



(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)



1-0025916

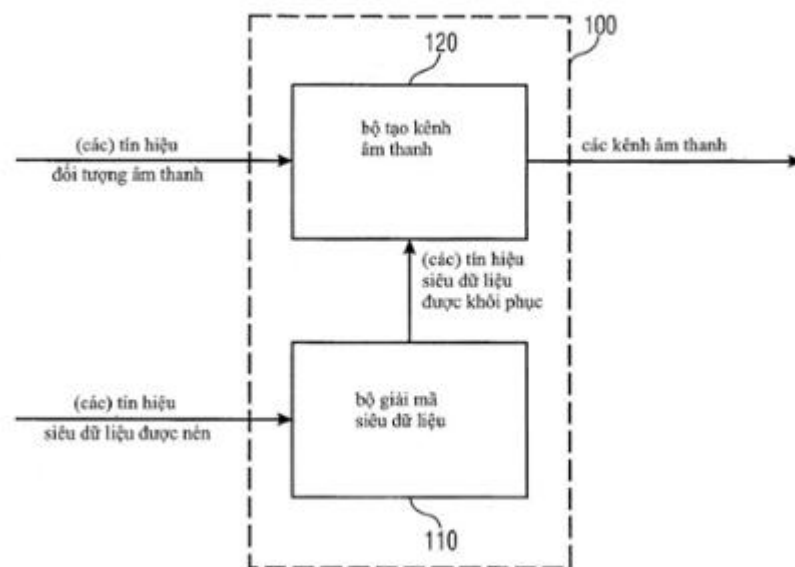
(51)⁷ G10L 19/008

(13) B

- (21) 1-2016-00554 (22) 16/07/2014
(86) PCT/EP2014/065299 16/07/2014 (87) WO 2015/011000 29/01/2015
(30) 13177365.7 22/07/2013 EP; 13177367.3 22/07/2013 EP; 13177378.0 22/07/2013 EP;
13189284.6 18/10/2013 EP
(45) 26/10/2020 391 (43) 25/05/2016 338A
(73) Fraunhofer-Gesellschaft zur Foerderung der angewandten Forschung e.V. (DE)
Hansastrasse 27c, 80686 Muenchen, Germany
(72) BORSS, Christian (DE); ERTEL, Christian (DE).
(74) CÔNG TY LUẬT TRÁCH NHIỆM HỮU HẠN AMBYS HÀ NỘI (AMBYS
HANOI)

(54) THIẾT BỊ VÀ PHƯƠNG PHÁP TẠO RA MỘT HOẶC NHIỀU KÊNH ÂM THANH, THIẾT BỊ VÀ PHƯƠNG PHÁP TẠO RA THÔNG TIN ÂM THANH ĐƯỢC MÃ HÓA, HỆ THỐNG TẠO RA THÔNG TIN ÂM THANH ĐƯỢC MÃ HÓA VÀ NHẬN MỘT HOẶC NHIỀU TÍN HIỆU ÂM THANH ĐƯỢC MÃ HÓA, THIẾT BỊ MÃ HÓA DỮ LIỆU ĐẦU VÀO ÂM THANH, THIẾT BỊ GIẢI MÃ DỮ LIỆU ÂM THANH ĐƯỢC MÃ HÓA

(57) Sáng chế đề cập đến thiết bị và phương pháp tạo ra một hoặc nhiều kênh âm thanh, thiết bị và phương pháp tạo ra thông tin âm thanh được mã hóa, hệ thống tạo ra thông tin âm thanh được mã hóa và nhận một hoặc nhiều tín hiệu âm thanh được mã hóa, thiết bị mã hóa dữ liệu đầu vào âm thanh, thiết bị giải mã dữ liệu âm thanh được mã hóa. Thiết bị (100) tạo ra một hoặc nhiều kênh âm thanh được đề xuất. Thiết bị (100) bao gồm bộ giải mã siêu dữ liệu (110) để nhận một hoặc nhiều tín hiệu siêu dữ liệu được nén. Mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhiều mẫu siêu dữ liệu thứ nhất. Các mẫu siêu dữ liệu thứ nhất của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén biểu thị thông tin được đính kèm với tín hiệu đối tượng âm thanh trong số một hoặc nhiều tín hiệu đối tượng âm thanh. Bộ giải mã siêu dữ liệu (110) được tạo cấu hình để tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục, để mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm các mẫu siêu dữ liệu thứ nhất của một trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén và còn bao gồm mẫu siêu dữ liệu thứ hai. Hơn nữa, bộ giải mã siêu dữ liệu (110) được tạo cấu hình để tạo ra mỗi mẫu trong số các mẫu siêu dữ liệu thứ hai của mỗi tín hiệu siêu dữ liệu được khôi phục trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục phụ thuộc vào ít nhất hai trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu. Hơn nữa, thiết bị (100) bao gồm bộ tạo kênh âm thanh (120) để tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục. Hơn nữa, thiết bị tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén được đề xuất.



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến sự mã hóa/sự giải mã âm thanh, cụ thể, đến sự lập mã âm thanh không gian và sự lập mã đối tượng âm thanh không gian, và cụ thể hơn, đến thiết bị và phương pháp cho sự lập mã siêu dữ liệu đối tượng hiệu quả.

Tình trạng kỹ thuật của sáng chế

Các công cụ lập mã âm thanh không gian được biết đến rộng rãi trong lĩnh vực và, ví dụ, được tiêu chuẩn hóa trong tiêu chuẩn âm thanh nổi MPEG. Sự lập mã âm thanh không gian bắt đầu từ các kênh đầu vào ban đầu như năm hoặc bảy kênh mà được nhận biết bởi sự thay thế trong thiết lập tái tạo, tức là, kênh bên trái, kênh ở giữa, kênh bên phải, kênh âm thanh nổi bên trái, kênh âm thanh nổi bên phải và kênh nâng cao tần số thấp. Bộ mã hóa âm thanh không gian thường rút một hoặc nhiều kênh trộn giảm từ các kênh ban đầu và, ngoài ra, rút dữ liệu tham số liên quan đến các ký hiệu điều khiển không gian như các chênh lệch mức liên kênh trong các giá trị nhất quán kênh, các chênh lệch pha liên kênh, các chênh lệch thời gian liên kênh, v.v. Một hoặc nhiều kênh trộn giảm được truyền dẫn cho nhau với thông tin phụ theo tham số biểu thị các ký hiệu điều khiển không gian cho bộ giải mã âm thanh trong không gian mà giải mã kênh trộn giảm và dữ liệu tham số được kết hợp để cuối cùng thu được các kênh đầu ra mà là phiên bản được làm gần đúng của các kênh đầu vào ban đầu. Sự sắp đặt các kênh trong thiết lập đầu ra thường cố định và, ví dụ, là định dạng 5.1, định dạng 7.1, v.v.

Các định dạng âm thanh trên cơ sở kênh như vậy được sử dụng rộng rãi để lưu trữ hoặc truyền dẫn nội dung âm thanh đa kênh trong đó mỗi kênh liên quan đến loa phát thanh cụ thể tại vị trí đã cho. Sự tái tạo trung thành của các loại định dạng này yêu cầu thiết lập loa phát thanh trong đó các loa được đặt tại cùng vị trí như các loa mà

đã được sử dụng trong suốt sự sản xuất các tín hiệu âm thanh. Trong khi tăng số lượng các loa phát thanh cải thiện sự tái tạo của các ngữ cảnh âm thanh 3D thực sự chìm, nó trở nên khó khăn hơn để thỏa mãn yêu cầu này - đặc biệt trong môi trường gia đình như phòng khách.

Sự cần thiết của việc có thiết lập loa phát thanh có thể được khắc phục bằng phương pháp trên cơ sở đối tượng trong đó các tín hiệu loa phát thanh được kết xuất đặc biệt cho thiết lập phát lại.

Ví dụ, các công cụ lập mã đối tượng âm thanh không gian được biết đến rộng rãi trong lĩnh vực kỹ thuật và được tiêu chuẩn hóa trong tiêu chuẩn lập mã đối tượng âm thanh không gian (spatial audio object coding - SAOC) MPEG. Trái với sự lập mã âm thanh không gian bắt đầu từ các kênh ban đầu, sự lập mã đối tượng âm thanh không gian bắt đầu từ các đối tượng âm thanh mà không tự động chuyên dùng cho sự thiết lập tái tạo kết xuất nhất định. Thay vào đó, sự sắp đặt các đối tượng âm thanh trong ngữ cảnh tái tạo là có thể linh động và có thể được xác định bởi người dùng, bằng cách nhập thông tin kết xuất nhất định vào bộ giải mã lập mã đối tượng âm thanh không gian. Ngoài ra hoặc cách khác, thông tin kết xuất, tức là, thông tin tại đó vị trí trong thiết lập sự tái tạo đối tượng âm thanh nhất định thường được đặt theo thời gian có thể được truyền dẫn như thông tin phụ hoặc siêu dữ liệu bổ sung. Để thu được sự nén dữ liệu nhất định, số lượng các đối tượng âm thanh được mã hóa bằng bộ mã hóa SAOC mà tính toán, từ các đối tượng âm thanh đầu vào, một hoặc nhiều kênh vận chuyển bằng cách trộn giảm các đối tượng phù hợp với thông tin trộn giảm. Hơn nữa, bộ mã hóa SAOC tính toán thông tin phụ theo tham số biểu diễn các tín hiệu điều khiển như sự chênh lệch mức đối tượng (object level differences - OLD), các giá trị cố kết đối tượng, v.v. Như trong sự lập mã âm thanh không gian (Spatial Audio Coding – SAC), dữ liệu tham số liên đối tượng được tính toán cho các ô thời gian/tần số riêng, tức là, đối với khung nhất định của tín hiệu âm thanh bao gồm, ví dụ, 1024 hoặc 2048 mẫu, 24, 32, hoặc 64, v.v., các băng tần được xem xét để, cuối cùng, dữ liệu theo tham số tồn tại cho mỗi khung và mỗi băng tần. Như ví dụ, khi đoạn âm thanh có 20 khung và khi mỗi khung được chia nhỏ thành 32 băng tần, vậy số lượng ô thời gian/tần số là 640.

Trong phương pháp trên cơ sở đối tượng, trường âm được mô tả bằng các đối tượng âm thanh rời rạc. Điều này yêu cầu siêu dữ liệu đối tượng mà mô tả với nhau vị trí biến thiên thời gian của mỗi nguồn âm trong không gian 3D.

Khái niệm lập mã siêu dữ liệu thứ nhất trong lĩnh vực kỹ thuật trong định dạng trao đổi mô tả âm không gian (spatial sound description interchange format - SpatDIF), định dạng mô tả ngữ cảnh âm thanh mà vẫn đang phát triển [1]. Nó được thiết kế như định dạng trao đổi cho các ngữ cảnh âm trên cơ sở đối tượng và không cung cấp bất kỳ phương pháp nén nào cho các quỹ đạo đối tượng. SpatDIF sử dụng định dạng sự điều khiển âm mở (Open Sound Control - OSC) trên cơ sở văn bản để kết cấu siêu dữ liệu đối tượng [2]. Sự biểu diễn trên cơ sở văn bản đơn giản, tuy nhiên, không phải là sự lựa chọn cho sự truyền dẫn được nén của các quỹ đạo đối tượng.

Khái niệm siêu dữ liệu khác trong tình trạng kỹ thuật là định dạng mô tả ngữ cảnh âm thanh (Audio Scene Description Format - ASDF) [3], giải pháp trên cơ sở văn bản mà có chung bất lợi. Dữ liệu được kết cấu bằng sự mở rộng của ngôn ngữ tích hợp đa phương tiện được đồng bộ hóa (Synchronized Multimedia Integration Language - SMIL) mà là tập hợp con của Ngôn ngữ đánh dấu mở rộng (Extensible Markup Language - XML) [4,5].

Khái niệm siêu dữ liệu khác trong tình trạng kỹ thuật là định dạng nhị phân âm thanh cho ngữ cảnh (AudioBIFS), định dạng nhị phân mà là phần của sự mô tả MPEG-4 [6,7]. Nó đề cập một cách kỹ lưỡng đến Ngôn ngữ mô hình hóa thực tế ảo (Virtual Reality Modeling Language - VRML) trên cơ sở XML mà được phát triển cho sự mô tả ngữ cảnh 3D âm thanh - hình ảnh và các ứng dụng thực tế ảo tương tác [8]. Sự mô tả AudioBIFS phức hợp sử dụng các đồ thị ngữ cảnh để chỉ rõ các đường đi của các di chuyển đối tượng. Bất lợi lớn của AudioBIFS là nó không được thiết kế cho thao tác thời gian thực trong đó độ trễ hệ thống được giới hạn và truy cập ngẫu nhiên đến dòng dữ liệu là yêu cầu. Hơn nữa, sự mã hóa của các vị trí đối tượng không khai thác hiệu suất định vị bị giới hạn của người nghe. Đối với vị trí người nghe cố định nằm trong ngữ cảnh âm thanh - hình ảnh, dữ liệu đối tượng có thể được lượng tử hóa với số lượng các bit ít hơn nhiều [9]. Do đó, sự mã hóa siêu dữ liệu đối tượng mà được áp dụng trong AudioBIFS không hiệu quả đối với sự nén dữ liệu.

Do đó nó sẽ được đánh giá cao, nếu được cải thiện, các khái niệm lập mã siêu dữ liệu đối tượng hiệu quả được đề xuất.

Tài liệu tham khảo

- [1] Peters, N., Lossius, T. and Schacher J. C., "SpatDIF: Principles, Specification, and Examples", 9th Sound and Music Computing Conference, Copenhagen, Denmark, Jul. 2012.
- [2] Wright, M., Freed, A., "Open Sound Control: A New Protocol for Communicating with Sound Synthesizers", International Computer Music Conference, Thessaloniki, Greece, 1997.
- [3] Matthias Geier, Jens Ahrens, and Sascha Spors. (2010), "Object-based audio reproduction and the audio scene description format", Org. Sound, Vol. 15, No. 3, pp. 219-227, December 2010.
- [4] W3C, "Synchronized Multimedia Integration Language (SMIL 3.0)", Dec. 2008.
- [5] W3C, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", Nov. 2008.
- [6] MPEG, "ISO/IEC International Standard 14496-3 - Coding of audio-visual objects, Part 3 Audio", 2009.
- [7] Schmidt, J.; Schroeder, E. F. (2004), "New and Advanced Features for Audio Presentation in the MPEG-4 Standard", 116th AES Convention, Berlin, Germany, May 2004
- [8] Web3D, "International Standard ISO/IEC 14772-1:1997 - The Virtual Reality Modeling Language (VRML), Part 1: Functional specification and UTF-8 encoding", 1997.
- [9] Sporer, T. (2012), "Codierung räumlicher Audiosignale mit leicht-gewichtigen Audio-Objekten", Proc. Annual Meeting of the German Audiological Society (DGA), Erlangen, Germany, Mar. 2012.
- [10] Ramer, U. (1972), "An iterative procedure for the polygonal approximation of plane curves", Computer Graphics and Image Processing, 1(3), 244–256.

[11] Douglas, D.; Peucker, T. (1973), "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature", The Canadian Cartographer 10(2), 112–122.

[12] Ville Pulkki, "Virtual Sound Source Positioning Using Vector Base Amplitude Panning"; J. Audio Eng. Soc., Volume 45, Issue 6, pp. 456-466, June 1997.

Bản chất kỹ thuật của sáng chế

Mục đích của sáng chế là đề xuất các khái niệm được cải thiện để lập mã siêu dữ liệu đối tượng. Mục đích của sáng chế đạt được bởi thiết bị theo điểm 1, thiết bị theo điểm 8, hệ thống theo điểm 14, phương pháp theo điểm 15, phương pháp theo điểm 16 và chương trình máy tính theo điểm 17.

Thiết bị tạo ra một hoặc nhiều kênh âm thanh được đề xuất. Thiết bị bao gồm bộ giải mã siêu dữ liệu để nhận một hoặc nhiều tín hiệu siêu dữ liệu được nén. Mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhiều mẫu siêu dữ liệu thứ nhất. Các mẫu siêu dữ liệu thứ nhất của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén biểu thị thông tin được đính kèm với tín hiệu đối tượng âm thanh trong số một hoặc nhiều tín hiệu đối tượng âm thanh. Bộ giải mã siêu dữ liệu được cấu hình để tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục, để mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm các mẫu siêu dữ liệu thứ nhất của một trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén và còn bao gồm mẫu siêu dữ liệu thứ hai. Hơn nữa, bộ giải mã siêu dữ liệu được cấu hình để tạo ra mỗi mẫu trong số các mẫu siêu dữ liệu thứ hai của mỗi tín hiệu siêu dữ liệu được khôi phục trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục phụ thuộc vào ít nhất hai trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu. Hơn nữa, thiết bị bao gồm bộ tạo kênh âm thanh để tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục.

Hơn nữa, thiết bị tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén được đề xuất. Thiết bị bao gồm bộ mã hóa siêu dữ liệu để nhận một hoặc nhiều tín hiệu siêu dữ liệu ban đầu. Mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu bao

gồm nhiều mẫu siêu dữ liệu. Các mẫu siêu dữ liệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu biểu thị thông tin được đính kèm với tín hiệu đối tượng âm thanh trong số một hoặc nhiều tín hiệu đối tượng âm thanh. Các bộ mã hóa siêu dữ liệu được tạo cấu hình để tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được nén, để mỗi tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhóm thứ nhất gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu của một trong số các tín hiệu siêu dữ liệu ban đầu, và để tín hiệu siêu dữ liệu được nén đã nêu không bao gồm bất kỳ mẫu siêu dữ liệu của nhóm thứ hai gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu khác của một trong số các tín hiệu siêu dữ liệu ban đầu đã nêu. Hơn nữa, thiết bị bao gồm bộ mã hóa âm thanh để mã hóa một hoặc nhiều tín hiệu đối tượng âm thanh để thu được một hoặc nhiều tín hiệu âm thanh được mã hóa.

Hơn nữa, hệ thống được đề xuất. Hệ thống bao gồm thiết bị tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén như được mô tả trên đây. Hơn nữa, hệ thống bao gồm thiết bị để nhận một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén, và để tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu âm thanh được mã hóa và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được nén như được mô tả trên đây.

Theo các phương án, khái niệm nén dữ liệu cho siêu dữ liệu đối tượng được đề xuất, mà đạt được cơ cấu nén hiệu quả cho các kênh truyền dẫn với tốc độ dữ liệu giới hạn. Hơn nữa, tốc độ nén tốt cho các thay đổi góc phương vị đơn thuần, ví dụ, sự quay máy ảnh, đạt được. Hơn nữa, các khái niệm được đề xuất hỗ trợ các quỹ đạo gián đoạn, ví dụ, các nhánh vị trí. Hơn nữa, độ phức tạp giải mã thấp được nhận biết. Hơn nữa, sự truy cập ngẫu nhiên với thời gian tái khởi tạo bị hạn chế đạt được.

Hơn nữa, phương pháp tạo ra một hoặc nhiều kênh âm thanh được đề xuất. Phương pháp bao gồm:

- Nhận một hoặc nhiều tín hiệu siêu dữ liệu được nén, trong đó mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhiều mẫu siêu dữ liệu thứ nhất, trong đó các mẫu siêu dữ liệu thứ nhất của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén biểu thị thông tin được kết hợp với tín hiệu đối tượng

âm thanh của một hoặc nhiều tín hiệu đối tượng âm thanh.

- Tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục, để mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm các mẫu siêu dữ liệu thứ nhất của một trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén và còn bao gồm nhiều mẫu siêu dữ liệu thứ hai, trong đó tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục, bao gồm bước tạo ra mỗi mẫu trong số các mẫu siêu dữ liệu thứ hai của mỗi tín hiệu siêu dữ liệu được khôi phục của một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục phụ thuộc vào ít nhất hai trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu. Và:

- Tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục.

Hơn nữa, phương pháp tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén được đề xuất. Phương pháp bao gồm:

- Nhận một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, trong đó mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu bao gồm nhiều mẫu siêu dữ liệu, trong đó các mẫu siêu dữ liệu của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu biểu thị thông tin được kết hợp với tín hiệu đối tượng âm thanh của một hoặc nhiều tín hiệu đối tượng âm thanh.

- Tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được nén, để mỗi tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhóm thứ nhất gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu của một trong số các tín hiệu siêu dữ liệu ban đầu, và để tín hiệu siêu dữ liệu được nén đã nêu không bao gồm bất kỳ mẫu siêu dữ liệu của nhóm thứ hai gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu khác của một trong số các tín hiệu siêu dữ liệu ban đầu đã nêu. Và:

- Mã hóa một hoặc nhiều tín hiệu đối tượng âm thanh để thu được một hoặc nhiều tín hiệu âm thanh được mã hóa.

Hơn nữa, chương trình máy tính để thực hiện phương pháp được mô tả bên trên

khi được chạy trên máy tính hoặc bộ xử lý tín hiệu được đề xuất.

Mô tả vắn tắt các hình vẽ

Dưới đây, các phương án của sáng chế được mô tả chi tiết hơn với sự viện dẫn đến các hình vẽ, trong đó:

Fig.1 minh họa thiết bị tạo ra một hoặc nhiều kênh âm thanh theo phương án,

Fig.2 minh họa thiết bị tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén theo phương án,

Fig.3 minh họa hệ thống theo như phương án khác,

Fig.4 minh họa vị trí của đối tượng âm thanh trong không gian ba chiều từ góc được thể hiện bởi góc phương vị, góc nâng và bán kính,

Fig.5 minh họa các vị trí của các đối tượng âm thanh và thiết lập loa phát thanh được giả định bởi bộ tạo kênh âm thanh,

Fig.6 minh họa sự mã hóa siêu dữ liệu theo phương án,

Fig.7 minh họa sự giải mã siêu dữ liệu theo phương án,

Fig.8 minh họa sự mã hóa siêu dữ liệu theo phương án khác,

Fig.9 minh họa sự giải mã siêu dữ liệu theo phương án khác,

Fig.10 minh họa sự mã hóa siêu dữ liệu theo phương án khác,

Fig.11 minh họa sự mã hóa siêu dữ liệu theo phương án khác,

Fig.12 minh họa phương án thứ nhất của bộ mã hóa âm thanh,

Fig.13 minh họa phương án thứ nhất của bộ giải mã âm thanh,

Fig.14 minh họa phương án thứ hai của bộ mã hóa âm thanh,

Fig.15 minh họa phương án thứ hai của bộ giải mã âm thanh,

Fig.16 minh họa phương án thứ ba của bộ mã hóa âm thanh,

Fig.17 minh họa phương án thứ ba của bộ giải mã âm thanh,

Mô tả chi tiết sáng chế

Fig.2 minh họa thiết bị 250 tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén theo phương án.

Thiết bị 250 bao gồm bộ mã hóa siêu dữ liệu 210 để nhận một hoặc nhiều tín hiệu siêu dữ liệu ban đầu. Mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu bao gồm nhiều mẫu siêu dữ liệu. Các mẫu siêu dữ liệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu biểu thị thông tin được đính kèm với tín hiệu đối tượng âm thanh trong số một hoặc nhiều tín hiệu đối tượng âm thanh. Bộ mã hóa siêu dữ liệu 210 được tạo cấu hình để tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được nén, để mỗi tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhóm thứ nhất gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu của một trong số các tín hiệu siêu dữ liệu ban đầu, và để tín hiệu siêu dữ liệu được nén đã nêu không bao gồm bất kì mẫu siêu dữ liệu của nhóm thứ hai gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu khác của một trong số các tín hiệu siêu dữ liệu ban đầu đã nêu.

Hơn nữa, thiết bị 250 bao gồm bộ mã hóa âm thanh 220 để mã hóa một hoặc nhiều tín hiệu đối tượng âm thanh để thu được một hoặc nhiều tín hiệu âm thanh được mã hóa. Ví dụ, bộ tạo kênh âm thanh có thể bao gồm bộ mã hóa SAOC theo tình trạng kỹ thuật để mã hóa một hoặc nhiều tín hiệu đối tượng âm thanh để thu được một hoặc nhiều kênh vận chuyển SAOC như một hoặc nhiều tín hiệu âm thanh được mã hóa. Nhiều công nghệ mã hóa khác để mã hóa một hoặc nhiều kênh đối tượng âm thanh có thể ngoài ra hoặc cách khác được sử dụng để mã hóa một hoặc nhiều kênh đối tượng âm thanh.

Fig.1 minh họa thiết bị 100 tạo ra một hoặc nhiều kênh âm thanh theo phương án.

Thiết bị 100 bao gồm bộ giải mã siêu dữ liệu 110 để nhận một hoặc nhiều tín hiệu siêu dữ liệu được nén. Mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhiều mẫu siêu dữ liệu thứ nhất. Các mẫu siêu dữ liệu thứ nhất của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén biểu thị thông tin được đính kèm với tín hiệu đối tượng âm thanh trong số một hoặc nhiều tín hiệu đối tượng âm thanh. Bộ giải mã siêu dữ liệu 110 được tạo cấu hình để tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục, để mỗi tín hiệu trong số một hoặc nhiều tín

hiệu siêu dữ liệu được khôi phục bao gồm các mẫu siêu dữ liệu thứ nhất của một trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén và còn bao gồm mẫu siêu dữ liệu thứ hai. Hơn nữa, bộ giải mã siêu dữ liệu 110 được tạo cấu hình để tạo ra mỗi mẫu trong số các mẫu siêu dữ liệu thứ hai của mỗi tín hiệu siêu dữ liệu được khôi phục trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục phụ thuộc vào ít nhất hai trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu.

Hơn nữa, thiết bị 100 bao gồm bộ tạo kênh âm thanh 120 để tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục.

Khi đề cập đến các mẫu siêu dữ liệu, nên lưu ý rằng, mẫu siêu dữ liệu được đặc trưng bởi giá trị mẫu siêu dữ liệu của nó, nhưng cũng bởi thời điểm, mà nó liên quan đến. Ví dụ, thời điểm như vậy có thể liên quan đến sự bắt đầu của chuỗi âm thanh hoặc tương tự. Ví dụ, chỉ số n hoặc k có thể xác định vị trí của mẫu siêu dữ liệu trong tín hiệu siêu dữ liệu và bằng cách này, thời điểm (liên quan) (liên quan đến thời gian bắt đầu) được biểu thị. Nên lưu ý rằng khi hai mẫu siêu dữ liệu liên quan đến các thời điểm khác nhau, hai mẫu siêu dữ liệu này là các mẫu siêu dữ liệu khác nhau, thậm chí khi các giá trị mẫu siêu dữ liệu bằng nhau, mà thường có thể là một trường hợp.

Các phương án trên được dựa trên việc tìm ra rằng thông tin siêu dữ liệu (được bao gồm bởi tín hiệu siêu dữ liệu) mà được kết hợp với tín hiệu đối tượng âm thanh thường thay đổi dần dần.

Ví dụ, tín hiệu siêu dữ liệu có thể biểu thị thông tin vị trí về đối tượng âm thanh (ví dụ, góc phương vị, góc nâng hoặc bán kính định rõ vị trí của đối tượng âm thanh). Có thể giả định rằng, hầu hết thời gian, vị trí của đối tượng âm thanh hoặc không thay đổi hoặc chỉ thay đổi dần dần.

Hoặc, tín hiệu siêu dữ liệu có thể, ví dụ, biểu thị âm lượng (ví dụ, độ khuếch đại) của đối tượng âm thanh, và có thể cũng được giả định, rằng hầu hết thời gian, âm lượng của đối tượng âm thanh thay đổi dần dần.

Vì lí do này, không cần thiết truyền dẫn thông tin siêu dữ liệu (đầy đủ) tại mọi thời điểm. Thay vào đó, thông tin siêu dữ liệu (đầy đủ) chỉ được truyền dẫn tại các thời

điểm nhất định, ví dụ, theo chu kỳ, ví dụ, tại mỗi thời điểm thứ N , ví dụ, tại thời điểm $0, N, 2N, 3N, \dots$. Tại phía bộ giải mã, đối với mỗi thời điểm trung gian (ví dụ, các thời điểm $1, 2, \dots, N-1$) siêu dữ liệu có thể sau đó được làm gần đúng dựa trên các mẫu siêu dữ liệu cho hai hoặc nhiều hơn hai thời điểm. Ví dụ, các mẫu siêu dữ liệu cho các thời điểm $1, 2, \dots, N-1$ có thể được làm gần đúng tại các phía bộ giải mã phụ thuộc vào các mẫu siêu dữ liệu cho các thời điểm 0 và N , ví dụ, bằng cách sử dụng phép nội suy tuyến tính. Như đã nêu trước đây, phương pháp như vậy được dựa trên việc tìm ra rằng thông tin siêu dữ liệu về các đối tượng âm thanh nhìn chung thay đổi từ từ.

Ví dụ, trong các phương án, ba tín hiệu siêu dữ liệu chỉ rõ vị trí của đối tượng âm thanh trong không gian 3D. Tín hiệu thứ nhất trong số các tín hiệu siêu dữ liệu có thể, ví dụ, chỉ rõ góc phương vị của vị trí của đối tượng âm thanh. Tín hiệu thứ hai trong số các tín hiệu siêu dữ liệu có thể, ví dụ, chỉ rõ góc nâng của vị trí của đối tượng âm thanh. Tín hiệu thứ ba trong số các tín hiệu siêu dữ liệu có thể, ví dụ, chỉ rõ bán kính liên quan đến khoảng cách của đối tượng âm thanh.

Góc phương vị, góc nâng và bán kính chỉ rõ ràng vị trí của đối tượng âm thanh trong không gian 3D từ gốc. Điều này được mô tả với sự viện dẫn đến Fig.4.

Fig.4 minh họa vị trí 410 của đối tượng âm thanh trong không gian ba chiều (three-dimensional - 3D) từ gốc 400 được thể hiện bởi góc phương vị, góc nâng và bán kính.

Góc nâng chỉ rõ, ví dụ, góc giữa đường thẳng từ gốc đến vị trí đối tượng và phép chiếu bình thường của đường thẳng này lên mặt phẳng xy (mặt phẳng được xác định bởi trục x và trục y). Góc phương vị định rõ, ví dụ, góc giữa trục x và phép chiếu bình thường đã nêu. Bằng cách chỉ rõ góc phương vị và góc nâng đường thẳng 415 qua gốc 400 và vị trí 410 của đối tượng âm thanh có thể được định rõ. Hơn nữa, bằng cách chỉ rõ bán kính, vị trí chính xác 410 của đối tượng âm thanh có thể được định rõ.

Trong phương án, góc phương vị được định rõ trong phạm vi: $-180^\circ < \text{góc phương vị} \leq 180^\circ$, góc nâng được định rõ trong phạm vi: $-90^\circ \leq \text{góc nâng} \leq 90^\circ$ và bán kính có thể, ví dụ, được định rõ theo mét [m] (lớn hơn hoặc bằng 0m).

Trong phương án khác, trong đó nó, có thể ví dụ, được giả định rằng tất cả các

giá trị x của các vị trí đối tượng âm thanh trong hệ tọa độ xyz là lớn hơn hoặc bằng 0, góc phương vị có thể được định rõ trong phạm vi: $-90^\circ \leq \text{góc phương vị} \leq 90^\circ$, góc nâng có thể được định rõ trong phạm vi: $-90^\circ \leq \text{góc nâng} \leq 90^\circ$ và bán kính có thể, ví dụ, được định rõ theo mét [m].

Trong phương án khác, các tín hiệu siêu dữ liệu có thể được định tỉ lệ sao cho góc phương vị được định rõ trong tỉ lệ: $-128^\circ < \text{góc phương vị} \leq 128^\circ$, góc nâng được định rõ trong phạm vi: $-32^\circ \leq \text{góc nâng} \leq 32^\circ$ và bán kính có thể, ví dụ, được định rõ trên sự định tỉ lệ theo logarit. Trong một số phương án, các tín hiệu siêu dữ liệu ban đầu, các tín hiệu siêu dữ liệu được nén và các tín hiệu siêu dữ liệu được khôi phục lần lượt có thể bao gồm sự biểu diễn được định tỉ lệ của thông tin vị trí và/hoặc sự biểu diễn được định tỉ lệ của âm lượng của một trong số một hoặc nhiều tín hiệu đối tượng âm thanh.

Bộ tạo kênh âm thanh 120 có thể, ví dụ, được tạo cấu hình để tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào các tín hiệu siêu dữ liệu được khôi phục, trong đó các tín hiệu siêu dữ liệu được khôi phục có thể, ví dụ, biểu thị vị trí của các đối tượng âm thanh.

Fig.5 minh họa các vị trí của các đối tượng âm thanh và thiết lập loa phát thanh được giả định bởi bộ tạo kênh âm thanh. Góc 500 của hệ tọa độ xyz được minh họa. Hơn nữa, vị trí 510 của đối tượng âm thanh thứ nhất và vị trí 520 của đối tượng âm thanh thứ hai được minh họa. Hơn nữa, Fig.5 minh họa cảnh, trong đó bộ tạo kênh âm thanh 120 tạo ra bốn kênh âm thanh cho bốn loa phát thanh. Bộ tạo kênh âm thanh 120 giả định rằng bốn loa phát thanh 511, 512, 513 và 514 được định vị tại các vị trí trên Fig.5.

Trên Fig.5, đối tượng âm thanh thứ nhất được định vị tại vị trí 510 gần với các vị trí được giả định của các loa phát thanh 511 và 512, và được định vị xa khỏi các loa phát thanh 513 và 514. Do đó, bộ tạo kênh âm thanh 120 có thể tạo ra bốn kênh âm thanh sao cho đối tượng âm thanh thứ nhất 510 được tái tạo bởi các loa phát thanh 511 và 512 nhưng không bởi các loa phát thanh 513 và 514.

Trong các phương án khác, bộ tạo kênh âm thanh 120 có thể tạo ra bốn kênh âm thanh sao cho đối tượng âm thanh thứ nhất 510 được tái tạo với âm lượng cao bởi các

loa phát thanh 511 và 512 và với âm lượng thấp bởi các loa phát thanh 513 và 514.

Hơn nữa, đối tượng âm thanh thứ hai được định vị tại vị trí 520 gần với các vị trí được giả định của các loa phát thanh 513 và 514, và được định vị xa khỏi các loa phát thanh 511 và 512. Do đó, bộ tạo kênh âm thanh 120 có thể tạo ra bốn kênh âm thanh sao cho đối tượng âm thanh thứ hai 520 được tái tạo bởi các loa phát thanh 513 và 514 nhưng không bởi các loa phát thanh 511 và 512.

Trong các phương án khác, bộ tạo kênh âm thanh 120 có thể tạo ra bốn kênh âm thanh sao cho đối tượng âm thanh thứ hai 520 được tái tạo với âm lượng cao bởi các loa phát thanh 513 và 514 và với âm lượng thấp bởi các loa phát thanh 511 và 512.

Trong các phương án thay thế, chỉ hai tín hiệu siêu dữ liệu được sử dụng để chỉ rõ vị trí của đối tượng âm thanh. Ví dụ, chỉ góc phương vị và bán kính có thể được chỉ rõ, ví dụ, khi được giả thiết rằng tất cả các đối tượng âm thanh được định vị trong mặt phẳng đơn.

Trong các phương án khác, đối với mỗi đối tượng âm thanh, chỉ tín hiệu siêu dữ liệu đơn được mã hóa và được truyền dẫn như thông tin vị trí. Ví dụ, chỉ góc phương vị có thể được chỉ rõ như thông tin vị trí cho đối tượng âm thanh (ví dụ, nó có thể được giả định rằng tất cả các đối tượng âm thanh được định vị trong cùng mặt phẳng có cùng khoảng cách từ tâm điểm, và do đó được giả định có cùng bán kính). Thông tin phương vị có thể, ví dụ, đủ để xác định rằng đối tượng âm thanh được định vị gần với loa phát thanh bên trái và xa khỏi loa phát thanh bên phải. Trong trường hợp như vậy, bộ tạo kênh âm thanh 120 có thể, ví dụ, tạo ra một hoặc nhiều kênh âm thanh sao cho đối tượng âm thanh được tái tạo bởi loa phát thanh bên trái, nhưng không bởi loa phát thanh bên phải.

Ví dụ, sự ghép nhấn biên độ trên cơ sở vectơ (Vector Base Amplitude Panning - VBAP) có thể được sử dụng (xem, ví dụ, [12]) để xác định trọng số của tín hiệu đối tượng âm thanh nằm trong mỗi kênh âm thanh của các loa phát thanh. Ví dụ, đối với VBAP, giả định rằng đối tượng âm thanh liên quan đến nguồn ảo.

Trong các phương án, tín hiệu siêu dữ liệu khác có thể chỉ rõ âm lượng, ví dụ, độ khuếch đại (ví dụ, được biểu diễn theo deciben [dB]) cho mỗi đối tượng âm thanh.

Ví dụ, trên Fig.5, giá trị độ khuếch đại thứ nhất có thể được chỉ rõ bởi tín hiệu siêu dữ liệu cho đối tượng âm thanh thứ nhất được định vị tại vị trí 510 mà cao hơn giá trị độ khuếch đại thứ hai được chỉ rõ bởi tín hiệu siêu dữ liệu khác cho đối tượng âm thanh thứ hai được định vị tại vị trí 520. Trong tình huống như vậy, các loa phát thanh 511 và 512 có thể tái tạo đối tượng âm thanh thứ nhất với âm lượng cao hơn âm lượng mà các loa phát thanh 513 và 514 tái tạo đối tượng âm thanh thứ hai.

Các phương án cũng giả định rằng các giá trị độ khuếch đại như vậy của các đối tượng âm thanh thường thay đổi dần dần. Do đó, không cần thiết truyền dẫn thông tin siêu dữ liệu tại mọi thời điểm. Thay vào đó, thông tin siêu dữ liệu chỉ được truyền dẫn tại các thời điểm thích hợp. Tại các thời điểm trung gian, thông tin siêu dữ liệu có thể, ví dụ, được làm gần đúng sử dụng mẫu siêu dữ liệu đứng trước và mẫu siêu dữ liệu tiếp theo, mà đã được truyền dẫn. Ví dụ, phép nội suy tuyến tính có thể được sử dụng cho sự làm gần đúng các giá trị trung gian. Ví dụ, độ khuếch đại, góc phương vị, góc nâng và/hoặc bán kính của mỗi đối tượng trong số các đối tượng âm thanh có thể được làm gần đúng đối với các thời điểm, trong đó siêu dữ liệu như vậy không được truyền dẫn.

Bằng phương pháp như vậy, sự tiết kiệm đáng kể trong tốc độ truyền dẫn của siêu dữ liệu có thể đạt được.

Fig.3 minh họa hệ thống theo như phương án khác.

Hệ thống bao gồm thiết bị 250 tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén như được mô tả trên đây.

Hơn nữa, hệ thống bao gồm thiết bị 100 để nhận một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén, và để tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu âm thanh được mã hóa và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được nén như được mô tả trên đây.

Ví dụ, một hoặc nhiều tín hiệu âm thanh được mã hóa có thể được giải mã bằng thiết bị 100 để tạo ra một hoặc nhiều kênh âm thanh bằng cách sử dụng bộ giải mã

SAOC theo tình trạng kỹ thuật để thu được một hoặc nhiều tín hiệu đối tượng âm thanh, trong đó thiết bị 250 để mã hóa đã sử dụng bộ mã hóa SAOC để mã hóa một hoặc nhiều đối tượng âm thanh.

Xem xét các vị trí đối tượng chỉ như ví dụ cho siêu dữ liệu, để cho phép sự truy cập ngẫu nhiên với thời gian được tái khởi tạo được giới hạn, các phương án cung cấp sự tái truyền dẫn đầy đủ của tất cả các vị trí đối tượng trên cơ sở bình thường.

Theo phương án, thiết bị 100 được tạo cấu hình để nhận thông tin truy cập ngẫu nhiên, trong đó, đối với mỗi tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén, thông tin truy cập ngẫu nhiên biểu thị phần tín hiệu được truy cập của tín hiệu siêu dữ liệu được nén đã nêu, trong đó ít nhất một phần tín hiệu khác của tín hiệu siêu dữ liệu đã nêu không được biểu thị bởi thông tin truy cập ngẫu nhiên, và trong đó bộ giải mã siêu dữ liệu 110 được tạo cấu hình để tạo ra một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục phụ thuộc vào các mẫu siêu dữ liệu thứ nhất của phần tín hiệu được truy cập đã nêu của tín hiệu siêu dữ liệu được nén đã nêu, nhưng không phụ thuộc vào bất kì mẫu siêu dữ liệu thứ nhất nào khác của bất kì phần tín hiệu khác của tín hiệu siêu dữ liệu được nén đã nêu. Nói cách khác, bằng cách chỉ rõ thông tin truy cập ngẫu nhiên, phần của mỗi tín hiệu trong số các tín hiệu siêu dữ liệu được nén có thể được chỉ rõ, trong đó các phần khác của tín hiệu siêu dữ liệu đã nêu không được chỉ rõ. Trong trường hợp này, chỉ phần được chỉ rõ của tín hiệu siêu dữ liệu được nén được khôi phục như một trong số các tín hiệu siêu dữ liệu được khôi phục, nhưng không phải các phần khác. Việc khôi phục là khả thi, như các mẫu siêu dữ liệu thứ nhất được truyền dẫn của tín hiệu siêu dữ liệu được nén đã nêu biểu diễn thông tin siêu dữ liệu đầy đủ của tín hiệu siêu dữ liệu đã nêu cho các thời điểm nhất định (cho các thời điểm khác, tuy nhiên, thông tin siêu dữ liệu không được truyền dẫn).

Fig.6 minh họa sự mã hóa siêu dữ liệu theo phương án. Bộ mã hóa siêu dữ liệu 210 theo các phương án có thể được tạo cấu hình để thực hiện sự mã hóa siêu dữ liệu được minh họa bởi Fig.6.

Trong Fig.6, $s(n)$ có thể biểu diễn một trong số các tín hiệu siêu dữ liệu ban đầu. Ví dụ, $s(n)$ có thể, ví dụ, biểu diễn hàm của góc phương vị của một trong số các đối

tượng âm thanh, và n có thể biểu thị thời gian (ví dụ, bằng cách biểu thị các vị trí mẫu trong tín hiệu siêu dữ liệu ban đầu).

Thành phần quỹ đạo biến thiên thời gian $s(n)$, mà được tạo mẫu tại tốc độ lấy mẫu mà thấp hơn đáng kể (ví dụ, 1:1024 hoặc thấp hơn) so với tốc độ lấy mẫu âm thanh, được lượng tử hóa (xem 611) và được lấy mẫu giảm (xem 612) bởi hệ số N . Điều này dẫn đến tín hiệu kỹ thuật số được truyền dẫn thông thường được nêu trước đây mà chúng ta biểu thị là $z(k)$.

$z(k)$ là một trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén. Ví dụ, mỗi mẫu siêu dữ liệu thứ N của $s(n)$ cũng là mẫu siêu dữ liệu của tín hiệu siêu dữ liệu được nén $z(k)$, trong khi $N-1$ mẫu siêu dữ liệu khác của $s(n)$ giữa mỗi mẫu siêu dữ liệu thứ N không phải các mẫu siêu dữ liệu của tín hiệu siêu dữ liệu được nén $z(k)$.

Ví dụ, giả định rằng trong $s(n)$, n biểu thị thời gian (ví dụ, bằng cách biểu thị các vị trí mẫu trong tín hiệu siêu dữ liệu ban đầu), trong đó n là số nguyên dương hoặc 0. (ví dụ, thời gian bắt đầu: $n = 0$). N là hệ số lấy mẫu giảm. Ví dụ, $N = 32$ hoặc hệ số lấy mẫu giảm thích hợp khác.

Ví dụ, lấy mẫu giảm trong 612 để thu tín hiệu siêu dữ liệu được nén z từ tín hiệu siêu dữ liệu ban đầu s có thể, ví dụ, được nhận ra, sao cho:

$$z(k) = s(k \cdot N); \quad \text{trong đó } k \text{ là số nguyên dương hoặc } 0 \text{ (} k = 0, 1, 2, \dots \text{)}$$

Do đó:

$$z(0) = s(0); \quad z(1) = s(32); \quad z(2) = s(64); \quad z(3) = s(96), \quad \dots$$

Fig.7 minh họa sự giải mã siêu dữ liệu theo phương án. Bộ giải mã siêu dữ liệu 110 theo các phương án có thể được tạo cấu hình để thực hiện sự giải mã siêu dữ liệu được minh họa bởi Fig.7.

Theo phương án được minh họa bởi Fig.7, bộ giải mã siêu dữ liệu 110 được tạo cấu hình để tạo ra mỗi tín hiệu siêu dữ liệu được khôi phục trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bằng cách lấy mẫu tăng một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén, trong đó bộ giải mã siêu dữ liệu 110 được tạo cấu hình để tạo ra mỗi mẫu trong số các mẫu siêu dữ liệu thứ hai trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bằng cách tiến hành phép nội suy tuyến tính

phụ thuộc vào ít nhất hai mẫu trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu.

Do đó, mỗi tín hiệu siêu dữ liệu được khôi phục bao gồm tất cả các mẫu siêu dữ liệu của tín hiệu siêu dữ liệu được nén (các mẫu này được đề cập đến như "các mẫu siêu dữ liệu thứ nhất" của một hoặc nhiều tín hiệu siêu dữ liệu được nén).

Bằng cách tiến hành việc lấy mẫu tăng, các mẫu siêu dữ liệu ("thứ hai") bổ sung được thêm vào tín hiệu siêu dữ liệu được khôi phục. Bước lấy mẫu tăng xác định, tại các vị trí nào trong tín hiệu siêu dữ liệu được khôi phục (ví dụ, tại các thời điểm "liên quan") các mẫu siêu dữ liệu (thứ hai) bổ sung được thêm vào tín hiệu siêu dữ liệu.

Bằng cách tiến hành phép nội suy tuyến tính, các giá trị mẫu siêu dữ liệu của các mẫu siêu dữ liệu thứ hai được xác định. Phép nội suy tuyến tính được tiến hành dựa trên hai mẫu siêu dữ liệu của tín hiệu siêu dữ liệu được nén (mà đã trở thành các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục).

Theo các phương án, sự lấy mẫu giảm vào tạo ra các mẫu siêu dữ liệu thứ hai bằng cách tiến hành phép nội suy tuyến tính có thể, ví dụ, được tiến hành trong bước đơn.

Trên Fig.7, quy trình lấy mẫu tăng nghịch đảo (xem 721) tổ hợp với phép nội suy tuyến tính (xem 722) dẫn đến kết quả là sự làm gần đúng thô của tín hiệu ban đầu. Quy trình lấy mẫu tăng nghịch đảo (xem 721) và phép nội suy tuyến tính (xem 722), có thể, ví dụ, được tiến hành trong bước đơn.

Ví dụ, sự lấy mẫu tăng (721) và phép nội suy tuyến tính (722) trên phía bộ giải mã có thể, ví dụ, được tiến hành, sao cho:

$$s'(k \cdot N) = z(k); \quad \text{trong đó } k \text{ là số nguyên dương hoặc } 0$$

$$s'(k \cdot N + j) = z(k-1) + \frac{j}{N} [z(k) - z(k-1)]; \quad \text{trong đó } j \text{ là số nguyên với } 1 \leq j \leq N - 1$$

Tại đây, $z(k)$ là mẫu siêu dữ liệu được nhận thực sự của tín hiệu siêu dữ liệu được nén z , và $z(k-1)$ là mẫu siêu dữ liệu của tín hiệu siêu dữ liệu được nén z , mà được nhận ngay trước mẫu siêu dữ liệu được nhận thực sự $z(k)$.

Fig.8 minh họa sự mã hóa siêu dữ liệu theo phương án khác. Bộ mã hóa siêu dữ liệu 210 theo các phương án có thể được tạo cấu hình để thực hiện sự mã hóa siêu dữ liệu được minh họa bởi Fig.8.

Trong các phương án, ví dụ, như được minh họa bởi Fig.8, trong sự mã hóa siêu dữ liệu, cấu trúc mịn có thể được chỉ rõ bởi sự chênh lệch được mã hóa giữa tín hiệu đầu vào được bù độ trễ và sự làm gần đúng thô được nội suy tuyến tính.

Theo các phương án như vậy, quy trình lấy mẫu tăng nghịch đảo tổ hợp với phép nội suy tuyến tính cũng được tiến hành như phần của sự mã hóa siêu dữ liệu trên phía bộ mã hóa (xem 621 và 622 trên Fig.6). Lần nữa, quy trình lấy mẫu tăng nghịch đảo (xem 621) và phép nội suy tuyến tính (xem 622), có thể, ví dụ, được tiến hành trong bước đơn.

Như được mô tả trên đây, bộ mã hóa siêu dữ liệu 210 được tạo cấu hình để tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được nén, để mỗi tín hiệu siêu dữ liệu được nén của một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhóm thứ nhất gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu của tín hiệu siêu dữ liệu ban đầu của một hoặc nhiều tín hiệu siêu dữ liệu ban đầu. Tín hiệu siêu dữ liệu được nén đã nêu có thể được xem như được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu.

Mỗi mẫu trong số các mẫu siêu dữ liệu mà được bao gồm bởi tín hiệu siêu dữ liệu ban đầu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu và nó cũng được bao gồm bởi tín hiệu siêu dữ liệu được nén, mà được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu, có thể được xem xét như một mẫu trong số nhiều mẫu siêu dữ liệu thứ nhất.

Hơn nữa, mỗi mẫu trong số các mẫu siêu dữ liệu mà được bao gồm bởi tín hiệu siêu dữ liệu ban đầu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu và nó không được bao gồm bởi tín hiệu siêu dữ liệu được nén, mà được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu, có thể được xem xét như một mẫu trong số nhiều mẫu siêu dữ liệu thứ hai.

Theo phương án của Fig.8, bộ mã hóa siêu dữ liệu 210 được tạo cấu hình để tạo ra mẫu siêu dữ liệu được làm gần đúng cho mỗi mẫu trong số nhiều mẫu siêu dữ liệu

thứ hai của một trong số các tín hiệu siêu dữ liệu ban đầu bằng cách tiến hành phép nội suy tuyến tính phụ thuộc vào ít nhất hai mẫu trong số các mẫu siêu dữ liệu thứ nhất của một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu.

Hơn nữa, trong phương án của Fig.8, bộ mã hóa siêu dữ liệu 210 được tạo cấu hình để tạo ra giá trị chênh lệch cho mỗi mẫu siêu dữ liệu thứ hai trong số nhiều mẫu siêu dữ liệu thứ hai đã nêu của một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, để giá trị chênh lệch đã nêu biểu thị sự chênh lệch giữa mẫu siêu dữ liệu thứ hai đã nêu và mẫu siêu dữ liệu được làm gần đúng của mẫu siêu dữ liệu thứ hai đã nêu.

Trong phương án được ưu tiên, mà được mô tả sau với sự viện dẫn đến Fig.10, bộ mã hóa siêu dữ liệu 210 có thể, ví dụ, được tạo cấu hình để xác định ít nhất một giá trị trong số các giá trị chênh lệch của nhiều mẫu siêu dữ liệu thứ hai đã nêu của một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, liệu mỗi giá trị trong số ít nhất một trong số giá trị chênh lệch đã nêu lớn hơn giá trị ngưỡng.

Trong các phương án theo Fig.8, các mẫu siêu dữ liệu được làm tròn có thể, ví dụ, được xác định (ví dụ, như các mẫu $s''(n)$ của tín hiệu s'') bằng cách tiến hành sự lấy mẫu tăng trên tín hiệu siêu dữ liệu được nén $z(k)$ và bằng cách tiến hành phép nội suy tuyến tính. Sự lấy mẫu tăng và phép nội suy tuyến tính có thể, ví dụ, được tiến hành như phần của sự mã hóa siêu dữ liệu trên phía bộ mã hóa (xem 621 và 622 trên Fig.6), ví dụ, theo cùng một cách, như được mô tả cho sự giải mã siêu dữ liệu viện dẫn đến 721 và 722:

$$s''(k \cdot N) = z(k); \quad \text{trong đó } k \text{ là số nguyên dương hoặc } 0$$

$$s''(k \cdot N + j) = z(k-1) + \frac{j}{N} [z(k) - z(k-1)]; \quad \text{trong đó } j \text{ là số nguyên với } 1 \leq j \leq N-1$$

Ví dụ, trong phương án được minh họa bởi Fig.8, khi tiến hành sự mã hóa siêu dữ liệu, các giá trị chênh lệch có thể được xác định trong 630 cho sự chênh lệch

$$s(n) - s''(n), \quad \text{ví dụ, cho tất cả } n \text{ với } (k-1) \cdot N < n < k \cdot N, \text{ hoặc}$$

$$\text{ví dụ, đối với tất cả } n \text{ với } (k-1) \cdot N < n \leq k \cdot N$$

Trong các phương án, một hoặc nhiều của các giá trị chênh lệch này được truyền dẫn đến bộ giải mã siêu dữ liệu.

Fig.9 minh họa sự giải mã siêu dữ liệu theo phương án khác. Bộ giải mã siêu dữ liệu 110 theo các phương án có thể được tạo cấu hình để thực hiện sự giải mã siêu dữ liệu được minh họa bởi Fig.9.

Như được mô tả trên đây, mỗi tín hiệu siêu dữ liệu được khôi phục trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được nén của một hoặc nhiều tín hiệu siêu dữ liệu được nén. Tín hiệu siêu dữ liệu được khôi phục đã nêu được xem xét để được kết hợp với tín hiệu siêu dữ liệu được nén.

Trong các phương án được minh họa bởi Fig.9, bộ giải mã siêu dữ liệu 110 được tạo cấu hình để tạo ra các mẫu siêu dữ liệu thứ hai của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bằng cách tạo ra nhiều mẫu siêu dữ liệu được làm gần đúng cho tín hiệu siêu dữ liệu được khôi phục đã nêu, trong đó, bộ giải mã siêu dữ liệu 110 được tạo cấu hình để tạo ra mỗi mẫu trong số nhiều mẫu siêu dữ liệu được làm gần đúng phụ thuộc vào ít nhất hai mẫu trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu. Ví dụ, các mẫu siêu dữ liệu được làm gần đúng này có thể được tạo ra bởi phép nội suy tuyến tính như được mô tả viện dẫn đến Fig.7.

Theo phương án được minh họa bởi Fig.9, bộ giải mã siêu dữ liệu 110 được tạo cấu hình để nhận nhiều giá trị chênh lệch cho tín hiệu siêu dữ liệu được nén của một hoặc nhiều tín hiệu siêu dữ liệu được nén. Bộ giải mã siêu dữ liệu 110 còn được tạo cấu hình để thêm mỗi giá trị trong số nhiều giá trị chênh lệch vào một mẫu trong số các mẫu siêu dữ liệu được làm gần đúng của tín hiệu siêu dữ liệu được khôi phục được kết hợp với tín hiệu siêu dữ liệu được nén đã nêu để thu được các mẫu siêu dữ liệu thứ hai của tín hiệu siêu dữ liệu được khôi phục đã nêu.

Đối với các mẫu siêu dữ liệu được làm gần đúng, mà giá trị chênh lệch đã được nhận, rằng giá trị chênh lệch được thêm vào mẫu siêu dữ liệu được làm gần đúng để thu được các mẫu siêu dữ liệu thứ hai.

Theo phương án, mẫu siêu dữ liệu được làm gần đúng, mà không có giá trị chênh lệch nào được nhận, được sử dụng như mẫu siêu dữ liệu thứ hai của tín hiệu siêu dữ liệu được khôi phục.

Theo phương án khác, tuy nhiên, nếu không có giá trị chênh lệch được nhận cho mẫu sự chênh lệch được làm gần đúng, giá trị chênh lệch được làm gần đúng được tạo ra cho mẫu siêu dữ liệu được làm gần đúng đã nêu phụ thuộc vào một hoặc nhiều giá trị chênh lệch được nhận, và mẫu siêu dữ liệu được làm gần đúng đã nêu được thêm vào mẫu siêu dữ liệu được làm gần đúng đã nêu, xem dưới đây.

Theo phương án được minh họa bởi Fig.9, các giá trị chênh lệch được nhận được thêm (xem 730) vào các mẫu siêu dữ liệu tương ứng của tín hiệu siêu dữ liệu được lấy mẫu tăng. Bằng cách này, các mẫu siêu dữ liệu được nội suy tương ứng, mà các giá trị chênh lệch đã được truyền dẫn, có thể được hiệu chỉnh, nếu cần, để thu được các mẫu siêu dữ liệu chính xác.

Quay trở lại sự mã hóa siêu dữ liệu trên Fig.8, trong các phương án được ưu tiên, ít số bit hơn được sử dụng cho việc mã hóa các mẫu siêu dữ liệu. Các phương án này được dựa trên việc tìm ra rằng (ví dụ, N) mẫu siêu dữ liệu tiếp theo hầu hết thời gian chỉ thay đổi một chút. Ví dụ, nếu một loại mẫu siêu dữ liệu được mã hóa, ví dụ, bởi 8 bit, các mẫu siêu dữ liệu này có thể nhận một trong số 256 giá trị khác nhau. Nhìn chung, bởi vì sự thay đổi một chút của (ví dụ, N) giá trị siêu dữ liệu tiếp theo, có thể được xem xét đầy đủ, để mã hóa các giá trị chênh lệch chỉ, ví dụ, bởi 5 bit. Do đó, thậm chí, các giá trị chênh lệch được truyền dẫn, số lượng các bit được truyền dẫn có thể được giảm.

Trong phương án được ưu tiên, một hoặc nhiều giá trị chênh lệch được truyền dẫn, mỗi giá trị trong số một hoặc nhiều giá trị chênh lệch với ít bit hơn mỗi mẫu trong số các mẫu siêu dữ liệu, và mỗi giá trị chênh lệch là giá trị số nguyên.

Theo phương án, bộ mã hóa siêu dữ liệu 110 được tạo cấu hình để mã hóa một hoặc nhiều của các mẫu siêu dữ liệu của một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén với số lượng các bit thứ nhất, trong đó mỗi mẫu đã nêu trong số một hoặc nhiều mẫu siêu dữ liệu của một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén biểu thị số nguyên. Hơn nữa bộ mã hóa siêu dữ liệu (110)

được tạo cấu hình để mã hóa một hoặc nhiều giá trị chênh lệch với số lượng các bit thứ hai, trong đó mỗi giá trị trong số một hoặc nhiều giá trị chênh lệch đã nêu biểu thị số nguyên, trong đó số lượng các bit thứ hai là nhỏ hơn số lượng các bit thứ nhất.

Ví dụ, xem xét rằng trong phương án, các mẫu siêu dữ liệu có thể biểu diễn góc phương vị được mã hóa bởi 8 bit. Ví dụ, góc phương vị có thể là số nguyên nằm giữa $-90 \leq \text{góc phương vị} \leq 90$. Do đó, góc phương vị có thể nhận 181 giá trị khác nhau. Tuy nhiên, nếu có thể giả định rằng (ví dụ, N) mẫu góc phương vị tiếp theo chỉ khác bởi không nhiều hơn, ví dụ, ± 15 , sau đó, 5 bit ($2^5 = 32$) có thể đủ để mã hóa các giá trị chênh lệch. Nếu các giá trị chênh lệch được biểu diễn như các số nguyên, sau đó sự xác định các giá trị chênh lệch tự động biến đổi các giá trị bổ sung, để được truyền dẫn, đến phạm vi giá trị thích hợp.

Ví dụ, xem xét trường hợp trong đó giá trị góc phương vị thứ nhất của đối tượng âm thanh thứ nhất là 60° và các giá trị tiếp theo thay đổi trong khoảng từ 45° đến 75° . Hơn nữa, xem xét rằng giá trị góc phương vị thứ hai của đối tượng âm thanh thứ hai là -30° và các giá trị tiếp theo thay đổi trong khoảng từ -45° đến -15° . Bằng cách xác định các giá trị chênh lệch cho cả hai giá trị tiếp theo của đối tượng âm thanh thứ nhất và cho cả hai giá trị tiếp theo của đối tượng âm thanh thứ hai, các giá trị chênh lệch của giá trị góc phương vị thứ nhất và giá trị góc phương vị thứ hai là cả hai nằm trong phạm vi từ -15° đến $+15^\circ$, để 5 bit là đủ để mã hóa mỗi giá trị trong số các giá trị chênh lệch và để chuỗi bit, mà mã hóa các giá trị chênh lệch, có cùng nghĩa cho các giá trị chênh lệch của góc phương vị thứ nhất và các giá trị chênh lệch của giá trị góc phương vị thứ hai.

Trong phương án, mỗi giá trị chênh lệch, mà không mẫu siêu dữ liệu nào tồn tại trong tín hiệu siêu dữ liệu được nén, được truyền dẫn đến phía bộ giải mã. Hơn nữa, theo phương án, mỗi giá trị chênh lệch, mà không mẫu siêu dữ liệu nào tồn tại trong tín hiệu siêu dữ liệu được nén, được nhận và được xử lý bởi bộ giải mã siêu dữ liệu. Một số phương án được ưu tiên được minh họa bởi Fig.10 và Fig.11, tuy nhiên, nhận biết khái niệm khác.

Fig.10 minh họa sự mã hóa siêu dữ liệu theo phương án khác. Bộ mã hóa siêu dữ liệu 210 theo các phương án có thể được tạo cấu hình để thực hiện sự mã hóa siêu dữ

liệu được minh họa bởi Fig.10.

Như trong một số phương án trước đây, trên Fig.10, các giá trị chênh lệch, ví dụ, được xác định cho mỗi mẫu siêu dữ liệu của tín hiệu siêu dữ liệu ban đầu mà không được bao gồm bởi tín hiệu siêu dữ liệu được nén. Ví dụ, khi các mẫu siêu dữ liệu tại thời điểm $n=0$ và thời điểm $n=N$ được bao gồm bởi tín hiệu siêu dữ liệu được nén, nhưng các mẫu siêu dữ liệu được nén, nhưng các mẫu siêu dữ liệu tại các thời điểm từ $n=1$ đến $n=N-1$, khi đó các giá trị chênh lệch được xác định cho thời điểm $n=1$ đến $n=N-1$.

Tuy nhiên, theo phương án của Fig.10, phép tính gần đúng đa giác khi đó được thực hiện trong 640. Bộ mã hóa siêu dữ liệu 210 được tạo cấu hình để quyết định, các giá trị chênh lệch nào sẽ được truyền dẫn, và liệu các giá trị chênh lệch có được truyền dẫn hay không.

Ví dụ, bộ mã hóa siêu dữ liệu 210 có thể được tạo cấu hình để truyền dẫn chỉ các giá trị chênh lệch này có giá trị chênh lệch mà lớn hơn giá trị ngưỡng.

Trong phương án khác, bộ mã hóa siêu dữ liệu 210 có thể được tạo cấu hình để truyền dẫn chỉ các giá trị chênh lệch này, khi tỉ lệ của giá trị chênh lệch so với mẫu siêu dữ liệu tương ứng là lớn hơn giá trị ngưỡng.

Trong phương án, bộ mã hóa siêu dữ liệu 210 kiểm tra cho giá trị chênh lệch tuyệt đối lớn nhất, liệu giá trị chênh lệch tuyệt đối này lớn hơn giá trị ngưỡng không. Nếu giá trị chênh lệch tuyệt đối này lớn hơn giá trị ngưỡng, khi đó giá trị chênh lệch được truyền dẫn, nếu không không giá trị chênh lệch nào được truyền dẫn và sự kiểm tra kết thúc. Sự kiểm tra liên tục đối với giá trị chênh lệch lớn nhất thứ hai, đối với giá trị lớn nhất thứ ba và tiếp tục, cho đến khi tất cả các giá trị chênh lệch là nhỏ hơn giá trị ngưỡng.

Vì không phải tất cả các giá trị chênh lệch cần thiết, theo các phương án, bộ mã hóa siêu dữ liệu 210 không chỉ mã hóa (kích thước của) giá trị chênh lệch tự bản thân nó (một trong số các giá trị $y_1[k] \dots y_{N-1}[k]$ trên Fig.10), mà cũng truyền dẫn thông tin đến mẫu siêu dữ liệu của tín hiệu siêu dữ liệu ban đầu nào, giá trị chênh lệch liên quan (một trong số các giá trị $x_1[k] \dots x_{N-1}[k]$ trên Fig.10). Ví dụ, bộ mã hóa siêu dữ liệu 210

có thể mã hóa thời điểm mà giá trị chênh lệch liên quan. Ví dụ, bộ mã hóa siêu dữ liệu 210 có thể mã hóa giá trị nằm giữa 1 và $N-1$ để biểu thị mẫu siêu dữ liệu nào nằm giữa các mẫu siêu dữ liệu 0 và N , mà đã được truyền dẫn trong tín hiệu siêu dữ liệu được nén, giá trị chênh lệch liên quan. Liệt kê các giá trị $x_1[k] \dots x_{N-1}[k]$ $y_1[k] \dots y_{N-1}[k]$ tại đầu ra của phép tính gần đúng đa giác không có nghĩa rằng tất cả các giá trị này được truyền dẫn một cách cần thiết, nhưng thay vào đó có nghĩa rằng không, một, một số hoặc tất cả các cặp giá trị này được truyền dẫn, phụ thuộc vào các giá trị chênh lệch.

Trong phương án, bộ mã hóa siêu dữ liệu 210 có thể xử lý phân đoạn của, ví dụ, N , giá trị chênh lệch liên tiếp và làm gần đúng mỗi phân đoạn bởi tiến trình đa giác mà được tạo ra bởi số lượng biến thiên các điểm đa giác được lượng tử hóa $[x_i, y_i]$.

Nó có thể mong muốn rằng số lượng các điểm đa giác mà cần thiết để làm gần đúng tín hiệu chênh lệch với độ chính xác đủ là trung bình nhỏ hơn đáng kể so với N . Và như $[x_i, y_i]$ là các số nguyên nhỏ, chúng có thể được mã hóa với số lượng bit thấp.

Fig.11 minh họa sự mã hóa siêu dữ liệu theo phương án khác. Bộ giải mã siêu dữ liệu 110 theo các phương án có thể được tạo cấu hình để thực hiện sự giải mã siêu dữ liệu được minh họa bởi Fig.11.

Trong các phương án, bộ giải mã siêu dữ liệu 110 nhận một số các giá trị chênh lệch và bổ sung các giá trị chênh lệch này đến các mẫu siêu dữ liệu được nội suy tuyến tính trong 730.

Trong một số phương án, bộ giải mã siêu dữ liệu 110 bổ sung các giá trị chênh lệch được nhận chỉ đến các mẫu siêu dữ liệu được nội suy tuyến tính tương ứng trong 730 và rời các mẫu siêu dữ liệu được nội suy tuyến tính khác, mà không có giá trị chênh lệch nào được nhận, không bị thay đổi.

Tuy nhiên, các phương án mà nhận ra khái niệm khác giờ được mô tả.

Theo phương án như vậy, bộ giải mã siêu dữ liệu 110 được tạo cấu hình để nhận nhiều giá trị chênh lệch cho tín hiệu siêu dữ liệu được nén của một hoặc nhiều tín hiệu siêu dữ liệu được nén. Mỗi giá trị trong số các giá trị chênh lệch có thể được đề cập đến như "giá trị chênh lệch được nhận". Giá trị chênh lệch được nhận được gán cho một mẫu trong số các mẫu siêu dữ liệu được làm gần đúng của tín hiệu siêu dữ liệu

được khôi phục, mà được kết hợp với (được dựng từ) tín hiệu siêu dữ liệu được nén đã nêu, mà các giá trị chênh lệch được nhận liên quan.

Như đã được mô tả đối với Fig.9, bộ giải mã siêu dữ liệu 110 được tạo cấu hình để cộng mỗi giá trị chênh lệch được nhận trong số nhiều giá trị chênh lệch được nhận vào mẫu siêu dữ liệu được làm gần đúng kết hợp với giá trị chênh lệch được nhận đã nêu. Bằng cách thêm giá trị chênh lệch được nhận vào mẫu siêu dữ liệu được làm gần đúng của nó, một trong số các mẫu siêu dữ liệu thứ hai của tín hiệu siêu dữ liệu được khôi phục thu được.

Tuy nhiên, đối với một số (hoặc thỉnh thoảng, hầu hết) các mẫu siêu dữ liệu được làm gần đúng, thường, không giá trị chênh lệch nào được nhận.

Trong một số phương án, bộ giải mã siêu dữ liệu 110 có thể, ví dụ, được tạo cấu hình để xác định giá trị chênh lệch được làm gần đúng phụ thuộc vào một hoặc nhiều giá trị trong số nhiều giá trị chênh lệch được nhận cho mỗi mẫu siêu dữ liệu được làm gần đúng trong số nhiều mẫu siêu dữ liệu được làm gần đúng của tín hiệu siêu dữ liệu được khôi phục được kết hợp với tín hiệu siêu dữ liệu được nén đã nêu, khi không có giá trị nào trong số nhiều giá trị chênh lệch được nhận được kết hợp với mẫu siêu dữ liệu được làm gần đúng đã nêu.

Nói cách khác, đối với tất cả các mẫu siêu dữ liệu được làm gần đúng, mà không có giá trị chênh lệch nào được nhận, giá trị chênh lệch được làm gần đúng được tạo ra phụ thuộc vào một hoặc nhiều giá trị chênh lệch được nhận.

Bộ giải mã siêu dữ liệu 110 được tạo cấu hình để thêm mỗi giá trị chênh lệch được làm gần đúng trong số nhiều giá trị chênh lệch được làm gần đúng cho mẫu siêu dữ liệu được làm gần đúng của giá trị chênh lệch được làm gần đúng để thu được một mẫu khác trong số các mẫu siêu dữ liệu thứ hai của tín hiệu siêu dữ liệu được khôi phục đã nêu.

Trong các phương án khác, tuy nhiên, bộ giải mã siêu dữ liệu 110 làm gần đúng các giá trị chênh lệch cho các mẫu siêu dữ liệu này, mà không có các giá trị chênh lệch nào được nhận, bằng cách tiến hành phép nội suy tuyến tính phụ thuộc vào các giá trị chênh lệch này mà đã được nhận trong bước 740.

Ví dụ, nếu các giá trị chênh lệch thứ nhất và giá trị chênh lệch thứ hai được nhận, sau đó các giá trị chênh lệch được định vị giữa các giá trị chênh lệch được nhận có thể được làm gần đúng, ví dụ, sử dụng sự nội suy tuyến tính.

Ví dụ, khi giá trị chênh lệch thứ nhất tại thời điểm $n=15$ có giá trị chênh lệch $d[15]=5$. Và khi giá trị chênh lệch thứ hai tại thời điểm $n = 18$ có giá trị chênh lệch $d[18]=2$, sau đó, các giá trị chênh lệch cho $n = 16$ và $d=17$ có thể được làm gần đúng một cách tuyến tính như $d[16]=4$ và $d[17] = 3$.

Trong phương án khác, khi các mẫu siêu dữ liệu được bao gồm bởi tín hiệu siêu dữ liệu được nén, các giá trị chênh lệch của các mẫu siêu dữ liệu đã nêu được giả định bằng 0, và phép nội suy tuyến tính của các giá trị chênh lệch mà không được nhận có thể được tiến hành bởi bộ giải mã siêu dữ liệu trên cơ sở các mẫu siêu dữ liệu đã nêu mà được giả định bằng 0.

Ví dụ, khi giá trị chênh lệch đơn $d=8$ được truyền dẫn cho $n = 16$, và khi cho $n = 0$ và $n = 32$, mẫu siêu dữ liệu được truyền dẫn trong tín hiệu siêu dữ liệu được nén, khi đó, các giá trị chênh lệch không được truyền dẫn tại $n=0$ và $n=32$ được giả định là 0.

Để n biểu thị thời gian và để $d[n]$ là giá trị chênh lệch tại thời điểm n . Khi đó:

$$d[16] = 8 \quad (\text{giá trị chênh lệch được nhận})$$

$$d[0] = 0 (\text{giá trị chênh lệch được giả định, như mẫu siêu dữ liệu tồn tại trong } z(k))$$

$$d[32] = 0 \quad (\text{giá trị chênh lệch được giả định, như mẫu siêu dữ liệu tồn tại trong } z(k))$$

các giá trị chênh lệch được làm gần đúng:

$$d[1] = 0,5; d[2] = 1; d[3] = 1,5; d[4] = 2; d[5] = 2,5; d[6] = 3; d[7] = 3,5; d[8] = 4;$$

$$d[9] = 4,5; d[10] = 5; d[11] = 5,5; d[12] = 6; d[13] = 6,5; d[14] = 7; d[15] = 7,5;$$

$$d[17] = 7,5; d[18] = 7; d[19] = 6,5; d[20] = 6; d[21] = 5,5; d[22] = 5; d[23] = 4,5; d[24] = 4;$$

$$d[25] = 3,5; d[26] = 3; d[27] = 2,5; d[28] = 2; d[29] = 1,5; d[30] = 1; d[31] = 0,5.$$

Trong các phương án, các giá trị được nhận cũng như các giá trị chênh lệch được

làm gần đúng được thêm vào các mẫu nội suy tuyến tính tương ứng (trong 730).

Trong phần tiếp theo, các phương án được ưu tiên được mô tả.

Bộ mã hóa siêu dữ liệu (đối tượng) có thể, ví dụ, mã hóa kết hợp chuyển đổi các giá trị quỹ đạo được lấy mẫu (phụ) thường xuyên sử dụng vùng đệm dùng trước của kích thước N đã cho. Ngay khi vùng đệm này được điền đầy, toàn bộ khối dữ liệu sẽ được mã hóa và được truyền dẫn. Dữ liệu đối tượng được mã hóa có thể bao gồm 2 phần, dữ liệu đối tượng được lập mã trong và tùy chọn phần dữ liệu chênh lệch mà chứa cấu trúc mịn của mỗi phân đoạn.

Dữ liệu đối tượng được lập mã trong bao gồm các giá trị được lượng tử hóa $z(k)$ mà được lấy mẫu trên lưới thông thường (ví dụ, mỗi 32 khung âm thanh có chiều dài 1024). Các biến số Boolean có thể được sử dụng để biểu thị rằng các giá trị được chỉ rõ một cách riêng lẻ cho mỗi đối tượng hoặc giá trị theo sau đó là chung cho tất cả các đối tượng.

Bộ giải mã có thể được tạo cấu hình để lấy quỹ đạo thô từ dữ liệu đối tượng được lập mã trong bằng phép nội suy tuyến tính. Cấu trúc mịn của các quỹ đạo được cho bởi phần dữ liệu chênh lệch mà bao gồm sự chênh lệch được mã hóa giữa quỹ đạo đầu vào và phép nội suy tuyến tính. Phép biểu diễn đa giác tổ hợp với các bước lượng tử hóa khác nhau cho góc phương vị, góc nâng, bán kính, và các giá trị độ khuếch đại dẫn đến việc giảm sự không thích hợp như mong muốn.

Phép biểu diễn đa giác có thể thu được từ sự biến thiên của thuật toán Ramer-Douglas-Peucker [10,11] mà không sử dụng phép đệ quy và khác với phương pháp ban đầu bởi tiêu chí hủy bỏ bổ sung, tức là, số lượng tối đa các điểm đa giác cho tất cả các đối tượng và các thành phần đối tượng.

Các điểm đa giác thu được có thể được mã hóa trong phần dữ liệu chênh lệch sử dụng độ dài lệnh biến thiên mà được chỉ rõ trong dòng bit. Các biến số Boolean biểu thị sự mã hóa chung của các giá trị bằng nhau.

Dưới đây, các khung siêu dữ liệu đối tượng theo các phương án và phép biểu diễn kí hiệu theo các phương án được mô tả.

Đối với các lí do hiệu suất, chuỗi các giá trị quỹ đạo được lấy mẫu (phụ) thông

thường được mã hóa kết hợp. Bộ mã hóa có thể sử dụng vùng đệm dùng trước của kích thước đã cho và ngay khi vùng đệm này được điền đầy, toàn bộ khối dữ liệu sẽ được mã hóa và được truyền dẫn. Dữ liệu đối tượng được mã hóa này (ví dụ, các dung lượng có ích cho siêu dữ liệu đối tượng) có thể, ví dụ, bao gồm hai phần, dữ liệu đối tượng được lập mã trong (phần thứ nhất) và, tùy chọn, phần dữ liệu chênh lệch (phần thứ hai).

Một số hoặc tất cả các phần của cú pháp dưới đây có thể, ví dụ, được sử dụng:

	Số lượng các bit	Gợi ý
<pre> object_metadata() { intracoded_object_metadata() has_differential_metadata; if (has_differential_metadata) { differential_object_metadata(); } } </pre>	1	bslbf

Dưới đây, dữ liệu đối tượng được lập mã trong theo phương án sẽ được mô tả:

Để hỗ trợ sự truy cập ngẫu nhiên của siêu dữ liệu đối tượng được mã hóa, đặc điểm kỹ thuật đầy đủ và độc lập của tất cả các siêu dữ liệu đối tượng cần được truyền dẫn một cách thông thường. Điều này được nhận ra qua dữ liệu đối tượng được lập mã trong (“các khung I”) mà chứa các giá trị được lượng tử hóa được lấy mẫu trên lưới thông thường (ví dụ, mỗi 32 khung có độ dài 1024). Các khung I này có cú pháp sau, trong đó *position_azimuth*, *position_elevation*, *position_radius*, và *gain_factor* chỉ rõ các giá trị được lượng tử hóa trong các khung *iframe_period* sau khung I hiện thời:

	Số lượng các bit	Gợi ý
--	------------------	-------

```

intracoded_object_metadata()
{
    Ifperiod;                                6                                uimsbf
    if (num_objects>1) {
        common_azimuth;                      1                                bslbf
        if (common_azimuth) {
            default_azimuth;                 8                                tcimsbf
        }
        else {
            for (o=1:num_objects) {
                position_azimuth[o];           8                                tcimsbf
            }
        }
        common_elevation;                    1                                bslbf
        if (common_elevation) {
            default_elevation;               6                                tcimsbf
        }
        else {
            for (o=1:num_objects) {
                position_elevation[o];         6                                tcimsbf
            }
        }
        common_radius;                       1                                bslbf
        if (common_radius) {
            default_radius;                  4                                uimsbf
        }
        else {
            for (o=1:num_objects) {
                position_radius[o];            4                                uimsbf
            }
        }
    }
}

```

common_gain;	1	bslbf
if (common_radius) {		
default_gain;	7	tcimsbf
}		
else {		
for (o=1:num_objects) {		
gain_factor[o];	7	tcimsbf
}		
}		
}		
else {		
position_azimuth;	8	tcimsbf
position_elevation;	6	tcimsbf
position_radius;	4	tcimsbf
gain_factor;	7	tcimsbf
}		
}		
Lưu ý: iframe_period = ifperiod + 1;		

Dưới đây, dữ liệu đối tượng chệch lệch trong theo phương án sẽ được mô tả.

Phép tính gần đúng với độ chính xác cao hơn đạt được bằng cách truyền dẫn các tiến trình đa giác trên cơ sở số lượng được giảm của các điểm lấy mẫu. Theo đó, ma trận ba chiều mỏng có thể được truyền dẫn, trong đó chiều thứ nhất có thể là chỉ số đối tượng, chiều thứ hai có thể được tạo ra bởi các thành phần siêu dữ liệu (góc phương vị, góc nâng, bán kính, và độ khuếch đại), và chiều thứ ba có thể là chỉ số khung của các điểm lấy mẫu đa giác. Không có các số đo khác, sự biểu thị mà các phần tử của ma trận bao gồm các giá trị đã yêu cầu $num_objects * num_components * (iframe_period - 1)$ bit. Bước thứ nhất để giảm lượng bit này có thể thêm bốn cờ hiệu mà biểu thị liệu có ít nhất một giá trị mà thuộc về một trong số bốn thành phần. Ví dụ, có thể hi vọng rằng chỉ trong một ít trường hợp sẽ có bán kính hoặc các giá trị độ khuếch đại chệch lệch. Chiều thứ ba của ma trận 3 chiều được giảm bao gồm vectơ với $iframe_period - 1$

phần tử. Nếu chỉ số lượng nhỏ các điểm đa giác được mong, khi đó nó có thể hiệu quả hơn để tham số hóa vectơ này bởi tập hợp các chỉ số khung và số các yếu tố trong tập hợp này Ví dụ, đối với *iframe_period* của $N_{period} = 32$ khung, số lượng tối đa của 16 điểm đa giác, phương pháp này có thể có lợi đối với $N_{points} < (32 - \log_2(16)) / \log_2(32) = 5,6$ điểm đa giác. Theo các phương án, cú pháp sau đối với sơ đồ lập mã như vậy được sử dụng:

	Số lượng các bit	Gợi ý
differential_object_metadata() {		
bits_per_point;	4	uimsbf
fixed_azimuth;	1	bslbf
if (!fixed_azimuth) {		
for (o=1:num_objects) {		
flag_azimuth;	1	bslbf
if (flag_azimuth) {		
num_points = offset_data();		
nbits_azimuth;	3	uimsbf
for (p=1:num_points) {		
differential_azimuth[o][p];	num_bits	tcimsbf
}		
}		
}		
fixed_elevation;	1	bslbf
if (!fixed_elevation) {		
for (o=1:num_objects) {		
flag_elevation;	1	bslbf
if (flag_elevation) {		
num_points = offset_data();		

nbits_elevation;	3	uimsbf
for (p=1:num_points) {		
	num_bits	tcimsbf
differential_elevation[o][p];		
}		
}		
}		
fixed_radius;	1	bslbf
if (!fixed_radius) {		
for (o=1:num_objects) {		
flag_radius;	1	bslbf
if (flag_radius) {		
num_points = offset_data();		
nbits_radius	3	uimsbf
for (p=1:num_points) {		
differential_radius[o][p];	num_bits	tcimsbf
}		
}		
}		
}		
fixed_gain;	1	bslbf
if (!fixed_gain) {		
for (o=1:num_objects) {		
flag_gain;	1	bslbf
if (flag_gain) {		
num_points = offset_data();		
nbits_gain;	3	uimsbf
for (p=1:num_points) {		
differential_gain[o][p];	num_bits	tcimsbf
}		

<pre> } } } } </pre>
Lưu ý: num_bits = nbits_* + 2;

	Số lượng các bit	Gợi ý
int offset_data() {		
bitfield_syntax	1	bslbf
if (bitfield_syntax) {		
offset_bitfield	iframe_period-1	bslbf array
num_points = sum(offset_bitfield)		
}		
else {		
npoints;	bits_per_point	uimsbf
num_points = npoints + 1;		
for (p=1:num_points) {		
foffset[p];	ceil(log2(iframe_period	uimsbf
	-1))	
}		
}		
return num_points;		
}		

Macro *offset_data()* mã hóa các vị trí (sự kết thúc khung) của các điểm đa giác, hoặc như trường bit đơn giản hoặc sử dụng các khái niệm được mô tả trên đây. Các giá trị *num_bits* cho phép mã hóa các sự rẽ nhánh vị trí lớn khi phần còn lại của dữ liệu chênh lệch được mã hóa với kích thước lệnh nhỏ hơn.

Cụ thể, trong phương án, các macro nêu trên có thể, ví dụ, có nghĩa sau:

Định nghĩa của các dung lượng có ích `object_metadata()` theo phương án:

has_differential_metadata biểu thị liệu siêu dữ liệu đối tượng chênh lệch có mặt không.

Định nghĩa của các dung lượng có ích `intracoded_object_metadata()` theo phương án:

ifperiod định rõ số lượng các khung ở giữa các khung độc lập.

common_azimuth biểu thị liệu góc phương vị chung có được sử dụng cho tất cả các đối tượng không.

default_azimuth định rõ giá trị của góc phương vị chung.

position_azimuth nếu không có giá trị góc phương vị chung, giá trị cho mỗi đối tượng được truyền dẫn.

common_elevation biểu thị liệu góc nâng có được sử dụng cho tất cả các đối tượng.

default_elevation định rõ giá trị của góc nâng chung.

position_elevation nếu không có giá trị nâng chung, giá trị cho mỗi đối tượng được truyền dẫn.

common_radius biểu thị liệu giá trị bán kính chung có được sử dụng cho tất cả các đối tượng.

default_radius định rõ giá trị của bán kính chung.

position_radius nếu không có giá trị bán kính chung, giá trị cho mỗi đối tượng được truyền dẫn.

common_gain biểu thị liệu giá trị độ khuếch đại chung có được sử dụng cho tất cả các đối tượng.

default_gain biểu thị giá trị của hệ số khuếch đại chung.

gain_factor nếu không có giá trị độ khuếch đại chung, giá trị cho mỗi đối tượng được truyền dẫn.

position_azimuth nếu chỉ có một đối tượng, đây là góc phương vị của nó.

position_elevation nếu chỉ có một đối tượng, đây là góc nâng của nó.

position_radius nếu chỉ có một đối tượng, đây là bán kính của nó.

gain_factor nếu chỉ có một đối tượng, đây là hệ số khuếch đại của nó.

Định nghĩa của các dung lượng có ích differential_object_metadata() payloads
theo phương án:

bits_per_point số lượng các bit được yêu cầu để biểu diễn số lượng các điểm đa giác.

fixed_azimuth cờ hiệu biểu thị liệu giá trị góc phương vị có cố định cho tất cả các đối tượng không.

flag_azimuth cờ hiệu mỗi đối tượng biểu thị liệu giá trị góc phương vị có thay đổi không.

nbits_azimuth bao nhiêu bit được yêu cầu để biểu diễn giá trị chênh lệch.

differential_azimuth giá trị của sự chênh lệch giữa giá trị được nội suy tuyến tính và giá trị thực tế.

fixed_elevation cờ hiệu biểu thị liệu giá trị góc nâng có cố định cho tất cả các đối tượng không.

flag_elevation cờ hiệu mỗi đối tượng biểu thị liệu giá trị góc nâng có thay đổi không.

nbits_elevation bao nhiêu bit được yêu cầu để biểu diễn giá trị chênh lệch.

differential_elevation giá trị của sự chênh lệch giữa giá trị được nội suy tuyến tính và giá trị thực tế.

fixed_radius cờ hiệu biểu thị liệu giá trị bán kính có cố định cho tất cả các đối tượng không.

flag_radius cờ hiệu mỗi đối tượng biểu thị liệu giá trị bán kính có thay đổi không.

nbits_radius bao nhiêu bit được yêu cầu để biểu diễn giá trị chênh lệch.

differential_radius giá trị của sự chênh lệch giữa giá trị được nội suy tuyến tính và giá trị thực tế.

fixed_gain cờ hiệu biểu thị liệu hệ số khuếch đại có cố định cho tất cả các đối tượng không.

flag_gain cờ hiệu mỗi đối tượng biểu thị liệu bán kính độ khuếch đại có thay đổi không.

nbits_gain bao nhiêu bit được yêu cầu để biểu diễn giá trị chênh lệch.

differential_gain giá trị của sự chênh lệch giữa giá trị được nội suy tuyến tính và giá trị thực tế.

Định nghĩa của các dung lượng có ích offset_data() theo phương án:

bitfield_syntax cờ hiệu biểu thị liệu véc tơ với các chỉ số đa giác có mặt trong dòng bit.

offset_bitfield mảng bool chứa cờ hiệu cho mỗi điểm của iframe_period liệu đó có là điểm đa giác hay không.

npoints số lượng các điểm đa giác trừ 1 ($\text{num_points} = \text{npoints} + 1$).

foffset chỉ số lát thời gian của các điểm đa giác nằm trong iframe_period ($\text{frame_offset} = \text{foffset} + 1$).

Theo phương án, siêu dữ liệu có thể, ví dụ, được truyền cho mọi đối tượng âm thanh như các vị trí đã cho (ví dụ, được biểu thị bằng góc phương vị, góc nâng, và bán kính) tại các tem thời gian được định rõ.

Trong tình trạng kỹ thuật, không công nghệ linh hoạt nào tồn tại một mặt tổ hợp kênh lập mã và một mặt lập mã đối tượng để thu được các chất lượng âm thanh có thể chấp nhận được tại các tốc độ bit thấp.

Hạn chế này được khắc phục bởi hệ thống bộ mã hóa-giải mã âm thanh 3D. Bây giờ, hệ thống bộ mã hóa-giải mã âm thanh 3D được mô tả.

Fig.12 minh họa bộ mã hóa âm thanh 3D phù hợp với phương án của sáng chế.

Bộ mã hóa âm thanh 3D được tạo cấu hình để mã hóa dữ liệu đầu vào âm thanh 101 thành dữ liệu đầu ra âm thanh 501. Bộ mã hóa âm thanh 3D bao gồm giao diện đầu vào để nhận nhiều kênh âm thanh được biểu thị bởi CH và nhiều đối tượng âm thanh được biểu thị bởi OBJ. Hơn nữa, như được minh họa trên Fig.12, giao diện đầu vào 1100 ngoài ra nhận siêu dữ liệu liên quan đến một hoặc nhiều đối tượng trong số nhiều đối tượng âm thanh OBJ. Hơn nữa, bộ mã hóa âm thanh 3D bao gồm bộ trộn 200 để trộn nhiều đối tượng và nhiều kênh để thu được nhiều kênh được trộn trước, trong đó mỗi kênh được trộn trước bao gồm dữ liệu âm thanh của kênh và dữ liệu âm thanh của ít nhất một đối tượng..

Hơn nữa, bộ mã hóa âm thanh 3D bao gồm bộ mã hóa lõi 300 để mã hóa lõi dữ liệu đầu vào bộ mã hóa lõi, bộ nén siêu dữ liệu 400 để nén siêu dữ liệu liên quan đến một hoặc nhiều đối tượng trong số nhiều đối tượng âm thanh.

Hơn nữa, bộ mã hóa âm thanh 3D có thể bao gồm bộ điều khiển chế độ 600 để điều khiển bộ trộn, bộ mã hóa lõi và/hoặc giao diện đầu ra 500 trong một trong số các chế độ vận hành, trong đó trong chế độ thứ nhất, bộ mã hóa lõi được tạo cấu hình để mã hóa nhiều kênh âm thanh và nhiều đối tượng âm thanh được nhận bởi giao diện đầu vào 1100 mà không có sự tương tác nào bởi bộ trộn, tức là, không có sự trộn nào bởi bộ trộn 200. Tuy nhiên, trong chế độ thứ hai, trong đó bộ trộn 200 hoạt động, bộ mã hóa lõi mã hóa nhiều kênh được trộn, tức là, đầu ra được tạo bởi khối 200. Trong trường hợp sau, được ưu tiên không mã hóa bất kỳ dữ liệu đối tượng nào nữa. Thay vào đó, siêu dữ liệu biểu thị các vị trí của đối tượng âm thanh đã được sử dụng bởi bộ trộn 200 để kết xuất các đối tượng lên các kênh như được biểu thị bởi siêu dữ liệu. Nói cách khác, bộ trộn 200 sử dụng siêu dữ liệu liên quan đến nhiều đối tượng âm thanh để kết xuất trước các đối tượng âm thanh và sau đó các đối tượng âm thanh được kết xuất trước được trộn với các kênh để thu được các kênh được trộn tại đầu ra của bộ trộn. Trong phương án này, bất kỳ đối tượng nào có thể không cần thiết được truyền dẫn và điều này cũng áp dụng cho siêu dữ liệu được nén như đầu ra bởi khối 400. Tuy nhiên, nếu không phải tất cả các đối tượng được nhập vào trong giao diện 1100 được trộn nhưng chỉ lượng nhất định các đối tượng được trộn, khi đó chỉ các đối tượng không được trộn còn lại và siêu dữ liệu được kết hợp tuy nhiên được truyền dẫn lần lượt tới bộ mã hóa lõi 300 hoặc bộ nén siêu dữ liệu 400.

Trên Fig.12, bộ nén siêu dữ liệu 400 là bộ giải mã siêu dữ liệu 210 của thiết bị 250 để tạo ra thông tin âm thanh được mã hóa theo một trong số các phương án được mô tả ở trên. Hơn nữa, trên Fig.12, bộ trộn 200 và bộ mã hóa lõi 300 cùng tạo ra bộ mã hóa âm thanh 220 của thiết bị 250 để tạo ra thông tin âm thanh được mã hóa theo một trong số các phương án được mô tả ở trên.

Fig.14 minh họa phương án khác của bộ mã hóa âm thanh 3D mà, ngoài ra, bao gồm bộ mã hóa SAOC 800. Bộ mã hóa SAOC 800 được tạo cấu hình để tạo ra một hoặc nhiều kênh vận chuyển và dữ liệu theo tham số từ dữ liệu đầu vào bộ mã hóa đối tượng âm thanh không gian. Như được minh họa trên Fig.14, dữ liệu đầu vào bộ mã hóa đối tượng âm thanh không gian là các đối tượng mà chưa được xử lý bởi bộ kết xuất trước/bộ trộn. Cách khác, giả sử rằng bộ kết xuất trước/bộ trộn đã được rẽ nhánh như trong chế độ một trong đó sự lập mã kênh/đối tượng riêng hoạt động, tất cả các đối tượng được nhập vào giao diện nhận 1100 được mã hóa bởi bộ mã hóa SAOC 800.

Ngoài ra, như được minh họa trên Fig.14, bộ mã hóa lõi 300 được ưu tiên thực hiện như bộ mã hóa USAC, tức là, như bộ mã hóa như được định rõ và được tiêu chuẩn hóa trong tiêu chuẩn lập mã tiếng nói và âm thanh hợp nhất (unified speech and audio coding - USAC) MPEG. Đầu ra của toàn bộ bộ mã hóa âm thanh 3D được minh họa trên Fig.14 là dòng dữ liệu MPEG 4 có các cấu trúc tương tự bộ chứa cho các loại dữ liệu riêng. Hơn nữa, siêu dữ liệu được biểu thị như dữ liệu "OAM" và bộ nén siêu dữ liệu 400 trên Fig.12 tương ứng với bộ mã hóa OAM 400 để thu dữ liệu OAM được nén mà được nhập vào bộ mã hóa USAC 300 mà, như có thể được nhìn thấy trên Fig.14, ngoài ra bao gồm giao diện đầu ra để thu được dòng dữ liệu đầu ra MP4 không chỉ có dữ liệu kênh/đối tượng được mã hóa mà cũng có dữ liệu OAM được nén.

Trên Fig.14, bộ mã hóa OAM 400 là bộ giải mã siêu dữ liệu 210 của thiết bị 250 để tạo ra thông tin âm thanh được mã hóa theo một trong số các phương án được mô tả ở trên. Hơn nữa, trên Fig.14, bộ mã hóa SAOC 800 và bộ mã hóa USAC 300 cùng tạo ra bộ mã hóa âm thanh 220 của thiết bị 250 để tạo ra thông tin âm thanh được mã hóa theo một trong số các phương án được mô tả ở trên.

Fig.16 minh họa phương án khác của bộ mã hóa âm thanh 3D, trong đó ngược lại với Fig.14, bộ mã hóa SAOC có thể được tạo cấu hình để hoặc mã hóa, với thuật toán

mã hóa SAOC, các kênh được cung cấp tại bộ kết xuất trước/bộ trộn 200 không hoạt động trong chế độ này, hoặc, cách khác, để mã hóa SAOC các kênh được kết xuất trước cùng với các đối tượng. Do đó, trên Fig.16, bộ mã hóa SAOC có thể vận hành ba loại dữ liệu đầu vào khác nhau, tức là, các kênh không có bất kỳ đối tượng được kết xuất trước nào, các kênh và các đối tượng được kết xuất trước hoặc chỉ mình các đối tượng. Hơn nữa, được ưu tiên để cung cấp bộ giải mã OAM 420 bổ sung trên Fig.16 để bộ mã hóa SAOC 800 sử dụng, cho sự xử lý của nó, dữ liệu giống nhau như trên phía bộ giải mã, tức là, dữ liệu được thu bởi sự nén giảm hơn là dữ liệu OAM ban đầu.

Bộ mã hóa âm thanh 3D trên Fig.16 có thể vận hành trong một số chế độ riêng.

Ngoài các chế độ thứ nhất và thứ hai như được thảo luận trong ngữ cảnh của Fig.12, bộ mã hóa âm thanh trên Fig.16 có thể ngoài ra vận hành trong chế độ thứ ba trong đó bộ mã hóa lõi tạo ra một hoặc nhiều kênh vận chuyển từ các đối tượng riêng khi bộ kết xuất trước/bộ trộn 200 không hoạt động. Ngoài ra hoặc cách khác, trong chế độ thứ ba này, bộ mã hóa SAOC 800 có thể tạo ra một hoặc nhiều kênh vận chuyển thay thế hoặc bổ sung từ các kênh ban đầu, tức là, một lần nữa khi bộ kết xuất trước/bộ trộn 200 liên quan đến bộ trộn 200 của Fig.12 không hoạt động.

Cuối cùng, bộ mã hóa SAOC 800 có thể mã hóa, khi bộ mã hóa âm thanh 3D được tạo cấu hình trong chế độ thứ tư, các kênh kèm các đối tượng được kết xuất trước như được tạo ra bởi bộ kết xuất trước/bộ trộn. Do đó, trong chế độ thứ tư, sự áp dụng tốc độ bit thấp nhất sẽ cung cấp chất lượng tốt do thực tế rằng các kênh và các đối tượng đã được biến đổi hoàn toàn thành các kênh vận chuyển SAOC và thông tin phụ được kết hợp như được biểu thị trên Fig.3 và Fig.5 như “SAOC-SI” và, ngoài ra, bất kỳ siêu dữ liệu được nén không phải được truyền dẫn trong chế độ thứ tư này.

Trên Fig.16, bộ mã hóa OAM 400 là bộ giải mã siêu dữ liệu 210 của thiết bị 250 để tạo ra thông tin âm thanh được mã hóa theo một trong số các phương án được mô tả ở trên. Hơn nữa, trên Fig.16, bộ mã hóa SAOC 800 và bộ mã hóa USAC 300 cùng tạo ra bộ mã hóa âm thanh 220 của thiết bị 250 để tạo ra thông tin âm thanh được mã hóa theo một trong số các phương án được mô tả ở trên.

Theo phương án, thiết bị mã hóa dữ liệu đầu vào âm thanh 101 để thu được dữ liệu đầu ra âm thanh 501 được đề xuất. Thiết bị mã hóa dữ liệu đầu vào âm thanh 101

bao gồm:

- giao diện đầu vào 1100 để nhận nhiều kênh âm thanh, nhiều đối tượng âm thanh và siêu dữ liệu liên quan đến một hoặc nhiều đối tượng trong số nhiều đối tượng âm thanh.

- bộ trộn 200 để trộn nhiều đối tượng và nhiều kênh để thu được nhiều kênh được trộn trước, mỗi kênh được trộn trước bao gồm dữ liệu âm thanh của kênh và dữ liệu âm thanh của ít nhất một đối tượng, và

- thiết bị 250 để tạo ra thông tin âm thanh được mã hóa mà bao gồm bộ mã hóa siêu dữ liệu và bộ mã hóa âm thanh như được mô tả trên đây.

Bộ mã hóa âm thanh 220 của thiết bị 250 để tạo ra thông tin âm thanh được mã hóa là bộ mã hóa lõi 300 để mã hóa lõi dữ liệu đầu vào bộ mã hóa lõi.

Bộ mã hóa siêu dữ liệu 210 của thiết bị 250 để tạo ra thông tin âm thanh được mã hóa là bộ nén siêu dữ liệu 400 để nén siêu dữ liệu liên quan đến một hoặc nhiều đối tượng âm thanh.

Fig.13 minh họa bộ giải mã âm thanh 3D phù hợp với phương án của sáng chế. Bộ giải mã âm thanh 3D nhận, như đầu vào, dữ liệu âm thanh được mã hóa, tức là, dữ liệu 501 của Fig.12.

Bộ giải mã âm thanh 3D bao gồm bộ giải nén siêu dữ liệu 1400, bộ giải mã lõi 1300, bộ xử lý đối tượng 1200, bộ điều khiển chế độ 1600 và bộ xử lý sau 1700.

Cụ thể, bộ giải mã âm thanh 3D được tạo cấu hình để giải mã dữ liệu âm thanh được mã hóa và giao diện đầu vào được tạo cấu hình để nhận dữ liệu âm thanh được mã hóa, dữ liệu âm thanh được mã hóa bao gồm nhiều kênh được mã hóa và nhiều đối tượng được mã hóa và siêu dữ liệu được nén liên quan đến nhiều đối tượng trong chế độ nhất định.

Hơn nữa, bộ giải mã lõi 1300 được tạo cấu hình để giải mã hóa nhiều kênh được mã hóa và nhiều đối tượng được mã hóa và, ngoài ra, bộ giải nén siêu dữ liệu được tạo cấu hình để giải nén siêu dữ liệu được nén.

Hơn nữa, bộ xử lý đối tượng 1200 được tạo cấu hình để xử lý nhiều đối tượng

được giải mã như được tạo ra bởi giải mã lõi 1300 sử dụng siêu dữ liệu được giải nén để thu được số lượng các kênh đầu ra được định trước bao gồm dữ liệu đối tượng và các kênh được giải mã. Các kênh đầu ra như được biểu thị tại 1205 sau đó được nhập vào bộ xử lý sau 1700. Bộ xử lý sau 1700 được tạo cấu hình để chuyển đổi số lượng các kênh đầu ra 1205 thành định dạng đầu ra nhất định mà có thể là định dạng đầu ra lập thể hoặc định dạng đầu ra loa phát thanh như định dạng đầu ra 5.1, 7.1, v.v..

Cụ thể, bộ giải mã âm thanh 3D bao gồm bộ điều khiển chế độ 1600 mà được tạo cấu hình để phân tích dữ liệu được mã hóa để lựa chọn bộ chỉ thị chế độ. Do đó, bộ điều khiển chế độ 1600 được nối với giao diện đầu vào 1100 trong Fig.13. Tuy nhiên, ngoài ra, bộ điều khiển chế độ không nhất thiết phải ở đây. Thay vào đó, bộ giải mã âm thanh có thể được thiết lập trước bởi bất kỳ loại dữ liệu điều khiển như đầu vào người sử dụng hoặc bất kỳ sự điều khiển khác. Bộ giải mã âm thanh 3D trên Fig.13 và, cụ thể được điều khiển bởi bộ điều khiển chế độ 1600, được tạo cấu hình để hoặc rẽ nhánh bộ xử lý đối tượng và để nạp nhiều kênh được giải mã vào bộ xử lý sau 1700. Đây là sự vận hành trong chế độ 2, trong đó, chỉ các kênh được kết xuất trước được nhận, tức là, khi chế độ 2 đã được áp dụng trong bộ mã hóa âm thanh 3D của Fig.12. Ngoài ra, khi chế độ 1 đã được áp dụng trong bộ mã hóa âm thanh 3D, tức là, khi bộ mã hóa âm thanh 3D đã thực hiện sự lập mã kênh/đối tượng đơn, thì bộ xử lý đối tượng 1200 không được rẽ nhánh, nhưng nhiều kênh được giải mã và nhiều đối tượng được giải mã được nạp vào bộ xử lý đối tượng 1200 cùng với siêu dữ liệu được giải nén bởi bộ giải nén siêu dữ liệu 1400.

Cụ thể hơn, sự biểu thị liệu chế độ 1 hoặc chế độ 2 để được áp dụng được bao gồm trong dữ liệu âm thanh được mã hóa và khi đó bộ điều khiển chế độ 1600 phân tích dữ liệu được mã hóa để phát hiện sự biểu thị chế độ. Chế độ 1 được sử dụng khi sự biểu thị chế độ biểu thị rằng dữ liệu âm thanh được mã hóa bao gồm kênh được mã hóa và các đối tượng được mã hóa và chế độ 2 được áp dụng khi sự biểu thị chế độ biểu thị rằng dữ liệu âm thanh được mã hóa không chứa bất kỳ đối tượng âm thanh nào, tức là, chỉ chứa các kênh được kết xuất trước thu được bởi chế độ 2 của bộ mã hóa âm thanh của Fig.12.

Trên Fig.13, bộ giải nén siêu dữ liệu 1400 là bộ giải mã siêu dữ liệu 110 của thiết

bị 100 để tạo ra một hoặc nhiều kênh âm thanh theo một trong số các phương án được mô tả ở trên. Hơn nữa, trên Fig.13, bộ giải mã lõi 1300, bộ xử lý đối tượng 1200 và bộ xử lý sau 1700 cùng tạo ra bộ giải mã âm thanh 120 của thiết bị 100 để tạo ra một hoặc nhiều kênh âm thanh theo một trong số các phương án được mô tả ở trên.

Fig.15 minh họa phương án so với bộ giải mã âm thanh 3D của Fig.13 và phương án của Fig.15 tương ứng với bộ mã hóa âm thanh 3D của Fig.14. Ngoài sự thực hiện bộ giải mã âm thanh 3D của Fig.13, bộ giải mã âm thanh 3D của Fig.15 bao gồm bộ giải mã SAOC 1800. Hơn nữa, bộ xử lý đối tượng 1200 của Fig.13 được thực hiện như bộ kết xuất đối tượng tách riêng 1210 trong bộ trộn 1220 trong khi, phụ thuộc vào chế độ, chức năng của bộ kết xuất đối tượng 1210 có thể cũng được thực hiện bởi bộ giải mã SAOC 1800.

Hơn nữa, bộ xử lý sau 1700 có thể được thực hiện như bộ kết xuất lập thể 1710 hoặc bộ chuyển đổi định dạng 1720. Ngoài ra, đầu ra trực tiếp của dữ liệu 1205 của Fig.13 có thể cũng được thực hiện như được mô tả bởi 1730. Do đó, ưu tiên để thực hiện sự xử lý trong bộ giải mã trên số lượng cao nhất các kênh như 22.2 hoặc 32 để có sự linh hoạt và để sau đó xử lý sau nếu định dạng nhỏ hơn được yêu cầu. Tuy nhiên, khi rõ ràng rằng ngay từ đầu rằng chỉ định dạng nhỏ như định dạng 5.1 được yêu cầu, khi đó được ưu tiên, như được biểu thị bởi Fig.13 hoặc 6 bởi đường tắt 1727, rằng sự điều khiển nhất định qua bộ giải mã SAOC và/hoặc bộ giải mã USAC có thể được áp dụng để tránh các thao tác trộn tăng không cần thiết và các thao tác trộn giảm tiếp theo.

Trong phương án được ưu tiên của sáng chế, bộ xử lý đối tượng 1200 bao gồm bộ giải mã SAOC 1800 và bộ giải mã SAOC được tạo cấu hình để giải mã một hoặc nhiều kênh vận chuyển được phát ra bởi bộ giải mã lõi và dữ liệu tham số được kết hợp và sử dụng siêu dữ liệu được giải nén để thu được nhiều đối tượng âm thanh được kết xuất. Để đạt được điều này, đầu ra OAM được nối với hộp 1800.

Hơn nữa, bộ xử lý đối tượng 1200 được tạo cấu hình để kết xuất các đối tượng được giải mã được phát ra bởi bộ giải mã lõi mà không được mã hóa trong các kênh vận chuyển SAOC nhưng được mã hóa riêng lẻ trong các phần tử được tạo kênh đơn tiêu biểu như được biểu thị bởi bộ kết xuất đối tượng 1210. Hơn nữa, bộ giải mã bao

gồm giao diện đầu ra tương ứng với đầu ra 1730 để phát ra đầu ra của bộ trộn đến các loa phát thanh.

Trong phương án khác, bộ xử lý đối tượng 1200 bao gồm bộ giải mã sự lập mã đối tượng âm thanh không gian 1800 để giải mã một hoặc nhiều kênh vận chuyển và thông tin phụ theo tham số được kết hợp biểu diễn các tín hiệu âm thanh được mã hóa hoặc các kênh âm thanh được mã hóa, trong đó bộ giải mã sự lập mã đối tượng âm thanh không gian được tạo cấu hình để vận chuyển thông tin theo tham số được kết hợp và siêu dữ liệu được giải nén thành thông tin phụ theo tham số được chuyển mã có thể sử dụng cho sự kết xuất trực tiếp định dạng đầu ra, như ví dụ được định rõ trong phiên bản trước của SAOC. Bộ xử lý sau 1700 được tạo cấu hình để tính toán các kênh âm thanh của định dạng đầu ra sử dụng các kênh vận chuyển được giải mã và thông tin phụ theo tham số được chuyển mã. Sự xử lý được thực hiện bởi bộ xử lý sau có thể tương tự với sự xử lý bao quanh MPEG hoặc có thể là bất kỳ sự xử lý khác như sự xử lý BCC hoặc tương tự.

Trong phương án khác, bộ xử lý đối tượng 1200 bao gồm bộ giải mã sự lập mã đối tượng âm thanh không gian 1800 được tạo cấu hình để trực tiếp trộn tăng và kết xuất các tín hiệu kênh cho định dạng đầu ra sử dụng các kênh vận chuyển được giải mã (bởi bộ giải mã lõi) và thông tin phụ theo tham số.

Hơn nữa, và quan trọng hơn, bộ xử lý đối tượng 1200 của Fig.13 ngoài ra còn bao gồm bộ trộn 1220 mà nhận, như đầu vào, đầu ra dữ liệu bởi bộ giải mã USAC 1300 một cách trực tiếp khi các đối tượng được kết xuất trước được trộn với các kênh tồn tại, tức là, khi bộ trộn 200 của Fig.12 hoạt động. Ngoài ra, bộ trộn 1220 nhận dữ liệu từ bộ kết xuất đối tượng thực hiện sự kết xuất đối tượng mà không có sự giải mã SAOC. Hơn nữa, bộ trộn nhận dữ liệu đầu ra bộ giải mã SAOC, tức là, các đối tượng được kết xuất SAOC.

Bộ trộn 1220 được nối với giao diện đầu ra 1730, bộ kết xuất lập thể 1710 và bộ chuyển đổi 1720. Bộ kết xuất lập thể 1710 được tạo cấu hình để kết xuất các kênh đầu ra thành hai kênh lập thể sử dụng các hàm chuyển đổi liên quan đến đầu hoặc các đáp ứng xung trong phòng lập thể (binaural room impulse responses - BRIR). Bộ chuyển đổi định dạng 1720 được tạo cấu hình để chuyển đổi các kênh đầu ra thành định dạng

đầu ra có số lượng ít các kênh hơn là các kênh đầu ra 1205 của bộ trộn và bộ chuyển đổi định dạng 1720 yêu cầu thông tin về sự sắp xếp sự tái tạo như các loa 5.1 hoặc tương tự.

Trên Fig.15, bộ giải mã OAM 1400 là bộ giải mã siêu dữ liệu 110 của thiết bị 100 để tạo ra một hoặc nhiều kênh âm thanh theo một trong số các phương án được mô tả ở trên. Hơn nữa, trên Fig.15, bộ kết xuất đối tượng 1210, bộ giải mã USAC 1300 và bộ trộn 1220 cùng tạo ra bộ giải mã âm thanh 120 của thiết bị 100 để tạo ra một hoặc nhiều kênh âm thanh theo một trong số các phương án được mô tả ở trên.

Bộ giải mã âm thanh 3D của Fig.17 khác với bộ giải mã âm thanh 3D của Fig.15 ở điểm rằng bộ giải mã SAOC không thể chỉ tạo ra các đối tượng được kết xuất mà cũng các kênh được kết xuất và đây là trường hợp khi bộ mã hóa âm thanh 3D của Fig.16 đã được sử dụng và kết nối 900 giữa các kênh/các đối tượng được kết xuất trước và bộ mã hóa SAOC 800 giao diện đầu vào hoạt động.

Hơn nữa, giai đoạn ghép nhẵn biên độ trên cơ sở vectơ (vector base amplitude panning - VBAP) 1810 được tạo cấu hình mà nhận, từ bộ giải mã SAOC, thông tin sự sắp xếp sự tái tạo và phát ra ma trận kết xuất đến bộ giải mã để bộ giải mã SAOC có thể, cuối cùng, cung cấp các kênh được kết xuất mà không có bất kỳ thao tác khác của bộ trộn trong định dạng kênh cao của 1205, tức là, 32 loa phát thanh.

khối VBAP ưu tiên nhận dữ liệu OAM được giải mã để kết xuất các ma trận kết xuất. Chung hơn, nó ưu tiên yêu cầu thông tin hình học không chỉ của sự sắp xếp sự tái tạo mà cũng là các vị trí mà các tín hiệu đầu vào nên được kết xuất đến sự sắp đặt sự tái tạo. Dữ liệu đầu vào hình học có thể là dữ liệu OAM cho các đối tượng hoặc thông tin vị trí kênh cho các kênh mà đã được truyền dẫn sử dụng SAOC.

Tuy nhiên, nếu chỉ giao diện đầu ra cụ thể được yêu cầu khi đó tình trạng VBAP 1810 có thể cung cấp ma trận kết xuất được yêu cầu cho đầu ra, ví dụ 5.1. Bộ giải mã SAOC 1800 khi đó thực hiện sự kết xuất trực tiếp từ các kênh vận chuyển SAOC, dữ liệu theo tham số được kết hợp và siêu dữ liệu được giải nén, sự kết xuất trực tiếp thành dữ liệu đầu ra thành định dạng đầu ra được yêu cầu mà không có bất kỳ tương tác của bộ trộn 1220. Tuy nhiên, khi sự trộn nhất định giữa các chế độ được áp dụng, ví dụ, trong đó một số kênh được mã hóa SAOC nhưng không phải tất cả các kênh

được mã hóa SAOC hoặc trong đó một số đối tượng được mã hóa SAOC nhưng không phải tất cả các đối tượng được mã hóa SAOC hoặc khi chỉ một lượng nhất định các đối tượng được kết xuất trước với các kênh được giải mã SAOC và các kênh còn lại không được xử lý SAOC thì bộ trộn sẽ được đặt cùng với dữ liệu từ các phần đầu vào, tức là, trực tiếp từ bộ giải mã lõi 1300, từ bộ kết xuất đối tượng 1210 và từ bộ giải mã SAOC 1800.

Trên Fig.17, bộ giải mã OAM 1400 là bộ giải mã siêu dữ liệu 110 của thiết bị 100 để tạo ra một hoặc nhiều kênh âm thanh theo một trong số các phương án được mô tả ở trên. Hơn nữa, trên Fig.17, bộ kết xuất đối tượng 1210, bộ giải mã USAC 1300 và bộ trộn 1220 cùng tạo ra bộ giải mã âm thanh 120 của thiết bị 100 để tạo ra một hoặc nhiều kênh âm thanh theo một trong số các phương án được mô tả ở trên.

Thiết bị để giải mã dữ liệu âm thanh được mã hóa được đề xuất. Thiết bị để giải mã dữ liệu âm thanh được mã hóa bao gồm:

- giao diện đầu vào 1100 để nhận dữ liệu âm thanh được mã hóa, dữ liệu âm thanh được mã hóa bao gồm nhiều kênh được mã hóa hoặc nhiều đối tượng được mã hóa hoặc nén siêu dữ liệu liên quan đến nhiều đối tượng, và
- thiết bị 100 bao gồm bộ giải mã siêu dữ liệu 110 và bộ tạo kênh âm thanh 120 để tạo ra một hoặc nhiều kênh âm thanh như được mô tả trên đây.

Bộ giải mã siêu dữ liệu 110 của thiết bị 100 để tạo ra một hoặc nhiều kênh âm thanh là bộ giải nén siêu dữ liệu 400 để giải nén siêu dữ liệu được nén.

Bộ tạo kênh âm thanh 120 của thiết bị 100 để tạo ra một hoặc nhiều kênh âm thanh bao gồm bộ giải mã lõi 1300 để giải mã nhiều kênh được mã hóa và nhiều đối tượng được mã hóa.

Hơn nữa, bộ tạo kênh âm thanh 120 còn bao gồm bộ xử lý đối tượng 1200 để xử lý nhiều đối tượng được giải mã sử dụng siêu dữ liệu được giải nén để thu được số lượng các kênh đầu ra 1205 bao gồm dữ liệu âm thanh từ các đối tượng và các kênh được giải mã.

Hơn nữa, bộ tạo kênh âm thanh 120 còn bao gồm bộ xử lý sau 1700 để chuyển đổi số lượng các kênh đầu ra 1205 thành định dạng đầu ra.

Mặc dù một số khía cạnh đã được mô tả trong phạm vi của thiết bị, rõ ràng rằng các khía cạnh này cũng thể hiện sự mô tả của phương pháp tương ứng, trong đó khối hoặc thiết bị tương ứng với bước phương pháp hoặc dấu hiệu của bước phương pháp. Tương tự, các khía cạnh được mô tả trong phạm vi của bước phương pháp cũng biểu diễn sự mô tả của khối hoặc mục hoặc dấu hiệu tương ứng của thiết bị tương ứng.

Các tín hiệu phân tích đối mới có thể được lưu trữ trên môi trường lưu trữ kỹ thuật số hoặc có thể được truyền trong môi trường truyền như môi trường truyền không dây hoặc môi trường truyền có dây như Internet.

Phụ thuộc vào các yêu cầu thực thi nhất định, các phương án của sáng chế có thể được thực thi trong phần cứng hoặc trong phần mềm. Phương án có thể được thực hiện sử dụng vật lưu trữ số, ví dụ, đĩa mềm, DVD, CD, ROM, PROM, EPROM, EEPROM hoặc bộ nhớ FLASH, có các tín hiệu điều khiển có thể đọc được bằng điện được lưu trữ trên đó, mà kết hợp (hoặc có thể kết hợp) với hệ thống máy tính khả trình sao cho phương pháp tương ứng được thực hiện.

Một số phương án theo sáng chế gồm có sóng mang dữ liệu không tạm thời có các tín hiệu điều khiển đọc được nhờ điện tử, chúng có khả năng kết hợp với hệ thống máy tính lập trình được, do đó một trong các phương pháp được mô tả ở đây được thực hiện.

Thông thường, các phương án theo sáng chế có thể được thực hiện như sản phẩm chương trình máy tính với mã chương trình, mã chương trình hoạt động để thực hiện một trong các phương pháp khi sản phẩm chương trình máy tính chạy trên máy tính. Mã chương trình có thể ví dụ được lưu trữ trên sóng mang máy đọc được.

Các phương án khác bao gồm chương trình máy tính để thực hiện một trong các phương pháp được mô tả ở đây, được lưu trữ trên vật mang có thể đọc được bằng máy.

Nói cách khác, do đó, phương án của phương pháp theo sáng chế là, chương trình máy tính có mã chương trình để thực hiện một trong số các phương pháp được mô tả ở đây, khi chương trình máy tính chạy trên máy tính.

Phương án khác của các phương pháp theo sáng chế là, sóng mang dữ liệu (hoặc môi trường lưu trữ số, hoặc môi trường máy tính đọc được) gồm có, đã được ghi lại

trên đó, chương trình máy tính để thực hiện một trong các phương pháp được mô tả ở đây.

Phương án khác của các phương pháp theo sáng chế là, dòng dữ liệu hoặc chuỗi tín hiệu thể hiện chương trình máy tính để thực hiện một trong các phương pháp đã được mô tả ở đây. Ví dụ, có thể cấu hình dòng dữ liệu hoặc chuỗi tín hiệu để được truyền thông qua sự kết nối truyền thông dữ liệu, ví dụ thông qua Liên mạng.

Phương án khác gồm có phương tiện xử lý, ví dụ máy tính, hoặc thiết bị logic lập trình được, được tạo cấu hình để hoặc được làm thích ứng để thực hiện một trong số các phương pháp được mô tả ở đây.

Phương án nữa bao gồm máy tính có chương trình máy tính được cài đặt trên máy tính để thực hiện một trong số các phương pháp được mô tả ở đây.

Theo một số phương án, thiết bị logic lập trình được (ví dụ, mảng cổng lập trình được dạng trường) có thể được sử dụng để thực hiện một số hoặc tất cả các chức năng của các phương pháp được mô tả ở đây. Trong một số phương án, mảng cổng lập trình được dạng trường có thể kết hợp với bộ vi xử lý để thực hiện một trong số các phương pháp được mô tả ở đây. Thông thường, các phương pháp ưu tiên được thực hiện bởi thiết bị phần cứng bất kỳ.

Các phương án được mô tả bên trên chỉ mang tính minh họa cho các nguyên lý của sáng chế. Cần hiểu rằng các sự biến thể và biến đổi của các phương án và các chi tiết được mô tả ở đây sẽ là rõ ràng đối với người có trình độ trong lĩnh vực kỹ thuật tương ứng. Do đó, mục đích chỉ bị giới hạn bởi phạm vi của các điểm yêu cầu bảo hộ sắp đưa ra dưới đây và không bị giới hạn bởi các chi tiết cụ thể được biểu diễn bằng cách mô tả và giải thích của các phương án ở đây.

YÊU CẦU BẢO HỘ

1. Thiết bị tạo ra một hoặc nhiều kênh âm thanh (100), trong đó thiết bị bao gồm:

bộ giải mã siêu dữ liệu (110) để nhận một hoặc nhiều tín hiệu siêu dữ liệu được nén, trong đó mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhiều mẫu siêu dữ liệu thứ nhất, trong đó các mẫu siêu dữ liệu thứ nhất của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén biểu thị thông tin được kết hợp với tín hiệu đối tượng âm thanh trong số một hoặc nhiều tín hiệu đối tượng âm thanh, trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục, để mỗi tín hiệu siêu dữ liệu được khôi phục trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén, tín hiệu siêu dữ liệu được khôi phục đã nêu được kết hợp với tín hiệu siêu dữ liệu được nén đã nêu, và còn bao gồm nhiều mẫu siêu dữ liệu thứ hai, trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để tạo ra các mẫu siêu dữ liệu thứ hai của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bằng cách tạo ra nhiều mẫu siêu dữ liệu được làm gần đúng cho tín hiệu siêu dữ liệu đã nêu, trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để tạo ra mỗi mẫu trong số nhiều mẫu siêu dữ liệu được làm gần đúng phụ thuộc vào ít nhất hai mẫu trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu, và

bộ tạo kênh âm thanh (120) để tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục,

trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để nhận nhiều giá trị chênh lệch cho tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén, và được tạo cấu hình để cộng mỗi giá trị trong số nhiều giá trị chênh lệch vào một mẫu trong số các mẫu siêu dữ liệu được làm gần đúng của tín hiệu siêu dữ liệu được khôi phục được kết hợp với tín hiệu siêu dữ liệu được nén đã nêu để thu được các mẫu siêu dữ liệu thứ hai của tín hiệu siêu dữ liệu được khôi phục đã nêu.

2 Thiết bị (100) theo điểm 1, trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu

hình để tạo ra mỗi tín hiệu siêu dữ liệu được khôi phục trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bằng cách lấy mẫu tăng một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén, trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để tạo ra mỗi mẫu trong số các mẫu siêu dữ liệu thứ hai trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bằng cách tiến hành phép nội suy tuyến tính phụ thuộc vào ít nhất hai mẫu trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu.

3 Thiết bị (100) theo điểm 1 hoặc 2,

trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để nhận nhiều giá trị chênh lệch cho tín hiệu siêu dữ liệu được nén của một hoặc nhiều tín hiệu siêu dữ liệu được nén, trong đó mỗi giá trị trong số các giá trị chênh lệch là giá trị chênh lệch được nhận được gán cho một mẫu trong số các mẫu siêu dữ liệu được làm gần đúng của tín hiệu siêu dữ liệu được khôi phục được kết hợp với tín hiệu siêu dữ liệu được nén đã nêu,

trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để cộng mỗi giá trị chênh lệch được nhận trong số nhiều giá trị chênh lệch được nhận vào mẫu siêu dữ liệu được làm gần đúng được kết hợp với giá trị chênh lệch được nhận đã nêu để thu được một mẫu trong số các mẫu siêu dữ liệu thứ hai của tín hiệu siêu dữ liệu được khôi phục đã nêu,

trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để xác định giá trị chênh lệch được làm gần đúng phụ thuộc vào một hoặc nhiều giá trị trong số nhiều giá trị chênh lệch được nhận cho mỗi mẫu siêu dữ liệu được làm gần đúng trong số nhiều mẫu siêu dữ liệu được làm gần đúng của tín hiệu siêu dữ liệu được khôi phục được kết hợp với tín hiệu siêu dữ liệu được nén đã nêu, khi không có giá trị nào trong số nhiều giá trị chênh lệch được nhận được kết hợp với mẫu siêu dữ liệu được làm gần đúng đã nêu,

trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để cộng mỗi giá trị chênh lệch được làm gần đúng trong số nhiều giá trị chênh lệch được làm gần đúng vào mẫu siêu dữ liệu được làm gần đúng của giá trị chênh lệch được làm gần đúng để thu được một mẫu khác trong số các mẫu siêu dữ liệu thứ hai của tín hiệu siêu dữ liệu

được khôi phục đã nêu.

4. Thiết bị (100) theo một trong số các điểm nêu trên,

trong đó ít nhất một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm thông tin vị trí về một tín hiệu trong số một hoặc nhiều tín hiệu đối tượng âm thanh, hoặc bao gồm phép biểu diễn được định tỉ lệ của thông tin vị trí về một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu đối tượng âm thanh, và

trong đó bộ tạo kênh âm thanh (120) được tạo cấu hình để tạo ra ít nhất một kênh trong số một hoặc nhiều kênh âm thanh phụ thuộc vào một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào thông tin vị trí đã nêu.

5. Thiết bị (100) theo một trong số các điểm nêu trên,

trong đó ít nhất một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm âm lượng của một tín hiệu trong số một hoặc nhiều tín hiệu đối tượng âm thanh, hoặc bao gồm phép biểu diễn được định tỉ lệ của âm lượng của một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu đối tượng âm thanh, và

trong đó bộ tạo kênh âm thanh (120) được tạo cấu hình để tạo ra ít nhất một kênh trong số một hoặc nhiều kênh âm thanh phụ thuộc vào một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào âm lượng đã nêu.

6. Thiết bị (100) theo một trong số các điểm nêu trên, trong đó thiết bị (100) được tạo cấu hình để nhận thông tin truy cập ngẫu nhiên, trong đó, đối với mỗi tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén, thông tin truy cập ngẫu nhiên biểu thị phần tín hiệu được truy cập của tín hiệu siêu dữ liệu được nén đã nêu, trong đó ít nhất một phần tín hiệu khác của tín hiệu siêu dữ liệu đã nêu không được biểu thị bởi thông tin truy cập ngẫu nhiên, và trong đó bộ giải mã siêu dữ liệu (110) được tạo cấu hình để tạo ra một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục phụ thuộc vào các mẫu siêu dữ liệu thứ nhất của phần tín hiệu được truy cập đã nêu của tín hiệu siêu dữ liệu được nén đã nêu, nhưng không phụ thuộc vào bất kì mẫu siêu dữ liệu thứ nhất nào khác của bất kì phần tín hiệu khác của tín hiệu siêu dữ liệu được nén đã nêu.

7. Thiết bị tạo ra thông tin âm thanh được mã hóa (250) bao gồm một hoặc nhiều

tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén, trong đó thiết bị bao gồm:

bộ mã hóa siêu dữ liệu (210) để nhận một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, trong đó mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu bao gồm nhiều mẫu siêu dữ liệu, trong đó các mẫu siêu dữ liệu của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu biểu thị thông tin được kết hợp với tín hiệu đối tượng âm thanh trong số một hoặc nhiều tín hiệu đối tượng âm thanh, trong đó bộ mã hóa (210) được tạo cấu hình để tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được nén, để mỗi tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhóm thứ nhất gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu của tín hiệu siêu dữ liệu ban đầu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, tín hiệu siêu dữ liệu được nén đã nêu được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu, và để tín hiệu siêu dữ liệu được nén đã nêu không bao gồm bất kì mẫu siêu dữ liệu trong số nhóm thứ hai gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu khác của một tín hiệu đã nêu trong số các tín hiệu siêu dữ liệu ban đầu, và

bộ mã hóa âm thanh (220) để mã hóa một hoặc nhiều tín hiệu đối tượng âm thanh để thu được một hoặc nhiều tín hiệu âm thanh được mã hóa,

trong đó mỗi mẫu trong số các mẫu siêu dữ liệu mà được bao gồm bởi tín hiệu siêu dữ liệu ban đầu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu và nó cũng được bao gồm bởi tín hiệu siêu dữ liệu được nén, mà được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu, là một mẫu trong số nhiều mẫu siêu dữ liệu thứ nhất,

trong đó mỗi mẫu trong số các mẫu siêu dữ liệu mà được bao gồm bởi tín hiệu siêu dữ liệu ban đầu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu và nó không được bao gồm bởi tín hiệu siêu dữ liệu được nén, mà được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu, là một mẫu trong số nhiều mẫu siêu dữ liệu thứ hai,

trong đó bộ mã hóa siêu dữ liệu (210) được tạo cấu hình để tạo ra mẫu siêu dữ liệu được làm gần đúng cho mỗi mẫu trong số nhiều mẫu siêu dữ liệu thứ hai của một trong số các tín hiệu siêu dữ liệu ban đầu bằng cách tiến hành phép nội suy tuyến tính phụ thuộc vào ít nhất hai mẫu trong số các mẫu siêu dữ liệu thứ nhất của một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, và

trong đó bộ mã hóa siêu dữ liệu (210) được tạo cấu hình để tạo ra giá trị chênh lệch cho mỗi mẫu siêu dữ liệu thứ hai trong số nhiều mẫu siêu dữ liệu thứ hai đã nêu của một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, để giá trị chênh lệch đã nêu biểu thị sự chênh lệch giữa mẫu siêu dữ liệu thứ hai đã nêu và mẫu siêu dữ liệu được làm gần đúng của mẫu siêu dữ liệu thứ hai đã nêu.

8. Thiết bị (250) theo điểm 7,

trong đó bộ mã hóa siêu dữ liệu (210) được tạo cấu hình để xác định ít nhất một giá trị trong số các giá trị chênh lệch của nhiều mẫu siêu dữ liệu thứ hai đã nêu của một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, liệu mỗi giá trị trong số ít nhất một trong số giá trị chênh lệch đã nêu lớn hơn giá trị ngưỡng.

9. Thiết bị (250) theo điểm 7 hoặc 8,

trong đó bộ mã hóa siêu dữ liệu (210) được tạo cấu hình để mã hóa một hoặc nhiều mẫu trong số các mẫu siêu dữ liệu của một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén với số lượng các bit thứ nhất, trong đó mỗi mẫu đã nêu trong số một hoặc nhiều mẫu siêu dữ liệu đã nêu của một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén biểu thị số nguyên,

trong đó bộ mã hóa siêu dữ liệu (210) được tạo cấu hình để mã hóa một hoặc nhiều giá trị chênh lệch của nhiều mẫu siêu dữ liệu thứ hai đã nêu với số lượng các bit thứ hai, trong đó mỗi giá trị trong số một hoặc nhiều giá trị chênh lệch đã nêu của nhiều mẫu siêu dữ liệu thứ hai biểu thị số nguyên, và

trong đó số lượng các bit thứ hai là nhỏ hơn số lượng các bit thứ nhất.

10. Thiết bị (250) theo một trong số các điểm từ 7 đến 9,

trong đó ít nhất một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu bao gồm thông tin vị trí về một tín hiệu trong số một hoặc nhiều tín hiệu đối tượng âm thanh, hoặc bao gồm phép biểu diễn được định tỉ lệ của thông tin vị trí về một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu đối tượng âm thanh, và

trong đó bộ mã hóa siêu dữ liệu (210) được tạo cấu hình để tạo ra ít nhất một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén phụ thuộc vào ít nhất một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu.

11. Thiết bị (250) theo một trong số các điểm từ 7 đến 10,

trong đó ít nhất một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu bao gồm âm lượng của một tín hiệu trong số một hoặc nhiều tín hiệu đối tượng âm thanh, hoặc bao gồm phép biểu diễn được định tỉ lệ của âm lượng của một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu đối tượng âm thanh, và

trong đó bộ mã hóa siêu dữ liệu (210) được tạo cấu hình để tạo ra ít nhất một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén phụ thuộc vào ít nhất một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu.

12. Hệ thống tạo ra thông tin âm thanh được mã hóa và nhận một hoặc nhiều tín hiệu âm thanh được mã hóa, hệ thống bao gồm:

thiết bị (250) theo một trong số các điểm từ 7 đến 11 tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén, và

thiết bị (100) theo một trong số các điểm từ 1 đến 6 để nhận một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén, và để tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu âm thanh được mã hóa và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được nén.

13. Phương pháp tạo ra một hoặc nhiều kênh âm thanh, trong đó phương pháp bao gồm:

nhận một hoặc nhiều tín hiệu siêu dữ liệu được nén, trong đó mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhiều mẫu siêu dữ liệu thứ nhất, trong đó các mẫu siêu dữ liệu thứ nhất của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén biểu thị thông tin được kết hợp với tín hiệu đối tượng âm thanh của một hoặc nhiều tín hiệu đối tượng âm thanh,

tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục, để mỗi tín hiệu siêu dữ liệu được khôi phục trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén, tín hiệu siêu dữ liệu được khôi phục đã nêu được kết hợp với tín hiệu siêu dữ liệu được nén đã nêu, và còn bao gồm nhiều mẫu

siêu dữ liệu thứ hai, trong đó tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bao gồm tạo ra các mẫu siêu dữ liệu thứ hai của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục bằng cách tạo ra nhiều mẫu siêu dữ liệu được làm gần đúng cho tín hiệu siêu dữ liệu được khôi phục đã nêu, trong đó tạo ra mỗi mẫu trong số nhiều mẫu siêu dữ liệu được làm gần đúng được thực hiện phụ thuộc vào ít nhất hai trong số các mẫu siêu dữ liệu thứ nhất của tín hiệu siêu dữ liệu được khôi phục đã nêu, và

tạo ra một hoặc nhiều kênh âm thanh phụ thuộc vào một hoặc nhiều tín hiệu đối tượng âm thanh và phụ thuộc vào một hoặc nhiều tín hiệu siêu dữ liệu được khôi phục,

trong đó phương pháp còn bao gồm nhận nhiều giá trị chênh lệch cho tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén, và cộng mỗi giá trị trong số nhiều giá trị chênh lệch vào một mẫu trong số các mẫu siêu dữ liệu được làm gần đúng của tín hiệu siêu dữ liệu được khôi phục được kết hợp với tín hiệu siêu dữ liệu được nén đã nêu để thu được các mẫu siêu dữ liệu thứ hai của tín hiệu siêu dữ liệu được khôi phục đã nêu.

14. Phương pháp tạo ra thông tin âm thanh được mã hóa bao gồm một hoặc nhiều tín hiệu âm thanh được mã hóa và một hoặc nhiều tín hiệu siêu dữ liệu được nén, trong đó phương pháp bao gồm:

nhận một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, trong đó mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu bao gồm nhiều mẫu siêu dữ liệu, trong đó các mẫu siêu dữ liệu của mỗi tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu biểu thị thông tin được kết hợp với tín hiệu đối tượng âm thanh của một hoặc nhiều tín hiệu đối tượng âm thanh,

tạo ra một hoặc nhiều tín hiệu siêu dữ liệu được nén, để mỗi tín hiệu siêu dữ liệu được nén trong số một hoặc nhiều tín hiệu siêu dữ liệu được nén bao gồm nhóm thứ nhất gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu của tín hiệu siêu dữ liệu ban đầu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, tín hiệu siêu dữ liệu được nén đã nêu được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu, và để tín hiệu siêu dữ liệu được nén đã nêu không bao gồm bất kì mẫu siêu dữ liệu của nhóm thứ hai gồm hai hoặc nhiều hơn hai mẫu siêu dữ liệu khác của một trong số các tín hiệu siêu dữ liệu

ban đầu đã nêu, và

mã hóa một hoặc nhiều tín hiệu đối tượng âm thanh để thu được một hoặc nhiều tín hiệu âm thanh được mã hóa,

trong đó mỗi mẫu trong số các mẫu siêu dữ liệu mà được bao gồm bởi tín hiệu siêu dữ liệu ban đầu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu và nó cũng được bao gồm bởi tín hiệu siêu dữ liệu được nén, mà được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu, là một mẫu trong số nhiều mẫu siêu dữ liệu thứ nhất,

trong đó mỗi mẫu trong số các mẫu siêu dữ liệu mà được bao gồm bởi tín hiệu siêu dữ liệu ban đầu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu và nó không được bao gồm bởi tín hiệu siêu dữ liệu được nén, mà được kết hợp với tín hiệu siêu dữ liệu ban đầu đã nêu, là một mẫu trong số nhiều mẫu siêu dữ liệu thứ hai,

trong đó phương pháp còn bao gồm tạo ra mẫu siêu dữ liệu được làm gần đúng cho mỗi mẫu trong số nhiều mẫu siêu dữ liệu thứ hai của một trong số các tín hiệu siêu dữ liệu ban đầu bằng cách tiến hành phép nội suy tuyến tính phụ thuộc vào ít nhất hai mẫu trong số các mẫu siêu dữ liệu thứ nhất của một tín hiệu đã nêu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, và

trong đó phương pháp còn bao gồm tạo ra giá trị chênh lệch cho mỗi mẫu siêu dữ liệu thứ hai trong số nhiều mẫu siêu dữ liệu thứ hai đã nêu của một tín hiệu trong số một hoặc nhiều tín hiệu siêu dữ liệu ban đầu, để giá trị chênh lệch đã nêu biểu thị sự chênh lệch giữa mẫu siêu dữ liệu thứ hai đã nêu và mẫu siêu dữ liệu được làm gần đúng của mẫu siêu dữ liệu thứ hai đã nêu.

15. Vật ghi có thể đọc được bằng máy tính bao gồm chương trình máy tính để thực hiện phương pháp theo điểm 13 khi chạy trên máy tính hoặc bộ xử lý tín hiệu.

16. Vật ghi có thể đọc được bằng máy tính bao gồm chương trình máy tính để thực hiện phương pháp theo điểm 14 khi chạy trên máy tính hoặc bộ xử lý tín hiệu.

17. Thiết bị mã hóa dữ liệu đầu vào âm thanh (101) để thu được dữ liệu đầu ra âm thanh (501), thiết bị bao gồm:

giao diện đầu vào (1100) để nhận nhiều kênh âm thanh, nhiều đối tượng âm thanh và siêu dữ liệu liên quan đến một hoặc nhiều đối tượng trong số nhiều đối tượng

âm thanh,

bộ trộn (200) để trộn nhiều đối tượng và nhiều kênh để thu được nhiều kênh được trộn trước, mỗi kênh được trộn trước bao gồm dữ liệu âm thanh của kênh và dữ liệu âm thanh của ít nhất một đối tượng, và

thiết bị (250) theo một trong số các điểm từ 7 đến 11,

trong đó bộ mã hóa âm thanh (220) của thiết bị (250) theo một trong số các điểm từ 7 đến 11 là bộ mã hóa lõi (300) để mã hóa lõi dữ liệu đầu vào bộ mã hóa lõi, và

trong đó bộ mã hóa siêu dữ liệu (210) của thiết bị (250) theo một trong số các điểm từ 7 đến 11 là bộ nén siêu dữ liệu (400) để nén siêu dữ liệu liên quan đến một hoặc nhiều đối tượng trong số nhiều đối tượng âm thanh.

18. Thiết bị giải mã dữ liệu âm thanh được mã hóa, thiết bị bao gồm:

giao diện đầu vào (1100) để nhận dữ liệu âm thanh được mã hóa, dữ liệu âm thanh được mã hóa bao gồm nhiều kênh được mã hóa hoặc nhiều đối tượng được mã hóa hoặc nén siêu dữ liệu liên quan đến nhiều đối tượng, và

thiết bị (100) theo một trong số các điểm từ 1 đến 6,

trong đó bộ giải mã siêu dữ liệu (110) của thiết bị (100) theo một trong số các điểm từ 1 đến 6 là bộ giải nén siêu dữ liệu (400) để giải nén siêu dữ liệu được nén,

trong đó bộ tạo kênh âm thanh (120) của thiết bị (100) theo một trong số các điểm từ 1 đến 6 bao gồm bộ giải mã lõi (1300) để giải mã nhiều kênh được mã hóa và nhiều đối tượng được mã hóa,

trong đó bộ tạo kênh âm thanh (120) còn bao gồm bộ xử lý đối tượng (1200) để xử lý nhiều đối tượng được giải mã sử dụng siêu dữ liệu được giải nén để thu được số lượng các kênh đầu ra (1205) bao gồm dữ liệu âm thanh từ các đối tượng và các kênh được giải mã,

trong đó bộ tạo kênh âm thanh (120) còn bao gồm bộ xử lý sau (1700) để chuyển đổi số lượng các kênh đầu ra (1205) thành định dạng đầu ra.

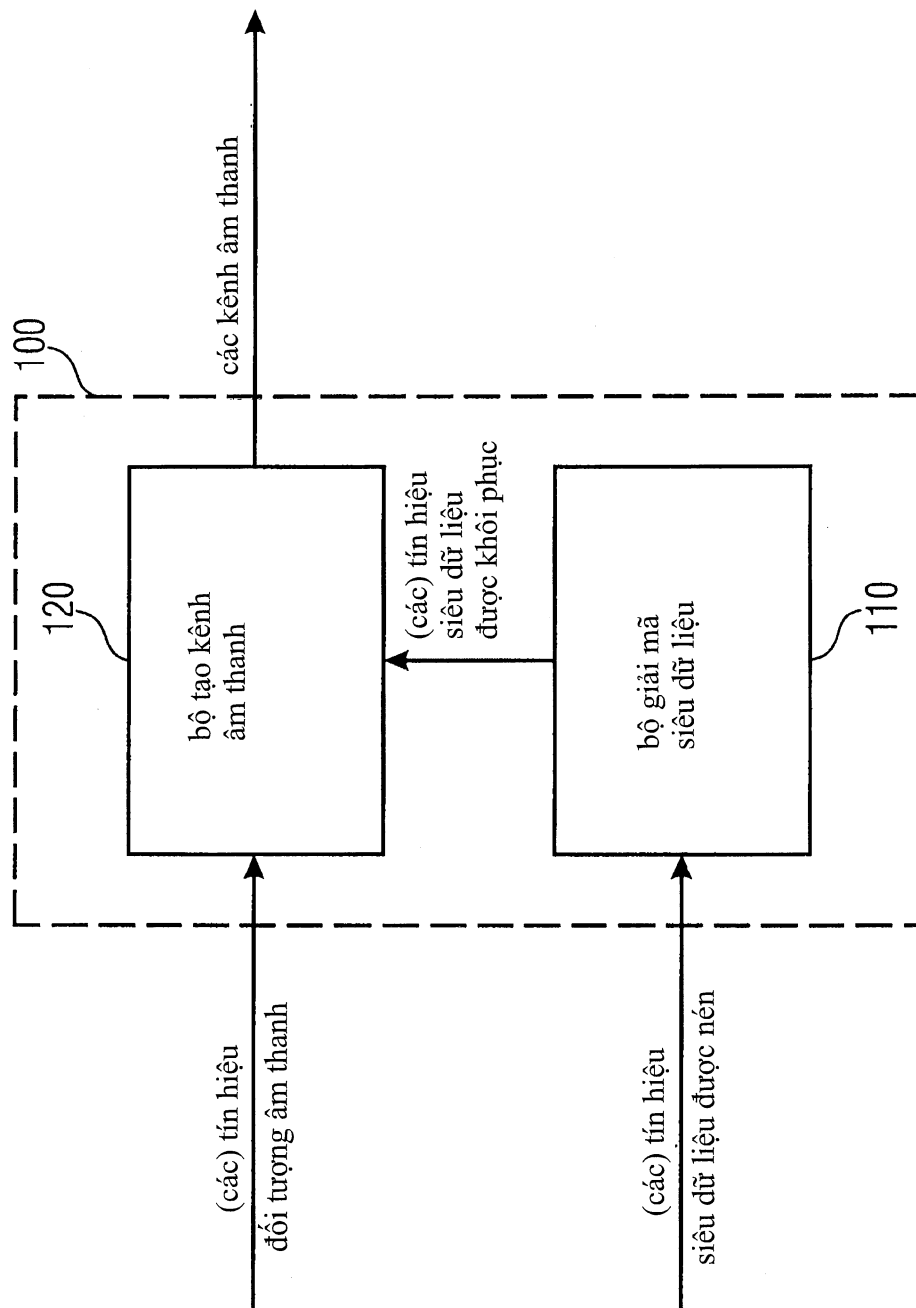


FIGURE 1

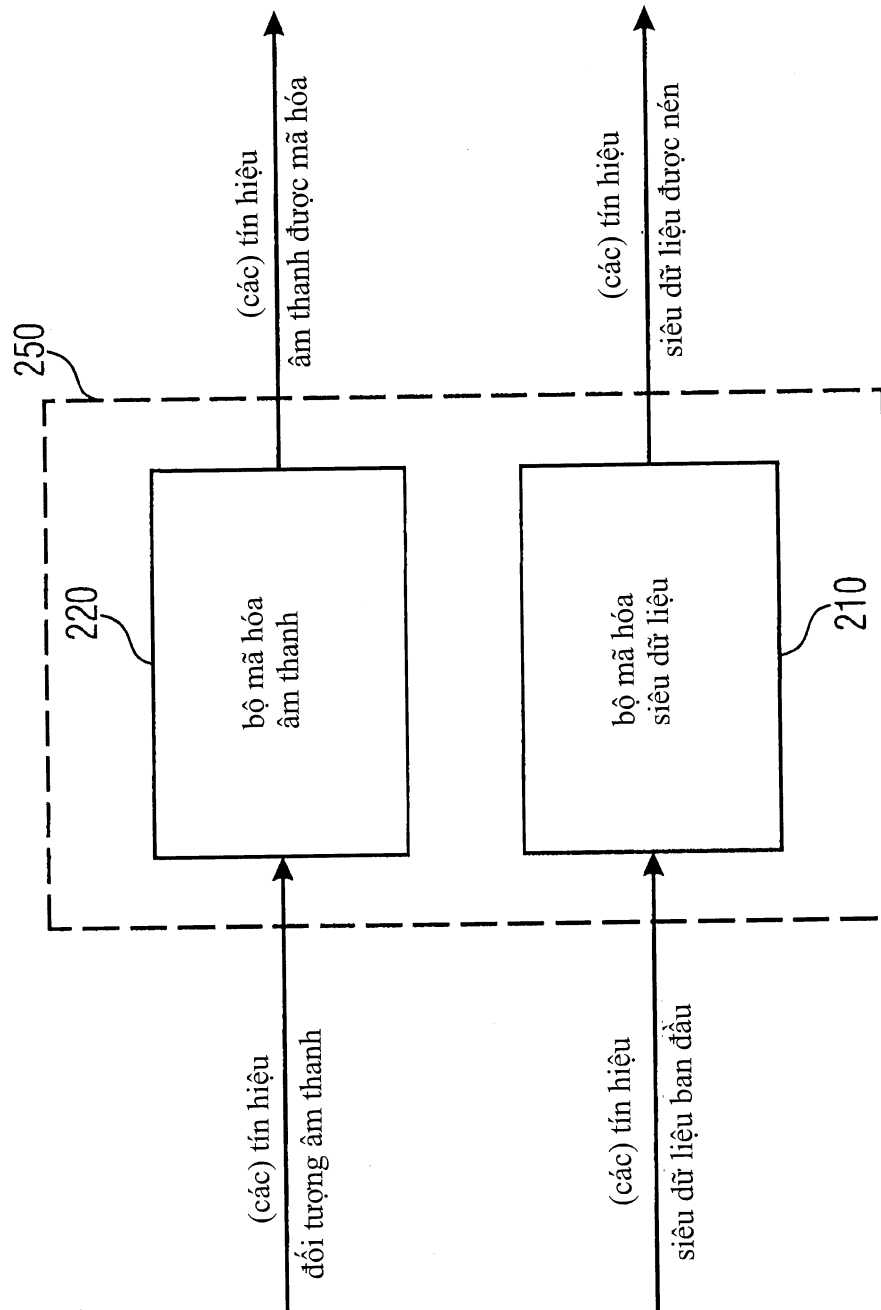


FIGURE 2

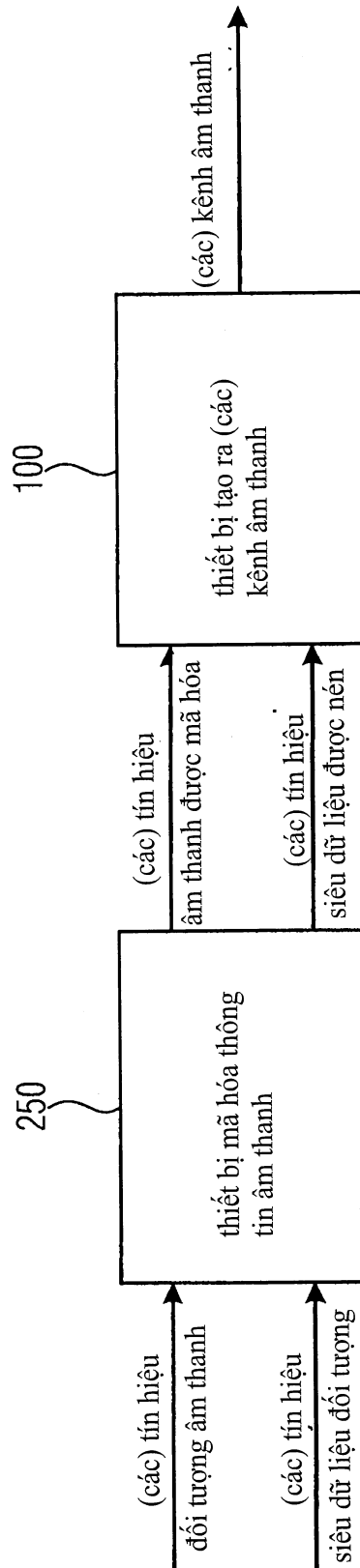


FIGURE 3

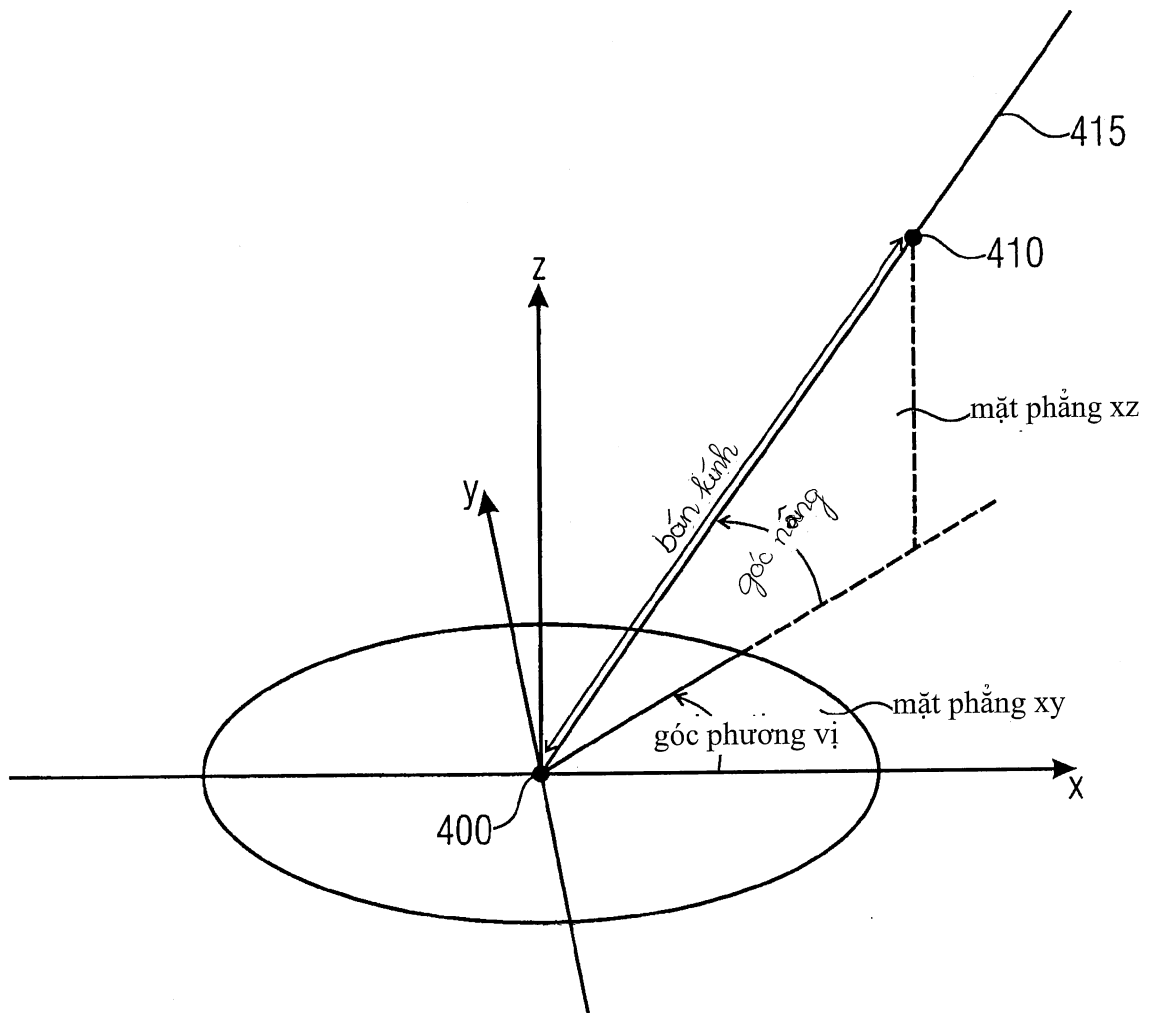


FIGURE 4

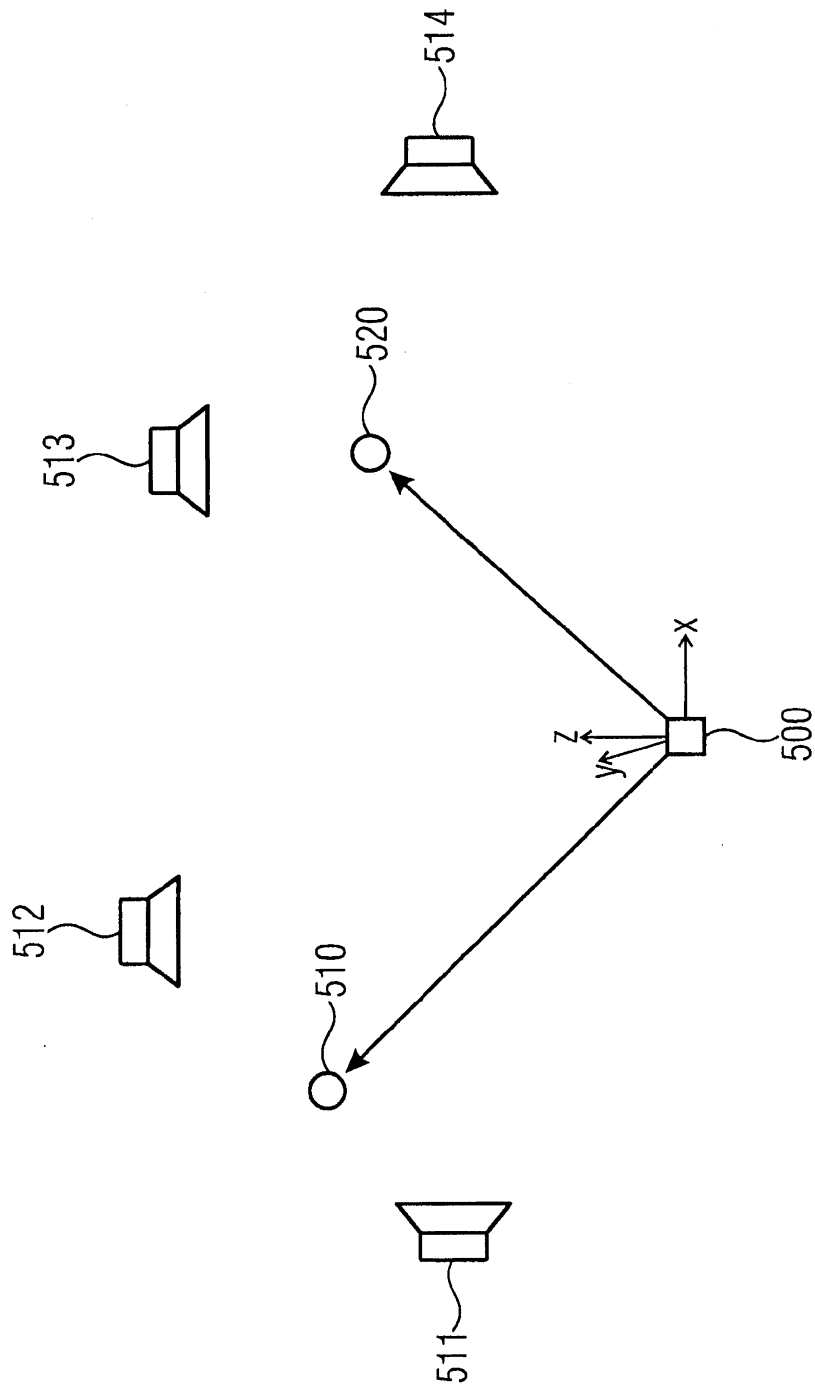


FIGURE 5

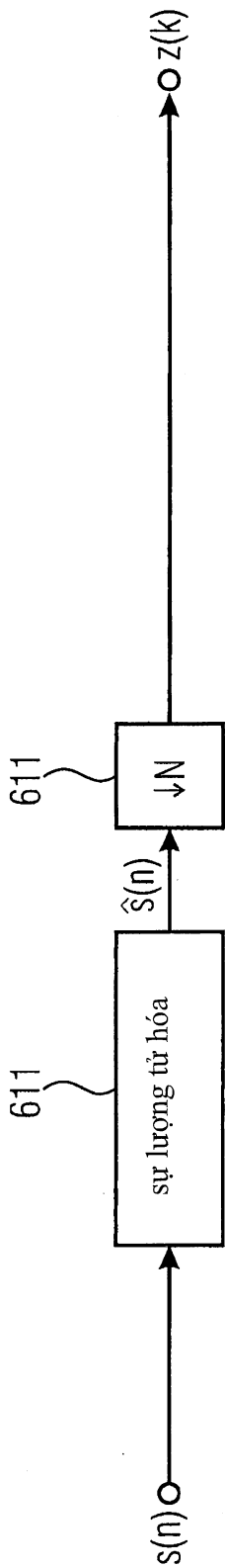


FIGURE 6

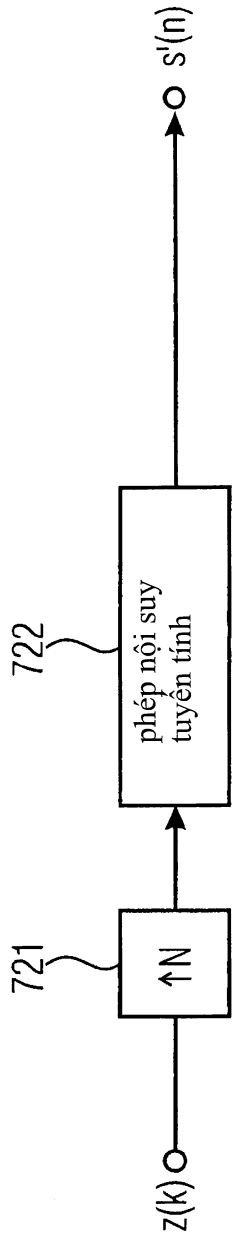


FIGURE 7

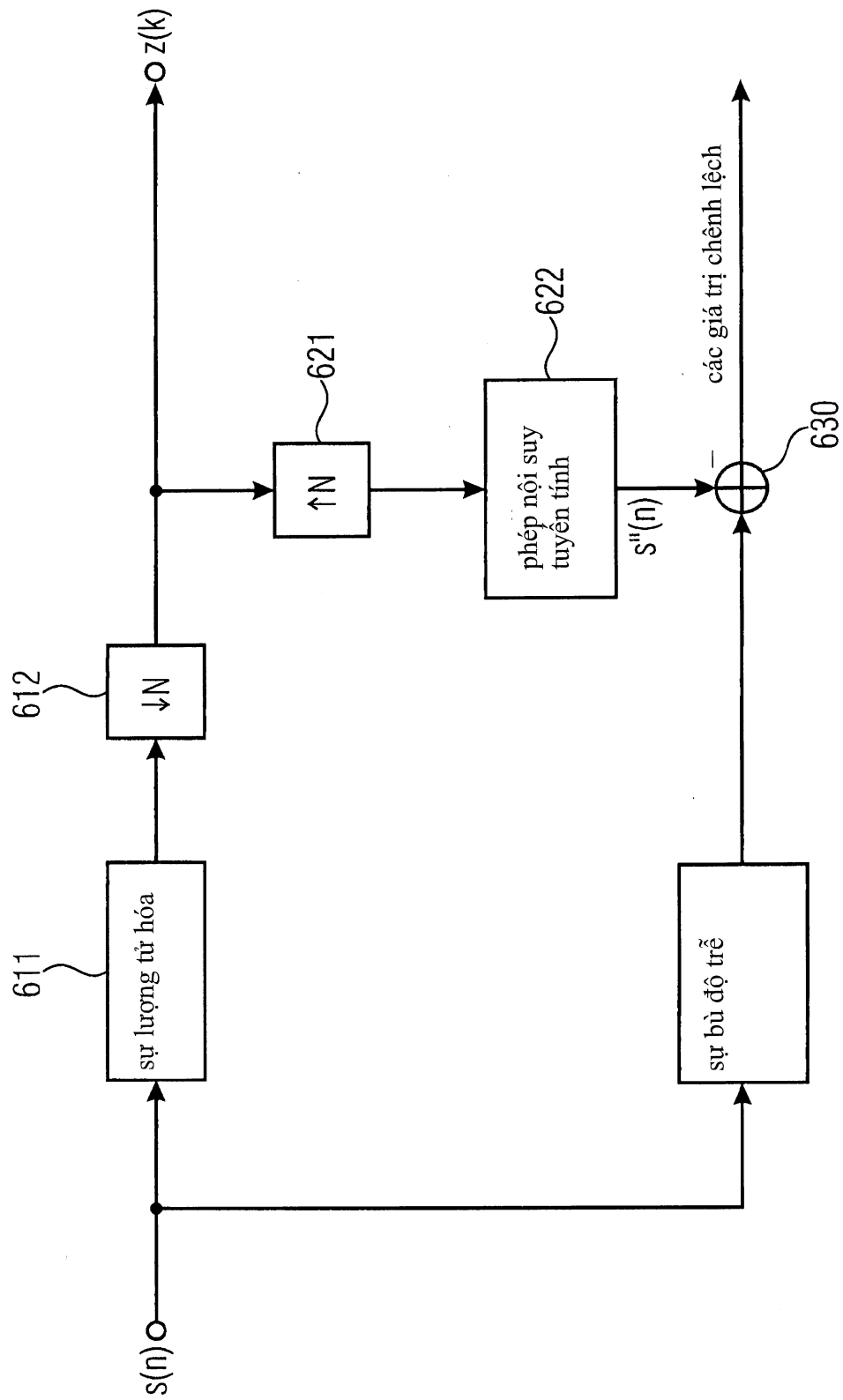


FIGURE 8

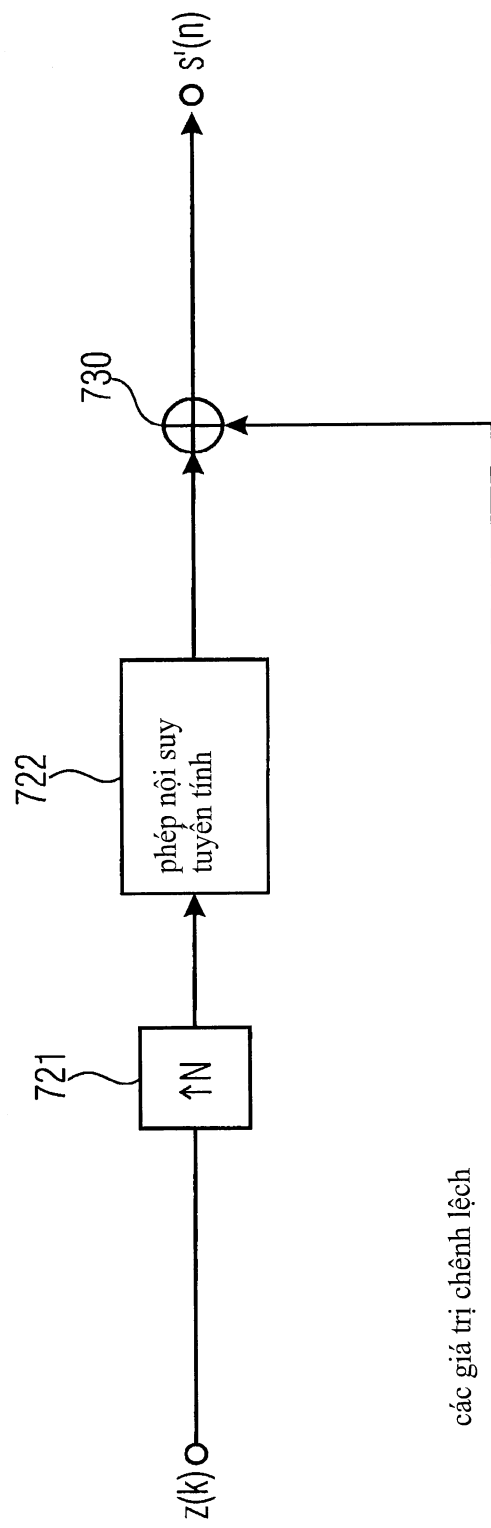


FIGURE 9

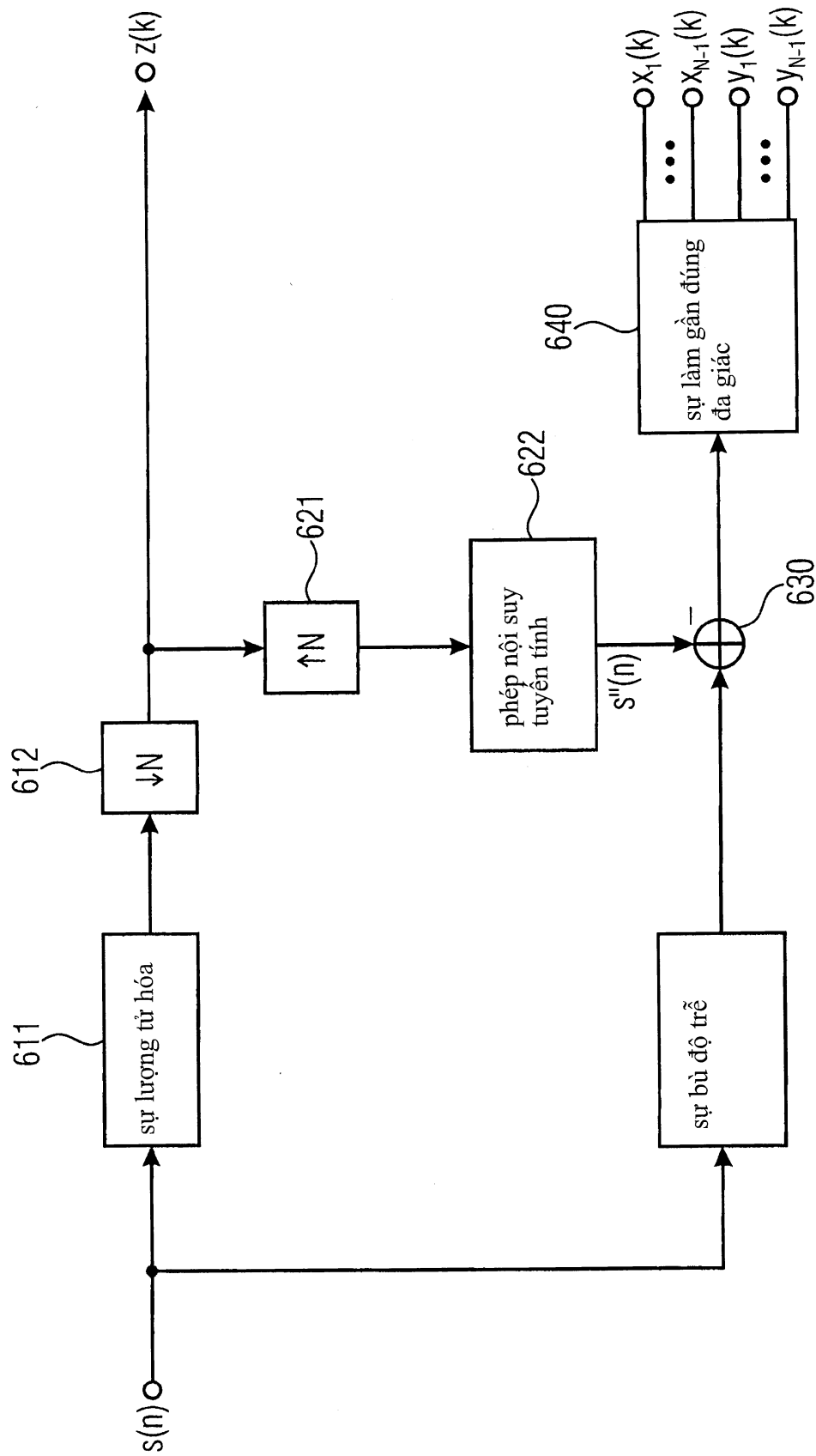


FIGURE 10

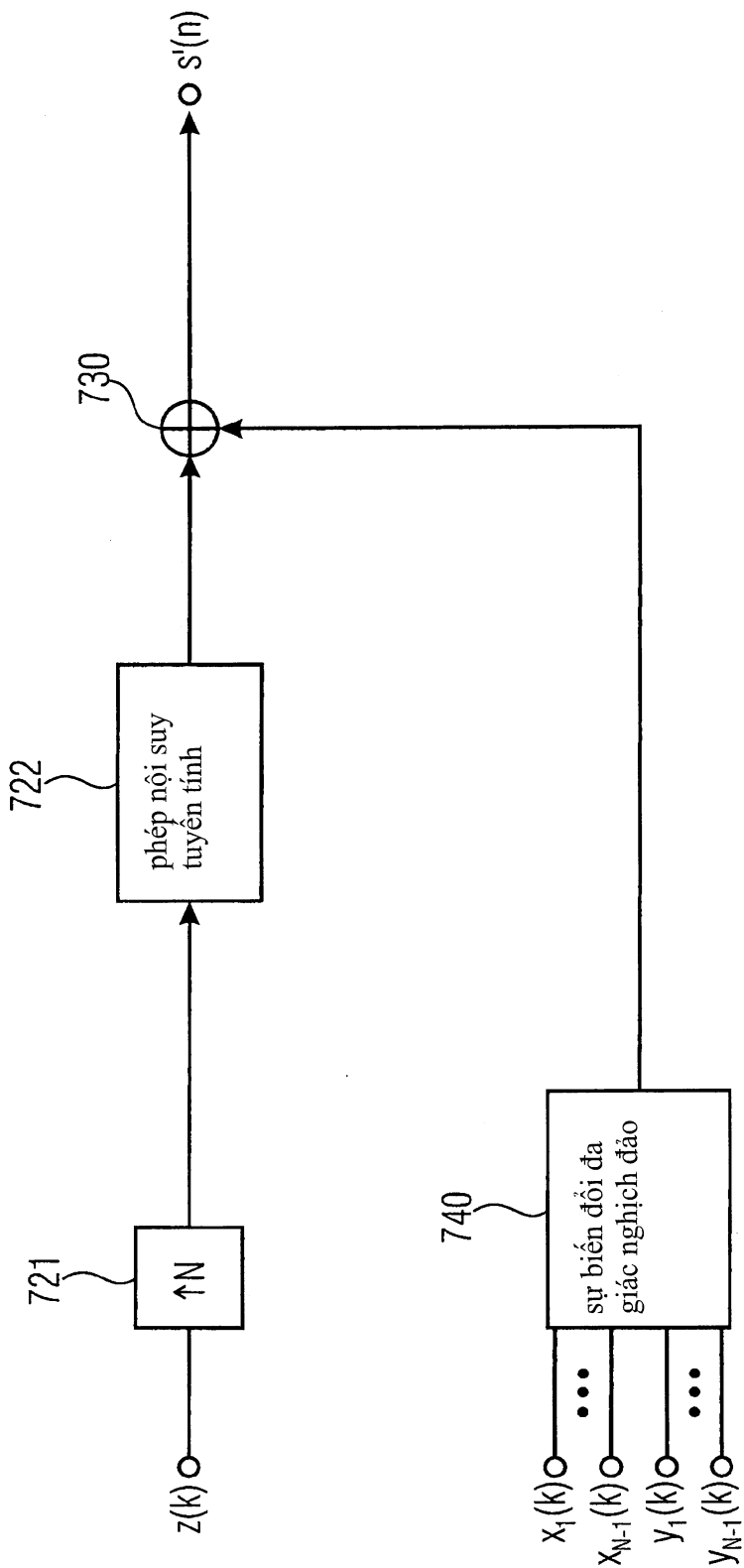
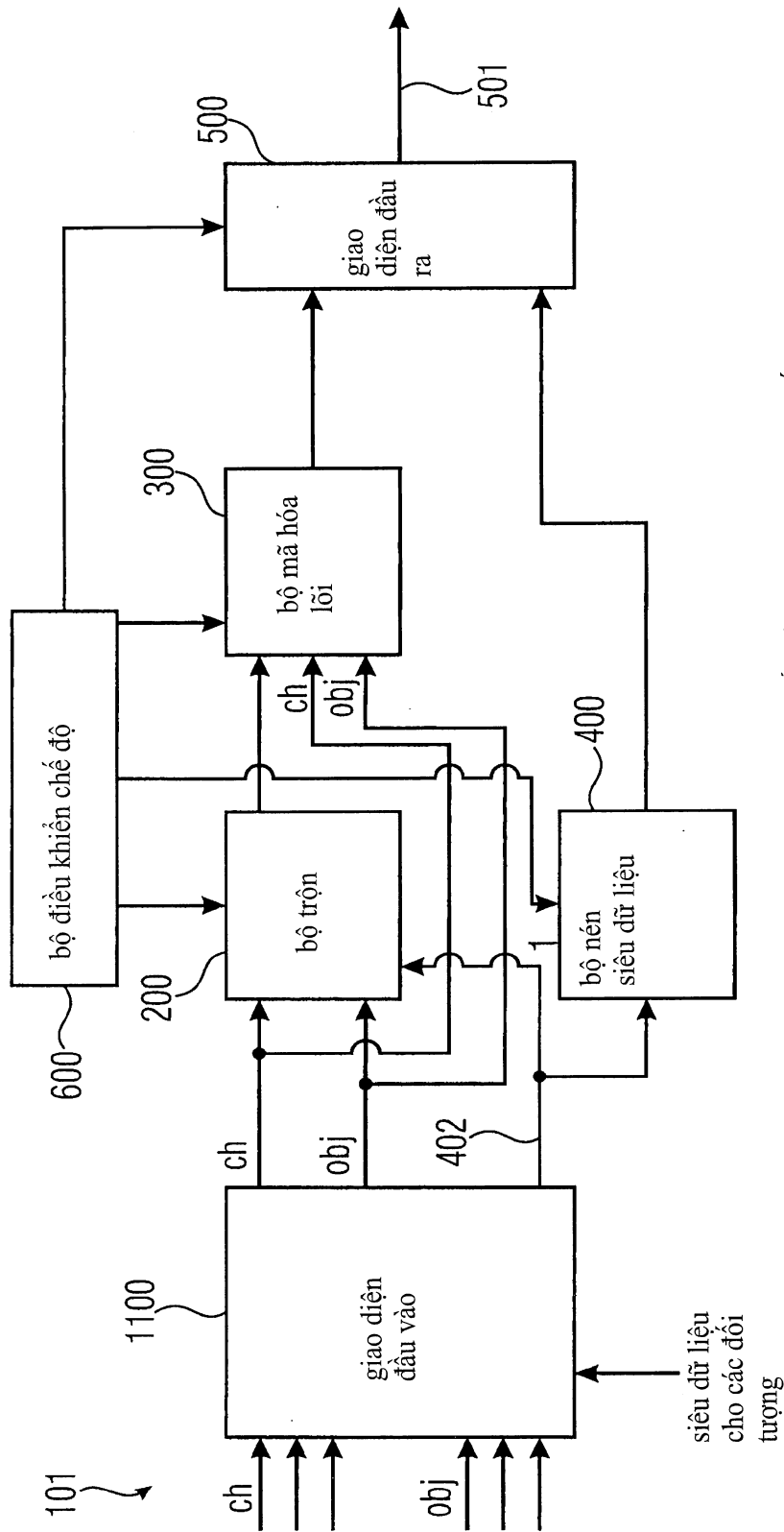


FIGURE 11



CHẾ ĐỘ 1: sự mã hóa kênh/đối tượng riêng
CHẾ ĐỘ 2: trộn các kênh và các đối tượng được kết xuất

FIGURE 12
(BỘ MÃ HÓA)

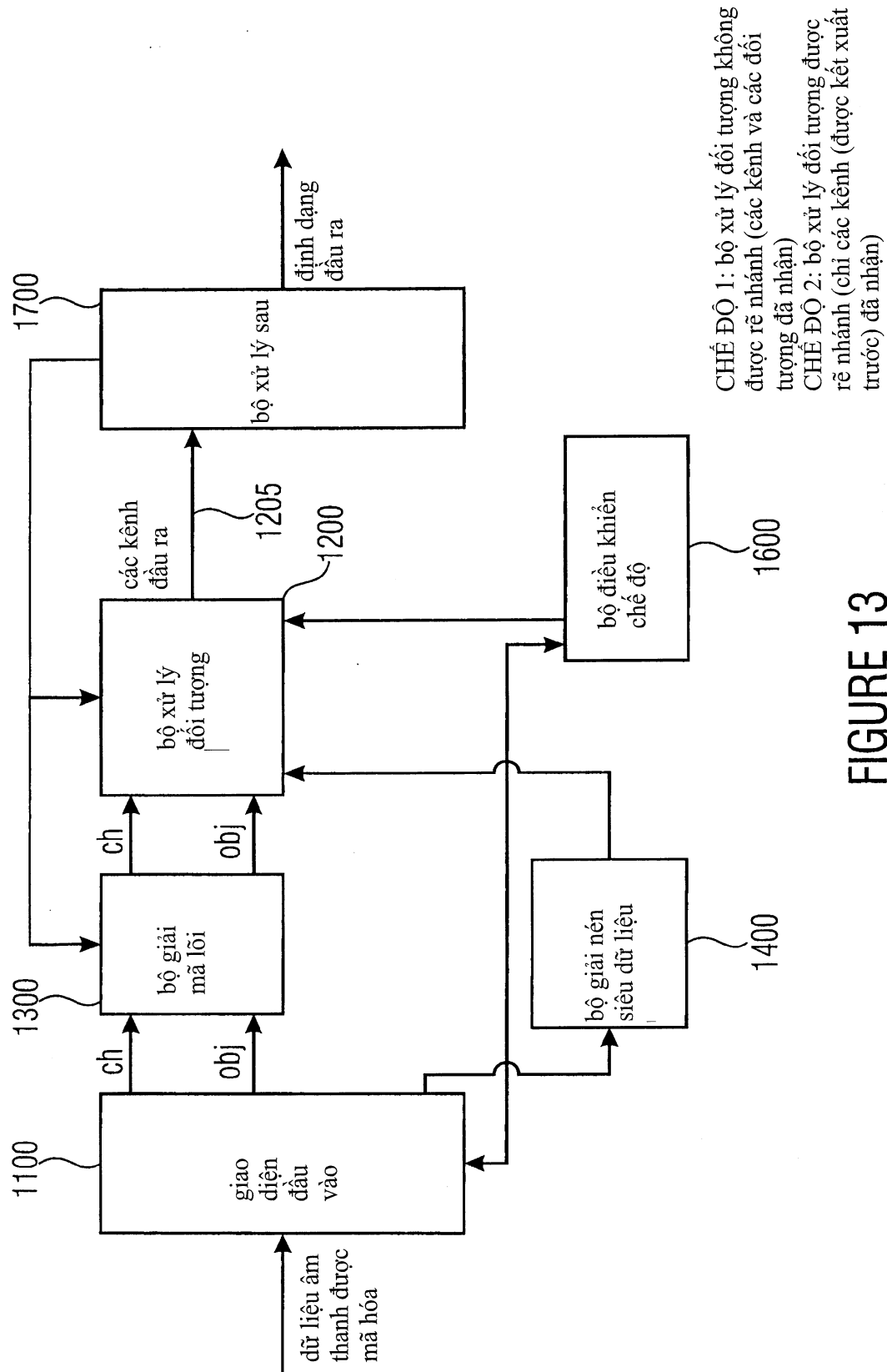


FIGURE 13

(BỘ GIẢI MÃ)

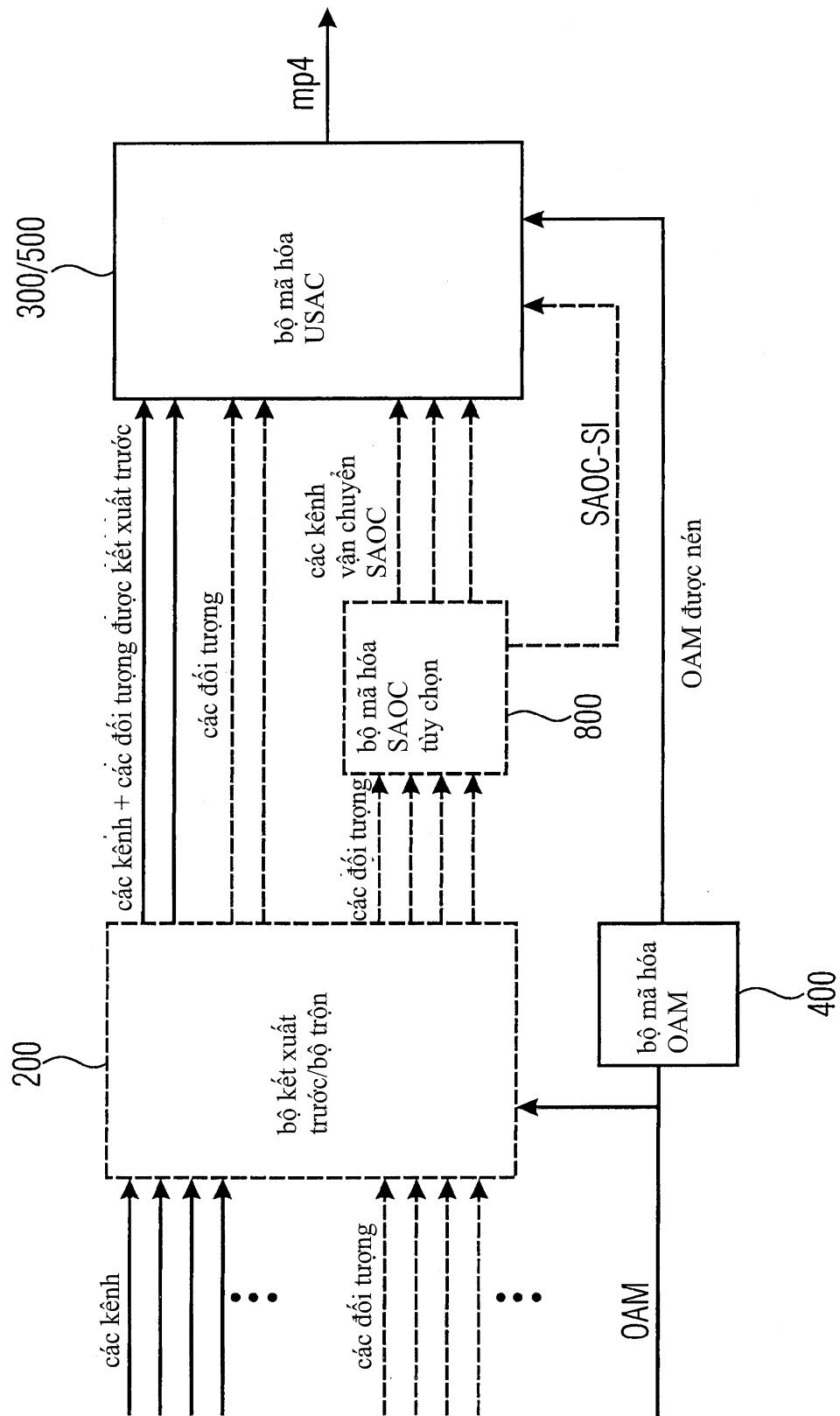


FIGURE 14
(BỘ MÃ HÓA)

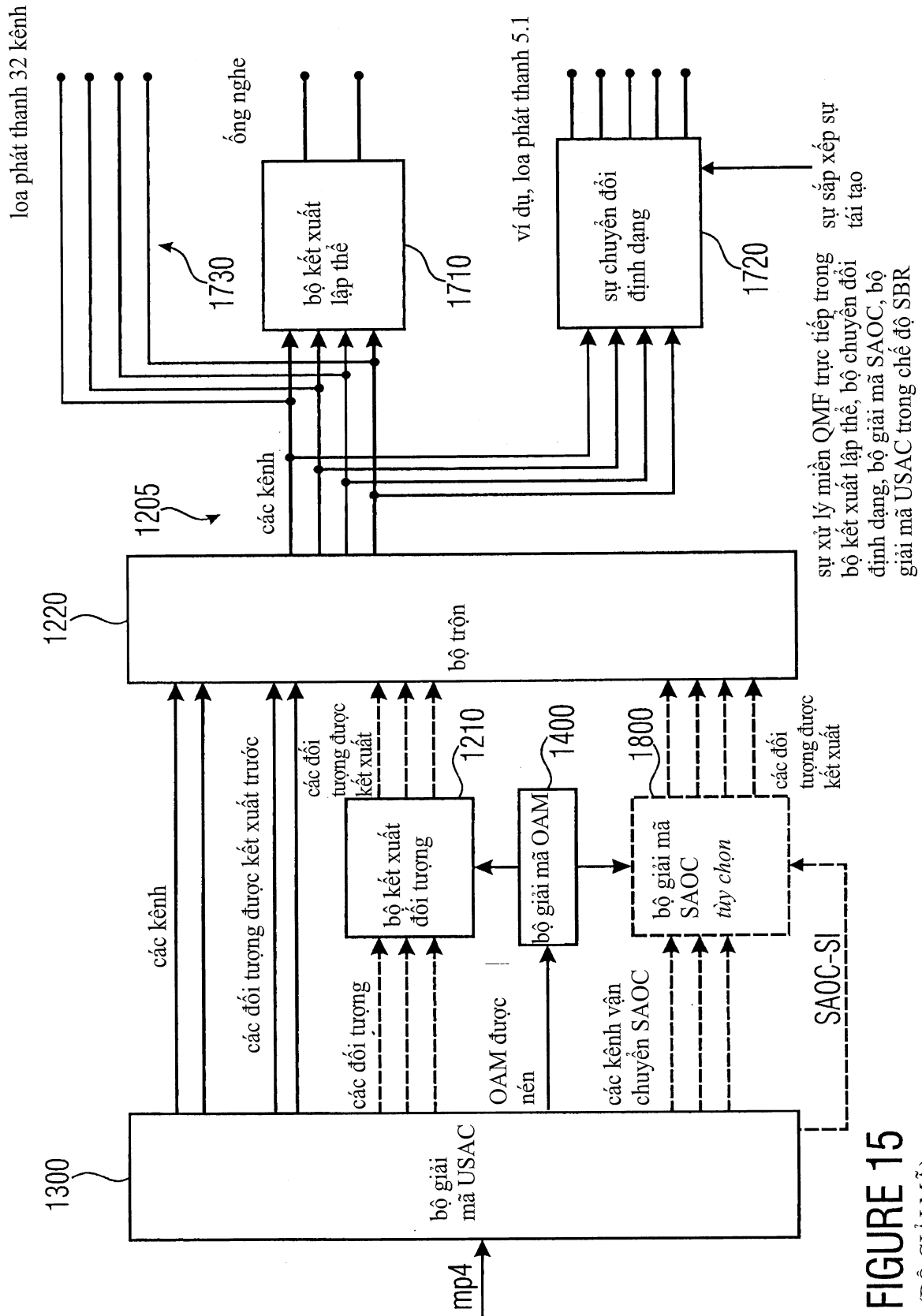


FIGURE 15
(BỘ GIẢI MÃ)

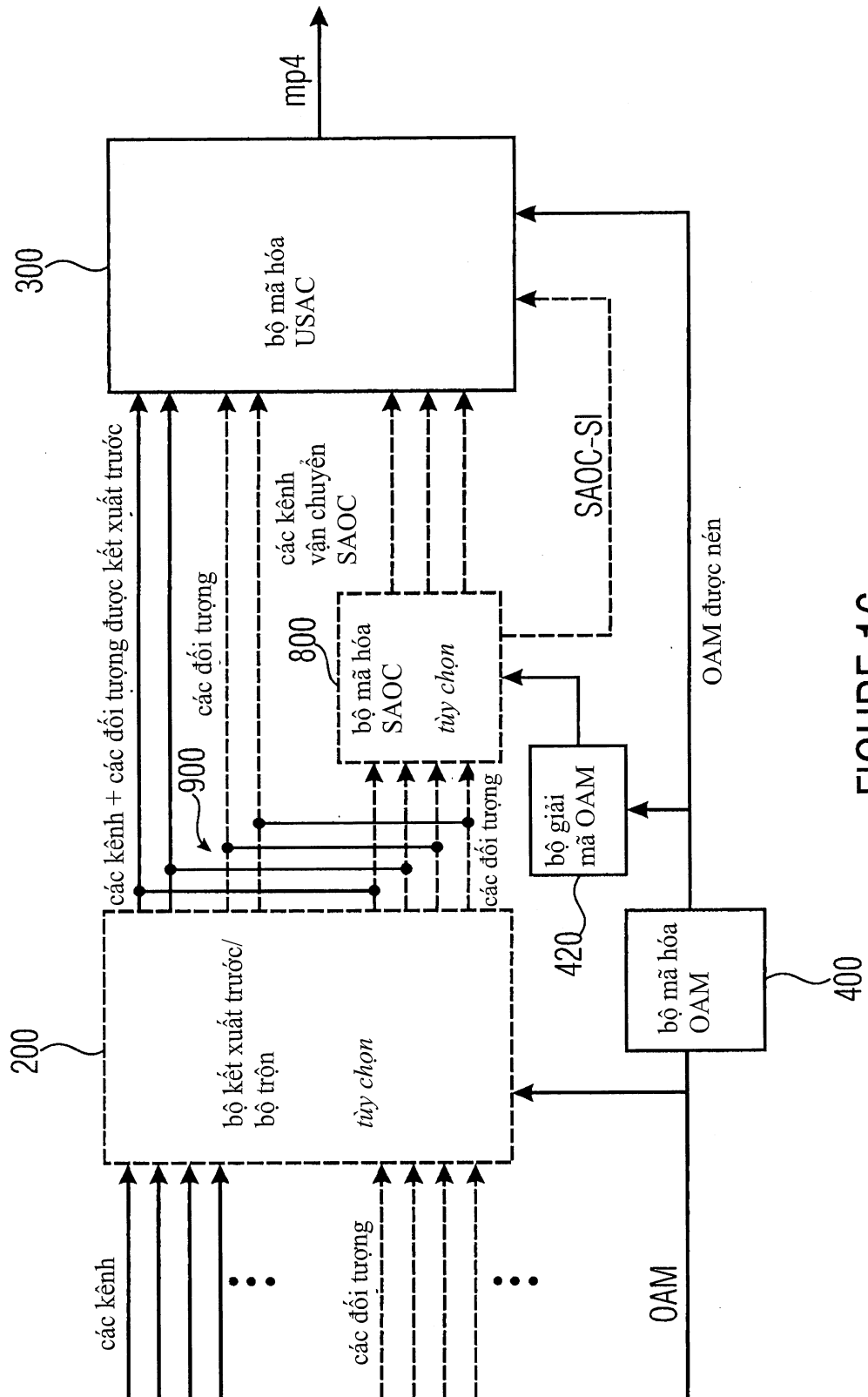


FIGURE 16
(BỘ MÃ HÓA)

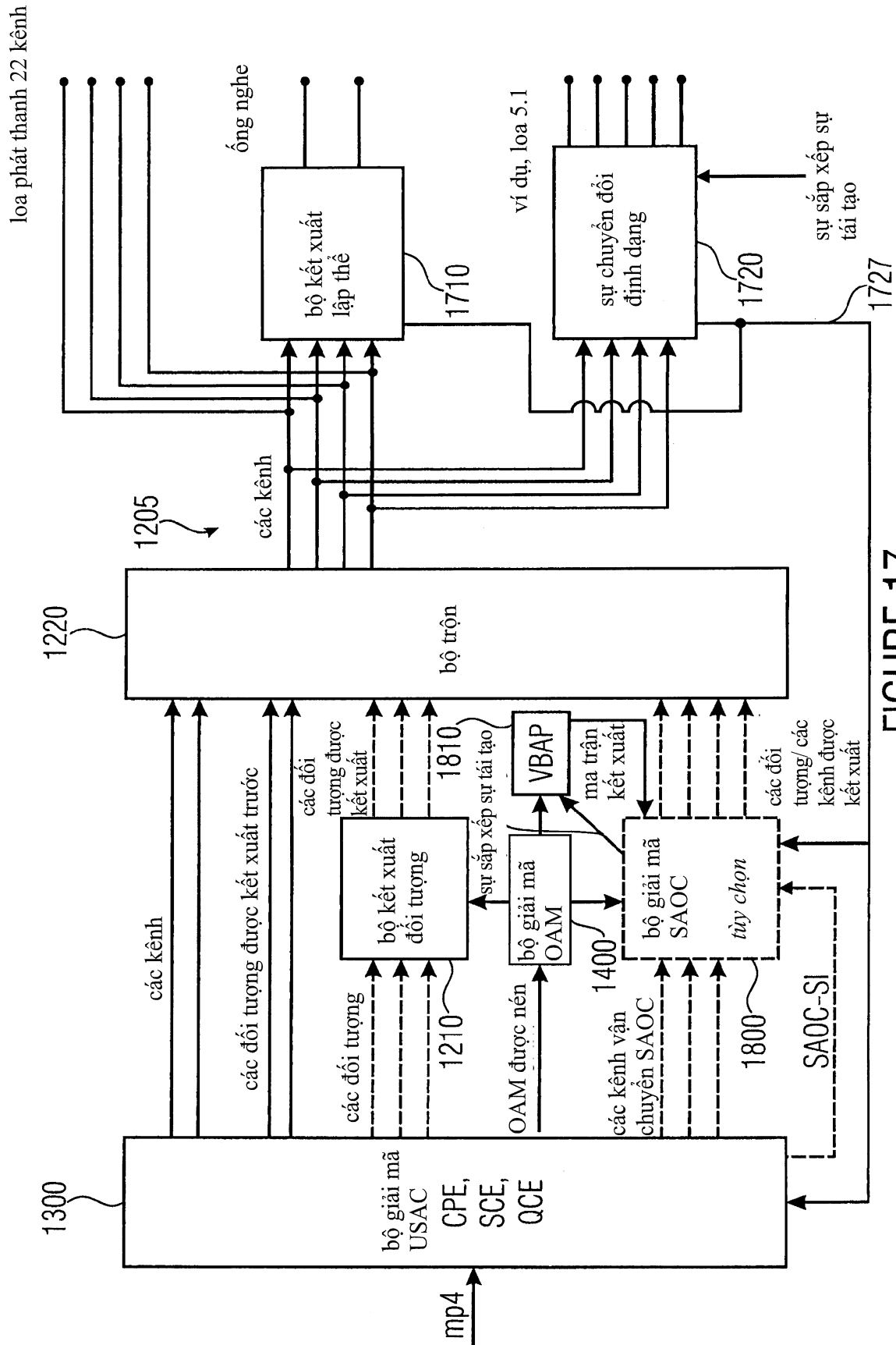


FIGURE 17
(BỘ GIẢI MÃ)