(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 771 200**

(51) Int. CI.:

H03G 3/00 (2006.01)
H03G 5/16 (2006.01)
H03G 5/00 (2006.01)
G10L 19/032 (2013.01)
G10L 19/26 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **10.02.2017 PCT/EP2017/053068**

(87) Fecha y número de publicación internacional: **24.08.2017 WO17140600**

(96) Fecha de presentación y número de la solicitud europea: **10.02.2017 E 17703771 (0)**

(97) Fecha y número de publicación de la concesión europea: **04.12.2019 EP 3417544**

(54) Título: **Postprocesador, preprocesador, codificador de audio, decodificador de audio y métodos relacionados para mejorar el procesamiento de transitorios**

(30) Prioridad:

17.02.2016 EP 16156200

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
06.07.2020

(73) Titular/es:

FRAUNHOFER-GESELLSCHAFT ZUR FÖRDERUNG DER ANGEWANDTEN FORSCHUNG E.V. (100.0%)
Hansastraße 27c
80686 München, DE

(72) Inventor/es:

GHIDO, FLORIN;
DISCH, SASCHA;
HERRE, JÜRGEN;
ADAMI, ALEXANDER y
REUTELHUBER, FRANZ

(74) Agente/Representante:

ARIZTI ACHA, Monica

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Postprocesador, preprocesador, codificador de audio, decodificador de audio y métodos relacionados para mejorar el procesamiento de transitorios

La presente invención se relaciona con el procesamiento de audio y, en particular, con el procesamiento de audio en el contexto del preprocesamiento y postprocesamiento de audio.

Pre-ecos: El problema del enmascaramiento temporal

Los codificadores perceptuales clásicos basados en bancos de filtro como MP3 o AAC están destinados principalmente para aprovechar el efecto perceptual del enmascaramiento simultáneo, aunque también pueden tener que tratar el aspecto temporal del fenómeno del enmascaramiento.: El ruido es enmascarado por poco tiempo antes y después de la presentación de una señal de enmascaramiento (fenómeno de pre-enmascaramiento y post-enmascaramiento). El post-enmascaramiento se observa durante un período de tiempo mucho más largo que el pre-enmascaramiento (en el orden de 10.0-50.0ms en lugar de 0.5-2.0ms, dependiendo del nivel y duración del enmascarador).

Por consiguiente, el aspecto temporal del enmascaramiento da lugar a una necesidad adicional para un esquema de codificación perceptual: Para obtener una calidad de codificación perceptualmente transparente, el ruido de cuantificación no puede exceder además el umbral enmascarado dependiente del tiempo.

En la práctica, este requisito no es fácil de obtener en el caso de los codificadores perceptuales, puesto que el uso de la descomposición de una señal espectral para la cuantificación y codificación implica que un error de cuantificación introducido en este dominio se ha de difundir en el tiempo tras la reconstrucción por el banco de filtro de síntesis (principio de incerteza de tiempo/ frecuencia). En el caso de los diseños de bancos de filtro utilizados corrientemente (por ej. una MDCT (por sus siglas en inglés, Modified Discrete Cosine Transform, Transformada de Coseno Discreta Modificada de 1024 líneas) esto significa que el ruido de cuantificación se puede haber difundido durante un período de más de 40 milisegundos a una frecuencia o tasa de muestreo de CD. Esto conduce a problemas cuando la señal a codificar contiene Fuertes componentes de señal solo en partes de la ventana de banco de filtros de análisis, es decir, para las señales transitorias. En particular, el ruido de cuantificación se dispersa antes del inicio de la señal y, en casos extremos, incluso puede exceder el nivel de los componentes de la señal original durante ciertos intervalos de tiempo. Un conocido ejemplo de una señal percusiva crítica es el de una grabación de castañuelas en la cual, después de decodificar el ruido de cuantificación los componentes se dispersan un cierto tiempo antes del “ataque” de la señal original. Esa constelación se conoce tradicionalmente como “fenómeno pre-eco” [Joh92b].

Debido a las propiedades del Sistema auditivo humano, esos “pre-ecos” solo son enmascarados si no hay una cantidad significativa de ruido de codificación presente durante más de alrededor de 2,0ms antes del inicio de la señal. De lo contrario, el ruido de codificación se percibe en forma de error pre-eco, es decir un breve evento de ruido que precede al inicio de la señal. Para evitar ese tipo de errores, se debe tener cuidado de mantener las características temporales correctas del ruido de cuantificación de tal manera que se satisfagan las condiciones para el enmascaramiento temporal. Este problema de modelado de ruido temporal ha dificultado, tradicionalmente, la obtención de una buena calidad de señal perceptual a bajas tasas de bits en el caso de las señales transitorias como de castañuelas, glockenspiel (carrillón), triángulo, etc.

Señales similares a aplauso: Una clase extremadamente crítica de señales

Aunque las señales transitorias antes citadas pueden disparar pre-ecos en los códecs de audio perceptuales, éstas exhiben ataques individuales aislados, es decir que hay un cierto lapso mínimo hasta que aparece el siguiente ataque. Por consiguiente, le lleva a un codificador de audio perceptual cierto tiempo recuperarse del procesamiento del último ataque y puede recoger nuevamente, por ej., bits libres para hacer frente al siguiente ataque (véase “reserva de bits” que se describe más adelante). Por el contrario, el sonido de una audiencia que aplaude consiste en un flujo constante de palmadas densamente espaciadas, cada una de las cuales es un evento transitorio en sí. La Fig. 11 expone una ilustración de la envolvente temporal de alta frecuencia de una señal estéreo de aplauso. Como se puede ver, el período promedio entre los eventos de palmas subsiguientes es significativamente inferior a 10ms.

Por esta razón, las señales de aplauso y similares a aplauso (como de gotas de lluvia o fuegos artificiales crepitantes) constituyen una clase de señales sumamente difíciles de codificar, si bien son comunes en numerosas grabaciones en vivo. Este también es el caso cuando se emplean métodos paramétricos para la codificación conjunta de dos o más canales [Hot08].

Estrategias tradicionales para la codificación de señales transitorias

Se ha propuesto una serie de técnicas para evitar los errores pre-eco en la señal codificada / decodificada:

Control pre-eco y reserva de bits

Una manera consiste en aumentar la precisión de codificación en el caso de los coeficientes espectrales de la ventana de banco de filtros que cubre en primer lugar la porción transitoria de la señal (el llamado “control pre-eco”, [MPEG1]). Dado que esto aumenta considerablemente la cantidad de bits necesarios para la codificación de esas tramas, este método no se puede aplicar en un codificador con tasa de bits constante. Hasta cierto punto, se puede hacer frente a las variaciones locales en la demanda de tasa de bits utilizando una reserva de bits ([Bra87], [MPEG1]). Esta técnica permite manejar las demandas pico de tasa de bits utilizando bits que han sido reservados durante la codificación de tramas anteriores mientras la tasa de bits promedio se mantiene constante.

Conmutación adaptativa de ventana

Una estrategia diferente empleada en muchos codificadores de audio perceptuales es la conmutación adaptativa de ventanas introducida por Edler [Edl89]. Esta técnica adapta el tamaño de las ventas de bancos de filtros a las características de la señal de entrada. Aunque las partes estacionarias de la señal son codificadas utilizando una longitud de ventana extensa, se utilizan ventanas cortas para codificar las partes transitorias de la señal. De esta manera, se puede reducir considerablemente la demanda pico de bits, puesto que la región para la cual es indispensable una elevada precisión de codificación está limitada en el tiempo. Los pre-ecos tienen, implícitamente, duración limitada por el menor tamaño de transformada.

Modelado Temporal de Ruido (TNS)

El Modelado Temporal de Ruido (TNS, por sus siglas en inglés Temporal Noise Shaping) fue introducido en [Her96] y obtiene un modelado temporal del ruido de cuantificación mediante la aplicación de codificación predictiva de lazo abierto en la dirección de la frecuencia en los bloques temporales en el dominio espectral.

Modificación de ganancia (control de ganancia)

Otra manera de evitar la dispersión temporal del ruido de cuantificación consiste en aplicar una modificación ganancia de ganancia (proceso de control de ganancia) a la señal ante de calcular su descomposición espectral y codificación.

En la Fig. 12 se ilustra el principio de esta estrategia. La dinámica de la señal de entrada se reduce mediante una modificación de ganancia (preprocesamiento multiplicativo) antes de su codificación. De esta manera, se atenúan los “picos” de la señal con anterioridad a la codificación. Los parámetros de la modificación de ganancia son transmitidos en el flujo de bits. Utilizando esta información se revierte el proceso del lado del decodificador, es decir que después de la decodificación otra modificación de ganancia restablece la dinámica original de la señal.

[Lin93] propuso un control de ganancia como adición a un codificador perceptual en que la modificación de ganancia se lleva a cabo en la señal en el dominio del tiempo (y por consiguiente en todo el espectro de la señal).

Ya se ha utilizado la modificación / control de ganancia dependiente de la frecuencia en una cantidad de casos:

Control de ganancia basado en filtros: En su disertación [Vau91], Vaupel nota que el control de ganancia de la banda completa no da buen resultado. Para obtener un control de ganancia dependiente de la frecuencia propone un par de compresor y filtro expansor que puede ser controlado dinámicamente en sus características de ganancia. Este esquema está ilustrado en las Figs. 13a y 13b.

La variación de la respuesta a la frecuencia del filtro está expuesta en la Fig. 13b.

Control de ganancia con banco de filtros híbrido (ilustrado en la Fig. 14): En el perfil SSR del esquema de Codificación Avanzada de Audio de MPEG-2 [Bos96], se utiliza el control de ganancia dentro de una estructura de banco de filtros híbrido. Una primera etapa del banco de filtros (PQF) divide la señal de entrada en cuatro bandas de anchos iguales. A continuación, un detector de ganancia y un modificador de ganancia ejecutan el procesamiento de codificación con control de ganancia. Por último, en una segunda etapa, cuatro bancos de filtro de MDCT con tamaño reducido (256 en lugar de 1024) dividen adicionalmente la señal obtenida y producen los componentes espectrales que se utilizan para la codificación subsiguiente.

Modelado guiado de envolvente (GES, por sus siglas en inglés) es una herramienta contenida en MPEG Surround (por sus siglas en inglés, Moving Picture Expert Group, Grupo de Expertos en Imágenes en Movimiento Surround o Envolvente) que transmite parámetros de envolvente temporal individuales para cada canal y restablece las envolventes temporales del lado del decodificador. Téngase en cuenta que, a diferencia del procesamiento HREP

(por sus siglas en inglés, High Resolution Envelope Processing, Procesamiento de Envolvente de Alta Resolución), no hay aplanamiento de la envolvente del lado del codificador para mantener la compatibilidad con sistemas anteriores en la mezcla descendente. Otra herramienta que da resultado para efectuar el modelado de envolvente es el Procesamiento Temporal de Sub-bandas (del inglés Temporal Processing (STP). En este caso, se aplican filtros de LPC (por sus siglas en inglés, Linear Prediction Coding, Codificación de Predicción Lineal) dentro de una representación de bancos de filtro QMF (por sus siglas en inglés Quadrature Mirror Filters, Filtros Espejo en Cuadratura) de las señales de audio.

La técnica anterior relacionada está documentada en las publicaciones de patente WO 2006/045373 A1, WO 2006/045371 A1, WO2007/042108 A1, WO 2006/108543 A1 o WO 2007/110101 A1.

Referencias

[Bos96] M. Bosi, K. Brandenburg, S. Quackenbush, L. Fielder, K. Akagiri, H. Fuchs, M. Dietz, J. Herre, G. Davidson, Oikawa: "MPEG-2 Advanced Audio Coding", 101a Convención de AES, Los Ángeles 1996

[Bra87] K. Brandenburg: "OCF - A New Coding Algorithm for High Quality Sound Signals", Proc. IEEE ICASSP, 1987

[Joh92b]J. D. Johnston, K. Brandenburg: "Wideband Coding Perceptual Considerations for Speech and Music", in S. Furui and M. M. Sondhi, editores: "Advances in Speech Signal Processing", Marcel Dekker, New York, 1992

[Edl89] B. Edler: "Codierung von Audiosignalen mit überlappender Transformation und adaptiven Fensterfunktionen", Frequenz, Vol. 43, pp. 252-256, 1989

[Her96] J. Herre, J. D. Johnston: "Enhancing the Performance of Perceptual Audio Coders by Using Temporal Noise Shaping (TNS)", 101a Convención de AES, Los Angeles 1996, Preimpresión 4384

[Hot08] Gerard Hotho, Steven van de Par, and Jeroen Breebaart: "Multichannel coding of applause signals", EURASIP Journal of Advances in Signal Processing, Hindawi, Enero 2008, doi: 10.1155/2008/531693

[Lin93] M. Link: "An Attack Processing of Señales de audio for Optimizing the Temporal Characteristics of a Low Bit-Rate Audio Coding System", 95a Convención de AES, New York 1993, Preimpresión 3696

[MPEG1] ISO/IEC JTC1/SC29/WG11 MPEG, International Standard ISO 11172-3 "Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s"

[Vau91] T. Vaupel: "Ein Beitrag zur Transformationscodierung von Audiosignalen unter Verwendung der Methode der 'Time Domain Aliasing Cancellation (TDAC)' und einer Signalkompandierung im Zeitbereich", PhD Thesis, Universität-Gesamthochschule Duisburg, Alemania, 1991

Una reserva de bits puede contribuir a manejar demandas pico de tasas de bit en un codificador perceptual y, de esa manera, mejorar la calidad perceptual de las señales transitorias. Sin embargo, en la práctica, el tamaño de la reserva de bits tiene que ser de un tamaño poco realista para evitar errores al codificar señales de entrada de naturaleza muy transitoria sin más precauciones.

La conmutación adaptativa de ventanas limita la demanda de bits de las partes transitorias de la señal y los pre-ecos reducidos mediante la confinación de los transitorios a bloques de transformación cortos. Una limitación de la conmutación adaptativa de ventanas está dada por su tiempo de latencia y repetición: El ciclo de restablecimiento más rápido posible entre dos secuencias de bloques requiere por lo menos tres bloques (“corto”→“fin”→“comienzo”→“corto”, de aproximadamente. 30.0 - 60.0 ms para tamaños de bloques típicos de 512 - 1024 muestras), o sea muy prolongados para ciertos tipos de señales de entrada, incluyendo el aplauso. En consecuencia, solo se podría evitar la dispersión temporal del ruido de cuantificación para señales que se asemejan a aplausos seleccionando de modo permanente el tamaño de ventana corto, lo que habitualmente lleva a una reducción de la eficiencia en la codificación de fuente del codificador.

El TNS (por sus siglas en inglés, Temporal Noise Shaping, Modelado de Ruido Temporal) ejecuta el aplanamiento temporal en el codificador y el modelado temporal en el decodificador. En principio, la resolución temporal arbitrariamente fina es posible. Sin embargo, en la práctica la eficiencia está limitada por la superposición “aliasing” temporal del banco de filtros del codificador (por lo general una MDCT (por sus siglas en inglés, Modified Discrete Cosine Transform, Transformada de Coseno Discreta Modificada), es decir un bloque de transformación superpuesto con 50% de superposición). Por consiguiente, el ruido de codificación modelado aparece también de manera espejada a la salida del banco de filtros de síntesis.

Las técnicas de control de ganancia en banda ancha adolecen de una falta de resolución espectral. No obstante, para tener buena eficiencia para muchas señales, es importante que se pueda aplicar el proceso de modificación de ganancia de manera independiente en partes diferentes del espectro de audio porque con frecuencia los eventos transitorios son dominantes sólo en partes del espectro (en la práctica los eventos que son difíciles de codificar están presentes casi siempre en la parte de alta frecuencia del espectro). En efecto, la aplicación de una modificación multiplicativa dinámica de la señal de entrada antes de su descomposición espectral en un codificador es equivalente a una modificación dinámica de la ventana de análisis del banco de filtros. Dependiendo de la forma de la función de modificación de ganancia, la respuesta a la frecuencia de los filtros de análisis se modifica de acuerdo con la función de generación de ventana compuesta. Sin embargo, no es conveniente ampliar la respuesta de frecuencia de los canales de filtros de baja frecuencia del banco de filtros, puesto que esto aumenta las discordancias a la escala crítica de ancho de banda.

El control de ganancia empleando un banco de filtros híbrido tiene la desventaja de una mayor complejidad informática, ya que el banco de filtros de la primera etapa tiene que lograr una selectividad considerable para evitar distorsiones por superposición después de esta última escisión por el banco de filtros de la segunda etapa. Además, las frecuencias de cruce entre las bandas de control de ganancia se finan a un cuarto de la frecuencia de Nyquist, es decir son de 6, 12 y 18kHz para una frecuencia de muestreo de 48kHz. Para la mayor parte de las señales, un primer cruce a 6kHz es demasiado alto para una buena eficiencia.

Se sabe que las técnicas de modelado de envolvente contenidas en las soluciones de codificación multicanal semiparamétrica como MPEG Surround (STP, GES) mejoran la calidad perceptual de los transitorios por medio de un remodelado temporal de la señal de salida o partes de la misma en el decodificador. Sin embargo, estas técnicas no ejecutan el aplanamiento temporal antes del codificador. Por ende, la señal transitoria entra de todas maneras al codificador con su dinámica de corto tiempo original e impone una elevada demanda de tasa de bits en el presupuesto de bits de los codificadores.

También se conoce de acuerdo con la publicación Martin Link: “Un Procesamiento de Ataque de Señales de Audio para Optimizar las Características Temporales de un Sistema de Codificación de Audio de Baja Velocidad de Bits”, 95ª Convención AES, 1er de octubre de 1993, un método para disminuir el ruido de cuantificación debajo de la audibilidad mediante el preprocesamiento de una señal cuando se producen ataques agudos y el uso de una información lateral para señalar el procesamiento a un decodificador.

SUMARIO DE LA INVENCIÓN

Un objetivo de la presente invención es dar a conocer un concepto mejorado de preprocesamiento de audio, postprocesamiento de audio o codificación de audio o, por otro lado, decodificación de audio.

Este objetivo se alcanza mediante un postprocesador de audio de acuerdo con la reivindicación 1, un preprocesador de audio de acuerdo con la reivindicación 10, un aparato codificador de audio de acuerdo con la reivindicación 16, un aparato decodificador de audio de acuerdo con la reivindicación 17, un método de postprocesamiento de acuerdo con la reivindicación 18, un método de preprocesamiento de acuerdo con la reivindicación 19, un método de codificación de acuerdo con la reivindicación 20, un método de decodificación de audio de acuerdo con la reivindicación 21 o un programa de computación de acuerdo con la reivindicación 22.

Un primer aspecto de la presente invención consiste en un postprocesador de audio para postprocesar una señal de audio que contiene información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral, que comprende un extractor de banda para extraer una banda de alta frecuencia de la señal de audio y una banda de baja frecuencia de la señal de audio; un procesador de banda alta para ejecutar una modificación variable en el tiempo de la banda alta de conformidad con la información de ganancia de alta frecuencia variable en el tiempo para obtener una banda de alta frecuencia procesada; y un combinador para combinar la banda de alta frecuencia procesada y la banda de baja frecuencia.

Un segundo aspecto de la presente invención consiste en un preprocesador de audio para preprocesar una señal de audio, que comprende un analizador de señales para analizar la señal de audio para determinar información de ganancia de alta frecuencia variable en el tiempo; un extractor de banda para extraer una banda de alta frecuencia de la señal de audio y una banda de baja frecuencia de la señal de audio; un procesador de banda alta para ejecutar una modificación variable en el tiempo de la banda alta de conformidad con la información de ganancia de alta frecuencia variable en el tiempo para obtener una banda de alta frecuencia procesada; un combinador para combinar la banda de alta frecuencia procesada y la banda de baja frecuencia para obtener una señal de audio preprocesada; y una interfaz de salida para generar una señal de salida que comprende la señal de audio preprocesada y la información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral.

Un tercer aspecto de la presente invención se refiere a un aparato codificador de audio para codificar una señal de audio, que comprende el preprocesador de audio del primer aspecto, configurado para generar la señal de salida

que contiene la información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral; un codificador de núcleo para generar una señal codificada en el núcleo e información lateral de núcleo; y una interfaz de salida para generar una señal codificada que comprende la señal codificada en el núcleo, la información lateral de núcleo y la información de ganancia de alta frecuencia variable en el tiempo como información lateral adicional.

Un cuarto aspecto de la presente invención se refiere a un aparato decodificador de audio, que comprende una interfaz de entrada para recibir una señal de audio codificada que comprende la señal codificada en el núcleo, la información lateral de núcleo y la información de ganancia de alta frecuencia variable en el tiempo como información lateral adicional; un decodificador de núcleo para decodificar la señal codificada en el núcleo utilizando la información lateral de núcleo para obtener una señal de núcleo decodificada; y un postprocesador para postprocesar la señal de núcleo decodificada utilizando la información de ganancia de alta frecuencia variable en el tiempo de conformidad con el segundo aspecto antes expuesto.

Un quinto aspecto de la presente invención se refiere a un método de postprocesamiento una señal de audio que contiene información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral, que comprende extraer una banda de alta frecuencia de la señal de audio y una banda de baja frecuencia de la señal de audio; ejecutar una modificación variable en el tiempo de la banda alta de conformidad con la información de ganancia de alta frecuencia variable en el tiempo para obtener una banda de alta frecuencia procesada; y combinar la banda de alta frecuencia procesada y la banda de baja frecuencia.

Un sexto aspecto de la presente invención es un método de preprocesamiento una señal de audio, que comprende analizar la señal de audio para determinar información de ganancia de alta frecuencia variable en el tiempo; extraer una banda de alta frecuencia de la señal de audio y una banda de baja frecuencia de la señal de audio; ejecutar una modificación variable en el tiempo de la banda alta de conformidad con la información de ganancia de alta frecuencia variable en el tiempo para obtener una banda de alta frecuencia procesada; combinar la banda de alta frecuencia procesada y la banda de baja frecuencia para obtener una señal de audio preprocesada; y generar una señal de salida que comprende la señal de audio preprocesada y la información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral.

Un séptimo aspecto de la presente invención se refiere a un método de codificación una señal de audio, que comprende el método de preprocesamiento de audio del sexto aspecto, configurado para generar la señal de salida que contiene la información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral; generar una señal codificada en el núcleo e información lateral de núcleo; y generar una señal codificada que comprende la señal codificada en el núcleo, la información lateral de núcleo, y la información de ganancia de alta frecuencia variable en el tiempo como información lateral adicional.

Un octavo aspecto de la presente invención se refiere a un método de decodificación de audio, que comprende recibir una señal de audio codificada que comprende una señal codificada en el núcleo, información lateral de núcleo y la información de ganancia de alta frecuencia variable en el tiempo como información lateral adicional; decodificar la señal codificada en el núcleo utilizando la información lateral de núcleo para obtener una señal de núcleo decodificada; y postprocesar la señal de núcleo decodificada utilizando la información de ganancia de alta frecuencia variable en el tiempo de conformidad con el quinto aspecto.

Un noveno aspecto de la presente invención se relaciona con un programa de computación o un medio de almacenamiento no transitorio en el cual se ha almacenado el programa de computación para ejecutar, al correr en un ordenador o un procesador, cualquiera de los métodos de conformidad con el quinto, sexto, séptimo u octavo aspecto antes expuestos.

La presente invención da a conocer un procesamiento de alta frecuencia selectivo de las bandas tal como una atenuación selectiva en un preprocesador o una amplificación selectiva en un postprocesador para codificar de manera selectiva una cierta clase de señales tales como señales transitorias con información de ganancia de alta frecuencia variable en el tiempo para la banda alta. Por consiguiente, la señal preprocesada es una señal que consta de la información lateral adicional en forma de información llana de ganancia de alta frecuencia variable en el tiempo y de la señal en sí, por lo que cierta clase de señales, como las señales transitorias, ya no aparecen en la señal preprocesada o sólo aparecen en menor grado. En el postprocesamiento de audio, se recupera la forma original de la señal mediante la ejecución de la multiplicación variable en el tiempo de la banda de alta frecuencia, de conformidad con la información de ganancia de alta frecuencia variable en el tiempo asociada a la señal de audio en forma de información lateral por lo que, al final, es decir, con posterioridad a un encadenamiento que consiste en preprocesamiento, codificación, decodificación y postprocesamiento, el oyente no percibe diferencias sustanciales con la señal original y, en particular, no percibe una señal de naturaleza transitoria reducida, si bien los bloques de codificador de núcleo / decodificador de núcleo internos, en los cuales la posición para procesar una señal menos transitoria ha dado como resultado, para el procesamiento del codificador, una cantidad reducida de bits necesarios por un lado una calidad de audio incrementada por el otro, ya que se ha eliminado de la señal la clase de señales de difícil codificación antes de que el codificador haya iniciado en realidad su tarea. Sin embargo, esta eliminación de

las porciones difíciles de codificar de la señal no da lugar a una calidad de audio reducida, ya que estas porciones de la señal son reconstruidas por el postprocesamiento de audio posterior a la operación del decodificador.

En realizaciones preferidas, el preprocesador también amplifica partes ligeramente más silenciosas que un nivel de fondo promedio y el postprocesador las atenúa. Este procedimiento adicional es de potencial utilidad tanto para ataques fuertes individuales como para partes entre eventos transitorios consecutivos.

A continuación, se esbozan determinadas ventajas de las realizaciones preferidas.

HREP (por sus siglas en inglés, High Resolution Envelope Processing, Procesamiento de Envolvente de Alta Resolución) es una herramienta para la codificación mejorada de señales que consisten predominantemente en muchos eventos transitorios densos, tales como sonidos de aplausos, gotas de lluvia, etc. Del lado del codificador, la herramienta actúa como un preprocesador con alta resolución temporal antes del códec de audio perceptual en sí mediante el análisis de la señal de entrada, atenuando, y de esa manera también aplanando temporalmente la parte de alta frecuencia de los eventos transitorios, y generando una pequeña cantidad de información lateral (1-4 kbps en el caso de las señales estéreo). Del lado del decodificador, la herramienta actúa como un postprocesador posterior al códec de audio mediante el refuerzo y, de esa manera, el modelado temporal de la parte de alta frecuencia de los eventos transitorios, haciendo uso de la información lateral que se ha generado durante la codificación. Los beneficios de la aplicación de HREP son dobles: HREP afloja la demanda de tasa de bits impuesta sobre el codificador mediante la reducción de la dinámica de corto tiempo de la señal de entrada; además, HREP garantiza un restablecimiento correcto de la envolvente en la etapa de mezcla (ascendente) en el decodificador, que es tanto más importante si se han aplicado técnicas de codificación paramétrica multicanal dentro del códec.

Por añadidura, la presente invención es ventajosa porque mejora la eficiencia de la codificación en el caso de las señales similares a aplauso utilizando los métodos apropiados de procesamiento de señales, por ejemplo, en el preprocesamiento por un lado o el postprocesamiento por el otro.

Otra ventaja de la presente invención es que el procesamiento de envolvente de alta resolución (HREP) de la invención, es decir, el preprocesamiento de audio o el postprocesamiento de audio soluciona los problemas de la técnica anterior llevando a cabo un pre-aplanamiento antes del codificador o un correspondiente aplanamiento inverso con posterioridad a un decodificador.

A continuación, se resumen las características particulares y novedosas de las realizaciones de la presente invención relacionadas con el procesamiento de señales HREP y se describen sus singulares ventajas.

HREP procesa señales de audio en solo dos bandas de frecuencia que son divididas por filtros. Esto hace que el procesamiento sea sencillo y de baja complejidad informática y estructural. Sólo se procesa la banda alta; la banda baja se transmite sin modificaciones.

Estas bandas de frecuencia se derivan mediante filtrado de paso bajo de la señal de entrada para computar la primera banda. La banda de paso alto (segunda) se obtiene simplemente restando el componente de paso bajo de la señal de entrada. De esta manera, solo es necesario calcular un filtro de manera explícita en lugar de dos, lo que reduce la complejidad. Por otro lado, se puede computar explícitamente la señal filtrada de paso alto y se puede derivar el componente de paso bajo como diferencia entre la señal de entrada y la señal de paso alto.

Para admitir implementaciones de postprocesador de baja complejidad, son posibles las siguientes restricciones

- Limitación de canales/ objetos activos de HREP
- Limitación a los factores de ganancia máximos transmitidos g(k) que no son triviales (los factores de ganancia triviales de 0dB mitigan la necesidad de un par DFT/iDFT asociado (por sus siglas en inglés, Discrete Fourier Transform, Transformada Discreta de Fourier, iDFT Inverse Discrete Fourier Transform, Transformada Discreta de Fourier Inversa)
- El cálculo de la DFT/iDFT en una topología eficiente dispersa de radix-2 de base partida.

En una realización el codificador o el preprocesador de audio asociado al codificador de núcleo está configurado para limitar el número máximo de canales u objetos en que el HREP está activo al mismo tiempo, o el decodificador o el postprocesador de audio asociado al decodificador de núcleo está configurado para ejecutar solamente un postprocesamiento con el número máximo de canales u objetos en que el HREP está activo al mismo tiempo. Un número preferido para la limitación de canales u objetos activos es 16 y es aún más preferible que sean 8.

En una realización adicional, el codificador HREP o el preprocesador de audio asociado al codificador de núcleo está configurado para limitar la salida a un máximo de factores de ganancia no triviales o el decodificador o el

postprocesador de audio asociado al decodificador de núcleo está configurado de tal manera que los factores de ganancia triviales de valor “1” no computen un par DFT/iDFT, sino que se transmiten en la señal en el dominio del tiempo (en ventana) sin modificar. Un número preferido para la limitación de factores de ganancia no triviales es 24 y es aún más preferible que sean 16 por trama y canal u objeto.

En una realización adicional, el codificador HREP o el preprocesador de audio asociado al codificador de núcleo está configurado para calcular la DFT/iDFT en una topología dispersa y eficiente de radix 2 (de base partida) o el decodificador o el postprocesador de audio asociado al decodificador de núcleo está configurado para calcular asimismo la DFT/iDFT en una topología dispersa y eficiente de radix 2 o de base partida.

El filtro de paso bajo de HREP puede ser implementado de manera eficiente utilizando un algoritmo de FFT. Se presenta aquí un ejemplo a partir de una decimación (o diezmado) en tiempo de la topología de FFT radix 2 de N=8 puntos, en la cual sólo se necesita X(0) y X(1) para continuar el procesamiento; en consecuencia, no se necesita E(2) y E(3) y O(2) ni O(3); seguidamente, supongamos que ambas DFTs de N/2 puntos se subdividen a su vez en DFTs de N/4 puntos + sus consiguientes mariposas posteriores. Ahora se pueden repetir las omisiones antes descritas de manera análoga, etc., como se ilustra en la Fig. 15. A diferencia de un esquema de control de ganancia basado en bancos de filtros híbridos (en que las frecuencias de cruce de las bandas de procesamiento están dictadas por la primera etapa de banco de filtros, y están prácticamente atadas a las fracciones de potencia de dos de la frecuencia de Nyquist), la frecuencia partida de HREP puede/podría ser ajustada libremente mediante la adaptación del filtro. Esto permite una adaptación óptima a las características de la señal y los requisitos psicoacústicos.

A diferencia de un esquema de control de ganancia basado en bancos de filtros híbridos, no hay necesidad de largos filtros para separar las bandas de procesamiento a fin de evitar problemas de superposición o aliasing después de la segunda etapa de banco de filtros. Esto es posible porque HREP es un pre-/postprocesador autónomo que no tiene que operar con un banco de filtros con muestreo crítico.

A diferencia de otros esquemas de control de ganancia, HREP se adapta dinámicamente a la estadística local de la señal (computando una media deslizante de dos colas de la envolvente de energía de fondo de alta frecuencia de entrada). Reduce la dinámica de la señal de entrada a una determinada fracción de su tamaño original (el denominado factor alfa). Esto habilita una operación “suave” del esquema sin introducir errores por la interacción perjudicial con el códec de audio.

A diferencia de otros esquemas de control de ganancia, HREP puede compensar la pérdida adicional de dinámica por medio de un códec de audio de baja tasa de bits modelando esto como “pérdida de una determinada fracción de dinámica energética” (el llamado factor beta) y revirtiendo esta pérdida.

El par pre/postprocesador de HREP está (casi) reconstruyendo perfectamente en ausencia de cuantificación (es decir, sin un códec).

Para lograrlo, el postprocesador utiliza una pendiente adaptativa para el filtro de partición que depende del factor de ponderación de amplitud de alta frecuencia y corrige el error de interpolación que se produce al revertir las ponderaciones espectrales variantes en el tiempo aplicadas a las transformaciones T/F superpuestas mediante la aplicación de un factor de corrección en el dominio del tiempo.

Las implementaciones de HREP pueden contener un denominado Control de meta ganancia (MGC, por sus siglas en inglés, Meta Gain Control) que controla correctamente la potencia del efecto perceptual provisto por el procesamiento HREP y puede evitar errores al procesar señales que no son de aplauso. De esa manera, mitiga los requisitos de precisión de una clasificación externa de señales de entrada para controlar la aplicación del HREP.

Mapeo del resultado de la clasificación de aplausos sobre el MGC y los ajustes de HREP.

HREP es un pre-/postprocesador autónomo que abarca todos los demás componentes del codificador incluyendo las herramientas de extensión de ancho de banda y codificación espacial paramétrica.

HREP relaja los requisitos del codificador de audio de baja tasa de bits por medio del pre-aplanamiento de la envolvente temporal de alta frecuencia. En efecto, se activan menos bloques cortos en el codificador y se necesitan menos filtros de TNS activos.

HREP mejora también la codificación paramétrica multicanal reduciendo la comunicación cruzada entre los canales procesados que se produce normalmente también debido a la resolución de la referencia espacial temporal.

Topología del códec: interacción con TNS/TTS, IGF y relleno estéreo

Formato del flujo de bits: señalización HREP

A continuación, se describen las realizaciones preferidas de la presente invención en el contexto de las figuras adjuntas, en las cuales:

Fig. 1 ilustra un postprocesador de audio de conformidad con una realización;

Fig. 2 ilustra una implementación preferida del extractor de banda de la Fig. 1;

Fig. 3a es una representación esquemática de la señal de audio que contiene información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral;

Fig. 3b es una representación esquemática de un procesamiento realizado por el extractor de banda, el procesador de banda alta o el combinador con bloques superpuestos que presentan una región superpuesta;

Fig. 3c ilustra un postprocesador de audio que consta de un sumador de superposición;

Fig. 4 ilustra una implementación preferida del extractor de banda de la Fig. 1;

Fig. 5a ilustra otra implementación preferida del postprocesador de audio;

Fig. 5b ilustra una incorporación preferida del postprocesador de audio (HREP) en la estructura de un decodificador de audio MPEG-H 3D;

Fig. 5c ilustra otra incorporación preferida del postprocesador de audio (HREP) en la estructura de un decodificador de audio MPEG-H 3D;

Fig. 6a ilustra una realización preferida de la información lateral que contiene la correspondiente información de posición;

Fig. 6b ilustra un extractor de información lateral combinado con un decodificador de información lateral para un postprocesador de audio;

Fig. 7 ilustra un preprocesador de audio de conformidad con una realización preferida;

Fig. 8a ilustra un gráfico de flujo de los pasos ejecutados por el preprocesador de audio;

Fig. 8b ilustra un gráfico de flujo de los pasos ejecutados por el analizador de señales del preprocesador de audio;

Fig. 8c ilustra un gráfico de flujo de los procedimientos ejecutados por el analizador de señales, el procesador de banda alta y la interfaz de salida del preprocesador de audio;

Fig. 8d ilustra un procedimiento ejecutado por el preprocesador de audio de la Fig. 7;

Fig. 9a ilustra un aparato codificador de audio con un preprocesador de audio de conformidad con una realización;

Fig. 9b ilustra un aparato decodificador de audio que comprende un postprocesador de audio;

Fig. 9c ilustra una implementación preferida de un preprocesador de audio;

Fig. 10a ilustra un aparato codificador de audio con funcionalidad multicanal/ multiobjeto;

Fig. 10b ilustra un aparato decodificador de audio con funcionalidad multicanal/ multiobjeto;

Fig. 10c ilustra una implementación adicional de una incorporación del preprocesador y el postprocesador a una cadena de codificación / decodificación;

Fig. 11 ilustra una envolvente temporal de alta frecuencia de una señal de aplauso estéreo;

Fig. 12 ilustra una funcionalidad de un procesamiento con modificación de ganancia;

Fig. 13a ilustra un procesamiento de control de ganancia basado en filtros;

Fig. 13b ilustra diferentes funcionalidades de filtro para el filtro correspondiente de la Fig. 13a;

Fig. 14 ilustra un control de ganancia con banco de filtros híbrido;

Fig. 15 ilustra una implementación de una implementación digital dispersa de transformada de Fourier;

Fig. 16 ilustra una reseña general de una prueba de escucha;

Fig. 17a ilustra puntuaciones MUSHRA (por sus siglas en inglés, Multiple Stimuli with Hidden Reference and Anchor, Estímulos Múltiples con Referencia Oculta y Ancla) absolutas correspondientes a la prueba de 128 kbps 5,1can;

Fig. 17b ilustra puntuaciones MUSHRA diferentes correspondientes a una prueba de 128 kbps 5,1can;

Fig. 17c ilustra puntuaciones MUSHRA correspondientes a las señales de aplauso de la prueba de 128 kbps 5.1can;

Fig. 17d ilustra puntuaciones MUSHRA diferentes correspondientes a las señales de aplauso de la prueba de 128 kbps 5.1can;

Fig. 17e ilustra puntuaciones MUSHRA absolutas correspondientes a la prueba de 48 kbps;

Fig. 17f ilustra puntuaciones MUSHRA diferentes correspondientes a la prueba estéreo de 48 kbps;

Fig. 17g ilustra puntuaciones MUSHRA absolutas correspondientes a la prueba estéreo de 128 kbps y

Fig. 17h ilustra puntuaciones MUSHRA diferentes correspondientes a la prueba estéreo de 128.

La Fig. 1 ilustra una realización preferida de un postprocesador 100 de audio para postprocesar una señal 102 de audio que contiene información 104 de ganancia de alta frecuencia variable en el tiempo en forma de información 106 lateral ilustrada en la Fig. 3a. El postprocesador de audio comprende un extractor 110 de banda para extraer una banda 112 de alta frecuencia de la señal 102 de audio y una banda 114 de baja frecuencia de la señal 102 de audio. Además, el postprocesador de audio de conformidad con esta realización comprende un procesador 120 de banda alta para ejecutar una modificación variable en el tiempo de la banda 112 de alta frecuencia de conformidad con la información 104 de ganancia de alta frecuencia variable en el tiempo para obtener una banda 122 de alta frecuencia procesada. Además, el postprocesador de audio comprende un combinador 130 para combinar la banda 122 de alta frecuencia procesada y la banda 114 de baja frecuencia.

Preferiblemente, el procesador 120 de banda alta lleva a cabo una amplificación selectiva de una banda de alta frecuencia de conformidad con la información de ganancia de alta frecuencia variable en el tiempo correspondiente a esta banda específica. Esto es para deshacer o reconstruir la banda de alta frecuencia original, puesto que la banda de alta frecuencia correspondiente ha sido atenuada previamente en un preprocesador de audio tal como el preprocesador de audio de la Fig. 7 que se describe más adelante.

En particular, en la realización, el extractor 110 de banda se presenta con la señal 102 de audio, a la entrada de ésta, extraída de la señal de audio que tiene información lateral asociada. Además, una salida del extractor de banda está conectada a una entrada del combinador. Asimismo, una segunda entrada del combinador está conectada a una salida del procesador 120 de banda alta para alimentar la banda 122 de alta frecuencia procesada al combinador 130. Más aún, otra salida del extractor 110 de banda está conectada a una entrada del procesador 120 de banda alta. Asimismo, el procesador de banda alta tiene además una entrada de control para recibir la información de ganancia de alta frecuencia variable en el tiempo como se ilustra en la Fig. 1.

La Fig. 2 ilustra una implementación preferida del extractor 110 de banda. En particular, el extractor 110 de banda comprende un filtro 111 de paso bajo que, a su salida, produce la banda 114 de baja frecuencia. Asimismo, la banda 112 de alta frecuencia se genera restando la banda 114 de baja frecuencia de la señal 102 de audio, es decir, la señal de audio que ha sido ingresada al filtro 111 de paso bajo. Sin embargo, el restador 113 puede ejecutar algún tipo de preprocesamiento con anterioridad a la substracción real típicamente por banda como se demuestra con respecto al generador 121 de ventanas para la señal de audio de la Fig. 4 o el bloque correspondiente 121 de la Fig. 5a. De esa manera, el extractor 110 de banda puede comprender, como se ilustra en la Fig. 2, un filtro 111 de paso bajo y el restador 113 conectado subsiguientemente, es decir, el restador 113 que consta de una entrada que está conectada a una salida del filtro 111 de paso bajo y que consta de otra entrada que está conectada a la entrada del filtro 111 de paso bajo.

Por otra parte, sin embargo, el extractor 110 de banda también puede ser implementado utilizando, en realidad, un filtro de paso alto y restando la señal de salida de paso alto o la banda de alta frecuencia de la señal de audio para obtener la banda de baja frecuencia. O, por otra parte, el extractor de banda puede ser implementado sin restador alguno, es decir, mediante una combinación de un filtro de paso bajo y un filtro de paso alto a la manera de un banco

de filtros de dos canales, por ejemplo. Preferiblemente, el extractor 110 de banda de la Fig. 1 (o Fig. 2) es implementado para extraer sólo dos bandas, es decir, una sola banda de baja frecuencia y una sola banda de alta frecuencia, en tanto que estas bandas juntas abarcan todo el rango de frecuencias de la señal de audio.

Preferiblemente, una frecuencia de corte de la banda de baja frecuencia extraída por el extractor 110 de banda es entre 1/8 y 1/3 de una frecuencia máxima de la señal de audio y preferentemente igual a 1/6 de la frecuencia máxima de la señal de audio.

La Fig. 3a ilustra una representación esquemática de la señal 102 de audio que presenta información útil en la secuencia de bloques 300, 301, 302, 303 en que, por motivos de ilustración, el bloque 301 se toma como primer bloque de los valores de muestreo y se considera que el bloque 302 es un segundo bloque posterior de los valores de muestreo de la señal de audio. El bloque 300 precede al primer bloque 301 en el tiempo y el bloque 303 sigue al bloque 302 en el tiempo y el primer bloque 301 y el segundo bloque 302 son adyacentes entre sí en el tiempo. Más aún, como se ilustra en 106 en la Fig. 3a, cada bloque tiene información lateral asociada al mismo 106 que comprende, en el caso del primer bloque 301, la primera información 311 de ganancia y que comprende, en el caso del segundo bloque, una segunda información 312 de ganancia.

La Fig. 3b ilustra un procesamiento del extractor 110 de banda (y el procesador 120 de banda alta y el combinador 130) en bloques superpuestos. De esa manera, la ventana 313 usada para calcular para calcular el primer bloque 301 se traslapa con la ventana 314 usada para extraer el segundo bloque 302 y ambas ventanas 313 y 314 se superponen dentro de un rango 321 de superposición.

Si bien la escala de las Figs. 3a y 3b señalan que la longitud de cada bloque tiene la mitad de tamaño de longitud de una ventana, la situación también puede ser diferente, es decir, que la longitud de cada bloque tiene el mismo tamaño que una ventana usada para incluir en ventana el bloque correspondiente. En realidad, ésta es la implementación preferida para estas realizaciones preferidas subsiguientes ilustradas en la Fig. 4 o, en particular, la Fig. 5a en el caso del postprocesador o la Fig. 9c en el caso del preprocesador.

Luego, la longitud del rango 321 de superposición es la mitad de tamaño de una ventana, correspondiendo a la mitad de tamaño o longitud de un bloque de valores de muestreo.

En particular, se presenta la información de ganancia de alta frecuencia variable en el tiempo para una secuencia de bloques 300 a 303 de valores de muestreo de la señal 102 de audio de manera que el primer bloque 301 de valores de muestreo tiene la primera información 311 de ganancia asociada al mismo y el segundo bloque posterior 302 de valores de muestreo de la señal de audio tiene una segunda información 312 de ganancia diferente, donde el extractor 110 de banda está configurado para extraer, del primer bloque 301 de valores de muestreo, una primera banda de baja frecuencia y una primera banda de alta frecuencia y para extraer, del segundo bloque 302 de valores de muestreo, una segunda banda de baja frecuencia y una segunda banda de alta frecuencia. Asimismo, el procesador 120 de banda alta está configurado para modificar la primera banda de alta frecuencia utilizando la primera información 311 de ganancia para obtener la primera banda de alta frecuencia procesada y para modificar la segunda banda de alta frecuencia usando la segunda información 312 de ganancia para obtener una segunda banda de alta frecuencia procesada. Asimismo, el combinador 130 está configurado, luego, para combinar la primera banda de baja frecuencia y la primera banda de alta frecuencia procesada para obtener un primer bloque combinado y para combinar la segunda banda de baja frecuencia y la segunda banda de alta frecuencia procesada para obtener un segundo bloque combinado.

Como se ilustra en la Fig. 3c, el extractor 110 de banda, el procesador 120 de banda alta y el combinador 130 están configurados para operar con los bloques superpuestos ilustrados en la Fig. 3b. Asimismo, el postprocesador 100 de audio asimismo comprende un sumador 140 de superposición para calcular una porción postprocesada mediante la suma de las muestras de audio de un primer bloque 301 y muestras de audio de un segundo bloque 302 en el rango 321 de superposición de bloques. Preferiblemente, el sumador 140 de superposición está configurado para ponderar muestras de audio de una segunda mitad de un primer bloque empleando una función de fundido de salida o atenuación gradual y para ponderar una primera mitad de un segundo bloque posterior al primer bloque usando una función de fundido de entrada o incremento progresivo. La función fundido de salida y la función de fundido de entrada pueden ser funciones lineales o no lineales que aumentan de manera monótona en el caso de la función de fundido de entrada y reducen de manera monótona en el caso de la función del fundido de salida.

A la salida del sumador 140 de superposición, existe una secuencia de muestras de la señal de audio postprocesada, como por ejemplo, la ilustrada en la Fig. 3a, aunque ahora sin información lateral alguna, ya que la información lateral ha sido “consumida” por el postprocesador 100 de audio.

La Fig. 4 ilustra una implementación preferida del extractor 110 de banda del postprocesador de audio ilustrado en la Fig. 1 o, por otra parte, del extractor 210 de banda del preprocesador 200 de audio de la Fig. 7. Tanto el extractor 110 de banda de la Fig. 1 como el extractor 210 de banda de la Fig. 7 pueden ser implementados de la misma

manera ilustrada en la Fig. 4 o como se ilustra en la Fig. 5a en el caso del postprocesador o en la Fig. 9c en el caso del preprocesador. En una realización, el postprocesador de audio comprende el extractor de banda que tiene, como características particulares, un generador 115 de ventanas de análisis para generar una secuencia de bloques de valores de muestreo de la señal de audio usando una ventana de análisis, donde los bloques están superpuestos en el tiempo como se ilustra en la Fig. 3b en un rango 321 de superposición. Asimismo, el extractor 110 de banda comprende un procesador 116 de DFT para ejecutar una transformada discreta de Fourier a fin de generar una secuencia de bloques de valores espectrales. Por consiguiente, cada bloque individual de valores de muestreo se convierte a una representación espectral que es un bloque de valores espectrales. Por lo tanto, se genera el mismo número de bloques de valores espectrales como si fueran bloques de valores de muestreo.

El procesador 116 de DFT tiene una salida conectada a una entrada de un conformador 117 de paso bajo. El conformador 117 de paso bajo lleva a cabo, en realidad, la acción de filtrado de paso bajo, y la salida del conformador 117 de paso bajo está conectada a un procesador 118 de DFT inversa para generar una secuencia de bloques de valores de muestreo de paso bajo en el dominio del tiempo. Por último, se incluye un generador 119 de ventanas de síntesis a una salida del procesador de DFT inversa para incluir en ventanas la secuencia de bloques de valores de muestreo de paso bajo en el dominio del tiempo usando una ventana de síntesis. La salida del generador 119 de ventanas de síntesis es una señal de paso bajo en el dominio del tiempo. Por consiguiente, los bloques 115 a 119 corresponden al bloque 111 de “filtro de paso bajo” de la Fig. 2, y los bloques 121 y 113 corresponden al “restador” 113 de la Fig. 2. Por consiguiente, en la realización ilustrada en la Fig. 4, el extractor de banda comprende además el generador 121 de ventanas para la señal de audio para incluir en ventanas la señal 102 de audio usando la ventana de análisis y la ventana de síntesis para obtener una secuencia de valores de señal de audio de los bloques incluidos en ventanas. En particular, el generador 121 de ventanas para la señal de audio está sincronizado con el generador 115 de ventanas de análisis y/o el generador 119 de ventanas de síntesis por lo que la secuencia de bloques de valores de muestreo de paso bajo en el dominio del tiempo emitida por el generador 119 de ventanas de síntesis está temporalmente sincronizada con la secuencia de valores de señal de audio de los bloques incluidos en ventanas emitida por el bloque 121, que es la señal de banda completa.

Sin embargo, la señal de banda completa se incluye ahora en ventana utilizando el generador 121 de ventanas para la señal de audio y, por lo tanto, se lleva a cabo una resta muestra por muestra mediante el restador 113 muestra por muestra de la Fig. 4 para obtener en última instancia la señal de paso alto. De esa manera, la señal de paso alto está disponible, además, en una secuencia de bloques, ya que se ha realizado la resta 113 muestra por muestra por cada bloque.

Asimismo, el procesador 120 de banda alta está configurado para aplicar la modificación a cada muestra de cada bloque de la secuencia de bloques de valores de muestreo de paso alto en el dominio del tiempo generados por el bloque 110 de la Fig. 3c. Preferiblemente, la modificación de una muestra de un bloque depende, una vez más, de la información de un bloque anterior y, una vez más, la información del bloque actual, o, por otro lado, también, la información del bloque actual y, otra vez, la información del bloque siguiente. En particular, y de preferencia, la modificación es efectuada por un multiplicador 125 de la Fig. 5a y la modificación es precedida por una corrección de interpolación el bloque 124. Como se ilustra en la Fig. 5a, la corrección de interpolación se realiza entre los valores de ganancia precedentes g[k-1], g[k] y, una vez más, el factor g[k+1] del siguiente bloque posterior al bloque actual.

Asimismo, como se señalará, el multiplicador 125 se controla mediante un bloque de compensación de ganancia 126 que es controlado, por un lado, por el factor 500 beta y, por el otro, por el factor 104 de ganancia g[k] en el caso del bloque actual. En particular, se utiliza el factor beta para calcular la modificación real aplicada por el multiplicador 125 indicada como 1/gc[k] del factor de ganancia g[k] asociado al bloque actual.

Por consiguiente, el factor beta representa una atenuación adicional de los transitorios que es modelada aproximadamente por este factor beta, donde esta atenuación adicional de los eventos transitorios es un efecto secundario de un codificador o un decodificador que opera antes que el postprocesador ilustrado en la Fig. 5a.

El preprocesamiento y el postprocesamiento se aplican dividiendo la señal de entrada en una parte de paso bajo (LP, por sus siglas en inglés) y una parte de paso alto (HP, por sus siglas en inglés). Esto se puede lograr: a) usando la FFT para computar la parte LP o la parte HP, b) usando un filtro FIR (por sus siglas en inglés, Finite Impulse Response, Respuesta Finita al Impulso) de fase cero para computar la parte LP o la parte HP, o c) usando un filtro IIR (por sus siglas en inglés, Infinite Impulse Response, Respuesta Infinita al Impulso) aplicado en ambas direcciones, obteniendo así una fase cero, para computar la parte LP o la parte HP. Dada la parte LP o la parte HP, se puede obtener la otra parte mediante una simple resta en el dominio del tiempo. Se aplica una ganancia escalar dependiente del tiempo a la parte HP, que se vuelve a sumar a la parte LP para crear la salida preprocesada o postprocesada.

División de la señal en una parte LP y una parte HP usando FFT (Figs. 5a, 9c)

En la implementación propuesta, se utiliza la FFT para computar la parte LP. Digamos que el tamaño de

transformación por FFT es N, en particular $N = 128$. La señal de entrada s se divide en bloques de tamaño N, que están superpuestos en la mitad, produciendo los bloques de entrada $ib[k][i] = s[k \times \frac{N}{2} + i]$, donde k es el índice de bloque e i es la posición de la muestra en el bloque k. Se aplica una ventana $w[i]$ (115, 215) a $ib[k]$, en particular la ventana de seno, que se define como

$$w[i] = \text{seno}\frac{\pi(i + 0.5)}{N}, \text{ en el caso de } 0 \leq i < N,$$

y luego también aplicando FFT (116, 216), se obtienen coeficientes complejos $c[k][f]$ de la siguiente manera

$$c[k][f] = FFT(w[i] \times ib[k][i]), \text{ en el caso de } 0 \leq f \leq \frac{N}{2}.$$

Del lado del codificador (Fig. 9c) (217a), para obtener la parte LP, se aplica una multiplicación por cada elemento (217a) de $c[k][f]$ con la forma de procesamiento $ps[f]$, que consiste en lo siguiente:

$$ps[f] = \begin{cases} 1, \text{ en el caso } 0 \leq f < lp_size \\ 1 - \frac{f - lp_size+1}{tr_size + 1}, \text{ en el caso } lp_size \leq f < lp_size + tr_size \\ 0, \text{ en caso } lp_size + tr_size \leq f \leq \frac{N}{2} \end{cases}$$

El parámetro $lp_size = lastFFTLine[sig] + 1 - transitionWidthLines[sig]$ representa el ancho en líneas de FFT de la región de paso bajo y el parámetro $tr_size = transitionWidthLines[sig]$ representa el ancho en líneas de FFT de la región de transición. La forma del procesamiento propuesto es lineal, aunque se puede utilizar cualquier forma arbitraria.

El bloque de LP $lpb[k]$ se obtiene aplicando IFFT (218) y generación de ventanas (219) una vez más, así:

$$lpb[k][i] = w[i] \times IFFT(ps[f] \times c[k][f]), \text{ en el caso de } 0 \leq i < N.$$

La ecuación expuesta es válida para el codificador/preprocesador de la Fig. 9c. En el caso del decodificador o postprocesador, se utiliza la forma de procesamiento adaptativo rs[f] en lugar de ps[f].

Luego se obtiene el bloque de HP $hpb[k]$ mediante una simple resta (113, 213) en el dominio del tiempo

$$hpb[k][i] = in[k][i] \times w^2[i] - lpb[k][i], \text{ en el caso de } 0 \leq i < N.$$

El bloque de salida $ob[k]$ se obtiene aplicando la ganancia escalar $g[k]$ al bloque de HP como (225) (230)

$$ob[k][i] = lpb[k][i] + g[k] \times hpb[k][i]$$

Por último, se combina el bloque de salida $ob[k]$ usando superposición y suma con el bloque de salida anterior $ob[k-1]$ para crear $\frac{N}{2}$ muestras finales adicionales para la señal de salida pre-procesada o de la siguiente manera

$$o\left[k \times \frac{N}{2} + j\right] = ob[k-1][j + \frac{N}{2}] + ob[k][j], \text{ donde } j = \{0, \cdots, \frac{N}{2} - 1\}.$$

Todo el procesamiento se realiza por separado por cada canal de entrada, que se indexa por sig.

Forma de reconstrucción adaptativa del lado del postprocesamiento (Fig. 5a)

Del lado del decodificador, para lograr una reconstrucción perfecta en la región de transición, se debe recurrir a una forma de reconstrucción adaptativa $rs[f]$ (117b) en la región de transición, en lugar de la forma de procesamiento $ps[f]$ (217b) utilizada del lado del codificador, dependiendo de la forma de procesamiento $ps[f]$ y $g[k]$ según

$$rs[f] = 1 - (1 - ps[f]) \times \frac{g[k]}{1 + (g[k] - 1) \times (1 - ps[f])}$$

En la región de LP, tanto $ps[f]$ como $rs[f]$ son uno, en la región HP tanto $ps[f]$ como $rs[f]$ son cero, sólo difieren en la región de transición. Más aún, cuando $g[k] = 1$, luego se tiene $rs[f] = ps[f]$.

La forma de reconstrucción adaptativa se puede deducir asegurando que la magnitud de una línea de FFT en la región de transición se restablezca después del postprocesamiento, lo que da la relación

$$(\mathrm{ps}[\mathrm{f}] + (1 - \mathrm{ps}[\mathrm{f}]) \times \mathrm{g}[\mathrm{k}]) \times \left(\mathrm{rs}[\mathrm{f}] + (1 - \mathrm{rs}[\mathrm{f}]) \times \frac{1}{\mathrm{g}[\mathrm{k}]}\right) = 1.$$

El procesamiento es similar al lado del preprocesamiento, excepto que se utiliza $\mathrm{rs}[\mathrm{f}]$ en lugar de $\mathrm{ps}[\mathrm{f}]$ según

$$\mathrm{lpb}[\mathrm{k}][\mathrm{i}] = \mathrm{w}[\mathrm{i}] \times \mathrm{IFFT}(\mathrm{rs}[\mathrm{f}] \times \mathrm{c}[\mathrm{k}][\mathrm{f}]), \text{ donde } \mathrm{i} = \{0, \cdots, \mathrm{N} - 1\}$$

y el bloque de salida $\mathrm{ob}[\mathrm{k}][\mathrm{i}]$ se computa usando la inversa de la ganancia escalar $\mathrm{g}[\mathrm{k}]$ según (125)

$$\mathrm{ob}[\mathrm{k}][\mathrm{i}] = \mathrm{lpb}[\mathrm{k}][\mathrm{i}] + \frac{1}{\mathrm{g}[\mathrm{k}]} \times \mathrm{hpb}[\mathrm{k}][\mathrm{i}].$$

Corrección de interpolación (124) del lado del postprocesamiento (Fig. 5a)

La primera mitad de la contribución del bloque de salida k a la salida pre-procesada final está dada por $\mathrm{o}[\mathrm{k} \times \frac{\mathrm{N}}{2} + \mathrm{j}] = \mathrm{ob}[\mathrm{k} - 1][\mathrm{j} + \frac{\mathrm{N}}{2}] + \mathrm{ob}[\mathrm{k}][\mathrm{j}]$, donde $\mathrm{j} = \{0, \cdots, \frac{\mathrm{N}}{2}\}$. Por lo tanto, las ganancias $\mathrm{g}[\mathrm{k} - 1]$ y $\mathrm{g}[\mathrm{k}]$ aplicadas al lado del preprocesamiento se interpolan implícitamente debido a las operaciones de generación de ventanas y superposición y suma. La magnitud de cada línea de FFT en la región HP se multiplica, en efecto, en el dominio del tiempo por el factor de escala $\mathrm{g}[\mathrm{k} - 1] \times \mathrm{w}^2\left[\mathrm{j} + \frac{\mathrm{N}}{2}\right] + \mathrm{g}[\mathrm{k}] \times \mathrm{w}^2[\mathrm{j}]$.

De modo similar, del lado del postprocesamiento, la magnitud de cada línea de FFT en la región HP se multiplica, en efecto, en el dominio del tiempo por el factor de escala

$$\frac{1}{\mathrm{g}[\mathrm{k} - 1]} \times \mathrm{w}^2\left[\mathrm{j} + \frac{\mathrm{N}}{2}\right] + \frac{1}{\mathrm{g}[\mathrm{k}]} \times \mathrm{w}^2[\mathrm{j}].$$

Para obtener una reconstrucción perfecta, el producto de los dos términos anteriores,

$$\mathrm{corr}[\mathrm{j}] = \left(\mathrm{g}[\mathrm{k} - 1] \times \mathrm{w}^2\left[\mathrm{j} + \frac{\mathrm{N}}{2}\right] + \mathrm{g}[\mathrm{k}] \times \mathrm{w}^2[\mathrm{j}]\right) \times \left(\frac{1}{\mathrm{g}[\mathrm{k}-1]} \times \mathrm{w}^2\left[\mathrm{j} + \frac{\mathrm{N}}{2}\right] + \frac{1}{\mathrm{g}[\mathrm{k}]} \times \mathrm{w}^2[\mathrm{j}]\right),$$

que representa la ganancia total en el dominio del tiempo en la posición j por cada línea de FFT en la región HP, debería normalizarse en la primera mitad del bloque de salida k según

$$\mathrm{ob}[\mathrm{k}][\mathrm{j}] = \mathrm{lpb}[\mathrm{k}][\mathrm{j}] + \frac{1}{\mathrm{g}[\mathrm{k}]} \times \mathrm{hpb}[\mathrm{k}][\mathrm{j}] \times \frac{1}{\mathrm{corr}[\mathrm{j}]}.$$

Se puede simplificar y reescribir el valor $\mathrm{corr}[\mathrm{j}]$ de la siguiente manera

$$\mathrm{corr}[\mathrm{j}] = 1 + \left(\frac{\mathrm{g}[\mathrm{k} - 1]}{\mathrm{g}[\mathrm{k}]} + \frac{\mathrm{g}[\mathrm{k}]}{\mathrm{g}[\mathrm{k} - 1]} - 2\right) \times \mathrm{w}^2[\mathrm{j}] \times (1 - \mathrm{w}^2[\mathrm{j}]), \text{ en el caso de } 0 \leq \mathrm{j} < \frac{\mathrm{N}}{2}.$$

La contribución de la segunda mitad del bloque de salida k a la salida pre-procesada final está dada por $\mathrm{o}[(\mathrm{k} + 1) \times \frac{\mathrm{N}}{2} + \mathrm{j}] = \mathrm{ob}[\mathrm{k}][\mathrm{j} + \frac{\mathrm{N}}{2}] + \mathrm{ob}[\mathrm{k} + 1][\mathrm{j}]$, y la corrección de interpolación se puede escribir basándose en las ganancias $\mathrm{g}[\mathrm{k}]$ y $\mathrm{g}[\mathrm{k} + 1]$ como

$$\mathrm{corr}\left[\mathrm{j} + \frac{\mathrm{N}}{2}\right] = 1 + \left(\frac{\mathrm{g}[\mathrm{k}]}{\mathrm{g}[\mathrm{k} + 1]} + \frac{\mathrm{g}[\mathrm{k} + 1]}{\mathrm{g}[\mathrm{k}]} - 2\right) \times \mathrm{w}^2[\mathrm{j}] \times (1 - \mathrm{w}^2[\mathrm{j}]), \text{ en el caso de } 0 \leq \mathrm{j} < \frac{\mathrm{N}}{2}.$$

El valor actualizado de la segunda mitad del bloque de salida k está dado por

$$\mathrm{ob}[\mathrm{k}]\left[\mathrm{j} + \frac{\mathrm{N}}{2}\right] = \mathrm{lpb}[\mathrm{k}]\left[\mathrm{j} + \frac{\mathrm{N}}{2}\right] + \frac{1}{\mathrm{g}[\mathrm{k}]} \times \mathrm{hpb}[\mathrm{k}]\left[\mathrm{j} + \frac{\mathrm{N}}{2}\right] \times \frac{1}{\mathrm{corr}[\mathrm{j} + \frac{\mathrm{N}}{2}]}.$$

Cómputo de ganancia del lado del preprocesamiento (Fig. 9c)

Del lado del preprocesamiento, la parte HP del bloque k, que supuestamente contiene un evento transitorio, se ajusta usando la ganancia escalar $\mathrm{g}[\mathrm{k}]$ a fin de tornarla más similar al fondo en su vecindad. La energía de la parte HP del bloque k está indicada por $\mathrm{hp_e}[\mathrm{k}]$ y la energía promedio del fondo de HP en la vecindad del bloque k ha de ser indicada por $\mathrm{hp_bg_e}[\mathrm{k}]$.

El parámetro $\alpha \in [0,1]$, que controla la cantidad de ajuste se define como

$$g_{\text{float}}[k] = \begin{cases} \dfrac{\alpha \times \text{hp_bg_e}[k] + (1-\alpha) \times \text{hp_e}[k]}{\text{hp_e}[k]}, \text{ cuando hp_e}[k] \geq T_{\text{quiet}} \\ 1, \text{ de lo contrario} \end{cases},$$

Se cuantifica y recorta el valor de $g_{\text{float}}[k]$ al rango admitido por el valor elegido de la opción de configuración extendedGainRange para producir el índice de ganancia gainIdx$[k][sig]$ como

$$g_{\text{idx}} = \lfloor \log_2(4 \times g_{\text{float}}[k]) + 0.5 \rfloor + \text{GAIN_INDEX_0dB},$$

$$\text{gainIdx}[k][sig] = \min(\max(0, g_{\text{idx}}), 2 \times \text{GAIN_INDEX_0dB} - 1).$$

El valor $g[k]$ empleado para el procesamiento es el valor cuantificado, definido del lado del decodificador como

$$g[k] = 2^{\frac{\text{gainIdx}[k][sig] - \text{GAIN_INDEX_0dB}}{4}}.$$

Cuando α es 0, la ganancia tiene el valor $g_{\text{float}}[k] = 1$, por lo tanto, no se realiza ningún ajuste, y cuando α es 1, la ganancia tiene el valor $g_{\text{float}}[k] = \text{hp_bg_e}[k]/\text{hp_e}[k]$, por lo tanto, se procura que la energía ajustada coincida con la energía promedio del fondo. La relación anterior se puede reescribir de la siguiente manera

$$g_{\text{float}}[k] \times \text{hp_e}[k] = \text{hp_bg_e}[k] + (1-\alpha) \times (\text{hp_e}[k] - \text{hp_bg_e}[k]),$$

indicando que la variación de la energía ajustada $g_{\text{float}}[k] \times \text{hp_e}[k]$ alrededor de la correspondiente energía promedio del fondo hp_bg_e$[k]$ se reduce en un factor de $(1-\alpha)$. En el sistema propuesto, se utiliza $\alpha = 0{,}75$, por consiguiente, la variación de la energía HP de cada bloque alrededor de la energía promedio correspondiente del fondo se reduce al 25% de la original.

Compensación de ganancia (126) del lado del postprocesamiento (Fig. 5a)

El codificador y el decodificador de núcleo introducen una atenuación adicional de los eventos transitorios, que se modela aproximadamente introduciendo un paso extra de atenuación, usando el parámetro $\beta \in [0,1]$ dependiendo de la configuración del codificador de núcleo y de las características de señal de la trama, según

$$gc_{\text{float}}[k] = \frac{\beta \times \text{hp_bg_e}[k] + (1-\beta) \times [g_{\text{float}}[k] \times \text{hp_e}[k]]}{\text{hp_e}[k]}$$

que indica que, después de pasar a través del codificador y el decodificador de núcleo, la variación de la energía decodificada $gc_{\text{float}}[k] \times \text{hp_e}[k]$ alrededor de la correspondiente energía promedio del fondo hp_bg_e$[k]$ se reduce a su vez en un factor adicional de $(1-\beta)$.

Usando sólo $g[k]$, α, y β, es posible computar un estimativo de $gc[k]$ del lado del decodificador como

$$gc[k] = (1 + \frac{\beta \times (1-\alpha)}{\alpha}) \times g[k] - \frac{\beta \times (1-\alpha)}{\alpha}$$

El parámetro beta_factor $= \frac{\beta \times (1-\alpha)}{\alpha}$ se cuantifica a betaFactorIdx[sig] y se transmite en forma de información lateral por cada trama. La ganancia compensada $gc[k]$ se puede computar empleando el factor beta ("beta_factor) según

$$gc[k] = (1 + \text{beta_factor}) \times g[k] - \text{beta_factor}$$

Control de meta ganancia (MGC)

Las señales de aplauso de los conciertos en vivo, etc. habitualmente no contienen solo el sonido de golpes de palmas, sino también gritos de la multitud, silbidos pronunciados y golpes con los pies de las audiencias. Con frecuencia, el artista emite un anuncio durante el aplauso o los sonidos (manipulación) de un instrumento se superponen con el aplauso sostenido. En este caso, los métodos existentes de modelado de la envolvente temporal como STP o GES podrían menoscabar estos componentes que no son de aplauso si se activan en el mismo instante de los sonidos interferentes. Por lo tanto, un clasificador de señales garantiza la desactivación durante esas señales. HREP ofrece la característica del denominado Control de meta ganancia (MGC). El MGC se utiliza para relajar de manera satisfactoria el efecto perceptual del procesamiento HREP, evitando la necesidad de una clasificación muy precisa de la señal de entrada. Con el MGC, se pueden manejar los aplausos mezclados con el ambiente y los

sonidos interferentes de todo tipo sin introducir errores de distorsión no deseados.

Como se explicó anteriormente, una realización preferida tiene además un parámetro 807 de control o, por otra parte, el parámetro de control beta_factor indicado en 500 en la Fig. 5a. Por otra parte, o además, los factores alfa o beta individuales antes descritos se pueden transmitir en forma de información lateral adicional, aunque es preferible contar con el único parámetro de control beta_factor que consiste en beta por un lado y alfa por el otro, donde beta es el parámetro entre 0 y 1 y depende de la configuración del codificador de núcleo y también, opcionalmente, de las características de la señal y, además, el factor alfa determina la variación de la energía de una parte de alta frecuencia de cada bloque alrededor de la correspondiente energía promedio del fondo, y alfa también es un parámetro entre 0 y 1. Si el número de transitorios en una trama es muy pequeño, como 1-2, luego TNS puede potencialmente conservarlos mejor y, como resultado de la atenuación adicional por medio del codificador y el decodificador correspondiente a la trama se puede reducir. Por lo tanto, un codificador Avanzado puede reducir ligeramente, de manera correspondiente, el beta_factor para impedir la sobreamplificación.

En otras palabras, el MGC modifica actualmente las ganancias computadas g (indicadas aquí por g_float[k]) usando un parámetro de probabilidades p, como g' = g ^ p, que comprime las ganancias hacia 1 antes de su cuantificación. El parámetro factor beta es un mecanismo adicional para controlar la expansión de las ganancias cuantificadas; sin embargo, la presente implementación utiliza un valor fijo basado en la configuración del codificador de núcleo, como por ejemplo la tasa de bits.

Beta_factor está determinado por β x (1-α)/α y se calcula preferentemente del lado del codificador y se cuantifica, y el índice beta_factor cuantificado betaFactorIdx se transmite en forma de información lateral una vez por trama además de la información de ganancia de alta frecuencia variable en el tiempo g[k].

En particular, el parámetro de control adicional 807 tal como beta o beta_factor 500 tiene una resolución temporal menor que la resolución de la información de ganancia de alta frecuencia variable en el tiempo o el parámetro de control adicional es incluso estacionario en el caso de una configuración de codificador de núcleo o pieza de audio.

Preferiblemente, el procesador de banda alta, el extractor de banda y el combinador operan en bloques superpuestos, donde la superposición varía entre el 40% y el 60% de la longitud del bloque y preferentemente se utiliza un rango 321 de superposición del 50%.

En otras realizaciones o en las mismas realizaciones, la longitud del bloque es de entre 0,8 ms y 5,0 ms.

Asimismo, preferentemente o además, la modificación ejecutada por el procesador 120 de banda alta es un factor de multiplicación dependiente del tiempo aplicado a cada muestra de un bloque en el dominio del tiempo de conformidad con g[k], además de conformidad con el parámetro 500 de control y, además, en línea con la corrección de interpolación descrita en el contexto del bloque 124 de la Fig. 5a.

Asimismo, una frecuencia de corte o límite de la banda de baja frecuencia es de entre 1/8 y 1/3 de una frecuencia máxima de la señal de audio y preferentemente igual a 1/6 de la frecuencia máxima de la señal de audio.

Asimismo, el conformador de paso bajo que consiste en 117b y 117a de la Fig. 5a en la realización preferida está configurado para aplicar la función de modelado rs[f] que depende de la información de ganancia de alta frecuencia variable en el tiempo correspondiente al bloque consiguiente. Ya se ha descrito una implementación preferida de la función de modelado rs[f], aunque también se pueden utilizar funciones alternativas.

Además, de preferencia, la función de modelado rs[f] depende también de una función de modelado ps[f] utilizado en un preprocesador 200 de audio para modificar o atenuar una banda de alta frecuencia de la señal de audio utilizando la información de ganancia de alta frecuencia variable en el tiempo correspondiente al bloque correspondiente. Ya se ha descripto una dependencia específica de rs[f] de ps[f] con respecto a la Fig. 5a, aunque también se pueden utilizar otras dependencias.

Asimismo, como ya se ha expuesto con respecto al bloque 124 de la Fig. 5a, la modificación de una muestra de un bloque depende además de un factor de generación de ventanas aplicado para cierta muestra según lo definido por la función de ventana de análisis o la función ventana de síntesis como ya se ha expuesto, por ejemplo, con respecto al factor de corrección que depende de una función de ventana w[j] y aún más preferentemente de un cuadrado de un factor de ventana w[j].

Como se mencionó anteriormente, especialmente con respecto a la Fig. 3b, el procesamiento ejecutado por el extractor de banda, el combinador y el procesador de banda alta es ejecutado en bloques superpuestos de manera que una porción posterior de un bloque anterior se derive de las mismas muestras de audio de la señal de audio que una porción anterior de un bloque posterior que está adyacente en el tiempo a un bloque anterior, es decir que el procesamiento se realiza dentro y usando el rango 321 de superposición. Este rango 321 de superposición de los

bloques 313 y 314 superpuestos es igual a la mitad del bloque anterior y el bloque posterior tiene la misma longitud que el bloque anterior con respecto a un número de valores de muestreo y el postprocesador comprende además el sumador 140 de superposición para ejecutar la operación de superposición y suma como se ilustra en la Fig. 3c.

En particular, el extractor 110 de banda está configurado para aplicar la pendiente del filtro de partición 111 entre un rango de parada y un rango de pase del filtro de partición a un bloque de muestras de audio, donde está pendiente depende de la información de ganancia de alta frecuencia variable en el tiempo correspondiente al bloque de muestras. Se da una pendiente preferida con respecto a la pendiente rs[f] que depende de la información de ganancia g[k] antes definida y descrita en el contexto de la Fig. 5a, aunque también son útiles otras dependencias.

En términos generales, la información de ganancia de alta frecuencia tiene preferentemente los valores de ganancia g[k] para un bloque actual k, donde la pendiente se incrementa más potentemente para un valor de ganancia más elevado en comparación con un aumento de la pendiente correspondiente a un valor de ganancia más bajo.

La Fig. 6a ilustra una representación más detallada de la información 106 lateral de la Fig. 3. En particular, la información lateral comprende una secuencia 601 de índices de ganancia, información 602 de precisión de ganancia, una información 603 de compensación de ganancia y una información 604 de precisión de compensación.

Preferiblemente, el postprocesador de audio comprende un extractor 610 de información lateral para extraer la señal 102 de audio y la información 106 lateral de una señal de audio con información lateral y la información lateral es enviada a un decodificador 620 de información lateral que genera y calcula una ganancia 621 decodificada y/o un valor 622 de compensación de ganancia decodificado basado en la correspondiente información de precisión de ganancia y la correspondiente información de precisión de compensación.

En particular, la información de precisión determina un número de valores diferentes, donde una elevada información de precisión de ganancia define un mayor número de valores que puede tener el índice de ganancia en comparación con una información de precisión de ganancia más baja que indica un número de valores más bajo que puede tener un valor de ganancia.

Por consiguiente, una información de ganancia de alta precisión puede indicar un número más elevado de bits utilizados para transmitir un índice de ganancia en comparación con la información de ganancia de menor precisión que indica un número menor de bits empleados para transmitir la información de ganancia. La información de alta precisión puede indicar 4 bits (16 valores correspondientes a la información de ganancia) y la información de ganancia menor puede ser de sólo 3 bits (8 valores) para la cuantificación de ganancia. Por lo tanto, la información de precisión de ganancia puede ser, por ejemplo, una simple bandera indicada como “extendedGainRange”. En este último caso, la bandera de configuración extendedGainRange no indica exactitud ni precisión, sino que indica si las ganancias tienen un rango normal o un rango extendido. El rango extendido contiene todos los valores del rango normal y, además, valores más altos que los posibles usando el rango normal. El rango extendido que se puede utilizar en ciertas realizaciones permite potencialmente aplicar un efecto de preprocesamiento más intento para los eventos transitorios fuertes, que de lo contrario serían recortados al rango normal.

De modo similar, en el caso de la precisión del factor beta, es decir, el caso de la información de precisión de compensación de ganancia, se puede utilizar también una bandera, que define si los índices del factor beta utilizan 3 bits o 4 bits, y esta bandera se puede denominar extendedBetaFactorPrecision.

Preferiblemente, el procesador de FFT 116 está configurado para ejecutar una transformada discreta de Fourier bloque por bloque, con una longitud de bloque de N valores de muestreo para obtener un número de valores espectrales que es menor que un número de N/2 valores espectrales complejos mediante la ejecución de un algoritmo de transformada discreta de Fourier disperso, en el cual se omiten los cálculos de ramas correspondientes a valores espectrales por encima de una frecuencia máxima, y el extractor de banda está configurado para calcular la señal de la banda de baja frecuencia usando los valores espectrales hasta un rango de frecuencia de inicio de transición y ponderando los valores espectrales dentro del rango frecuencia de transición, donde el rango de frecuencia de transición sólo se extiende hasta la frecuencia máxima o una frecuencia que es menor que la frecuencia máxima.

Este procedimiento está ilustrado en la Fig. 15, por ejemplo, donde se ilustran ciertas operaciones de mariposa. Se presenta un ejemplo a partir de la topología de FFT de decimación en tiempo de base partida de N=8 puntos, donde sólo se necesita X(0) y X(1) para continuar el procesamiento; en consecuencia, E(2) y E(3) y O(2) y O(3) no son necesarios. A continuación, supongamos que ambas DFTs de N/2 puntos se subdividen a su vez en dos DFT de N/4 puntos y sus posteriores mariposas consiguientes. Ahora se puede repetir la omisión antes descrita de manera análoga a la ilustrada en la Fig. 15.

A continuación, se describe el preprocesador 200 de audio con más detalle con respecto a Fig. 7.

El preprocesador 200 de audio comprende un analizador 260 de señales para analizar la señal 202 de audio a fin de determinar la información 204 de ganancia de alta frecuencia variable en el tiempo.

Además, el preprocesador 200 de audio comprende un extractor 210 de banda para extraer una banda 212 de alta frecuencia de la señal 202 de audio y una banda 214 de baja frecuencia de la señal 202 de audio. Asimismo, se incluye un procesador 220 de banda alta para ejecutar una modificación variable en el tiempo de la banda 212 de alta frecuencia de conformidad con la información 204 de ganancia de alta frecuencia variable en el tiempo para obtener una banda 222 de alta frecuencia procesada.

El preprocesador 200 de audio comprende además un combinador 230 para combinar la banda 222 de alta frecuencia procesada y la banda 214 de baja frecuencia para obtener una señal 232 de audio preprocesada. Además, se incluye una interfaz 250 de salida para generar una señal 252 de salida que comprende la señal 232 de audio preprocesada y la información 204 de ganancia de alta frecuencia variable en el tiempo en forma de información lateral 206 que corresponde a la información 106 lateral descrita en el contexto de la Fig. 3.

Preferiblemente, el analizador 260 de señales está configurado para analizar la señal de audio y determinar una primera característica en un primer bloque 301 de tiempo indicado por el bloque 801 de la Fig. 8a y una segunda característica en un segundo bloque 302 de tiempo de la señal de audio, donde la segunda característica es más transitoria que la primera característica indicada en el bloque 802 de la Fig. 8a.

Asimismo, el analizador 260 está configurado para determinar una primera información 311 de ganancia correspondiente a la primera característica y una segunda información 312 de ganancia correspondiente a la segunda característica como se ilustra en el bloque 803 en la Fig. 8a. Seguidamente, el procesador 220 de banda alta está configurado para atenuar la porción de banda alta del segundo bloque 302 de tiempo de conformidad con la segunda información de ganancia más fuerte que la porción de banda alta del primer bloque 301 de tiempo de conformidad con la primera información de ganancia, como se ilustra en el bloque 804 de la Fig. 8a.

Asimismo, el analizador 260 de señales está configurado para calcular la medida del fondo correspondiente a una energía de fondo de la banda alta de uno o más bloques de tiempo adyacentes en el tiempo colocados antes del bloque de tiempo actual o colocados a continuación del bloque de tiempo actual o colocados antes y a continuación del bloque de tiempo actual o incluyendo el bloque de tiempo actual o excluyendo el bloque de tiempo actual como se ilustra en el bloque 805 de la Fig. 8b. Asimismo, como se ilustra en el bloque 808, una medida de la energía para una banda alta del bloque actual y, como se esboza en el bloque 809, se calcula un factor de ganancia usando la medida del fondo por un lado y la medida de la energía por el otro. Por consiguiente, el resultado del bloque 809 es el factor de ganancia ilustrado en 810 en la Fig. 8b.

Preferiblemente, el analizador 260 de señales está configurado para calcular el factor 810 de ganancia sobre la base de la ecuación ilustrada antes de g_float, aunque también se pueden adoptar otras modalidades de cálculo.

Asimismo, el parámetro alfa influye sobre el factor de ganancia por lo que una variación de la energía de cada bloque alrededor de una energía promedio correspondiente de un fondo se reduce por lo menos 50 % y preferentemente 75 %. Por consiguiente, la variación de la energía de paso alto de cada bloque alrededor de la correspondiente energía promedio del fondo se reduce preferentemente a 25 % de la original por medio del factor alfa.

Además, el bloque de control de meta ganancia /funcionalidad 806 está configurado para generar un factor de control p. En una realización, el bloque de MGC 806 utiliza un método de detección estadística para identificar los transitorios potenciales. Por cada bloque (de, por ej., 128 muestras), produce un factor de “confianza” tipo probabilidad p de entre 0 y 1. La ganancia final que se ha de aplicar al bloque es g' = g ^ p, donde g es la ganancia original. Cuando p es cero, g' = 1, por lo tanto, no se aplica procesamiento alguno, y cuando p es uno, g' = g, se aplica toda la potencia de procesamiento.

Se utiliza el MGC 806 para comprimir las ganancias hacia 1 antes de la cuantificación durante el preprocesamiento, para controlar la potencia del procesamiento entre sin cambios y efecto total. Se utiliza el parámetro beta_factor (que es una parametrización mejorada del parámetro beta) para expandir las ganancias después de la descuantificación durante el postprocesamiento, y una posibilidad consiste en usar un valor fijo para cada configuración del codificador, definido por la tasa de bits.

En una realización, se fija el parámetro alfa en 0,75. Por ende, el factor α es la reducción de la variación de energía alrededor de un fondo promedio, y se lo fija, en la implementación de MPEG-H, en el 75%. El factor de control p de la Fig. 8b sirve como factor de confianza tipo probabilidad p.

Como se ilustra en la Fig. 8c, el analizador de señales está configurado para cuantificar y recortar una secuencia bruta de valores de información de ganancia para obtener la información de ganancia de alta frecuencia variable en

el tiempo como secuencia de valores cuantificados, y el procesador 220 de banda alta está configurado para ejecutar la modificación variable en el tiempo de la banda alta de conformidad con la secuencia de valores cuantificados en lugar de usar valores no cuantificados.

Asimismo, la interfaz 250 de salida está configurada para introducir la secuencia de valores cuantificados en la información lateral 206 como información 204 de ganancia de alta frecuencia variable en el tiempo como se ilustra en la Fig. 8c en el bloque 814.

Más aún, el preprocesador 200 de audio está configurado para determinar 815 otro valor de compensación de ganancia que describe una pérdida de una variación de energía introducida por un codificador o decodificador conectado seguidamente y, además, el preprocesador 200 de audio cuantifica 816 esta información adicional de compensación de ganancia e introduce 817 esta información adicional de compensación de ganancia cuantificada en la información lateral y, además, el analizador de señales está configurado preferentemente para aplicar el Control de meta ganancia en la determinación de la información de ganancia de alta frecuencia variable en el tiempo para reducir gradualmente o aumentar gradualmente un efecto del procesador de banda alta sobre la señal de audio de conformidad con datos 807 de control adicionales.

Preferiblemente, el extractor 210 de banda del preprocesador 200 de audio está implementado de modo más detallado como se ilustra en la Fig. 4, o en la Fig. 9c. Por lo tanto, el extractor 210 de banda está configurado para extraer la banda de baja frecuencia usando un dispositivo de filtro 111 de paso bajo y para extraer una banda de alta frecuencia restando 113 la banda de baja frecuencia de la señal de audio exactamente de la misma manera descrita anteriormente con respecto al dispositivo postprocesador.

Además, el extractor 210 de banda, el procesador 220 de banda alta y el combinador 230 están configurados para operar en bloques solapados o superpuestos. El combinador 230 comprende además un sumador de superposición para calcular una porción postprocesada mediante la suma de las muestras de audio de un primer bloque y muestras de audio de un segundo bloque en el rango de superposición de bloques. Por lo tanto, el sumador de superposición asociado al combinador 230 de la Fig. 7 puede ser implementado de manera igual al sumador de superposición para el postprocesador ilustrado en la Fig. 3c en el número de referencia 130.

En una realización, en el caso del preprocesador de audio, el rango 320 de superposición es de entre el 40% de la longitud de un bloque y el 60% de la longitud de un bloque. En otras realizaciones, la longitud de un bloque es de entre 0,8 ms y 5,0 ms y/o la modificación ejecutada por el procesador 220 de banda alta es un factor de multiplicación aplicado a cada muestra de un bloque en el dominio del tiempo para que el resultado de todo el preprocesamiento sea una señal con naturaleza transitoria reducida.

En una realización adicional, una frecuencia de corte o límite de la banda de baja frecuencia es de entre 1/8 y 1/3 del rango de frecuencia máxima de la señal 202 de audio y preferentemente igual a 1/6 de la frecuencia máxima de la señal de audio.

Como se ilustra, por ejemplo, en la Fig. 9c y como también se ha descrito con respecto al postprocesador de la Fig. 4, el extractor 210 de banda comprende un generador 215 de ventanas de análisis para generar una secuencia de bloques de valores de muestreo de la señal de audio usando una ventana de análisis, donde estos bloques son superpuestos en el tiempo como se ilustra en 321 en la Fig. 3b. Asimismo, se incluye un procesador 216 de transformada discreta de Fourier para generar una secuencia de bloques de valores espectrales y se incluye asimismo un conformador o modelador de paso bajo conectado a continuación 217a, 217b, para modelar cada bloque de valores espectrales para obtener una secuencia de bloques de valores espectrales modelados de paso bajo. Asimismo, se incluye un procesador 218 de transformada discreta de Fourier inversa para generar una secuencia de bloques de valores de muestreo en el dominio del tiempo y se conecta un generador 219 de ventanas de síntesis a una salida del procesador 218 de transformada discreta de Fourier inversa para incluir en ventanas la secuencia de bloques correspondientes a los valores de muestreo de paso bajo en el dominio del tiempo usando una ventana de síntesis.

Preferiblemente, el conformador de paso bajo que consiste en los bloques 217a, 217b aplica la forma de paso bajo ps[f] multiplicando las líneas de FFT individuales como se indica con el multiplicador 217a. La forma de paso bajo ps[f] se calcula según lo indicado anteriormente con respecto a la Fig. 9c.

Además, la señal de audio en sí, es decir, la señal de audio de banda completa también se circunscribe en ventana utilizando el generador 221 de ventanas para la señal de audio a fin de obtener una secuencia de valores de señal de audio de los bloques incluidos en ventanas, donde este generador 221 de ventana para señales de audio está sincronizado con el generador 215 de ventanas de análisis y/o el generador 219 de ventanas de síntesis de tal manera que la secuencia de bloques de valores de muestreo de paso bajo en el dominio del tiempo sea sincrónica con la secuencia de bloques de valores de señales de audio incluidos en ventanas.

Asimismo, el analizador 260 de la Fig. 7 está configurado para incluir además el parámetro 807 de control, usado para controlar la potencia del preprocesamiento entre ningún efecto y el total, y 500, es decir, el factor beta como información lateral adicional, donde el procesador 220 de banda alta está configurado para aplicar la modificación también de conformidad con el parámetro de control adicional 807, donde la resolución temporal del parámetro factor beta es menor que la resolución temporal de la información de ganancia de alta frecuencia variante en el tiempo, o bien el parámetro de control adicional es fijo para una pieza de audio específica. Como se mencionó anteriormente, se utiliza el parámetro de control tipo probabilidad del MGC para comprimir las ganancias hacia 1 antes de la cuantificación, y no se la transmite explícitamente en forma de información lateral.

Asimismo, el combinador 230 está configurado para ejecutar una suma muestra por muestra de los correspondientes bloques de la secuencia de bloques de valores de muestreo de paso bajo en el dominio del tiempo y la secuencia de bloques de valores de muestreo de paso alto en el dominio del tiempo modificados, es decir, procesados, para obtener una secuencia de bloques de valores de señal combinatoria como se ilustra con respecto al lado del postprocesador en la Fig. 3c.

La Fig. 9a ilustra un aparato codificador de audio para codificar una señal de audio que comprende el preprocesador 200 de audio ya expuesto, que está configurado para generar la señal 252 de salida que contiene la información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral. Asimismo, se incluye un codificador 900 de núcleo para generar una señal 902 codificada en el núcleo y una información 904 lateral de núcleo. Además, el aparato codificador de audio comprende una interfaz de salida 910 para generar una señal codificada 912 que comprende la señal 902 codificada en el núcleo, la información 904 lateral de núcleo y la información de ganancia de alta frecuencia variable en el tiempo como información 106 lateral adicional.

Preferiblemente, el preprocesador 200 de audio lleva a cabo un preprocesamiento de cada canal o cada objeto por separado, como se ilustra en la Fig. 10a. En este caso, la señal de audio es una señal multicanal o multiobjeto. En una realización adicional, ilustrada en la Fig. 5c, el preprocesador 200 de audio lleva a cabo un preprocesamiento de cada canal de transporte SAOC (por sus siglas en inglés, Spatial Audio Object Coding, Codificación de Objetos de Audio Espacial) o cada canal de transporte de Ambisonics de Alto Orden (HOA, por sus siglas en inglés) por separado, como se ilustra en la Fig. 10a. En este caso, la señal de audio es un canal de transporte de objetos de audio espacial o un canal de transporte de Ambisonics de Alto Orden.

Por el contrario, el codificador 900 de núcleo está configurado para aplicar un procesamiento del codificador multicanal conjunto o un procesamiento por el codificador multiobjeto conjunto o un relleno de brechas del codificador o un procesamiento de extensión de ancho de banda del codificador en los canales 232 preprocesados.

Por consiguiente, por lo general, la señal 902 codificada en el núcleo tiene menos canales que fueron introducidos en el codificador 900 de núcleo multicanal / multiobjeto conjunto, ya que el codificador 900 de núcleo comprende por lo general un tipo de operación de mezcla descendente.

En la Fig. 9b se ilustra un aparato decodificador de audio. El aparato decodificador de audio tiene una interfaz 920 de entrada de audio para recibir la señal 912 de audio codificada que comprende una señal 902 codificada en el núcleo, información 904 lateral de núcleo y la información 104 de ganancia de alta frecuencia variable en el tiempo como información 106 lateral adicional. Asimismo, el aparato decodificador de audio comprende un decodificador 930 de núcleo para decodificar la señal 902 codificada en el núcleo utilizando la información 904 lateral de núcleo para obtener la señal 102 de núcleo decodificada. Además, el aparato decodificador de audio consta del postprocesador 100 para postprocesar la señal 102 de núcleo decodificada utilizando la información 104 de ganancia de alta frecuencia variable en el tiempo.

Preferiblemente, y como se ilustra en la Fig. 10b, el decodificador 930 de núcleo está configurado para aplicar un procesamiento del decodificador multicanal o un procesamiento del decodificador multiobjeto o un procesamiento del decodificador de extensión de ancho de banda o un procesamiento del decodificador de relleno de brechas para generar canales decodificados de una señal 102 multicanal u objetos 102 decodificados de una señal multiobjeto. Por consiguiente, en otras palabras, el procesador 930 decodificador conjunto comprende por lo general algún tipo de mezcla ascendente a fin de generar, a partir de un número más bajo de canales en la señal 902 de audio codificado, un número más elevado de objetos/ canales individuales. Estos canales/objetos individuales son ingresados en un postprocesamiento de canales individuales por el postprocesador 100 de audio usando la información de ganancia de alta frecuencia variable en el tiempo individual por cada canal o cada objeto como se ilustra en 104 en la Fig. 10b. El postprocesador 100 de canales individuales emite canales postprocesadas que pueden ser transmitidos a un convertidor digital/analógico y seguidamente a altavoces conectados o que pueden ser transmitidos a algún tipo de procesamiento adicional o almacenamiento o cualquier otro procedimiento adecuado para procesar objetos de audio o canales de audio.

La Fig. 10c ilustra una situación similar a lo ilustrado en las Figs. 9a o 9b, es decir, una cadena completa que está compuesta por un preprocesador de procesamiento de envolvente 100 de alta resolución conectado a un codificador

900 para generar un flujo de bits y el flujo de bits es decodificado por el decodificador 930 y la salida del decodificador es postprocesada por el postprocesador de procesamiento de envolvente 100 de alta resolución para generar la señal de salida final.

La Fig. 16 y las Figs. 17a a 17h ilustran resultados de pruebas de escucha correspondientes a una escucha por altavoz de 5.1 canales (128 kbps). Además, se presentan los resultados de una escucha con auriculares de mediana (48 kbps) y alta (128 kbps) calidad. La Fig. 16a resume las configuraciones de las pruebas de escucha. Los resultados están consignados en las Figs. 17a a 17h.

En la Fig. 17a, la calidad perceptual está en el rango de “buena” a “excelente”. Cabe señalar que las señales tipo aplauso están entre los elementos de menor puntuación en el rango “buena”.

La Fig. 17b ilustra que todos los elementos de aplauso exhiben una mejora significativa, en tanto que no se observan cambios significativos en la calidad perceptual en el caso de los ítems que no son de aplauso. Ninguno de los ítems se ha degradado de modo significativo.

En lo que respecta a las Figs. 17c y 17d, se define que la calidad perceptual absoluta está en el rango “buena”. En las diferencias, en general, hay una ganancia significativa de siete puntos. Las ganancias de calidad individual varían entre 4 y 9 puntos, todas las cuales son significativas.

En la Fig. 17e, todas las señales de la serie de pruebas son señales de aplauso. La calidad perceptual está en el rango de “regular” a “buena”. De manera acorde, las condiciones de “HREP” obtienen mejor puntuación que la condición “NOHREP”. En la Fig. 17f, es visible que, en todos los ítems excepto uno, las puntuaciones de “HREP” son significativamente mejores que “NOHREP”. Se observan mejoras en el rango de 3 a 17 puntos. En general, hay una ganancia promedio significativa de 12 puntos. Ninguno de los elementos resulta degradado en forma significativa.

En lo que respecta a las Fig. 17g y 17h, es visible que, en las puntuaciones absolutas, todas las señales están en el rango “excelente”. En las puntuaciones de diferencia se puede apreciar que, si bien la calidad perceptual es casi transparente, en seis de las ocho señales hay una mejora significativa de tres a nueve puntos en total, lo que asciende a una media de cinco puntos MUSHRA. Ninguno de los elementos se degrada de modo significativo.

Los resultados demuestran claramente que la tecnología HREP de las realizaciones preferidas es de significativo valor para la codificación de señales tipo aplauso en un amplio rango de tasas de bits/ calidades absolutas. Más aún, se demuestra que no hay absolutamente ningún desmedro de las señales que no son de aplauso. HREP es una herramienta para la codificación perceptual mejorada que consiste predominantemente en numerosos eventos transitorios densos, tales como aplausos, sonidos de lluvia, etc. Los beneficios de la aplicación de HREP son dobles: HREP relaja la demanda de tasa de bits impuesta sobre el codificador mediante la reducción de la dinámica de corto tiempo de la señal de entrada; además, HREP garantiza el restablecimiento correcto de la envolvente en la etapa de mezcla (ascendente) del decodificador, que es tanto más importante si se han aplicado técnicas de codificación paramétrica multicanal con el códec. Las pruebas subjetivas han demostrado una mejora de alrededor de 12 puntos MUSHRA mediante el procesamiento HREP estéreo a 48 kbps y 7 puntos MUSHRA a 128 kbps y 5.1 canales.

A continuación, se hace referencia a la Fig. 5b que ilustra la implementación del postprocesamiento por un lado o el preprocesamiento por el otro, dentro de una estructura de codificador/ decodificador de audio MPEG-H 3D. Específicamente, la Fig. 5b ilustra el postprocesador HREP 100 implementado dentro de un decodificador de audio MPEG-H 3D. Específicamente, el postprocesador de la invención está indicado en 100 en la Fig. 5b.

Se puede ver que el decodificador HREP está conectado a una salida del decodificador de núcleo de audio 3D ilustrado en 550. Además, entre el elemento 550 y el bloque 100 de la porción superior, se ilustra un elemento Surround MPEG que, por lo general, lleva a cabo una mezcla ascendente implementada en MPEG Surround de los canales base a la entrada del bloque 560 para obtener más canales de salida a la salida del bloque 560.

Además, la Fig. 5b ilustra otros elementos además de la porción de núcleo de audio. Estos son, en la porción de renderización de audio, un drc_1 570 en el caso de los canales por un lado y los objetos por el otro. Asimismo, se incluye un bloque de conversión anterior 580, un renderizador de objetos 590, un decodificador de metadatos de objeto 592, un decodificador 3D SAOC 594 y un decodificador de Ambisonics de Alto Orden (HOA) 596.

Todos estos elementos alimentan un remuestreador 582 y el remuestreador alimenta sus datos de salida a un mezclador 584. El mezclador envía sus canales de salida a una alimentación de altavoces 586 o a una alimentación de auriculares 588, que representan elementos al “final de la cadena” y que representan un postprocesamiento adicional posterior a la salida 584 del mezclador.

La Fig. 5c ilustra otra incorporación preferida del postprocesador de audio (HREP) en la estructura de un

decodificador de audio MPEG-H 3D. A diferencia de la Fig. 5b, el procesamiento HREP se aplica asimismo a los canales de transporte de SAOC y/o a los canales de transporte de HOA. Las demás funcionalidades de la Fig. 5c son similares a los de la Fig. 5b.

Se debe tener en cuenta que las reivindicaciones adjuntas relacionadas con el extractor de banda se aplican para el extractor de banda en el postprocesador de audio y también para el preprocesador de audio, aun cuando sólo se presenta una reivindicación correspondiente a un postprocesador en uno del postprocesador o el preprocesador. Lo mismo es válido con respecto al procesador de banda alta y el combinador.

Se hace especial referencia a las otras realizaciones ilustradas en el Anexo y en el Anexo A.

Si bien la presente invención ha sido descrita en términos de varias realizaciones, el alcance de protección pretendido se define solo siguiendo las reivindicaciones adjuntas.

Si bien se han descrito algunos aspectos en el contexto de un aparato, es obvio que estos aspectos también representan una descripción del método correspondiente, en el cual un bloque o dispositivo corresponde a un paso del método o a una característica de un paso del método. De manera análoga, los aspectos descritos en el contexto de un paso del método también representan una descripción de un bloque o ítem correspondiente o de una característica de un aparato correspondiente. Algunos o todos los pasos del método pueden ser ejecutados por (o utilizando) un aparato de hardware, como por ejemplo, un microprocesador, un ordenador programable o un circuito electrónico. En algunas realizaciones, uno cualquiera o más de las etapas más importantes del método pueden ser ejecutados por ese tipo de aparato.

La señal de audio codificada de la invención puede ser almacenada en un medio de almacenamiento digital o puede ser transmitida por un medio de transmisión tal como un medio de transmisión inalámbrico o un medio de transmisión conectado por cables tal como internet.

Dependiendo de ciertos requisitos de implementación, las formas pueden ser implementadas en hardware o en software o por lo menos parcialmente en software, o por lo menos parcialmente en hardware o por lo menos parcialmente en software. La implementación se puede realizar empleando un medio de almacenamiento digital, por ejemplo, un disco blando, un DVD, un Blu-Ray, un CD, una ROM, una PROM, una EPROM, una EEPROM o una memoria FLASH, que tiene almacenadas en la misma señales control legibles electrónicamente, que cooperan (o tienen capacidad para cooperar) con un sistema de computación programable de tal manera que se ejecute el método respectivo. Por lo tanto, el medio de almacenamiento digital puede ser legible por un ordenador.

Algunas realizaciones de acuerdo con la invención comprenden un transportador de datos no transitorio que comprende señales de control legibles electrónicamente, con capacidad para cooperar con un sistema de computación programable de tal manera que se ejecute uno de los métodos descritos en la presente.

En general, las realizaciones de la presente invención pueden ser implementadas en forma de producto programa de computación con un código de programa, donde el código de programa cumple la función de ejecutar uno de los métodos al ejecutarse el programa de computación en un ordenador. El código de programa puede ser almacenado, por ejemplo, en un portador legible por una máquina.

Otras realizaciones comprenden el programa de computación para ejecutar uno de los métodos aquí descritos, almacenado en un portador legible por una máquina.

En otras palabras, una realización del método de la invención consiste, por lo tanto, en un programa de computación que consta de un código de programa para ejecutar uno de los métodos aquí descritos al ejecutarse el programa de computación en un ordenador.

Otra realización de los métodos de la invención consiste, por lo tanto, en un portador de datos (o medio de almacenamiento digital, o medio legible por ordenador) que comprende, grabado en el mismo, el programa de computación para ejecutar uno de los métodos aquí descritos. El portador de datos, el medio de almacenamiento digital o el medio grabado son por lo general tangibles y/o no transitorios.

Otra realización del método de la invención es, por lo tanto, un flujo de datos o una secuencia de señales que representa el programa de computación para ejecutar uno de los métodos aquí descritos. El flujo de datos o la secuencia de señales puede estar configurada, por ejemplo, para ser transferida a través de una conexión de comunicación de datos, por ejemplo, a través de Internet.

Una realización adicional comprende un medio de procesamiento, por ejemplo, un ordenador, un dispositivo lógico programable, configurado o adaptado para ejecutar uno de los métodos aquí descritos.

Otra realización comprende un ordenador en la que se ha instalado el programa de computación para ejecutar uno de los métodos aquí descritos.

Otra de las realizaciones de acuerdo con la invención comprende un aparato o un sistema configurado para transferir (por ejemplo, por vía electrónica u óptica) un programa de computación para ejecutar uno de los métodos aquí descritos a un receptor. El receptor puede ser, por ejemplo, un ordenador, un dispositivo móvil, un dispositivo de memoria o similar. El aparato o sistema puede comprender, por ejemplo, un servidor de archivos para transferir un programa de computación al receptor.

En algunas realizaciones, se puede utilizar un dispositivo lógico programable (por ejemplo, una matriz de puertas programables en el campo) para ejecutar algunas o todas las funcionalidades de los métodos aquí descritos. En algunas realizaciones, una matriz de puertas programables en el campo puede cooperar con un microprocesador para ejecutar uno de los métodos aquí descritos. Por lo general, los métodos son ejecutados preferentemente por cualquier aparato de hardware.

El aparato descrito en la presente puede ser implementado empleando un aparato de hardware o utilizando un ordenador, o utilizando una combinación de aparato de hardware y un ordenador.

Los métodos aquí descritos se pueden poner en práctica empleando un aparato de hardware o utilizando un ordenador, o utilizando una combinación de aparato de hardware y ordenador.

Las realizaciones antes descritas son meramente ilustrativas de los principios de la presente invención. Se entiende que las modificaciones y variaciones de las disposiciones y los detalles aquí descritos han de resultar obvias para un experto en la técnica. Por lo tanto, sólo se pretende quedar limitados por el alcance de las siguientes reivindicaciones de patente y no por los detalles específicos presentados a manera de descripción y explicación de las realizaciones de la presente.

Anexo

Descripción de una realización adicional de HREP en 3DAudio MPEG-H

El procesamiento de envolvente de alta resolución (HREP) es una herramienta para la codificación perceptual mejorada que consiste predominantemente en numerosos eventos transitorios densos, tales como aplausos, sonidos de lluvia, etc. Históricamente, estas señales han sido muy difíciles de codificar para los códecs de audio de MPEG, especialmente a bajas tasas de bits. Las pruebas subjetivas han demostrado una mejora de alrededor de 12 puntos MUSHRA mediante el procesamiento HREP estéreo a 48 kbps.

Resumen ejecutivo

La herramienta HREP otorga una eficiencia de codificación mejorada en el caso de señales que contienen eventos transitorios densamente espaciados, tales como señales de aplauso, ya que son una parte importante de las grabaciones en vivo. De modo similar, el sonido de las gotas de lluvia u otros sonidos como de fuegos artificiales pueden exhibir esas características. Lamentablemente, esta clase de sonidos presenta dificultades a los códecs de audio existentes, especialmente cuando se los codifica a bajas tasas de bits y/o con herramientas de codificación paramétrica.

La Figura 10c ilustra el flujo de señales en un códec equipado con HREP. Del lado del codificador, la herramienta que actúa como preprocesador que aplana temporalmente la señal para altas frecuencias generando una pequeña cantidad de información lateral (1-4 kbps en el caso de las señales estéreo). Del lado del decodificador, la herramienta que actúa como un postprocesador que modela temporalmente la señal para altas frecuencias, haciendo uso de la información lateral. Los beneficios de la aplicación de HREP son dobles: HREP afloja la demanda de tasa de bits impuesta sobre el codificador mediante la reducción de la dinámica de corto tiempo de la señal de entrada; además, HREP garantiza un restablecimiento correcto de la envolvente en la etapa de mezcla (ascendente) en el decodificador, que es tanto más importante si se han aplicado técnicas de codificación paramétrica multicanal dentro del códec.

Figura 10c: Reseña general del flujo de señales en un códec equipado con HREP.

La herramienta HREP funciona para todas las configuraciones de canales (mono, estéreo, multicanal, incluyendo 3D) y también para objetos de audio.

En el experimento central, presentamos los resultados de las pruebas de escucha MUSHRA, que demuestran el mérito de HREP para la codificación de señales de aplauso. Se demuestra una mejora significativa de la calidad perceptual para los siguientes pasos de pruebas

- Mejora promedio de 7 puntos MUSHRA para 5.1 canales a 128kbit/s
- Mejora promedio de 12 puntos MUSHRA para estéreo de 48kbit/s
- Mejora promedio de 5 puntos MUSHRA para estéreo de 128kbit/s

A título de ejemplo, mediante la evaluación de la calidad perceptual para señales de 5.1can empleado la conocida serie completa de pruebas de MPEG Surround, probamos que la calidad de las señales que no son de aplauso no son dañadas por HREP.

Descripción detallada de HREP

La Figura 10c ilustra el flujo de señales en un códec equipado con HREP. Del lado del codificador, la herramienta actúa como un preprocesador con alta resolución temporal antes del códec de audio perceptual en sí mediante el análisis de la señal de entrada, atenuando, y de esa manera también aplanando temporalmente la parte de alta frecuencia de los eventos transitorios, y generando una pequeña cantidad de información lateral (1-4 kbps en el caso de las señales estéreo). Un clasificador de aplausos puede guiar la decisión del codificador de activar o no HREP. Del lado del decodificador, la herramienta actúa como un postprocesador posterior al códec de audio mediante el refuerzo y, de esa manera, el modelado temporal de la parte de alta frecuencia de los eventos transitorios, haciendo uso de la información lateral que se ha generado durante la codificación.

Figura 9c: Flujo de señales HREP detallado en el codificador.

La Figura 9c exhibe el flujo de señales dentro del procesador HREP. El preprocesamiento se aplica dividiendo la señal de entrada en una parte de paso bajo (LP) y una parte de paso alto (HP). Esto se logra usando FFT para computar la parte LP. Dada la parte LP, se obtiene la parte HP mediante resta en el dominio del tiempo. Se aplica una ganancia escalar dependiente del tiempo a la parte HP, que se vuelve a sumar a la parte LP para crear la salida preprocesada.

La información lateral comprende información de forma de paso bajo (LP) y ganancias escalares que se estiman dentro del bloque de análisis de HREP (no se ilustra). El bloque de análisis HREP puede contener otros mecanismos que reducen satisfactoriamente el efecto del procesamiento HREP sobre el contenido de la señal (“señales que no son de aplauso”) donde HREP no es aplicable por completo. Por consiguiente, se debilitan considerablemente los requisitos de precisión en la detección de aplausos.

Figura 5a: Flujo de señales HREP detallado en el decodificador.

En la Figura 5a se ilustra el procesamiento del lado del decodificador. La información lateral sobre la forma de la HP y ganancias escalares es analizada a partir del flujo de bits (no se ilustra) y aplicada a la señal, asemejándose a un postprocesamiento del decodificador inverso al del preprocesamiento realizado por el codificador. El postprocesamiento se aplica, una vez más, dividiendo la señal de entrada en una parte de paso bajo (LP) y una parte de paso alto (HP). Esto se logra usando FFT para computar la parte LP. Dada la parte LP, se obtiene la parte HP mediante resta en el dominio del tiempo. Se aplica una ganancia escalar dependiente del tiempo a la parte HP, que se vuelve a sumar a la parte LP para crear la salida preprocesada.

Toda la información lateral de HREP es señalizada en una carga útil de extensión y una compatibilidad con versiones anteriores incorporada dentro del flujo de bits de Audio 3D MPEG-H.

Especificación de la memoria

Los cambios de WD necesarios, la sintaxis propuesta del flujo de bits, la semántica y una descripción detallada del proceso de decodificación se pueden encontrar en el Anexo A del documento en forma de texto comparativo.

Complejidad

La complejidad informática del procesamiento HREP está dominado por el cálculo de los pares DFT/IDFT que implementan la división LP/HP de la señal. Por cada trama de audio que comprende 1024 valores en el dominio del tiempo, se tienen que calcular 16 pares de DFT/IDFTs de valor real de 128 puntos.

Para la inclusión en el perfil de baja complejidad (LC, por sus siglas en inglés), proponemos las siguientes restricciones

- Limitación de canales/ objetos HREP activos

- Limitación a los factores de ganancia transmitida máximos g(k) que no son triviales (los factores de ganancia triviales de 0dB mitigan la necesidad de un par DFT/IDFT asociado)
- Cálculo de la DFT/iDFT en una topología dispersa y eficiente de radix 2 o de base partida

Evidencia de mérito

Pruebas de escucha

Como evidencia del mérito, se presentan los resultados de las pruebas de escucha correspondientes a la escucha por altavoces de 5.1 canales (128kbps). Además, se presentan los resultados de la escucha por auriculares estéreo con mediana (48kbps) y alta (128kbps) calidad. La Figura 16 resume las configuraciones de las pruebas de escucha.

Fig. 16 — Reseña general de las pruebas de escucha.

Resultados

128kbps 5.1canales

La Figura muestra las puntuaciones MUSHRA absolutas de la prueba de 128kbps 5.1canales. La calidad perceptual está en el rango de “buena” a “excelente”. Cabe señalar que las señales tipo aplauso están entre los elementos de menor puntuación en el rango “buena”.

Figura 17a: Puntuaciones MUSHRA absolutas de la prueba de 128kbps 5.1 canales.

Figura 17b ilustra las puntuaciones MUSHRA de diferencia de la prueba de 128kbps 5.1 canales. Todos los elementos de aplauso exhiben una mejora significativa, en tanto que no se observan cambios significativos en la calidad perceptual en el caso de los ítems que no son de aplauso. Ninguno de los elementos se ha degradado de modo significativo.

Figura 17b: Puntuaciones MUSHRA de diferencia de la prueba de 128kbps 5.1 canales.

La Figura 17c ilustra las puntuaciones MUSHRA absolutas correspondientes a todos los elementos de aplauso contenidos en la serie de pruebas y la Figura 17d ilustra las puntuaciones MUSHRA de diferencia correspondientes a todos los elementos de aplauso contenidos en la serie de pruebas. La calidad perceptual absoluta está en el rango “buena”. En las diferencias, en general, hay una ganancia significativa de 7 puntos. Las ganancias de calidad individuales están en el rango de entre 4 y 9 puntos y todas son significativas.

Figura 17c: Puntuaciones MUSHRA absolutas de señales de aplauso en la prueba de 128kbps 5.1 canales.

Figura 17d: Puntuaciones MUSHRA de diferencia de señales de aplauso en la prueba de 128kbps 5.1 canales.

48kbps estéreo

La Figura 17e ilustra las puntuaciones MUSHRA absolutas correspondientes a la prueba de estéreo de 48kbps. En este caso, todas las señales de la serie son señales de aplauso. La calidad perceptual está en el rango de “regular” a “buena”. De manera acorde, la condición de “hrep” obtiene mejor puntuación que la condición “noherp”. La Figura 17f ilustra las puntuaciones MUSHRA de diferencia. En todos los ítems excepto uno, las puntuaciones de “hrep” son significativamente mejores que “nohrep”. Se observan mejoras en el rango de 3 a 17 puntos. En general, hay una ganancia promedio significativa de 12 puntos. Ninguno de los elementos resulta degradado en forma significativa

Figura 17e: Puntuaciones MUSHRA absolutas correspondientes a la prueba estéreo de 48kbps.

Figura 17f: Puntuaciones MUSHRA de diferencia de la prueba estéreo de 48kbps.

Estéreo de 128kbps

La Figura 17g y la Figura 17h exhiben las puntuaciones MUSHRA absolutas y de diferencias de la prueba estéreo de 128kbps, respectivamente. En las puntuaciones absolutas, todas las señales reciben puntuaciones en el rango “excelente”. En las puntuaciones de diferencia, se observa que, aunque la calidad perceptual es casi transparente, en el caso de 6 de 8 señales hay una mejora significativa de 3 a 9 puntos, que en general ascienden a una media de 5 puntos MUSHRA. Ninguno de los elementos se degradó de modo significativo.

Figura 17g: Puntuaciones MUSHRA absolutas correspondientes a la prueba estéreo de 128kbps.

Figura17h: Puntuaciones MUSHRA de diferencia de la prueba estéreo de 128kbps.

Los resultados demuestran claramente que la tecnología HREP de la propuesta de la CE es de significativo valor para la codificación de señales tipo aplauso en un amplio rango de tasas de bits/ calidades absolutas. Más aún, se comprueba que no hay disfunción alguna de las señales que no son de aplauso.

Conclusión

HPREP es una herramienta para la codificación perceptual mejorada de señales que, predominantemente, consisten en numerosos eventos transitorios densos, tales como aplauso, sonidos de las gotas de lluvia, etc. Los beneficios de la aplicación de HREP son dobles: HREP afloja la demanda de tasa de bits impuesta sobre el codificador mediante la reducción de la dinámica de corto tiempo de la señal de entrada; además, HREP garantiza un restablecimiento correcto de la envolvente en la etapa de mezcla (ascendente) en el decodificador, lo que es tanto más importante si se han aplicado técnicas de codificación paramétrica multicanal dentro del códec Las pruebas subjetivas han demostrado una mejora de alrededor de 12 puntos MUSHRA por el procesamiento HREP a 48 kbps estéreo, y 7 puntos MUSHRA a 128kbps 5.1 canales.

Anexo A

Realización preferida de HREP dentro de Audio 3D MPEG-H

A continuación, se presentan las modificaciones de datos correspondientes a los cambios necesarios para HREP con respecto a los documentos ISO/IEC 23008-3:2015 e ISO/IEC 23008-3:2015/EAM3.

Se agrega la siguiente línea a la Tabla 1, "Bloques funcionales de MPEG-H 3DA y dominio del procesamiento interno. $f_{s,core}$ denota la frecuencia de muestreo de salida del decodificador de núcleo, $f_{s,out}$ denota la frecuencia de muestreo de salida del decodificador", en la Sección 10.2:

Tabla 1 — Bloques funcionales de MPEG-H 3DA y dominio del procesamiento interno. $f_{s,core}$ denota la frecuencia de muestreo de salida del decodificador de núcleo, $f_{s,out}$ denota la frecuencia de muestreo de salida del decodificador.

Contexto de Procesamiento	Bloque funcional		Dominio de procesamiento	Retardo Muestras [$1/f_{s,core}$] o [$1/f_{s,out}$]	Contribución al Retardo Máximo Alto perfil Muestras [$1/f_{s,out}$]	Contribución al Retardo Máximo Perfil de Baja Complejidad Muestras [$1/f_{s,out}$]
Núcleo de Audio	HREP		TD, longitud trama de núcleo = 1024	64		64 * RSRmax
		Par Síntesis QMF y Análisis QMF y alineamiento a entramado de 64 muestras	FD TD FD	64 + 257 + 320 + 63	(64 + 257 + 320 + 63) * RSR_{max}	

Se agrega el siguiente caso a la Tabla 13, "Sintaxis de mpegh3daExtElementConfig()", en Sección 5.2.2.3:

Tabla 13 - Sintaxis de mpegh3daExtElementConfig()

```
…
        case ID_EXT_ELE_HREP:
                HREPConfig(current_signal_group);
                break;
…
```

Se agrega la siguiente definición de valores a la Tabla 50, "Valor de usacExtElementType", en la Sección 5.3.4:

Tabla 50 - Valor de usacExtElementType

ID_EXT_ELE_HREP	12
/* reservado para uso de ISO */	13 - 127

Se agrega la siguiente interpretación a la Tabla 51, "Interpretación de los bloques de datos para la decodificación de carga útil de extensión", en la Sección 5.3.4:

Tabla 51 - Interpretación de los bloques de datos para la decodificación de carga útil de extensión

ID_EXT_ELE_HREP	HREPFrame(outputFrameLength, current_signal_group)

Se agrega una nueva subcláusula al final de 5.2.2 y se agrega la siguiente Tabla:

5.2.2.X Configuraciones de elementos de extensión

Tabla 2 — Sintaxis de HREPConfig()

Sintaxis	No. de bits	Mnemónico

HREPConfig(current_signal_group)
{
signal_type = signalGroupType[current_signal_group];
signal_count = bsNumberOfSignals[current_signal_group] + 1;
if (signal_type == SignalGroupTypeChannels) {
channel_layout = audioChannelLayout[current_signal_group];
}
 	
extendedGainRange;
extendedBetaFactorPrecision;
 	
for (sig = 0; sig < signal_count; sig++) {
if ((signal_type == SignalGroupTypeChannels) && isLFEChannel(channel_layout, sig)) {
isHREPActive[sig] = 0;
} else {
isHREPActive[sig];
}
if (isHREPActive[sig]) {
if (sig == 0) {
lastFFTLine[0];
transitionWidthLines[0];
defaultBetaFactorIdx[0];
} else {
if (useCommonSettings) {
lastFFTLine[sig] = lastFFTLine[0];
transitionWidthLines[sig] = transitionWidthLines[0];
defaultBetaFactorIdx[sig] = defaultBetaFactorIdx[0];
} else {
lastFFTLine[sig];
transitionWidthLines[sig];
defaultBetaFactorIdx[sig];
}
}
}
}
}
	
	1
	1
	
	NOTE 1
	
	1
	
	NOTE 2
	4
	4
	nBitsBeta
	NOTE 3
	1
	
	4
	4
	nBitsBeta
		
		uimsbf
		uimsbf
		
		uimsbf
		uimsbf
		uimsbf
		
		uimsbf
		
		uimsbf
		uimsbf
		uimsbf

NOTA 1: La función helper isLFEChannel(channel_layout, sig) retorna 1 si el canal en la posición sig in channel_layout es un canal LFE o 0 de lo contrario. NOTA 3: nBitsBeta = 3 + extendedBetaFactorPrecision.		

Al final de 5.2.2.3 se agregan las siguientes Tablas:

Tabla 3 — Sintaxis de HREPFrame()

Sintaxis	No. de bits	Mnemónico
HREPFrame(outputFrameLength, current_signal_group)		

```
{
   gain_count = outputFrameLength / 64;
   signal_count = bsNumberOfSignals[current_signal_group] + 1;

   useRawCoding;                                          1            uimsbf
   if (useRawCoding) {
      for (pos = 0; pos < gain_count; pos++) {
         for (sig = 0; sig < signal_count; sig++) {       NOTE 1
            if (isHREPActive[sig] == 0) continue;
            gainIdx[pos][sig];                          nBitsGain    uimsbf
         }
      }
   } else {
      HREP_decode_ac_data(gain_count, signal_count);
   }

   for (sig = 0; sig < signal_count; sig++) {
      if (isHREPActive[sig] == 0) continue;
      all_zero = 1; /* all gains are zero for the current channel */
      for (pos = 0; pos < gain_count; pos++) {
         if (gainIdx[pos][sig] != GAIN_INDEX_0dB) {
            all_zero = 0;
            break;
         }
      }
      if (all_zero == 0) {
         useDefaultBetaFactorIdx;                       1            uimsbf
         if (useDefaultBetaFactorIdx) {
            betaFactorIdx[sig] = defaultBetaFactorIdx[sig];
         } else {
            betaFactorIdx[sig];                         nBitsBeta    uimsbf
         }
      }
   }
}
```

NOTA 1: nBitsGain = 3 + extendedGainRainge.

La función helper HREP_decode_ac_data(gain_count, signal_count) describe la lectura de los valores de ganancia en la matriz gainIdx usando las siguientes funciones de codificación aritmética de bajo nivel USAC.

arith_decode(*ari_state, cum_freq, cfl),
arith_start_decoding(*ari_state),
arith_done_decoding(*ari_state).

Se introducen dos funciones helper adicionales,

ari_decode_bit_with_prob(*ari_state, count_0, count_total),

que decodifica un bit donde $p_0 = count_0/total_count$ y $p_1 = 1 - p_0$, y ari_decode_bit(*ari_state),

que decodifica un bit sin modelado, donde $p_0 = 0.5$ y $p_1 = 0.5$.

```
                ari_decode_bit_with_prob(*ari_state, count_0, count_total)
                {
                   prob_scale = 1 << 14;
                   tbl[0] = probScale - (count_0 * prob_scale) / count_total;
                   tbl[1] = 0;

   res = arith_decode(ari_state, tbl, 2);
   return res;
}

ari_decode_bit(*ari_state)
{
```

```
    prob_scale = 1 << 14;
    tbl[0] = prob_scale >> 1;
    tbl[1] = 0;
    res = arith_decode(ari_state, tbl, 2);
    return res;
}


HREP_decode_ac_data(gain_count, signal_count)
{
    cnt_mask[2] = {1, 1};
    cnt_sign[2] = {1, 1};
    cnt_neg[2] = {1, 1};
    cnt_pos[2] = {1, 1};

    arith_start_decoding(&ari_state);

    for (pos = 0; pos < gain_count; pos++) {
        for (sig = 0; sig < signal_count; sig++) {
            if (!isHREPActive[sig]) {
                continue;
            }
            mask_bit = ari_decode_bit_with_prob(&ari_state, cnt_mask[0], cnt_mask[0] + cnt_mask[1]);
            cnt_mask[mask_bit]++;

            if (mask_bit) {
                sign_bit = ari_decode_bit_with_prob(&ari_state, cnt_sign[0], cnt_sign[0] + cnt_sign[1]);
                cnt_sign[sign_bit] += 2;

                if (sign_bit) {
                    large_bit = ari_decode_bit_with_prob(&ari_state, cnt_neg[0], cnt_neg[0] + cnt_neg[1]);
                    cnt_neg[large_bit] += 2;
                    last_bit = ari_decode_bit(&ari_state);
                    gainIdx[pos][sig] = -2 * large_bit - 2 + last_bit;
                } else {
                    large_bit = ari_decode_bit_with_prob(&ari_state, cnt_pos[0], cnt_pos[0] + cnt_pos[1]);
                    cnt_pos[large_bit] += 2;
                    if (large_bit) {
                        gainIdx[pos][sig] = 3;
                    } else {
                        last_bit = ari_decode_bit(&ari_state);
                        gainIdx[pos][sig] = 2 - last_bit;
                    }
                }
            } else {
                gainIdx[pos][sig] = 0;
            }

            if (extendedGainRange) {
                prob_scale = 1 << 14;
                esc_cnt = prob_scale / 5;
                tbl_esc[5] = {prob_scale - esc_cnt, prob_scale - 2 * esc_cnt, prob_scale - 3 * esc_cnt, prob_scale - 4 *
esc_cnt, 0};
                sym = gainIdx[pos][sig];
                if (sym <= -4) {
                    esc = arith_decode(ari_state, tbl_esc, 5);
                    sym = -4 - esc;
                } else if (sym >= 3) {
                    esc = arith_decode(ari_state, tbl_esc, 5);
                    sym = 3 + esc;
                }
                gainIdx[pos][sig] = sym;
            }
```

```
            gainIdx[pos][sig] += GAIN_INDEX_0dB;
        }
    }

    arith_done_decoding(&ari_state);
}
```

Se agregan las siguientes nuevas subcláusulas "Herramienta de Procesamiento de Envolvente de Alta Resolución (HREP) 5.5.X" al final de la subcláusula 5.5:

5.5.X Herramienta de procesamiento de envolvente de alta resolución (HREP)

5.5.X.1 Descripción de la herramienta

La herramienta HREP otorga una eficiencia de codificación mejorada en el caso de señales que contienen eventos transitorios densamente espaciados, tales como señales de aplauso, ya que son una parte importante de las grabaciones en vivo. De modo similar, el sonido de las gotas de lluvia u otros sonidos como de fuegos artificiales pueden exhibir esas características. Lamentablemente, esta clase de sonidos presenta dificultades a los códecs de audio existentes, especialmente cuando se los codifica a bajas tasas de bits y/o con herramientas de codificación paramétrica.

La Fig. 5b o 5c ilustra el flujo de señales en un códec equipado con HREP. Del lado del codificador, la herramienta actúa como preprocesador que aplana temporalmente la señal para altas frecuencias generando una pequeña cantidad de información lateral (1-4 kbps en el caso de las señales estéreo). Del lado del decodificador, la herramienta que actúa como un postprocesador que modela temporalmente la señal para altas frecuencias, haciendo uso de la información lateral. Los beneficios de la aplicación de HREP son dobles: HREP afloja la demanda de tasa de bits impuesta sobre el codificador mediante la reducción de la dinámica de corto tiempo de la señal de entrada; además, HREP garantiza un restablecimiento correcto de la envolvente en la etapa de mezcla (ascendente) en el decodificador, lo que es tanto más importante si se han aplicado técnicas de codificación paramétrica multicanal dentro del códec. La herramienta HREP funciona para todas las configuraciones de canales (mono, estéreo, multicanal, incluyendo 3D) y también para objetos de audio.

5.5.X.2 Datos y elementos de ayuda

current_signal_group	El parámetro current_signal_group se basa en el elemento de sintaxis Signals3d()y el elemento de sintaxis mpegh3daDecoderConfig().
signal_type	El tipo de grupo de señales actual, utilizado para diferenciar entre señales de canales y señales de objeto, HOA, y SAOC.
signal_count	El número de señales en el grupo de señales en curso.
channel_layout	En caso de que el grupo de señales en curso contenga señales de canales, contiene las propiedades de los altavoces por cada canal, utilizadas para identificar los altavoces LFE.
extendedGainRange	Indica si los índices de ganancia utilizan 3 bits (8 valores) o 4 bits (16 valores), computados por nBitsGain.
extendedBetaFactorPrecision	Indica si los índices de factor beta utilizan 3 bits o 4 bits, computados por nBitsBeta.
isHREPActive[sig]	Indica si la herramienta está activa para la señal en index sig en el grupo de señales en curso.
lastFFTLine[sig]	La posición de la última línea no cero utilizada en el procedimiento de paso bajo implementado usando FFT.
transitionWidthLines[sig]	El ancho en líneas de la región de transición utilizada en el procedimiento de paso bajo implementado usando FFT.
defaultBetaFactorIdx[sig]	El índice de factor beta por defecto empleado para modificar las ganancias en el procedimiento de compensación de ganancia.

outputFrameLength	El número equivalente de muestras por trama, usando la frecuencia de muestreo original, como se define en la norma de USAC.
gain_count	El número de ganancias por señal en una trama.
useRawCoding	Indica si los índices de ganancia son codificados en bruto, usando nBitsGain en cada caso, o si se los codifica usando codificación aritmética.
gainIdx[pos][sig]	El índice de ganancia correspondiente al bloque en la posición pos de la señal en la posición sig en el grupo de señales en curso. Si extendedGainRange = 0, los valores posibles están en el rango {0, …, 7}, y si extendedGainRange = 1, los valores posibles están en el rango {0, …, 15}.
GAIN_INDEX_0dB	El índice de ganancia que corresponde a 0 dB, donde se utiliza un valor de 4 si extendedGainRange = 0, y donde se utiliza un valor de 8 si extendedGainRange = 1. Los índices de ganancia se transmiten en forma de valores sin signo mediante la suma de GAIN_INDEX_0dB a sus rangos de datos originales con signo.
all_zero	Indica si todos los índices de ganancia contenidos en una trama correspondiente a la señal actual están teniendo el valor GAIN_INDEX_0dB.
useDefaultBetaFactorIdx	Indica si el índice de factor beta correspondiente a la señal actual tiene el valor por defecto estipulado por defaultBetaFactor[sig].
betaFactorIdx[sig]	El índice de factor beta utilizado para modificar las ganancias en el procedimiento de compensación de ganancia.

5.5.X.2.1 Limitaciones del perfil de baja complejidad

Si el número total de señales contadas en todos los grupos de señales es 6 como máximo no hay limitaciones.

De lo contrario, si el número total de señales donde HREP está activo, indicado por el elemento de sintaxis isHREPActive[sig] en HREPConfig(), y contado en todos los grupos de señales es como máximo 4, no hay más limitaciones.

De lo contrario, si el número total de señales donde HREP está activo, indicado por el elemento de sintaxis isHREPActive[sig] en HREPConfig(), y contado en todos los grupos de señales se ha de limitar a un máximo de 8.

Además, por cada trama, el número total de índices de ganancia que son diferentes de GAIN_INDEX_0dB, contado tomando en cuenta las señales en que HREP está activo y en todos los grupos de señales, ha de ser a lo sumo $4 \times$ gain_count. En el caso de los bloques que tienen un índice de ganancia igual a GAIN_INDEX_0dB, se han de omitir la FFT, la corrección de interpolación, y la IFFT. En este caso, se debe multiplicar la entrada del bloque por el cuadrado de la ventana de seno y utilizarlo directamente en el procedimiento de superposición y suma.

5.5.X.3 Proceso de decodificación

5.5.X.3.1 General

En el elemento de sintaxis mpegh3daExtElementConfig() el campo usacExtElementPayloadFrag ha de ser cero en el caso de un elemento ID_EXT_ELE_HREP. La herramienta HREP es aplicable sólo a grupos de señales del tipo SignalGroupTypeChannels y SignalGroupTypeObject, definidos por SignalGroupType[grp] en el elemento de sintaxis Signals3d(). Por lo tanto, los elementos ID_EXT_ELE_HREP han de estar presentes solo en el caso de grupos de señales del tipo SignalGroupTypeChannels y SignalGroupTypeObject.

El tamaño del bloque y el correspondiente tamaño de FFT utilizado es $\mathrm{N} = 128$.

La totalidad del procesamiento se realiza de modo independiente en cada señal del grupo de señales en curso. Por lo tanto, para simplificar la notación, sólo se describe el proceso de decodificación para una señal en la posición sig.

Figura 5a: Diagrama de bloques de la herramienta de procesamiento de envolvente de alta resolución (HREP) del lado de la decodificación

5.5.X.3.2 Decodificación de factores beta cuantificados

Se deben utilizar las siguientes tablas de búsqueda para convertir el índice de factor beta betaFactorIdx[sig] al factor beta beta_factor, dependiendo del valor de extendedBetaFactorPrecision.

```
tab_beta_factor_dequant_coarse[8] = {
   0.000f, 0.035f, 0.070f, 0.120f, 0.170f, 0.220f, 0.270f, 0.320f
}

tab_beta_factor_dequant_precise[16] = {
   0.000f, 0.035f, 0.070f, 0.095f, 0.120f, 0.145f, 0.170f, 0.195f,
   0.220f, 0.245f, 0.270f, 0.295f, 0.320f, 0.345f, 0.370f, 0.395f
}
```

Si extendedBetaFactorPrecision = 0, la conversión se computa según
beta_factor = tab_beta_factor_dequant_coarse[betaFactorIndex[sig]]
Si extendedBetaFactorPrecision = 1, la conversión se computa según
beta_factor = tab_beta_factor_dequant_precise[betaFactorIndex[sig]]

5.5.X.3.3 Decodificación de ganancias cuantificadas

Una trama es procesada como bloques de gain_count que consisten en N muestras cada uno, que se traslapan en la mitad. Se derivan las ganancias escalares por cada bloque, basándose en el valor de extendedGainRange.

$$g[k] = 2^{\frac{\text{gainIdx[k][sig]} - \text{GAIN_INDEX_0dB}}{4}}, \text{ en el caso } 0 \le k < \text{gain_count}$$

5.5.X.3.4 Cómputo de la parte LP y la parte HP

La señal de entrada s se divide en bloques de tamaño N, que están superpuestos por la mitad, produciendo los bloques de entrada $ib[k][i] = s[k \times \frac{N}{2} + i]$, donde k es el índice de bloque e i es la posición de las muestras en el bloque k. Se aplica una ventana $w[i]$ a $ib[k]$, en particular la ventana de seno, que se define como

$$w[i] = \sin\frac{\pi(i + 0.5)}{N}, \text{ en el caso de } 0 \le i < N,$$

y después también aplicando FFT, se obtienen los coeficientes complejos $c[k][f]$ de la siguiente manera

$$c[k][f] = \text{FFT}(w[i] \times ib[k]), \text{ en el caso de } 0 \le f \le \frac{N}{2}.$$

Del lado del codificador, para obtener la parte LP, aplicamos una multiplicación por elemento de $c[k]$ con la forma de procesamiento $ps[f]$, que consiste en lo siguiente:

$$ps[f] = \begin{cases} 1, \text{ en el caso } 0 \le f < \text{lp_size} \\ 1 - \frac{f - \text{lp_size}+1}{\text{tr_size} + 1}, \text{ si lp_size} \le f < \text{lp_size} + \text{tr_size} \\ 0, \text{ si lp_size} + \text{tr_size} \le f \le \frac{N}{2} \end{cases}$$

El parámetro $\text{lp_size} = \text{lastFFTLine}[sig] + 1 - \text{transitionWidthLines}[sig]$ representa el ancho en líneas de FFT de la región de paso bajo y el parámetro $\text{tr_size} = \text{transitionWidthLines}[sig]$ representa el ancho en líneas FFT de la región de transición.

Del lado del decodificador, para obtener una reconstrucción perfecta en la región de transición, se debe emplear una forma de reconstrucción adaptativa $rs[f]$ en la región de transición, en lugar de la forma de procesamiento $ps[f]$ usada del lado del codificador, dependiendo de la forma de procesamiento $ps[f]$ y $g[k]$ según

$$rs[f] = 1 - (1 - ps[f]) \times \frac{g[k]}{1 + (g[k] - 1) \times (1 - ps[f])}$$

El bloque de LP $lpb[k]$ se obtiene aplicando IFFT y circunscripción en ventanas nuevamente según

$$lpb[k][i] = w[i] \times \text{IFFT}(rs[f] \times c[k][f]), \text{ en el caso de } 0 \le i < N,$$

A continuación se obtiene el bloque de HP hpb[k] mediante simple resta en el dominio del tiempo según

$$\text{hpb}[k][i] = \text{in}[k][i] \times w^2[i] - \text{lpb}[k][i], \text{ en el caso de } 0 \le i < N.$$

5.5.X.3.5 Cómputo de la corrección de interpolación

Las ganancias $g[k-1]$ y $g[k]$ aplicadas del lado del codificador a los bloques en las posiciones $k-1$ y k se interpolan implícitamente debido a las operaciones de generación de ventanas y superposición y suma. Para obtener una reconstrucción perfecta en la parte HP por encima de la región de transición, se necesita una corrección del factor de interpolación, según

$$\text{corr}[j] = 1 + \left(\frac{g[k-1]}{g[k]} + \frac{g[k]}{g[k-1]} - 2\right) \times w^2[j] \times (1 - w^2[j]), \text{ si } 0 \le j < \frac{N}{2}.$$

$$\text{corr}\left[j + \frac{N}{2}\right] = 1 + \left(\frac{g[k]}{g[k+1]} + \frac{g[k+1]}{g[k]} - 2\right) \times w^2[j] \times (1 - w^2[j]), \text{ si } 0 \le j < \frac{N}{2}.$$

5.5.X.3.6 Cómputo de las ganancias compensadas

El codificador de núcleo y el decodificador introducen una atenuación adicional de eventos transitorios, que se compensa ajustando las ganancias $g[k]$ usando beta_factor anteriormente computado, según

$$gc[k] = (1 + \text{beta_factor})g[k] - \text{beta_factor}$$

5.5.X.3.7 Cómputo de la señal de salida

Basándose en $gc[k]$ y $\text{corr}[i]$, se computa el valor del bloque de salida ob[k] de la siguiente manera

$$\text{ob}[k][i] = \text{lpb}[k][i] + \frac{1}{gc[k]} \times \frac{1}{\text{corr}[i]} \times \text{hpb}[k][i], \text{ si } 0 \le i < N$$

Por último, se computa la señal de salida usando los bloques de salida por medio de superposición y suma según

$$o\left[k \times \frac{N}{2} + j\right] = \text{ob}[k-1]\left[j + \frac{N}{2}\right] + \text{ob}[k][j], \text{ si } 0 \le j < \frac{N}{2}$$

$$o\left[(k+1) \times \frac{N}{2} + j\right] = \text{ob}[k]\left[j + \frac{N}{2}\right] + \text{ob}[k+1][j], \text{ si } 0 \le j < \frac{N}{2}$$

5.5.X.4 Descripción del codificador (informativa)

Figura 9c: Diagrama de bloques de la herramienta de procesamiento de envolvente de alta resolución (HREP) del lado de la codificación

5.5.X.4.1 Cómputo de las ganancias y del factor beta

Del lado del preprocesamiento, se ajusta la parte HP del bloque k, que se supone contiene un evento transitorio, usando la ganancia escalar $g[k]$ para tornarla más similar al fondo en sus proximidades. La energía de la parte HP del bloque k está indicada por hp_e[k] y la energía promedio del fondo HP en las proximidades del bloque k se indica por hp_bg_e[k].

Definimos el parámetro $\alpha \in [0, 1]$, que controla la cantidad de ajuste, como :

$$g_{\text{float}}[k] = \begin{cases} \dfrac{\alpha \times \text{hp_bg_e}[k] + (1-\alpha) \times \text{hp_e}[k]}{\text{hp_e}[k]}, \text{ cuando hp_e}[k] \ge T_{\text{quiet}}, \\ 1, \text{ de lo contrario} \end{cases}$$

Se cuantifica y recorta el valor de $g_{\text{float}}[k]$ al rango permitido por el valor elegido de la opción de configuración extendedGainRange para producir el índice de ganancia gainIdx[k][sig] según

$$g_{\text{idx}} = \lfloor \log_2(4 \times g_{\text{float}}[k]) + 0.5 \rfloor + \text{GAIN_INDEX_0dB},$$

$$\text{gainIdx}[k][sig] = \min(\max(0, g_{\text{idx}}), 2 \times \text{GAIN_INDEX_0dB} - 1).$$

El valor $g[k]$ usado para el procesamiento es el valor cuantificado, definido del lado del decodificador según

$$g[k] = 2^{\frac{\text{gainIdx[k][sig]} - \text{GAIN_INDEX_0dB}}{4}}.$$

Cuando α es 0, la ganancia tiene el valor $g_{float}[k] = 1$, por lo tanto no se realiza ajuste alguno, y cuando α es 1, la ganancia tiene el valor $g_{float}[k] = hp_bg_e[k]/hp_e[k]$, por lo tanto se hace que la energía ajustada coincida con la energía promedio del fondo. Podemos reescribir la relación expuesta de la siguiente manera

$$g_{float}[k] \times hp_e[k] = hp_bg_e[k] + (1 - \alpha) \times (hp_e[k] - hp_bg_e[k]),$$

Indicando que la variación de la energía ajustada $g_{float}[k] \times hp_e[k]$ alrededor de la correspondiente energía promedio del fondo $hp_bg_e[k]$ se reduce en un factor de $(1 - \alpha)$. En el sistema propuesto, se utiliza $\alpha = 0.75$, por consiguiente, la variación de la energía de HP de cada bloque alrededor de la correspondiente energía promedio del fondo se reduce a 25 % de la original.

El codificador y el decodificador de núcleo introducen una atenuación adicional de los eventos transitorios, que se modela aproximadamente introduciendo un paso de atenuación extra, usando el parámetro $\beta \in [0, 1]$ dependiendo de la configuración del codificador de núcleo y las características de la señal de la trama, según

$$gc_{float}[k] = \frac{\beta \times hp_bg_e[k] + (1 - \beta) \times [g_{float}[k] \times hp_e[k]]}{hp_e[k]}$$

Indicando que, después de pasar a través del codificador y el decodificador de núcleo, la variación de la energía decodificada $gc_{float}[k] \times hp_e[k]$ alrededor de la correspondiente energía promedio del fondo $hp_bg_e[k]$ se reduce aún más con un factor adicional de $(1 - \beta)$.

Usando sólo $g[k]$, α, y β, es posible computar un estimativo de $gc[k]$ del lado del decodificador de acuerdo con

$$gc[k] = (1 + \frac{\beta \times (1 - \alpha)}{\alpha}) \times g[k] - \frac{\beta \times (1 - \alpha)}{\alpha}$$

El parámetro $beta_factor = \frac{\beta \times (1 - \alpha)}{\alpha}$ es cuantificado a betaFactorIdx[sig] y transmitido en forma de información lateral por cada trama. La ganancia compensada $gc[k]$ se puede computar usando beta_factor según

$$gc[k] = (1 + \text{beta_factor}) \times g[k] - \text{beta_factor}$$

5.5.X.4.2 Cómputo de la parte LP y la parte HP

El procesamiento es idéntico al que corresponde del lado del decodificador definido anteriormente, excepto que se utiliza la forma de procesamiento $ps[f]$ en lugar de la forma de reconstrucción adaptativa $rs[f]$ en el cómputo del bloque de LP $lpb[k]$, que se obtiene aplicando IFFT y la generación de ventanas una vez más según

$$lpb[k][i] = w[i] \times \text{IFFT}(ps[f] \times c[k][f]), \text{ si } 0 \leq i < N.$$

5.5.X.4.3 Cómputo de la señal de salida

Basándose en $g[k]$, se computa el valor del bloque de salida $ob[k]$ según

$$ob[k][i] = lpb[k][i] + g[k] \times hpb[k][i], \text{ si } 0 \leq i < N.$$

De manera idéntica al lado del decodificador, se computa la señal de salida usando los bloques de salida empleando superposición y suma de acuerdo con

$$o\left[k \times \frac{N}{2} + j\right] = ob[k-1]\left[j + \frac{N}{2}\right] + ob[k][j], \text{ si } 0 \leq j < \frac{N}{2},$$
$$o\left[(k+1) \times \frac{N}{2} + j\right] = ob[k]\left[j + \frac{N}{2}\right] + ob[k+1][j], \text{ si } 0 \leq j < \frac{N}{2}.$$

5.5.X.4.4 Codificación de las ganancias usando codificación aritmética

La función helper HREP_encode_ac_data(gain_count, signal_count) describe la escritura de los valores de ganancia a partir de la matriz gainIdx usando las siguientes funciones de codificación aritmética de bajo nivel de USAC:

arith_encode(*ari_state, symbol, cum_freq),

arith_encoder_open(*ari_state),
arith_encoder_flush(*ari_state).

Se introducen dos funciones helper adicionales,
ari_encode_bit_with_prob(*ari_state, bit_value, count_0, count_total),

que codifica el valor de un bit bit_value con $p_0 = \text{count_0/total_count}$ y $p_1 = 1 - p_0$, y ari_encode_bit(*ari_state, bit_value),

que codifica el valor de un bit bit_value sin modelado, con $p_0 = 0.5$ y $p_1 = 0.5$.

```
ari_encode_bit_with_prob(*ari_state, bit_value, count_0, count_total)
{
   prob_scale = 1 << 14;
   tbl[0] = prob_scale - (count_0 * prob_scale) / count_total;
   tbl[1] = 0;
   arith_encode(ari_state, bit_value, tbl);
}

ari_encode_bit(*ari_state, bit_value)
{
   prob_scale = 1 << 14;
   tbl[0] = prob_scale >> 1;
   tbl[1] = 0;
   ari_encode(ari_state, bit_value, tbl);
}

HREP_encode_ac_data(gain_count, signal_count)
{
   cnt_mask[2] = {1, 1};
   cnt_sign[2] = {1, 1};
   cnt_neg[2] = {1, 1};
   cnt_pos[2] = {1, 1};

   arith_encoder_open(&ari_state);

   for (pos = 0; pos < gain_count; pos++) {
      for (sig = 0; sig < signal_count; sig++) {
         if (!isHREPActive[sig]) {
            continue;
         }

        sym = gainIdx[pos][sig] - GAIN_INDEX_0dB;
         if (extendedGainRange) {
            sym_ori = sym;
            sym = max(min(sym_ori, GAIN_INDEX_0dB / 2 - 1), -GAIN_INDEX_0dB / 2);
         }

         mask_bit = (sym != 0);
         arith_encode_bit_with_prob(ari_state, mask_bit, cnt_mask[0], cnt_mask[0] + cnt_mask[1]);
         cnt_mask[mask_bit]++;

         if (mask_bit) {
            sign_bit = (sym < 0);
            arith_encode_bit_with_prob(ari_state, sign_bit, cnt_sign[0], cnt_sign[0] + cnt_sign[1]);
            cnt_sign[sign_bit] += 2;

            if (sign_bit) {
               large_bit = (sym < -2);
               arith_encode_bit_with_prob(ari_state, large_bit, cnt_neg[0], cnt_neg[0] + cnt_neg[1]);
               cnt_neg[large_bit] += 2;
               last_bit = sym & 1;
               arith_encode_bit(ari_state, last_bit);
            } else {
```

```
                large_bit = (sym > 2);
                arith_encode_bit_with_prob(ari_state, large_bit, cnt_pos[0], cnt_pos[0] + cnt_pos[1]);
                cnt_pos[large_bit] += 2;
                if (large_bit == 0) {
                    last_bit = sym & 1;
                    ari_encode_bit(ari_state, last_bit);
                }
            }
        }

        if (extendedGainRange) {
            prob_scale = 1 << 14;
            esc_cnt = prob_scale / 5;
            tbl_esc[5] = {prob_scale - esc_cnt, prob_scale - 2 * esc_cnt, prob_scale - 3 * esc_cnt, prob_scale - 4 *
esc_cnt, 0};
            if (sym_ori <= -4) {
                esc = -4 - sym_ori;
                arith_encode(ari_state, esc, tbl_esc);
            } else if (sym_ori >= 3) {
                esc = sym_ori - 3;
                arith_encode(ari_state, esc, tbl_esc);
            }

        }
    }

    arith_encode_flush(ari_state);
}
```

REIVINDICACIONES

1. Postprocesador (100) de audio para post-procesar una señal (102) de audio que contiene información (104) de ganancia de alta frecuencia variable en el tiempo en forma de información (106) lateral, que comprende:

un extractor (110) de banda para extraer una banda (112) de alta frecuencia de la señal (102) de audio y una banda (114) de baja frecuencia de la señal (102) de audio;

un procesador (120) de banda alta para ejecutar una amplificación variable en el tiempo de la banda (112) de alta frecuencia de acuerdo con la información (104) de ganancia de alta frecuencia variable en el tiempo para obtener una banda (122) de alta frecuencia procesada;

un combinador (130) para combinar la banda (122) de alta frecuencia procesada y la banda (114) de baja frecuencia.

2. Postprocesador de audio de acuerdo con la reivindicación 1, en el cual el extractor (110) de banda está configurado para extraer la banda de baja frecuencia usando un dispositivo (111) de filtro de paso bajo y para extraer la banda (113) de alta frecuencia restando la banda de baja frecuencia de la señal de audio.

3. Postprocesador de audio de acuerdo con la reivindicación 1 o 2, en el cual se incluye la información (104) de ganancia de alta frecuencia variable en el tiempo correspondiente a una secuencia (300-303) de bloques de valores de muestreo de la señal (102) de audio de manera que un primer bloque (301) de valores de muestreo tenga una primera información (311) de ganancia asociada al mismo y un segundo bloque posterior (302) de valores de muestreo de la señal de audio tenga una segunda información de ganancia (312) diferente, donde el extractor (110) de banda está configurado para extraer, del primer bloque (301) de valores de muestreo, una primera banda de baja frecuencia y una primera banda de alta frecuencia y para extraer, del segundo bloque (302) de valores de muestreo, una segunda banda de baja frecuencia y una segunda banda de alta frecuencia y donde el procesador (120) de banda alta está configurado para modificar la primera banda de alta frecuencia usando la primera información (311) de ganancia para obtener una primera banda de alta frecuencia procesada y para modificar la segunda banda de alta frecuencia usando la segunda información de ganancia (312) para obtener una segunda banda de alta frecuencia procesada y

donde el combinador (130) está configurado para combinar la primera banda de baja frecuencia y la primera banda de alta frecuencia procesada para obtener un primer bloque combinado y para combinar la segunda banda de baja frecuencia y la segunda banda de alta frecuencia procesada para obtener un segundo bloque combinado, o

donde el extractor (110) de banda y el procesador (120) de banda alta y el combinador (130) están configurados para operar en bloques superpuestos y donde el postprocesador (100) de audio comprende además un sumador (140) de superposición para calcular una porción postprocesada mediante la suma de las muestras de audio de un primer bloque (301) y muestras de audio de un segundo bloque (302) en un rango de superposición de bloques, o

donde el extractor (110) de banda comprende: un generador (115) de ventanas de análisis para generar una secuencia de bloques de valores de muestreo de la señal de audio usando una ventana de análisis, donde los bloques están superpuestos en el tiempo; un procesador (116) de transformada discreta de Fourier para generar una secuencia de bloques de valores espectrales; un conformador (117) de paso bajo para modelar cada bloque de valores espectrales para obtener una secuencia de bloques de valores espectrales modelados de paso bajo; un procesador (118) de transformada discreta inversa de Fourier para generar una secuencia de bloques de valores de muestreo de paso bajo en el dominio del tiempo y un generador (119) de ventanas de síntesis para incluir en ventanas la secuencia de bloques de valores de muestreo de paso bajo en el dominio del tiempo usando una ventana de síntesis.

4. Postprocesador de audio de acuerdo con las reivindicaciones 1 a 3, en el cual la señal de audio comprende un parámetro (500) de control adicional como información lateral adicional, donde el procesador (120) de banda alta está configurado para aplicar la amplificación variable en el tiempo también de conformidad con el parámetro (500) de control adicional, donde una resolución temporal del parámetro (500) de control adicional es menor que la resolución temporal de la información de ganancia de alta frecuencia variable en el tiempo o el parámetro de control adicional es fijo para una pieza de audio específica.

5. Postprocesador de audio de acuerdo con una de las reivindicaciones anteriores, en el cual el extractor (110) de banda, el procesador (120) de banda alta y el combinador (130) están configurados para procesar secuencias de bloques (300-303) derivados de la señal de audio en forma de bloques superpuestos, de

manera que una porción posterior de un bloque anterior se derive de las mismas muestras de audio de la señal de audio que una porción anterior de un bloque posterior que es adyacente en el tiempo al bloque anterior, en el cual un rango (321) de superposición de los bloques superpuestos es igual a la mitad del bloque anterior y donde el bloque posterior tiene la misma longitud que el bloque anterior con respecto a un número de valores de muestreo y donde el postprocesador comprende además un sumador (140) de superposición para ejecutar la operación de superposición y suma, y en el cual el extractor (110) de banda está configurado para aplicar una pendiente de un filtro (111) de partición entre un rango de parada y un rango de pase del filtro de partición a un bloque de muestras de audio, donde la pendiente depende de la información de ganancia de alta frecuencia variable en el tiempo correspondiente al bloque de muestras.

6. Postprocesador de audio de acuerdo con la reivindicación 5,

en el cual el procesador (120) de banda alta está configurado para compensar además la atenuación de los eventos transitorios introducida en la señal de audio por un procesamiento ejecutado antes de un procesamiento por el postprocesador (100) de audio.

7. Postprocesador de audio de acuerdo con una de las reivindicaciones 1 a 6, en el cual la información de ganancia de alta frecuencia variante en el tiempo comprende una secuencia de índices (600) de ganancia e información (602) de rango extendido de ganancia o donde la información lateral comprende además una información (603) de compensación de ganancia y una información (604) de precisión de compensación de ganancia, donde el postprocesador de audio comprende

un decodificador (620) para decodificar los índices (601) de ganancia dependiendo de la información (602) de precisión de ganancia para obtener una ganancia (621) decodificada de un primer número de valores diferentes correspondientes a una primera información de precisión o una ganancia (621) decodificada de un segundo número de valores diferentes correspondientes a una segunda información de precisión, donde el segundo número es mayor que el primer número o

un decodificador (620) para decodificar los índices (603) de compensación de ganancia dependiendo de la información (604) de precisión de compensación para obtener un valor (622) de compensación de ganancia decodificada de un primer número de valores diferentes correspondientes a una información de precisión de compensación o un valor (622) de compensación de ganancia decodificada de un segundo número de valores diferentes correspondientes a una segunda información de precisión de compensación diferente, donde el primer número es mayor que el segundo número.

8. Postprocesador de audio de acuerdo con una de las reivindicaciones precedentes,

en el cual el extractor (110) de banda está configurado para ejecutar una transformada (116) discreta de Fourier por bloques con una longitud de bloque de N valores de muestreo para obtener un número de valores espectrales que es menor que un número de valores espectrales complejos N/2 mediante la ejecución de un algoritmo de transformada discreta de Fourier dispersa en la cual se omiten los cálculos de las ramas de valores espectrales superiores a una frecuencia máxima y

donde el extractor (110) de banda está configurado para calcular la señal de banda de baja frecuencia usando los valores espectrales hasta un rango de frecuencia de inicio de transición y ponderando (117a, 117b) los valores espectrales dentro del rango de frecuencia de inicio de transición, donde el rango de frecuencia de inicio de transición sólo se extiende hasta la frecuencia máxima o una frecuencia menor que la frecuencia máxima.

9. Postprocesador de audio de acuerdo con una de las reivindicaciones 1 a 8,

que está configurado para ejecutar solamente un postprocesamiento con un número máximo de canales u objetos, para la cual se dispone de información (106) lateral para la amplificación variable en el tiempo de la banda de alta frecuencia y para no ejecutar postprocesamiento alguno con un número de canales u objetos para el cual no se dispone de información lateral para la amplificación variable en el tiempo de la banda de alta frecuencia o

donde el extractor (110) de banda está configurado para no ejecutar extracción de banda alguna o no computar un par de Transformada Discreta de Fourier y Transformada Discreta de Fourier inversa en el caso de factores de ganancia triviales para la amplificación variable en el tiempo de la banda de alta frecuencia, y para el paso de una señal en el dominio del tiempo sin cambios ni inclusión en ventanas con los factores de ganancia triviales.

10. Preprocesador (200) de audio para preprocesar una señal (202) de audio, que comprende:

un analizador (260) de señales para analizar la señal (202) de audio para determinar información (204) de ganancia de alta frecuencia variable en el tiempo;

un extractor (210) de banda para extraer una banda (212) de alta frecuencia de la señal (202) de audio y una banda (214) de baja frecuencia de la señal de audio;

un procesador (220) de banda alta para ejecutar una modificación variable en el tiempo de la banda (212) de alta frecuencia de acuerdo con la información de ganancia de alta frecuencia variable en el tiempo para obtener una banda (222) de alta frecuencia procesada;

un combinador (230) para combinar la banda (222) de alta frecuencia procesada y la banda (214) de baja frecuencia para obtener una señal (232) de audio preprocesada y

una interfaz (250) de salida para generar una señal (252) de salida que comprende la señal (232) de audio preprocesada y la información (204) de ganancia de alta frecuencia variable en el tiempo en forma de información (206) lateral.

11. Preprocesador de audio de acuerdo con la reivindicación 10,

en el cual el analizador (260) de señales está configurado para analizar la señal de audio a fin de determinar (801, 802) una primera característica en un primer bloque (301) de tiempo de la señal de audio y una segunda característica en un segundo bloque (302) de tiempo de la señal de audio, donde la segunda característica es más transitoria que la primera característica o que tiene un nivel de energía de alta frecuencia más elevado que la primera característica,

donde el analizador (260) de señales está configurado para determinar (803) una primera información (311) de ganancia correspondiente a la primera característica y una segunda información (312) de ganancia correspondiente a la segunda característica, y

donde el procesador (220) de banda alta está configurado para aplicar un factor (804) de multiplicación a la porción de banda alta del segundo bloque (302) de tiempo de acuerdo con la segunda información de ganancia más fuerte que la porción de banda alta del primer bloque (301) de tiempo de conformidad con la primera información de ganancia.

12. Preprocesador de audio de acuerdo con cualquiera de las reivindicaciones 10 a 11, en el cual el analizador (260) de señales está configurado:

para calcular (805) la medida del fondo correspondiente a una energía de fondo de la banda alta para uno o más bloques temporales cercanos en el tiempo colocados antes del bloque de tiempo actual o colocados a continuación del bloque de tiempo actual o colocados antes y después del bloque de tiempo actual o incluyendo el bloque de tiempo actual o excluyendo el bloque de tiempo actual;

para calcular (808) una medida de la energía correspondiente a una banda alta del bloque actual; y

para calcular (809) un factor de ganancia usando la medida del fondo y la medida de la energía.

13. Preprocesador de audio de acuerdo con una de las reivindicaciones 10 a 12,

en el cual el analizador (260) de señales está configurado para cuantificar y recortar (812) una secuencia bruta de valores de información de ganancia para obtener la información de ganancia de alta frecuencia variable en el tiempo como secuencia de valores cuantificados,

donde el procesador (220) de banda alta está configurado para ejecutar (813) la modificación variable en el tiempo de la banda alta de conformidad con la secuencia de valores cuantificados, y

donde la interfaz (250) de salida está configurada para introducir (814) la secuencia de valores cuantificados en la información (206) lateral como información (204) de ganancia de alta frecuencia variable en el tiempo.

14. Preprocesador de audio de acuerdo con una de las reivindicaciones 10 a 12, en el cual el preprocesador (200) de audio está configurado

para determinar (815) otro valor de compensación de ganancia que describe una variación de pérdida de

energía introducida por un codificador o decodificador conectado subsiguientemente, y

para cuantificar (816) la información adicional de compensación de ganancia, y

donde la interfaz (250) de salida está configurada para introducir (817) la información adicional de compensación de ganancia en la información lateral.

15. Preprocesador de audio de acuerdo con una de las reivindicaciones 10 a 14,

en el cual se incluye la información (204) de ganancia de alta frecuencia variable en el tiempo para una secuencia (300-303) de bloques de valores de muestreo de la señal de audio de manera que un primer bloque (301) de valores de muestreo tenga una a primera información (311) de ganancia asociado al mismo y un segundo bloque (302) posterior de valores de muestras de la señal de audio tenga una segunda información (312) de ganancia diferente, donde el extractor de banda está configurado para extraer, del primer bloque de valores de muestreo, una primera banda de baja frecuencia y una primera banda de alta frecuencia y para extraer, del segundo bloque de valores de muestreo, una segunda banda de baja frecuencia y una segunda banda de alta frecuencia, y

donde el procesador (220) de banda alta está configurado para modificar la primera banda de alta frecuencia usando la primera información (311) de ganancia para obtener una primera banda de alta frecuencia procesada y para modificar la segunda banda de alta frecuencia usando la segunda información (312) de ganancia para obtener una segunda banda de alta frecuencia procesada, y

donde el combinador (230) está configurado para combinar la primera banda de baja frecuencia y la primera banda de alta frecuencia procesada para obtener un primer bloque combinado y para combinar la segunda banda de baja frecuencia y la segunda banda de alta frecuencia procesada para obtener un segundo bloque combinado.

16. Aparato codificador de audio para codificar una señal de audio, que comprende:

el preprocesador de audio de acuerdo con una cualquiera de las reivindicaciones 10 a 15, configurado para generar la señal (252) de salida que contiene la información de ganancia de alta frecuencia variable en el tiempo en forma de información lateral;

un codificador (900) de núcleo para generar una señal (902) codificada en el núcleo e información (904) lateral de núcleo; y

una interfaz (910) de salida para generar una señal (912) codificada que comprende la señal (902) codificada en el núcleo, la información (904) lateral de núcleo y la información de ganancia de alta frecuencia variable en el tiempo como información (106) lateral adicional.

17. Aparato decodificador de audio, que comprende:

una interfaz (920) de entrada para recibir una señal (912) de audio codificada que comprende una señal (902) codificada en el núcleo, información (904) lateral de núcleo e información (104) de ganancia de alta frecuencia variable en el tiempo como información lateral adicional;

un decodificador (930) de núcleo para decodificar la señal (902) codificada en el núcleo utilizando la información (904) lateral de núcleo para obtener una señal de núcleo decodificada; y

un postprocesador (100) para postprocesar la señal (102) de núcleo decodificada utilizando la información (104) de ganancia de alta frecuencia variable en el tiempo de conformidad con cualquiera de las reivindicaciones 1 a 9.

18. Método de postprocesamiento (100) de una señal (102) de audio que contiene información (104) de ganancia de alta frecuencia variable en el tiempo en forma de información (106) lateral, que comprende:

extraer (110) una banda (112) de alta frecuencia de la señal de audio y una banda (114) de baja frecuencia de la señal de audio;

ejecutar (120) una modificación variable en el tiempo de la banda alta de conformidad con la información (104) de ganancia de alta frecuencia variable en el tiempo para obtener una banda (122) de alta frecuencia procesada; y

combinar (130) la banda (122) de alta frecuencia procesada y la banda (114) de baja frecuencia.

19. Método de preprocesamiento (200) de una señal (202) de audio, que comprende:

analizar (260) la señal (202) de audio para determinar información (204) de ganancia de alta frecuencia variable en el tiempo;

extraer (210) una banda (212) de alta frecuencia de la señal de audio y una banda (214) de baja frecuencia de la señal de audio;

ejecutar (220) una modificación variable en el tiempo de la banda de alta frecuencia de acuerdo con la información de ganancia de alta frecuencia variable en el tiempo para obtener una banda de alta frecuencia procesada;

combinar (230) la banda (222) de alta frecuencia procesada y la banda (214) de baja frecuencia para obtener una señal de audio preprocesada; y

generar (250) una señal (252) de salida que comprende la señal (232) de audio preprocesada y la información (204) de ganancia de alta frecuencia variable en el tiempo en forma de información (106) lateral.

20. Método de codificación de una señal de audio, que comprende:

el método de preprocesamiento (200) de audio de acuerdo con la reivindicación 19 configurado para generar la señal de salida que contiene la información (204) de ganancia de alta frecuencia variable en el tiempo en forma de información (106) lateral;

generar una señal (902) codificada en el núcleo e información (904) lateral de núcleo; y

generar (910) una señal codificada (912) que comprende la señal (902) codificada en el núcleo, la información (904) lateral de núcleo y la información (204) de ganancia de alta frecuencia variable en el tiempo como información (106) lateral adicional.

21. Método de decodificación de audio, que comprende:

recibir (920) una señal (912) de audio codificada que comprende una señal (902) codificada en el núcleo, información (904) lateral de núcleo e información (204) de ganancia de alta frecuencia variable en el tiempo como información (106) lateral adicional ;

decodificar (930) la señal (902) codificada en el núcleo utilizando la información (904) lateral de núcleo para obtener una señal (102) de núcleo decodificada; y

postprocesar (100) la señal (102) de núcleo decodificada utilizando la información (104) de ganancia de alta frecuencia variable en el tiempo de acuerdo con el método de la reivindicación 18.

22. Programa informático para realizar, cuando se ejecuta en un ordenador o un procesador, cualquiera de los métodos de acuerdo con las reivindicaciones 18, 19, 20, 21.

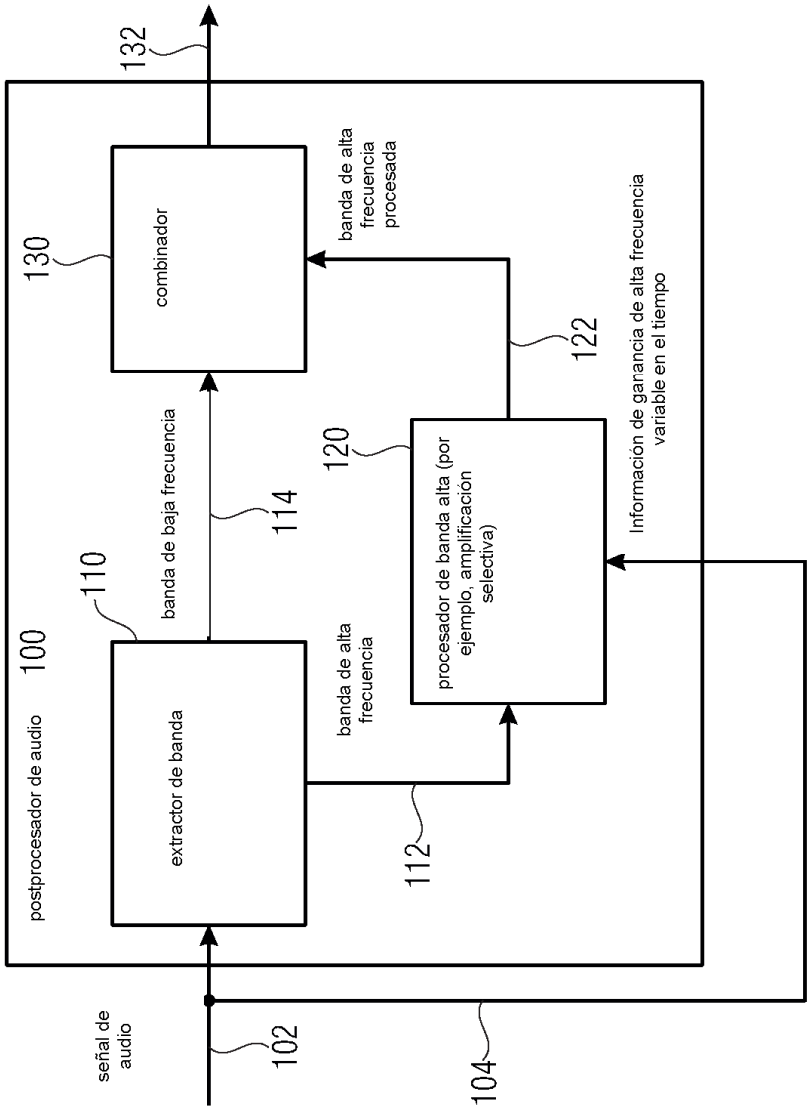
132
130
combinador
banda de alta frecuencia procesada
122
Información de ganancia de alta frecuencia variable en el tiempo
120
procesador de banda alta (por ejemplo, amplificación selectiva)
114
banda de baja frecuencia
110
100
postprocesador de audio
extractor de banda
banda de alta frecuencia
112
señal de audio
102
104

Fig. 1

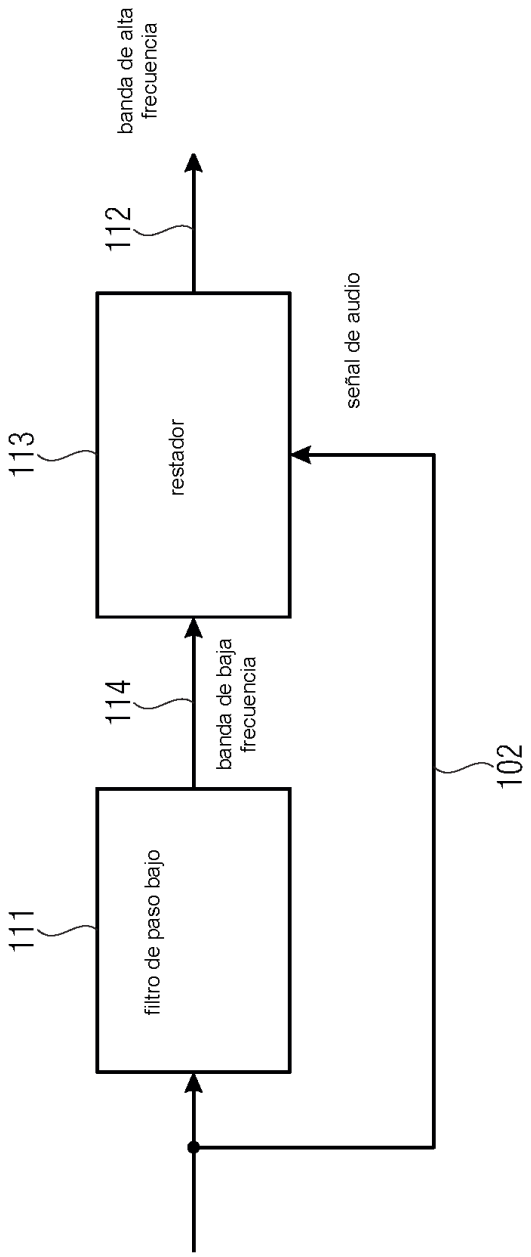
banda de alta frecuencia
112
señal de audio
113
restador
114
banda de baja frecuencia
102
111
filtro de paso bajo

Fig. 2

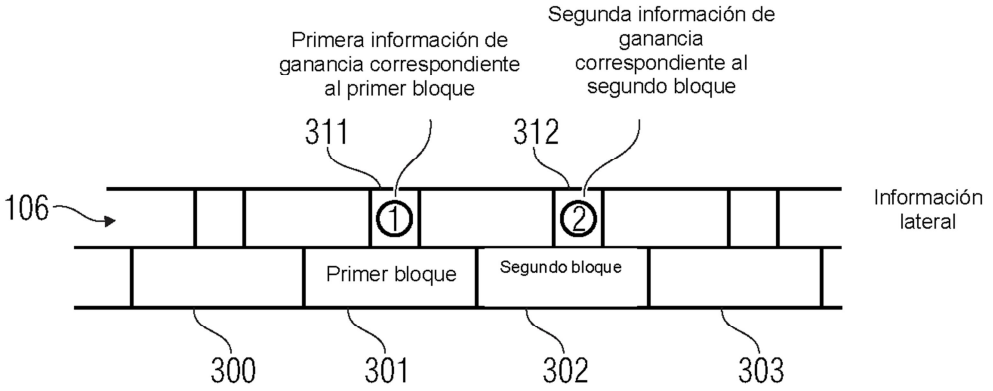

Fig. 3a

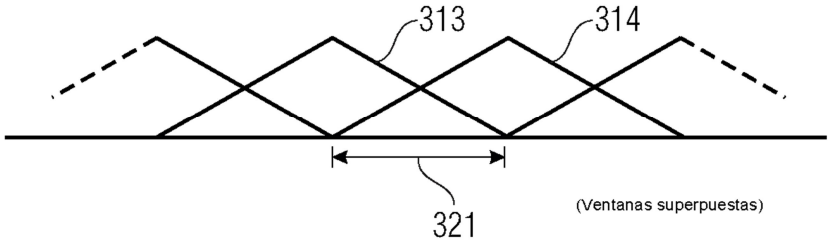

Fig. 3b

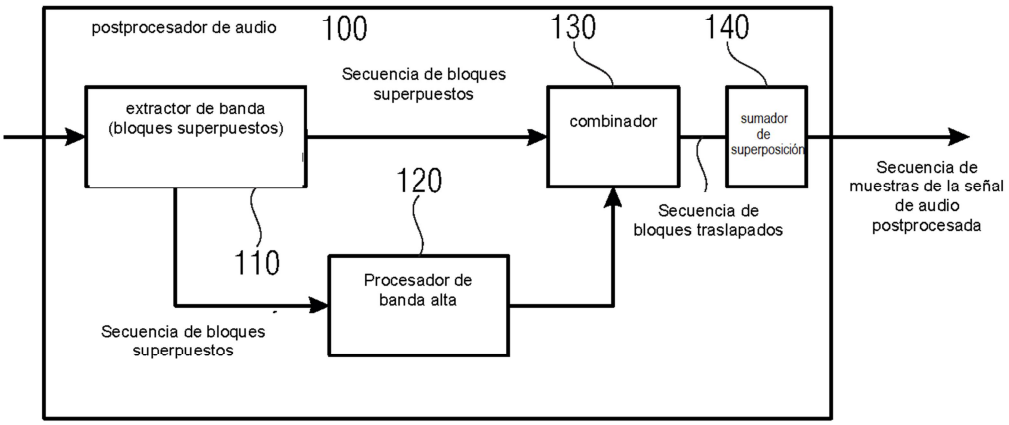

Fig. 3c

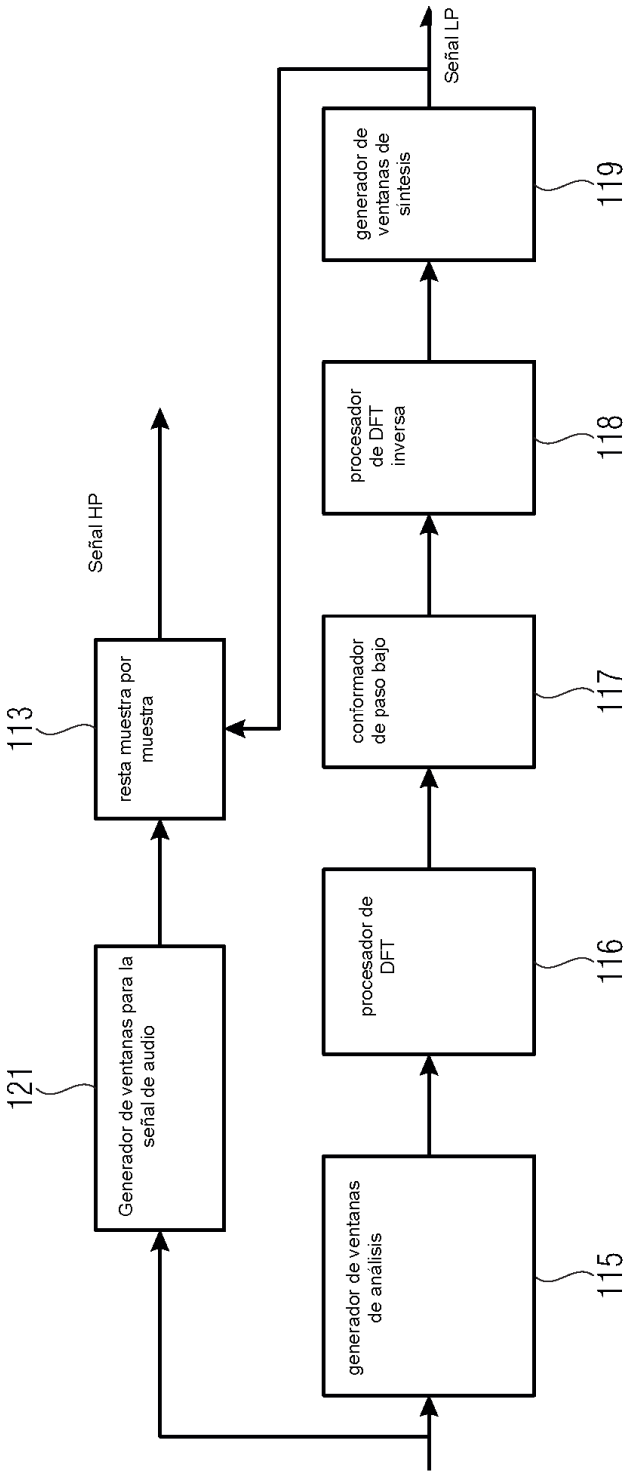
Señal LP
generador de ventanas de síntesis
119
procesador de DFT inversa
118
conformador de paso bajo
117
procesador de DFT
116
generador de ventanas de análisis
115
Señal HP
resta muestra por muestra
113
Generador de ventanas para la señal de audio
121

Fig. 4

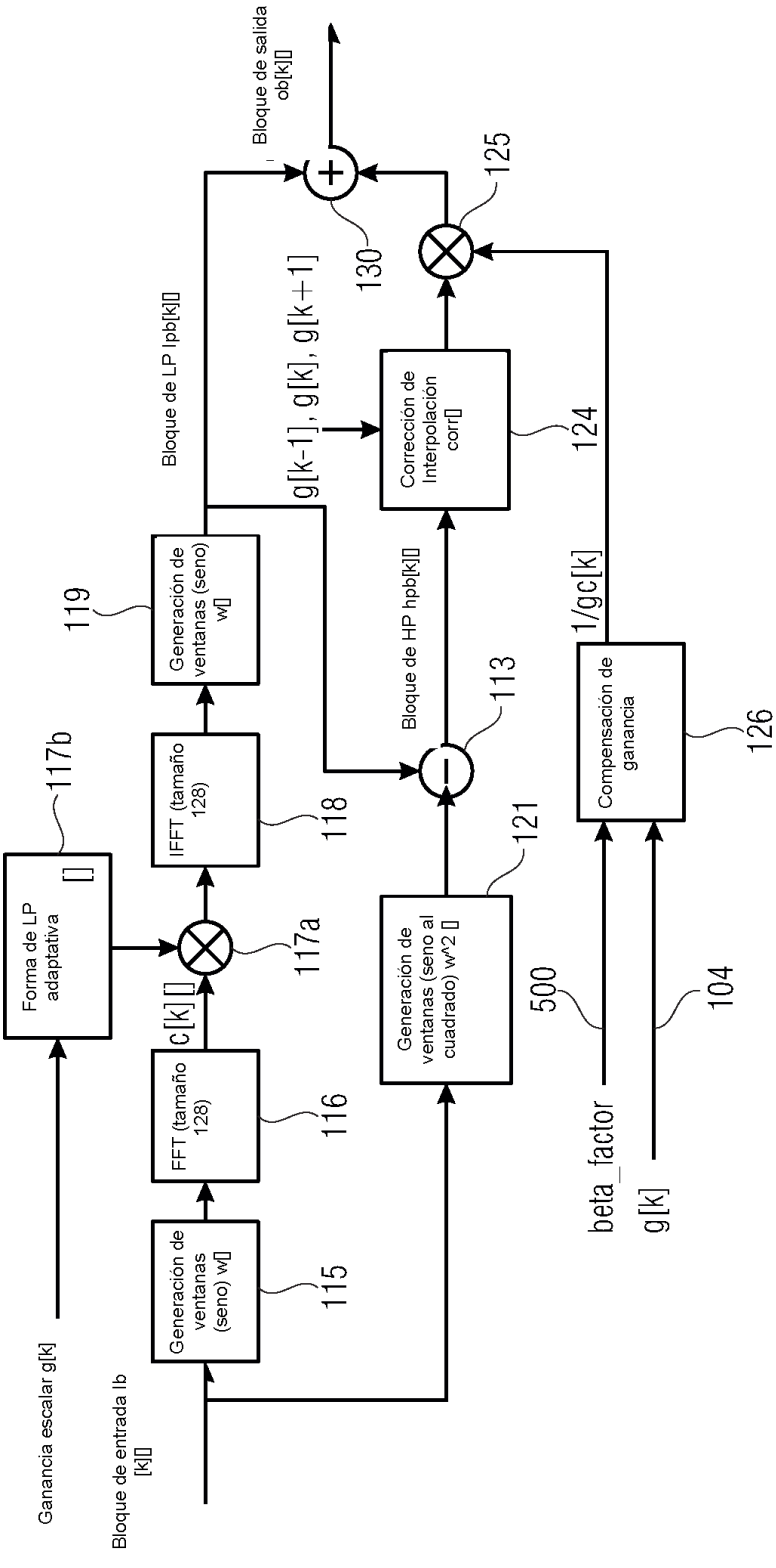
Ganancia escalar g[k]
Bloque de entrada Ib [k][]
Generación de ventanas (seno) w[]
115
FFT (tamaño 128)
116
c[k][]
Forma de LP adaptativa []
117b
117a
IFFT (tamaño 128)
118
Generación de ventanas (seno) w[]
119
Bloque de LP lpb[k][]
Bloque de salida ob[k][]
130
125
g[k-1], g[k], g[k+1]
Corrección de Interpolación corr[]
124
Bloque de HP hpb[k][]
113
Generación de ventanas (seno al cuadrado) w^2 []
121
500
beta_factor
1/gc[k]
Compensación de ganancia
126
g[k]
104

Fig. 5a

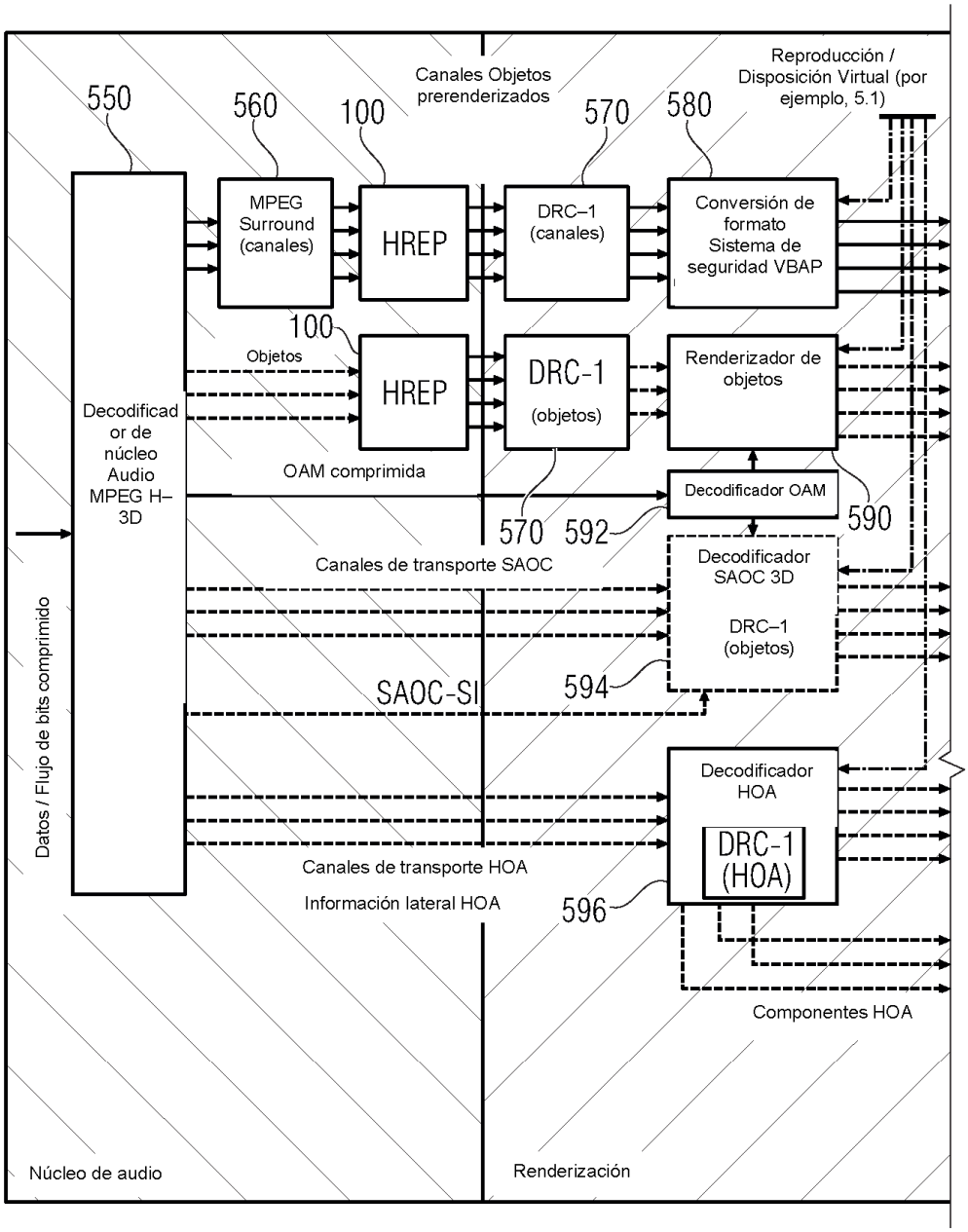

Fig. 5b1

Fig. 5b	Fig. 5b1
	Fig. 5b2

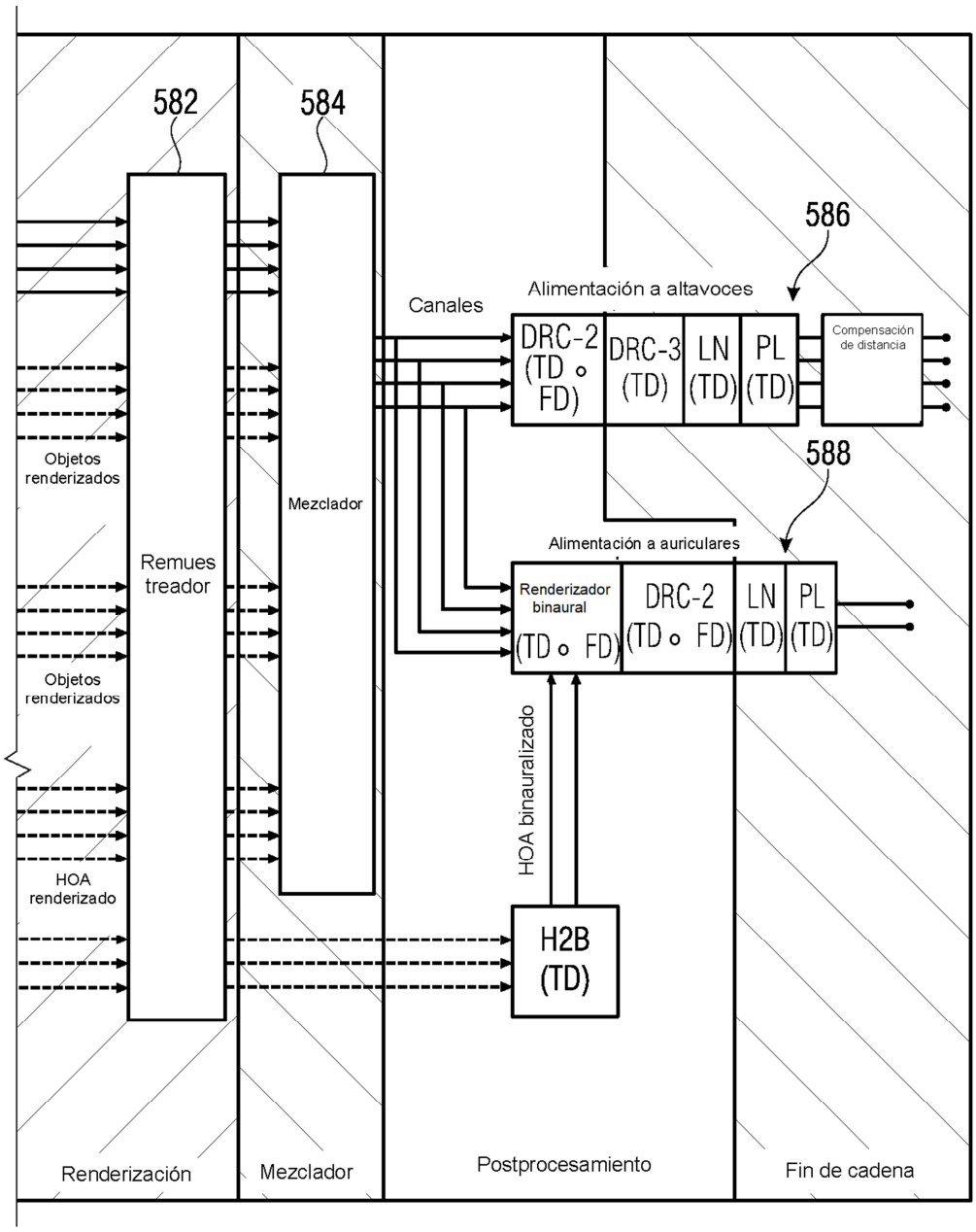

Fig. 5b2

Fig. 5b	Fig. 5b1
	Fig. 5b2

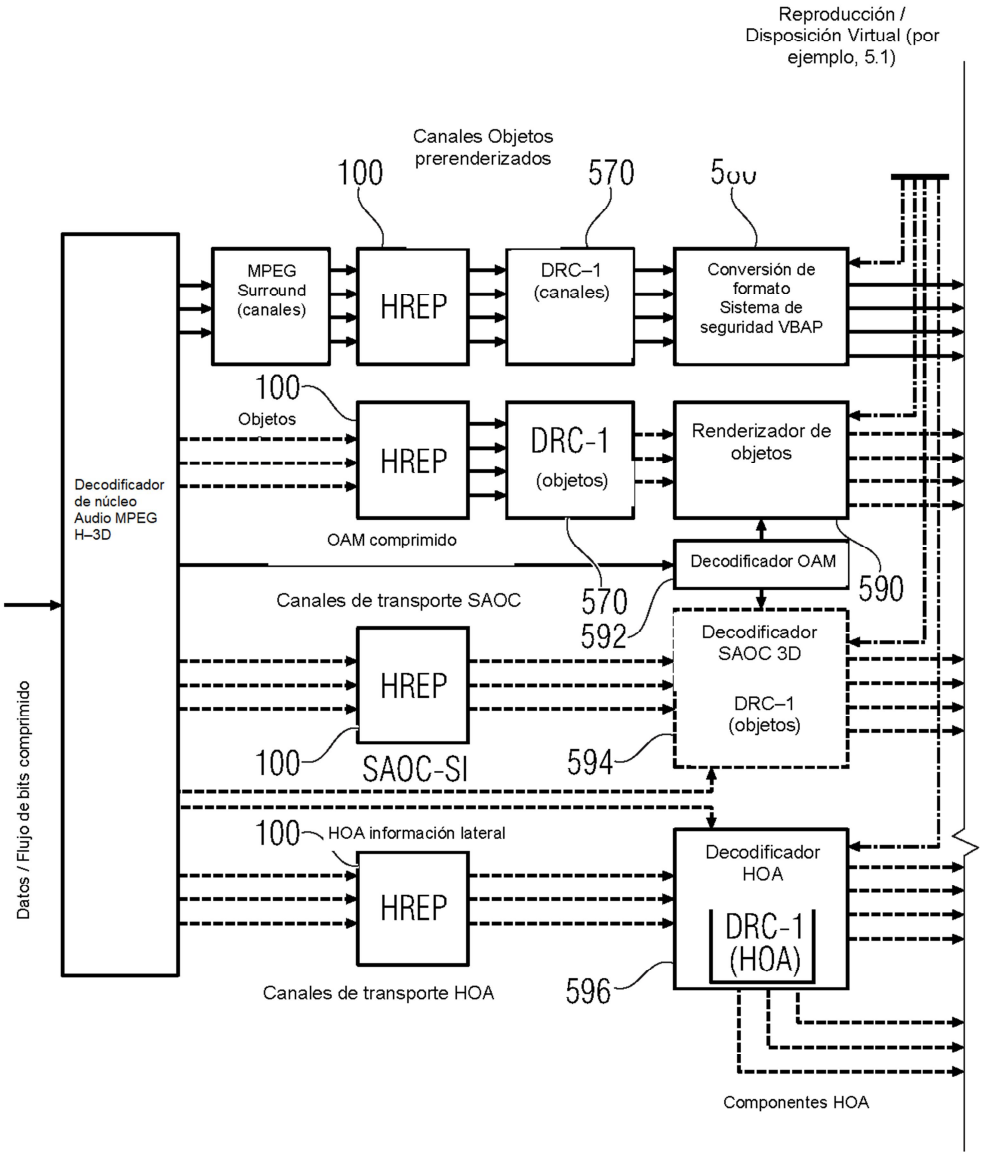

Fig. 5c1

Fig. 5c	Fig. 5c1
	Fig. 5c2

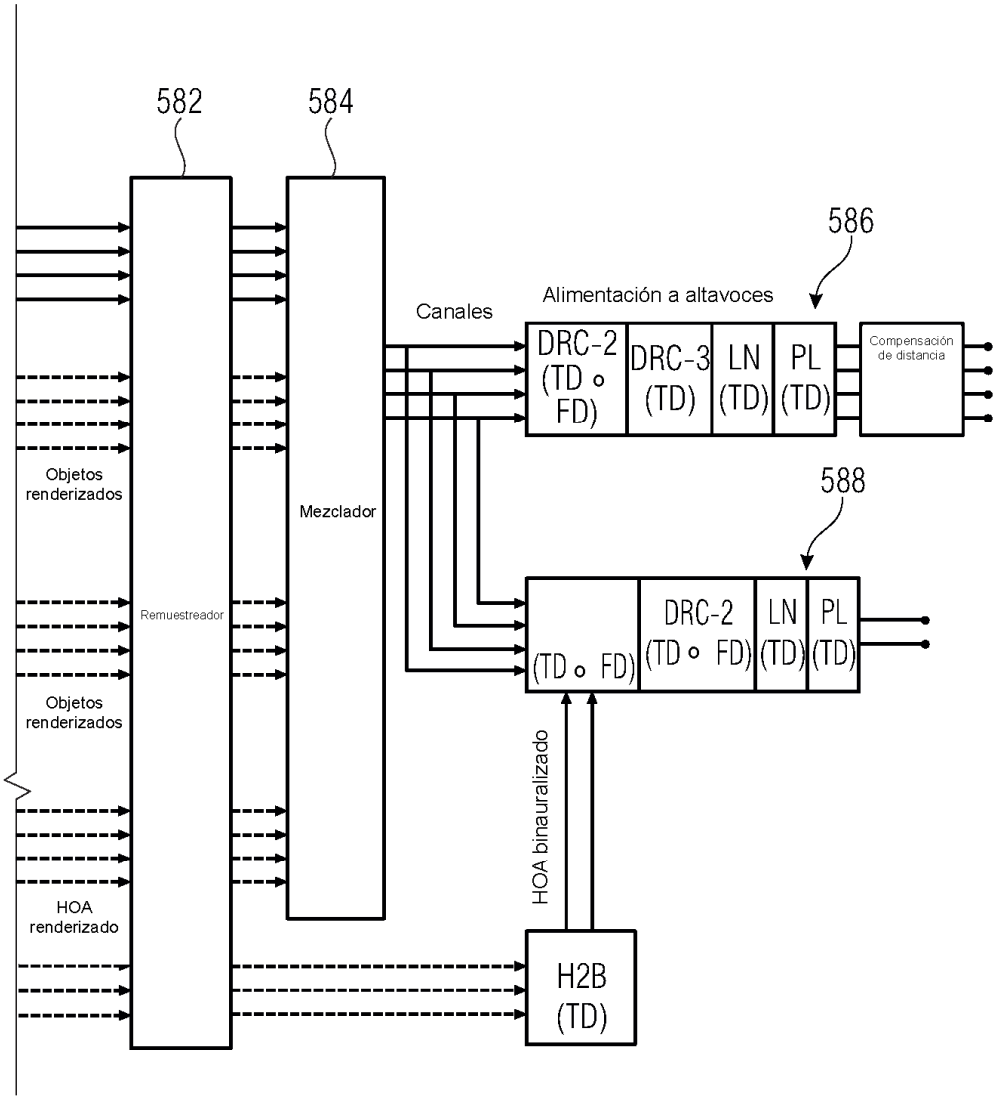

Fig. 5c2

Fig. 5c	Fig. 5c1
	Fig. 5c2

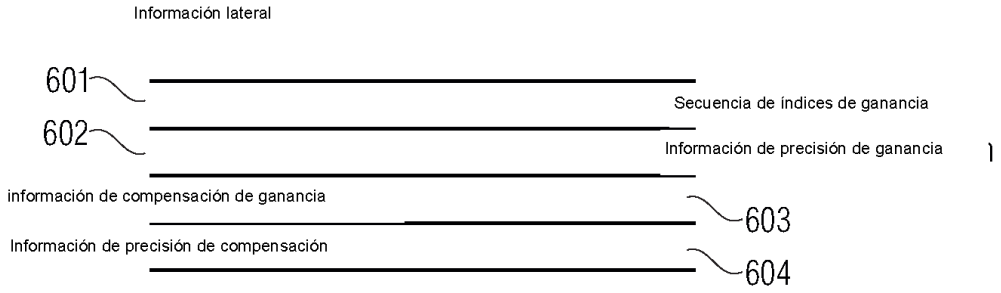
Información lateral
601
Secuencia de índices de ganancia
602
Información de precisión de ganancia
información de compensación de ganancia
603
Información de precisión de compensación
604

Fig. 6a

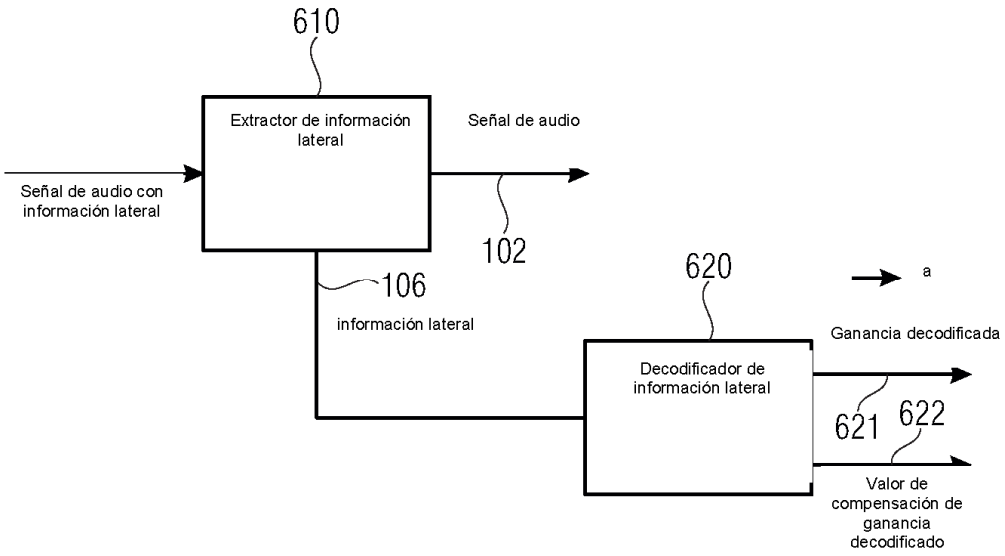
610
Extractor de información lateral
Señal de audio
Señal de audio con información lateral
102
106
información lateral
620
a
Ganancia decodificada
Decodificador de información lateral
621
622
Valor de compensación de ganancia decodificado

Fig. 6b

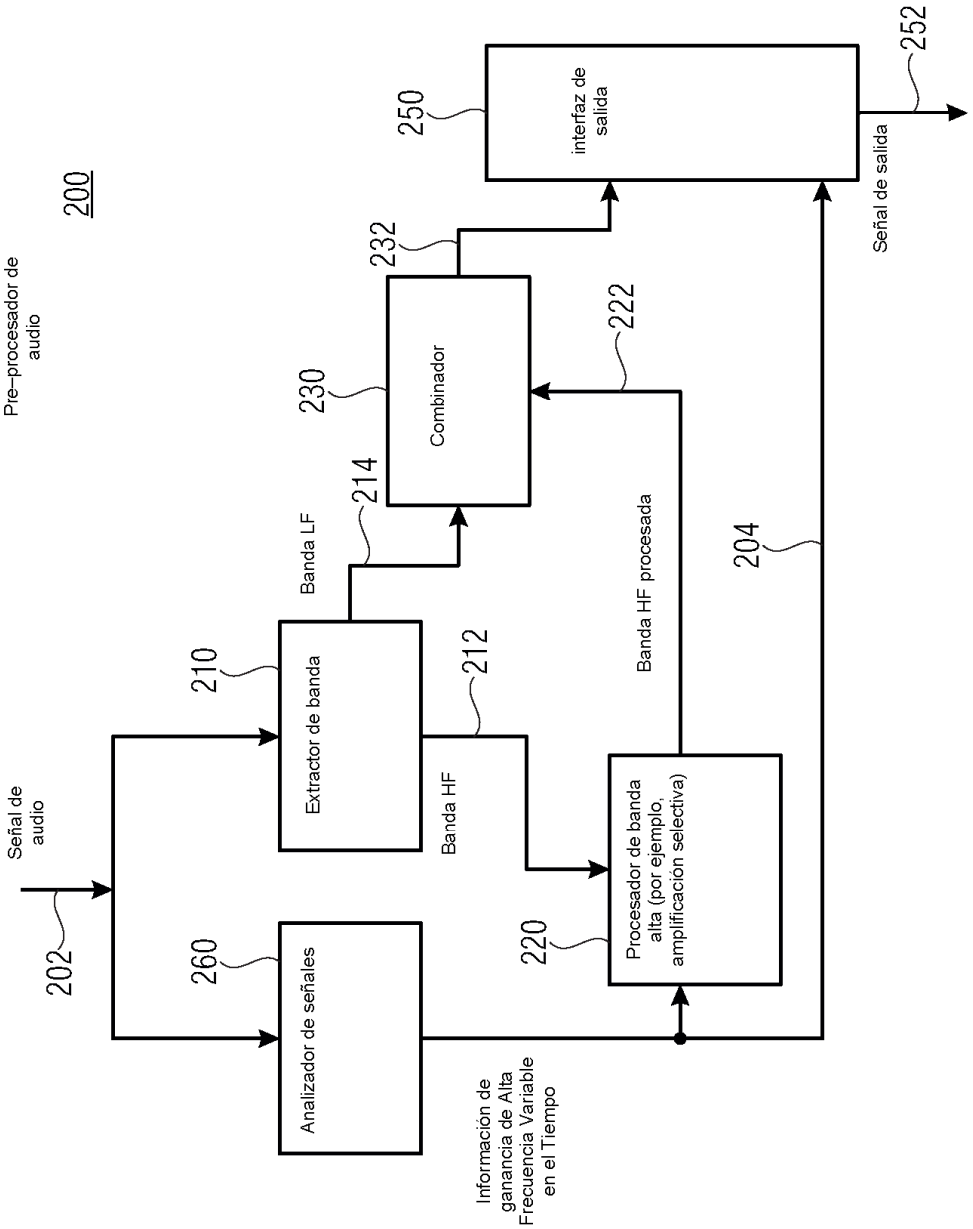
200
Pre-procesador de audio
Señal de audio
202
260
Analizador de señales
210
Extractor de banda
Banda LF
214
Banda HF
212
230
Combinador
232
222
Banda HF procesada
220
Procesador de banda alta (por ejemplo, amplificación selectiva)
204
250
interfaz de salida
252
Señal de salida
Información de ganancia de Alta Frecuencia Variable en el Tiempo

Fig. 7

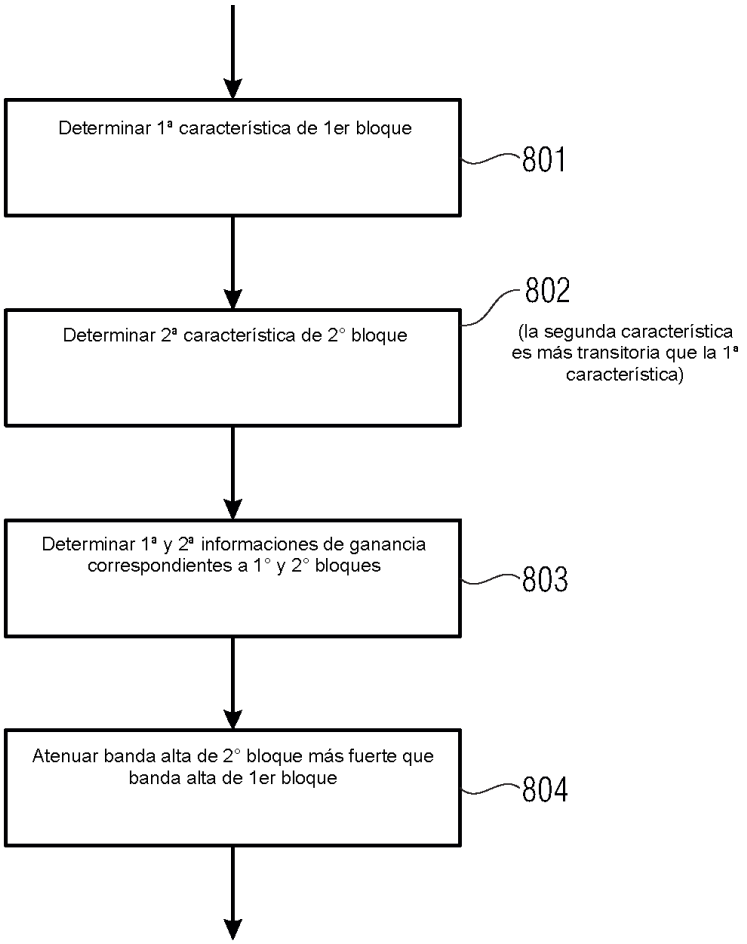
Determinar 1ª característica de 1er bloque
801
Determinar 2ª característica de 2° bloque
802
(la segunda característica es más transitoria que la 1ª característica)
Determinar 1ª y 2ª informaciones de ganancia correspondientes a 1° y 2° bloques
803
Atenuar banda alta de 2° bloque más fuerte que banda alta de 1er bloque
804

Fig. 8a

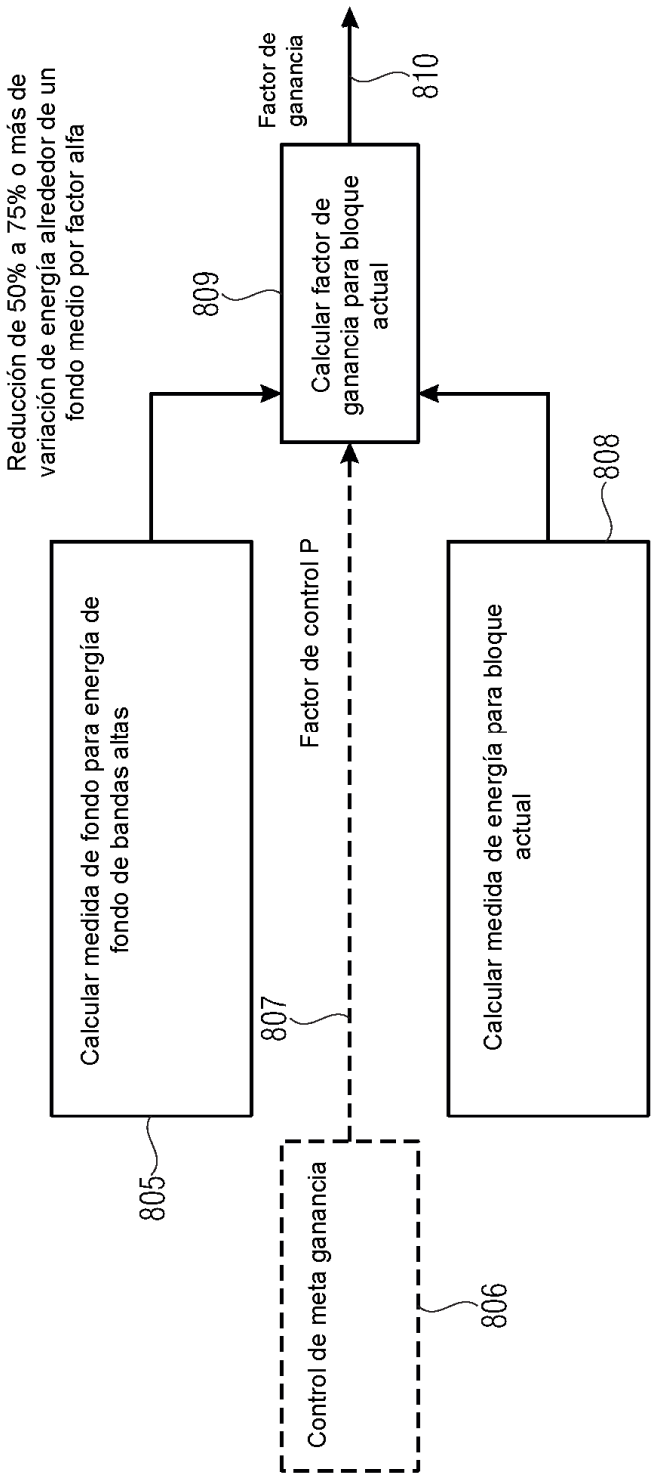
Reducción de 50% a 75% o más de variación de energía alrededor de un fondo medio por factor alfa
Factor de ganancia
810
809
Calcular factor de ganancia para bloque actual
808
Factor de control P
Calcular medida de fondo para energía de fondo de bandas altas
Calcular medida de energía para bloque actual
807
805
Control de meta ganancia
806

Fig. 8b

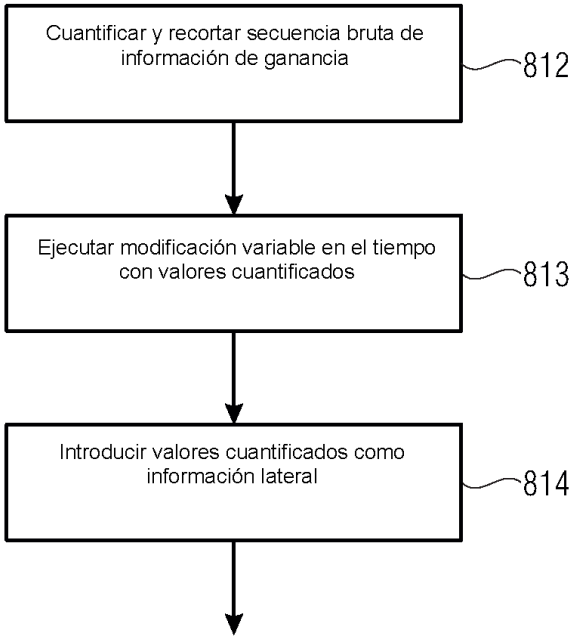
Cuantificar y recortar secuencia bruta de información de ganancia
812
Ejecutar modificación variable en el tiempo con valores cuantificados
813
Introducir valores cuantificados como información lateral
814

Fig. 8c

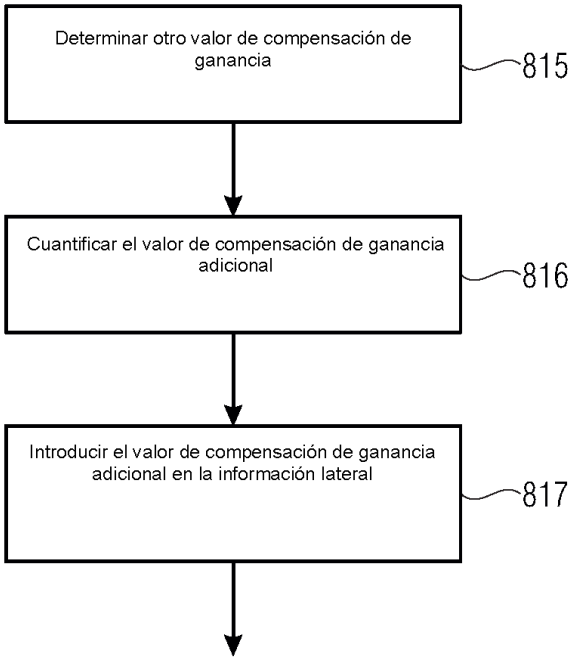
Determinar otro valor de compensación de ganancia
815
Cuantificar el valor de compensación de ganancia adicional
816
Introducir el valor de compensación de ganancia adicional en la información lateral
817

Fig. 8d

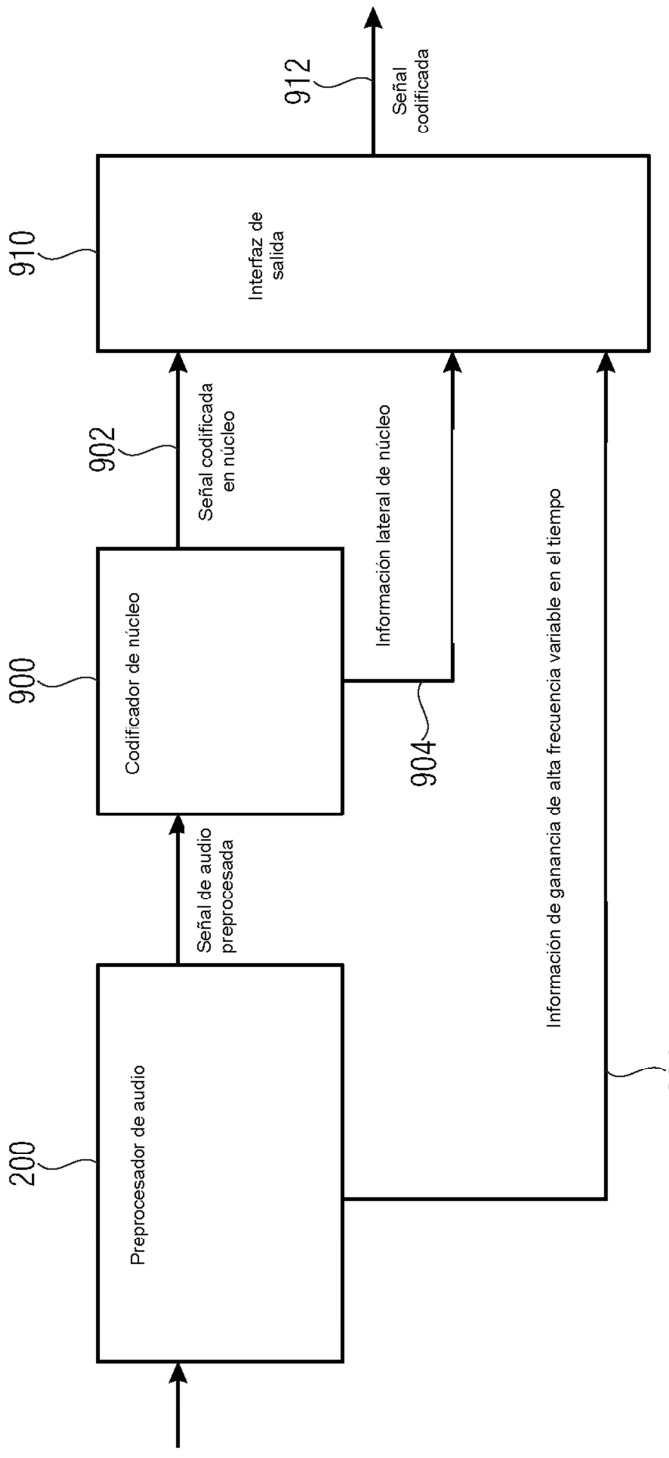
912
Señal codificada
910
Interfaz de salida
902
Señal codificada en núcleo
Información lateral de núcleo
900
Codificador de núcleo
904
Señal de audio preprocesada
Información de ganancia de alta frecuencia variable en el tiempo
200
Preprocesador de audio
204

Fig. 9a

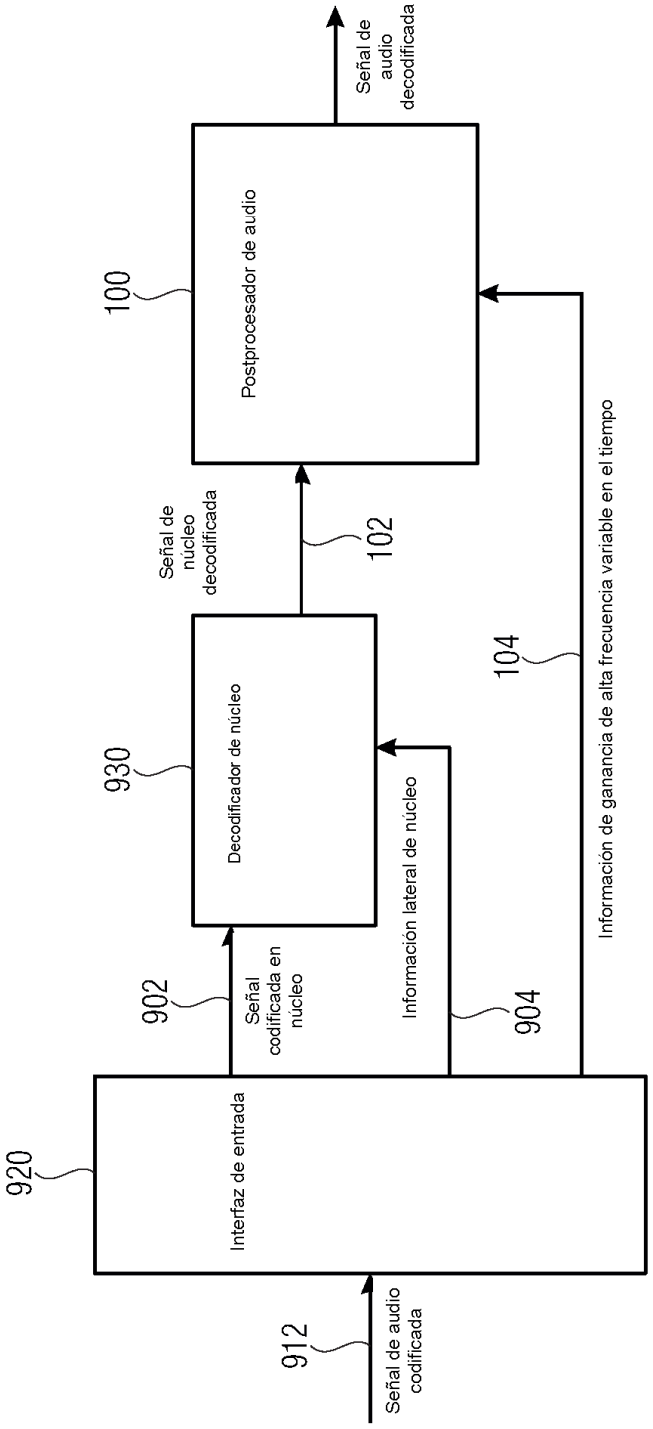
Señal de audio codificada
912
Interfaz de entrada
920
Señal codificada en núcleo
902
Información lateral de núcleo
904
Información de ganancia de alta frecuencia variable en el tiempo
104
Decodificador de núcleo
930
Señal de núcleo decodificada
102
Postprocesador de audio
100
Señal de audio decodificada

Fig. 9b

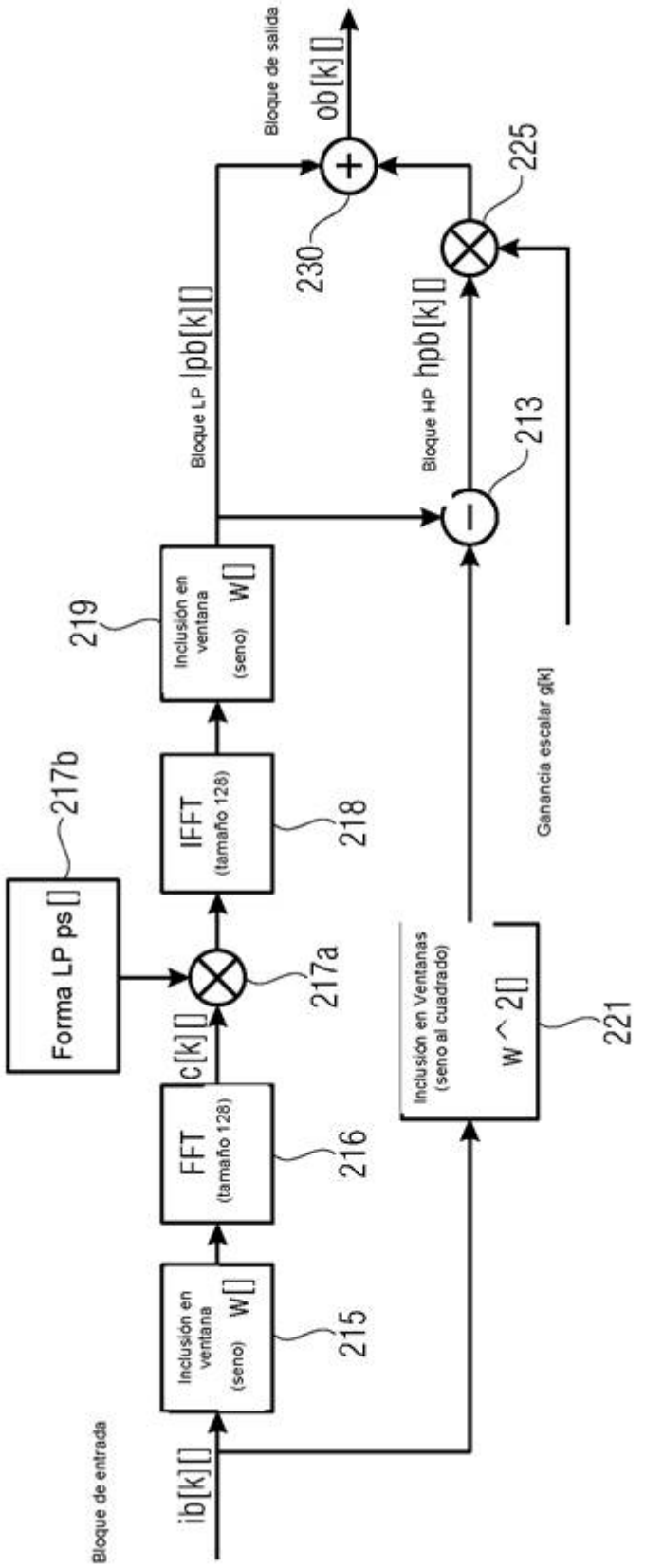
Bloque de entrada
ib[k][]
Inclusión en ventana (seno) w[]
215
FFT (tamaño 128)
216
c[k][]
Forma LP ps[]
217b
217a
IFFT (tamaño 128)
218
219
Inclusión en ventana (seno) w[]
Bloque LP lpb[k][]
Inclusión en Ventanas (seno al cuadrado)
w^2[]
221
Ganancia escalar g[k]
213
Bloque HP hpb[k][]
225
230
Bloque de salida
ob[k][]

Fig. 9c

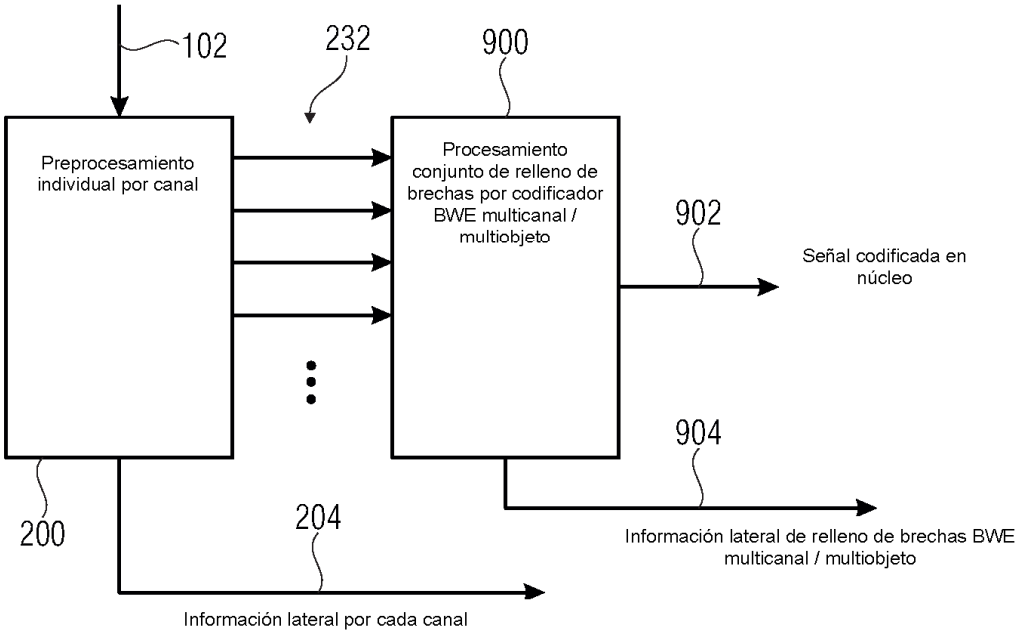

Fig. 10a

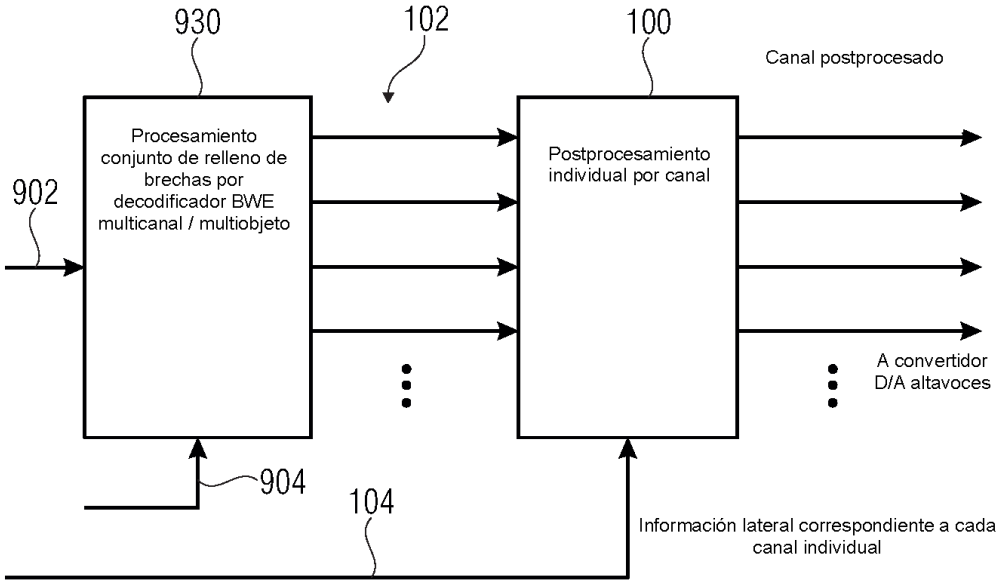

Fig. 10b

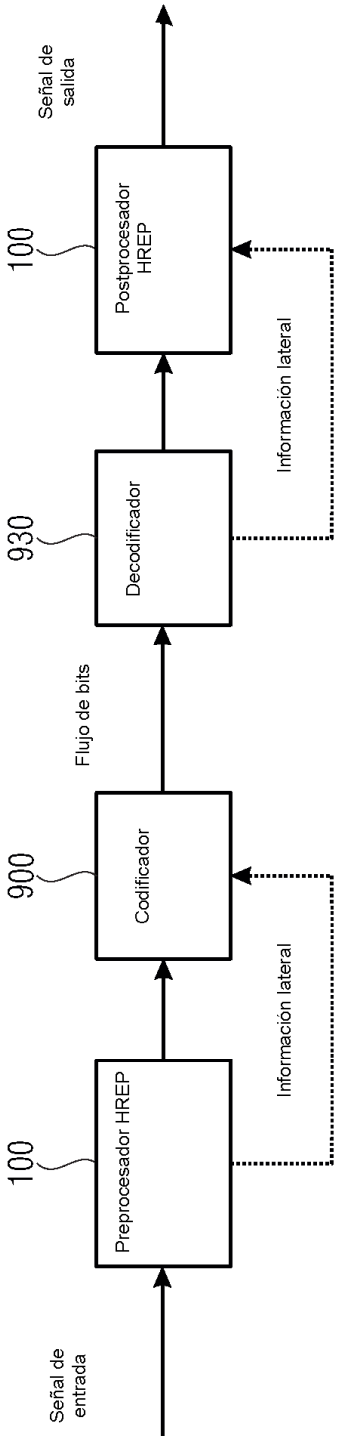
Señal de entrada
100
Preprocesador HREP
Información lateral
900
Codificador
Flujo de bits
930
Decodificador
Información lateral
100
Postprocesador HREP
Señal de salida

Fig. 10c

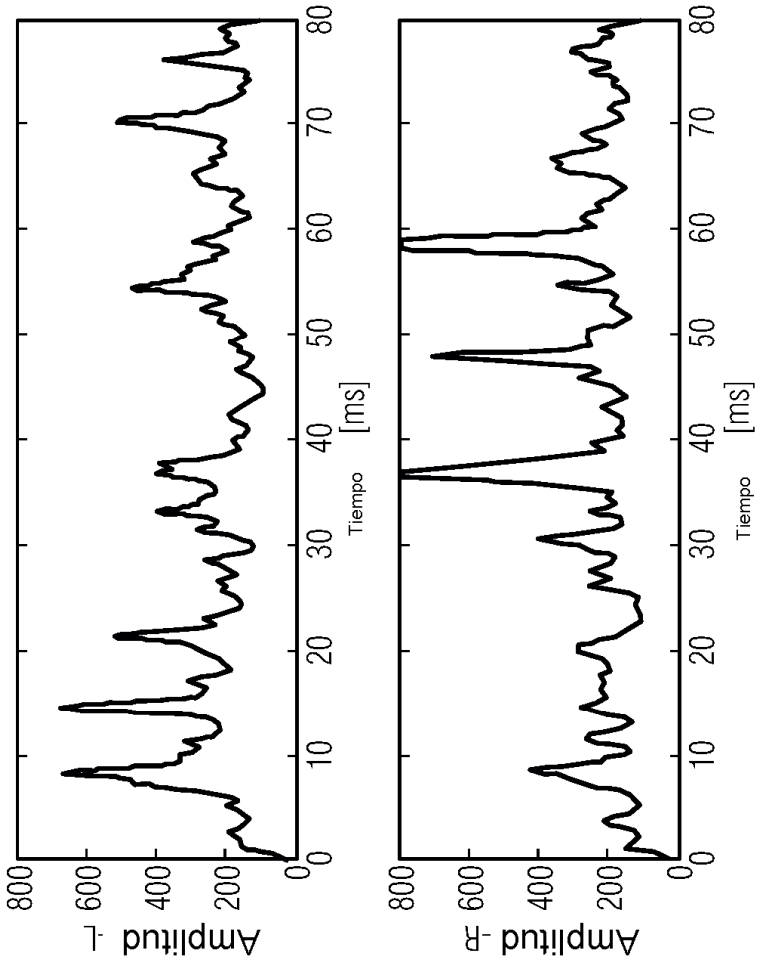
Amplitud -L
800
600
400
200
0
0
10
20
30
40
50
60
70
80
Tiempo [ms]
Amplitud -R
800
600
400
200
0
0
10
20
30
40
50
60
70
80
Tiempo [ms]

Fig. 11

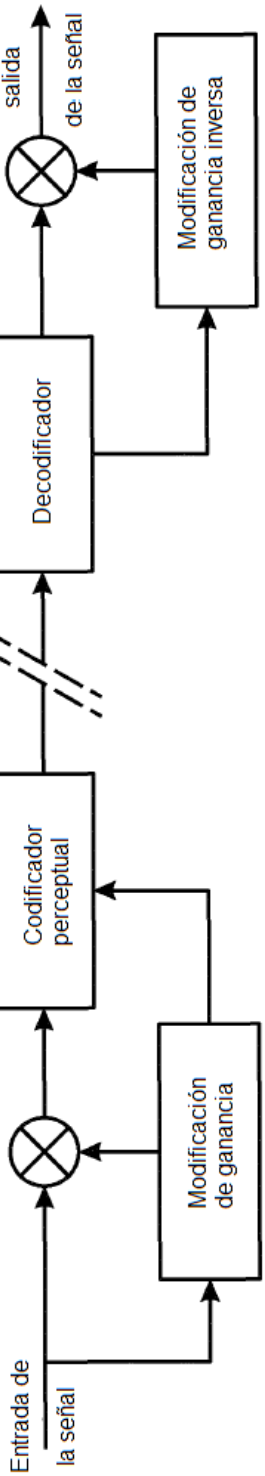
Entrada de la señal
Modificación de ganancia
Codificador perceptual
Decodificador
Modificación de ganancia inversa
salida de la señal

Fig. 12

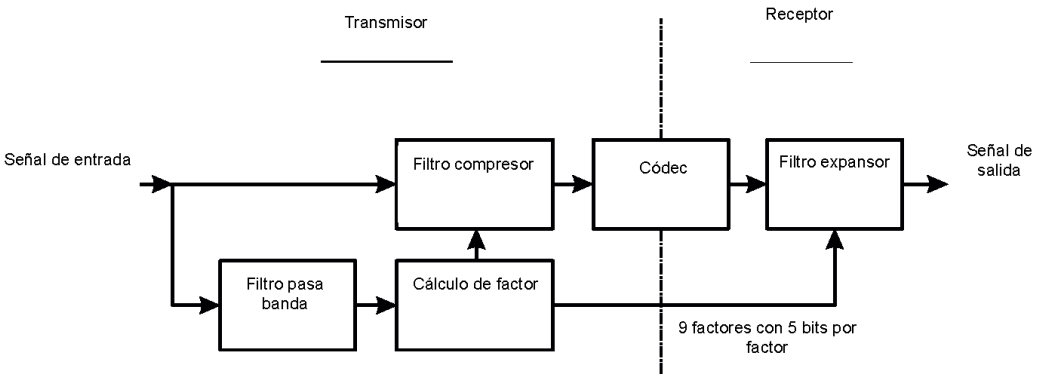

Fig. 13a

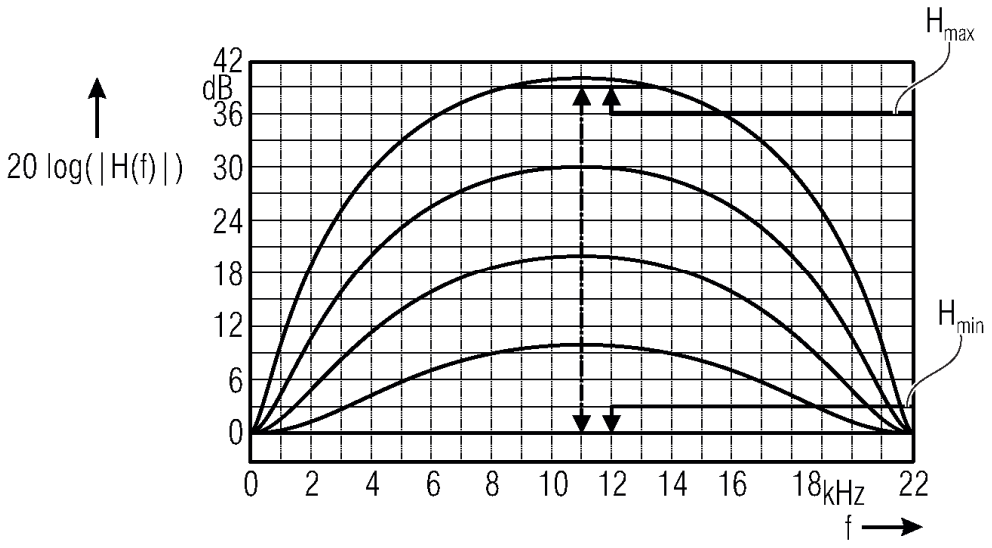

Fig. 13b

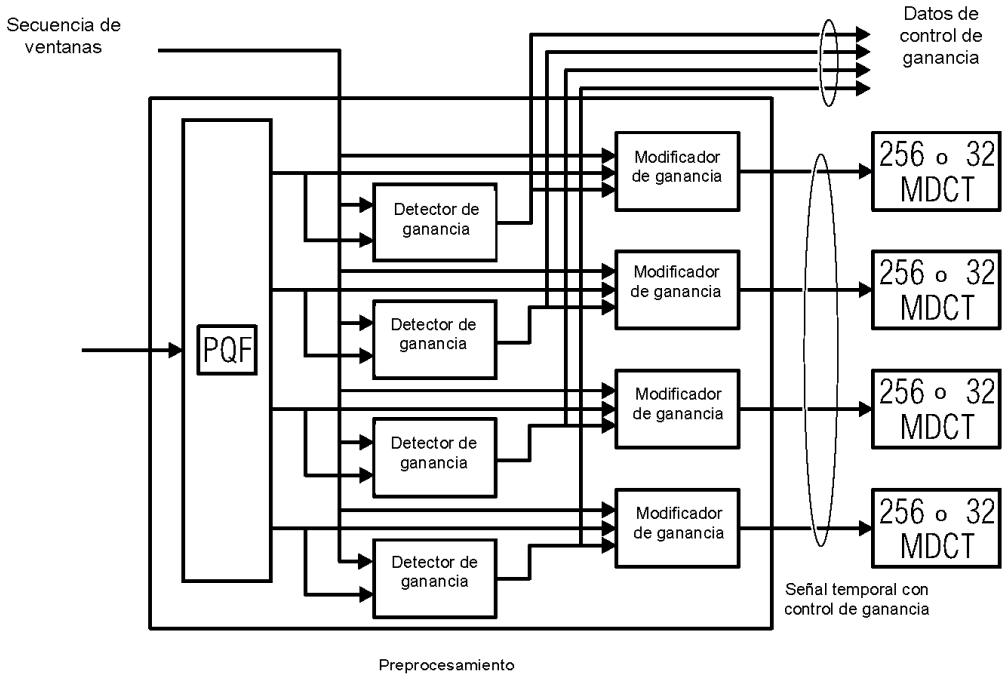
Secuencia de ventanas
Datos de control de ganancia
PQF
Detector de ganancia
Detector de ganancia
Detector de ganancia
Detector de ganancia
Modificador de ganancia
Modificador de ganancia
Modificador de ganancia
Modificador de ganancia
256 o 32 MDCT
256 o 32 MDCT
256 o 32 MDCT
256 o 32 MDCT
Señal temporal con control de ganancia
Preprocesamiento

Fig. 14

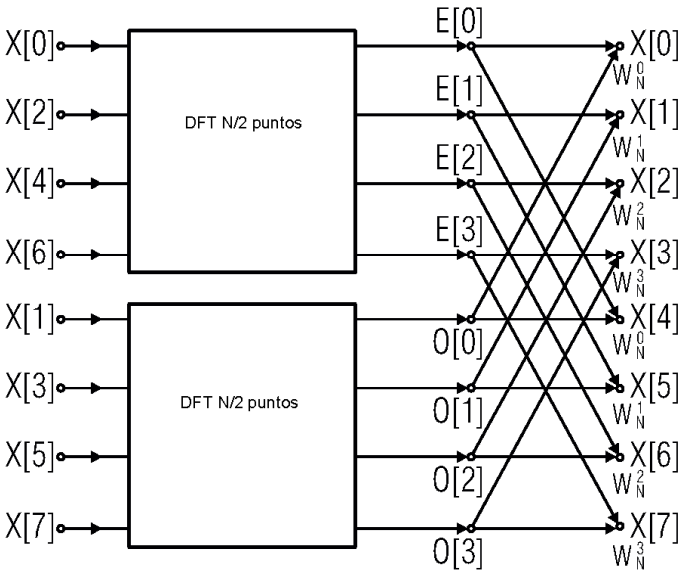
X[0]
X[2]
X[4]
X[6]
X[1]
X[3]
X[5]
X[7]
DFT N/2 puntos
DFT N/2 puntos
E[0]
E[1]
E[2]
E[3]
O[0]
O[1]
O[2]
O[3]
X[0]
X[1]
X[2]
X[3]
X[4]
X[5]
X[6]
X[7]

Fig. 15

Metodología	MUSHRA ITU-R BS.1534
Laboratorio	Fraunhofer IIS
Condiciones de prueba	"nohrep": Codificador de Calidad de Referencia (núcleo FD que utiliza IGF) "hrep": Configuración propuesta
Tasa de bits / Canales	128'kbps 5.1 canales
Número de oyentes después de la post–clasificación	14
Señales de prueba	• 1: ARL_applause • 2: Appl3D • 3: StudioAppl (Aplauso en estudio) • 4: applse • 5: chostakovitch • 6: fountain_music • 7: glock • 8: indie2 • 9: jackson1 • 10: pops • 11: poulenc • 12: rock_concert • 13: stomp

Fig. 16a

Fig. 16	Fig. 16a
	Fig. 16b

Tasa de bits / Canales	48 kbps estéreo
Número de oyentes después de la postclasificación	15
Señales de prueba	• 1: ARLapplause • 2: Ejercicio • 3: HeavyApplause (Aplauso cerrado) • 4: Intro • 5: Klatschen • 6: MediumApplause (Aplauso medio) • 7: SallyBrown • 8: Aplauso
Tasa de bits / Canales	128 kbps estéreo
Número de oyentes después de la post	14
Señales de prueba	• 1: ARLapplause • 2: Ejercicio • 3: HeavyApplause (Aplauso cerrado) • 4: Intro • 5: Klatschen • 6: MediumApplause Aplauso Medio • 7: SallyBrown • 8: applse aplauso

Fig. 16b

Fig. 16	Fig. 16a
	Fig. 16b

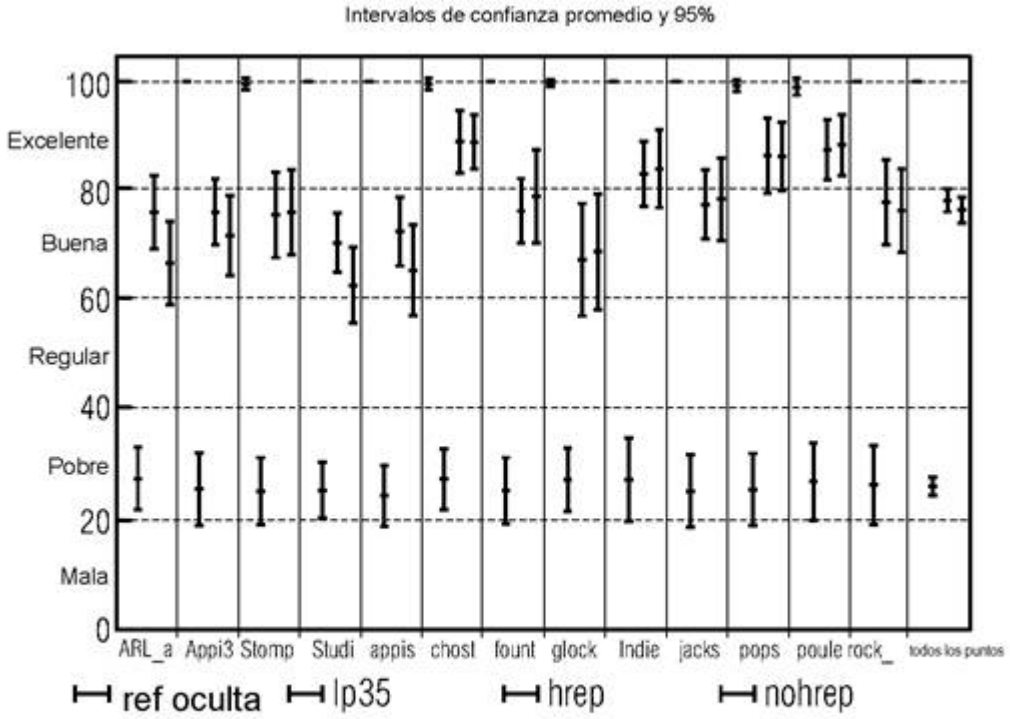

Fig. 17a

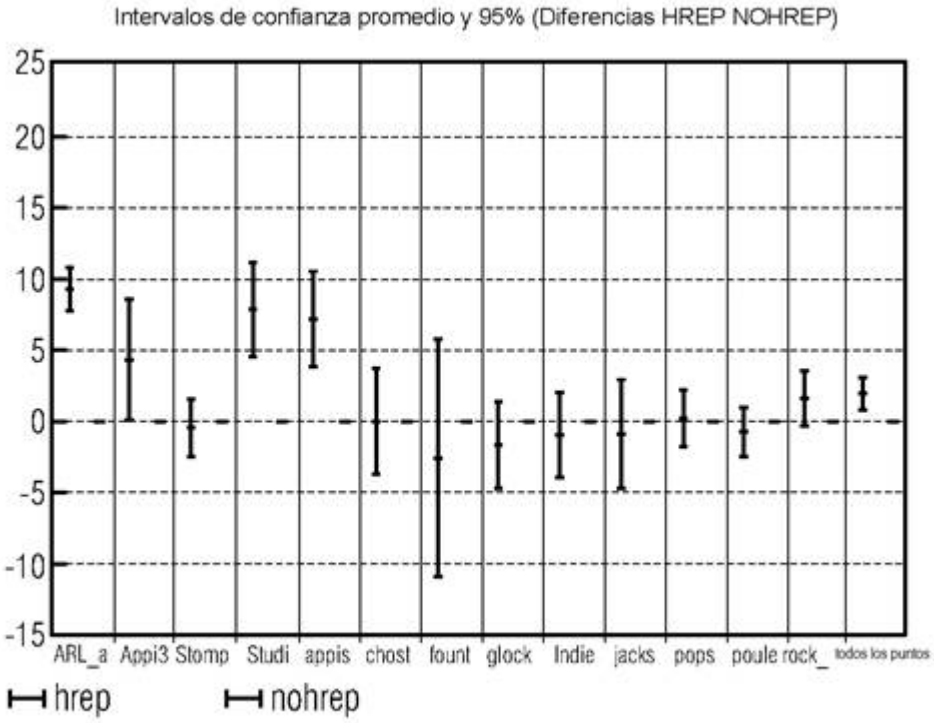

Fig. 17b

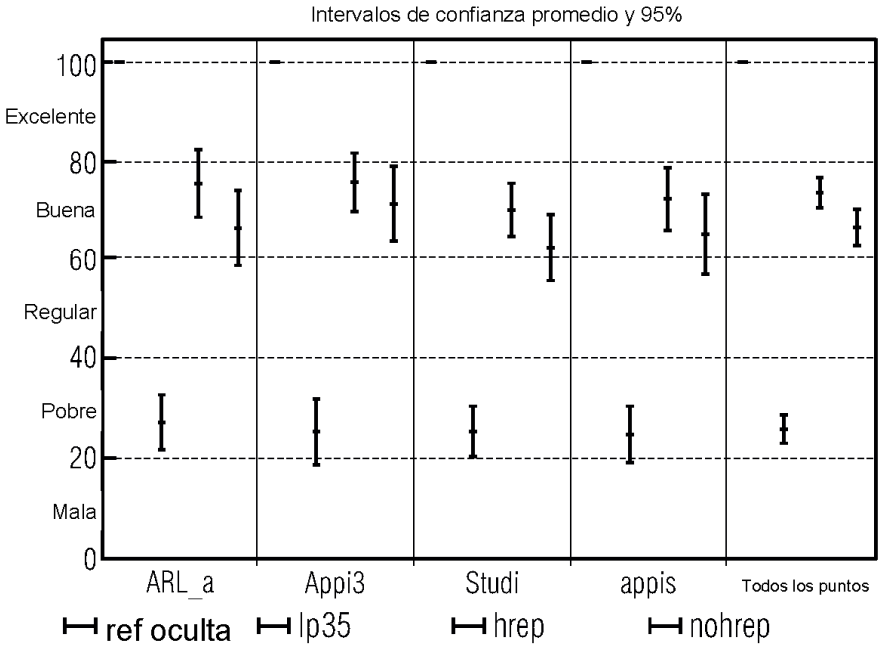

Fig. 17c

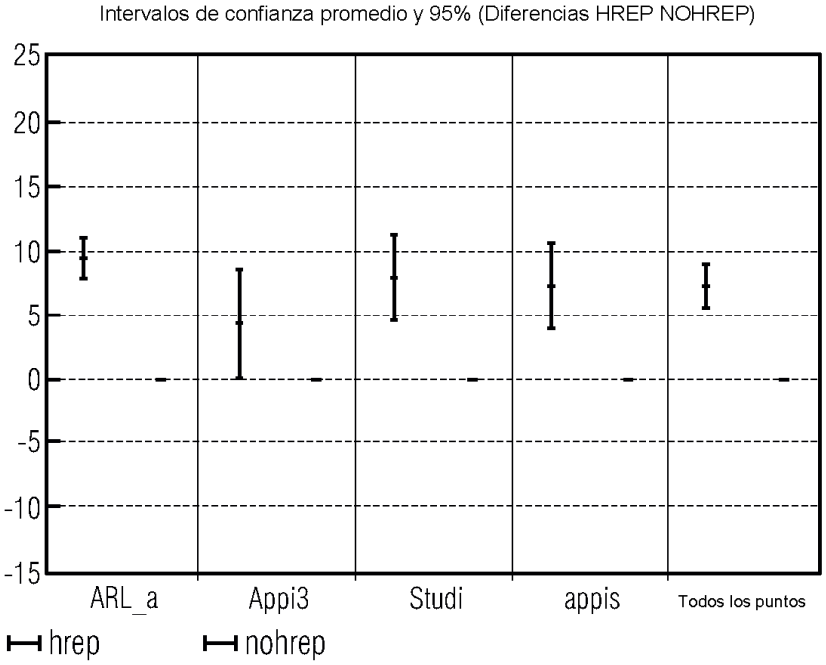

Fig. 17d

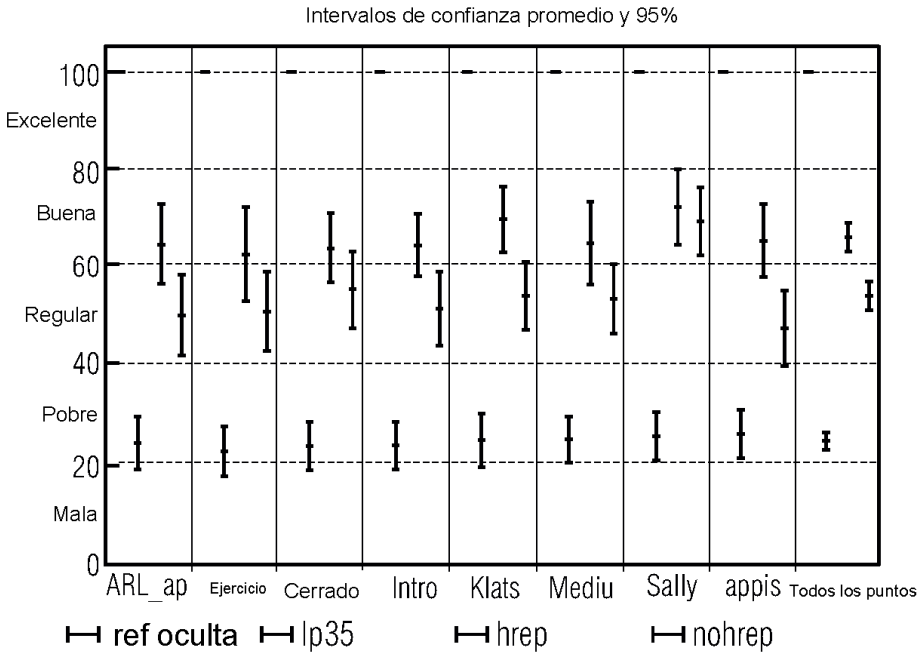

Fig. 17e

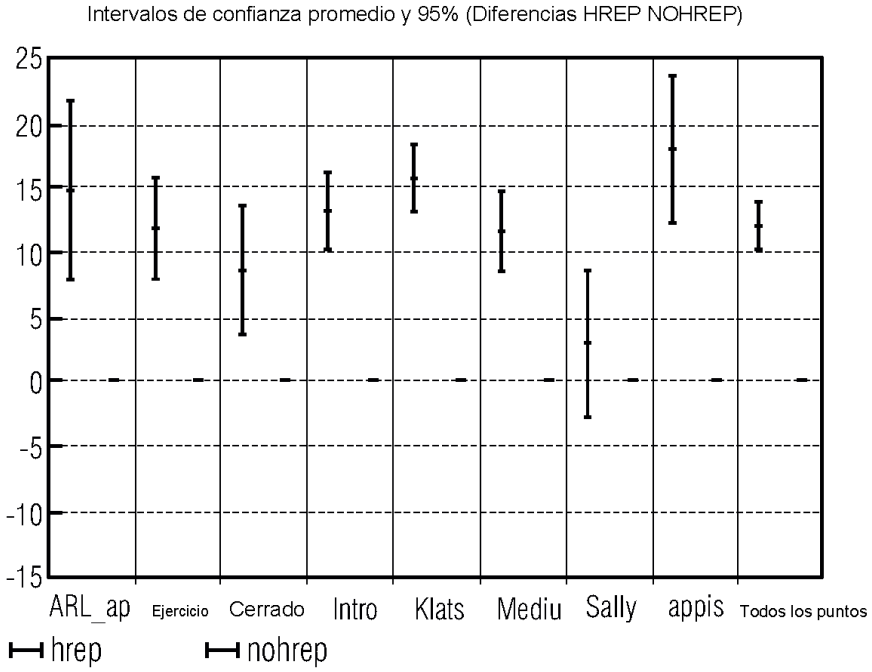

Fig. 17f

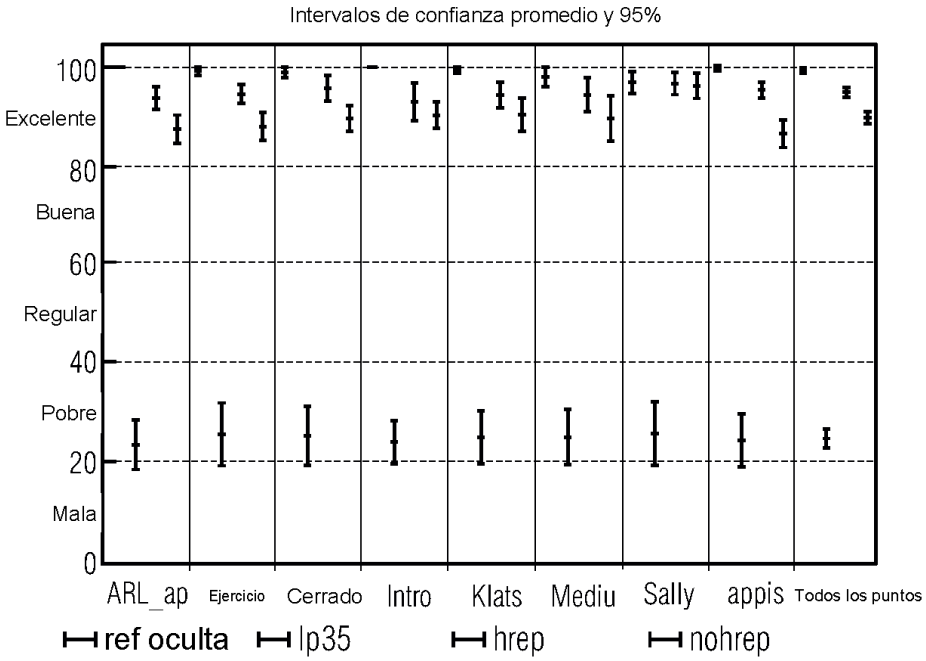

Fig. 17g

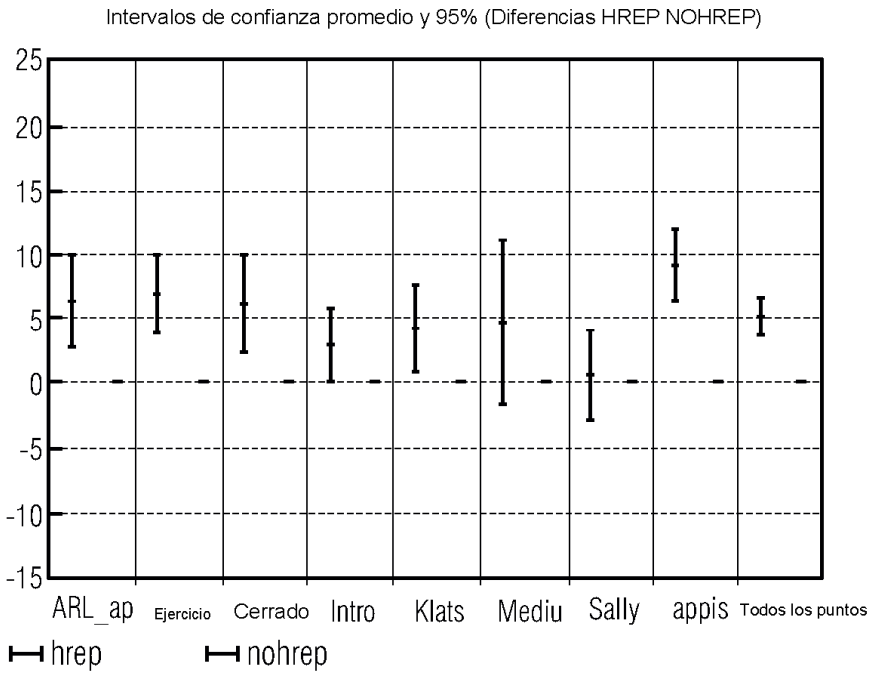

Fig. 17h