



(12) 发明专利

(10) 授权公告号 CN 101868970 B

(45) 授权公告日 2015. 01. 21

(21) 申请号 200880116700. X

(22) 申请日 2008. 10. 20

(30) 优先权数据

10-2007-0105831 2007. 10. 19 KR

10-2007-0131535 2007. 12. 14 KR

(85) PCT国际申请进入国家阶段日

2010. 05. 19

(86) PCT国际申请的申请数据

PCT/KR2008/006193 2008. 10. 20

(87) PCT国际申请的公布数据

W02009/051457 EN 2009. 04. 23

(73) 专利权人 三星电子株式会社

地址 韩国京畿道

专利权人 庆熙大学校产学协力团

(72) 发明人 郑度泳 朴泰诚 吴润济 宋在涓

黄栖煥 徐德荣 朴光勋 金奎宪

李润辰 李长远

(74) 专利代理机构 北京铭硕知识产权代理有限公司

公司 11286

代理人 韩明星 王青芝

(51) Int. Cl.

H04N 5/92 (2006. 01)

(56) 对比文件

US 20030030675 A1, 2003. 02. 13,

US 20030048354 A1, 2003. 03. 13,

US 20030095177 A1, 2003. 05. 22,

CN 1860503 A, 2006. 11. 08,

Jon Siragusa, David C Swift. general purpose stereoscopic data descriptor.

《VREX, INC. 》. 1997, 1-6.

Man Bae Kim. the adaption of 3d stereoscopic video in mpeg-21 dia. 《Signal Processing: Image Communication》. 2003, (第 18 期), 685-697.

审查员 吴永兴

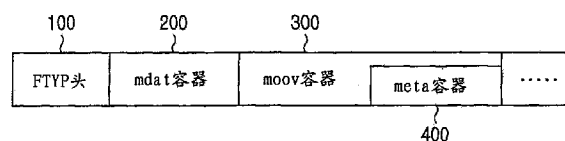
权利要求书2页 说明书17页 附图8页

(54) 发明名称

记录三维图像数据的方法

(57) 摘要

一种将 3D 图像数据存储于记录介质中的方法, 包括: 在预定的文件类型字段加载图像类型信息, 所述图像类型信息表明是单一的立体图像还是包括单视点图像和立体图像的复合图像; 在预定的图像构造信息容器字段加载表明包括在三维图像中的多个媒体对象之间的时间和空间的关系的场景描述符信息、表明所述多个媒体对象的关系的各媒体对象的属性以及多个媒体对象的编码流之间的构造关系的对象构造信息、编码的流的构造信息; 在图像数据容器字段加载将要存储的图像的媒体数据; 在 meta 容器字段加载包括用于播放 3D 图像数据的信息的元数据。



1. 一种将三维 (3D) 图像数据存储在记录介质中的方法,所述方法包括:

在图像类型字段加载立体内容指示符或立体-单视点混合内容指示符,其中,所述立体内容指示符表示 3D 图像包括立体内容,所述立体-单视点混合内容指示符表示 3D 图像包括立体-单视点混合内容,其中,所述立体内容包括立体图像,所述立体-单视点混合内容包括单视点图像和立体图像;

在图像构造信息容器字段加载:场景描述符信息,表明包括在三维图像中的多个媒体对象之间的时间和空间关系;对象构造信息,表明所述多个媒体对象的各媒体对象的属性以及多个媒体对象的编码流之间的构造关系;编码的流的构造信息;

在图像数据容器字段加载将要存储的图像的媒体数据;

在 meta 容器字段加载包括用于播放 3D 图像数据的信息的元数据,其中,所述元数据包括:关于立体内容和立体-单视点混合内容的构造信息、关于包括在立体内容和立体-单视点混合内容中的图像的大小的构造信息、拍摄信息和显示信息,

其中,所述元数据包括:

相机信息,所述相机信息包括:多个相机之间的距离、关于目标的多个相机镜头的焦距、指示相机是否交叉的信息以及关于对象的旋转角度;

显示信息,所述显示信息包括:通过多个相机镜头拍摄的图像之间的最大视差和最小视差、以及立体图像的观看距离。

2. 如权利要求 1 所述的方法,其中,meta 容器字段被包括在预定的图像构造信息容器字段。

3. 如权利要求 1 所述的方法,其中,meta 容器字段作为单独的字段被形成。

4. 如权利要求 1 所述的方法,其中,所述元数据被记录在 meta 容器字段的 snmi 包中。

5. 如权利要求 4 所述的方法,其中,元数据包括:关于包括在立体-单视点混合内容中的单视点图像和立体图像的位置的信息,关于单视点图像和立体图像的数据的大小的信息以及关于单视点图像和立体图像的图像类型标识的信息,

位置信息和数据大小被记录在 meta 容器字段的 iloc 包中,

图像类型标识被记录在 meta 容器字段的 iinf 包中。

6. 如权利要求 4 所述的方法,其中,当各个媒体数据的帧率互不不同时,snmi 包包括:表明参考媒体数据的信息和表明根据组合图像的类型首先被压缩的媒体数据的信息。

7. 如权利要求 2 所述的方法,其中,立体图像包括至少一个片段。

8. 如权利要求 7 所述的方法,其中,由于立体图像包括多个片段,将包含相同图像信息的片段分组为片段组,并根据 meta 容器字段中的片段组来记录所述片段。

9. 如权利要求 8 所述的方法,其中,计算包含不同图像信息的片段组的数量,并基于计算的片段组的数量,将通过各个片段表明的图像信息记录在 meta 容器字段中。

10. 如权利要求 3 所述的方法,其中,所述元数据被记录在 meta 容器字段的 snmi 包中。

11. 如权利要求 10 所述的方法,其中,元数据包括:包括在立体-单视点混合内容中的单视点图像和立体图像的位置信息、单视点图像和立体图像的数据大小以及单视点图像和立体图像的图像类型标识,

位置信息和数据大小被记录在 meta 容器字段的 iloc 包中,

图像类型标识被记录在 meta 容器字段的 iinf 包中。

12. 如权利要求 10 所述的方法,其中,拍摄信息包括:多个相机之间的距离、关于目标的多个相机镜头的焦距、关于对象的旋转角度、最大垂直视差以及通过多个相机镜头拍摄的图像之间的最大视差和最小视差。

13. 如权利要求 10 所述的方法,其中,当各个媒体数据的帧率互不相同,snmi 包包括表明参考媒体数据的信息和表明根据组合图像的类型首先被压缩的媒体数据的信息。

14. 如权利要求 3 所述的方法,其中,立体图像包括至少一个片段。

15. 如权利要求 14 所述的方法,其中,由于立体图像包括多个片段,将包含相同图像信息的片段分组为片段组,并根据 meta 容器字段中的片段组来记录所述片段。

16. 如权利要求 15 所述的方法,其中,计算包含不同图像信息的片段组的数量,并基于计算的片段组的数量,将通过各个片段表明的图像信息记录在 meta 容器字段中。

记录三维图像数据的方法

技术领域

[0001] 本发明涉及一种记录三维 (3D) 图像数据的方法,更具体地讲,涉及一种按照与 ISO 基本媒体文件格式兼容的格式生成 3D 图像数据,并存储生成的 3D 图像数据的方法。

背景技术

[0002] 典型的媒体文件格式包括:头部分,用于描述关于对应媒体的信息;图像数据部分,用于存储压缩的媒体数据。尽管可使用典型的媒体文件格式存储简单的图像数据,但是它并不适合作为一种适合多种媒体类型的全面的结构。

[0003] 国际标准化组织运动图像专家组 (MPEG) 已将 ISO 基本媒体文件格式定义为可通用于多种应用的基本文件格式。所述 ISO 基本媒体文件格式已被设计为将数据(例如,压缩的媒体流和与媒体流相关的构造信息)分级地存储到多个容器中。ISO 基本媒体文件格式未定义编码或解码方案,但是基本地定义了用于有效地存储编码或解码的媒体流的基本结构。

[0004] 根据组合左图像和右图像的方法,可按照多种方式构造立体图像。因此,为在显示设备上展现立体图像,需要将关于组合左图像和右图像的方法、左图像和右图像的划分的数量和大小、划分的图像的位置等的信息存储在显示设备的存储器中。此外,立体图像必需需要时间信息,基于该时间信息使左图像和右图像同步。

[0005] 因此,难以通过使用传统的典型媒体文件格式有效地构造用于存储立体图像的文件格式。由于 ISO 基本媒体文件格式是未考虑到立体图像的结构而定义的,并不适合立体图像。

[0006] 因此,需要一种考虑到立体图像的结构而定义的数据存储格式。同时,用于立体图像的这样的数据存储格式需要与国际标准文件格式(即,ISO 基本媒体文件格式)兼容,以便其能够应用于多种多媒体应用。

发明内容

[0007] 因此,本发明提供了一种数据存储格式,该数据存储格式包括关于立体图像的构造的信息并与国际标准文件格式(即,ISO 基本媒体文件格式)兼容,以及一种生成所述数据存储格式的方法。

[0008] 根据本发明,将三维 (3D) 图像数据存储于记录介质的方法包括:在预定的文件类型字段加载图像类型信息,所述信息表明是单一的立体图像还是包括单视点图像和立体图像的复合图像;在预定的图像构造信息容器字段加载表明包括在三维图像中的多个媒体对象之间的时间和空间的关系的场景描述符信息、表明所述多个媒体对象的各媒体对象的属性以及多个媒体对象的编码流之间的构造关系的对象构造信息、编码的流的构造信息;在图像数据容器字段加载将要存储的图像的媒体数据;并在 meta 容器字段加载包括用于播放 3D 图像数据的信息的元数据。

附图说明

[0009] 通过下面结合附图进行的详细描述,本发明的上述特点和优点将会变得更加清楚,其中:

[0010] 图 1 是应用本发明的立体成像设备的框图;

[0011] 图 2A 至 2C 是示出根据本发明示例性实施例的包含在三维图像中的立体图像的结构示例的示意图;

[0012] 图 3A 和 3B 是示出根据本发明示例性实施例的三维图像数据的存储格式的示意图;

[0013] 图 4 是示出根据本发明示例性实施例的包括在三维图像数据的存储格式中的 moov 容器的详细结构的示意图;

[0014] 图 5 是示出根据本发明示例性实施例的包括在三维图像数据的存储格式中的 meta 容器的详细结构的示意图;

[0015] 图 6 是根据本发明示例性实施例的根据三维图像数据的存储格式的示例的示意图;

[0016] 图 7 是根据本发明示例性实施例的根据三维图像数据的存储格式的另一示例的示意图;

[0017] 图 8 是示出根据本发明另一示例性实施例的包含在三维图像中的立体图像的结构示例的示意图;

[0018] 图 9 是根据本发明的另一示例性实施例的根据三维图像数据的存储格式的示例的示意图;

[0019] 图 10 是根据本发明的另一示例性实施例的根据三维图像数据的存储格式的另一示例的示意图;

具体实施方式

[0020] 以下,将参照附图描述本发明的示例性实施例。应该注意,尽管在不同的附图中示出相似的元件,但相似的元件用相似的标号指定。由于合并在这里的公知的功能和构造的详细描述会使本发明的主题不清晰,因此,为了清楚和简明的目的,将省略对所述公知的功能和构造的详细描述。此外,应该注意,仅描述了对于理解根据本发明的操作的所必要的部分,将省略除必要部分之外的其他部分的描述以使本发明的要点更清晰。

[0021] 在本发明的示例性实施例中,三维 (3D) 图像包括:包含左图像和右图像的组合作为立体图像以及包含立体 (stereoscopic) 图像和单视点 (monoscopic) 图像的组合作为复合图像。在此,单视点图像是指仅包含左图像和右图像之一的图像。左图像和右图像对应于媒体数据,并在本发明的实施例中可被称为媒体数据。此外,这样的媒体数据可包括视频数据和音频数据。此外,作为示例,所述视频数据和音频数据可包含至少一个轨道的媒体数据。

[0022] 图 1 示出应用本发明的立体成像设备。

[0023] 所述立体成像设备包括:图像输入单元 10、编码器 20、3D 图像生成器 30、数据存储控制器 40、存储器 50、显示器 60 和通信模块 70。

[0024] 图像输入单元 10 与用于接收生成立体图像所必需的多个输入图像的装置对应,并包括:多个镜头,用于用不同的角度投射从目标反射的特定波长的颜色信号;多个传感器模块(例如,包括传感器的模块(例如,CCD 和 CMOS)),用于将通过多个镜头输入的颜色

信号转换为电信号。图像输入单元 10 还包括：视频信号处理设备，用于将从多个传感器模块中的每一个输入的数据转换为包括时间 / 空间信息的图像信号，从而生成与传感器模块一样多的图像数据。例如，当立体图像包括左图像和右图像的媒体数据的组合时，图像输入单元 10 包括用于拍摄目标的左边部分的镜头和传感器模块以及用于拍摄目标的右边部分的镜头和传感器模块，并通过使用从两个传感器模块输入的数据生成各个图像数据（即，左图像的媒体数据和右图像的媒体数据）。

[0025] 编码器 20 通过将立体图像构造方法应用到从图像输入单元 10 输出的多个图像数据（左图像和右图像数据）来构造立体图像，然后对立体图像编码。根据运动图像编码方案（例如，MPEG-1、MPEG-2、MPEG-3、MPEG-4 和 H. 264 方案）对立体图像编码的典型设备可用作编码器 20。

[0026] 此外，图像输入单元 10 还可包括用于将特定频段的音频信号转换为电信号的麦克风，并且编码器 20 可通过使用本领域技术人员公知的典型的音频压缩方案来压缩从麦克风输入的音频信号。

[0027] 考虑到与拍摄对象的距离、拍摄角度、拍摄设备的运动等，3D 图像生成器 30 从多个编码的媒体数据生成 3D 图像数据。

[0028] 同时，如果用户长时间地观看仅包括立体图像的 3D 图像，则他 / 她的眼睛比当他 / 她观看单视点图像时感到更加疲劳。因为这个原因，用户难以长时间地观看仅包括立体图像的 3D 图像。因此，3D 图像可以仅包括立体图像，但为通过减轻眼睛的疲劳以保证对 3D 图像的长时间观看，3D 图像最好包括立体图像与单视点图像的适度组合。可根据拍摄设备中的预定设置在拍摄设备的用户的控制下，或通过编辑拍摄的立体图像和单视点图像来组合立体图像和单视点图像。预定设置可以是以固定间隔周期地并重复地拍摄立体图像或单视点图像，根据将要被拍摄的对象性质或种类选择性地拍摄立体图像或单视点图像，或适当地编辑拍摄的图像。

[0029] 图 2 示出根据本发明示例性实施例的包含在 3D 图像中的立体图像的结构。根据本实施例的包含在 3D 图像中的立体图像包括左图像 1 和右图像 5 的组合。这样的立体图像可通过如下方法构造：在图像内将左图像 1 和右图像 5 按 1 : 1 的比例垂直地组合构成一帧，并将帧沿时间轴排列（图 2A）；分别将左图像 1 和右图像 5 垂直地划分，然后按顺序将划分的图像交替地组合构成一帧，并最终将帧沿时间轴排列（图 2B），或通过包括左图像 1 或右图像 5 的帧沿时间轴排列（图 2C）。

[0030] 此外，3D 图像生成器 30 向数据存储控制器 40 提供 3D 图像构造信息，所述 3D 图像构造信息包括当生成 3D 图像时产生的组成视频数据和音频数据的信息。

[0031] 数据存储控制器 40 将 3D 图像数据存储在存储器 50 中，使用从 3D 图像生成器 30 提供的 3D 图像构造信息和 3D 图像的数据生成所述 3D 图像数据。此外，显示器 60 输出由 3D 图像生成器 30 生成的 3D 图像。通信模块 70 包括能够将存储在存储器 50 中的 3D 图像发送到外部装置（例如，移动终端、PDA、装备有无线 LAN 的便携式终端和支持 USB 或串行通信的个人计算机）的接口。

[0032] 图 3A 示出根据本发明示例性实施例的 3D 图像数据的存储格式。参照图 3A，通过数据存储控制器 40 存储在存储器 50 中的 3D 图像数据的存储格式包括：fryp 头 100、mdat 容器 200、moov 容器 300 和 meta 容器 400。

[0033] ftyp 头 100 包含关于存储的数据的文件类型和兼容性的信息。具体地讲,ftyp 头 100 包含表明 3D 图像是仅由立体图像还是由包括立体图像和单视点图像的组复合图像组成的信息。

[0034] 各信道的视频或音频数据以帧为单位被记录在 mdat 容器 200 中。

[0035] moov 容器 300 是作为基于对象的结构被构成的,并且上述 3D 图像构造信息被记录在 moov 300 中。内容信息和更进一步的媒体数据的同步信息也被记录在 moov 容器 300 中,所述内容信息包括:帧率、比特率和媒体数据的图像大小。

[0036] meta 容器 400 包含用于播放单一的立体图像或复合图像的信息。

[0037] 最好,meta 容器 400 被包括在 moov 容器 300 中。

[0038] 尽管在本实施例中已经示出包括在 moov 容器 300 中的 meta 容器 400,但本发明并不局限于此。例如,记录元数据的 meta 容器 400 可作为单独的容器被形成(参照图 3B)。

[0039] 图 4 示出根据本发明示例性实施例的包括在 3D 图像数据的存储格式中的 moov 容器 300 的详细结构。

[0040] 参照图 4,根据本实施例的 moov 容器 300 包括:mvhd 包(box)301、iods 包 302 和 trak 包 303。

[0041] mvhd 包 301 是表示影片头的包,并包含:“mvhd”的大小、类型、版本、文件创建时间、文件修改时间、时间缩放因数和总的播放时间。

[0042] 初始化信息(例如,用于场景构造和对象描述的和级)被记录在 iods 包 302 中。iods 包 302 还可包含用于场景构造的 BIFS(二进制格式场景)流和用于对象描述的 OD(对象描述符)流的设置信息。

[0043] trak 包 303 是记录包括在 3D 图像中的轨道或流的构造信息的字段,并包括 tkhd 包 304、tref 包 305、edts 包 306 和 mdia 包 307。

[0044] 基本轨道信息被记录在 tkhd 包 304 中,所述基本轨道信息包括:诸如包括在轨道中的图像的大小和播放时间、轨道创建时间和轨道修改时间。

[0045] tref 包 305 与数据参考包对应。

[0046] 包括轨道的时间同步信息的列表被记录在 edts 包 306 中。

[0047] 提供 mdia 包 307 以记录关于轨道内的媒体数据的信息,并且 mdia 包 307 包括 mdhd 包 308、hdlr 包 309 和 minf 包 310。

[0048] mdhd 包 308 是代表媒体头的包,包括包含在轨道中的媒体的播放时间、创建时间和修改时间的信息被记录在 mdhd 包 308 中。

[0049] hdlr 包 309 与定义媒体类型的处理包对应。

[0050] 提供 minf 包 310 以记录媒体数据信息,并且 minf 包 310 包括:vmhd 包 311,记录视频媒体头信息;smhd 包 312,记录声音媒体头信息;hmhd 包 313,记录提示媒体头;nmhd 包 314,记录空媒体头信息;stbl 包 315,记录对应的轨道采样的头信息。

[0051] stbl 包 315 包括:stsd 包 316,记录编解码类型信息、初始化信息等;stsc 包 317,记录关于每块(chunk)包含的采样的数量的信息;stts 包 318,记录关于每个采样的解码时间的信息;stsz 包 319,记录关于每个采样的比特流大小的信息;stz2 包 320,记录紧密采样的比特流大小的信息;stco 包 321,记录关于文件中开始地址值的信息,块从所述地址值开始;co64 包 323,记录在大文件的情况下关于文件中开始地址值的信息,块从所述地址值

开始。

[0052] 图 5 是示出根据本发明示例性实施例的包括在三维图像数据的存储格式中的 meta 容器的详细结构的视图。参照图 5, 记录元数据的 meta 容器 400 包括 :hdlr 包 401、iloc 包 402、iinf 包 403、xml 包 404、bxml 包 405 和 snmi 包 406。

[0053] hdlr 包 401 与定义元数据的类型的处理包对应。

[0054] 组成立体图像或单视点图像的各个项目的标识码、各个项目的位置信息和关于各个项目的大小的信息被记录在 iloc 包 402 中。例如, 位置信息可以是存储在与项目对应的媒体数据中的存储器的地址值, 并且关于大小的信息可以是表明与项目对应的比特流的长度的值。

[0055] iinf 包 403 包含表明图像的类型图像类型标识。图像类型标识最好包括用于对项目编码的信息。例如, 当项目为立体图像时, 将图像类型标识设置为具有表明立体图像的“S”的标识码。详细地讲, 图像类型标识可被设置为 S1、S2、S3……Sn。此外, 当项目为单视点图像时, 将图像类型标识设置为用具有表明单视点图像的“M”的标识码。详细地讲, 图像类型标识可被设置为 M1、M2、M3……Mn。

[0056] xml 包是记录 xml 数据的容器, 并且 bxml 包 405 是记录二进制 xml 数据的容器。

[0057] 单视点图像和包括由多个图像 (例如, 左图像和右图像) 组成的立体图像的大小的信息、关于拍摄立体图像的相机的信息、显示信息和立体图像构造信息被记录在 snmi 包 406 中。在本实施例中, snmi 包 406 被作为记录包括由多个图像 (例如, 左图像和右图像) 组成的立体图像的大小的信息、关于拍摄立体图像的相机的信息、显示信息和立体图像构造信息的包示出。然而, 本发明并不局限于此, 仅记录包含在 snmi 包 406 中的信息就足够。

[0058] 更特别地, 可将如下信息记录在 snmi 包 406 中, 所述信息包括: 复合图像的图像宽度和高度、拍摄左图像和右图像的两个相机之间的距离、关于目标的相机镜头的焦距、关于对象的旋转角度、中心焦点、详细的相机排列 (表明是否将拍摄左图像的相机和拍摄右图像的相机分别排列在左边和右边, 或是否将拍摄左图像的相机和拍摄右图像的相机按相互交叉的方式排列的信息)、3D 图像观众与显示设备之间的光程、用于 3D 效果的垂直视差的最大值、左图像和右图像之间的最小视差、左图像和右图像之间的最大视差、用于设置 3D 图像类型的值 (格式如图 2 示出, 格式包括仅为单一的左图像或右图像等)、根据各自的类型排列的图像的大小和顺序、用于在包括在 3D 图像的帧之间指定同步的值、用于设置图像以在左图像和右图像之间进行初始编码的值等。

[0059] 图 6 概念地示出根据本发明示例性实施例的 3D 图像数据的存储格式的示例。参照图 6, 基于 ISO 基本媒体文件格式, 根据本实施例的 3D 图像数据的存储格式举例说明复合图像数据的构造信息被存储在 moov 容器 300 中的情况。复合图像数据可包括至少一个立体图像流和至少一个单视点图像流。考虑到这点, meta 容器 400 中的 snmi 包 406 包含: 单视点图像和由多个图像 (例如, 左图像和右图像) 组成的立体图像的大小、关于拍摄立体图像的相机的信息、显示信息和立体图像构造信息。iloc 包 402 包含: 按顺序分配给立体图像流和单视点图像流的标识 (例如, 项目_ID(item_ID))、存储器的地址值 (例如, 偏移量) 和图像流的长度 (例如, 长度)。此外, iinf 包 403 包含图像类型标识 (例如, 项目_名称(item_name))。

[0060] 最好, 将 iloc 包 402 与 iinf 包 403 形成一个包。

[0061] 图 7 概念地示出根据本发明示例性实施例的 3D 图像数据的存储格式的另一示例。参照图 7, 与图 6 不同, 根据本实施例的 3D 图像数据的存储格式举例说明复合图像包括两个基本流。例如, 两个基本流可以是一个左图像流和一个右图像流。因此, moov 容器 300 包括两个 trak 包 303, 所述 trak 包 303 记录关于各个流的构造信息。meta 容器 400 中的 snmi 包 406 包含: 单视点图像和立体图像的大小、关于拍摄立体图像的相机的信息、显示信息和立体图像构造信息。iloc 包 402 包含: 按顺序分配给立体图像流和单视点图像流的标识 (例如, 项目_ID)、存储器的地址值 (例如, 偏移量) 和图像流的长度 (例如, 长度), 并且 iinf 包 403 包含图像类型标识 (例如, 项目_名称)。

[0062] 可注意到, 即使当复合图像包括两个基本流时, 也可容易地使用 meta 容器 400 提供的 snmi 包 406、iloc 包 402 和 iinf 包 403 存储复合图像。

[0063] 在根据本发明示例性实施例的 3D 图像数据的存储格式中, 尽管包括在 moov 容器 300 中的 meta 容器 400 如图 6 和图 7 被示出, 但本发明并不局限于此。例如, 记录元数据的 meta 容器 400 可被包括在 ftyp 头 100 中或作为单独的容器。

[0064] 由于本发明中提出的格式与从国际标准文件格式 (即, ISO 基本媒体文件格式) 扩展的文件格式兼容, 可将包括复合图像的数据发送或应用到多种多媒体应用。

[0065] 同时, 根据本发明, 数据存储控制器 40 按 3D 图像数据存储格式将 3D 图像生成器 30 生成的 3D 图像数据存储于存储器中。根据本发明示例性实施例, 参照上述组成元件和 3D 图像数据存储格式, 现在将描述生成 3D 图像数据的存储格式的过程。

[0066] 数据存储控制器 40 生成 ftyp 头 100、mdat 容器 200、moov 容器 300 和 meta 容器 400, 并按包括生成的数据的格式存储 3D 图像数据。

[0067] 首先, 数据存储控制器 40 生成 ftyp 头 100。关于将要存储的 3D 图像数据的文件类型和兼容性的信息被包括在 ftyp 头 100 中。数据存储控制器 40 从 3D 图像生成器 30 接收信息, 并将该信息包括并记录在 ftyp 头 100 中, 所述信息表明 3D 图像是仅包括立体图像还是包括包含立体图像与单视点图像的组的复合图像。例如, 数据存储控制器 40 可通过执行与下面表 1 中给出的程序对应的操作来设置信息, 该信息表明 3D 图像仅包括立体图像还是包括包含立体图像与单视点图像的组的复合图像。

[0068] 表 1

[0069]

```
aligned(8) class FileTypeBox extends Box('type'){
    unsigned int(32)    major_brand;
        unsigned int(32) minor_version;
        unsigned int(32) compatible_brands[];    //至包的末尾
}
```

[0070] 即, 数据存储控制器 40 通过使用 ftyp 头的标记设置标识信息, 所述信息代表关于立体的内容是否部分地包括单视点数据的立体 MAF (多媒体应用文件格式)。例如, 当 3D 图像仅包括单一的立体图像时, 将作为标识信息使用的标记设置为 “ss01”, 并在当 3D 图像包括立体图像与单视点图像的组时设置为 “ss02”。在设置关于立体图像的文件类型和兼

容性的信息时,本实施例没有具体地指定 ftyp 头的标记,而是设置可能使用的诸如 major_brand, minor_brand, compatible_brand 等多种标记。

[0071] 接下来,数据存储控制器 40 基于从 3D 图像生成器 30 接收的数据生成 mdat 容器 200。各信道的视频数据或音频数据以帧为单位被记录在 mdat 容器 200 中。

[0072] 数据存储控制器 40 还基于从 3D 图像生成器 30 接收的数据生成 moov 容器 300。3D 图像构造信息作为基于对象的结构被记录在 moov 300 中,所述 3D 图像构造信息与存储在 mdat 容器 200 中的各个 3D 图像数据对应。此外,包括媒体数据的帧率、比特率和图像大小的内容信息以及媒体数据的同步信息被记录在 moov 容器 300 中。

[0073] 特别地,moov 容器 300 包括 mvhd 包 301 和 trak 包 303。mvhd 包 301 是代表影片头的包,并包含“mvhd”的大小、类型、版本、文件创建时间、文件修改时间、时间缩放因数和总的文件播放时间。

[0074] 最好,moov 容器 300 可根据 3D 图像相关信息的存储格式选择性地包括 iods 包 302。初始化信息(例如,用于场景构造和对象描述的类和级)被记录在 idos 包 302 中。idos 包 302 还可包含用于场景构造的 BIFS 流和用于对象描述的 OD 流的设置信息。

[0075] 此外,trak 包 303 可包括:tkhd 包 304、tref 包 305、edts 包 306 和 mdia 包 307。包括诸如包括在轨道中的图像的大小和播放时间、轨道创建时间和 轨道修改时间的基本轨道信息被记录在 tkhd 包 304 中。数据参考可被包含在 tref 包 305 中,并且包括轨道的时间同步信息的列表被记录在 edts 包 306 中。此外,关于轨道内媒体数据的信息可被记录在 mdia 包 307 中。

[0076] 更特别地,mdia 包 307 可包括:mdhd 包 308、hdlr 包 309 和 minf 包 310。mdhd 包 308 是代表媒体头的包,并包含:包括在轨道中的媒体的播放时间、创建时间和修改时间的信息。此外,定义媒体类型的处理数据可被记录在 hdlr 包 309 中。minf 包 310 是被提供用于记录媒体数据信息的字段,并可包括:vmhd 包 311,记录视频媒体头信息;smhd 包 312,记录声音媒体头信息;hmhd 包 313,记录提示媒体头信息;nmhd 包 314,记录空媒体头信息;stbl 包 315,记录相应的轨道采样的头信息。

[0077] 此外,stbl 包 315 可包括:stsd 包 316,记录编解码类型信息、初始化信息等;stsc 包 317,记录关于每块包含的采样的数量的信息;stts 包 318,记录关于每个采样的解码时间的信息;stsz 包 319,记录关于每个采样的比特流大小的信息;stz2 包 320,记录紧密采样的比特流大小信息;stco 包 321,记录关于文件中开始地址值的信息,块从所述地址值开始;co64 包 323,记录在大文件的情况下关于文件中开始地址值的信息,块从所述地址值开始。

[0078] 另外,数据存储控制器 40 生成 meta 容器 400,并将从 3D 图像生成器 30 接收的关于 3D 图像数据的元数据的信息存储到 meta 容器 400 中。这样,meta 容器 400 包括:hdlr 包 401、iloc 包 402、iinf 包 403、xml 包 404 和 bxml 包 405(见图 5)。此外,数据存储控制器 40 可进一步的将记录立体图像信息的 snmi 包 406 合并到 meta 容器 400。在根据本发明示例性实施例的 3D 图像数据的存储格式中,即使包括在 moov 容器 300 中的 meta 容器 400 已被示出,但本发明也不局限于此。例如,记录元数据的 meta 容器 400 可被包括在 ftyp 头 100 中或作为单独的容器。

[0079] 定义元数据类型的数据被记录在 hdlr 包 401 中。此外,组成立体图像或单视点图

像的各个项目的标识码和关于各个项目的长度的信息被记录在 iloc 包 402 中,并且关于各个项目的信息被记录在 iinf 包 403 中。此外,xml 数据被记录在 xml 包 404 中,并且二进制 xml 数据被记录在 bxml 包 405 中。此外,由多个图像(例如,左图像和右图像)组成的立体图像的大小、关于拍摄立体图像的相机的信息、显示信息和立体图像构造信息可被记录在 snmi 包 406 中。例如,可通过与执行下面表 2 给出的程序对应的操作来设置将要记录在 snmi 包 406 中的信息,所述程序被存储在数据存储控制器 40 中。

[0080] 表 2

[0081]

```
aligned(8) class StereoscopicandMonoInformation box extend FullBox('snmi')
version=0,0){
    //复合图像大小
    unsigned int(16)    compound_image_width;
    unsigned int(16)    compound_image_height;

    //立体相机信息
    unsigned int(32)    baseline;
    unsigned int(32)    focallength;
    unsigned int(32)    rotation;
    unsigned int(32)    convergence;
    unsigned int(1)     camera_setting;
    unsigned int(7)     reserved;

    //立体显示信息
    unsigned int(16)    ViewingDisplaySize;
    unsigned int(16)    MaxVerticalDisparity;
    int(16)             MinofDisparity;
    int(16)             MaxofDisparity;

    //
    unsigned int(8)     SteroScopi_ES_Type;
    unsigned int(1)     frame_sync;
    unsigned int(1)     LR_first;
    unsigned int(6)     reserved;
    if(StereoScopic_ES_Type==1)    //并排格式
    {
        unsigned int(16)    left_image_width;
        unsigned int(16)    right_image_width;
```

[0082]

```

    }
    else if(StereoScopic_ES_Type==1)    //垂直线交错格式
    {
        unsigned int(16)    odd_line_width;
        unsigned int(16)    odd_line_count;
        unsigned int(16)    even_line_width;
        unsigned int(16)    even_line_count;
    }
    else if(StereoScopic_ES_Type==3)    //场序制格式
    {
        unsigned int(16)    field_width;
        unsigned int(16)    field_height;
    }
}

```

[0083] 在表 2 中,“compound_image_width”表示复合图像显示宽度,且“compound_image_height”表示复合图像显示高度。此外,表现相机信息的“baseline”表示拍摄左图像和右图像的相机之间的距离,“focallength”表示关于目标的相机镜头的焦距,“rotation”表示两个相机之间的旋转角度,“convergence”表示中心焦点与基线之间的距离,“camera_setting”表示根据下面给出的表 3 定义的详细的相机排列。此外,“left_image_width”表示并排格式中左图像的宽度,“right_image_width”表示并排格式中右图像的宽度,“odd_line_width”表示垂直线交错格式中奇数线的宽度,“odd_line_count”表示垂直线交错格式中奇数线的顺序,“even_line_width”表示垂直线交错格式中偶数线的宽度,“even_line_count”表示垂直线交错格式中偶数线的顺序。此外,“field_width”表示场序制格式中的图像的宽度,并且“field_height”表示场序制格式中的图像的高度。

[0084] 表 3

[0085]

值	说明
0	平行排列
1	交叉排列

[0086] 在表 3 中,“平行排列”表明拍摄左图像的相机和拍摄右图像的相机分别排列在左边和右边,“交叉排列”表明拍摄左图像的相机和拍摄右图像的相机按照相互交叉的方式排列。

[0087] 此外,在表 2 中,表现关于显示设备的信息的“ViewingDisplaySize”表示 3D 图像观众与显示设备之间的光程,“MaxVerticalDisparity”表示用于 3D 效果的垂直视差的最大值,“MinofDisparity”表示左图像和右图像之间的最小视差,“MaxofDisparity”表示左图像和右图像之间的最大视差。此外,“StereoScopic_ES_type”表示用于设置 ES 类型的值,由表 4 在下面给出。

[0088] 表 4

[0089]

立体组成类型	标识
0	并排格式
1	垂直线交错格式
2	帧序制格式
3	场序制格式
4	立体的左视图顺序
5	立体的右视图顺序

[0090] 在表 4 中,“并排格式”是如图 2A 中形成左图像和右图像的格式,“垂直线交错格式”是如图 2B 中形成左图像和右图像的格式,“帧序格式”是如图 2C 中形成左图像和右图像的格式,“立体的左视图顺序”表示仅由左图像形成的 3D 图像,“立体的右视图顺序”表示仅由右图像形成的 3D 图像。

[0091] 此外,在表 2 中,“frame_sync”表示用于在包括在 3D 图像中的帧之间指定同步的值,“LR_first”表示用于设置图像以在左图像和右图像之间进行初始编码的值。“LR_first”根据下面给出的表 5 指定。

[0092] 表 5

[0093]

标识	LR_First=0		LR_First=1	
	左视图顺序	右视图顺序	左视图顺序	右视图顺序
并排	左边	右边	右边	左边
垂直线交错	奇数线	偶数线	偶数线	奇数线
帧序制	奇数帧	偶数帧	偶数帧	奇数帧
场序制	奇数场	偶数场	偶数场	奇数场

[0094]

n ES	主媒体	子媒体	子媒体	主媒体
------	-----	-----	-----	-----

[0095] 尽管在本实施例中 ftyp 头 100、mdat 容器 200、moov 容器 300 和 meta 容器 400 按生成顺序被示出并被存储在存储器 50 中,但本发明并不局限于此。不管 ftyp 头 100、mdat 容器 200、moov 容器 300 和 meta 容器 400 的生成顺序如何,按包括 ftyp 头 100、mdat 容器 200、moov 容器 300 和 meta 容器 400 的格式已足以将 3D 图像数据存储在存储器 50 中。

[0096] 通过以上过程,可将通过 3D 图像生成器 30 生成的 3D 图像数据按包括 ftyp 头 100、mdat 容器 200、moov 容器 300 和 meta 容器 400 的格式存储在存储器 50 中。这样,能够按照与国际标准文件格式(即,ISO 基本媒体文件格式)兼容的方式管理 3D 图像数据。

[0097] 图 8 示出根据本发明另一示例性实施例的立体图像的结构示例。参照图 8,根据本实施例的包括在复合图像中的立体图像可包括多个片段。例如,立体图像可包括片段 S1、S2、S3、S4 和 S5。可在立体图像信息被改变的点相互区分各个片段。

[0098] 图 9 是根据本发明的另一示例性实施例的根据三维图像数据的存储格式的示例的示意图。参照图 9,基于 ISO 基本媒体文件格式,根据本实施例的 3D 图像数据的存储格式举例说明将复合图像数据的构造信息存储在 moov 容器 300 的情况。

[0099] meta 容器 400 中的 snmi 包 406 包含:单视点图像和由多个图像(例如,左图像和右图像)组成的立体图像的大小、关于拍摄立体图像的相机的信息、显示信息和立体图像构造信息。具体地讲,包括在立体图像中的片段的数量在 snmi 包 406 中被定义。通过相同的构造信息对片段分组。此外,对应于分组的信息被记录在 snmi 包 406 中。

[0100] iloc 包 402 包含:基于立体图像信息被改变的点按顺序分配给立体图像流和单视点图像流的标识(例如,项目_ID)、存储器的地址值(例如,偏移量)和图像流的长度(例如,长度),并且 iinf 包 403 包含图像类型标识(例如,项目_名称)。例如,当项目为立体图像时,图像类型标识被设置为具有表明立体图像的“S”的标识码。详细地讲,图像类型标识可被设置为 S1、S2、S3……Sn。此外,当项目为单视点图像时,图像类型标识被设置为具有表明单视点图像的“M”的标识码。详细地讲,图像类型标识可设置为 M1、M2、M3……Mn。

[0101] 最好,将 iloc 包 402 和 iinf 包 403 形成一个包。

[0102] 此外,即使当复合图像包含两个基本流时,也可使用上述的包括多个片段的立体图像。图 10 示出根据本发明另一示例性实施例的 3D 图像数据的存储格式的另一示例。参照图 10,与图 6 不同,根据本实施例的 3D 图像数据的存储格式举例说明复合图像包括两个基本流。例如,两个基本流可以是左图像流和右图像流。因此,moov 容器 300 包含两个记录关于各个流的构造信息的 trak 包 303。meta 容器 400 中的 snmi 包 406 包含:单视点图像和由多个图像(例如,左图像和右图像)组成的立体图像的大小、关于拍摄立体图像的相机的信息、显示信息和立体图像构造信息。具体地讲,包括在立体图像中的片段的数量在 snmi 包 406 中被定义。通过相同的构造信息对片段分组。此外,对应于分组的信息被记录在 snmi 包 406 中。

[0103] iloc 包 402 包含:基于立体图像信息被改变的点按顺序分配给立体图像流和单视点图像流的标识(例如,项目_ID)、存储器的地址值(例如,偏移量)和图像流的长度(例如,长度),iinf 包 403 包含图像类型标识(例如,项目_名称)。这里,用于分配标识(例如,项目_ID)和图像类型标识(例如,项目_名称)的根据是立体图像信息被改变的点。

[0104] 在根据本发明实施例的 3D 图像数据的存储格式中,尽管已在图 9 和图 10 中示出包括在 moov 容器 300 中的 meta 容器 400,但本发明并不局限于此。例如,记录元数据的 meta 容器 400 可在 ftyp 头 100 中或是作为单独的容器被形成。

[0105] 在生成 meta 容器 400 时,数据存储控制器 40 考虑到各立体图像以片段为单位被分离的事实。即,数据存储控制器 40 通过执行与下面的表 6 给出的程序对应的操作来识别包括在各立体图像中的片段数量(item_count)并按顺序分配项目_ID(item_ID)。此外,当存在包含相同信息的片段时,数据存储控制器 40 通过使用单独的标识符(附属标志(dependence flag))表示自己。此外,数据存储控制器 40 通过执行与表 6 的程序对应的操作将单独的项目_ID分配给包含相同信息的片段。例如,单独的项目_ID可以是附属_项目_ID(dependence_item_ID)。单独的项目_ID(即,附属_项目_ID)如包含在特定项目_ID的片段中一样表明包含相同信息的片段,并可通过参考现有的项目_ID被使用。

[0106] 表 6

[0107]

Syntax

```
aligned(8) class StereoscopicandMonoInformation box extend FullBox('snmi'  
version=0,0){  
    //立体视觉类型信息  
    unsigned int(1)    Is_VideoSafety;  
    if(Is_VideoSafety){  
    }  
    unsigned int(16)    item_count;  
    for(i=0;i<item_count;i++){  
        unsigned int(16)    item_ID;  
        unsigned int(1)    Is_StereoScopic_Information;  
        unsigned int(7)    reserved;  
        if(Is_StereoScopic_Information){  
            unsigned int(1)    dependence_flag;  
            unsigned int(7)    reserved;  
            if(dependence_flag==1){  
                unsigned int(16)    dependence_item_ID;  
            }else{  
                //立体相机信息  
                unsigned int(1)    Is_CamParams;  
                //立体显示信息  
                unsigned int(1)    Is_DisplayInformation;  
                unsigned int(6)    reserved;  
  
                if(Is_CamParams) {  
                    unsigned int(32)    baseline;  
                    unsigned int(32)    focallength;  
                    unsigned int(32)    ConvergenceDistance;  
                    unsigned int(1)    Is_camera_cross;  
                    unsigned int(7)    reserved;
```

[0108]

```

        if(Is_camera_cross){
            unsigned int(32)    rotation[];
        }
    }

    if(Is_DisplayInformation){
        unsigned int(16)    ViewingDistance;
        int(16)             MinofDisparity;
        int(16)             MaxofDisparity;
    }
}
}
}
}

```

[0109] 另外,数据存储控制器 40 可通过执行与下面的表 7 给出的程序对应的操作对包含相同信息的片段分组。即,使用在 ISO 基本媒体文件格式中提供的“extent_count”,数据存储控制器 40 计算并表明包括在各立体图像中的不同片段的类型的数量,并根据各自的类型记录立体图像相关信息。

[0110] 表 7

[0111]

```

Syntax
aligned(8) class StereoscopicandMonoInformation box extend FullBox('snmi'
version=0,0){
    //立体视觉类型信息
    unsigned int(8)    Stereoscopic_Composition_Type;
    unsigned int(1)    LR_First;
    unsigned int(1)    Is_VideoSafety;
    unsigned int(6)    reserved;
    if(Is_VideoSafety){
    }
    unsigned int(16)    item_count;
}

```

[0112]

```
for(i=0;i<item_count;i++){
    unsigned int(16)    item_ID;
    unsigned int(16)    extent_count;
    for(j=0;j<extent_count;j++){
        //立体相机信息
        unsigned int(1)    Is_CamParams;
        //立体显示信息
        unsigned int(1)    Is_DisplayInformation;
        unsigned int(6)    reserved;
        if(Is_CamParams) {
            unsigned int(32)    baseline;
            unsigned int(32)    focallength;
            unsigned int(32)    ConvergenceDistance;
            unsigned int(1)    Is_camera_cross;
            unsigned int(7)    reserved;
            if(Is_camera_cross){
                unsigned int(32)    rotation[];
            }
        }
        if(Is_DisplayInformation){
            unsigned int(16)    ViewingDistance;
            int(16)            MinofDisparity;
            int(16)            MaxofDisparity;
        }
    }
}
```

[0113] 此外,数据存储控制器 40 可执行综合操作,所述综合操作是用于分配与包括在各立体图像中的片段的数量一样多的项目_ID的操作(例如,与表 6 的程序对应的操作)和用于对包含相同信息的片段分组的操作(例如,与表 7 的程序对应的操作)。这样的综合操作可由与下面的表 8 中给出的程序对应的操作实现。

[0114] 表 8

[0115]

Syntax

```
aligned(8) class StereoscopicandMonoInformation box extend FullBox('snmi'  
version=0,0){
```

```
    //立体视觉类型信息
```

```
        unsigned int(8)    Stereoscopic_Composition_Type;
```

```
        unsigned int(1)    LR_First;
```

```
        unsigned int(1)    Is_VideoSafety;
```

```
        unsigned int(6)    reserved;
```

```
    if(Is_VideoSafety){
```

```
        }
```

```
        unsigned int(16)    item_count;
```

```
        for(i=0;i<item_count;i++){
```

```
            unsigned int(16)    item_ID;
```

```
            unsigned int(16)    extent_count;
```

```
            unsigned int(8)    Is_extent_information[extent_count];
```

```
            for(j=0;j<extent_count;j++){
```

```
                if(Is_extent_information[j]){
```

```
                    unsigned int(16)    dependence_extent_index;
```

```
                }else{
```

```
                    //立体相机信息
```

```
                        unsigned int(1)    Is_CamParams;
```

```
                    //立体显示信息
```

```
                        unsigned int(1)    Is_DisplayInformation;
```

```
                        unsigned int(6)    reserved;
```

```
                    if(Is_CamParams) {
```

```
                        unsigned int(32)    baseline;
```

```
                        unsigned int(32)    focallength;
```

[0116]

```
        unsigned int(32)    ConvergenceDistance;
        unsigned int(1)     Is_camera_cross;
        unsigned int(7)     reserved;
        if(Is_camera_cross){
            unsigned int(32)    rotation[];
        }
    }
    if(Is_DisplayInformation){
        unsigned int(16)    ViewingDistance;
        int(16)             MinofDisparity;
        int(16)             MaxofDisparity;
    }
}
}
```

[0117] 根据在本发明中提出的数据存储格式,3D 图像数据可按照与国际标准文件格式(即,ISO 基本媒体文件格式)兼容的格式被存储,并且可将存储的数据发送或是应用于多种多媒体应用。

[0118] 尽管已经参照本发明的特定示例性实施例对本发明进行了显示和描述,但是本领域的技术人员应该理解,在不脱离由权利要求所限定的本发明的精神和范围的情况下,可以对其进行形式和细节上的各种改变。

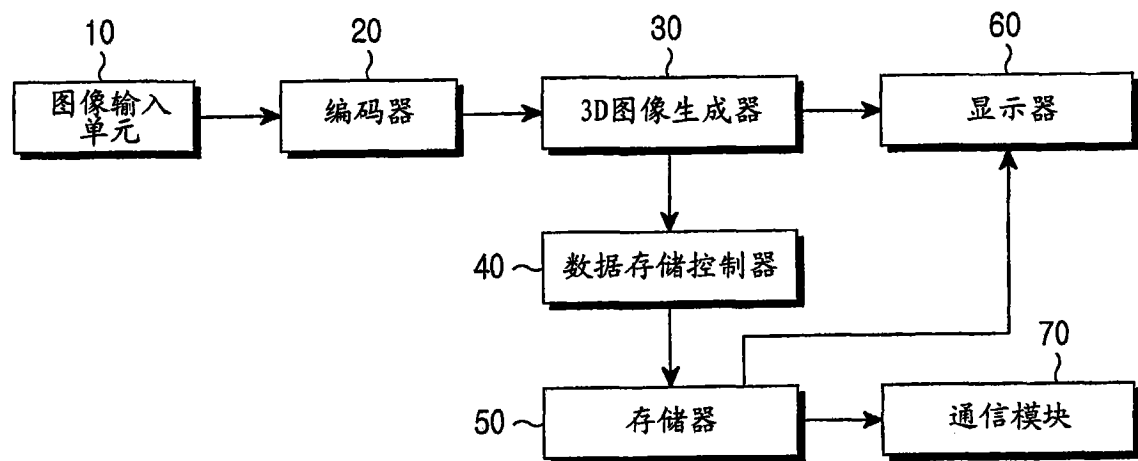


图 1

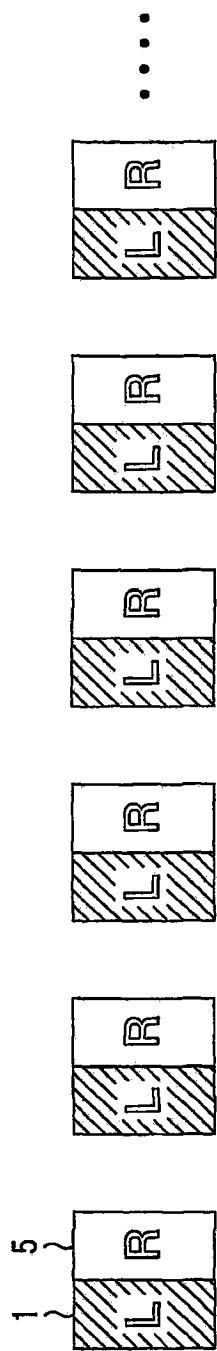


图 2A

n 并排格式

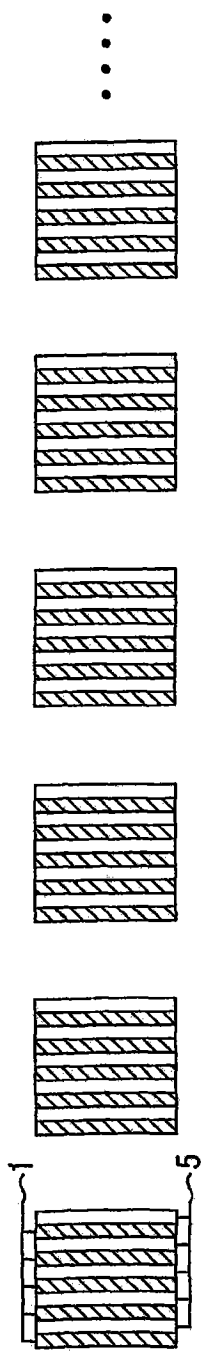


图 2B

n 垂直线交错格式

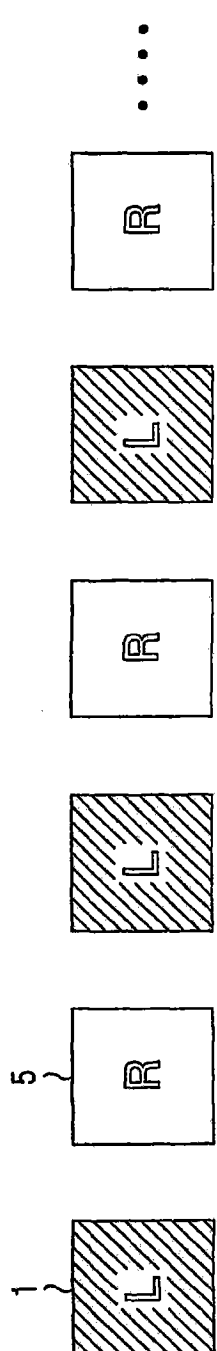
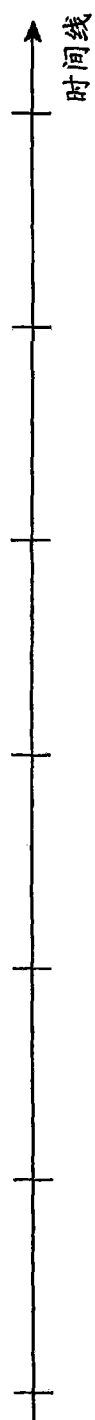


图 2C

n 帧或场序列格式



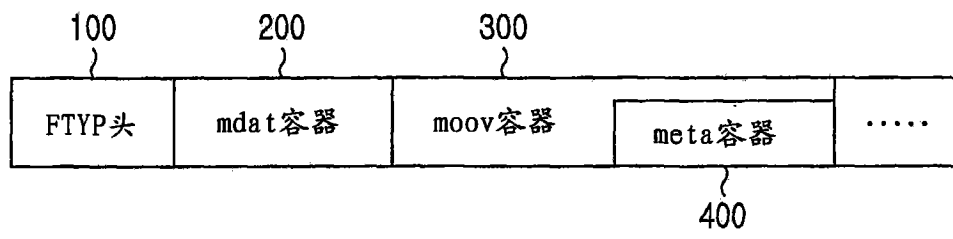


图 3A

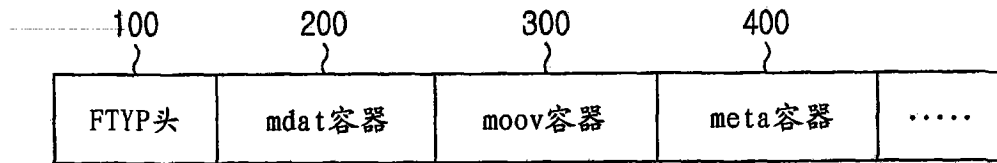


图 3B

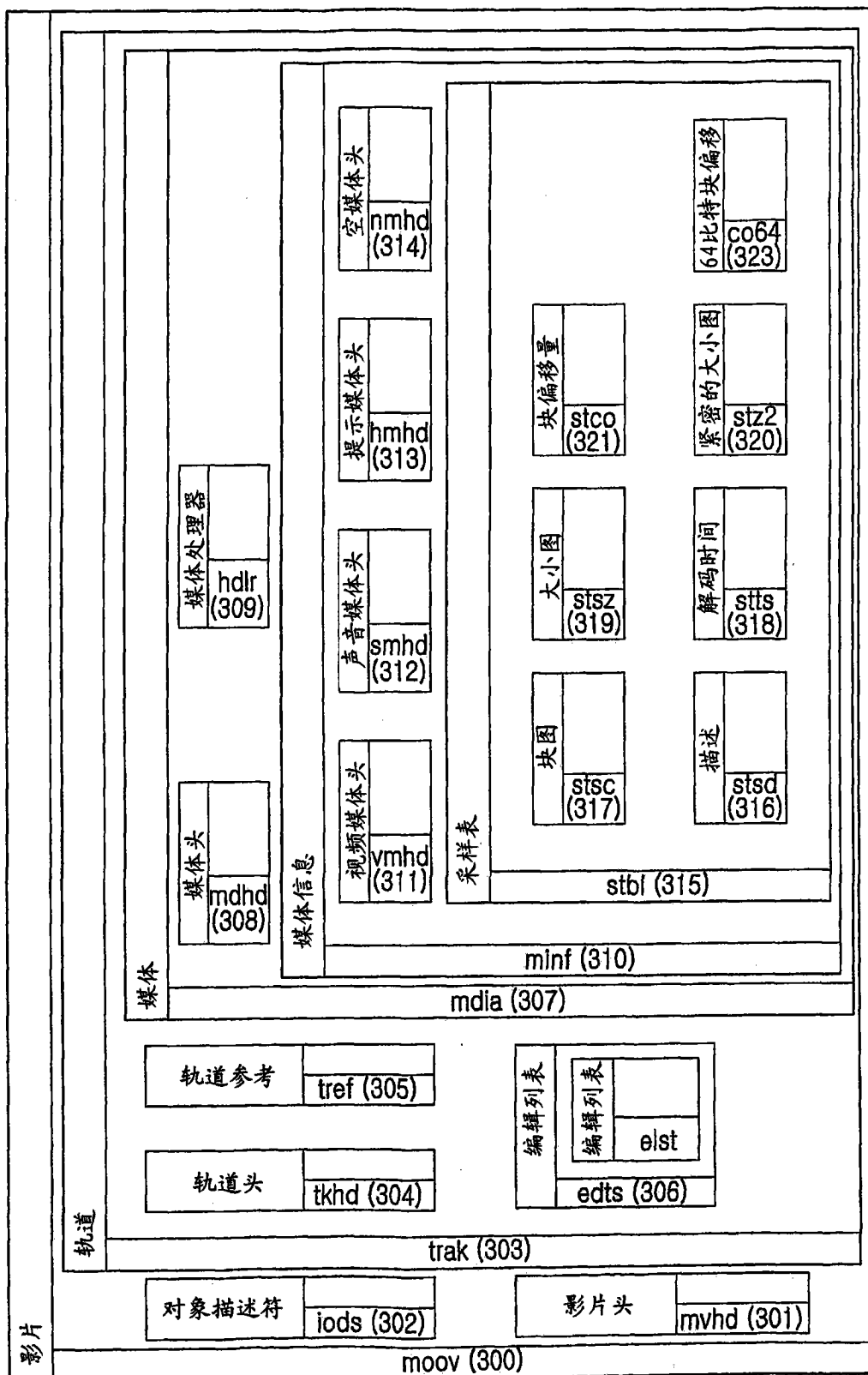


图 4

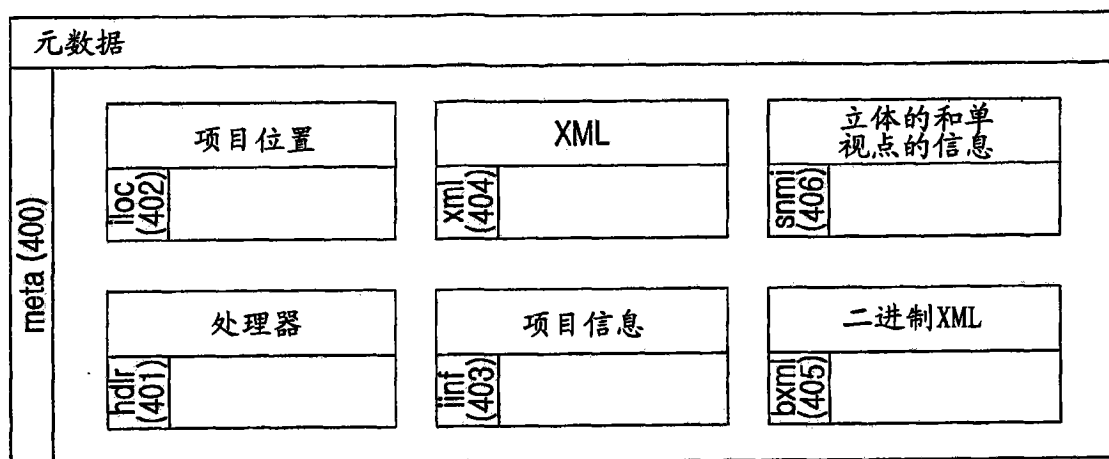


图 5

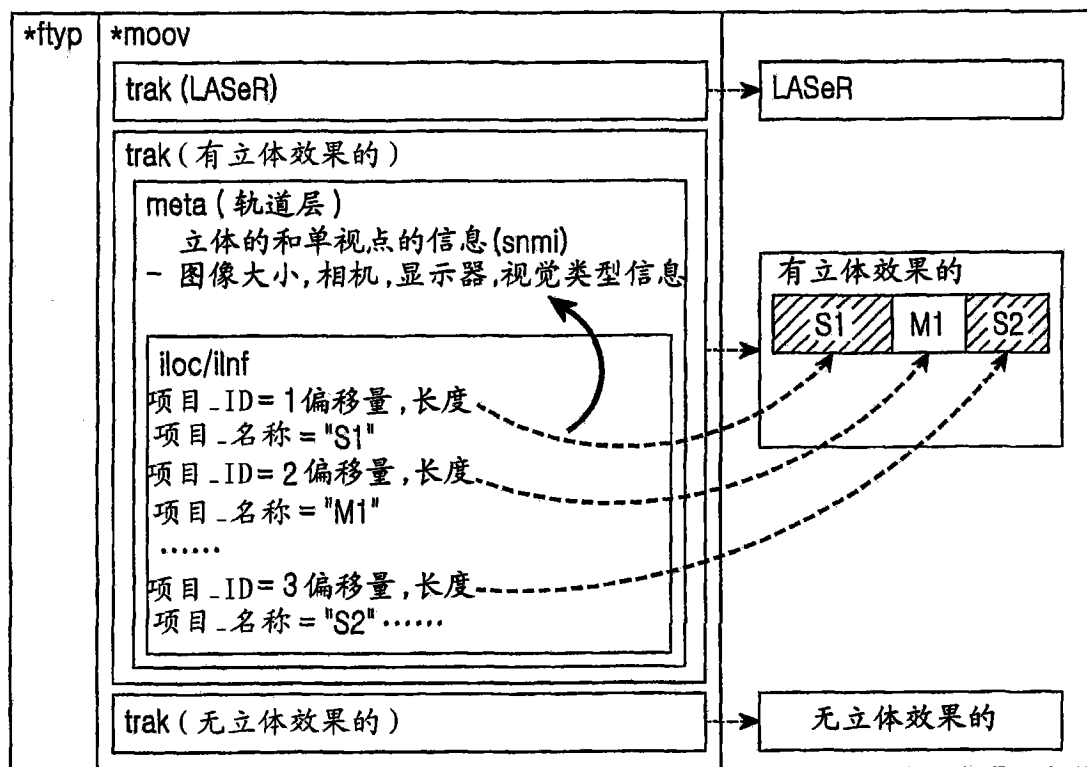


图 6

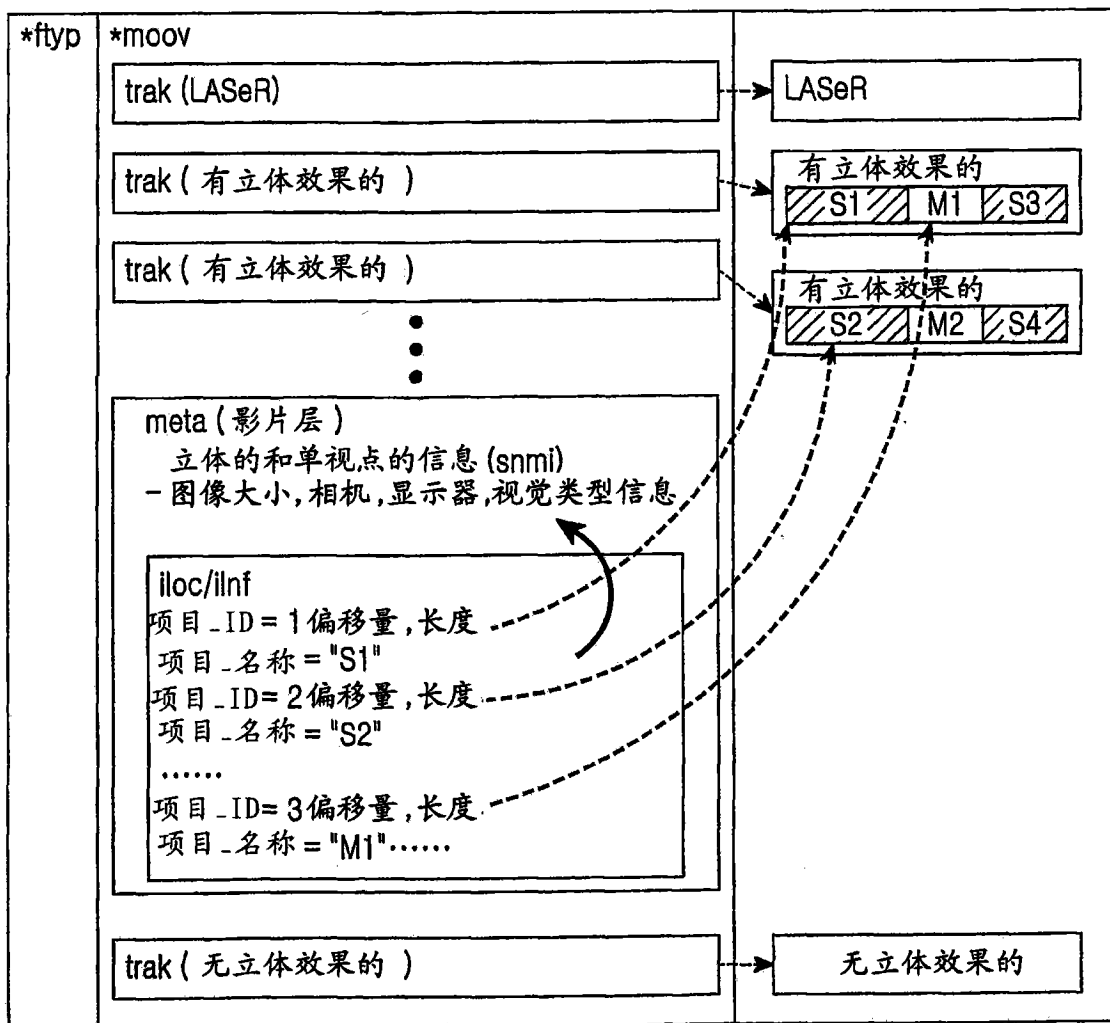


图 7

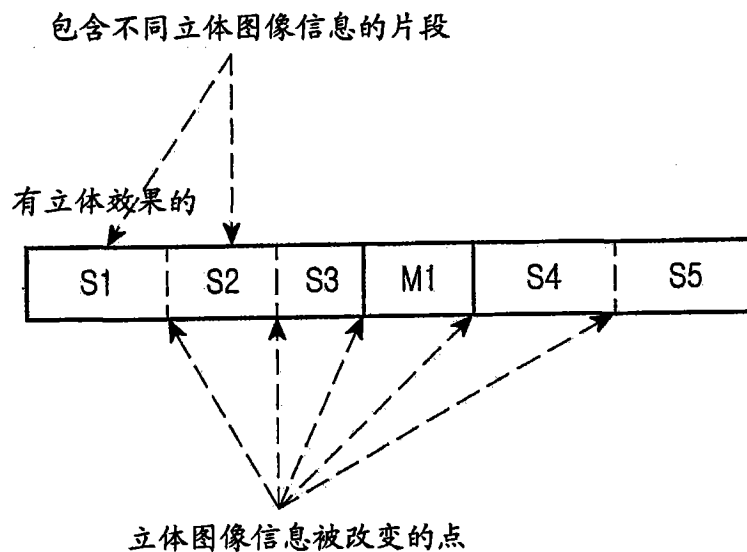


图 8

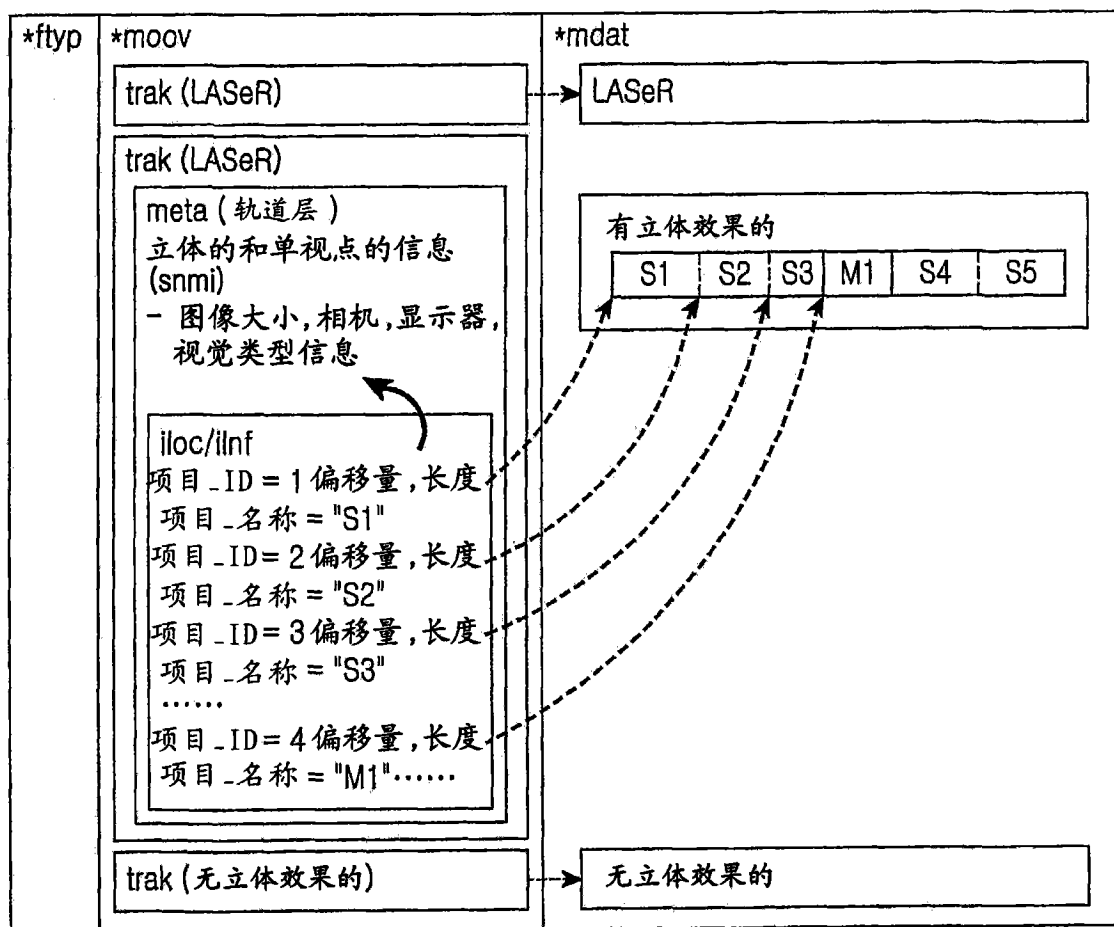


图 9

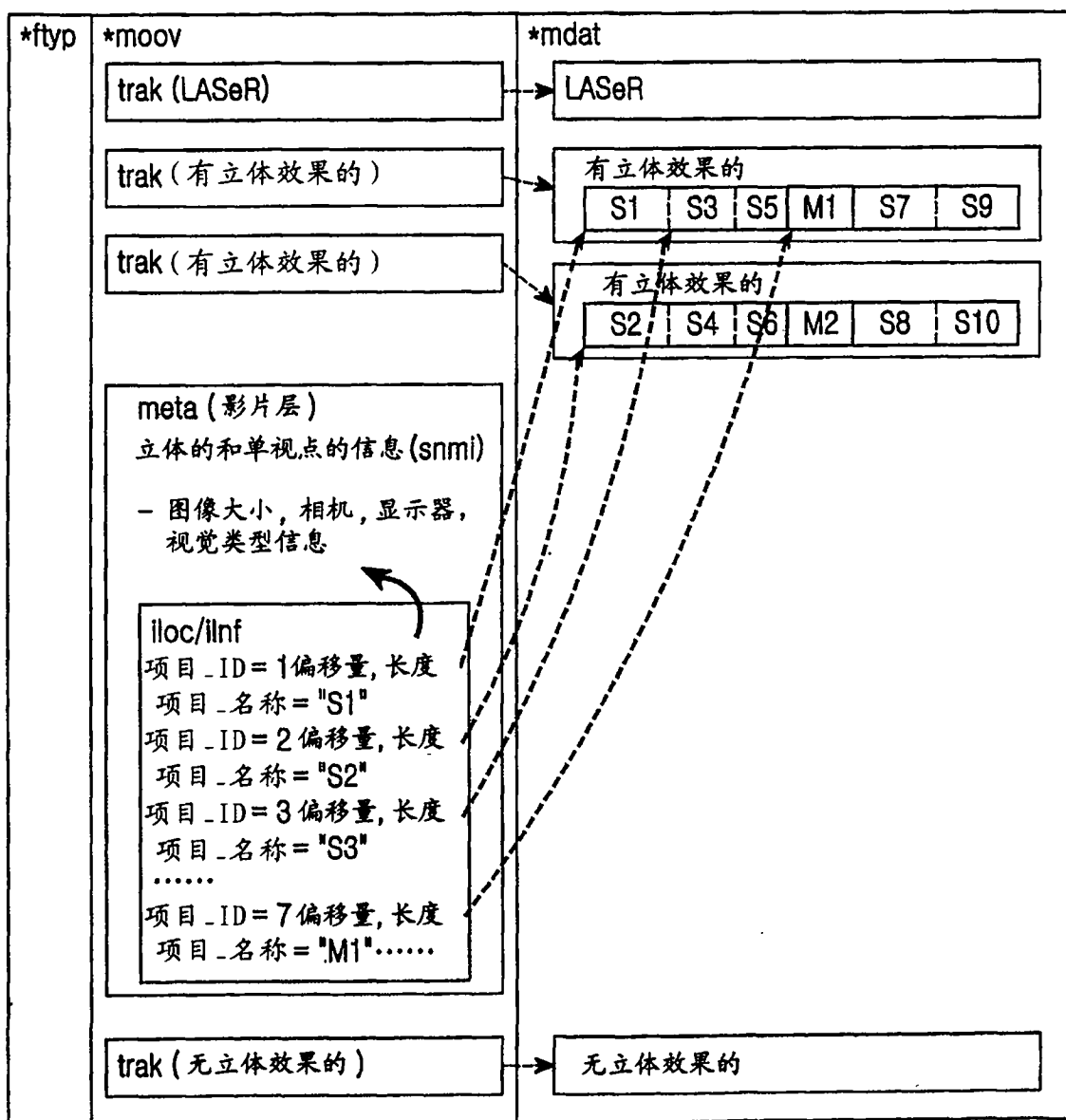


图 10