



(12)发明专利

(10)授权公告号 CN 105474309 B

(45)授权公告日 2019.08.23

(21)申请号 201480041458.X

(22)申请日 2014.07.16

(65)同一申请的已公布的文献号  
申请公布号 CN 105474309 A

(43)申请公布日 2016.04.06

(30)优先权数据  
13177367.3 2013.07.22 EP  
13177365.7 2013.07.22 EP  
13177378.0 2013.07.22 EP  
13189284.6 2013.10.18 EP

(85)PCT国际申请进入国家阶段日  
2016.01.21

(86)PCT国际申请的申请数据  
PCT/EP2014/065299 2014.07.16

(87)PCT国际申请的公布数据  
W02015/011000 EN 2015.01.29

(73)专利权人 弗朗霍夫应用科学研究促进协会  
地址 德国慕尼黑

(72)发明人 克里斯蒂安·鲍斯  
克里斯蒂安·埃特尔

(74)专利代理机构 北京清亦华知识产权代理事  
务所(普通合伙) 11201  
代理人 宋融冰

(51)Int.Cl.  
G10L 19/008(2006.01)

(56)对比文件  
CN 102449689 A,2012.05.09,  
CN 102931969 A,2013.02.13,  
CN 101743586 A,2010.06.16,  
CN 101542597 A,2009.09.23,  
CN 102171755 A,2011.08.31,  
CN 101632118 A,2010.01.20,  
CN 102387005 A,2012.03.21,

审查员 颜博

权利要求书5页 说明书26页 附图17页

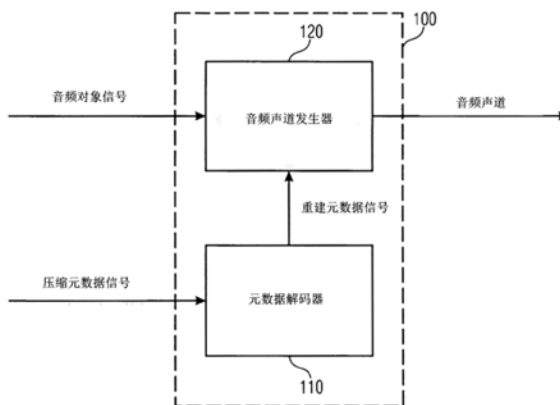
(54)发明名称

高效率对象元数据编码的装置及方法

(57)摘要

提供了一种用于产生至少一个音频声道的装置(100)。装置(100)包含用于接收至少一个压缩元数据信号的元数据解码器(110)。每一压缩元数据信号包含多个第一元数据样本。每一压缩元数据信号的第一元数据样本指示与至少一个音频对象信号中的音频对象信号相关联的信息。元数据解码器(110)用于产生至少一个重建元数据信号,使得每一重建元数据信号包含至少一个压缩元数据信号中的一个压缩元数据信号的第一元数据样本以及进一步包含多个第二元数据样本。而且,元数据解码器(110)根据重建元数据信号的至少两个第一元数据样本来产生每一重建元数据信号的每一第二元数据样本。而且,装置(100)包含音频声道发生器(120),其根据至少一个音频对象信号以及至少一个重建元数据信

号而产生至少一个音频声道。此外,提供了一种用于产生编码音频信息的装置,编码音频信息包含至少一个编码音频信号以及至少一个压缩元数据信号。



CN 105474309 B

1. 一种用于产生一个或多个音频声道的装置(100),其中该装置包含:

元数据解码器(110),用于接收一个或多个压缩元数据信号,其中每一所述一个或多个压缩元数据信号包含多个第一元数据样本,其中每一所述一个或多个压缩元数据信号的所述第一元数据样本指示与一个或多个音频对象信号中的音频对象信号相关联的信息,其中所述元数据解码器(110)用于产生一个或多个重建元数据信号,使得每一所述一个或多个重建元数据信号包含所述一个或多个压缩元数据信号中的压缩元数据信号的所述第一元数据样本,而所述重建元数据信号与所述压缩元数据信号相关联,每一所述一个或多个重建元数据信号进一步包含多个第二元数据样本,其中所述元数据解码器(110)用于通过产生针对所述重建元数据信号的多个近似元数据样本,产生每一所述一个或多个重建元数据信号的所述第二元数据样本,其中所述元数据解码器(110)用于根据所述重建元数据信号的至少两个所述第一元数据样本,产生每一所述多个近似元数据样本,

音频声道发生器(120),用于根据所述一个或多个音频对象信号以及所述一个或多个重建元数据信号而产生所述一个或多个音频声道,

其中所述元数据解码器(110)用于接收针对所述一个或多个压缩元数据信号的压缩元数据信号的多个差值,以及将每一所述多个差值与所述重建元数据信号的所述多个近似元数据样本中的其中一个相加,以获得所述重建元数据信号的所述第二元数据样本,而所述重建元数据信号与所述压缩元数据信号相关联。

2. 如权利要求1所述的装置(100),其中所述元数据解码器(110)用于通过对所述一个或多个压缩元数据信号中的其中一个作升采样,产生每一所述一个或多个重建元数据信号,其中所述元数据解码器(110)用于根据所述重建元数据信号的至少两个所述第一元数据样本进行线性内插,产生每一所述一个或多个重建元数据信号的每一所述第二元数据样本。

3. 如权利要求1所述的装置(100),

其中所述元数据解码器(110)用于接收针对所述一个或多个压缩元数据信号中的压缩元数据信号的所述多个差值,其中每一所述多个差值为指派至与所述压缩元数据信号相关联的所述重建元数据信号的所述多个近似元数据样本中的其中一个的所接收的差值,

其中所述元数据解码器(110)将每一所述多个所接收的差值加上与所述所接收的差值相关联的所述近似元数据样本,以获得所述重建元数据信号的一个所述第二元数据样本,

其中当所述多个所接收的差值没有一个与所述近似元数据样本相关联时,所述元数据解码器(110)用于根据一个或多个所述多个所接收的差值确定近似差值,所述近似差值针对与所述压缩元数据信号相关联的所述重建元数据信号的每一所述多个近似元数据样本,

其中所述元数据解码器(110)将每一所述多个近似差值与所述近似差值的所述近似元数据样本相加,以获得所述重建元数据信号的另一个所述第二元数据样本。

4. 如权利要求1所述的装置(100),其中所述一个或多个重建元数据信号中的至少一个重建元数据信号包含所述一个或多个音频对象信号中的其中一个的位置信息,或者包含关于所述一个或多个音频对象信号中的其中一个的位置信息的缩放表现,以及

其中所述音频声道发生器(120)用于根据所述一个或多个音频对象信号的所述一个音频对象信号以及所述位置信息,产生所述一个或多个音频声道中的至少一个音频声道。

5. 如权利要求1所述的装置(100),其中所述一个或多个重建元数据信号中的至少一个

重建元数据信号包含所述一个或多个音频对象信号中的其中一个的音量,或者包含所述一个或多个音频对象信号中的所述一个音频对象信号的所述音量的缩放表现,以及

其中所述音频声道发生器(120)用于根据所述一个或多个音频对象信号的所述一个音频对象信号以及所述音量,产生所述一个或多个音频声道中的至少一个音频声道。

6.如权利要求1所述的装置(100),其中所述装置(100)用于接收随机存取信息,其中针对每一所述一个或多个压缩元数据信号,所述随机存取信息指示所述压缩元数据信号的存取信息部分,其中所述元数据信号的至少一个其他信号部分并非由所述随机存取信息所指示,以及其中所述元数据解码器(110)用于根据所述压缩元数据信号的所述存取信息部分的所述第一元数据样本,但不根据所述压缩元数据信号的任何其他信号部分的任何其他第一元数据样本,产生所述一个或多个重建元数据信号中的其中一个。

7.一种用于产生编码音频信息的装置(250),所述编码音频信息包含一个或多个编码音频信号以及一个或多个压缩元数据信号,其中所述装置(250)包含:

元数据编码器(210),用于接收一个或多个原始元数据信号,其中每一所述一个或多个原始元数据信号包含多个元数据样本,其中每一所述一个或多个原始元数据信号的所述元数据样本指示与一个或多个音频对象信号中的音频对象信号相关联的信息,其中所述元数据编码器(210)用于产生所述一个或多个压缩元数据信号,使得每一所述一个或多个压缩元数据信号包含所述一个或多个原始元数据信号中的原始元数据信号的所述元数据样本的至少两个的第一组,所述压缩元数据信号与所述原始元数据信号相关联,以及使得所述压缩元数据信号不包含所述一个所述原始元数据信号的另外至少两个所述元数据样本的第二组的任何元数据样本,以及

音频编码器(220),用于编码所述一个或多个音频对象信号以获得所述一个或多个编码音频信号,

其中所述一个或多个原始元数据信号中的原始元数据信号所包含且也是所述压缩元数据信号所包含的每一所述元数据样本为多个第一元数据样本中的其中一个,每一所述第一元数据样本与所述原始元数据信号相关联,

其中所述一个或多个原始元数据信号中的原始元数据信号所包含且不是所述压缩元数据信号所包含的每一所述元数据样本为多个第二元数据样本中的其中一个,每一所述第二元数据样本与所述原始元数据信号相关联,

其中所述元数据编码器(210)用于根据所述一个或多个原始元数据信号中的所述其中一个的至少两个所述第一元数据本来执行线性内插,以针对一个所述原始元数据信号中的每一所述多个第二元数据样本产生近似元数据样本,以及

其中所述元数据编码器(210)用于针对所述一个或多个原始元数据信号中的所述其中一个的每一所述多个第二元数据样本而产生差值,使得所述差值指示所述第二元数据样本以及所述第二元数据样本的所述近似元数据样本之间的差值。

8.如权利要求7所述的装置(250),其中所述元数据编码器(210)用于针对所述一个或多个原始元数据信号中的所述其中一个的所述多个第二元数据样本的至少一个所述差值,判断每一所述至少一个所述差值是否大于阈值。

9.如权利要求7所述的装置(250),

其中所述元数据编码器(210)用于使用第一位数将所述一个或多个压缩元数据信号中

的其中一个的一个或多个所述元数据样本进行编码,其中所述一个或多个压缩元数据信号中的所述其中一个的每一所述一个或多个元数据样本表示整数,

其中所述元数据编码器(210)用于使用第二位数将所述多个第二元数据样本的一个或多个所述差值进行编码,其中所述多个第二元数据样本的所述一个或多个所述差值表示整数,以及

其中所述第二位数小于所述第一位数。

10. 如权利要求7所述的装置(250),

其中所述一个或多个原始元数据信号中的至少一个包含所述一个或多个音频对象信号中的其中一个的位置信息,或者包含所述一个或多个音频对象信号中的其中一个的所述位置信息的缩放表现,以及

其中所述元数据编码器(210)用于根据所述一个或多个原始元数据信号中的所述至少一个来产生所述一个或多个压缩元数据信号中的至少一个。

11. 如权利要求7所述的装置(250),

其中所述一个或多个原始元数据信号中的至少一个包含所述一个或多个音频对象信号中的其中一个的音量,或者包含所述一个或多个音频对象信号中的所述其中一个的所述音量的缩放表现,以及

其中所述元数据编码器(210)用于根据所述一个或多个原始元数据信号中的所述至少一个来产生所述一个或多个压缩元数据信号中的至少一个。

12. 一种用于编码和解码的系统,包含:

如权利要求7至11中任一项所述的装置(250),用于产生编码音频信息,所述编码音频信息包含一个或多个编码音频信号以及一个或多个压缩元数据信号;以及

如权利要求1至6中任一项所述的装置(100),用于接收所述一个或多个编码音频信号以及所述一个或多个压缩元数据信号,并根据所述一个或多个编码音频信号以及所述一个或多个压缩元数据信号产生一个或多个音频声道。

13. 一种用于产生一个或多个音频声道的方法,其中所述方法包含:

接收一个或多个压缩元数据信号,其中每一所述一个或多个压缩元数据信号包含多个第一元数据样本,其中每一所述一个或多个压缩元数据信号的所述第一元数据样本指示与一个或多个音频对象信号中的音频对象信号相关联的信息,

产生一个或多个重建元数据信号,使得每一所述一个或多个重建元数据信号包含所述一个或多个压缩元数据信号中的压缩元数据信号的所述第一元数据样本,所述重建元数据信号与所述压缩元数据信号相关联,以及每一所述一个或多个重建元数据信号进一步包含多个第二元数据样本,其中产生所述一个或多个重建元数据信号的步骤包含通过产生所述重建元数据信号的多个近似元数据样本而产生所述每一所述一个或多个重建元数据信号的所述第二元数据样本的步骤,其中产生每一所述多个近似元数据样本根据所述重建元数据信号的至少两个所述第一元数据样本执行,以及

根据所述一个或多个音频对象信号以及所述一个或多个重建元数据信号产生所述一个或多个音频声道,

其中所述方法进一步包含接收针对所述一个或多个压缩元数据信号的压缩元数据信号的多个差值,以及将每一所述多个差值与所述重建元数据信号的所述多个近似元数据样

本中的其中一个相加,以获得所述重建元数据信号的所述第二元数据样本,而所述重建元数据信号与所述压缩元数据信号相关联。

14. 一种用于产生编码音频信息的方法,所述编码音频信息包含一个或多个编码音频信号以及一个或多个压缩元数据信号,其中所述方法包含:

接收一个或多个原始元数据信号,其中每一所述一个或多个原始元数据信号包含多个元数据样本,其中每一所述一个或多个原始元数据信号的所述元数据样本指示与一个或多个音频对象信号中的音频对象信号相关联的信息,

产生所述一个或多个压缩元数据信号,使得每一所述一个或多个压缩元数据信号包含所述一个或多个原始元数据信号中的原始元数据信号的至少两个所述元数据样本的第一组,所述压缩元数据信号与所述原始元数据信号相关联,以及使得所述压缩元数据信号不包含所述一个所述原始元数据信号的另外至少两个所述元数据样本的第二组的任何元数据样本,以及

编码所述一个或多个音频对象信号以获得所述一个或多个编码音频信号,

其中所述一个或多个原始元数据信号中的原始元数据信号所包含且也是所述压缩元数据信号所包含的每一所述元数据样本为多个第一元数据样本中的其中一个,每一所述第一元数据样本与所述原始元数据信号相关联,

其中所述一个或多个原始元数据信号中的原始元数据信号所包含且不是所述压缩元数据信号所包含的每一所述元数据样本为多个第二元数据样本中的其中一个,每一所述第二元数据样本与所述原始元数据信号相关联,

其中所述方法进一步包含根据所述一个或多个原始元数据信号中的所述其中一个的至少两个所述第一元数据本来执行线性内插,以针对一个所述原始元数据信号中的每一所述多个第二元数据样本产生近似元数据样本,以及

其中所述方法进一步包含针对所述一个或多个原始元数据信号中的所述其中一个的每一所述多个第二元数据样本而产生差值,使得所述差值指示所述第二元数据样本以及所述第二元数据样本的所述近似元数据样本之间的差值。

15. 一种计算机可读存储介质,包括计算机程序,当所述计算机程序于计算机或者信号处理器上执行时,所述计算机程序用于实现权利要求13或14所述的方法。

16. 一种对音频输入数据(101)进行编码以获得音频输出数据(501)的装置,包含:

输入界面(1100),用于接收多个音频声道、多个音频对象以及关于一个或多个所述多个音频对象的元数据;

混合器(200),用于混合所述多个音频对象以及所述多个音频声道以获得多个预混合声道,每一所述多个预混合声道包含音频声道的音频数据以及至少一个音频对象的音频数据;以及

权利要求7至11的任一项所述的装置(250),

其中权利要求7至11的任一项所述的装置(250)的所述音频编码器(220)为对核心编码器输入数据进行核心编码的核心编码器(300),以及

其中权利要求7至11的任一项所述的装置(250)的所述元数据编码器(210)为对与所述多个音频对象中的一个或多个相关的所述元数据进行压缩的元数据压缩器(400)。

17. 一种对编码音频数据进行解码的装置,包含:

输入界面(1100),用于接收所述编码音频数据,所述编码音频数据包含多个编码声道、或者多个编码对象、或者关于所述多个编码对象的压缩元数据;以及

权利要求1至6的任一项所述的装置(100),

其中权利要求1至6的任一项所述的装置(100)的所述元数据解码器(110)为对所述压缩元数据进行解压缩的元数据解压缩器(1400);

其中权利要求1至6的任一项所述的装置(100)的所述音频声道发生器(120)包含用于解码所述多个编码声道以及所述多个编码对象的核心解码器(1300),

其中所述音频声道发生器(120)进一步包含对象处理器(1200),所述对象处理器(1200)用于使用解压缩元数据处理多个解码对象,以从所述多个解码对象以及解码声道获得包含音频数据的多个输出声道(1205),以及

其中所述音频声道发生器(120)进一步包含后置处理器(1700),所述后置处理器(1700)用于将所述多个输出声道(1205)转换成输出格式。

## 高效率对象元数据编码的装置及方法

### 技术领域

[0001] 本发明涉及音频编码/解码,特别地涉及空间音频编码以及空间音频对象编码,更特别地涉及高效率对象元数据编码。

### 背景技术

[0002] 空间音频编码工具是此技术领域中所熟知的,例如,在环绕MPEG标准中已有标准化规范。空间音频编码从原始输入声道开始,例如在再现装备中根据其位置而识别的五个或七个声道,即左声道、中间声道、右声道、左环绕声道、右环绕声道以及低频增强声道。空间音频编码器通常从原始声道得到至少一个降混合声道,以及另外得到关于空间线索的参数数据,例如声道相干数值的声道间水平差异、声道间相位差异、声道间时间差异等等。至少一个降混合声道与指示空间线索的参数化辅助信息(parametric side information,或称为参数边信息、参数侧信息或参数侧边信息)一起传送到空间音频解码器,空间音频解码器解码降混声道以及相关参数数据,最后获得为原始输入声道的近似版本的输出声道。声道在输出装备中的放置通常为固定,例如,5.1声道格式或7.1声道格式等等。

[0003] 此种基于声道的音频格式广泛使用于储存或者传送多声道音频内容,而每一个声道关于在给定位置的特定扬声器。这些种类格式的忠实再现,需要扬声器设备,其中扬声器放置在与音频信号生产期间使用的扬声器相同的位置。增加扬声器数量可改进真实三维虚拟现实场景,但是满足此要求是越来越困难的,尤其是在家庭环境中,像是客厅。

[0004] 可用于对象为基础的方法来克服对特殊扬声器设备的需求,在以对象为基础的方法中扬声器信号特别针对播放方案来渲染。

[0005] 例如,空间音频对象编码工具是此技术领域中所熟知的且在MPEG SAOC(SAOC=spatial audio object coding空间音频对象编码)标准中已成标准。相比于空间音频编码从原始声道开始,空间音频对象编码从非自动专为特定渲染再现装备的音频对象开始。代替地,音频对象在再现场景中的位置可变化,且可由使用者通过将特定的渲染信息输入至空间音频对象编码解码器来确定。可选地或另外,渲染信息,即在再现装备中特定音频对象待放置的位置信息,以额外的辅助信息或元数据来传送。为了获得特定的数据压缩,由SAOC编码器来编码多个音频对象,SAOC编码器根据特定的降混合信息来降混合对象以从输入对象计算至少一个传输声道。此外,SAOC编码器计算参数化辅助信息,其代表对象间线索,例如对象水平差异(OLD)、对象相干数值等等。当在空间音频编码(SAC)中,对象间参数数据针对单独时间平铺/频率平铺来计算,即,针对音频信号的特定帧(例如,1024或2048个样本),考虑多个频带(例如24、32或64个频带等等),使得对于每一帧以及每一频带皆存在参数数据。作为举例,当音频片具有20个帧且当每一帧细分成32个频带,则时间/频率平铺的数量为640。

[0006] 在以对象为基础的方法中,以分离式音频对象来描述音场。此需要对象元数据,其描述在3D空间中每一个声源的时变位置。

[0007] 在现有技术中,第一元数据编码概念为空间声音描述交换格式(SpatDIF),而音频

场景描述格式目前尚在开发中[1]。音频场景描述格式为以对象为基础的声音场景交换格式,其并没有提供任何压缩对象轨迹的方法。SpatDIF将以文字为基础的开放性声音控制(OSC)格式使用于对象元数据的结构[2]。然而,简单以文字为基础的表现并非为对象轨迹的压缩传输的选项。

[0008] 在现有技术中,另一个元数据概念为音频场景描述格式(ASDF)[3],其是具有相同的缺点的以文字为基础的解决方案。此数据通过同步多介质集成语言(SMIL)的延伸所建构,该同步多介质集成语言(SMIL)为可延伸标记式语言(XML)[4,5]的子集合。

[0009] 在现有技术中的另一个元数据概念为场景的音频二进制格式(AudioBIFS),为MPEG-4标准的一部分的二进制格式[6,7]。其高度关于基于XML的虚拟现实建模语言(VRML),其已开发应用于音频虚拟3D场景以及交互式虚拟现实[8]。复杂的AudioBIFS标准使用场景图以指定对象移动的路径。AudioBIFS主要的缺点在于并非设计用于实时操作,其中会使有限的系统延迟并且需要随机读取数据流。此外,对象位置的编码不运用受限的听者的定位能力。在音频虚拟场景中的听者有固定位置时,则对象数据可量化成较低的位数[9]。因此,应用于AudioBIFS的对象元数据的编码对于数据压缩是无效的。

[0010] 如果能提供改善的高效率的对象元数据编码概念,将会获得高度的赞赏。

## 发明内容

[0011] 本发明的目的用于提供改善的高效率的对象元数据编码的概念。

[0012] 本发明提供一种用于产生至少一个音频声道的装置。该装置包含元数据解压缩器,用于接收至少一个压缩元数据信号。每一个压缩元数据信号包含多个第一元数据样本。每一个压缩元数据信号中的第一元数据样本指示与至少一个音频对象信号中的音频对象信号相关联的信息。元数据解码器用于产生至少一个重建元数据信号,使得每一个重建元数据信号包含至少一个压缩元数据信号中的其中一个的多个第一元数据样本以及进一步包含多个第二元数据样本。元数据解码器用于根据重建元数据信号的至少两个第一元数据样本,产生每一个重建元数据信号的每一个第二元数据样本。此外,该装置包含音频声道发生器,音频声道发生器用于根据至少一个音频对象信号以及至少一个重建元数据信号而产生至少一个音频声道。

[0013] 此外,本发明提供一种用于产生编码音频信息的装置,该编码音频信息包含至少一个编码音频信号以及至少一个压缩元数据信号。此装置包含:元数据编码器,用于接收至少一个原始元数据信号。每一个原始元数据信号包含多个元数据样本。每一个原始元数据信号中的元数据样本指示与至少一个音频对象信号中的音频对象信号相关联的信息。元数据编码器用于产生至少一个压缩元数据信号,使得每一压缩元数据信号包含一个原始元数据信号的至少两个元数据样本的第一组,以及使得压缩元数据信号不包含所述一个原始元数据信号的另外至少两个元数据样本的第二组的任何元数据样本。此外,该装置包含音频编码器,该音频编码器用于编码至少一个音频对象信号以获得至少一个编码音频信号。

[0014] 此外,提供了一种系统。该系统包含用于产生编码音频信息的装置,该编码音频信息包含至少一个编码音频信号以及至少一个压缩元数据信号,如上所述。此外,该系统包含用于接收至少一个编码音频信号以及至少一个压缩元数据信号的装置,该装置用于根据至少一个编码音频信号以及至少一个压缩元数据信号产生至少一个音频声道,如上所述。

[0015] 根据实施例,提供用于对象元数据的数据压缩概念,其达成用于具有有限的数据速率的传输声道为有效的压缩机制。此外,对于纯方位变化的良好压缩率得以实现,例如照相机旋转。此外,该提供的概念支持不连续的轨迹,例如位置的跳跃。此外,也能实现低解码复杂度。此外,可实现有限的重新初始化时间下的随机存取。

[0016] 此外,本发明提供一种用于产生至少一个音频声道的方法。该方法包含:

[0017] -接收至少一个压缩元数据信号,其中每一个压缩元数据信号包含多个第一元数据样本,其中每一个压缩元数据信号中的第一元数据样本指示与至少一个音频对象信号中的音频对象信号相关联的信息;

[0018] -产生至少一个重建元数据信号,使得每一个重建元数据信号包含至少一个压缩元数据信号中的其中一个的第一元数据样本,以及进一步包含多个第二元数据样本,其中产生至少一个重建元数据信号的步骤包含根据重建元数据信号的至少两个第一元数据样本产生每一个重建元数据信号的每一个第二元数据样本的步骤;

[0019] -根据至少一个音频对象信号以及至少一个重建元数据信号产生至少一个音频声道。

[0020] 此外,提供了一种用于产生编码音频信息的方法,编码音频信息包含至少一个编码音频信号以及至少一个压缩元数据信号。此方法包含:

[0021] -接收至少一个原始元数据信号,其中每一原始元数据信号包含多个元数据样本,其中每一原始元数据信号的元数据样本指示与至少一个音频对象信号中的音频对象信号相关联的信息;

[0022] -产生至少一个压缩元数据信号,使得每一压缩元数据信号包含一个原始元数据信号的至少两个元数据样本的第一组,以及使得压缩元数据信号不包含所述一个原始元数据信号的另外至少两个元数据样本的第二组的任何元数据样本;

[0023] -编码至少一个音频对象信号以获得至少一个编码音频信号。

[0024] 此外,本发明提供一种计算机程序,当此计算机程序于计算机或者信号处理器上执行时,计算机程序用于实现上述的方法。

## 附图说明

[0025] 下面参考附图讨论本发明的实施例,其中:

[0026] 图1示出根据实施例的用于产生至少一个音频声道的装置;

[0027] 图2示出根据实施例的用于产生编码音频信息的装置,编码音频信息包含至少一个编码音频信号以及至少一个压缩元数据信号;

[0028] 图3示出根据实施例的系统;

[0029] 图4示出在从原点开始的三维空间中通过方位角、仰角以及半径表示的音频对象的位置;

[0030] 图5示出音频声道发生器采用的音频对象以及扬声器装备的位置;

[0031] 图6示出根据实施例的元数据编码;

[0032] 图7示出根据实施例的元数据解码;

[0033] 图8示出根据另一实施例的元数据编码;

[0034] 图9示出根据另一实施例的元数据解码;

- [0035] 图10示出根据另一实施例的元数据编码;
- [0036] 图11示出根据另一实施例的元数据解码;
- [0037] 图12示出3D音频编码器的第一实施例;
- [0038] 图13示出3D音频解码器的第一实施例;
- [0039] 图14示出3D音频编码器的第二实施例;
- [0040] 图15示出3D音频解码器的第二实施例;
- [0041] 图16示出3D音频编码器的第三实施例;
- [0042] 图17示出3D音频解码器的第三实施例。

### 具体实施方式

[0043] 图2示出根据实施例的用于产生编码音频信息的装置250,编码音频信息包含至少一个编码音频信号以及至少一个压缩元数据信号。

[0044] 装置250包含元数据编码器210,用于接收至少一个原始元数据信号。每一个原始元数据信号包含多个元数据样本。至少一个原始元数据信号中的每一个的元数据样本指示与至少一个音频对象信号中的音频对象信号相关联的信息。元数据编码器210用于产生至少一个压缩元数据信号,使得每一压缩元数据信号能包含一个原始元数据信号的至少两个元数据样本的第一组,以及使得压缩元数据信号不包含该一个原始元数据信号的另外至少两个元数据样本的第二组的任何元数据样本。

[0045] 此外,装置250包含音频编码器220,用于编码至少一个音频对象信号以获得至少一个编码音频信号。例如,音频声道发生器可包含SAOC编码器,该SAOC编码器根据现有技术编码至少一个音频对象信号,以获得至少一个SAOC传输声道并作为至少一个编码音频信号。各种其他用于编码至少一个音频对象声道的编码技术可替换或额外地用于编码所述至少一个音频对象声道。

[0046] 图1示出根据实施例的用于产生至少一个音频声道的装置100。

[0047] 装置100包含元数据解码器110,用于接收至少一个压缩元数据信号。每一个压缩元数据信号包含多个第一元数据样本。每一个压缩元数据信号的第一元数据样本指示与至少一个音频对象信号中的音频对象信号相关联的信息。元数据解码器110用于产生至少一个重建元数据信号,使得每一个重建元数据信号包含至少一个压缩元数据信号中的其中一个的第一元数据样本以及进一步包含多个第二元数据样本。此外,元数据解码器110用于根据重建元数据信号的至少两个第一元数据样本,产生每一个重建元数据信号的每一个第二元数据样本。

[0048] 此外,装置100包含音频声道发生器120,该音频声道发生器120用于根据至少一个音频对象信号以及至少一个重建元数据信号而产生至少一个音频声道。

[0049] 当参阅元数据样本时,应当注意的是,元数据样本的特征在于其元数据样本值以及与其相关的时间点。例如,此类时间点可与音序列或其相似物的起始相关。例如,指数n或k可辨识在元数据信号内的元数据样本的位置,并因此指示出(相关的)时间点(其与起始时间相关)。应当注意的是,当两个元数据样本与不同时间点相关时,该两个元数据样本不同于其他的元数据样本,即使当它们的元数据样本值相同时,有时也会出现这样的情况。

[0050] 上述的实施例基于以下发现:与音频对象信号相关联的(包含于元数据信号的)元

数据信息常变化缓慢。

[0051] 例如,元数据信号可指示音频对象的位置信息(例如用于定义音频对象的位置的方位角、仰角或半径)。可以假设音频对象的位置在大部分的时间不会改变或仅缓慢地改变。

[0052] 或者,元数据信号可例如指示音频对象的音量(例如增益),并且也可以假设音频对象的音量在大部分的时间缓慢地改变。

[0053] 基于这个原因,在每个时间点并不需要传递(完整的)元数据信息。相反地,(完整的)元数据信息仅在特定时间点传递,例如周期性地,例如在每N个时间点,例如在时间点0、N、2N、3N等。在解码器侧上,对于中间的时间点(例如时间点1、2...N-1),元数据可接着基于至少两个时间点的元数据样本进行近似。在解码器侧上,例如,时间点1、2...N-1的元数据样本可根据时间点0以及N的元数据样本进行近似,例如采用线性内插法。如前所述,此类方法基于以下发现:音频对象的元数据信息通常缓慢地改变。

[0054] 例如,在实施例中,三个元数据信号指定在3D空间中的音频对象的位置。元数据信号中的第一个可例如指定音频对象所在位置的方位角。元数据信号中的第二个可例如指定音频对象所在位置的仰角。元数据信号中的第三个可例如指定与音频对象距离相关的半径。

[0055] 请参阅图4,如图所示,方位角、仰角以及半径明确地定义在从原点开始的3D空间中的音频对象的位置。

[0056] 图4示出在从原点400开始的三维(3D)空间中通过方位角、仰角以及半径表示的音频对象的位置410。

[0057] 仰角例如指定从原点到对象位置的直线以及在xy平面(x轴以及y轴所定义的平面)上的直线的正交投影之间的角度。方位角,例如定义在x轴以及该正交投影之间的角度。通过指定方位角以及仰角,可定义出原点400以及音频对象的位置410之间的直线415。通过更进一步指定半径,可定义音频对象的精确位置410。

[0058] 在实施例中,方位角定义为: $-180^{\circ} < \text{方位角} \leq 180^{\circ}$ ,仰角定义为: $-90^{\circ} \leq \text{仰角} \leq 90^{\circ}$ ,半径的单位可例如定义为米[m](大于或等于0m)。

[0059] 在另一实施例中,可假设在xyz坐标系中的音频对象位置的所有x值大于或等于零,方位角的范围可定义为 $-90^{\circ} \leq \text{方位角} \leq 90^{\circ}$ ,仰角的范围可定义为: $-90^{\circ} \leq \text{仰角} \leq 90^{\circ}$ ,半径的单位可例如定义为米[m]。

[0060] 在另一实施例中,可调整元数据信号以使方位角的范围被定义为: $-128^{\circ} < \text{方位角} \leq 128^{\circ}$ 、仰角的范围被定义为: $-32^{\circ} \leq \text{仰角} \leq 32^{\circ}$ 以及半径可例如被定义为对数标度。在一些实施例中,原始元数据信号、压缩元数据信号以及重建元数据信号可包含位置信息的缩放表现和/或至少一个音频对象信号中的其中一个的音量的缩放表现。

[0061] 音频声道发生器120可例如用于根据至少一个音频对象信号以及重建元数据信号产生至少一个音频声道,其中重建元数据信号可例如指示音频对象的位置。

[0062] 图5示出音频声道发生器采用的音频对象以及扬声器装备的位置。xyz坐标系的原点500被示出。此外,第一音频对象的位置510以及第二音频对象的位置520被示出。此外,图5示出音频声道发生器120产生四个扬声器的四个音频声道的场景。音频声道发生器120采用四个扬声器511、512、513及514位于图5中示出的位置。

[0063] 在图5中,第一音频对象所在的位置510接近于采用的扬声器511及512的位置并远离扬声器513及514。因此,音频声道发生器120可产生四个音频声道,以使第一音频对象510通过扬声器511及512而不通过扬声器513及514播放。

[0064] 在其他实施例中,音频声道发生器120可产生四个音频声道,以使第一音频对象510通过扬声器511及512以高音量播放以及通过扬声器513及514以低音量播放。

[0065] 此外,第二音频对象所在的位置520接近于扬声器513及514采用的位置以及远离扬声器511和512。因此,音频声道发生器120可产生四个音频声道,以使第二音频对象520通过扬声器513及514而不通过扬声器511及512播放。

[0066] 在其他实施例中,音频声道发生器120可产生四个音频声道,以使第二音频对象520通过扬声器513及514以高音量播放以及通过扬声器511及512以低音量播放。

[0067] 在替代实施例中,仅两个元数据信号被用于指定音频对象的位置。举例来说,当假设所有音频对象位于单一平面时,例如仅方位角以及半径可被指定。

[0068] 在其他实施例中,每个音频对象仅有单一元数据信号被编码以及传递作为位置信息。例如,仅方位角可被指定作为音频对象的位置信息(例如可假设所有音频对象在与中心点相隔相同距离的相同平面,因此被假设为具有相同半径)。方位角信息可例如用于确定音频对象的位置接近于左扬声器以及远离右扬声器。在此情况下,音频声道发生器120可例如产生至少一个音频声道,以使音频对象通过左扬声器而不通过右扬声器播放。

[0069] 例如,矢量基幅值相移(Vector Base Amplitude Panning,VBAP)可被用于确定在扬声器的每个音频声道内的音频对象信号的权重(例如,请见参考文献[12])。例如关于VBAP,假定音频对象与虚拟源相关。

[0070] 在实施例中,另一元数据信号可指定每个音频对象的音量,例如增益(例如以分贝[dB]表示)。

[0071] 例如,在图5中,第一增益值可通过在位置510的第一音频对象的另一元数据信号指定,第二增益值通过位置520的第二音频对象的另一元数据信号指定,其中第一增益值大于第二增益值。在此情况下,扬声器511及512播放的第一音频对象的音量大于扬声器513及514播放的第二音频对象的音量。

[0072] 实施例也假定音频对象的此类增益值通常缓慢地改变。因此,不需要在每个时间点传送此类元数据信息。相反地,仅在特定时间点传送元数据信息。在中间的时间点,元数据信息可例如使用上述的元数据样本以及随后的元数据样本被近似并且被传送。例如,线性内插法可用于中间值的近似。例如,对于该元数据未被传送的时间点,每个音频对象的增益、方位角、仰角和/或半径被近似。

[0073] 通过此方式,可有效节省元数据传输的速率。

[0074] 图3示出根据实施例的系统。

[0075] 该系统包含装置250,用于产生编码音频信息,编码音频信息包含至少一个编码音频信号以及至少一个压缩元数据信号,如上所述。

[0076] 此外,该系统包含装置100,用于接收至少一个编码音频信号以及至少一个压缩元数据信号,并根据至少一个编码音频信号以及至少一个压缩元数据信号产生至少一个音频声道,如上所述。

[0077] 例如,当用于编码的装置250的确使用SAOC编码器用于编码至少一个音频对象时,

至少一个编码音频信号可通过用于产生至少一个音频声道的装置100通过根据现有技术采用SAOC解码器以获得至少一个音频对象信号进行解码。

[0078] 考虑对象位置仅作为元数据的示例,为了允许在有限的重新初始化时间可随机存取,而在实施例中提供定期重新传输所有对象的位置。

[0079] 根据实施例,装置100用于接收随机存取信息,其中针对每一个压缩元数据信号,随机存取信息指示压缩元数据信号的存取信号部分,其中元数据信号的至少一个其他信号部分并非由随机存取信息所指示,以及元数据解码器110用于根据压缩元数据信号的存取信号部分的第一元数据样本,但不根据压缩元数据信号的任何其他信号部分的任何其他第一元数据样本,产生至少一个重建元数据信号中的其中一个。换句话说,通过指定随机存取信息,每一个压缩元数据信号的一部分可以被指定,而元数据信号的其他部分没有被指定。在此情况中,仅压缩元数据信号的特定部分而无其他部分被重建作为重建元数据信号的其中一个。因此,针对特定的时间点进行重建是可能的,因为压缩元数据信号传送的第一元数据样本代表压缩元数据信号完整的元数据信息(然而对于其他时间点,元数据信息不会被传送)。

[0080] 图6示出根据实施例的元数据编码。根据实施例的元数据编码器210可用于实现图6所示出的元数据编码。

[0081] 在图6中, $s(n)$ 可表示原始元数据信号中的其中一个。举例来说, $s(n)$ 可例如代表音频对象中的其中一个的方位角的函数, $n$ 可指示时间(例如通过指示在原始元数据信号内的样本位置)。

[0082] 随时间变化轨迹分量 $s(n)$ 被以明显低于音频采样速率的采样速率进行采样(例如,等于或低于1:1024),并通过因子 $N$ 进行量化(请见611)以及降采样(请见612)。这产生表示为 $z(k)$ 的上述定期传送数字信号。

[0083]  $z(k)$ 为至少一个压缩元数据信号中的其中一个。例如, $\hat{s}(n)$ 的每 $N$ 个元数据样本也是压缩元数据信号 $z(k)$ 的元数据样本,在每 $N$ 个元数据样本之间的 $\hat{s}(n)$ 的其他 $N-1$ 个元数据样本并非为压缩元数据信号 $z(k)$ 的元数据样本。

[0084] 例如,假设在 $s(n)$ 内, $n$ 指示时间(例如通过指示在原始元数据信号内的样本位置),其中 $n$ 为正整数或0。(例如起始时间: $n=0$ )。 $N$ 为降采样因子。例如, $N=32$ 或任何其他适合的降采样因子。

[0085] 例如,在612的降样本用于从原始元数据信号中获得压缩元数据信号 $z$ ,可例如被实现,使得:

$$[0086] \quad z(k) = \hat{s}(k \cdot N); \text{其中 } k \text{ 为正整数或 } 0 (k=0, 1, 2, \dots)$$

[0087] 因此:

[0088]

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

[0089] 图7示出根据实施例的元数据解码。实施例中的元数据解码器110可被用于实现图7所示出的元数据解码。

[0090] 根据图7所示出的实施例,元数据解码器110用于通过升采样至少一个压缩元数据信号中的一个,产生每一个重建元数据信号,其中元数据解码器110用于根据重建元数据信

号的至少两个第一元数据样本进行线性内插,产生每一个重建元数据信号的每一个第二元数据样本。

[0091] 因此,每一个重建元数据信号包含其压缩元数据信号的所有元数据样本(该样本被称为至少一个压缩元数据信号的“第一元数据样本”)。

[0092] 额外的(“第二”)元数据样本通过执行升采样被加入于重建元数据信号内。升采样的步骤用于确定被加入于重建元数据信号内的额外的(“第二”)元数据样本的位置。

[0093] 通过执行线性内插,判断第二元数据样本的元数据样本值。线性内插基于压缩元数据信号的两个元数据样本被执行(该压缩元数据信号已成为重建元数据信号的第一元数据样本)。

[0094] 根据实施例,通过执行线性内插法升采样以及产生第二元数据样本,例如,可在单一步骤中进行。

[0095] 在图7中,反升采样处理(见721)结合于线性内插法(见722)导致原始信号的粗略近似。反升采样处理(见721)以及线性内插法(见722)可在单一步骤中进行。

[0096] 例如,在解码器侧上的升采样(见721)以及线性内插(见722)可被执行,使得:

[0097]  $s'(k \cdot N) = z(k)$ ;其中k为正整数或0

[0098]  $s'(k \cdot N + j) = z(k-1) + \frac{j}{N} [z(k) - z(k-1)]$ ;

[0099] 其中j为正整数,并且: $1 \leq j \leq N-1$ 。

[0100] 在此, $z(k)$ 为压缩元数据信号z的实际接收的元数据样本, $z(k-1)$ 为压缩元数据信号z的元数据样本,在实际接收元数据样本z(k)之前, $z(k-1)$ 被立即接收。

[0101] 图8示出根据另一实施例的元数据编码。根据实施例,元数据编码器210可用于实现图8所示出的元数据编码。

[0102] 在实施例中,如图8所示出,在元数据编码中,良好的结构可通过在延迟补偿输入信号以及线性内插粗略近似之间的编码差异指定。

[0103] 根据此实施例,与线性内插结合的反升采样过程也被执行作为编码器侧上的元数据编码的一部分(见图8中的621以及622)。此外,反升采样过程(见621)以及线性内插(见622)例如可在单一步骤中进行。

[0104] 如上所述,元数据编码器210用于产生至少一个压缩元数据信号,以使每一个压缩元数据信号包含一个或多个原始元数据信号中的原始元数据信号的至少两个元数据样本的第一组。该压缩元数据信号可被认为与原始元数据信号相关联。

[0105] 每一个元数据样本,其被包含于至少一个原始元数据信号中的原始元数据信号以及被包含于压缩元数据信号中并与原始元数据信号相关联,可被当作为多个第一元数据样本中的其中一个。

[0106] 此外,每一个元数据样本,其被包含于至少一个原始元数据信号中的原始元数据信号但不被包含于压缩元数据信号且与原始元数据信号相关联,则为多个第二元数据样本中的其中一个。

[0107] 根据图8的实施例,元数据编码器210用于根据至少一个原始元数据信号中的其中一个的至少两个第一元数据本来执行线性内插,以针对该原始元数据信号中的其中一个的多个第二元数据样本中的每一个产生近似元数据样本。

[0108] 此外,图8的实施例中,元数据编码器210用于针对至少一个原始元数据信号中的其中一个的每一个第二元数据样本产生差值,使得此差值代表第二元数据样本与该第二元数据样本的近似元数据样本之间的差异。

[0109] 在图10中所述的优选的实施例中,针对至少一个原始元数据信号中的其中一个的多个第二元数据样本的至少一个差值,元数据编码器210可以例如用于判断每一差值是否大于阈值。

[0110] 在图8的实施例中,近似元数据样本可例如通过对压缩元数据信号 $z(k)$ 执行升采样以及线性内插来确定(例如,作为信号 $s''(n)$ 的样本 $s''(n)$ )。升采样以及线性内插可作为在编码器侧上的元数据编码的一部分执行(见图8的621以及622),同样的方法也可见于721与722的元数据解码:

[0111]  $s''(k \cdot N) = z(k)$ ; 其中 $k$ 为正整数或0

[0112]  $s''(k \cdot N + j) = z(k-1) + \frac{j}{N} [z(k) - z(k-1)]$ ; 其中 $j$ 为整数且: $1 \leq j \leq N-1$ 。

[0113] 例如,在图8所示出的实施例中,当执行元数据编码时,差值可在630内针对差异被确定:

[0114]  $s(n) - s''(n)$ , 例如,  $(k-1) \cdot N < n < k \cdot N$ 的所有 $n$ 值,或者

[0115] 例如,  $(k-1) \cdot N < n \leq k \cdot N$ 的所有 $n$ 值。

[0116] 在实施例中,至少一个差值传送至元数据解码器。

[0117] 图9示出根据另一实施例的元数据解码。根据实施例的元数据解码器110可用于实现图9所示出的元数据解码。

[0118] 如上所述,每一个重建元数据信号包含至少一个压缩元数据信号中的压缩元数据信号的第一元数据样本。此重建元数据信号被认为与压缩元数据信号相关联。

[0119] 在图9所示的实施例中,元数据解码器110用于通过产生重建元数据信号的多个近似元数据样本,产生每一个重建元数据信号中的第二元数据样本,其中元数据解码器110用于根据重建元数据信号的至少两个第一元数据样本,产生多个近似元数据样本中的每一个。该近似元数据样本可例如通过线性内插产生,如图7所示出。

[0120] 根据图9所示出的实施例,元数据解码器110用于接收针对至少一个压缩元数据信号中的压缩元数据信号的多个差值。更进一步,元数据解码器110用于将每一个差值与重建元数据信号的近似元数据样本中的其中一个相加,以获得重建元数据信号的第二元数据样本,而重建元数据信号与压缩元数据信号相关联。

[0121] 对于差值已被接收的所有近似元数据样本,差值与近似元数据样本相加,以获得第二元数据样本。

[0122] 根据实施例,对于没有接收差值的近似元数据样本被作为重建元数据信号的第二元数据样本使用。

[0123] 然而,根据不同的实施例,如果近似元数据样本没有差值被接收,则针对近似元数据样本根据至少一个所接收的差值产生近似差值,以及将近似元数据样本与近似元数据样本相加,如下所述。

[0124] 根据图9所示出的实施例,所接收的差值与升采样元数据信号的对应的元数据样本相加(见730)。因此,当差值已被传输,相对应的内插元数据样本的差值可以被校正,如果

需要的话,以获得正确的元数据样本。

[0125] 请参阅图8的元数据编码,在优选实施例中,用于编码差值的位数少于用于编码元数据样本的位数。这些实施例基于以下发现:在大部分的时间里随后的(例如N个)元数据样本仅有略有变化。举例来说,如果一种元数据样本(例如以8位)被编码,则元数据样本可从256个不同的差值中取出一个差值。因为随后(例如N个)的元数据值通常有略微变化,仅对差值进行编码(例如以5位)被认为是足够的。因此,即使差值被传送,依然可减少传输的位数。

[0126] 在优选实施例中,至少一个差值被传送,并且每一个差值以少于每一个元数据样本的位数进行编码,其中每个差值皆为整数。

[0127] 根据实施例,元数据编码器110用于将该至少一个压缩元数据信号中的其中一个的该至少一个元数据样本以第一位数进行编码,其中至少一个压缩元数据信号中的其中一个的每一个元数据样本表示整数。此外,元数据编码器110用于将至少一个差值以第二位数进行编码,其中至少一个差值中的每一个表示整数,其中第二位数小于第一位数。

[0128] 在实施例中,元数据样本可例如代表以8位进行编码的方位角。例如,方位角为整数并且: $-90 \leq \text{方位角} \leq 90$ 。因此,方位角可采用181个不同的数值。如果可假设随后的(例如N个)方位角样本相差不大,例如不超过 $\pm 15$ ,则5位( $2^5 = 32$ )可足以编码差值。如果差值可代表整数,则判断差值自动地传送额外的待传送数值到适当的数值范围。

[0129] 例如,考虑第一音频对象的第一方位角值为 $60^\circ$ ,且随后的方位角值会在 $45^\circ$ 至 $75^\circ$ 之间改变的情况。此外,考虑第二音频对象的第二方位角值为 $-30^\circ$ ,且随后的方位角值会在 $-45^\circ$ 至 $-15^\circ$ 之间改变。通过确定第二音频对象以及第一音频对象两者的随后的数值的差值,第二方位角值以及第一方位角值两者的差值皆介于 $-15^\circ$ 至 $+15^\circ$ 的数值范围内,使得5位足以编码每一个差值以及使得编码差值的位序列对于第二方位角值的差值以及第一方位角值的差值具有相同的含义。

[0130] 在实施例中,对于没有元数据样本存在于压缩元数据信号中的每一个差值被传送到解码侧上。此外,根据实施例,对于没有元数据样本存在于压缩元数据信号中的每一个差值被元数据解码器接收并处理。然而,图10以及图11所示出的一些优选实施例实现不同的概念。

[0131] 图10示出根据另一实施例的元数据编码。根据实施例的元数据编码器210可用于实现图10所示出的元数据编码。

[0132] 在一些实施例中,如图10所示出,例如,对于未包含于压缩元数据信号的原始元数据信号的每个元数据样本,确定差值。例如,当在时间点 $n=0$ 以及 $n=N$ 的元数据样本包含于压缩元数据信号,但不包含时间点 $n=1$ 至 $n=N-1$ 之间的元数据样本时,则需确定时间点 $n=1$ 至 $n=N-1$ 的差值。

[0133] 然而,根据图10的实施例,接着在640执行多边形近似。元数据编码器210用于决定将传送多个差值中的哪一个以及决定是否传送所有的差值。

[0134] 例如,元数据编码器210可用于仅传送具有大于阈值的差值的差值。

[0135] 在另一实施例中,当差值与对应元数据样本的比值大于阈值时,元数据编码器210可用于仅传送该差值。

[0136] 在实施例中,元数据编码器210检查最大的绝对差值是否大于阈值。如果最大的绝

对差值大于阈值,则传送该差值,否则,不会传送任何的差值并结束检查。继续检查第二大的差值以及第三大差值等,直到所有的差值皆小于阈值。

[0137] 根据实施例,因为并非所有的差值皆一定会被传送,所以元数据编码器210不仅编码其(图10中的数值 $y_1[k] \cdots y_{N-1}[k]$ 中的其中一个)差值(的大小),并且传送与(图10中的数值 $x_1[k] \cdots x_{N-1}[k]$ 中的其中一个)差值相关联的原始元数据信号的元数据样本的信息。例如,元数据编码器210可编码与差值相关联的时间点。例如,元数据编码器210可编码介于1到N-1之间的数值以指示出与差值相关联并在压缩元数据信号中传送的介于0到N之间的元数据样本。根据差值,在多边形近似的输出处所列出的多个数值 $x_1[k] \cdots x_{N-1}[k] y_1[k] \cdots y_{N-1}[k]$ 并非意指所有数值一定会被传送,相反地,其意指没有一个、一个、一些或全部的数值对会被传送。

[0138] 在实施例中,元数据编码器210可处理部分(例如N个)连续的差值,并通过可变数量的量化的多边形点 $[x_i, y_i]$ 形成的多边形过程来近似每个部分。

[0139] 可预期必须足够精确地近似差异信号的多边形点的数量的平均值明显地小于N。此外,因为 $[x_i, y_i]$ 为较小的整数值,它们将以低位进行编码。

[0140] 图11示出根据另一实施例的元数据解码。根据实施例的元数据解码器110可用于实现图11所示出的元数据解码。

[0141] 在实施例中,元数据解码器110接收一些差值,并将这些差值与在730内的相对应的线性内插的元数据样本相加。

[0142] 在一些实施例中,元数据解码器110仅将所接收的差值与在730内的相对应的线性内插的元数据样本相加,并将没有接收到任何的差值的其他线性内插的元数据样本保持不变。

[0143] 然而,实现另一个概念的实施例如下所述。

[0144] 根据此类的实施例,元数据解码器110用于针对至少一个压缩元数据信号中的压缩元数据信号接收多个差值。每一个差值可称为“所接收的差值”。所接收的差值被指派为重建元数据信号的近似元数据样本中的其中一个,其中所接收的差值与压缩元数据信号相关联或从其构建,所接收的差值与压缩元数据信号相关联。

[0145] 请参阅已描述的图9,元数据解码器110用于将接收到的多个差值中的每一个与近似元数据样本相加,该近似元数据样本与所接收的差值相关联。重建元数据信号的第二元数据样本中的其中一个通过将所接收的差值与其近似元数据样本相加而获得。

[0146] 然而,针对一些(或者有时大部分)近似元数据样本,通常没有差值被接收。

[0147] 在一些实施例中,当多个所接收的差值没有一个与近似元数据样本相关联时,针对重建元数据信号的每一个近似元数据样本,元数据解码器110可用于例如根据多个所接收的差值中的至少一个来确定近似差值,该重建元数据信号与压缩元数据信号相关联。

[0148] 换句话说,对于所有的近似元数据样本而言,没有差值被接收时,近似差值仍根据至少一个所接收的差值所产生。

[0149] 元数据解码器110用于将多个近似差值的每一个与近似差值的近似元数据样本相加,以获得重建元数据信号的第二元数据样本中的另一个。

[0150] 然而,在另一实施例中,针对没有接收差值的元数据样本,元数据解码器110通过根据在步骤740内被接收的差值来执行线性内插,而对差值进行近似。

[0151] 举例来说,如果接收第一差值以及第二差值,则位于所接收的差值之间的差值可以被近似,例如采用线性内插。

[0152] 例如,当在时间点 $n=15$ 的第一差值具有差值 $d[15]=5$ 。以及当在时间点 $n=18$ 的第二差值具有差值 $d[18]=2$ 时,对于 $n=16$ 以及 $n=17$ 的差值可被线性近似作为 $d[16]=4$ 以及 $d[17]=3$ 。

[0153] 在另一实施例中,当元数据样本被包含于压缩元数据信号时,元数据样本的差值被假设为0,元数据解码器可基于被假设为0的元数据样本来执行没有被接收的差值的线性内插。

[0154] 例如,当在 $n=16$ 的单个差值 $d=8$ 被传送时以及当在 $n=0$ 以及 $n=32$ 的元数据样本在压缩元数据信号内被传送时,则在 $n=0$ 以及 $n=32$ 没有被传送的差值被假设为0。

[0155] 假设 $n$ 代表时间以及假设 $d[n]$ 为在时间点 $n$ 的差值。接着:

[0156]  $d[16]=8$  (接收的差值)

[0157]  $d[0]=0$  (假设的差值,在元数据样本存在于 $z(k)$ 时)

[0158]  $d[32]=0$  (假设的差值,在元数据样本存在于 $z(k)$ 时)

[0159] 则近似差值:

[0160]  $d[1]=0.5; d[2]=1; d[3]=1.5; d[4]=2; d[5]=2.5; d[6]=3; d[7]=3.5; d[8]=4;$

[0161]  $d[9]=4.5; d[10]=5; d[11]=5.5; d[12]=6; d[13]=6.5; d[14]=7; d[15]=7.5;$

[0162]  $d[17]=7.5; d[18]=7; d[19]=6.5; d[20]=6; d[21]=5.5; d[22]=5; d[23]=4.5; d[24]=4;$

[0163]  $d[25]=3.5; d[26]=3; d[27]=2.5; d[28]=2; d[29]=1.5; d[30]=1; d[31]=0.5。$

[0164] 在实施例中,所接收的近似差值与(在730中)相对应的线性内插样本相加。

[0165] 优选实施例被描述如下。

[0166] (对象)元数据编码器可例如使用给定大小 $N$ 的前瞻缓冲器来编码规则(子)采样轨迹值序列。一旦缓冲器被填充,整体数据区块被编码以及传送。所编码的对象数据可由两个部分组成,分别为内部编码对象数据以及包含每个部分的精细结构的任选差分数据部分。

[0167] 内部编码对象数据包含被采样于规则网格(每32个长度1024的音频帧)上的量化值 $z(k)$ 。布尔变量可被用于针对每个对象指示数值被单独指定或用于指示适用于所有对象的数值。

[0168] 解码器可用于通过线性内插从内部编码对象数据提取粗略轨迹。轨迹的精细结构由差分部分给定,该差分数据部分包含在输入轨迹以及线性内插之间的编码差值。针对方位角、仰角以及半径,多边形表现与不同的量化步骤结合,导致所预期的非相关性减少。

[0169] 多边形表现可从不使用递归的道格拉斯-普克算法[10,11]的变体中获得,其中道格拉斯-普克算法通过使用额外的中断循环(即对于所有对象和所述对象部件的多边形点的最大数量)使其不同于原始的方法。

[0170] 所产生的多边形点可使用可变的字长被编码于差分数据部分,该字长在比特流内被指定。额外的布尔变量指示相同数值的共同编码。



[0178]

```

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

```

[0179]

	}		
	<b>common_gain;</b>	<b>1</b>	<b>bslbf</b>
	if (common_gain) {		
	<b>default_gain;</b>	<b>7</b>	<b>tcimsbf</b>
	}		
	else {		
	for (o=1:num_objects) {		
	<b>gain_factor[o];</b>	<b>7</b>	<b>tcimsbf</b>
	}		
	}		
	}		
	else {		
	<b>position_azimuth;</b>	<b>8</b>	<b>tcimsbf</b>
	<b>position_elevation;</b>	<b>6</b>	<b>tcimsbf</b>
	<b>position_radius;</b>	<b>4</b>	<b>tcimsbf</b>
	<b>gain_factor;</b>	<b>7</b>	<b>tcimsbf</b>
	}		
	}		
	}		
	Note: iframe_period = ifperiod + 1;		

[0180] 以下描述根据实施例的差分对象数据。

[0181] 通过传送基于较少数量的样本点的多边形路线,实现较精确的近似。因此,非常稀疏的三维矩阵被传送,其中第一维度可以为对象索引,第二维度可由元数据分量(方位角,仰角,半径,和增益)形成,以及第三维度可为多个多边形采样点的帧索引。不需进一步的量测,哪个矩阵的元素包括数值的指示已需要num\_objects\*num\_components\*(iframe\_period-1)个位数。第一步骤为减少位数,可以是加入四个旗标,该四个旗标用于指示是否有至少一个数值属于四个分量中的其中一个。例如,可预期仅在少数的情况下会出现差分半径值或增益值。降低的三维矩阵的第三维度包含具有iframe\_period-1元素的向量。如果仅预期少量的多边形点,通过一组帧索引以及该组的基数来参数化向量会更有效率。例如,针对Nperiod=32帧的iframe\_period,最多数量的16个多边形点,此方法对Npoints<(32-log<sub>2</sub>(16))/log<sub>2</sub>(32)=5.6个多边形点会更有利。根据实施例,采用以下用于此类编码方案的句法:

[0182]

	位数	Mnemonic
differential_object_metadata() {		
<b>bits_per_point;</b>	<b>4</b>	<b>uimsbf</b>
<b>fixed_azimuth;</b>	<b>1</b>	<b>bslbf</b>
if (!fixed_azimuth) {		
for (o=1:num_objects) {		
<b>flag_azimuth;</b>	<b>1</b>	<b>bslbf</b>
if (flag_azimuth) {		
num_points = offset_data();		
<b>nbits_azimuth;</b>	<b>3</b>	<b>uimsbf</b>
for (p=1:num_points) {		
<b>differential_azimuth[o][p]; num_bits</b>		<b>tcimsbf</b>
}		
}		
}		
}		
<b>fixed_elevation;</b>	<b>1</b>	<b>bslbf</b>
if (!fixed_elevation) {		
for (o=1:num_objects) {		
<b>flag_elevation;</b>	<b>1</b>	<b>bslbf</b>
if (flag_elevation) {		
num_points = offset_data();		
<b>nbits_elevation;</b>	<b>3</b>	<b>uimsbf</b>
for (p=1:num_points) {		
<b>differential_elevation[o][p]; num_bits</b>		<b>tcimsbf</b>
}		
}		
}		
}		

[0183]

```

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

```

Note: num\_bits = nbits\_\* + 2;

	位数	Mnemonic
int offset_data() {		
<b>bitfield_syntax</b>	<b>1</b>	<b>bslbf</b>
if (bitfield_syntax) {		
<b>offset_bitfield</b>	<b>iframe_period-1</b>	<b>bslbf array</b>
num_points = sum(offset_bitfield)		
}		
else {		
[0184] <b>npoints;</b>	<b>bits_per_point</b>	<b>uimsbf</b>
num_points = npoints + 1;		
for (p=1:num_points) {		
<b>foffset[p];</b>	<b>ceil(log2(iframe_period-1</b>	<b>uimsbf</b>
	<b>))</b>	
}		
}		
return num_points;		
}		

[0185] 宏offset\_data() 编码多边形点的位置(帧偏移),作为简单的位域或使用上述概念。num\_bits数值允许较大的位置跳跃编码,同时,差分数据的其余部分以较小的字长进行编码。

[0186] 特别地,在实施例中,上述宏可例如具有下面的含义:

[0187] 根据实施例,object\_metadata() payloads的定义如下:

[0188] has\_differential\_metadata指示差分对象元数据是否存在。

[0189] 根据实施例,intracoded\_object\_metadata() payloads的定义如下:

[0190] ifperiod 定义在独立帧之间的帧数量。

[0191] common\_azimuth 指示共同方位角是否使用于所有的对象。

[0192] default\_azimuth 定义共同方位角的数值。

[0193] position\_azimuth 如果不存在共同方位角值,则传送每个对象的数值。

[0194] common\_elevation 指示共同仰角是否使用于所有的对象。

[0195] default\_elevation 定义共同仰角的数值。

[0196] position\_elevation 如果不存在共同仰角值,则传送每个对象的数值。

[0197] common\_radius 指示共同半径值是否被使用于所有的对象。

[0198] default\_radius 定义共同半径的值。

[0199] position\_radius 如果不存在共同半径值,则传送每个对象的数值。

[0200] common\_gain 指示共同增益值是否使用于所有的对象。

- [0201] default\_gain 定义共同增益因子值。
- [0202] gain\_factor 如果不存在共同增益因子值,则传送每个对象的数值。
- [0203] position\_azimuth 如果仅存在一个对象,这是它的方位角。
- [0204] position\_elevation 如果仅存在一个对象,这是它的仰角。
- [0205] position\_radius 如果仅存在一个对象,这是它的半径。
- [0206] gain\_factor 如果仅存在一个对象,这是它的增益因子。
- [0207] 根据实施例,differential\_object\_metadata() payloads的定义如下:
- [0208] bits\_per\_point 用于代表多边形点数量所需要的位数。
- [0209] fixed\_azimuth 用于指示所有对象的方位角值是否为固定不变的旗标。
- [0210] flag\_azimuth 用于指示方位角值是否有改变的每个对象的旗标。
- [0211] nbits\_azimuth 用于表示差值所需要的多少位。
- [0212] differential\_azimuth 在线性内插值以及实际值之间的差值。
- [0213] fixed\_elevation 用于指示所有对象的仰角值是否为固定不变的旗标。
- [0214] flag\_elevation 用于指示仰角值是否有改变的每个对象的旗标。
- [0215] nbits\_elevation 用于表示差值所需要的多少位。
- [0216] differential\_elevation 在线性内插值以及实际值之间的差值。
- [0217] fixed\_radius 用于指示所有对象的半径是否为固定不变的旗标。
- [0218] flag\_radius 用于指示半径是否有改变的每个对象的旗标。
- [0219] nbits\_radius 用于表示差值所需要的多少位。
- [0220] differential\_radius 在线性内插值以及实际值之间的差值。
- [0221] fixed\_gain 用于指示所有对象的增益因子是否为固定不变的旗标。
- [0222] flag\_gain 用于指示增益因子是否有改变的每个对象的旗标。
- [0223] nbits\_gain 用于表示差值所需要的多少位。
- [0224] differential\_gain 在线性内插值以及实际值之间的差值。
- [0225] 根据实施例,offset\_data() payloads的定义如下:
- [0226] bitfield\_syntax 用于指示具有多边形索引的向量是否存在于比特流内的旗标。
- [0227] offset\_bitfield 布尔数组,包含旗标,其针对iframe\_period的每个点是否为多边形点。
- [0228] npoints 多边形点数减1 ( $\text{num\_points} = \text{npoints} + 1$ )。
- [0229] foffset 在frame\_period ( $\text{frame\_offset} = \text{foffset} + 1$ ) 内的多边形点的时间片索引。
- [0230] 根据实施例,元数据可例如被传送作为每个音频对象在所定义的时间戳上的给定位置(例如方位角、仰角以及半径所指示的)。
- [0231] 在现有技术中,不存在结合一方面声道编码和另一方面对象编码的可变技术,使得可接受的音频质量以低比特率获得。
- [0232] 3D音频编码解码系统克服此限制,并且被描述如下。
- [0233] 图12示出根据本发明的实施例的3D音频编码器。3D音频编码器用于编码音频输入数据101以获得音频输出数据501。3D音频编码器包含输入界面,该输入界面用于接收CH所

指示的多个音频声道以及OBJ所指示的多个音频对象。此外,图12所示出的输入界面1100额外地接收与多个音频对象OBJ中的至少一个相关的元数据。此外,3D音频编码器包含混合器200,该混合器200用于混合多个对象以及多个声道以获得多个预混合的声道,其中每个预混合的声道包含声道的音频数据以及至少一个对象的音频数据。

[0234] 此外,3D音频编码器包含核心编码器300以及元数据压缩器400,其中核心编码器300用于核心编码核心编码器输入数据,元数据压缩器400用于压缩与多个音频对象中的至少一个相关的元数据。

[0235] 此外,3D音频编码器可包含模式控制器600,其在多个操作模式中的其中一个下控制混合器,核心编码器和/或输出界面500,其中核心编码器在第一模式用于编码多个音频声道以及通过输入界面1100接收而不受混合器影响(也即不通过混合器200混合)的多个音频对象。然而,在第二模式下混合器200是激活的,核心编码器编码多个混合的声道,也即区块200所产生的输出。在后者的情况下,优选地,不要再编码任何对象数据。代替地,指示音频对象位置的元数据已被使用于混合器200,以将对象渲染于元数据所指示的声道上。换句话说,混合器200使用与多个音频对象相关的元数据以预渲染多个音频对象,接着,所预渲染的音频对象与声道混和以获得在混合器输出处的混合声道。在此实施例中,可以不必传输任何对象,也可将音频对象应用于压缩元数据并作为区块400的输出。然而,如果并非输入界面1100的所有对象皆被混合而仅有特定数量的对象被混合,则仅剩余的没有被混合的对象以及相关元数据仍分别被传送到核心编码器300或元数据压缩器400。

[0236] 根据上述实施例中的其中一个,在图12中的元数据压缩器400为装置250的元数据编码器210,用于产生编码音频信息。此外,根据上述实施例中的其中一个,在图12中的混合器200以及核心编码器300一起形成装置250的音频编码器220,用于产生编码音频信息。

[0237] 图14示出3D音频编码器的另一实施例,3D音频编码器进一步包含SAOC编码器800。该SAOC编码器800用于从空间音频对象编码器输入数据中产生至少一个传输声道以及参数化数据。如图14所示出,空间音频对象编码器的输入数据为尚未经由预渲染器/混合器处理的对象。另外,当单独声道/对象编码在第一模式下是激活时,则预渲染器/混合器被绕过,所有被输入到输入界面1100的对象被SAOC编码器800编码。

[0238] 此外,如图14所示出,优选地,核心编码器300被实现作为USAC编码器,也即作为MPEG-USAC标准(USAC=联合语音以及音频编码)中所定义以及标准化的编码器。针对单独数据类型,描绘于图14中的3D音频编码器的所有输出为具有容器状结构的MPEG 4数据流。此外,元数据被指示作为“OAM”数据,图12中的元数据压缩器400对应于OAM编码器400,以获得输入到USAC编码器300内的压缩OAM数据,如图14所示出,USAC编码器300进一步包含输出界面,用于获得具有编码声道/对象数据以及压缩OAM数据的MP4输出数据流。

[0239] 根据上述实施例中的其中一个,在图14中的OAM编码器400为装置250的元数据编码器210,用于产生编码音频信息。此外,根据上述实施例中的其中一个,在图14中的SAOC编码器800以及USAC编码器300一起形成装置250的音频编码器220,用于产生编码音频信息。

[0240] 图16示出3D音频编码器的另一实施例,其中与图14相比,SAOC编码器可用于使用SAOC编码算法进行编码此模式下不被激活的在预渲染器/混合器200上所设置的声道,或者,SAOC编码器用于SAOC编码预渲染声道和对象。因此,在图16中的SAOC编码器800可对三种不同类型的输入数据进行操作,也即不具有任何预渲染对象的声道、声道以及多个预渲

染对象、或者单独对象。此外,优选地,在图16中提供另一OAM解码器420,以使SAOC编码器800用于处理使用与在编码器侧上相同的数据,也即有损压缩所获得的数据,而非原始的OAM数据。

[0241] 在图16中,3D音频编码器可在多个单独模式下操作。

[0242] 除了在图12的上下文中所描述的第一模式以及第二模式下外,在图16中的3D音频编码器可额外地在第三模式下操作,当预渲染/混合器200没有激活时,核心编码器在第三模式下从独立对象中产生至少一个传输声道。另外或额外地,当对应于图12中的混合器200的预渲染/混合器200未激活,SAOC编码器在第三模式下从原始信号中产生至少一个另外的或额外的传输声道。

[0243] 最后,当3D音频编码器使用于第四模式时,SAOC编码器800可对声道和预渲染/混合器所产生的预渲染对象进行编码。因此,在第四模式下,由于声道以及对象完整地传送到独立的SAOC传输声道内,最低的比特率应用将提供良好的质量,并在第四模式下,图3以及图5中作为“SAOC-SI”所指示的相关联辅助信息,和另外,任何的压缩元数据不会被传送。

[0244] 根据上述实施例中的其中一个,在图16中的OAM编码器400为装置250的元数据编码器210,用于产生编码音频信息。此外,根据上述实施例中的其中一个,在图16中的SAOC编码器800以及USAC编码器300一起形成装置250的音频编码器220,用于产生编码音频信息。

[0245] 根据另一实施例,提供一种对音频输入数据101进行编码以获得音频输出数据501的装置。对音频输入数据101进行编码的装置包含:

[0246] -输入界面1100,用于接收多个音频声道、多个音频对象以及关于多个音频对象的至少一个的元数据;

[0247] -混合器200,用于混合多个对象以及多个声道以获得多个预混合声道,多个预混合声道中的每一个包含声道的音频数据以及至少一个对象的音频数据;和

[0248] -装置250,用于产生包含元数据编码器以及音频编码器的编码音频信息,如上所述。

[0249] 用于产生编码音频信息的装置250的音频编码器220为对核心编码器输入数据进行核心编码的核心编码器300。

[0250] 用于产生编码音频信息的装置250的元数据编码器210为对关于多个音频对象中的至少一个的元数据进行压缩的元数据压缩器400。

[0251] 图13示出根据本发明的实施例的3D音频解码器。3D音频解码器接收编码音频数据作为输入,也即图12的数据501。

[0252] 3D音频解码器包含元数据解压缩器1400、核心解码器1300、对象处理器1200、模式控制器1600以及后置处理器1700。

[0253] 具体地,3D音频解码器用于解码编码音频数据,输入界面用于接收编码音频数据,编码音频数据包含多个编码声道以及多个编码对象以及在特定的模式下与多个对象相关联的压缩元数据。

[0254] 此外,核心解码器1300用于解码多个编码声道以及多个编码对象,额外地,元数据解压缩器用于解压缩压缩元数据。

[0255] 此外,对象处理器1200用于使用解压缩元数据处理核心解码器1300所产生的多个解码对象,以获得包含对象数据以及解码声道的预定数量的输出声道。该输出声道在1205

处被指示并接着被输入到后置处理器1700内。后置处理器1700用于将多个输出声道1205转换成特定输出格式,该特定输出格式可以为二进制输出格式或扬声器输出格式,例如5.1以及7.1等输出格式。

[0256] 优选地,3D音频解码器包含模式控制器1600,该模式控制器1600用于分析编码数据以检测模式指示。因此,模式控制器1600连接到图13内的输入界面1100。然而,模式控制器在此并非为必要的。代替地,可调式音频解码器可通过任何其他种类的控制数据进行预设置,例如用户输入或任何其他控制。优选地,在图13中的3D音频解码器通过模式控制器1600进行控制,并用于绕过任何对象处理器并将多个解码声道馈入后置处理器1700。当第二模式应用于图12的3D音频编码器时,3D音频编码器在第二模式下操作,则仅有预渲染声道被接收。另外,当第一模式应用于3D音频编码器时,也即当3D音频编码器已执行单独的声道/对象编码时,对象处理器1200不会被绕过,而多个解码声道以及多个解码对象与元数据解压缩器1400产生的解压缩元数据一同被馈入到对象处理器1200。

[0257] 优选地,应用第一模式或第二模式的指示被包含于解码音频数据,然后模式控制器1600分析解码数据以检测模式指示。当模式指示表示编码音频数据包含编码声道以及编码对象时,使用第一模式;而当模式指示表示编码音频数据不包含任何音频对象(也即仅包含由图12中的3D音频编码器获得的预渲染声道)时,使用第二模式。

[0258] 在图13中,根据上述实施例中的其中一个,元数据解压缩器1400为装置100的元数据解码器110,用于产生至少一个音频声道。此外,根据上述实施例中的其中一个,图13中的核心解码器1300、对象处理器1200以及后置处理器1700一起形成装置100的音频解码器120,用于产生多个音频声道。

[0259] 图15示出与图13相比的3D音频解码器的优选实施例,图15的实施例对应于图14的3D音频编码器。除了在图13中的3D音频解码器的实施方式之外,在图15中的3D音频解码器包含SAOC解码器1800。此外,图13的对象处理器1200被实施作为独立的对象渲染器1210以及混合器1220,而对象渲染器1210的功能也可通过SAOC解码器1800根据该模式来实施。

[0260] 此外,后置处理器1700可被实施作为立体渲染器1710或格式转换器1720。另外,也可实施图13的数据1205的直接输出,如1730所示出。因此,为了具有可变性,优选的是对较多数量(例如22.2或32)的声道执行解码器内的处理,如果需要较小的格式,再接着进行后处理。然而,当一开始就清楚知道仅需要小格式(例如5.1格式),优选地,如图13或图6的快捷1727所示出,可施加跨越SAOC解码器和/或USAC解码器的特别控制,以避免不必要的升混合操作以及随后的降混合操作。

[0261] 在本发明的优选实施例中,对象处理器1200包含SAOC解码器1800,该SAOC解码器1800用于解码核心解码器所输出的至少一个传输声道以及相关参数化数据,并使用解压缩元数据以获得多个渲染音频对象。为此,OAM输出被连接至方块1800。

[0262] 此外,对象处理器1200用于渲染核心解码器所输出的解码对象,其并未被编码于SAOC传输声道,而是独立编码于对象渲染器1210所指示的典型单一声道单元。此外,解码器包含相对应于输出1730的输出界面,用于将混合器的输出输出到扬声器。

[0263] 在另一实施例中,对象处理器1200包含空间音频对象编码解码器1800,用于解码至少一个传输声道以及相关参数化辅助信息,其代表编码音频信号或编码音频声道,其中空间音频对象编码解码器用于将相关参数化信息以及解压缩元数据转码到可用

于直接地渲染输出格式的经转码的参数化辅助信息,例如在SAOC的早期版本所定义的示例。后置处理器1700用于使用解码传输声道以及经转码的参数化辅助信息,计算输出格式的音频声道。后置处理器所执行的处理可相似于MPEG环绕处理或可以为任何其他处理,例如BCC处理等。

[0264] 在另一实施例中,对象处理器1200包含空间音频对象编码解码器1800,用于使用解码(通过核心解码器)传输声道以及参数化辅助信息,针对输出格式直接升混合以及渲染声道信号。

[0265] 此外,重要的是,图13的对象处理器1200另外包含混合器1220,当存在与声道混合的预渲染对象时(也即当图12的混合器200激活时),混合器1220直接地接收USAC解码器1300所输出的数据并作为输入。此外,混合器1220从执行对象渲染的对象渲染器接收没有经SAOC解码的数据。此外,混合器接收SAOC解码器输出数据,也即SAOC渲染的对象。

[0266] 混合器1220连接到输出界面1730、立体渲染器1710以及格式转换器1720。立体渲染器1710用于使用头部相关传递函数或立体空间脉冲响应(BRIR),将输出声道渲染成两个立体声道。格式转换器1720用于将输出声道转换成输出格式,该输出格式具有数量少于混合器的输出声道1205的声道,格式转换器1720需要再现布局的信息,例如5.1扬声器等。

[0267] 根据上述实施例中的其中一个,在图15中的OAM解码器1400为装置100的元数据解码器110,用于产生至少一个音频声道。此外,根据上述实施例中的其中一个,在图15中的对象渲染器1210、USAC解码器1300以及混合器1220一起形成装置100的音频解码器120,用于产生至少一个音频声道。

[0268] 图17中的3D音频解码器不同于图15中的3D音频解码器,不同之处在于SAOC解码器不仅能产生渲染对象,也能产生渲染声道,在此情况下,图16中的3D音频解码器已被使用,且在声道/预渲染对象以及SAOC编码器800输入界面之间的连接900为激活的。

[0269] 此外,矢量基幅值相移(VBAP)阶段1810用于从SAOC解码器接收再现布局的信息,并将渲染矩阵输出到SAOC解码器,以使SAOC解码器最后能以1205的高声道格式(也即32声道扬声器)来提供渲染声道,而不需混合器的任何额外的操作。

[0270] 优选地,VBAP方块接收解码OAM数据以得到渲染矩阵。更普遍地,优选的是需要再现布局以及输入信号应被渲染到再现布局的位置的几何信息。几何输入数据可以为对象的OAM数据或已使用SAOC传送的声道的声道位置信息。

[0271] 然而,如果仅需要特定的输出界面,则VBAP状态1810已经针对例如5.1输出而提供所需要的渲染矩阵。SAOC解码器1800执行来自SAOC传输声道、相关联的参数数据以及解压缩元数据的直接渲染,而不需混合器1220的交互下直接渲染成所需要的输出格式。然而,当模式之间采用特定的混合时,即几个声道SAOC编码但非所有声道皆为SAOC编码;或者几个对象SAOC编码但非所有对象皆SAOC编码;或者仅特定数量的预渲染对象和声道SAOC解码而剩余声道不以SAOC处理,然后混合器将来自单独输入部分,即直接来自核心解码器1300、对象渲染器1210以及SAOC解码器1800的数据放在一起。

[0272] 在图17中,根据一个上述实施例的用于产生至少一个音频声道的装置100的元数据解码器110为OAM解码器1400。而且,在图17中,根据一个上述实施例的用于产生至少一个音频声道的装置100的音频解码器120由对象渲染器1210、USAC解码器1300以及混合器1220一起形成。

[0273] 本发明提供一种对编码音频数据进行解码的装置。对编码音频数据进行解码的装置包含：

[0274] -输入界面1100,用于接收编码音频数据,此编码音频数据包含多个编码声道、或者多个编码对象、或者关于多个对象的压缩元数据;以及

[0275] -装置100,包含元数据解码器110以及音频声道发生器120,用于产生至少一个如上所述的音频声道。

[0276] 用于产生至少一个音频声道的装置100的元数据解码器110为对压缩元数据进行解压缩的元数据解压缩器400。

[0277] 用于产生至少一个音频声道的装置100的音频声道发生器120包含用于解码多个编码声道以及多个编码对象的核心解码器1300。

[0278] 而且,音频声道发生器120进一步包含对象处理器1200,其使用解压缩元数据处理多个解码对象,以从对象以及解码声道获得包含音频数据的多个输出声道1205。

[0279] 此外,音频声道发生器120进一步包含后置处理器1700,其将多个输出声道1205转换成输出格式。

[0280] 虽然一些方面已经在装置的内容中描述,清楚的是这些方面也代表相对应的方法的描述,而方块或者装置对应方法步骤或者方法步骤的特征。同样地,在方法步骤的内容中描述的方面也代表相对应的方块或者项目或者相对应装置的特征的描述。

[0281] 本发明的解压缩信号可储存在数字存储介质上或者可传送至传送介质(例如无线传送介质或者有线传送介质(例如因特网))上。

[0282] 取决于特定的执行需求,本发明的实施例可在硬件或者在软件上实现。此实现可使用数字储存介质,例如软盘、DVD、CD、ROM、PROM、EPROM、EEPROM或者FLASH内存实施,其储存有电子可读控制信号,其能与可编程计算机系统合作(或者能够合作)以执行上述方法。

[0283] 根据本发明的一些实施例包含具有电子可读控制信号的非临时性数据载体,其能够与可编程计算机系统配合,以执行上述方法中的其中一种。

[0284] 通常,本发明的实施例可实现为具有程序代码的计算机程序产品,当此计算机程序产品在计算机上运行时此程序代码可操作以执行上述方法中的其中一种。例如此程序代码可储存在机器可读载体上。

[0285] 其他实施例包含用于执行上述方法中的其中一种的计算机程序,其储存在机器可读载体上。

[0286] 换句话说,因此本发明的方法的实施例为具有当此计算机程序在计算机上运行时,能执行上述方法中的其中一种的程序代码的计算机程序。

[0287] 因此,本发明的方法的另一实施例为数据载体(或者数字存储介质或者计算机可读介质),包含纪录于其上的用于执行上述方法中的其中一种的计算机程序。

[0288] 因此,本发明的方法的另一实施例为数据流或者信号序列,其代表用于执行上述方法中的其中一种的计算机程序。例如数据流或者信号序列可配置为经由数据通讯连接传输,例如经由因特网。

[0289] 另一实施例包含处理装置,例如计算机,或者可编程逻辑设备,用于或者适于执行上述方法中的其中一种。

[0290] 另一实施例包含安装有用于执行上述方法中的其中一种的计算机程序的计算机。

[0291] 在一些实施例中,可编程逻辑设备(例如现场可编程门阵列)可用于执行上述方法的一些或者全部功能。在一些实施例中,为了执行上述方法中的其中一种,现场可编程门阵列可配合微处理器。通常,此方法可优选通过任何硬件装置执行。

[0292] 上述实施例仅为本发明原理的说明。应理解的是,本文中所描述的修改和有关布置的变化和细节对本领域的其他技术人员来说是明显的。因此,其意图是由即将发生的专利权利要求范围来限制,而不是由本文描述的实施例和解释的方式呈现的特定细节来限制。

[0293] 参考文献:

[0294] [1]Peters,N.,Lossius,T.and Schacher J.C.,“SpatDIF:Principles, Specification,and Examples”,9th Sound and Music Computing Conference,Copenhagen,Denmark,2012年7月.

[0295] [2]Wright,M.,Freed,A.,“Open Sound Control:A New Protocol for Communicating with Sound Synthesizers”,International Computer Music Conference,Thessaloniki,Greece,1997.

[0296] [3]Matthias Geier,Jens Ahrens,and Sascha Spors.(2010),“Object-based audio reproduction and the audio scene description format”,Org.Sound,第15卷,第3期,第219-227页,2010年12月.

[0297] [4]W3C,“Synchronized Multimedia Integration Language (SMIL 3.0)”,2008年12月.

[0298] [5]W3C,“Extensible Markup Language (XML) 1.0 (Fifth Edition)”,2008月11月.

[0299] [6]MPEG,“ISO/IEC International Standard 14496-3-Coding of audio-visual objects,Part 3 Audio”,2009.

[0300] [7]Schmidt,J.;Schroeder,E.F.(2004),“New and Advanced Features for Audio Presentation in the MPEG-4Standard”,116th AES Convention,Berlin,Germany,2004年5月

[0301] [8]Web3D,“International Standard ISO/IEC 14772-1:1997-The Virtual Reality Modeling Language (VRML),Part 1:Functional specification and UTF-8encoding”,1997.

[0302] [9]Sporer,T.(2012),“Codierung räumlicher Audiosignale mit leichtgewichtigen Audio-Objekten”,Proc.Annual Meeting of the German Audiological Society (DGA),Erlangen,Germany,2012年3月.

[0303] [10]Ramer,U.(1972),“An iterative procedure for the polygonal approximation of plane curves”,Computer Graphics and Image Processing,1(3),244-256.

[0304] [11]Douglas,D.;Peucker,T.(1973),“Algorithms for the reduction of the number of points required to represent a digitized line or its caricature”,The Canadian Cartographer 10(2),112-122.

[0305] [12]Ville Pulkki,“Virtual Sound Source Positioning Using Vector Base

---

Amplitude Panning”;J.Audio Eng.Soc.,第45卷,第6期,第456-466页,1997年6月.

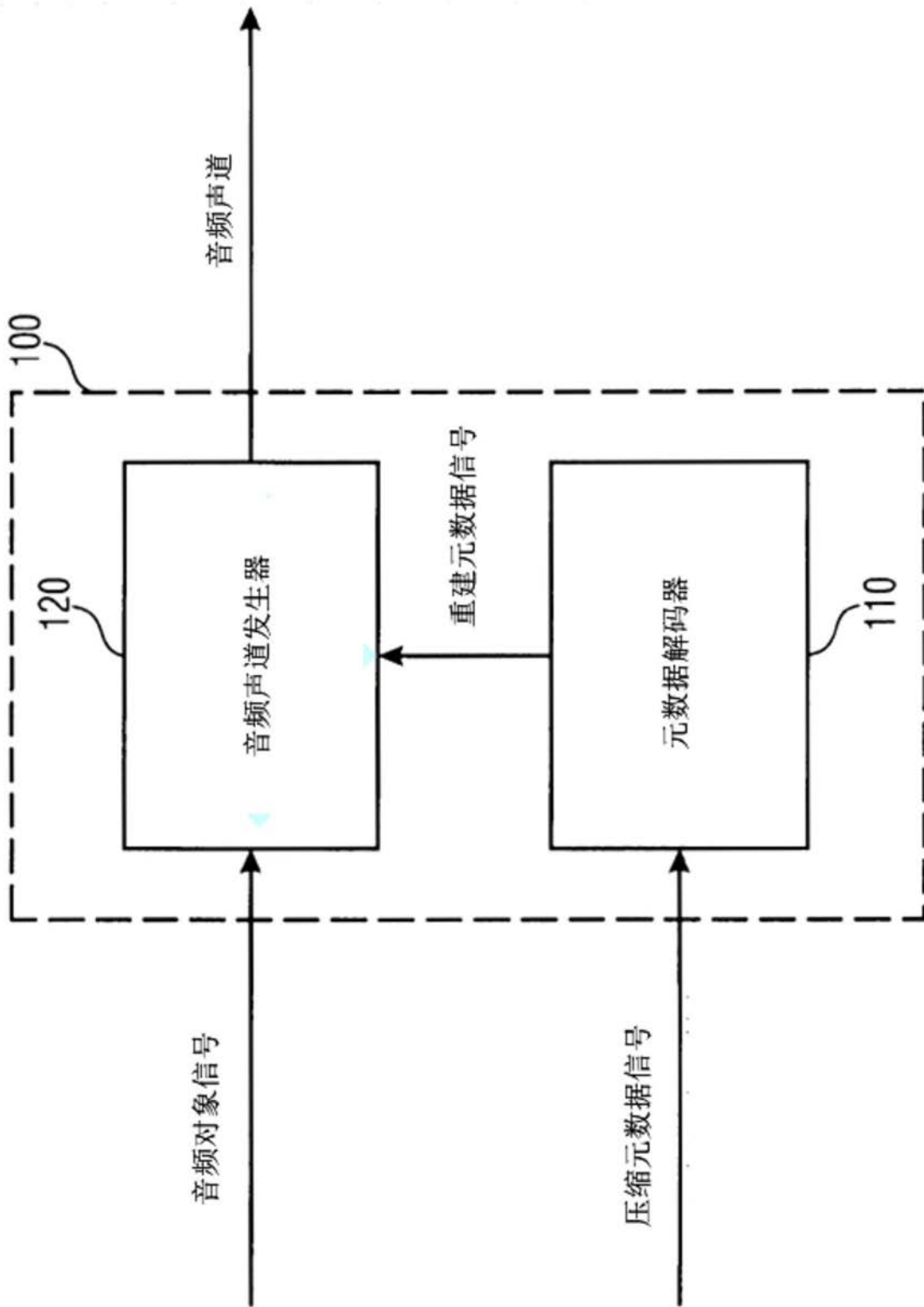


图1

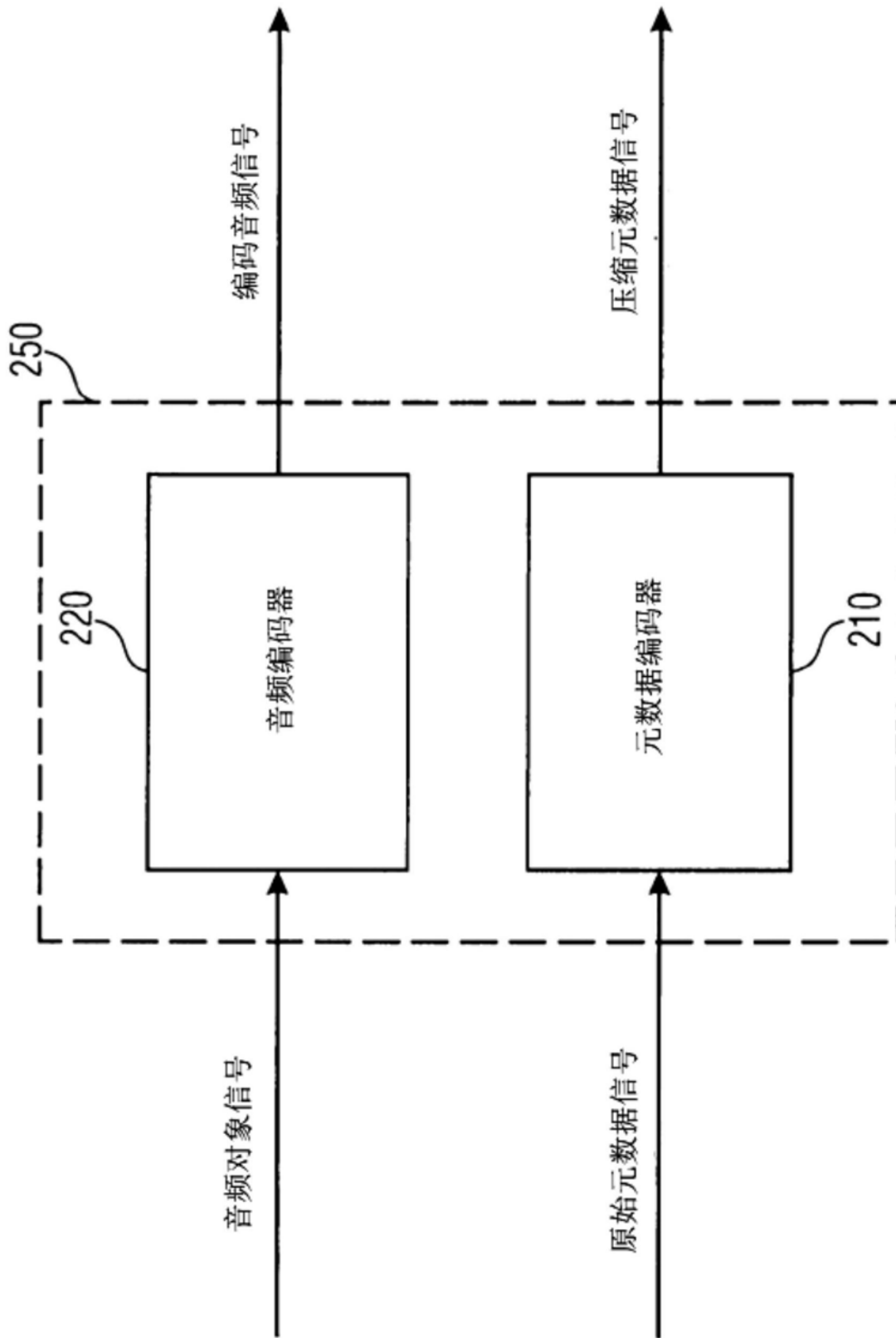


图2

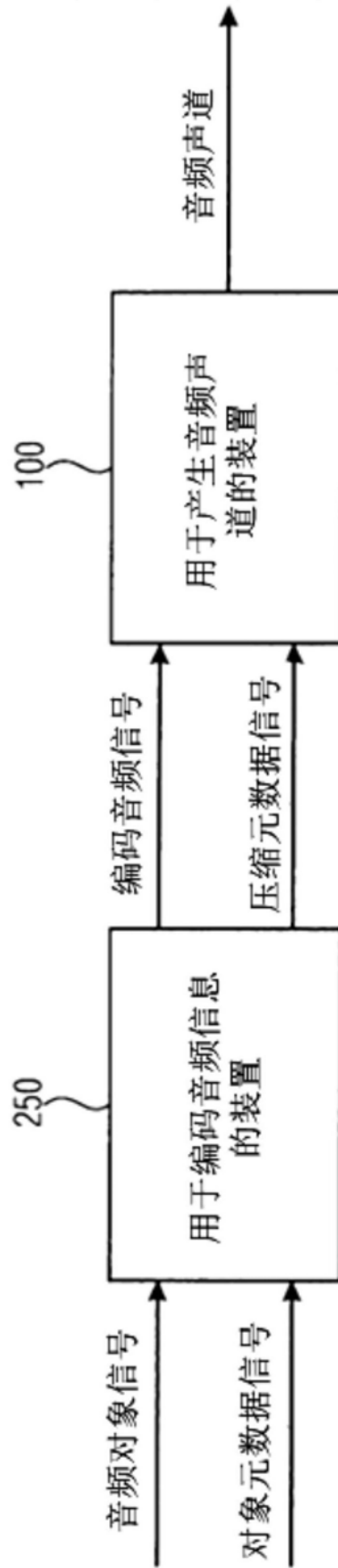


图3

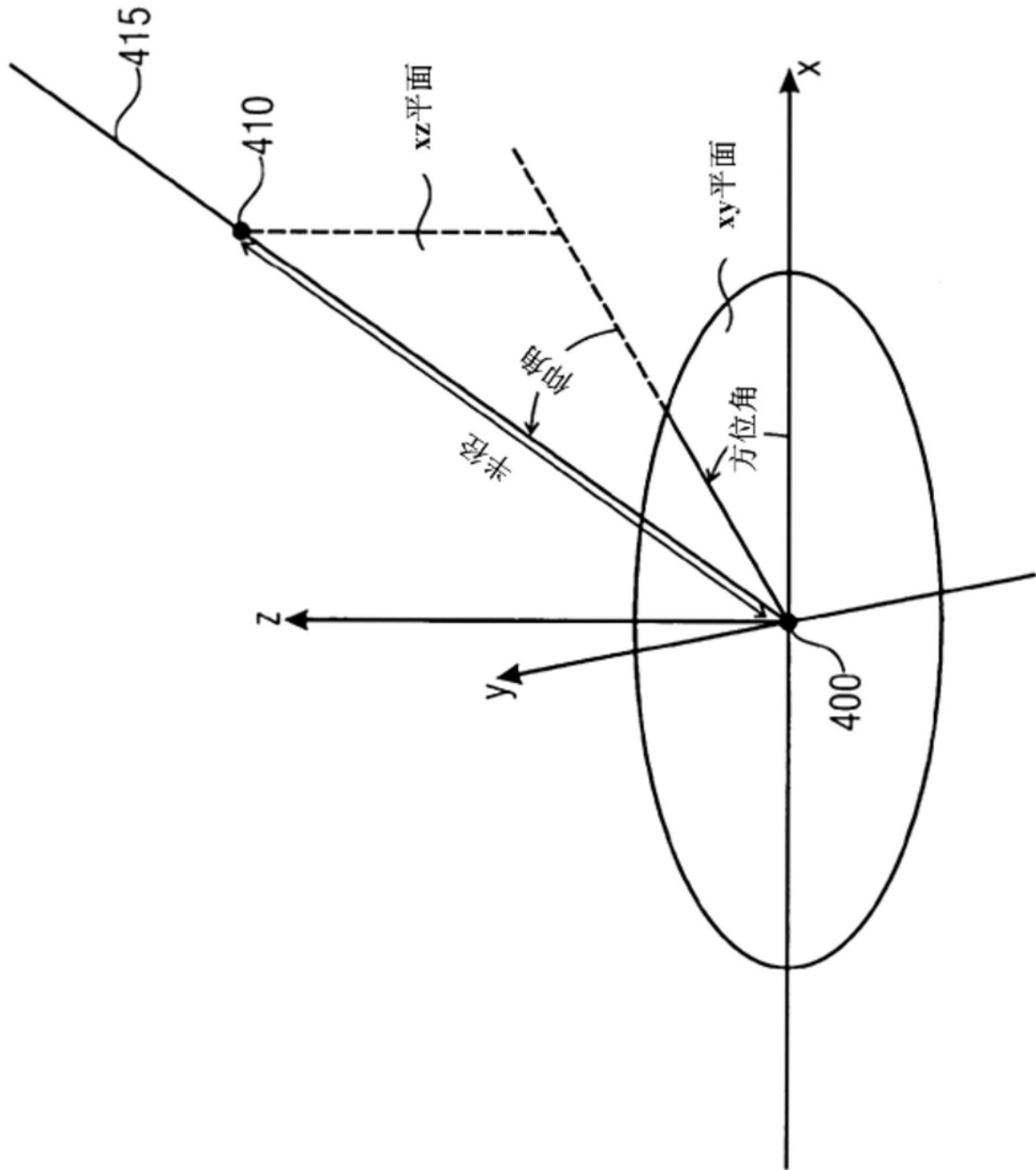


图4

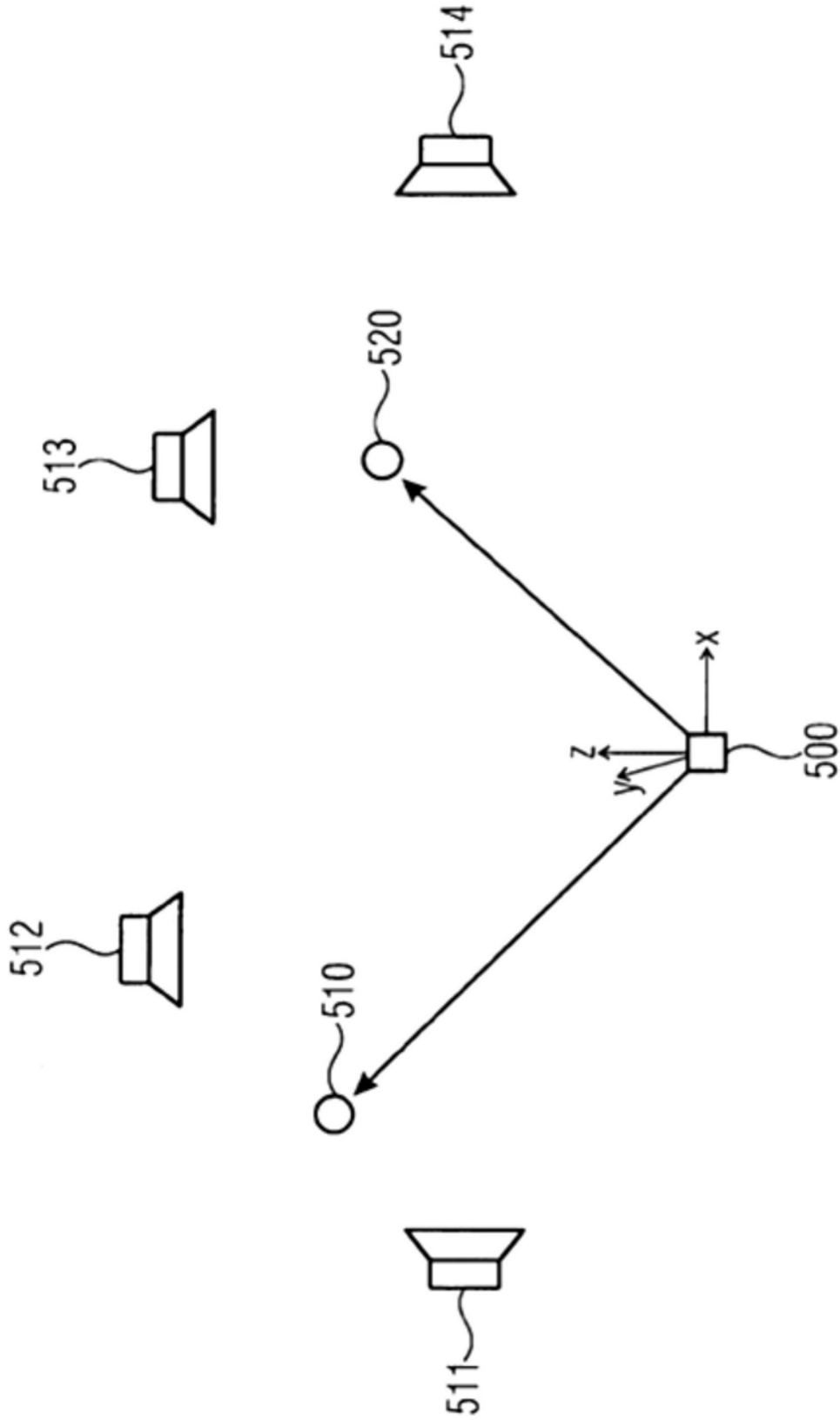


图5

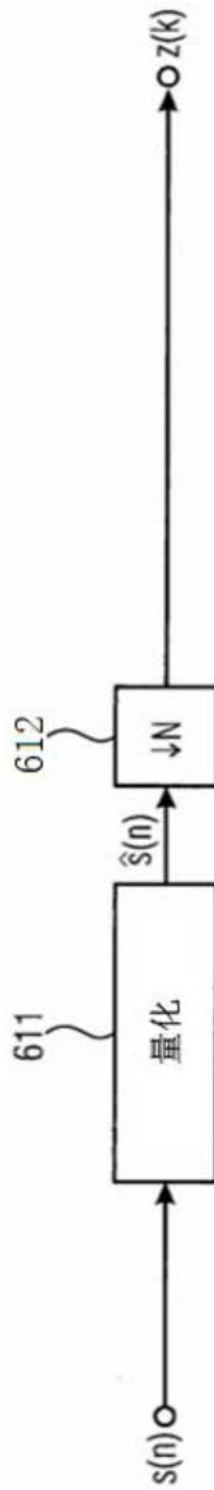


图6



图7

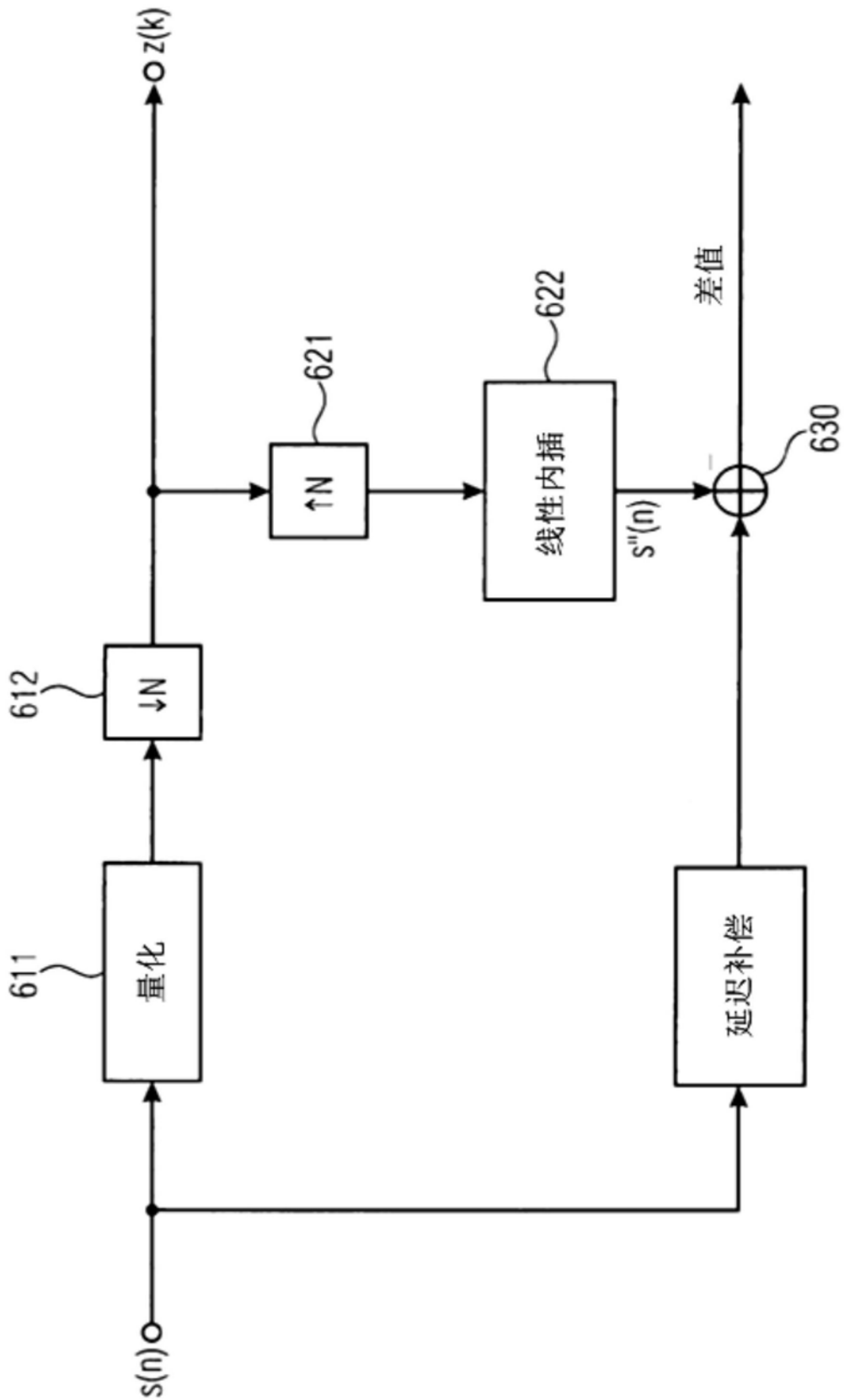


图8

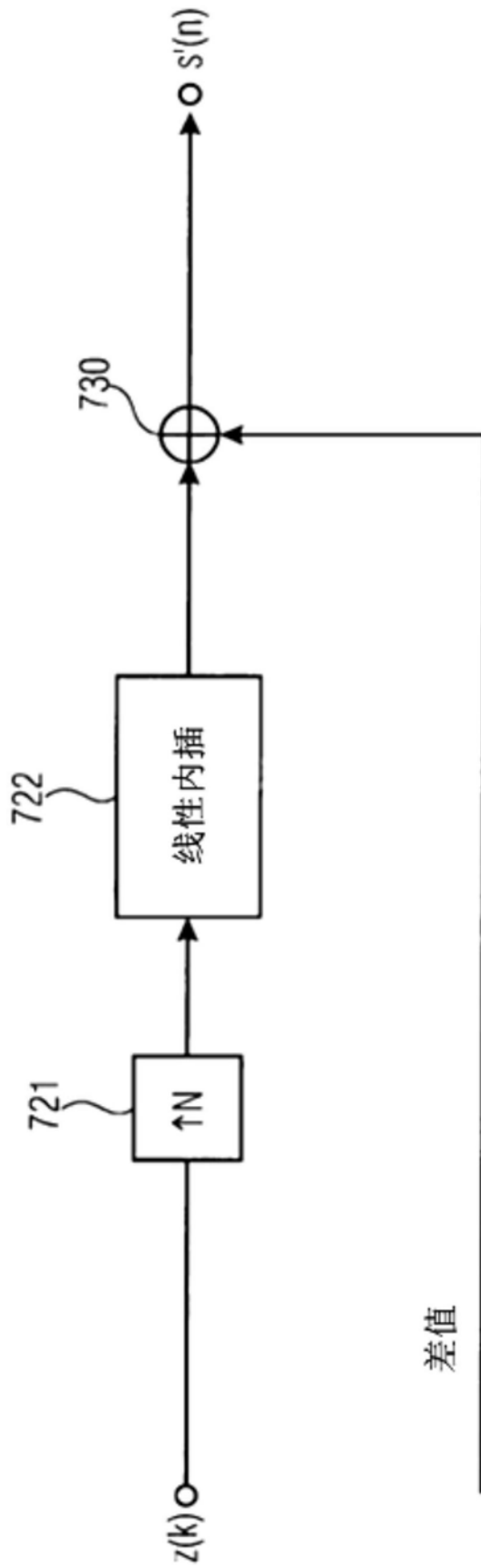


图9

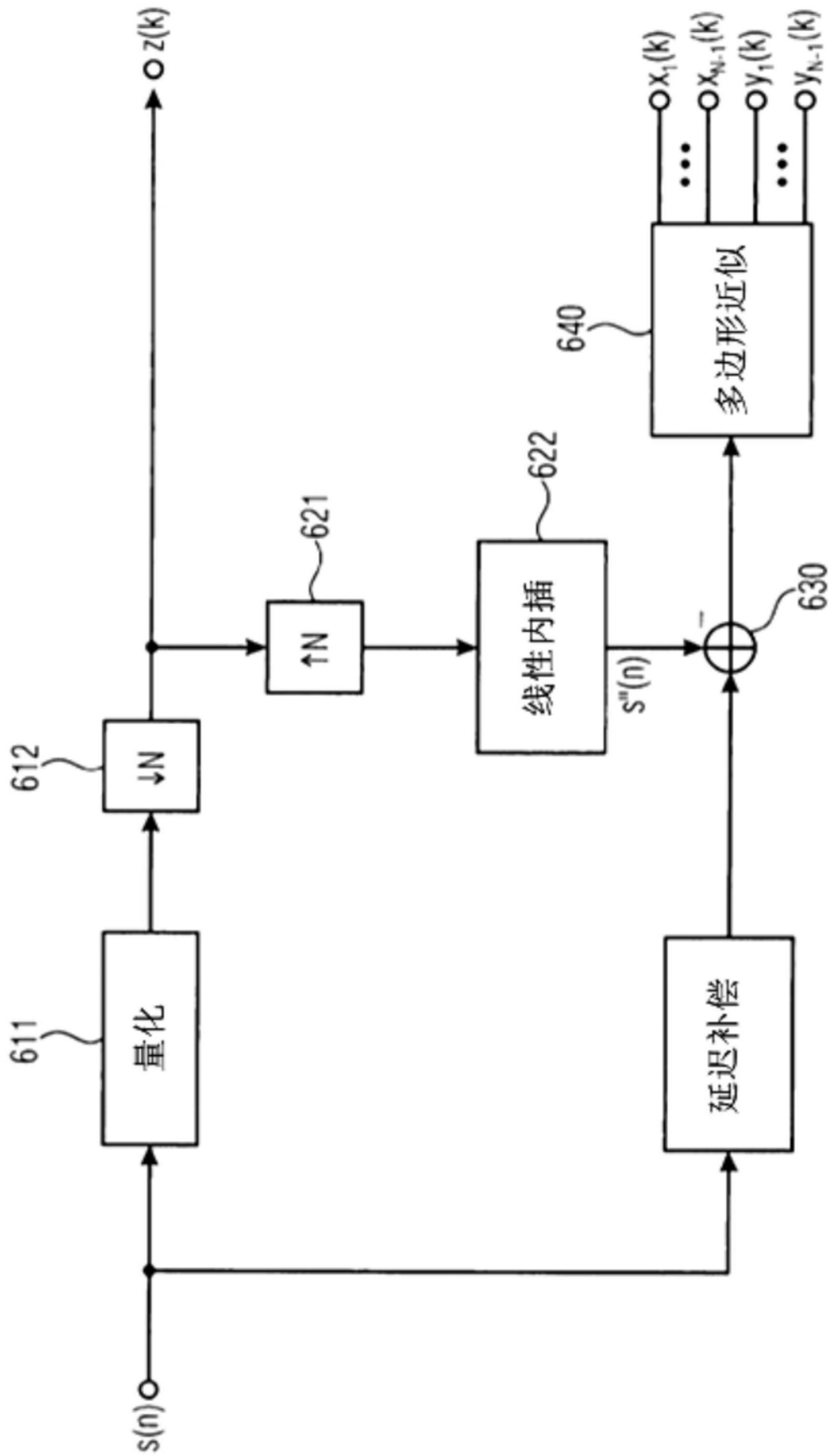


图10

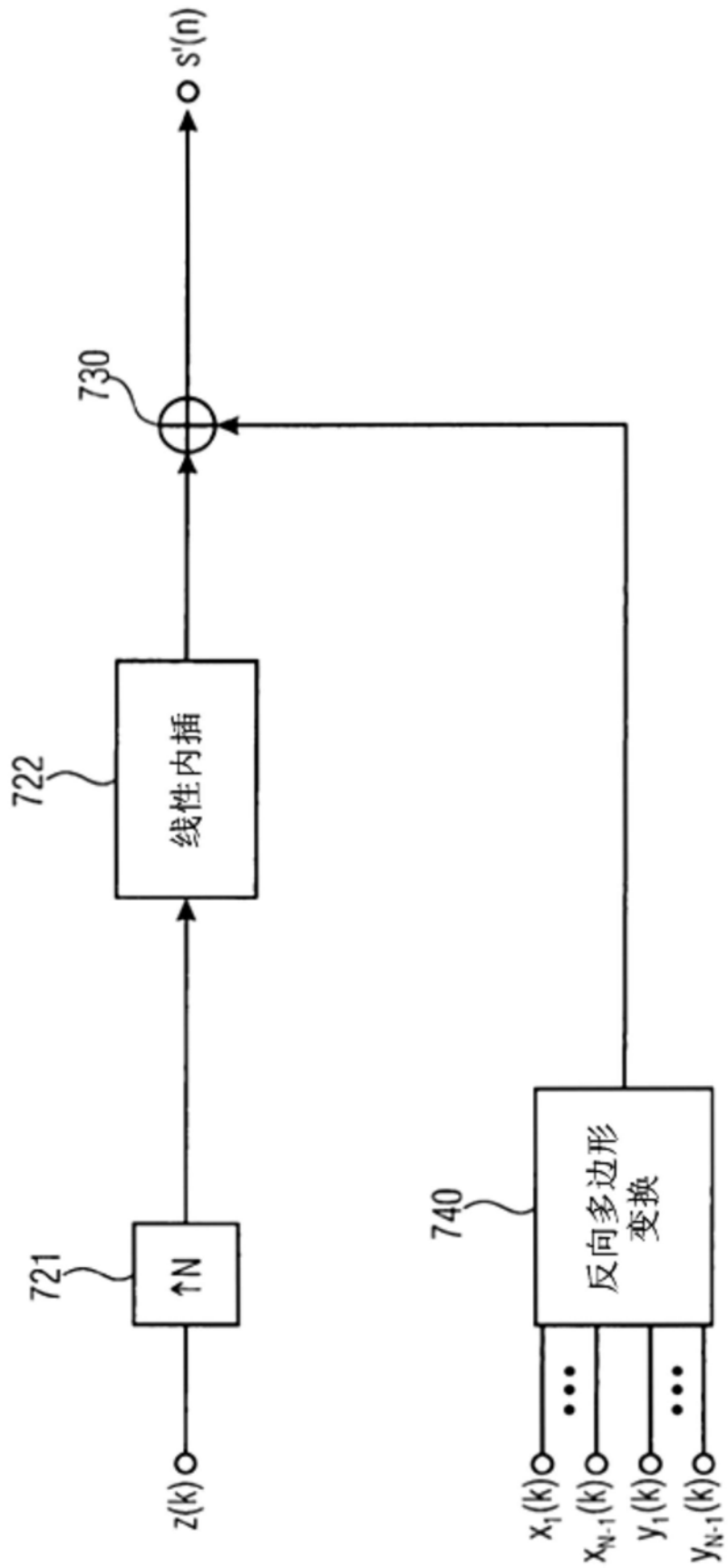
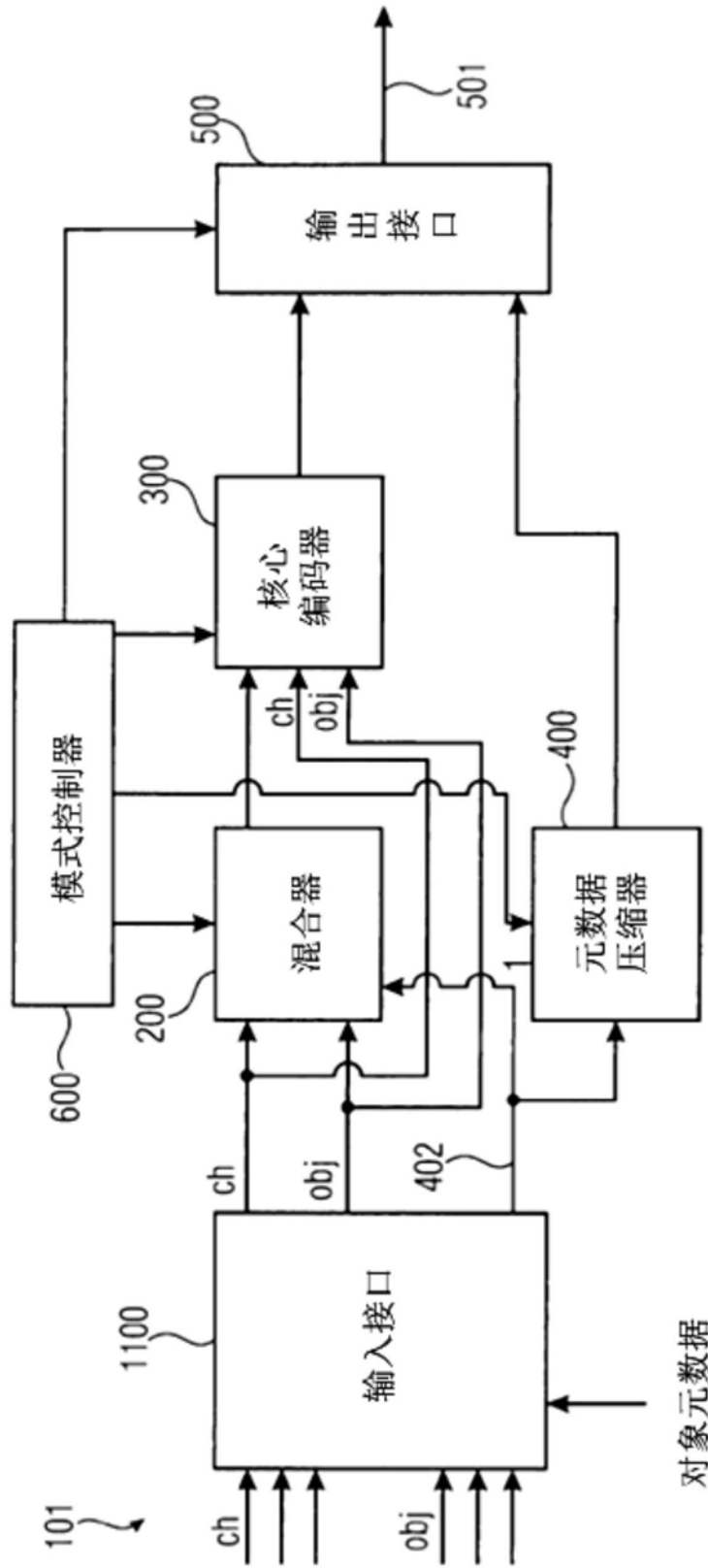


图11



第一模式: 单独声道/对象编码  
第二模式: 声道和渲染对象的混合

图12(编码器)

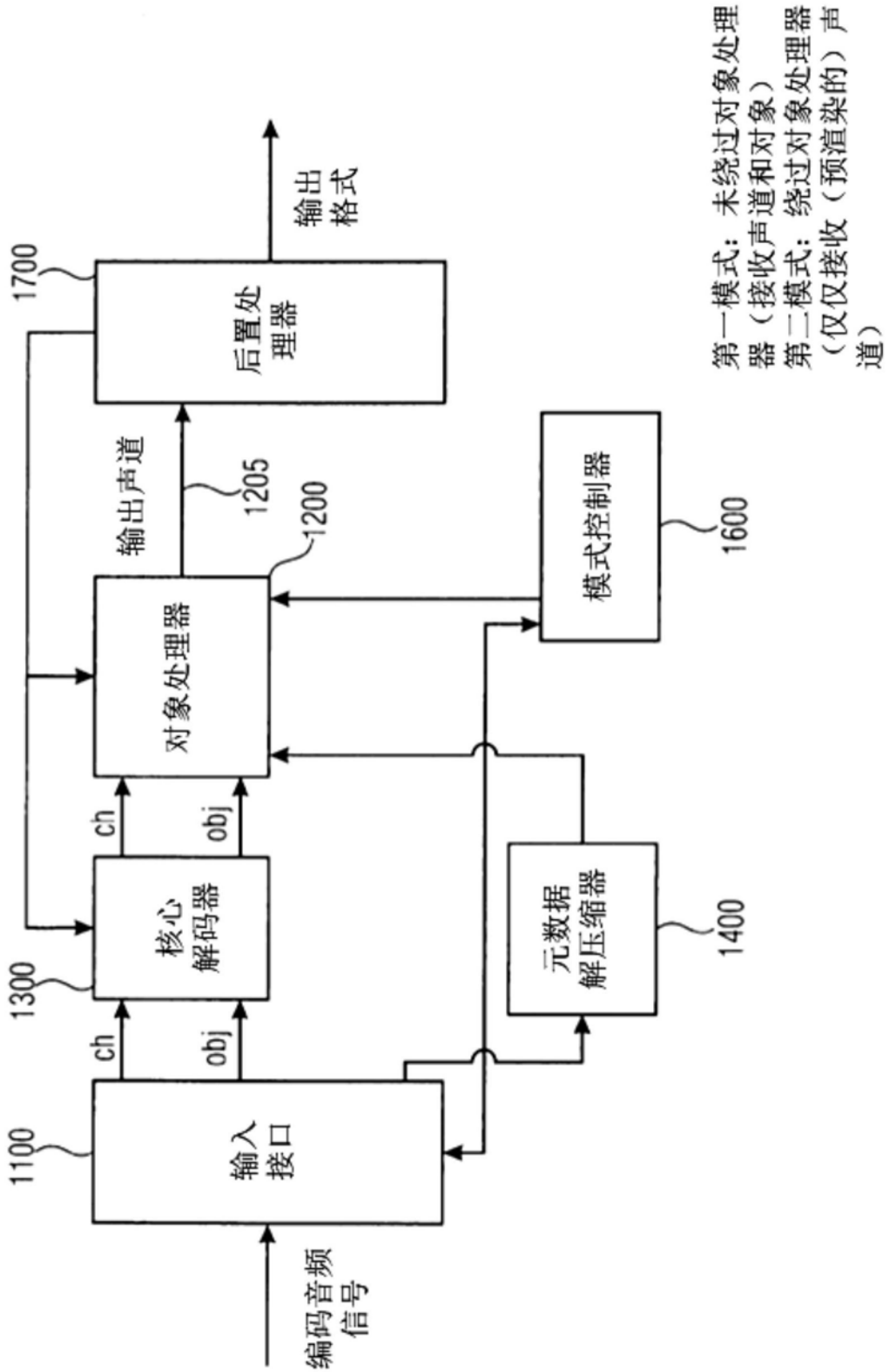


图13 (解码器)

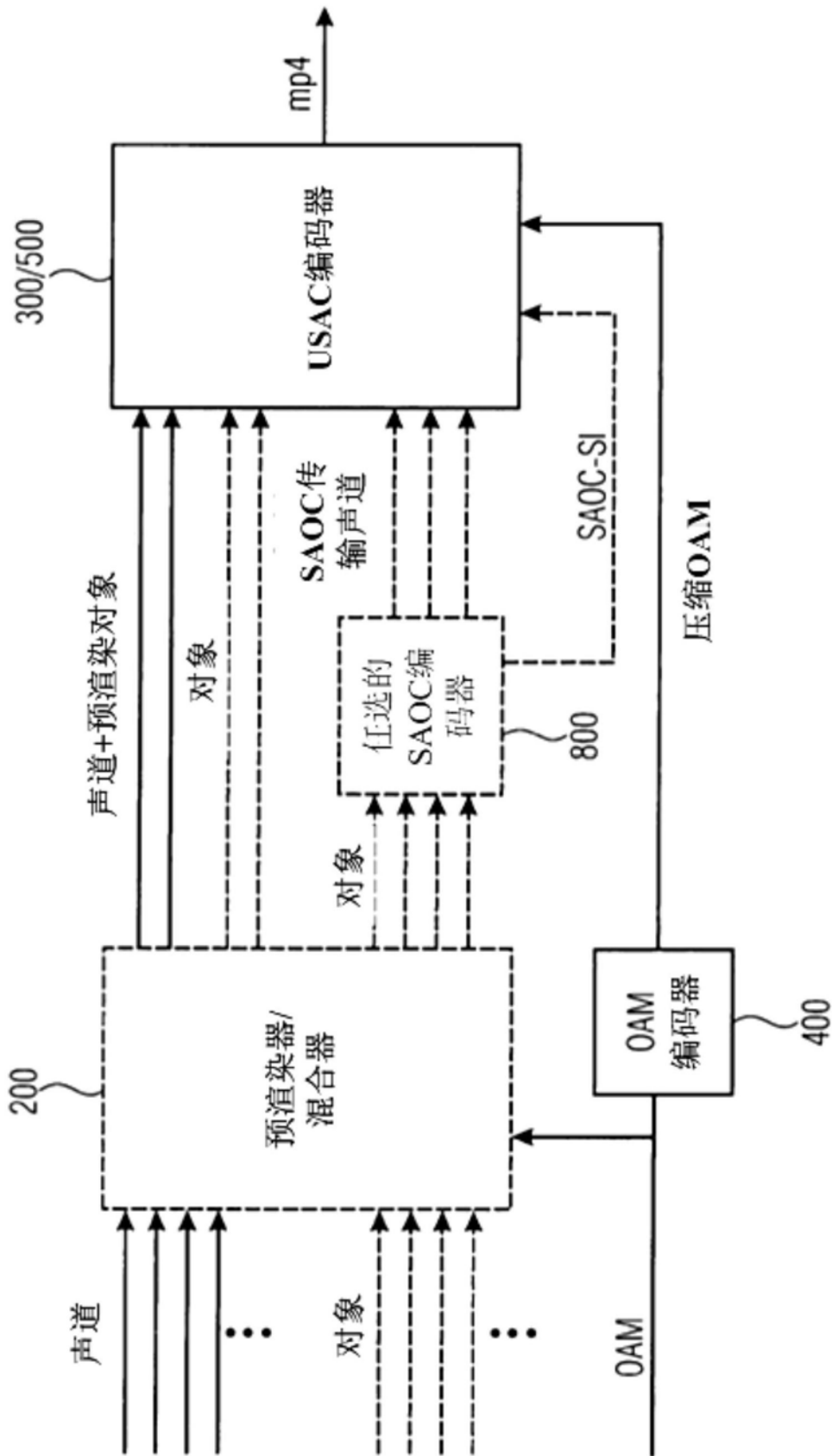


图14 (编码器)

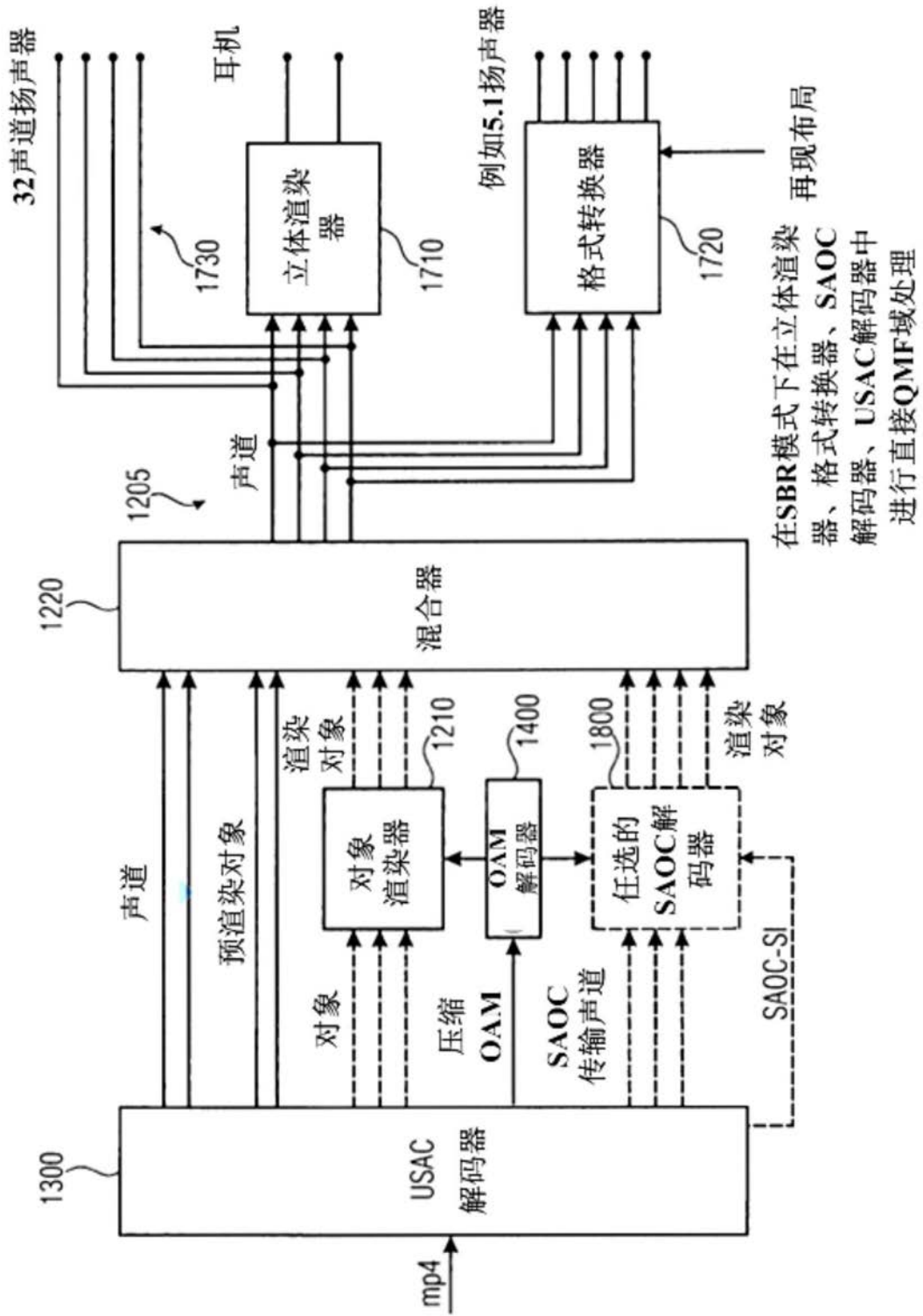


图15 (解码器)

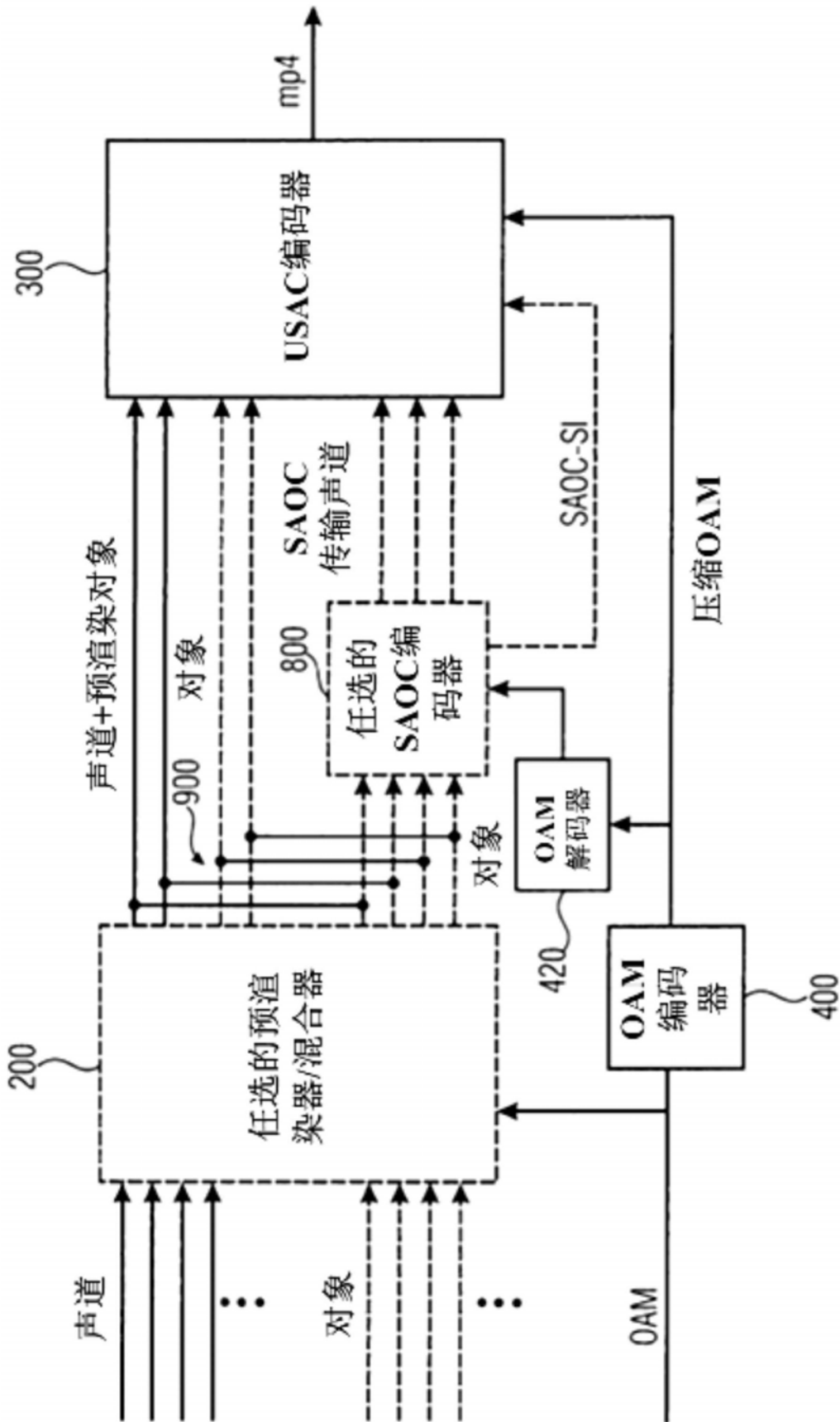


图16(编码器)

