thị liệu chế độ hợp nhất dựa trên khối phụ có được áp dụng cho khối hiện thời hay không.

5. Phương pháp xử lý tín hiệu video theo điểm 1, trong đó điều kiện định trước thứ hai bao gồm điều kiện nhờ đó chế độ thứ tư sử dụng được.

6. Phương pháp xử lý tín hiệu video theo điểm 1, trong đó điều kiện định trước thứ hai bao gồm ít nhất một điều kiện trong số các điều kiện liên quan đến việc liệu chế độ thứ ba sử dụng được theo trình tự hiện thời hay không, liệu chế độ thứ tư sử dụng được theo trình tự hiện thời hay không, liệu số lượng ứng viên tối đa dùng cho chế độ thứ tư có lớn hơn 1 hay không, liệu chiều rộng của khối hiện thời có nhỏ hơn kích thước định trước thứ nhất hay không, và liệu chiều cao của khối hiện thời có nhỏ hơn kích thước định trước thứ hai hay không.

7. Phương pháp xử lý tín hiệu video theo điểm 1, phương pháp này còn bao gồm bước: khi phần tử cú pháp thứ hai có giá trị bằng 1, thì thu được phần tử cú pháp thứ năm biểu thị liệu chế độ được áp dụng cho khối hiện thời là chế độ thứ nhất hay chế độ thứ hai.

8. Thiết bị xử lý tín hiệu video bao gồm bộ xử lý, trong đó bộ xử lý được tạo cấu hình để:

phân tích cú pháp phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp nhất có được áp dụng cho khối hiện thời hay không;

xác định liệu có phân tích cú pháp phần tử cú pháp thứ hai hay không dựa vào điều kiện định trước thứ nhất khi chế độ hợp nhất được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ hai này biểu thị liệu chế độ thứ nhất hay chế độ thứ hai được áp dụng cho khối hiện thời;

xác định liệu có phân tích cú pháp phần tử cú pháp thứ ba hay không dựa vào điều kiện định trước thứ hai khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ ba này biểu thị chế độ trong số chế độ thứ ba và chế độ thứ tư được áp dụng cho khối hiện thời;

xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba;

thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định; và

tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời,

trong đó điều kiện định trước thứ nhất bao gồm ít nhất một điều kiện trong số điều kiện nhờ đó chế độ thứ ba sử dụng được và điều kiện nhờ đó chế độ thứ tư sử dụng được.

9. Thiết bị xử lý tín hiệu video theo điểm 8, trong đó chế độ thứ ba và chế độ thứ tư được

đặt muộn hơn so với chế độ thứ nhất trong trình tự giải mã theo cú pháp dữ liệu hợp nhất.

10. Thiết bị xử lý tín hiệu video theo điểm 8, trong đó bộ xử lý được tạo cấu hình để phân tích cú pháp phần tử cú pháp thứ hai khi điều kiện định trước thứ nhất được thỏa mãn; và

trong đó khi điều kiện định trước thứ nhất không được thỏa mãn, thì giá trị của phần tử cú pháp thứ hai được suy ra là 1.

11. Thiết bị xử lý tín hiệu video theo điểm 8, trong đó khi điều kiện định trước thứ nhất không được thỏa mãn, thì phần tử cú pháp thứ hai được suy ra dựa vào phần tử cú pháp thứ tư biểu thị liệu chế độ hợp nhất dựa trên khối phụ có được áp dụng cho khối hiện thời hay không.

12. Thiết bị xử lý tín hiệu video theo điểm 8, trong đó điều kiện định trước thứ hai bao gồm điều kiện nhờ đó chế độ thứ tư sử dụng được.

13. Thiết bị xử lý tín hiệu video theo điểm 8, trong đó điều kiện định trước thứ hai bao gồm ít nhất một điều kiện trong số các điều kiện liên quan đến việc liệu chế độ thứ ba sử dụng được theo trình tự hiện thời hay không, liệu chế độ thứ tư sử dụng được theo trình tự hiện thời hay không, liệu số lượng ứng viên tối đa dùng cho chế độ thứ tư có lớn hơn 1 hay không, liệu chiều rộng của khối hiện thời có nhỏ hơn kích thước định trước thứ nhất hay không và liệu chiều cao của khối hiện thời có nhỏ hơn kích thước định trước thứ hai hay không.

14. Thiết bị xử lý tín hiệu video theo điểm 8, trong đó khi phần tử cú pháp thứ hai có giá trị bằng 1, thì bộ xử lý được tạo cấu hình để thu được phần tử cú pháp thứ năm biểu thị liệu chế độ được áp dụng cho khối hiện thời là chế độ thứ nhất hay chế độ thứ hai.

15. Phương pháp xử lý tín hiệu video, phương pháp này bao gồm các bước:

mã hóa phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp nhất có được áp dụng cho khối hiện thời hay không;

xác định liệu có mã hóa phần tử cú pháp thứ hai hay không dựa vào điều kiện định trước thứ nhất khi chế độ hợp nhất được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ hai biểu thị liệu chế độ thứ nhất hay chế độ thứ hai được áp dụng cho khối hiện thời;

xác định liệu có mã hóa phần tử cú pháp thứ ba hay không dựa vào điều kiện định trước thứ hai khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ ba biểu thị chế độ được áp dụng cho khối hiện thời trong số chế độ thứ ba hoặc chế độ thứ tư;

xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba;

thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định; và
tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời,

trong đó điều kiện định trước thứ nhất bao gồm ít nhất một điều kiện trong số điều kiện nhờ đó chế độ thứ ba sử dụng được và điều kiện nhờ đó chế độ thứ tư sử dụng được.

16. Vật ghi không khả biến đọc được bởi máy tính lưu trữ dòng bit, dòng bit này được giải mã bằng phương pháp giải mã,

trong đó phương pháp giải mã bao gồm các bước:

phân tích cú pháp phần tử cú pháp thứ nhất biểu thị liệu chế độ hợp nhất có được áp dụng cho khối hiện thời hay không;

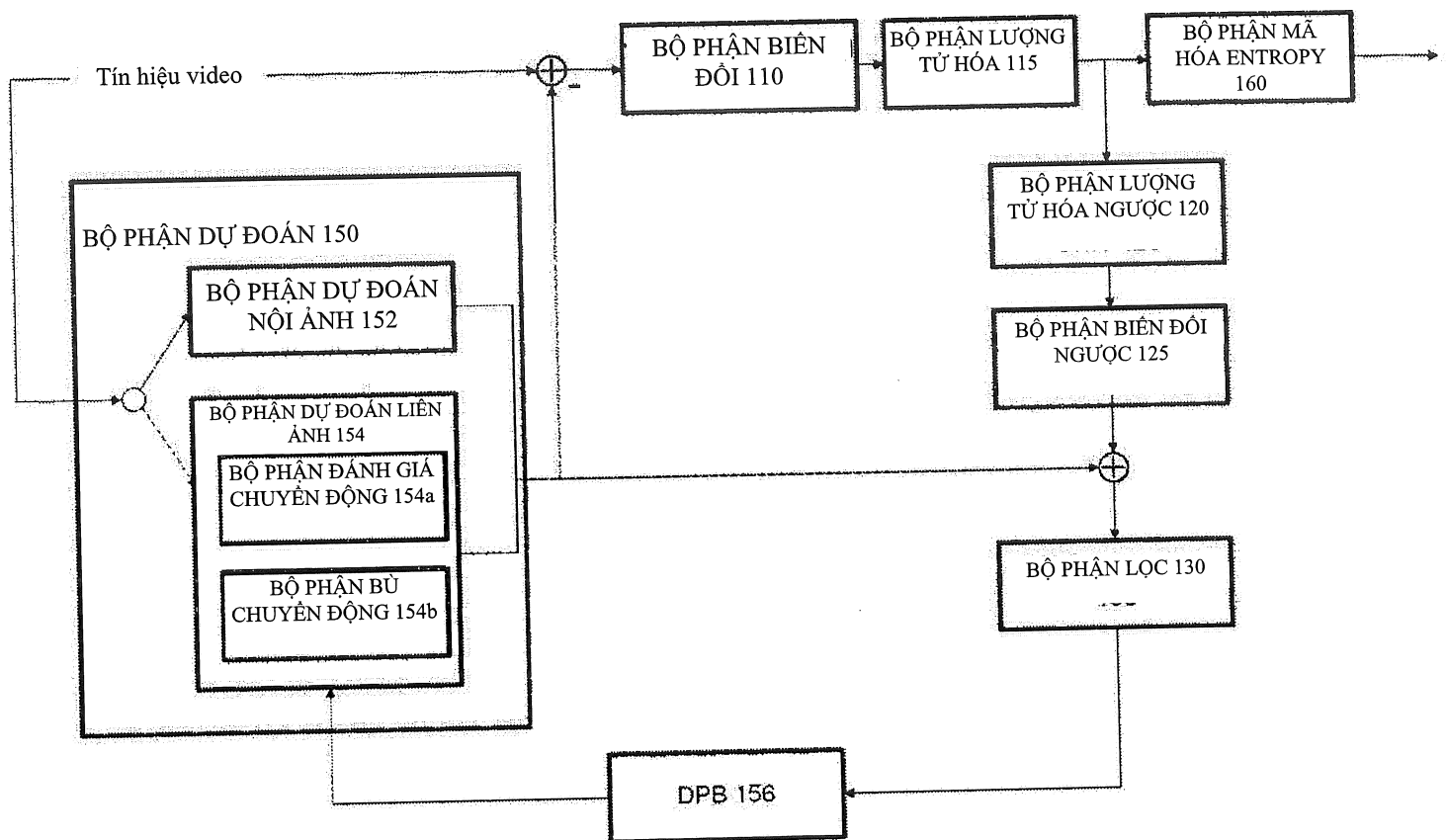
xác định liệu có phân tích cú pháp phần tử cú pháp thứ hai hay không dựa vào điều kiện định trước thứ nhất khi chế độ hợp nhất được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ hai này biểu thị liệu chế độ thứ nhất hay chế độ thứ hai được áp dụng cho khối hiện thời;

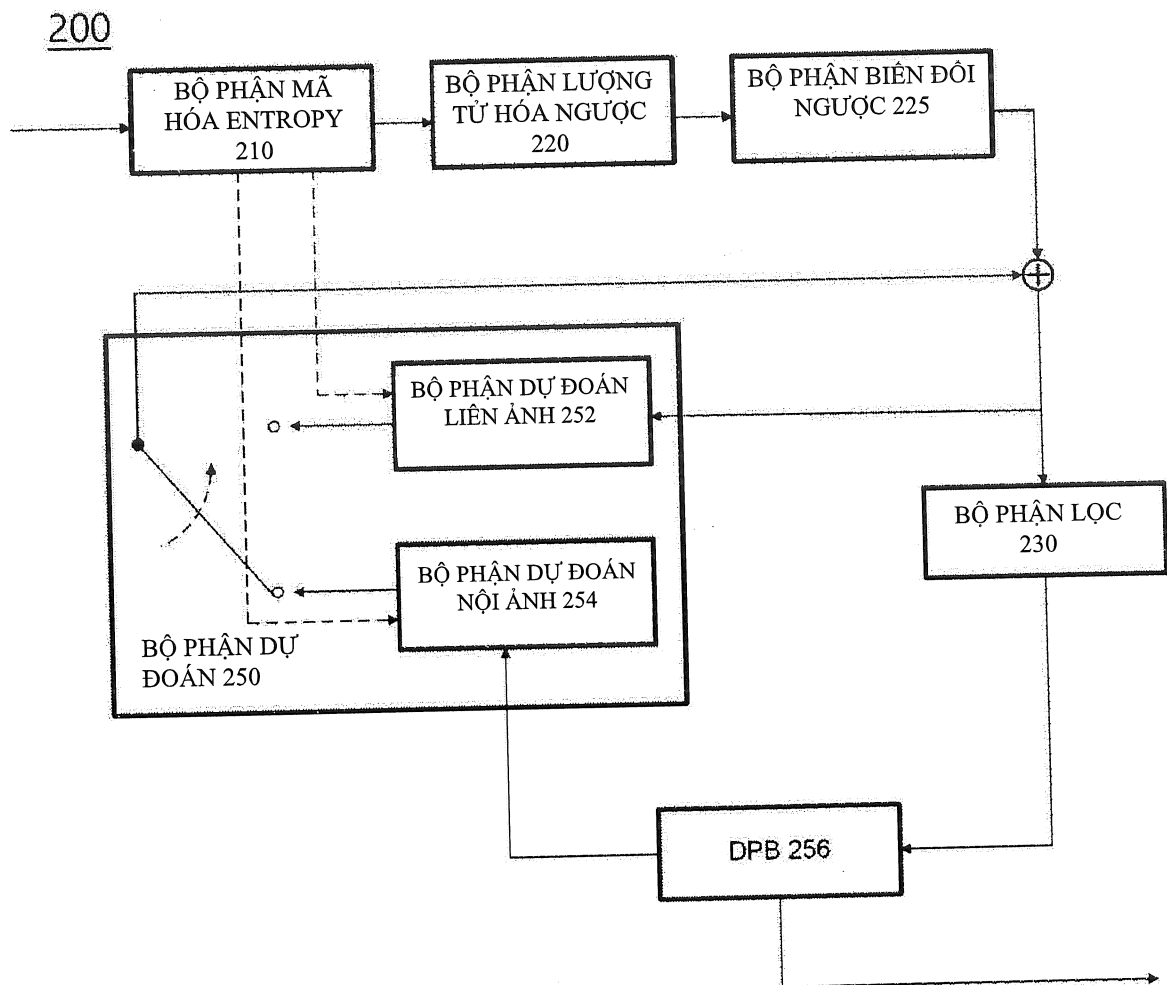
xác định liệu có phân tích cú pháp phần tử cú pháp thứ ba hay không dựa vào điều kiện định trước thứ hai khi chế độ thứ nhất và chế độ thứ hai không được áp dụng cho khối hiện thời, trong đó phần tử cú pháp thứ ba này biểu thị chế độ trong số chế độ thứ ba và chế độ thứ tư được áp dụng cho khối hiện thời;

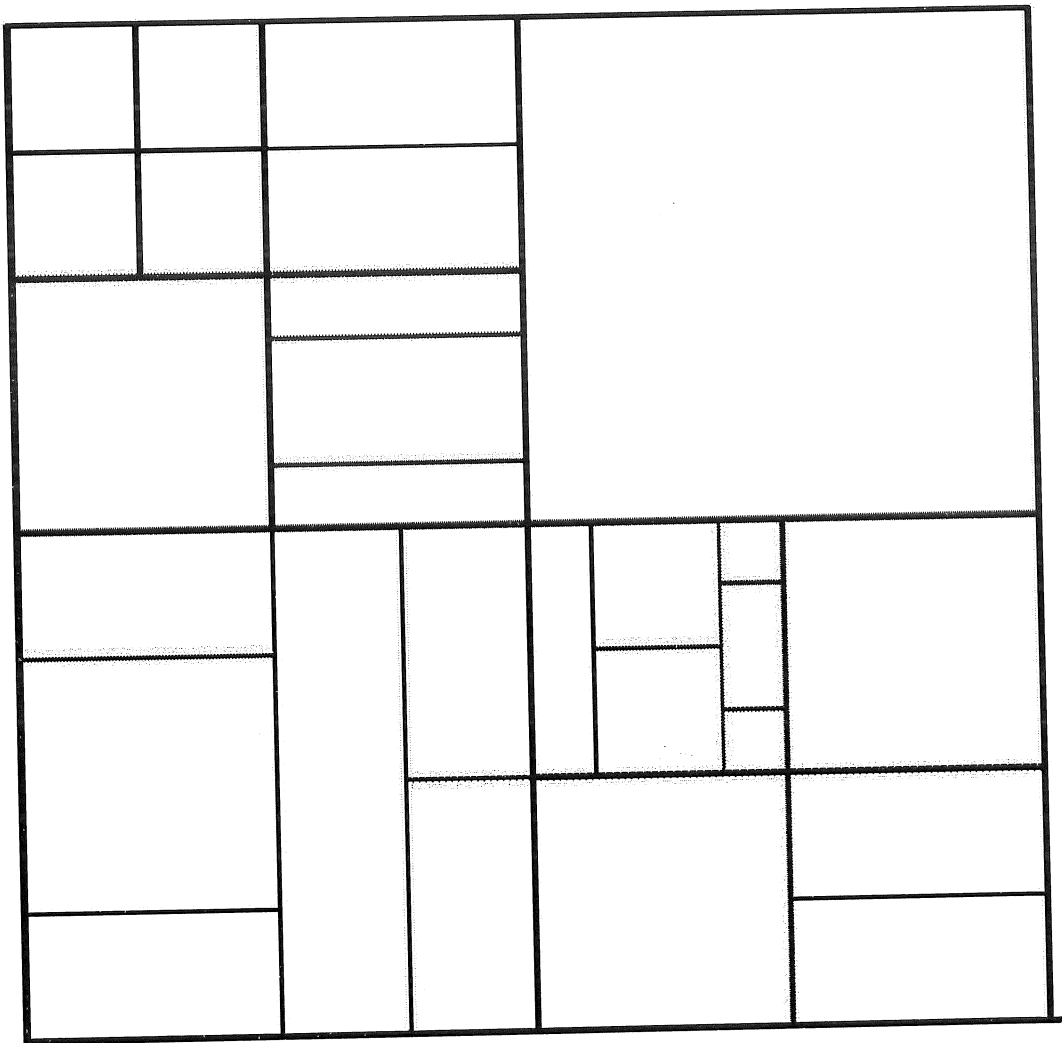
xác định chế độ được áp dụng cho khối hiện thời dựa vào phần tử cú pháp thứ hai hoặc phần tử cú pháp thứ ba;

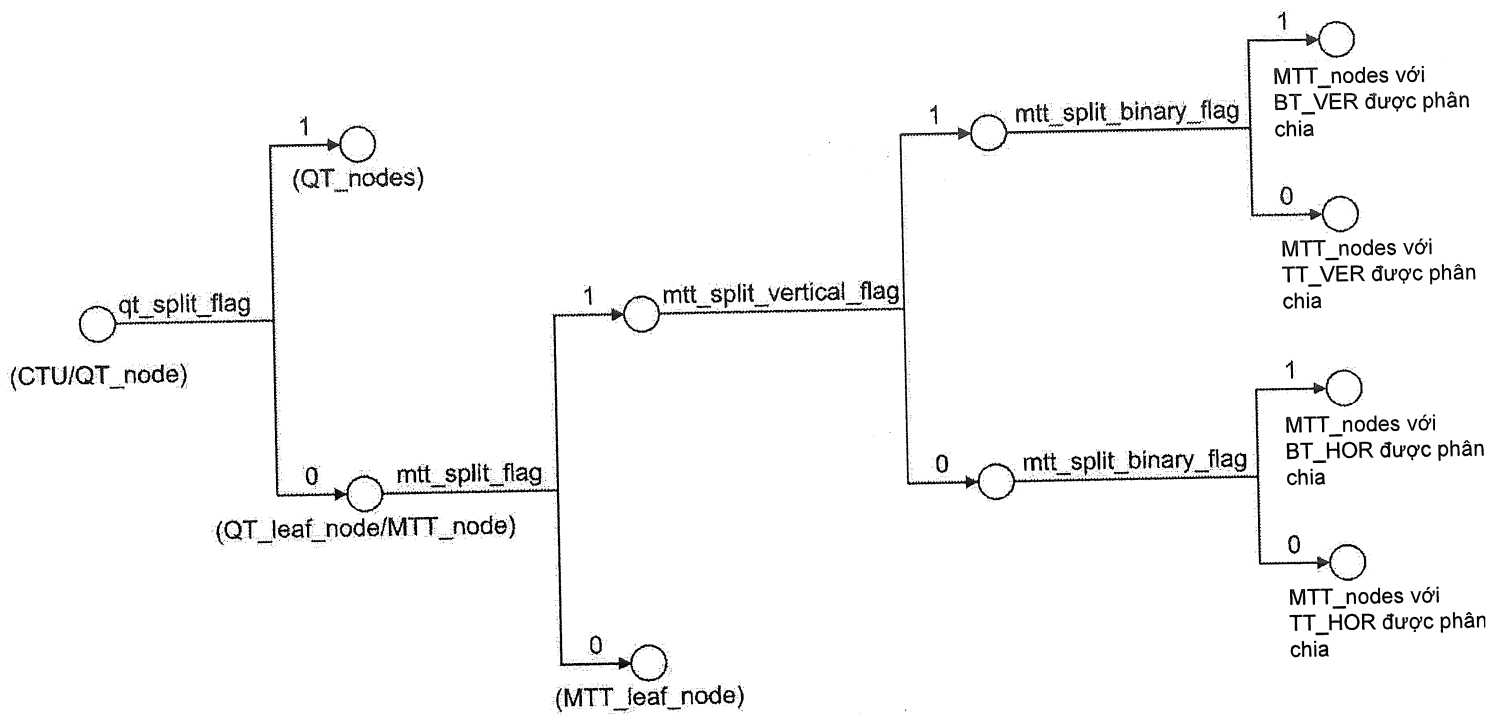
thu được thông tin chuyển động của khối hiện thời dựa vào chế độ đã xác định; và
tạo ra khối dự đoán của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối hiện thời,

trong đó điều kiện định trước thứ nhất bao gồm ít nhất một điều kiện trong số điều kiện nhờ đó chế độ thứ ba sử dụng được và điều kiện nhờ đó chế độ thứ tư sử dụng được.

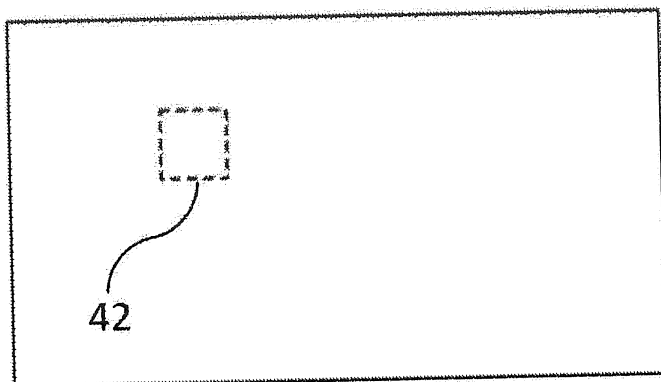
100



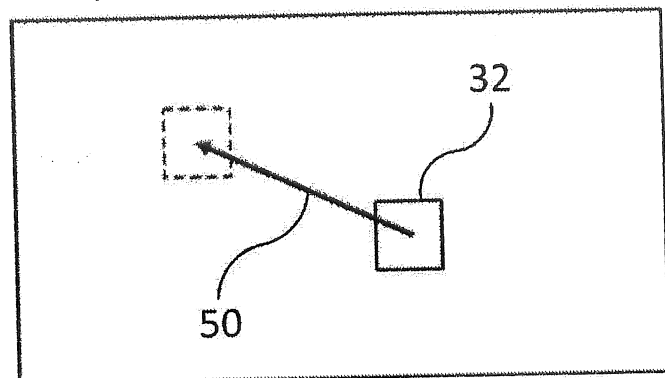


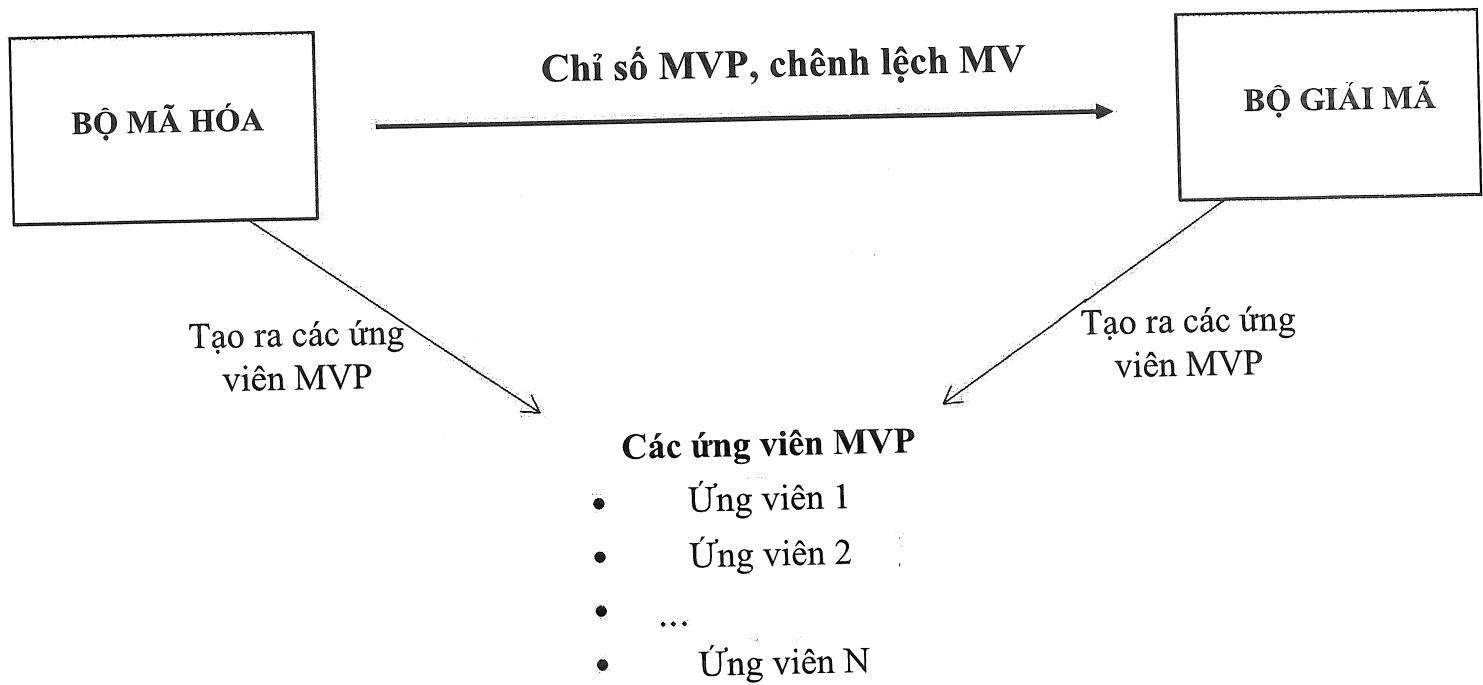


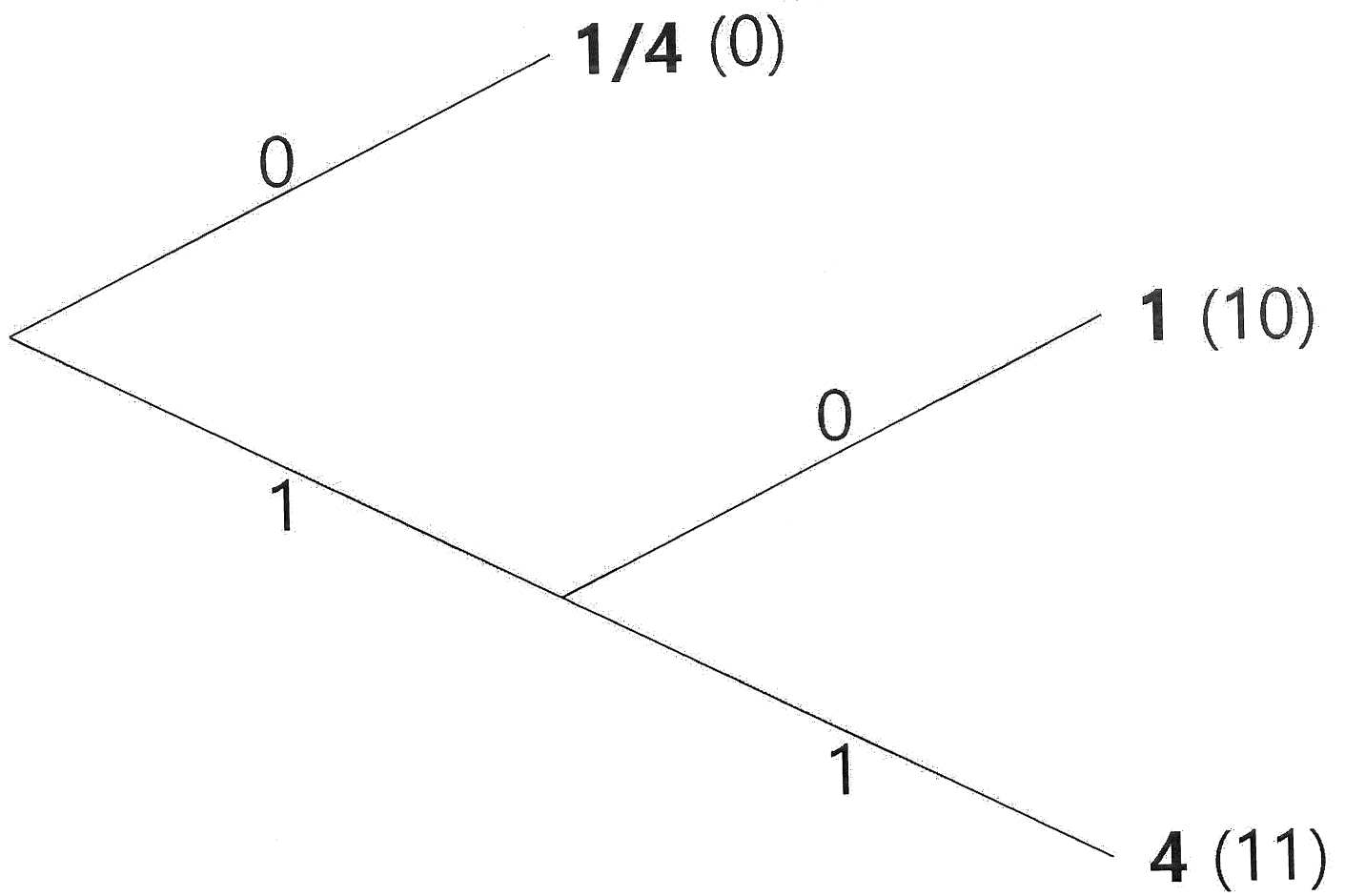
ẢNH THAM CHIẾU



ẢNH HIỆN THỜI







coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	
if(slice_type != I) {	
cu_skip_flag [x0][y0]	
if(cu_skip_flag[x0][y0] == 0)	
pred_mode_flag	
}	
if(CuPredMode[x0][y0] == MODE_INTRA) { /* intra */	
} else { /* MODE_INTER */	
if(cu_skip_flag[x0][y0] == 0) {	
merge_flag [x0][y0]	
if(merge_flag[x0][y0]) {	
merge_data(x0, y0, cbWidth, cbHeight)	
} else {	S801
mmvd_flag [x0][y0]	
if(mmvd_flag[x0][y0] == 1) {	
mmvd_merge_flag [x0][y0]	
mmvd_distance_idx [x0][y0]	S802
mmvd_direction_idx [x0][y0]	
} else {	
}	
}	
}	
}	
if(!pcm_flag[x0][y0]) {	
if(CuPredMode[x0][y0] != MODE_INTRA && cu_skip_flag[x0][y0] == 0)	S803
cu_cbf	
if(cu_cbf)	S804
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
}	
}	

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	
if(slice_type != I) {	
cu skip flag [x0][y0]	
if(cu skip flag[x0][y0] == 0)	
pred mode flag	
}	
if(CuPredMode[x0][y0] == MODE_INTRA) { /* intra */	
} else { /* MODE_INTER */	
if(cu skip flag[x0][y0] == 0) {	
merge flag [x0][y0]	
if(merge flag[x0][y0]) {	
merge_data(x0, y0, cbWidth, cbHeight)	
} else {	
mmvd flag [x0][y0]	
if(mmvd flag[x0][y0] == 1) {	
mmvd merge flag [x0][y0]	
mmvd distance idx [x0][y0]	
mmvd direction idx [x0][y0]	
} else { /* AMVP, affine Inter, etc. */	
}	
}	
}	
}	
if(!pcm_flag[x0][y0]) {	
if(CuPredMode[x0][y0] != MODE_INTRA && cu_skip_flag[x0][y0] == 0 && mmvd_flag[	S901
x0][y0] == 0)	
cu cbf	
if(cu cbf)	
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
}	
}	
}	

(a)

Các chế độ được báo hiệu	Cờ chính quy	Cờ MMVD	Cờ khối phụ	Cờ CIIP
Chính quy	1	-	-	-
MMVD	0	1	-	-
Khối phụ	0	0	1	-
CIIP	0	0	0	1
Tam giác	0	0	0	0

(b)

Các chế độ được báo hiệu	Cờ chính quy	Cờ MMVD	Cờ khối phụ
Chính quy	1	-	-
MMVD	0	1	-
Khối phụ	0	0	1
Tam giác	0	0	0

merge_data(x0, y0, cbWidth, cbHeight) {	
regular_merge_flag [x0][y0]	S1101
if(!(cbWidth == 4 && cbHeight == 4) && !(cbWidth == 8 && cbHeight == 4)	S1102
&& !(cbWidth == 4 && cbHeight == 8) && !regular_merge_flag[x0][y0])	
{	
mmvd_flag [x0][y0]	
if(mmvd_flag[x0][y0] == 1) {	
...	
} else {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	S1103
merge_subblock_flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx [x0][y0]	
} else {	
if(sps_mh_intra_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	S1104
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	
mh_intra_flag [x0][y0]	
if(mh_intra_flag[x0][y0]) {	
...	
}	
}	
}	
merge_triangle_flag[x0][y0] =	S1105
!regular_merge_flag[x0][y0] && !mmvd_flag[x0][y0]	
&& !merge_subblock_flag[x0][y0] && !mh_intra_flag[x0][y0]	
if(merge_triangle_flag[x0][y0])	
merge_triangle_idx [x0][y0]	
else if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
regular merge flag [x0][y0]	S1201
if(regular merge flag[x0][y0] && MaxNumMergeCand > 1)	
merge idx [x0][y0]	
if(!regular merge flag[x0][y0])	
{	
mmvd flag [x0][y0]	
if(mmvd flag[x0][y0] == 1) {	
...	
} else if (mmvd flag[x0][y0] == 0) {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge subblock flag [x0][y0]	
if(merge subblock flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge subblock idx [x0][y0]	
} else {	
if(sps_mh_intra_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	
mh intra flag [x0][y0]	
if(mh_intra_flag[x0][y0]) {	
if (cbWidth <= 2 * cbHeight cbHeight <= 2 * cbWidth)	
mh intra luma mpm flag [x0][y0]	
if(mh_intra_luma_mpm_flag[x0][y0])	
mh intra luma mpm idx [x0][y0]	
}	
}	
}	
if(merge_triangle_flag[x0][y0]) {	
merge triangle idx [x0][y0]	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
regular merge flag [x0][y0]	
if(regular merge flag[x0][y0] && MaxNumMergeCand > 1)	
merge idx [x0][y0]	
if(!regular merge flag[x0][y0])	
{	
if(merge_subblock_conditions mh_intra_conditions merge_triangle_conditions	S1301
)	
mmvd flag [x0][y0]	
if(mmvd flag[x0][y0] == 1) {	
...	
} else if (mmvd flag[x0][y0] == 0) {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8	
&& (mh_intra_conditions merge_triangle_conditions))	
merge subblock flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge subblock idx [x0][y0]	
} else {	
if(sps_mh_intra_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	
mh intra flag [x0][y0]	
if(mh_intra_flag[x0][y0]) {	
...	
}	
}	
if(merge_triangle_flag[x0][y0]) {	
merge triangle idx [x0][y0]	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if (CuPredMode[x0][y0] == MODE_IBC) {	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	S1401
} else {	
regular_merge_flag [x0][y0]	
if(regular_merge_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
if(!regular_merge_flag[x0][y0] && !(cbWidth == 8 && cbHeight == 4) && !(cbWidth == 4 && cbHeight == 8))	
{	S1402
if(sps_mmvd_enabled_flag)	
mmvd_flag [x0][y0]	
if(mmvd_flag[x0][y0] == 1) {	
...	
} else if(mmvd_flag[x0][y0] == 0) {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	S1403
merge_subblock_flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx [x0][y0]	
} else {	
if(sps_mh_intra_enabled_flag && cu_skip_flag[x0][y0] == 0 && (cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	S1404
mh_intra_flag [x0][y0]	
if(mh_intra_flag[x0][y0]) {	
...	
}	
}	
if(merge_triangle_flag[x0][y0]) {	
merge_triangle_split_dir [x0][y0]	
merge_triangle_idx0 [x0][y0]	
merge_triangle_idx1 [x0][y0]	
}	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if (CuPredMode[x0][y0] == MODE_IBC) { // ...	
} else {	
if(sps mmvd enabled flag cbWidth + cbHeight > 12))	S1501
regular merge flag [x0][y0]	
if(regular merge flag[x0][y0] && MaxNumMergeCand > 1)	
merge idx [x0][y0]	
if(!regular_merge_flag[x0][y0] && !(cbWidth == 8 && cbHeight == 4) && !(cbWidth == 4 && cbHeight == 8))	
{	
if(sps mmvd enabled flag)	
mmvd flag [x0][y0]	
if(mmvd flag[x0][y0] == 1) {	
...	
} else if (mmvd_flag[x0][y0] == 0) {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge subblock flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge subblock idx [x0][y0]	
} else {	
if(sps_mh_intra_enabled_flag && cu_skip_flag[x0][y0] == 0 && (cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	
mh intra flag [x0][y0]	
if(mh_intra_flag[x0][y0]) {	
....	
}	
}	
if(merge_triangle_flag[x0][y0]) {	
merge triangle split dir [x0][y0]	
merge triangle idx0 [x0][y0]	
merge triangle idx1 [x0][y0]	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if(CuPredMode[x0][y0] == MODE_IBC) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
} else {	
regular_merge_flag[x0][y0]	
if(regular_merge_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
if(!regular_merge_flag[x0][y0] && !(cbWidth == 8 && cbHeight == 4) && !(cbWidth == 4 && cbHeight == 8))	
{	
if(sps_mmvd_enabled_flag)	
mmvd_flag[x0][y0]	
if(mmvd_flag[x0][y0] == 1) {	
...	
} else if(mmvd_flag[x0][y0] == 0) {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge_subblock_flag[x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx[x0][y0]	
} else {	
if(sps_triangle_enabled_flag && tile_group_type == B && cbWidth * cbHeight >= 64)	
merge_triangle_flag[x0][y0]	
if(merge_triangle_flag[x0][y0]) {	
...	
} else {	
if(ciip_flag[x0][y0]) {	
// CIIP related syntax elements	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
}	
}	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if (CuPredMode[x0][y0] == MODE_IBC) {	
...	
} else {	
regular_merge_flag [x0][y0]	
if(regular_merge_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
if(!regular_merge_flag[x0][y0] && !(cbWidth == 8 && cbHeight == 4) && !(cbWidth == 4 && cbHeight == 8))	
{	
if(sps_mmvd_enabled_flag)	
mmvd_flag [x0][y0]	
if(mmvd_flag[x0][y0] == 1) {	
...	
} else if(mmvd_flag[x0][y0] == 0) {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge_subblock_flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx [x0][y0]	
} else {	
if(sps_triangle_enabled_flag && tile_group_type == B && cbWidth * cbHeight >= 64	
&& sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128)	
merge_triangle_flag [x0][y0]	
if(merge_triangle_flag[x0][y0]) {	
...	
} else {	
if(ciip_flag[x0][y0]) {	
// CIIP related syntax elements	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
}	
}	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if(CuPredMode[x0][y0] == MODE_IBC) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
} else {	
regular_merge_flag[x0][y0]	
if(regular_merge_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
if(!regular_merge_flag[x0][y0] && !(cbWidth == 8 && cbHeight == 4) && !(cbWidth == 4	
&& cbHeight == 8))	
{	
if(sps_mmvd_enabled_flag)	
mmvd_flag[x0][y0]	
if(mmvd_flag[x0][y0] == 1) {	
...	
} else if(mmvd_flag[x0][y0] == 0) {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge_subblock_flag[x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx[x0][y0]	
} else {	
if(sps_triangle_enabled_flag && tile_group_type == B && cbWidth * cbHeight >= 64	
&& cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128)	
merge_triangle_flag[x0][y0]	
if(merge_triangle_flag[x0][y0]) {	
...	
} else {	
if(ciip_flag[x0][y0]) {	
// CIIP related syntax elements	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
}	
}	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if(CuPredMode[x0][y0] == MODE_IBC) {	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
} else {	
regular_merge_flag [x0][y0]	
if(regular_merge_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
if(!regular_merge_flag[x0][y0] && !(cbWidth == 8 && cbHeight == 4) && !(cbWidth == 4	
&& cbHeight == 8))	
{	
if(sps_mmvd_enabled_flag)	
mmvd_flag [x0][y0]	
if(mmvd_flag[x0][y0] == 1) {	
...	
} else if(mmvd_flag[x0][y0] == 0) {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge_subblock_flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx [x0][y0]	
} else {	
if(sps_triangle_enabled_flag && tile_group_type == B && cbWidth * cbHeight >= 64	
&& cu_skip_flag[x0][y0] == 0)	
merge_triangle_flag [x0][y0]	
if(merge_triangle_flag[x0][y0]) {	
...	
} else {	
if(ciip_flag[x0][y0]) {	
// CIIP related syntax elements	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
}	
}	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if(CuPredMode[x0][y0] == MODE_IBC) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
} else {	
if(sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	
regular_merge_flag[x0][y0]	
if(regular_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32)	
mmvd_merge_flag[x0][y0]	
if(mmvd_merge_flag[x0][y0] == 1) {	
...	
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	
ciip_flag[x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
}	
if(ciip_flag[x0][y0] == 0) {	
if(MaxNumTriangleMergeCand >= 2 && sps_triangle_enabled_flag && slice_type == B	
&& cbWidth * cbHeight >= 64)	
merge_triangle_flag[x0][y0]	
if(merge_triangle_flag[x0][y0]) {	
...	
} else {	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx[x0][y0]	
}	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if(CuPredMode[x0][y0] == MODE_IBC) {	
...	
} else {	
if(sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	
regular merge flag [x0][y0]	
if(regular_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge idx [x0][y0]	
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32)	
mmvd merge flag [x0][y0]	
if(mmvd_merge_flag[x0][y0] == 1) {	
...	
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	
ciip flag [x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge idx [x0][y0]	
}	
if(ciip_flag[x0][y0] == 0) {	
if(MaxNumTriangleMergeCand >= 2 && sps_triangle_enabled_flag && slice_type == B	
&& cbWidth * cbHeight >= 64	
&& MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge triangle flag [x0][y0]	
if(merge_triangle_flag[x0][y0]) {	
...	
} else {	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge subblock idx [x0][y0]	
}	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if (CuPredMode[x0][y0] == MODE_IBC) {	
...	
} else {	
if(sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	
regular merge flag [x0][y0]	
if (regular merge flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge idx [x0][y0]	
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32)	
mmvd merge flag [x0][y0]	
if(mmvd merge flag[x0][y0] == 1) {	
...	
} else {	
if(MaxNumTriangleMergeCand >= 2 && sps_triangle_enabled_flag && slice_type == B	
&& cbWidth * cbHeight >= 64	
&& ((MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128))	
merge triangle flag [x0][y0]	
if(merge triangle flag[x0][y0]) {	
...	
} else {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8	
&& sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128)	
merge subblock flag [x0][y0]	
if(merge subblock flag[x0][y0] == 1) {	
...	
} else {	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge idx [x0][y0]	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if (CuPredMode[x0][y0] == MODE_IBC) {	
...	
} else {	
if(sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	
regular_merge_flag [x0][y0]	
if (regular_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32)	
mmvd_merge_flag [x0][y0]	
if(mmvd_merge_flag[x0][y0] == 1) {	
...	
} else {	
if(MaxNumTriangleMergeCand >= 2 && sps_triangle_enabled_flag && slice_type == B	
&& cbWidth * cbHeight >= 64	
&& ((MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128)) {	
merge_triangle_flag [x0][y0]	
if(merge_triangle_flag[x0][y0]) {	
...	
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128	
&& MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8) {	
ciip_flag [x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
}	
if(ciip_flag[x0][y0] == 0) {	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx [x0][y0]	
}	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
if (CuPredMode[x0][y0] == MODE_IBC) {	
...	
} else {	
if(sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	
regular_merge_flag [x0][y0]	
if (regular_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32)	
mmvd_merge_flag [x0][y0]	
if(mmvd_merge_flag[x0][y0] == 1) {	
...	
} else {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge_subblock_flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx [x0][y0]	
} else {	
if(MaxNumTriangleMergeCand >= 2 && sps_triangle_enabled_flag && slice_type == B	
&& cbWidth * cbHeight >= 64 && sps_ciip_enabled_flag && cu_skip_flag	
[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128)	
merge_triangle_flag [x0][y0]	
if(merge_triangle_flag[x0][y0]) {	
...	
} else {	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
}	
}	
}	
}	

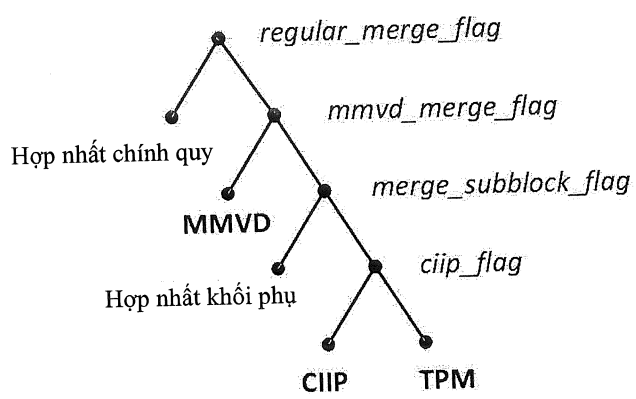
merge_data(x0, y0, cbWidth, cbHeight) {	
if (CuPredMode[x0][y0] == MODE_IBC) {	
...	
} else {	
if(sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	2501
regular merge flag [x0][y0]	
if (regular_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge idx [x0][y0]	
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32)	2502
mmvd merge flag [x0][y0]	
if(mmvd_merge_flag[x0][y0] == 1) {	
...	
} else {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	2503
merge subblock flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge subblock idx [x0][y0]	
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && (cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	2504
ciip flag [x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge idx [x0][y0]	
}	
if(MergeTriangleFlag[x0][y0]) {	2505
merge triangle split dir [x0][y0]	
merge triangle idx0 [x0][y0]	
merge triangle idx1 [x0][y0]	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
...	
if ((sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	2601
&& (mmvd_condition subblock_merge_condition ciip_condition triangle_merge_condition))	
regular_merge_flag[x0][y0]	
if (regular_merge_flag[x0][y0] == 1) {	
if (MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
} else {	
if (sps_mmvd_enabled_flag && cbWidth * cbHeight != 32	2602
&& (subblock_merge_condition ciip_condition triangle_merge_condition))	
mmvd_merge_flag[x0][y0]	
if (mmvd_merge_flag[x0][y0] == 1) {	
...	
} else {	
if (MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8	2603
&& (ciip_condition triangle_merge_condition))	
merge_subblock_flag[x0][y0]	
if (merge_subblock_flag[x0][y0] == 1) {	
if (MaxNumSubblockMergeCand > 1)	
merge_subblock_idx[x0][y0]	
} else {	
if (sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	2604
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128	
&& triangle_merge_condition) {	
ciip_flag[x0][y0]	
if (ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
}	
if (MergeTriangleFlag[x0][y0]) {	2605
...	
}	
}	

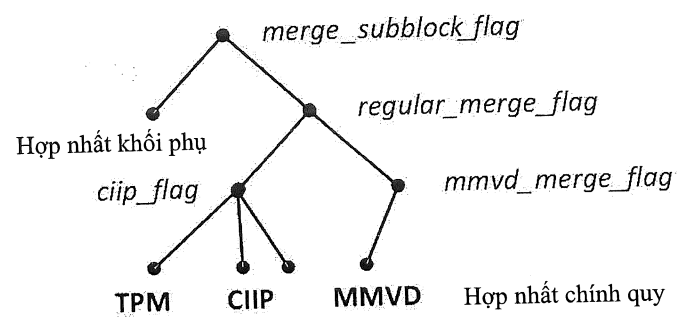
merge_data(x0, y0, cbWidth, cbHeight) {	
...	
if((sps_mmvd_enabled_flag cbWidth * cbHeight != 32) && (sps_mmvd_enabled_flag (sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2 && (cbWidth * cbHeight) >= 64) (sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128 && (cbWidth * cbHeight) >= 64) (MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)))	2701
regular_merge_flag[x0][y0]	
if(regular_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32 && (sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2 && (cbWidth * cbHeight) >= 64) (sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128 && (cbWidth * cbHeight) >= 64) (MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)))	2702
mmvd_merge_flag[x0][y0]	
if(mmvd_merge_flag[x0][y0] == 1) {	
...	
} else {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8 && ((sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) (sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128)))	2703
merge_subblock_flag[x0][y0]	
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && (cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128 && sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) {	2704
ciip_flag[x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
}	
if(MergeTriangleFlag[x0][y0]) {	2705
...	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
...	
if((sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	2801
&& (sps_mmvd_enabled_flag (sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2 &	
& (cbWidth * cbHeight) >= 64)	
(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128 && (cbWidth * cbHeight) >= 64)	
(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)) {	
regular_merge_flag[x0][y0]	
if(regular_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32	2802
&& ((sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2)	
(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128)	
(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)) {	
mmvd_merge_flag[x0][y0]	
if(mmvd_merge_flag[x0][y0] == 1) {	
...	
} else {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8	2803
&& ((sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2)	
(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128)) {	
merge_subblock_flag[x0][y0]	
...	
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	2804
(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128	
&& sps_triangle_enabled_flag && slice_type == B && MaxNumTriangleMergeCand >= 2) {	
ciip_flag[x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
}	
if(MergeTriangleFlag[x0][y0]) {	2805
...	
}	
}	

(a)



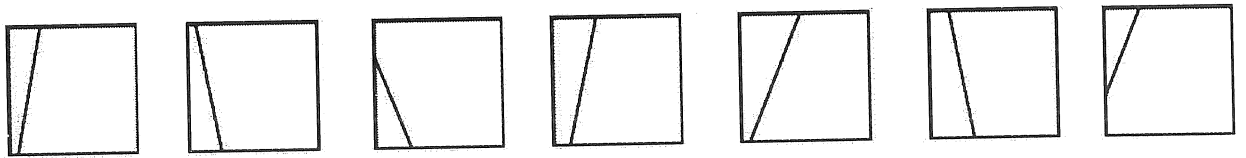
(b)



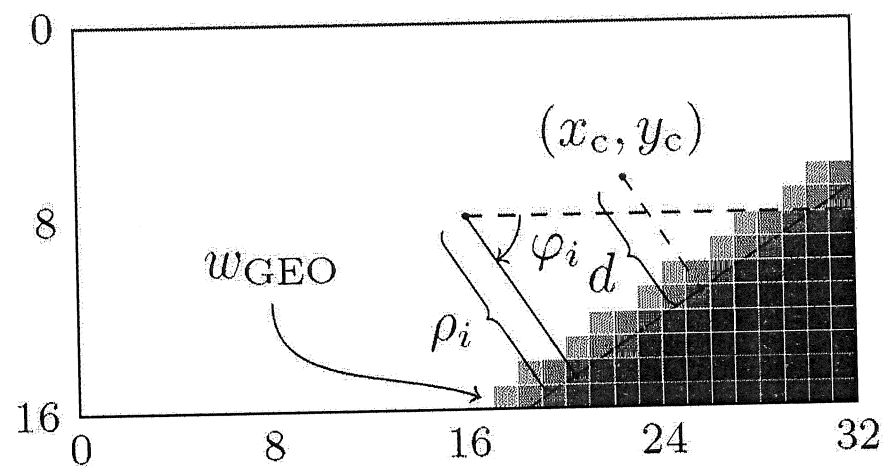
merge_data(x0, y0, cbWidth, cbHeight) {	
...	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge_subblock_flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx [x0][y0]	
} else {	
if((cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128 && ((sps_ciip_enabled_flag &	
& cu_skip_flag[x0][y0] == 0) (sps_triangle_enabled_flag && slice_type == B)) {	
regular_merge_flag [x0][y0]	
}	
if(regular_merge_flag[x0][y0]) {	
if(sps_mmvd_enabled_flag)	
{	
mmvd_merge_flag [x0][y0]	
...	
} else {	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
}	
}	
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && sps_triangle_enabled_flag && slice_type == B) {	
ciip_flag [x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
}	
if(! ciip_flag[x0][y0]) {	
...	
}	
}	

merge_data(x0, y0, cbWidth, cbHeight) {	
...	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge_subblock_flag [x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx [x0][y0]	
} else {	
if((cbWidth * cbHeight) >= 64 && ((sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
cbWidth < 128 && cbHeight < 128) (sps_triangle_enabled_flag && slice_type == B)) {	
regular_merge_flag [x0][y0]	
}	
if(regular_merge_flag[x0][y0]) {	
if(sps_mmvd_enabled_flag)	
{	
mmvd_merge_flag [x0][y0]	
...	
}	
}	
if(mmvd_merge_flag[x0][y0] == 0 && MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && cbWidth < 128 && cbHeight < 128 &	
& sps_triangle_enabled_flag && slice_type == B) {	
ciip_flag [x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx [x0][y0]	
}	
if(! ciip_flag[x0][y0]) {	
...	
}	
}	

CÁC VÍ DỤ VỀ PHÂN VÙNG HÌNH HỌC



PHÂN MÔ TẢ PHÂN VÙNG HÌNH HỌC



merge_data(x0, y0, cbWidth, cbHeight, chType) {	
if (CuPredMode[chType][x0][y0] == MODE_IBC) {	
if(MaxNumIbcMergeCand > 1)	
merge_idx[x0][y0]	
} else {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	3301
merge_subblock_flag[x0][y0]	
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx[x0][y0]	
} else {	
if((sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && (cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) (sps_triangle_enabled_flag && MaxNumTriangleMergeCand > 1 && cbWidth >= 8 && cbHeight >= 8))	3302
regular_merge_flag[x0][y0]	
if(regular_merge_flag[x0][y0] == 1) {	
if(sps_mmvd_enabled_flag)	
mmvd_merge_flag[x0][y0]	
if(mmvd_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
mmvd_cand_flag[x0][y0]	
mmvd_distance_idx[x0][y0]	
mmvd_direction_idx[x0][y0]	
} else {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
}	
} else {	
if(sps_ciip_enabled_flag && sps_triangle_enabled_flag && MaxNumTriangleMergeCand > 1 && cu_skip_flag[x0][y0] == 0 && cbWidth >= 8 && cbHeight >= 8 && cbWidth < 128 && cbHeight < 128) {	3303
ciip_flag[x0][y0]	
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx[x0][y0]	
if(!ciip_flag[x0][y0] && MaxNumTriangleMergeCand > 1) {	
merge_geo_idx[x0][y0]	
merge_triangle_idx0[x0][y0]	
merge_triangle_idx1[x0][y0]	
}	
}	
}	
}	
}	
}	
}	
}	

