



(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)



1-0053998

(51)^{2020.01} H04N 7/12

(13) B

(21) 1-2022-02457

(22) 21/09/2020

(86) PCT/US2020/051857 21/09/2020

(87) WO 2021/022262 04/02/2021

(30) 62/905,144 24/09/2019 US

(45) 25/11/2025 452

(43) 25/07/2022 412A

(73) HUAWEI TECHNOLOGIES CO., LTD. (CN)

Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong
518129, China

(72) WANG, Ye-Kui (US).

(74) Công ty Luật TNHH T&G (TGVN)

(54) PHƯƠNG PHÁP GIẢI MÃ, PHƯƠNG PHÁP MÃ HÓA, BỘ GIẢI MÃ, BỘ MÃ
HÓA, THIẾT BỊ LẬP MÃ VÀ PHƯƠNG TIỆN ĐỌC ĐƯỢC BẰNG MÁY TÍNH
KHÔNG TẠM THỜI

(21) 1-2022-02457

(57) Sáng chế đề cập đến phương pháp giải mã, phương pháp mã hóa, bộ giải mã, và bộ mã hóa. Cơ chế lập mã video được bộc lộ. Cơ chế này bao gồm bước mã hóa dòng bit bao gồm một hoặc nhiều lớp. Tín hiệu thông tin nâng cao bổ sung (SEI) lồng ghép có thể mở rộng được mã hóa thành dòng bit ở bộ phận lớp trừu tượng hóa mạng (NAL) SEI hiện thời. Tín hiệu SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tín hiệu SEI được lồng ghép có thể mở rộng và một hoặc nhiều ký hiệu nhận dạng (Id) lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tín hiệu SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời. Tập hợp của các thử nghiệm sự phù hợp dòng bit được thực hiện trên các lớp dựa vào các tín hiệu SEI được lồng ghép có thể mở rộng. Dòng bit được lưu trữ cho việc truyền thông về phía bộ giải mã.

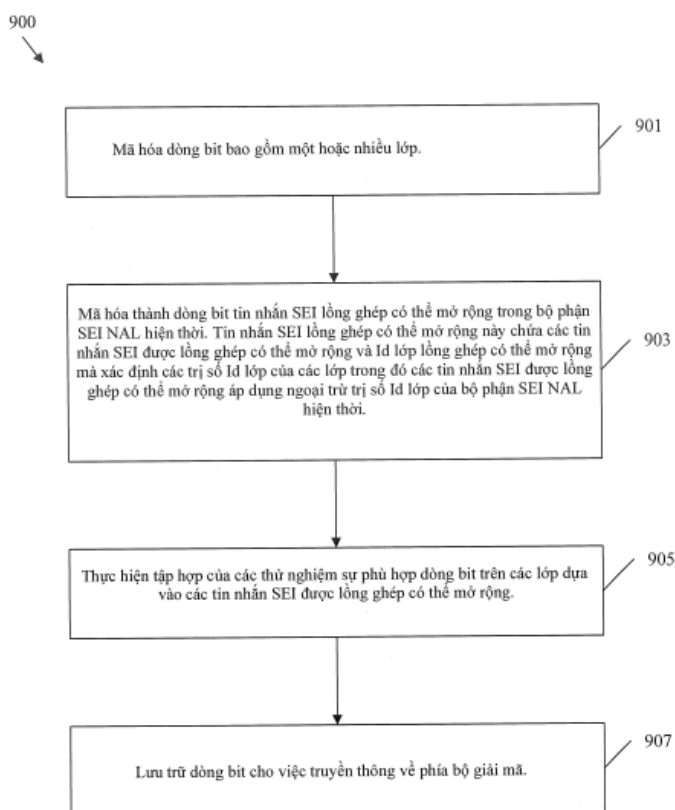


FIG. 9

Lĩnh vực kỹ thuật được đề cập

Sáng chế nói chung đề cập đến việc lập mã video, và cụ thể là đề cập đến các tin nhắn thông tin nâng cao bổ sung (supplemental enhancement information, viết tắt là SEI) lồng ghép có thể mở rộng được sử dụng để hỗ trợ các lớp mã hóa ở các dòng bit đa lớp.

Tình trạng kỹ thuật của sáng chế

Lượng dữ liệu video được cần để mô tả một video dù tương đối ngắn có thể là đáng kể, điều này có thể dẫn đến những khó khăn khi dữ liệu được tạo dòng hoặc theo cách khác được truyền thông qua mạng truyền thông với dung lượng độ rộng dải giới hạn. Do đó, dữ liệu video nói chung được nén trước khi được truyền thông qua các mạng viễn thông hiện nay. Kích thước của video cũng có thể là vấn đề khi video được lưu trữ trên thiết bị lưu trữ bởi vì các tài nguyên bộ nhớ có thể bị giới hạn. Các thiết bị nén video thường sử dụng phần mềm và/hoặc phần cứng ở nguồn để lập mã dữ liệu video trước khi truyền hoặc lưu trữ, nhờ đó làm giảm số lượng dữ liệu được cần để biểu diễn các ảnh video kỹ thuật số. Dữ liệu được nén sau đó được thu ở đích bởi thiết bị giải nén video mà giải mã dữ liệu video. Với các tài nguyên mạng giới hạn và các nhu cầu ngày càng tăng về chất lượng video cao hơn, các kỹ thuật nén và giải nén được cải tiến mà nâng cao tỷ lệ nén với chất lượng hình ảnh ở mức độ bị suy giảm ít đến không bị suy giảm là điều mong muốn.

Bản chất kỹ thuật của sáng chế

Theo phương án, sáng chế bao gồm phương pháp được thực hiện ở bộ giải mã, phương pháp này bao gồm các bước: thu, bởi bộ thu của bộ giải mã, dòng bit bao gồm một hoặc nhiều lớp và tin nhắn thông tin nâng cao bổ sung (supplemental enhancement information, viết tắt là SEI) lồng ghép có thể mở rộng ở bộ phận lớp trừu tượng hóa mạng (network abstraction layer, viết tắt là NAL) SEI hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều ký hiệu nhận dạng (identifier, viết tắt là Id) lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời; giải mã, bởi bộ xử lý của bộ giải

mã, ảnh được lập mã từ một hoặc nhiều lớp để tạo ra ảnh được giải mã; và chuyển tiếp, bởi bộ xử lý, ảnh được giải mã cho việc hiển thị như phần của chuỗi video được giải mã.

Một số hệ thống lập mã video ứng dụng các tin nhắn SEI. Tin nhắn SEI chứa thông tin mà không được cần bởi quy trình giải mã để xác định các trị số của các mẫu trong các ảnh được giải mã. Ví dụ, các tin nhắn SEI có thể chứa các thông số được sử dụng để kiểm tra dòng bit về sự phù hợp với các chuẩn. Trong một số trường hợp, video được lập mã thành dòng bit nhờ sử dụng nhiều lớp. Các tin nhắn SEI lồng ghép có thể mở rộng có thể được ứng dụng để chứa các tin nhắn SEI cho nhiều lớp. Ví dụ, tin nhắn SEI lồng ghép có thể mở rộng chứa các tin nhắn SEI được lồng ghép có thể mở rộng. Vấn đề có thể xảy ra khi danh mục của các Id lớp được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng để chỉ báo sự tương ứng giữa các tin nhắn SEI được lồng ghép có thể mở rộng và các lớp. Tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận SEI NAL, mà có thể được biểu thị như bộ phận SEI NAL hiện thời nhằm rõ ràng cho việc thảo luận. Bộ phận SEI NAL hiện thời được kết hợp với Id lớp. Tin nhắn SEI lồng ghép có thể mở rộng luôn đề cập đến lớp được kết hợp với bộ phận SEI NAL hiện thời. Điều này dẫn đến báo hiệu phần dư khi Id lớp của bộ phận SEI NAL hiện thời được bao gồm trong bộ phận SEI NAL hiện thời và cũng được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận NAL hiện thời.

Ví dụ này bao gồm các sự cải tiến với tin nhắn SEI lồng ghép có thể mở rộng dùng cho việc sử dụng cùng với các lớp được quy định. Cụ thể là, Id lớp có thể được gọi là Id lớp đoạn đầu bộ phận NAL (`nuh_layer_id`). `Nuh_layer_id` của bộ phận SEI NAL hiện thời được loại trừ khỏi tin nhắn SEI lồng ghép có thể mở rộng. Điều này có thể được thực hiện bằng cách ứng dụng thành phần cú pháp Id lớp lồng ghép có thể mở rộng (`layer_id[i]`) mà xác định `nuh_layer_id` cho mỗi lớp trong đó tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. `Layer_id[i]` lồng ghép có thể mở rộng bị ràng buộc để bao gồm chỉ `nuh_layer_ids` mà lớn hơn `nuh_layer_id` của bộ phận SEI NAL hiện thời. Do đó, `nuh_layer_id` của bộ phận SEI NAL hiện thời được bỏ qua khỏi tin nhắn SEI lồng ghép có thể mở rộng và được suy luận từ bộ phận SEI NAL hiện thời. Điều này nâng cao tin nhắn SEI lồng ghép có thể mở rộng bằng cách bỏ qua Id phần dư. Ví dụ, vòng lặp mã hóa/giải mã `layer_id[i]` lồng ghép có thể mở rộng thực thi ít hơn một lần, mà làm giảm các tài nguyên bộ xử lý trong suốt thời gian mã hóa và/hoặc giải mã. Hơn nữa, điều này làm giảm kích thước của dòng bit được mã hóa cho mỗi trong số các tin nhắn SEI lồng ghép có thể mở rộng ở dòng bit. Kết quả là, hiệu quả lập

mã được tăng lên, mà làm giảm việc sử dụng tài nguyên của bộ xử lý, bộ nhớ, và/hoặc bảo hiệu mạng ở cả bộ mã hóa và bộ giải mã.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó các Id lớp lồng ghép có thể mở rộng là thành phần cú pháp `layer_id[i]` lồng ghép có thể mở rộng mà xác định trị số `nuh_layer_id` của lớp thứ `i` trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng khi cờ tất cả các lớp lồng ghép có thể mở rộng (`all_layers_flag`) bằng không.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó mỗi trị số của `layer_id[i]` lồng ghép có thể mở rộng lớn hơn trị số `nuh_layer_id` của bộ phận SEI NAL hiện thời.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, còn bao gồm bước dẫn xuất, bởi bộ xử lý, biến số lượng lớp lồng ghép (`nestingNumLayers`) mà xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, còn bao gồm bước dẫn xuất, bởi bộ xử lý, danh mục của các Id lớp lồng ghép (`NestingLayerId[i]`) mà xác định danh mục của các trị số `nuh_layer_id` của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng đối với `i` nằm trong đoạn từ không đến `nestingNumLayers` trừ một.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó `nestingNumLayers` và `NestingLayerId[i]` được dẫn xuất như sau:

```
if( scalable nesting all_layers_flag ) {
    nestingNumLayers = vps_max_layers_minus1 + 1 -
GeneralLayerIdx[ nuh_layer_id ]
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
} else {
    nestingNumLayers = scalable nesting num_layers_minus1 + 1
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
}
```

trong đó `all_layers_flag` lồng ghép có thể mở rộng xác định xem các tin nhắn SEI được lồng ghép có thể mở rộng có áp dụng cho tất cả các lớp hay không, `vps_max_layers_minus1` cộng một xác định số lượng lớp được quy định bởi tập hợp thông số video (video parameter set, viết tắt là VPS), `GeneralLayerIdx` là biến xác định chỉ số lớp của lớp với `nuh_layer_id` bằng `vps_layer_id[i]`, `vps_layer_id[i]` xác định trị số `nuh_layer_id` của lớp thứ `i`, và `num_layers_minus1` cộng một lồng ghép có thể mở rộng xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó `num_layers_minus1` lồng ghép có thể mở rộng nằm trong đoạn từ không đến `vps_max_layers_minus1 - GeneralLayerIdx[nuh_layer_id]`, trong đó `nuh_layer_id` là `nuh_layer_id` của bộ phận SEI NAL hiện thời.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó ảnh được lập mã được giải mã dựa vào các tin nhắn SEI được lồng ghép có thể mở rộng.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó tin nhắn SEI lồng ghép có thể mở rộng bao gồm ít nhất một tin nhắn SEI được lồng ghép có thể mở rộng mà áp dụng cho lớp của bộ phận SEI NAL hiện thời.

Theo phương án, sáng chế bao gồm phương pháp được thực hiện ở bộ mã hóa, trong đó phương pháp này bao gồm các bước: mã hóa, bởi bộ xử lý của bộ mã hóa, dòng bit bao gồm một hoặc nhiều lớp; mã hóa thành dòng bit, bởi bộ xử lý, tin nhắn SEI lồng ghép có thể mở rộng ở bộ phận SEI NAL hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều Id lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng trong khi áp dụng sự ràng buộc để bỏ qua trị số Id lớp của bộ phận SEI NAL hiện thời; thực hiện, bởi bộ xử lý, tập hợp của các thử nghiệm sự phù hợp dòng bit trên các lớp dựa vào các tin nhắn SEI được lồng ghép có thể mở rộng; và lưu trữ, bởi bộ nhớ được ghép nối với bộ xử lý, dòng bit cho việc truyền thông về phía bộ giải mã.

Một số hệ thống lập mã video ứng dụng các tin nhắn SEI. Tin nhắn SEI chứa thông tin mà không được cần bởi quy trình giải mã để xác định các trị số của các mẫu ở các ảnh được giải mã. Ví dụ, các tin nhắn SEI có thể chứa các thông số được sử dụng để kiểm tra

dòng bit về sự phù hợp với các chuẩn. Trong một số trường hợp, video được lập mã thành dòng bit nhờ sử dụng nhiều lớp. Các tin nhắn SEI lồng ghép có thể mở rộng có thể được ứng dụng để chứa các tin nhắn SEI cho nhiều lớp. Ví dụ, tin nhắn SEI lồng ghép có thể mở rộng chứa các tin nhắn SEI được lồng ghép có thể mở rộng. Vấn đề có thể xảy ra khi danh mục của các Id lớp được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng để chỉ báo sự tương ứng giữa các tin nhắn SEI được lồng ghép có thể mở rộng và các lớp. Tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận SEI NAL, mà có thể được biểu thị như bộ phận SEI NAL hiện thời nhằm rõ ràng cho việc thảo luận. Bộ phận SEI NAL hiện thời được kết hợp với Id lớp. Tin nhắn SEI lồng ghép có thể mở rộng luôn đề cập đến lớp được kết hợp với bộ phận SEI NAL hiện thời. Điều này dẫn đến báo hiệu phần dư khi Id lớp của bộ phận SEI NAL hiện thời được bao gồm trong bộ phận SEI NAL hiện thời và cũng được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận NAL hiện thời.

Ví dụ này bao gồm các sự cải tiến với tin nhắn SEI lồng ghép có thể mở rộng dùng cho việc sử dụng cùng với các lớp được quy định. Cụ thể là, Id lớp có thể được đề cập đến là `nuh_layer_id`. `Nuh_layer_id` của bộ phận SEI NAL hiện thời được loại trừ khỏi tin nhắn SEI lồng ghép có thể mở rộng. Điều này có thể được thực hiện bằng cách ứng dụng thành phần cú pháp `layer_id[i]` lồng ghép có thể mở rộng mà xác định `nuh_layer_id` cho mỗi lớp trong đó tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. `Layer_id[i]` lồng ghép có thể mở rộng bị ràng buộc để bao gồm chỉ `nuh_layer_ids` mà lớn hơn `nuh_layer_id` của bộ phận SEI NAL hiện thời. Do đó, `nuh_layer_id` của bộ phận SEI NAL hiện thời được bỏ qua khỏi tin nhắn SEI lồng ghép có thể mở rộng và được suy luận từ bộ phận SEI NAL hiện thời. Điều này nâng cao tin nhắn SEI lồng ghép có thể mở rộng bằng cách bỏ qua Id phần dư. Ví dụ, vòng lặp mã hóa/giải mã `layer_id[i]` lồng ghép có thể mở rộng thực thi ít hơn một lần, mà làm giảm các tài nguyên bộ xử lý trong suốt thời gian mã hóa và/hoặc giải mã. Hơn nữa, điều này làm giảm kích thước của dòng bit được mã hóa cho mỗi trong số các tin nhắn SEI lồng ghép có thể mở rộng ở dòng bit. Kết quả là, hiệu quả lập mã được tăng lên, mà làm giảm việc sử dụng tài nguyên của bộ xử lý, bộ nhớ, và/hoặc báo hiệu mạng ở cả bộ mã hóa và bộ giải mã.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó Id lớp lồng ghép có thể mở rộng là thành phần cú pháp `layer_id[i]` lồng ghép có thể mở rộng mà xác định trị số `nuh_layer_id` của lớp thứ *i* trong đó

các tín hiệu SEI được lồng ghép có thể mở rộng áp dụng khi all_layers_flag lồng ghép có thể mở rộng bằng không.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó mỗi trị số của layer_id[i] lồng ghép có thể mở rộng lớn hơn trị số nuh_layer_id của bộ phận SEI NAL hiện thời.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó bước thực hiện tập hợp của các thử nghiệm sự phù hợp dòng bit còn bao gồm bước dẫn xuất biến số lượng lớp lồng ghép (nestingNumLayers) mà xác định số lượng lớp trong đó các tín hiệu SEI được lồng ghép có thể mở rộng áp dụng.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó bước thực hiện tập hợp của các thử nghiệm sự phù hợp dòng bit còn bao gồm bước dẫn xuất danh mục của các Id lớp lồng ghép (NestingLayerId[i]) mà xác định danh mục của các trị số nuh_layer_id của các lớp trong đó các tín hiệu SEI được lồng ghép có thể mở rộng áp dụng đối với i nằm trong đoạn từ không đến nestingNumLayers trừ một.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó nestingNumLayers và NestingLayerId[i] được dẫn xuất như sau:

```
if( scalable nesting all_layers_flag ) {
    nestingNumLayers = vps_max_layers_minus1 + 1 -
GeneralLayerIdx[ nuh_layer_id ]
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
} else {
    nestingNumLayers = scalable nesting num_layers_minus1 + 1
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
}
```

trong đó all_layers_flag lồng ghép có thể mở rộng xác định xem các tín hiệu SEI được lồng ghép có thể mở rộng có áp dụng cho tất cả các lớp hay không, vps_max_layers_minus1 cộng một xác định số lượng lớp được quy định bởi tập hợp thông số video (VPS), GeneralLayerIdx là biến xác định chỉ số lớp của lớp với nuh_layer_id bằng vps_layer_id[i], vps_layer_id[i]

xác định trị số `nuh_layer_id` của lớp thứ *i*, và `num_layers_minus1` cộng một lồng ghép có thể mở rộng xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó `num_layers_minus1` lồng ghép có thể mở rộng nằm trong đoạn từ không đến `vps_max_layers_minus1 - GeneralLayerIdx[nuh_layer_id]`, trong đó `nuh_layer_id` là `nuh_layer_id` của bộ phận SEI NAL hiện thời.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó tin nhắn SEI lồng ghép có thể mở rộng bao gồm ít nhất một tin nhắn SEI được lồng ghép có thể mở rộng mà áp dụng cho lớp của bộ phận SEI NAL hiện thời.

Theo phương án, sáng chế bao gồm thiết bị lập mã video bao gồm: bộ xử lý, bộ thu được ghép nối với bộ xử lý, bộ nhớ được ghép nối với bộ xử lý, và bộ truyền được ghép nối với bộ xử lý, trong đó bộ xử lý, bộ thu, bộ nhớ, và bộ truyền này được tạo cấu hình để thực hiện phương pháp theo khía cạnh bất kỳ trong số các khía cạnh nêu trên.

Theo phương án, sáng chế bao gồm phương tiện đọc được bởi máy tính không tạm thời bao gồm sản phẩm chương trình máy tính dùng cho việc sử dụng bởi thiết bị lập mã video, sản phẩm chương trình máy tính này bao gồm các lệnh có thể thực thi được bởi máy tính được lưu trữ trên phương tiện đọc được bởi máy tính không tạm thời sao cho khi được thực thi bởi bộ xử lý khiến thiết bị lập mã video thực hiện phương pháp theo khía cạnh bất kỳ trong số các khía cạnh nêu trên.

Theo phương án, sáng chế bao gồm bộ giải mã bao gồm: phương tiện thu để thu dòng bit bao gồm một hoặc nhiều lớp và tin nhắn SEI lồng ghép có thể mở rộng ở bộ phận SEI NAL hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều Id lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời; phương tiện giải mã để giải mã ảnh được lập mã từ một hoặc nhiều lớp để tạo ra ảnh được giải mã; và phương tiện chuyển tiếp để chuyển tiếp ảnh được giải mã cho việc hiển thị như phần của chuỗi video được giải mã.

Một số hệ thống lập mã video ứng dụng các tin nhắn SEI. Tin nhắn SEI chứa thông tin mà không được cần bởi quy trình giải mã để xác định các trị số của các mẫu ở các ảnh

được giải mã. Ví dụ, các tin nhắn SEI có thể chứa các thông số được sử dụng để kiểm tra dòng bit về sự phù hợp với các chuẩn. Trong một số trường hợp, video được lập mã thành dòng bit nhờ sử dụng nhiều lớp. Các tin nhắn SEI lồng ghép có thể mở rộng có thể được ứng dụng để chứa các tin nhắn SEI cho nhiều lớp. Ví dụ, tin nhắn SEI lồng ghép có thể mở rộng chứa các tin nhắn SEI được lồng ghép có thể mở rộng. Vấn đề có thể xảy ra khi danh mục của các Id lớp được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng để chỉ báo sự tương ứng giữa các tin nhắn SEI được lồng ghép có thể mở rộng và các lớp. Tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận SEI NAL, mà có thể được biểu thị như bộ phận SEI NAL hiện thời nhằm rõ ràng cho việc thảo luận. Bộ phận SEI NAL hiện thời được kết hợp với Id lớp. Tin nhắn SEI lồng ghép có thể mở rộng luôn đề cập đến lớp được kết hợp với bộ phận SEI NAL hiện thời. Điều này dẫn đến báo hiệu phần dư khi Id lớp của bộ phận SEI NAL hiện thời được bao gồm trong bộ phận SEI NAL hiện thời và cũng được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận NAL hiện thời.

Ví dụ này bao gồm các sự cải tiến với tin nhắn SEI lồng ghép có thể mở rộng dùng cho việc sử dụng cùng với các lớp được quy định. Cụ thể là, Id lớp có thể được đề cập đến là `nuh_layer_id`. `Nuh_layer_id` của bộ phận SEI NAL hiện thời được loại trừ khỏi tin nhắn SEI lồng ghép có thể mở rộng. Điều này có thể được thực hiện bằng cách ứng dụng thành phần cú pháp `layer_id[i]` lồng ghép có thể mở rộng mà xác định `nuh_layer_id` cho mỗi lớp trong đó tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. `Layer_id[i]` lồng ghép có thể mở rộng bị ràng buộc để bao gồm chỉ `nuh_layer_ids` mà lớn hơn `nuh_layer_id` của bộ phận SEI NAL hiện thời. Do đó, `nuh_layer_id` của bộ phận SEI NAL hiện thời được bỏ qua khỏi tin nhắn SEI lồng ghép có thể mở rộng và được suy luận từ bộ phận SEI NAL hiện thời. Điều này nâng cao tin nhắn SEI lồng ghép có thể mở rộng bằng cách bỏ qua Id phần dư. Ví dụ, vòng lặp mã hóa/giải mã `layer_id[i]` lồng ghép có thể mở rộng thực thi ít hơn một lần, mà làm giảm các tài nguyên bộ xử lý trong suốt thời gian mã hóa và/hoặc giải mã. Hơn nữa, điều này làm giảm kích thước của dòng bit được mã hóa cho mỗi trong số các tin nhắn SEI lồng ghép có thể mở rộng ở dòng bit. Kết quả là, hiệu quả lập mã được tăng lên, mà làm giảm việc sử dụng tài nguyên của bộ xử lý, bộ nhớ, và/hoặc báo hiệu mạng ở cả bộ mã hóa và bộ giải mã.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó bộ giải mã còn được tạo cấu hình để thực hiện phương pháp theo khía cạnh bất kỳ trong số các khía cạnh nêu trên.

Theo phương án, sáng chế bao gồm bộ mã hóa bao gồm: phương tiện mã hóa để: mã hóa dòng bit bao gồm một hoặc nhiều lớp; và mã hóa thành dòng bit tin nhắn SEI lồng ghép có thể mở rộng ở bộ phận SEI NAL hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều Id lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng trong khi áp dụng sự ràng buộc để bỏ qua trị số Id lớp của bộ phận SEI NAL hiện thời; phương tiện HRD để thực hiện tập hợp của các thử nghiệm sự phù hợp dòng bit trên các lớp dựa vào các tin nhắn SEI được lồng ghép có thể mở rộng; và phương tiện lưu trữ để lưu trữ dòng bit cho việc truyền thông về phía bộ giải mã.

Một số hệ thống lập mã video ứng dụng các tin nhắn SEI. Tin nhắn SEI chứa thông tin mà không được cần bởi quy trình giải mã để xác định các trị số của các mẫu ở các ảnh được giải mã. Ví dụ, các tin nhắn SEI có thể chứa các thông số được sử dụng để kiểm tra dòng bit về sự phù hợp với các chuẩn. Trong một số trường hợp, video được lập mã thành dòng bit nhờ sử dụng nhiều lớp. Các tin nhắn SEI lồng ghép có thể mở rộng có thể được ứng dụng để chứa các tin nhắn SEI cho nhiều lớp. Ví dụ, tin nhắn SEI lồng ghép có thể mở rộng chứa các tin nhắn SEI được lồng ghép có thể mở rộng. Vấn đề có thể xảy ra khi danh mục của các Id lớp được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng để chỉ báo sự tương ứng giữa các tin nhắn SEI được lồng ghép có thể mở rộng và các lớp. Tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận SEI NAL, mà có thể được biểu thị như bộ phận SEI NAL hiện thời nhằm rõ ràng cho việc thảo luận. Bộ phận SEI NAL hiện thời được kết hợp với Id lớp. Tin nhắn SEI lồng ghép có thể mở rộng luôn đề cập đến lớp được kết hợp với bộ phận SEI NAL hiện thời. Điều này dẫn đến báo hiệu phần dư khi Id lớp của bộ phận SEI NAL hiện thời được bao gồm trong bộ phận SEI NAL hiện thời và cũng được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận NAL hiện thời.

Ví dụ này bao gồm các sự cải tiến với tin nhắn SEI lồng ghép có thể mở rộng dùng cho việc sử dụng cùng với các lớp được quy định. Cụ thể là, Id lớp có thể được đề cập đến là `nuh_layer_id`. `Nuh_layer_id` của bộ phận SEI NAL hiện thời được loại trừ khỏi tin nhắn

SEI lồng ghép có thể mở rộng. Điều này có thể được thực hiện bằng cách ứng dụng thành phần cú pháp `layer_id[i]` lồng ghép có thể mở rộng mà xác định `nuh_layer_id` cho mỗi lớp trong đó tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. `Layer_id[i]` lồng ghép có thể mở rộng bị ràng buộc để bao gồm chỉ `nuh_layer_ids` mà lớn hơn `nuh_layer_id` của bộ phận SEI NAL hiện thời. Do đó, `nuh_layer_id` của bộ phận SEI NAL hiện thời được bỏ qua khỏi tin nhắn SEI lồng ghép có thể mở rộng và được suy luận từ bộ phận SEI NAL hiện thời. Điều này nâng cao tin nhắn SEI lồng ghép có thể mở rộng bằng cách bỏ qua Id phần dư. Ví dụ, vòng lặp mã hóa/giải mã `layer_id[i]` lồng ghép có thể mở rộng thực thi ít hơn một lần, mà làm giảm các tài nguyên bộ xử lý trong suốt thời gian mã hóa và/hoặc giải mã. Hơn nữa, điều này làm giảm kích thước của dòng bit được mã hóa cho mỗi trong số các tin nhắn SEI lồng ghép có thể mở rộng ở dòng bit. Kết quả là, hiệu quả lập mã được tăng lên, mà làm giảm việc sử dụng tài nguyên của bộ xử lý, bộ nhớ, và/hoặc báo hiệu mạng ở cả bộ mã hóa và bộ giải mã.

Một cách tùy chọn, theo bất kỳ một trong số các khía cạnh nêu trên, cách thực hiện khác của khía cạnh đề xuất, trong đó bộ mã hóa còn được tạo cấu hình để thực hiện phương pháp theo khía cạnh bất kỳ trong số các khía cạnh nêu trên.

Nhằm mục đích rõ ràng, bất kỳ một trong số các phương án nêu trên có thể được kết hợp với bất kỳ một hoặc nhiều trong số các phương án nêu trên còn lại để tạo ra phương án mới nằm trong phạm vi của sáng chế.

Các dấu hiệu này và các dấu hiệu khác sẽ được hiểu một cách rõ ràng hơn từ phần mô tả chi tiết sau đây được thực hiện cùng với các hình vẽ và các điểm yêu cầu bảo hộ kèm theo.

Mô tả vắn tắt các hình vẽ

Để hiểu đầy đủ hơn về sáng chế này, tham chiếu bây giờ được thực hiện tới phần mô tả ngắn gọn sau đây, được thực hiện kết hợp với các hình vẽ và phần mô tả chi tiết kèm theo, trong đó các số chỉ dẫn giống nhau biểu diễn các phần giống nhau.

Fig.1 là lưu đồ của phương pháp ví dụ về việc lập mã tín hiệu video.

Fig.2 là hình vẽ giản lược của hệ thống lập mã và giải mã (mã hóa và giải mã) ví dụ về việc lập mã video.

Fig.3 là hình vẽ giản lược minh họa bộ mã hóa video ví dụ.

Fig.4 là hình vẽ giản lược minh họa bộ giải mã video ví dụ.

Fig.5 là hình vẽ giản lược minh họa bộ giải mã tham chiếu giả định (hypothetical reference decoder, viết tắt là HRD) ví dụ.

Fig.6 là hình vẽ giản lược minh họa chuỗi video đa lớp ví dụ được tạo cấu hình cho việc dự đoán liên lớp.

Fig.7 là hình vẽ giản lược minh họa dòng bit ví dụ.

Fig.8 là hình vẽ giản lược của thiết bị lập mã video ví dụ.

Fig.9 là lưu đồ của phương pháp ví dụ về việc mã hóa chuỗi video thành dòng bit bao gồm các tin nhắn SEI lồng ghép có thể mở rộng mà áp dụng cho các lớp.

Fig.10 là lưu đồ của phương pháp ví dụ về việc giải mã chuỗi video từ dòng bit bao gồm các tin nhắn SEI lồng ghép có thể mở rộng mà áp dụng cho các lớp.

Fig.11 là hình vẽ giản lược của hệ thống ví dụ về việc lập mã chuỗi video nhờ sử dụng dòng bit bao gồm các tin nhắn SEI lồng ghép có thể mở rộng mà áp dụng cho các lớp.

Mô tả chi tiết sáng chế

Cần hiểu ngay từ đầu rằng mặc dù cách thực hiện minh họa của một hoặc nhiều phương án được đưa ra dưới đây, nhưng các hệ thống và/hoặc các phương pháp được bộc lộ có thể được thực hiện nhờ sử dụng số các kỹ thuật bất kỳ, cho dù hiện đã biết hay đang tồn tại hay không. Sáng chế sẽ không giới hạn ở các cách thực hiện, các hình vẽ, và các kỹ thuật minh họa được minh họa dưới đây, bao gồm các thiết kế làm ví dụ và các cách thực hiện được minh họa và được mô tả ở đây, nhưng có thể được điều chỉnh nằm trong phạm vi của các điểm yêu cầu bảo hộ kèm theo cùng với phạm vi đầy đủ của các sự tương đương của chúng.

Các thuật ngữ sau đây được định rõ như sau trừ khi được sử dụng theo ngữ cảnh trái ngược ở đây. Cụ thể là, các định nghĩa sau đây được nhằm để đưa ra sự rõ ràng thêm cho sáng chế. Tuy nhiên, các thuật ngữ có thể được mô tả khác nhau theo các ngữ cảnh khác nhau. Theo đó, các định nghĩa sau đây sẽ được coi là phần bổ sung và sẽ không được coi là giới hạn các định nghĩa khác bất kỳ của các phần mô tả được đưa ra đối với các thuật ngữ như vậy ở đây.

Dòng bit có thể là chuỗi của các bit bao gồm dữ liệu video mà được nén cho việc truyền giữa bộ mã hóa và bộ giải mã. Bộ mã hóa có thể là thiết bị mà được tạo cấu hình để ứng dụng các quy trình mã hóa để nén dữ liệu video thành dòng bit. Bộ giải mã có thể là thiết bị mà được tạo cấu hình để ứng dụng các quy trình giải mã để tái cấu trúc dữ liệu video từ dòng bit nhằm hiển thị. Ảnh có thể là mảng của các mẫu độ chói và/hoặc mảng của các mẫu sắc độ mà tạo ra khung hoặc trường của chúng. Ảnh mà đang được mã hóa hoặc được giải mã có thể được đề cập đến là ảnh hiện thời nhằm rõ ràng cho việc thảo luận. Ảnh được lập mã có thể là sự biểu diễn được lập mã của ảnh bao gồm các bộ phận lớp trừu tượng hóa

mạng (network abstraction layer, viết tắt là NAL) lớp lập mã video (video coding layer, viết tắt là VCL) với trị số cụ thể của ký hiệu nhận dạng lớp đoạn đầu bộ phận NAL (`nuh_layer_id`) nằm trong bộ phận truy cập (access unit, viết tắt là AU) và chứa tất cả các bộ phận cây lập mã (coding tree unit, viết tắt là CTU) của ảnh. Ảnh được giải mã có thể là ảnh được tạo ra bằng cách áp dụng quy trình giải mã cho ảnh được lập mã. AU có thể là tập hợp của các ảnh được lập mã mà được bao gồm ở các lớp khác nhau và được kết hợp với cùng thời gian cho việc đưa ra từ bộ đệm ảnh được giải mã (decoded picture buffer, viết tắt là DPB). Bộ phận NAL có thể là cấu trúc cú pháp chứa dữ liệu ở dạng của tải trọng chuỗi byte thô (Raw Byte Sequence Payload, viết tắt là RBSP), sự chỉ báo của loại dữ liệu, và được xen kẽ như được mong muốn với các byte ngăn ngừa mô phỏng. Bộ phận VCL NAL có thể là bộ phận NAL được lập mã để chứa dữ liệu video, chẳng hạn như lát được lập mã của ảnh. Bộ phận không phải VCL NAL có thể là bộ phận NAL mà chứa không phải dữ liệu video chẳng hạn như cú pháp và/hoặc các thông số mà hỗ trợ việc giải mã dữ liệu video, hiệu suất của việc kiểm tra sự phù hợp, hoặc các thao tác khác. Lớp có thể là tập hợp của các bộ phận VCL NAL mà dùng chung đặc tính được quy định (ví dụ, độ phân giải chung, tỷ lệ khung, kích thước hình ảnh chẳng hạn) như được chỉ báo bởi Id (identifier) lớp và các bộ phận không phải NAL VCL được kết hợp. Ký hiệu nhận dạng lớp đoạn đầu bộ phận NAL (`nuh_layer_id`) có thể là thành phần cú pháp mà xác định ký hiệu nhận dạng của lớp mà bao gồm bộ phận NAL. Tập hợp thông số video (VPS) có thể là bộ phận dữ liệu mà chứa các thông số đề cập đến toàn bộ video. Chuỗi video được lập mã có thể là tập hợp của một hoặc nhiều ảnh được lập mã. Chuỗi video được giải mã có thể là tập hợp của một hoặc nhiều ảnh được giải mã.

Bộ giải mã tham chiếu giả định (hypothetical reference decoder, viết tắt là HRD) có thể là mô hình bộ giải mã hoạt động trên bộ mã hóa mà kiểm tra sự biến đổi của các dòng bit được tạo ra bởi quy trình mã hóa để xác minh sự phù hợp với các sự ràng buộc được quy định. Thử nghiệm sự phù hợp dòng bit có thể là thử nghiệm để xác định xem dòng bit được mã hóa tuân theo chuẩn hay không, chẳng hạn như lập mã video đa năng (Versatile Video Coding, viết tắt là VVC). Các thông số HRD là các thành phần cú pháp mà khởi tạo và/hoặc định rõ các điều kiện hoạt động của HRD. Các thông số HRD có thể được bao gồm trong các tin nhắn thông tin nâng cao bổ sung (SEI). Tin nhắn SEI có thể là cấu trúc cú pháp với ngữ nghĩa được quy định mà bao gồm thông tin mà không được cần bởi quy trình giải mã để xác định các trị số của các mẫu ở các ảnh được giải mã. Bộ phận SEI NAL có thể là bộ phận NAL mà chứa một hoặc nhiều tin nhắn SEI. Bộ phận SEI NAL cụ thể có thể được đề

cập đến là bộ phận SEI NAL hiện thời. Tin nhắn SEI lồng ghép có thể mở rộng có thể là tin nhắn mà chứa nhiều tin nhắn SEI mà tương ứng với một hoặc nhiều tập hợp lớp đầu ra (output layer set, viết tắt là OLS) hoặc một hoặc nhiều lớp. Tin nhắn SEI khoảng đệm (buffer period, viết tắt là BP) có thể là tin nhắn SEI mà chứa các thông số HRD để khởi tạo HRD để quản lý bộ đệm ảnh được lập mã (coded picture buffer, viết tắt là CPB). Tin nhắn SEI thời điểm ảnh (picture timing, viết tắt là PT) có thể là tin nhắn SEI mà chứa các thông số HRD để quản lý thông tin phân phối cho các bộ phận truy cập (AU) ở CPB và/hoặc bộ đệm ảnh được giải mã (DPB). Tin nhắn SEI thông tin bộ phận giải mã (decoding unit information, viết tắt là DUI) có thể là tin nhắn SEI mà chứa các thông số HRD để quản lý thông tin phân phối cho các DU ở CPB và/hoặc DPB.

Tin nhắn SEI lồng ghép có thể mở rộng có thể là tập hợp của các tin nhắn SEI được lồng ghép có thể mở rộng. Tin nhắn SEI được lồng ghép có thể mở rộng có thể là tin nhắn SEI mà được lồng ghép bên trong tin nhắn SEI lồng ghép có thể mở rộng. Id lớp lồng ghép có thể mở rộng (layer_id[i]) có thể là thành phần cú pháp trong tin nhắn SEI lồng ghép có thể mở rộng mà xác định trị số nuh_layer_id của lớp thứ i trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. Cờ tất cả các lớp lồng ghép có thể mở rộng (all_layers_flag) có thể là thành phần cú pháp trong tin nhắn SEI lồng ghép có thể mở rộng mà xác định xem các tin nhắn SEI được lồng ghép có thể mở rộng trong tin nhắn SEI lồng ghép có thể mở rộng có áp dụng cho tất cả các lớp hay không. Số các lớp trừ một lồng ghép có thể mở rộng (num_layers_minus1) có thể là thành phần cú pháp trong tin nhắn SEI lồng ghép có thể mở rộng mà xác định số các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. Các lớp lớn nhất VPS trừ một (vps_max_layers_minus1) có thể là thành phần cú pháp trong VPS mà xác định số lượng lớp được quy định bởi VPS. Id lớp VPS (vps_layer_id[i]) có thể là thành phần cú pháp trong VPS mà xác định trị số nuh_layer_id của lớp thứ i. Số lượng lớp lồng ghép (nestingNumLayers) có thể là biến mà được thiết đặt tới số các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng dựa vào dữ liệu ở dòng bit. Danh mục của các Id lớp lồng ghép (NestingLayerId[i]) có thể là biến mà được thiết đặt để xác định danh mục của các trị số nuh_layer_id của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng cho mỗi lớp i dựa vào dữ liệu ở dòng bit. Chỉ số lớp chung (GeneralLayerIdx) có thể là biến được thiết đặt để xác định chỉ số lớp của mỗi lớp với nuh_layer_id dựa vào dữ liệu ở dòng bit.

Các từ viết tắt sau đây được sử dụng ở đây, bộ phận truy cập (AU), khối cây lập mã (Coding Tree Block, viết tắt là CTB), bộ phận cây lập mã (Coding Tree Unit, viết tắt là CTU), bộ phận lập mã (Coding Unit, viết tắt là CU), chuỗi video lớp được lập mã (Coded Layer Video Sequence, viết tắt là CLVS), bắt đầu chuỗi video lớp được lập mã (Coded Layer Video Sequence Start, viết tắt là CLVSS), chuỗi video được lập mã (Coded Video Sequence, viết tắt là CVS), bắt đầu chuỗi video được lập mã (Coded Video Sequence Start, viết tắt là CVSS), nhóm chuyên gia video liên kết (joint video experts team, viết tắt là JVET), tập hợp miếng bị ràng buộc chuyển động (Motion Constrained Tile Set, viết tắt là MCTS), bộ phận chuyển đổi lớn nhất (Maximum Transfer Unit, viết tắt là MTU), lớp trừu tượng hóa mạng (NAL), tập hợp lớp đầu ra (Output Layer Set, viết tắt là OLS), số đếm thứ tự ảnh (Picture Order Count, viết tắt là POC), điểm truy cập ngẫu nhiên (Random Access Point, viết tắt là RAP), tải trọng chuỗi byte thô (RBSP), tập hợp thông số chuỗi (Sequence Parameter Set, viết tắt là SPS), tập hợp thông số video (VPS), lập mã video đa năng (VVC).

Nhiều kỹ thuật nén video có thể được ứng dụng để làm giảm kích thước của các tệp video với sự tổn thất dữ liệu thấp nhất. Ví dụ, các kỹ thuật nén video có thể bao gồm bước thực hiện việc dự đoán theo không gian (ví dụ, ảnh bên trong) và/hoặc việc dự đoán theo thời gian (ví dụ, ảnh liên kết) để làm giảm hoặc loại bỏ phần dư dữ liệu trong các chuỗi video. Đối với việc lập mã video trên cơ sở khối, lát video (ví dụ, ảnh video hoặc phần của ảnh video) có thể được phân chia thành các khối video, mà cũng có thể được đề cập đến là các khối cây, các khối cây lập mã (coding tree block, viết tắt là CTB), các bộ phận cây lập mã (CTU), các bộ phận lập mã (CU), và/hoặc các nút lập mã. Các khối video trong lát được lập mã bên trong (I) của ảnh được lập mã nhờ sử dụng việc dự đoán theo không gian đối với các mẫu tham chiếu trong các khối lân cận trong cùng ảnh. Các khối video trong lát dự đoán đơn hướng (unidirectional prediction, viết tắt là P) hoặc dự đoán hai chiều (bidirectional prediction, viết tắt là B) được lập mã liên kết của ảnh có thể được lập mã bằng cách ứng dụng việc dự đoán theo không gian đối với các mẫu tham chiếu trong các khối lân cận trong cùng ảnh hoặc việc dự đoán theo thời gian đối với các mẫu tham chiếu trong các ảnh tham chiếu khác. Các ảnh có thể được đề cập đến là các khung và/hoặc các hình ảnh, và các ảnh tham chiếu có thể được đề cập đến là các khung tham chiếu và/hoặc các hình ảnh tham chiếu. Các kết quả dự đoán theo không gian hoặc theo thời gian trong khối dự đoán biểu diễn khối hình ảnh. Dữ liệu phần dư biểu diễn các độ chênh lệch điểm ảnh giữa khối hình ảnh ban đầu và khối dự đoán. Theo đó, khối được lập mã liên kết được mã hóa theo vectơ chuyển động

mà trở tới khối của các mẫu tham chiếu tạo nên khối dự đoán và dữ liệu phần dư chỉ báo độ chênh lệch giữa khối được lập mã và khối dự đoán. Khối được lập mã bên trong được mã hóa theo chế độ lập mã bên trong và dữ liệu phần dư. Đối với việc nén thêm, dữ liệu phần dư có thể được biến đổi từ miền điểm ảnh tới miền biến đổi. Những điều này dẫn đến các hệ số biến đổi phần dư, mà có thể được lượng tử hóa. Các hệ số biến đổi được lượng tử hóa ban đầu có thể được bố trí trong mảng hai chiều. Các hệ số biến đổi được lượng tử hóa có thể được quét để tạo ra vector một chiều của các hệ số biến đổi. Việc lập mã entropi có thể được áp dụng để đạt được thậm chí việc nén nhiều hơn nữa. Các kỹ thuật nén video như vậy được thảo luận chi tiết hơn dưới đây.

Để đảm bảo video được mã hóa có thể được giải mã một cách chính xác, video được mã hóa và được giải mã theo các chuẩn lập mã video tương ứng. Các chuẩn lập mã video bao gồm lĩnh vực chuẩn hóa (Standardization Sector) thuộc tổ chức viễn thông quốc tế (International Telecommunication Union, viết tắt là ITU) (International Telecommunication Union - Standardization Sector, viết tắt là ITU-T) H.261, nhóm chuyên gia ảnh động (Motion Picture Experts Group, viết tắt là MPEG)-1 thuộc tổ chức chuẩn hóa quốc tế/ủy ban kỹ thuật điện quốc tế (International Organization for Standardization/International Electrotechnical Commission, viết tắt là ISO/IEC) phần 2, ITU-T H.262 hoặc ISO/IEC MPEG-2 phần 2, ITU-T H.263, ISO/IEC MPEG-4 phần 2, lập mã video nâng cao (Advanced Video Coding, viết tắt là AVC), cũng được biết đến là ITU-T H.264 hoặc ISO/IEC MPEG-4 phần 10, và lập mã video hiệu quả cao (High Efficiency Video Coding, viết tắt là HEVC), cũng được biết đến là ITU-T H.265 hoặc MPEG-H phần 2. AVC bao gồm các phần mở rộng chẳng hạn như lập mã video có thể mở rộng (Scalable Video Coding, viết tắt là SVC), lập mã video đa điểm (Multiview Video Coding, viết tắt là MVC) và lập mã video đa điểm cộng độ sâu (Multiview Video Coding plus Depth, viết tắt là MVC+D), và AVC ba chiều (three dimensional, viết tắt là 3D) (three dimensional (3D) AVC, viết tắt là 3D-AVC). HEVC bao gồm các phần mở rộng chẳng hạn như HEVC có thể mở rộng (Scalable HEVC, viết tắt là SHVC), HEVC đa điểm (Multiview HEVC, viết tắt là MV-HEVC), và 3D HEVC (3D-HEVC). Nhóm chuyên gia video liên kết (joint video experts team, viết tắt là JVET) của ITU-T và ISO/IEC đã đang bắt đầu phát triển chuẩn lập mã video được đề cập đến là lập mã video đa năng (VVC). VVC được bao gồm trong bản thảo (Working Draft, viết tắt là WD), mà bao gồm JVET-O2001-v14.

Một số hệ thống lập mã video ứng dụng các tin nhắn thông tin nâng cao bổ sung (SEI). Tin nhắn SEI chứa thông tin mà không được cần bởi quy trình giải mã để xác định các trị số của các mẫu ở các ảnh được giải mã. Ví dụ, các tin nhắn SEI có thể chứa các thông số được sử dụng để kiểm tra dòng bit về sự phù hợp với các chuẩn. Trong một số trường hợp, video được lập mã thành dòng bit nhờ sử dụng nhiều lớp. Các tin nhắn SEI lồng ghép có thể mở rộng có thể được ứng dụng để chứa các tin nhắn SEI cho nhiều lớp. Ví dụ, tin nhắn SEI lồng ghép có thể mở rộng chứa các tin nhắn SEI được lồng ghép có thể mở rộng.

Vấn đề thứ nhất có thể xảy ra khi danh mục của các ký hiệu nhận dạng (identifier, viết tắt là Id) lớp được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng để chỉ báo sự tương ứng giữa các tin nhắn SEI được lồng ghép có thể mở rộng và các lớp. Tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận lớp trừu tượng hóa mạng (NAL) SEI, mà có thể được biểu thị như bộ phận SEI NAL hiện thời nhằm rõ ràng cho việc thảo luận. Bộ phận SEI NAL hiện thời được kết hợp với Id lớp. Tin nhắn SEI lồng ghép có thể mở rộng luôn đề cập đến lớp được kết hợp với bộ phận SEI NAL hiện thời. Điều này dẫn đến báo hiệu phần dư khi Id lớp của bộ phận SEI NAL hiện thời được bao gồm trong bộ phận SEI NAL hiện thời và cũng được bao gồm trong tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận NAL hiện thời.

Theo ví dụ thứ nhất, được bộc lộ ở đây là các sự cải tiến với tin nhắn SEI lồng ghép có thể mở rộng dùng cho việc sử dụng cùng với các lớp được quy định. Cụ thể là, Id lớp có thể được đề cập đến là Id lớp đoạn đầu bộ phận NAL (nuh_layer_id). Nuh_layer_id của bộ phận SEI NAL hiện thời được loại trừ khỏi tin nhắn SEI lồng ghép có thể mở rộng. Điều này có thể được thực hiện bằng cách ứng dụng thành phần cú pháp Id lớp lồng ghép có thể mở rộng (layer_id[i]) mà xác định nuh_layer_id cho mỗi lớp trong đó tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. Layer_id[i] lồng ghép có thể mở rộng bị ràng buộc để bao gồm chỉ nuh_layer_ids mà lớn hơn nuh_layer_id của bộ phận SEI NAL hiện thời. Do đó, nuh_layer_id của bộ phận SEI NAL hiện thời được bỏ qua khỏi tin nhắn SEI lồng ghép có thể mở rộng và được suy luận từ bộ phận SEI NAL hiện thời. Điều này nâng cao tin nhắn SEI lồng ghép có thể mở rộng bằng cách bỏ qua Id phần dư. Ví dụ, vòng lặp mã hóa/giải mã layer_id[i] lồng ghép có thể mở rộng thực thi ít hơn một lần, mà làm giảm các tài nguyên bộ xử lý trong suốt thời gian mã hóa và/hoặc giải mã. Hơn nữa, điều này làm giảm kích thước của dòng bit được mã hóa cho mỗi trong số các tin nhắn SEI lồng ghép có thể mở rộng.

ở dòng bit. Kết quả là, hiệu quả lập mã được tăng lên, mà làm giảm việc sử dụng tài nguyên của bộ xử lý, bộ nhớ, và/hoặc báo hiệu mạng ở cả bộ mã hóa và bộ giải mã.

Vấn đề thứ hai xảy ra khi cờ được sử dụng để chỉ báo rằng các tin nhắn SEI được lồng ghép có thể mở rộng trong tin nhắn SEI lồng ghép có thể mở rộng áp dụng cho tất cả các lớp ở bộ phận truy cập (AU). AU là tập hợp của các ảnh được lập mã mà được bao gồm ở các lớp khác nhau và được kết hợp với cùng thời gian đầu ra. Cách tiếp cận này làm việc khi tất cả các AU chứa các ảnh cho tất cả các lớp. Tuy nhiên, trong một số trường hợp, các ảnh có thể có mặt trong thời gian đầu ra ở số lượng lớp và được bỏ qua ở các lớp khác. Ví dụ, lớp tốc độ khung cao có thể bao gồm các ảnh trong mỗi AU, trong khi lớp tốc độ khung thấp bỏ qua các ảnh trong nhiều trong số các AU mà được chiếm giữ bởi lớp tốc độ khung cao hơn. Kết quả là cờ khiến các tin nhắn SEI chỉ áp dụng cho các lớp mà bao gồm các ảnh trong AU hiện thời. Điều này gây ra các lỗi thử nghiệm sự phù hợp bởi vì bộ giải mã tham chiếu giả định (HRD) có thể không thể kết hợp các tin nhắn SEI được lồng ghép có thể mở rộng với các lớp mà bỏ qua các ảnh trong AU hiện thời. Ví dụ, các lớp như vậy có thể chứa các ảnh trong các AU sau đó sau AU hiện thời. Trong trường hợp như vậy, HRD không nhận ra các tin nhắn SEI được lồng ghép có thể mở rộng khi áp dụng cho các lớp như vậy khi các AU sau đó được thử nghiệm về sự phù hợp. Theo đó, HRD có thể không thể thử nghiệm một cách chính xác các lớp như vậy về sự phù hợp ở các AU sau đó khi cờ cho tất cả các lớp được ứng dụng ở AU không hoàn chỉnh.

Theo ví dụ thứ hai, được bộc lộ ở đây là cờ tất cả các lớp lồng ghép có thể mở rộng (`all_layers_flag`) để điều chỉnh các lỗi nêu trên. `all_layers_flag` lồng ghép có thể mở rộng có thể được thiết đặt để chỉ báo rằng các tin nhắn SEI được lồng ghép có thể mở rộng trong tin nhắn SEI lồng ghép có thể mở rộng áp dụng cho tất cả các lớp được quy định trong tập hợp thông số video (video parameter set, viết tắt là VPS). VPS liệt kê tất cả các lớp. Do đó, cách tiếp cận này cho phép bộ mã hóa chỉ báo một cách rõ ràng khi nào các tin nhắn SEI được lồng ghép có thể mở rộng được áp dụng toàn bộ cho tất cả các lớp. Quy trình thử nghiệm sự phù hợp HRD sau đó có thể kiểm tra dòng bit về sự phù hợp mà không gửi trả lại các lỗi không thể dự đoán ở các AU mà theo sau các AU không hoàn chỉnh mà được kết hợp với tin nhắn SEI lồng ghép có thể mở rộng. Do đó, `all_layers_flag` lồng ghép có thể mở rộng làm tăng tính năng của bộ mã hóa và/hoặc bộ giải mã bằng cách khiến dòng bit được kiểm tra một cách chính xác về sự phù hợp khi các AU không hoàn chỉnh có mặt. Hơn nữa, `all_layers_flag` lồng ghép có thể mở rộng làm giảm kích thước của dòng bit được mã hóa

bằng cách bỏ qua báo hiệu lớp rõ ràng cho các tín hiệu SEI được lồng ghép có thể mở rộng khi các tín hiệu như vậy áp dụng cho tất cả các lớp. Kết quả là, hiệu quả lập mã được tăng lên, mà làm giảm việc sử dụng tài nguyên của bộ xử lý, bộ nhớ, và/hoặc báo hiệu mạng ở cả bộ mã hóa và bộ giải mã.

Fig.1 là lưu đồ của phương pháp thao tác 100 ví dụ về việc lập mã tín hiệu video. Cụ thể là, tín hiệu video được mã hóa ở bộ mã hóa. Quy trình mã hóa nén tín hiệu video bằng cách ứng dụng các cơ chế khác nhau để làm giảm kích thước tệp video. Kích thước tệp nhỏ hơn cho phép tệp video được nén được truyền về phía người dùng, trong khi làm giảm phí tồn độ rộng dải được kết hợp. Bộ giải mã sau đó giải mã tệp video được nén để tái cấu trúc tín hiệu video ban đầu cho việc hiển thị tới người dùng cuối. Quy trình giải mã nói chung phản chiếu quy trình mã hóa để cho phép bộ giải mã được tái cấu trúc một cách nhất quán tín hiệu video.

Ở bước 101, tín hiệu video được đưa vào bộ mã hóa. Ví dụ, tín hiệu video có thể là tệp video không được nén được lưu trữ trong bộ nhớ. Với ví dụ khác, tệp video có thể được chụp bởi thiết bị chụp video, chẳng hạn như camera video, và được mã hóa để hỗ trợ việc tạo dòng trực tiếp của video. Tệp video có thể bao gồm cả thành phần audio và thành phần video. Thành phần video chứa loạt của các khung hình ảnh mà khi được xem theo chuỗi, đưa ra ấn tượng trực quan về chuyển động. Các khung chứa các điểm ảnh mà được biểu diễn xét về độ sáng, được đề cập đến ở đây là các thành phần độ chói (hoặc các mẫu độ chói), và màu sắc, mà được đề cập đến là các thành phần sắc độ (hoặc các mẫu màu sắc). Theo một số ví dụ, các khung cũng có thể chứa các trị số độ sâu để hỗ trợ việc xem ba chiều.

Ở bước 103, video được phân chia thành các khối. Việc phân chia bao gồm việc chia nhỏ các điểm ảnh trong mỗi khung thành các khối hình vuông và/hoặc hình chữ nhật cho việc nén. Ví dụ, trong lập mã video hiệu quả cao (HEVC) (cũng được biết đến là H.265 và MPEG-H phần 2) khung trước tiên có thể được chia thành các bộ phận cây lập mã (CTU), mà là các khối có kích thước được định rõ trước (ví dụ, sáu mươi tư điểm ảnh nhân sáu mươi tư điểm ảnh). Các CTU chứa cả các mẫu độ chói và sắc độ. Các cây lập mã có thể được ứng dụng để chia các CTU thành các khối và sau đó chia nhỏ đệ quy các khối cho đến khi các cấu hình đạt được mà hỗ trợ việc mã hóa thêm. Ví dụ, các thành phần độ chói của khung có thể được chia nhỏ cho đến khi các khối riêng lẻ chứa các trị số phát sáng tương đối đồng nhất. Hơn nữa, các thành phần sắc độ của khung có thể được chia nhỏ cho đến khi các khối

riêng lẻ chứa các trị số màu sắc tương đối đồng nhất. Theo đó, các cơ chế phân chia thay đổi tùy thuộc vào nội dung của các khung video.

Ở bước 105, các cơ chế nén khác nhau được ứng dụng để nén các khối ảnh được phân chia ở bước 103. Ví dụ, việc dự đoán liên kết và/hoặc việc dự đoán bên trong có thể được ứng dụng. Việc dự đoán liên kết được thiết kế để nắm lợi thế của thực tế là các đối tượng trong cảnh chung có xu hướng xuất hiện trong các khung liên tiếp. Theo đó, khối mô tả đối tượng trong các khung tham chiếu không cần được mô tả lặp lại trong các khung liền kề. Cụ thể là, đối tượng, chẳng hạn như bảng, có thể vẫn ở vị trí không đổi qua nhiều khung. Do đó bảng được mô tả một lần và các khung liền kề có thể tham chiếu ngược lại khung tham chiếu. Các cơ chế so khớp mẫu hình có thể được ứng dụng để so khớp các đối tượng qua nhiều khung. Hơn nữa, các đối tượng di chuyển có thể được biểu diễn qua nhiều khung, ví dụ do sự di chuyển đối tượng hoặc sự di chuyển camera. Với ví dụ cụ thể, video có thể thể hiện ô tô mà di chuyển qua màn hình qua nhiều khung. Các vector chuyển động có thể được ứng dụng để mô tả sự di chuyển như vậy. Vector chuyển động là vector hai chiều mà đưa ra dịch vị từ các tọa độ của đối tượng trong khung tới các tọa độ của đối tượng trong khung tham chiếu. Như vậy, việc dự đoán liên kết có thể mã hóa khối ảnh trong khung hiện thời là tập hợp của các vector chuyển động chỉ báo dịch vị từ khối tương ứng trong khung tham chiếu.

Việc dự đoán bên trong mã hóa các khối trong khung chung. Việc dự đoán bên trong lấy lợi thế của thực tế là các thành phần sắc độ và độ chói có xu hướng tạo chùm trong khung. Ví dụ, mảng màu xanh lá cây trong phần của cây có xu hướng được định vị sát với các mảng màu xanh lá cây tương tự. Việc dự đoán bên trong ứng dụng các chế độ dự đoán đa hướng (ví dụ, ba mươi ba trong HEVC), chế độ mặt phẳng, và chế độ dòng một chiều (direct current, viết tắt là DC). Các chế độ hướng chỉ báo rằng khối hiện thời tương tự/giống các mẫu của khối lân cận theo hướng tương ứng. Chế độ mặt phẳng chỉ báo rằng loạt của các khối dọc hàng/cột (ví dụ, mặt phẳng) có thể được nội suy dựa vào các khối lân cận ở các mép của hàng. Chế độ mặt phẳng, thực tế, chỉ báo sự chuyển tiếp mượt mà của ánh sáng/màu sắc qua hàng/cột bằng cách ứng dụng độ dốc gần như không đổi về việc thay đổi các trị số. Chế độ DC được ứng dụng cho việc làm mịn ranh giới và chỉ báo rằng khối tương tự/giống như trị số trung bình được kết hợp với các mẫu của tất cả các khối lân cận được kết hợp với các hướng góc của các chế độ dự đoán hướng. Theo đó, các khối dự đoán bên trong có thể biểu diễn các khối ảnh như các trị số chế độ dự đoán liên quan khác nhau thay vì các trị số thực tế. Hơn nữa, các khối dự đoán liên kết có thể biểu diễn các khối ảnh như các trị số vector

chuyển động thay vì các trị số thực tế. Trong cả hai trường hợp, các khối dự đoán có thể không biểu diễn một cách chính xác các khối ảnh trong một số trường hợp. Các sự chênh lệch bất kỳ được lưu trữ trong các khối phần dư. Các sự biến đổi có thể được áp dụng cho các khối phần dư để nén thêm tệp.

Ở bước 107, các kỹ thuật lọc khác nhau có thể được áp dụng. Trong HEVC, các bộ lọc được áp dụng theo sơ đồ lọc trong vòng lặp. Việc dự đoán trên cơ sở khối được thảo luận ở trên có thể dẫn đến việc tạo ra của các ảnh khối ở bộ giải mã. Hơn nữa, sơ đồ dự đoán trên cơ sở khối có thể mã hóa khối và sau đó tái cấu trúc khối được mã hóa cho việc sử dụng sau đó như khối tham chiếu. Sơ đồ lọc trong vòng lặp áp dụng lặp lại các bộ lọc triệt nhiễu, các bộ lọc tách khối, các bộ lọc vòng lặp thích ứng, và các bộ lọc dịch vị thích ứng mẫu (sample adaptive offset, viết tắt là SAO) cho các khối/các khung. Các bộ lọc này làm giảm các tạo tác tạo khối như vậy sao cho tệp được mã hóa có thể được tái cấu trúc một cách chính xác. Hơn nữa, các bộ lọc này làm giảm các tạo tác trong các khối tham chiếu được tái cấu trúc sao cho các tạo tác ít có khả năng tạo ra các tạo tác bổ sung trong các khối tiếp theo mà được mã hóa dựa vào các khối tham chiếu được tái cấu trúc.

Khi tín hiệu video đã được phân chia, được nén, và được lọc, dữ liệu nhận được được mã hóa ở dòng bit ở bước 109. Dòng bit bao gồm dữ liệu được thảo luận ở trên cũng như dữ liệu báo hiệu bất kỳ được mong muốn để hỗ trợ việc tái cấu trúc tín hiệu video thích hợp ở bộ giải mã. Ví dụ, dữ liệu như vậy có thể bao gồm dữ liệu phân chia, dữ liệu dự đoán, các khối phần dư, và các cờ khác nhau cung cấp các lệnh lập mã tới bộ giải mã. Dòng bit có thể được lưu trữ trong bộ nhớ cho việc truyền về phía bộ giải mã khi yêu cầu. Dòng bit cũng có thể phát rộng và/hoặc phát đa hướng về phía nhiều bộ giải mã. Việc tạo ra của dòng bit là quy trình lặp đi lặp lại. Theo đó, các bước 101, 103, 105, 107, và 109 có thể xảy ra liên tục và/hoặc đồng thời qua nhiều khung và nhiều khối. Thứ tự được thể hiện trên Fig.1 được biểu diễn nhằm làm rõ và dễ dàng cho việc thảo luận, và không nhằm giới hạn quy trình lập mã video ở thứ tự cụ thể.

Bộ giải mã thu dòng bit và bắt đầu quy trình giải mã ở bước 111. Cụ thể là, bộ giải mã ứng dụng sơ đồ giải mã entropi để chuyển đổi dòng bit thành cú pháp và dữ liệu video tương ứng. Bộ giải mã ứng dụng dữ liệu cú pháp từ dòng bit để xác định các phần chia cho các khung ở bước 111. Việc phân chia sẽ so khớp các kết quả của việc phân chia khối ở bước 103. Việc mã hóa/giải mã entropi như được ứng dụng ở bước 111 bây giờ được mô tả. Bộ mã hóa đưa ra nhiều lựa chọn trong suốt thời gian quy trình nén, chẳng hạn như lựa chọn các

sơ đồ phân chia khối từ nhiều lựa chọn khả thi dựa vào việc định vị theo không gian của các trị số trong (các) ảnh đầu vào. Việc báo hiệu các lựa chọn chính xác có thể ứng dụng số lượng lớn các ngăn. Như được sử dụng ở đây, ngăn là trị số nhị phân mà được xử lý như biến (ví dụ, trị số bit mà có thể thay đổi tùy thuộc vào ngữ cảnh). Việc lập mã entropi cho phép bộ mã hóa loại bỏ bất kỳ các tùy chọn nào mà rõ ràng là không khả thi đối với trường hợp cụ thể, để lại tập hợp của các tùy chọn có thể được phép. Mỗi tùy chọn có thể được phép sau đó được gán từ mã. Độ dài của các từ mã dựa vào số các tùy chọn có thể được phép (ví dụ, một ngăn cho hai tùy chọn, hai ngăn cho ba đến bốn tùy chọn chẳng hạn) bộ mã hóa sau đó mã hóa từ mã cho tùy chọn được lựa chọn. Sơ đồ này làm giảm kích thước của các từ mã vì các từ mã lớn đến mức được mong muốn để chỉ báo duy nhất sự lựa chọn từ tập hợp con nhỏ của các tùy chọn có thể được phép như trái ngược với việc chỉ báo duy nhất sự lựa chọn từ tập hợp lớn có khả năng của tất cả các tùy chọn có thể. Bộ giải mã sau đó giải mã sự lựa chọn bằng cách xác định tập hợp của các tùy chọn có thể được phép theo cách thức tương tự với bộ mã hóa. Bằng cách xác định tập hợp của các tùy chọn có thể được phép, bộ giải mã có thể đọc từ mã và xác định sự lựa chọn được thực hiện bởi bộ mã hóa.

Ở bước 113, bộ giải mã thực hiện việc giải mã khối. Cụ thể là, bộ giải mã ứng dụng các sự biến đổi ngược để tạo ra các khối phần dư. Sau đó bộ giải mã ứng dụng các khối phần dư và các khối dự đoán tương ứng để tái cấu trúc các khối hình ảnh theo việc phân chia. Các khối dự đoán có thể bao gồm cả các khối dự đoán bên trong và các khối dự đoán liên kết khi được tạo ra ở bộ mã hóa ở bước 105. Các khối hình ảnh được tái cấu trúc sau đó được định vị vào các khung của tín hiệu video được tái cấu trúc theo dữ liệu phân chia được xác định ở bước 111. Cú pháp cho bước 113 cũng có thể được báo hiệu ở dòng bit qua việc lập mã entropi như được thảo luận ở trên.

Ở bước 115, việc lọc được thực hiện trên các khung của tín hiệu video được tái cấu trúc theo cách thức tương tự với bước 107 ở bộ mã hóa. Ví dụ, các bộ lọc triệt nhiễu, các bộ lọc tách khối, các bộ lọc vòng lặp thích ứng, và các bộ lọc SAO có thể được áp dụng cho các khung để loại bỏ các tạo tác tạo khối. Khi các khung được lọc, tín hiệu video có thể được đưa ra tới màn hình ở bước 117 để xem bởi người dùng cuối.

Fig.2 là hình vẽ giản lược của hệ thống lập mã và giải mã (mã hóa và giải mã) 200 ví dụ về việc lập mã video. Cụ thể là, hệ thống mã hóa-giải mã 200 cung cấp tính năng để hỗ trợ việc thực hiện của phương pháp vận hành 100. Hệ thống mã hóa-giải mã 200 được khái quát để mô tả các thành phần được ứng dụng trong cả bộ mã hóa và bộ giải mã. Hệ thống

mã hóa-giải mã 200 thu và phân chia tín hiệu video như được thảo luận đối với các bước 101 và 103 theo phương pháp vận hành 100, mà dẫn đến tín hiệu video được phân chia 201. Hệ thống mã hóa-giải mã 200 sau đó nén tín hiệu video được phân chia 201 thành dòng bit được lập mã khi hoạt động như bộ mã hóa như được thảo luận đối với các bước 105, 107, và 109 theo phương pháp 100. Khi hoạt động như bộ giải mã, hệ thống mã hóa-giải mã 200 tạo ra tín hiệu video đầu ra từ dòng bit như được thảo luận đối với các bước 111, 113, 115, và 117 theo phương pháp vận hành 100. Hệ thống mã hóa-giải mã 200 bao gồm thành phần điều khiển bộ lập mã chung 211, thành phần định tỷ lệ và lượng tử hóa biến đổi 213, thành phần đánh giá ảnh bên trong 215, thành phần dự đoán ảnh bên trong 217, thành phần bù chuyển động 219, thành phần đánh giá chuyển động 221, thành phần định tỷ lệ và biến đổi ngược 229, thành phần phân tích điều khiển bộ lọc 227, thành phần các bộ lọc trong vòng lặp 225, thành phần bộ đệm ảnh được giải mã 223, và thành phần lập mã số học nhị phân thích ứng ngữ cảnh (context adaptive binary arithmetic coding, viết tắt là CABAC) và định dạng đoạn đầu 231. Các thành phần như vậy được ghép nối như được thể hiện. Trên Fig.2, các đường nét liền chỉ báo sự di chuyển của dữ liệu được mã hóa/được giải mã trong khi các đường nét đứt chỉ báo sự di chuyển của dữ liệu điều khiển mà điều khiển hoạt động của các thành phần khác. Các thành phần của hệ thống mã hóa-giải mã 200 đều có thể có mặt ở bộ mã hóa. Bộ giải mã có thể bao gồm tập hợp con của các thành phần của hệ thống mã hóa-giải mã 200. Ví dụ, bộ giải mã có thể bao gồm thành phần dự đoán ảnh bên trong 217, thành phần bù chuyển động 219, thành phần định tỷ lệ và biến đổi ngược 229, thành phần các bộ lọc trong vòng lặp 225, và thành phần bộ đệm ảnh được giải mã 223. Các thành phần này bây giờ được mô tả.

Tín hiệu video được phân chia 201 là chuỗi video được chụp mà đã được phân chia thành các khối của các điểm ảnh bởi cây lập mã. Cây lập mã ứng dụng các chế độ phân tách khác nhau để chia nhỏ khối của các điểm ảnh thành các khối nhỏ hơn của các điểm ảnh. Các khối này sau đó có thể được chia nhỏ thêm thành các khối nhỏ hơn. Các khối có thể được đề cập đến là các nút trên cây lập mã. Các nút cha lớn hơn được phân tách thành các nút con nhỏ hơn. Số các lần nút được chia nhỏ được đề cập đến là độ sâu của nút/cây lập mã. Các khối được chia có thể được bao gồm trong các bộ phận lập mã (coding unit, viết tắt là CU) trong một số trường hợp. Ví dụ, CU có thể là phần con của CTU mà chứa khối độ chói, (các) khối sắc độ khác biệt màu đỏ (red difference chroma, viết tắt là Cr), và (các) khối sắc độ khác biệt màu xanh da trời (blue difference chroma, viết tắt là Cb) cùng với các lệnh cú pháp

tương ứng cho CU. Các chế độ phân tách có thể bao gồm cây nhị phân (binary tree, viết tắt là BT), cây tam phân (triple tree, viết tắt là TT), và cây tứ phân (quad tree, viết tắt là QT) được ứng dụng để phân chia nút lần lượt thành hai, ba, hoặc bốn nút con, có các hình dạng thay đổi tùy thuộc vào các chế độ phân tách được ứng dụng. Tín hiệu video được phân chia 201 được chuyển tiếp tới thành phần điều khiển bộ lập mã chung 211, thành phần định tỷ lệ và lượng tử hóa biến đổi 213, thành phần đánh giá ảnh bên trong 215, thành phần phân tích điều khiển bộ lọc 227, và thành phần đánh giá chuyển động 221 cho việc nén.

Thành phần điều khiển bộ lập mã chung 211 được tạo cấu hình để đưa ra các quyết định liên quan đến việc lập mã của các ảnh của chuỗi video thành dòng bit theo các sự ràng buộc ứng dụng. Ví dụ, thành phần điều khiển bộ lập mã chung 211 quản lý việc tối ưu hóa của kích thước dòng bit/tốc độ bit đối với chất lượng tái cấu trúc. Các quyết định như vậy có thể được thực hiện dựa vào tính khả dụng độ rộng dải/khoảng trống lưu trữ và các yêu cầu độ phân giải hình ảnh. Thành phần điều khiển bộ lập mã chung 211 cũng quản lý việc ứng dụng bộ đệm xét về tốc độ truyền để làm giảm các vấn đề thấp tải và quá tải bộ đệm. Để quản lý các vấn đề này, thành phần điều khiển bộ lập mã chung 211 quản lý việc phân chia, dự đoán, và việc lọc bởi các thành phần khác. Ví dụ, thành phần điều khiển bộ lập mã chung 211 có thể tăng một cách cơ động độ phức tạp nén để tăng độ phân giải và tăng việc sử dụng độ rộng dải hoặc giảm độ phức tạp nén để giảm độ phân giải và việc sử dụng độ rộng dải. Do đó, thành phần điều khiển bộ lập mã chung 211 điều khiển các thành phần khác của hệ thống mã hóa-giải mã 200 cân bằng chất lượng tái cấu trúc tín hiệu video với tốc độ bit liên quan. Thành phần điều khiển bộ lập mã chung 211 tạo ra dữ liệu điều khiển, mà điều khiển hoạt động của các thành phần còn lại. Dữ liệu điều khiển cũng được chuyển tiếp tới thành phần CABAC và định dạng đoạn đầu 231 được mã hóa ở dòng bit để báo hiệu các thông số để giải mã ở bộ giải mã.

Tín hiệu video được phân chia 201 cũng được gửi tới thành phần đánh giá chuyển động 221 và thành phần bù chuyển động 219 cho việc dự đoán liên kết. Khung hoặc lát của tín hiệu video được phân chia 201 có thể được chia thành nhiều khối video. Thành phần đánh giá chuyển động 221 và thành phần bù chuyển động 219 thực hiện việc lập mã dự đoán liên kết của khối video được thu liên quan đến một hoặc nhiều khối trong một hoặc nhiều khung tham chiếu để đưa ra việc dự đoán theo thời gian. Hệ thống mã hóa-giải mã 200 có thể thực hiện nhiều sự chuyển tác lập mã, ví dụ, để lựa chọn chế độ lập mã thích hợp cho mỗi khối của dữ liệu video.

Thành phần đánh giá chuyển động 221 và thành phần bù chuyển động 219 có thể được tích hợp cao, nhưng được minh họa một cách riêng biệt cho các mục đích khái niệm. Việc đánh giá chuyển động, được thực hiện bởi thành phần đánh giá chuyển động 221, là quy trình tạo ra các vector chuyển động, mà đánh giá chuyển động cho các khối video. Vector chuyển động, ví dụ, có thể chỉ báo sự dịch chuyển của đối tượng được lập mã liên quan đến khối dự đoán. Khối dự đoán là khối mà được tìm thấy để so khớp kỹ khối được lập mã, xét về độ chênh lệch điểm ảnh. Khối dự đoán cũng có thể được đề cập đến là khối tham chiếu. Độ chênh lệch điểm ảnh như vậy có thể được xác định bởi tổng của độ chênh lệch tuyệt đối (sum of absolute difference, viết tắt là SAD), tổng của độ chênh lệch bình phương (sum of square difference, viết tắt là SSD), hoặc các số liệu chênh lệch khác. HEVC ứng dụng nhiều đối tượng được lập mã bao gồm CTU, các khối cây lập mã (CTB), và các CU. Ví dụ, CTU có thể được chia thành các CTB, mà sau đó có thể được chia thành các CB cho việc đưa vào các CU. CU có thể được mã hóa như bộ phận dự đoán (prediction unit, viết tắt là PU) chứa dữ liệu dự đoán và/hoặc bộ phận biến đổi (transform unit, viết tắt là TU) chứa dữ liệu phần dư được biến đổi cho CU. Thành phần đánh giá chuyển động 221 tạo ra các vector chuyển động, các PU, và các TU nhờ sử dụng phân tích méo tỷ lệ như phần của quy trình tối ưu hóa méo tỷ lệ. Ví dụ, thành phần đánh giá chuyển động 221 có thể xác định nhiều khối tham chiếu, nhiều vector chuyển động, v.v. cho khung/khối hiện thời, và có thể lựa chọn các khối tham chiếu, các vector chuyển động, v.v. có các đặc tính méo tỷ lệ tốt nhất. Các đặc tính méo tỷ lệ tốt nhất cân bằng cả chất lượng của việc tái cấu trúc video (ví dụ, lượng tổn thất dữ liệu bởi việc nén) với hiệu quả lập mã (ví dụ, kích thước của việc mã hóa cuối cùng).

Theo một số ví dụ, hệ thống mã hóa-giải mã 200 có thể tính các trị số cho các vị trí điểm ảnh số nguyên phụ của các ảnh tham chiếu được lưu trữ trong thành phần bộ đệm ảnh được giải mã 223. Ví dụ, hệ thống mã hóa-giải mã video 200 có thể nội suy các trị số của các vị trí điểm ảnh một phần tư, các vị trí điểm ảnh một phần tám, hoặc các vị trí điểm ảnh phân số khác của ảnh tham chiếu. Do đó, thành phần đánh giá chuyển động 221 có thể thực hiện việc tìm kiếm chuyển động liên quan đến các vị trí điểm ảnh đầy đủ và các vị trí điểm ảnh phân số và đưa ra vector chuyển động với độ chính xác điểm ảnh phân số. Thành phần đánh giá chuyển động 221 tính vector chuyển động cho PU của khối video trong lát được lập mã liên kết bằng cách so sánh vị trí của PU với vị trí của khối dự đoán của ảnh tham chiếu. Thành phần đánh giá chuyển động 221 đưa ra vector chuyển động được tính như dữ liệu

chuyển động tới thành phần CABAC và định dạng đoạn đầu 231 để mã hóa và chuyển động tới thành phần bù chuyển động 219.

Việc bù chuyển động, được thực hiện bởi thành phần bù chuyển động 219, có thể bao hàm việc tìm nạp hoặc tạo ra khối dự đoán dựa vào vector chuyển động được xác định bởi thành phần đánh giá chuyển động 221. Hơn nữa, thành phần đánh giá chuyển động 221 và thành phần bù chuyển động 219 có thể được tích hợp ở dạng chức năng, theo một số ví dụ. Ngay khi thu vector chuyển động cho PU của khối video hiện tại, thành phần bù chuyển động 219 có thể định vị khối dự đoán trong đó các điểm vector chuyển động trở tới. Khối video phần dư sau đó được tạo nên bằng cách trừ các trị số điểm ảnh của khối dự đoán từ các trị số điểm ảnh của khối video hiện thời được lập mã, tạo nên các trị số độ chênh lệch điểm ảnh. Nói chung, thành phần đánh giá chuyển động 221 thực hiện việc đánh giá chuyển động liên quan đến các thành phần độ chói, và thành phần bù chuyển động 219 sử dụng các vector chuyển động được tính dựa vào các thành phần độ chói cho cả các thành phần sắc độ và các thành phần độ chói. Khối dự đoán và khối phần dư được chuyển tiếp tới thành phần định tỷ lệ và lượng tử hóa biến đổi 213.

Tín hiệu video được phân chia 201 cũng được gửi tới thành phần đánh giá ảnh bên trong 215 và thành phần dự đoán ảnh bên trong 217. Như với thành phần đánh giá chuyển động 221 và thành phần bù chuyển động 219, thành phần đánh giá ảnh bên trong 215 và thành phần dự đoán ảnh bên trong 217 có thể được tích hợp cao, nhưng được minh họa một cách riêng biệt cho các mục đích khái niệm. Thành phần đánh giá ảnh bên trong 215 và thành phần dự đoán ảnh bên trong 217 dự đoán bên trong khối hiện thời liên quan đến các khối trong khung hiện thời, như sự thay thế cho việc dự đoán liên kết được thực hiện bởi thành phần đánh giá chuyển động 221 và thành phần bù chuyển động 219 giữa các khung, như được nêu trên. Cụ thể là, thành phần đánh giá ảnh bên trong 215 xác định chế độ dự đoán bên trong để sử dụng để mã hóa khối hiện thời. Theo một số ví dụ, thành phần đánh giá ảnh bên trong 215 lựa chọn chế độ dự đoán bên trong thích hợp để mã hóa khối hiện thời từ nhiều chế độ dự đoán bên trong được thử nghiệm. Các chế độ dự đoán bên trong được lựa chọn sau đó được chuyển tiếp tới thành phần CABAC và định dạng đoạn đầu 231 để mã hóa.

Ví dụ, thành phần đánh giá ảnh bên trong 215 tính các trị số méo tỷ lệ nhờ sử dụng phân tích méo tỷ lệ cho các chế độ dự đoán bên trong được thử nghiệm khác nhau, và lựa chọn chế độ dự đoán bên trong có các đặc tính méo tỷ lệ tốt nhất trong số các chế độ được thử nghiệm. Phân tích méo tỷ lệ nói chung xác định lượng méo (hoặc lỗi) giữa khối được mã

hóa và khối chưa được mã hóa ban đầu mà đã được mã hóa để tạo ra khối được mã hóa, cũng như tốc độ bit (ví dụ, số các bit) được sử dụng để tạo ra khối được mã hóa. Thành phần đánh giá ảnh bên trong 215 tính các hệ số từ các sự méo và các tỷ lệ cho các khối được mã hóa khác nhau để xác định chế độ dự đoán bên trong nào thể hiện trị số méo tỷ lệ tốt nhất cho khối. Ngoài ra, thành phần đánh giá ảnh bên trong 215 có thể được tạo cấu hình để lập mã các khối độ sâu của bản đồ độ sâu nhờ sử dụng chế độ mô hình hóa độ sâu (depth modeling mode, viết tắt là DMM) dựa vào việc tối ưu hóa méo tỷ lệ (rate-distortion optimization, viết tắt là RDO).

Thành phần dự đoán ảnh bên trong 217 có thể tạo ra khối phần dư từ khối dự đoán dựa vào các chế độ dự đoán bên trong được lựa chọn được xác định bởi thành phần đánh giá ảnh bên trong 215 khi được thực hiện trên bộ mã hóa hoặc đọc khối phần dư từ dòng bit khi được thực hiện trên bộ giải mã. Khối phần dư bao gồm độ chênh lệch ở các trị số giữa khối dự đoán và khối ban đầu, được biểu diễn như ma trận. Khối phần dư sau đó được chuyển tiếp tới thành phần định tỷ lệ và lượng tử hóa biến đổi 213. Thành phần đánh giá ảnh bên trong 215 và thành phần dự đoán ảnh bên trong 217 có thể hoạt động trên cả hai thành phần độ chói và sắc độ.

Thành phần định tỷ lệ và lượng tử hóa biến đổi 213 được tạo cấu hình để nén thêm khối phần dư. Thành phần định tỷ lệ và lượng tử hóa biến đổi 213 áp dụng sự biến đổi, chẳng hạn như biến đổi cosin rời rạc (discrete cosine transform, viết tắt là DCT), biến đổi sin rời rạc (discrete sine transform, viết tắt là DST), hoặc biến đổi tương tự về mặt khái niệm, tới khối phần dư, tạo ra khối video bao gồm các trị số hệ số biến đổi phần dư. Các sự biến đổi wavelet, các sự biến đổi số nguyên, các sự biến đổi dải con hoặc các loại khác của các sự biến đổi cũng có thể được sử dụng. Sự biến đổi có thể chuyển đổi thông tin phần dư từ miền trị số điểm ảnh tới miền biến đổi, chẳng hạn như miền tần số. Thành phần định tỷ lệ và lượng tử hóa biến đổi 213 cũng được tạo cấu hình để định tỷ lệ thông tin phần dư được biến đổi, ví dụ dựa vào tần số. Việc định tỷ lệ như vậy bao gồm việc áp dụng hệ số định tỷ lệ cho thông tin phần dư sao cho thông tin tần số khác nhau được lượng tử hóa ở các độ hạt khác nhau, mà có thể tác động đến chất lượng hình ảnh cuối cùng của video được tái cấu trúc. Thành phần định tỷ lệ và lượng tử hóa biến đổi 213 cũng được tạo cấu hình để lượng tử hóa các hệ số biến đổi để làm giảm thêm tốc độ bit. Quy trình lượng tử hóa có thể làm giảm độ sâu bit được kết hợp với một số hoặc tất cả các hệ số. Mức độ lượng tử hóa có thể được điều chỉnh bằng cách điều chỉnh thông số lượng tử hóa. Theo một số ví dụ, thành phần định tỷ lệ và

lượng tử hóa biến đổi 213 sau đó có thể thực hiện việc quét của ma trận bao gồm các hệ số biến đổi được lượng tử hóa. Các hệ số biến đổi được lượng tử hóa được chuyển tiếp tới thành phần CABAC và định dạng đoạn đầu 231 được mã hóa ở dòng bit.

Thành phần định tỷ lệ và biến đổi ngược 229 áp dụng thao tác ngược của thành phần định tỷ lệ và lượng tử hóa biến đổi 213 để hỗ trợ việc đánh giá chuyển động. Thành phần định tỷ lệ và biến đổi ngược 229 áp dụng việc định tỷ lệ, biến đổi, và/hoặc lượng tử hóa ngược để tái cấu trúc khối phần dư trong miền điểm ảnh, ví dụ, cho việc sử dụng sau đó như khối tham chiếu mà có thể trở thành khối dự đoán cho khối hiện thời khác. Thành phần đánh giá chuyển động 221 và/hoặc thành phần bù chuyển động 219 có thể tính khối tham chiếu bằng cách cộng khối phần dư trở lại vào khối dự đoán tương ứng để sử dụng trong việc đánh giá chuyển động của khung/khối sau đó. Các bộ lọc được áp dụng cho các khối tham chiếu được tái cấu trúc để làm giảm các tạo tác được tạo ra trong suốt thời gian định tỷ lệ, lượng tử hóa, và biến đổi. Các tạo tác như vậy theo cách khác có thể khiến việc dự đoán không chính xác (và tạo ra các tạo tác bổ sung) khi các khối tiếp theo được dự đoán.

Thành phần phân tích điều khiển bộ lọc 227 và thành phần các bộ lọc trong vòng lặp 225 áp dụng các bộ lọc cho các khối phần dư và/hoặc cho các khối ảnh được tái cấu trúc. Ví dụ, khối phần dư được biến đổi từ thành phần định tỷ lệ và biến đổi ngược 229 có thể được kết hợp với khối dự đoán tương ứng từ thành phần dự đoán ảnh bên trong 217 và/hoặc thành phần bù chuyển động 219 để tái cấu trúc khối hình ảnh ban đầu. Các bộ lọc sau đó có thể được áp dụng cho khối hình ảnh được tái cấu trúc. Theo một số ví dụ, các bộ lọc có thể được áp dụng thay thế cho các khối phần dư. Như với các thành phần khác trên Fig.2, thành phần phân tích điều khiển bộ lọc 227 và thành phần các bộ lọc trong vòng lặp 225 được tích hợp cao và có thể được thực hiện cùng nhau, nhưng được mô tả một cách riêng biệt cho các mục đích khái niệm. Các bộ lọc được áp dụng cho các khối tham chiếu được tái cấu trúc được áp dụng cho các vùng theo không gian cụ thể và bao gồm nhiều thông số để điều chỉnh cách thức các bộ lọc như vậy được áp dụng. Thành phần phân tích điều khiển bộ lọc 227 phân tích các khối tham chiếu được tái cấu trúc để xác định nơi các bộ lọc như vậy sẽ được áp dụng và thiết đặt các thông số tương ứng. Dữ liệu như vậy được chuyển tiếp tới thành phần CABAC và định dạng đoạn đầu 231 như dữ liệu điều khiển bộ lọc để mã hóa. Thành phần các bộ lọc trong vòng lặp 225 áp dụng các bộ lọc như vậy dựa vào dữ liệu điều khiển bộ lọc. Các bộ lọc có thể bao gồm bộ lọc tách khối, bộ lọc triệt nhiễu, bộ lọc SAO, và bộ lọc vòng

lập thích nghi. Các bộ lọc như vậy có thể được áp dụng trong miền điểm ảnh/không gian (ví dụ, trên khối điểm ảnh được tái cấu trúc) hoặc trong miền tần số, tùy thuộc vào ví dụ.

Khi hoạt động như bộ mã hóa, khối hình ảnh được tái cấu trúc được lọc, khối phần dư, và/hoặc khối dự đoán được lưu trữ trong thành phần bộ đệm ảnh được giải mã 223 cho việc sử dụng sau đó trong việc đánh giá chuyển động như được thảo luận ở trên. Khi hoạt động như bộ giải mã, thành phần bộ đệm ảnh được giải mã 223 lưu trữ và chuyển tiếp các khối được tái cấu trúc và được lọc về phía màn hình như phần của tín hiệu video đầu ra. Thành phần bộ đệm ảnh được giải mã 223 có thể là thiết bị nhớ bất kỳ có khả năng lưu trữ các khối dự đoán, các khối phần dư, và/hoặc các khối hình ảnh được tái cấu trúc.

Thành phần CABAC và định dạng đoạn đầu 231 thu dữ liệu từ các thành phần khác nhau của hệ thống mã hóa-giải mã 200 và mã hóa dữ liệu như vậy thành dòng bit được lập mã cho việc truyền về phía bộ giải mã. Cụ thể là, thành phần CABAC và định dạng đoạn đầu 231 tạo ra các đoạn đầu khác nhau để mã hóa dữ liệu điều khiển, chẳng hạn như dữ liệu điều khiển chung và dữ liệu điều khiển bộ lọc. Hơn nữa, dữ liệu dự đoán, bao gồm dữ liệu chuyển động và dự đoán bên trong, cũng như dữ liệu phần dư ở dạng dữ liệu hệ số biến đổi được lượng tử hóa tất cả được mã hóa ở dòng bit. Dòng bit cuối cùng bao gồm tất cả thông tin được mong muốn bởi bộ giải mã để tái cấu trúc tín hiệu video được phân chia ban đầu 201. Thông tin như vậy cũng có thể bao gồm các bảng chỉ số chế độ dự đoán bên trong (cũng được đề cập đến là các bảng ánh xạ từ mã), các định nghĩa của các ngữ cảnh mã hóa cho các khối khác nhau, các chỉ báo của các chế độ dự đoán bên trong có thể xảy ra nhất, chỉ báo của thông tin phân chia, v.v. Dữ liệu như vậy có thể được mã hóa bằng cách ứng dụng việc lập mã entropi. Ví dụ, thông tin có thể được mã hóa bằng cách ứng dụng việc lập mã độ dài thay đổi thích ứng ngữ cảnh (context adaptive variable length coding, viết tắt là CAVLC), CABAC, lập mã số học nhị phân thích ứng ngữ cảnh dựa trên cú pháp (syntax-based context-adaptive binary arithmetic coding, viết tắt là SBAC), lập mã phân chia entropi theo khoảng xác suất (probability interval partitioning entropy, viết tắt là PIPE), hoặc kỹ thuật lập mã entropi khác. Theo sau việc lập mã entropi, dòng bit được lập mã có thể được truyền tới thiết bị khác (ví dụ, bộ giải mã video) hoặc được lưu trữ cho việc truyền hoặc truy xuất sau đó.

Fig.3 là sơ đồ khối minh họa bộ mã hóa video 300 ví dụ. Bộ mã hóa video 300 có thể được ứng dụng để thực hiện các chức năng mã hóa của hệ thống mã hóa-giải mã 200 và/hoặc thực hiện các bước 101, 103, 105, 107, và/hoặc 109 của phương pháp vận hành 100. Bộ mã hóa 300 phân chia tín hiệu video đầu vào, dẫn đến tín hiệu video được phân chia 301, mà về

cơ bản tương tự với tín hiệu video được phân chia 201. Tín hiệu video được phân chia 301 sau đó được nén và được mã hóa thành dòng bit bởi các thành phần của bộ mã hóa 300.

Cụ thể là, tín hiệu video được phân chia 301 được chuyển tiếp tới thành phần dự đoán ảnh bên trong 317 cho việc dự đoán bên trong. Thành phần dự đoán ảnh bên trong 317 về cơ bản có thể tương tự với thành phần đánh giá ảnh bên trong 215 và thành phần dự đoán ảnh bên trong 217. Tín hiệu video được phân chia 301 cũng được chuyển tiếp tới thành phần bù chuyển động 321 cho việc dự đoán liên kết dựa vào các khối tham chiếu trong thành phần bộ đệm ảnh được giải mã 323. Thành phần bù chuyển động 321 về cơ bản có thể tương tự với thành phần đánh giá chuyển động 221 và thành phần bù chuyển động 219. Các khối dự đoán và các khối phần dư từ thành phần dự đoán ảnh bên trong 317 và thành phần bù chuyển động 321 được chuyển tiếp tới thành phần biến đổi và lượng tử hóa 313 cho việc biến đổi và lượng tử hóa của các khối phần dư. Thành phần biến đổi và lượng tử hóa 313 về cơ bản có thể tương tự với thành phần định tỷ lệ và lượng tử hóa biến đổi 213. Các khối phần dư được biến đổi và được lượng tử hóa và các khối dự đoán tương ứng (cùng với dữ liệu điều khiển được kết hợp) được chuyển tiếp tới thành phần lập mã entropi 331 để lập mã thành dòng bit. Thành phần lập mã entropi 331 về cơ bản có thể tương tự với thành phần CABAC và định dạng đoạn đầu 231.

Các khối phần dư được biến đổi và được lượng tử hóa và/hoặc các khối dự đoán tương ứng cũng được chuyển tiếp từ thành phần biến đổi và lượng tử hóa 313 tới thành phần biến đổi và lượng tử hóa ngược 329 cho việc tái cấu trúc thành các khối tham chiếu để sử dụng bởi thành phần bù chuyển động 321. Thành phần biến đổi và lượng tử hóa ngược 329 về cơ bản có thể tương tự với thành phần định tỷ lệ và biến đổi ngược 229. Các bộ lọc trong vòng lặp trong thành phần các bộ lọc trong vòng lặp 325 cũng được áp dụng cho các khối phần dư và/hoặc các khối tham chiếu được tái cấu trúc, tùy thuộc vào ví dụ. Thành phần các bộ lọc trong vòng lặp 325 về cơ bản có thể tương tự với thành phần phân tích điều khiển bộ lọc 227 và thành phần các bộ lọc trong vòng lặp 225. Thành phần các bộ lọc trong vòng lặp 325 có thể bao gồm nhiều bộ lọc như được thảo luận đối với thành phần các bộ lọc trong vòng lặp 225. Các khối được lọc sau đó được lưu trữ trong thành phần bộ đệm ảnh được giải mã 323 để sử dụng như các khối tham chiếu bởi thành phần bù chuyển động 321. Thành phần bộ đệm ảnh được giải mã 323 về cơ bản có thể tương tự với thành phần bộ đệm ảnh được giải mã 223.

Fig.4 là sơ đồ khối minh họa bộ giải mã video 400 ví dụ. Bộ giải mã video 400 có thể được ứng dụng để thực hiện các chức năng giải mã của hệ thống mã hóa-giải mã 200 và/hoặc thực hiện các bước 111, 113, 115, và/hoặc 117 của phương pháp vận hành 100. Bộ giải mã 400 thu dòng bit, ví dụ từ bộ mã hóa 300, và tạo ra tín hiệu video đầu ra được tái cấu trúc dựa vào dòng bit cho việc hiển thị tới người dùng cuối.

Dòng bit được thu bởi thành phần giải mã entropi 433. Thành phần giải mã entropi 433 được tạo cấu hình để thực hiện sơ đồ giải mã entropi, chẳng hạn như lập mã CAVLC, CABAC, SBAC, PIPE, hoặc các kỹ thuật lập mã entropi khác. Ví dụ, thành phần giải mã entropi 433 có thể ứng dụng thông tin đoạn đầu để cung cấp ngữ cảnh để diễn dịch dữ liệu bổ sung được mã hóa như các từ mã ở dòng bit. Thông tin được giải mã bao gồm thông tin bất kỳ được mong muốn để giải mã tín hiệu video, chẳng hạn như dữ liệu điều khiển chung, dữ liệu điều khiển bộ lọc, thông tin phân chia, dữ liệu chuyển động, dữ liệu dự đoán, và các hệ số biến đổi được lượng tử hóa từ các khối phần dư. Các hệ số biến đổi được lượng tử hóa được chuyển tiếp tới thành phần biến đổi và lượng tử hóa ngược 429 cho việc tái cấu trúc thành các khối phần dư. Thành phần biến đổi và lượng tử hóa ngược 429 có thể tương tự với thành phần biến đổi và lượng tử hóa ngược 329.

Các khối dự đoán và/hoặc các khối phần dư được tái cấu trúc được chuyển tiếp tới thành phần dự đoán ảnh bên trong 417 cho việc tái cấu trúc thành các khối hình ảnh dựa vào các thao tác dự đoán bên trong. Thành phần dự đoán ảnh bên trong 417 có thể tương tự với thành phần đánh giá ảnh bên trong 215 và thành phần dự đoán ảnh bên trong 217. Cụ thể là, thành phần dự đoán ảnh bên trong 417 ứng dụng các chế độ dự đoán để định vị khối tham chiếu trong khung và áp dụng khối phần dư cho kết quả để tái cấu trúc các khối ảnh được dự đoán bên trong. Các khối ảnh được dự đoán bên trong được tái cấu trúc và/hoặc các khối phần dư và dữ liệu dự đoán liên kết tương ứng được chuyển tiếp tới thành phần bộ đệm ảnh được giải mã 423 qua thành phần các bộ lọc trong vòng lặp 425, mà lần lượt về cơ bản có thể tương tự với thành phần bộ đệm ảnh được giải mã 223 và thành phần các bộ lọc trong vòng lặp 225. Thành phần các bộ lọc trong vòng lặp 425 lọc các khối ảnh được tái cấu trúc, các khối phần dư và/hoặc các khối dự đoán, và thông tin như vậy được lưu trữ trong thành phần bộ đệm ảnh được giải mã 423. Các khối ảnh được tái cấu trúc từ thành phần bộ đệm ảnh được giải mã 423 được chuyển tiếp tới thành phần bù chuyển động 421 cho việc dự đoán liên kết. Thành phần bù chuyển động 421 về cơ bản có thể tương tự với thành phần đánh giá chuyển động 221 và/hoặc thành phần bù chuyển động 219. Cụ thể là, thành phần bù chuyển

động 421 ứng dụng các vectơ chuyển động từ khối tham chiếu để tạo ra khối dự đoán và áp dụng khối phần dư cho kết quả để tái cấu trúc khối hình ảnh. Các khối được tái cấu trúc nhận được cũng có thể được chuyển tiếp qua thành phần các bộ lọc trong vòng lặp 425 tới thành phần bộ đệm ảnh được giải mã 423. Thành phần bộ đệm ảnh được giải mã 423 tiếp tục lưu trữ các khối hình ảnh được tái cấu trúc bổ sung, mà có thể được tái cấu trúc thành các khung qua thông tin phân chia. Các khung như vậy cũng có thể được đặt trong chuỗi. Chuỗi được đưa ra về phía màn hình như tín hiệu video đầu ra được tái cấu trúc.

Fig.5 là hình vẽ giản lược minh họa HRD 500 ví dụ. HRD 500 có thể được ứng dụng ở bộ mã hóa, chẳng hạn như hệ thống mã hóa-giải mã 200 và/hoặc bộ mã hóa 300. HRD 500 có thể kiểm tra dòng bit được tạo ra ở bước 109 của phương pháp 100 trước khi dòng bit được chuyển tiếp tới bộ giải mã, chẳng hạn như bộ giải mã 400. Theo một số ví dụ, dòng bit có thể liên tục được chuyển tiếp thông qua HRD 500 khi dòng bit được mã hóa. Trong trường hợp phần của dòng bit không tuân theo các sự ràng buộc được kết hợp, HRD 500 có thể chỉ báo lỗi như vậy tới bộ mã hóa để khiến bộ mã hóa mã hóa lại phần tương ứng của dòng bit với các cơ chế khác.

HRD 500 bao gồm bộ lập lịch dòng giả định (hypothetical stream scheduler, viết tắt là HSS) 541. HSS 541 là thành phần được tạo cấu hình để thực hiện cơ chế phân phối giả định. Cơ chế phân phối giả định được sử dụng để kiểm tra sự phù hợp của dòng bit hoặc bộ giải mã liên quan đến thời điểm và luồng dữ liệu của dòng bit 551 được đưa vào HRD 500. Ví dụ, HSS 541 có thể thu dòng bit 551 được đưa ra từ bộ mã hóa và quản lý quy trình thử nghiệm sự phù hợp trên dòng bit 551. Theo ví dụ cụ thể, HSS 541 có thể điều khiển tốc độ mà các ảnh được lập mã di chuyển thông qua HRD 500 và xác minh rằng dòng bit 551 không chứa dữ liệu không phù hợp.

HSS 541 có thể chuyển tiếp dòng bit 551 tới CPB 543 ở tốc độ được định rõ trước. HRD 500 có thể quản lý dữ liệu ở các bộ phận giải mã (decoding unit, viết tắt là DU) 553. DU 553 là bộ phận truy cập (AU) hoặc tập hợp con của AU và các bộ phận lớp trừu tượng mạng (NAL) không phải lớp lập mã video (VCL) được kết hợp. Cụ thể là, AU chứa một hoặc nhiều ảnh được kết hợp với thời gian đầu ra. Ví dụ, AU có thể chứa ảnh đơn ở dòng bit lớp đơn, và có thể chứa ảnh cho mỗi lớp ở dòng bit đa lớp. Mỗi ảnh của AU có thể được chia thành các lát mà đều được bao gồm trong bộ phận VCL NAL tương ứng. Do đó, DU 553 có thể chứa một hoặc nhiều ảnh, một hoặc nhiều lát của ảnh, hoặc các sự kết hợp của chúng. Ngoài ra, các thông số được sử dụng để giải mã AU, các ảnh, và/hoặc các lát có thể được

bao gồm trong các bộ phận không phải NAL VCL. Như vậy, DU 553 chứa các bộ phận không phải NAL VCL mà chứa dữ liệu được cần để hỗ trợ việc giải mã các bộ phận VCL NAL trong DU 553. CPB 543 là bộ đệm vào trước ra trước trong HRD 500. CPB 543 chứa các DU 553 bao gồm dữ liệu video theo thứ tự giải mã. CPB 543 lưu trữ dữ liệu video để sử dụng trong suốt thời gian xác minh sự phù hợp dòng bit.

CPB 543 chuyển tiếp các DU 553 tới thành phần quy trình giải mã 545. Thành phần quy trình giải mã 545 là thành phần mà phù hợp với chuẩn VVC. Ví dụ, thành phần quy trình giải mã 545 có thể mô phỏng bộ giải mã 400 được ứng dụng bởi người dùng cuối. Thành phần quy trình giải mã 545 giải mã các DU 553 ở tốc độ mà có thể đạt được bởi bộ giải mã người dùng cuối ví dụ. Nếu thành phần quy trình giải mã 545 không thể giải mã các DU 553 đủ nhanh để ngăn ngừa việc tràn của CPB 543, thì sau đó dòng bit 551 không phù hợp với chuẩn và sẽ được mã hóa lại.

Thành phần quy trình giải mã 545 giải mã các DU 553, mà tạo ra các DU 555 được giải mã. DU 555 được giải mã chứa ảnh được giải mã. Các DU 555 được giải mã được chuyển tiếp tới DPB 547. DPB 547 về cơ bản có thể tương tự với thành phần bộ đệm ảnh được giải mã 223, 323, và/hoặc 423. Để hỗ trợ việc dự đoán liên kết, các ảnh mà được đánh dấu để sử dụng như các ảnh tham chiếu 556 mà được thu nhận từ các DU 555 được giải mã được gửi trả lại tới thành phần quy trình giải mã 545 để hỗ trợ việc giải mã thêm. DPB 547 đưa ra chuỗi video được giải mã như loạt của các ảnh 557. Các ảnh 557 là các ảnh được tái cấu trúc mà nói chung phản chiếu các ảnh được mã hóa thành dòng bit 551 bởi bộ mã hóa.

Các ảnh 557 được chuyển tiếp tới thành phần cắt xén đầu ra 549. Thành phần cắt xén đầu ra 549 được tạo cấu hình để áp dụng cửa sổ cắt xén sự phù hợp cho các ảnh 557. Điều này dẫn đến các ảnh được cắt xén đầu ra 559. Ảnh được cắt xén đầu ra 559 là ảnh được tái cấu trúc hoàn toàn. Theo đó, ảnh được cắt xén đầu ra 559 bất chước điều mà người dùng cuối sẽ thấy khi giải mã dòng bit 551. Như vậy, bộ mã hóa có thể xem trước các ảnh được cắt xén đầu ra 559 để đảm bảo việc mã hóa đạt yêu cầu.

HRD 500 được khởi tạo dựa vào các thông số HRD ở dòng bit 551. Ví dụ, HRD 500 có thể đọc các thông số HRD từ VPS, SPS, và/hoặc các tín hiệu SEI. HRD 500 sau đó có thể thực hiện các thao tác thử nghiệm sự phù hợp trên dòng bit 551 dựa vào thông tin trong các thông số HRD như vậy. Với ví dụ cụ thể, HRD 500 có thể xác định một hoặc nhiều việc lập lịch phân phối CPB từ các thông số HRD. Việc lập lịch phân phối xác định thời điểm cho việc phân phối của dữ liệu video tới và/hoặc từ vị trí bộ nhớ, chẳng hạn như CPB và/hoặc

DPB. Do đó, việc lập lịch phân phối CPB xác định thời điểm cho việc phân phối của các AU, các DU 553, và/hoặc các ảnh, tới/từ CPB 543. Cần lưu ý rằng HRD 500 có thể ứng dụng các việc lập lịch phân phối DPB cho DPB 547 mà tương tự với các việc lập lịch phân phối CPB.

Video có thể được lập mã thành các lớp và/hoặc các OLS khác nhau để sử dụng bởi các bộ giải mã với việc thay đổi các mức độ của các dung lượng phần cứng cũng như để thay đổi các điều kiện mạng. Các việc lập lịch phân phối CPB được lựa chọn để phản ánh các vấn đề này. Theo đó, các dòng bit con lớp cao hơn được chỉ định cho phần cứng tốt nhất và các điều kiện mạng và do đó các lớp cao hơn có thể thu một hoặc nhiều việc lập lịch phân phối CPB mà ứng dụng lượng lớn của bộ nhớ trong CPB 543 và các sự trì hoãn ngắn cho các việc chuyển đổi của các DU 553 về phía DPB 547. Tương tự như vậy, các dòng bit con lớp thấp hơn được chỉ định các dung lượng phần cứng bộ giải mã giới hạn và/hoặc các điều kiện mạng kém. Do đó, các lớp thấp hơn có thể thu một hoặc nhiều việc lập lịch phân phối CPB mà ứng dụng lượng nhỏ của bộ nhớ trong CPB 543 và các sự trì hoãn dài hơn cho các việc chuyển đổi của các DU 553 về phía DPB 547. Các OLS, các lớp, các lớp con, hoặc các sự kết hợp của chúng sau đó có thể được thử nghiệm theo việc lập lịch phân phối tương ứng để đảm bảo rằng dòng bit con nhận được có thể được giải mã một cách chính xác theo các điều kiện mà được mong muốn cho dòng bit con. Theo đó, các thông số HRD ở dòng bit 551 có thể chỉ báo các việc lập lịch phân phối CPB cũng như bao gồm dữ liệu đủ để cho phép HRD 500 xác định các việc lập lịch phân phối CPB và tương quan các việc lập lịch phân phối CPB với các OLS, các lớp, và/hoặc các lớp con tương ứng.

Fig.6 là hình vẽ giản lược minh họa chuỗi video đa lớp 600 ví dụ được tạo cấu hình cho việc dự đoán liên lớp 621. Chuỗi video đa lớp 600 có thể được mã hóa bởi bộ mã hóa, chẳng hạn như hệ thống mã hóa-giải mã 200 và/hoặc bộ mã hóa 300 và được giải mã bởi bộ giải mã, chẳng hạn như hệ thống mã hóa-giải mã 200 và/hoặc bộ giải mã 400, ví dụ theo phương pháp 100. Hơn nữa, chuỗi video đa lớp 600 có thể được kiểm tra về sự phù hợp chuẩn bởi HRD, chẳng hạn như HRD 500. Chuỗi video đa lớp 600 được bao gồm để mô tả ứng dụng ví dụ cho các lớp trong chuỗi video được lập mã. Chuỗi video đa lớp 600 là chuỗi video bất kỳ mà ứng dụng nhiều lớp, chẳng hạn như lớp N 631 và lớp N+1 632.

Theo ví dụ, chuỗi video đa lớp 600 có thể ứng dụng việc dự đoán liên lớp 621. Việc dự đoán liên lớp 621 được áp dụng giữa các ảnh 611, 612, 613, và 614 và các ảnh 615, 616, 617, và 618 ở các lớp khác nhau. Theo ví dụ được thể hiện, các ảnh 611, 612, 613, và 614 là

phần của lớp N+1 632 và các ảnh 615, 616, 617, và 618 là phần của lớp N 631. Lớp, chẳng hạn như lớp N 631 và/hoặc lớp N+1 632, là nhóm của các ảnh mà tất cả được kết hợp với trị số tương tự của đặc tính, chẳng hạn như kích thước, chất lượng, độ phân giải, tỷ lệ tín hiệu trên nhiễu, dung lượng tương tự, v.v. Lớp có thể được định rõ chính thức là tập hợp của các bộ phận VCL NAL và các bộ phận không phải NAL VCL được kết hợp. Bộ phận VCL NAL là bộ phận NAL được lập mã để chứa dữ liệu video, chẳng hạn như lát được lập mã của ảnh. Bộ phận không phải VCL NAL là bộ phận NAL mà chứa không phải dữ liệu video chẳng hạn như cú pháp và/hoặc các thông số mà hỗ trợ việc giải mã dữ liệu video, hiệu suất của việc kiểm tra sự phù hợp, hoặc các thao tác khác.

Theo ví dụ được thể hiện, lớp N+1 632 được kết hợp với kích thước hình ảnh lớn hơn lớp N 631. Theo đó, các ảnh 611, 612, 613, và 614 ở lớp N+1 632 có kích thước ảnh lớn hơn (ví dụ, độ cao và độ rộng lớn hơn và do đó nhiều mẫu hơn) so với các ảnh 615, 616, 617, và 618 ở lớp N 631 theo ví dụ này. Tuy nhiên, các ảnh như vậy có thể được tách riêng giữa lớp N+1 632 và lớp N 631 bởi các đặc tính khác. Trong khi chỉ hai lớp, lớp N+1 632 và lớp N 631, được thể hiện, tập hợp của các ảnh có thể được tách riêng thành số các lớp bất kỳ dựa vào các đặc tính được kết hợp. Lớp N+1 632 và lớp N 631 cũng có thể được biểu thị bởi Id lớp. Id lớp là mục của dữ liệu mà được kết hợp với ảnh và biểu thị ảnh là phần của lớp được chỉ báo. Theo đó, mỗi ảnh 611-618 có thể được kết hợp với Id lớp tương ứng để chỉ báo lớp N+1 632 hoặc lớp N 631 nào bao gồm ảnh tương ứng. Ví dụ, Id lớp có thể bao gồm ký hiệu nhận dạng lớp đoạn đầu bộ phận NAL (nuh_layer_id), mà là thành phần cú pháp mà xác định ký hiệu nhận dạng của lớp mà bao gồm bộ phận NAL (ví dụ, mà bao gồm các lát và/hoặc các thông số của các ảnh trong lớp). Lớp được kết hợp với kích thước dòng bit/chất lượng thấp hơn, chẳng hạn như lớp N 631, nói chung được gán Id lớp thấp hơn và được đề cập đến là lớp thấp hơn. Hơn nữa, lớp được kết hợp với kích thước dòng bit/chất lượng cao hơn, chẳng hạn như lớp N+1 632, nói chung được gán Id lớp cao hơn và được đề cập đến là lớp cao hơn.

Các ảnh 611-618 ở các lớp khác nhau 631-632 được tạo cấu hình để được hiển thị theo cách khác. Với ví dụ cụ thể, bộ giải mã có thể giải mã và hiển thị ảnh 615 ở thời gian hiển thị hiện thời nếu ảnh nhỏ hơn được mong muốn hoặc bộ giải mã có thể giải mã và hiển thị ảnh 611 ở thời gian hiển thị hiện thời nếu ảnh lớn hơn được mong muốn. Như vậy, các ảnh 611-614 ở lớp N+1 632 cao hơn chứa một cách đáng kể dữ liệu hình ảnh giống như các ảnh 615-618 tương ứng ở lớp N 631 thấp hơn (bất kể độ chênh lệch về kích thước ảnh). Cụ

thể là, ảnh 611 chứa một cách đáng kể dữ liệu hình ảnh giống như ảnh 615, ảnh 612 chứa một cách đáng kể dữ liệu hình ảnh giống như ảnh 616, v.v.

Các ảnh 611-618 có thể được lập mã bằng cách tham chiếu tới các ảnh 611-618 khác trong cùng lớp N 631 hoặc N+1 632. Việc lập mã ảnh tham chiếu tới ảnh khác ở cùng lớp dẫn đến việc dự đoán liên kết 623. Việc dự đoán liên kết 623 được mô tả bởi các mũi tên nét liền. Ví dụ, ảnh 613 có thể được lập mã bằng cách ứng dụng việc dự đoán liên kết 623 nhờ sử dụng một hoặc hai trong số các ảnh 611, 612, và/hoặc 614 ở lớp N+1 632 như tham chiếu, trong đó một ảnh được tham chiếu cho việc dự đoán liên kết đơn hướng và/hoặc hai ảnh được tham chiếu cho việc dự đoán liên kết hai chiều. Hơn nữa, ảnh 617 có thể được lập mã bằng cách ứng dụng việc dự đoán liên kết 623 nhờ sử dụng một hoặc hai trong số các ảnh 615, 616, và/hoặc 618 ở lớp N 631 như tham chiếu, trong đó một ảnh được tham chiếu cho việc dự đoán liên kết đơn hướng và/hoặc hai ảnh được tham chiếu cho việc dự đoán liên kết hai chiều. Khi ảnh được sử dụng như tham chiếu cho ảnh khác ở cùng lớp khi thực hiện việc dự đoán liên kết 623, ảnh có thể được đề cập đến là ảnh tham chiếu. Ví dụ, ảnh 612 có thể là ảnh tham chiếu được sử dụng để lập mã ảnh 613 theo việc dự đoán liên kết 623. Việc dự đoán liên kết 623 cũng có thể được đề cập đến là việc dự đoán lớp bên trong trong ngữ cảnh đa lớp. Như vậy, việc dự đoán liên kết 623 là cơ chế lập mã các mẫu của ảnh hiện thời bằng cách tham chiếu tới các mẫu được chỉ báo trong ảnh tham chiếu mà khác với ảnh hiện thời trong đó ảnh tham chiếu và ảnh hiện thời ở cùng lớp.

Các ảnh 611-618 cũng có thể được lập mã bằng cách tham chiếu tới các ảnh 611-618 khác ở các lớp khác nhau. Quy trình xử lý này được biết đến là việc dự đoán liên lớp 621, và được mô tả bởi các mũi tên nét đứt. Việc dự đoán liên lớp 621 là cơ chế lập mã các mẫu của ảnh hiện thời bằng cách tham chiếu tới các mẫu được chỉ báo trong ảnh tham chiếu trong đó ảnh hiện thời và ảnh tham chiếu ở các lớp khác nhau và do đó có các Id lớp khác nhau. Ví dụ, ảnh ở lớp N 631 thấp hơn có thể được sử dụng như ảnh tham chiếu để lập mã ảnh tương ứng ở lớp N+1 632 cao hơn. Với ví dụ cụ thể, ảnh 611 có thể được lập mã bằng cách tham chiếu tới ảnh 615 theo việc dự đoán liên lớp 621. Trong trường hợp như vậy, ảnh 615 được sử dụng như ảnh tham chiếu liên lớp. Ảnh tham chiếu liên lớp là ảnh tham chiếu được sử dụng cho việc dự đoán liên lớp 621. Trong hầu hết các trường hợp, việc dự đoán liên lớp 621 bị ràng buộc sao cho ảnh hiện thời, chẳng hạn như ảnh 611, có thể chỉ sử dụng (các) ảnh tham chiếu liên lớp mà được bao gồm trong cùng AU và ở lớp thấp hơn, chẳng hạn như ảnh 615. Khi nhiều lớp (ví dụ, nhiều hơn hai) khả dụng, việc dự đoán liên lớp 621 có thể mã

hóa/giải mã ảnh hiện thời dựa vào nhiều (các) ảnh tham chiếu liên lớp ở các mức độ thấp hơn so với ảnh hiện thời.

Bộ mã hóa video có thể ứng dụng chuỗi video đa lớp 600 để mã hóa các ảnh 611-618 qua nhiều sự kết hợp khác nhau và/hoặc các sự hoán vị của việc dự đoán liên kết 623 và việc dự đoán liên lớp 621. Ví dụ, ảnh 615 có thể được lập mã theo việc dự đoán bên trong. Các ảnh 616-618 sau đó có thể được lập mã theo việc dự đoán liên kết 623 nhờ sử dụng ảnh 615 như ảnh tham chiếu. Hơn nữa, ảnh 611 có thể được lập mã theo việc dự đoán liên lớp 621 nhờ sử dụng ảnh 615 như ảnh tham chiếu liên lớp. Các ảnh 612-614 sau đó có thể được lập mã theo việc dự đoán liên kết 623 nhờ sử dụng ảnh 611 như ảnh tham chiếu. Như vậy, ảnh tham chiếu có thể dùng làm cả ảnh tham chiếu lớp đơn và ảnh tham chiếu liên lớp đối với các cơ chế lập mã khác nhau. Bằng cách lập mã các ảnh lớp $N+1$ 632 cao hơn dựa vào các ảnh lớp N 631 thấp hơn, lớp $N+1$ 632 cao hơn có thể tránh khỏi việc ứng dụng việc dự đoán bên trong, mà có hiệu quả lập mã thấp hơn nhiều so với việc dự đoán liên kết 623 và việc dự đoán liên lớp 621. Như vậy, hiệu quả lập mã kém của việc dự đoán bên trong có thể giới hạn ở các ảnh chất lượng thấp nhất/nhỏ nhất, và do đó giới hạn ở việc lập mã lượng dữ liệu video nhỏ nhất. Các ảnh được sử dụng như các ảnh tham chiếu và/hoặc các ảnh tham chiếu liên lớp có thể được chỉ báo trong các mục nhập của (các) danh mục ảnh tham chiếu được chứa trong cấu trúc danh mục ảnh tham chiếu.

Các ảnh 611-618 cũng có thể được bao gồm trong các bộ phận truy cập (AU) 627 và 628. AU 627-628 là tập hợp của các ảnh được lập mã mà được bao gồm ở các lớp khác nhau và được kết hợp với cùng thời gian đầu ra trong suốt thời gian giải mã. Theo đó, các ảnh được lập mã trong cùng AU 627-628 được lập lịch để được đưa ra từ DPB ở bộ giải mã đồng thời. Ví dụ, các ảnh 614 và 618 trong cùng AU 628. Các ảnh 613 và 617 ở AU 627 khác với các ảnh 614 và 618. Các ảnh 614 và 618 trong cùng AU 628 có thể được hiển thị theo cách khác. Ví dụ, ảnh 618 có thể được hiển thị khi kích thước ảnh nhỏ được mong muốn và ảnh 614 có thể được hiển thị khi kích thước ảnh lớn được mong muốn. Khi kích thước ảnh lớn được mong muốn, ảnh 614 được đưa ra và ảnh 618 được sử dụng chỉ cho việc dự đoán liên lớp 621. Trong trường hợp này, ảnh 618 được loại bỏ mà không được đưa ra khi việc dự đoán liên lớp 621 là hoàn chỉnh. AU 627 mà bao gồm ảnh ở tất cả các lớp có thể được đề cập đến là AU hoàn chỉnh 627. AU 628 mà không bao gồm các ảnh ở tất cả các lớp có thể được đề cập đến là AU không hoàn chỉnh 628.

Lớp N-1 633 được bao gồm để mô tả kịch bản mà AU 628 không hoàn chỉnh. Lớp N-1 633 bao gồm các ảnh 625 và 626, mà lần lượt về cơ bản giống với các ảnh 615 và 617. Tuy nhiên, lớp N-1 633 không chứa các ảnh mà tương ứng với các ảnh 616 và 618. Theo đó, lớp N-1 633 về cơ bản tương tự với lớp N 631, nhưng chứa tỷ lệ khung thấp hơn. Theo đó, AU 627 mà chứa các ảnh 611, 615, và 625 là hoàn chỉnh. Tuy nhiên, AU 628 mà chứa các ảnh 612 và 616 là AU không hoàn chỉnh 628 bởi vì AU 628 này đang thiếu ảnh ở lớp N-1 633.

Việc mã hóa các AU không hoàn chỉnh 628 có thể hữu dụng khi điều này cho phép chuỗi video đa lớp 600 được truyền tới các người dùng khác nhau ở các tỷ lệ khung khác nhau. Tuy nhiên, các AU không hoàn chỉnh 628 có thể tạo ra các thách thức khi kiểm tra chuỗi video được lập mã về sự phù hợp. Ví dụ, các tin nhắn SEI có thể được ứng dụng để truyền các thông số mà được sử dụng bởi HRD, chẳng hạn như HRD 500, khi kiểm tra chuỗi video đa lớp 600 về sự phù hợp với các chuẩn. Một số hệ thống lập mã video cho phép các tin nhắn SEI áp dụng cho tất cả các lớp trong AU 627-628. Vấn đề là các tin nhắn SEI như vậy có thể ổn định qua nhiều AU 627-628. Khi tin nhắn SEI áp dụng cho tất cả các lớp trong AU không hoàn chỉnh 628, các tin nhắn SEI như vậy không được áp dụng cho lớp N-1 633 bởi vì lớp N-1 633 không được bao gồm trong AU không hoàn chỉnh 628. Theo đó, tin nhắn SEI không được áp dụng cho lớp N-1 633 trong các AU hoàn chỉnh 627 theo sau AU không hoàn chỉnh 628. Điều này có thể gây ra các lỗi không thể dự đoán khi các tin nhắn SEI được áp dụng đúng cách cho lớp N-1 633 khi tin nhắn như vậy xảy ra được kết hợp với AU hoàn chỉnh 627 và không được áp dụng cho lớp N-1 633 khi tin nhắn như vậy xảy ra được kết hợp với AU không hoàn chỉnh 628. Vấn đề này và các vấn đề khác được xử lý bởi sơ đồ báo hiệu được mô tả dưới đây.

Fig.7 là hình vẽ giản lược minh họa dòng bit 700 ví dụ. Ví dụ, dòng bit 700 có thể được tạo ra bởi hệ thống mã hóa-giải mã 200 và/hoặc bộ mã hóa 300 để giải mã bởi hệ thống mã hóa-giải mã 200 và/hoặc bộ giải mã 400 theo phương pháp 100. Hơn nữa, dòng bit 700 có thể bao gồm chuỗi video đa lớp 600. Ngoài ra, dòng bit 700 có thể bao gồm các thông số khác nhau để điều khiển hoạt động của HRD, chẳng hạn như HRD 500. Dựa vào các thông số như vậy, HRD 500 có thể kiểm tra dòng bit 700 về sự phù hợp với các chuẩn trước khi truyền về phía bộ giải mã để giải mã.

Dòng bit 700 bao gồm VPS 711, một hoặc nhiều SPS 713, nhiều tập hợp thông số ảnh (PPS) 715, nhiều đoạn đầu lát 717, dữ liệu hình ảnh 720, và các tin nhắn SEI 719. VPS

711 chứa dữ liệu liên quan đến toàn bộ dòng bit 700. Ví dụ, VPS 711 có thể chứa các OLS, các lớp, và/hoặc các lớp con liên quan đến dữ liệu được sử dụng ở dòng bit 700. SPS 713 chứa dữ liệu chuỗi phổ biến với tất cả các ảnh trong chuỗi video được lập mã được chứa ở dòng bit 700. Ví dụ, mỗi lớp có thể chứa một hoặc nhiều chuỗi video được lập mã, và mỗi chuỗi video được lập mã có thể tham chiếu SPS 713 cho các thông số tương ứng. Các thông số trong SPS 713 có thể bao gồm việc định cỡ ảnh, độ sâu bit, các thông số công cụ lập mã, các giới hạn tốc độ bit, v.v. Cần lưu ý rằng, trong khi mỗi chuỗi đề cập đến SPS 713, SPS 713 đơn có thể chứa dữ liệu cho nhiều chuỗi theo một số ví dụ. PPS 715 chứa các thông số mà áp dụng cho toàn bộ ảnh. Do đó, mỗi ảnh trong chuỗi video có thể đề cập đến PPS 715. Cần lưu ý rằng, trong khi mỗi ảnh đề cập đến PPS 715, PPS 715 đơn có thể chứa dữ liệu cho nhiều ảnh theo một số ví dụ. Ví dụ, nhiều ảnh tương tự có thể được lập mã theo các thông số tương tự. Trong trường hợp như vậy, PPS 715 đơn có thể chứa dữ liệu cho các ảnh tương tự như vậy. PPS 715 có thể chỉ báo các công cụ lập mã khả dụng cho các lát trong các ảnh, các thông số lượng tử hóa, các dịch vị tương ứng, v.v.

Đoạn đầu lát 717 chứa các thông số mà cụ thể với mỗi lát trong ảnh. Do đó, có thể có một đoạn đầu lát 717 trên mỗi lát trong chuỗi video. Đoạn đầu lát 717 có thể chứa thông tin loại lát, các POC, các danh mục ảnh tham chiếu, các trọng số dự đoán, các điểm mục nhập miếng, các thông số tách khối, v.v. Cần lưu ý rằng theo một số ví dụ, dòng bit 700 cũng có thể bao gồm đoạn đầu ảnh, mà là cấu trúc cú pháp chứa các thông số mà áp dụng cho tất cả các lát trong ảnh đơn. Vì lý do này, đoạn đầu ảnh và đoạn đầu lát 717 có thể được sử dụng thay thế cho nhau trong một số ngữ cảnh. Ví dụ, các thông số nhất định có thể được di chuyển giữa đoạn đầu lát 717 và đoạn đầu ảnh tùy thuộc vào xem các thông số như vậy phổ biến với tất cả các lát trong ảnh hay không.

Dữ liệu hình ảnh 720 chứa dữ liệu video được mã hóa theo việc dự đoán liên kết và/hoặc việc dự đoán bên trong cũng như dữ liệu phân dư được biến đổi và được lượng tử hóa tương ứng. Ví dụ, dữ liệu hình ảnh 720 có thể bao gồm các lớp 723, các ảnh 725, và/hoặc các lát 727. Lớp 723 là tập hợp của các bộ phận VCL NAL mà dùng chung đặc tính được quy định (ví dụ, độ phân giải chung, tỷ lệ khung, kích thước hình ảnh chẳng hạn) như được chỉ báo bởi Id lớp, chẳng hạn như `nuh_layer_id` 729, và các bộ phận không phải NAL VCL được kết hợp. Ví dụ, lớp 723 có thể bao gồm tập hợp của các ảnh 725 mà dùng chung cùng `nuh_layer_id`. Lớp 723 về cơ bản có thể tương tự với các lớp 631, 632, và/hoặc 633. `Nuh_layer_id` 729 là thành phần cú pháp mà xác định ký hiệu nhận dạng của lớp 723 mà bao

gồm ít nhất một bộ phận NAL. Ví dụ, lớp chất lượng thấp nhất 723, được biết đến là lớp cơ sở, có thể bao gồm trị số thấp nhất của `nuh_layer_id` 729 với các trị số tăng lên của `nuh_layer_id` 729 cho các lớp 723 có chất lượng cao hơn. Do đó, lớp thấp hơn là lớp 723 với trị số nhỏ hơn của `nuh_layer_id` 729 và lớp cao hơn là lớp 723 với trị số lớn hơn của `nuh_layer_id` 729.

Ảnh 725 là mảng của các mẫu độ chói và/hoặc mảng của các mẫu sắc độ mà tạo ra khung hoặc trường của chúng. Ví dụ, ảnh 725 là ảnh được lập mã mà có thể được đưa ra cho việc hiển thị hoặc được sử dụng để hỗ trợ việc lập mã của (các) ảnh khác 725 cho đầu ra. Ảnh 725 chứa một hoặc nhiều lát 727. Lát 727 có thể được định rõ là số nguyên của các miếng hoàn chỉnh hoặc số nguyên của các hàng bộ phận cây lập mã (CTU) hoàn chỉnh liên tiếp (ví dụ, nằm trong miếng) của ảnh 725 mà được chứa duy nhất trong bộ phận NAL đơn. Các lát 727 còn được chia thành các CTU và/hoặc các khối cây lập mã (CTB). CTU là nhóm của các mẫu có kích thước được định rõ trước mà có thể được phân chia bởi cây lập mã. CTB là tập hợp con của CTU và chứa các thành phần độ chói hoặc các thành phần sắc độ của CTU. Các CTU /các CTB còn được chia thành các khối lập mã dựa vào các cây lập mã. Các khối lập mã sau đó có thể được mã hóa/được giải mã theo các cơ chế dự đoán.

Dòng bit 700 có thể được lập mã như chuỗi của các bộ phận NAL. Bộ phận NAL là bộ chứa cho dữ liệu video và/hoặc cú pháp hỗ trợ. Bộ phận NAL có thể là bộ phận VCL NAL hoặc bộ phận không phải VCL NAL. Bộ phận VCL NAL là bộ phận NAL được lập mã để chứa dữ liệu video, chẳng hạn như dữ liệu hình ảnh 720 và đoạn đầu lát được kết hợp 717. Bộ phận không phải VCL NAL là bộ phận NAL mà chứa không phải dữ liệu video chẳng hạn như cú pháp và/hoặc các thông số mà hỗ trợ việc giải mã dữ liệu video, hiệu suất của việc kiểm tra sự phù hợp, hoặc các thao tác khác. Ví dụ, bộ phận không phải VCL NAL có thể chứa VPS 711, SPS 713, PPS 715, tin nhắn SEI 719, hoặc cú pháp hỗ trợ khác.

Tin nhắn SEI 719 là cấu trúc cú pháp với ngữ nghĩa được quy định mà bao gồm thông tin mà không được cần bởi quy trình giải mã để xác định các trị số của các mẫu ở các ảnh được giải mã. Ví dụ, các tin nhắn SEI 719 có thể chứa dữ liệu để hỗ trợ các quy trình xử lý HRD hoặc dữ liệu hỗ trợ khác mà không liên quan trực tiếp đến việc giải mã dòng bit 700 ở bộ giải mã. Tập hợp của các tin nhắn SEI 719 có thể được thực hiện như tin nhắn SEI lồng ghép có thể mở rộng 741. Tin nhắn SEI lồng ghép có thể mở rộng 741 cung cấp cơ chế để kết hợp các tin nhắn SEI 719 với các lớp 723 cụ thể. Tin nhắn SEI lồng ghép có thể mở rộng 741 là tin nhắn mà chứa nhiều tin nhắn SEI được lồng ghép có thể mở rộng 742. Tin nhắn

SEI được lồng ghép có thể mở rộng 742 là tin nhắn SEI 719 mà tương ứng với một hoặc nhiều OLS hoặc một hoặc nhiều lớp 723. OLS là tập hợp của các lớp 723 trong đó ít nhất một trong số các lớp 723 là lớp đầu ra. Theo đó, tin nhắn SEI lồng ghép có thể mở rộng 741 có thể được cho là để bao gồm tập hợp của các tin nhắn SEI được lồng ghép có thể mở rộng 742 hoặc được cho là để bao gồm tập hợp của các tin nhắn SEI 719, tùy thuộc vào ngữ cảnh. Hơn nữa, tin nhắn SEI lồng ghép có thể mở rộng 741 chứa tập hợp của các tin nhắn SEI được lồng ghép có thể mở rộng 742 của cùng loại. Các tin nhắn SEI 719 có thể bao gồm tin nhắn BP SEI mà chứa các thông số HRD để khởi tạo HRD để quản lý CPB để thử nghiệm các OLS tương ứng và/hoặc các lớp 723. Các tin nhắn SEI 719 cũng có thể bao gồm tin nhắn PT SEI mà chứa các thông số HRD để quản lý thông tin phân phối cho các AU ở CPB và/hoặc DPB để thử nghiệm các OLS tương ứng và/hoặc các lớp 723. Các tin nhắn SEI 719 cũng có thể bao gồm tin nhắn DUI SEI mà chứa các thông số HRD để quản lý thông tin phân phối cho các DU ở CPB và/hoặc DPB để thử nghiệm các OLS tương ứng và/hoặc các lớp 723.

Như được lưu ý ở trên, tin nhắn SEI 719 có thể được chứa trong bộ phận không phải VCL NAL. Ví dụ, tin nhắn SEI lồng ghép có thể mở rộng 741 có thể được chứa trong bộ phận SEI NAL 743. Bộ phận SEI NAL 743 là bộ phận NAL với dữ liệu loại chỉ báo rằng bộ phận SEI NAL 743 chứa dữ liệu SEI. Một trong số các lợi ích của tin nhắn SEI lồng ghép có thể mở rộng 741 là tin nhắn SEI lồng ghép có thể mở rộng 741 cho phép tập hợp của các tin nhắn SEI được lồng ghép có thể mở rộng 742 được chứa trong bộ phận SEI NAL 743 đơn. Khi các tin nhắn SEI lồng ghép có thể mở rộng 741 không được ứng dụng, mỗi tin nhắn SEI 719 được chứa trong bộ phận SEI NAL 743 riêng biệt. Nhằm rõ ràng cho việc thảo luận, bộ phận SEI NAL 743 cụ thể có thể được đề cập đến là bộ phận SEI NAL 743 hiện thời.

Một trong số những vấn đề với các tin nhắn SEI lồng ghép có thể mở rộng 741 là toàn bộ tập hợp của các tin nhắn SEI được lồng ghép có thể mở rộng 742 được chứa trong bộ phận SEI NAL 743 đơn. Tuy nhiên, tập hợp của các tin nhắn SEI được lồng ghép có thể mở rộng 742 có thể liên quan đến các lớp 723 tương ứng theo nhiều cách. Ví dụ, mỗi trong số các tin nhắn SEI được lồng ghép có thể mở rộng 742 trong tập hợp có thể liên quan đến tất cả các lớp 723. Theo ví dụ khác, mỗi trong số các tin nhắn SEI được lồng ghép có thể mở rộng 742 trong tập hợp có thể liên quan đến một hoặc nhiều trong số các lớp 723. Các cờ khác nhau có thể được ứng dụng để chỉ báo sự tương quan giữa các tin nhắn SEI được lồng ghép có thể mở rộng 742 và các lớp 723. Như được lưu ý ở trên, một vấn đề với các cờ tin nhắn SEI được lồng ghép có thể mở rộng 742 là các cờ chỉ báo các tin nhắn SEI được lồng

ghép có thể mở rộng 742 áp dụng cho tất cả các lớp trong AU có thể gây ra các lỗi khi chuỗi video bao gồm các AU không hoàn chỉnh. Vấn đề khác là bộ phận SEI NAL 743 mà chứa tin nhắn SEI lồng ghép có thể mở rộng 741 luôn được kết hợp với lớp 723, ví dụ lớp 723 với `nuh_layer_id` 729 thấp nhất. Khi tin nhắn SEI được lồng ghép có thể mở rộng 742 áp dụng cho lớp 723 được kết hợp với bộ phận SEI NAL 743, sự kết hợp được báo hiệu trong cả bộ phận SEI NAL 743 và trong tin nhắn SEI lồng ghép có thể mở rộng 741. Điều này là dư thừa và làm lãng phí các bit.

Dòng bit 700 bao gồm các cờ khác nhau để báo hiệu cấu hình của các tin nhắn SEI 719 trong khi xử lý các vấn đề được xử lý ở trên. Ví dụ, tin nhắn SEI lồng ghép có thể mở rộng 741 có thể bao gồm cờ tất cả các lớp lồng ghép có thể mở rộng (`all_layers_flag`) 735. Thành phần cú pháp `all_layers_flag` lồng ghép có thể mở rộng 735 xác định xem các tin nhắn SEI được lồng ghép có thể mở rộng 742 trong tin nhắn SEI lồng ghép có thể mở rộng 741 có áp dụng cho tất cả các lớp 723 hay không, ví dụ như được quy định trong VPS 711. Bằng cách ứng dụng `all_layers_flag` lồng ghép có thể mở rộng 735 mà không xử lý các AU, vấn đề AU không hoàn chỉnh được tránh khỏi. Theo đó, khi `all_layers_flag` lồng ghép có thể mở rộng 735 được thiết đặt, tin nhắn SEI được lồng ghép có thể mở rộng 742 được chứa trong tin nhắn SEI lồng ghép có thể mở rộng 741 áp dụng cho tất cả các lớp 723 không quan tâm đến số các lớp 723 mà được bao gồm trong AU trong đó tin nhắn SEI lồng ghép có thể mở rộng 741 được áp dụng. Theo ví dụ cụ thể, `all_layers_flag` lồng ghép có thể mở rộng 735 được thiết đặt bằng một khi xác định rằng các tin nhắn SEI được lồng ghép có thể mở rộng 742 áp dụng cho tất cả các lớp 723 mà có `nuh_layer_id` 729 lớn hơn hoặc bằng `nuh_layer_id` 729 của bộ phận SEI NAL 743 hiện thời. Hơn nữa, `all_layers_flag` lồng ghép có thể mở rộng 735 có thể được thiết đặt bằng không khi xác định rằng các tin nhắn SEI được lồng ghép có thể mở rộng 742 có thể áp dụng hoặc có thể không áp dụng cho tất cả các lớp 723 mà có `nuh_layer_id` 729 lớn hơn hoặc bằng `nuh_layer_id` 729 của bộ phận SEI NAL 743 hiện thời. Do đó, `all_layers_flag` lồng ghép có thể mở rộng 735 làm tăng tính năng của bộ mã hóa và/hoặc bộ giải mã bằng cách khiến dòng bit 700 được kiểm tra một cách chính xác về sự phù hợp khi các AU không hoàn chỉnh có mặt. Hơn nữa, `all_layers_flag` lồng ghép có thể mở rộng 735 làm giảm kích thước của dòng bit được mã hóa 700 bằng cách bỏ qua báo hiệu lớp rõ ràng cho các tin nhắn SEI được lồng ghép có thể mở rộng 742 khi tin nhắn như vậy áp dụng cho tất cả các lớp 723. Kết quả là, hiệu quả lập mã được tăng lên, mà làm giảm việc

sử dụng tài nguyên của bộ xử lý, bộ nhớ, và/hoặc báo hiệu mạng ở cả bộ mã hóa và bộ giải mã.

Để xử lý vấn đề báo hiệu phân dư, tin nhắn SEI lồng ghép có thể mở rộng 741 có thể bao gồm Id lớp lồng ghép có thể mở rộng (`layer_id[i]`) 739. `Layer_id[i]` lồng ghép có thể mở rộng 739 là thành phần cú pháp mà xác định trị số `nuh_layer_id` 729 của lớp thứ *i* trong đó các tin nhắn SEI được lồng ghép có thể mở rộng 742 áp dụng. `Layer_id[i]` lồng ghép có thể mở rộng 739 có thể được ứng dụng khi `all_layers_flag` lồng ghép có thể mở rộng 735 được thiết đặt tới không. Như vậy, `layer_id[i]` 739 có thể được sử dụng để tương quan mỗi trong số các tin nhắn SEI được lồng ghép có thể mở rộng 742 với các lớp 723 tương ứng. Hơn nữa, `layer_id[i]` lồng ghép có thể mở rộng 739 được tạo cấu hình để báo hiệu `nuh_layer_id` 729 của tất cả các lớp 723 mà tương ứng với các tin nhắn SEI được lồng ghép có thể mở rộng 742 ngoại trừ `nuh_layer_id` 729 của lớp 723 mà tương ứng với bộ phận SEI NAL 743. Khi tin nhắn SEI được lồng ghép có thể mở rộng 742 không được kết hợp với trị số `nuh_layer_id` 729 trong `layer_id[i]` lồng ghép có thể mở rộng 739, tin nhắn SEI được lồng ghép có thể mở rộng 742 có thể được suy luận liên quan đến lớp 723 được kết hợp với bộ phận SEI NAL 743. Ví dụ, bộ phận SEI NAL 743 có thể được kết hợp với trị số thấp nhất của `nuh_layer_id` 729. Do đó, `layer_id[i]` lồng ghép có thể mở rộng 739 có thể bị ràng buộc sao cho trị số của `layer_id[i]` lồng ghép có thể mở rộng 739 cho mỗi lớp *i* phải lớn hơn `nuh_layer_id` 729 của bộ phận SEI NAL 743 hiện thời. Điều này đảm bảo rằng `nuh_layer_id` 729 của bộ phận SEI NAL 743 hiện thời được báo hiệu ở bộ phận SEI NAL 743 hiện thời, nhưng không được lặp lại trong `layer_id[i]` lồng ghép có thể mở rộng 739 trong tin nhắn SEI lồng ghép có thể mở rộng 741. Điều này nâng cao tin nhắn SEI lồng ghép có thể mở rộng 741 bằng cách bỏ qua Id phân dư. Ví dụ, vòng lặp mà mã hóa/giải mã `layer_id[i]` lồng ghép có thể mở rộng 739 thực thi ít hơn một lần, mà làm giảm các tài nguyên bộ xử lý trong suốt thời gian mã hóa và/hoặc giải mã. Hơn nữa, điều này làm giảm kích thước của dòng bit được mã hóa 700 cho mỗi trong số các tin nhắn SEI lồng ghép có thể mở rộng 741 ở dòng bit 700. Kết quả là, hiệu quả lập mã được tăng lên, mà làm giảm việc sử dụng tài nguyên của bộ xử lý, bộ nhớ, và/hoặc báo hiệu mạng ở cả bộ mã hóa và bộ giải mã.

HRD và/hoặc bộ giải mã có thể sử dụng `layer_id[i]` lồng ghép có thể mở rộng 739 và `all_layers_flag` lồng ghép có thể mở rộng 735 để tương quan các tin nhắn SEI được lồng ghép có thể mở rộng 742 với các lớp 723, ví dụ dựa vào các thành phần cú pháp khác ở dòng bit 700. Ví dụ, HRD và/hoặc bộ giải mã có thể sử dụng biến số lượng lớp lồng ghép

(nestingNumLayers) và danh mục của biến các Id lớp lồng ghép (NestingLayerId[i]). Với vấn đề thuật ngữ, các thành phần cú pháp với các dấu gạch dưới có thể được báo hiệu ở dòng bit 700 bởi bộ mã hóa, trong khi các thành phần cú pháp mà không có các dấu gạch dưới hoặc các khoảng trống có thể là các biến mà được xác định khi đọc dữ liệu từ dòng bit 700 (ví dụ, ở HRD và/hoặc bộ giải mã). Biến nestingNumLayers xác định số lượng lớp 723 trong đó các tin nhắn SEI được lồng ghép có thể mở rộng 742 áp dụng dựa vào dữ liệu ở dòng bit 700. Biến NestingLayerId[i] là biến mà được thiết đặt để xác định danh mục của các trị số nuh_layer_id của các lớp 723 trong đó các tin nhắn SEI được lồng ghép có thể mở rộng 742 áp dụng cho mỗi lớp i nằm trong đoạn từ không đến nestingNumLayers trừ một, dựa vào dữ liệu ở dòng bit 700. NestingNumLayers và NestingLayerId[i] có thể được xác định như sau:

```
if( scalable nesting all_layers_flag ) {
```

```
    nestingNumLayers = vps_max_layers_minus1 + 1 - GeneralLayerIdx[ nuh_layer_id ]
    for( i = 0; i < nestingNumLayers; i ++)
```

```
        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
```

```
} else {
```

```
    nestingNumLayers = scalable nesting num_layers_minus1 + 1
```

```
    for( i = 0; i < nestingNumLayers; i ++)
```

```
        NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
```

```
}
```

Mã nêu trên dựa vào các lớp lớn nhất VPS trừ một (vps_max_layers_minus1) 731. Vps_max_layers_minus1 731 là thành phần cú pháp được mang trong VPS 711 mà xác định số lượng lớp 723 ở dòng bit 700. Mã nêu trên cũng dựa vào Id lớp VPS (vps_layer_id[i]) 733. Vps_layer_id[i] 733 là thành phần cú pháp được mang trong VPS 733 mà xác định trị số nuh_layer_id 729 của lớp thứ i. Do đó, vps_layer_id[i] 733 xác định nuh_layer_id 729 cho mọi lớp 723 sao cho nuh_layer_ids 729 có thể được truy cập bởi chỉ số lớp i.

Mã nêu trên cũng dựa vào biến chỉ số lớp chung (GeneralLayerIdx). GeneralLayerIdx là biến được xác định bởi HRD và/hoặc bộ giải mã mà xác định chỉ số lớp của lớp 723 với nuh_layer_id bằng vps_layer_id[i] 733. Mã nêu trên cũng dựa vào thành phần cú pháp số các lớp trừ một lồng ghép có thể mở rộng (num_layers_minus1) 737. Num_layers_minus1 lồng ghép có thể mở rộng 737 là thành phần cú pháp được mang trong tin nhắn SEI lồng ghép có thể mở rộng 741 mà xác định số các lớp 723 trong đó các tin nhắn SEI được lồng ghép có thể mở rộng 742 trong tin nhắn SEI lồng ghép có thể mở rộng 741 áp dụng.

Num_layers_minus1 lồng ghép có thể mở rộng 737 ứng dụng định dạng trừ một và do đó bao gồm ít hơn một trị số thực tế. Ví dụ, nếu tin nhắn SEI lồng ghép có thể mở rộng 741 bao gồm các tin nhắn SEI được lồng ghép có thể mở rộng 742 mà liên quan đến năm lớp 723, sau đó num_layers_minus1 lồng ghép có thể mở rộng 737 được thiết đặt tới trị số bốn. Theo một số ví dụ, num_layers_minus1 lồng ghép có thể mở rộng 737 bị ràng buộc nằm trong đoạn từ không đến vps_max_layers_minus1 731 trừ GeneralLayerIdx[nuh_layer_id], trong đó nuh_layer_id 729 là nuh_layer_id 729 của bộ phận SEI NAL 743 hiện thời.

Thông tin nêu trên bây giờ được mô tả chi tiết hơn ở đây dưới đây. Việc lập mã video được phân lớp cũng được đề cập đến là việc lập mã video có thể mở rộng hoặc việc lập mã video với khả năng mở rộng. Khả năng mở rộng trong việc lập mã video có thể được hỗ trợ nhờ sử dụng các kỹ thuật lập mã đa lớp. Dòng bit đa lớp bao gồm lớp cơ sở (base layer, viết tắt là BL) và một hoặc nhiều lớp nâng cao (enhancement layer, viết tắt là EL). Ví dụ về các khả năng mở rộng bao gồm khả năng mở rộng theo không gian, khả năng mở rộng chất lượng/tỷ lệ tín hiệu trên nhiễu (signal noise ratio, viết tắt là SNR), khả năng mở rộng đa điểm, khả năng mở rộng tỷ lệ khung, v.v. Khi kỹ thuật lập mã đa lớp được sử dụng, ảnh hoặc phần của chúng có thể được lập mã mà không sử dụng ảnh tham chiếu (dự đoán bên trong), có thể được lập mã bằng cách tham chiếu các ảnh tham chiếu mà ở cùng lớp (dự đoán liên kết), và/hoặc có thể được lập mã bằng cách tham chiếu các ảnh tham chiếu mà ở (các) lớp khác (dự đoán liên lớp). Ảnh tham chiếu được sử dụng cho việc dự đoán liên lớp của ảnh hiện thời được đề cập đến là ảnh tham chiếu liên lớp (inter-layer reference picture, viết tắt là ILRP). Fig.6 minh họa ví dụ về việc lập mã đa lớp cho khả năng mở rộng theo không gian trong đó các ảnh ở các lớp khác nhau có các độ phân giải khác nhau.

Một số gia đình lập mã video cung cấp sự hỗ trợ cho khả năng mở rộng trong (các) hồ sơ được tách riêng từ (các) hồ sơ dùng cho việc lập mã lớp đơn. Lập mã video có thể mở rộng (SVC) là việc mở rộng có thể mở rộng của việc lập mã video nâng cao (AVC) mà cung cấp sự hỗ trợ cho các khả năng mở rộng theo không gian, theo thời gian, và chất lượng. Đối với SVC, cờ được báo hiệu trong mỗi khối macro (macroblock, viết tắt là MB) trong các ảnh EL để chỉ báo xem EL MB có được dự đoán nhờ sử dụng khối được sắp xếp vào một chỗ từ lớp thấp hơn hay không. Việc dự đoán từ khối được sắp xếp vào một chỗ có thể bao gồm kết cấu, các vector chuyển động, và/hoặc các chế độ lập mã. Các cách thực hiện của SVC có thể không sử dụng lại trực tiếp các cách thực hiện AVC không được điều chỉnh theo thiết kế của

chúng. Cú pháp khối macro SVC EL và quy trình giải mã khác với quy trình giải mã và cú pháp AVC.

HEVC có thể mở rộng (scalable HEVC, viết tắt là SHVC) là sự mở rộng của HEVC mà cung cấp sự hỗ trợ cho các khả năng mở rộng theo không gian và chất lượng. HEVC đa điểm (MV-HEVC) là sự mở rộng của HEVC mà cung cấp sự hỗ trợ cho khả năng mở rộng đa điểm. 3D HEVC (3D-HEVC) là sự mở rộng của HEVC mà cung cấp sự hỗ trợ cho việc lập mã video 3D mà được nâng cao hơn và hữu hiệu hơn so với MV-HEVC. Khả năng mở rộng theo thời gian có thể được bao gồm như phần nguyên của bộ mã hóa-giải mã HEVC lớp đơn. Theo sự mở rộng đa lớp của HEVC, các ảnh được giải mã được sử dụng cho việc dự đoán liên lớp đến chỉ từ cùng AU và được xử lý như các ảnh tham chiếu dài hạn (long-term reference picture, viết tắt là LTRP). Các ảnh như vậy được gán các chỉ số tham chiếu trong (các) danh mục ảnh tham chiếu cùng với các ảnh tham chiếu theo thời gian khác ở lớp hiện thời. Việc dự đoán liên lớp (ILP) đạt được ở mức bộ phận dự đoán (PU) bằng cách thiết đặt trị số của chỉ số tham chiếu đề cập đến (các) ảnh tham chiếu liên lớp trong (các) danh mục ảnh tham chiếu. Khả năng mở rộng theo không gian lấy mẫu lại ảnh tham chiếu hoặc phần của chúng khi ILRP có độ phân giải theo không gian khác so với ảnh hiện thời được mã hóa hoặc được giải mã. Việc lấy mẫu lại ảnh tham chiếu có thể được thực hiện ở hoặc mức ảnh hoặc mức khối lập mã.

VVC cũng có thể hỗ trợ việc lập mã video được phân lớp. Dòng bit VVC có thể bao gồm nhiều lớp. Các lớp đều có thể độc lập với nhau. Ví dụ, mỗi lớp có thể được lập mã mà không sử dụng việc dự đoán liên lớp. Trong trường hợp này, các lớp cũng được đề cập đến là các lớp đồng phát sóng. Trong một số trường hợp, một vài trong số các lớp được lập mã nhờ sử dụng ILP. Cờ trong VPS có thể chỉ báo xem các lớp là các lớp đồng phát sóng hay không hoặc xem số lượng lớp sử dụng ILP hay không. Khi số lượng lớp sử dụng ILP, mối tương quan phụ thuộc lớp trong số các lớp cũng được báo hiệu trong VPS. Không giống SHVC và MV-HEVC, VVC có thể không xác định các OLS. OLS bao gồm tập hợp của các lớp được quy định, trong đó một hoặc nhiều lớp trong tập hợp của các lớp được quy định là các lớp đầu ra. Lớp đầu ra là lớp của OLS mà được đưa ra. Theo một số cách thực hiện của VVC, chỉ một lớp có thể được lựa chọn để giải mã và được đưa ra khi các lớp là các lớp đồng phát sóng. Theo một số cách thực hiện của VVC, toàn bộ dòng bit bao gồm tất cả các lớp được quy định được giải mã khi lớp bất kỳ sử dụng ILP. Hơn nữa, các lớp nhất định trong số các lớp được quy định là các lớp đầu ra. Các lớp đầu ra có thể được chỉ báo chỉ là

lớp cao nhất, tất cả các lớp, hoặc lớp cao nhất cộng tập hợp của các lớp thấp hơn được chỉ báo.

Các khía cạnh nêu trên chứa các vấn đề nhất định. Một số hệ thống lập mã video xác định rằng các tin nhắn SEI lồng ghép có thể mở rộng để kết hợp các tin nhắn SEI với các tập hợp con dòng bit tương ứng với các điểm thao tác khác nhau, với các lớp cụ thể, và/hoặc các lớp con. Vấn đề thứ nhất xảy ra khi các tin nhắn SEI lồng ghép cho các lớp cụ thể nhờ sử dụng tin nhắn SEI lồng ghép có thể mở rộng được chứa trong bộ phận SEI NAL với trị số cụ thể `nuh_layer_id` của `nuhLayerId`. Trong trường hợp này, lớp có thể áp dụng với `nuh_layer_id` bằng `nuhLayerId` cũng được báo hiệu một cách rõ ràng khi các tin nhắn SEI được lồng ghép không áp dụng cho tất cả các lớp. Tuy nhiên, lớp có thể áp dụng với `nuh_layer_id` bằng `nuhLayerId` luôn là lớp có thể áp dụng. Vì vậy việc báo hiệu của `nuh_layer_id` của bộ phận SEI NAL là không cần thiết (ví dụ, phần dư) và là sự lãng phí của các bit.

Vấn đề thứ hai xảy ra khi các tin nhắn SEI được lồng ghép áp dụng cho tất cả các lớp. Ngữ nghĩa có thể cho biết rằng `all_layers_flag` được thiết đặt bằng một để xác định rằng danh mục `NestingLayerIdList[ 0 ]` bao gồm tất cả các trị số của `nuh_layer_id` có mặt trong bộ phận truy cập hiện thời mà lớn hơn hoặc bằng `nuh_layer_id` của bộ phận SEI NAL hiện thời, theo thứ tự tăng lên của các trị số. Tuy nhiên, AU hiện thời có thể là AU không hoàn chỉnh. Điều này có thể xảy ra khi AU không có các ảnh có mặt cho tất cả các lớp. Phạm vi ổn định của tin nhắn SEI được lồng ghép có thể bao gồm các AU khác mà chứa các ảnh cho nhiều lớp hơn so với AU hiện thời. Do đó, các lớp mà đang thiếu trong AU hiện thời (AU chứa tin nhắn SEI lồng ghép có thể mở rộng) không được quy định có thể áp dụng. Khi một số AU khác trong phạm vi ổn định của tin nhắn SEI được lồng ghép có các ảnh của các lớp mà đang thiếu trong AU hiện thời, thông tin được mang trong tin nhắn SEI được lồng ghép được diễn dịch là không thể áp dụng cho các ảnh này. Do đó, các lỗi có thể xảy ra. Ví dụ, nếu khung đóng gói tin nhắn SEI được bao gồm trong tin nhắn SEI được lồng ghép, thì các ảnh được kết xuất có thể bao gồm các tạo tác dẫn đến trải nghiệm người dùng kém.

Nói chung, sáng chế này mô tả các cách tiếp cận cho việc lồng ghép có thể mở rộng của các tin nhắn SEI cho các lớp ở các dòng bit video đa lớp. Các phần mô tả về các kỹ thuật dựa vào VVC. Tuy nhiên, các kỹ thuật cũng áp dụng cho việc lập mã video được phân lớp dựa vào các thông số kỹ thuật mã hóa-giải mã video khác.

Một hoặc nhiều trong số các vấn đề nêu trên có thể được giải quyết như sau. Cụ thể là, sáng chế này bao gồm các phương pháp cho các tin nhắn SEI lồng ghép một cách hữu hiệu để sử dụng với các lớp cùng với các AU không hoàn chỉnh như được nêu trên. Thứ nhất, lớp có Id lớp bằng Id lớp của bộ phận SEI NAL chứa tin nhắn SEI lồng ghép có thể mở rộng được suy luận là lớp trong đó các tin nhắn SEI được lồng ghép áp dụng thay vì được báo hiệu một cách rõ ràng. Thứ hai, khi các tin nhắn SEI được lồng ghép áp dụng cho tất cả các lớp, các tin nhắn SEI được lồng ghép áp dụng cho tất cả các lớp được quy định bởi VPS, mà là tất cả các lớp mà có thể có mặt ở dòng bit, thay vì tất cả các lớp có mặt trong AU hiện thời.

Cách thực hiện ví dụ của các cơ chế nêu trên như sau. Cú pháp tin nhắn SEI lồng ghép có thể mở rộng ví dụ như sau.

scalable_nesting(payloadSize) {	Ký hiệu mô tả
nesting_ols_flag	u(1)
if(nesting_ols_flag) {	
nesting_num_olss_minus1	ue(v)
for(i = 0; i <= nesting_num_olss_minus1; i++) {	
nesting_ols_idx_delta_minus1[i]	ue(v)
if(NumLayersInOls[NestingOlsIdx[i]] > 1) {	
nesting_num_ols_layers_minus1[i]	ue(v)
for(j = 0; j <= nesting_num_ols_layers_minus1[i]; j++)	
nesting_ols_layer_idx_delta_minus1[i][j]	ue(v)
}	
}	
} else {	
nesting_all_layers_flag	u(1)
if(!nesting_all_layers_flag) {	
nesting_num_layers_minus1	ue(v)
for(i = 1; i <= nesting_num_layers_minus1; i++)	

nesting_layer_id[i]	u(6)
}	
}	
nesting_num_seis_minus1	ue(v)
while(!byte_aligned())	
nesting_not_bit /* bằng 0 */	u(1)
for(i = 0; i <= nesting_num_seis_minus1; i++)	
sei_message()	
}	

Theo ví dụ thay thế, thành phần cú pháp nesting_layer_id[i] là việc lập mã ue(v) của chỉ số lớp trong số các lớp được quy định bởi VPS. Theo ví dụ thay thế khác, thành phần cú pháp nesting_layer_id[i] là việc lập mã ue(v) của delta của chỉ số lớp trong số các lớp được quy định bởi VPS. Theo ví dụ thay thế khác, thành phần cú pháp nesting_layer_id[i] là việc lập mã ue(v) của delta của nuh_layer_id.

Ngữ nghĩa tin nhắn SEI lồng ghép có thể mở rộng ví dụ như sau. Tin nhắn SEI lồng ghép có thể mở rộng cung cấp cơ chế để kết hợp các tin nhắn SEI với các lớp cụ thể theo ngữ cảnh của OLS cụ thể hoặc với các lớp cụ thể không theo ngữ cảnh của OLS. Tin nhắn SEI lồng ghép có thể mở rộng chứa một hoặc nhiều tin nhắn SEI. Các tin nhắn SEI được chứa trong tin nhắn SEI lồng ghép có thể mở rộng cũng được đề cập đến là các tin nhắn SEI được lồng ghép có thể mở rộng. Sự phù hợp dòng bit có thể yêu cầu rằng các hạn chế sau đây áp dụng khi các tin nhắn SEI được chứa trong tin nhắn SEI lồng ghép có thể mở rộng.

Tin nhắn SEI mà có payloadType bằng một trăm ba mươi hai (bấm ảnh được giải mã) hoặc một trăm ba mươi ba (lồng ghép có thể mở rộng) sẽ không được chứa trong tin nhắn SEI lồng ghép có thể mở rộng. Khi tin nhắn SEI lồng ghép có thể mở rộng chứa khoảng đệm, thời điểm ảnh, hoặc tin nhắn SEI thông tin bộ phận giải mã, tin nhắn SEI lồng ghép có thể mở rộng sẽ không chứa tin nhắn SEI khác bất kỳ với payloadType không bằng không (khoảng đệm), một (thời điểm ảnh), hoặc một trăm ba mươi (thông tin bộ phận giải mã).

Sự phù hợp dòng bit cũng có thể yêu cầu rằng các hạn chế sau đây áp dụng trên trị số của nal_unit_type của bộ phận SEI NAL chứa tin nhắn SEI lồng ghép có thể mở rộng. Khi tin nhắn SEI lồng ghép có thể mở rộng chứa tin nhắn SEI mà có payloadType bằng không

(khoảng đệm), một (thời điểm ảnh), một trăm ba mươi (thông tin bộ phận giải mã), một trăm bốn mươi lăm (sự chỉ báo RAP phụ thuộc), hoặc một trăm sáu mươi tám (thông tin trường khung), bộ phận SEI NAL chứa tin nhắn SEI lồng ghép có thể mở rộng sẽ có nal_unit_type được thiết đặt bằng PREFIX_SEI_NUT. Khi tin nhắn SEI lồng ghép có thể mở rộng chứa tin nhắn SEI mà có payloadType bằng một trăm ba mươi hai (bấm ảnh được giải mã), bộ phận SEI NAL chứa tin nhắn SEI lồng ghép có thể mở rộng sẽ có nal_unit_type được thiết đặt bằng SUFFIX_SEI_NUT.

Nesting_ols_flag có thể được thiết đặt bằng một để xác định rằng các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng cho các lớp cụ thể theo ngữ cảnh của OLS cụ thể. Nesting_ols_flag có thể được thiết đặt bằng không để xác định rằng các tin nhắn SEI được lồng ghép có thể mở rộng nói chung áp dụng (ví dụ, không theo ngữ cảnh của OLS) cho các lớp cụ thể.

Sự phù hợp dòng bit có thể yêu cầu rằng các hạn chế sau đây được áp dụng cho trị số của nesting_ols_flag. Khi tin nhắn SEI lồng ghép có thể mở rộng chứa tin nhắn SEI mà có payloadType bằng không (khoảng đệm), một (thời điểm ảnh), hoặc một trăm ba mươi (thông tin bộ phận giải mã), trị số của nesting_ols_flag sẽ bằng một. Khi tin nhắn SEI lồng ghép có thể mở rộng chứa tin nhắn SEI mà có payloadType bằng trị số trong VclAssociatedSeiList, trị số của nesting_ols_flag sẽ bằng không.

Nesting_num_olss_minus1 cộng một xác định số các OLS trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. Trị số của nesting_num_olss_minus1 sẽ nằm trong đoạn từ không đến TotalNumOlss - 1. Nesting_ols_idx_delta_minus1[i] được sử dụng để dẫn xuất biến NestingOlsIdx[i] mà xác định chỉ số OLS của OLS thứ i trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng khi nesting_ols_flag bằng một. Trị số của nesting_ols_idx_delta_minus1[i] sẽ nằm trong đoạn từ không đến TotalNumOlss - 2. Biến NestingOlsIdx[i] có thể được dẫn xuất như sau:

```
if( i == 0 )
```

```
    NestingOlsIdx[ i ] = nesting_ols_idx_delta_minus1[ i ]
```

```
else
```

```
    NestingOlsIdx[ i ] = NestingOlsIdx[ i - 1 ] + nesting_ols_idx_delta_minus1[ i ] +
```

```
1
```

Nesting_num_ols_layers_minus1[i] cộng một xác định số các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng theo ngữ cảnh của OLS thứ NestingOlsIdx[

i]. Trị số của $\text{nesting_num_ols_layers_minus1}[i]$ sẽ nằm trong đoạn từ không đến $\text{NumLayersInOls}[\text{NestingOlsIdx}[i]] - 1$.

$\text{Nesting_ols_layer_idx_delta_minus1}[i][j]$ được sử dụng để dẫn xuất biến $\text{NestingOlsLayerIdx}[i][j]$ mà xác định chỉ số lớp OLS của lớp thứ j trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng theo ngữ cảnh của OLS thứ $\text{NestingOlsIdx}[i]$ khi nesting_ols_flag bằng một. Trị số của $\text{nesting_ols_layer_idx_delta_minus1}[i]$ sẽ nằm trong đoạn từ không đến $\text{NumLayersInOls}[\text{nestingOlsIdx}[i]] - \text{hai}$.

Biến $\text{NestingOlsLayerIdx}[i][j]$ có thể được dẫn xuất như sau:

if($j = 0$)

$\text{NestingOlsLayerIdx}[i][j] = \text{nesting_ols_layer_idx_delta_minus1}[i][j]$

else

$\text{NestingOlsLayerIdx}[i][j] = \text{NestingOlsLayerIdx}[i][j - 1] +$

$\text{nesting_ols_layer_idx_delta_minus1}[i][j] + 1$

Trị số thấp nhất trong số tất cả các trị số của $\text{LayerIdxInOls}[\text{NestingOlsIdx}[i]][\text{NestingOlsLayerIdx}[i][0]]$ đối với i nằm trong đoạn từ không đến $\text{nesting_num_olss_minus1}$, sẽ bằng nuh_layer_id của bộ phận SEI NAL hiện thời (ví dụ, bộ phận SEI NAL chứa tin nhắn SEI lồng ghép có thể mở rộng). $\text{Nesting_all_layers_flag}$ có thể được thiết đặt bằng một để xác định rằng các tin nhắn SEI được lồng ghép có thể mở rộng nói chung áp dụng cho tất cả các lớp mà có nuh_layer_id lớn hơn hoặc bằng nuh_layer_id của bộ phận SEI NAL hiện thời. $\text{Nesting_all_layers_flag}$ có thể được thiết đặt bằng không để xác định rằng các tin nhắn SEI được lồng ghép có thể mở rộng nói chung có thể áp dụng hoặc có thể không áp dụng cho tất cả các lớp mà có nuh_layer_id lớn hơn hoặc bằng nuh_layer_id của bộ phận SEI NAL hiện thời.

$\text{Nesting_num_layers_minus1}$ cộng một xác định số các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng nói chung áp dụng. Trị số của $\text{nesting_num_layers_minus1}$ sẽ nằm trong đoạn từ không đến $\text{vps_max_layers_minus1} - \text{GeneralLayerIdx}[\text{nuh_layer_id}]$, trong đó nuh_layer_id là nuh_layer_id của bộ phận SEI NAL hiện thời. $\text{Nesting_layer_id}[i]$ xác định trị số nuh_layer_id của lớp thứ i trong đó các tin nhắn SEI được lồng ghép có thể mở rộng nói chung áp dụng khi $\text{nesting_all_layers_flag}$ bằng không. Trị số của $\text{nesting_layer_id}[i]$ sẽ lớn hơn nuh_layer_id , trong đó nuh_layer_id là nuh_layer_id của bộ phận SEI NAL hiện thời.

Khi `nesting_ols_flag` bằng một, biến `NestingNumLayers`, xác định số lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng nói chung áp dụng, và danh mục `NestingLayerId[ i ]` đối với `i` nằm trong đoạn từ không đến `NestingNumLayers - 1`, xác định danh mục của trị số `nuh_layer_id` của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng nói chung áp dụng, được dẫn xuất như sau, trong đó `nuh_layer_id` là `nuh_layer_id` của bộ phận SEI NAL hiện thời:

```
if( nesting_all_layers_flag ) {
    NestingNumLayers =
vps_max_layers_minus1 + 1 - GeneralLayerIdx[ nuh_layer_id ]
    for( i = 0; i < NestingNumLayers; i ++ )
        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ] (D-2)
} else {
    NestingNumLayers = nesting_num_layers_minus1 + 1
    for( i = 0; i < NestingNumLayers; i ++ )
        NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : nesting_layer_id[ i ]
}
```

`Nesting_num_seis_minus1` cộng một xác định số các tin nhắn SEI được lồng ghép có thể mở rộng. Trị số của `nesting_num_seis_minus1` sẽ nằm trong đoạn từ không đến sáu mươi ba. `Nesting_not_bit` sẽ được thiết đặt bằng không.

Fig. 8 là hình vẽ giản lược của thiết bị lập mã video 800 ví dụ. Thiết bị lập mã video 800 thích hợp cho việc thực hiện các ví dụ/các phương án được bộc lộ như được mô tả ở đây. Thiết bị lập mã video 800 bao gồm các cổng xuôi dòng 820, các cổng ngược dòng 850, và/hoặc các bộ phận thu phát (Tx/Rx) 810, bao gồm các bộ truyền và/hoặc các bộ thu để truyền thông ngược dòng và/hoặc xuôi dòng dữ liệu qua mạng. Thiết bị lập mã video 800 cũng bao gồm bộ xử lý 830 bao gồm bộ phận logic và/hoặc bộ xử lý trung tâm (central processing unit, viết tắt là CPU) để xử lý dữ liệu và bộ nhớ 832 để lưu trữ dữ liệu. Thiết bị lập mã video 800 cũng có thể bao gồm các thành phần điện, quang-điện (optical-electrical, viết tắt là OE), các thành phần điện-quang (electrical-to-optical, viết tắt là EO), và/hoặc các thành phần truyền thông không dây được ghép nối với các cổng ngược dòng 850 và/hoặc các cổng xuôi dòng 820 cho việc truyền thông của dữ liệu qua các mạng truyền thông điện, quang, hoặc không dây. Thiết bị lập mã video 800 cũng có thể bao gồm các thiết bị đầu vào và/hoặc đầu ra (input and/or output, viết tắt là I/O) 860 để truyền thông dữ

liệu tới và từ người dùng. Các thiết bị I/O 860 có thể bao gồm các thiết bị đầu ra chẳng hạn như màn hình để hiển thị dữ liệu video, các loa để đưa ra dữ liệu audio, v.v. Các thiết bị I/O 860 cũng có thể bao gồm các thiết bị đầu vào, chẳng hạn như bàn phím, chuột, bi xoay, v.v., và/hoặc các giao diện tương ứng để tương tác với các thiết bị đầu ra như vậy.

Bộ xử lý 830 được thực hiện bởi phần cứng và phần mềm. Bộ xử lý 830 có thể được thực hiện như một hoặc nhiều chip CPU, các lõi (ví dụ, như bộ xử lý nhiều lõi), các mảng cổng lập trình được dạng trường (field-programmable gate array, viết tắt là FPGA), các mạch tích hợp chuyên dụng (application specific integrated circuit, viết tắt là ASIC), và các bộ xử lý tín hiệu kỹ thuật số (digital signal processor, viết tắt là DSP). Bộ xử lý 830 truyền thông với các cổng xuôi dòng 820, Tx/Rx 810, các cổng ngược dòng 850, và bộ nhớ 832. Bộ xử lý 830 bao gồm môđun lập mã 814. Môđun lập mã 814 thực hiện các phương án được bộc lộ được mô tả ở đây, chẳng hạn như các phương pháp 100, 900, và 1000, mà có thể ứng dụng chuỗi video đa lớp 600 và/hoặc dòng bit 700. Môđun lập mã 814 cũng có thể thực hiện phương pháp/cơ chế khác bất kỳ được mô tả ở đây. Hơn nữa, môđun lập mã 814 có thể thực hiện hệ thống mã hóa-giải mã 200, bộ mã hóa 300, bộ giải mã 400, và/hoặc HRD 500. Ví dụ, môđun lập mã 814 có thể được ứng dụng để thực hiện HRD. Hơn nữa, môđun lập mã 814 có thể được ứng dụng để mã hóa các tin nhắn SEI lồng ghép có thể mở rộng với các cờ tương ứng để hỗ trợ việc báo hiệu rõ ràng và ngắn gọn của sự tương quan giữa các tin nhắn SEI được lồng ghép có thể mở rộng trong các tin nhắn SEI lồng ghép có thể mở rộng và các lớp tương ứng. Theo đó, môđun lập mã 814 có thể được tạo cấu hình để thực hiện các cơ chế để xử lý một hoặc nhiều trong số các vấn đề được thảo luận ở trên. Do đó, môđun lập mã 814 khiến thiết bị lập mã video 800 cung cấp tính năng bổ sung và/hoặc hiệu quả lập mã khi lập mã dữ liệu video. Như vậy, môđun lập mã 814 nâng cao tính năng của thiết bị lập mã video 800 cũng như xử lý các vấn đề mà cụ thể với các lĩnh vực kỹ thuật lập mã video. Hơn nữa, môđun lập mã 814 tác động đến việc biến đổi của thiết bị lập mã video 800 tới trạng thái khác. Theo cách khác, môđun lập mã 814 có thể được thực hiện như các lệnh được lưu trữ ở bộ nhớ 832 và được thực thi bởi bộ xử lý 830 (ví dụ, như sản phẩm chương trình máy tính được lưu trữ trên phương tiện không tạm thời).

Bộ nhớ 832 bao gồm một hoặc nhiều loại bộ nhớ chẳng hạn như các đĩa, các ổ đĩa băng, các ổ đĩa trạng thái rắn, bộ nhớ chỉ đọc (read only memory, viết tắt là ROM), bộ nhớ truy cập ngẫu nhiên (random access memory, viết tắt là RAM), bộ nhớ tia chớp, bộ nhớ có thể lập địa chỉ nội dung bậc ba (ternary content-addressable memory, viết tắt là TCAM), bộ

nhớ truy cập ngẫu nhiên tĩnh (static random-access memory, viết tắt là SRAM), v.v. Bộ nhớ 832 có thể được sử dụng như thiết bị lưu trữ dữ liệu tràn, để lưu trữ các chương trình khi các chương trình như vậy được lựa chọn cho việc thực thi, và để lưu trữ các lệnh và dữ liệu mà được đọc trong suốt thời gian thực thi chương trình.

Fig.9 là lưu đồ của phương pháp 900 ví dụ về việc mã hóa chuỗi video thành dòng bit, chẳng hạn như dòng bit 700, bao gồm các tin nhắn SEI lồng ghép có thể mở rộng mà áp dụng cho các lớp. Phương pháp 900 có thể được ứng dụng bởi bộ mã hóa, chẳng hạn như hệ thống mã hóa-giải mã 200, bộ mã hóa 300, và/hoặc thiết bị lập mã video 800 khi thực hiện phương pháp 100. Hơn nữa, phương pháp 900 có thể hoạt động trên HRD 500 và do đó có thể thực hiện các thử nghiệm sự phù hợp trên chuỗi video đa lớp 600.

Phương pháp 900 có thể bắt đầu khi bộ mã hóa thu chuỗi video và xác định mã hóa chuỗi video đó thành dòng bit đa lớp, ví dụ dựa vào đầu vào người dùng. Ở bước 901, bộ mã hóa mã hóa chuỗi video thành một hoặc nhiều lớp và mã hóa các lớp thành dòng bit đa lớp. Do đó, dòng bit bao gồm một hoặc nhiều lớp. Lớp có thể bao gồm tập hợp của các bộ phận VCL NAL với cùng Id lớp và các bộ phận không phải NAL VCL được kết hợp. Ví dụ, lớp có thể bao gồm tập hợp của các bộ phận VCL NAL mà chứa dữ liệu video của các ảnh được mã hóa cũng như các tập hợp thông số bất kỳ được sử dụng để lập mã các ảnh như vậy. Một hoặc nhiều trong số các lớp có thể là các lớp đầu ra. Các lớp mà không là lớp đầu ra được mã hóa để hỗ trợ việc tái cấu trúc (các) lớp đầu ra, nhưng các lớp hỗ trợ như vậy không được dự định cho đầu ra ở bộ giải mã. Theo cách này, bộ mã hóa có thể mã hóa các sự kết hợp khác nhau của các lớp cho việc truyền tới bộ giải mã khi yêu cầu. Lớp có thể được truyền như được mong muốn để cho phép bộ giải mã thu nhận sự biểu diễn khác nhau của chuỗi video tùy thuộc vào các điều kiện mạng, các dung lượng phần cứng, và/hoặc các sự thiết đặt người dùng.

Ở bước 903, bộ mã hóa mã hóa một hoặc nhiều tin nhắn SEI lồng ghép có thể mở rộng thành dòng bit ở bộ phận SEI NAL hiện thời. Tin nhắn SEI là cấu trúc cú pháp mà chứa dữ liệu mà không được sử dụng để giải mã. Ví dụ, các tin nhắn SEI có thể chứa dữ liệu để hỗ trợ việc thử nghiệm sự phù hợp để đảm bảo dòng bit phù hợp với các chuẩn. Để hỗ trợ báo hiệu được đơn giản hóa khi được sử dụng cùng với dòng bit đa lớp, các tin nhắn SEI được mã hóa như các tin nhắn SEI lồng ghép có thể mở rộng. Tin nhắn SEI lồng ghép có thể mở rộng chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng. Các tin nhắn SEI được lồng ghép có thể mở rộng đều có thể áp dụng cho một hoặc nhiều trong số các

OLS và/hoặc một hoặc nhiều trong số các lớp. Để hỗ trợ báo hiệu được đơn giản hóa, tin nhắn SEI lồng ghép có thể mở rộng chứa một hoặc nhiều Id lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời. Sự ngoại trừ này ngăn ngừa báo hiệu phần dư của Id lớp của bộ phận SEI NAL hiện thời khi bộ phận SEI NAL hiện thời cũng chứa Id lớp tương ứng. Cần lưu ý rằng tin nhắn SEI lồng ghép có thể mở rộng bỏ qua Id lớp của bộ phận SEI NAL hiện thời thậm chí khi tin nhắn SEI lồng ghép có thể mở rộng bao gồm ít nhất một tin nhắn SEI được lồng ghép có thể mở rộng mà áp dụng cho lớp của bộ phận SEI NAL hiện thời.

Theo một số ví dụ, các Id lớp lồng ghép có thể mở rộng là thành phần cú pháp `layer_id[i]` lồng ghép có thể mở rộng mà xác định trị số `nuh_layer_id` của lớp thứ *i* trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng khi `all_layers_flag` lồng ghép có thể mở rộng bằng không. Theo một số ví dụ, mỗi trị số của `layer_id[i]` lồng ghép có thể mở rộng bị ràng buộc lớn hơn trị số `nuh_layer_id` của bộ phận SEI NAL hiện thời. Cần lưu ý rằng các thành phần cú pháp được mô tả ở đây khi bao gồm các dấu gạch dưới có thể được bao gồm ở dòng bit, trong khi các thành phần cú pháp mà không có các khoảng trống có thể được dẫn xuất. Các sự ràng buộc và/hoặc các yêu cầu nêu trên đảm bảo rằng dòng bit phù hợp với, ví dụ, VVC hoặc một số chuẩn khác, được điều chỉnh như được chỉ báo ở đây. Tuy nhiên, bộ mã hóa cũng có thể có khả năng thao tác ở các chế độ khác trong đó nó không bị quá ràng buộc, chẳng hạn như khi thao tác theo chuẩn khác hoặc phiên bản khác của cùng chuẩn.

Ở bước 905, HRD thao tác ở bộ mã hóa có thể thực hiện tập hợp của các thử nghiệm sự phù hợp dòng bit trên các lớp dựa vào tin nhắn SEI lồng ghép có thể mở rộng. Ví dụ, HRD có thể đọc các cờ trong tin nhắn SEI lồng ghép có thể mở rộng để xác định cách thức để diễn dịch các tin nhắn SEI được lồng ghép có thể mở rộng được chứa trong tin nhắn SEI lồng ghép có thể mở rộng. HRD sau đó có thể đọc các tin nhắn SEI được lồng ghép có thể mở rộng để xác định cách thức để kiểm tra các lớp về sự phù hợp với các chuẩn. HRD sau đó có thể thực hiện các thử nghiệm sự phù hợp trên các lớp dựa vào các tin nhắn SEI được lồng ghép có thể mở rộng và/hoặc các cờ tương ứng trong tin nhắn SEI lồng ghép có thể mở rộng.

Theo một số ví dụ, HRD có thể ứng dụng biến `nestingNumLayers` và biến `NestingLayerId[i]` để xác định các mối tương quan giữa các tin nhắn SEI được lồng ghép có

thể mở rộng và các lớp. Biến `nestingNumLayers` xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. Biến `NestingLayerId[i]` xác định danh mục của các trị số `nuh_layer_id` của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng đối với `i` nằm trong đoạn từ không đến `nestingNumLayers` trừ một. `NestingNumLayers` và `NestingLayerId[i]` có thể được dẫn xuất bởi HRD dựa vào dữ liệu từ dòng bit. HRD sau đó có thể ứng dụng biến `nestingNumLayers` và biến `NestingLayerId[i]` như được mong muốn để hỗ trợ việc thử nghiệm sự phù hợp. Theo một số ví dụ, `nestingNumLayers` và `NestingLayerId[i]` được dẫn xuất như sau:

```
if( scalable nesting all_layers_flag ) {
    nestingNumLayers = vps_max_layers_minus1 + 1 - GeneralLayerIdx[ nuh_layer_id ]
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
} else {
    nestingNumLayers = scalable nesting num_layers_minus1 + 1
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
}
```

trong đó `all_layers_flag` lồng ghép có thể mở rộng xác định xem các tin nhắn SEI được lồng ghép có thể mở rộng có áp dụng cho tất cả các lớp hay không, `vps_max_layers_minus1` cộng một xác định số lượng lớp được quy định bởi VPS, `GeneralLayerIdx` là biến xác định chỉ số lớp của lớp với `nuh_layer_id` bằng `vps_layer_id[ i ]`, `vps_layer_id[i]` xác định trị số `nuh_layer_id` của lớp thứ `i`, và `num_layers_minus1` cộng một lồng ghép có thể mở rộng xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. Theo một số ví dụ, `num_layers_minus1` lồng ghép có thể mở rộng bị ràng buộc vẫn nằm trong đoạn từ không đến `vps_max_layers_minus1 - GeneralLayerIdx[nuh_layer_id]`, trong đó `nuh_layer_id` là `nuh_layer_id` của bộ phận SEI NAL hiện thời. Cần lưu ý rằng các thành phần cú pháp được mô tả ở đây khi bao gồm các dấu gạch dưới có thể được bao gồm ở dòng bit, trong khi các thành phần cú pháp mà không có các khoảng trống có thể được dẫn xuất.

Ở bước 907, bộ mã hóa có thể lưu trữ dòng bit cho việc truyền thông về phía bộ giải mã khi yêu cầu. Bộ mã hóa cũng có thể truyền dòng bit về phía bộ mã hóa như mong muốn.

Fig.10 là lưu đồ của phương pháp 1000 ví dụ về việc giải mã chuỗi video từ dòng bit, chẳng hạn như dòng bit 700, bao gồm các tin nhắn SEI lồng ghép có thể mở rộng mà áp dụng

cho các lớp. Phương pháp 1000 có thể được ứng dụng bởi bộ giải mã, chẳng hạn như hệ thống mã hóa-giải mã 200, bộ giải mã 400, và/hoặc thiết bị lập mã video 800 khi thực hiện phương pháp 100. Hơn nữa, phương pháp 1000 có thể được ứng dụng trên chuỗi video đa lớp 600 mà đã được kiểm tra về sự phù hợp bởi HRD, chẳng hạn như HRD 500.

Phương pháp 1000 có thể bắt đầu khi bộ giải mã bắt đầu thu dòng bit của dữ liệu được lập mã biểu diễn chuỗi video đa lớp, ví dụ kết quả của phương pháp 900. Ở bước 1001, bộ giải mã thu dòng bit bao gồm một hoặc nhiều lớp. Lớp có thể bao gồm tập hợp của các bộ phận VCL NAL với cùng Id lớp và các bộ phận không phải NAL VCL được kết hợp. Ví dụ, lớp có thể bao gồm tập hợp của các bộ phận VCL NAL mà chứa dữ liệu video của các ảnh được mã hóa cũng như các tập hợp thông số bất kỳ được sử dụng để lập mã các ảnh như vậy. Một hoặc nhiều trong số các lớp có thể là các lớp đầu ra. Các lớp mà không là lớp đầu ra được giải mã để hỗ trợ việc giải mã (các) lớp đầu ra, nhưng các lớp hỗ trợ như vậy không được đưa ra.

Dòng bit cũng bao gồm một hoặc nhiều tin nhắn SEI lồng ghép có thể mở rộng. Tin nhắn SEI là cấu trúc cú pháp mà chứa dữ liệu mà không được sử dụng để giải mã. Ví dụ, các tin nhắn SEI có thể chứa dữ liệu để hỗ trợ việc thử nghiệm sự phù hợp để đảm bảo dòng bit phù hợp với các chuẩn. Để hỗ trợ báo hiệu được đơn giản hóa khi được sử dụng cùng với dòng bit đa lớp, các tin nhắn SEI được lập mã trong các tin nhắn SEI lồng ghép có thể mở rộng. Tin nhắn SEI lồng ghép có thể mở rộng bao gồm một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng. Tin nhắn SEI lồng ghép có thể mở rộng được quy định có thể được bao gồm trong bộ phận SEI NAL hiện thời, mà là bộ phận không phải VCL NAL mà chứa (các) tin nhắn SEI. Các tin nhắn SEI được lồng ghép có thể mở rộng đều có thể áp dụng cho một hoặc nhiều trong số các OLS và/hoặc một hoặc nhiều trong số các lớp. Để hỗ trợ báo hiệu được đơn giản hóa, tin nhắn SEI lồng ghép có thể mở rộng chứa một hoặc nhiều Id lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời. Sự ngoại trừ này ngăn ngừa báo hiệu phần dư của Id lớp của bộ phận SEI NAL hiện thời khi bộ phận SEI NAL hiện thời cũng chứa Id lớp tương ứng. Cần lưu ý rằng tin nhắn SEI lồng ghép có thể mở rộng bỏ qua Id lớp của bộ phận SEI NAL hiện thời thậm chí khi tin nhắn SEI lồng ghép có thể mở rộng bao gồm ít nhất một tin nhắn SEI được lồng ghép có thể mở rộng mà áp dụng cho lớp của bộ phận SEI NAL hiện thời.

Theo một số ví dụ, các Id lớp lồng ghép có thể mở rộng là thành phần cú pháp `layer_id[i]` lồng ghép có thể mở rộng mà xác định trị số `nuh_layer_id` của lớp thứ `i` trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng khi `all_layers_flag` lồng ghép có thể mở rộng bằng không. Theo một số ví dụ, mỗi trị số của `layer_id[i]` lồng ghép có thể mở rộng bị ràng buộc lớn hơn trị số `nuh_layer_id` của bộ phận SEI NAL hiện thời.

Theo một số ví dụ, bộ giải mã có thể ứng dụng biến `nestingNumLayers` và biến `NestingLayerId[i]` để xác định các mối tương quan giữa các tin nhắn SEI được lồng ghép có thể mở rộng và các lớp. Biến `nestingNumLayers` xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. Biến `NestingLayerId[i]` xác định danh mục của các trị số `nuh_layer_id` của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng đối với `i` nằm trong đoạn từ không đến `nestingNumLayers` trừ một. `NestingNumLayers` và `NestingLayerId[i]` có thể được dẫn xuất bởi bộ giải mã dựa vào dữ liệu từ dòng bit. Bộ giải mã sau đó có thể ứng dụng biến `nestingNumLayers` và biến `NestingLayerId[i]` như được mong muốn để hỗ trợ việc giải mã và/hoặc hiển thị. Theo một số ví dụ, `nestingNumLayers` và `NestingLayerId[i]` được dẫn xuất như sau:

```
if( scalable nesting all_layers_flag ) {
    nestingNumLayers = vps_max_layers_minus1 + 1 - GeneralLayerIdx[ nuh_layer_id ]
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
} else {
    nestingNumLayers = scalable nesting num_layers_minus1 + 1
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
}
```

trong đó `all_layers_flag` lồng ghép có thể mở rộng xác định xem các tin nhắn SEI được lồng ghép có thể mở rộng có áp dụng cho tất cả các lớp hay không, `vps_max_layers_minus1` cộng một xác định số lượng lớp được quy định bởi VPS, `GeneralLayerIdx` là biến xác định chỉ số lớp của lớp với `nuh_layer_id` bằng `vps_layer_id[ i ]`, `vps_layer_id[i]` xác định trị số `nuh_layer_id` của lớp thứ `i`, và `num_layers_minus1` cộng một lồng ghép có thể mở rộng xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng. Theo một số ví dụ, `num_layers_minus1` lồng ghép có thể mở rộng bị ràng buộc vẫn nằm trong

đoạn từ không đến $vps_max_layers_minus1 - GeneralLayerIdx[nuh_layer_id]$, trong đó nuh_layer_id là nuh_layer_id của bộ phận SEI NAL hiện thời.

Theo phương án, bộ giải mã video kỳ vọng tin nhắn SEI lồng ghép có thể mở rộng bỏ qua Id lớp của bộ phận SEI NAL hiện thời như được nêu trên dựa vào VVC hoặc một số chuẩn khác. Nếu, tuy nhiên, bộ giải mã xác định rằng điều kiện này là không đúng, thì bộ giải mã có thể phát hiện lỗi, báo hiệu lỗi, yêu cầu rằng dòng bit được sửa đổi (hoặc phần của chúng) được gửi lại, hoặc thực hiện một số biện pháp điều chỉnh khác để đảm bảo rằng dòng bit phù hợp được thu.

Ở bước 1003, bộ giải mã có thể giải mã ảnh được lập mã từ một hoặc nhiều lớp để tạo ra ảnh được giải mã. Ví dụ, sự có mặt của tin nhắn SEI lồng ghép có thể mở rộng có thể chỉ báo rằng dòng bit đã được kiểm tra bởi HRD ở bộ mã hóa và do đó phù hợp với các chuẩn. Theo đó, sự có mặt của tin nhắn SEI lồng ghép có thể mở rộng chỉ báo dòng bit có thể được giải mã. Ở bước 1005, bộ giải mã có thể chuyển tiếp ảnh được giải mã cho việc hiển thị như phần của chuỗi video được giải mã. Ví dụ, ảnh và/hoặc chuỗi video được giải mã có thể được hiển thị tới người dùng trên màn hình hoặc màn hiển thị của thiết bị điện tử (ví dụ, điện thoại thông minh, máy tính bảng, máy tính xách tay, máy tính cá nhân chẳng hạn).

Fig.11 là hình vẽ giản lược của hệ thống 1100 ví dụ về việc lập mã chuỗi video nhờ sử dụng dòng bit bao gồm các tin nhắn SEI lồng ghép có thể mở rộng mà áp dụng cho các lớp. Hệ thống 1100 có thể được thực hiện bởi bộ mã hóa và bộ giải mã chẳng hạn như hệ thống mã hóa-giải mã 200, bộ mã hóa 300, bộ giải mã 400, và/hoặc thiết bị lập mã video 800. Hơn nữa, hệ thống 1100 có thể ứng dụng HRD 500 để thực hiện các thử nghiệm sự phù hợp trên chuỗi video đa lớp 600 và/hoặc dòng bit 700. Ngoài ra, hệ thống 1100 có thể được ứng dụng khi thực hiện phương pháp 100, 900, và/hoặc 1000.

Hệ thống 1100 bao gồm bộ mã hóa video 1102. Bộ mã hóa video 1102 bao gồm môđun mã hóa 1103 để mã hóa dòng bit bao gồm một hoặc nhiều lớp. Môđun mã hóa 1103 còn để mã hóa thành dòng bit tin nhắn SEI lồng ghép có thể mở rộng ở bộ phận SEI NAL hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều Id lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời. Bộ mã hóa video 1102 còn bao gồm môđun HRD 1105 để thực hiện tập hợp của các thử nghiệm sự phù

hợp dòng bit trên các lớp dựa vào các tin nhắn SEI được lồng ghép có thể mở rộng. Bộ mã hóa video 1102 còn bao gồm môđun lưu trữ 1106 để lưu trữ dòng bit cho việc truyền thông về phía bộ giải mã. Bộ mã hóa video 1102 còn bao gồm môđun truyền 1107 để truyền dòng bit về phía bộ giải mã video 1110. Bộ mã hóa video 1102 còn có thể được tạo cấu hình để thực hiện bất kỳ trong số các bước của phương pháp 900.

Hệ thống 1100 cũng bao gồm bộ giải mã video 1110. Bộ giải mã video 1110 bao gồm môđun thu 1111 để thu dòng bit bao gồm một hoặc nhiều lớp và tin nhắn SEI lồng ghép có thể mở rộng ở bộ phận SEI NAL hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều Id lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời. Bộ giải mã video 1110 còn bao gồm môđun giải mã 1113 để giải mã ảnh được lập mã từ một hoặc nhiều lớp để tạo ra ảnh được giải mã. Bộ giải mã video 1110 còn bao gồm môđun chuyển tiếp 1115 để chuyển tiếp ảnh được giải mã cho việc hiển thị như phần của chuỗi video được giải mã. Bộ giải mã video 1110 còn có thể được tạo cấu hình để thực hiện bất kỳ trong số các bước của phương pháp 1000.

Bộ phận thứ nhất được ghép nối trực tiếp với bộ phận thứ hai khi không có các bộ phận xen kẽ, ngoại trừ dòng, dấu hiệu, hoặc phương tiện khác giữa bộ phận thứ nhất và bộ phận thứ hai. Bộ phận thứ nhất được ghép nối gián tiếp với bộ phận thứ hai khi có các bộ phận xen kẽ khác ngoài dòng, dấu hiệu, hoặc phương tiện khác giữa bộ phận thứ nhất và bộ phận thứ hai. Thuật ngữ “được ghép nối” và các biến thể của nó bao gồm cả được ghép nối trực tiếp và được ghép nối gián tiếp. Việc sử dụng thuật ngữ “khoảng” nghĩa là khoảng bao gồm $\pm 10\%$ của số tiếp theo trừ khi được nêu theo cách khác.

Cũng cần được hiểu rằng các bước của các phương pháp làm ví dụ được nêu ở đây không nhất thiết phải được thực hiện theo thứ tự được mô tả, và thứ tự của các bước của các phương án như vậy cần được hiểu chỉ là làm ví dụ. Tương tự như vậy, các bước bổ sung có thể được bao gồm trong các phương pháp như vậy, và các bước nhất định có thể được bỏ qua hoặc được kết hợp, trong các phương pháp phù hợp với các phương án khác nhau của sáng chế.

Trong khi nhiều phương án đã được đề xuất theo sáng chế, có thể được hiểu rằng các hệ thống và các phương pháp được bộc lộ có thể được thể hiện ở nhiều dạng cụ thể khác mà không chệch khỏi tinh thần hoặc phạm vi của sáng chế. Các ví dụ này được xem là có tính

minh họa và không có tính giới hạn, và dự định không bị giới hạn ở chi tiết được đưa ra ở đây. Ví dụ, các thành phần hoặc các bộ phận khác nhau có thể được kết hợp hoặc được tích hợp ở hệ thống khác hoặc các dấu hiệu nhất định có thể được bỏ qua, hoặc không được thực hiện.

Ngoài ra, các kỹ thuật, các hệ thống, các hệ thống con, và các phương pháp được mô tả và được minh họa theo các phương án khác nhau là rời rạc hoặc tách riêng có thể được kết hợp hoặc được tích hợp với các hệ thống, các bộ phận, các kỹ thuật, hoặc các phương pháp khác mà không chệch khỏi phạm vi của sáng chế. Các ví dụ khác về các sự thay đổi, các sự thay thế, và các sự sửa đổi có thể xác định được bởi người có hiểu biết trong lĩnh vực kỹ thuật tương ứng và có thể được thực hiện mà không chệch khỏi tinh thần và phạm vi được bộc lộ ở đây.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã được thực hiện ở bộ giải mã, trong đó phương pháp này bao gồm các bước:

thu, bởi bộ thu của bộ giải mã, dòng bit bao gồm một hoặc nhiều lớp và tin nhắn thông tin nâng cao bổ sung (supplemental enhancement information, viết tắt là SEI) lồng ghép có thể mở rộng ở bộ phận lớp trừu tượng hóa mạng (network abstraction layer, viết tắt là NAL) SEI hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều ký hiệu nhận dạng (identifier, viết tắt là Id) lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời;

dẫn xuất, số lượng lớp lồng ghép, nestingNumLayers, và danh mục Id lớp lồng ghép, NestingLayerId[i], trong đó nestingNumLayers xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng, và trong đó NestingLayerId[i] xác định danh mục trị số nuh_layer_id của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng cho i trong đoạn từ không đến nestingNumLayers trừ một, bao gồm, và trong đó nestingNumLayers và NestingLayerId[i] được dẫn xuất như sau:

```

if(          scalable          nesting          all_layers_flag          )          {
    nestingNumLayers
    =
    vps_max_layers_minus1 + 1 -          GeneralLayerIdx[ nuh_layer_id ]
    for(      i      =      0;      i      <      nestingNumLayers;      i      ++ )
        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
    }
    else          {
        nestingNumLayers          =          scalable          nesting          num_layers_minus1 + 1
        for(      i      =      0;      i      <      nestingNumLayers;      i      ++ )
            NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
    }

```

trong đó vps_max_layers_minus1 cộng một xác định số lượng lớp được quy định bởi tập hợp thông số video, (video parameter set, viết tắt là VPS), GeneralLayerIdx là biến xác định chỉ số lớp của lớp với nuh_layer_id bằng vps_layer_id[i], vps_layer_id[i] xác định trị số nuh_layer_id của lớp thứ i, num_layers_minus1 cộng một lồng ghép có thể mở rộng xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng, và

layer_id[i] lồng ghép có thể mở rộng xác định trị số nuh_layer_id của lớp thứ i trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng;

giải mã, bởi bộ xử lý của bộ giải mã, ảnh được lập mã từ một hoặc nhiều lớp để tạo ra ảnh được giải mã.

2. Phương pháp theo điểm 1, trong đó mỗi trị số layer_id[i] lồng ghép có thể mở rộng lớn hơn so với trị số nuh_layer_id của bộ phận SEI NAL hiện thời.

3. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 2, trong đó num_layers_minus1 lồng ghép có thể mở rộng trong đoạn từ không đến vps_max_layers_minus1 – GeneralLayerIdx[nuh_layer_id], bao gồm, trong đó nuh_layer_id là nuh_layer_id của bộ phận SEI NAL hiện thời.

4. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 2, trong đó ảnh được lập mã được giải mã dựa vào các tin nhắn SEI được lồng ghép có thể mở rộng.

5. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 2, trong đó, tin nhắn SEI lồng ghép có thể mở rộng bao gồm ít nhất một tin nhắn SEI được lồng ghép có thể mở rộng mà áp dụng cho lớp của bộ phận SEI NAL hiện thời.

6. Phương pháp mã hóa được thực hiện ở bộ mã hóa video, trong đó phương pháp này bao gồm các bước:

mã hóa, bởi bộ xử lý của bộ mã hóa, dòng bit bao gồm một hoặc nhiều lớp;

mã hóa thành dòng bit, bởi bộ xử lý, tin nhắn thông tin nâng cao bổ sung (SEI) lồng ghép có thể mở rộng ở bộ phận lớp trừu tượng hóa mạng (NAL) SEI hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều ký hiệu nhận dạng (Id) lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng trong khi áp dụng sự ràng buộc để bỏ qua trị số Id lớp của bộ phận SEI NAL hiện thời;

đẫn xuất, bởi bộ xử lý, số lượng lớp lồng ghép, nestingNumLayers, và danh mục Id lớp lồng ghép, NestingLayerId[i], trong đó nestingNumLayers xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng, và trong đó NestingLayerId[i] xác định danh mục trị số nuh_layer_id của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng cho i trong đoạn từ không đến nestingNumLayers trừ một, bao gồm, và trong đó nestingNumLayers và NestingLayerId[i] được dẫn xuất như sau:

```

if( scalable nesting all_layers_flag ) {
    nestingNumLayers =
vps_max_layers_minus1 + 1 - GeneralLayerIdx[ nuh_layer_id ]
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
}
else {
    nestingNumLayers = scalable nesting num_layers_minus1 + 1
    for( i = 0; i < nestingNumLayers; i ++ )
        NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
}

```

trong đó `vps_max_layers_minus1` cộng một xác định số lượng lớp được quy định bởi tập hợp thông số video, (VPS), `GeneralLayerIdx` là biến xác định chỉ số lớp của lớp với `nuh_layer_id` bằng `vps_layer_id[ i ]`, `vps_layer_id[i]` xác định trị số `nuh_layer_id` của lớp thứ `i`, `num_layers_minus1` cộng một lồng ghép có thể mở rộng xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng, và `layer_id[i]` lồng ghép có thể mở rộng xác định trị số `nuh_layer_id` của lớp thứ `i` trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng.

7. Phương pháp theo điểm 6, trong đó mỗi trị số `layer_id[i]` lồng ghép có thể mở rộng lớn hơn so với trị số `nuh_layer_id` của bộ phận SEI NAL hiện thời.

8. Phương pháp theo điểm bất kỳ trong số các điểm từ 6 đến 7, trong đó `num_layers_minus1` lồng ghép có thể mở rộng trong đoạn từ không đến `vps_max_layers_minus1 - GeneralLayerIdx[nuh_layer_id]`, bao gồm, trong đó `nuh_layer_id` là `nuh_layer_id` của bộ phận SEI NAL hiện thời.

9. Phương pháp theo điểm bất kỳ trong số các điểm từ 6 đến 7, trong đó, tin nhắn SEI lồng ghép có thể mở rộng bao gồm ít nhất một tin nhắn SEI được lồng ghép có thể mở rộng mà áp dụng cho lớp của bộ phận SEI NAL hiện thời.

10. Thiết bị lập mã video bao gồm:

bộ xử lý, máy thu được ghép nối với bộ xử lý, bộ nhớ được ghép nối với bộ xử lý và máy phát được ghép nối với bộ xử lý, trong đó bộ xử lý, máy thu, bộ nhớ, và máy phát được cấu hình để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 5.

11. Thiết bị lập mã video bao gồm:

bộ xử lý, máy thu được ghép nối với bộ xử lý, bộ nhớ được ghép nối với bộ xử lý và máy phát được ghép nối với bộ xử lý, trong đó bộ xử lý, máy thu, bộ nhớ, và máy phát được cấu hình để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 6 đến 9.

12. Phương tiện đọc được bằng máy tính không tạm thời bao gồm sản phẩm chương trình máy tính cho việc sử dụng bởi thiết bị lập mã video, sản phẩm chương trình máy tính này bao gồm các lệnh thi hành được bởi máy tính được lưu trữ trên phương tiện đọc được bằng máy tính không tạm thời sao cho khi được thi hành bởi bộ xử lý khiến thiết bị lập mã video thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 5.

13. Phương tiện đọc được bằng máy tính không tạm thời bao gồm sản phẩm chương trình máy tính cho việc sử dụng bởi thiết bị lập mã video, sản phẩm chương trình máy tính này bao gồm các lệnh thi hành được bởi máy tính được lưu trữ trên phương tiện đọc được bằng máy tính không tạm thời sao cho khi được thi hành bởi bộ xử lý khiến thiết bị lập mã video thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 6 đến 9.

14. Bộ giải mã bao gồm:

phương tiện thu để thu dòng bit bao gồm một hoặc nhiều lớp và tin nhắn thông tin nâng cao bổ sung (SEI) lồng ghép có thể mở rộng ở bộ phận lớp trừu tượng hóa mạng (NAL) SEI hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều ký hiệu nhận dạng (Id) lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng ngoại trừ trị số Id lớp của bộ phận SEI NAL hiện thời;

phương tiện dẫn xuất để dẫn xuất số lượng lớp lồng ghép, `nestingNumLayers`, và danh mục Id lớp lồng ghép, `NestingLayerId[i]`, trong đó `nestingNumLayers` xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng, và trong đó `NestingLayerId[i]` xác định danh mục trị số `nuh_layer_id` của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng cho `i` trong đoạn từ không đến `nestingNumLayers` trừ một, bao gồm, và trong đó `nestingNumLayers` và `NestingLayerId[i]` được dẫn xuất như sau:

```

if(          scalable          nesting          all_layers_flag          )          {
    nestingNumLayers          =
vps_max_layers_minus1 + 1 -          GeneralLayerIdx[ nuh_layer_id ]
    for(      i      =      0;      i      <      nestingNumLayers;      i      ++)
```

```

        NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
    }
    else {
        nestingNumLayers = scalable nesting num_layers_minus1 + 1
        for( i = 0; i < nestingNumLayers; i ++ )
            NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
    }

```

trong đó `vps_max_layers_minus1` cộng một xác định số lượng lớp được quy định bởi tập hợp thông số video, (VPS), `GeneralLayerIdx` là biến xác định chỉ số lớp của lớp với `nuh_layer_id` bằng `vps_layer_id[ i ]`, `vps_layer_id[i]` xác định trị số `nuh_layer_id` của lớp thứ `i`, `num_layers_minus1` cộng một lồng ghép có thể mở rộng xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng, và `layer_id[i]` lồng ghép có thể mở rộng xác định trị số `nuh_layer_id` của lớp thứ `i` trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng;

phương tiện giải mã để giải mã ảnh được lập mã từ một hoặc nhiều lớp để tạo ra ảnh được giải mã.

15. Bộ giải mã theo điểm 14, trong đó bộ giải mã còn được tạo cấu hình để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 5.

16. Bộ mã hóa video bao gồm:

phương tiện mã hóa để:

mã hóa dòng bit bao gồm một hoặc nhiều lớp; và

mã hóa thành dòng bit tin nhắn thông tin nâng cao bổ sung (SEI) lồng ghép có thể mở rộng ở bộ phận lớp trừu tượng hóa mạng (NAL) SEI hiện thời, trong đó tin nhắn SEI lồng ghép có thể mở rộng này chứa một hoặc nhiều tin nhắn SEI được lồng ghép có thể mở rộng và một hoặc nhiều ký hiệu nhận dạng (Id) lớp lồng ghép có thể mở rộng mà xác định các trị số Id lớp của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng trong khi áp dụng sự ràng buộc để bỏ qua trị số Id lớp của bộ phận SEI NAL hiện thời;

phương tiện dẫn xuất để dẫn xuất số lượng lớp lồng ghép, `nestingNumLayers`, và danh mục Id lớp lồng ghép, `NestingLayerId[i]`, trong đó `nestingNumLayers` xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng, và trong đó `NestingLayerId[i]` xác định danh mục trị số `nuh_layer_id` của các lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng cho `i` trong đoạn từ không đến `nestingNumLayers` trừ một, bao gồm, và trong đó `nestingNumLayers` và `NestingLayerId[i]`

được dẫn xuất như sau:

```

    if(          scalable          nesting          all_layers_flag          )          {
        nestingNumLayers
vps_max_layers_minus1 + 1 -          GeneralLayerIdx[ nuh_layer_id ]
        for(      i      =      0;      i      <      nestingNumLayers;      i      ++ )
            NestingLayerId[ i ] = vps_layer_id[ GeneralLayerIdx[ nuh_layer_id ] + i ]
    }
        else          {
            nestingNumLayers      =      scalable      nesting      num_layers_minus1 + 1
            for(      i      =      0;      i      <      nestingNumLayers;      i      ++ )
                NestingLayerId[ i ] = ( i == 0 ) ? nuh_layer_id : scalable nesting layer_id[ i ]
        }

```

trong đó `vps_max_layers_minus1` cộng một xác định số lượng lớp được quy định bởi tập hợp thông số video, (VPS), `GeneralLayerIdx` là biến xác định chỉ số lớp của lớp với `nuh_layer_id` bằng `vps_layer_id[ i ]`, `vps_layer_id[i]` xác định trị số `nuh_layer_id` của lớp thứ `i`, `num_layers_minus1` cộng một lồng ghép có thể mở rộng xác định số lượng lớp trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng, và `layer_id[i]` lồng ghép có thể mở rộng xác định trị số `nuh_layer_id` của lớp thứ `i` trong đó các tin nhắn SEI được lồng ghép có thể mở rộng áp dụng.

17. Bộ mã hóa theo điểm 16, trong đó bộ mã hóa này còn được cấu hình để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 6 đến 9.

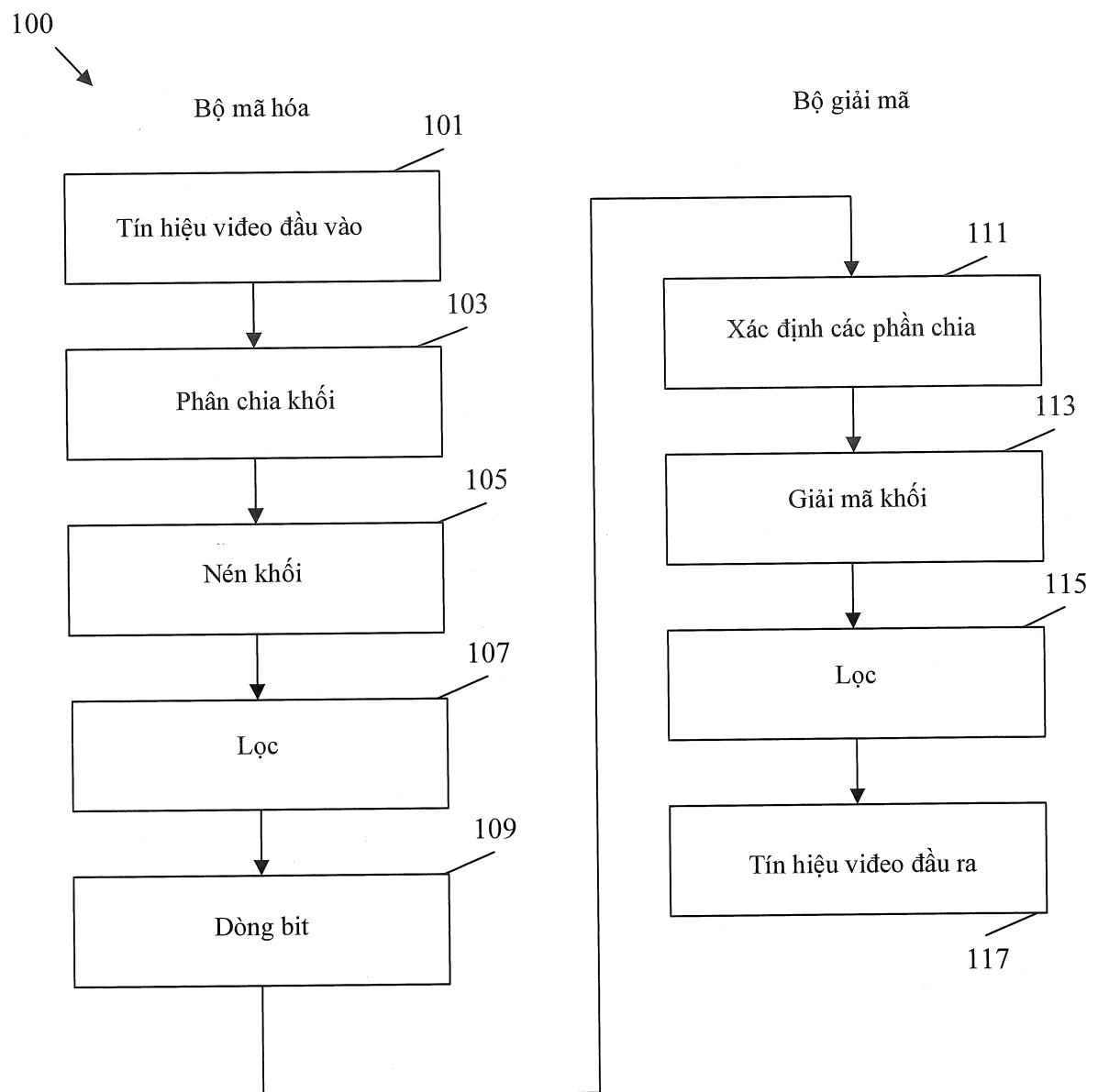


FIG. 1

2/11

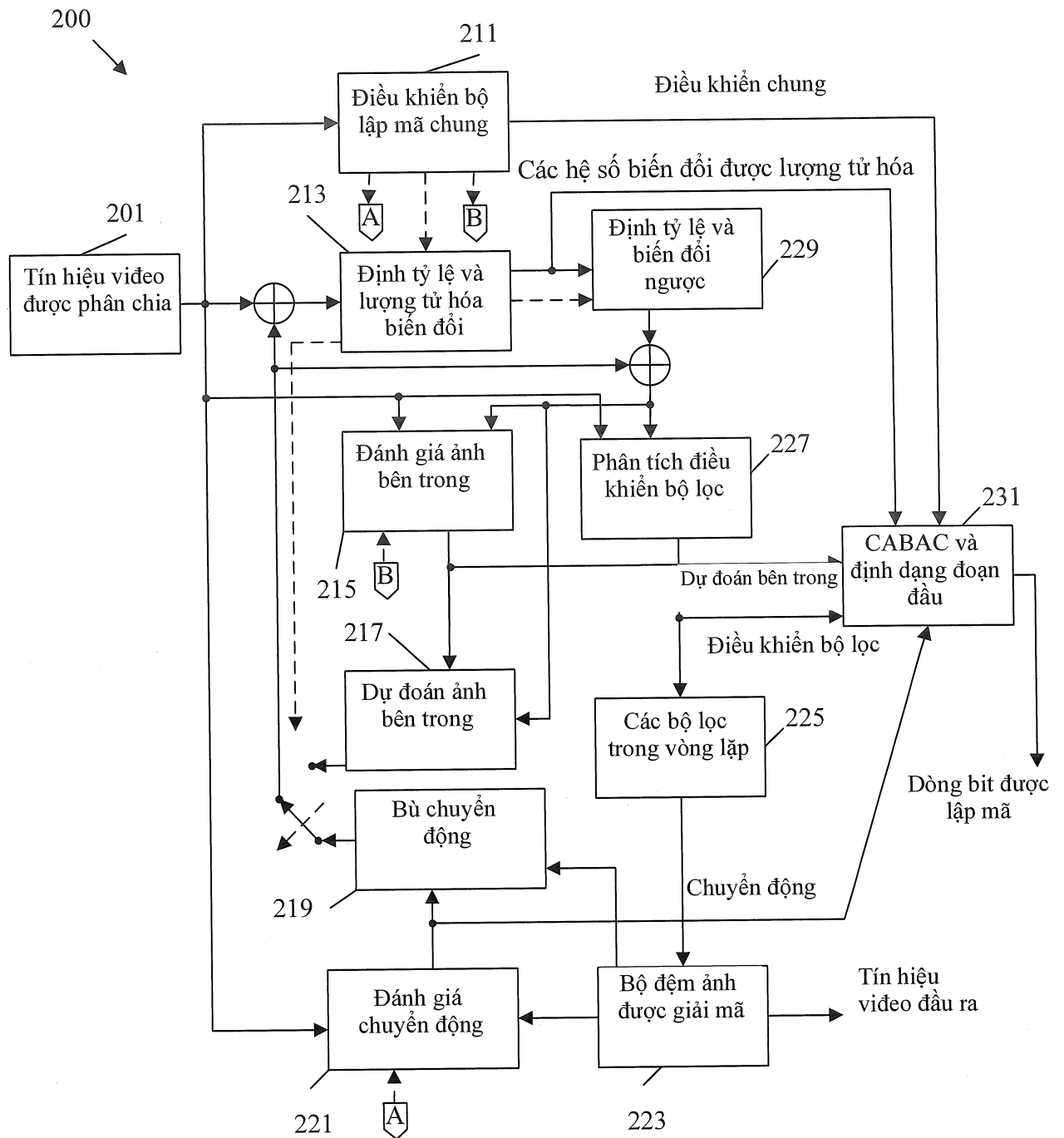


FIG. 2

3/11

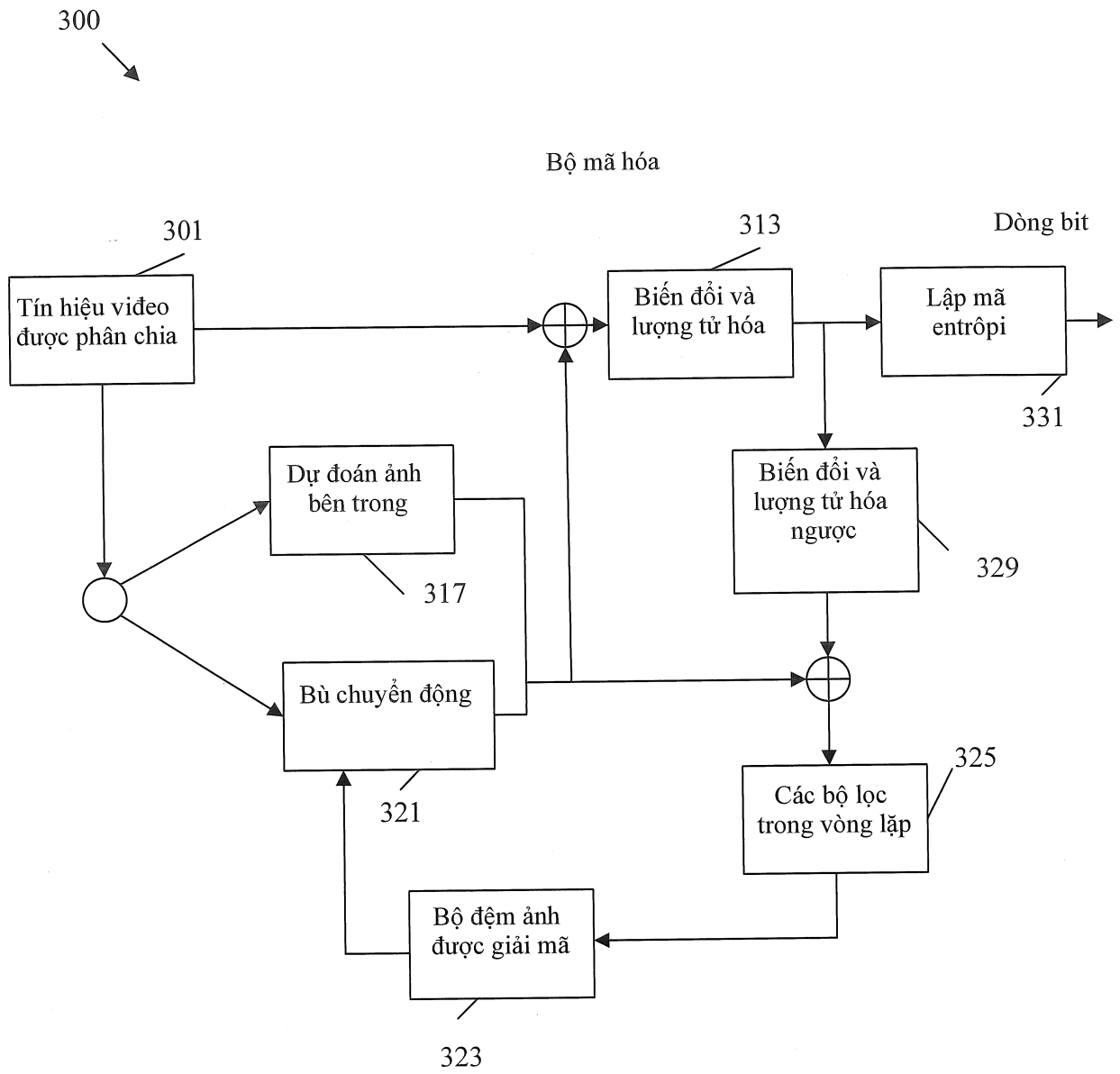


FIG. 3

4/11

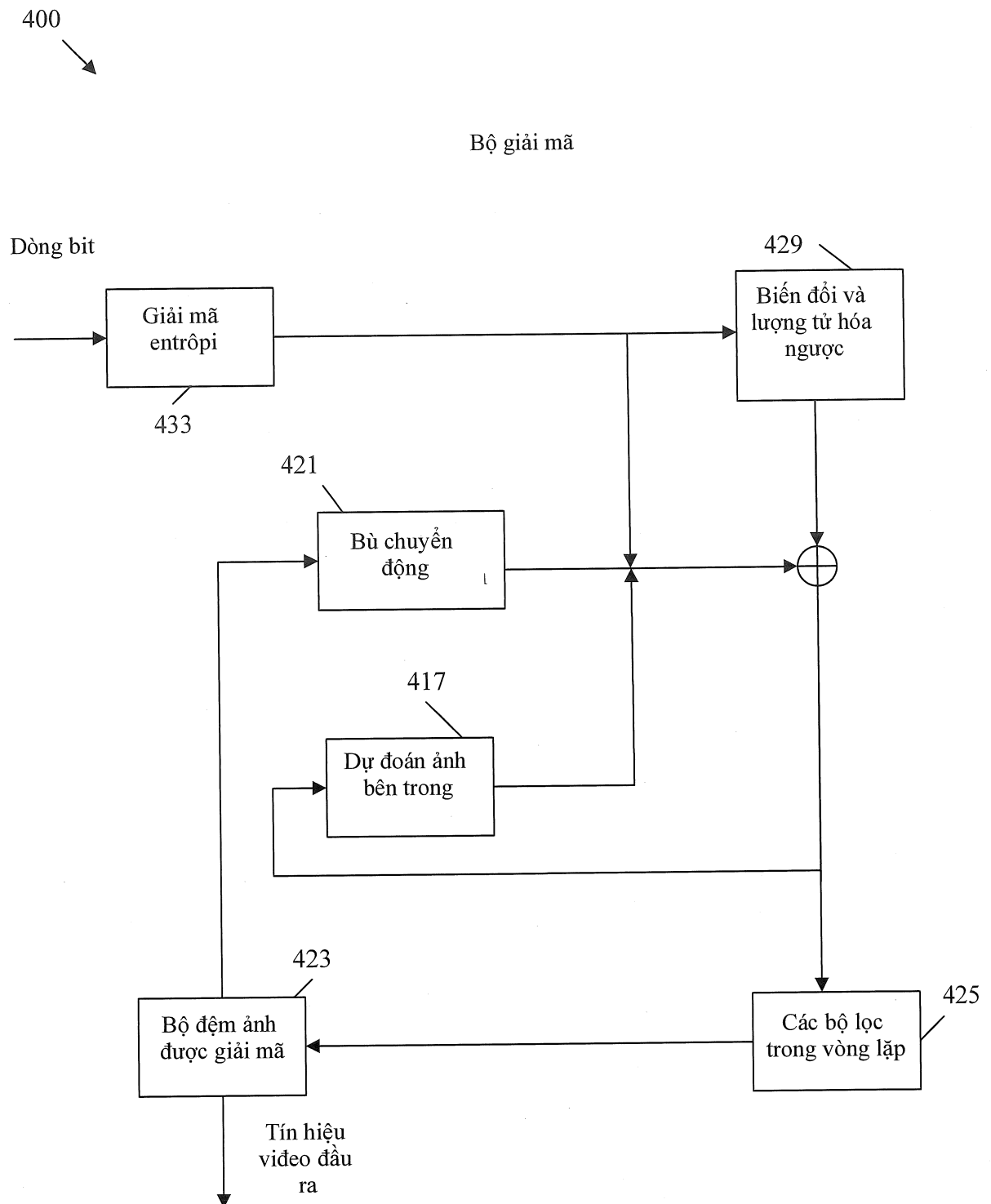


FIG. 4

5/11

Bộ giải mã tham chiếu giả định

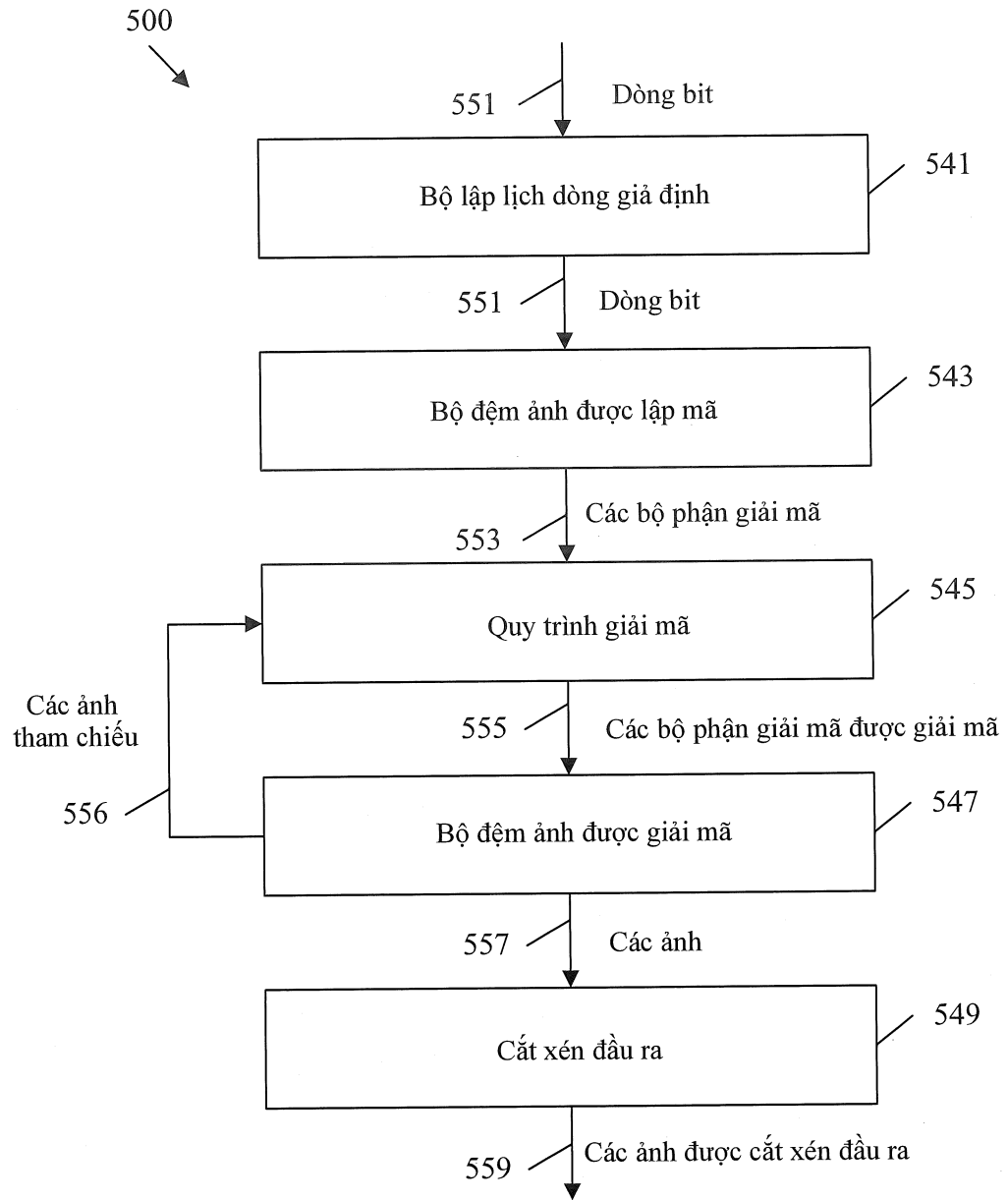


FIG. 5

6/11

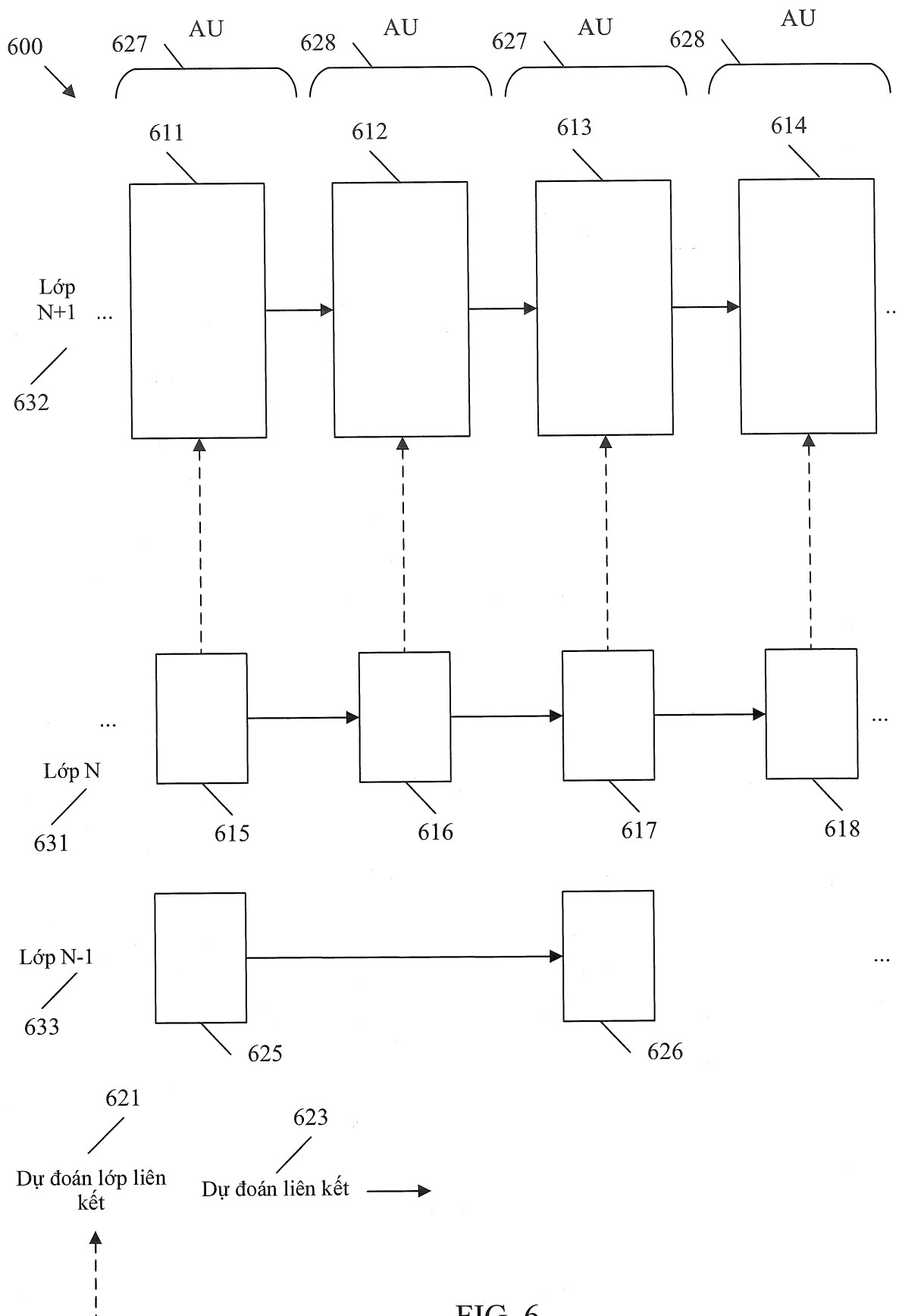


FIG. 6

7/11

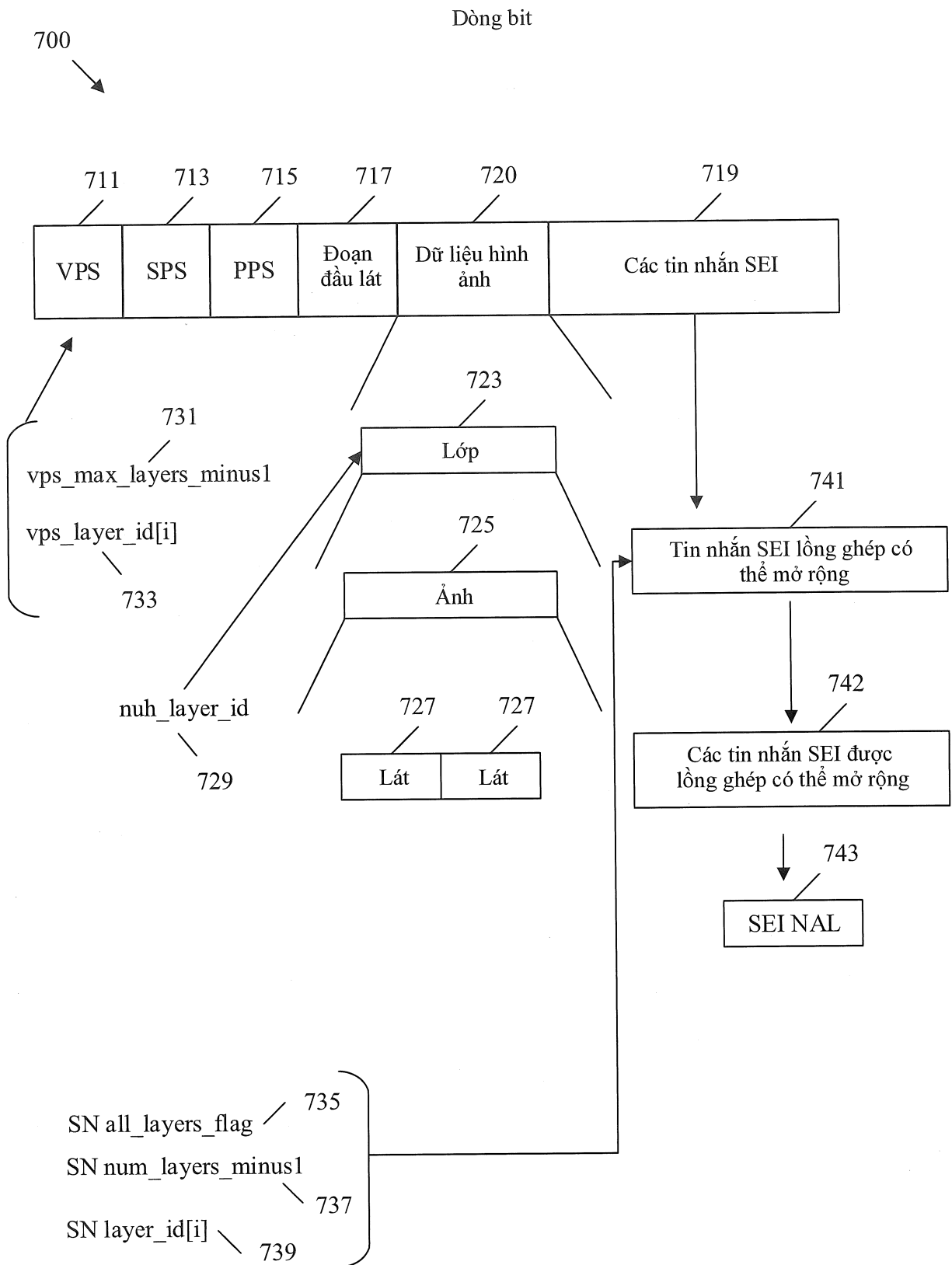


FIG. 7

8/11

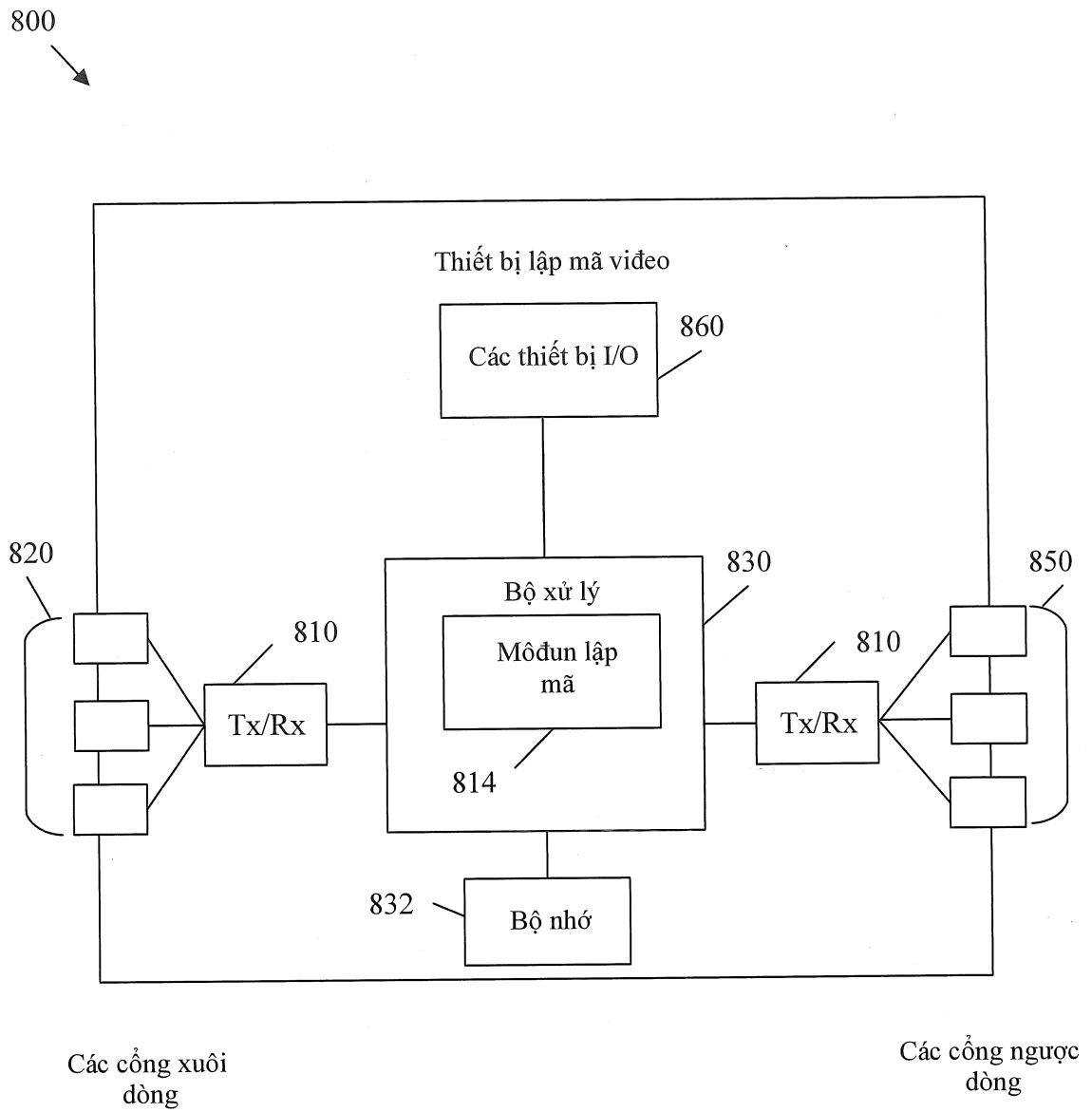


FIG. 8

9/11

900

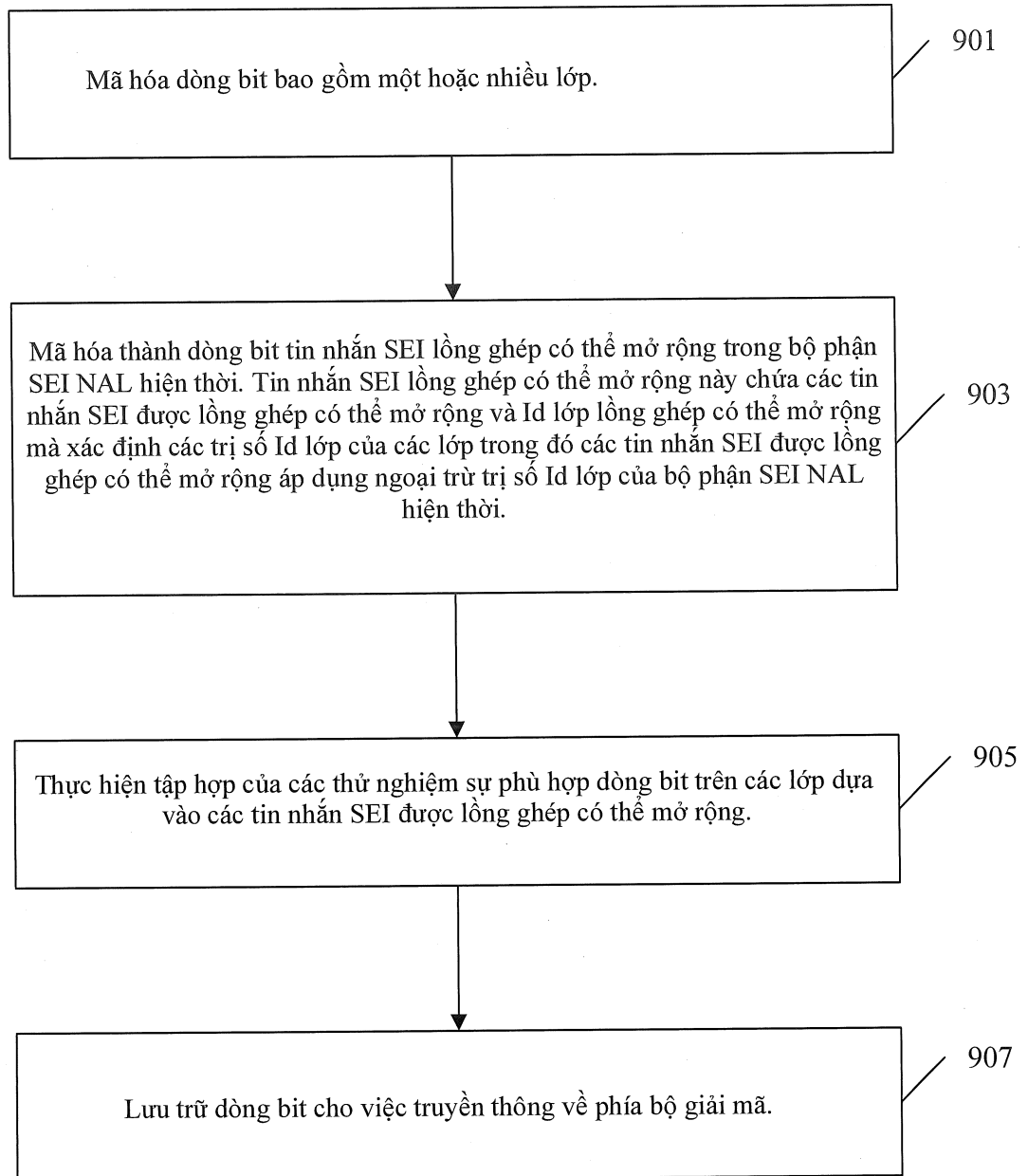


FIG. 9

10/11

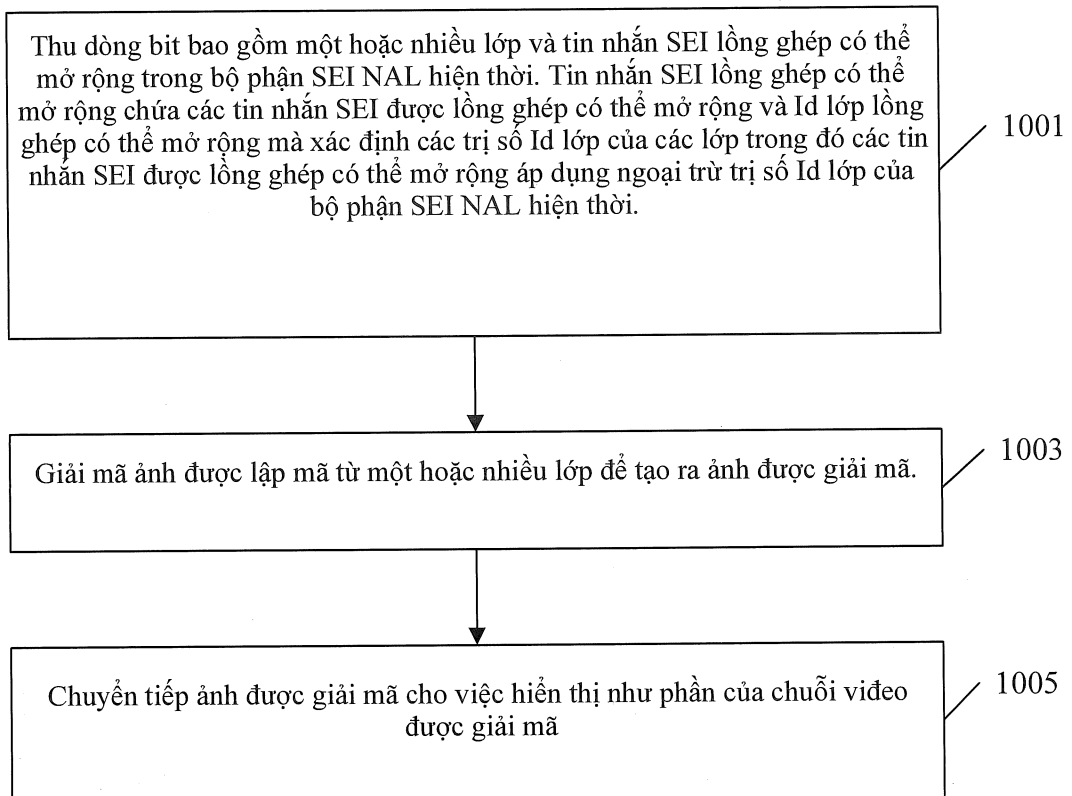
1000
↓

FIG. 10

11/11

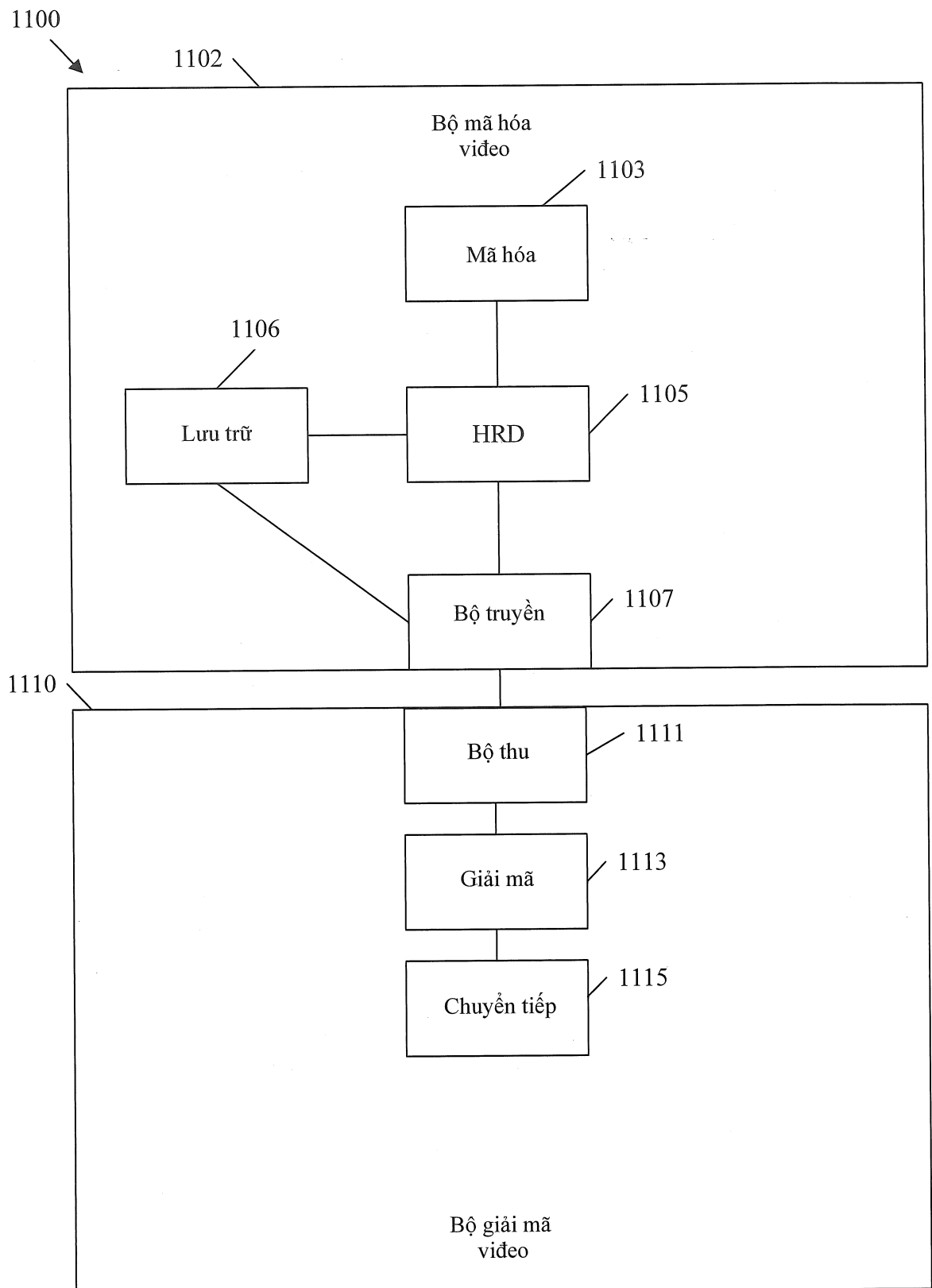


FIG. 11