



(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)



1-0048262

(51)<sup>2006.01</sup> H04N 7/26

(13) B

(21) 1-2021-07689

(22) 27/04/2020

(86) PCT/US2020/030104 27/04/2020

(87) WO2020/226930 12/11/2020

(30) 62/843,047 03/05/2019 US; 62/864,966 21/06/2019 US

(45) 25/07/2025 448

(43) 25/02/2022 407A

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

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

(72) HENDRY, Fnu (ID); WANG, Ye-Kui (US).

(74) Công ty Luật TNHH T&G (TGVN)

(54) PHƯƠNG PHÁP GIẢI MÃ, THIẾT BỊ GIẢI MÃ, HỆ THỐNG GIẢI MÃ, MÔĐUN  
GIẢI MÃ, BỘ GIẢI MÃ VIDEO VÀ PHƯƠNG TIỆN LƯU TRỮ ĐỌC ĐƯỢC  
BẰNG MÁY TÍNH

(21) 1-2021-07689

(57) Sáng chế đề cập đến phương pháp giải mã, phương pháp mã hóa, thiết bị giải mã, thiết bị mã hóa, bộ giải mã, bộ mã hóa, hệ thống tạo mã, phương tiện tạo mã, và phương tiện lưu trữ. Phương pháp giải mã bao gồm bước kết hợp cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video, mà chỉ báo rằng khối cây tạo mã (CTB) hiện hành là CTB cuối cùng trong phiên; kết hợp cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video, mà chỉ báo rằng việc xử lý song song dạng sóng (WPP) là được phép và rằng CTB hiện hành là CTB cuối cùng trong hàng CTB nhưng không phải CTB cuối cùng trong phiên; và tái tạo các CTB trong phiên dựa trên cuối của bit phiên với giá trị thứ nhất, cuối của bit hàng CTB với giá trị thứ nhất, và các bit cân chỉnh bai.

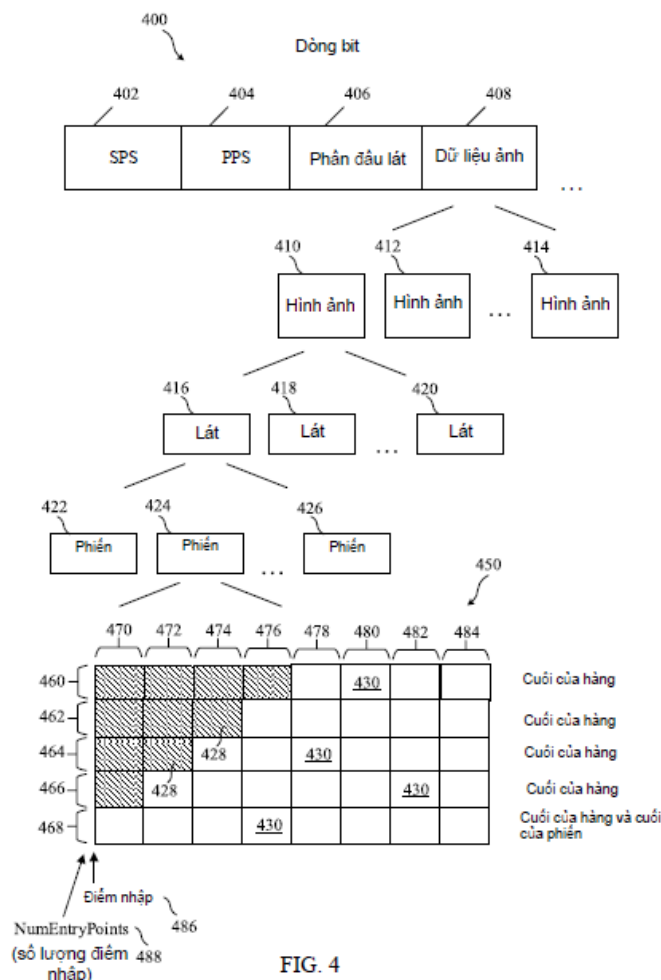


FIG. 4

### **Lĩnh vực kỹ thuật được đề cập**

Nói chung, sáng chế đề cập đến các kỹ thuật hỗ trợ xử lý song song mặt sóng (wavefront parallel processing, WPP) khi tạo mã video. Cụ thể hơn, sáng chế giúp ngăn chặn sự sao chép bit cân chỉnh bit và bai (byte) không cần thiết trong WPP.

### **Tình trạng kỹ thuật của sáng chế**

Lượng dữ liệu video cần để mô tả thậm chí video tương đối ngắn có thể là quan trọng, mà có thể dẫn đến những khó khăn khi dữ liệu cần được tạo luồng hoặc nếu không thì được truyền thông qua mạng truyền thông với dung lượng dải thông hạn chế. Vì vậy, dữ liệu video thường được nén trước khi được truyền thông qua các mạng viễn thông đương thời. Kích thước của video cũng có thể là vấn đề khi video được lưu trữ trên thiết bị lưu trữ do 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 tạo mã dữ liệu video trước khi truyền hoặc lưu trữ, nhờ đó làm giảm lượng dữ liệu cần để đại diện cho các ảnh video số. Dữ liệu đã nén sau đó được nhận tại điểm đến bởi thiết bị giải nén video để giải mã dữ liệu video. Với các tài nguyên mạng hạn chế và các nhu cầu luôn tăng lên về chất lượng video cao hơn, các kỹ thuật nén và giải nén cải tiến để nâng cao tỷ lệ nén với việc đánh đổi một ít cho đến không phải đánh đổi về chất lượng ảnh luôn được mong đợi.

### **Bản chất kỹ thuật của sáng chế**

Khía cạnh thứ nhất đề cập đến phương pháp giải mã dòng bit video được tạo mã được thực hiện bởi bộ giải mã video. Phương pháp này bao gồm các bước nhận, bởi bộ giải mã video, dòng bit video được tạo mã, trong đó dòng bit video được tạo mã chứa hình ảnh, hình ảnh này bao gồm một hoặc nhiều lát có một hoặc nhiều phiến (tile), mỗi phiến chứa các khối cây tạo mã (coding tree block, CTB); kết hợp, bởi bộ giải mã video, cuối của bit phiến với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video được tạo mã, cuối của bit phiến với giá trị thứ nhất và các bit cân chỉnh bai chỉ báo rằng CTB

hiện hành từ các CTB là CTB cuối cùng trong phiên; kết hợp, bởi bộ giải mã video, cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video được tạo mã, cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai chỉ báo rằng việc xử lý song song dạng sóng (waveform parallel processing, WPP) là được phép và rằng CTB hiện hành từ các CTB là CTB cuối cùng trong hàng CTB nhưng không phải CTB cuối cùng trong phiên; và tái tạo, bởi bộ giải mã video, các CTB trong phiên dựa trên cuối của bit phiên với giá trị thứ nhất, cuối của bit hàng CTB với giá trị thứ nhất, và các bit cân chỉnh bai.

Phương pháp đề xuất các kỹ thuật để ngăn chặn sự sao chép báo hiệu và cân chỉnh bai trong WPP. Bằng cách loại bỏ sự sao chép báo hiệu và cân chỉnh bai trong WPP, số lượng bit được sử dụng để báo hiệu cuối của hàng/phiên và số lượng bit được sử dụng làm đệm được giảm bớt. Nhờ làm giảm số lượng bit cần cho WPP, bộ tạo mã / bộ giải mã (cũng được gọi là “codecs”) trong tạo mã video được nâng cao so với các codecs hiện hành. Như một vấn đề thực tế, quy trình tạo mã video được nâng cao đem lại cho người dùng trải nghiệm người dùng tốt hơn khi các video được gửi, được nhận, và/hoặc được xem.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc cuối của bit phiên được biểu thị là `end_of_tile_one_bit`.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc cuối của bit hàng CTB được biểu thị là `end_of_subset_bit`.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc WPP được phép bởi cờ được bố trí trong tập hợp thông số.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc WPP được phép bởi cờ được biểu thị là `entropy_coding_sync_enabled_flag`.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc giá trị thứ nhất là một (1) khi WPP được phép.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất hiển thị ảnh được tạo ra dựa trên các CTB như được tái tạo.

Khía cạnh thứ hai đề cập đến phương pháp mã hóa dòng bit video được thực hiện bởi bộ mã hóa video. Phương pháp này bao gồm bước phân chia, bởi bộ mã hóa video, hình ảnh thành một hoặc nhiều lát, mỗi lát chứa một hoặc nhiều phiên, mỗi phiên chứa các khối cây tạo mã (coding tree block, CTB); mã hóa, bởi bộ mã hóa video, cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh lại thành dòng bit video khi CTB hiện hành từ các CTB là CTB cuối cùng trong phiên; mã hóa, bởi bộ mã hóa video, cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh lại thành dòng bit video khi việc xử lý song song dạng sóng (WPP) được phép và khi CTB hiện hành là CTB cuối cùng trong hàng CTB nhưng không phải CTB cuối cùng trong phiên; và lưu trữ, bởi bộ mã hóa video, dòng bit video để truyền đến bộ giải mã video.

Phương pháp đề xuất các kỹ thuật để ngăn chặn sự sao chép báo hiệu và cân chỉnh lại trong WPP. Bằng cách loại bỏ sự sao chép báo hiệu và cân chỉnh lại trong WPP, số lượng bit được sử dụng để báo hiệu cuối của hàng/phiên và số lượng bit được sử dụng làm đệm được giảm bớt. Nhờ làm giảm số lượng bit cần cho WPP, bộ tạo mã / bộ giải mã (cũng được gọi là “codec”) trong tạo mã video được nâng cao so với các codec hiện hành. Như một vấn đề thực tế, quy trình tạo mã video được nâng cao đem lại cho người dùng trải nghiệm người dùng tốt hơn khi các video được gửi, được nhận, và/hoặc được xem.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc cuối của bit phiên được biểu thị là `end_of_tile_one_bit`.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc cuối của bit hàng CTB được biểu thị là `end_of_subset_bit`.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc WPP được phép bởi cờ được bố trí trong tập hợp thông số.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc WPP được phép bởi cờ được biểu thị là `entropy_coding_sync_enabled_flag`.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc giá trị thứ nhất là một (1) khi WPP được phép.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc truyền dòng bit video đến bộ giải mã video.

Khía cạnh thứ ba đề cập đến thiết bị giải mã. Thiết bị giải mã bao gồm bộ nhận được tạo cấu hình để nhận dòng bit video được tạo mã; bộ nhớ được ghép nối với bộ nhận, bộ nhớ lưu trữ các lệnh; và bộ xử lý được ghép nối với bộ nhớ, bộ xử lý được tạo cấu hình để thực hiện các lệnh để làm cho thiết bị giải mã: nhận dòng bit video được tạo mã, trong đó dòng bit video được tạo mã chứa hình ảnh, hình ảnh này bao gồm một hoặc nhiều lát có một hoặc nhiều phiên, mỗi phiên chứa các khối cây tạo mã (CTB); kết hợp cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video được tạo mã, cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bai chỉ báo rằng CTB hiện hành từ các CTB là CTB cuối cùng trong phiên; kết hợp cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video được tạo mã, cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai chỉ báo rằng việc xử lý song song dạng sóng (waveform parallel processing, WPP) là được phép và rằng CTB hiện hành từ các CTB là CTB cuối cùng trong hàng CTB nhưng không phải CTB cuối cùng trong phiên; và tái tạo các CTB trong phiên dựa trên cuối của bit phiên với giá trị thứ nhất, cuối của bit hàng CTB với giá trị thứ nhất, và các bit cân chỉnh bai.

Thiết bị giải mã đề xuất các kỹ thuật để ngăn chặn sự sao chép báo hiệu và cân chỉnh bai trong WPP. Bằng cách loại bỏ sự sao chép báo hiệu và cân chỉnh bai trong WPP, số lượng bit được sử dụng để báo hiệu cuối của hàng/phiên và số lượng bit được sử dụng làm đệm được giảm bớt. Nhờ làm giảm số lượng bit cần cho WPP, bộ tạo mã / bộ giải mã (cũng được gọi là “codec”) trong tạo mã video được nâng cao so với các

codec hiện hành. Như một vấn đề thực tế, quy trình tạo mã video được nâng cao đem lại cho người dùng trải nghiệm người dùng tốt hơn khi các video được gửi, được nhận, và/hoặc được xem.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc cuối của bit phiên được biểu thị là `end_of_tile_one_bit`, trong đó cuối của bit hàng CTB được biểu thị là `end_of_subset_bit`, và giá trị thứ nhất là một.

Khía cạnh thứ tư đề cập đến thiết bị mã hóa. Thiết bị mã hóa bao gồm bộ nhớ chứa các lệnh; bộ xử lý được ghép nối với bộ nhớ, bộ xử lý được tạo cấu hình để thực hiện các lệnh để làm cho thiết bị mã hóa: phân chia hình ảnh thành một hoặc nhiều lát, mỗi lát chứa một hoặc nhiều phiên, mỗi phiên chứa các khối cây tạo mã (CTB); mã hóa cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bai thành dòng bit video khi CTB hiện hành từ các CTB là CTB cuối cùng trong phiên; mã hóa cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai thành dòng bit video khi việc xử lý song song dạng sóng (WPP) được phép và khi CTB hiện hành là CTB cuối cùng trong hàng CTB nhưng không phải CTB cuối cùng trong phiên; và lưu trữ dòng bit video để truyền đến bộ giải mã video.

Thiết bị mã hóa đề xuất các kỹ thuật để ngăn chặn sự sao chép báo hiệu và cân chỉnh bai trong WPP. Bằng cách loại bỏ sự sao chép báo hiệu và cân chỉnh bai trong WPP, số lượng bit được sử dụng để báo hiệu cuối của hàng/phiên và số lượng bit được sử dụng làm đệm được giảm bớt. Nhờ làm giảm số lượng bit cần cho WPP, bộ tạo mã / bộ giải mã (cũng được gọi là “codec”) trong tạo mã video được nâng cao so với các codec hiện hành. Như một vấn đề thực tế, quy trình tạo mã video được nâng cao đem lại cho người dùng trải nghiệm người dùng tốt hơn khi các video được gửi, được nhận, và/hoặc được xem.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất thiết bị mã hóa còn bao gồm bộ truyền được ghép nối với bộ xử lý, bộ truyền được tạo cấu hình để truyền dòng bit video đến bộ giải mã video.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất việc cuối của bit phiên được biểu thị là `end_of_tile_one_bit`, trong đó cuối của bit hàng CTB được biểu thị là `end_of_subset_bit`, và giá trị thứ nhất là một.

Khía cạnh thứ năm đề cập đến máy tạo mã. Máy tạo mã bao gồm bộ nhận được tạo cấu hình để nhận hình ảnh để mã hóa hoặc để nhận dòng bit để giải mã; bộ truyền được ghép nối với bộ nhận, bộ truyền được tạo cấu hình để truyền dòng bit đến bộ giải mã hoặc để truyền ảnh đã giải mã đến bộ hiển thị; bộ nhớ được ghép nối với ít nhất một trong số bộ nhận hoặc bộ truyền, bộ nhớ được tạo cấu hình để lưu trữ các lệnh; và bộ xử lý được ghép nối với bộ nhớ, bộ xử lý được tạo cấu hình để thực hiện các lệnh được lưu trữ trong bộ nhớ để thực hiện phương pháp bất kỳ trong số các phương pháp được bộc lộ ở đây.

Máy tạo mã đề xuất các kỹ thuật để ngăn chặn sự sao chép báo hiệu và cân chỉnh bai trong WPP. Bằng cách loại bỏ sự sao chép báo hiệu và cân chỉnh bai trong WPP, số lượng bit được sử dụng để báo hiệu cuối của hàng/phiên và số lượng bit được sử dụng làm đệm được giảm bớt. Nhờ làm giảm số lượng bit cần cho WPP, bộ tạo mã / bộ giải mã (cũng được gọi là “codec”) trong tạo mã video được nâng cao so với các codec hiện hành. Như một vấn đề thực tế, quy trình tạo mã video được nâng cao đem lại cho người dùng trải nghiệm người dùng tốt hơn khi các video được gửi, được nhận, và/hoặc được xem.

Tùy chọn là, theo khía cạnh bất kỳ trong số các khía cạnh trên đây, một cách thực hiện khác của khía cạnh này đề xuất bộ hiển thị được tạo cấu hình để hiển thị ảnh.

Khía cạnh thứ sáu đề cập đến hệ thống. Hệ thống này bao gồm bộ mã hóa; và bộ giải mã truyền thông với bộ mã hóa, trong đó bộ mã hóa hoặc bộ giải mã bao gồm thiết bị giải mã, thiết bị mã hóa, hoặc máy tạo mã được bộc lộ ở đây.

Hệ thống này đề xuất các kỹ thuật để ngăn chặn sự sao chép báo hiệu và cân chỉnh bai trong WPP. Bằng cách loại bỏ sự sao chép báo hiệu và cân chỉnh bai trong WPP, số lượng bit được sử dụng để báo hiệu cuối của hàng/phiên và số lượng bit được sử dụng làm đệm được giảm bớt. Nhờ làm giảm số lượng bit cần cho WPP, bộ tạo mã / bộ giải mã (cũng được gọi là “codec”) trong tạo mã video được nâng cao so với các codec hiện



hành. Như một vấn đề thực tế, quy trình tạo mã video được nâng cao đem lại cho người dùng trải nghiệm người dùng tốt hơn khi các video được gửi, được nhận, và/hoặc được xem.

Khía cạnh thứ bảy đề cập đến phương tiện tạo mã. Phương tiện tạo mã bao gồm phương tiện nhận được tạo cấu hình để nhận hình ảnh để mã hóa hoặc để nhận dòng bit để giải mã; phương tiện truyền được ghép nối với phương tiện nhận, phương tiện truyền được tạo cấu hình để truyền dòng bit đến phương tiện giải mã hoặc để truyền ảnh đã giải mã đến phương tiện hiển thị; phương tiện lưu trữ được ghép nối với ít nhất một trong số phương tiện nhận hoặc phương tiện truyền, phương tiện lưu trữ được tạo cấu hình để lưu trữ các lệnh; và phương tiện xử lý được ghép nối với phương tiện lưu trữ, phương tiện xử lý được tạo cấu hình để thực hiện các lệnh được lưu trữ trong phương tiện lưu trữ để thực hiện phương pháp bất kỳ trong số các phương pháp được bộc lộ ở đây.

Phương tiện tạo mã này đề xuất các kỹ thuật để ngăn chặn sự sao chép báo hiệu và cân chỉnh bại trong WPP. Bằng cách loại bỏ sự sao chép báo hiệu và cân chỉnh bại trong WPP, số lượng bit được sử dụng để báo hiệu cuối của hàng/phiên và số lượng bit được sử dụng làm đệm được giảm bớt. Nhờ làm giảm số lượng bit cần cho WPP, bộ tạo mã / bộ giải mã (cũng được gọi là “codec”) trong tạo mã video được nâng cao so với các codec hiện hành. Như một vấn đề thực tế, quy trình tạo mã video được nâng cao đem lại cho người dùng trải nghiệm người dùng tốt hơn khi các video được gửi, được nhận, và/hoặc được xem.

### **Mô tả vắn tắt các hình vẽ**

Để hiểu đầy đủ hơn về sáng chế, có thể tham khảo phần mô tả vắn tắt sau đây, kết hợp với các hình vẽ kèm theo và phần mô tả chi tiết, trong đó các số chỉ dẫn giống nhau đại diện cho các phần giống nhau.

Fig.1 là sơ đồ khối minh họa hệ thống tạo mã làm ví dụ mà có thể sử dụng các kỹ thuật tạo mã video.

Fig.2 là sơ đồ khối minh họa bộ mã hóa video làm ví dụ mà có thể thực hiện các kỹ thuật tạo mã video.

Fig.3 là sơ đồ khối minh họa một ví dụ về bộ giải mã video mà có thể thực hiện các kỹ thuật tạo mã video.

Fig.4 minh họa dòng bit video được tạo cấu hình để thực hiện việc xử lý song song mặt sóng.

Fig.5 thể hiện một phương án của phương pháp giải mã dòng bit video được tạo mã.

Fig.6 thể hiện một phương án của phương pháp mã hóa dòng bit video được tạo mã.

Fig.7 là hình vẽ giản lược của thiết bị tạo mã video.

Fig.8 là hình vẽ giản lược của một phương án của phương tiện tạo mã.

### **Mô tả chi tiết sáng chế**

Nên được hiểu 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 lượng kỹ thuật bất kỳ, liệu hiện đã được biết đến hay đang tồn tại. Sáng chế sẽ hoàn toàn không bị giới hạn ở các cách thực hiện minh họa, các hình vẽ, và các kỹ thuật được minh họa dưới đây, bao gồm các thiết kế và các cách thực hiện làm ví dụ được minh họa và được mô tả ở đây, mà có thể được cải biến nằm trong phạm vi của yêu cầu bảo hộ kèm theo cùng với phạm vi tương đương đầy đủ của chúng.

Các thuật ngữ sau đây được định nghĩa như sau trừ khi được sử dụng trong ngữ cảnh ngược lại ở đây. Cụ thể, các định nghĩa sau đây được dự định để làm rõ ràng thêm nữa cho sáng chế. Tuy nhiên, các thuật ngữ này có thể được mô tả khác nhau trong các ngữ cảnh khác nhau. Theo đó, các định nghĩa sau đây sẽ được xem xét như là sự bổ sung và sẽ không được coi là giới hạn các định nghĩa khác bất kỳ của các phần mô tả được tạo cho cho các thuật ngữ này ở đây.

Dòng bit là một chuỗi các bit bao gồm dữ liệu mà được nén để truyền giữa bộ mã hóa và bộ giải mã. Bộ mã hóa là thiết bị mà được tạo cấu hình để sử dụng các quy trình mã hóa để nén dữ liệu video thành dòng bit. Bộ giải mã là thiết bị mà được tạo cấu hình để sử dụng các quy trình giải mã để tái tạo dữ liệu video từ dòng bit để hiển thị. Hình

ảnh là ảnh hoàn chỉnh mà được dự định để hiển thị hoàn toàn hoặc một phần cho người dùng tại thời điểm tương ứng trong chuỗi video. Hình ảnh tham chiếu là hình ảnh chứa các mẫu tham chiếu mà có thể được sử dụng khi tạo mã các hình ảnh khác bằng cách tham chiếu theo dự đoán liên ảnh. Hình ảnh được tạo mã là sự đại diện của hình ảnh mà được tạo mã theo dự đoán liên ảnh hoặc dự đoán trong ảnh, có trong đơn vị truy cập riêng lẻ trong dòng bit, và chứa tập hợp hoàn chỉnh của các đơn vị cây tạo mã (các CTU) của hình ảnh. Một lát là một phần chia của hình ảnh mà chứa số lượng nguyên của các phiến hoàn chỉnh hoặc số lượng nguyên của các hàng CTU hoàn chỉnh liên tiếp trong phiến của hình ảnh, trong đó lát và tất cả các sự chia nhỏ được bao gồm một cách riêng biệt trong một đơn vị lớp trừu tượng mạng (network abstraction layer, NAL) đơn lẻ. Lát tham chiếu là lát của hình ảnh tham chiếu mà chứa các mẫu tham chiếu hoặc được sử dụng khi tạo mã các lát khác bằng cách tham chiếu theo dự đoán liên ảnh. Phần đầu lát là một phần của lát được tạo mã chứa các phần tử dữ liệu liên quan đến tất cả các phiến hoặc các hàng CTU trong phiến được đại diện trong lát. Điểm nhập là vị trí bit trong dòng bit chứa bit thứ nhất của dữ liệu video đối với tập con tương ứng của lát được tạo mã. Độ lệch là khoảng cách về bit giữa vị trí bit đã biết và điểm nhập. Tập con là sự chia nhỏ của tập hợp, như phiến, hàng CTU, hoặc CTU. CTU là tập con của lát. Đơn vị cây tạo mã (CTU) là nhóm của các mẫu có kích thước được xác định trước mà có thể được phân chia bởi cây tạo mã. Các CTU được chia đôi với mỗi thành phần độ chói/sắc độ thành các khối cây tạo mã (coding tree block, CTB). CTB có thể là  $64 \times 64$ ,  $32 \times 32$ , hoặc  $16 \times 16$  với kích thước khối điểm ảnh lớn hơn thường làm tăng hiệu suất tạo mã. Các CTB sau đó được chia thành một hoặc nhiều đơn vị tạo mã (coding unit, CU), sao cho kích thước CTU cũng là kích thước đơn vị tạo mã lớn nhất.

Hàng CTU là nhóm của các CTU mà kéo dài theo chiều ngang giữa biên lát bên trái và biên lát bên phải. Hàng CTB là nhóm của các CTB mà kéo dài theo chiều ngang giữa biên lát bên trái và biên lát bên phải. Cột CTU là nhóm của các CTU mà kéo dài theo chiều thẳng đứng giữa biên lát trên cùng và biên lát dưới cùng. Cột CTB là nhóm của các CTB mà kéo dài theo chiều thẳng đứng giữa biên lát trên cùng và biên lát dưới cùng. Cuối của bit hàng CTB là bit ở cuối của hàng CTB. Các bit cân chỉnh bại là các bit được thêm vào cuối của tập con dữ liệu, hàng CTU, hàng CTB, phiến, v.v., làm đệm. Các bit cân chỉnh bại có thể được sử dụng để xử lý hoặc bù trễ được tạo ra bởi WPP.

WPP là cơ chế tạo mã các hàng CTU của lát với trể để cho phép mỗi hàng được giải mã song song bởi các tuyến đoạn (thread) khác nhau. Địa chỉ lát là vị trí nhận dạng được của lát hoặc phân con của nó.

Các từ viết tắt sau đây được sử dụng ở đây: khối cây tạo mã (Coding Tree Block, CTB), đơn vị cây tạo mã (Coding Tree Unit, CTU), đơn vị tạo mã (Coding Unit, CU), chuỗi video được tạo mã (Coded Video Sequence, CVS), nhóm chuyên gia video chung (Joint Video Experts Team, JVET), tập hợp phiên ràng buộc chuyển động (Motion-Constrained Tile Set, MCTS), đơn vị truyền tối đa (Maximum Transfer Unit, MTU), lớp trừu tượng mạng (Network Abstraction Layer, NAL), đếm thứ tự hình ảnh (Picture Order Count, POC), phần tải dữ liệu chuỗi byte thô (Raw Byte Sequence Payload, RBSP), tập hợp thông số chuỗi (Sequence Parameter Set, SPS), đơn vị hình ảnh con (Sub-Picture Unit, SPU), tạo mã video đa năng (Versatile Video Coding, VVC), và bản phác thảo công việc (Working Draft, WD).

Fig.1 là sơ đồ khối minh họa hệ thống tạo mã 10 làm ví dụ mà có thể sử dụng các kỹ thuật tạo mã video như được mô tả ở đây. Như được thể hiện trên Fig.1, hệ thống tạo mã 10 bao gồm thiết bị nguồn 12 để cung cấp dữ liệu video được mã hóa để được giải mã vào thời điểm muộn hơn bởi thiết bị đích 14. Cụ thể, thiết bị nguồn 12 có thể cung cấp dữ liệu video cho thiết bị đích 14 qua phương tiện đọc được bằng máy tính 16. Thiết bị nguồn 12 và thiết bị đích 14 có thể bao gồm thiết bị bất kỳ trong phạm vi thiết bị rộng, bao gồm các máy tính để bàn, các máy tính xách tay loại notebook (ví dụ, máy tính xách tay loại laptop), các máy tính bảng, các thiết bị giải mã tín hiệu và truyền dữ liệu (set-top box), các máy điện thoại như các điện thoại "thông minh", các bàn điều khiển (pad) "thông minh", các máy thu hình, các camera, các thiết bị hiển thị, các máy phát phương tiện số, các bàn giao tiếp trò chơi video, thiết bị tạo luồng video, hoặc tương tự. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể được trang bị để truyền thông vô tuyến.

Thiết bị đích 14 có thể nhận dữ liệu video được mã hóa để được giải mã qua phương tiện đọc được bằng máy tính 16. Phương tiện đọc được bằng máy tính 16 có thể bao gồm loại phương tiện hoặc thiết bị bất kỳ có khả năng chuyển dữ liệu video được mã hóa từ thiết bị nguồn 12 đến thiết bị đích 14. Theo một ví dụ, phương tiện đọc được

bằng máy tính 16 có thể bao gồm phương tiện truyền thông để cho phép thiết bị nguồn 12 truyền dữ liệu video được mã hóa một cách trực tiếp đến thiết bị đích 14 theo thời gian thực. Dữ liệu video được mã hóa có thể được điều biến theo tiêu chuẩn truyền thông, như giao thức truyền thông vô tuyến, và được truyền đến thiết bị đích 14. Phương tiện truyền thông có thể bao gồm phương tiện truyền thông vô tuyến hoặc hữu tuyến bất kỳ, như phổ tần số radiô (RF) hoặc một hoặc nhiều đường truyền vật lý. Phương tiện truyền thông có thể tạo thành một phần của mạng dựa trên gói, như mạng cục bộ, mạng diện rộng, hoặc mạng toàn cầu như Internet. Phương tiện truyền thông có thể bao gồm các bộ định tuyến, các bộ chuyển mạch, các trạm gốc, hoặc thiết bị khác bất kỳ mà có thể là hữu ích để hỗ trợ truyền thông từ thiết bị nguồn 12 đến thiết bị đích 14.

Trong một số ví dụ, dữ liệu được mã hóa có thể được kết xuất từ giao diện đầu ra 22 đến thiết bị lưu trữ. Tương tự, dữ liệu được mã hóa có thể được truy cập từ thiết bị lưu trữ bởi giao diện đầu vào. Thiết bị lưu trữ có thể bao gồm loại phương tiện bất kỳ trong số các loại phương tiện lưu trữ dữ liệu truy cập cục bộ hoặc phân tán như ổ cứng, các đĩa Blu-ray, các đĩa video số (digital video disk, DVD), các bộ nhớ chỉ đọc dạng đĩa compac (CD-ROM), bộ nhớ tác động nhanh (flash memory), bộ nhớ khả biến hoặc bất khả biến, hoặc phương tiện lưu trữ kỹ thuật số thích hợp khác bất kỳ để lưu trữ dữ liệu video được mã hóa. Trong ví dụ thêm nữa, thiết bị lưu trữ có thể tương ứng với máy chủ tệp hoặc thiết bị lưu trữ trung gian khác mà có thể lưu trữ video đã mã hóa được tạo ra bởi thiết bị nguồn 12. Thiết bị đích 14 có thể truy cập dữ liệu video được lưu trữ từ thiết bị lưu trữ qua tạo luồng hoặc tải xuống. Máy chủ tệp có thể là loại máy chủ bất kỳ có khả năng lưu trữ dữ liệu video được mã hóa và truyền dữ liệu video được mã hóa đó đến thiết bị đích 14. Các máy chủ tệp làm ví dụ bao gồm máy chủ web (ví dụ, đối với vị trí web), máy chủ giao thức truyền tệp (file transfer protocol, FTP), các thiết bị lưu trữ gắn vào mạng (network attached storage, NAS), hoặc ổ đĩa cục bộ. Thiết bị đích 14 có thể truy cập dữ liệu video được mã hóa qua kết nối dữ liệu chuẩn bất kỳ, bao gồm kết nối Internet. Điều này có thể bao gồm kênh vô tuyến (ví dụ, kết nối Wi-Fi), kết nối hữu tuyến (ví dụ, đường thuê bao số (digital subscriber line, DSL), modem cáp, v.v.), hoặc sự kết hợp của cả hai loại mà là thích hợp để truy cập dữ liệu video được mã hóa được lưu trữ trên máy chủ tệp. Hoạt động truyền dữ liệu video được mã hóa từ thiết bị lưu trữ

có thể là hoạt động truyền tạo luồng, hoạt động truyền tải xuống, hoặc sự kết hợp của chúng.

Các kỹ thuật của sáng chế không nhất thiết phải bị giới hạn ở các ứng dụng hoặc các thiết đặt vô tuyến. Các kỹ thuật này có thể được áp dụng vào tạo mã video với sự hỗ trợ của ứng dụng đa phương tiện bất kỳ trong số nhiều loại ứng dụng đa phương tiện, như phát rộng vô tuyến truyền hình qua không trung, truyền vô tuyến truyền hình cáp, truyền vô tuyến truyền hình vệ tinh, truyền video tạo luồng Internet, như tạo luồng thích ứng động qua HTTP (dynamic adaptive streaming over HTTP, DASH), video số mà được mã hóa trên phương tiện lưu trữ dữ liệu, giải mã video số được lưu trữ trên phương tiện lưu trữ dữ liệu, hoặc các ứng dụng khác. Trong một số ví dụ, hệ thống tạo mã 10 có thể được tạo cấu hình để hỗ trợ truyền video một-hướng hoặc hai-hướng để hỗ trợ các ứng dụng như tạo luồng video, phát lại video, phát rộng video, và/hoặc điện thoại video.

Trong ví dụ trên Fig.1, thiết bị nguồn 12 bao gồm nguồn video 18, bộ mã hóa video 20, và giao diện đầu ra 22. Thiết bị đích 14 bao gồm giao diện đầu vào 28, bộ giải mã video 30, và thiết bị hiển thị 32. Theo sáng chế, bộ mã hóa video 20 của thiết bị nguồn 12 và/hoặc bộ giải mã video 30 của thiết bị đích 14 có thể được tạo cấu hình để áp dụng các kỹ thuật để tạo mã video. Trong các ví dụ khác, thiết bị nguồn và thiết bị đích có thể bao gồm các thành phần hoặc các cách bố trí khác. Ví dụ, thiết bị nguồn 12 có thể nhận dữ liệu video từ nguồn video ngoài, như camera ngoài. Tương tự, thiết bị đích 14 có thể giao tiếp với thiết bị hiển thị ngoài, ngoài việc bao gồm thiết bị hiển thị được tích hợp.

Hệ thống tạo mã được minh họa 10 trên Fig.1 chỉ là một ví dụ. Các kỹ thuật để tạo mã video có thể được thực hiện bởi thiết bị mã hóa và/hoặc giải mã video số bất kỳ. Mặc dù các kỹ thuật của sáng chế thường được thực hiện bởi thiết bị tạo mã video, nhưng các kỹ thuật này cũng có thể được thực hiện bởi bộ mã hóa/bộ giải mã video, thường được gọi là “CODEC”. Hơn nữa, các kỹ thuật của sáng chế cũng có thể được thực hiện bởi bộ tiền xử lý video. Bộ mã hóa video và/hoặc bộ giải mã có thể là bộ phận xử lý đồ họa (GPU) hoặc thiết bị tương tự.

Thiết bị nguồn 12 và thiết bị đích 14 chỉ là các ví dụ của các thiết bị tạo mã như vậy trong đó thiết bị nguồn 12 tạo ra dữ liệu video được tạo mã để truyền đến thiết bị

đích 14. Trong một số ví dụ, thiết bị nguồn 12 và thiết bị đích 14 có thể hoạt động theo cách gần như đối xứng sao cho mỗi thiết bị trong số thiết bị nguồn 12 và thiết bị đích 14 bao gồm các thành phần mã hóa và giải mã video. Vì vậy, hệ thống tạo mã 10 có thể hỗ trợ truyền video một-hướng hoặc hai-hướng giữa các thiết bị video 12, 14, ví dụ, để tạo luồng video, phát lại video, phát rộng video, hoặc điện thoại video.

Nguồn video 18 của thiết bị nguồn 12 có thể bao gồm thiết bị thu video, như camera video, kho trữ video chứa video thu được trước đó, và/hoặc giao diện cung cấp video để nhận video từ bên cung cấp nội dung video. Theo một cách khác nữa, nguồn video 18 có thể tạo dữ liệu dựa trên đồ họa máy tính làm video nguồn, hoặc sự kết hợp của video trực tiếp, video lưu trữ, và video được tạo ra bởi máy tính.

Trong một số trường hợp, khi nguồn video 18 là camera video, thiết bị nguồn 12 và thiết bị đích 14 có thể tạo thành các điện thoại camera hoặc các điện thoại video. Tuy nhiên, như đã nêu trên, các kỹ thuật được mô tả trong sáng chế có thể áp dụng được vào tạo mã video nói chung, và có thể được áp dụng vào các ứng dụng vô tuyến và/hoặc hữu tuyến. Trong từng trường hợp, video thu được, video thu được từ trước, hoặc video được tạo ra bởi máy tính có thể được mã hóa bởi bộ mã hóa video 20. Thông tin video đã mã hóa sau đó có thể được kết xuất bởi giao diện đầu ra 22 vào phương tiện đọc được bằng máy tính 16.

Phương tiện đọc được bằng máy tính 16 có thể bao gồm phương tiện nhất thời, như phát rộng vô tuyến hoặc truyền mạng hữu tuyến, hoặc phương tiện lưu trữ (tức là, phương tiện lưu trữ lâu dài), như đĩa cứng, ổ tác động nhanh, đĩa compac, đĩa video số, đĩa Blu-ray, hoặc phương tiện đọc được bằng máy tính khác. Trong một số ví dụ, mạng máy chủ (không được thể hiện trên hình vẽ) có thể nhận dữ liệu video được mã hóa từ thiết bị nguồn 12 và cung cấp dữ liệu video được mã hóa cho thiết bị đích 14, ví dụ, qua hoạt động truyền mạng. Tương tự, thiết bị tính toán của nhà máy sản xuất phương tiện, như nhà máy dập khuôn đĩa, có thể nhận dữ liệu video được mã hóa từ thiết bị nguồn 12 và tạo ra đĩa chứa dữ liệu video được mã hóa. Do đó, phương tiện đọc được bằng máy tính 16 có thể được hiểu là bao gồm một hoặc nhiều phương tiện đọc được bằng máy tính ở các dạng khác nhau, trong các ví dụ khác nhau.

Giao diện đầu vào 28 của thiết bị đích 14 nhận thông tin từ phương tiện đọc được bằng máy tính 16. Thông tin của phương tiện đọc được bằng máy tính 16 có thể bao gồm thông tin cú pháp được xác định bởi bộ mã hóa video 20, mà cũng được sử dụng bởi bộ giải mã video 30, mà bao gồm các phần tử cú pháp mô tả các đặc tính và/hoặc các hoạt động xử lý các khối và các đơn vị được tạo mã khác, ví dụ, nhóm các hình ảnh (group of pictures, GOP). Thiết bị hiển thị 32 hiển thị dữ liệu video được giải mã cho người dùng, và có thể bao gồm thiết bị hiển thị bất kỳ trong số nhiều loại thiết bị hiển thị như ống tia catốt (CRT), bộ hiển thị tinh thể lỏng (LCD), bộ hiển thị plasma, bộ hiển thị diot phát quang hữu cơ (OLED), hoặc loại thiết bị hiển thị khác.

Bộ mã hóa video 20 và bộ giải mã video 30 có thể hoạt động theo tiêu chuẩn tạo mã video, như tiêu chuẩn tạo mã video hiệu suất cao (High Efficiency Video Coding, HEVC) hiện đang sự phát triển, và có thể phù hợp với mô hình thử nghiệm HEVC (HM). Theo cách khác, bộ mã hóa video 20 và bộ giải mã video 30 có thể hoạt động theo các tiêu chuẩn độc quyền hoặc công nghiệp khác, như tiêu chuẩn H.264 của lĩnh vực chuẩn hóa viễn thông của hiệp hội viễn thông quốc tế (ITU-T), theo cách khác được gọi là nhóm chuyên gia hình ảnh động (Moving Picture Expert Group, MPEG)-4, phần 10, tạo mã video tiên tiến (AVC), H.265/HEVC, hoặc các sự mở rộng của các tiêu chuẩn này. Tuy nhiên, các kỹ thuật của sáng chế không bị giới hạn ở tiêu chuẩn tạo mã cụ thể bất kỳ. Các ví dụ khác về các tiêu chuẩn tạo mã video bao gồm MPEG-2 và ITU-T H.263. Mặc dù không được thể hiện trên Fig.1, nhưng theo một số khía cạnh, bộ mã hóa video 20 và bộ giải mã video 30, mỗi bộ có thể mỗi được tích hợp với bộ mã hóa và bộ giải mã audio, và có thể bao gồm các bộ phận dồn kênh-bộ phận phân kênh (MUX-DEMUX) thích hợp, hoặc phần cứng và phần mềm khác, để xử lý việc mã hóa cả audio lẫn video trong một luồng dữ liệu chung hoặc các luồng dữ liệu riêng rẽ. Nếu áp dụng được, các bộ phận MUX-DEMUX có thể phù hợp với giao thức bộ dồn kênh ITU H.223, hoặc các giao thức khác như giao thức gói dữ liệu người dùng (user datagram protocol, UDP).

Bộ mã hóa video 20 và bộ giải mã video 30, mỗi bộ có thể được thực thi như là loại bất kỳ trong số nhiều loại mạch bộ mã hóa thích hợp, như một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (digital signal processor, DSP), mạch tích hợp chuyên dụng (application specific integrated circuit, ASIC), mảng công lập trình được dạng trường



(field programmable gate array, FPGA), bộ logic rời rạc, phần mềm, phần cứng, phần sụn hoặc các sự kết hợp bất kỳ của chúng. Khi các kỹ thuật được thực hiện một phần trong phần mềm, thì thiết bị có thể lưu trữ các lệnh đối với phần mềm trong phương tiện lâu dài đọc được bằng máy tính thích hợp và thực hiện các lệnh trong phần cứng nhờ sử dụng một hoặc nhiều bộ xử lý để thực hiện các kỹ thuật của sáng chế. Mỗi bộ trong số bộ mã hóa video 20 và bộ giải mã video 30 có thể có trong một hoặc nhiều bộ mã hóa hoặc bộ giải mã, mỗi loại có thể được tích hợp như là một phần của bộ mã hóa/bộ giải mã kết hợp (CODEC - CODEC) trong thiết bị tương ứng. Thiết bị bao gồm bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể bao gồm mạch tích hợp, bộ vi xử lý, và/hoặc thiết bị truyền thông vô tuyến, như điện thoại mạng dạng ô.

Fig.2 là sơ đồ khối minh họa một ví dụ về bộ mã hóa video 20 mà có thể thực hiện các kỹ thuật tạo mã video. Bộ mã hóa video 20 có thể thực hiện tạo mã trong ảnh và liên ảnh của các khối video trong các lát video. Tạo mã trong ảnh dựa trên dự đoán không gian để làm giảm hoặc loại bỏ dư thừa không gian trong video trong khung video hoặc hình ảnh đã cho. Tạo mã liên ảnh dựa trên dự đoán thời gian để làm giảm hoặc loại bỏ dư thừa thời gian trong video trong các khung liên kế hoặc các hình ảnh của chuỗi video. Chế độ trong ảnh (chế độ I) có thể liên quan đến chế độ bất kỳ trong số một số chế độ tạo mã dựa trên không gian. Các chế độ liên ảnh, như dự đoán đơn hướng (cũng được gọi là dự đoán đơn) (chế độ P) hoặc song dự đoán (cũng được gọi là dự đoán song) (chế độ B), có thể liên quan đến chế độ bất kỳ trong số một số chế độ tạo mã dựa trên thời gian.

Như được thể hiện trên Fig.2, bộ mã hóa video 20 nhận khối video hiện hành trong khung video cần được mã hóa. Trong ví dụ trên Fig.2, bộ mã hóa video 20 bao gồm bộ phận chọn chế độ 40, bộ nhớ khung tham chiếu 64, bộ cộng 50, bộ phận xử lý biến đổi 52, bộ phận lượng tử hóa 54, và bộ phận tạo mã entropi 56. Bộ phận chọn chế độ 40 bao gồm bộ phận bù chuyển động 44, bộ phận ước lượng chuyển động 42, bộ phận dự đoán trong ảnh (cũng được gọi là dự đoán-trong ảnh) 46, và bộ phận phân chia 48. Đối với việc tái tạo khối video, bộ mã hóa video 20 cũng bao gồm bộ phận lượng tử hóa ngược 58, bộ phận biến đổi ngược 60, và bộ cộng 62. Bộ lọc giải khối (không được thể hiện trên Fig.2) cũng có thể được bao gồm để lọc các biên khối để loại bỏ các vật tạo tác dạng khối từ video được tái tạo. Nếu muốn, bộ lọc giải khối sẽ thường lọc đầu ra của

bộ cộng 62. Các bộ lọc bỏ sung (nội vòng hoặc sau vòng) cũng có thể được sử dụng ngoài bộ lọc giải khối. Các bộ lọc này không được thể hiện trên hình vẽ cho ngắn gọn, nhưng nếu muốn, có thể lọc đầu ra của bộ cộng 50 (như là bộ lọc nội vòng).

Trong suốt quá trình mã hóa, bộ mã hóa video 20 nhận khung video hoặc lát cắt được tạo mã. Khung hoặc lát có thể được chia thành nhiều khối video. Bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 thực hiện việc tạo mã dự đoán liên ảnh đối với khối video nhận được so với một hoặc nhiều khối trong một hoặc nhiều khung tham chiếu để cung cấp dự đoán thời gian. Bộ phận dự đoán trong ảnh 46 theo cách khác có thể thực hiện việc tạo mã dự đoán trong ảnh đối với khối video nhận được so với một hoặc nhiều khối lân cận trong cùng khung hoặc lát như là khối cần được tạo mã để cung cấp dự đoán không gian. Bộ mã hóa video 20 có thể thực hiện nhiều sự cho phép tạo mã, ví dụ, để chọn chế độ tạo mã thích hợp cho mỗi khối của dữ liệu video.

Hơn nữa, bộ phận phân chia 48 có thể phân chia các khối của dữ liệu video thành các khối con, dựa trên việc đánh giá các sơ đồ phân chia trước đó trong các sự cho phép tạo mã trước đó. Ví dụ, bộ phận phân chia 48 có thể phân chia ban đầu khung hoặc lát thành các đơn vị tạo mã lớn nhất (largest coding unit, LCU), và phân chia mỗi trong số các LCU này thành các đơn vị tạo mã con (các CU con) dựa trên phân tích tỷ lệ-méo (ví dụ, tối ưu hóa tỷ lệ-méo). Bộ phận chọn chế độ 40 còn có thể tạo ra cấu trúc dữ liệu cây tứ phân chỉ báo sự phân chia LCU thành các CU con. Các CU nút lá của cây tứ phân có thể bao gồm một hoặc nhiều đơn vị dự đoán (prediction unit, PU) và một hoặc nhiều đơn vị biến đổi (transform unit, TU).

Sáng chế sử dụng thuật ngữ “khối” để đề cập đến loại bất kỳ trong số CU, PU, hoặc TU, trong ngữ cảnh của HEVC, hoặc các cấu trúc dữ liệu tương tự trong ngữ cảnh của các tiêu chuẩn khác (ví dụ, các khối macrô và các khối con của nó trong H.264/AVC). CU bao gồm nút tạo mã, các PU, và các TU được liên kết với nút tạo mã. Kích thước của CU tương ứng với kích thước của nút tạo mã và có hình vuông. Kích thước của CU có thể nằm trong khoảng từ  $8 \times 8$  điểm ảnh đến kích thước của khối cây với tối đa là  $64 \times 64$  điểm ảnh hoặc lớn hơn. Mỗi CU có thể chứa một hoặc nhiều PU và một hoặc nhiều TU. Dữ liệu cú pháp được liên kết với CU có thể mô tả, ví dụ, việc phân chia CU thành một hoặc nhiều PU. Các chế độ phân chia có thể khác nhau giữa việc liệu

CU là chế độ bỏ qua hay chế độ trực tiếp được mã hóa, chế độ dự đoán trong ảnh được mã hóa, hoặc chế độ dự đoán liên ảnh (cũng được gọi là dự đoán-liên ảnh) được mã hóa. Các PU có thể được phân chia để có hình dạng không phải hình vuông. Dữ liệu cú pháp được liên kết với CU cũng có thể mô tả, ví dụ, việc phân chia CU thành một hoặc nhiều TU theo cây tứ phân. TU có thể là hình vuông hoặc hình không vuông (ví dụ, hình chữ nhật).

Bộ phận chọn chế độ 40 có thể chọn một trong số các chế độ tạo mã, trong ảnh hoặc liên ảnh, ví dụ, dựa trên các kết quả sai số, và cung cấp khối được tạo mã liên ảnh hoặc trong ảnh tạo thành cho bộ cộng 50 để tạo dữ liệu khối dư và cho bộ cộng 62 để tái tạo khối được mã hóa để sử dụng làm khung tham chiếu. Bộ phận chọn chế độ 40 cũng cung cấp các phần tử cú pháp, như các vector chuyển động, các bộ chỉ báo chế độ trong ảnh, thông tin phân chia, và thông tin cú pháp khác, cho bộ phận tạo mã entropi 56.

Bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 có thể được tích hợp ở mức cao, nhưng được minh họa riêng rẽ nhằm các mục đích liên quan đến khái niệm. Ước lượng chuyển động, được thực hiện bởi bộ phận ước lượng chuyển động 42, là quy trình tạo các vector chuyển động, mà ước lượng chuyển động đối với các khối video. Vector chuyển động, ví dụ, có thể chỉ báo sự dịch chuyển của PU của khối video trong khung video hoặc hình ảnh hiện hành so với khối dự đoán trong khung tham chiếu (hoặc đơn vị tạo mã khác) so với khối hiện hành đang được tạo mã trong khung hiện hành (hoặc đơn vị tạo mã khác). Khối dự đoán là khối mà được thấy là khớp gần với khối cần được tạo mã, trên phương diện chênh lệch điểm ảnh, mà có thể được xác định bởi tổng sai phân tuyệt đối (SAD), tổng sai phân bình phương (SSD), hoặc các chuẩn đo chênh lệch khác. Trong một số ví dụ, bộ mã hóa video 20 có thể tính toán các giá trị đối với các vị trí điểm ảnh nguyên con của các hình ảnh tham chiếu được lưu trữ trong bộ nhớ khung tham chiếu 64. Ví dụ, bộ mã hóa video 20 có thể nội suy các giá trị 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 hình ảnh tham chiếu. Do đó, bộ phận ước lượng chuyển động 42 có thể thực hiện việc tìm kiếm chuyển động so với các vị trí điểm ảnh đầy đủ và các vị trí điểm ảnh phân số và kết xuất vector chuyển động với độ chính xác điểm ảnh thập phân.

Bộ phận ước lượng chuyển động 42 tính toán vector chuyển động đối với PU của khối video trong lát được tạo mã liên ảnh bằng cách so sánh vị trí của PU với vị trí của khối dự đoán của hình ảnh tham chiếu. Hình ảnh tham chiếu có thể được chọn từ danh sách hình ảnh tham chiếu thứ nhất (List 0) hoặc danh sách hình ảnh tham chiếu thứ hai (List 1), mỗi danh sách trong số đó nhận dạng một hoặc nhiều hình ảnh tham chiếu được lưu trữ trong bộ nhớ khung tham chiếu 64. Bộ phận ước lượng chuyển động 42 gửi vector chuyển động tính toán được đến bộ phận mã hóa entropi 56 và bộ phận bù chuyển động 44.

Bù chuyển động, được thực hiện bởi bộ phận bù chuyển động 44, có thể bao gồm việc tìm hoặc tạo khối dự đoán dựa trên vector chuyển động được xác định bởi bộ phận ước lượng chuyển động 42. Mặt khác, bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 có thể được tích hợp về mặt chức năng trong một số ví dụ. Khi nhận vector chuyển động đối với PU của khối video hiện hành, bộ phận bù chuyển động 44 có thể định vị khối dự đoán mà vector chuyển động trỏ đến trong một trong số các danh sách hình ảnh tham chiếu. Bộ cộng 50 tạo ra khối video dư bằng cách trừ các giá trị điểm ảnh của khối dự đoán từ các giá trị điểm ảnh của khối video hiện hành được tạo mã, tạo thành các giá trị chênh lệch điểm ảnh, như được thảo luận dưới đây. Nói chung, bộ phận ước lượng chuyển động 42 thực hiện ước lượng chuyển động so với các thành phần độ chói, và bộ phận bù chuyển động 44 sử dụng các vector chuyển động được tính toán dựa trên các thành phần độ chói đối với cả các thành phần sắc độ lẫn các thành phần độ chói. Bộ phận chọn chế độ 40 cũng có thể tạo các phần tử cú pháp được liên kết với các khối video và lát video để sử dụng bởi bộ giải mã video 30 khi giải mã các khối video của lát video.

Bộ phận dự đoán trong ảnh 46 có thể dự đoán nội bộ khối hiện hành, như là một sự lựa chọn đối với việc dự đoán liên ảnh được thực hiện bởi bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44, như được mô tả trên đây. Cụ thể, bộ phận dự đoán trong ảnh 46 có thể xác định chế độ dự đoán trong ảnh để sử dụng để mã hóa khối hiện hành. Trong một số ví dụ, bộ phận dự đoán trong ảnh 46 có thể mã hóa khối hiện hành sử dụng các chế độ dự đoán trong ảnh khác nhau, ví dụ, trong suốt quá trình cho phép mã hóa riêng rẽ, và bộ phận dự đoán trong ảnh 46 (hoặc bộ phận chọn chế độ 40,

trong một số ví dụ) có thể chọn chế độ dự đoán trong ảnh thích hợp để sử dụng từ các chế độ đã kiểm tra.

Ví dụ, bộ phận dự đoán trong ảnh 46 có thể tính toán các giá trị tỷ lệ méo sử dụng phân tích tỷ lệ méo đối với các chế độ dự đoán trong ảnh đã kiểm tra khác nhau, và chọn chế độ dự đoán trong ảnh có các đặc tính tỷ lệ méo tốt nhất trong số các chế độ đã kiểm tra. Việc phân tích tỷ lệ méo xác định chung lượng méo (hoặc sai số) giữa khối được mã hóa và khối chưa được mã hóa gốc mà đã được mã hóa để tạo ra khối được mã hóa, cũng như tốc độ bit (tức là, số lượng bit) được sử dụng để tạo khối được mã hóa. Bộ phận dự đoán trong ảnh 46 có thể tính toán các tỷ lệ từ các phần méo và các tỷ lệ đối với các khối được mã hóa khác nhau để xác định chế độ dự đoán trong ảnh nào thể hiện giá trị tỷ lệ méo tốt nhất đối với khối.

Ngoài ra, bộ phận dự đoán trong ảnh 46 có thể được tạo cấu hình để tạo mã các khối độ sâu của bản đồ độ sâu sử dụng chế độ tạo mô hình độ sâu (depth modeling mode, DMM). Bộ phận chọn chế độ 40 có thể xác định liệu chế độ DMM khả dụng tạo ra các kết quả tạo mã tốt hơn so với chế độ dự đoán trong ảnh và các chế độ DMM khác, ví dụ, sử dụng tối ưu hóa tỷ lệ-méo (rate-distortion optimization, RDO). Dữ liệu cho ảnh bố cục tương ứng với bản đồ độ sâu có thể được lưu trữ trong bộ nhớ khung tham chiếu 64. Bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 cũng có thể được tạo cấu hình để dự đoán liên ảnh các khối độ sâu của bản đồ độ sâu.

Sau khi chọn chế độ dự đoán trong ảnh đối với khối (ví dụ, chế độ dự đoán trong ảnh hoặc một trong số các chế độ DMM), bộ phận dự đoán trong ảnh 46 có thể cung cấp thông tin chỉ báo chế độ dự đoán trong ảnh được chọn đối với khối cho bộ phận tạo mã entropi 56. Bộ phận tạo mã entropi 56 có thể mã hóa thông tin chỉ báo chế độ dự đoán trong ảnh được chọn. Bộ mã hóa video 20 có thể bao gồm trong dữ liệu cấu hình dòng bit được truyền, mà có thể bao gồm các bảng chỉ số chế độ dự đoán trong ảnh và các bảng chỉ số chế độ dự đoán trong ảnh được thay đổi (cũng được gọi là các bảng ánh xạ từ mã), các định nghĩa về các ngữ cảnh mã hóa đối với các khối khác nhau, và các chỉ báo của chế độ dự đoán trong ảnh có khả năng nhất, bảng chỉ số chế độ dự đoán trong ảnh, và bảng chỉ số chế độ dự đoán trong ảnh thay đổi để sử dụng cho mỗi ngữ cảnh trong số các ngữ cảnh này.

Bộ mã hóa video 20 tạo thành khối video dư bằng cách trừ dữ liệu dự đoán từ bộ phận chọn chế độ 40 từ khối video gốc đang được tạo mã. Bộ cộng 50 đại diện cho thành phần hoặc các thành phần thực hiện phép toán trừ này.

Bộ phận xử lý biến đổi 52 áp dụng biến đổi, như biến đổi cosin rời rạc (DCT) hoặc biến đổi tương tự về mặt khái niệm, vào khối dư, tạo ra khối video bao gồm các giá trị hệ số biến đổi dư. Bộ phận xử lý biến đổi 52 có thể thực hiện các sự biến đổi khác mà tương tự về mặt khái niệm với DCT. Biến đổi sóng nhỏ (Wavelet), biến đổi nguyên, biến đổi dải con hoặc các loại biến đổi khác cũng có thể được sử dụng.

Bộ phận xử lý biến đổi 52 áp dụng biến đổi vào khối dư, tạo ra khối của các hệ số biến đổi dư. Việc biến đổi có thể chuyển đổi thông tin còn lại từ miền giá trị điểm ảnh sang miền biến đổi, như miền tần số. Bộ phận xử lý biến đổi 52 có thể gửi các hệ số biến đổi tạo thành đến bộ phận lượng tử hóa 54. Bộ phận lượng tử hóa 54 lượng tử hóa các hệ số biến đổi để làm giảm tốc độ bit thêm nữa. Quy trình lượng tử hóa có thể làm giảm độ sâu bit được liên kết với một số hoặc tất cả hệ số. Mức độ lượng tử hóa có thể được thay đổi bằng cách điều chỉnh thông số lượng tử hóa. Trong một số ví dụ, bộ phận lượng tử hóa 54 sau đó có thể thực hiện việc quét ma trận bao gồm các hệ số biến đổi được lượng tử hóa. Theo cách khác, bộ phận mã hóa entropi 56 có thể thực hiện việc quét.

Tiếp theo việc lượng tử hóa, bộ phận tạo mã entropi 56 tạo mã entropi các hệ số biến đổi được lượng tử hóa. Ví dụ, bộ phận tạo mã entropi 56 có thể thực hiện tạo mã độ dài biến số thích ứng ngữ cảnh (context adaptive variable length coding, CAVLC), tạo mã số học nhị phân thích ứng ngữ cảnh (context adaptive binary arithmetic coding, CABAC), tạo mã số học nhị phân thích ứng ngữ cảnh dựa trên cú pháp (syntax-based context-adaptive binary arithmetic coding, SBAC), tạo mã entropi phân chia khoảng xác suất (probability interval partitioning entropy, PIPE) hoặc kỹ thuật tạo mã entropi khác. Trong trường hợp tạo mã entropi dựa trên ngữ cảnh, ngữ cảnh có thể dựa trên các khối lân cận. Tiếp theo tạo mã entropi bởi bộ phận tạo mã entropi 56, dòng bit được mã hóa có thể được truyền đến thiết bị khác (ví dụ, bộ giải mã video 30) hoặc thu được để truyền hoặc truy hồi sau này.

Bộ phận lượng tử hóa ngược 58 và bộ phận biến đổi ngược 60 áp dụng tương ứng lượng tử hóa ngược và biến đổi ngược để tái tạo khối dư trong miền điểm ảnh, ví dụ, để

sử dụng sau này làm khối tham chiếu. Bộ phận bù chuyển động 44 có thể tính toán khối tham chiếu bằng cách cộng khối dư vào khối dự đoán của một trong số các khung của bộ nhớ khung tham chiếu 64. Bộ phận bù chuyển động 44 cũng có thể áp dụng một hoặc nhiều bộ lọc nội suy vào khối dư được tái tạo để tính toán các giá trị điểm ảnh nguyên con để sử dụng khi ước lượng chuyển động. Bộ cộng 62 cộng khối dư được tái tạo vào khối dự đoán bù chuyển động được tạo ra bởi bộ phận bù chuyển động 44 để tạo ra khối video được tái tạo để lưu trữ trong bộ nhớ khung tham chiếu 64. Khối video được tái tạo có thể được sử dụng bởi bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 làm khối tham chiếu để tạo mã liên ảnh một khối trong khung video sau đó.

Fig.3 là sơ đồ khối minh họa một ví dụ về bộ giải mã video 30 mà có thể thực hiện các kỹ thuật tạo mã video. Trong ví dụ trên Fig.3, bộ giải mã video 30 bao gồm bộ phận giải mã entropi 70, bộ phận bù chuyển động 72, bộ phận dự đoán trong ảnh 74, bộ phận lượng tử hóa ngược 76, bộ phận biến đổi ngược 78, bộ nhớ khung tham chiếu 82, và bộ cộng 80. Bộ giải mã video 30 có thể, trong một số ví dụ, thực hiện cách giải mã thường ngược với cách mã hóa được mô tả liên quan đến bộ mã hóa video 20 (Fig.2). Bộ phận bù chuyển động 72 có thể tạo dữ liệu dự đoán dựa trên các vector chuyển động được nhận từ bộ phận giải mã entropi 70, trong khi bộ phận dự đoán trong ảnh 74 có thể tạo dữ liệu dự đoán dựa trên các bộ chỉ báo chế độ dự đoán trong ảnh được nhận từ bộ phận giải mã entropi 70.

Trong suốt quá trình giải mã, bộ giải mã video 30 nhận dòng bit video được mã hóa mà đại diện cho các khối video của lát video được mã hóa và các phần tử cú pháp liên kết từ bộ mã hóa video 20. Bộ phận giải mã entropi 70 của bộ giải mã video 30 giải mã entropi dòng bit để tạo các hệ số được lượng tử hóa, các vector chuyển động hoặc các bộ chỉ báo chế độ dự đoán trong ảnh, và các phần tử cú pháp khác. Bộ phận giải mã entropi 70 chuyển tiếp các vector chuyển động và các phần tử cú pháp khác đến bộ phận bù chuyển động 72. Bộ giải mã video 30 có thể nhận các phần tử cú pháp ở mức lát video và/hoặc mức khối video.

Khi lát video được tạo mã dưới dạng lát được tạo mã trong ảnh (I), bộ phận dự đoán trong ảnh 74 có thể tạo dữ liệu dự đoán đối với khối video của lát video hiện hành dựa trên chế độ dự đoán trong ảnh và dữ liệu được báo hiệu từ các khối được giải mã

trước đó của khung hoặc hình ảnh hiện hành. Khi khung video được tạo mã như là lát được tạo mã liên ảnh (ví dụ, B, P, hoặc GPB), bộ phận bù chuyển động 72 tạo ra các khối dự đoán đối với khối video của lát video hiện hành dựa trên các vector chuyển động và các phần tử cú pháp khác được nhận từ bộ phận giải mã entropi 70. Các khối dự đoán có thể được tạo ra từ một trong số các hình ảnh tham chiếu trong một trong số các danh sách hình ảnh tham chiếu. Bộ giải mã video 30 có thể xây dựng các danh sách khung tham chiếu, List (danh sách) 0 và List (danh sách) 1, sử dụng các kỹ thuật xây dựng mặc định dựa trên các hình ảnh tham chiếu được lưu trữ trong bộ nhớ khung tham chiếu 82.

Bộ phận bù chuyển động 72 xác định thông tin dự đoán đối với khối video của lát video hiện hành bằng cách phân tách các vector chuyển động và các phần tử cú pháp khác, và sử dụng thông tin dự đoán để tạo ra các khối dự đoán đối với khối video hiện hành đang được giải mã. Ví dụ, bộ phận bù chuyển động 72 sử dụng một số trong số các phần tử cú pháp được nhận để xác định chế độ dự đoán (ví dụ, dự đoán trong ảnh hoặc liên ảnh) được sử dụng để tạo mã các khối video của lát video, loại lát dự đoán liên ảnh (ví dụ, lát B, lát P, hoặc lát GPB), thông tin xây dựng đối với một hoặc nhiều danh sách trong số các danh sách hình ảnh tham chiếu đối với lát, các vector chuyển động đối với mỗi khối video được mã hóa liên ảnh của lát, trạng thái dự đoán liên ảnh đối với mỗi khối video được mã hóa liên ảnh của lát, và thông tin khác để giải mã các khối video trong lát video hiện hành.

Bộ phận bù chuyển động 72 cũng có thể thực hiện nội suy dựa trên các bộ lọc nội suy. Bộ phận bù chuyển động 72 có thể sử dụng các bộ lọc nội suy như được sử dụng bởi bộ mã hóa video 20 trong suốt quá trình mã hóa các khối video để tính toán các giá trị được nội suy đối với các điểm ảnh nguyên con của các khối tham chiếu. Trong trường hợp này, bộ phận bù chuyển động 72 có thể xác định các bộ lọc nội suy được sử dụng bởi bộ mã hóa video 20 từ các phần tử cú pháp được nhận và sử dụng các bộ lọc nội suy để tạo ra các khối dự đoán.

Dữ liệu cho ảnh bố cục tương ứng với bản đồ độ sâu có thể được lưu trữ trong bộ nhớ khung tham chiếu 82. Bộ phận bù chuyển động 72 cũng có thể được tạo cấu hình để dự đoán liên ảnh các khối độ sâu của bản đồ độ sâu.



Theo một phương án, bộ giải mã video 30 bao gồm giao diện người dùng (UI) 84. Giao diện người dùng 84 được tạo cấu hình để nhận đầu vào từ người dùng của bộ giải mã video 30 (ví dụ, người quản trị mạng). Qua giao diện người dùng 84, người dùng có thể quản lý hoặc thay đổi các thiết đặt trên bộ giải mã video 30. Ví dụ, người dùng có thể nhập hoặc nếu không thì cung cấp giá trị đối với thông số (ví dụ, cờ) để điều khiển cấu hình và/hoặc hoạt động của bộ giải mã video 30 theo quyền ưu tiên của người dùng. Giao diện người dùng 84 có thể, ví dụ, là giao diện người dùng đồ họa (GUI) mà cho phép người dùng tương tác với bộ giải mã video 30 qua các biểu tượng đồ họa, các bảng chọn thả xuống, các hộp kiểm tra, v.v.. Trong một số trường hợp, giao diện người dùng 84 có thể nhận thông tin từ người dùng qua bàn phím, chuột, hoặc thiết bị ngoại vi khác. Theo một phương án, người dùng có thể truy cập giao diện người dùng 84 qua điện thoại thông minh, thiết bị máy tính bảng, máy tính cá nhân được đặt ở xa từ bộ giải mã video 30, v.v.. Như được sử dụng ở đây, giao diện người dùng 84 có thể được gọi là đầu vào bên ngoài hoặc phương tiện bên ngoài.

Lưu ý, các kỹ thuật nén video thực hiện dự đoán không gian (trong hình ảnh) và/hoặc dự đoán thời gian (liên hình ảnh) để làm giảm hoặc loại bỏ phần dư thừa vốn có trong các chuỗi video. Đối với việc tạo mã video dựa trên khối, lát video (tức là, hình ảnh video hoặc một phần của hình ảnh video) có thể được phân chia thành các khối video, mà cũng có thể được gọi là các khối cây (treeblocks), các khối cây tạo mã (coding tree block, CTB), các đơn vị cây tạo mã (coding tree unit, CTU), các đơn vị tạo mã (coding unit, CU) và/hoặc các nút tạo mã. Các khối video trong lát được tạo mã trong ảnh (I) của hình ảnh được mã hóa sử dụng dự đoán không gian đối với các mẫu tham chiếu trong các khối lân cận trong cùng một hình ảnh. Các khối video trong lát được tạo mã liên ảnh (P hoặc B) của hình ảnh có thể sử dụng dự đoán không gian đối với các mẫu tham chiếu trong các khối lân cận trong cùng một hình ảnh hoặc dự đoán thời gian đối với các mẫu tham chiếu trong các hình ảnh tham chiếu khác. Các hình ảnh có thể được gọi là các khung, và các hình ảnh tham chiếu có thể được gọi các khung tham chiếu.

Dự đoán không gian hoặc thời gian dẫn đến khối dự đoán cho khối cần được tạo mã. Dữ liệu dư thể hiện các sự chênh lệch điểm ảnh giữa khối gốc cần được tạo mã và khối dự đoán. Khối được tạo mã liên ảnh được mã hóa theo vectơ chuyển động mà trở

đến khối của các mẫu tham chiếu tạo thành khối dự đoán, và dữ liệu dư chỉ báo sự chênh lệch giữa khối được tạo mã và khối dự đoán. Khối được tạo mã trong ảnh được mã hóa theo chế độ tạo mã trong ảnh và dữ liệu dư. Để nén thêm, dữ liệu dư có thể được biến đổi từ miền điểm ảnh sang miền biến đổi, tạo ra các hệ số biến đổi dư, mà sau đó có thể được lượng tử hóa. Các hệ số biến đổi được lượng tử hóa, ban đầu được bố trí trong mảng hai chiều, có thể được quét để tạo ra vector một chiều của các hệ số biến đổi, và mã hóa entropi có thể được áp dụng để thu được kết quả nén thậm chí là tốt hơn.

Ảnh và video nén đã trải qua tăng trưởng nhanh chóng, dẫn đến các tiêu chuẩn tạo mã khác nhau. Các tiêu chuẩn tạo mã video này bao gồm ITU-T H.261, MPEG-1 phần 2 của tổ chức chuẩn hóa quốc tế/Ủy ban kỹ thuật điện quốc tế (ISO/IEC), ITU-T H.262 hoặc ISO/IEC MPEG-2 phần 2, ITU-T H.263, ISO/IEC MPEG-4 phần 2, tạo mã video tiên tiến (Advanced Video Coding, AVC), cũng được gọi là ITU-T H.264 hoặc ISO/IEC MPEG-4 phần 10, và tạo mã video hiệu suất cao (HEVC), cũng được gọi là ITU-T H.265 hoặc MPEG-H phần 2. AVC bao gồm các loại mở rộng như tạo mã video thay đổi được (Scalable Video Coding, SVC), tạo mã video đa cảnh nhìn (Multiview Video Coding, MVC) và tạo mã video đa cảnh nhìn cộng độ sâu (Multiview Video Coding plus Depth, MVC+D), và 3D AVC (3D-AVC). HEVC bao gồm các loại mở rộng như HEVC thay đổi được (SHVC), HEVC đa cảnh nhìn (MV-HEVC), và 3D HEVC (3D-HEVC).

Cũng có tiêu chuẩn tạo mã video mới, được đặt tên là tạo mã video đa năng (Versatile Video Coding, VVC), được phát triển bởi nhóm chuyên gia video chung (joint video experts team, JVET) của ITU-T và ISO/IEC. Mặc dù tiêu chuẩn VVC có một số bản phác thảo công việc, nhưng một bản phác thảo công việc (WD) của VVC cụ thể, cụ thể B. Bross, J. Chen, và S. Liu, “Versatile Video Coding (Draft 5),” JVET-N1001-v3, 13th JVET Meeting, March 27, 2019 (phác thảo VVC 5) được tham chiếu ở đây.

Phần mô tả về các kỹ thuật được bộc lộ ở đây dựa trên tiêu chuẩn tạo mã video đang phát triển Tạo mã video đa năng (Versatile Video Coding, VVC) bởi nhóm chuyên gia video chung (joint video experts team, JVET) của ITU-T và ISO/IEC. Tuy nhiên, các kỹ thuật này cũng áp dụng vào các đặc tả codec video khác.

Các sơ đồ phân chia hình ảnh trong HEVC được thảo luận.

HEVC bao gồm bốn sơ đồ phân chia hình ảnh khác nhau, cụ thể các lát đều, các lát phụ thuộc, các phiên, và việc xử lý song song mặt sóng (Wavefront Parallel Processing, WPP), mà có thể được áp dụng cho việc so khớp kích thước đơn vị truyền tối đa (Maximum Transfer Unit, MTU), việc xử lý song song, và trễ đầu-cuối giảm bớt.

Các lát đều tương tự như trong H.264/AVC. Lát đều được đóng gói trong đơn vị trừu tượng mạng (network abstraction layer, NAL) của chính nó, và dự đoán trong hình ảnh (dự đoán trong mẫu, dự đoán thông tin chuyển động, dự đoán chế độ tạo mã) và sự phụ thuộc vào mã hóa entropi qua các biên lát là không được phép. Vì vậy, lát đều có thể được tái tạo một cách độc lập từ các lát đều khác trong cùng một hình ảnh (mặc dù có thể vẫn là các sự phụ thuộc lẫn nhau do các hoạt động lọc vòng).

Lát đều là công cụ duy nhất có thể được sử dụng cho quá trình song song hóa mà cũng khả dụng, ở dạng giống hệt ảo, trong H.264/AVC. Sự song song hóa dựa trên lát đều không sử dụng nhiều hoạt động truyền thông liên bộ xử lý hoặc liên lõi (trừ việc chia sẻ dữ liệu liên bộ xử lý hoặc liên lõi đối với việc bù chuyển động khi giải mã hình ảnh được tạo mã dự đoán, mà thường nặng hơn nhiều so với việc chia sẻ dữ liệu liên bộ xử lý hoặc liên lõi do dự đoán trong hình ảnh). Tuy nhiên, vì cùng lý do, việc sử dụng các lát đều có thể gặp phải lượng thông tin thừa tạo mã quan trọng do chi phí bit của phần đầu lát và do thiếu sự dự đoán qua các biên lát. Hơn nữa, các lát đều (ngược lại với các công cụ khác được nêu dưới đây) cũng dùng làm cơ chế chủ đạo để phân chia dòng bit để so khớp với các yêu cầu kích thước MTU, mà là do sự độc lập trong hình ảnh của các lát đều và do mỗi lát đều được đóng gói trong đơn vị NAL của chính nó. Trong nhiều trường hợp, mục tiêu song song hóa và mục tiêu so khớp kích thước MTU đặt ra các nhu cầu trái ngược cho bố cục lát trong hình ảnh. Sự thực hiện tình huống này dẫn đến sự phát triển của các công cụ song song hóa được nêu dưới đây.

Các lát phụ thuộc có các phần đầu lát ngắn và cho phép phân chia dòng bit ở các biên khối cây mà không cần phá vỡ dự đoán trong hình ảnh bất kỳ. Về cơ bản, các lát phụ thuộc cung cấp sự phân mảnh các lát đều thành nhiều đơn vị NAL, mà cung cấp trễ đầu-cuối giảm bớt bằng cách cho phép một phần của lát đều được gửi ra ngoài trước khi việc mã hóa toàn bộ lát đều được kết thúc.

Trong WPP, hình ảnh được phân chia thành các hàng đơn lẻ của các khối cây tạo mã (coding tree block, CTB). Giải mã entropi và dự đoán được phép sử dụng dữ liệu từ các CTB trong các phân chia khác. Việc xử lý song song là khả thi qua giải mã song song các hàng CTB, trong đó việc bắt đầu giải mã hàng CTB bị trễ bởi hai CTB, vì vậy, để đảm bảo rằng dữ liệu liên quan đến CTB phía trên và bên phải của CTB chủ thể là khả dụng trước khi CTB chủ thể được giải mã. Nhờ sử dụng sự bắt đầu so le này (mà xuất hiện như mặt sóng được biểu thị bằng đồ thị), việc song song hóa là khả thi với nhiều bộ xử lý/lõi như hình ảnh chứa các hàng CTB. Do dự đoán trong hình ảnh giữa các hàng khối cây lân cận trong hình ảnh được phép, hoạt động truyền thông liên bộ xử lý/liên lõi cần thiết để cho phép dự đoán trong hình ảnh có thể là quan trọng. Việc phân chia WPP không dẫn đến việc tạo ra các đơn vị NAL bổ sung so với khi nó không được áp dụng, vì vậy WPP không phải là công cụ để so khớp kích thước MTU. Tuy nhiên, khi việc so khớp kích thước MTU được sử dụng, các lát đều có thể được sử dụng với WPP, với lượng thông tin thừa tạo mã nhất định.

Các phiên xác định các biên ngang và thẳng đứng mà phân chia hình ảnh thành các cột và các hàng phiên. Thứ tự quét của các CTB được thay đổi để là cục bộ trong phiên (theo thứ tự quét kiểu mảnh CTB của phiên), trước khi giải mã CTB trên cùng bên trái của phiên tiếp theo theo thứ tự quét kiểu mảnh phiên của hình ảnh. Tương tự như các lát đều, các phiên phá vỡ các sự phụ thuộc dự đoán trong hình ảnh cũng như các sự phụ thuộc giải mã entropi. Tuy nhiên, chúng không cần được bao gồm vào các đơn vị NAL cá nhân (giống WPP về điểm này); vì vậy các phiên không thể được sử dụng để so khớp kích thước MTU. Mỗi phiên có thể được xử lý bởi một bộ xử lý/lõi, và hoạt động truyền thông liên bộ xử lý/liên lõi cần để dự đoán trong hình ảnh giữa các bộ phận xử lý giải mã các phiên lân cận được giới hạn để truyền phần đầu lát được chia sẻ trong các trường hợp mà lát đang mở rộng nhiều hơn một phiên, và việc lọc vòng liên quan đến việc chia sẻ các mẫu tái tạo và siêu dữ liệu. Khi nhiều hơn một phiên hoặc phân đoạn WPP có trong lát, độ lệch bại điểm mục nhập đối với mỗi phiên hoặc phân đoạn WPP không phải là phân đoạn đầu tiên trong lát được báo hiệu trong phần đầu lát.

Để đơn giản, các hạn chế trên ứng dụng của bốn sơ đồ phân chia hình ảnh khác nhau đã được quy định trong HEVC. Chuỗi video được tạo mã đã cho không thể bao gồm cả phiên và các mặt sóng đối với hầu hết các hồ sơ được quy định trong HEVC.

Đối với mỗi lát và phiến, một trong số hoặc cả hai điều kiện sau đây nên được đáp ứng: 1) tất cả các khối cây được tạo mã trong lát thuộc về cùng phiến; 2) tất cả các khối cây được tạo mã trong phiến thuộc về cùng lát. Cuối cùng, phân đoạn mặt sóng chứa chính xác một hàng CTB, và khi WPP đang sử dụng, nếu lát bắt đầu trong hàng CTB, thì nó nên kết thúc trong cùng hàng CTB.

Các sơ đồ phân chia hình ảnh trong VVC được thảo luận.

Như được lưu ý trên đây, HEVC bao gồm bốn sơ đồ phân chia hình ảnh khác nhau, cụ thể các lát, các phiến và các khối gạch (brick), và việc xử lý song song mặt sóng (Wavefront Parallel Processing, WPP), mà có thể được áp dụng cho việc so khớp kích thước đơn vị truyền tối đa (Maximum Transfer Unit, MTU), việc xử lý song song, và trễ đầu-cuối giảm bớt.

Các phiến trong VVC tương tự như các phiến trong HEVC. Các phiến xác định các biên ngang và thẳng đứng mà phân chia hình ảnh thành các cột và các hàng phiến. Trong VVC, khái niệm về các phiến còn được cải tiến thêm nữa bằng cách cho phép phiến được chia tách thêm theo chiều ngang để tạo thành các khối gạch. Phiến mà không được chia tách thêm nữa cũng được coi là khối gạch. Thứ tự quét của các CTB được thay đổi để là cục bộ trong khối gạch (theo thứ tự quét kiểu mảnh CTB của khối gạch), trước khi giải mã CTB trên cùng bên trái của khối gạch tiếp theo theo thứ tự quét kiểu mảnh khối gạch của hình ảnh.

Các lát trong VVC bao gồm một hoặc nhiều khối gạch. Mỗi lát được đóng gói trong đơn vị NAL của chính nó, và dự đoán trong hình ảnh (dự đoán trong mẫu, dự đoán thông tin chuyển động, dự đoán chế độ tạo mã) và sự phụ thuộc vào mã hóa entropi qua các biên lát là không được phép. Vì vậy, lát đều có thể được tái tạo một cách độc lập từ các lát đều khác trong cùng một hình ảnh (mặc dù có thể vẫn có các sự phụ thuộc lẫn nhau do các hoạt động lọc vòng). VVC định nghĩa hai loại lát, đó là: lát hình chữ nhật và lát quét kiểu mảnh. Lát hình chữ nhật bao gồm một hoặc nhiều khối gạch chiếm vùng hình chữ nhật trong hình ảnh. Lát quét kiểu mảnh bao gồm một hoặc nhiều khối gạch mà theo thứ tự quét kiểu mảnh của các khối gạch trong hình ảnh.

Dấu hiệu WPP trong VVC tương tự như dấu hiệu WPP trong HEVC trừ việc HEVC WPP có hai trễ CTU trong khi đó VVC WPP có một trễ CTU. Đối với HEVC

WPP, tuyến đoạn (thread) giải mã mới có thể bắt đầu giải mã CTU thứ nhất trong hàng CTU được ấn định của nó sau khi hàng CTU trước đó có hai CTU đầu tiên của nó đã được giải mã; mặt khác, đối với VVC WPP, tuyến đoạn giải mã mới có thể bắt đầu giải mã CTU thứ nhất trong hàng CTU được ấn định của nó sau khi hàng CTU trước đó có CTU thứ nhất đã được giải mã.

Việc báo hiệu các lát hình chữ nhật được thảo luận.

Cấu trúc của các lát hình chữ nhật được báo hiệu trong tập hợp thông số hình ảnh (PPS) bằng cách mô tả số lượng lát hình chữ nhật trong hình ảnh. Đối với mỗi lát, tập hợp chỉ số khối gạch trên cùng bên trái và giá trị delta để dẫn xuất chỉ số của khối gạch dưới cùng bên phải được báo hiệu để mô tả vị trí của lát trong hình ảnh và kích thước của nó (tức là, đơn vị của khối gạch). Đối với lát quét kiểu mảnh, thông tin của nó được báo hiệu trong phần đầu lát nhờ sử dụng chỉ số của khối gạch thứ nhất trong lát quét kiểu mảnh và số lượng khối gạch trong lát.

Phần của bảng cú pháp PPS được thể hiện dưới đây bao gồm các phần tử cú pháp mà mô tả báo hiệu của các phiên, các khối gạch, và thông tin lát hình chữ nhật trong PPS.

|                                    |                 |
|------------------------------------|-----------------|
| <b>pic_parameter_set_rbsp( ) {</b> | <b>Bộ mô tả</b> |
| ...                                |                 |
| <b>single_tile_in_pic_flag</b>     | u(1)            |
| if( !single_tile_in_pic_flag ) {   |                 |
| <b>uniform_tile_spacing_flag</b>   | u(1)            |
| if( uniform_tile_spacing_flag ) {  |                 |
| <b>tile_cols_width_minus1</b>      | ue(v)           |
| <b>tile_rows_height_minus1</b>     | ue(v)           |
| } else {                           |                 |
| <b>num_tile_columns_minus1</b>     | ue(v)           |
| <b>num_tile_rows_minus1</b>        | ue(v)           |

|                                                                           |       |
|---------------------------------------------------------------------------|-------|
| for( i = 0; i < num_tile_columns_minus1; i++ )                            |       |
| <b>tile_column_width_minus1[ i ]</b>                                      | ue(v) |
| for( i = 0; i < num_tile_rows_minus1; i++ )                               |       |
| <b>tile_row_height_minus1[ i ]</b>                                        | ue(v) |
| }                                                                         |       |
| <b>brick_splitting_present_flag</b>                                       | u(1)  |
| for( i = 0; brick_splitting_present_flag && i < NumTilesInPic; i++ )<br>{ |       |
| <b>brick_split_flag[ i ]</b>                                              | u(1)  |
| if( brick_split_flag[ i ] ) {                                             |       |
| <b>uniform_brick_spacing_flag[ i ]</b>                                    | u(1)  |
| if( uniform_brick_spacing_flag[ i ] )                                     |       |
| <b>brick_height_minus1[ i ]</b>                                           | ue(v) |
| else {                                                                    |       |
| <b>num_brick_rows_minus1[ i ]</b>                                         | ue(v) |
| for( j = 0; j < num_brick_rows_minus1[ i ]; j++ )                         |       |
| <b>brick_row_height_minus1[ i ][ j ]</b>                                  | ue(v) |
| }                                                                         |       |
| }                                                                         |       |
| }                                                                         |       |
| <b>single_brick_per_slice_flag</b>                                        | u(1)  |
| if( !single_brick_per_slice_flag )                                        |       |
| <b>rect_slice_flag</b>                                                    | u(1)  |
| if( rect_slice_flag && !single_brick_per_slice_flag ) {                   |       |
| <b>num_slices_in_pic_minus1</b>                                           | ue(v) |

|                                                    |      |
|----------------------------------------------------|------|
| for( i = 0; i <= num_slices_in_pic_minus1; i++ ) { |      |
| if( i > 0 )                                        |      |
| <b>top_left_brick_idx[ i ]</b>                     | u(v) |
| <b>bottom_right_brick_idx_delta[ i ]</b>           | u(v) |
| }                                                  |      |
| }                                                  |      |
| <b>loop_filter_across_bricks_enabled_flag</b>      | u(1) |
| if( loop_filter_across_bricks_enabled_flag )       |      |
| <b>loop_filter_across_slices_enabled_flag</b>      | u(1) |
| }                                                  |      |
| ...                                                |      |
| }                                                  |      |

single\_brick\_per\_slice\_flag bằng 1 quy định rằng mỗi lát mà liên quan đến PPS này bao gồm một khối gạch. single\_brick\_per\_slice\_flag bằng 0 quy định rằng lát mà liên quan đến PPS này có thể bao gồm nhiều hơn một khối gạch. Khi không có mặt, giá trị của single\_brick\_per\_slice\_flag được suy ra là bằng 1.

rect\_slice\_flag bằng 0 quy định rằng các khối gạch trong mỗi lát theo thứ tự quét kiểu mảnh và thông tin lát không được báo hiệu trong PPS. rect\_slice\_flag bằng 1 quy định rằng các khối gạch trong mỗi lát bao trùm vùng hình chữ nhật của hình ảnh và thông tin lát được báo hiệu trong PPS. Khi single\_brick\_per\_slice\_flag bằng 1 thì rect\_slice\_flag được suy ra là bằng 1.

num\_slices\_in\_pic\_minus1 cộng 1 quy định số lượng lát trong mỗi hình ảnh tham chiếu đến PPS. Giá trị của num\_slices\_in\_pic\_minus1 sẽ nằm trong đoạn từ 0 đến NumBricksInPic – 1. Khi không có mặt và single\_brick\_per\_slice\_flag bằng 1, giá trị của num\_slices\_in\_pic\_minus1 được suy ra là bằng NumBricksInPic – 1.



`top_left_brick_idx[ i ]` quy định chỉ số khối gạch của khối gạch được bố trí tại góc trên cùng bên trái của lát thứ  $i$ . Giá trị của `top_left_brick_idx[ i ]` sẽ không bằng giá trị của `top_left_brick_idx[ j ]` với  $i$  bất kỳ không bằng  $j$ . Khi không có mặt, giá trị của `top_left_brick_idx[ i ]` được suy ra là bằng  $i$ . Độ dài của phần tử cú pháp `top_left_brick_idx[ i ]` là  $\text{Ceil}(\text{Log2}(\text{NumBricksInPic}))$  bit.

`bottom_right_brick_idx_delta[ i ]` quy định sự chênh lệch giữa chỉ số khối gạch của khối gạch được bố trí tại góc dưới bên phải của lát thứ  $i$  và `top_left_brick_idx[ i ]`. Khi `single_brick_per_slice_flag` bằng 1, giá trị của `bottom_right_brick_idx_delta[ i ]` được suy ra là bằng 0. Độ dài của phần tử cú pháp `bottom_right_brick_idx_delta[ i ]` là  $\text{Ceil}(\text{Log2}(\text{NumBricksInPic} - \text{top\_left\_brick\_idx}[ i ]))$  bit.

Đây là sự đòi hỏi về sự tương hợp dòng bit mà lát sẽ bao gồm số lượng phiên hoàn chỉnh hoặc chỉ chuỗi liên tiếp của các khối gạch hoàn chỉnh của một phiên.

Biến số `NumBricksInSlice[ i ]` và `BricksToSliceMap[ j ]`, mà quy định số lượng khối gạch trong lát thứ  $i$  và ánh xạ của các khối gạch đến các lát, được dẫn xuất như sau:

```

NumBricksInSlice[ i ] = 0
botRightBkIdx = top_left_brick_idx[ i ] + bottom_right_brick_idx_delta[ i ]
for( j = 0; j < NumBricksInPic; j++ ) {
    if( BrickColBd[ j ] >= BrickColBd[ top_left_brick_idx[ i ] ] &&
        BrickColBd[ j ] <= BrickColBd[ botRightBkIdx ] &&
        BrickRowBd[ j ] >= BrickRowBd[ top_left_brick_idx[ i ] ] &&
        BrickRowBd[ j ] <= BrickColBd[ botRightBkIdx ] ) {
        NumBricksInSlice[ i ]++
        BricksToSliceMap[ j ] = i
    }
}
...

```

(7-35)

Việc báo hiệu WPP trong VVC được thảo luận.

Phương pháp báo hiệu đối với WPP trong VVC được mô tả trong bảng cú pháp và các ngữ nghĩa của PPS, phần đầu lát, và dữ liệu lát.

Cờ trong PPS được gọi là `entropy_coding_sync_enabled_flag` quy định liệu WPP có được sử dụng để tạo mã các hình ảnh mà tham chiếu đến PPS như được thể hiện trong phần của bảng cú pháp PPS dưới đây.

|                                               |                   |
|-----------------------------------------------|-------------------|
| <code>pic_parameter_set_rbsp( ) {</code>      | <b>Bộ mô tả</b>   |
| <code>...</code>                              |                   |
| <code>entropy_coding_sync_enabled_flag</code> | <code>u(1)</code> |
| <code>...</code>                              |                   |
| <code>}</code>                                |                   |

Khi WPP được phép để tạo mã hình ảnh, phần đầu lát của tất cả các lát của hình ảnh bao gồm thông tin về điểm nhập (tức là, độ lệch từ đầu của dữ liệu phân tải dữ liệu lát). Điểm nhập được sử dụng để truy cập mỗi tập con của hàng CTU để xử lý theo phương pháp WPP. Thông tin này được báo hiệu như được thể hiện trong phần của bảng cú pháp phần đầu lát dưới đây.

|                                                       |                    |
|-------------------------------------------------------|--------------------|
| <code>slice_header( ) {</code>                        | <b>Bộ mô tả</b>    |
| <code>...</code>                                      | <code>ue(v)</code> |
| <code>if ( entropy_coding_sync_enabled_flag )</code>  |                    |
| <code>num_entry_point_offsets</code>                  | <code>ue(v)</code> |
| <code>if( NumEntryPoints &gt; 0 ) {</code>            |                    |
| <code>offset_len_minus1</code>                        | <code>ue(v)</code> |
| <code>for( i = 0; i &lt; NumEntryPoints; i++ )</code> |                    |
| <code>entry_point_offset_minus1[ i ]</code>           | <code>u(v)</code>  |
| <code>}</code>                                        |                    |
| <code>byte_alignment( )</code>                        |                    |
| <code>}</code>                                        |                    |

Khi WPP được phép, mỗi hàng CTU được gọi là tập con dữ liệu trong phần tải dữ liệu của dữ liệu lát. Ở cuối của mỗi tập con dữ liệu, bit được chỉ định `designated_end_of_subset_one_bit` được báo hiệu để chỉ báo cuối của tập con dữ liệu. Hơn nữa, để đảm bảo rằng kích thước của tập con dữ liệu là bội số của bai (tức là, 8 bit), việc cân chỉnh bai được thực hiện để bổ sung các bit cân chỉnh bai vào cuối của mỗi tập con dữ liệu. Việc báo hiệu `end_of_subset_one_bit` và việc cân chỉnh bai ở cuối của mỗi tập con được thể hiện trong bảng cú pháp dữ liệu lát dưới đây.

|                                                                                                     |          |
|-----------------------------------------------------------------------------------------------------|----------|
| slice_data( ) {                                                                                     | Bộ mô tả |
| for( i = 0; i < NumBricksInCurrSlice; i++ ) {                                                       |          |
| CtbAddrInBs = FirstCtbAddrBs[ SliceBrickIdx[ i ] ]                                                  |          |
| for( j = 0; j < NumCtusInBrick[ SliceBrickIdx[ i ] ]; j++,<br>CtbAddrInBs++ ) {                     |          |
| if( ( j % BrickWidth[ SliceBrickIdx[ i ] ] ) == 0 ) {                                               |          |
| NumHmvpCand = 0                                                                                     |          |
| NumHmvpIbcCand = 0                                                                                  |          |
| }                                                                                                   |          |
| CtbAddrInRs = CtbAddrBsToRs[ CtbAddrInBs ]                                                          |          |
| coding_tree_unit( )                                                                                 |          |
| if( entropy_coding_sync_enabled_flag &&<br>(( j + 1 ) % BrickWidth[ SliceBrickIdx[ i ] ] == 0 ) ) { |          |
| <b>end_of_subset_one_bit</b> /* bằng 1 */                                                           | ae(v)    |
| if( j < NumCtusInBrick[ SliceBrickIdx[ i ] ] - 1 )                                                  |          |
| byte_alignment( )                                                                                   |          |
| }                                                                                                   |          |
| if( !entropy_coding_sync_enabled_flag ) {                                                           |          |
| <b>end_of_brick_one_bit</b> /* bằng 1 */                                                            | ae(v)    |
| if( i < NumBricksInCurrSlice - 1 )                                                                  |          |
| byte_alignment( )                                                                                   |          |
| }                                                                                                   |          |
| }                                                                                                   |          |
| }                                                                                                   |          |

Một số vấn đề trong số các vấn đề với WPP và các khối gạch được thảo luận.

Thứ nhất, khi lát chứa nhiều khối gạch và WPP được phép tạo mã hình ảnh chứa lát, mỗi hàng CTU của mỗi khối gạch trong lát là tập con dữ liệu. Ở cuối mỗi tập con dữ liệu, hoặc phần tử cú pháp `end_of_subset_one_bit` được báo hiệu để chỉ báo cuối hàng CTU hoặc phần tử cú pháp `end_of_brick_one_bit` được báo hiệu để chỉ báo cuối CTU của khối gạch. Tuy nhiên, không nhất thiết phải báo hiệu cả hai phần tử cú pháp. Tương tự, ở cuối mỗi tập con dữ liệu, việc cân chỉnh bai nên có mặt nhưng không cần sao chép.

Khi các phiên, các khối gạch, và WPP được sử dụng cùng nhau, xem xét rằng lát có thể chứa một hoặc nhiều phiên và mỗi phiên có thể chứa một hoặc nhiều khối gạch, việc thực thi WPP có thể phức tạp hơn.

Để giải quyết vấn đề được mô tả trên đây, sáng chế đề xuất các khía cạnh sau đây (mỗi khía cạnh trong số đó có thể được áp dụng một cách riêng lẻ và một số khía cạnh trong số đó có thể được áp dụng kết hợp).

Giải pháp thứ nhất bao gồm phương pháp giải mã dòng bit video. Theo một phương án, dòng bit video bao gồm ít nhất một hình ảnh chứa các lát, mỗi lát trong số các lát này bao gồm các khối gạch, và mỗi khối gạch trong số các khối gạch này bao gồm các khối cây tạo mã (các CTU). Phương pháp này bao gồm bước phân tách tập hợp thông số để xác định liệu việc xử lý song song mặt sóng có được phép đối với hình ảnh hiện hành và/hoặc lát hiện hành hay không. Phương pháp này bao gồm bước phân tách dữ liệu lát của lát hiện hành để thu nhận các khối gạch và các CTU trong mỗi khối gạch. Phương pháp này còn bao gồm bước phân tách CTU hiện hành, mà ở trong khối gạch; và xác định vị trí của CTU hiện hành. Ngoài ra, phương pháp này bao gồm bước báo hiệu bit để chỉ báo cuối của hàng CTU và báo hiệu các bit cân chỉnh bai khi tất cả các điều kiện sau đây được đáp ứng: WPP là được phép cho việc tạo mã lát hiện hành, CTU hiện hành không phải là CTU cuối cùng của khối gạch hiện hành; và CTU tiếp theo theo thứ tự giải mã khối gạch không phải là CTU thứ nhất của hàng CTU trong khối gạch hiện hành. Phương pháp này bao gồm bước báo hiệu bit để chỉ báo cuối của khối gạch khi CTU hiện hành là CTU cuối cùng trong khối gạch hiện hành, và báo hiệu các bit cân chỉnh bai khi khối gạch hiện hành là CTU cuối cùng trong khối gạch hiện hành nhưng không phải là CTU cuối cùng của lát hiện hành.

Giải pháp thứ hai bao gồm phương pháp để mã hóa dòng bit video. Dòng bit video bao gồm ít nhất một hình ảnh chứa các lát, mỗi lát trong số các lát này bao gồm các phiên và các khối gạch, và mỗi phiên bao gồm một hoặc nhiều khối gạch. Phương pháp này bao gồm bước ràng buộc mỗi lát của hình ảnh hiện hành để chứa chỉ một phiên và mỗi phiên để chứa chỉ một khối gạch khi WPP là được phép để mã hóa hình ảnh hiện hành.

Giải pháp thứ hai khác bao gồm phương pháp để mã hóa dòng bit video. Dòng bit video bao gồm ít nhất một hình ảnh chứa các lát, mỗi lát trong số các lát này bao gồm các phiên và các khối gạch, và mỗi phiên bao gồm một hoặc nhiều khối gạch. Phương pháp này bao gồm bước ràng buộc mỗi phiên của hình ảnh hiện hành để chứa chỉ một khối gạch khi WPP là được phép để mã hóa hình ảnh hiện hành. Điều này có nghĩa là, khi giá trị của `entropy_coding_sync_enabled_flag` bằng 1, giá trị của `brick_splitting_present_flag` sẽ bằng 0.

Giải pháp thứ hai khác bao gồm phương pháp để mã hóa dòng bit video. Dòng bit video bao gồm ít nhất một hình ảnh chứa các lát, mỗi lát trong số các lát này bao gồm các phiên và các khối gạch, và mỗi phiên bao gồm một hoặc nhiều khối gạch. Phương pháp này bao gồm bước ràng buộc mỗi lát của hình ảnh hiện hành để chứa chỉ một khối gạch khi WPP là được phép để mã hóa hình ảnh hiện hành. Điều này có nghĩa là, khi giá trị của `entropy_coding_sync_enabled_flag` bằng 1, giá trị của `NumBricksInCurrSlice` được thay đổi sẽ bằng 1.

Fig.4 minh họa dòng bit video 400 được tạo cấu hình để thực thi WPP 450. Như được sử dụng ở đây, dòng bit video 400 cũng có thể được gọi là dòng bit video được tạo mã, dòng bit, hoặc các biến thể của chúng. Như được thể hiện trên Fig.4, dòng bit 400 bao gồm tập hợp thông số chuỗi (SPS) 402, tập hợp thông số hình ảnh (PPS) 404, phần đầu lát 406, và dữ liệu ảnh 408.

SPS 402 chứa dữ liệu mà là chung cho tất cả các hình ảnh trong chuỗi các hình ảnh (SOP). Ngược lại, PPS 404 chứa dữ liệu mà là chung cho toàn bộ hình ảnh. Phần đầu lát 406 chứa thông tin về lát hiện hành như, ví dụ, loại lát, mà là của các hình ảnh tham chiếu sẽ được sử dụng, v.v.. SPS 402 và PPS 404 thường có thể được gọi là tập hợp thông số. SPS 402, PPS 404, và phần đầu lát 406 là các loại đơn vị lớp trừu tượng

mạng (NAL). Đơn vị NAL là cấu trúc cú pháp chứa chỉ báo về loại dữ liệu cần tuân theo (ví dụ, dữ liệu video đã tạo mã). Các đơn vị NAL được phân loại thành các đơn vị NAL lớp tạo mã video (VCL) và các đơn vị NAL phi VCL. Các đơn vị NAL VCL chứa dữ liệu mà đại diện cho các giá trị của các mẫu trong các hình ảnh video, và các đơn vị NAL phi VCL chứa thông tin bổ sung liên kết bất kỳ như các tập hợp thông số (dữ liệu phần đầu quan trọng mà có thể áp dụng vào số lượng đơn vị NAL VCL lớn) và thông tin nâng cao phụ (thông tin định thời và dữ liệu bổ sung khác mà có thể nâng cao khả năng sử dụng của tín hiệu video được giải mã nhưng không cần giải mã các giá trị của các mẫu trong các hình ảnh video). Người có hiểu biết trung bình trong lĩnh vực tương ứng sẽ hiểu rằng dòng bit 400 có thể chứa các thông số khác và thông tin trong các ứng dụng thực tế.

Dữ liệu ảnh 408 trên Fig.4 bao gồm dữ liệu được liên kết với các ảnh hoặc video đang được mã hóa hoặc được giải mã. Dữ liệu ảnh 408 có thể được gọi đơn giản là phần tải dữ liệu hoặc dữ liệu đang được mang trong dòng bit 400. Dữ liệu ảnh 408 có thể được phân chia thành một hoặc nhiều hình ảnh, như hình ảnh 410, hình ảnh 412, và hình ảnh 414. Mặc dù ba hình ảnh 410-414 được thể hiện trên Fig.4, nhưng nhiều hình ảnh hơn hoặc ít hình ảnh hơn có thể có trong các ứng dụng thực tế.

Theo một phương án, các hình ảnh 410-414, mỗi hình ảnh được phân chia thành các lát, như lát 416, lát 418, và lát 420. Mặc dù ba lát (ví dụ, các lát 416-420) được thể hiện, nhiều lát hơn hoặc ít lát hơn có thể có trong các ứng dụng thực tế. Theo một phương án, các lát 416-420, mỗi lát được phân chia thành các phiên, như phiên 422, phiên 424, và phiên 426. Mặc dù ba phiên (ví dụ, các phiên 422-426) được thể hiện, nhưng nhiều phiên hơn hoặc ít phiên hơn có thể có trong các ứng dụng thực tế. Theo một phương án, các phiên 422-426, mỗi phiên được phân chia thành các CTB, như CTB 428 và CTB 430. Mặc dù bốn mươi CTB (ví dụ, CTB 428-430) được thể hiện, nhưng nhiều CTB hơn hoặc ít CTB hơn có thể có trong các ứng dụng thực tế.

WPP 450 có thể được sử dụng để mã hóa và/hoặc giải mã lát (ví dụ, lát 416-420). Như vậy, WPP 450 có thể được sử dụng bởi bộ mã hóa (ví dụ, bộ mã hóa video 20) hoặc bộ giải mã (ví dụ, bộ giải mã video 30).

Theo một phương án, WPP 450 được áp dụng vào phiên 424, mà là phần chia của lát 416, mà là phần chia của hình ảnh 410. Phiên chứa các CTB, như CTB 428 và CTB 430. Mỗi CTB (ví dụ, CTB 428-430) là nhóm của các mẫu có kích thước định trước mà có thể được phân chia thành các khối tạo mã bởi cây tạo mã. Các CTB 428 và các CTB 430 có thể được sắp xếp thành các hàng CTB 460, 462, 464, 466, và 468 và các cột CTB 470, 472, 474, 476, 478, 480, 482, và 484. Hàng CTB 460-468 là nhóm của các CTB 428-430 mà kéo dài theo chiều ngang giữa biên trái của phiên 424 và biên phải của phiên 424. Cột CTB 470-484 là nhóm của các CTB 428-430 kéo dài theo chiều thẳng đứng giữa biên trên cùng của phiên 424 và biên dưới cùng phiên 424. Theo một phương án, WPP 450 được áp dụng vào lát (ví dụ, lát 416) thay vì phiên (ví dụ, 424). Điều này có nghĩa là, các phiên là tùy chọn theo một số phương án.

WPP 450 có thể sử dụng nhiều tuyến đoạn tính toán hoạt động song song để tạo mã các CTB 428-430. Trong ví dụ được thể hiện, các CTB 428 (được phủ bóng) đã được tạo mã trong khi các CTB 430 (không được phủ bóng) chưa được tạo mã. Ví dụ, tuyến đoạn thứ nhất có thể bắt đầu tạo mã hàng CTB 460 trước tiên. Trong VVC, một khi một CTB 428 đã được tạo mã trong hàng CTB thứ nhất 460, tuyến đoạn thứ hai có thể bắt đầu tạo mã hàng CTB 462. Một khi một CTB 428 đã được tạo mã trong hàng CTB thứ hai 462, tuyến đoạn thứ ba có thể bắt đầu tạo mã hàng CTB 464. Một khi CTB 428 đã được tạo mã trong hàng CTB thứ ba 464, tuyến đoạn thứ tư có thể bắt đầu tạo mã hàng CTB 466. Một khi một CTB 428 đã được tạo mã trong hàng CTB thứ tư 466, tuyến đoạn thứ năm có thể bắt đầu tạo mã hàng CTB thứ năm 468. Điều này dẫn đến mẫu hình như được thể hiện trên Fig.4. Các tuyến đoạn bổ sung có thể được sử dụng như mong muốn. Điều này có nghĩa là, quy trình bắt đầu hàng CTB mới sau khi CTB ở hàng trước đó đã được tạo mã có thể được lặp lại. Cơ chế này tạo mẫu hình với ngoại hình dạng mặt sóng, và vì vậy có tên WPP 450. Một số cơ chế tạo mã video tạo mã CTB hiện hành 430 dựa trên CTB được tạo mã 428 được định vị phía trên hoặc ở bên trái của CTB hiện hành 430. Trong VVC, WPP 450 để lại một trễ tạo mã CTB 430 giữa lúc khởi tạo mỗi tuyến đoạn để đảm bảo các CTB 428 này đã được tạo mã nhờ đạt đến CTB hiện hành bất kỳ 430 cần được tạo mã. Trong HEVC, WPP 450 để lại hai trễ tạo mã CTB 430 giữa lúc khởi tạo mỗi tuyến đoạn để đảm bảo các CTB 428 này đã được tạo mã nhờ đạt đến CTB hiện hành bất kỳ 430 cần được tạo mã.



Các CTB 428 được tạo mã thành dòng bit (ví dụ, dòng bit 400) trong các hàng CTB 460-468. Theo đó, mỗi hàng CTB 460-468 có thể là tập con đánh địa chỉ được một cách độc lập của phiên 424 trong dòng bit 400. Ví dụ, mỗi hàng CTB 460-468 có thể được đánh địa chỉ tại điểm nhập 486. Điểm nhập 486 là vị trí bit trong dòng bit 400 chứa bit thứ nhất của dữ liệu video dùng cho tập con tương ứng của phiên 424 sau khi phiên 424 được mã hóa. Khi WPP 450 được sử dụng, điểm nhập 486 là vị trí bit chứa bit thứ nhất của hàng CTB tương ứng 460-468. Như vậy, số lượng điểm nhập (NumEntryPoints) 488 là số lượng điểm nhập 486 đối với các hàng CTB 460-468.

Nhờ sử dụng phiên 424 trên Fig.4 làm ví dụ, bộ mã hóa bổ sung cuối của bit hàng CTB ở cuối tại mỗi hàng CTB 460-468 trong WPP. Cuối của bit hàng CTB báo hiệu cuối của hàng CTB 460-468 cho bộ giải mã. Bộ mã hóa sau đó thực hiện cân chỉnh bai để bổ sung các bit cân chỉnh bai như là đệm. Ngoài ra, bộ mã hóa cũng bổ sung cuối của bit phiên ở cuối của hàng CTB 468 trong WPP. Cuối của bit phiên báo hiệu cuối của phiên 424 cho bộ giải mã. Bộ mã hóa sau đó thực hiện cân chỉnh bai để bổ sung các bit cân chỉnh bai như là đệm. Do cuối của hàng CTB 468 cũng là cuối của phiên 424, nên bộ mã hóa trong WPP mã hóa cuối của bit hàng CTB và cuối của bit phiên sau khi CTB cuối cùng 430 trong hàng CTB 468 đã được tạo mã và thực hiện cân chỉnh bai hai lần. Vì vậy, có sự sao chép báo hiệu và cân chỉnh bai trong WPP.

Được bộc lộ ở đây là các kỹ thuật để ngăn chặn sự sao chép báo hiệu và cân chỉnh bai trong WPP. Bằng cách loại bỏ sự sao chép báo hiệu và cân chỉnh bai trong WPP, số lượng bit được sử dụng để báo hiệu cuối của phiên và số lượng bit được sử dụng làm đệm được giảm bớt. Nhờ làm giảm số lượng bit cần cho WPP, bộ tạo mã / bộ giải mã (cũng được gọi là “codec”) trong tạo mã video được nâng cao so với các codec hiện hành. Như một vấn đề thực tế, quy trình tạo mã video được nâng cao đem lại cho người dùng trải nghiệm người dùng tốt hơn khi các video được gửi, được nhận, và/hoặc được xem.

Không giống như WPP được mô tả trên đây, sáng chế chỉ báo hiệu cuối của bit phiên và thực hiện cân chỉnh bai chỉ một lần sau khi CTB cuối cùng 430 trong hàng CTB 468 đã được tạo mã. Khi thực hiện như vậy, số lượng bit báo hiệu và số lượng bit được sử dụng làm đệm được giảm đi so với WPP.

Fig.5 thể hiện một phương án của phương pháp 500 để giải mã dòng bit video được tạo mã được thực hiện bởi bộ giải mã video (ví dụ, bộ giải mã video 30). Phương pháp 500 có thể được thực hiện sau khi dòng bit được giải mã đã được nhận trực tiếp hoặc gián tiếp từ bộ mã hóa video (ví dụ, bộ mã hóa video 20). Phương pháp 500 nâng cao quy trình giải mã bằng cách làm giảm số lượng bit báo hiệu và số lượng bit được sử dụng làm đệm theo sau việc tạo mã CTB cuối cùng (ví dụ, CTB 430) trong hàng CTB cuối cùng (ví dụ, hàng CTB 468) của phiên (ví dụ, phiên 424). Do đó, như một vấn đề thực tế, hiệu năng của codec được cải thiện, mà dẫn đến trải nghiệm tốt hơn của người dùng.

Trong khối 502, bộ giải mã video nhận dòng bit video được tạo mã (ví dụ, dòng bit 400). Theo một phương án, dòng bit video được tạo mã chứa hình ảnh (ví dụ, hình ảnh 410). Theo một phương án, hình ảnh bao gồm một hoặc nhiều lát (ví dụ, các lát 416-420) có một hoặc nhiều phiên (ví dụ, các phiên 422-426). Theo một phương án, mỗi phiên chứa các khối cây tạo mã (ví dụ, các CTB 428-430).

Trong khối 504, bộ giải mã video kết hợp cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bại trong dòng bit video được tạo mã. Theo một phương án, cuối của bit phiên được biểu thị là `end_of_tile_one_bit`. Theo một phương án, giá trị thứ nhất là một (1). Theo một phương án, các bit cân chỉnh bại là kết quả của quy trình cân chỉnh bit được thực hiện bởi bộ mã hóa (ví dụ, bộ mã hóa video 20). Theo một phương án, cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bại chỉ báo rằng CTB hiện hành (ví dụ, CTB 430) từ các CTB (ví dụ, các CTB 428-430) là CTB cuối cùng trong phiên (ví dụ, phiên 424).

Trong khối 506, bộ giải mã video kết hợp cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bại trong dòng bit video được tạo mã. Theo một phương án, cuối của bit hàng CTB được biểu thị là `end_of_subset_bit`. Theo một phương án, giá trị thứ nhất là một (1). Theo một phương án, các bit cân chỉnh bại là kết quả của quy trình cân chỉnh bit được thực hiện bởi bộ mã hóa (ví dụ, bộ mã hóa video 20). Theo một phương án, cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bại chỉ báo rằng WPP là được phép và rằng CTB hiện hành (ví dụ, CTB 430) từ các CTB (ví dụ, các

CTB 428-430) là CTB cuối cùng trong hàng CTB (ví dụ, các hàng CTB 460-466) nhưng không phải CTB cuối cùng trong phiên (ví dụ, phiên 424).

Ở khối 508, bộ giải mã video tái tạo các CTB trong phiên dựa trên cuối của bit phiên với giá trị thứ nhất, cuối của bit hàng CTB với giá trị thứ nhất, và các bit cân chỉnh bai. Theo một phương án, ảnh được tạo ra dựa trên các CTB như được tái tạo. Theo một phương án, ảnh có thể được hiển thị cho người dùng thiết bị điện tử (ví dụ, điện thoại thông minh, máy tính bảng, máy tính xách tay loại laptop, máy tính cá nhân, v.v.).

Fig.6 thể hiện một phương án của phương pháp 600 để mã hóa dòng bit video được thực hiện bởi bộ mã hóa video (ví dụ, bộ mã hóa video 20). Phương pháp 600 có thể được thực hiện khi hình ảnh (ví dụ, từ video) cần được mã hóa thành dòng bit video và sau đó được truyền đến bộ giải mã video (ví dụ, bộ giải mã video 30). Phương pháp 600 nâng cao quy trình mã hóa bằng cách làm giảm số lượng bit báo hiệu và số lượng bit được sử dụng làm đệm theo sau việc tạo mã CTB cuối cùng (ví dụ, CTB 430) trong hàng CTB cuối cùng (ví dụ, hàng CTB 468) của phiên (ví dụ, phiên 424). Do đó, như một vấn đề thực tế, hiệu năng của codec được cải thiện, mà dẫn đến trải nghiệm tốt hơn của người dùng.

Trong khối 602, bộ mã hóa video phân chia hình ảnh (ví dụ, hình ảnh 410) thành một hoặc nhiều lát (ví dụ, các lát 416-420). Theo một phương án, mỗi lát chứa một hoặc nhiều phiên (ví dụ, các phiên 422-426). Theo một phương án, mỗi phiên chứa các khối cây tạo mã (ví dụ, các CTB 428-430).

Ở khối 604, bộ mã hóa video mã hóa cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bai thành dòng bit video khi CTB hiện hành từ các CTB là CTB cuối cùng trong phiên. Theo một phương án, cuối của bit phiên được biểu thị là `end_of_tile_one_bit`. Theo một phương án, giá trị thứ nhất là một (1). Theo một phương án, các bit cân chỉnh bai là kết quả của quy trình cân chỉnh bit được thực hiện bởi bộ mã hóa (ví dụ, bộ mã hóa video 20). Theo một phương án, cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bai chỉ báo rằng CTB hiện hành (ví dụ, CTB 430) từ các CTB (ví dụ, các CTB 428-430) là CTB cuối cùng trong phiên (ví dụ, phiên 424).

Ở khối 606, bộ mã hóa video mã hóa cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai thành dòng bit video khi WPP được phép và khi CTB hiện hành là CTB cuối cùng trong hàng CTB nhưng không phải CTB cuối cùng trong phiên. Theo một phương án, cuối của bit hàng CTB được biểu thị là `end_of_subset_bit`. Theo một phương án, giá trị thứ nhất là một (1). Theo một phương án, các bit cân chỉnh bai là kết quả của quy trình cân chỉnh bit được thực hiện bởi bộ mã hóa (ví dụ, bộ mã hóa video 20). Theo một phương án, cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai chỉ báo rằng WPP là được phép và rằng CTB hiện hành (ví dụ, CTB 430) từ các CTB (ví dụ, các CTB 428-430) là CTB cuối cùng trong hàng CTB (ví dụ, các hàng CTB 460-466) nhưng không phải CTB cuối cùng trong phiên (ví dụ, phiên 424).

Ở khối 608, bộ mã hóa video lưu trữ dòng bit video để truyền đến bộ giải mã video. Theo một phương án, bộ mã hóa video truyền dòng bit video đến bộ giải mã video.

Cú pháp và các ngữ nghĩa sau đây có thể được sử dụng để thực hiện các phương án được bộc lộ ở đây. Phần mô tả sau đây có liên quan đến văn bản cơ bản, mà là đặc tả phác thảo VVC mới nhất. Nói cách khác, chỉ delta được mô tả, trong khi các văn bản trong văn bản cơ bản mà không được nêu dưới đây áp dụng đúng như chính chúng. Phần văn bản bổ sung vào văn bản cơ bản được thể hiện đậm, và phần văn bản được loại bỏ được thể hiện nghiêng.

Fig.7 là hình vẽ giản lược của thiết bị tạo mã video 700 (ví dụ, bộ mã hóa video 20 hoặc bộ giải mã video 30) theo một phương án của sáng chế. Thiết bị tạo mã video 700 thích hợp để thực hiện các phương án được bộc lộ như được mô tả ở đây. Thiết bị tạo mã video 700 bao gồm các cổng vào 710 và các bộ phận nhận (Rx) 720 để nhận dữ liệu; bộ xử lý, bộ phận logic, hoặc bộ xử lý trung tâm (CPU) 730 để xử lý dữ liệu; các bộ phận truyền (Tx) 740 và các cổng ra 750 để truyền dữ liệu; và bộ nhớ 760 để lưu trữ dữ liệu. Thiết bị tạo mã video 700 cũng có thể bao gồm các thành phần quang sang điện (optical-to-electrical, OE) và các thành phần điện sang quang (electrical-to-optical, EO) được ghép nối với các cổng vào 710, các bộ phận nhận 720, các bộ phận truyền 740, và các cổng ra 750 để lấy ra hoặc lấy vào các tín hiệu điện hoặc quang.

Bộ xử lý 730 được thực thi bởi phần cứng và phần mềm. Bộ xử lý 730 có thể được thực thi như là một hoặc nhiều chip CPU, nhân (ví dụ, bộ xử lý nhiều nhân), mảng công lập trình được dạng trường (field-programmable gate array, FPGA), mạch tích hợp chuyên dụng (application specific integrated circuit, ASIC), và bộ xử lý tín hiệu số (digital signal processor, DSP). Bộ xử lý 730 truyền thông với các cổng vào 710, các bộ phận nhận 720, các bộ phận truyền 740, các cổng ra 750, và bộ nhớ 760. Bộ xử lý 730 bao gồm môđun tạo mã 770. Môđun tạo mã 770 thực thi các phương án được bộc lộ được mô tả trên đây. Ví dụ, môđun tạo mã 770 thực hiện, xử lý, chuẩn bị, hoặc cung cấp các chức năng codec khác nhau. Việc bao gồm môđun tạo mã 770 do đó sẽ mang lại sự cải tiến quan trọng cho chức năng của thiết bị tạo mã video 700 và đem lại sự biến đổi thiết bị tạo mã video 700 sang trạng thái khác. Theo cách khác, môđun tạo mã 770 được thực thi dưới dạng các lệnh được lưu trữ trong bộ nhớ 760 và được thực hiện bởi bộ xử lý 730.

Thiết bị tạo mã video 700 cũng có thể bao gồm các thiết bị đầu vào và/hoặc đầu ra (I/O) 780 để truyền thông dữ liệu đến và từ người dùng. Các thiết bị I/O 780 có thể bao gồm thiết bị đầu ra như bộ hiển thị để hiển thị dữ liệu video, loa để kết xuất dữ liệu audio, v.v.. Các thiết bị I/O 780 cũng có thể bao gồm các thiết bị đầu vào, như bàn phím, chuột, bi xoay, v.v., và/hoặc các giao diện tương ứng để tương tác với thiết bị đầu ra đó.

Bộ nhớ 760 bao gồm một hoặc nhiều đĩa, ổ băng, và ổ trạng thái rắn và có thể được sử dụng làm thiết bị lưu trữ dữ liệu tràn, để lưu trữ các chương trình khi các chương trình này được chọn để thực hiện, và để lưu trữ các lệnh và dữ liệu được đọc trong suốt quá trình thực hiện chương trình. Bộ nhớ 760 có thể là khả biến và/hoặc bất khả biến và có thể là bộ nhớ chỉ đọc (ROM), bộ nhớ truy cập ngẫu nhiên (RAM), bộ nhớ khả lập địa chỉ nội dung bậc ba (ternary content-addressable memory, TCAM), và/hoặc bộ nhớ truy cập ngẫu nhiên tĩnh (SRAM).

Fig.8 là hình vẽ giản lược của một phương án của phương tiện tạo mã 800. Theo một phương án, phương tiện tạo mã 800 được thực thi trong thiết bị tạo mã video 802 (ví dụ, bộ mã hóa video 20 hoặc bộ giải mã video 30). Thiết bị tạo mã video 802 bao gồm phương tiện nhận 801. Phương tiện nhận 801 được tạo cấu hình để nhận hình ảnh để mã hóa hoặc để nhận dòng bit để giải mã. Thiết bị tạo mã video 802 bao gồm phương

tiện truyền 807 được ghép nối với phương tiện nhận 801. Phương tiện truyền 807 được tạo cấu hình để truyền dòng bit đến bộ giải mã hoặc để truyền ảnh đã giải mã đến phương tiện hiển thị (ví dụ, một trong số các thiết bị I/O 780).

Thiết bị tạo mã video 802 bao gồm phương tiện lưu trữ 803. Phương tiện lưu trữ 803 được ghép nối với ít nhất một phương tiện trong số phương tiện nhận 801 hoặc phương tiện truyền 807. Phương tiện lưu trữ 803 được tạo cấu hình để lưu trữ các lệnh. Thiết bị tạo mã video 802 cũng bao gồm phương tiện xử lý 805. Phương tiện xử lý 805 được ghép nối với phương tiện lưu trữ 803. Phương tiện xử lý 805 được tạo cấu hình để thực hiện các lệnh được lưu trữ trong phương tiện lưu trữ 803 để thực hiện các phương pháp được bộc lộ ở đây.

Cũng cần 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 pháp này nên được hiểu là chỉ làm ví dụ. Tương tự, các bước bổ sung có thể có trong các phương pháp này, 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.

Mặc dù một số phương án đã được đề xuất trong sáng chế, nhưng cần hiểu rằng các hệ thống và các phương pháp được bộc lộ có thể có trong nhiều dạng cụ thể khác mà không nằm ngoài mục đích hoặc phạm vi của sáng chế. Các ví dụ này cần được coi là minh họa và không phải giới hạn, và sáng chế không bị giới hạn ở các nội dung chi tiết được đưa ra ở đây. Ví dụ, các phần tử hoặc các thành phần khác nhau có thể được kết hợp hoặc được tích hợp trong 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 trong các phương án khác nhau rời rạc hoặc riêng rẽ có thể được kết hợp hoặc được tích hợp với các hệ thống, các môđun, các kỹ thuật, hoặc các phương pháp khác mà không nằm ngoài phạm vi của sáng chế. Các mục được thể hiện hoặc thảo luận là được ghép nối hoặc được ghép nối trực tiếp hoặc truyền thông với nhau có thể được ghép nối gián tiếp hoặc truyền thông qua giao diện, thiết bị, hoặc thành phần trung gian nào đó bằng điện, cơ học, hoặc loại khác. Các ví dụ khác về các sự thay đổi, thay thế, và cải biến có thể biết được bởi người có hiểu biết trung bình trong

lĩnh vực tương ứng và có thể được tạo ra mà không nằm ngoài mục đích và phạm vi được bộc lộ ở đây.

## YÊU CẦU BẢO HỘ

1. Phương pháp giải mã được thực hiện bởi bộ giải mã video, trong đó phương pháp này bao gồm các bước: nhận dòng bit video được tạo mã, trong đó dòng bit video được tạo mã bao gồm hình ảnh, hình ảnh này bao gồm một hoặc nhiều lát (slice) có một hoặc nhiều phiên, mỗi phiên (tile) bao gồm nhiều khối cây tạo mã (coding tree block, CTB); thu nhận cuối của bit phiên với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video được tạo mã, trong đó cuối của bit phiên với giá trị thứ nhất và bit cân chỉnh bai chỉ báo CTB hiện hành trong các CTB là CTB cuối cùng trong phiên; thu nhận cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video được tạo mã, trong đó cuối của bit hàng CTB với giá trị thứ nhất và bit cân chỉnh bai chỉ báo rằng việc xử lý song song dạng sóng (waveform parallel processing, WPP) được cho phép, và rằng CTB hiện hành từ các CTB này là CTB cuối cùng trong hàng CTB, nhưng không phải CTB cuối cùng trong phiên; tái tạo các CTB trong phiên dựa trên cuối của bit phiên với giá trị thứ nhất, cuối của bit hàng CTB với giá trị thứ nhất, và các bit cân chỉnh bai.
2. Phương pháp theo điểm 1, trong đó cuối của bit phiên được biểu thị là `end_of_tile_one_bit`.
3. Phương pháp theo điểm bất kỳ trong số các điểm 1 và 2, trong đó cuối của bit hàng CTB được biểu thị là `end_of_subset_bit`.
4. Phương pháp theo điểm 1, trong đó WPP được cho phép bởi cờ được bố trí trong tập hợp thông số.
5. Phương pháp theo điểm 4, trong đó WPP được cho phép bởi cờ được biểu thị là `entropy_coding_sync_enabled_flag`.
6. Phương pháp theo điểm 4, trong đó khi WPP được cho phép, giá trị thứ nhất là một (1).
7. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước: hiển thị một hình ảnh được tạo ra theo các CTB được tái tạo.
8. Thiết bị giải mã, trong đó thiết bị giải mã bao gồm: bộ nhận được tạo cấu hình để nhận dòng bit video được tạo mã; bộ nhớ được ghép nối với bộ nhận, trong đó bộ nhớ



lưu trữ các lệnh; bộ xử lý được ghép nối với bộ nhớ, trong đó bộ xử lý được sử dụng để thực hiện lệnh, sao cho thiết bị giải mã thực hiện các hoạt động sau đây: nhận dòng bit video được tạo mã, trong đó dòng bit video được tạo mã bao gồm hình ảnh, hình ảnh này bao gồm một hoặc nhiều lát (slice) có một hoặc nhiều phiến, mỗi phiến (tile) bao gồm nhiều khối cây tạo mã (CTB); thu nhận cuối của bit phiến với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video được tạo mã, trong đó cuối của bit phiến với giá trị thứ nhất và bit cân chỉnh bai chỉ báo CTB hiện hành trong các CTB là CTB cuối cùng trong phiến; thu nhận cuối của bit hàng CTB với giá trị thứ nhất và các bit cân chỉnh bai trong dòng bit video được tạo mã, trong đó cuối của bit hàng CTB với giá trị thứ nhất và bit cân chỉnh bai chỉ báo rằng việc xử lý song song dạng sóng (WPP) được cho phép, và rằng CTB hiện hành từ các CTB này là CTB cuối cùng trong hàng CTB, nhưng không phải CTB cuối cùng trong phiến; tái tạo các CTB trong phiến dựa trên cuối của bit phiến với giá trị thứ nhất, cuối của bit hàng CTB với giá trị thứ nhất, và các bit cân chỉnh bai.

9. Thiết bị giải mã theo điểm 8, trong đó cuối của bit phiến được biểu thị là `end_of_tile_one_bit`.

10. Thiết bị giải mã theo điểm 8 hoặc 9, trong đó WPP được cho phép bởi cờ được bố trí trong tập hợp thông số.

11. Thiết bị giải mã theo điểm 10, trong đó WPP được cho phép bởi cờ được biểu thị là `entropy_coding_sync_enabled_flag`.

12. Thiết bị giải mã, trong đó thiết bị giải mã bao gồm: bộ nhận, để nhận dòng bit để giải mã; bộ truyền được ghép nối với bộ nhận, trong đó bộ truyền được sử dụng để truyền hình ảnh được giải mã đến bộ hiển thị; bộ nhớ, được ghép nối với ít nhất một trong số bộ nhận hoặc bộ truyền, trong đó bộ nhớ được sử dụng để lưu trữ các lệnh; bộ xử lý, được ghép nối với bộ nhớ được sử dụng để thực hiện các lệnh được lưu trữ trong bộ nhớ, để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.

13. Thiết bị giải mã theo điểm 12, trong đó thiết bị giải mã còn bao gồm bộ hiển thị để hiển thị hình ảnh.

14. Hệ thống giải mã, trong đó hệ thống này bao gồm: bộ mã hóa; bộ giải mã, truyền thông với bộ mã hóa, trong đó bộ giải mã bao gồm thiết bị giải mã theo điểm bất kỳ trong số các điểm từ 8 đến 11 hoặc thiết bị giải mã theo điểm bất kỳ trong số các điểm từ 12 đến 13.

15. Môđun giải mã, trong đó môđun giải mã bao gồm: môđun nhận để nhận dòng bit để giải mã; môđun gửi, được ghép nối với môđun gửi được sử dụng để gửi hình ảnh được giải mã đến môđun hiển thị; môđun lưu trữ, được ghép nối với ít nhất một trong số môđun nhận hoặc môđun gửi, trong đó môđun lưu trữ được sử dụng để lưu trữ các lệnh; môđun xử lý, được ghép nối với môđun xử lý được sử dụng để thực hiện lệnh được lưu trữ trong môđun lưu trữ, để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.

16. Bộ giải mã video, trong đó bộ giải mã này bao gồm: bộ nhớ để lưu trữ dòng bit video được tạo mã; bộ giải mã, để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.

17. Phương tiện lưu trữ đọc được bằng máy tính, trong đó phương tiện lưu trữ đọc được bằng máy tính lưu trữ chương trình để được thực hiện bởi bộ xử lý, chương trình này làm cho bộ xử lý thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.

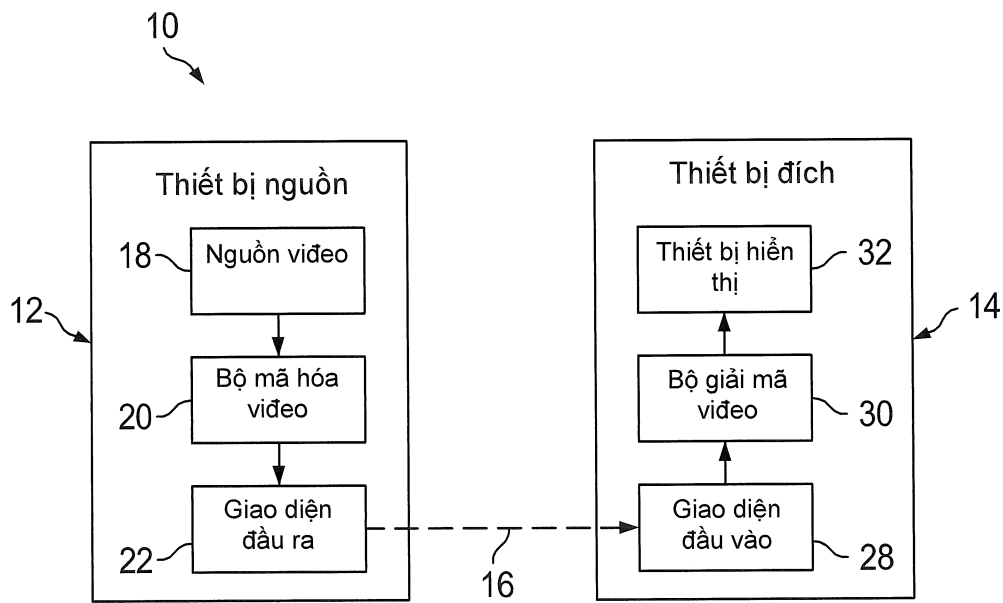


FIG. 1

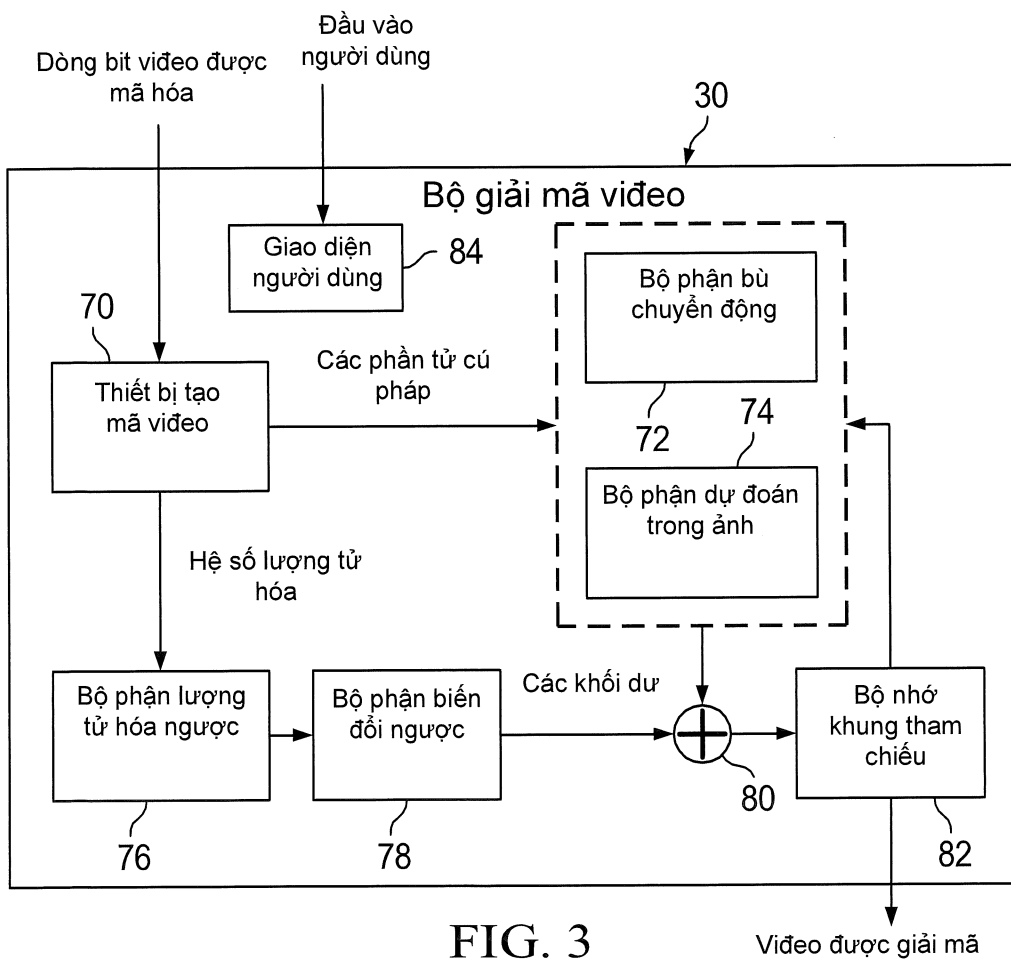


FIG. 3

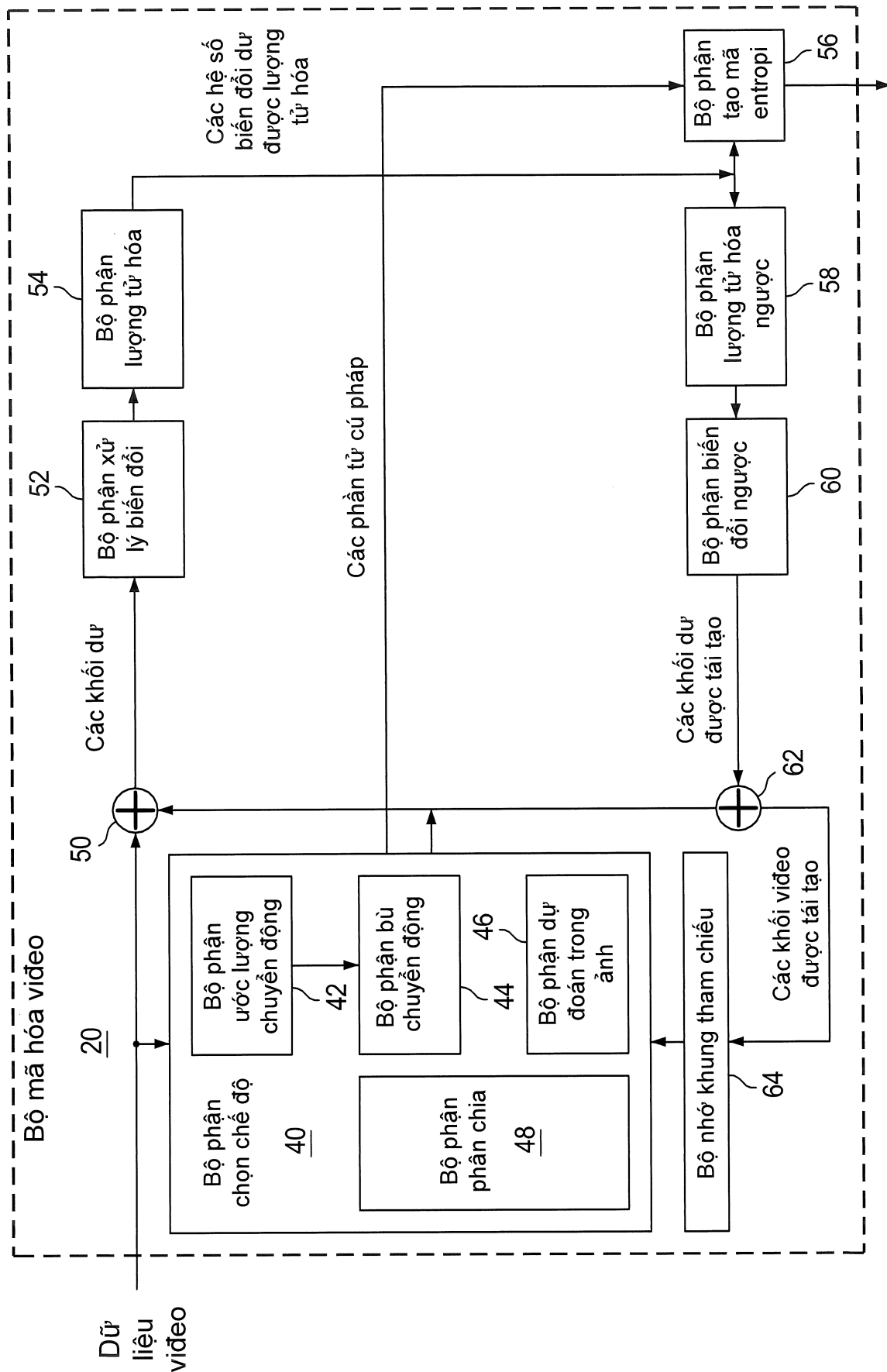
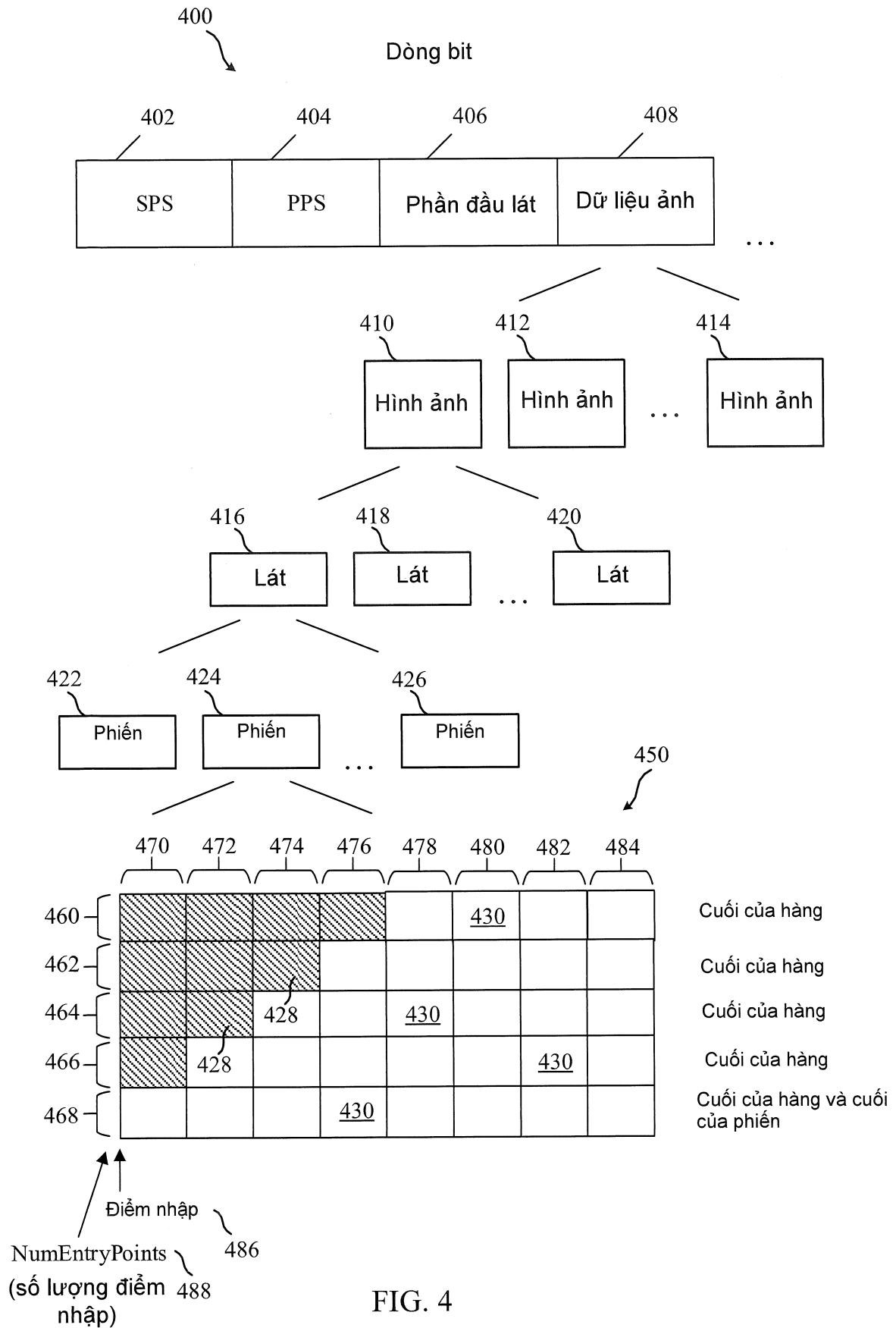


FIG. 2

3/7



500

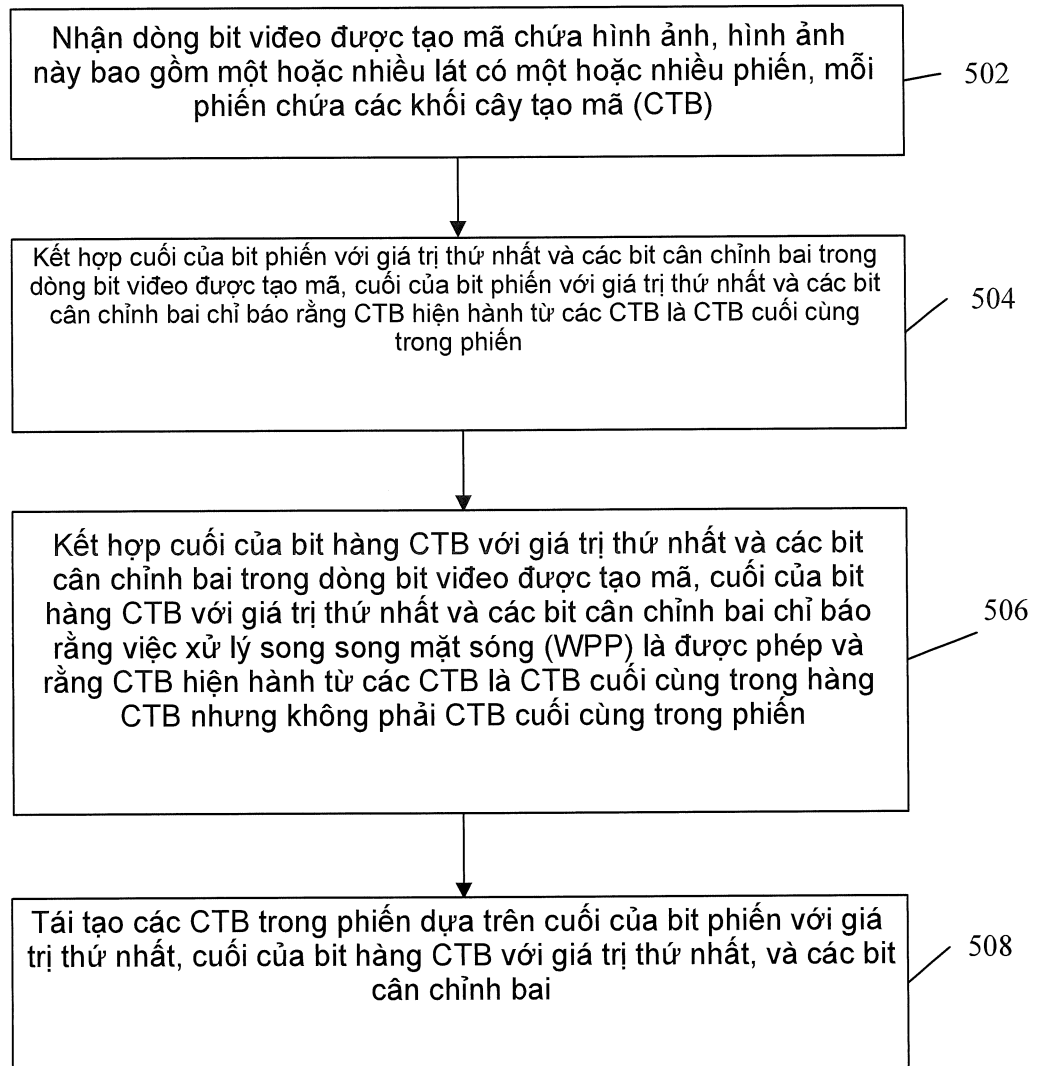


FIG. 5

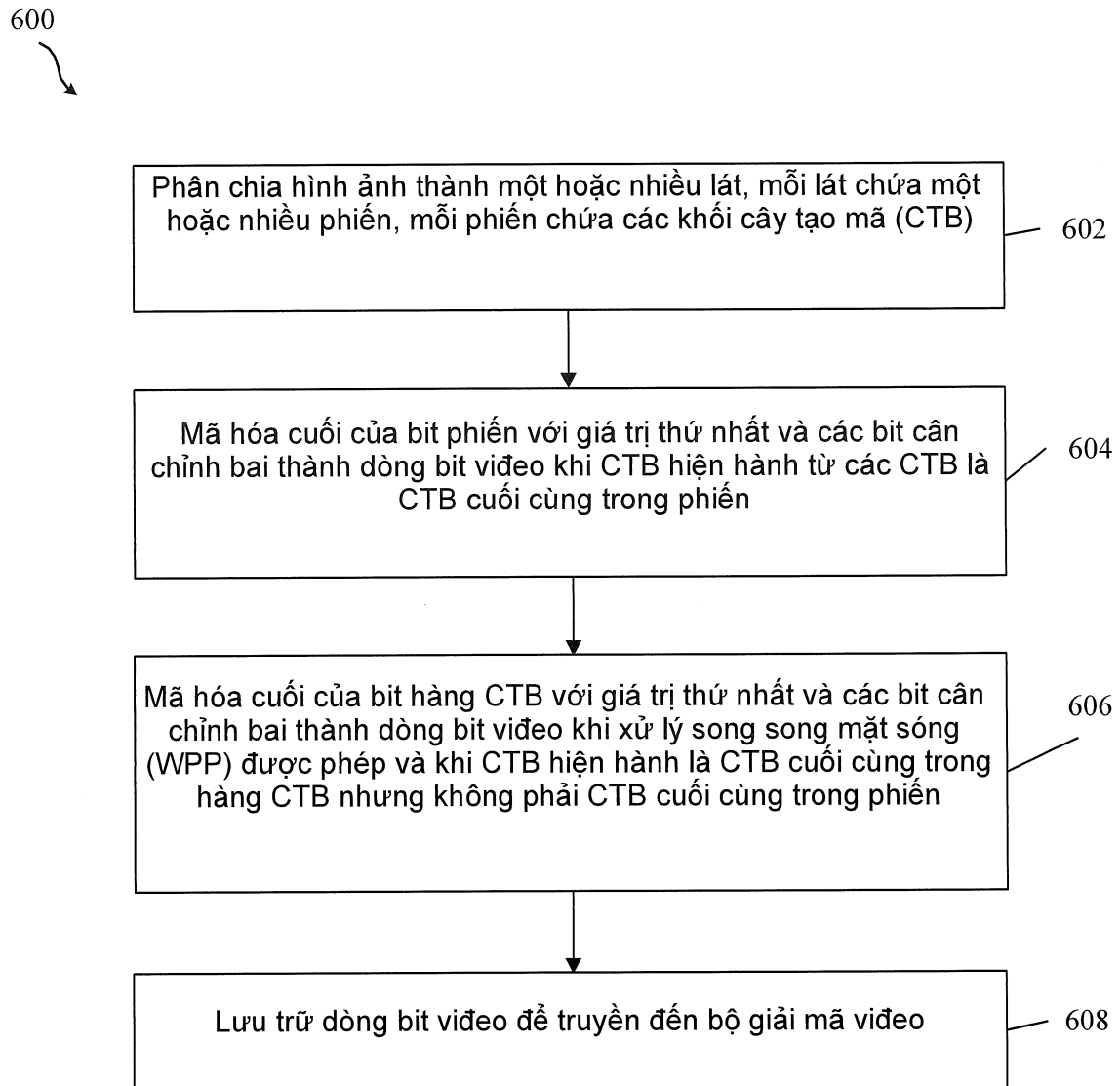


FIG. 6

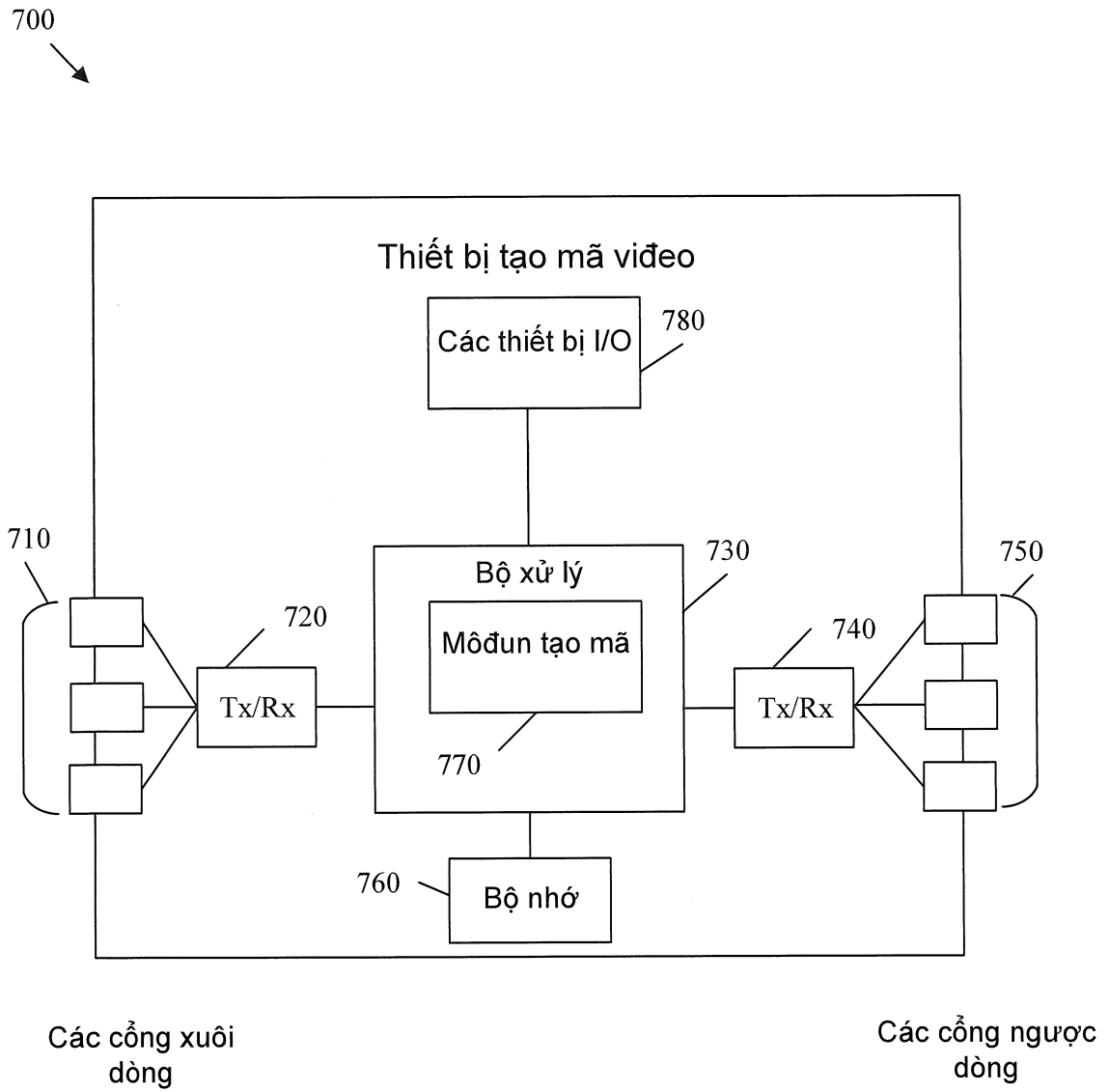


FIG. 7



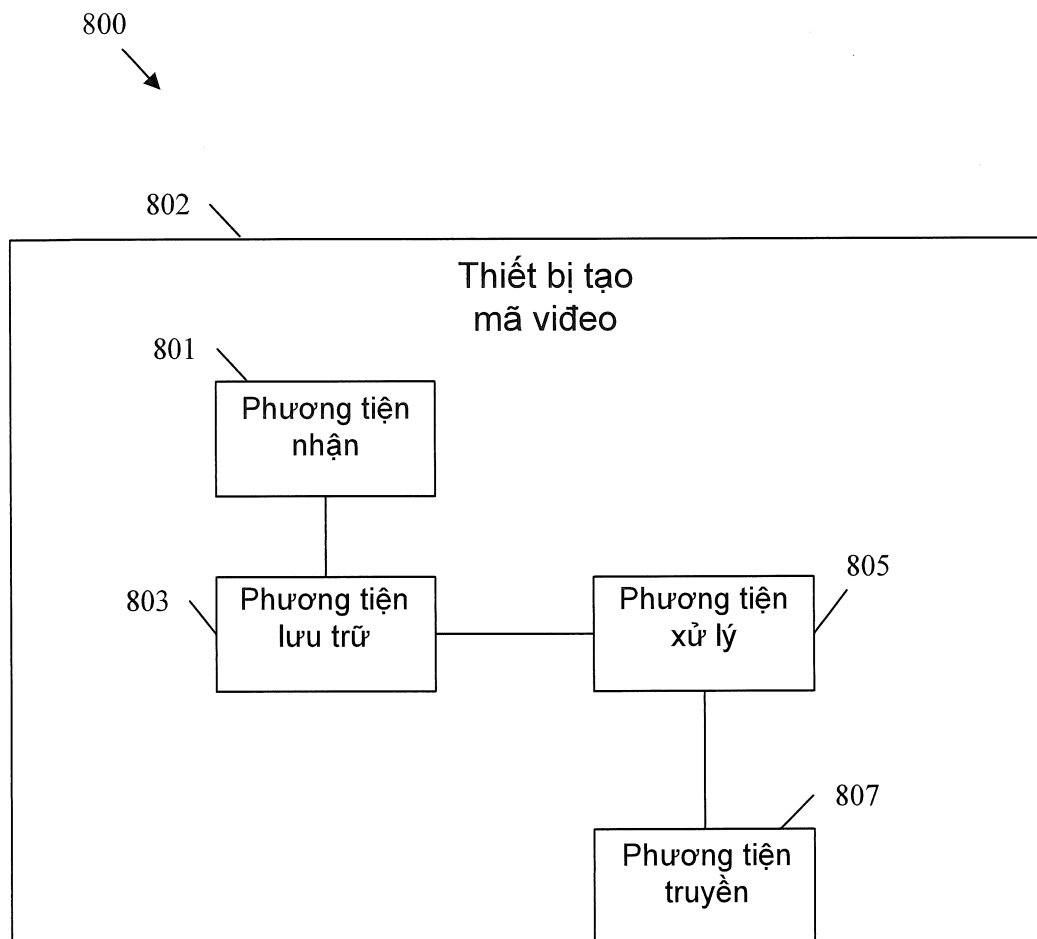


FIG. 8