



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

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

(11) 
1-0021517

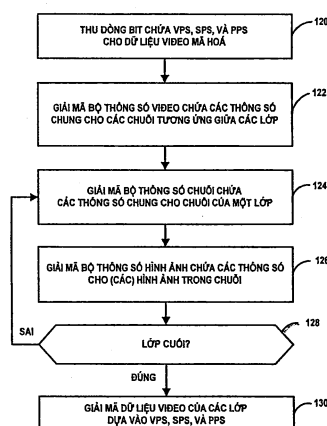
(51)⁷ **H04N 19/31, 19/46, 19/70**

(13) **B**

- (21) 1-2014-02717 (22) 11.01.2013
(86) PCT/US2013/021227 11.01.2013 (87) WO2013/106705 18.07.2013
(30) 61/586,777 14.01.2012 US
61/587,777 16.01.2012 US
61/588,629 19.01.2012 US
61/637,195 23.04.2012 US
61/637,774 24.04.2012 US
13/738,377 10.01.2013 US
(45) 26.08.2019 377 (43) 25.02.2015 323
(73) QUALCOMM INCORPORATED (US)
Attn: International IP Administration, 5775 Morehouse Drive, San Diego, California
92121, United States of America
(72) CHEN, Ying (CN), WANG, Ye-Kui (CN)
(74) Công ty TNHH Quốc tế D & N (D&N INTERNATIONAL CO.,LTD.)

(54) **PHƯƠNG PHÁP VÀ THIẾT BỊ MÃ HÓA DỮ LIỆU VIDEO**

(57) Theo sáng chế, bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video hoặc bộ giải mã dữ liệu video, được tạo cấu hình để mã hoá bộ thông số video (VPS: Video Parameter Set) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS. Bộ mã hoá dữ liệu video có thể mã hoá bộ thông số VPS cho dữ liệu video theo chuẩn mã hoá dữ liệu video đạt hiệu quả cao, chuẩn mã hoá dữ liệu video có nhiều cảnh nhìn, chuẩn mã hoá dữ liệu video có thể mở rộng cấp độ, hoặc các chuẩn mã hoá dữ liệu video khác hoặc các phiên bản mở rộng của các chuẩn mã hoá dữ liệu video. Bộ thông số VPS có thể chứa dữ liệu xác định các thông số cho các chuỗi tương ứng của dữ liệu video trong các lớp khác nhau (ví dụ, cảnh nhìn, lớp chất lượng, hoặc tương tự). Các thông số của bộ thông số VPS có thể cung cấp thông tin chỉ báo cách thức mã hoá dữ liệu video tương ứng.



Lĩnh vực kỹ thuật được đề cập

Sáng chế liên quan đến lĩnh vực mã hoá dữ liệu video.

Tình trạng kỹ thuật của sáng chế

Các tính năng video kỹ thuật số có thể được đưa vào áp dụng trong rất nhiều thiết bị, bao gồm máy thu hình kỹ thuật số, hệ thống phát rộng trực tiếp kỹ thuật số, hệ thống phát rộng không dây, thiết bị kỹ thuật số hỗ trợ cá nhân (*PDA: Personal Digital Assistant*), máy tính xách tay hoặc máy tính để bàn, máy tính bảng, thiết bị đọc sách điện tử, camera kỹ thuật số, thiết bị ghi kỹ thuật số, thiết bị phát lại đa phương tiện kỹ thuật số, thiết bị trò chơi có hình ảnh, bàn giao tiếp trò chơi có hình ảnh, máy điện thoại di động hoặc máy điện thoại vô tuyến vệ tinh, thiết bị được gọi là “máy điện thoại thông minh”, thiết bị hội thảo từ xa có truyền hình, thiết bị truyền dòng dữ liệu video và các thiết bị tương tự khác. Thiết bị video kỹ thuật số áp dụng kỹ thuật nén dữ liệu video, như kỹ thuật được mô tả trong các chuẩn MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Part 10, Advanced Video Coding (AVC), chuẩn mã hoá dữ liệu video đạt hiệu quả cao (*HEVC: High Efficiency Video Coding*) đang được phát triển hiện nay, và phiên bản mở rộng của các chuẩn nêu trên. Phiên bản dự thảo hiện nay của chuẩn HEVC sắp tới được công bố tại địa chỉ http://phenix.int-evry.fr/jct/doc_end_user/documents/7_Geneva/wg11/JCTVC-G1103-v3.zip. Các thiết bị video có thể truyền, thu, mã hoá, giải mã và/hoặc lưu trữ thông tin video dạng số hiệu quả hơn nhờ áp dụng các kỹ thuật nén dữ liệu video như vậy.

Các kỹ thuật mã hoá dữ liệu video thực hiện kỹ thuật dự báo không gian (dự báo nội hình ảnh) và/hoặc dự báo thời gian (dự báo liên hình ảnh) để giảm bớt hoặc loại bỏ dữ liệu dư vốn có trong các chuỗi dữ liệu video. Đối với phương pháp mã hoá dữ liệu video theo khối, lát video (tức là, một khung video hoặc một phần khung video) có thể được phân tách ra thành các khối video, khối video này cũng có thể được gọi là khối cấu trúc cây, đơn vị mã hoá (*CU: Coding Unit*) và/hoặc nút mã hoá. Các khối video trong lát hình ảnh mã hoá dự báo nội cấu trúc (I) được mã hoá bằng cách áp dụng kỹ thuật dự báo không gian dựa vào các mẫu chuẩn ở các khối liền kề trong cùng một hình ảnh. Các khối video trong lát hình ảnh mã hoá dự báo liên cấu

trúc (P hoặc B) có thể áp dụng kỹ thuật dự báo không gian dựa vào các mẫu chuẩn ở các khối liền kề trong cùng một hình ảnh hoặc áp dụng kỹ thuật dự báo thời gian dựa vào các mẫu chuẩn trong các hình ảnh chuẩn khác. Hình ảnh có thể được gọi là khung, và hình ảnh chuẩn có thể được gọi là khung chuẩn.

Kỹ thuật dự báo không gian hoặc thời gian tạo ra khối dự báo cho khối cần mã hoá. Dữ liệu dư biểu diễn giá trị chênh lệch điểm ảnh giữa khối ban đầu cần mã hoá và khối dự báo. Khối mã hoá dự báo liên cấu trúc được mã hoá theo vectơ chuyển động tham chiếu đến một khối mẫu chuẩn tạo nên khối dự báo, và dữ liệu dư chỉ báo giá trị chênh lệch giữa khối mã hoá và khối dự báo. Khối mã hoá dự báo nội cấu trúc được mã hoá theo chế độ mã hoá dự báo nội cấu trúc và dữ liệu dư. Để nén thêm, dữ liệu dư có thể được biến đổi từ miền điểm ảnh sang miền biến đổi, tạo ra các hệ số biến đổi dư, sau đó các hệ số này có thể được lượng tử hoá. Các hệ số biến đổi đã lượng tử hoá, ban đầu được sắp xếp thành ma trận hai chiều, có thể được quét để tạo ra vectơ một chiều của các hệ số biến đổi, và kỹ thuật mã hoá entropy có thể được áp dụng để đạt hiệu quả nén cao hơn.

Bản chất kỹ thuật của sáng chế

Nói chung, sáng chế đề cập đến kỹ thuật mã hoá các bộ thông số và các đơn vị lớp trừu tượng hoá mạng (*NAL: Network Abstraction Layer*) để mã hoá dữ liệu video. Các kỹ thuật này có thể được áp dụng cho dữ liệu mã hoá một lớp, như dữ liệu video hai chiều, và còn được áp dụng cho dữ liệu video theo chuẩn mã hoá dữ liệu video có thể mở rộng cấp độ (*SVC: Scalable Video Coding*) và dữ liệu video theo chuẩn mã hoá dữ liệu video có nhiều cảnh nhìn (*MVC: Multiview Video Coding*). Do đó, các bộ thông số và các đơn vị NAL có thể tương thích lẫn nhau giữa các loại dữ liệu video khác nhau. Ví dụ, bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video hoặc bộ giải mã dữ liệu video, có thể mã hoá bộ thông số video (*VPS: Video Parameter Set*) xác định các thông số cho một hoặc nhiều lớp dữ liệu video. Các lớp này có thể tương ứng với, ví dụ, các lớp SVC (có tốc độ khung, độ phân giải không gian và/hoặc các mức chất lượng khác nhau) và/hoặc các cảnh nhìn của dữ liệu MVC (ví dụ, các chuỗi hình ảnh của một cảnh được chụp từ các góc chụp khác nhau của camera so với trục nằm ngang).

Theo một phương án làm ví dụ, phương pháp mã hoá dữ liệu video bao gồm

các bước mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

Theo phương án khác làm ví dụ, thiết bị mã hoá dữ liệu video có bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video hoặc bộ giải mã dữ liệu video, được tạo cấu hình để mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

Theo phương án khác làm ví dụ, thiết bị mã hoá dữ liệu video bao gồm phương tiện mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và phương tiện mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

Theo phương án khác làm ví dụ, vật ghi đọc được bằng máy tính trên đó lưu trữ các lệnh khi được thi hành sẽ ra lệnh cho bộ xử lý mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

Một hoặc nhiều phương án làm ví dụ sẽ được mô tả chi tiết dưới đây có dựa vào hình vẽ kèm theo và phần mô tả chi tiết sáng chế. Các dấu hiệu, mục đích và ưu điểm khác của sáng chế sẽ được hiểu rõ hơn khi xem phần mô tả chi tiết sáng chế và hình vẽ, cùng với các điểm yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ khối thể hiện ví dụ về hệ thống mã hoá và giải mã dữ liệu video có thể áp dụng kỹ thuật mã hoá các bộ thông số và các đơn vị lớp trừu tượng hoá mạng (NAL) cho một hoặc nhiều lớp dữ liệu video.

Fig.2 là sơ đồ khối thể hiện ví dụ về bộ mã hoá dữ liệu video có thể áp dụng kỹ thuật mã hoá các bộ thông số và các đơn vị NAL cho một hoặc nhiều lớp dữ liệu

video.

Fig.3 là sơ đồ khối thể hiện ví dụ về bộ giải mã dữ liệu video có thể áp dụng kỹ thuật mã hoá các bộ thông số và các đơn vị NAL cho một hoặc nhiều lớp dữ liệu video.

Fig.4 là sơ đồ khái niệm thể hiện ví dụ về mẫu dự báo MVC.

Fig.5 là sơ đồ khái niệm thể hiện bộ thông số video (VPS) và các bộ thông số lớp (*LPS: Layer Parameter Set*).

Fig.6 là sơ đồ khái niệm thể hiện ví dụ về bộ thông số nhóm (*GPS: Grouping Parameter Set*) và các quan hệ của bộ thông số GPS với các bộ thông số khác và các phần đầu lát.

Fig.7 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video theo các kỹ thuật nêu trong sáng chế.

Fig.8 là lưu đồ thể hiện ví dụ về phương pháp giải mã dữ liệu video theo các kỹ thuật nêu trong sáng chế.

Fig.9 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào số lượng lớp thời gian như được báo hiệu trong bộ thông số VPS.

Fig.10 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào số lượng hình ảnh cần sắp xếp lại trong một hoặc nhiều lớp và số lượng hình ảnh cần lưu trữ vào bộ nhớ đệm hình ảnh giải mã.

Fig.11 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào các thông số bộ giải mã chuẩn giả định (*HRD: Hypothetical Reference Decoder*) được báo hiệu trong bộ thông số VPS.

Fig.12 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào dữ liệu mở rộng được báo hiệu trong bộ thông số VPS.

Mô tả chi tiết sáng chế

Nói chung, sáng chế đề cập đến phương pháp và thiết bị mã hoá dữ liệu video sử dụng bộ thông số video (VPS). Dữ liệu video có thể được phân loại theo cấu trúc phân cấp gồm nhiều lớp, chuỗi hình ảnh trong một lớp cho trước, hình ảnh trong một

chuỗi, các lát trong một hình ảnh, và các khối (ví dụ, khối ảnh lớn hoặc các đơn vị cấu trúc cây mã hoá) trong một lát. Bộ thông số chuỗi (*SPS: Sequence Parameter Set*) có thể được dùng để báo hiệu các thông số ít khi thay đổi cho chuỗi hình ảnh, và bộ thông số hình ảnh (*PPS: Picture Parameter Set*) có thể được dùng để báo hiệu các thông số ít khi thay đổi cho các hình ảnh riêng biệt.

Theo các kỹ thuật nêu trong sáng chế, bộ thông số VPS có thể báo hiệu các thông số ít khi thay đổi cho nhiều chuỗi trên các lớp tương ứng. Có nghĩa là, bộ thông số VPS có thể chứa các thông số cho một tập hợp chuỗi đồng vị theo thời gian trên các lớp khác nhau. Các lớp khác nhau có thể có, ví dụ, các cảnh nhìn khác nhau cho dữ liệu video có nhiều cảnh nhìn, các lớp chất lượng khác nhau, các lớp độ phân giải không gian khác nhau, các lớp có thể chuyển đổi cấp độ thời gian (tức là, các lớp cho phép có tốc độ khung khác nhau), và các loại tương tự khác. Theo cách này, một bộ thông số VPS có thể được cung cấp cho nhiều lớp khác nhau, cho nên bộ thông số VPS đó báo hiệu các thông số là thông số chung cho mỗi lớp tương ứng (ví dụ, các chuỗi tương ứng trong các lớp tương ứng). Dòng bit có thể được coi là chứa mỗi lớp, và các lớp tương ứng có thể tạo ra các dòng bit con tương ứng. Ngoài ra, dòng bit con có thể tương ứng với dạng kết hợp của từ hai lớp trở lên.

Sáng chế mô tả các ví dụ về dữ liệu có thể được đưa vào trong bộ thông số VPS. Trong một số ví dụ, dữ liệu như vậy có thể có thông tin chỉ báo số lượng lớp con (ví dụ, số lượng lớp con tối đa) trong các lớp tương ứng. Ví dụ, bộ thông số VPS có thể chứa dữ liệu báo hiệu số lượng lớp thời gian và/hoặc số lượng lớp thời gian tối đa (ví dụ, số nhận dạng lớp thời gian cao nhất).

Trong ví dụ khác, theo cách khác hoặc theo cách bổ sung, bộ thông số VPS có thể có dữ liệu gần giống như dữ liệu bất kỳ đã được báo hiệu trước đó trong bộ thông số SPS (tức là, được báo hiệu trong các bộ thông số SPS thông thường). Theo cách này, khi các chuỗi của từ hai lớp trở lên trong dòng bit chứa các thông số giống nhau hoặc gần giống nhau, thì bộ mã hoá dữ liệu video có thể mã hoá bộ thông số VPS để báo hiệu các thông số cho các chuỗi trên các lớp, chứ không mã hoá phần dữ liệu dư của các dữ liệu đó trong các bộ thông số SPS tương ứng cho các chuỗi khác nhau trên các lớp khác nhau.

Theo cách khác hoặc theo cách bổ sung, bộ thông số VPS có thể chứa dữ liệu

xác định thông tin sử dụng dữ liệu video (*VUI: Video Usability Information*), như thông tin biểu diễn dữ liệu video, các thông số bộ giải mã chuẩn giả định (*HRD*), và/hoặc thông tin giới hạn dòng bit. Thông tin giới hạn dòng bit có thể chứa các giới hạn về khoảng vector chuyển động, dung lượng bộ nhớ đệm hình ảnh giải mã (*DPB: Decoded Picture Buffer*) (ví dụ, tính theo số lượng hình ảnh được lưu trữ trong bộ nhớ đệm *DPB*), số lượng khung cần sắp xếp lại (tức là, thông tin chỉ báo số lượng khung được sắp xếp lại từ thứ tự giải mã chuyển thành thứ tự hiển thị), kích thước của các khối mã hoá (ví dụ, các khối ảnh lớn (*MB: MacroBlock*) hoặc các đơn vị cấu trúc cây mã hoá), và kích thước của các hình ảnh mã hoá. Bộ thông số *VPS* có thể còn cung cấp dữ liệu cho một hoặc nhiều bộ thông số *VPS* mở rộng, sao cho bộ thông số *VPS* có thể được mở rộng theo các chuẩn tương lai hoặc các phiên bản mở rộng của chuẩn *HEVC* sắp tới.

Fig.1 là sơ đồ khối thể hiện ví dụ về hệ thống mã hoá và giải mã dữ liệu video 10 có thể áp dụng kỹ thuật mã hoá các bộ thông số và các đơn vị lớp trừu tượng hoá mạng (*NAL*) cho một hoặc nhiều lớp dữ liệu video. Như được thể hiện trên Fig.1, hệ thống 10 bao gồm thiết bị nguồn 12 tạo ra dữ liệu video mã hoá sẽ được giải mã vào một thời điểm sau đó ở thiết bị đích 14. Cụ thể, thiết bị nguồn 12 cung cấp dữ liệu video đến thiết bị đích 14 thông qua vật ghi đọc được bằng máy tính 16. Thiết bị nguồn 12 và thiết bị đích 14 có thể là một thiết bị bất kỳ trong số rất nhiều thiết bị, như máy tính để bàn, máy tính sở tay (tức là, máy tính xách tay), máy tính bảng, bộ giải mã để bàn, máy điện thoại cầm tay như máy điện thoại “thông minh”, máy tính bảng “thông minh”, máy thu hình, camera, thiết bị hiển thị, thiết bị phát lại dữ liệu đa phương tiện kỹ thuật số, bàn giao tiếp trò chơi có hình ảnh, thiết bị truyền dòng dữ liệu video, hoặc các thiết bị tương tự khác. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể có tính năng truyền thông không dây.

Thiết bị đích 14 có thể thu dữ liệu video mã hoá cần giải mã thông qua vật ghi đọc được bằng máy tính 16. Vật ghi đọc được bằng máy tính 16 có thể là loại phương tiện hoặc thiết bị bất kỳ có khả năng chuyển dữ liệu video mã hoá từ thiết bị nguồn 12 đến thiết bị đích 14. Theo một phương án làm ví dụ, vật ghi đọc được bằng máy tính 16 có thể là phương tiện truyền thông cho phép thiết bị nguồn 12 truyền trực tiếp dữ liệu video mã hoá đến thiết bị đích 14 theo thời gian thực. Dữ liệu video mã hoá có thể

được điều biến theo một chuẩn truyền thông, như giao thức truyền thông không dây, và được truyền đến thiết bị đích 14. Phương tiện truyền thông có thể là phương tiện truyền thông không dây hoặc nối dây bất kỳ, như đường truyền phổ tần số vô tuyến (*RF: Radio Frequency*) hay một hoặc nhiều đường truyền vật lý. Phương tiện truyền thông có thể tạo nên một phần mạng truyền dữ liệu gói, như mạng cục bộ, mạng diện rộng, hoặc mạng toàn cầu như mạng internet. Phương tiện truyền thông có thể là bộ định tuyến, chuyển mạch, trạm cơ sở, hoặc mọi thiết bị khác có thể dùng để hỗ trợ truyền thông từ thiết bị nguồn 12 đến thiết bị đích 14.

Trong một số ví dụ, dữ liệu mã hoá có thể được xuất ra từ giao diện xuất 22 gửi đến thiết bị lưu trữ. Tương tự, dữ liệu mã hoá có thể được truy nhập từ thiết bị lưu trữ bằng giao diện nhập. Thiết bị lưu trữ có thể là một loại bất kỳ trong số rất nhiều loại phương tiện lưu trữ dữ liệu được truy nhập cục bộ hoặc phân tán như ổ đĩa cứng, đĩa Blu-ray, đĩa đa năng kỹ thuật số (*DVD: Digital Versatile Disc*), đĩa compac-bộ nhớ chỉ đọc (*CD-ROM: Compact Disc-Read Only Memory*), bộ nhớ tốc độ nhanh, bộ nhớ khả biến hoặc không khả biến, hoặc mọi phương tiện lưu trữ kỹ thuật số khác phù hợp để lưu trữ dữ liệu video mã hoá. Theo phương án khác làm ví dụ, thiết bị lưu trữ có thể tương ứng với máy chủ tệp hoặc thiết bị lưu trữ trung gian khác có thể lưu trữ dữ liệu video mã hoá được tạo ra bằng thiết bị nguồn 12. Thiết bị đích 14 có thể truy nhập dữ liệu video đã lưu trữ từ thiết bị lưu trữ thông qua cơ chế truyền dòng hoặc tải xuống. Máy chủ tệp có thể là loại máy chủ bất kỳ có khả năng lưu trữ dữ liệu video mã hoá và truyền dữ liệu video mã hoá đó đến thiết bị đích 14. Ví dụ về máy chủ tệp là máy chủ web (ví dụ, cho website), máy chủ giao thức truyền tệp (*FTP: File Transfer Protocol*), thiết bị lưu trữ mạng (*NAS: Network Attached Storage*), hoặc ổ đĩa cục bộ. Thiết bị đích 14 có thể truy nhập dữ liệu video mã hoá thông qua kết nối dữ liệu tiêu chuẩn bất kỳ, trong đó có kết nối internet. Kết nối này có thể là kênh không dây (ví dụ, kết nối Wi-Fi), kết nối nối dây (ví dụ, đường thuê bao kỹ thuật số (*DSL: Digital Subscriber Line*), môđem cáp, v.v.), hoặc kết hợp hai loại này phù hợp với việc truy nhập dữ liệu video mã hoá lưu trữ trên máy chủ tệp. Cơ chế truyền dữ liệu video mã hoá từ thiết bị lưu trữ có thể là truyền dòng, truyền tải xuống, hoặc kết hợp hai loại này.

Các kỹ thuật nêu trong sáng chế không nhất thiết chỉ giới hạn ở các ứng dụng

hoặc hệ thống không dây. Các kỹ thuật này có thể được áp dụng để mã hoá dữ liệu video khi hỗ trợ một ứng dụng đa phương tiện bất kỳ trong số nhiều ứng dụng đa phương tiện, như truyền hình phát rộng theo giao thức truyền vô tuyến, truyền hình cáp, truyền hình vệ tinh, truyền dòng dữ liệu video qua mạng internet, như truyền dòng thích ứng động theo giao thức HTTP (*DASH: Dynamic Adaptive Streaming over HTTP*), mã hoá dữ liệu video dạng số lưu trữ trên phương tiện lưu trữ dữ liệu, giải mã dữ liệu video dạng số lưu trữ trên phương tiện lưu trữ dữ liệu, hoặc các ứng dụng khác. Trong một số ví dụ, hệ thống 10 có thể được tạo cấu hình để hỗ trợ truyền dữ liệu video một chiều hoặc hai chiều nhằm hỗ trợ các ứng dụng như truyền dòng dữ liệu video, phát lại dữ liệu video, phát rộng dữ liệu video, và/hoặc điện thoại có truyền hình.

Trong ví dụ được thể hiện trên Fig.1, thiết bị nguồn 12 bao gồm nguồn dữ liệu video 18, bộ mã hoá dữ liệu video 20 và giao diện xuất 22. Thiết bị đích 14 bao gồm giao diện nhập 28, bộ giải mã dữ liệu video 30 và thiết bị hiển thị 32. Theo sáng chế, bộ mã hoá dữ liệu video 20 của thiết bị nguồn 12 có thể được tạo cấu hình để áp dụng các kỹ thuật mã hoá các bộ thông số và các đơn vị NAL cho một hoặc nhiều lớp dữ liệu video. Theo các phương án khác làm ví dụ, thiết bị nguồn và thiết bị đích có thể bao gồm các bộ phận hoặc cơ cấu khác. Ví dụ, thiết bị nguồn 12 có thể thu dữ liệu video từ nguồn dữ liệu video bên ngoài 18, như camera bên ngoài. Tương tự như vậy, thiết bị đích 14 có thể giao diện với thiết bị hiển thị bên ngoài, mà không có thiết bị hiển thị tích hợp.

Hệ thống 10 được thể hiện trên Fig.1 chỉ là một ví dụ. Kỹ thuật mã hoá các bộ thông số và các đơn vị NAL cho một hoặc nhiều lớp dữ liệu video có thể được thực hiện bằng mọi thiết bị mã hoá và/hoặc giải mã dữ liệu video dạng số. Mặc dù các kỹ thuật nêu trong sáng chế thường được thực hiện bằng thiết bị mã hoá dữ liệu video, nhưng các kỹ thuật này cũng có thể được thực hiện bằng bộ mã hoá/giải mã dữ liệu video, thường được gọi là “CODEC”. Ngoài ra, các kỹ thuật nêu trong sáng chế cũng có thể được thực hiện bằng bộ xử lý trước dữ liệu video. Thiết bị nguồn 12 và thiết bị đích 14 chỉ là các ví dụ về thiết bị mã hoá như vậy, trong đó thiết bị nguồn 12 tạo ra dữ liệu video mã hoá để truyền đến thiết bị đích 14. Trong một số ví dụ, các thiết bị 12, 14 có thể hoạt động theo cách gần như đối xứng sao cho mỗi thiết bị 12, 14 đều có

các bộ phận mã hoá và giải mã dữ liệu video. Vì vậy, hệ thống 10 có thể hỗ trợ truyền dữ liệu video một chiều hoặc hai chiều giữa các thiết bị video 12, 14, ví dụ, cho ứng dụng truyền dòng dữ liệu video, phát lại dữ liệu video, phát rộng dữ liệu video, hoặc điện thoại có truyền hình.

Nguồn dữ liệu video 18 của thiết bị nguồn 12 có thể là thiết bị chụp hình, như camera ghi hình, phương tiện lưu trữ dữ liệu video có dữ liệu video đã được ghi từ trước và/hoặc giao diện cung cấp dữ liệu video để thu dữ liệu video từ nhà cung cấp nội dung video. Ví dụ khác, nguồn dữ liệu video 18 có thể tạo ra dữ liệu đồ hoạ máy tính dùng làm nguồn dữ liệu video, hoặc kết hợp dữ liệu video truyền hình trực tiếp, dữ liệu video đã được ghi từ trước và dữ liệu video được tạo ra trên máy tính. Trong một số trường hợp, nếu nguồn dữ liệu video 18 là camera ghi hình, thì thiết bị nguồn 12 và thiết bị đích 14 có thể tạo nên thiết bị được gọi là máy điện thoại có camera hoặc máy điện thoại có truyền hình. Tuy nhiên, như đã nêu trên, các kỹ thuật nêu trong sáng chế có thể được áp dụng để mã hoá dữ liệu video nói chung, và có thể được áp dụng cho các ứng dụng không dây và/hoặc nối dây. Trong mỗi trường hợp, dữ liệu video được chụp, được chụp từ trước hoặc được tạo ra trên máy tính có thể được mã hoá bằng bộ mã hoá dữ liệu video 20. Sau đó, dữ liệu video mã hoá có thể được xuất ra bằng giao diện xuất 22 để ghi lên vật ghi đọc được bằng máy tính 16.

Vật ghi đọc được bằng máy tính 16 có thể là phương tiện khả biến, như tín hiệu phát rộng không dây hoặc tín hiệu truyền qua mạng nối dây, hoặc phương tiện lưu trữ (tức là, phương tiện lưu trữ không khả biến), như ổ đĩa cứng, ổ đĩa tác động nhanh, đĩa compac, đĩa video kỹ thuật số, đĩa Blu-ray, hoặc phương tiện khác đọc được bằng máy tính. Trong một số ví dụ, máy chủ mạng (không được thể hiện trên hình vẽ) có thể thu dữ liệu video mã hoá từ thiết bị nguồn 12 và cung cấp dữ liệu video mã hoá này cho thiết bị đích 14, ví dụ, thông qua tín hiệu truyền trên mạng. Tương tự, thiết bị máy tính của nhà máy sản xuất phương tiện lưu trữ, như nhà máy sản xuất đĩa, có thể thu dữ liệu video mã hoá từ thiết bị nguồn 12 và chế tạo ra đĩa chứa dữ liệu video mã hoá. Vì vậy, vật ghi đọc được bằng máy tính 16 có thể được hiểu là một hoặc nhiều vật ghi đọc được bằng máy tính có nhiều dạng khác nhau, trong nhiều ứng dụng khác nhau.

Giao diện nhập 28 của thiết bị đích 14 thu thông tin từ vật ghi đọc được bằng máy tính 16. Thông tin trên vật ghi đọc được bằng máy tính 16 có thể là thông tin cú

pháp được xác định bằng bộ mã hoá dữ liệu video 20, thông tin cú pháp này cũng được sử dụng cho bộ giải mã dữ liệu video 30, chứa các phần tử cú pháp mô tả các đặc trưng và/hoặc cách xử lý các khối và các đơn vị mã hoá khác, ví dụ, các nhóm hình ảnh (*GOP: Group of Pictures*). Thiết bị hiển thị 32 hiển thị dữ liệu video đã giải mã cho người dùng xem, và có thể là thiết bị hiển thị bất kỳ trong số nhiều thiết bị hiển thị như màn hình ống tia catôt (*CRT: Cathode Ray Tube*), màn hình tinh thể lỏng (*LCD: Liquid Crystal Display*), màn hình plasma, màn hình điốt phát quang hữu cơ (*OLED: Organic Light Emitting Diode*), hoặc loại thiết bị hiển thị khác.

Bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể hoạt động theo một chuẩn mã hoá dữ liệu video, như chuẩn mã hoá dữ liệu video đạt hiệu quả cao (*HEVC*) đang được phát triển hiện nay, và có thể tuân theo mô hình thử nghiệm *HEVC (HM: HEVC Test Model)*. Theo cách khác, bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể hoạt động theo các chuẩn độc quyền hoặc thông dụng khác, như chuẩn ITU-T H.264, được gọi theo cách khác là chuẩn *MPEG-4, Part 10, Advanced Video Coding (AVC)*, hoặc phiên bản mở rộng của các chuẩn này. Tuy nhiên, các kỹ thuật nêu trong sáng chế không bị giới hạn ở bất kỳ chuẩn mã hoá cụ thể nào. Ví dụ khác về các chuẩn mã hoá dữ liệu video là *MPEG-2* và *ITU-T H.263*. Tuy không được thể hiện trên Fig.1, nhưng theo một số khía cạnh, mỗi bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể được tích hợp với bộ mã hoá và giải mã dữ liệu audio, và có thể có các bộ dồn kênh-phân kênh thích hợp, hoặc phần cứng và phần mềm khác, để xử lý việc mã hoá cả dữ liệu audio và dữ liệu video trong một dòng dữ liệu chung hoặc trong các dòng dữ liệu riêng biệt. Nếu có thể, thì các bộ dồn kênh-phân kênh có thể tuân theo giao thức dồn kênh *ITU H.223*, hoặc các giao thức khác như giao thức gói dữ liệu người dùng (*UDP: User Datagram Protocol*).

Chuẩn *ITU-T H.264/MPEG-4 (AVC)* được tạo ra bởi nhóm chuyên gia mã hoá dữ liệu video (*VCEG: Video Coding Experts Group*) của tổ chức *ITU-T* kết hợp với nhóm chuyên gia ảnh động (*MPEG: Moving Picture Experts Group*) của tổ chức *ISO/IEC* tạo thành một nhóm hợp tác chung gọi là nhóm hợp tác liên kết mã hoá dữ liệu video (*JVT: Joint Video Team*). Theo một số khía cạnh, các kỹ thuật nêu trong sáng chế có thể được áp dụng cho các thiết bị thường tuân theo chuẩn *H.264*. Chuẩn *H.264* được mô tả trong tài liệu *ITU-T Recommendation H.264, Advanced Video*

Coding áp dụng cho các dịch vụ nghe nhìn chung, của nhóm nghiên cứu ITU-T Study Group, vào tháng 3 năm 2005, trong sáng chế, chuẩn này có thể được gọi là chuẩn H.264 hoặc đặc tả H.264, hay chuẩn hoặc đặc tả H.264/AVC. Nhóm Joint Video Team (JVT) tiếp tục làm việc để mở rộng chuẩn H.264/MPEG-4 AVC.

Mỗi bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể được thực hiện dưới dạng mạch mã hoá hoặc giải mã bất kỳ trong số rất nhiều mạch mã hoá hoặc giải mã phù hợp, nếu có, như một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (*DSP: Digital Signal Processor*), mạch tích hợp chuyên dụng (*ASIC: Application Specific Integrated Circuit*), mảng cửa lập trình được bằng trường (*FPGA: Field Programmable Gate Array*), mạch logic rời rạc, phần mềm, phần cứng, phần sụn hoặc mọi dạng kết hợp của các loại trên. Mỗi bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể nằm trong một hoặc nhiều bộ mã hoá hoặc bộ giải mã, mỗi bộ phận này có thể được tích hợp làm một phần của bộ mã hoá/giải mã kết hợp (*CODEC*) trong thiết bị tương ứng.

Nhóm hợp tác liên kết mã hoá dữ liệu video (*JCT-VC: Joint Collaboration Team on Video Coding*) đang làm việc để phát triển chuẩn HEVC. Nỗ lực tiêu chuẩn hoá HEVC dựa trên mô hình cải tiến thiết bị mã hoá dữ liệu video được gọi là mô hình thử nghiệm HEVC (*HM: HEVC Test Model*). Mô hình HM cho là các thiết bị mã hoá video có thêm vài khả năng so với các thiết bị hiện có, ví dụ, theo chuẩn ITU-T H.264/AVC. Ví dụ, trong khi chuẩn H.264 có chín chế độ mã hoá dự báo nội cấu trúc, thì mô hình HM có tới ba mươi ba chế độ mã hoá dự báo nội cấu trúc.

Nói chung, phiên bản dự thảo làm việc của mô hình HM mô tả khung video hoặc hình ảnh có thể được phân tách ra thành một chuỗi gồm các khối cấu trúc cây hoặc các đơn vị mã hoá lớn nhất (*LCU: Largest Coding Unit*) chứa cả mẫu độ chói lẫn các mẫu màu. Dữ liệu cú pháp trong dòng bit có thể xác định kích thước của đơn vị LCU, đó là đơn vị mã hoá lớn nhất tính theo số lượng điểm ảnh. Một lát gồm nhiều khối cấu trúc cây liên tiếp theo thứ tự giải mã. Khung video hoặc hình ảnh có thể được phân tách ra thành một hoặc nhiều lát. Mỗi khối cấu trúc cây có thể được phân tách ra thành các đơn vị mã hoá (CU) theo cấu trúc cây tứ phân. Thông thường, cấu trúc dữ liệu cây tứ phân có một nút cho mỗi đơn vị CU, với nút gốc tương ứng với khối cấu trúc cây. Nếu đơn vị CU được phân tách ra thành bốn đơn vị CU con, nút này tương

ứng với đơn vị CU gồm bốn nút lá, mỗi nút lá tương ứng với một trong số các đơn vị CU.

Mỗi nút của cấu trúc dữ liệu cây tứ phân có thể cung cấp dữ liệu cú pháp cho đơn vị CU tương ứng. Ví dụ, nút trong cấu trúc cây tứ phân có thể có cờ phân tách, chỉ báo việc đơn vị CU tương ứng với nút đó có được phân tách ra thành các đơn vị CU con hay không. Các phần tử cú pháp cho đơn vị CU có thể được xác định đệ quy, và có thể phụ thuộc vào việc đơn vị CU được phân tách ra thành các đơn vị CU con. Nếu đơn vị CU không được phân tách nữa, thì đơn vị CU đó được gọi là đơn vị CU nút lá. Trong sáng chế này, bốn đơn vị CU con của một đơn vị CU nút lá được gọi là đơn vị CU nút lá ngay cả khi không có sự phân tách rõ ràng của đơn vị CU nút lá ban đầu. Ví dụ, nếu đơn vị CU có kích thước 16x16 điểm ảnh không được phân tách nữa, thì bốn đơn vị CU con có kích thước 8x8 điểm ảnh cũng sẽ được gọi là các đơn vị CU nút lá mặc dù đơn vị CU có kích thước 16x16 điểm ảnh chưa từng được phân tách.

Đơn vị CU có mục đích giống như khối ảnh lớn theo chuẩn H.264, ngoại trừ việc đơn vị CU không có sự khác biệt về kích thước. Ví dụ, khối cấu trúc cây có thể được phân tách ra thành bốn nút con (còn được gọi là đơn vị CU con), và mỗi nút con có thể lại là nút cha và được phân tách ra thành bốn nút con khác. Nút con cuối cùng không phân tách nữa, được gọi là nút lá của cấu trúc cây tứ phân, là nút mã hoá, còn được gọi là đơn vị CU nút lá. Dữ liệu cú pháp liên quan đến dòng bit mã hoá có thể xác định số lần tối đa có thể phân tách khối cấu trúc cây, được gọi là độ sâu cao nhất của đơn vị CU, và cũng có thể xác định kích thước tối đa của các nút mã hoá. Do đó, dòng bit cũng có thể xác định đơn vị mã hoá nhỏ nhất (*SCU: Smallest Coding Unit*). Sáng chế sử dụng thuật ngữ “khối” để chỉ đơn vị bất kỳ trong số các đơn vị CU, PU hoặc TU trong trường hợp chuẩn HEVC, hoặc các cấu trúc dữ liệu tương tự trong trường hợp các chuẩn khác (ví dụ, khối ảnh lớn và các khối con của khối ảnh lớn theo chuẩn H.264/AVC).

Đơn vị CU có một nút mã hoá và các đơn vị dự báo (*PU: Prediction Unit*) và các đơn vị biến đổi (*TU: Transform Unit*) liên quan đến nút mã hoá đó. Kích thước của đơn vị CU tương ứng với kích thước của nút mã hoá và phải có dạng hình vuông. Kích thước của đơn vị CU có thể nằm trong khoảng từ 8x8 điểm ảnh đến kích thước của khối cấu trúc cây có giá trị tối đa là 64x64 điểm ảnh hoặc lớn hơn. Mỗi đơn vị CU

có thể có một hoặc nhiều đơn vị PU và một hoặc nhiều đơn vị TU. Dữ liệu cú pháp liên quan đến đơn vị CU có thể mô tả, ví dụ, chế độ phân tách đơn vị CU thành một hoặc nhiều đơn vị PU. Các chế độ phân tách có thể là khác nhau tùy thuộc vào việc đơn vị CU được mã hoá ở chế độ bỏ qua hay chế độ trực tiếp, được mã hoá ở chế độ dự báo nội cấu trúc hay được mã hoá ở chế độ dự báo liên cấu trúc. Các đơn vị PU có thể được phân tách ra thành khối có dạng không phải hình vuông. Dữ liệu cú pháp liên quan đến đơn vị CU cũng có thể mô tả, ví dụ, chế độ phân tách đơn vị CU thành một hoặc nhiều đơn vị TU theo cấu trúc cây tứ phân. Đơn vị TU có thể có dạng hình vuông hoặc dạng không phải hình vuông (ví dụ, dạng hình chữ nhật).

Chuẩn HEVC cho phép thực hiện các quy trình biến đổi theo đơn vị TU, các quy trình biến đổi đó có thể là khác nhau với các đơn vị CU khác nhau. Các đơn vị TU thường có kích thước dựa vào kích thước của các đơn vị PU trong một đơn vị CU cho trước được xác định đối với đơn vị LCU đã phân tách, tuy nhiên có thể không phải lúc nào cũng như vậy. Các đơn vị TU thường có kích thước nhỏ hơn hoặc bằng kích thước của các đơn vị PU. Trong một số ví dụ, các mẫu dữ liệu dư tương ứng với đơn vị CU có thể được phân tách ra thành các đơn vị nhỏ hơn sử dụng cấu trúc cây tứ phân gọi là “cấu trúc cây tứ phân dữ liệu dư” (*RQT: Residual QuadTree*). Các nút lá của cấu trúc RQT có thể được gọi là các đơn vị biến đổi (TU). Giá trị chênh lệch điểm ảnh liên quan đến các đơn vị TU có thể được biến đổi để tạo ra các hệ số biến đổi, các hệ số biến đổi này có thể được lượng tử hoá.

Đơn vị CU nút lá có thể có một hoặc nhiều đơn vị dự báo (PU). Nói chung, đơn vị PU thể hiện vùng không gian tương ứng với toàn bộ hoặc một phần đơn vị CU tương ứng, và có thể chứa dữ liệu để tìm mẫu chuẩn cho đơn vị PU. Ngoài ra, đơn vị PU còn chứa dữ liệu liên quan đến việc dự báo. Ví dụ, khi đơn vị PU được mã hoá dự báo nội cấu trúc, thì dữ liệu cho đơn vị PU có thể được đưa vào trong cấu trúc cây tứ phân dữ liệu dư (RQT), dữ liệu này có thể là dữ liệu mô tả chế độ dự báo nội cấu trúc cho đơn vị TU tương ứng với đơn vị PU. Ví dụ khác, khi đơn vị PU được mã hoá dự báo liên cấu trúc, thì đơn vị PU có thể chứa dữ liệu xác định một hoặc nhiều vector chuyển động cho đơn vị PU. Dữ liệu xác định vector chuyển động cho đơn vị PU có thể mô tả, ví dụ, thành phần theo phương nằm ngang của vector chuyển động, thành phần theo phương thẳng đứng của vector chuyển động, độ phân giải cho vector chuyển

động (ví dụ, độ chính xác bằng một phần tư điểm ảnh hoặc độ chính xác bằng một phần tám điểm ảnh), hình ảnh chuẩn mà vector chuyển động tham chiếu đến đó, và/hoặc danh mục hình ảnh chuẩn (ví dụ, danh mục 0, danh mục 1 hoặc danh mục C) cho vector chuyển động.

Đơn vị CU nút lá có một hoặc nhiều đơn vị PU cũng có thể có một hoặc nhiều đơn vị biến đổi (TU). Các đơn vị biến đổi có thể được xác định bằng cách sử dụng cấu trúc RQT (còn được gọi là cấu trúc cây tứ phân cho đơn vị TU), như đã nêu trên. Ví dụ, cờ phân tách có thể chỉ báo việc đơn vị CU nút lá được phân tách ra thành bốn đơn vị biến đổi. Sau đó, mỗi đơn vị biến đổi có thể được phân tách tiếp thành các đơn vị TU con khác. Khi đơn vị TU không được phân tách nữa, thì đơn vị TU đó có thể được gọi là đơn vị TU nút lá. Thông thường, đối với chế độ mã hoá dự báo nội cấu trúc, tất cả các đơn vị TU nút lá thuộc về một đơn vị CU nút lá đều dùng chung một chế độ dự báo nội cấu trúc. Có nghĩa là, cùng một chế độ dự báo nội cấu trúc thường được áp dụng để tính các giá trị dự báo cho tất cả các đơn vị TU của đơn vị CU nút lá. Đối với chế độ mã hoá dự báo nội cấu trúc, bộ mã hoá dữ liệu video có thể tính giá trị dư cho mỗi đơn vị TU nút lá sử dụng chế độ dự báo nội cấu trúc, dùng làm giá trị chênh lệch giữa phần đơn vị CU tương ứng với đơn vị TU và khối ban đầu. Đơn vị TU không nhất định chỉ giới hạn ở kích thước của đơn vị PU. Do đó, các đơn vị TU có thể lớn hơn hoặc nhỏ hơn so với đơn vị PU. Đối với chế độ mã hoá dự báo nội cấu trúc, đơn vị PU có thể nằm ở vị trí đồng vị với đơn vị TU nút lá tương ứng cho cùng một đơn vị CU đó. Trong một số ví dụ, kích thước lớn nhất của đơn vị TU nút lá có thể tương ứng với kích thước của đơn vị CU nút lá tương ứng.

Ngoài ra, các đơn vị TU của các đơn vị CU nút lá cũng có thể được liên hệ với các cấu trúc dữ liệu cây tứ phân tương ứng, được gọi là cấu trúc cây tứ phân dữ liệu dư (RQT). Có nghĩa là, đơn vị CU nút lá có thể có cấu trúc cây tứ phân chỉ báo cách phân tách đơn vị CU nút lá thành các đơn vị TU. Nút gốc của cấu trúc cây tứ phân cho đơn vị TU thường tương ứng với đơn vị CU nút lá, trong khi đó nút gốc của cấu trúc cây tứ phân cho đơn vị CU thường tương ứng với khối cấu trúc cây (hoặc LCU). Các đơn vị TU trong cấu trúc RQT không được phân tách nữa thì được gọi là đơn vị TU nút lá. Nói chung, sáng chế này sử dụng các thuật ngữ đơn vị CU và đơn vị TU để chỉ đơn vị CU nút lá và đơn vị TU nút lá tương ứng, trừ trường hợp có quy định khác.

Chuỗi dữ liệu video thường là một chuỗi khung video hoặc hình ảnh. Nhóm hình ảnh (GOP) thường là một chuỗi gồm một hoặc nhiều hình ảnh video. Nhóm GOP có thể chứa dữ liệu cú pháp ở phần đầu của nhóm GOP, ở phần đầu của một hoặc nhiều hình ảnh, hoặc ở vị trí khác, để mô tả số lượng hình ảnh có trong nhóm GOP. Mỗi lát hình ảnh có thể chứa dữ liệu cú pháp lát để mô tả chế độ mã hoá cho lát tương ứng. Bộ mã hoá dữ liệu video 20 thường hoạt động trên các khối video trong các lát video riêng biệt để mã hoá dữ liệu video. Khối video có thể tương ứng với nút mã hoá trong đơn vị CU. Các khối video có thể có kích thước cố định hoặc thay đổi, và có thể có kích thước khác nhau theo chuẩn mã hoá đã xác định.

Ví dụ, mô hình HM hỗ trợ dự báo cho nhiều kích thước đơn vị PU. Giả sử kích thước của một đơn vị CU cụ thể là $2N \times 2N$, thì mô hình HM hỗ trợ dự báo nội cấu trúc cho đơn vị PU có kích thước $2N \times 2N$ hoặc $N \times N$, và dự báo liên cấu trúc cho đơn vị PU đối xứng có kích thước $2N \times 2N$, $2N \times N$, $N \times 2N$ hoặc $N \times N$. Mô hình HM còn hỗ trợ phân tách không đối xứng để dự báo liên cấu trúc cho đơn vị PU có kích thước $2N \times nU$, $2N \times nD$, $nL \times 2N$ và $nR \times 2N$. Khi phân tách không đối xứng, một chiều của đơn vị CU không được phân tách, trong khi chiều còn lại được phân tách ra thành 25% và 75%. Phần đơn vị CU tương ứng với phần 25% có ký hiệu “n” sau thông tin chỉ báo “trên”, “dưới”, “trái” hoặc “phải”. Do đó, ví dụ, “ $2N \times nU$ ” dùng để chỉ đơn vị CU có kích thước $2N \times 2N$ được phân tách theo chiều ngang thành đơn vị PU có kích thước $2N \times 0,5N$ ở trên và đơn vị PU có kích thước $2N \times 1,5N$ ở dưới.

Trong sáng chế này, “ $N \times N$ ” và “N nhân N” có thể được sử dụng hoán đổi lẫn nhau để chỉ kích thước điểm ảnh của khối video tính theo chiều dọc và chiều ngang, ví dụ, 16×16 điểm ảnh hoặc 16 nhân 16 điểm ảnh. Thông thường, khối 16×16 sẽ có 16 điểm ảnh theo chiều dọc ($y = 16$) và 16 điểm ảnh theo chiều ngang ($x = 16$). Tương tự, khối $N \times N$ thường có N điểm ảnh theo chiều dọc và N điểm ảnh theo chiều ngang, trong đó N là giá trị nguyên không âm. Các điểm ảnh trong khối có thể được sắp xếp thành hàng và cột. Ngoài ra, các khối không nhất thiết phải có số lượng điểm ảnh theo chiều ngang bằng số lượng điểm ảnh theo chiều dọc. Ví dụ, các khối có thể có $N \times M$ điểm ảnh, trong đó M không nhất thiết phải bằng N.

Sau khi mã hoá dự báo nội cấu trúc hoặc mã hoá dự báo liên cấu trúc sử dụng các đơn vị PU của đơn vị CU, bộ mã hoá dữ liệu video 20 có thể tính dữ liệu dư cho

các đơn vị TU của đơn vị CU. Đơn vị PU có thể chứa dữ liệu cú pháp mô tả phương pháp hoặc chế độ tạo ra dữ liệu điểm ảnh dự báo ở miền không gian (còn được gọi là miền điểm ảnh) và các đơn vị TU có thể có các hệ số ở miền biến đổi sau khi áp dụng một quy trình biến đổi, ví dụ, biến đổi cosin rời rạc (*DCT: Discrete Cosine Transform*), biến đổi số nguyên, biến đổi dạng sóng, hoặc biến đổi tương tự về mặt khái niệm, đối với dữ liệu video dư. Dữ liệu dư có thể tương ứng với các giá trị chênh lệch điểm ảnh giữa các điểm ảnh của hình ảnh chưa được mã hoá và các giá trị dự báo tương ứng với các đơn vị PU. Bộ mã hoá dữ liệu video 20 có thể tạo ra các đơn vị TU chứa dữ liệu dư cho đơn vị CU, và sau đó biến đổi các đơn vị TU để tạo ra các hệ số biến đổi cho đơn vị CU.

Sau khi biến đổi để tạo ra các hệ số biến đổi, bộ mã hoá dữ liệu video 20 có thể thực hiện thao tác lượng tử hoá cho các hệ số biến đổi. Lượng tử hoá thường là quy trình trong đó các hệ số biến đổi được lượng tử hoá để có thể giảm bớt lượng dữ liệu dùng để biểu diễn các hệ số, nhằm đạt được hiệu quả nén cao hơn. Quy trình lượng tử hoá có thể giảm độ sâu bit liên quan đến một số hoặc tất cả các hệ số. Ví dụ, giá trị n -bit có thể được làm tròn xuống thành giá trị m -bit trong quy trình lượng tử hoá, trong đó n lớn hơn m .

Sau khi lượng tử hoá, bộ mã hoá dữ liệu video có thể quét các hệ số biến đổi, tạo ra vector một chiều từ ma trận hai chiều chứa các hệ số biến đổi đã lượng tử hoá. Quy trình quét này được thiết kế sao cho đặt các hệ số có năng lượng cao hơn (và do đó có tần số thấp hơn) ở phía trước trong ma trận và đặt các hệ số có năng lượng thấp hơn (và do đó có tần số cao hơn) ở phía sau trong ma trận. Trong một số ví dụ, bộ mã hoá dữ liệu video 20 có thể áp dụng thứ tự quét định trước để quét các hệ số biến đổi đã lượng tử hoá tạo ra vector tuyến tính có thể được mã hoá entropy. Theo các phương án khác làm ví dụ, bộ mã hoá dữ liệu video 20 có thể thực hiện kỹ thuật quét thích ứng. Sau khi quét các hệ số biến đổi đã lượng tử hoá để tạo ra vector một chiều, bộ mã hoá dữ liệu video 20 có thể mã hoá entropy vector một chiều, ví dụ, theo phương pháp mã hoá độ dài thay đổi thích ứng với ngữ cảnh (*CAVLC: Context-Adaptive Variable Length Coding*), mã hoá số học nhị phân thích ứng với ngữ cảnh (*CABAC: Context-Adaptive Binary Arithmetic Coding*), mã hoá số học nhị phân thích ứng với ngữ cảnh theo cú pháp (*SBAC: Syntax-Based Context-Adaptive Binary Arithmetic Coding*), mã

hoá entropy phân tách khoảng xác suất (*PIPE: Probability Interval Partitioning Entropy*) hoặc phương pháp mã hoá entropy khác. Bộ mã hoá dữ liệu video 20 cũng có thể mã hoá entropy cho các phần tử cú pháp liên quan đến dữ liệu video mã hoá để cho bộ giải mã dữ liệu video 30 sử dụng khi giải mã dữ liệu video.

Để thực hiện phương pháp CABAC, bộ mã hoá dữ liệu video 20 có thể phân định một ngữ cảnh trong mẫu ngữ cảnh cho ký hiệu cần truyền. Ngữ cảnh có thể liên quan đến việc, ví dụ, các giá trị liên kề của ký hiệu có khác không hay không. Để thực hiện phương pháp CAVLC, bộ mã hoá dữ liệu video 20 có thể chọn mã độ dài thay đổi cho ký hiệu cần truyền. Các từ mã trong phương pháp mã hoá độ dài thay đổi (*VLC: Variable Length Coding*) có thể được thiết lập sao cho các từ mã tương đối ngắn sẽ tương ứng với các ký hiệu có xác suất xuất hiện cao hơn, còn các từ mã dài sẽ tương ứng với các ký hiệu có xác suất xuất hiện thấp hơn. Theo cách này, việc áp dụng phương pháp VLC có thể tiết kiệm bit, ví dụ, sử dụng các từ mã có độ dài bằng nhau cho mỗi ký hiệu cần truyền. Việc xác định xác suất có thể dựa vào ngữ cảnh được phân định cho ký hiệu.

Theo các kỹ thuật nêu trong sáng chế, bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video 20 hoặc bộ giải mã dữ liệu video 30, có thể được tạo cấu hình để mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS. Bảng 2 và bảng 5, được mô tả chi tiết dưới đây, là ví dụ về các tập hợp phần tử cú pháp của bộ thông số VPS. Mỗi lớp trong số một hoặc nhiều lớp dữ liệu video có thể tham chiếu đến bộ thông số VPS, tức là, cùng một bộ thông số VPS. Nói cách khác, bộ thông số VPS có thể áp dụng cho tất cả các lớp của một tập hợp dữ liệu video chung, ví dụ, tất cả các lớp của dữ liệu video SVC và/hoặc tất cả các cảnh nhìn của dữ liệu video MVC.

Bộ thông số VPS có thể chứa nhiều loại thông tin khác nhau. Ví dụ, bộ thông số VPS có thể chứa bộ mô tả số đếm kích thước mẫu (*SDCD: Sample Dimension Counter Description*). Có nghĩa là, với mỗi kích thước, bộ mã hoá dữ liệu video có thể báo hiệu một tập hợp chỉ số. Các kích thước có thể có là *cnt_p*: số lượng lớp ưu tiên có trong chuỗi dữ liệu video mã hoá; *cnt_d*: số lượng lớp phụ thuộc khác nhau trong dòng bit, nhiều lớp có cùng một độ phân giải không gian và cùng một độ sâu bit có thể thuộc về các lớp phụ thuộc khác nhau; *cnt_t*: số lượng lớp thời gian trong dòng bit;

cnt_q: số lượng lớp chất lượng tối đa cho một lớp phụ thuộc bất kỳ trong dòng bit; và cnt_v: số lượng cảnh nhìn tối đa. Các thông số độ sâu bit có thể dài 8 bit hoặc 12 bit và có thể là khác nhau với các thành phần màu khác nhau. Các định dạng mẫu thành phần màu có thể là 4:0:0, 4:2:0 và 4:4:4.

Bộ thông số VPS cũng có thể chứa ánh xạ từ chỉ số mẫu đến đặc trưng. Nếu với mỗi kích thước, thông tin chỉ báo đặc trưng không bằng chỉ số nằm trong khoảng từ 0 đến số đếm kích thước mẫu trừ 1, thì vòng lặp có thể được đưa vào để xác định thông tin chỉ báo đặc trưng cho mỗi chỉ số đặc trưng. Ánh xạ này có thể có, với mỗi chỉ số lớp phụ thuộc, độ phân giải không gian cụ thể có giá trị độ sâu bit cụ thể và định dạng mẫu thành phần màu cụ thể. Lưu ý rằng, ánh xạ này có thể được bỏ qua, nếu luôn có bảng tra cứu cố định ở thiết bị giải mã, ví dụ, giá trị 0 có thể tương ứng với định dạng 4:2:0, giá trị 1 có thể tương ứng với định dạng 4:4:4, và giá trị 2 có thể tương ứng với định dạng 4:0:0. Theo cách khác hoặc theo cách bổ sung, ánh xạ này có thể có: với mỗi chỉ số thời gian, tốc độ khung hoặc tốc độ khung trung bình cụ thể; với mỗi chỉ số cảnh nhìn, id cảnh nhìn cụ thể; với mỗi chỉ số độ sâu bit, cặp giá trị độ sâu bit cụ thể cho thành phần độ chói và thành phần màu; và với mỗi định dạng mẫu thành phần màu, thông tin chỉ báo định dạng mẫu thành phần màu cụ thể.

Bộ thông số VPS cũng có thể chứa các thông số điều khiển và các cờ kích hoạt/bất hoạt công cụ, như các loại sau đây: thông số pcm_bit_depth_luma_minus1, thông số pcm_bit_depth_chroma_minus1, cờ loop_filter_across_slice_flag, cờ pcm_loop_filter_disable_flag, cờ temporal_id_nesting_flag, một hoặc nhiều phần tử cú pháp liên quan đến ô, cờ chroma_pred_from_luma_enabled_flag, cờ sample_adaptive_offset_enabled_flag, cờ adaptive_loop_filter_enabled_flag và cờ inter_4x4_enabled_flag.

Bộ thông số VPS cũng có thể chứa một hoặc nhiều bộ mô tả điểm hoạt động. Các điểm hoạt động thường mô tả một tập hợp con của tất cả cảnh nhìn của dữ liệu video có trong dòng bit. Điểm hoạt động có thể chứa một số lượng cảnh nhìn cụ thể gửi đến thiết bị đích để xuất ra, cũng như các cảnh nhìn khác có thể được dùng làm chuẩn khi giải mã, xuất ra, hoặc cả hai trường hợp đó. Dòng bit có thể có một hoặc nhiều điểm hoạt động được mô tả bằng các bộ mô tả điểm hoạt động. Bộ mô tả điểm hoạt động có thể chứa thông tin xác định số lượng điểm hoạt động tối đa, tính phụ

thuộc giữa các lớp hoặc các cảnh nhìn khác nhau, thông tin profin và mức cho mỗi điểm hoạt động, tỷ lệ bit cho mỗi điểm hoạt động, tính phụ thuộc giữa các điểm hoạt động cho mỗi điểm hoạt động, các quy định hạn chế khác, thông tin sử dụng dữ liệu video (VUI) hoặc một phần thông tin VUI cho mỗi điểm hoạt động, và/hoặc thông tin VUI hoặc một phần thông tin VUI cho mỗi lớp hoặc cảnh nhìn. Theo cách khác hoặc theo cách bổ sung, các bộ mô tả điểm hoạt động có thể chứa, với mỗi điểm hoạt động, dạng biểu diễn đơn vị lớp trừu tượng hoá mạng (NAL) có lớp mã hoá dữ liệu video (*VCL: Video Coding Layer*) chứa điểm hoạt động. Trong một số ví dụ, dạng biểu diễn đơn vị NAL có lớp VCL chứa điểm hoạt động, với mỗi kích thước, có thể có ba sự lựa chọn: (1) giá trị chỉ số cụ thể: ví dụ, với độ phân giải không gian, cho độ sâu bit của định dạng mẫu thành phần màu; (2) khoảng giá trị chỉ số: ví dụ, với các lớp thời gian, từ 0 đến ID của lớp thời gian cao nhất, với các lớp chất lượng, từ 0 đến ID của lớp chất lượng cao nhất; hoặc (3) danh mục giá trị chỉ số: ví dụ, với các cảnh nhìn, danh mục giá trị chỉ số cảnh nhìn.

Trong một số ví dụ, bộ thông số VPS có thể chứa thông tin chỉ báo số lượng lớp thời gian tối đa trên các lớp của dòng bit. Có nghĩa là, bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để mã hoá bộ thông số VPS chứa thông tin chỉ báo số lượng lớp thời gian tối đa cho dòng bit tương ứng. Ví dụ, bộ mã hoá dữ liệu video 20 có thể xác định số lượng lớp thời gian tối đa và mã hoá bộ thông số VPS sao cho có chứa dữ liệu biểu diễn số lượng lớp thời gian tối đa đã xác định, còn bộ giải mã dữ liệu video 30 có thể giải mã bộ thông số VPS để xác định số lượng lớp thời gian tối đa. Bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 cũng có thể mã hoá dữ liệu video của dòng bit dựa vào số lượng lớp thời gian tối đa đã xác định. Ví dụ, số lượng lớp thời gian tối đa có thể ảnh hưởng đến số lượng thông tin nhận dạng lớp thời gian cần dùng để biểu diễn các lớp thời gian khác nhau. Ví dụ khác, số lượng lớp thời gian tối đa có thể ảnh hưởng đến cách thức mà bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 mã hoá các thông tin nhận dạng hình ảnh chuẩn, ví dụ, sử dụng giá trị số thứ tự hình ảnh (*POC: Picture Order Count*).

Theo phương án khác nữa làm ví dụ, bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để mã hoá dữ liệu của một lớp thời gian cụ thể chỉ sử dụng dữ liệu chuẩn cho tới và bao gồm cả lớp thời gian đó. Nói cách khác,

bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để tránh mã hoá dữ liệu của một lớp thời gian cụ thể bằng cách sử dụng dữ liệu chuẩn của lớp thời gian cao hơn. Theo cách này, bộ giải mã dữ liệu video 30 có thể được đảm bảo là giải mã chính xác dữ liệu video của một tập hợp lớp thời gian cho trước ngay cả sau khi đã tách dòng bit con. Có nghĩa là, nếu việc tách dòng bit con được thực hiện, thì một số lớp thời gian ở cao hơn lớp cao nhất của dòng bit con đã tách sẽ không có sẵn để làm chuẩn. Do việc mã hoá dữ liệu của mỗi lớp thời gian chỉ dựa vào dữ liệu của các lớp ở lớp hiện thời hoặc thấp hơn lớp hiện thời, nên có thể tránh được sai sót vì dữ liệu ở một lớp cụ thể phụ thuộc vào dữ liệu ở lớp cao hơn, mà lớp cao hơn đó đã bị mất do việc tách dòng bit con.

Trong một số phương án làm ví dụ, theo cách khác hoặc theo cách bổ sung, bộ thông số VPS, chứa thông tin chỉ báo một hoặc cả hai số lượng hình ảnh cần sắp xếp lại trong một hoặc nhiều lớp của dòng bit và/hoặc số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã (DPB). Như đã nêu trên, thông tin đó có thể được gọi là thông tin giới hạn dòng bit. Vì vậy, thiết bị đích 14 có thể xác định các tính năng của bộ giải mã dữ liệu video 30 và sử dụng thông tin giới hạn dòng bit để xác định xem dòng bit tương ứng có thích hợp để được giải mã bằng bộ giải mã dữ liệu video 30 hay không, hoặc xác định xem thiết bị đích 14 có nên chọn nội dung khác hay không (ví dụ, từ nhà cung cấp nội dung qua mạng, nếu có nhiều phiên bản của nội dung đó).

Ngoài ra, bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể sử dụng thông tin giới hạn dòng bit khi mã hoá dữ liệu video. Ví dụ, bộ mã hoá dữ liệu video 20 có thể đảm bảo rằng thông tin giới hạn dòng bit không bị vi phạm. Có nghĩa là, giả sử thông tin giới hạn dòng bit chỉ báo rằng nhiều nhất là N hình ảnh được lưu trữ vào bộ nhớ đệm DPB, thì bộ mã hoá dữ liệu video 20 có thể đảm bảo rằng không có nhiều hơn N hình ảnh trong một tổ hợp bất kỳ của một hoặc nhiều danh mục hình ảnh chuẩn ở một thời điểm cho trước bất kỳ. Ví dụ khác, giả sử thông tin sắp xếp lại hình ảnh chỉ báo rằng một hình ảnh cần được dịch chuyển đi nhiều nhất là M hình ảnh, thì bộ mã hoá dữ liệu video 20 có thể đảm bảo rằng không có hình ảnh nào được dịch chuyển đi vượt quá M hình ảnh. Việc dịch chuyển hình ảnh theo cách này thường tương ứng với độ chênh lệch giữa thứ tự giải mã và thứ tự hiển thị của hình ảnh.

Tương tự như vậy, bộ giải mã dữ liệu video 30 có thể sử dụng thông tin này khi giải mã, ví dụ, để thực hiện việc quản lý bộ nhớ DPB, như xả bộ nhớ DPB. Bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 cũng có thể sử dụng thông tin giới hạn dòng bit, như số lượng hình ảnh tối đa sẽ được lưu trữ vào bộ nhớ đệm DPB và/hoặc số lượng hình ảnh cần sắp xếp lại, khi mã hoá giá trị của các thông tin nhận dạng hình ảnh chuẩn.

Trong một số phương án làm ví dụ, theo cách khác hoặc theo cách bổ sung, bộ thông số VPS chứa thông tin chỉ báo các thông số bộ giải mã chuẩn giả định (HRD). Các thông số HRD chứa, ví dụ, thông tin mô tả thời điểm khi mà dữ liệu được loại bỏ ra khỏi bộ nhớ đệm hình ảnh mã hoá (CPB: *Coded Picture Buffer*). Trong các bộ giải mã, như bộ giải mã dữ liệu video 30, bộ nhớ CPB là bộ nhớ đệm mà trong đó dữ liệu video mã hoá được lưu trữ cho tới khi dữ liệu này đã sẵn sàng để giải mã. Các bộ giải mã như bộ giải mã dữ liệu video 30 cũng có thể có bộ nhớ đệm hình ảnh giải mã (DPB), trong đó dữ liệu video đã giải mã được lưu trữ, ví dụ, được dùng làm dữ liệu chuẩn cho dữ liệu được dự báo liên cấu trúc và để sắp xếp lại các hình ảnh từ thứ tự giải mã chuyển thành thứ tự hiển thị.

Các thông số HRD có thể chứa thông tin chỉ báo thời điểm khi mà các hình ảnh cụ thể được loại bỏ ra khỏi bộ nhớ CPB và được giải mã. Do đó, bộ mã hoá dữ liệu video 20 có thể mã hoá các thông số HRD của bộ thông số VPS để chỉ báo thời điểm khi mà các hình ảnh có thể được loại bỏ ra khỏi bộ nhớ CPB và được giải mã, còn bộ giải mã dữ liệu video 30 có thể giải mã các thông số HRD của bộ thông số VPS để xác định thời điểm loại bỏ các hình ảnh ra khỏi bộ nhớ CPB. Tương tự như vậy, bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể mã hoá các hình ảnh theo các thông số HRD, ví dụ, theo thứ tự mã hoá được chỉ báo bằng các thông số HRD. Theo cách này, bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để mã hoá bộ thông số VPS chứa các thông số HRD, và để mã hoá dữ liệu video tương ứng với bộ thông số VPS ít nhất một phần dựa vào các thông số HRD.

Bộ thông số VPS cũng có thể chứa dữ liệu mở rộng chỉ báo việc bộ thông số VPS đã được mở rộng, ví dụ, để cung cấp dữ liệu cho một hoặc nhiều công cụ mã hoá khác. Các công cụ mã hoá như vậy có thể là các công cụ khác với các công cụ theo

chuẩn mã hoá dữ liệu video tương ứng, ví dụ như chuẩn ITU-T H.264/AVC hoặc chuẩn HEVC sắp tới. Ngoài ra, các công cụ mã hoá như vậy có thể cần có dữ liệu cấu hình. Dữ liệu cấu hình này có thể được cung cấp trong dữ liệu mở rộng của bộ thông số VPS. Theo cách này, khi mã hoá dữ liệu video sử dụng các công cụ mã hoá như vậy, bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể mã hoá bộ thông số VPS chỉ báo việc dữ liệu mở rộng có mặt hay không, và nếu có mặt, thì chỉ báo dữ liệu mở rộng của bộ thông số VPS. Ngoài ra, khi dữ liệu mở rộng đó có mặt, thì bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể thực hiện các công cụ mã hoá tương ứng để mã hoá dữ liệu video sử dụng dữ liệu mở rộng.

Các chuẩn mã hoá dữ liệu video xác định cú pháp tương ứng, ngữ nghĩa, và quy trình giải mã cho các dòng bit không có lỗi, tuân theo một profin hoặc mức nhất định. Các chuẩn mã hoá dữ liệu video thường không xác định bộ mã hoá, mà bộ mã hoá có nhiệm vụ phải đảm bảo sao cho các dòng bit đã tạo ra tương thích với chuẩn áp dụng cho bộ giải mã. Trong trường hợp các chuẩn mã hoá dữ liệu video, “profin” tương ứng với tập hợp con của các thuật toán, dấu hiệu, hoặc công cụ và các quy định hạn chế áp dụng cho các chuẩn đó. Như được xác định theo chuẩn H.264, ví dụ, “profin” là một tập hợp con của toàn bộ cú pháp dòng bit được xác định theo chuẩn H.264. “Mức” tương ứng với các quy định hạn chế về mức sử dụng tài nguyên của bộ giải mã, ví dụ như dung lượng bộ nhớ và công suất tính toán của bộ giải mã, liên quan đến độ phân giải của các hình ảnh, tỷ lệ bit, và tốc độ xử lý khối. Profin có thể được báo hiệu bằng giá trị `profin_idc` (thông tin chỉ báo profin), còn mức có thể được báo hiệu bằng giá trị `level_idc` (thông tin chỉ báo mức). Theo các kỹ thuật nêu trong sáng chế, thông tin profin và mức có thể được xác định trong các bộ mô tả điểm hoạt động, như đã nêu trên.

Trong một số ví dụ, mỗi lớp hoặc cảnh nhìn trong dòng bit tham chiếu đến một bộ thông số video (VPS), và bộ thông số chuỗi phân lớp (*LPS: Layered sequence Parameter Set*) có thể được kích hoạt cho mỗi lớp. Bộ thông số LPS có thể được thiết kế sao cho càng nhẹ càng tốt khi tham chiếu đến bộ thông số VPS trong cấu trúc thiết kế. Bộ thông số LPS có thể chứa một thông tin bất kỳ hoặc tất cả các thông tin được mô tả dưới đây. Bộ thông số LPS có thể chứa thông tin chỉ báo kích thước mẫu để chỉ báo, với mỗi kích thước, chỉ số cho mỗi kích thước. Ví dụ, nếu trong bộ thông số

VPS, chỉ số chỉ đến độ phân giải không gian 0 được phân định cho đặc trưng không gian có độ phân giải bằng 320x240 và chỉ số chỉ đến độ phân giải không gian 1 được phân định cho đặc trưng không gian có độ phân giải bằng 640x480, và lớp hiện thời được phân định có độ phân giải bằng 640x480, thì bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể mã hoá phần tử cú pháp có giá trị bằng 1 cho lớp hiện thời. Có nghĩa là, bộ mã hoá dữ liệu video 20 có thể báo hiệu giá trị 1 cho phần tử cú pháp để xác định độ phân giải bằng 640x480, còn bộ giải mã dữ liệu video 30 có thể xác định rằng lớp hiện thời có phần tử cú pháp có giá trị bằng 1 với độ phân giải bằng 640x480, dựa vào giá trị bằng 1 của phần tử cú pháp.

Bộ thông số LPS cũng có thể chứa các thông số điều khiển và các cờ kích hoạt/bất hoạt công cụ. Ví dụ, thông số điều khiển và cờ kích hoạt/bất hoạt công cụ có thể là thông số `pcm_bit_depth_luma_minus1`, thông số `pcm_bit_depth_luma_minus1`, cờ `loop_filter_across_slice_flag`, cờ `pcm_loop_filter_disable_flag`, một hoặc nhiều phần tử cú pháp liên quan đến ô, cờ `chroma_pred_from_luma_enabled_flag`, cờ `sample_adaptive_offset_enabled_flag`, cờ `adaptive_loop_filter_enabled_flag`, và cấu trúc phân cấp của đơn vị mã hoá (CU).

Bộ thông số LPS có thể còn chứa thông tin về các loại bộ thông số khác áp dụng cho lát, nhóm lát, hình ảnh, hoặc vài hình ảnh. Mỗi bộ thông số đó có thể chỉ đến một bộ thông số hình ảnh (PPS) cụ thể.

Bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30, có thể được tạo cấu hình để đảm bảo và/hoặc xác định rằng bộ thông số PPS không chỉ đến bộ thông số LPS hoặc bộ thông số VPS. Do đó, bộ mã hoá dữ liệu video có thể đảm bảo rằng mỗi bộ thông số PPS của dòng bit không chỉ đến bộ thông số LPS hoặc bộ thông số VPS. Việc phân tích cú pháp cho bộ thông số PPS có thể là độc lập. Khi bộ thông số PPS có một hoặc nhiều phần tử cú pháp giống như các phần tử cú pháp của bộ thông số VPS hoặc bộ thông số LPS, thì các phần tử cú pháp của bộ thông số PPS có thể ghi đè lên các phần tử cú pháp của bộ thông số VPS hoặc bộ thông số LPS.

Bộ mã hoá dữ liệu video có thể còn được tạo cấu hình để mã hoá bộ thông số nhóm (GPS) để nhóm tất cả các bộ thông số với nhau. Bộ mã hoá dữ liệu video có thể mã hoá nhiều nhóm khác nhau trong một bộ thông số GPS, mỗi bộ thông số GPS có

thông tin nhận dạng GPS (*GPS id: GPS identifier*) riêng biệt. Mỗi nhóm trong bộ thông số GPS có thể là một tổ hợp khác nhau của các bộ thông số. Theo cách này, phần đầu lát chỉ cần có thông tin tham chiếu đến GPS id tương ứng, và không cần có thông tin chỉ báo loại bộ thông số. Đơn yêu cầu cấp patent Mỹ số 61/590,702, nộp ngày 25.01.2012, mô tả chi tiết các kỹ thuật trong đó các loại bộ thông số khác nhau được nhóm với nhau và chỉ có ID của cấu trúc cú pháp tải hữu ích chứa chuỗi byte thô (*RBSP: Raw Byte Sequence Payload*) cho nhóm bộ thông số được báo hiệu trong phần đầu lát.

Như đã nêu trên, bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video 20 hoặc bộ giải mã dữ liệu video 30, có thể được tạo cấu hình để mã hoá bộ thông số video và/hoặc bộ thông số nhóm. Ví dụ về bộ thông số video được mô tả chi tiết dưới đây dựa vào Fig.5, còn ví dụ về bộ thông số nhóm được mô tả chi tiết dưới đây dựa vào Fig.6.

Bộ mã hoá dữ liệu video 20 có thể còn truyền dữ liệu cú pháp, như dữ liệu cú pháp theo khối, dữ liệu cú pháp theo khung, và dữ liệu cú pháp theo nhóm GOP, đến bộ giải mã dữ liệu video 30, ví dụ, trong phần đầu khung, phần đầu khối, phần đầu lát, hoặc phần đầu nhóm GOP. Dữ liệu cú pháp theo nhóm GOP có thể mô tả số lượng khung trong nhóm GOP tương ứng, và dữ liệu cú pháp theo khung có thể chỉ báo chế độ mã hoá/dự báo được dùng để mã hoá khung tương ứng.

Mỗi bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể được thực hiện dưới dạng mạch mã hoá bất kỳ trong số rất nhiều mạch mã hoá phù hợp, như một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (*DSP: Digital Signal Processor*), mạch tích hợp chuyên dụng (*ASIC: Application Specific Integrated Circuit*), mảng cửa lập trình được bằng trường (*FPGA: Field Programmable Gate Array*), mạch logic rời rạc, phần mềm, phần cứng, phần sụn hoặc mọi dạng kết hợp của các loại trên. Mỗi bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể nằm trong một hoặc nhiều bộ mã hoá hoặc bộ giải mã, mỗi bộ phận này có thể được tích hợp làm một phần của bộ mã hoá/giải mã kết hợp (*CODEC*) trong thiết bị tương ứng. Thiết bị có bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể là mạch tích hợp, bộ vi xử lý, và/hoặc thiết bị truyền thông không dây, như máy điện thoại di động.

Fig.2 là sơ đồ khối thể hiện ví dụ về bộ mã hoá dữ liệu video 20 có thể áp dụng

kỹ thuật mã hoá các bộ thông số và các đơn vị NAL cho một hoặc nhiều lớp dữ liệu video. Bộ mã hoá dữ liệu video 20 có thể thực hiện kỹ thuật mã hoá dự báo nội cấu trúc và mã hoá dự báo liên cấu trúc cho các khối video trong các lát video. Mã hoá dự báo nội cấu trúc dựa vào kỹ thuật dự báo không gian để giảm bớt hoặc loại bỏ dữ liệu dư theo không gian trong dữ liệu video của khung video hoặc hình ảnh cho trước. Mã hoá dự báo liên cấu trúc dựa vào kỹ thuật dự báo thời gian để giảm bớt hoặc loại bỏ dữ liệu dư theo thời gian trong dữ liệu video của các khung hoặc hình ảnh liên kế trong chuỗi dữ liệu video. Chế độ dự báo nội cấu trúc (chế độ I) có thể dùng để chỉ một chế độ bất kỳ trong một số chế độ mã hoá theo không gian. Chế độ dự báo liên cấu trúc, như chế độ dự báo một chiều (chế độ P) hoặc chế độ dự báo hai chiều (chế độ B) có thể dùng để chỉ một chế độ bất kỳ trong một số chế độ mã hoá theo thời gian.

Như được thể hiện trên Fig.2, bộ mã hoá dữ liệu video 20 thu khối video hiện thời trong khung video cần mã hoá. Trong ví dụ được thể hiện trên Fig.2, bộ mã hoá dữ liệu video 20 bao gồm bộ chọn chế độ 40, bộ nhớ hình ảnh chuẩn 64, bộ cộng 50, bộ xử lý biến đổi 52, bộ lượng tử hoá 54 và bộ mã hoá entropy 56. Bộ chọn chế độ 40 lại bao gồm bộ bù chuyển động 44, bộ đánh giá chuyển động 42, bộ dự báo nội cấu trúc 46 và bộ phân tách 48. Để khôi phục khối video, bộ mã hoá dữ liệu video 20 còn bao gồm bộ lượng tử hoá ngược 58, bộ biến đổi ngược 60 và bộ cộng 62. Bộ lọc tách khối (không được thể hiện trên Fig.2) cũng có thể được đưa vào để lọc các ranh giới khối nhằm loại bỏ các thành phần lạ dạng khối ra khỏi dữ liệu video đã khôi phục. Nếu muốn, bộ lọc tách khối thường lọc tín hiệu đầu ra của bộ cộng 62. Các bộ lọc vòng lặp khác (trong vòng lặp hoặc sau vòng lặp) cũng có thể được sử dụng cùng với bộ lọc tách khối. Vì để cho dễ hiểu nên các bộ lọc đó không được thể hiện trên hình vẽ, nhưng nếu muốn, thì các bộ lọc đó có thể lọc tín hiệu đầu ra của bộ cộng 50 (dưới dạng bộ lọc trong vòng lặp).

Trong quy trình mã hoá, bộ mã hoá dữ liệu video 20 thu khung hoặc lát video cần mã hoá. Khung hoặc lát này có thể được phân tách ra thành nhiều khối video. Bộ đánh giá chuyển động 42 và bộ bù chuyển động 44 thực hiện bước mã hoá dự báo liên cấu trúc cho khối video thu được dựa vào một hoặc nhiều khối trong một hoặc nhiều khung chuẩn để thực hiện bước dự báo theo thời gian. Theo cách khác, bộ dự báo nội cấu trúc 46 có thể thực hiện bước mã hoá dự báo nội cấu trúc cho khối video thu được

dựa vào một hoặc nhiều khối liền kề trong cùng một khung hoặc lát với khối cần mã hoá để thực hiện bước dự báo theo không gian. Bộ mã hoá dữ liệu video 20 có thể thực hiện nhiều bước mã hoá, ví dụ, để chọn chế độ mã hoá thích hợp cho mỗi khối dữ liệu video.

Ngoài ra, bộ phân tách 48 có thể phân tách các khối dữ liệu video ra thành các khối con, dựa vào sự đánh giá các sơ đồ phân tách trước đó trong các bước mã hoá trước. Ví dụ, ban đầu bộ phân tách 48 có thể phân tách khung hoặc lát ra thành các đơn vị LCU, và phân tách mỗi đơn vị LCU ra thành các đơn vị CU con dựa vào phương pháp phân tích tốc độ mã hoá-méo (ví dụ, tối ưu hoá tốc độ mã hoá-méo). Bộ chọn chế độ 40 có thể còn tạo ra cấu trúc dữ liệu cây tứ phân chỉ báo cách phân tách đơn vị LCU ra thành các đơn vị CU con. Các đơn vị CU nút lá của cây tứ phân có thể có một hoặc nhiều đơn vị PU và một hoặc nhiều đơn vị TU.

Bộ chọn chế độ 40 có thể chọn một trong số các chế độ mã hoá, mã hoá dự báo nội cấu trúc hoặc mã hoá dự báo liên cấu trúc, ví dụ, dựa vào kết quả sai số, và cung cấp khối đã được mã hoá dự báo nội cấu trúc hoặc mã hoá dự báo liên cấu trúc thu được cho bộ cộng 50 để tạo ra khối dữ liệu dư và cung cấp cho bộ cộng 62 để khôi phục khối mã hoá dùng làm khung chuẩn. Bộ chọn chế độ 40 còn cung cấp các phần tử cú pháp, như vector chuyển động, thông tin chỉ báo chế độ dự báo nội cấu trúc, thông tin phân tách, và các thông tin cú pháp khác, cho bộ mã hoá entropy 56.

Bộ đánh giá chuyển động 42 và bộ bù chuyển động 44 có thể được tích hợp chức năng với nhau, nhưng vì để giúp cho người đọc thấy dễ hiểu nên các bộ phận này được thể hiện riêng biệt trên hình vẽ. Quy trình đánh giá chuyển động, được thực hiện bằng bộ đánh giá chuyển động 42, là quy trình tạo ra các vector chuyển động, để đánh giá chuyển động cho các khối video. Vector chuyển động, ví dụ, có thể chỉ báo sự dịch chuyển của đơn vị PU ở khối video trong khung video hoặc hình ảnh video hiện thời so với khối dự báo trong khung chuẩn (hoặc đơn vị mã hoá khác) so với khối hiện thời đang được mã hoá trong khung hiện thời (hoặc đơn vị mã hoá khác). Khối dự báo là khối được nhận thấy là rất phù hợp với đơn vị PU của khối video cần mã hoá xét về giá trị chênh lệch điểm ảnh, giá trị này có thể được xác định bằng tổng hiệu số tuyệt đối (*SAD: Sum of Absolute Difference*), tổng hiệu số bình phương (*SSD: Sum of Square Difference*), hoặc các giá trị đo hiệu số khác. Trong một số ví dụ, bộ mã hoá

dữ liệu video 20 có thể tính giá trị cho các vị trí điểm ảnh dưới số nguyên của các hình ảnh chuẩn lưu trữ trong bộ nhớ hình ảnh chuẩn 64. Ví dụ, bộ mã hoá dữ liệu video 20 có thể nội suy giá trị của các vị trí phân tử điểm ảnh, các vị trí phân tám điểm ảnh, hoặc các vị trí phân số điểm ảnh khác của hình ảnh chuẩn. Vì vậy, bộ đánh giá chuyển động 42 có thể thực hiện chức năng tìm kiếm chuyển động dựa vào các vị trí số nguyên điểm ảnh và các vị trí phân số điểm ảnh, và xuất ra vector chuyển động với độ chính xác bằng phân số điểm ảnh.

Bộ đánh giá chuyển động 42 tính vector chuyển động cho đơn vị PU của khối video trong lát được mã hoá dự báo liên cấu trúc bằng cách so sánh vị trí của đơn vị PU với vị trí của khối dự báo trong hình ảnh chuẩn. Hình ảnh chuẩn có thể được chọn từ danh mục hình ảnh chuẩn thứ nhất (danh mục 0) hoặc danh mục hình ảnh chuẩn thứ hai (danh mục 1), mỗi danh mục hình ảnh chuẩn xác định một hoặc nhiều hình ảnh chuẩn lưu trữ trong bộ nhớ hình ảnh chuẩn 64. Bộ đánh giá chuyển động 42 truyền vector chuyển động tính được đến bộ mã hoá entropy 56 và bộ bù chuyển động 44.

Quy trình bù chuyển động, được thực hiện bằng bộ bù chuyển động 44, có thể là quy trình tìm hoặc tạo ra khối dự báo dựa vào vector chuyển động được xác định ở bộ đánh giá chuyển động 42. Xin nhắc lại, bộ đánh giá chuyển động 42 và bộ bù chuyển động 44 có thể được tích hợp chức năng với nhau, trong một số ví dụ. Khi thu được vector chuyển động cho đơn vị PU của khối video hiện thời, bộ bù chuyển động 44 có thể tìm khối dự báo mà vector chuyển động tham chiếu đến đó ở một trong số các danh mục hình ảnh chuẩn. Bộ cộng 50 tạo ra khối dữ liệu video dư bằng cách lấy giá trị điểm ảnh của khối video hiện thời đang được mã hoá trừ đi giá trị điểm ảnh của khối dự báo, thu được các giá trị chênh lệch điểm ảnh, như được mô tả dưới đây. Nói chung, bộ đánh giá chuyển động 42 thực hiện chức năng đánh giá chuyển động liên quan đến các thành phần độ chói, và bộ bù chuyển động 44 sử dụng các vector chuyển động tính được dựa trên các thành phần độ chói cho cả thành phần màu lẫn thành phần độ chói. Bộ chọn chế độ 40 cũng có thể tạo ra các phần tử cú pháp liên hệ với các khối video và lát video để cho bộ giải mã dữ liệu video 30 sử dụng khi giải mã các khối video trong lát video.

Bộ dự báo nội cấu trúc 46 có thể dự báo nội cấu trúc cho khối hiện thời, theo một phương án thay thế khác với phương án dự báo liên cấu trúc được thực hiện bằng

bộ đánh giá chuyển động 42 và bộ bù chuyển động 44, như đã nêu trên. Cụ thể là, bộ dự báo nội cấu trúc 46 có thể xác định chế độ dự báo nội cấu trúc cần dùng để mã hoá khối hiện thời. Trong một số ví dụ, bộ dự báo nội cấu trúc 46 có thể mã hoá khối hiện thời bằng cách sử dụng các chế độ dự báo nội cấu trúc khác nhau, ví dụ, trong những lần mã hoá khác nhau, và bộ dự báo nội cấu trúc 46 (hoặc bộ chọn chế độ 40, trong một số ví dụ) có thể chọn một chế độ dự báo nội cấu trúc thích hợp để sử dụng trong số các chế độ đã được thử nghiệm.

Ví dụ, bộ dự báo nội cấu trúc 46 có thể tính giá trị tốc độ mã hoá-méo bằng cách sử dụng phép phân tích tốc độ mã hoá-méo cho các chế độ dự báo nội cấu trúc được thử nghiệm, và chọn chế độ dự báo nội cấu trúc có đặc trưng tốc độ mã hoá-méo phù hợp nhất trong số các chế độ được thử nghiệm. Phép phân tích tốc độ mã hoá-méo thường xác định độ méo (hoặc sai số) giữa khối mã hoá và khối ban đầu chưa mã hoá, khối ban đầu này được mã hoá để tạo ra khối mã hoá, cũng như tỷ lệ bit (nghĩa là, số bit) dùng để tạo ra khối mã hoá. Bộ dự báo nội cấu trúc 46 có thể tính tỷ lệ từ các độ méo và các tốc độ mã hoá cho các khối mã hoá khác nhau để xác định chế độ dự báo nội cấu trúc có giá trị tốc độ mã hoá-méo phù hợp nhất cho khối.

Sau khi chọn chế độ dự báo nội cấu trúc cho khối, bộ dự báo nội cấu trúc 46 có thể cung cấp cho bộ mã hoá entropy 56 thông tin chỉ báo chế độ dự báo nội cấu trúc được chọn cho khối. Bộ mã hoá entropy 56 có thể mã hoá thông tin chỉ báo chế độ dự báo nội cấu trúc được chọn theo các kỹ thuật mã hoá entropy nêu trong sáng chế. Bộ mã hoá dữ liệu video 20 có thể đưa dữ liệu cấu hình vào trong dòng bit được truyền, dữ liệu cấu hình này có thể có nhiều bảng chỉ số chế độ dự báo nội cấu trúc và nhiều bảng chỉ số chế độ dự báo nội cấu trúc sửa đổi (còn gọi là các bảng ánh xạ từ mã), các định nghĩa ngữ cảnh mã hoá cho các khối khác nhau, và thông tin chỉ báo chế độ dự báo nội cấu trúc, bảng chỉ số chế độ dự báo nội cấu trúc và bảng chỉ số chế độ dự báo nội cấu trúc sửa đổi phù hợp nhất sẽ được sử dụng cho mỗi ngữ cảnh.

Bộ mã hoá dữ liệu video 20 tạo ra khối dữ liệu video dư bằng cách lấy khối video ban đầu đang được mã hoá trừ đi dữ liệu dự báo từ bộ chọn chế độ 40. Bộ cộng 50 có một hoặc nhiều bộ phận để thực hiện phép tính trừ này. Bộ xử lý biến đổi 52 áp dụng một quy trình biến đổi, như biến đổi cosin rời rạc (DCT) hoặc biến đổi tương tự về mặt khái niệm, cho khối dữ liệu dư, để tạo ra khối video có các giá trị hệ số biến

đôi dư. Bộ xử lý biến đổi 52 có thể thực hiện các quy trình biến đổi khác tương tự về mặt khái niệm với DCT. Biến đổi dạng sóng, biến đổi số nguyên, biến đổi dải con hoặc các loại biến đổi khác cũng có thể được sử dụng.

Trong mọi trường hợp, bộ xử lý biến đổi 52 áp dụng quy trình biến đổi cho khối dữ liệu dư, tạo ra khối hệ số biến đổi dư. Quy trình biến đổi này có thể chuyển dữ liệu dư từ miền giá trị điểm ảnh sang miền biến đổi, như miền tần số. Bộ xử lý biến đổi 52 có thể truyền các hệ số biến đổi thu được đến bộ lượng tử hoá 54. Bộ lượng tử hoá 54 lượng tử hoá các hệ số biến đổi để tiếp tục giảm tỷ lệ bit. Quy trình lượng tử hoá có thể giảm độ sâu bit liên quan đến một số hoặc tất cả các hệ số. Mức lượng tử hoá có thể được sửa đổi bằng cách điều chỉnh thông số lượng tử hoá. Trong một số ví dụ, bộ lượng tử hoá 54 có thể thực hiện thao tác quét ma trận hệ số biến đổi đã lượng tử hoá. Theo cách khác, bộ mã hoá entropy 56 có thể thực hiện thao tác quét này.

Sau khi lượng tử hoá, bộ mã hoá entropy 56 mã hoá entropy cho các hệ số biến đổi đã lượng tử hoá. Ví dụ, bộ mã hoá entropy 56 có thể thực hiện phương pháp mã hoá độ dài thay đổi thích ứng với ngữ cảnh (CAVLC), mã hoá số học nhị phân thích ứng với ngữ cảnh (CABAC), mã hoá số học nhị phân thích ứng với ngữ cảnh theo cú pháp (SBAC), mã hoá entropy phân tách khoảng xác suất (PIPE) hoặc phương pháp mã hoá entropy khác. Trong trường hợp mã hoá entropy thích ứng với ngữ cảnh, ngữ cảnh có thể dựa vào các khối liền kề. Sau khi mã hoá entropy bằng bộ mã hoá entropy 56, dòng bit mã hoá có thể được truyền đến thiết bị khác (ví dụ, bộ giải mã dữ liệu video 30) hoặc được lưu trữ để sau này sẽ được truyền hoặc tìm kiếm.

Bộ lượng tử hoá ngược 58 và bộ biến đổi ngược 60 lần lượt áp dụng quy trình lượng tử hoá ngược và biến đổi ngược để khôi phục khối dữ liệu dư ở miền điểm ảnh, ví dụ, để sau này dùng làm khối chuẩn. Bộ bù chuyển động 44 có thể tính khối chuẩn bằng cách cộng khối dữ liệu dư với khối dự báo của một trong số các khung trong bộ nhớ hình ảnh chuẩn 64. Bộ bù chuyển động 44 cũng có thể áp dụng một hoặc nhiều bộ lọc nội suy cho khối dữ liệu dư đã được khôi phục để tính các giá trị điểm ảnh dưới số nguyên để sử dụng khi đánh giá chuyển động. Bộ cộng 62 cộng khối dữ liệu dư đã được khôi phục với khối dự báo đã bù chuyển động được tạo ra bằng bộ bù chuyển động 44 sẽ thu được khối video đã được khôi phục để lưu trữ vào bộ nhớ hình ảnh chuẩn 64. Khối video đã khôi phục có thể được bộ đánh giá chuyển động 42 và bộ bù

chuyển động 44 dùng làm khối chuẩn để mã hoá dự báo liên cấu trúc cho khối trong khung video sau đó.

Bộ mã hoá dữ liệu video 20 có thể còn được tạo cấu hình để mã hoá bộ thông số video (VPS), bộ thông số lớp (LPS) và/hoặc bộ thông số nhóm, theo các kỹ thuật nêu trong sáng chế, cũng như bộ thông số chuỗi (SPS), bộ thông số hình ảnh (PPS), bộ thông số thích ứng (*APS: Adaptation Parameter Set*), hoặc các cấu trúc dữ liệu báo hiệu khác. Cụ thể hơn, bộ mã hoá entropy 56 có thể được tạo cấu hình để mã hoá một cấu trúc dữ liệu bất kỳ hoặc tất cả các cấu trúc dữ liệu. Các thông số của các cấu trúc dữ liệu này có thể ảnh hưởng đến hiệu suất mã hoá ở một mức độ nào đó, bộ chọn chế độ 40 có thể chọn các thông số thích hợp và chuyển các thông số đó đến bộ mã hoá entropy 56 để đưa vào trong, ví dụ, bộ thông số VPS. Các thông số khác, như số lượng lớp thời gian, số lượng hình ảnh cần sắp xếp lại và số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã, có thể được người dùng, ví dụ, người quản lý, lựa chọn. Theo các phương án khác làm ví dụ, một số thông số, như các thông số HRD, có thể phát sinh trong quy trình mã hoá.

Bộ mã hoá entropy 56 có thể mã hoá bộ thông số VPS sao cho có chứa một hoặc tất cả các loại dữ liệu nêu trong sáng chế. Bộ mã hoá dữ liệu video 20 cũng có thể mã hoá dữ liệu theo các thông số của bộ thông số VPS. Cụ thể hơn, bộ mã hoá dữ liệu video 20 có thể mã hoá các chuỗi hình ảnh trên một hoặc nhiều lớp dữ liệu video tương ứng với bộ thông số VPS theo các thông số của bộ thông số VPS.

Theo cách này, bộ mã hoá dữ liệu video 20 trên Fig.2 là ví dụ về bộ mã hoá dữ liệu video được tạo cấu hình để mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

Mặc dù thường được mô tả dựa vào bộ mã hoá dữ liệu video, nhưng quy trình mã hoá bộ thông số VPS có thể được thực hiện bằng các thiết bị khác, ví dụ, phần tử mạng nhận biết phương tiện (*MANE: Media-Aware Network Element*). MANE có thể tương ứng với phần tử mạng giữa thiết bị nguồn (như thiết bị nguồn 12 trên Fig.1) và thiết bị đích (như thiết bị đích 14). MANE có thể được tạo cấu hình để mã hoá bộ thông số VPS theo các kỹ thuật nêu trong sáng chế. MANE có thể tạo ra bộ thông số

VPS sử dụng dữ liệu của các cấu trúc dữ liệu khác thu được bằng MANE, ví dụ, các bộ thông số chuỗi.

Fig.3 là sơ đồ khối thể hiện ví dụ về bộ giải mã dữ liệu video 30 có thể áp dụng kỹ thuật mã hoá các bộ thông số và các đơn vị NAL cho một hoặc nhiều lớp dữ liệu video. Trong ví dụ được thể hiện trên Fig.3, bộ giải mã dữ liệu video 30 bao gồm bộ giải mã entropy 70, bộ bù chuyển động 72, bộ dự báo nội cấu trúc 74, bộ lượng tử hoá ngược 76, bộ biến đổi ngược 78, bộ nhớ hình ảnh chuẩn 82 và bộ cộng 80. Bộ nhớ hình ảnh chuẩn 82 cũng có thể được gọi là “bộ nhớ đệm hình ảnh giải mã”, hoặc bộ nhớ DPB. Bộ giải mã dữ liệu video 30 có thể, trong một số ví dụ, thực hiện quy trình giải mã thường ngược với quy trình mã hoá đã được mô tả trên đây liên quan đến bộ mã hoá dữ liệu video 20 (Fig.2). Bộ bù chuyển động 72 có thể tạo ra dữ liệu dự báo dựa vào các vectơ chuyển động thu được từ bộ giải mã entropy 70, còn bộ dự báo nội cấu trúc 74 có thể tạo ra dữ liệu dự báo dựa vào thông tin chỉ báo chế độ dự báo nội cấu trúc thu được từ bộ giải mã entropy 70.

Trong quy trình giải mã, bộ giải mã dữ liệu video 30 thu dòng bit video mã hoá có các khối video của lát video mã hoá và các phần tử cú pháp liên quan từ bộ mã hoá dữ liệu video 20. Bộ giải mã entropy 70 của bộ giải mã dữ liệu video 30 giải mã entropy cho dòng bit để tạo ra các hệ số đã lượng tử hoá, các vectơ chuyển động hoặc thông tin chỉ báo chế độ dự báo nội cấu trúc, và các phần tử cú pháp khác. Bộ giải mã entropy 70 truyền các vectơ chuyển động đến bộ lượng tử hoá ngược 76 và truyền các phần tử cú pháp khác đến bộ bù chuyển động 72. Bộ giải mã dữ liệu video 30 có thể thu các phần tử cú pháp ở mức lát video và/hoặc mức khối video.

Khi lát video được mã hoá là lát mã hoá dự báo nội cấu trúc (I), thì bộ dự báo nội cấu trúc 74 có thể tạo ra dữ liệu dự báo cho khối video của lát video hiện thời dựa vào chế độ dự báo nội cấu trúc đã được báo hiệu và dữ liệu từ các khối đã giải mã trước đó của khung hoặc hình ảnh hiện thời. Khi lát video được mã hoá là lát mã hoá dự báo liên cấu trúc (tức là, B, P hoặc GPB), thì bộ bù chuyển động 72 tạo ra các khối dự báo cho khối video của lát video hiện thời dựa vào các vectơ chuyển động và các phần tử cú pháp khác thu được từ bộ giải mã entropy 70. Khối dự báo có thể được tạo ra từ một trong số các hình ảnh chuẩn ở một trong số các danh mục hình ảnh chuẩn. Bộ giải mã dữ liệu video 30 có thể thiết lập các danh mục khung chuẩn, danh mục 0

và danh mục 1, sử dụng kỹ thuật thiết lập ngầm định dựa vào các hình ảnh chuẩn lưu trữ trong bộ nhớ hình ảnh chuẩn 82.

Bộ bù chuyển động 72 xác định thông tin dự báo cho khối video của lát video hiện thời bằng cách phân tích cú pháp cho các vector chuyển động và các phần tử cú pháp khác, và sử dụng thông tin dự báo để tạo ra các khối dự báo cho khối video hiện thời đang được giải mã. Ví dụ, bộ bù chuyển động 72 sử dụng một số phần tử cú pháp thu được để xác định chế độ dự báo (ví dụ, dự báo nội cấu trúc hoặc dự báo liên cấu trúc) được dùng để mã hoá các khối video trong lát video, loại lát dự báo liên cấu trúc (ví dụ, lát B, lát P, hoặc lát GPB), thông tin thiết lập của một hoặc nhiều danh mục hình ảnh chuẩn cho lát, các vector chuyển động cho mỗi khối video mã hoá dự báo liên cấu trúc trong lát, trạng thái dự báo liên cấu trúc cho mỗi khối video mã hoá dự báo liên cấu trúc trong lát, và thông tin khác để giải mã các khối video trong lát video hiện thời.

Bộ bù chuyển động 72 cũng có thể thực hiện phép nội suy dựa vào các bộ lọc nội suy. Bộ bù chuyển động 72 có thể sử dụng các bộ lọc nội suy giống như các bộ lọc nội suy mà bộ mã hoá dữ liệu video 20 đã sử dụng khi mã hoá các khối video để tính các giá trị nội suy cho các điểm ảnh dưới số nguyên của các khối chuẩn. Trong trường hợp này, bộ bù chuyển động 72 có thể xác định các bộ lọc nội suy dùng cho bộ mã hoá dữ liệu video 20 từ các phần tử cú pháp thu được và sử dụng các bộ lọc nội suy để tạo ra các khối dự báo.

Bộ lượng tử hoá ngược 76 lượng tử hoá ngược, tức là, khử lượng tử hoá, các hệ số biến đổi đã lượng tử hoá được cung cấp trong dòng bit và được giải mã bằng bộ giải mã entropy 80. Quy trình lượng tử hoá ngược có thể bao gồm bước sử dụng thông số lượng tử hoá QPY được tính bằng bộ giải mã dữ liệu video 30 cho mỗi khối video trong lát video để xác định mức lượng tử hoá và, tương tự như vậy, mức lượng tử hoá ngược sẽ được áp dụng. Bộ biến đổi ngược 78 áp dụng quy trình lượng tử hoá ngược, ví dụ, biến đổi DCT ngược, biến đổi số nguyên ngược, hoặc biến đổi ngược tương tự về mặt khái niệm, cho các hệ số biến đổi để tạo ra các khối dữ liệu dư ở miền điểm ảnh.

Sau khi bộ bù chuyển động 72 tạo ra khối dự báo cho khối video hiện thời dựa vào các vector chuyển động và các phần tử cú pháp khác, bộ giải mã dữ liệu video 30

tạo ra các khối video đã giải mã bằng cách cộng các khối dữ liệu dư từ bộ biến đổi ngược 78 với các khối dự báo tương ứng được tạo ra bằng bộ bù chuyển động 72. Bộ cộng 90 có một hoặc nhiều bộ phận để thực hiện phép tính cộng này. Nếu muốn, bộ lọc tách khối cũng có thể được đưa vào để lọc các khối đã giải mã nhằm loại bỏ các thành phần lạ dạng khối. Các bộ lọc vòng lặp khác (trong vòng lặp mã hoá hoặc sau vòng lặp mã hoá) cũng có thể được dùng để làm trơn những chỗ chuyển tiếp điểm ảnh, hoặc, theo cách khác, để nâng cao chất lượng tín hiệu video. Sau đó, các khối video đã giải mã trong khung hoặc hình ảnh cho trước được lưu trữ vào bộ nhớ hình ảnh chuẩn 82, bộ nhớ này lưu trữ các hình ảnh chuẩn dùng để bù chuyển động sau này. Bộ nhớ hình ảnh chuẩn 82 còn lưu trữ các khối video đã giải mã để sau này hiển thị trên thiết bị hiển thị, như thiết bị hiển thị 32 trên Fig.1.

Theo các kỹ thuật nêu trong sáng chế, bộ giải mã dữ liệu video 30 có thể giải mã bộ thông số video (VPS), bộ thông số lớp (LPS) và/hoặc bộ thông số nhóm, theo các kỹ thuật nêu trong sáng chế, cũng như bộ thông số chuỗi (SPS), bộ thông số hình ảnh (PPS), bộ thông số thích ứng (APS), hoặc các cấu trúc dữ liệu báo hiệu khác. Cụ thể hơn, bộ giải mã entropy 70 có thể được tạo cấu hình để giải mã một cấu trúc dữ liệu bất kỳ hoặc tất cả các cấu trúc dữ liệu. Khi giải mã các cấu trúc dữ liệu như vậy, bộ giải mã entropy 70 có thể xác định các thông số sẽ được dùng để giải mã dữ liệu video tương ứng. Ví dụ, bộ giải mã dữ liệu video 30 có thể giải mã các chuỗi tương ứng của dữ liệu video trên một hoặc nhiều lớp sử dụng các thông số của bộ thông số VPS đã giải mã.

Tuy không được thể hiện trên Fig.3, nhưng bộ giải mã dữ liệu video 30 có thể còn có bộ nhớ đệm hình ảnh mã hoá (CPB). Bộ nhớ CPB thường được đặt ở trước bộ giải mã entropy 70. Theo cách khác, bộ nhớ CPB có thể được kết nối với bộ giải mã entropy 70 để lưu trữ tạm thời, hoặc ở đầu ra của bộ giải mã entropy 70 để lưu trữ dữ liệu đã giải mã entropy cho tới khi dữ liệu đó được giải mã. Nói chung, bộ nhớ CPB lưu trữ dữ liệu video mã hoá cho đến khi dữ liệu video mã hoá được giải mã, ví dụ, như được chỉ báo bằng các thông số HRD mà bộ giải mã dữ liệu video 30 có thể tách ra từ bộ thông số VPS đã giải mã. Tương tự như vậy, các bộ phận khác của bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để giải mã dữ liệu video bằng cách sử dụng, ví dụ, bộ thông số VPS. Ví dụ, bộ giải mã dữ liệu video 30 có thể giải mã các thông

tin nhận dạng lớp thời gian cho các hình ảnh của các lớp thời gian khác nhau, dữ liệu chỉ báo số lượng hình ảnh cần sắp xếp lại và/hoặc số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ hình ảnh chuẩn 82 (trên hình vẽ thể hiện bộ nhớ đệm DPB).

Ngoài ra, bộ giải mã dữ liệu video 30 có thể có các bộ xử lý khác để xử lý dữ liệu video theo các công cụ mã hoá được tạo ra bằng cách mở rộng chuẩn mã hoá dữ liệu video. Theo cách khác, các bộ phận hiện có của bộ giải mã dữ liệu video 30 được thể hiện trên Fig.3 có thể được tạo cấu hình để thi hành các công cụ mã hoá mở rộng. Bộ giải mã entropy 70 có thể được tạo cấu hình để giải mã dữ liệu mở rộng của bộ thông số VPS và cung cấp dữ liệu mở rộng đó cho các bộ phận được tạo cấu hình để thi hành các công cụ mã hoá được đề xuất cho phần mở rộng.

Theo cách này, bộ giải mã dữ liệu video 30 trên Fig.3 là ví dụ về bộ giải mã dữ liệu video được tạo cấu hình để mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

Mặc dù thường được mô tả dựa vào bộ giải mã dữ liệu video, nhưng quy trình giải mã bộ thông số VPS có thể được thực hiện bằng các thiết bị khác, ví dụ, phần tử mạng nhận biết phương tiện (MANE). MANE có thể được tạo cấu hình để giải mã bộ thông số VPS theo các kỹ thuật nêu trong sáng chế. MANE có thể còn tạo ra dữ liệu cho các bộ thông số khác, như một hoặc nhiều bộ thông số chuỗi, sử dụng dữ liệu của bộ thông số VPS. Theo cách này, MANE có thể có tính tương thích ngược với các chuẩn trước đây, như chuẩn ITU-T H.264/AVC.

Fig.4 là sơ đồ khái niệm thể hiện ví dụ về mẫu dự báo MVC. Phiên bản mã hoá dữ liệu video có nhiều cảnh nhìn (MVC) là phiên bản mở rộng của chuẩn ITU-T H.264/AVC. Kỹ thuật tương tự có thể được áp dụng cho chuẩn HEVC. Trong ví dụ được thể hiện trên Fig.4 có tám cảnh nhìn (với ID cảnh nhìn từ “S0” đến “S7”) và mười hai vị trí thời gian (từ “T0” đến “T11”) cho mỗi cảnh nhìn. Có nghĩa là, mỗi hàng trên Fig.4 tương ứng với một cảnh nhìn, còn mỗi cột thể hiện một vị trí thời gian.

Cấu trúc dự báo MVC thông thường (trong đó có cả dự báo liên hình ảnh trong mỗi cảnh nhìn và dự báo liên cảnh nhìn) để mã hoá dữ liệu video có nhiều cảnh nhìn

được thể hiện trên Fig.4, trong đó hướng dự báo được thể hiện bằng các mũi tên, đối tượng có mũi tên chỉ đến sử dụng đối tượng có mũi tên chỉ đi làm chuẩn dự báo. Theo phiên bản mở rộng MVC, chế độ dự báo liên cảnh nhìn được hỗ trợ bằng cách bù chuyển động từ cảnh nhìn khác, có thể sử dụng cú pháp bù chuyển động theo chuẩn H.264/AVC, nhưng cho phép hình ảnh trong một cảnh nhìn khác được dùng làm hình ảnh chuẩn.

Chế độ mã hoá hai cảnh nhìn cũng có thể được hỗ trợ trong phiên bản mở rộng MVC, và một ưu điểm của phiên bản mở rộng MVC là bộ mã hoá MVC có thể chọn nhiều hơn hai cảnh nhìn làm đầu vào dữ liệu video 3D và bộ giải mã MVC có thể giải mã dạng biểu diễn có nhiều cảnh nhìn như vậy. Vì vậy, mọi bộ phận kết xuất có bộ giải mã MVC đều có thể được tạo cấu hình để thu nội dung video 3D có nhiều hơn hai cảnh nhìn.

Mặc dù phiên bản mở rộng MVC có cảnh nhìn gọi là cảnh nhìn cơ bản có thể giải mã được bằng các bộ giải mã theo chuẩn H.264/AVC, và phiên bản mở rộng MVC cũng có thể hỗ trợ cặp cảnh nhìn lập thể, nhưng một ưu điểm của phiên bản mở rộng MVC là có thể hỗ trợ, ví dụ, cho việc sử dụng nhiều hơn hai cảnh nhìn làm đầu vào dữ liệu video 3D và giải mã dữ liệu video 3D được biểu diễn bằng nhiều cảnh nhìn như vậy. Thiết bị kết xuất của máy khách có bộ giải mã MVC có thể đoán trước nội dung video 3D có nhiều cảnh nhìn.

Thứ tự giải mã MVC thông thường được gọi là giải mã ưu tiên theo thời gian. Đơn vị truy nhập có thể chứa các hình ảnh mã hoá của tất cả các cảnh nhìn với một thời điểm xuất ra. Ví dụ, mọi hình ảnh ở vị trí thời gian T0 có thể được đưa vào trong đơn vị truy nhập chung thứ nhất, mọi hình ảnh ở vị trí thời gian T1 có thể được đưa vào trong đơn vị truy nhập chung thứ hai, v.v.. Thứ tự giải mã không nhất thiết phải giống như thứ tự xuất ra hoặc thứ tự hiển thị.

Các khung trên Fig.4 được thể hiện ở chỗ giao nhau của mỗi hàng và mỗi cột trên Fig.4 sử dụng khối màu sẫm có chữ cái để chỉ rõ khung tương ứng được mã hoá dự báo nội cấu trúc (tức là, khung-I), hay được mã hoá dự báo liên cấu trúc theo một chiều (tức là, khung-P) hay theo nhiều chiều (tức là, khung-B). Nói chung, chiều dự báo được thể hiện bằng các mũi tên, trong đó khung có mũi tên chỉ đến sử dụng khung có mũi tên chỉ đi làm chuẩn dự báo. Ví dụ, khung-P của cảnh nhìn S2 ở vị trí thời gian

T0 được dự báo từ khung-I của cảnh nhìn S0 ở vị trí thời gian T0.

Đối với phương pháp mã hoá dữ liệu video có một cảnh nhìn, các khung của chuỗi dữ liệu video theo chuẩn mã hoá video có nhiều cảnh nhìn có thể được mã hoá dự báo dựa vào các khung ở các vị trí thời gian khác nhau. Ví dụ, khung-b của cảnh nhìn S0 ở vị trí thời gian T1 có mũi tên chỉ đến đi xuất phát từ khung-I của cảnh nhìn S0 ở vị trí thời gian T0, chỉ báo rằng khung-b được dự báo từ khung-I. Ngoài ra, trong trường hợp mã hoá dữ liệu video có nhiều cảnh nhìn, các khung có thể được dự báo liên cảnh nhìn. Có nghĩa là, một cảnh nhìn thành phần có thể sử dụng các cảnh nhìn thành phần trong các cảnh nhìn khác làm chuẩn. Theo phiên bản mở rộng MVC, ví dụ, dự báo liên cảnh nhìn được thực hiện như thể cảnh nhìn thành phần trong một cảnh nhìn khác là chuẩn dự báo liên cảnh nhìn. Các chuẩn dự báo liên cảnh nhìn có thể có được báo hiệu trong phần mở rộng MVC có bộ thông số chuỗi (SPS) và có thể được sửa đổi bằng quy trình thiết lập danh mục hình ảnh chuẩn, quy trình này cho phép sắp xếp linh hoạt các chuẩn dự báo liên cấu trúc hoặc dự báo liên cảnh nhìn.

Trong phiên bản mở rộng MVC của chuẩn H.264/AVC, ví dụ, chế độ dự báo liên cảnh nhìn được hỗ trợ bằng cách bù chuyển động từ cảnh nhìn khác, sử dụng cú pháp bù chuyển động theo chuẩn H.264/AVC, nhưng cho phép hình ảnh trong một cảnh nhìn khác được dùng làm hình ảnh chuẩn. Chế độ mã hoá hai cảnh nhìn có thể được hỗ trợ trong phiên bản mở rộng MVC, thường được gọi là các cảnh nhìn lập thể. Một ưu điểm của phiên bản mở rộng MVC là bộ mã hoá MVC có thể chọn nhiều hơn hai cảnh nhìn làm đầu vào dữ liệu video 3D và bộ giải mã MVC có thể giải mã dữ liệu được biểu diễn bằng nhiều cảnh nhìn như vậy. Do đó, thiết bị kết xuất có bộ giải mã MVC có thể đoán trước nội dung video 3D có nhiều hơn hai cảnh nhìn.

Theo phiên bản mở rộng MVC, chế độ dự báo liên cảnh nhìn (*IVP: Inter-View Prediction*) được phép thực hiện trên các hình ảnh nằm trong cùng một đơn vị truy nhập (tức là, có cùng một vị trí thời gian). Thông thường, đơn vị truy nhập là đơn vị dữ liệu chứa tất cả các cảnh nhìn thành phần (ví dụ, tất cả các đơn vị NAL) với một vị trí thời gian chung. Do đó, theo phiên bản mở rộng MVC, chế độ dự báo liên cảnh nhìn được phép thực hiện trên các hình ảnh nằm trong cùng một đơn vị truy nhập. Khi mã hoá một hình ảnh trong một cảnh nhìn không phải là cảnh nhìn cơ bản, thì hình ảnh đó có thể được bổ sung vào danh mục hình ảnh chuẩn, nếu nó nằm trong một cảnh

nhìn khác nhưng có cùng vị trí thời gian (ví dụ, cùng giá trị POC, và do đó, nằm trong cùng một đơn vị truy nhập). Hình ảnh chuẩn dự báo liên cảnh nhìn có thể được đưa vào một vị trí bất kỳ trong danh mục hình ảnh chuẩn, giống như mọi hình ảnh chuẩn dự báo liên cấu trúc.

Trong trường hợp mã hoá dữ liệu video có nhiều cảnh nhìn, có hai loại vector chuyển động. Một loại là vector chuyển động thông thường chỉ đến các hình ảnh chuẩn theo thời gian, và chế độ dự báo liên cấu trúc tương ứng được gọi là dự báo có bù chuyển động (*MCP: Motion Compensated Prediction*). Một loại khác là vector chuyển động từ cảnh nhìn khác chỉ đến các hình ảnh trong cảnh nhìn khác, và chế độ dự báo liên cảnh nhìn tương ứng được gọi là dự báo có bù chuyển động từ cảnh nhìn khác (*DCP: Disparity-Compensated Prediction*).

Theo chuẩn HEVC thông thường, có hai chế độ dự báo cho các thông số chuyển động: một chế độ là chế độ kết hợp, và một chế độ nữa là chế độ dự báo vector chuyển động nâng cao (*AMVP: Advanced Motion Vector Prediction*). Ở chế độ kết hợp, danh mục thông số chuyển động dự bị (các hình ảnh chuẩn, và các vector chuyển động) được thiết lập, trong đó thông số chuyển động dự bị có thể được tìm ra từ các khối liên kề theo không gian hoặc theo thời gian. Các khối liên kề theo không gian hoặc theo thời gian có thể tạo ra danh mục dự bị, tức là, một tập hợp gồm các phần tử dự bị mà thông tin dự báo chuyển động có thể được chọn từ đó. Vì vậy, bộ mã hoá dữ liệu video 20 có thể mã hoá các thông số chuyển động được chọn làm thông tin dự báo chuyển động bằng cách mã hoá chỉ số thành danh mục dự bị. Sau khi bộ giải mã dữ liệu video 30 đã giải mã chỉ số, tất cả các thông số chuyển động của khối tương ứng mà chỉ số chỉ đến đó đều có thể được kế thừa, ở chế độ kết hợp.

Ở chế độ AMVP, do đó theo chuẩn HEVC thông thường, danh mục dự bị của các vector chuyển động dự báo cho mỗi giả thuyết chuyển động được tìm ra dựa vào chỉ số chuẩn mã hoá. Danh mục này có các vector chuyển động của các khối liên kề được liên hệ với chỉ số chuẩn giống như vector chuyển động dự báo theo thời gian được tìm ra dựa vào các thông số chuyển động của khối liên kề của khối đồng vị trong hình ảnh chuẩn theo thời gian. Các vector chuyển động đã chọn được báo hiệu bằng cách truyền chỉ số này trong danh mục dự bị. Ngoài ra, các giá trị chỉ số chuẩn và độ chênh lệch vector chuyển động cũng được báo hiệu.

Fig.4 thể hiện các ví dụ về dự báo liên cảnh nhìn. Các khung của cảnh nhìn S1, trong ví dụ trên Fig.4, được thể hiện dưới dạng được dự báo từ các khung ở các vị trí thời gian khác nhau của cảnh nhìn S1, cũng như được dự báo liên cảnh nhìn từ các khung của cảnh nhìn S0 và cảnh nhìn S2 ở cùng vị trí thời gian. Ví dụ, khung-b của cảnh nhìn S1 ở vị trí thời gian T1 được dự báo từ mỗi khung-b của cảnh nhìn S1 ở các vị trí thời gian T0 và T2, cũng như các khung-b của cảnh nhìn S0 và cảnh nhìn S2 ở vị trí thời gian T1.

Trong ví dụ thể hiện trên Fig.4, chữ “B” hoa và chữ “b” thường được dùng để chỉ báo các quan hệ phân cấp khác nhau giữa các khung, chứ không phải dùng để chỉ các phương pháp mã hoá khác nhau. Nói chung, các khung có chữ “B” hoa ở vị trí tương đối cao hơn trong cấu trúc phân cấp dự báo so với các khung có chữ “b” thường. Fig.4 còn thể hiện các dạng cải biến của cấu trúc phân cấp dự báo sử dụng các mức độ sẫm màu khác nhau, trong đó các khung có mức độ sẫm màu nhiều hơn (tức là, tương đối tối hơn) ở vị trí cao hơn trong cấu trúc phân cấp dự báo so với các khung có mức độ sẫm màu ít hơn (tức là, tương đối sáng hơn). Ví dụ, tất cả các khung-I trên Fig.4 được thể hiện bằng màu sẫm hoàn toàn, còn các khung-P có màu sáng hơn một chút, và các khung-B (và các khung-b chữ thường) có các mức độ sẫm màu khác nhau tương đối so với nhau, nhưng luôn sáng hơn so với màu của các khung-P và các khung-I.

Nói chung, cấu trúc phân cấp dự báo có liên quan đến các chỉ số thứ tự cảnh nhìn, trong đó các khung ở mức tương đối cao hơn trong cấu trúc phân cấp dự báo sẽ được giải mã trước khi giải mã các khung ở mức tương đối thấp hơn trong cấu trúc phân cấp, sao cho các khung ở mức tương đối cao hơn trong cấu trúc phân cấp có thể được dùng làm khung chuẩn khi giải mã các khung ở mức tương đối thấp hơn trong cấu trúc phân cấp. Chỉ số thứ tự cảnh nhìn là chỉ số chỉ báo thứ tự giải mã của các cảnh nhìn thành phần trong đơn vị truy nhập. Các chỉ số thứ tự cảnh nhìn được quy định trong phiên bản mở rộng MVC có bộ thông số SPS, như được mô tả trong phần Phụ lục H của đặc tả H.264/AVC (phiên bản sửa đổi MVC). Trong bộ thông số SPS, đối với mỗi chỉ số i , giá trị `view_id` tương ứng được báo hiệu. Trong một số ví dụ, việc giải mã các cảnh nhìn thành phần sẽ tuân theo thứ tự tăng dần của chỉ số thứ tự cảnh nhìn. Nếu tất cả các cảnh nhìn đều được hiển thị, thì các chỉ số thứ tự cảnh nhìn

được sắp xếp liên tiếp từ 0 đến num_views_minus_1 .

Theo cách này, các khung dùng làm khung chuẩn có thể được giải mã trước khi giải mã các khung được mã hoá dựa vào khung chuẩn. Chỉ số thứ tự cảnh nhìn là chỉ số chỉ báo thứ tự giải mã của các cảnh nhìn thành phần trong đơn vị truy nhập. Đối với mỗi chỉ số thứ tự cảnh nhìn i , giá trị view_id tương ứng được báo hiệu. Việc giải mã các cảnh nhìn thành phần tuân theo thứ tự tăng dần của các chỉ số thứ tự cảnh nhìn. Nếu tất cả các cảnh nhìn đều được hiển thị, thì tập hợp chỉ số thứ tự cảnh nhìn có thứ tự liên tiếp từ giá trị không đến giá trị bằng tổng số cảnh nhìn trừ một.

Đối với các cảnh nhìn ở cùng mức trong cấu trúc phân cấp, thứ tự giải mã có thể không khớp nhau. Ví dụ, khung-I của cảnh nhìn S0 ở vị trí thời gian T0 được dùng làm khung chuẩn cho khung-P của cảnh nhìn S2 ở vị trí thời gian T0, rồi khung-P của cảnh nhìn S2 ở vị trí thời gian T0 lại được dùng làm khung chuẩn cho khung-P của cảnh nhìn S4 ở vị trí thời gian T0. Do đó, khung-I của cảnh nhìn S0 ở vị trí thời gian T0 sẽ được giải mã trước khung-P của cảnh nhìn S2 ở vị trí thời gian T0, khung-P của cảnh nhìn S2 ở vị trí thời gian T0 sẽ được giải mã trước khung-P của cảnh nhìn S4 ở vị trí thời gian T0. Tuy nhiên, giữa cảnh nhìn S1 và cảnh nhìn S3, thứ tự giải mã không khớp nhau, vì cảnh nhìn S1 và cảnh nhìn S3 không dựa vào nhau khi dự báo, mà được dự báo từ các cảnh nhìn ở mức cao hơn trong cấu trúc phân cấp dự báo. Ngoài ra, cảnh nhìn S1 có thể được giải mã trước cảnh nhìn S4, miễn là cảnh nhìn S1 được giải mã sau cảnh nhìn S0 và cảnh nhìn S2.

Theo cách này, sự sắp xếp theo cấu trúc phân cấp có thể được dùng để mô tả các cảnh nhìn từ S0 đến S7. Ký hiệu $SA > SB$ có nghĩa là cảnh nhìn SA sẽ được giải mã trước cảnh nhìn SB. Nếu sử dụng ký hiệu này, thì trong ví dụ thể hiện trên Fig.4, $S0 > S2 > S4 > S6 > S7$. Ngoài ra, trong ví dụ thể hiện trên Fig.4, $S0 > S1$, $S2 > S1$, $S2 > S3$, $S4 > S3$, $S4 > S5$, và $S6 > S5$. Mọi thứ tự giải mã cho các cảnh nhìn nếu không vi phạm các quy định này thì đều có thể áp dụng được. Do đó, có thể áp dụng nhiều thứ tự giải mã khác nhau, với một số điều kiện hạn chế.

Theo các kỹ thuật nêu trong sáng chế, mỗi cảnh nhìn từ S0 đến S7 có thể được coi là một lớp tương ứng của dòng bit tương ứng. Do đó, bộ thông số VPS có thể mô tả các thông số của dòng bit áp dụng cho một hoặc tất cả các cảnh nhìn từ S0 đến S7, trong khi đó các bộ thông số lớp riêng biệt có thể được cung cấp cho một hoặc tất cả

các cảnh nhìn từ S0 đến S7. Ngoài ra, bộ thông số nhóm có thể được cung cấp cho một nhóm bộ thông số, sao cho các lát trong các hình ảnh riêng biệt của các cảnh nhìn từ S0 đến S7 có thể chỉ đơn giản chỉ đến thông tin nhận dạng của bộ thông số nhóm.

Như được thể hiện trên Fig.4, cảnh nhìn thành phần có thể sử dụng các cảnh nhìn thành phần trong cảnh nhìn khác để làm chuẩn. Chế độ đó được gọi là chế độ dự báo liên cảnh nhìn. Theo phiên bản mở rộng MVC, chế độ dự báo liên cảnh nhìn được thực hiện khi cảnh nhìn thành phần trong cảnh nhìn khác là chuẩn dự báo liên cảnh nhìn. Bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể mã hoá các chuẩn dự báo liên cảnh nhìn có thể có trong phiên bản mở rộng MVC có bộ thông số chuỗi (SPS) (như được thể hiện trong ví dụ nêu trong bảng 1). Bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể còn sửa đổi các chuẩn dự báo liên cảnh nhìn có thể có bằng cách thực hiện quy trình thiết lập danh mục hình ảnh chuẩn, quy trình này cho phép sắp xếp linh hoạt các chuẩn dự báo liên cấu trúc hoặc dự báo liên cảnh nhìn.

Bảng 1

seq_parameter_set_mvc_extension() {	C	Bộ mô tả
num_views_minus1	0	ue(v)
for(i = 0; i <= num_views_minus1; i++)		
view_id[i]	0	ue(v)
for(i = 1; i <= num_views_minus1; i++) {		
num_anchor_refs_l0[i]	0	ue(v)
for(j = 0; j < num_anchor_refs_l0[i]; j++)		
anchor_ref_l0[i][j]	0	ue(v)
num_anchor_refs_l1[i]	0	ue(v)
for(j = 0; j < num_anchor_refs_l1[i]; j++)		
anchor_ref_l1[i][j]	0	ue(v)
}		
for(i = 1; i <= num_views_minus1; i++) {		
num_non_anchor_refs_l0[i]	0	ue(v)
for(j = 0; j < num_non_anchor_refs_l0[i]; j++)		
non_anchor_ref_l0[i][j]	0	ue(v)

num_non_anchor_refs_l1[i]	0	ue(v)
for(j = 0; j < num_non_anchor_refs_l1[i]; j++)		
non_anchor_ref_l1[i][j]	0	ue(v)
}		
num_level_values_signaled_minus1	0	ue(v)
for(i = 0; i <= num_level_values_signaled_minus1; i++) {		
level_idc[i]	0	u(8)
num_applicable_ops_minus1[i]	0	ue(v)
for(j = 0; j <= num_applicable_ops_minus1[i]; j++) {		
applicable_op_temporal_id[i][j]	0	u(3)
applicable_op_num_target_views_minus1[i][j]	0	ue(v)
for(k = 0; k <= applicable_op_num_target_views_minus1[i][j]; k++)		
applicable_op_target_view_id[i][j][k]	0	ue(v)
applicable_op_num_views_minus1[i][j]	0	ue(v)
}		
}		
}		

Trong phiên bản mở rộng MVC có bộ thông số SPS được thể hiện trong bảng 1, với mỗi cảnh nhìn, số lượng cảnh nhìn có thể dùng để tạo ra danh mục hình ảnh chuẩn 0 và danh mục hình ảnh chuẩn 1 được báo hiệu. Quan hệ dự báo cho hình ảnh neo, như được báo hiệu trong phiên bản mở rộng MVC có bộ thông số SPS, có thể khác với quan hệ dự báo cho hình ảnh không phải hình ảnh neo (được báo hiệu trong phiên bản mở rộng MVC có bộ thông số SPS) của cùng một cảnh nhìn.

Các chuẩn mã hoá dữ liệu video là ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 hoặc ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual và ITU-T H.264 (còn gọi là ISO/IEC MPEG-4 AVC), kể cả các phiên bản mở rộng mã hoá dữ liệu video có thể mở rộng cấp độ (SVC) và mã hoá dữ liệu video có nhiều cảnh nhìn (MVC).

Ngoài ra, có một chuẩn mã hoá dữ liệu video mới, đó là chuẩn mã hoá dữ liệu video đạt hiệu quả cao (HEVC), đang được phát triển bởi nhóm hợp tác liên kết mã hoá dữ liệu video (JCT-VC) gồm nhóm chuyên gia mã hoá dữ liệu video (VCEG) của

tổ chức ITU-T và nhóm chuyên gia ảnh động (MPEG) của tổ chức ISO/IEC. Phiên bản dự thảo hiện thời (*WD: Working Draft*) của chuẩn HEVC, và dưới đây được gọi là HEVC WD4, công bố tại địa chỉ http://phenix.int-evry.fr/jct/doc_end_user/documents/6_Torino/wg11/JCTVC-F803-v3.zip, gọi tắt là HEVC WD4d1.

Cơ chế bộ thông số chuỗi và bộ thông số hình ảnh tách việc truyền thông tin ít khi thay đổi với việc truyền dữ liệu khối mã hoá. Chuỗi và các bộ thông số hình ảnh có thể, trong một số ứng dụng, được truyền “ngoài dải” sử dụng cơ chế truyền tin cậy. Cấu trúc cú pháp tải hữu ích chứa chuỗi byte thô (RBSP) cho bộ thông số hình ảnh có thể có các thông số được tham chiếu bởi các lớp trừu tượng hoá mạng (NAL) có lát mã hoá của một hoặc nhiều hình ảnh mã hoá. Cấu trúc cú pháp RBSP cho bộ thông số chuỗi có thể có các thông số được tham chiếu bởi một hoặc nhiều cấu trúc cú pháp RBSP cho bộ thông số hình ảnh hay một hoặc nhiều đơn vị NAL có thông tin nâng cao bổ sung (*SEI: Supplemental Enhancement Information*) chứa thông báo SEI về chu kỳ đưa vào bộ nhớ đệm. Cấu trúc cú pháp RBSP cho bộ thông số chuỗi có thể có các thông số được tham chiếu bởi một hoặc nhiều cấu trúc cú pháp RBSP cho bộ thông số hình ảnh hay một hoặc nhiều đơn vị NAL có thông tin SEI chứa thông báo SEI về chu kỳ đưa vào bộ nhớ đệm.

Bộ thông số chuỗi có thể có một tập hợp tùy chọn của các thông số được gọi là thông tin sử dụng dữ liệu video (VUI). Thông tin VUI có thể có ba loại thông tin tùy chọn như sau: thông tin biểu diễn dữ liệu video, thông tin bộ giải mã chuẩn giả định (HRD) và thông tin giới hạn dòng bit. Thông tin biểu diễn dữ liệu video có cỡ ảnh, các mức độ dịch pha của thành phần màu liên quan đến sự biến đổi không gian màu so với thành phần độ chói và tốc độ khung. Thông tin HRD có các thông số đưa dữ liệu video vào bộ nhớ đệm với các chuỗi dữ liệu video mã hoá. Thông tin giới hạn dòng bit có các giới hạn về khoảng vector chuyển động, dung lượng bộ nhớ đệm hình ảnh giải mã (DPB), và số lượng khung cần sắp xếp lại và kích thước của các khối mã hoá (ví dụ, khối ảnh lớn hoặc đơn vị mã hoá (CU)) và các hình ảnh.

Chuẩn HEVC WD5 hỗ trợ bộ thông số thích ứng (APS). Khái niệm bộ thông số thích ứng (APS) cũng được mô tả trong tài liệu JCTVC-F747, được công bố tại địa chỉ http://phenix.int-evry.fr/jct/doc_end_user/documents/6_Torino/wg11/JCTVC-F747-v4.zip.

Phần đầu đơn vị NAL thống nhất có thể được sử dụng cho cả các dòng bit không thể mở rộng cấp độ lẫn các dòng bit có thể mở rộng cấp độ theo chuẩn HEVC tuân theo các phiên bản mở rộng mã hoá có thể mở rộng cấp độ hoặc mã hoá có nhiều cảnh nhìn của chuẩn HEVC. Phần đầu đơn vị NAL thống nhất có thể khác với phần đầu đơn vị NAL theo chuẩn HEVC hiện thời ở các khía cạnh như sau: có thể có độ dài phần đầu đơn vị NAL cố định cho toàn bộ một chuỗi dữ liệu video mã hoá, trong khi độ dài này có thể thay đổi giữa các chuỗi dữ liệu video mã hoá khác nhau, và mã hoá có hiệu quả trên các phần tử cú pháp có thể mở rộng cấp độ trong phần đầu đơn vị NAL, và khi không cần có một phần tử cú pháp cụ thể thì phần tử cú pháp đó cũng không cần có mặt. Theo cấu trúc như vậy, một loại đơn vị NAL khác hoặc bộ thông số khác có thể được sử dụng cho toàn bộ dòng bit.

Fig.5 là sơ đồ khái niệm thể hiện bộ thông số video (VPS) và các bộ thông số lớp (LPS). Dấu ba chấm sau bộ thông số LPS thứ hai trên Fig.5 được dùng để chỉ báo rằng có thể có một số lượng bất kỳ N bộ thông số VPS, trong đó N là số nguyên. Ví dụ, mỗi lớp (ví dụ, mỗi lớp SVC hoặc mỗi cảnh nhìn MVC) có thể có một bộ thông số LPS tương ứng. Bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video 20 hoặc bộ giải mã dữ liệu video 30, có thể được tạo cấu hình để mã hoá một bộ thông số VPS và một hoặc nhiều bộ thông số LPS, như được thể hiện trên Fig.5.

Bảng 2 dưới đây thể hiện ví dụ về cú pháp tải hữu ích chứa chuỗi byte thô (RBPS) cho bộ thông số VPS.

Bảng 2

video_parameter_set_rbsp() {	Bộ mô tả
video_para_set_id	u(8)
// sample dimension counter description	
cnt_p	u(3)
cnt_d	u(3)
cnt_t	u(3)
cnt_q	u(3)
cnt_v	u(4)
cnt_f	u(4)
// sample index to characteristic mapping	

for(i = 0; i < cnt_d; i++) {	
pic_width_in_luma_samples[i]	ue(v)
pic_height_in_luma_samples[i]	ue(v)
bit_depth_luma_minus8[i]	ue(v)
bit_depth_chroma_minus8[i]	ue(v)
chroma_format_idc[i]	u(2)
}	
for(i = 0; i < cnt_t; i++)	
average_frame_rate[i]	u(16)
if(cnt_v > 1)	
for(i=0; i < cnt_v; i++)	
view_id[i]	ue(v)
// control parameters and tool enabling/disabling flags	
log2_max_pic_order_cnt_lsb_minus4	ue(v)
chroma_pred_from_luma_enabled_flag	u(1)
loop_filter_across_slice_flag	u(1)
sample_adaptive_offset_enabled_flag	u(1)
adaptive_loop_filter_enabled_flag	u(1)
pcm_loop_filter_disable_flag	u(1)
cu_qp_delta_enabled_flag	u(1)
temporal_id_nesting_flag	u(1)
inter_4x4_enabled_flag	u(1)
operation_point_description()	
vps_extension_flag	u(1)
if(vps_extension_flag)	
while(more_rbsp_data())	
vps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

Các bộ mã hoá dữ liệu video có thể được tạo cấu hình sao cho chuỗi dữ liệu video mã hoá (ví dụ, dòng bit có một hoặc nhiều lớp) có thể chỉ có một bộ thông số video (VPS) hoạt động. Bộ thông số VPS có thể được đóng gói trong đơn vị NAL

thuộc một loại cụ thể. Ví dụ, `nal_unit_type` của cấu trúc cú pháp RBSP cho bộ thông số VPS có thể bằng 10. Ví dụ về ngữ nghĩa cho bộ thông số VPS trong bảng 2 được mô tả dưới đây:

Trong ví dụ này, `video_para_set_id` xác định bộ thông số video (VPS) tương ứng.

Trong ví dụ này, `cnt_p` xác định số lượng tối đa cho các giá trị `priority_id` có mặt trong chuỗi dữ liệu video mã hoá tương ứng.

Trong ví dụ này, `cnt_d` xác định số lượng lớp phụ thuộc tối đa có mặt trong chuỗi dữ liệu video mã hoá tương ứng. Nhiều cảnh nhìn có cùng một độ phân giải có thể được coi là thuộc về cùng một lớp phụ thuộc. Hai lớp phụ thuộc có thể có cùng một độ phân giải không gian.

Trong ví dụ này, `cnt_t` xác định số lượng lớp thời gian tối đa có mặt trong chuỗi dữ liệu video mã hoá.

Trong ví dụ này, `cnt_q` xác định số lượng lớp chất lượng tối đa có mặt trong lớp phụ thuộc trong chuỗi dữ liệu video mã hoá.

Trong ví dụ này, `cnt_v` xác định số lượng cảnh nhìn tối đa có mặt trong chuỗi dữ liệu video mã hoá.

Trong ví dụ này, `cnt_f` xác định số lượng bit được dùng để thể hiện phần tử cú pháp `reserved_flags` trong phần đầu đơn vị NAL.

Trong ví dụ này, `pic_width_in_luma_samples[i]` và `pic_height_in_luma_samples[i]` lần lượt xác định chiều rộng và chiều cao cho độ phân giải của lớp phụ thuộc thứ *i* tính bằng đơn vị mẫu độ chói.

Trong ví dụ này, `bit_depth_luma_minus8[i]` cộng 8 và `bit_depth_chroma_minus8[i]` cộng 8 xác định độ sâu bit của thành phần độ chói và các thành phần màu của dạng biểu diễn độ sâu bit thứ *i*.

Trong ví dụ này, `chroma_format_idc[i]` xác định định dạng mẫu thành phần màu biểu diễn định dạng thành phần màu thứ *i*. Ví dụ, giá trị 0 có thể chỉ báo định dạng 4:2:0; giá trị 1 có thể chỉ báo định dạng 4:4:4, giá trị 2 có thể chỉ báo định dạng 4:2:2 và giá trị 3 có thể chỉ báo định dạng 4:0:0.

Trong ví dụ này, `average_frame_rate[i]` xác định tốc độ khung trung bình của

dạng biểu diễn lớp thời gian thứ i , tính theo đơn vị khung/256 giây.

Trong ví dụ này, `view_id[i]` xác định thông tin nhận dạng cảnh nhìn của cảnh nhìn thứ i , có chỉ số thứ tự cảnh nhìn bằng i . Nếu không có, thì giá trị `view_id[0]` có thể được coi là bằng 0. `vps_extension_flag` bằng 0 xác định rằng không có phần tử cú pháp `vps_extension_data_flag` trong cấu trúc cú pháp RBSP cho bộ thông số video. `vps_extension_flag` có thể bằng 0 trong các dòng bit tuân theo chuẩn HEVC sắp tới. Giá trị bằng 1 của `vps_extension_flag` có thể được dành riêng, ví dụ, để sau này sử dụng cho ITU-T | ISO/IEC. Các bộ giải mã, như bộ giải mã dữ liệu video 30, có thể bỏ qua tất cả các dữ liệu đứng sau giá trị bằng 1 của `vps_extension_flag` trong đơn vị NAL có bộ thông số video.

Trong ví dụ này, `vps_extension_data_flag` có thể có giá trị bất kỳ. Nó không ảnh hưởng đến sự phù hợp với các profile được xác định theo chuẩn HEVC sắp tới, mà cho phép phát triển tiếp chuẩn sắp tới.

Các phần tử cú pháp khác trong bộ thông số VPS có thể có ngữ nghĩa giống như các phần tử cú pháp có cùng tên gọi trong bộ thông số SPS theo phiên bản dự thảo hiện thời của chuẩn HEVC. Các phần tử cú pháp này có thể áp dụng cho chuỗi dữ liệu video mã hoá tham chiếu đến bộ thông số VPS, trừ trường hợp bị ghi đè bởi các bộ thông số mức thấp hơn.

Trong một số ví dụ, `3DV_flag` có thể còn được báo hiệu trong bộ thông số VPS để chỉ báo độ sâu bit có trong chuỗi dữ liệu video mã hoá hay không.

Trong một số ví dụ, các thông số VUI được báo hiệu trong bộ thông số LPS.

Trong một số ví dụ, các phần tử cú pháp `cnt_p`, `cnt_t`, `cnt_d`, `cnt_q` và `cnt_v` xác định số lượng bit lần lượt được dùng để mã hoá `priority_id`, `temporal_id`, `dependency_id`, `quality_id` và `view_idx`, và số lượng tối đa của các giá trị `priority_id`, các lớp thời gian, các lớp phụ thuộc, các lớp chất lượng và các cảnh nhìn có mặt trong chuỗi dữ liệu video mã hoá cũng có thể được báo hiệu trong bộ thông số VPS.

Trong một số ví dụ, một loại đơn vị NAL khác có thể được đưa vào sử dụng để chứa các phần tử cú pháp `cnt_p`, `cnt_t`, `cnt_d`, `cnt_q`, `cnt_v` và `cnt_f`. Loại đơn vị NAL mới này cũng có thể có thông tin nhận dạng (ID), và ID đó có thể được tham chiếu đến trong bộ thông số VPS.

Trong một số ví dụ, các phần tử cú pháp từ `log2_max_pic_order_cnt_lsb_minus4` đến `inter_4x4_enabled_flag` trong bảng 2 không được báo hiệu trong bộ thông số VPS, thay vào đó, bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể mã hoá các phần tử cú pháp này trong bộ thông số LPS.

Trong một số ví dụ, cấu trúc cú pháp `operation_point_description()` trong bảng 2 không có trong bộ thông số VPS; thay vào đó, bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30, hoặc các bộ phận khác (ví dụ, giao diện xuất 22 và/hoặc giao diện nhập 28), có thể mã hoá nội dung trong cấu trúc cú pháp `operation_point_description()` trong thông báo thông tin nâng cao bổ sung (SEI).

Trong một số ví dụ, bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể mã hoá các thông số thông tin sử dụng dữ liệu video (VUI) trong bộ thông số VPS. Ví dụ, bộ thông số VPS có thể chứa dữ liệu xác định thông tin giới hạn dòng bit, như các giới hạn về khoảng vector chuyển động, dung lượng bộ nhớ DPB, số lượng khung cần sắp xếp lại, và kích thước của các khối mã hoá (ví dụ, các khối ảnh lớn hoặc các đơn vị CU) và các hình ảnh. Theo cách này, bộ thông số VPS có thể xác định thông tin chỉ báo dung lượng bộ nhớ DPB cần thiết theo thứ tự cho bộ giải mã dữ liệu video (như bộ giải mã dữ liệu video 30) để giải mã chính xác dòng bit tương ứng, tức là, dòng bit có bộ thông số VPS. Tương tự như vậy, bộ thông số VPS có thể xác định thông tin sắp xếp thứ tự hình ảnh, có nghĩa là, số lượng hình ảnh có thể đứng trước một hình ảnh cho trước theo thứ tự giải mã và đứng sau hình ảnh cho trước đó theo thứ tự xuất ra (tức là, thứ tự hiển thị).

Theo cách khác hoặc theo cách bổ sung, bộ thông số VPS có thể chứa dữ liệu xác định thông tin bộ giải mã chuẩn giả định (HRD). Như đã nêu trên, bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể mã hoá (tức là, báo hiệu) các thông số VUI, các thông số này có thể chứa thông tin HRD, trong bộ thông số VPS. Do đó, bộ thông số VPS có thể chứa dữ liệu mô tả, ví dụ, các điểm hoạt động của dòng bit tương ứng. Ví dụ, bộ thông số VPS có thể chứa dữ liệu mô tả một hoặc nhiều thông tin trong số số lượng điểm hoạt động tối đa, tính phụ thuộc giữa các lớp hoặc các cảnh nhìn khác nhau, thông tin profile và mức cho mỗi điểm hoạt động, dạng biểu diễn đơn vị NAL có lớp VCL chứa điểm hoạt động cho mỗi điểm hoạt động, tỷ lệ bit

cho mỗi điểm hoạt động, tính phụ thuộc giữa các điểm hoạt động, các quy định hạn chế cho mỗi điểm hoạt động, thông tin VUI hoặc một phần thông tin VUI cho mỗi điểm hoạt động, và/hoặc thông tin VUI hoặc một phần thông tin VUI cho mỗi lớp hoặc cảnh nhìn.

Bộ thông số VPS cũng có thể chứa, với mỗi kích thước: giá trị chỉ số cụ thể, khoảng giá trị chỉ số hoặc danh mục giá trị chỉ số. Ví dụ, khi bộ thông số VPS chứa dữ liệu mô tả giá trị chỉ số cụ thể, thì giá trị chỉ số đó có thể tương ứng với, với độ phân giải không gian, độ sâu bit của định dạng mẫu thành phần màu. Ví dụ khác, khi bộ thông số VPS là khoảng giá trị chỉ số, với các lớp thời gian, thì khoảng này có thể là từ không (0) đến ID của lớp thời gian cao nhất, và với các lớp chất lượng, thì khoảng này có thể là từ không (0) đến ID của lớp chất lượng cao nhất. Ví dụ khác nữa, khi bộ thông số VPS chứa dữ liệu mô tả danh mục giá trị chỉ số, thì danh mục này có thể là danh mục giá trị chỉ số cảnh nhìn cho nhiều cảnh nhìn.

Trong một số ví dụ, bộ mã hoá dữ liệu video 20 có thể mã hoá (tức là, báo hiệu), và bộ giải mã dữ liệu video có thể giải mã, một hoặc nhiều thông số định dạng biểu diễn (chiều rộng, chiều cao, độ sâu bit, v.v.), và có thể có các tập hợp khác nhau của các thông số định dạng biểu diễn. Lớp hoặc điểm hoạt động có thể chỉ đến chỉ số của tập hợp thông số định dạng biểu diễn đó. Ví dụ về cấu trúc cú pháp cho tập hợp đó được thể hiện trong bảng 3 dưới đây.

Bảng 3

num_rep_formats_minus1	ue(v)
for(i = 0; i <= num_rep_formats_minus1; i++) {	
pic_width_in_luma_samples[i]	ue(v)
pic_height_in_luma_samples[i]	ue(v)
bit_depth_luma_minus8[i]	ue(v)
bit_depth_chroma_minus8[i]	ue(v)
chroma_format_idc[i]	u(2)
}	
for(i = 0; i < cnt_d; i++) {	
rep_format_idx[i]	ue(v)
}	

Trong một số ví dụ, `ref_format_idx` có thể được báo hiệu trong bộ thông số lớp.

Bảng 4 dưới đây thể hiện ví dụ về cú pháp cho các bộ mô tả điểm hoạt động.

Bảng 4

<code>operation_points_description() {</code>	Bộ mô tả
<code> num_operation_point_minus1</code>	<code>ue(v)</code>
<code> for(i = 0; i <= num_operation_points_minus1; i++) {</code>	
<code> op_profin_level_idc[i]</code>	<code>u(24)</code>
<code> operation_point_id[i]</code>	<code>ue(v)</code>
<code> priority_id[i]</code>	<code>ue(v)</code>
<code> temporal_id[i]</code>	<code>ue(v)</code>
<code> quality_id[i]</code>	<code>ue(v)</code>
<code> dependency_id[i]</code>	<code>ue(v)</code>
<code> if(cnt_v > 1) {</code>	
<code> num_target_output_views_minus1[i]</code>	<code>ue(v)</code>
<code> for(j = 0; j <= num_target_output_views_minus1[i]; j++)</code>	
<code> view_idx[i][j]</code>	<code>ue(v)</code>
<code> }</code>	
<code> frm_rate_info_present_flag[i]</code>	<code>u(1)</code>
<code> avg_bitrate[i]</code>	<code>u(16)</code>
<code> max_bitrate[i]</code>	<code>u(16)</code>
<code> max_bitrate_calc_window[i]</code>	<code>u(16)</code>
<code> constant_frm_rate_idc[i]</code>	<code>u(2)</code>
<code> if(cnt_v > 1) {</code>	
<code> num_directly_dependent_views[i]</code>	
<code> for(j = 0; j < num_directly_dependent_views[i]; j++) {</code>	
<code> directly_dependent_view_idx[i][j]</code>	
<code> }</code>	
<code> }</code>	
<code> if(cnt_v > 1)</code>	
<code> for(i = 1; i < cnt_v; i++) {</code>	
<code> num_ref_views[i]</code>	<code>ue(v)</code>
<code> for(j = 0; j < num_ref_views[i]; j++)</code>	

ref_view_idx[i][j]	ue(v)
}	
}	

Ví dụ về ngữ nghĩa cho các phần tử cú pháp trong bảng 4 được mô tả dưới đây:

Trong ví dụ này, num_operation_point_minus1 cộng 1 xác định số lượng điểm hoạt động tối đa có mặt trong chuỗi dữ liệu video mã hoá và cho điểm hoạt động được báo hiệu bằng các phần tử cú pháp dưới đây.

Trong ví dụ này, op_profin_level_idc[i], operation_point_id[i], priority_id[i], num_target_output_views_minus1[i], frm_rate_info_present_flag[i], avg_bitrate[i], max_bitrate[i], max_bitrate_calc_window[i], constant_frm_rate_idc[i] và num_directly_dependent_views[i] có thể có ngữ nghĩa giống như các phần tử cú pháp có cùng tên gọi trong thông báo SEI có thông tin mở rộng cấp độ cảnh nhìn theo chuẩn H.264.

Trong ví dụ này, quality_id[i] và dependency_id[i] có thể có ngữ nghĩa giống như các phần tử cú pháp có cùng tên gọi trong thông báo SEI có thông tin mở rộng cấp độ theo chuẩn H.264.

Trong ví dụ này, directly_dependent_view_idx[i][j] xác định chỉ số cảnh nhìn của cảnh nhìn thành phần thứ j mà cảnh nhìn xuất ra mục tiêu của điểm hoạt động hiện thời phụ thuộc trực tiếp vào cảnh nhìn thành phần đó trong dạng biểu diễn của điểm hoạt động hiện thời.

Trong ví dụ này, num_ref_views[i] xác định số lượng cảnh nhìn thành phần đề dự báo liên cảnh nhìn trong danh mục hình ảnh chuẩn ban đầu RefPicList0 và RefPicList1 khi giải mã các cảnh nhìn thành phần có chỉ số thứ tự cảnh nhìn bằng i. Trong ví dụ này, giá trị num_ref_views[i] không lớn hơn Min(15, num_views_minus1). Trong một số ví dụ, giá trị num_ref_views[0] bằng 0.

Trong ví dụ này, ref_view_idx[i][j] xác định chỉ số thứ tự cảnh nhìn của cảnh nhìn thành phần thứ j để dự báo liên cảnh nhìn trong danh mục hình ảnh chuẩn ban đầu RefPicList0 và RefPicList1 khi giải mã cảnh nhìn thành phần có chỉ số thứ tự cảnh nhìn bằng i. Trong ví dụ này, giá trị của ref_view_idx[i][j] sẽ nằm trong khoảng bao hàm từ 0 đến 31.

Trong một số ví dụ, theo phương án khác, một số phần tử cú pháp trong thông báo SEI chứa thông tin mở rộng cấp độ (như được mô tả trong chuẩn H.264), ví dụ, các phần tử cú pháp liên quan đến thông tin về tính phụ thuộc của lớp, có thể có mặt trong cấu trúc cú pháp `operation_points_description()` trong bảng 4.

Trong một số ví dụ, bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể mã hoá (tức là, báo hiệu) một số thông số VUI trong cấu trúc cú pháp `operation_points_description()` trong bảng 4.

Bảng 5 dưới đây thể hiện cú pháp khác cho bộ thông số video:

Bảng 5

<code>video_parameter_set_rbsp() {</code>	Bộ mô tả
<code>video_parameter_set_id</code>	<code>ue(v)</code>
<code>num_temporal_layers_minus1</code>	<code>u(3)</code>
<code>for(i = 0; i <= num_temporal_layers_minus1; i++) {</code>	
<code>profin_idc[i]</code>	<code>u(8)</code>
<code>reserved_zero_8bits[i]</code> /* equal to 0 */	<code>u(8)</code>
<code>level_idc[i]</code>	<code>u(8)</code>
<code>}</code>	
<code>bit_depth_luma_minus8</code>	<code>ue(v)</code>
<code>bit_depth_chroma_minus8</code>	<code>ue(v)</code>
<code>chroma_format_idc</code>	<code>u(2)</code>
<code>pic_width_in_luma_samples</code>	<code>ue(v)</code>
<code>pic_height_in_luma_samples</code>	<code>ue(v)</code>
<code>pic_cropping_flag</code>	<code>u(1)</code>
<code>if(pic_cropping_flag) {</code>	
<code>pic_crop_left_offset</code>	<code>ue(v)</code>
<code>pic_crop_right_offset</code>	<code>ue(v)</code>
<code>pic_crop_top_offset</code>	<code>ue(v)</code>
<code>pic_crop_bottom_offset</code>	<code>ue(v)</code>
<code>}</code>	
<code>temporal_id_nesting_flag</code>	<code>u(1)</code>
<code>bit_equal_to_one</code> /* equal to 1 */	<code>u(1)</code>

extension_type /* equal to 0 for 3DV */	ue(v)
num_layers_minus2	ue(v)
num_rep_formats_minus1	ue(v)
for(i = 1; i <= num_rep_formats_minus1; i++) {	
bit_depth_luma_minus8[i]	ue(v)
bit_depth_chroma_minus8[i]	ue(v)
chroma_format_idc[i]	u(2)
pic_width_in_luma_samples[i]	ue(v)
pic_height_in_luma_samples[i]	ue(v)
pic_cropping_flag[i]	u(1)
if(pic_cropping_flag[i]) {	
pic_crop_left_offset[i]	ue(v)
pic_crop_right_offset[i]	ue(v)
pic_crop_top_offset[i]	ue(v)
pic_crop_bottom_offset[i]	ue(v)
}	
}	
for(i = 1; i <= num_layers_minus1; i++) {	
rep_format_idx[i]	ue(v)
if(extension_type == 1) {	
dependency_id[i]	ue(v)
quality_id[i]	ue(v)
num_directly_dependent_layers[i]	ue(v)
for(j = 0; j < num_directly_dependency_layers[i]; j++)	
delta_reference_layer_id_minus1[i][j]	ue(v)
}	
}	
num_short_term_ref_pic_sets	ue(v)
for(i = 0; i < num_short_term_ref_pic_sets; i++)	
short_term_ref_pic_set(i)	
if(extension_type == 0)	
view_dependency()	
num_additional_profins_levels_minus1	ue(v)

for(i = 0; i <= num_additional_profin_levels_minus1; i++) {	
additional_profin_idc[i]	u(8)
additional_reserved_zero_8bits[i] /* equal to 0 */	u(8)
additional_level_idc[i]	u(8)
num_applicable_operation_points_minus1[i]	ue(v)
for(j = 0; j <= num_applicable_operation_points[i]; j++) {	
temporal_id[i][j]	ue(v)
layer_id[i][j]	ue(v)
if(extension_type == 0) { /* Always true for 3DV */	
depth_included_flag	u(1)
num_target_output_views_minus1[i][j]	ue(v)
for(k = 0; k < num_target_output_views_minus1[i][j]; k++)	
layer_id[i][j][k]	ue(v)
}	
else (extension_type == 1)	
layer_id[i][j]	ue(v)
}	
}	
vps_vui_parameters_present_flag	u(1)
if(vps_vui_parameters_present_flag)	
vps_vui_parameters()	
vps_extension_flag	u(1)
if(vps_extension_flag)	
while(more_rbsp_data())	
vps_extension_data_flag	u(1)
rbp_trailing_bits()	
}	

Ví dụ về ngữ nghĩa cho cú pháp của bộ thông số video trong bảng 5 được mô tả dưới đây. Nói chung, các phần tử cú pháp có tên gọi giống nhau không được mô tả dưới đây có thể có ngữ nghĩa giống như đã nêu trên dựa vào bảng 2. Ngữ nghĩa của các phần tử cú pháp khác có thể là như sau:

Trong ví dụ này, bit_equal_to_one bằng 1 (tức là, giá trị nhị phân “1”).

Trong ví dụ này, `extention_type` bằng 0 chỉ báo rằng các lớp có nhiều cảnh nhìn có thể có mặt trong dòng bit. Trong ví dụ này, `extension_type` bằng 1 xác định rằng nhiều lớp phụ thuộc và/hoặc lớp chất lượng có thể có mặt trong dòng bit.

Trong ví dụ này, `num_rep_formats_minus1` cộng 1 xác định số lượng tối đa của các tập hợp khác nhau gồm các định dạng biểu diễn được hỗ trợ bởi bộ thông số video này, định dạng biểu diễn có độ sâu bit và định dạng thành phần màu (tức là, các tập hợp của các giá trị `bit_depth_luma_minus8`, `bit_depth_chroma_minus8` và `chroma_format_idc`), độ phân giải hình ảnh và thông tin cửa sổ cắt trong chuỗi dữ liệu video mã hoá. Giá trị `num_rep_formats_minus1` có thể nằm trong khoảng bao hàm từ 0 đến X. Bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể mã hoá tập hợp gồm độ sâu bit và định dạng thành phần màu cho lớp cơ bản dựa vào `bit_depth_luma_minus8`, `bit_depth_chroma_minus8` và `chroma_format_idc`, và mã hoá các tập hợp gồm độ sâu bit và định dạng thành phần màu được báo hiệu cho các lớp nâng cao dựa vào các tập hợp của các phần tử cú pháp `bit_depth_luma_minus8[i]`, `bit_depth_chroma_minus8[i]` và `chroma_format_idc[i]`.

Bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể mã hoá tập hợp thứ nhất của định dạng biểu diễn dựa vào các phần tử cú pháp `bit_depth_luma_minus8`, `bit_depth_chroma_minus8`, `chroma_format_idc`, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`, `pic_cropping_flag`, `pic_crop_left_offset`, `pic_crop_right_offset`, `pic_crop_top_offset` và `pic_crop_bottom_offset`.

Trong ví dụ này, `bit_depth_luma_minus8[i]`, `bit_depth_chroma_minus8[i]` và `chroma_format_idc[i]` lần lượt xác định tập hợp thứ i của các giá trị `bit_depth_luma_minus8`, `bit_depth_chroma_minus8` và `chroma_format_idc` trong chuỗi dữ liệu video mã hoá.

Trong ví dụ này, `pic_width_in_luma_samples[i]` và `pic_height_in_luma_samples[i]` lần lượt xác định chiều rộng và chiều cao của mỗi hình ảnh đã giải mã tính bằng đơn vị mẫu độ chói sử dụng định dạng biểu diễn thứ i.

Trong ví dụ này, `pic_cropping_flag[i]`, `pic_crop_left_offset[i]`, `pic_crop_right_offset[i]`, `pic_crop_top_offset[i]` và `pic_crop_bottom_offset[i]` xác

định, cho tập hợp thứ i của định dạng biểu diễn, các mẫu của các hình ảnh trong chuỗi dữ liệu video mã hoá được xuất ra từ quy trình giải mã, theo vùng có dạng hình chữ nhật được xác định dựa vào các toạ độ hình ảnh xuất ra.

Trong ví dụ này, `rep_format_idx[i]` xác định chỉ số giá trị chỉ đến tập hợp gồm độ sâu bit và định dạng thành phần màu bổ sung áp dụng cho lớp có `layer_id` bằng i . Giá trị `bit_depth_luma_minus8`, `bit_depth_chroma_minus8` và `chroma_format_idc` cho lớp có `layer_id` bằng i có thể lần lượt bằng `bit_depth_luma_minus8[rep_format_idx[i]]`, `bit_depth_chroma_minus8[rep_format_idx[i]]` và `chroma_format_idc[rep_format_idx[i]]`. Giá trị `rep_format_idx[i]` sẽ nằm trong khoảng bao hàm từ 0 đến X .

Trong ví dụ này, `dependency_id[i]` xác định thông tin nhận dạng lớp phụ thuộc cho lớp có `layer_id` bằng i . `dependency_id[i]` có thể nằm trong khoảng bao hàm từ 0 đến X . Nếu không có, thì `dependency_id[i]` có thể được coi là bằng 0. Khi `num_directly_dependent_layers[i]` lớn hơn 0, thì `dependency_id[i]` có thể bằng hoặc lớn hơn thông tin nhận dạng lớp phụ thuộc của một lớp bất kỳ mà lớp có `layer_id` bằng i phụ thuộc vào đó.

Trong ví dụ này, `quality_id[i]` xác định thông tin nhận dạng chất lượng cho lớp có `layer_id` bằng i . `quality_id[i]` có thể nằm trong khoảng bao hàm từ 0 đến X . Nếu không có, thì `quality_id[i]` có thể được coi là bằng 0. Khi `num_directly_dependent_layers[i]` lớn hơn 0, thì `quality_id[i]` có thể bằng hoặc lớn hơn thông tin nhận dạng chất lượng của một lớp bất kỳ mà lớp có `layer_id` bằng i phụ thuộc vào đó và có thông tin nhận dạng lớp phụ thuộc `dependency_id[i]`.

Trong ví dụ này, `num_short_term_ref_pic_sets` xác định số lượng tập hợp hình ảnh chuẩn ngắn hạn được xác định trong bộ thông số video. Giá trị `num_short_term_ref_pic_sets` có thể nằm trong khoảng bao hàm từ 0 đến 64.

Trong ví dụ này, `depth_included_flag` bằng 1 chỉ báo rằng điểm hoạt động 3DV hiện thời có độ sâu. Trong ví dụ này, `depth_included_flag` bằng 0 chỉ báo rằng điểm hoạt động 3DV hiện thời không có độ sâu.

Ví dụ về cú pháp cho phần tử phụ thuộc vào cảnh nhìn trong bảng 5 được thể hiện trong bảng 6 dưới đây:

Bảng 6

view_dependency() {	
num_views_minus1	ue(v)
for(i = 0; i <= num_views_minus1; i++)	
view_id[i]	ue(v)
for(i = 1; i <= num_views_minus1; i++) {	
num_ref_views[i]	
for(j = 0; j < num_ref_views[i]; j++)	
ref_view_idx[i][j]	ue(v)
inter_view_texture_flag[i][j]	u(1)
}	
}	
}	

Bảng 7 dưới đây xác định tập hợp dữ liệu làm ví dụ trong đó tính phụ thuộc vào cảnh nhìn của mỗi cảnh nhìn không cơ bản được báo hiệu trực tiếp ở mức chuỗi.

Bảng 7

for(i = 1; i <= num_views_minus1; i++) {	
num_ref_views[i]	ue(v)
for(j = 0; j < num_ref_views[i]; j++)	
ref_view_idx[i][j]	ue(v)
}	

Trong ví dụ này, num_ref_views[i] xác định số lượng cảnh nhìn thành phần để dự báo liên cảnh nhìn trong danh mục hình ảnh chuẩn ban đầu RefPicList0 và RefPicList1 khi giải mã các cảnh nhìn thành phần có chỉ số thứ tự cảnh nhìn bằng i. Trong ví dụ này, giá trị num_ref_views[i] không lớn hơn Min(15, num_views_minus1). Trong ví dụ này, giá trị num_ref_views[0] bằng 0.

Trong ví dụ này, ref_view_idx[i][j] xác định chỉ số thứ tự cảnh nhìn của cảnh nhìn thành phần thứ j để dự báo liên cảnh nhìn trong danh mục hình ảnh chuẩn ban đầu RefPicList0 và RefPicList1 khi giải mã cảnh nhìn thành phần có chỉ số thứ tự cảnh nhìn bằng i. Trong ví dụ này, giá trị ref_view_idx[i][j] nằm trong khoảng bao hàm từ 0 đến 31.

Như đã nêu trên, một loại đơn vị NAL cụ thể (ví dụ, đơn vị NAL loại 10) có thể được dùng để đóng gói bộ thông số video. Cú pháp của đơn vị NAL có thể được sửa đổi như được thể hiện trong ví dụ nêu trong bảng 8 dưới đây.

Bảng 8

<code>nal_unit(NumBytesInNALunit) {</code>	Bộ mô tả
forbidden_zero_bit	f(1)
nal_ref_flag	u(1)
nal_unit_type	u(6)
<code>NumBytesInRBSP = 0</code>	
<code>nalUnitHeaderBytes = 1</code>	
<code>if(nal_unit_type != 10) { // not VPS NAL unit</code>	
<code>if(cnt_p > 1)</code>	
priority_id	u(v)
<code>if(cnt_t > 1)</code>	
temporal_id	u(v)
reserved_one_bit	u(1)
<code>if(cnt_d > 1)</code>	
dependency_id	u(v)
<code>if(cnt_q > 1)</code>	
quality_id	u(v)
reserved_one_bit	u(1)
<code>if(cnt_v > 1)</code>	
view_idx	u(v)
<code>if(cnt_f)</code>	
reserved_flags	u(v)
<code>m = Ceil(log2(cnt_p)) + Ceil(log2(cnt_t)) + Ceil(log2(cnt_d)) + Ceil(log2(cnt_q)) + Ceil(log2(cnt_v)) + cnt_f + 2</code>	
<code>if(((m + 7 >> 3) << 3) - m)</code>	
reserved_bits	u(v)
<code>nalUnitHeaderBytes += ((m + 7) >> 3)</code>	
<code>}</code>	
<code>for(i = nalUnitHeaderBytes; i < NumBytesInNALunit; i++) {</code>	
<code>if(i + 2 < NumBytesInNALunit && next_bits(24) == 0x000003) {</code>	

rbsp_byte [NumBytesInRBSP++]	b(8)
rbsp_byte [NumBytesInRBSP++]	b(8)
i += 2	
emulation_prevention_three_byte /* equal to 0x03 */	f(8)
} else	
rbsp_byte [NumBytesInRBSP++]	b(8)
}	
}	

Trong ví dụ này, các phần tử trong câu lệnh “if(nal_unit_type != 10)” được bổ sung thêm, so với cú pháp đơn vị NAL thông thường. Trong ví dụ này, số lượng bit được dùng để báo hiệu các phần tử cú pháp *priority_id*, *temporal_id*, *dependency_id*, *quality_id* và *view_idx* lần lượt là $\text{Ceil}(\log_2(\text{cnt}_p))$, $\text{Ceil}(\log_2(\text{cnt}_t))$, $\text{Ceil}(\log_2(\text{cnt}_d))$, $\text{Ceil}(\log_2(\text{cnt}_q))$ và $\text{Ceil}(\log_2(\text{cnt}_v))$. Tương tự như vậy, trong ví dụ này, khi một phần tử cú pháp trong số các phần tử cú pháp *priority_id*, *temporal_id*, *dependency_id*, *quality_id* và *view_idx* không có mặt, thì giá trị của phần tử cú pháp đó được coi là bằng 0.

Ngoại trừ trường hợp như đã nêu trên liên quan đến số lượng bit và các giá trị suy luận có thể tìm được, ngữ nghĩa cho các phần tử cú pháp trong bảng 8 có thể được xác định như sau. Ngữ nghĩa của *priority_id*, *dependency_id* và *quality_id* có thể được xác định như trong phiên bản mở rộng SVC của chuẩn ITU-T H.264/AVC. Ngữ nghĩa của *temporal_id* có thể được xác định như trong phiên bản WD4 của chuẩn HEVC. Trong ví dụ này, *reserved_one_bit* bằng 1. Giá trị 0 cho *reserved_one_bit* có thể được xác định theo các phiên bản mở rộng sau này của chuẩn HEVC. Các bộ giải mã, như bộ giải mã dữ liệu video 30, có thể được tạo cấu hình để bỏ qua giá trị của *reserved_one_bit*.

Trong ví dụ này, *view_idx* xác định chỉ số thứ tự cảnh nhìn cho một cảnh nhìn. Ngữ nghĩa của *view_idx* có thể giống như phần tử cú pháp “chỉ số thứ tự cảnh nhìn” như được xác định trong phiên bản mở rộng MVC của chuẩn ITU-T H.264/AVC.

Trong ví dụ này, mỗi bit của *reserved_flags* đều bằng 1. Các giá trị khác cho *reserved_flags* có thể được xác định theo các phiên bản mở rộng sau này của chuẩn HEVC sắp tới. Các bộ giải mã, như bộ giải mã dữ liệu video 30, có thể được tạo cấu

hình để bỏ qua giá trị của `reserved_flags`, trừ trường hợp được tạo cấu hình để hoạt động theo phiên bản mở rộng phân định ngữ nghĩa cho các bit của `reserved_flags`. Trong ví dụ này, số lượng bit được dùng để biểu diễn `reserved_flags` bằng `reserved_flags_len`.

Trong ví dụ này, mỗi bit của `reserved_bits` đều bằng 1. Các giá trị khác cho `reserved_bits` có thể được xác định theo các phiên bản mở rộng sau này của chuẩn HEVC sắp tới. Các bộ giải mã, như bộ giải mã dữ liệu video 30, có thể được tạo cấu hình để bỏ qua giá trị của `reserved_bits`, xin nhắc lại là, trừ trường hợp được tạo cấu hình để hoạt động theo một phiên bản mở rộng sau này. Số lượng bit được dùng để biểu diễn `reserved_bits`, trong ví dụ này, bằng $((m+7 \gg 3) \ll 3) - m$.

Bảng 9 dưới đây thể hiện ví dụ về cú pháp cho bộ thông số lớp. Cú pháp này có thể được sử dụng cho mọi bộ thông số LPS trên Fig.5, trong một số ví dụ.

Bảng 9

<code>layer_para_set() {</code>	Bộ mô tả
<code>depth_flag</code>	<code>u(1)</code>
<code>layer_para_set_id</code>	<code>ue(v)</code>
<code>vps_id</code>	<code>ue(v)</code>
<code>// cu hierarchy { {</code>	
<code>log2_min_coding_block_size_minus3</code>	<code>ue(v)</code>
<code>log2_diff_max_min_coding_block_size</code>	<code>ue(v)</code>
<code>log2_min_transform_block_size_minus2</code>	<code>ue(v)</code>
<code>log2_diff_max_min_transform_block_size</code>	<code>ue(v)</code>
<code>log2_min_pcm_coding_block_size_minus3</code>	<code>ue(v)</code>
<code>max_transform_hierarchy_depth_inter</code>	<code>ue(v)</code>
<code>max_transform_hierarchy_depth_intra</code>	<code>ue(v)</code>
<code>// cu hierarchy} }</code>	
<code>pcm_bit_depth_luma_minus1</code>	<code>u(4)</code>
<code>pcm_bit_depth_chroma_minus1</code>	<code>u(4)</code>
<code>loop_filter_across_slice_flag</code>	<code>u(1)</code>
<code>sample_adaptive_offset_enabled_flag</code>	<code>u(1)</code>
<code>adaptive_loop_filter_enabled_flag</code>	<code>u(1)</code>

pcm_loop_filter_disable_flag	u(1)
cu_qp_delta_enabled_flag	u(1)
//tiles	
num_tile_columns_minus1	ue(v)
num_tile_columns_minus1	ue(v)
if(num_tile_columns_minus1 != 0 num_tile_rows_minus1 != 0) {	
tile_boundary_independence_idc	u(1)
uniform_spacing_idc	u(1)
if(uniform_spacing_idc != 1) {	
for(i=0; i<num_tile_columns_minus1 ; i++)	
column_width[i]	ue(v)
for(i=0; i < num_tile_rows_minus1; i++)	
row_height[i]	ue(v)
}	
}	
lps_extension_flag	u(1)
if(lps_extension_flag)	
while(more_rbsp_data())	
lps_extension_data_flag	u(1)
rbp_trailing_bits()	
}	

Ví dụ về ngữ nghĩa của cú pháp bộ thông số LPS trong bảng 9 được mô tả dưới đây. Các lớp khác nhau (ví dụ, các cảnh nhìn khác nhau theo phiên bản mở rộng MVC hoặc các lớp khác nhau theo phiên bản mở rộng SVC) có thể tham chiếu đến các bộ thông số LPS khác nhau. Các lớp chất lượng khác nhau trong cùng một lớp phụ thuộc có thể dùng chung cùng một bộ thông số LPS. Các lớp thời gian khác nhau trong cùng một lớp phụ thuộc có thể dùng chung cùng một bộ thông số LPS. Theo cách khác, các cảnh nhìn khác nhau có thể tham chiếu đến cùng một bộ thông số LPS, và các lớp phụ thuộc khác nhau có thể tham chiếu đến cùng một bộ thông số LPS.

Trong ví dụ này, depth_flag bằng 1 xác định rằng bộ thông số LPS áp dụng cho các dạng biểu diễn số sâu được xác định dựa vào các giá trị temporal_id, dependency_id, quality_id và view_idx của đơn vị NAL có bộ thông số LPS.

Depth_flag bằng 0 xác định rằng bộ thông số LPS áp dụng cho các dạng biểu diễn cấu trúc được xác định dựa vào các giá trị temporal_id, dependency_id, quality_id và view_idx của đơn vị NAL có bộ thông số LPS.

Trong ví dụ này, layer_para_set_id xác định id của bộ thông số lớp (LPS) hiện thời. Các bộ thông số lớp khác nhau có các giá trị dependency_id và view_idx tương ứng bằng nhau dùng chung một khoảng giá trị cho layer_para_set_id, có nghĩa là các bộ thông số LPS khác nhau với tổ hợp dependency_id và view_idx khác nhau có thể có cùng một giá trị layer_para_set_id.

Theo cách khác, tất cả các bộ thông số LPS có thể dùng chung một khoảng giá trị, có nghĩa là mỗi LPS có một giá trị layer_para_set_id riêng biệt.

Trong ví dụ này, vps_id xác định bộ thông số video mà bộ thông số lớp này tham chiếu đến đó.

Trong ví dụ này, lps_extension_flag bằng 0 xác định rằng không có phần tử cú pháp lps_extension_data_flag trong cấu trúc cú pháp RBSP cho bộ thông số lớp. Trong ví dụ này, lps_extension_flag có thể bằng 0 trong các dòng bit tuân theo chuẩn HEVC sắp tới. Giá trị bằng 1 của lps_extension_flag được dành riêng để sau này sử dụng cho ITU-T | ISO/IEC. Các bộ giải mã, như bộ giải mã dữ liệu video 30, có thể bỏ qua tất cả các dữ liệu đứng sau giá trị bằng 1 của lps_extension_flag trong đơn vị NAL có bộ thông số lớp.

Trong ví dụ này, lps_extension_data_flag có thể có giá trị bất kỳ, và không ảnh hưởng đến sự phù hợp với các profin được xác định theo chuẩn HEVC sắp tới.

Các phần tử cú pháp khác có thể có ngữ nghĩa giống như các phần tử cú pháp có cùng tên gọi trong bộ thông số SPS theo chuẩn HEVC WD, nhưng chỉ áp dụng cho các hình ảnh tham chiếu đến bộ thông số LPS này.

Bộ thông số LPS có thể được đưa vào trong đơn vị NAL, phần đầu của đơn vị NAL này có thể được xác định theo bảng 8 trên đây. Các phần tử cú pháp dưới đây có ngữ nghĩa được sửa đổi đôi chút khi chúng được liên hệ với bộ thông số LPS.

Trong ví dụ này, priority_id bằng giá trị nhỏ nhất trong số các giá trị priority_id của tất cả các đơn vị NAL tham chiếu đến bộ thông số LPS này.

Trong ví dụ này, `temporal_id` bằng giá trị nhỏ nhất trong số các giá trị `temporal_id` của tất cả các đơn vị NAL tham chiếu đến bộ thông số LPS này.

Trong ví dụ này, `dependency_id` bằng `dependency_id` của tất cả các đơn vị NAL tham chiếu đến bộ thông số LPS này.

Trong ví dụ này, `quality_id` bằng giá trị nhỏ nhất trong số các giá trị `quality_id` của tất cả các đơn vị NAL tham chiếu đến bộ thông số LPS này.

Trong ví dụ này, `v_idx` là chỉ số cảnh nhìn của bộ thông số LPS. Tất cả các hình ảnh tham chiếu đến bộ thông số LPS này có thể đều có id cảnh nhìn bằng `view_id[v_idx]`.

Theo cách khác, các phần tử cú pháp nêu trên có thể được báo hiệu trực tiếp trong bảng cú pháp cho bộ thông số lớp, như được thể hiện trong ví dụ nêu trong bảng 10. Bảng cú pháp chi tiết hơn có thể được xác định theo bảng 10 dưới đây. Trong trường hợp này, các phần tử cú pháp không có trong phần đầu đơn vị NAL có bộ thông số LPS và việc phân tích cú pháp cho bộ thông số LPS có thể phụ thuộc vào bộ thông số VPS có ID bằng `vps_id`.

Bảng 10

<code>layer_para_set() {</code>	Bộ mô tả
<code>vps_id</code>	<code>u(8)</code>
<code>if(cnt_p > 1)</code>	
<code>priority_id</code>	<code>u(v)</code>
<code>if(cnt_t > 1)</code>	
<code>temporal_id</code>	<code>u(v)</code>
<code>reserved_one_bit</code>	<code>u(1)</code>
<code>if(cnt_d > 1)</code>	
<code>dependency_id</code>	<code>u(v)</code>
<code>if(cnt_q > 1)</code>	
<code>quality_id</code>	<code>u(v)</code>
<code>if(cnt_v > 1)</code>	
<code>view_idx</code>	<code>u(v)</code>
<code>depth_flag</code>	<code>u(1)</code>
<code>layer_para_set_id</code>	<code>ue(v)</code>

// cu hierarchy {{	
...	
rbsp_trailing_bits()	
}	

Bộ thông số LPS, trong trường hợp này, không cần phải có phần đầu đơn vị NAL sao chép các phần tử cú pháp nêu trên. Giả sử loại đơn vị NAL của đơn vị NAL đóng gói bộ thông số LPS là, ví dụ, loại 5, thì cú pháp phần đầu đơn vị NAL có thể được sửa đổi đôi chút như được thể hiện trong bảng 11, tức là bổ sung ngoại lệ “&& nal_unit_type != 5” vào câu lệnh “if” trong bảng 8:

Bảng 11

nal_unit(NumBytesInNALunit) {	Bộ mô tả
forbidden_zero_bit	f(1)
nal_ref_flag	u(1)
nal_unit_type	u(6)
NumBytesInRBSP = 0	
nalUnitHeaderBytes = 1	
if(nal_unit_type != 10 && nal_unit_type != 5) { // not VPS NAL unit	
if(cnt_p > 1)	
priority_id	u(v)
if(cnt_t > 1)	
temporal_id	u(v)
reserved_one_bit	u(1)
if(cnt_d > 1)	
dependency_id	u(v)
if(cnt_q > 1)	
quality_id	u(v)
reserved_one_bit	u(1)
if(cnt_v > 1)	
view_idx	u(v)
if(cnt_f)	
reserved_flags	u(v)

$m = \text{Ceil}(\log_2(\text{cnt_p})) + \text{Ceil}(\log_2(\text{cnt_t})) + \text{Ceil}(\log_2(\text{cnt_d})) + \text{Ceil}(\log_2(\text{cnt_q})) + \text{Ceil}(\log_2(\text{cnt_v})) + \text{cnt_f} + 2$	
$\text{if}(((m + 7) \gg 3) \ll 3) - m)$	
reserved_bits	u(v)
$\text{nalUnitHeaderBytes} += ((m + 7) \gg 3)$	
}	
for(i = nalUnitHeaderBytes; i < NumBytesInNALunit; i++) {	
if(i + 2 < NumBytesInNALunit && next_bits(24) == 0x000003) {	
rbsp_byte [NumBytesInRBSP++]	b(8)
rbsp_byte [NumBytesInRBSP++]	b(8)
i += 2	
emulation_prevention_three_byte /* equal to 0x03 */	f(8)
} else	
rbsp_byte [NumBytesInRBSP++]	b(8)
}	
}	

Theo các phương án khác làm ví dụ, bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể mã hoá các phần tử cú pháp liên quan đến các đặc trưng có thể mở rộng cấp độ bằng cách sử dụng phương pháp mã hoá độ dài cố định, như được thể hiện trong ví dụ nêu trong bảng 12 dưới đây.

Bảng 12

layer_para_set() {	Bộ mô tả
vps_id	u(8)
priority_id	u(5)
temporal_id	u(3)
dependency_id	u(3)
quality_id	u(3)
view_idx	u(8)
layer_para_set_id	ue(v)
// cu hierarchy { {	
}	

Bảng 13 dưới đây thể hiện ví dụ về cú pháp cho bộ thông số hình ảnh (PPS)

theo các kỹ thuật nêu trong sáng chế. Trong ví dụ này, bộ thông số hình ảnh không cần báo hiệu “seq_parameter_set_id”, khác với bộ thông số PPS theo chuẩn HEVC thông thường.

Bảng 13

pic_parameter_set_rbsp() {	Bộ mô tả
pic_parameter_set_id	ue(v)
...	
pps_extension_flag	u(1)
if(pps_extension_flag)	
while(more_rbsp_data())	
pps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

Ví dụ về ngữ nghĩa của bộ thông số PPS trong bảng 13 được mô tả dưới đây.

Trong ví dụ này, pps_extension_flag bằng 0 xác định rằng không có phần tử cú pháp pps_extension_data_flag trong cấu trúc cú pháp RBSP cho bộ thông số hình ảnh. Trong ví dụ này, pps_extension_flag bằng 0 trong các dòng bit tuân theo chuẩn HEVC sắp tới. Giá trị bằng 1 của pps_extension_flag có thể được dành riêng để sau này sử dụng cho ITU-T | ISO/IEC. Các bộ giải mã, như bộ giải mã dữ liệu video 30, có thể bỏ qua tất cả các dữ liệu đứng sau giá trị bằng 1 của pps_extension_flag trong đơn vị NAL có bộ thông số hình ảnh.

Trong ví dụ này, pps_extension_data_flag có thể có giá trị bất kỳ. Nó không ảnh hưởng đến sự phù hợp với các profin được xác định theo chuẩn HEVC sắp tới. Ngữ nghĩa của các giá trị pps_extension_data_flag có thể được phân định trong các phiên bản phát triển sau này của chuẩn HEVC hoặc các phiên bản mở rộng của chuẩn này mà không xung đột với các kỹ thuật nêu trong sáng chế.

Theo các kỹ thuật nêu trong sáng chế, id của bộ thông số chuỗi hoặc id của bộ thông số lớp không cần phải được báo hiệu trong bộ thông số PPS. Một số phần tử cú pháp khác trong bộ thông số PPS có thể được chuyển vào bộ thông số LPS. Có nghĩa là, bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo

cấu hình để mã hoá một hoặc nhiều bộ thông số LPS chứa dữ liệu giống như các phần tử cú pháp được thể hiện trong bảng 13.

Fig.6 là sơ đồ khái niệm thể hiện ví dụ về bộ thông số nhóm (GPS) và các quan hệ của bộ thông số GPS với các bộ thông số khác và các phần đầu lát. Trong ví dụ này, các bộ thông số khác là bộ thông số LPS, bộ thông số SPS, bộ thông số PPS, bộ thông số thích ứng (APS) loại 0 (ví dụ, bộ thông số APS báo hiệu các thông số bộ lọc vòng lặp thích ứng (*ALF: Adaptive Loop Filter*)), bộ thông số APS loại 1 (ví dụ, bộ thông số APS báo hiệu ma trận lượng tử hoá), và các bộ thông số khác. Trong ví dụ này, bộ thông số GPS có nhiều nhóm khác nhau, mỗi nhóm có GPS ID duy nhất (còn được gọi là ID nhóm), trong đó mỗi nhóm chỉ báo một bộ thông số cụ thể trong số nhiều bộ thông số khác nhau dựa vào ID bộ thông số. Theo cách này, các phần đầu lát chỉ cần xác định `group_id` để xác định mỗi bộ thông số tương ứng với nhóm có `group_id` đó.

Bảng 14 và bảng 15 dưới đây thể hiện ví dụ khác về cấu trúc cú pháp RBSP cho bộ thông số nhóm.

Bảng 14

<code>group_para_set_rbsp() {</code>	Bộ mô tả
<code> number_signaled_para_set_groups_minus1</code>	<code>ue(v)</code>
<code> for(i = 0; i <= number_signaled_para_set_groups_minus1; i++) {</code>	
<code> para_set_group_id[i]</code>	<code>ue(v)</code>
<code> lps_id[i]</code>	<code>ue(v)</code>
<code> pps_id[i]</code>	<code>ue(v)</code>
<code> for(j = 0; j < numParaSetTypes; j++)</code>	
<code> para_set_type_id[i][j]</code>	
<code> }</code>	
<code> gps_extension_flag</code>	<code>u(1)</code>
<code> if(gps_extension_flag)</code>	
<code> while(more_rbsp_data())</code>	
<code> gps_extension_data_flag</code>	<code>u(1)</code>
<code> rbsp_trailing_bits()</code>	
<code>}</code>	

Bảng 15

group_para_set_rbsp() {	Bộ mô tả
number_signaled_para_set_groups_minus1	ue(v)
for(i = 0; i <= number_signaled_para_set_groups_minus1; i++) {	
para_set_group_id[i]	ue(v)
lps_id[i]	ue(v)
pps_id[i]	ue(v)
aps_id[i]	ue(v)
}	
gps_extension_flag	u(1)
if(gps_extension_flag)	
while(more_rbsp_data())	
gps_extension_data_flag	u(1)
rbp_trailing_bits()	
}	

Các bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30, có thể được tạo cấu hình để mã hoá bộ thông số nhóm, ví dụ, theo bảng 14 hoặc bảng 15. Ví dụ về ngữ nghĩa cho cú pháp của các bộ thông số nhóm được thể hiện dưới đây.

Trong ví dụ này, **number_signaled_para_set_groups_minus1** cộng 1 xác định số lượng nhóm thông số được báo hiệu. Giá trị này có thể nằm trong khoảng bao hàm từ 0 đến 30.

Trong ví dụ này, **para_set_group_id[i]** xác định id của nhóm bộ thông số thứ i được báo hiệu. Giá trị **para_set_group_id[i]** sẽ nằm trong khoảng bao hàm từ 0 đến 31.

Trong ví dụ này, **para_set_type_id[i][j]** xác định id của loại bộ thông số thứ j cho nhóm bộ thông số thứ i.

Trong ví dụ này, **lps_id[i]** chỉ báo id của bộ thông số lớp được tham chiếu bởi nhóm bộ thông số có id nhóm bằng **para_set_group_id[i]**. Các giá trị **dependency_id** và **view_idx** của bộ thông số LPS có **layer_para_set_id** bằng **lps_id[i]** có thể bằng các giá trị **dependency_id** và **view_idx** tương ứng của đơn vị NAL có nhóm bộ thông số.

Các giá trị `dependency_id` và `view_idx` của cấu trúc cú pháp RBSP cho nhóm bộ thông số có mặt trong phần đầu đơn vị NAL của cấu trúc cú pháp RBSP này trong ví dụ được thể hiện trong bảng 14 và bảng 15, và các giá trị `dependency_id` và `view_idx` của bộ thông số LPS có thể có mặt trong phần đầu đơn vị NAL có bộ thông số LPS này hoặc trong bảng cú pháp của bộ thông số LPS.

Theo cách khác, các giá trị `dependency_id` và `view_idx` của bộ thông số LPS có `layer_para_set_id` bằng `lps_id[i]` có thể không bằng các giá trị `dependency_id` và `view_idx` tương ứng của đơn vị NAL có nhóm bộ thông số.

Trong ví dụ này, `pps_id[i]` chỉ báo id của bộ thông số hình ảnh được tham chiếu bởi nhóm bộ thông số có id nhóm bằng `para_set_group_id[i]`.

Trong ví dụ này, `aps_id[i]` chỉ báo id của bộ thông số thích ứng được tham chiếu bởi nhóm bộ thông số có id nhóm bằng `para_set_group_id[i]`.

Trong ví dụ này, `gps_extension_flag` bằng 0 xác định rằng không có phần tử cú pháp `gps_extension_data_flag` trong cấu trúc cú pháp RBSP cho nhóm bộ thông số. `Gps_extension_flag` có thể bằng 0 trong các dòng bit tuân theo chuẩn HEVC sắp tới. Giá trị bằng 1 của `gps_extension_flag` có thể được dành riêng để sau này sử dụng cho ITU-T | ISO/IEC. Các bộ giải mã, như bộ giải mã dữ liệu video 30, có thể bỏ qua tất cả các dữ liệu đứng sau giá trị bằng 1 của `gps_extension_flag` trong đơn vị NAL có nhóm bộ thông số. Nói chung, `gps_extension_data_flag` có thể có giá trị bất kỳ. Nó không ảnh hưởng đến sự phù hợp với các profin được xác định theo chuẩn HEVC sắp tới.

Trong một số ví dụ, `para_set_type_id[i][j]` có thể là `aps_id[i][j]`, với ngữ nghĩa giống như `aps_id[i]` như đã nêu trên.

Như được thể hiện trên Fig.6, thay vì tham chiếu đến ID bộ thông số hình ảnh trong phần đầu lát, theo các kỹ thuật nêu trong sáng chế, phần đầu lát có thể tham chiếu đến ID nhóm thông số, do đó tham chiếu gián tiếp đến bộ thông số LPS, bộ thông số PPS và bộ thông số APS thuộc mỗi loại (ví dụ, các bộ thông số APS cung cấp các thông số ALF và ma trận lượng tử hoá).

Bộ mã hoá dữ liệu video có thể kích hoạt bộ thông số video hoặc bộ thông số lớp khi đơn vị NAL có lớp VCL (chứa lát mã hoá) tham chiếu bộ thông số, theo cách

gián tiếp, ví dụ, dựa vào nguyên tắc thiết kế chuẩn H.264/AVC.

Trong một số ví dụ, các bộ thông số có thể được kích hoạt bằng một loại đơn vị NAL cụ thể, chứ không phải bằng lát mã hoá. Ví dụ, đơn vị NAL thuộc loại cụ thể (đơn vị NAL kích hoạt các bộ thông số), nếu có trong dòng bit, có thể kích hoạt một, và chỉ một, bộ thông số VPS. Theo cách khác hoặc theo cách bổ sung, loại đơn vị NAL như vậy có thể kích hoạt ít nhất một bộ thông số LPS. Loại đơn vị NAL như vậy có thể còn kích hoạt ít nhất một bộ thông số PPS. Loại đơn vị NAL như vậy có thể còn kích hoạt ít nhất một bộ thông số APS. Đơn vị NAL kích hoạt các bộ thông số có thể là cấu trúc cú pháp RBSP cho bộ thông số nhóm. Đơn vị NAL kích hoạt các bộ thông số (*PSA: Parameter Sets Activation*) có thể áp dụng cho một chuỗi dữ liệu video mã hoá. Đơn vị NAL PSA có thể được coi là đơn vị NAL không có lớp VCL, có nghĩa là, gián tiếp phù hợp với bộ mã hoá dữ liệu video. Cú pháp phân đầu đơn vị NAL của đơn vị NAL PSA có thể giống với cú pháp phân đầu đơn vị NAL của đơn vị NAL có bộ thông số VPS.

Trong một số ví dụ, đơn vị NAL PSA, nếu có trong đơn vị truy nhập, có thể đứng trước đơn vị NAL có lớp VCL thứ nhất của đơn vị truy nhập. Có thể có ít nhất một đơn vị NAL PSA trong đơn vị truy nhập thứ nhất của chuỗi dữ liệu video mã hoá, ví dụ, hình ảnh IDR. Nhiều đơn vị NAL PSA trong cùng một chuỗi dữ liệu video mã hoá có thể chứa cùng một id bộ thông số VPS; do đó, không cần kích hoạt các bộ thông số video khác nhau trong cùng một chuỗi dữ liệu video mã hoá. Đơn vị NAL PSA, nếu có trong đơn vị truy nhập, có thể đứng trước mọi đơn vị NAL chứa LPS, PPS, APS, hoặc SEI, nếu có. Đơn vị NAL có bộ thông số VPS, nếu có trong đơn vị truy nhập, có thể đứng trước đơn vị NAL có LPS, PPS, APS, hoặc SEI, nếu có. Theo cách khác hoặc theo cách bổ sung, đơn vị NAL PSA, nếu có trong đơn vị truy nhập, có thể đứng trước đơn vị NAL có bộ thông số VPS, nếu có.

Trong một số ví dụ, các bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30, có thể được tạo cấu hình để sử dụng cú pháp trong bảng 16 cho bộ thông số chuỗi (SPS), khác với cú pháp cho bộ thông số SPS thông thường, ví dụ, theo chuẩn HEVC.

Bảng 16

seq_parameter_set_rbsp() {	Bộ mô tả
video_parameter_set_id	ue(v)
rep_format_idx	ue(v)
pcm_enabled_flag	u(1)
if(pcm_enabled_flag) {	
pcm_bit_depth_luma_minus1	u(4)
pcm_bit_depth_chroma_minus1	u(4)
}	
qpprime_y_zero_transquant_bypass_flag	u(1)
log2_max_pic_order_cnt_lsb_minus4	ue(v)
for(i = 0; i <= max_temporal_layers_minus1; i++) {	
max_dec_pic_buffering[i]	ue(v)
num_reorder_pics[i]	ue(v)
max_latency_increase[i]	ue(v)
}	
restricted_ref_pic_lists_flag	<u>u(1)</u>
if(restricted_ref_pic_lists_flag)	
lists_modification_present_flag	<u>u(1)</u>
log2_min_coding_block_size_minus3	ue(v)
log2_diff_max_min_coding_block_size	ue(v)
log2_min_transform_block_size_minus2	ue(v)
log2_diff_max_min_transform_block_size	ue(v)
if(pcm_enabled_flag) {	
log2_min_pcm_coding_block_size_minus3	ue(v)
log2_diff_max_min_pcm_coding_block_size	ue(v)
}	
max_transform_hierarchy_depth_inter	ue(v)
max_transform_hierarchy_depth_intra	ue(v)
scaling_list_enable_flag	
chroma_pred_from_luma_enabled_flag	u(1)
deblocking_filter_in_aps_enabled_flag	u(1)
seq_loop_filter_across_slices_enabled_flag	u(1)
asymmetric_motion_partitions_enabled_flag	u(1)

non_square_quadtrees_enabled_flag	u(1)
sample_adaptive_offset_enabled_flag	u(1)
adaptive_loop_filter_enabled_flag	u(1)
if(adaptive_loop_filter_enabled_flag)	
alf_coef_in_slice_flag	u(1)
if(pcm_enabled_flag)	
pcm_loop_filter_disable_flag	u(1)
temporal_id_nesting_flag	u(1)
if(log2_min_coding_block_size_minus3 == 0)	
inter_4x4_enabled_flag	u(1)
long_term_ref_pics_present_flag	u(1)
tiles_or_entropy_coding_sync_idc	u(2)
if(tiles_or_entropy_coding_sync_idc == 1) {	
num_tile_columns_minus1	ue(v)
num_tile_rows_minus1	ue(v)
uniform_spacing_flag	u(1)
if(!uniform_spacing_flag) {	
for(i = 0; i < num_tile_columns_minus1; i++)	
column_width[i]	ue(v)
for(i = 0; i < num_tile_rows_minus1; i++)	
row_height[i]	ue(v)
}	
loop_filter_across_tiles_enabled_flag	u(1)
}	
vui_parameters_present_flag	u(1)
if(vui_parameters_present_flag)	
vui_parameters()	
sps_extension_flag	u(1)
if(sps_extension_flag)	
while(more_rbsp_data())	
sps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

Bộ thông số SPS làm ví dụ trong bảng 16 loại bỏ profin_idc,

reserved_zero_8bits, level_idc, chroma_format_idc, separate_colour_plane_flag và câu lệnh điều kiện “if” tương ứng, max_temporal_layers_minus1, pic_width_in_luma_samples, pic_height_in_luma_samples, pic_cropping_flag, pic_crop_left_offset, pic_crop_right_offset, pic_crop_top_offset, và pic_crop_bottom_offset và câu lệnh điều kiện “if” tương ứng, bit_depth_luma_minus8, bit_depth_chroma_minus8, num_short_term_ref_pic_sets, và short_term_ref_pic_set(i) và câu lệnh điều kiện “if” tương ứng ra khỏi cú pháp cho bộ thông số SPS thông thường. Ngoài ra, ví dụ về bộ thông số SPS trong bảng 16 bổ sung thêm video_parameter_set_id và rep_format_idx. Ngữ nghĩa của các phần tử cú pháp còn lại có thể giống như được xác định theo chuẩn HEVC thông thường. Ngữ nghĩa của các phần tử bổ sung video_parameter_set_id và rep_format_idx có thể được xác định như sau:

Trong ví dụ này, video_parameter_set_id xác định bộ thông số video (VPS) được tham chiếu bằng bộ thông số SPS hiện thời. Theo cách khác, không cần báo hiệu video_parameter_set_id, và bộ thông số GPS có thể được dùng để liên kết bộ thông số SPS với một bộ thông số VPS cụ thể.

Trong ví dụ này, rep_format_idx xác định chỉ số chỉ đến định dạng biểu diễn được báo hiệu trong bộ thông số video được tham chiếu.

Theo phương án khác nữa, bảng 17 thể hiện một ví dụ khác về cú pháp cho bộ thông số nhóm. Giả sử rằng, trong ví dụ này, phần tử cú pháp ID bộ thông số video không có mặt trong cú pháp cho bộ thông số SPS, như đã nêu trên.

Bảng 17

group_para_set_rbsp() {	Bộ mô tả
gps_id	ue(v)
vps_id	ue(v)
sps_id	ue(v)
pps_id	ue(v)
num_ref_aps_ids	ue(v)
for(i = 0; i < num_ref_aps_ids; i++) {	
ref_aps_id[i]	ue(v)
ref_aps_param_type[i]	ue(v)

}	
gps_extension_flag	u(1)
if(gps_extension_flag)	
while(more_rbsp_data())	
gps_extension_data_flag	u(1)
rbp_trailing_bits()	
}	

Ngữ nghĩa cho các phần tử cú pháp trong bảng 17 có thể được xác định như sau:

Trong ví dụ này, gps_id xác định thông tin nhận dạng của bộ thông số nhóm (GPS).

Trong ví dụ này, vps_id xác định thông tin nhận dạng của bộ thông số video được tham chiếu bởi bộ thông số GPS.

Trong ví dụ này, sps_id xác định thông tin nhận dạng của bộ thông số chuỗi được tham chiếu bởi bộ thông số GPS.

Trong ví dụ này, pps_id xác định thông tin nhận dạng của bộ thông số chuỗi hình ảnh được tham chiếu bởi bộ thông số GPS.

Trong ví dụ này, num_ref_aps_ids xác định số lượng phần tử cú pháp ref_aps_id[i]. Giá trị num_ref_aps_ids sẽ nằm trong khoảng bao hàm từ 0 đến 4.

Trong ví dụ này, ref_aps_id[i] xác định bộ thông số thích ứng thứ i được tham chiếu bởi bộ thông số nhóm.

Một giá trị của ref_aps_id[i] có thể xuất hiện trong vòng lặp nhiều hơn một lần, và do đó, nhiều hơn một loại của các thông số APS từ cùng một bộ thông số APS có thể được tham chiếu bằng cùng một bộ thông số GPS và có thể áp dụng cho lát mã hoá tham chiếu đến bộ thông số GPS.

Trong ví dụ này, ref_aps_param_type[i] xác định loại của các thông số APS có trong bộ thông số thích ứng thứ i được tham chiếu bởi bộ thông số nhóm. Giá trị của ref_aps_param_type[i] có thể nằm trong khoảng bao hàm từ 0 đến 3. Các giá trị bao hàm từ 0 đến 3 cho ref_aps_param_type[i] lần lượt tương ứng với các loại thông số APS của danh mục định tỷ lệ, bộ lọc tách khối, độ lệch thích ứng mẫu (*SAO: Sample Adaptive Offset*) và ALF. Giá trị ref_aps_param_type[i] với hai giá trị i bất kỳ khác

nhau sẽ không giống nhau, trong một số ví dụ.

Trong ví dụ này, cờ `gps_extension_flag` bằng 0 xác định rằng không có phần tử cú pháp `gps_extension_data_flag` trong cấu trúc cú pháp RBSP cho nhóm bộ thông số. `gps_extension_flag` có thể bằng 0 trong các dòng bit tuân theo chuẩn HEVC sắp tới. Giá trị bằng 1 của `gps_extension_flag` có thể được dành riêng để sau này sử dụng cho ITU-T | ISO/IEC. Các bộ giải mã, như bộ giải mã dữ liệu video 30, có thể bỏ qua tất cả các dữ liệu đứng sau giá trị bằng 1 của `gps_extension_flag` trong đơn vị NAL có nhóm bộ thông số.

Trong ví dụ này, `gps_extension_data_flag` có thể có giá trị bất kỳ. Nó không ảnh hưởng đến sự phù hợp với các profile được xác định theo chuẩn HEVC sắp tới.

Các bộ mã hoá dữ liệu video, như bộ mã hoá dữ liệu video 20 và bộ giải mã dữ liệu video 30, có thể áp dụng quy trình dưới đây để kích hoạt các bộ thông số cho các dòng bit một lớp hoặc các dòng bit một cảnh nhìn, khi bộ thông số GPS được xác định trong bảng 17, hoặc gần như tuân theo ví dụ thể hiện trong bảng 17.

Cấu trúc cú pháp RBSP cho bộ thông số thích ứng có thể có các thông số được tham chiếu bởi các đơn vị NAL có lát mã hoá của một hoặc nhiều hình ảnh mã hoá một cách gián tiếp qua một hoặc nhiều bộ thông số nhóm được tham chiếu bởi các đơn vị NAL có lát mã hoá. Mỗi cấu trúc cú pháp RBSP cho bộ thông số thích ứng ban đầu có thể được coi là không hoạt động khi bắt đầu tiến hành quy trình giải mã. Nhiều nhất một cấu trúc cú pháp RBSP cho bộ thông số thích ứng có thể được coi là hoạt động cho mỗi loại của các thông số APS ở một thời điểm cho trước bất kỳ trong khi tiến hành quy trình giải mã, và việc kích hoạt một cấu trúc cú pháp RBSP cho bộ thông số thích ứng cụ thể bất kỳ với một loại thông số APS cụ thể sẽ làm bất hoạt cấu trúc cú pháp RBSP cho bộ thông số thích ứng hoạt động trước đó (nếu có) với loại thông số APS cụ thể đó.

Khi cấu trúc cú pháp RBSP cho bộ thông số thích ứng (có một giá trị `aps_id` cụ thể) được coi là không hoạt động với một loại thông số APS cụ thể và được tham chiếu bởi đơn vị NAL có lát mã hoá với loại thông số APS cụ thể đó (sử dụng giá trị `aps_id` đó) một cách gián tiếp qua bộ thông số nhóm được tham chiếu bởi đơn vị NAL có lát mã hoá, thì nó có thể được kích hoạt cho loại thông số APS cụ thể đó. Cấu trúc

cú pháp RBSP cho bộ thông số thích ứng này được gọi là cấu trúc cú pháp RBSP cho bộ thông số thích ứng hoạt động cho loại thông số APS cụ thể đó cho tới khi nó được bất hoạt do việc kích hoạt một cấu trúc cú pháp RBSP cho bộ thông số thích ứng khác với loại thông số APS cụ thể đó. Cấu trúc cú pháp RBSP cho bộ thông số thích ứng, với giá trị `aps_id` cụ thể đó, có thể sẵn sàng để sử dụng cho quy trình giải mã trước khi kích hoạt.

Cấu trúc cú pháp RBSP cho bộ thông số hình ảnh có thể có các thông số được tham chiếu bởi các đơn vị NAL có lát mã hoá của một hoặc nhiều hình ảnh mã hoá một cách gián tiếp qua một hoặc nhiều bộ thông số nhóm được tham chiếu bởi các đơn vị NAL có lát mã hoá. Mỗi cấu trúc cú pháp RBSP cho bộ thông số hình ảnh ban đầu có thể được coi là không hoạt động khi bắt đầu tiến hành quy trình giải mã. Nhiều nhất một cấu trúc cú pháp RBSP cho bộ thông số hình ảnh có thể được coi là hoạt động ở một thời điểm cho trước bất kỳ trong khi tiến hành quy trình giải mã, và việc kích hoạt một cấu trúc cú pháp RBSP cụ thể bất kỳ cho bộ thông số hình ảnh sẽ làm bất hoạt cấu trúc cú pháp RBSP hoạt động trước đó cho bộ thông số hình ảnh (nếu có).

Khi cấu trúc cú pháp RBSP cho bộ thông số hình ảnh (có giá trị `pic_parameter_set_id` cụ thể) không hoạt động và được tham chiếu bởi đơn vị NAL có lát mã hoá (sử dụng giá trị `pic_parameter_set_id` đó) một cách gián tiếp qua bộ thông số nhóm được tham chiếu bởi đơn vị NAL có lát mã hoá, thì nó có thể được kích hoạt. Cấu trúc cú pháp RBSP cho bộ thông số hình ảnh này được gọi là cấu trúc cú pháp RBSP cho bộ thông số hình ảnh hoạt động cho tới khi nó được bất hoạt do việc kích hoạt một cấu trúc cú pháp RBSP cho bộ thông số hình ảnh khác. Cấu trúc cú pháp RBSP cho bộ thông số hình ảnh, có giá trị `pic_parameter_set_id` cụ thể đó, có thể sẵn sàng để sử dụng cho quy trình giải mã trước khi kích hoạt.

Mọi đơn vị NAL có bộ thông số hình ảnh chứa giá trị `pic_parameter_set_id` cho cấu trúc cú pháp RBSP hoạt động của bộ thông số hình ảnh cho hình ảnh mã hoá có thể có nội dung giống như cấu trúc cú pháp RBSP hoạt động của bộ thông số hình ảnh cho hình ảnh mã hoá trừ trường hợp nó đứng sau đơn vị NAL có lớp VCL cuối cùng của hình ảnh mã hoá và đứng trước đơn vị NAL có lớp VCL đầu tiên của hình ảnh mã hoá khác.

Cấu trúc cú pháp RBSP cho bộ thông số chuỗi có thể có các thông số được

tham chiếu bởi các đơn vị NAL có lát mã hoá của một hoặc nhiều hình ảnh mã hoá một cách gián tiếp qua một hoặc nhiều bộ thông số nhóm được tham chiếu bởi các đơn vị NAL có lát mã hoá, hoặc có thể được tham chiếu bởi một hoặc nhiều đơn vị NAL có thông tin SEI chứa thông báo SEI về chu kỳ đưa vào bộ nhớ đệm. Mỗi cấu trúc cú pháp RBSP cho bộ thông số chuỗi ban đầu có thể được coi là không hoạt động khi bắt đầu tiến hành quy trình giải mã. Nhiều nhất một cấu trúc cú pháp RBSP cho bộ thông số chuỗi có thể được coi là hoạt động ở một thời điểm cho trước bất kỳ trong khi tiến hành quy trình giải mã, và việc kích hoạt một cấu trúc cú pháp RBSP cho bộ thông số chuỗi cụ thể bất kỳ sẽ làm bất hoạt cấu trúc cú pháp RBSP cho bộ thông số chuỗi hoạt động trước đó (nếu có).

Khi cấu trúc cú pháp RBSP cho bộ thông số chuỗi (có giá trị `seq_parameter_set_id` cụ thể) chưa sẵn sàng hoạt động và được tham chiếu bởi đơn vị NAL có lát mã hoá một cách gián tiếp qua bộ thông số nhóm được tham chiếu bởi đơn vị NAL có lát mã hoá (sử dụng giá trị `seq_parameter_set_id` đó) hoặc được tham chiếu bởi đơn vị NAL có thông tin SEI chứa thông báo SEI về chu kỳ đưa vào bộ nhớ đệm (sử dụng giá trị `seq_parameter_set_id` đó), thì nó có thể được kích hoạt. Cấu trúc cú pháp RBSP cho bộ thông số chuỗi này được gọi là cấu trúc cú pháp RBSP cho bộ thông số chuỗi hoạt động cho tới khi nó được bất hoạt do việc kích hoạt một cấu trúc cú pháp RBSP cho bộ thông số chuỗi khác. Cấu trúc cú pháp RBSP cho bộ thông số chuỗi, có giá trị `seq_parameter_set_id` cụ thể đó và nằm trong đơn vị truy nhập có `temporal_id` bằng 0, có thể sẵn sàng để sử dụng cho quy trình giải mã trước khi kích hoạt. Cấu trúc cú pháp RBSP cho bộ thông số chuỗi đã được kích hoạt sẽ giữ nguyên trạng thái hoạt động trong toàn bộ chuỗi dữ liệu video mã hoá.

Cấu trúc cú pháp RBSP cho bộ thông số video có thể có các thông số được tham chiếu bởi các đơn vị NAL có lát mã hoá của một hoặc nhiều hình ảnh mã hoá một cách gián tiếp qua một hoặc nhiều bộ thông số nhóm được tham chiếu bởi các đơn vị NAL có lát mã hoá, hoặc có thể được tham chiếu bởi một hoặc nhiều đơn vị NAL có thông tin SEI chứa thông báo SEI về chu kỳ đưa vào bộ nhớ đệm. Mỗi cấu trúc cú pháp RBSP cho bộ thông số video ban đầu có thể được coi là không hoạt động khi bắt đầu tiến hành quy trình giải mã. Nhiều nhất một cấu trúc cú pháp RBSP cho bộ thông số video có thể được coi là hoạt động ở một thời điểm cho trước bất kỳ trong

khi tiến hành quy trình giải mã, và việc kích hoạt một cấu trúc cú pháp RBSP cho bộ thông số video cụ thể bất kỳ sẽ làm bất hoạt cấu trúc cú pháp RBSP cho bộ thông số video hoạt động trước đó (nếu có).

Khi cấu trúc cú pháp RBSP cho bộ thông số video (có giá trị `video_parameter_set_id` cụ thể) chưa sẵn sàng hoạt động và được tham chiếu bởi đơn vị NAL có lát mã hoá một cách gián tiếp qua bộ thông số nhóm được tham chiếu bởi đơn vị NAL có lát mã hoá (sử dụng giá trị `video_parameter_set_id` đó) hoặc được tham chiếu bởi đơn vị NAL có thông tin SEI chứa thông báo SEI về chu kỳ đưa vào bộ nhớ đệm (sử dụng giá trị `video_parameter_set_id` đó), thì nó có thể được kích hoạt. Cấu trúc cú pháp RBSP cho bộ thông số video này được gọi là cấu trúc cú pháp RBSP cho bộ thông số video hoạt động cho tới khi nó được bất hoạt do việc kích hoạt một cấu trúc cú pháp RBSP cho bộ thông số video khác. Cấu trúc cú pháp RBSP cho bộ thông số video, có giá trị `video_parameter_set_id` cụ thể đó và nằm trong đơn vị truy nhập có `temporal_id` bằng 0, sẽ sẵn sàng để sử dụng cho quy trình giải mã trước khi kích hoạt. Cấu trúc cú pháp RBSP cho bộ thông số video đã được kích hoạt sẽ giữ nguyên trạng thái hoạt động trong toàn bộ chuỗi dữ liệu video mã hoá.

Một đơn vị NAL có bộ thông số chuỗi bất kỳ chứa giá trị `seq_parameter_set_id` cho cấu trúc cú pháp RBSP cho bộ thông số chuỗi hoạt động đối với chuỗi dữ liệu video mã hoá có thể có thể có cùng nội dung với cấu trúc cú pháp RBSP cho bộ thông số chuỗi hoạt động đối với chuỗi dữ liệu video mã hoá trừ khi nó đứng sau đơn vị truy nhập cuối cùng của chuỗi dữ liệu video mã hoá này và đứng trước đơn vị NAL có lớp VCL thứ nhất và đơn vị NAL có thông tin SEI thứ nhất chứa thông báo SEI về chu kỳ đưa vào bộ nhớ đệm (nếu có) của một chuỗi dữ liệu video mã hoá khác.

Mọi đơn vị NAL có bộ thông số video chứa giá trị `video_parameter_set_id` cho cấu trúc cú pháp RBSP cho bộ thông số video hoạt động đối với chuỗi dữ liệu video mã hoá có thể có cùng nội dung với cấu trúc cú pháp RBSP cho bộ thông số video hoạt động đối với chuỗi dữ liệu video mã hoá trừ khi nó đứng sau đơn vị truy nhập cuối cùng của chuỗi dữ liệu video mã hoá này và đứng trước đơn vị NAL có lớp VCL thứ nhất và đơn vị NAL có thông tin SEI thứ nhất chứa thông báo SEI về chu kỳ đưa vào bộ nhớ đệm (nếu có) của một chuỗi dữ liệu video mã hoá khác.

Tất cả các quy định hạn chế được thể hiện trong quan hệ giữa các giá trị của

các phần tử cú pháp (và các giá trị của các biến được tìm ra dựa vào các phần tử cú pháp này) trong các bộ thông số video, các bộ thông số chuỗi, các bộ thông số hình ảnh và các bộ thông số thích ứng và các phần tử cú pháp khác là các quy định hạn chế có thể chỉ áp dụng cho bộ thông số video hoạt động, bộ thông số chuỗi hoạt động, bộ thông số hình ảnh hoạt động và các bộ thông số thích ứng hoạt động với mỗi loại thông số APS cụ thể. Nếu có cấu trúc cú pháp RBSP cho bộ thông số video không được kích hoạt trong dòng bit, thì các phần tử cú pháp của nó có thể có giá trị tuân theo các quy định hạn chế đã xác định nếu nó đã được kích hoạt bằng cách dựa vào trong dòng bit tuân theo một tiêu chuẩn khác. Nếu có cấu trúc cú pháp RBSP cho bộ thông số chuỗi không được kích hoạt trong dòng bit, thì các phần tử cú pháp của nó có thể có giá trị tuân theo các quy định hạn chế đã xác định nếu nó đã được kích hoạt bằng cách dựa vào dòng bit tuân theo một tiêu chuẩn khác. Nếu có cấu trúc cú pháp RBSP cho bộ thông số hình ảnh chưa từng được kích hoạt trong dòng bit, thì các phần tử cú pháp của nó có thể có giá trị tuân theo các quy định hạn chế đã xác định nếu nó đã được kích hoạt bằng cách dựa vào dòng bit tuân theo một tiêu chuẩn khác. Nếu có một cấu trúc cú pháp RBSP cho bộ thông số thích ứng bất kỳ chưa từng được kích hoạt trong dòng bit, thì các phần tử cú pháp của nó có thể có giá trị tuân theo các quy định hạn chế đã xác định nếu nó đã được kích hoạt bằng cách dựa vào dòng bit tuân theo một tiêu chuẩn khác.

Trong khi tiến hành quy trình giải mã, giá trị của các thông số của bộ thông số video hoạt động, bộ thông số chuỗi hoạt động, bộ thông số hình ảnh hoạt động và bộ thông số thích ứng hoạt động với mỗi loại của các thông số APS có thể được coi là có hiệu lực. Để thông dịch các thông báo SEI, giá trị của các thông số của bộ thông số video, bộ thông số chuỗi, bộ thông số hình ảnh và bộ thông số thích ứng đang hoạt động để tiến hành quy trình giải mã cho các đơn vị NAL có lớp VCL của hình ảnh mã hoá nằm trong cùng một đơn vị truy nhập có thể được coi là có hiệu lực trừ trường hợp có quy định khác về ngữ nghĩa của thông báo SEI.

Fig.7 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video theo các kỹ thuật nêu trong sáng chế. Mặc dù được mô tả dựa vào bộ mã hoá dữ liệu video 20, nhưng cần phải hiểu rằng, các thiết bị mã hoá dữ liệu video khác có thể được tạo cấu hình để thực hiện phương pháp trên Fig.7.

Ban đầu, trong ví dụ này, bộ mã hoá dữ liệu video 20 thu dòng bit có một hoặc nhiều lớp dữ liệu video thô (100). Ví dụ, nguồn dữ liệu video 18 (Fig.1) có thể cung cấp dữ liệu video có nhiều cảnh nhìn cho bộ mã hoá dữ liệu video 20. Theo cách khác, bộ mã hoá dữ liệu video 20, hoặc bộ tiền xử lý trong đó, có thể phân tách dòng bit video thô ra thành nhiều lớp, ví dụ, các lớp phân giải không gian, các lớp chất lượng, các lớp thời gian, hoặc tương tự. Trong các ví dụ khác, dòng bit có thể được phân tách ra thành một tổ hợp của các lớp khác nhau, ví dụ, một tổ hợp bất kỳ của các cảnh nhìn, các lớp phân giải không gian, các lớp chất lượng, các lớp thời gian, hoặc tương tự.

Bộ mã hoá dữ liệu video 20 có thể xác định một hoặc nhiều thông số chung cho các chuỗi tương ứng trên các tập hợp lớp (102). Các chuỗi tương ứng có thể là các chuỗi có vị trí thời gian tương ứng trong các lớp khác nhau. Có nghĩa là, chuỗi thứ nhất, có thời gian đầu (tính theo thời gian hiển thị) là T1 và thời gian cuối (cũng tính theo thời gian hiển thị) là T2, và chuỗi thứ hai, cũng có thời gian đầu là T1 và thời gian cuối là T2, có thể được coi là tương ứng với nhau. Cụ thể, chuỗi thứ nhất có thể tạo nên một phần lớp thứ nhất, và chuỗi thứ hai có thể tạo nên một phần lớp thứ hai khác. “Chuỗi” có thể là một loạt hình ảnh liên tiếp theo thứ tự giải mã, ví dụ, bắt đầu từ hình ảnh làm mới quy trình giải mã tức thời (*IDR: Instantaneous Decoding Refresh*) và kết thúc ngay trước hình ảnh IDR kế tiếp theo thứ tự giải mã. Nói chung, các thông số có thể tương ứng với một tập hợp chuỗi tương ứng của một hoặc nhiều lớp, ví dụ, N lớp, trong đó N là số nguyên. Bộ mã hoá dữ liệu video 20 có thể mã hoá bộ thông số VPS chứa dữ liệu cho các thông số đã xác định (104). Ví dụ, bộ mã hoá dữ liệu video 20 có thể mã hoá bộ thông số VPS tương ứng với một trong số các ví dụ được thể hiện trong bảng 2 hoặc bảng 5.

Bộ mã hoá dữ liệu video 20 cũng có thể xác định các thông số chung cho chuỗi trong một lớp (106). Chuỗi có thể là một trong số các chuỗi tương ứng với các chuỗi khác trong các lớp khác có bộ thông số VPS được mã hoá. Bộ mã hoá dữ liệu video 20 có thể mã hoá bộ thông số chuỗi (SPS) chứa các thông số chung cho chuỗi (108). Do đó, cần phải hiểu rằng, bộ thông số VPS và bộ thông số SPS là các cấu trúc dữ liệu khác nhau và tương ứng với các loại dữ liệu video khác nhau. Nếu bộ thông số VPS có thể tương ứng với một tập hợp chuỗi tương ứng trên nhiều lớp, thì bộ thông số SPS tương ứng với một chuỗi trong một lớp. Bộ thông số SPS có thể gần giống với bộ

thông số SPS theo chuẩn H.264/AVC, bộ thông số SPS theo chuẩn H.264/AVC như được mở rộng theo phiên bản mở rộng MVC (như được thể hiện trong bảng 1 trên đây), chuẩn HEVC sắp tới, hoặc ví dụ trong bảng 16 nêu trên. Ngoài ra, bộ mã hoá dữ liệu video 20 có thể mã hoá bộ thông số hình ảnh (PPS) cho một hình ảnh trong chuỗi (110). Bộ thông số PPS có thể gần giống với bộ thông số SPS theo chuẩn H.264/AVC, chuẩn HEVC sắp tới, hoặc ví dụ trong bảng 13 nêu trên. Mặc dù phương pháp trên Fig.7 là phương pháp mã hoá chỉ một bộ thông số PPS, nhưng cần phải hiểu rằng, có thể mã hoá nhiều bộ thông số PPS. Một hoặc nhiều hình ảnh có thể tham chiếu đến cùng một bộ thông số PPS.

Bộ mã hoá dữ liệu video 20 có thể xác định xem lớp hiện thời mà các bộ thông số SPS và PPS được mã hoá có phải là lớp cuối cùng hay không (112). Nếu chưa đến lớp cuối cùng (nhánh “sai” ở bước 112), thì bộ mã hoá dữ liệu video 20 có thể chọn lớp kế tiếp và mã hoá bộ thông số SPS và một hoặc nhiều bộ thông số PPS cho lớp kế tiếp, ví dụ, theo các bước từ bước 106 đến bước 110. Nếu đã đến lớp cuối cùng (nhánh “đúng” ở bước 112), thì bộ mã hoá dữ liệu video 20 có thể mã hoá dữ liệu video của các lớp khác nhau dựa vào dữ liệu của bộ thông số VPS, bộ thông số SPS và bộ thông số PPS. Ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào bộ thông số VPS được mô tả chi tiết dưới đây dựa vào các hình vẽ từ Fig.9 đến Fig.12.

Mặc dù không được thể hiện trong ví dụ trên Fig.7, nhưng trong một số ví dụ, bộ mã hoá dữ liệu video 20 có thể còn mã hoá một hoặc nhiều bộ thông số LPS và/hay một hoặc nhiều bộ thông số GPS, như đã nêu trên. Các bộ thông số LPS có thể gần như tuân theo ví dụ được thể hiện trong bảng 9, bảng 10 hoặc bảng 12, còn bộ thông số GPS có thể gần như tuân theo ví dụ được thể hiện trong bảng 14, bảng 15 hoặc bảng 17. Trong các ví dụ đó, bộ mã hoá dữ liệu video 20 cũng mã hoá dữ liệu video ít nhất một phần dựa vào các bộ thông số LPS và/hoặc các bộ thông số GPS.

Theo cách này, phương pháp trên Fig.7 là ví dụ về phương pháp bao gồm bước mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

Fig.8 là lưu đồ thể hiện ví dụ về phương pháp giải mã dữ liệu video theo các kỹ

thuật nêu trong sáng chế. Mặc dù được mô tả dựa vào bộ giải mã dữ liệu video 30, nhưng cần phải hiểu rằng, các thiết bị giải mã dữ liệu video khác có thể được tạo cấu hình để thực hiện phương pháp trên Fig.8.

Ban đầu, bộ giải mã dữ liệu video 30 thu dòng bit có bộ thông số VPS, một hoặc nhiều bộ thông số SPS, và một hoặc nhiều bộ thông số PPS cho các lớp dữ liệu video mã hoá (120). Sau đó, bộ giải mã dữ liệu video 30 có thể giải mã bộ thông số VPS, chứa các thông số chung cho các chuỗi tương ứng trên một hoặc nhiều lớp (122). Tương tự như vậy, bộ giải mã dữ liệu video 30 có thể giải mã bộ thông số chuỗi chứa các thông số chung cho một chuỗi của một lớp (124). Ngoài ra, bộ giải mã dữ liệu video 30 có thể giải mã bộ thông số hình ảnh có các thông số cho một hình ảnh trong chuỗi (126). Như đã nêu trên, một hoặc nhiều hình ảnh có thể tham chiếu đến cùng một bộ thông số PPS, và vì vậy, các thông số của bộ thông số PPS có thể được coi là các thông số chung cho một hoặc nhiều hình ảnh. Tương tự như vậy, bộ giải mã dữ liệu video 30 có thể giải mã nhiều bộ thông số PPS cho chuỗi, tuy không được thể hiện trên Fig.8.

Ngoài ra, bộ giải mã dữ liệu video 30 có thể xác định xem lớp mới nhất có phải là lớp cuối cùng hay không (128). Nếu lớp mới nhất chưa phải là lớp cuối cùng (nhánh “sai” ở bước 128), thì bộ giải mã dữ liệu video 30 có thể tiến hành giải mã bộ thông số SPS và một hoặc nhiều bộ thông số PPS cho lớp sau đó theo bước 124 và bước 126. Mặt khác, nếu lớp mới nhất chính là lớp cuối cùng (nhánh “đúng” ở bước 128), thì bộ giải mã dữ liệu video 30 có thể tiến hành giải mã dữ liệu video của các lớp dựa vào bộ thông số VPS, các bộ thông số SPS, và các bộ thông số PPS (130). Ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào bộ thông số VPS được mô tả chi tiết dưới đây dựa vào các hình vẽ từ Fig.9 đến Fig.12.

Mặc dù không được thể hiện trong ví dụ trên Fig.8, trong một số ví dụ, bộ giải mã dữ liệu video 30 có thể còn giải mã một hoặc nhiều bộ thông số LPS và/hay một hoặc nhiều bộ thông số GPS, như đã nêu trên. Các bộ thông số LPS có thể gần như tuân theo ví dụ được thể hiện trong bảng 9, bảng 10 hoặc bảng 12, còn bộ thông số GPS có thể gần như tuân theo ví dụ được thể hiện trong bảng 14, bảng 15 hoặc bảng 17. Trong các ví dụ đó, bộ giải mã dữ liệu video 30 cũng giải mã dữ liệu video ít nhất một phần dựa vào các bộ thông số LPS và/hoặc các bộ thông số GPS.

Theo cách này, phương pháp trên Fig.8 là ví dụ về phương pháp bao gồm bước mã hoá bộ thông số video (VPS) cho một hoặc nhiều lớp dữ liệu video, trong đó mỗi lớp trong số một hoặc nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và mã hoá một hoặc nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

Fig.9 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào số lượng lớp thời gian như được báo hiệu trong bộ thông số VPS. Phương pháp trên Fig.9 có thể được thực hiện bằng bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30. Ví dụ, phương pháp trên Fig.9 được mô tả liên quan đến bộ giải mã dữ liệu video 30.

Trong ví dụ này, bộ giải mã dữ liệu video 30 mã hoá (tức là, giải mã) bộ thông số VPS chỉ báo số lượng lớp thời gian trong dữ liệu video (150), ví dụ, của một hoặc nhiều lớp tương ứng với bộ thông số VPS. Ví dụ, bộ giải mã dữ liệu video 30 có thể giải mã “cnt_t” như đã mô tả dựa vào bảng 2 trên đây. Ví dụ khác, bộ giải mã dữ liệu video 30 có thể giải mã num_temporal_layers_minus1, như đã mô tả dựa vào bảng 5 trên đây.

Dựa vào thông tin chỉ báo này, trong ví dụ này, bộ giải mã dữ liệu video 30 giải mã các thông tin nhận dạng lớp thời gian cho mỗi lớp thời gian (152). Tương tự như vậy, bộ giải mã dữ liệu video 30 có thể xác định giá trị của các thông tin nhận dạng hình ảnh chuẩn dựa vào số lượng lớp thời gian (154). Ví dụ, bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để xác định, với hình ảnh hiện thời ở lớp N, là hình ảnh hiện thời sẽ không sử dụng các hình ảnh ở lớp N+1 hoặc cao hơn lớp N+1 để làm chuẩn. Vì vậy, bộ giải mã dữ liệu video 30 có thể xác định thông tin nhận dạng cho các hình ảnh chuẩn có thể có ở các lớp cho đến lớp N hoặc thấp hơn lớp N. Ngoài ra, bộ giải mã dữ liệu video 30 có thể giải mã dữ liệu của các hình ảnh ở lớp thời gian N sử dụng dữ liệu chuẩn của các lớp cho tới (và bao gồm cả) lớp N (156). Do đó, Fig.9 là ví dụ về phương pháp bao gồm bước mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng lớp thời gian tối đa trong một hoặc nhiều lớp dữ liệu video, và mã hoá một hoặc nhiều lớp ít nhất một phần dựa vào bộ thông số VPS.

Fig.10 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào số lượng hình ảnh cần sắp xếp lại trong một hoặc nhiều lớp và số

lượng hình ảnh cần lưu trữ vào bộ nhớ đệm hình ảnh giải mã. Phương pháp trên Fig.10 có thể được thực hiện bằng bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30. Ví dụ, phương pháp trên Fig.10 được mô tả liên quan đến bộ giải mã dữ liệu video 30.

Trong ví dụ này, bộ giải mã dữ liệu video 30 giải mã bộ thông số VPS chỉ báo số lượng hình ảnh cần sắp xếp lại trong một hoặc nhiều lớp dữ liệu video và số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã (ví dụ, bộ nhớ hình ảnh chuẩn 82) ở một thời điểm cho trước (160). Ví dụ, bộ giải mã dữ liệu video 30 có thể giải mã phần tử cú pháp của bộ thông số VPS gần như tương ứng với phần tử cú pháp `num_reorder_pics` như đã mô tả dựa vào bảng 16 trên đây và/hoặc thông tin giới hạn dòng bit xác định dung lượng bộ nhớ đệm DPB. Theo các phương án khác làm ví dụ, bộ thông số VPS có thể chỉ chứa thông tin này hoặc thông tin kia, và không nhất định phải chứa cả hai thông tin, trong số số lượng hình ảnh cần sắp xếp lại và số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã. Bộ giải mã dữ liệu video 30 có thể quản lý bộ nhớ đệm hình ảnh giải mã (ví dụ, bộ nhớ hình ảnh chuẩn 82) dựa vào số lượng hình ảnh cần sắp xếp lại và/hoặc số lượng hình ảnh sẽ được lưu trữ (162). Ví dụ, bộ giải mã dữ liệu video 30 có thể loại bỏ các hình ảnh ra khỏi bộ nhớ hình ảnh chuẩn 82 khi có nhiều hơn số lượng hình ảnh sẽ được lưu trữ được lưu trữ vào bộ nhớ hình ảnh chuẩn 82.

Bộ giải mã dữ liệu video 30 cũng có thể xác định giá trị của các thông tin nhận dạng hình ảnh chuẩn dựa vào số lượng hình ảnh trong bộ nhớ đệm DPB (tức là, trong bộ nhớ hình ảnh chuẩn 82) (164). Ngoài ra, bộ giải mã dữ liệu video 30 có thể giải mã dữ liệu của các hình ảnh dựa vào giá trị của các thông tin nhận dạng hình ảnh chuẩn (166). Do đó, phương pháp trên Fig.10 là ví dụ về phương pháp bao gồm bước mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã (DPB) trong khi giải mã một hoặc nhiều lớp và phương pháp bao gồm bước mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung được sắp xếp lại trên ít nhất một lớp trong số một hoặc nhiều lớp.

Fig.11 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào các thông số bộ giải mã chuẩn giả định (HRD) được báo hiệu trong bộ thông số VPS. Phương pháp trên Fig.11 có thể được thực hiện bằng bộ mã hoá dữ

liệu video 20 và/hoặc bộ giải mã dữ liệu video 30. Ví dụ, phương pháp trên Fig.11 được mô tả liên quan đến bộ giải mã dữ liệu video 30.

Trong ví dụ này, bộ giải mã dữ liệu video 30 giải mã bộ thông số VPS chỉ báo các thông số HRD (170). Bộ giải mã dữ liệu video 30 có thể còn xác định các thời điểm loại bỏ hình ảnh ra khỏi bộ nhớ đệm hình ảnh mã hoá (CPB) dựa vào các thông số HRD (172). Sau đó, bộ giải mã dữ liệu video 30 có thể loại bỏ dữ liệu ra khỏi bộ nhớ CPB dựa vào các thời điểm loại bỏ đã xác định (174), và giải mã dữ liệu đã được loại bỏ ra khỏi bộ nhớ CPB. Do đó, phương pháp trên Fig.11 là ví dụ về phương pháp bao gồm bước mã hoá dữ liệu của bộ thông số VPS chỉ báo một hoặc nhiều thông số bộ giải mã chuẩn giả định (HRD), và mã hoá dữ liệu của một hoặc nhiều lớp dựa vào các thông số HRD.

Fig.12 là lưu đồ thể hiện ví dụ về phương pháp mã hoá dữ liệu video ít nhất một phần dựa vào dữ liệu mở rộng được báo hiệu trong bộ thông số VPS. Phương pháp trên Fig.12 có thể được thực hiện bằng bộ mã hoá dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30. Ví dụ, phương pháp trên Fig.12 được mô tả liên quan đến bộ giải mã dữ liệu video 30.

Bộ giải mã dữ liệu video 30, trong ví dụ này, giải mã dữ liệu của bộ thông số VPS chỉ báo việc bộ thông số VPS chứa dữ liệu mở rộng (180). Ví dụ, bộ giải mã dữ liệu video 30 có thể giải mã cờ VPS_extension_flag của bộ thông số VPS. Sau đó, bộ giải mã dữ liệu video 30 xác định xem thông tin có chỉ báo rằng bộ thông số VPS chứa dữ liệu mở rộng hay không (182). Nếu thông tin chỉ báo rằng bộ thông số VPS chứa dữ liệu mở rộng (nhánh “có” ở bước 182), thì bộ giải mã dữ liệu video 30 mã hoá dữ liệu mở rộng của bộ thông số VPS với một hoặc nhiều công cụ mã hoá mở rộng (184), và giải mã dữ liệu video sử dụng các công cụ mã hoá mở rộng và dữ liệu mở rộng (186). Mặt khác, nếu thông tin chỉ báo rằng bộ thông số VPS không chứa dữ liệu mở rộng (nhánh “không” ở bước 182), thì bộ giải mã dữ liệu video 30 có thể giải mã dữ liệu video sử dụng các công cụ mã hoá thông thường (188). Theo cách này, phương pháp trên Fig.12 là ví dụ về phương pháp bao gồm bước mã hoá dữ liệu của bộ thông số VPS chỉ báo việc bộ thông số VPS có mở rộng vượt ra ngoài chuẩn tương ứng hay không, và khi bộ thông số VPS có mở rộng, thì mã hoá dữ liệu mở rộng, cũng như mã hoá dữ liệu video dựa vào dữ liệu mở rộng của bộ thông số VPS.

Cần phải hiểu rằng, tùy thuộc vào phương án làm ví dụ, một số thao tác hoặc sự kiện trong kỹ thuật bất kỳ trong số các kỹ thuật nêu trong sáng chế có thể được thực hiện theo một trình tự khác, có thể được bổ sung vào, được kết hợp với nhau, hoặc được loại bỏ bớt (ví dụ, không phải tất cả các thao tác hoặc sự kiện được mô tả đều là cần thiết để thực hiện các kỹ thuật này). Ngoài ra, trong một số ví dụ, các thao tác hoặc sự kiện có thể được thực hiện đồng thời, ví dụ, bằng quy trình xử lý đa khâu chuỗi, xử lý ngắt, hoặc đa xử lý, chứ không phải chỉ thực hiện tuần tự.

Theo một hoặc nhiều phương án làm ví dụ, các chức năng đã mô tả có thể được thực hiện bằng phần cứng, phần mềm, phần sụn, hoặc kết hợp các loại này. Nếu được thực hiện bằng phần mềm, thì các chức năng này có thể được lưu trữ hoặc được truyền dưới dạng một hoặc nhiều lệnh hoặc mã trên vật ghi đọc được bằng máy tính và được thi hành bằng bộ phận xử lý có dạng phần cứng. Vật ghi đọc được bằng máy tính có thể là phương tiện lưu trữ đọc được bằng máy tính, tương ứng với phương tiện lưu trữ hữu hình như phương tiện lưu trữ dữ liệu, hoặc phương tiện truyền thông là mọi phương tiện tạo điều kiện thuận lợi cho việc truyền chương trình máy tính từ nơi này đến nơi khác, ví dụ, theo giao thức truyền thông. Theo cách này, vật ghi đọc được bằng máy tính thường có thể tương ứng với (1) phương tiện lưu trữ hữu hình đọc được bằng máy tính thuộc loại không khả biến hoặc (2) phương tiện truyền thông như tín hiệu hoặc sóng mang. Phương tiện lưu trữ dữ liệu có thể là mọi loại phương tiện có sẵn có thể truy nhập được bằng một hoặc nhiều máy tính hoặc một hay nhiều bộ xử lý để tìm kiếm các lệnh, mã và/hoặc cấu trúc dữ liệu để thực hiện các kỹ thuật nêu trong sáng chế. Sản phẩm chứa chương trình máy tính có thể là vật ghi đọc được bằng máy tính.

Ví dụ, phương tiện lưu trữ đọc được bằng máy tính có thể là bộ nhớ truy nhập ngẫu nhiên (*RAM: Random Access Memory*), bộ nhớ chỉ đọc (*ROM: Read Only Memory*), bộ nhớ chỉ đọc lập trình được xóa được bằng điện (*EEPROM: Electrically Erasable Programmable Read Only Memory*)), đĩa compac-bộ nhớ chỉ đọc (*CD-ROM: Compact Disc-Read Only Memory*) hoặc đĩa quang khác, đĩa từ hoặc thiết bị nhớ từ tính khác, thiết bị nhớ truy nhập nhanh, hoặc mọi phương tiện khác có thể dùng để lưu trữ mã chương trình cần thiết dưới dạng lệnh hoặc cấu trúc dữ liệu và có thể truy nhập được bằng máy tính, nhưng không chỉ giới hạn ở đó. Ngoài ra, mọi loại kết

nổi được gọi theo cách thích hợp là phương tiện đọc được bằng máy tính. Ví dụ, nếu các lệnh được truyền từ website, máy chủ hoặc nguồn từ xa khác sử dụng cáp đồng trục, cáp sợi quang, cáp xoắn đôi, đường thuê bao số (*DSL: Digital Subscriber Line*), hoặc sử dụng công nghệ không dây như hồng ngoại, vô tuyến và vi ba, thì cáp đồng trục, cáp sợi quang, cáp xoắn đôi, DSL, hoặc công nghệ không dây như hồng ngoại, vô tuyến và vi ba đó cũng nằm trong định nghĩa phương tiện. Tuy nhiên, cần phải hiểu rằng, phương tiện lưu trữ đọc được bằng máy tính và phương tiện lưu trữ dữ liệu không bao gồm các loại kết nối, sóng mang, tín hiệu hoặc phương tiện khả biến khác, mà chỉ bao gồm phương tiện lưu trữ hữu hình, không khả biến. Đĩa từ và đĩa quang, như được sử dụng ở đây, bao gồm đĩa compac (*CD: Compact Disc*), đĩa laze, đĩa quang, đĩa số đa năng (*DVD: Digital Versatile Disc*), đĩa mềm và đĩa blu-ray, trong đó đĩa từ thường tái tạo dữ liệu bằng phương pháp từ tính, còn đĩa quang thì tái tạo dữ liệu bằng phương pháp quang học sử dụng laze. Dạng kết hợp của các loại phương tiện nêu trên cũng được coi là nằm trong phạm vi vật ghi đọc được bằng máy tính.

Các lệnh có thể được thi hành bằng một hoặc nhiều bộ xử lý, như một hoặc nhiều bộ xử lý tín hiệu số (*DSP*), bộ vi xử lý đa năng, mạch tích hợp chuyên dụng (*ASIC*), mảng cửa lập trình được bằng trường (*FPGA*), hoặc mạch logic rời rạc hoặc mạch tích hợp tương đương khác. Do đó, thuật ngữ “bộ xử lý”, như được sử dụng trong sáng chế này, có thể dùng để chỉ mọi cấu trúc nêu trên hoặc mọi cấu trúc khác phù hợp để thực hiện các kỹ thuật nêu trong sáng chế. Ngoài ra, theo một số khía cạnh, chức năng nêu trong sáng chế có thể được thực hiện bằng môđun phần cứng và/hoặc phần mềm chuyên dụng được tạo cấu hình để mã hoá và giải mã, hoặc được kết hợp thành một bộ mã hoá-giải mã kết hợp. Đồng thời, các kỹ thuật có thể được thực hiện hoàn toàn trong một hoặc nhiều mạch hoặc phần tử logic.

Kỹ thuật theo sáng chế có thể được áp dụng trong rất nhiều thiết bị, trong đó có tổ hợp thiết bị không dây, mạch tích hợp (*IC: Integrated Circuit*) hoặc bộ IC (ví dụ, bộ chip). Các thành phần, môđun, hoặc bộ phận được mô tả trong sáng chế để nhấn mạnh các khía cạnh chức năng của các thiết bị được tạo cấu hình để thực hiện các kỹ thuật đã mô tả, nhưng không nhất thiết phải được thực hiện bằng các bộ phận phần cứng khác nhau. Thực ra, như đã nêu trên, các bộ phận khác nhau có thể được kết hợp lại thành một bộ phận phần cứng mã hoá-giải mã hoặc được thực hiện bởi một tập hợp

các bộ phận phần cứng tương tác với nhau, có một hoặc nhiều bộ xử lý như đã nêu trên, kết hợp với phần mềm và/hoặc phần sụn phù hợp.

Nhiều phương án làm ví dụ đã được mô tả. Các phương án làm ví dụ này và các phương án khác làm ví dụ đều nằm trong phạm vi của sáng chế như được xác định theo các điểm yêu cầu bảo hộ dưới đây.

YÊU CẦU BẢO HỘ

1. Phương pháp mã hoá dữ liệu video bao gồm các bước:

mã hoá bộ thông số video (Video Parameter Set - VPS) cho dòng bit bao gồm nhiều lớp dữ liệu video, trong đó mỗi lớp trong số nhiều lớp dữ liệu video tham chiếu đến một bộ thông số VPS, và trong đó bước mã hoá bộ thông số VPS bao gồm:

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung được sắp xếp lại trên ít nhất một lớp trong số nhiều lớp dữ liệu video,

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã (Decoded Picture Buffer - DPB) trong khi giải mã nhiều lớp dữ liệu video,

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng lớp thời gian tối đa trong dòng bit bao gồm nhiều lớp dữ liệu video;

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung nhìn tối đa trong dòng bit bao gồm nhiều lớp dữ liệu video;

mã hóa thông tin xác định ánh xạ của chỉ số mẫu đến thông tin chỉ báo đặc trưng, trong đó bước mã hóa thông tin xác định ánh xạ này bao gồm việc mã hóa một hoặc nhiều: độ phân giải không gian tương ứng cho mỗi chỉ số trong số nhiều chỉ số tính phụ thuộc, tốc độ khung cho mỗi chỉ số trong số nhiều chỉ số thời gian, hoặc thông tin nhận dạng khung nhìn cho mỗi chỉ số trong số nhiều chỉ số khung nhìn; và

mã hóa nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

2. Phương pháp theo điểm 1, trong đó bước mã hoá bộ thông số VPS còn bao gồm bước mã hoá dữ liệu của bộ thông số VPS chỉ báo một hoặc nhiều bộ thông số bộ giải mã chuẩn giả định (Hypothetical Reference Decoder - HRD).

3. Phương pháp theo điểm 1, trong đó bước mã hoá bộ thông số VPS còn bao gồm bước mã hoá dữ liệu của bộ thông số VPS chỉ báo việc bộ thông số VPS có bao gồm sự mở rộng vượt ra ngoài chuẩn tương ứng hay không, và khi bộ thông số VPS có

bao gồm sự mở rộng này, thì mã hoá dữ liệu cho sự mở rộng này.

4. Phương pháp theo điểm 1, trong đó bước mã hoá nhiều lớp dữ liệu video bao gồm bước mã hoá nhiều lớp dữ liệu video theo chuẩn mã hoá dữ liệu video đạt hiệu quả cao (High Efficiency Video Coding - HEVC).

5. Phương pháp theo điểm 1, trong đó bước mã hoá nhiều lớp dữ liệu video bao gồm bước mã hoá nhiều lớp dữ liệu video theo ít nhất một chuẩn trong số chuẩn mã hoá video đa khung nhìn (Multiview Video Coding - MVC) và chuẩn mã hoá video mở rộng được (Scalable Video Coding - SVC).

6. Phương pháp theo điểm 1, trong đó bước mã hoá bộ thông số VPS bao gồm bước mã hoá thông tin định rõ, cho một hoặc nhiều kích thước của nhiều lớp dữ liệu video, một hoặc nhiều:

số lượng lớp ưu tiên trong nhiều lớp dữ liệu video,

số lượng lớp phụ thuộc trong nhiều lớp dữ liệu video,

số lượng lớp thời gian trong nhiều lớp dữ liệu video,

số lượng lớp chất lượng tối đa cho một lớp phụ thuộc bất kỳ trong một hoặc nhiều lớp dữ liệu video.

7. Phương pháp theo điểm 6, trong đó khi tập hợp con gồm nhiều lớp dữ liệu video có cùng độ phân giải không gian và cùng độ sâu bit, mỗi lớp trong tập hợp con đó tương ứng với một lớp phụ thuộc khác nhau.

8. Phương pháp theo điểm 7, trong đó bước mã hoá thông tin xác định ánh xạ này bao gồm bước mã hoá thông tin để xác định thông tin chỉ báo đặc trưng tương ứng cho mỗi chỉ số đặc trưng khi thông tin chỉ báo đặc trưng xác định các đặc trưng của kích thước của nhiều lớp dữ liệu video không nằm trong khoảng chỉ số từ không đến số đếm kích thước mẫu trừ 1 trong đó số đếm này được xác định bằng một chỉ số.

9. Phương pháp theo điểm 7, trong đó bước mã hoá thông tin xác định ánh xạ bao gồm bước mã hoá một hoặc nhiều:

cặp giá trị độ sâu cụ thể cho thành phần độ chói và thành phần màu với mỗi chỉ số độ sâu bit, hoặc

thông tin chỉ báo định dạng mẫu thành phần màu cụ thể với mỗi định dạng mẫu thành phần màu.

10. Phương pháp theo điểm 1, trong đó bước mã hoá bộ thông số VPS bao gồm bước mã hoá thông tin xác định các thông số điều khiển và một hoặc nhiều cờ kích hoạt/bất hoạt công cụ.

11. Phương pháp theo điểm 10, trong đó các thông số điều khiển và một hoặc nhiều cờ kích hoạt/bất hoạt công cụ là một hoặc nhiều:

thông số pcm_bit_depth_luma_minus1,

thông số pcm_bit_depth_luma_minus1,

cờ loop_filter_across_slice_flag,

cờ pcm_loop_filter_disable_flag,

cờ temporal_id_nesting_flag,

một hoặc nhiều phần tử cú pháp liên quan đến ô,

cờ chroma_pred_from_luma_enabled_flag,

cờ sample_adaptive_offset_enabled_flag,

cờ adaptive_loop_filter_enabled_flag, hoặc

cờ inter_4x4_enabled_flag.

12. Phương pháp theo điểm 1, trong đó bước mã hoá bộ thông số VPS bao gồm bước mã hoá thông tin xác định một hoặc nhiều bộ mô tả điểm hoạt động.

13. Phương pháp theo điểm 12, trong đó bước mã hoá thông tin xác định một hoặc nhiều bộ mô tả điểm hoạt động bao gồm bước mã hoá thông tin xác định một hoặc nhiều:

số lượng điểm hoạt động tối đa,

tính phụ thuộc giữa các lớp hoặc các khung nhìn khác nhau,

thông tin profin và mức cho mỗi điểm hoạt động,

tỷ lệ bit cho mỗi điểm hoạt động.

14. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước mã hoá bộ thông số chuỗi phân lớp (Layered Sequence Parameter Set - LPS) tương ứng cho mỗi lớp trong số nhiều lớp dữ liệu video, trong đó bước mã hoá nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS bao gồm bước mã hoá nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS và bộ thông số LPS tương ứng.

15. Phương pháp theo điểm 14, trong đó bước mã hoá các bộ thông số LPS tương ứng cho mỗi lớp trong số nhiều lớp dữ liệu video bao gồm bước mã hoá thông tin xác định thông tin chỉ báo kích thước mẫu để chỉ báo, với mỗi kích thước, chỉ số cho mỗi kích thước.

16. Phương pháp theo điểm 14, trong đó bước mã hoá các bộ thông số LPS tương ứng cho mỗi lớp trong số nhiều lớp dữ liệu video bao gồm bước mã hoá thông tin xác định các thông số điều khiển và các cờ kích hoạt/bất hoạt công cụ.

17. Phương pháp theo điểm 16, trong đó các thông số điều khiển và một hoặc nhiều cờ kích hoạt/bất hoạt công cụ là một hoặc nhiều:

thông số pcm_bit_depth_luma_minus1,

thông số pcm_bit_depth_chroma_minus1,

cờ loop_filter_across_slice_flag,

cờ pcm_loop_filter_disable_flag,

một hoặc nhiều phần tử cú pháp liên quan đến ô,

cờ chroma_pred_from_luma_enabled_flag,

cờ `sample_adaptive_offset_enabled_flag`,

cờ `adaptive_loop_filter_enabled_flag`, hoặc

cấu trúc phân cấp của đơn vị mã hoá (Coding Unit - CU).

18. Phương pháp theo điểm 14, trong đó bước mã hoá các bộ thông số LPS tương ứng cho mỗi lớp trong số nhiều lớp dữ liệu video bao gồm bước mã hoá thông tin xác định thông tin của một hoặc nhiều bộ thông số khác áp dụng cho ít nhất một trong số lát, nhóm lát, hình ảnh, hoặc vài hình ảnh tham chiếu đến bộ thông số hình ảnh (Picture Parameter Set - PPS) chung.

19. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước mã hoá một hoặc nhiều bộ thông số hình ảnh (PPS) sao cho các bộ thông số PPS không tham chiếu đến bộ thông số VPS, không tham chiếu đến các bộ thông số chuỗi phân lớp (LPS) của nhiều lớp dữ liệu video.

20. Phương pháp theo điểm 19, trong đó bước mã hoá nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS bao gồm bước mã hoá nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS, các bộ thông số PPS và các bộ thông số LPS, sao cho khi một phần tử cú pháp của một trong số các bộ thông số PPS xung đột với bộ thông số VPS hoặc một bộ thông số tương ứng trong số các bộ thông số LPS, thì mã hoá lớp tương ứng trong số nhiều lớp dữ liệu video dựa vào phần tử cú pháp đó của một trong số các bộ thông số PPS.

21. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước mã hoá bộ thông số nhóm (Grouping Parameter Set - GPS) để nhóm tất cả các bộ thông số, kể cả bộ thông số VPS, cho nhiều lớp dữ liệu video với nhau.

22. Phương pháp theo điểm 21, trong đó bước mã hoá bộ thông số GPS bao gồm bước mã hoá thông tin xác định thông tin nhận dạng của bộ thông số GPS, phương pháp này còn bao gồm bước mã hoá thông tin của phần đầu lát tương ứng với thông tin nhận dạng của bộ thông số GPS.

23. Phương pháp theo điểm 1, trong đó bước mã hoá nhiều lớp dữ liệu video bao gồm bước giải mã nhiều lớp dữ liệu video, và trong đó bước mã hoá bộ thông số VPS

bao gồm bước phân tích cú pháp cho bộ thông số VPS.

24. Phương pháp theo điểm 1, trong đó bước mã hoá nhiều lớp dữ liệu video bao gồm bước mã hoá nhiều lớp dữ liệu video, và trong đó bước mã hoá bộ thông số VPS bao gồm bước thiết lập bộ thông số VPS.

25. Thiết bị mã hoá dữ liệu video có bộ mã hoá dữ liệu video được tạo cấu hình để:

mã hoá bộ thông số video (VPS) dòng bit bao gồm nhiều lớp dữ liệu video, trong đó mỗi lớp trong số nhiều lớp dữ liệu video tham chiếu đến bộ thông số VPS này, và trong đó bộ mã hoá video mà được tạo cấu hình để mã hóa bộ thông số VPS này được tạo cấu hình để:

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung được sắp xếp lại trên ít nhất một lớp trong số nhiều lớp dữ liệu video,

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã (Decoded Picture Buffer - DPB) trong khi giải mã nhiều lớp dữ liệu video,

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng lớp thời gian tối đa trong dòng bit bao gồm nhiều lớp dữ liệu video;

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung nhìn tối đa trong dòng bit bao gồm nhiều lớp dữ liệu video;

mã hóa thông tin xác định ánh xạ của chỉ số mẫu đến thông tin chỉ báo đặc trưng, trong đó để mã hóa thông tin xác định ánh xạ này, bộ mã hóa video được tạo cấu hình để mã hóa một hoặc nhiều: độ phân giải không gian tương ứng cho mỗi chỉ số trong số nhiều chỉ số tính phụ thuộc, tốc độ khung cho mỗi chỉ số trong số nhiều chỉ số thời gian, hoặc thông tin nhận dạng khung nhìn cho mỗi chỉ số trong số nhiều chỉ số khung nhìn; và

mã hóa nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

26. Thiết bị theo điểm 25, trong đó bộ mã hoá dữ liệu video mà được tạo cấu hình để mã hóa bộ thông số VPS còn được tạo cấu hình để mã hoá dữ liệu của bộ thông số

VPS chỉ báo một hoặc nhiều bộ thông số bộ giải mã chuẩn giả định (HRD).

27. Thiết bị theo điểm 25, trong đó bộ mã hoá dữ liệu video mà được tạo cấu hình để mã hóa bộ thông số VPS còn được tạo cấu hình để mã hoá dữ liệu của bộ thông số VPS chỉ báo việc bộ thông số VPS có bao gồm sự mở rộng vượt ra ngoài chuẩn tương ứng hay không, và khi bộ thông số VPS có mở rộng, thì mã hoá dữ liệu cho sự mở rộng này.

28. Thiết bị theo điểm 25, trong đó bộ mã hoá dữ liệu video được tạo cấu hình để mã hoá nhiều lớp dữ liệu video theo một chuẩn trong số chuẩn mã hoá video hiệu quả cao (High Efficiency Video Coding - HEVC), chuẩn mã hoá video đa khung nhìn (MVC) và chuẩn mã hoá video mở rộng được (SVC).

29. Thiết bị theo điểm 25, trong đó bộ mã hoá dữ liệu video là bộ giải mã dữ liệu video, và trong đó thiết bị này còn bao gồm màn hiển thị được tạo cấu hình để hiển thị dữ liệu video.

30. Thiết bị theo điểm 25, trong đó bộ mã hoá dữ liệu video là bộ mã hoá dữ liệu video, và trong đó thiết bị này còn bao gồm camera được tạo cấu hình để tạo dữ liệu video.

31. Thiết bị theo điểm 25, trong đó thiết bị này bao gồm ít nhất một trong số:

mạch tích hợp;

bộ vi xử lý; hoặc

thiết bị truyền thông không dây di động có bộ mã hoá dữ liệu video.

32. Thiết bị theo điểm 25, trong đó bộ mã hóa video mà được tạo cấu hình để mã hóa bộ thông số VPS được tạo cấu hình để mã hóa thông tin xác định một hoặc nhiều bộ mô tả điểm hoạt động.

33. Thiết bị theo điểm 32, trong đó để mã hóa thông tin xác định một hoặc nhiều bộ mô tả điểm hoạt động, bộ mã hóa video được tạo cấu hình để mã hóa thông tin xác định một hoặc nhiều:

số lượng điểm hoạt động tối đa,

tính phụ thuộc giữa các lớp hoặc các khung nhìn khác nhau,

thông tin profin và mức cho mỗi điểm hoạt động,

tỷ lệ bit cho mỗi điểm hoạt động.

34. Thiết bị mã hoá dữ liệu video bao gồm:

phương tiện mã hoá bộ thông số video (VPS) cho dòng bit bao gồm nhiều lớp dữ liệu video, trong đó mỗi lớp trong số nhiều lớp dữ liệu video tham chiếu đến bộ thông số VPS này, và trong đó phương tiện để mã hóa bộ thông số VPS bao gồm:

phương tiện mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung được sắp xếp lại trên ít nhất một lớp trong số nhiều lớp dữ liệu video,

phương tiện mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã (Decoded Picture Buffer - DPB) trong khi giải mã nhiều lớp dữ liệu video,

phương tiện mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng lớp thời gian tối đa trong dòng bit bao gồm nhiều lớp dữ liệu video;

phương tiện mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung nhìn tối đa trong dòng bit bao gồm nhiều lớp dữ liệu video;

phương tiện mã hóa thông tin xác định ánh xạ của chỉ số mẫu đến thông tin chỉ báo đặc trưng, trong đó phương tiện mã hóa thông tin xác định ánh xạ này bao gồm phương tiện mã hóa một hoặc nhiều: độ phân giải không gian tương ứng cho mỗi chỉ số trong số nhiều chỉ số tính phụ thuộc, tốc độ khung cho mỗi chỉ số trong số nhiều chỉ số thời gian, hoặc thông tin nhận dạng khung nhìn cho mỗi chỉ số trong số nhiều chỉ số khung nhìn; và

phương tiện mã hóa nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

35. Thiết bị theo điểm 34, trong đó phương tiện mã hoá bộ thông số VPS còn bao gồm phương tiện mã hoá dữ liệu của bộ thông số VPS chỉ báo một hoặc nhiều bộ thông số bộ giải mã chuẩn giả định (HRD).

36. Thiết bị theo điểm 34, trong đó phương tiện mã hoá bộ thông số VPS còn bao gồm phương tiện mã hoá dữ liệu của bộ thông số VPS chỉ báo việc bộ thông số VPS có bao gồm sự mở rộng vượt ra ngoài chuẩn tương ứng hay không, và khi bộ thông số VPS có mở rộng, thì mã hoá dữ liệu cho sự mở rộng này.

37. Thiết bị theo điểm 34, trong đó phương tiện mã hoá bộ thông số VPS bao gồm phương tiện mã hoá nhiều lớp dữ liệu video theo một chuẩn trong số chuẩn mã hoá video hiệu quả cao (HEVC), chuẩn mã hoá video đa khung nhìn (MVC) và chuẩn mã hoá video mở rộng được (SVC).

38. Thiết bị theo điểm 34, trong đó phương tiện mã hóa bộ thông số VPS bao gồm phương tiện mã hóa thông tin xác định một hoặc nhiều bộ mô tả điểm hoạt động.

39. Thiết bị theo điểm 38, trong đó phương tiện mã hóa thông tin xác định một hoặc nhiều bộ mô tả điểm hoạt động bao gồm phương tiện mã hóa thông tin xác định một hoặc nhiều:

số lượng điểm hoạt động tối đa,

tính phụ thuộc giữa các lớp hoặc các khung nhìn khác nhau,

thông tin profin và mức cho mỗi điểm hoạt động,

tỷ lệ bit cho mỗi điểm hoạt động.

40. Vật ghi bất biến đọc được bằng máy tính trên đó lưu trữ các lệnh, mà khi được thực thi, sẽ khiến cho bộ xử lý:

mã hoá bộ thông số video (VPS) cho dòng bit bao gồm nhiều lớp dữ liệu video, trong đó mỗi lớp trong số nhiều lớp dữ liệu video tham chiếu đến bộ thông số VPS này; và trong đó các lệnh mà khiến cho bộ xử lý mã hóa bộ thông số VPS là các lệnh mà khiến cho bộ xử lý:

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung được sắp xếp lại trên ít nhất một lớp trong số nhiều lớp dữ liệu video,

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng hình ảnh sẽ được lưu trữ vào bộ nhớ đệm hình ảnh giải mã (Decoded Picture Buffer - DPB)

trong khi giải mã nhiều lớp dữ liệu video,

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng lớp thời gian tối đa trong dòng bit bao gồm nhiều lớp dữ liệu video;

mã hoá dữ liệu của bộ thông số VPS chỉ báo số lượng khung nhìn tối đa trong dòng bit bao gồm nhiều lớp dữ liệu video;

mã hóa thông tin xác định ánh xạ của chỉ số mẫu đến thông tin chỉ báo đặc trưng, trong đó các lệnh khiến cho bộ xử lý mã hóa thông tin xác định ánh xạ này còn bao gồm các lệnh, mà khi được thực thi, khiến cho bộ xử lý mã hóa một hoặc nhiều: độ phân giải không gian tương ứng cho mỗi chỉ số trong số nhiều chỉ số tính phụ thuộc, tốc độ khung cho mỗi chỉ số trong số nhiều chỉ số thời gian, hoặc thông tin nhận dạng khung nhìn cho mỗi chỉ số trong số nhiều chỉ số khung nhìn; và

mã hóa nhiều lớp dữ liệu video ít nhất một phần dựa vào bộ thông số VPS.

41. Vật ghi bất biến đọc được bằng máy tính theo điểm 40, trong đó các lệnh mà khiến cho bộ xử lý mã hoá bộ thông số VPS còn bao gồm các lệnh, mà khi được thực thi, khiến cho bộ xử lý mã hoá dữ liệu của bộ thông số VPS chỉ báo một hoặc nhiều bộ thông số bộ giải mã chuẩn giả định (HRD).

42. Vật ghi bất biến đọc được bằng máy tính theo điểm 40, trong đó các lệnh mà khiến cho bộ xử lý mã hoá bộ thông số VPS còn bao gồm các lệnh, mà khi được thực thi, khiến cho bộ xử lý mã hoá dữ liệu của bộ thông số VPS chỉ báo việc bộ thông số VPS có bao gồm sự mở rộng vượt ra ngoài chuẩn tương ứng hay không, và khi bộ thông số VPS có mở rộng, thì mã hoá dữ liệu cho sự mở rộng này.

43. Vật ghi bất biến đọc được bằng máy tính theo điểm 40, trong đó các lệnh mà khiến cho bộ xử lý mã hoá nhiều lớp dữ liệu video bao gồm các lệnh, mà khi được thực thi, khiến cho bộ xử lý mã hoá nhiều lớp dữ liệu video theo một chuẩn trong số chuẩn mã hoá video hiệu quả cao (HEVC), chuẩn mã hoá video đa khung nhìn (MVC) và chuẩn mã hoá video mở rộng được (SVC).

44. Vật ghi bất biến đọc được bằng máy tính theo điểm 40, trong đó các lệnh khiến

cho bộ xử lý mã hóa bộ thông số VPS còn bao gồm các lệnh, mà khi được thực thi, khiến cho bộ xử lý mã hoá thông tin xác định một hoặc nhiều bộ mô tả điểm hoạt động.

45. Vật ghi bất biến đọc được bằng máy tính theo điểm 44, trong đó các lệnh khiến cho bộ xử lý mã hóa thông tin xác định một hoặc nhiều bộ mô tả điểm hoạt động còn bao gồm các lệnh, mà khi được thực thi, khiến cho bộ xử lý mã hoá thông tin xác định một hoặc nhiều:

số lượng điểm hoạt động tối đa,

tính phụ thuộc giữa các lớp hoặc các khung nhìn khác nhau,

thông tin profin và mức cho mỗi điểm hoạt động,

tỷ lệ bit cho mỗi điểm hoạt động.

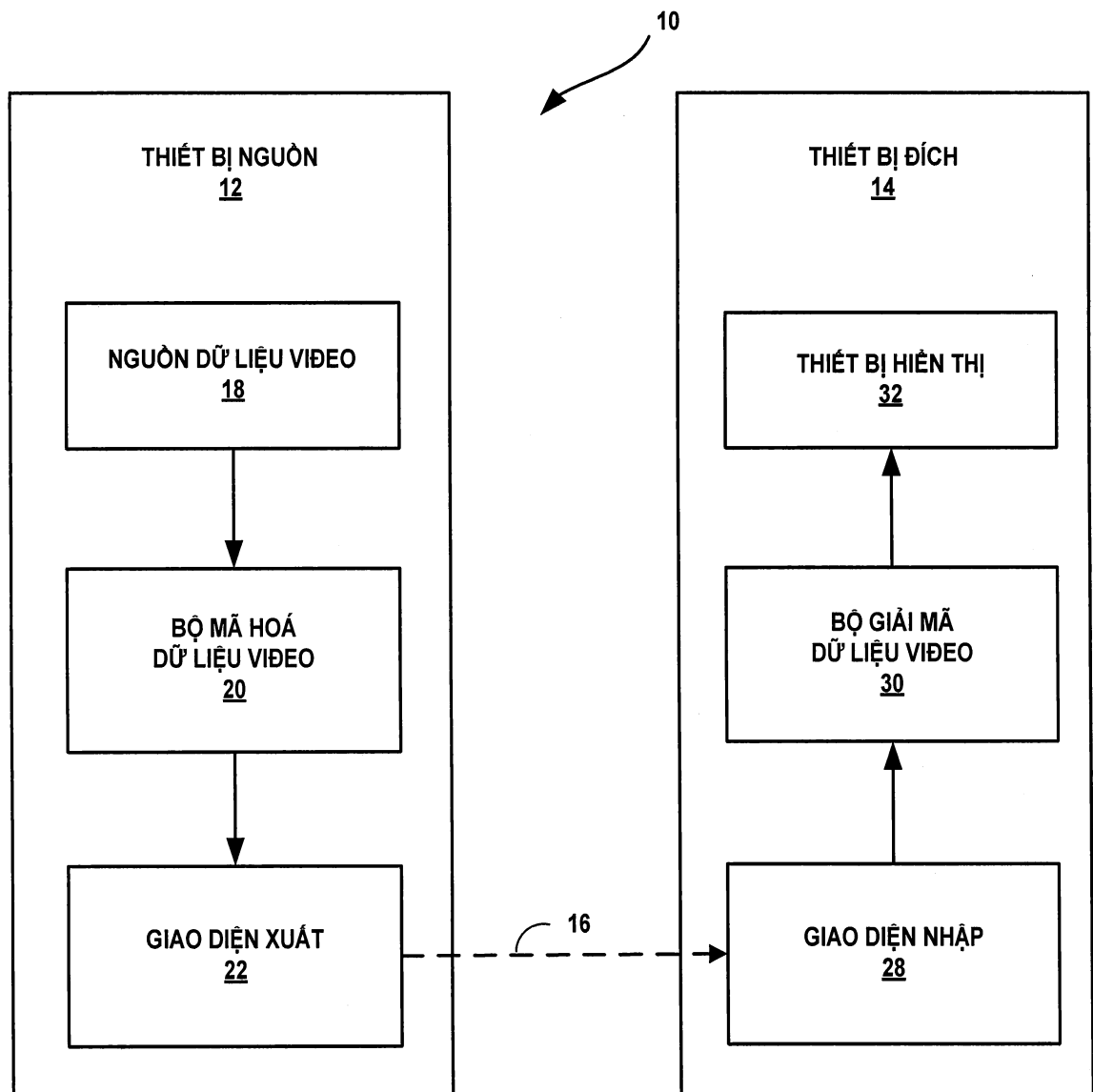


FIG. 1

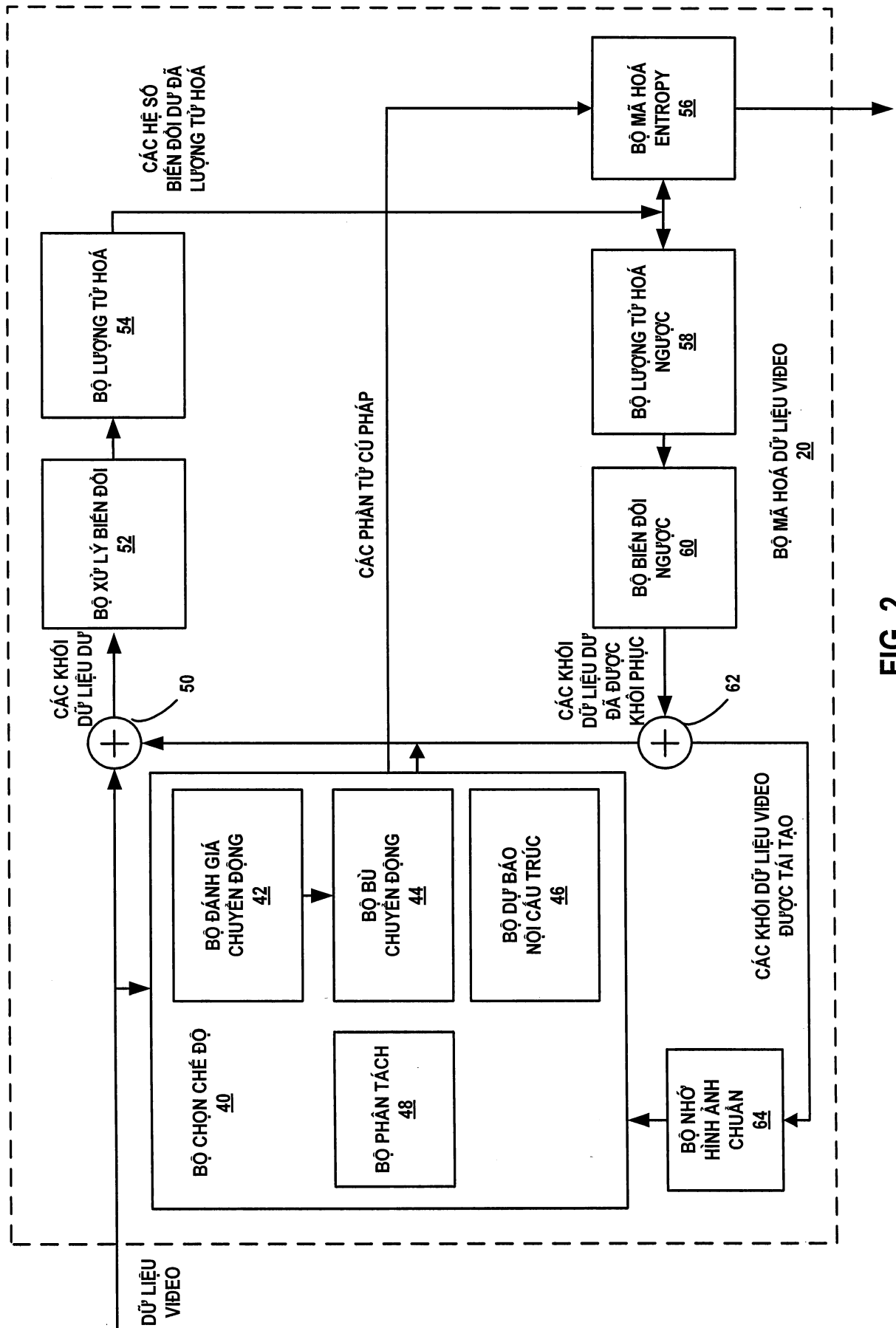


FIG. 2

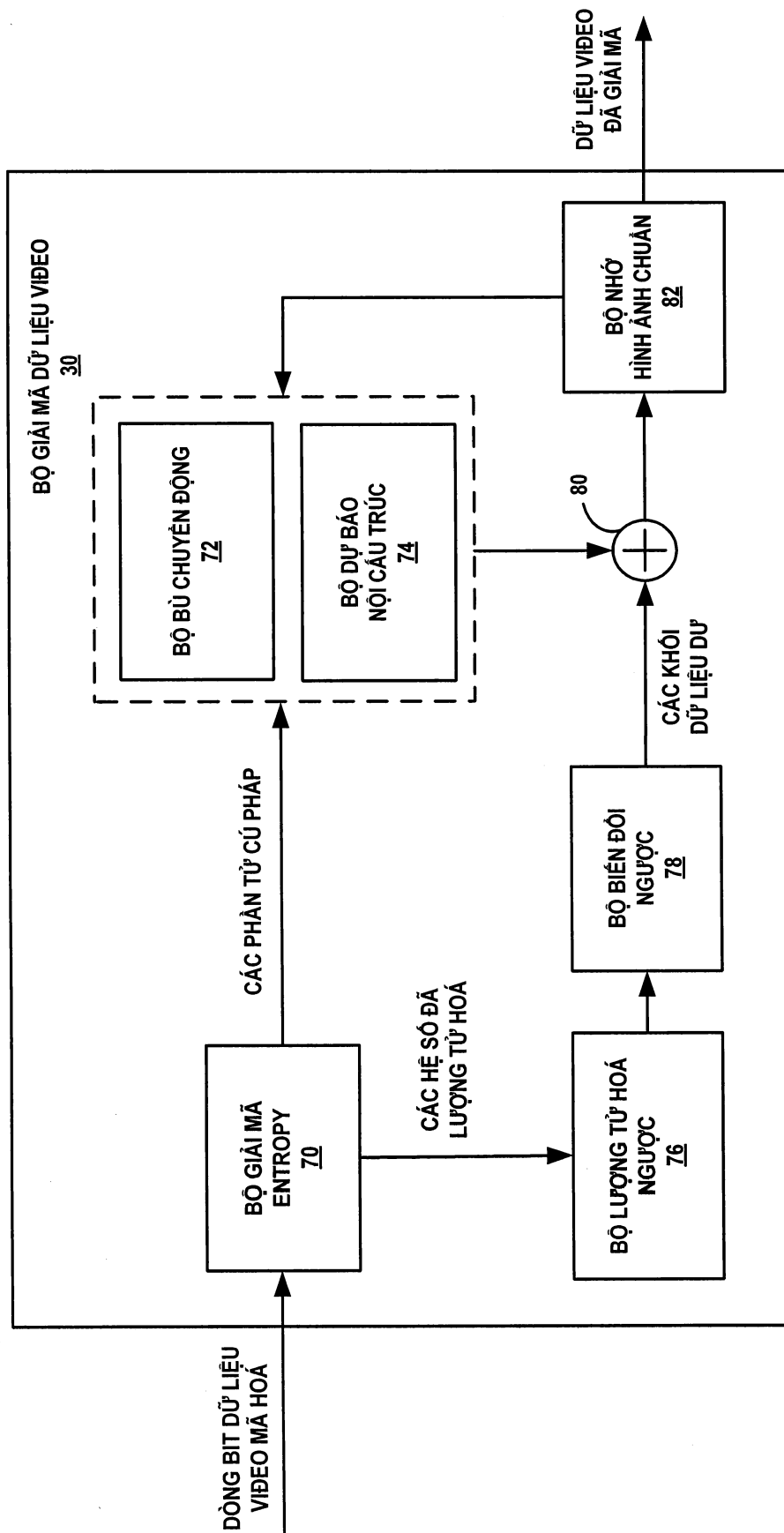
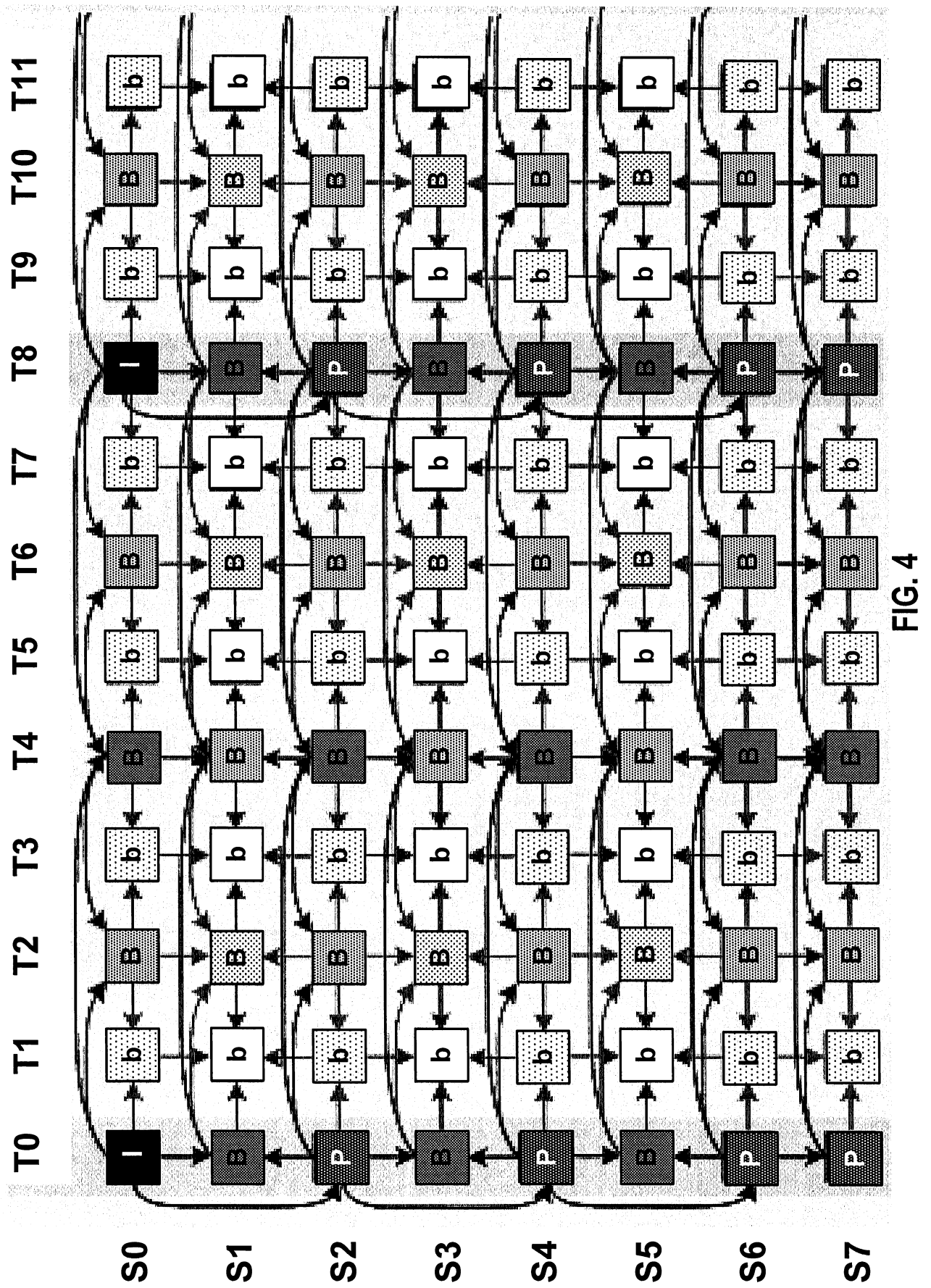


FIG. 3



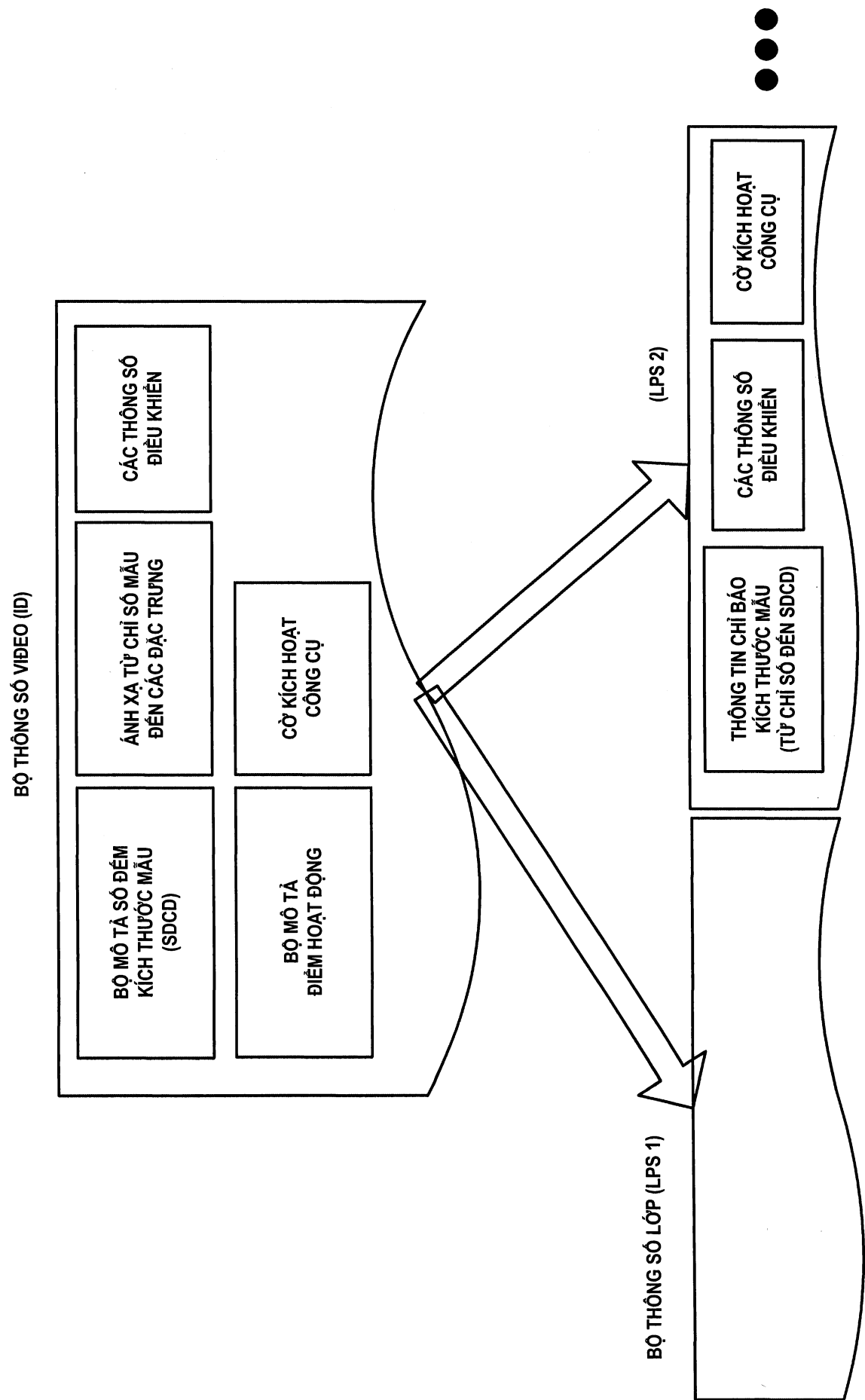


FIG. 5

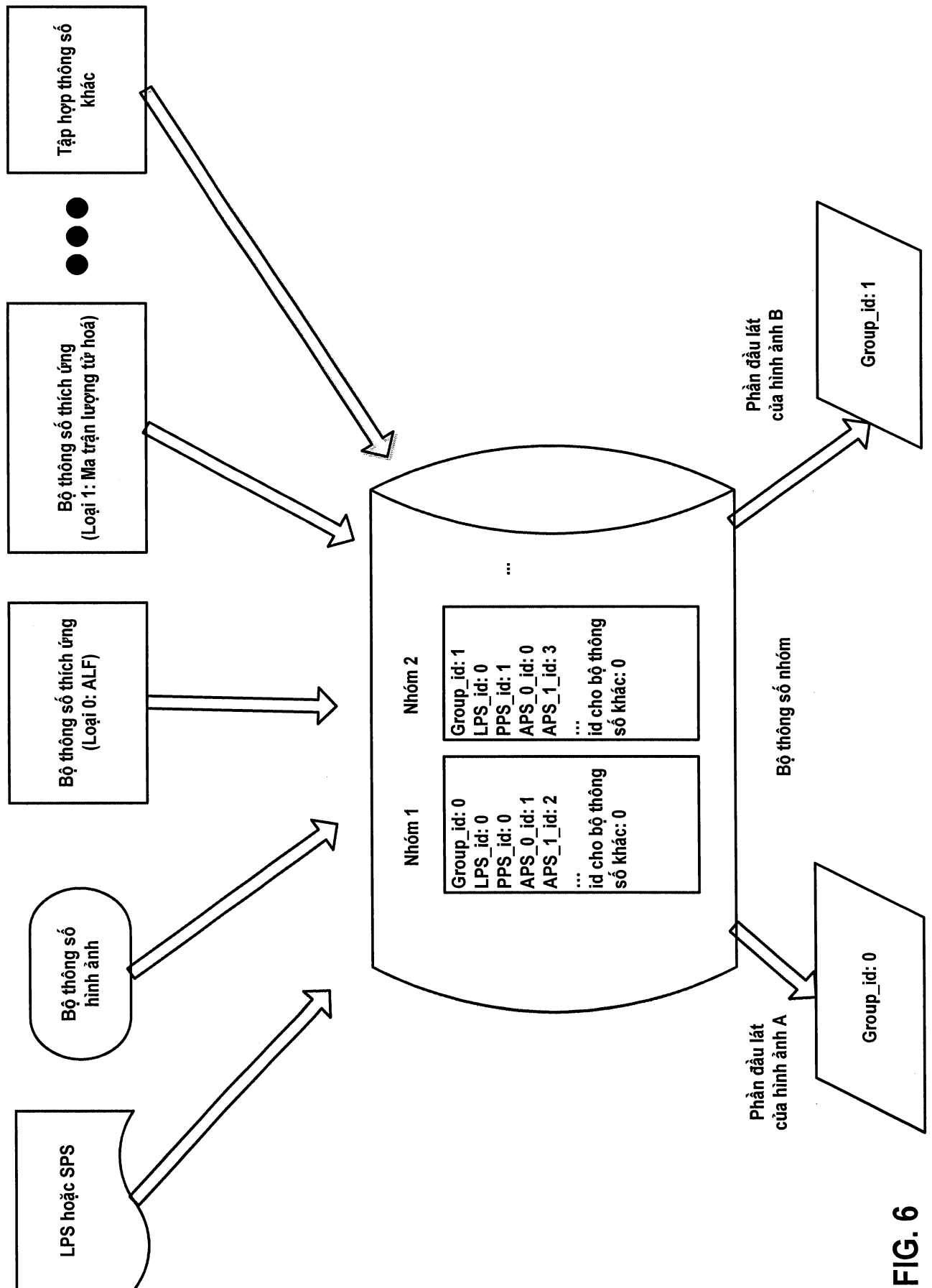


FIG. 6

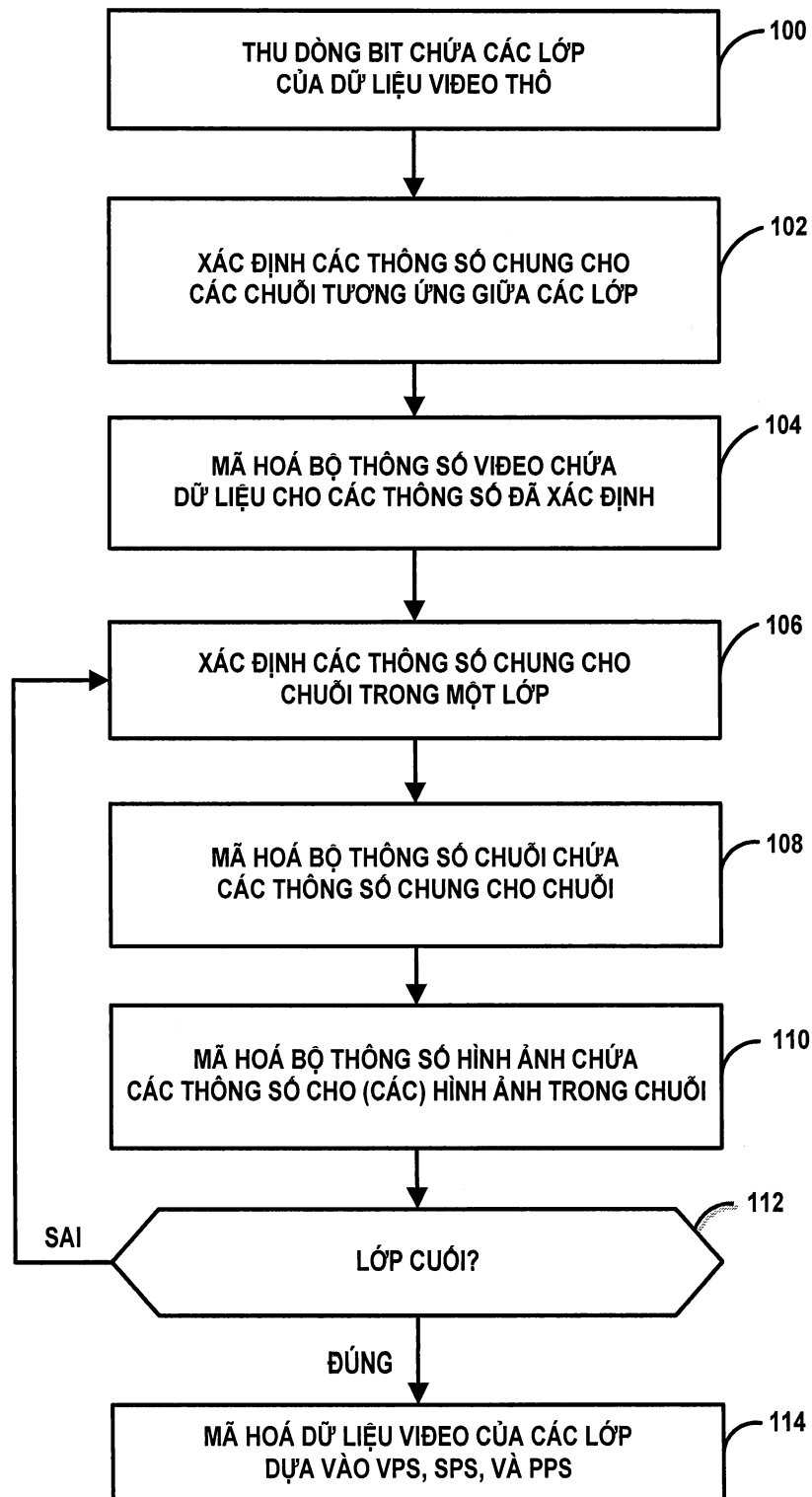


FIG. 7

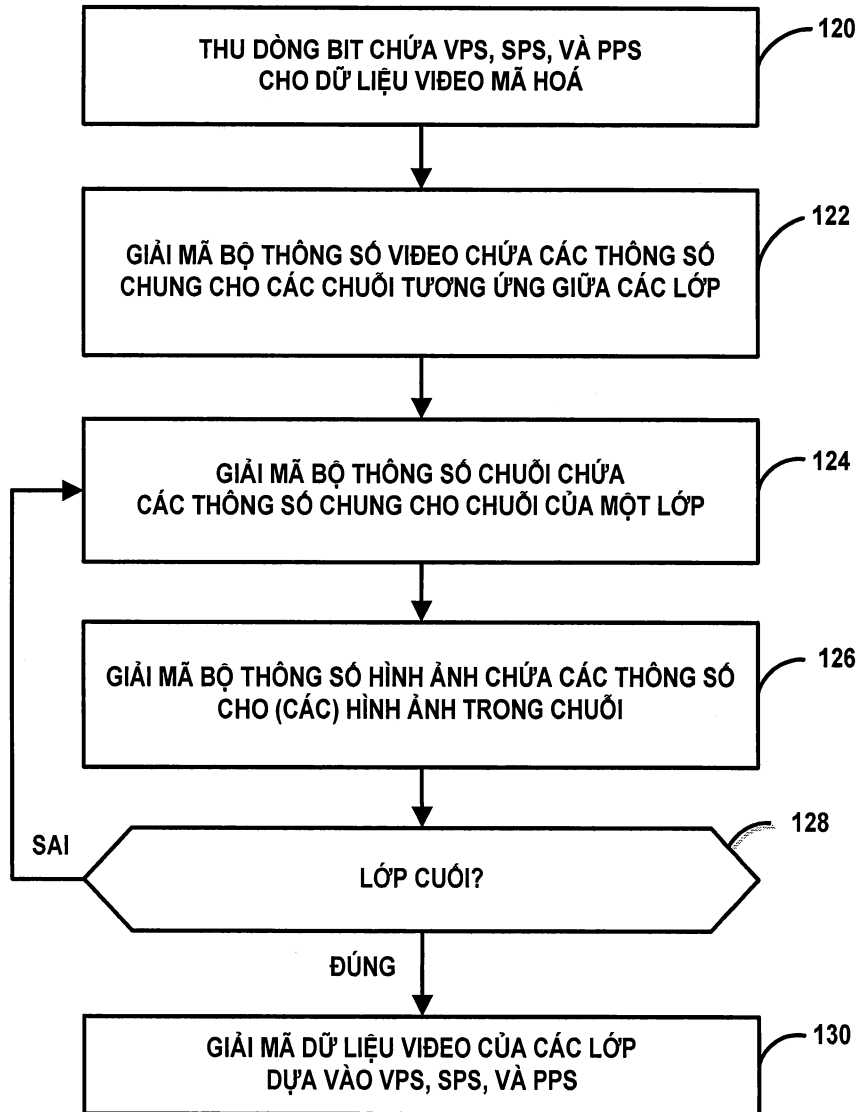


FIG. 8

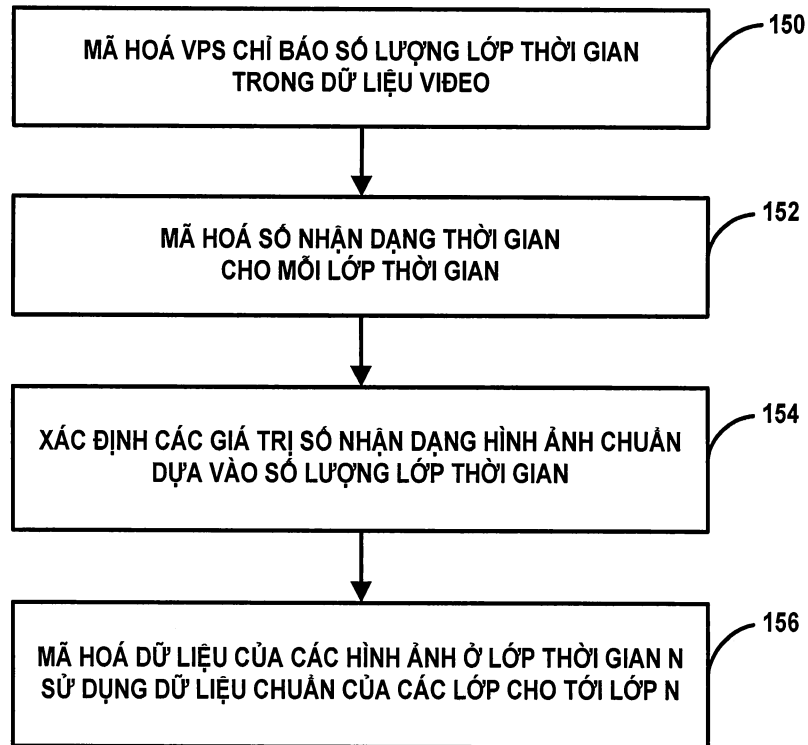


FIG. 9

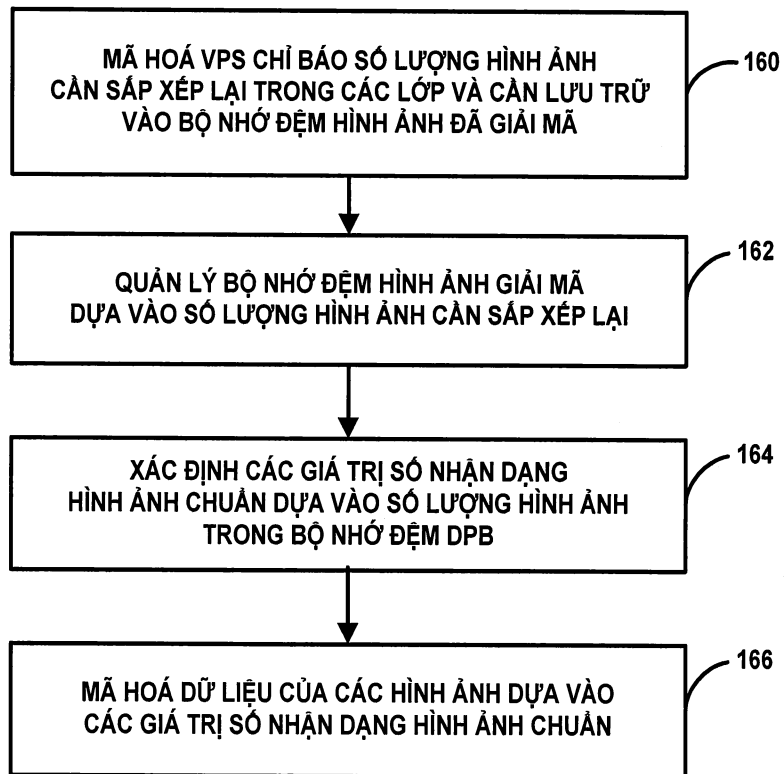


FIG. 10

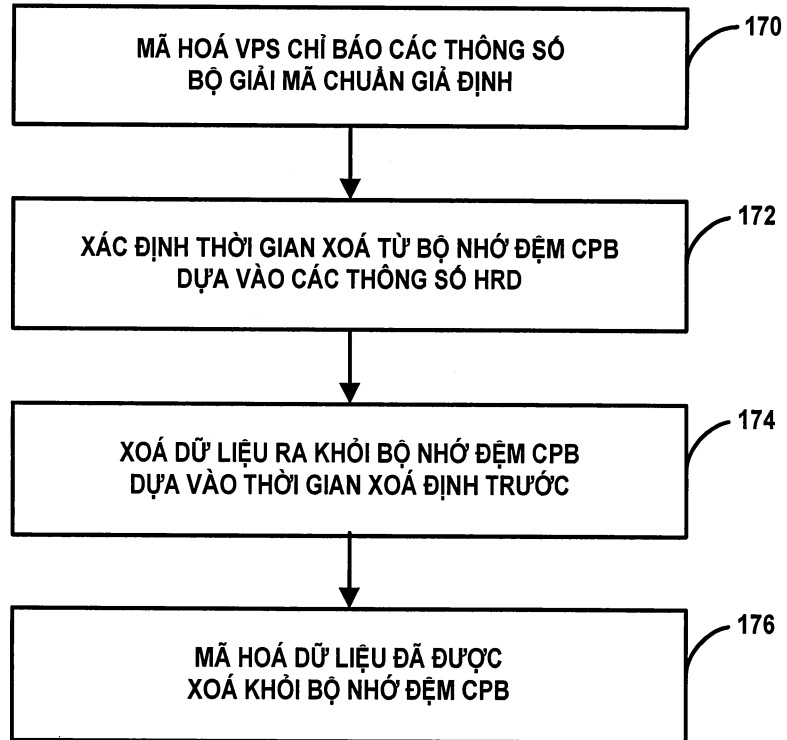


FIG. 11

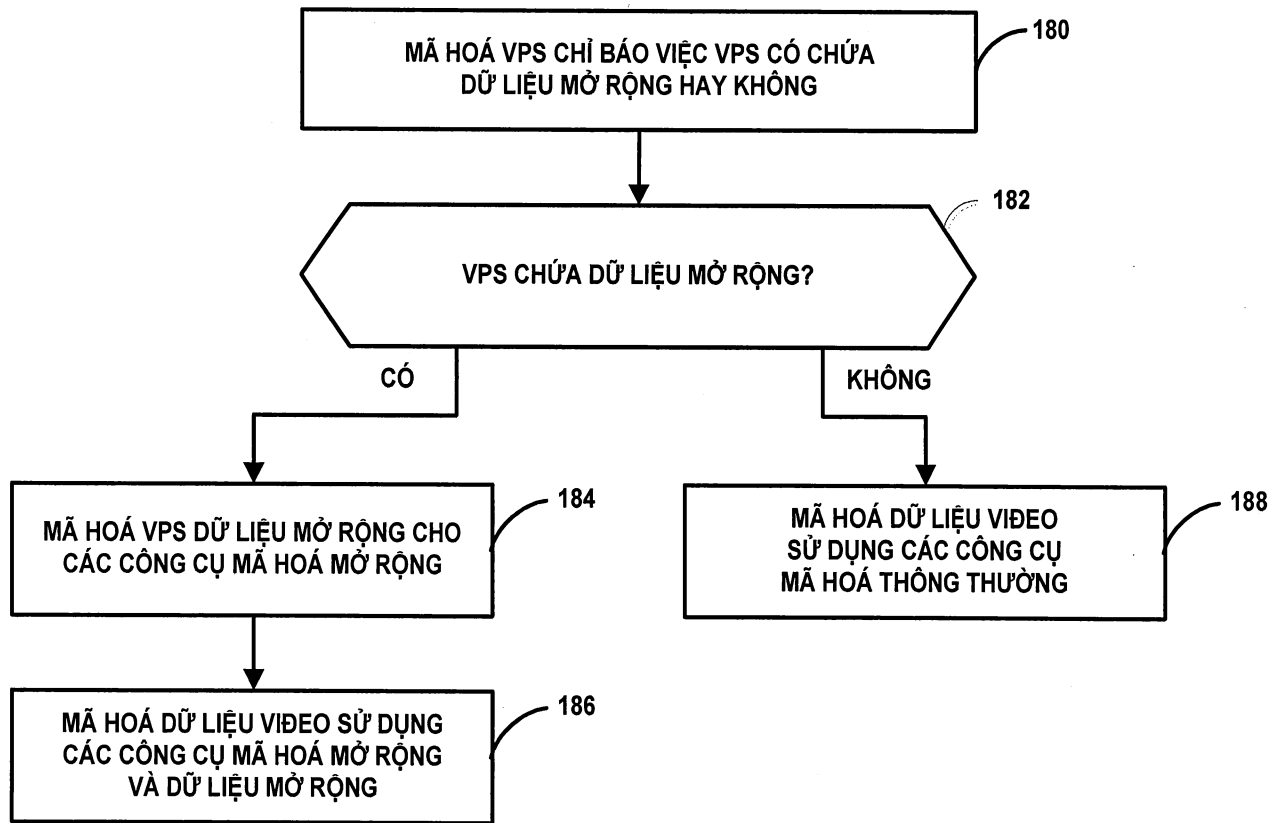


FIG. 12