



(12)发明专利

(10)授权公告号 CN 104702960 B

(45)授权公告日 2018.01.02

(21)申请号 201510144003.7

(22)申请日 2010.01.26

(65)同一申请的已公布的文献号
申请公布号 CN 104702960 A

(43)申请公布日 2015.06.10

(30)优先权数据
61/205,938 2009.01.26 US
61/269,955 2009.07.01 US

(62)分案原申请数据
201080013860.9 2010.01.26

(73)专利权人 汤姆森特许公司
地址 法国伊西莱穆利诺

(72)发明人 田栋 P.赖 罗健聪 尹鹏

(74)专利代理机构 北京市柳沈律师事务所
11105
代理人 吕晓章 叶齐峰

(51)Int.Cl.

H04N 19/597(2014.01)

H04N 19/70(2014.01)

H04N 19/61(2014.01)

H04N 19/117(2014.01)

H04N 19/80(2014.01)

H04N 19/86(2014.01)

H04N 19/90(2014.01)

(56)对比文件

WO 2008127676 A3,2008.12.18,
SULLIVAN等.Draft AVC amendment text
to specify Constrained Baseline profile
and supplemental enhancement information.
《JOINT VIDEO TEAM OF ISO/IEC JTC1/SC29/
WG11 AND ITU-T SG.16》.2008,

审查员 郑洪磊

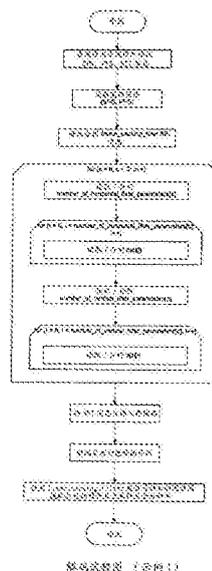
权利要求书1页 说明书44页 附图41页

(54)发明名称

用于视频解码的装置

(57)摘要

提供例如涉及视频编码和解码中的视图平铺的实现方式。一种具体实现方式访问包括组合成单个画面的多个画面的视频画面,并且访问指示如何组合访问的视频画面中的多个画面的附加信息。访问的信息包括空间交织信息和采样信息。另一种实现方式编码包括组合成单个画面的多个画面的视频画面,并且生成指示如何组合访问的视频画面中的多个画面的信息。生成的信息包括空间交织信息和采样信息。形成包括编码的视频画面和生成的信息的比特流。另一种实现方式提供用于传送生成的信息的数据结构。



1. 一种用于视频解码的装置,包括:

解码器,该解码器被配置为访问包括被组合成单个画面的多个画面的视频画面,所述视频画面是接收的视频流的一部分;

并被配置为访问作为接收的视频流的一部分的信息,所访问的信息指示如何组合访问的视频画面中的所述多个画面,其中,所访问的信息包括空间交织信息和采样信息,其中空间交织信息指示空间交织,所述空间交织是在形成所述单个画面中应用于所述多个画面的并排交织、顶部到底部交织、逐行交织、逐列交织或棋盘交织,并且其中采样信息指示用于将多个画面中的每一个恢复至另一分辨率的一个或多个与上采样滤波器有关的参数,该一个或多个与上采样滤波器有关的参数包括滤波方向的指示;并且

被配置为解码视频画面以提供该多个画面中的至少一个的解码表示。

2. 一种用于视频解码的装置,包括:

解调器,用于解调包括接收的视频流的信号;以及

解码器,被配置为访问包括被组合成单个画面的多个画面的视频画面,所述视频画面是接收的视频流的一部分;并被配置为访问作为接收的视频流的一部分的信息,访问的信息指示如何组合访问的视频画面中的所述多个画面,其中,访问的信息包括空间交织信息和采样信息,其中空间交织信息指示空间交织,所述空间交织是在形成所述单个画面中应用于所述多个画面的并排交织、顶部到底部交织、逐行交织、逐列交织或棋盘交织,并且其中采样信息指示用于将多个画面中的每一个恢复至另一分辨率的一个或多个与上采样滤波器有关的参数,该一个或多个与上采样滤波器有关的参数包括滤波方向的指示;并且被配置为解码视频画面以提供该多个画面中的至少一个的解码表示。

用于视频解码的装置

[0001] 本申请是申请日为2010年1月26日、申请号为201080013860.9、发明名称为“用于视频编码的帧打包”的发明专利申请的分案申请。

[0002] 相关申请的交叉引用

[0003] 本申请要求(1)名称为“Spatial Interleaving”、2009年1月26日提交的美国临时申请序列号61/205,938(代理人案号PU090010)以及(2)名称为“Communicating Filter Parameters For Frame Packing Applications”、2009年7月1日提交的美国临时申请序列号61/269,955中的每个的权益。针对所有目的在此通过引用明确地并入这两个申请的每一个的全部内容。

[0004] 本申请还涉及名称为“Tiling In Video Encoding And Decoding”、具有国际申请日2008年4月11日的国际申请第PCT/US2008/004747。本申请还被标识为美国专利申请序列号12/450,829(代理人案号PU070078)。

技术领域

[0005] 描述一般涉及视频编码和/或解码领域的实现方式。

背景技术

[0006] 随着包括立体和自动立体显示器的3D显示器在市场上的出现,存在对于更多可用的3D内容的强烈需求。编码通常涉及多个视图以及可能的相应深度图的3D内容典型地是挑战性的任务。3D内容的每个帧可能需要处理大量数据的系统。在典型的3D视频应用中,例如由于传输带宽中的限制、存储限制和处理限制,多视图视频信号需要被高效地传送或存储。多视图视频编码(MVC)使用高级语法将H.264/高级视频编码(AVC)扩展,以方便多个视图的编码。该语法有助于图像处理器连续处理3D图像。

[0007] 虽然表面上被设计用于2D视频,但H.264/AVC还可以利用帧打包技术来用于传送立体内容。帧打包技术简单地呈现如下:在编码器侧,两个视图或画面被一般地下采样用于打包为一个单个视频帧,该一个单个视频帧然后被提供给H.264/AVC编码器以作为比特流输出;在解码器侧,比特流被解码并且恢复的帧然后被解包。解包允许从恢复的帧提取两个原始视图并且一般涉及上采样操作,以便将每个视图复原为原始大小,使得可以呈现所述视图用于显示。该途径能够被用于诸如具有多视图图像或具有深度信息等之类的两个或多个视图。

[0008] 帧打包可能依靠现有的与该帧及其视图相关联的辅助信息。补充增强信息(SEI)消息可能被用于传递某些帧打包信息。作为示例,在AVC的起草修改中,已经提出SEI消息被用于通知解码器打包画面的各种空间交织特性,包括由棋盘空间交织形成构成画面。通过利用SEI消息,可以直接使用AVC来编码立体视频图像的棋盘交织的画面。图26示出棋盘交织的已知示例。然而,迄今为止,SEI消息内容和其他高级语法的内容已被限于传递与已经经历帧打包的画面或视图有关的信息。

发明内容

[0009] 根据一般方面,编码包括被组合为单个画面的多个画面的视频画面。生成指示在所访问的视频画面中如何组合多个画面的信息。所生成的信息包括空间交织信息和采样信息。空间交织信息指示在形成单个画面中应用于多个画面的空间交织。采样信息指示与用于将多个画面的每个恢复到期望分辨率的上采样滤波器相关的一个或多个参数。与上采样滤波器相关的一个或多个参数包括滤波方向的指示。形成包括编码视频画面和生成的信息的比特流。生成的信息提供用于处理编码视频画面的信息。

[0010] 根据另一个一般方面,视频信号或视频结构包括编码画面部分和信令部分。编码画面部分包括视频画面的编码,所述视频画面包括被组合成单个画面的多个画面。信令部分包括指示在访问的视频画面中如何组合多个画面的生成信息的编码。生成信息包括空间交织信息和采样信息。空间交织信息指示在形成单个画面中应用于多个画面的空间交织。采样信息指示与用于将多个画面中的每个恢复到期望分辨率的上采样滤波器有关的一个或多个参数。与上采样滤波器有关的一个或多个参数包括滤波方向的指示。生成信息提供用于解码所编码的视频画面的信息。

[0011] 根据另一个一般方面,访问包括被组合成单个画面的多个画面的视频画面,视频画面是所接收的视频流的一部分。访问作为所接收的视频流的一部分的信息,访问的信息指示在所访问的视频画面中如何组合多个画面。访问的信息包括空间交织信息和采样信息。空间交织信息指示在形成单个画面中应用于多个画面的空间交织。采样信息指示与用于将多个画面的每个恢复到期望分辨率的上采样滤波器有关的一个或多个参数。与上采样滤波器有关的该一个或多个参数包括滤波方向的指示。视频画面被编码以提供多个画面的至少一个的解码表示。

[0012] 在下面的附图和描述中阐述一个或多个实现方式的细节。即使以一种具体方式被描述,应当清楚实现方式可以以各种方式被配置或体现。例如,一种实现方式可能被执行为方法,或体现为被配置为执行一组操作的装置,或体现为存储用于执行一组操作的指令的装置,或体现为信号。其他方面和特征将从下面结合附图和权利要求的、所考虑的详细描述变得明显。

附图说明

[0013] 图1是示出平铺在单个帧上的四个视图的示例的图。

[0014] 图2是示出在单个帧上翻转并平铺的四个视图的示例的图。

[0015] 图3示出可以应用本原理的视频编码器的实现方式的框图。

[0016] 图4示出可以应用本原理的视频解码器的实现方式的框图。

[0017] 图5A-5B是使用MPEG-4 AVC标准编码多个视图的画面的方法的实现方式的流程图。

[0018] 图6A-6B是使用MPEG-4 AVC标准解码多个视图画面的方法的实现方式的流程图。

[0019] 图7A-7B是使用MPEG-4 AVC标准编码多个视图和深度的画面的方法的实现方式的流程图。

[0020] 图8A-8B是使用MPEG-4 AVC标准解码多个视图和深度的画面的方法的实现方式的

流程图。

[0021] 图9是示出深度信号示例的图。

[0022] 图10是示出添加为平铺的深度信号的示例的图。

[0023] 图11是示出平铺在信号帧上的5个视图的示例的图。

[0024] 图12是可以应用本原理的示例性多视图视频编码 (MVC) 编码器的框图。

[0025] 图13是可以应用本原理的示例性多视图视频编码 (MVC) 解码器的框图。

[0026] 图14是使用MPEG-4 AVC标准的多视图视频编码 (MVC) 扩展为编码多个视图的画面作准备而处理该画面的方法的实现方式的流程图。

[0027] 图15A-15B是使用MPEG-4 AVC标准的多视图视频编码 (MVC) 扩展的编码多个视图的画面的方法的实现方式的流程图。

[0028] 图16是使用MPEG-4 AVC标准的多视图视频编码 (MVC) 扩展为解码多个视图的画面作准备而处理该画面的方法的实现方式的流程图。

[0029] 图17A-17B是使用MPEG-4 AVC标准的多视图视频编码 (MVC) 扩展的解码多个视图的画面的方法的实现方式的流程图。

[0030] 图18是使用MPEG-4 AVC标准的多视图视频编码 (MVC) 扩展为编码多个视图和深度的画面作准备而处理该画面的方法的实现方式的流程图。

[0031] 图19A-19B是使用MPEG-4 AVC标准的多视图视频编码 (MVC) 扩展的编码多个视图和深度的画面的方法的实现方式的流程图。

[0032] 图20是使用MPEG-4 AVC标准的多视图视频编码 (MVC) 扩展为解码多个视图和深度的画面作准备而处理该画面的方法的实现方式的流程图。

[0033] 图21A-21B是使用MPEG-4 AVC标准的多视图视频编码 (MVC) 扩展的解码多个视图和深度的画面的方法的实现方式的流程图。

[0034] 图22是示出像素级的平铺示例的图。

[0035] 图23示出可以应用本原理的视频处理设备的实现方式的框图。

[0036] 图24示出示例性3D视频系统的简化图。

[0037] 图25示出来自不同参考视图的图像的示例性左深度图和右深度图。

[0038] 图26示出使用棋盘交织将两个构成画面空间交织为单个画面或帧的示例性框图。

[0039] 图27示出两个构成画面的并排空间交织的示例性画面。

[0040] 图28示出两个构成画面的从顶部至底部空间交织的示例性画面。

[0041] 图29示出两个构成画面的逐行空间交织的示例性画面。

[0042] 图30示出两个构成画面的逐列空间交织的示例性画面。

[0043] 图31示出两个构成画面的并排空间交织的示例性画面,其中,左手画面被水平翻转。

[0044] 图32示出两个构成画面的从顶部至底部空间交织的示例性画面,其中,底画面被垂直翻转。

[0045] 图33示出示例性交织的画面或帧,其中构成画面代表层深度视频 (LDV) 格式。

[0046] 图34示出示例性交织画面或帧,其中构成画面代表2D加深度格式。

[0047] 图35-38示出使用用于帧打包信息的SEI消息来处理视频图像的编码和解码的不同实施例的示例性流程图。

[0048] 图39示出可以应用本原理的示例性视频传输系统。

[0049] 图40示出可以应用本原理的示例性视频接收系统。

[0050] 图41示出可以应用本原理的示例性视频处理设备。

[0051] 在此陈述的示例性实施例例示各种实施例,并且这样的示例性实施例并不被解释为以任何方式限制本公开的范围。

具体实施方式

[0052] 各种实现方式关注用于视频编码和解码中视图平铺的方法和装置。由此应当理解,本领域技术人员将能够设计出虽然未在这里被显式地描述或示出的、但体现了本原理并被包括在本原理的精神和范围内的各种布置。

[0053] 在这里所叙述的所有示例和条件性语言意图在于辅助读者理解本原理和由(多个)发明人为发展本领域而贡献的构思的教导的目的,并且应当被理解为不受这样的具体叙述的示例和条件的限制。

[0054] 此外,在这里叙述本原理的原理、方面和实施例的所有陈述,以及其具体示例,意图包括其结构性和功能性等效物两者。另外,这样的等效物意图包括当前已知的等效物以及在将来开发的等效物,即所开发的任何无论结构如何都执行相同的功能的元件。

[0055] 由此,例如,本领域技术人员将理解,在这里所呈现的框图代表例示性系统部件和/或电路的构思视图。类似地,将理解,任何流程表、流程图、状态转换图、伪码等代表可以在计算机可读介质中被实质性地代表、并且由计算机或处理器如此执行的各种处理,而无论这样的计算机或处理器是否被显式地示出。

[0056] 可以通过使用专用硬件以及能够与恰当的软件相关联而执行软件的硬件来提供图中所示的各种元件的功能。在由处理器提供时,可以由单个专用处理器、由单个共享处理器、或由多个单独的处理器(其中的一些可以被共享)来提供所述功能。此外,术语“处理器”或“控制器”的显式使用不应被理解为排他性地指代能够执行软件的硬件,而可以隐式地包括但不限于数字信号处理器(“DSP”)硬件、用于存储软件的只读存储器(“ROM”)、随机存取存储器(“RAM”)和非易失性存储装置。

[0057] 还可以在实现各种实施方式时包括常规的和/或定制的其他硬件。例如,图中所示的任何切换器都仅仅是概念性的。可以通过程序逻辑的操作、通过专用逻辑、通过程序控制和专用逻辑的交互、或甚至手动地执行它们的功能,如同从上下文中更具体地理解的,具体的技术可由实施者选择。

[0058] 在本文的权利要求中,被表达为用于执行所指定的功能的部件的任何元件意图包括执行该功能的任何方式,所述任何方式例如包括:a)执行该功能的电路元件的组合或b)因此包括固件、微代码等的任何形式的软件与用于执行该软件的恰当的电路进行组合以执行该功能。如同由这样的权利要求所限定的本原理存在于以下事实中:以权利要求所要求的方式将由各种所叙述的部件所提供的功能组合并放在一起。因而认为可以提供那些功能的任何部件与这里所示出的那些部件等效。

[0059] 本说明书中对本原理的“一个实施例”(或“一个实现方式”)或者“实施例”(或“实现方式”)的引用意味着在本原理的至少一个实施例中包括与该实施例相联系地描述的特定的特征、结构、特性等。因而,贯穿本说明书的多处出现的短语“在一个实施例中”或“在实

施例中”的出现不一定都指代相同的实施例。

[0060] 此外,应理解虽然在这里关于MPEG-4 AVC标准描述本原理的一个或多个实施例,但在本申请中描述的本原理并不单独地限于该标准,并且由此可以关于其他标准、建议和其扩展,特别是视频编码标准、建议和其扩展,包括MPEG-4 AVC标准的扩展,而利用本原理同时保持本申请原理的精神。

[0061] 另外,应理解虽然在这里关于MPEG-4 AVC标准的多视图视频编码扩展描述一个或多个其他实施例,但本原理并不单独地限于该扩展和/或该标准,并且由此可以关于与多视图视频编码有关的其他视频编码标准、建议和其扩展利用本原理,同时保持本申请原理的精神。多视图视频编码(MVC)是用于编码多视图序列的压缩框架。多视图视频编码(MVC)序列是从不同视点捕捉同一场景的两个或更多视频序列的集合。

[0062] 此外,还应理解虽然在此描述使用关于视频内容的深度信息的一个或多个其他实施例,但是本申请的原理不限于这样的实施例并且由此可以实现不使用深度信息的其他实施例,同时保持本原理的精神。

[0063] 进一步,如在这里所使用的,“高级语法”指代分层级驻留在宏块层之上的比特流中呈现的语法。例如,如在这里所使用的,高级语法可以指代、但不限于在码片首标级、补充增强信息(SEI)级、画面参数集(PPS)级、序列参数集(SPS)级和视图参数集(VPS)以及网络抽象层(NAL)单元首标级处的语法。

[0064] 在基于国际标准化组织/国际电工委员会(ISO/IEC)运动图像专家组-4(MPEG-4)第10部分高级视频编码(AVC)标准/国际电信联盟电信分部(ITU-T)H.264建议(在下文中为“MPEG-4AVC标准”)的多视图编码(MVC)的当前实现方式中,参考软件通过用单个编码器编码每个视图并且考虑交叉视图参考来实现多视图预测。通过编码器以其原始分辨率将每个视图编码为单独的比特流并且随后将所有比特流组合以形成然后被解码的单个比特流。每个视图产生单独的YUV解码输出。

[0065] 在图24中示意性呈现支持产生和使用3D图像的示例性视频系统。该系统的内容产生侧示出由包括但不限于立体相机、深度相机、同时操作的多个相机的各种部件进行的图像捕捉,以及2D图像到3D图像的转换。在图25中示出为同一场景的左视图和右视图而捕捉的深度图信息(例如,Z信息)的示例。这些途径中的每一个不仅捕捉视频图像内容,而且一些还生成与捕捉的视频图像相关联的某些深度信息。一旦被处理并且被编码,所有该信息可用于被分配、被传送、以及最终被呈现。还利用视频内容生成元数据用于在3D视频的随后呈现中使用。使用2D显示系统或3D显示器可以进行呈现。3D显示器可以从立体显示器变化到多视图3D显示器,如各图中所示。

[0066] 多视图预测的另一途径涉及将视图集编组为伪视图。在该途径的一个示例中,可以利用可能的下采样或其他操作在较大帧或超级帧上平铺来自全部M个视图(在同一时间被采样)的每N个视图的画面。转向图1,由参考标号100一般地指示单个帧上平铺的四个视图的示例。所有四个图像在它们正常的方位(orientation)。

[0067] 转向图2,由参考标号200一般地指示在单个帧上翻转和平铺的四个视图的示例。左上视图在其正常方位。右上视图被水平翻转。左下视图被垂直翻转。右下视图被水平且垂直翻转。由此,如果有四个视图,那么以类似平铺的超级帧来布置每个视图的画面。这造成具有高分辨率的单个未编码输入序列。

[0068] 可替代地,我们可以将图像下采样,以产生较小的分辨率。由此,我们创建多个序列,其每一个包括一起被平铺的不同视图。每个这样的序列然后形成伪视图,其中每个伪视图包括N个不同的平铺视图。图1示出一个伪视图,而图2示出另一个伪视图。然后可以使用诸如ISO/IEC MPEG-2标准和MPEG-4 AVC标准之类的现有视频编码标准来编码这些伪视图。

[0069] 用于多视图编码的又一途径简单地涉及使用新标准独立地编码不同的视图并且在解码之后,由播放器按所需平铺视图。

[0070] 另外,在另一途径中,视图还可以以逐像素方式被平铺。例如,在由四个视图组成的超级视图中,像素(x,y)可以来自视图0,而像素(x+1,y)可以来自视图1,像素(x,y+1)可以来自视图2,以及像素(x+1,y+1)可以来自视图3。

[0071] 很多显示器制造商使用这样的框架:在单个帧上布置或平铺不同的视图,然后从它们各自的位置提取该视图并且将它们呈现。在这样的情形中,不存在用以确定该比特流是否具有这样的属性的标准方式。由此,如果系统使用在大型帧中平铺不同视图画面的方法,那么提取不同视图的方法是专用的。

[0072] 然而,不存在用以确定比特流是否具有这样的属性的标准方式。提出高级语法以方便呈现器或播放器提取这样的信息,以有助于显示或其他后处理。子画面还能够具有不同的分辨率并且可能需要某些上采样以逐渐呈现视图。用户也可能希望具有在高级语法中指示上采样的方法。此外,用以改变深度焦点的参数也可以被传送。

[0073] 在一实施例中,提出用于在MPEG-4 AVC标准相容比特流中用信号通知多视图信息的新的补充增强信息(SEI)消息,其中每个画面包括属于不同视图的子画面。该实施例意图是例如易于并且便于在可以使用这样的框架的三维(3D)监视器上显示多视图视频流。该构思可以被扩展至使用高级语法发信号通知这样的消息的其他视频编码标准和推荐。

[0074] 此外,在一实施例中,提出在视图被发送至多视图视频编码器和/或解码器之前如何布置它们的信令方法。有利地,该实施例可能导致简化的多视图编码的实现方式并且可能有益于编码效率。通过普通多视图视频编码器和/或解码器(例如,按照当前的基于MPEG-4 AVC标准的多视图视频编码的实现方式)某些视图可以被一起放置并且形成伪视图或超级视图并且然后平铺的超级视图被视作正常视图。在多视图视频编码的序列参数集(SPS)扩展中提出表1中示出的新的标志以便发信号通知伪视图技术的使用。该实施例意图是易于并且便于在可以使用这样的框架的3D监视器上显示多视图视频流。

[0075] 用于多视图编码的另一途径涉及利用可能的下采样操作在较大帧或超级帧上平铺来自每个视图(在同一时间被采样)的画面。转向图1,由参考标号100一般地指示在单个帧上平铺的四个视图的示例。转向图2,由参考标号200一般地指示在单个帧上翻转并平铺的四个视图的示例。由此,如果存在四个视图,那么以类似平铺的超级帧来布置来自每个视图的画面。这造成具有高分辨率的单个未编码的输入序列。该信号可以然后使用诸如ISO/IEC MPEG 2标准和MPEG-4 AVC标准之类的现有的视频编码标准来编码。

[0076] 转向图3,由参考标号300一般地指示根据MPEG-4 AVC标准、能够执行视频编码的视频编码器。

[0077] 视频编码器300包括帧排序缓冲器310,其具有与组合器385的非反相输入端信号通信的输出端。组合器385的输出端与变换器和量化器325的第一输入端信号通信地连接。变换器和量化器325的输出端与熵编码器345的第一输入端和逆变换器和逆量化器350的第

一输入端信号通信地连接。熵编码器345的输出端与组合器390的第一非反相输入端信号通信地连接。组合器390的输出端与输出缓冲器335的第一输入端信号通信地连接。

[0078] 编码器控制器305的第一输出端与帧排序缓冲器310的第二输入端、逆变换和逆量化器350的第二输入端、画面类型判定模块315的输入端、宏块类型(MB-类型)判定模块320的输入端、帧内预测模块360的第二输入端、去块滤波器365的第二输入端、运动补偿器370的第一输入端、运动估计器375的第一输入端,以及参考画面缓冲器380的第二输入端信号通信地连接。

[0079] 编码器控制器305的第二输出端与补充增强信息(SEI)插入器330的第一输入端、变换器和量化器325的第二输入端、熵编码器345的第二输入端、输出缓冲器335的第二输入端,以及序列参数集(SPS)和画面参数集(PPS)插入器340的输入端信号通信地连接。

[0080] 画面类型判定模块315的第一输出端与帧排序缓冲器310的第三输入端信号通信地连接。画面类型判定模块315的第二输出端与宏块类型判定模块320的第二输入端信号通信地连接。

[0081] 序列参数集(SPS)和画面参数集(PPS)插入器340的输出端与组合器390的第三非反相输入端信号通信地连接。SEI插入器330的输出端与组合器390的第二非反相输入端信号通信地连接。

[0082] 逆量化器和逆变换器350的输出端与组合器319的第一非反相输入端信号通信地连接。组合器319的输出端与帧内预测模块360的第一输入端和去块滤波器365的第一输入端信号通信地连接。去块滤波器365的输出端与参考画面缓冲器380的第一输入端信号通信地连接。参考画面缓冲器380的输出端与运动估计器375的第二输入端和运动补偿器370的第一输入端信号通信地连接。运动估计器375的第一输出端与运动补偿器370的第二输入端信号通信地连接。运动估计器375的第二输出端与熵编码器345的第三输入端信号通信地连接。

[0083] 运动补偿器370的输出端与开关397的第一输入端信号通信地连接。帧内预测模块360的输出端与开关397的第二输入端信号通信地连接。宏块类型判定模块320的输出端与开关397的第三输入端信号通信地连接,以对开关397提供控制输入端。开关397的第三输入端确定开关的“数据”输入端(与控制输入端比较,即第三输入端)是由运动补偿器370提供还是由帧内预测模块360提供。开关397的输出端与组合器319的第二非反相输入端以及组合器385的反相输入端信号通信地连接。

[0084] 帧排序缓冲器310和编码器控制器305的输入端可用作编码器300的输入端,用于接收输入画面301。此外,补充增强信息(SEI)插入器330的输入端可用作编码器300的输入端,用于接收元数据。输出缓冲器335的输出端可用作编码器300的输出端,用于输出比特流。

[0085] 转向图4,由参考标号400一般地指示能够根据MPEG-4 AVC标准执行视频解码的视频解码器。

[0086] 视频解码器400包括输入缓冲器410,其具有与熵解码器445的第一输入端信号通信地连接的输出端。熵解码器445的第一输出端与逆变换器和逆量化器450的第一输入端信号通信地连接。逆变换器和逆量化器450的输出端与组合器425的第二非反相输入端信号通信地连接。组合器425的输出端与去块滤波器465的第二输入端和帧内预测模块460的第一

输入端信号通信地连接。去块滤波器465的第二输出端与参考画面缓冲器480的第一输入端信号通信地连接。参考画面缓冲器480的输出端与运动补偿器470的第二输入端信号通信地连接。

[0087] 熵解码器445的第二输出端与运动补偿器470的第三输入端和去块滤波器465的第一输入端信号通信地连接。熵解码器445的第三输出端与解码器控制器405的输入端信号通信地连接。解码器控制器405的第一输出端与熵解码器445的第二输入端信号通信地连接。解码器控制器405的第二输出端与逆变换器和逆量化器450的第二输入端信号通信地连接。解码器控制器405的第三输出端与去块滤波器465的第三输入端信号通信地连接。解码器控制器405的第四输出端与帧内预测模块460的第二输入端、运动补偿器470的第一输入端、以及参考画面缓冲器480的第二输入端信号通信地连接。

[0088] 运动补偿器470的输出端与开关497的第一输入端信号通信地连接。帧内预测模块460的输出端与开关497的第二输入端信号通信地连接。开关497的输出端与组合器425的第一非反相输入端信号通信地连接。

[0089] 输入缓冲器410的输入端可用作解码器400的输入端,用于接收输入比特流。去块滤波器465的第一输出端可用作解码器400的输出端,用于对输出画面进行输出。

[0090] 转向图5,由参考标号500一般地指示使用MPEG-4 AVC标准编码多个视图的画面的示例性方法。

[0091] 方法500包括将控制传递给功能块504的起始块502。功能块504在特定时间实例(time instance)将每个视图布置为平铺格式的子画面,并且将控制传递给功能块506。功能块506设置语法元素num_coded_views_minus1,并将控制传递给功能块508。功能块508设置语法元素org_pic_width_in_mbs_minus1以及org_pic_height_in_mbs_minus1,并且将控制传递给功能块510。功能块510设置变量i等于零并且将控制传递给判定块512。判定块512确定变量i是否小于视图的数量。如果是这样,则将控制传递给功能块514。否则,将控制传递给功能块524。

[0092] 功能块514设置语法元素view_id[i],并且将控制传递给功能块516。功能块516设置语法元素num_parts[view_id[i]],并且将控制传递给功能块518。功能块518设置变量j等于零,并且将控制传递给判定块520。判定块520确定变量j的当前值是否小于语法元素num_parts[view_id[i]]的当前值。如果是这样,则将控制传递给功能块522。否则,将控制传递给功能块528。

[0093] 功能块522设置以下语法元素,递增变量j并然后将控制返回给判定块520:depth_flag[view_id[i]][j];flip_dir[view_id[i]][j];loc_left_offset[view_id[i]][j];loc_top_offset[view_id[i]][j];frame_crop_left_offset[view_id[i]][j];frame_crop_right_offset[view_id[i]][j];frame_crop_top_offset[view_id[i]][j];以及frame_crop_bottom_offset[view_id[i]][j]。

[0094] 功能块528设置语法元素upsample_view_flag[view_id[i]],并且将控制传递至判定块530。判定块530确定语法元素upsample_view_flag[view_id[i]]的当前值是否等于一。如果是这样,则将控制传递至功能块532。否则,将控制传递至判定块534。

[0095] 功能块532设置语法元素upsample_filter[view_id[i]],并且将控制传递至判定块534。

[0096] 判定块534确定语法元素upsample_filter[view_id[i]]的当前值是否等于三。如果是这样,则将控制传递至功能块536。否则,将控制传递至功能块540。

[0097] 功能块536设置以下语法元素并且将控制传递至功能块538:vert_dim[view_id[i]];hor_dim[view_id[i]];以及quantizer[view_id[i]]。

[0098] 功能块538设置每个YUV分量的滤波器系数,并且将控制传递至功能块540。

[0099] 功能块540递增变量i,并且将控制返回给判定块512。

[0100] 功能块524将这些语法元素写入序列参数集(SPS)、画面参数集(PPS)、补充增强信息(SEI)消息、网络抽象层(NAL)单元首标以及码片首标中的至少一个,并且将控制传递至功能块526。功能块526使用MPEG-4 AVC标准或其他单个画面编解码器来编码每个画面,并且将控制传递至结束块599。

[0101] 转向图6,由参考标号600一般地指示使用MPEG-4 AVC标准解码多个视图的画面的示例性方法。

[0102] 方法600包括将控制传递至功能块604的起始块602。功能块604从序列参数集(SPS)、画面参数集(PPS)、补充增强信息(SEI)消息、网络抽象层(NAL)单元首标以及码片首标中的至少一个分析以下语法元素,并且将控制传递至功能块606。功能块606分析语法元素num_coded_views_minus1,并且将控制传递至功能块608。功能块608分析语法元素

[0103] org_pic_width_in_mbs_minus1以及org_pic_height_in_mbs_minus1,并且将控制传递至功能块610。功能块610设置变量i等于零,并且将控制传递至判定块612。判定块612确定变量i是否小于视图的数量。如果是这样,则将控制传递至功能块614。否则,将控制传递至功能块624。

[0104] 功能块614分析语法元素view_id[i],并且将控制传递至功能块616。功能块616分析语法元素num_parts_minus1[view_id[i]],并且将控制传递至功能块618。功能块618设置变量j等于零,并且将控制传递至判定块620。判定块620确定变量j的当前值是否小于语法元素num_parts[view_id[i]]的当前值。如果是这样,则将控制传递至功能块622。否则,将控制传递至功能块628。

[0105] 功能块622分析以下语法元素,递增变量j,并且然后将控制返回给判定块620:depth_flag[view_id[i]][j];flip_dir[view_id[i]][j];loc_left_offset[view_id[i]][j];loc_top_offset[view_id[i]][j];frame_crop_left_offset[view_id[i]][j];frame_crop_right_offset[view_id[i]][j];frame_crop_top_offset[view_id[i]][j];以及frame_crop_bottom_offset[view_id[i]][j]。

[0106] 功能块628分析语法元素upsample_view_flag[view_id[i]],并且将控制传递至判定块630。判定块630确定语法元素upsample_view_flag[view_id[i]]的当前值是否等于一。如果是这样,则将控制传递至功能块632。否则,将控制传递至判定块634。

[0107] 功能块632分析语法元素upsample_filter[view_id[i]],并且将控制传递至判定块634。

[0108] 判定块634确定语法元素upsample_filter[view_id[i]]的当前值是否等于三。如果是这样的话,然后将控制传递至功能块636。否则,将控制传递至功能块640。

[0109] 功能块636分析以下语法元素并且将控制传递至功能块638:vert_dim[view_id[i]];hor_dim[view_id[i]];以及quantizer[view_id[i]]。

- [0110] 功能块638分析每个YUV分量的滤波器系数,并且将控制传递至功能块640。
- [0111] 功能块640递增变量*i*并且将控制返回给判定块612。
- [0112] 功能块624使用MPEG-4 AVC标准或其他单个视图编解码器来解码每个画面,并且将控制传递至功能块626。功能块626使用高级语法将每个视图从画面分离,并且将控制传递至结束块699。
- [0113] 转向图7,由参考标号700一般地指示使用MPEG-4 AVC标准来编码多个视图和深度的画面的示例性方法。
- [0114] 方法700包括将控制传递至功能块704的起始块702。功能块704在特定时间实例将每个视图和对应的深度布置为平铺格式的子画面,并且将控制传递给功能块706。功能块706设置语法元素num_coded_views_minus1,并且将控制传递给功能块708。功能块708设置语法元素org_pic_width_in_mbs_minus1以及org_pic_height_in_mbs_minus1,并且将控制传递给功能块710。功能块710设置变量*i*等于零,并且将控制传递给判定块712。判定块712确定变量*i*是否小于视图的数量。如果是这样,则将控制传递给功能块714。否则,将控制传递给功能块724。
- [0115] 功能块714设置语法元素view_id[*i*],并且将控制传递给功能块716。功能块716设置语法元素num_parts[view_id[*i*]],并且将控制传递给功能块718。功能块718设置变量*j*等于零,并且将控制传递给判定块720。判定块720确定变量*j*的当前值是否小于语法元素num_parts[view_id[*i*]]的当前值。如果是这样的话,则将控制传递给功能块722。否则将控制传递给功能块728。
- [0116] 功能块722设置以下语法元素,递增变量*j*,并且然后将控制返回给判定块720: depth_flag[view_id[*i*]][*j*];flip_dir[view_id[*i*]][*j*];loc_left_offset[view_id[*i*]][*j*];loc_top_offset[view_id[*i*]][*j*];frame_crop_left_offset[view_id[*i*]][*j*];frame_crop_right_offset[view_id[*i*]][*j*];frame_crop_top_offset[view_id[*i*]][*j*];以及frame_crop_bottom_offset[view_id[*i*]][*j*]。
- [0117] 功能块728设置语法元素upsample_view_flag[view_id[*i*]],并且将控制传递至判定块730。判定块730确定语法元素upsample_view_flag[view_id[*i*]]的当前值是否等于一。如果是这样的话,则将控制传递至功能块732。否则,将控制传递至判定块734。
- [0118] 功能块732设置语法元素upsample_filter[view_id[*i*]],并且将控制传递至判定块734。
- [0119] 判定块734确定语法元素upsample_filter[view_id[*i*]]的当前值是否等于三。如果是这样的话,则将控制传递至功能块736。否则,将控制传递至功能块740。
- [0120] 功能块736设置以下语法元素并将控制传递至功能块738:vert_dim[view_id[*i*]];hor_dim[view_id[*i*]];以及quantizer[view_id[*i*]]。
- [0121] 功能块738设置每个YUV分量的滤波器系数,并且将控制传递至功能块740。
- [0122] 功能块740递增变量*i*,并且将控制返回给判定块712。
- [0123] 功能块724将这些语法元素写入以下中的至少一个:序列参数集(SPS)、画面参数集(PPS)、补充增强信息(SEI)消息、网络抽象层(NAL)单元首标以及码片首标,并且将控制传递至功能块726。功能块726使用MPEG-4AVC标准或其他单个视图编解码器编码每个画面,并且将控制传递到结束块799。

[0124] 转向图8,由参考标号800一般地指示使用MPEG-4 AVC标准解码多个视图和深度的画面的示例性方法。

[0125] 方法800包括将控制传递至功能块804的起始块802。功能块分析来自序列参数集 (SPS)、画面参数集 (PPS)、补充增强信息 (SEI) 消息、网络抽象层 (NAL) 单元首标以及码片首标中的至少一个的以下语法元素,并且将控制传递至功能块806。功能块806分析语法元素 `num_coded_views_minus1`,并且将控制传递至功能块808。功能块808分析语法元素 `org_pic_width_in_mbs_minus1`以及`org_pic_height_in_mbs_minus1`,并且将控制传递至功能块810。功能块810设置变量*i*等于零,并且将控制传递至判定块812。判定块812确定变量*i*是否小于视图的数量。如果是这样的话,则将控制传递至功能块814。否则,将控制传递至功能块824。

[0126] 功能块814分析语法元素 `view_id[i]`,并且将控制传递到功能块816。功能块816分析语法元素 `num_parts_minus1[view_id[i]]`,并且将控制传递至功能块818。功能块818设置变量*j*等于零,并且将控制传递至判定块820。判定块820判定变量*j*的当前值是否小于语法元素 `num_parts[view_id[i]]`的当前值。如果是这样的话,则将控制传递至功能块822。否则,将控制传递至功能块828。

[0127] 功能块822分析以下语法元素,递增变量*j*,并且然后将控制返回给判定块820:
`depth_flag[view_id[i]][j]`;
`flip_dir[view_id[i]][j]`;
`loc_left_offset[view_id[i]][j]`;
`loc_top_offset[view_id[i]][j]`;
`frame_crop_left_offset[view_id[i]][j]`;
`frame_crop_right_offset[view_id[i]][j]`;
`frame_crop_top_offset[view_id[i]][j]`;
以及
`frame_crop_bottom_offset[view_id[i]][j]`。

[0128] 功能块828分析语法元素 `upsample_view_flag[view_id[i]]`,并且将控制传递至判定块830。判定块830确定语法元素 `upsample_view_flag[view_id[i]]`的当前值是否等于一。如果是这样的话,则将控制传递至功能块832。否则,将控制传递至判定块834。

[0129] 功能块832分析语法元素 `upsample_filter[view_id[i]]`,并且将控制传递至判定块834。

[0130] 判定块834确定语法元素 `upsample_filter[view_id[i]]`的当前值是否等于三。如果是这样的话,则将控制传递至功能块836。否则,将控制传递至功能块840。

[0131] 功能块836分析以下语法元素,并且将控制传递至功能块838:
`vert_dim[view_id[i]]`;
`hor_dim[view_id[i]]`;
以及
`quantizer[view_id[i]]`。

[0132] 功能块838分析每个YUV分量的滤波器系数,并且将控制传递至功能块840。

[0133] 功能块840递增变量*i*,并且将控制返回给判定块812。

[0134] 功能块824使用MPEG-4 AVC标准或其他单个视图编解码器来解码每个画面,并且将控制传递至功能块826。功能块826使用高级语法将每个视图和对应的深度从画面分离,并且将控制传递至功能块827。功能块827潜在地使用提取的视图和深度信号执行视图合成,并且将控制传递至结束块899。

[0135] 关于图7和图8中使用的深度,图9示出深度信号的示例900,其中提供深度作为图像的每个对应位置的像素值(未示出)。另外,图10示出在平铺1000中包括的两个深度信号的示例。平铺1000的右上部分是具有对应平铺1000的左上的图像的深度值的深度信号。平铺1000的右下部分是具有对应平铺1000的左下的图像的深度值的深度信号。

[0136] 转向图11,由参考标号1100一般地指示平铺在单个帧上的5个视图的示例。顶部的四个视图在正常方位。第五个视图也在正常方位,但是沿着平铺1100的底部被分为两个部分。第五个视图的左侧部分示出第五个视图的“顶部”,并且第五个视图的右侧部分示出第五个视图的“底部”。

[0137] 转向图12,由参考标号1200一般地指示示例性多视图视频编码(MVC)编码器。编码器1200包括组合器1205,其具有与变换器1210的输入端信号通信地连接的输出端。变换器1210的输出端与量化器1215的输入端信号通信地连接。量化器1215的输出端与熵编码器1220的输入端以及逆量化器1225的输入端信号通信地连接。逆量化器1225的输出端与逆变换器1230的输入端信号通信地连接。逆变换器1230的输出端与组合器1235的第一非反相输入端信号通信地连接。组合器1235的输出端与帧内预测器1245的输入端以及去块滤波器1250的输入端信号通信地连接。去块滤波器1250的输出端与参考画面存储器1255(用于视图i)的输入端信号通信地连接。参考画面存储器1255的输出端与运动补偿器1275的第一输入端以及运动估计器1280的第一输入端信号通信地连接。运动估计器1280的输出端与运动补偿器1275的第二输入端信号通信地连接。

[0138] 参考画面存储器1260(用于其他视图)的输出端与视差估计器1270的第一输入端以及视差补偿器1265的第一输入端信号通信地连接。视差估计器1270的输出端与视差补偿器1265的第二输入端信号通信地连接。

[0139] 熵解码器1220的输出端可用作编码器1200的输出端。组合器1205的非反相输入端可用作编码器1200的输入端,并且其与视差估计器1270的第二输入端以及运动估计器1280的第二输入端信号通信地连接。开关1285的输出端与组合器1235的第二非反相输入端以及组合器1205的反相输入端信号通信地连接。开关1285包括与运动补偿器1275的输出端信号通信地连接的第一输入端、与视差补偿器1265的输出端信号通信地连接的第二输入端,以及与帧内预测器1245的输出端信号通信地连接的第三输入端。

[0140] 模式判定模块1240具有连接至开关1285的输出端,用于控制由开关1285选择哪个输入端。

[0141] 转向图13,由参考标号1300一般地指示示例性多视图视频编码(MVC)解码器。解码器1300包括熵解码器1305,其具有与逆量化器1310的输入端信号通信地连接的输出端。逆量化器的输出端与逆变换器1315的输入端信号通信地连接。逆变换器1315的输出端与组合器1320的第一非反相输入端信号通信地连接。组合器1320的输出端与去块滤波器1325的输入端以及帧内预测器1330的输入端信号通信地连接。去块滤波器1325的输出端与参考画面存储器1340(用于视图i)的输入端信号通信地连接。参考画面存储器1340的输出端与运动补偿器1335的第一输入端信号通信地连接。

[0142] 参考画面存储器1345(用于其他视图)的输出端与视差补偿器1350的第一输入端信号通信地连接。

[0143] 熵编码器1305的输入端可用作解码器1300的输入端,用于接收残差比特流。此外,模式模块1360的输入端还用作解码器1300的输入端,用于接收控制语法以控制由开关1355选择哪个输入端。另外,运动补偿器1335的第二输入端可用作解码器1300的输入端,用于接收运动矢量。此外,视差补偿器1350的第二输入端可用作解码器1300的输入端,用于接收视差矢量。

[0144] 开关1355的输出端与组合器1320的第二非反相输入端信号通信地连接。开关1355的第一输入端与视差补偿器1350的输出端信号通信地连接。开关1355的第二输入端与运动补偿器1335的输出端信号通信地连接。开关1355的第三输入端与帧内预测器1330的输出端信号通信地连接。模式模块1360的输出端与开关1355信号通信地连接,用于控制由开关1355选择哪个开关。去块滤波器1325的输出端可用作解码器1300的输出端。

[0145] 转向图14,由参考标号1400一般地指示使用MPEG-4 AVC标准的多视图视频编码(MVC)扩展的、为编码多个视图的画面作准备而处理该画面的示例性方法。

[0146] 方法1400包括将控制传递给功能块1410的起始块1405。功能块1410在特定时间实例将所有M个视图中的每N个视图布置为平铺格式的超级画面,并且将控制传递给功能块1415。功能块1415设置语法元素num_coded_views_minus1,并且将控制传至功能块1420。功能块1420对所有(num_coded_views_minus1+1)个视图设置语法元素view_id[i],并且将控制传递至功能块1425。功能块1425设置锚点(anchor)画面的视图间参考相关性信息,并且将控制传递至功能块1430。功能块1430设置非锚点画面的视图间参考相关性信息,并且将控制传递至功能块1435。功能块1435设置语法元素pseudo_view_present_flag,并且将控制传递至判定块1440。判定块1440确定语法元素pseudo_view_present_flag的当前值是否等于真(true)。如果是这样的话,将控制传递至功能块1445。否则,将控制传递至结束块1499。

[0147] 功能块1445设置以下语法元素,并且将控制传递至功能块1450:tiling_mode; org_pic_width_in_mbs_minus1;以及org_pic_height_in_mbs_minus1。功能块1450调用每个编码视图的语法元素pseudo_view_info(view_id),并且将控制传递至结束块1499。

[0148] 转向图15,由参考标号1500一般地指示使用MPEG-4 AVC标准的多视图视频编码(MVC)扩展的、编码多个视图的画面的示例性方法。

[0149] 方法1500包括起始块1502,其具有输入参数pseudo_view_id,并且将控制传递至功能块1504。功能块1504设置语法元素num_sub_views_minus1,并且将控制传递至功能块1506。功能块1506设置变量i等于零,并且将控制传递至判定块1508。判定块1508确定变量i是否小于sub_views的数量。如果是这样的话,则将控制传递至功能块1510。否则,将控制传递至功能块1520。

[0150] 功能块1510设置语法元素sub_view_id[i],并且将控制传递至功能块1512。功能块1512设置语法元素num_parts_minus1[sub_view_id[i]],并且将控制传递至功能块1514。功能块1514设置变量j等于零,并且将控制传递至判定块1516。判定块1516确定变量j是否小于语法元素num_parts_minus1[sub_view_id[i]]。如果是这样的话,则将控制传递至功能块1518。否则将控制传递至判定块1522。

[0151] 功能块1518设置以下语法元素,递增变量j,并且将控制返回给判定块1516:loc_left_offset[sub_view_id[i]][j];loc_top_offset[sub_view_id[i]][j];frame_crop_left_offset[sub_view_id[i]][j];frame_crop_right_offset[sub_view_id[i]][j];frame_crop_top_offset[sub_view_id[i]][j];以及frame_crop_bottom_offset[sub_view_id[i]][j]。

[0152] 功能块1520使用多视图视频编码(MVC)编码当前视图的当前画面,并且将控制传递至结束块1599。

[0153] 判定块1522确定语法元素tiling_mode是否等于零。如果这样的话,则将控制传递至功能块1524。否则,将控制传递至功能块1538。

[0154] 功能块1524设置语法元素flip_dir[sub_view_id[i]]以及语法元素upsample_view_flag[sub_view_id[i]],并且将控制传递至判定块1526。判定块1526确定语法元素upsample_view_flag[sub_view_id[i]]的当前值是否等于一。如果这样的话,则将控制传递至功能块1528。否则,将控制传递至判定块1530。

[0155] 功能块1528设置语法元素upsample_filter[sub_view_id[i]],并且将控制传递至判定块1530。判定块1530确定语法元素upsample_filter[sub_view_id[i]]的值是否等于三。如果是这样的话,则将控制传递至功能块1532。否则,将控制传递至功能块1536。

[0156] 功能块1532设置以下语法元素,并且将控制传递至功能块1534:vert_dim[sub_view_id[i]];hor_dim[sub_view_id[i]];以及quantizer[sub_view_id[i]]。功能块1534设置每个YUV分量的滤波器系数,并且将控制传递至功能块1536。

[0157] 功能块1536递增变量i,并且将控制返回给判定块1508。

[0158] 功能块1538设置语法元素pixel_dist_x[sub_view_id[i]]以及语法元素flip_dist_y[sub_view_id[i]],并且将控制传递至功能块1540。功能块1540设置变量j等于零,并且将控制传递至判定块1542。判定块1542确定变量j的当前值是否小于语法元素num_parts[sub_view_id[i]]的当前值。如果是这样的话,则将控制传递至功能块1544。否则,将控制传递至功能块1536。

[0159] 功能块1544设置语法元素num_pixel_tiling_filter__coeffs_minus1[sub_view_id[i]],并且将控制传递至功能块1546。功能块1546设置所有像素平铺滤波器的系数,并且将控制传递至功能块1536。

[0160] 转向图16,由参考标号1600一般地指示使用MPEG-4 AVC标准的多视图视频编码(MVC)扩展的、为解码多个视图的画面做准备而处理该画面的示例性方法。

[0161] 方法1600包括将控制传递至功能块1615的起始块1605。功能块1615分析语法元素num_coded_views_minus1,并且将控制传递至功能块1620。功能块1620分析所有(num_coded_views_minus1+1)个视图的语法元素view_id[i],并且将控制传递至功能块1625。功能块1625分析锚点画面的视图间参考相关性信息,并且将控制传递至功能块1630。功能块1630分析非锚点画面的视图间参考相关性信息,并且将控制传递至功能块1635。功能块1635分析语法元素pseudo_view_present_flag,并且将控制传递至判定块1640。判定块1640确定语法元素pseudo_view_present_flag的当前值是否等于真。如果是这样的话,则将控制传递至功能块1645。否则将控制传递至结束块1699。

[0162] 功能块1645分析以下语法元素,并且将控制传递至功能块1650:tiling_mode;org_pic_width_in_mbs_minus1;以及org_pic_height_in_mbs_minus1。功能块1650调用每个编码视图的语法元素pseudo_view_info(view_id),并且将控制传递至结束块1699。

[0163] 转向图17,由参考标号1700一般地指示使用MPEG-4 AVC标准的多视图视频编码(MVC)扩展的、解码多个视图的画面的示例性方法。

[0164] 方法1700包括起始块1702,其以输入参数pseudo_view_id开始,并且将控制传递至功能块1704。功能块1704分析语法元素num_sub_views_minus1,并且将控制传递至功能块1706。功能块1706设置变量i等于零,并且将控制传递至判定块1708。判定块1708确定变

量*i*是否小于sub_views的数量。如果是这样的话,则将控制传递至功能块1710。否则,将控制传递至功能块1720。

[0165] 功能块1710分析语法元素sub_view_id[*i*],并且将控制传递至功能块1712。功能块1712分析语法元素num_parts_minus1[sub_view_id[*i*]],并且将控制传递至功能块1714。功能块1714设置变量*j*等于零,并且将控制传递至判定块1716。判定块1716确定变量*j*是否小于语法元素num_parts_minus1[sub_view_id[*i*]]。如果是这样的话,则将控制传递至功能块1718。否则,将控制传递至判定块1722。

[0166] 功能块1718设置以下元素,递增变量*j*,并且将控制返回给判定块1716:loc_left_offset[sub_view_id[*i*]][*j*];loc_top_offset[sub_view_id[*i*]][*j*];frame_crop_left_offset[sub_view_id[*i*]][*j*];frame_crop_right_offset[sub_view_id[*i*]][*j*];frame_crop_top_offset[sub_view_id[*i*]][*j*];以及frame_crop_bottom_offset[sub_view_id[*i*]][*j*]。

[0167] 功能块1720使用多视图视频编码(MVC)来解码当前视图的当前画面,并且将控制传递至功能块1721。功能块1721使用高级语法将每个视图从画面分离,并且将控制传递至结束块1799。

[0168] 使用比特流中指示的高级语法来完成将每个视图从解码画面的分离。该高级语法可以指示:画面中呈现的视图的精确位置和可能的方位(以及可能的对应深度)。

[0169] 判定块1722确定语法元素tiling_mode是否等于零。如果是这样的话,则将控制传递至功能块1724。否则,将控制传递至功能块1738。

[0170] 功能块1724分析语法元素flip_dir[sub_view_id[*i*]]以及语法元素upsample_view_flag[sub_view_id[*i*]],并且将控制传递至判定块1726。判定块1726确定语法元素upsample_view_flag[sub_view_id[*i*]]的当前值是否等于一。如果是这样的话,则将控制传递至功能块1728。否则,将控制传递至判定块1730。

[0171] 功能块1728分析语法元素upsample_filter[sub_view_id[*i*]],并且将控制传递至判定块1730。判定块1730确定语法元素upsample_filter[sub_view_id[*i*]]的值是否等于三。如果是这样的话,则将控制传递至功能块1732。否则,将控制传递至功能块1736。

[0172] 功能块1732分析以下语法元素,并且将控制传递至功能块1734:vert_dim[sub_view_id[*i*]];hor_dim[sub_view_id[*i*]];以及quantizer[sub_view_id[*i*]]。功能块1734分析每个YUV分量的滤波器系数,并且将控制传递至功能块1736。

[0173] 功能块1736递增变量*i*,并且将控制传递至判定块1708。

[0174] 功能块1738分析语法元素pixel_dist_x[sub_view_id[*i*]]以及语法元素flip_dist_y[sub_view_id[*i*]],并且将控制传递至功能块1740。功能块1740设置变量*j*等于零,并且将控制传递至判定块1742。判定块1742确定变量*j*的当前值是否小于语法元素num_parts[sub_view_id[*i*]]的当前值。如果是这样的话,则将控制传递至功能块1744。否则,将控制传递至功能块1736。

[0175] 功能块1744分析语法元素nurn_pixel_tiling_filter_coeffs_minus1[sub_view_id[*i*]],并且将控制传递至功能块1746。功能块1746分析所有像素平铺滤波器的系数,并且将控制传递至功能块1736。

[0176] 转向图18,由参考标号1800一般地指示使用MPEG-4 AVC标准的多视图视频编码

(MVC) 扩展的、为编码多个视图的画面做准备而处理该画面的示例性方法。

[0177] 方法1800包括将控制传递至功能块1810的起始块1805。功能块1810在特定时间实例将全部M个视图中的每N个视图和深度图布置为平铺格式的超级画面,并且将控制传递至功能块1815。功能块1815设置语法元素num_coded_views_minus1,并且将控制传递至功能块1820。功能块1820设置对应view_id[i]的所有(num_coded_views_minus1+1)个深度的语法元素view_id[i],并且将控制传递至功能块1825。功能块1825设置锚点深度画面的视图间参考相关性信息,并且将控制传递至功能块1830。功能块1830设置非锚点深度画面的视图间参考相关性信息,并且将控制传递至功能块1835。功能块1835设置语法元素pseudo_view_present_flag,并且将控制传递至判定块1840。判定块1840确定语法元素pseudo_view_present_flag的当前值是否等于真。如果这样的话,则将控制传递至功能块1845。否则,将控制传递至结束块1899。

[0178] 功能块1845设置以下语法元素,并且将控制传递至功能块1850:tiling mode; org_pic_width_in_mbs_minus1;以及org_pic_height_in_mbs_minus1。功能块1850调用每个编码视图的语法元素pseudo_view_info(view_id),并且将控制传递至结束块1899。

[0179] 转向图19,由参考标号1900一般地指示使用MPEG-4 AVC标准的多视图视频编码(MVC)扩展的、编码多个视图和深度的画面的示例性方法。

[0180] 方法1900包括将控制传递至功能块1904的起始块1902。功能块1904设置语法元素num_sub_views_minus1,并且将控制传递至功能块1906。功能块1906设置变量i等于零,并且将控制传递至判定块1908。判定块1908确定变量i是否小于sub_views的数量。如果是这样的话,则将控制传递至功能块1910。否则,将控制传递至功能块1920。

[0181] 功能块1910设置语法元素sub_view_id[i],并且将控制传递至功能块1912。功能块1912设置语法元素num_parts_minus1[sub_view_id[i]],并且将控制传递至功能块1914。功能块1914设置变量j等于零,并且将控制传递至判定块1916。判定块1916确定变量j是否小于语法元素num_parts_minus1[sub_view_id[i]]。如果是这样的,则将控制传递至功能块1918。否则,将控制传递至判定块1922。

[0182] 功能块1918设置以下语法元素,递增变量j,并且将控制返回给判定块1916:loc_left_offset[sub_view_id[i]][j];loc_top_offset[sub_view_id[i]][j];frame_crop_left_offset[sub_view_id[i]][j];frame_crop_right_offset[sub_view_id[i]][j];frame_crop_top_offset[sub_view_id[i]][j];以及frame_crop_bottom_offset[sub_view_id[i]][j]。

[0183] 功能块1920使用多视图视频编码(MVC)来编码当前视图的当前深度,并且将控制传递至结束块1999。深度信号可以以类似于其对应的视频信号被编码的方式而被编码。例如,视图的深度信号可以被包括在仅包括其它深度信号、或仅包括视频信号或深度和视频信号二者的平铺中。然后将该平铺(伪视图)视为MVC的单个视图,并且还假定存在被视为MVC的其它视图的其他平铺。

[0184] 判定块1922确定语法元素tiling_mode是否等于零。如果是这样的,则将控制传递至功能块1924,否则,将控制传递至功能块1938。

[0185] 功能块1924设置语法元素flip_dir[sub_view_id[i]]以及语法元素upsample_view_flag[sub_view_id[i]],并且将控制传递至判定块1926。判定块1926确定语法元素

upsample_view_flag[sub_view_id[i]]的当前值是否等于一。如果是这样的话,则将控制传递至功能块1928。否则,将控制传递至功能块1930。

[0186] 功能块1928设置语法元素upsample_filter[sub_view_id[i]],并且将控制传递至判定块1930。判定块1930确定语法元素upsample_filter[sub_view_id[i]]的值是否等于三。如果是这样的话,则将控制传递至功能块1932。否则将控制传递至功能块1936。

[0187] 功能块1932设置以下语法元素,并且将控制传递至功能块1934:vert_dim[sub_view_id[i]];hor_dim[sub_view_id[i]];以及quantizer[sub_view_id[i]]。功能块1934设置每个YUV分量的滤波器系数,并且将控制传递至功能块1936。

[0188] 功能块1936递增变量i,并且将控制返回给判定块1908。

[0189] 功能块1938设置语法元素pixel_dist_x[sub_view_id[i]]以及语法元素flip_dist_y[sub_view_id[i]],并且将控制传递至功能块1940。功能块1940设置变量j等于零,并且将控制传递至判定块1942。判定块1942确定变量j的当前值是否小于语法元素num_parts[sub_view_id[i]]的当前值。如果是这样的话,则将控制传递至功能块1944。否则,将控制传递至功能块1936。

[0190] 功能块1944设置语法元素num_pixel_tiling_filter_coeffs_minus1[sub_view_id[i]],并且将控制传递至功能块1946。功能块1946设置所有像素平铺滤波器的系数,并且将控制传递至功能块1936。

[0191] 转向图20,由参考标号2000一般地指示使用MPEG-4 AVC标准的多图图视频编码(MVC)扩展的、为解码多个视图和深度的画面做准备而处理该画面的示例性方法。

[0192] 方法2000包括将控制传递至功能块2015的起始块2005。功能块2015分析语法元素num_coded_views_minus1,并且将控制传递至功能块2020。功能块2020分析对应于view_id[i]的所有(num_coded_views_minus1+1)个深度的语法元素view_id[i],并且将控制传递至功能块2025。功能块2025分析锚点深度画面的视图间参考相关性信息,并且将控制传递至功能块2030。功能块2030分析非锚点深度画面的视图间参考相关性信息,并且将控制传递至功能块2035。功能块2035分析语法元素pseudo_view_present_flag,并且将控制传递至判定块2040。判定块2040确定语法元素pseudo_view_present_fiag的当前值是否等于真。如果是这样的话,将控制传递至功能块2045。否则,将控制传递至结束块2099。

[0193] 功能块2045分析以下语法元素,并且将控制传递至功能块2050:tiling_mode;org_pic_width_in_mbs_minus1;以及org_pic_height_in_mbs_minus1。功能块2050调用每个编码视图的语法元素pseudo_view_info(view_id),并且将控制传递至结束块2099。

[0194] 转向图21,由参考标号2100一般地指示使用MPEG-4 AVC标准的多视图视频编码(MVC)扩展的、解码多个视图和深度的画面的示例性方法。

[0195] 方法2100包括起始块2012,其以输入参数pseudo_view_id开始,并且将控制传递至功能块2104。功能块2104分析语法元素num_sub_views_minus1,并且将控制传递至功能块2106。功能块2106设置变量i为零,并且将控制传递至判定块2108。判定块2108确定变量i是否小于sub_views的数量。如果是这样的话,则将控制传递至功能块2110。否则,将控制传递至功能块2120。

[0196] 功能块2110分析语法元素sub_view_id[i],并且将控制传递至功能块2112。功能块2112分析语法元素num_parts_minus1[sub_view_id[i]],并且将控制传递至功能块

2114。功能块2114设置变量j等于零,并且将控制传递至判定块2116。判定块2116确定变量j是否小于语法元素num_parts_minus1[sub_view_id[i]]。如果是这样的话,则将控制传递至功能块2118。否则,将控制传递至判定块2122。

[0197] 功能块2118设置以下语法元素,递增变量j,并且将控制返回给判定块2116:loc_left_offset[sub_view_id[i]][j];loc_top_offset[sub_view_id[i]][j];frame_crop_left_offset[sub_view_id[i]][j];frame_crop_right_offset[sub_view_id[i]][j];frame_crop_top_offset[sub_view_id[i]][j];以及frame_crop_bottom_offset[sub_view_id[i]][j]。

[0198] 功能块2120使用多视图视频编码(MVC)解码当前画面,并且将控制传递至功能块2121。功能块2121使用高级语法将每个视图从画面分离,并且将控制传递至结束块2199。使用高级语法分离每个视图如前面所描述。

[0199] 判定块2122确定语法元素tiling_mode是否等于零。如果是这样的话,则将控制传递至功能块2124。否则,将控制传递至功能块2138。

[0200] 功能块2124分析语法元素flip_dir[sub_view_id[i]]以及语法元素upsample_view_flag[sub_view_id[i]],并且将控制传递至判定块2126。判定块2126确定语法元素upsample_view_flag[sub_view_id[i]]的当前值是否等于一。如果是这样的话,则将控制传递至功能块2128。否则,将控制传递至判定块2130。

[0201] 功能块2128分析语法元素upsample_filter[sub_view_id[i]],并且将控制传递至判定块2130。判定块2130确定语法元素upsample_filter[sub_view_id[i]]的值是否等于三。如果是这样的话,将控制传递至功能块2132。否则,将控制传递至功能块2136。

[0202] 功能块2132分析以下语法元素,并且将控制传递至功能块2134:vert_dim[sub_view_id[i]];hor_dim[sub_view_id[i]];以及quantizer[sub_view_id[i]]。功能块2134分析每个YUV分量的滤波器系数,并且将控制传递至功能块2136。

[0203] 功能块2136递增变量i,并且将控制返回给判定块2108。

[0204] 功能块2138分析语法元素pixel_dist_x[sub_view_id[i]]以及语法元素flip_dist_y[sub_view_id[i]],并且将控制传递至功能块2140。功能块2140设置变量j等于零,并且将控制传递至判定块2142。判定块2142确定变量j的当前值是否小于语法元素num_parts[sub_view_id[i]]的当前值。如果是这样的话,将控制传递至功能块2144。否则,将控制传递至功能块2136。

[0205] 功能块2144分析语法元素num_pixel_tiling_filter_coefs_minus1[sub_view_id[i]],并且将控制传递至功能块2146。功能块2146分析所有像素平铺滤波器的系数,并且将控制传递至功能块2136。

[0206] 转向图22,由参考标号2200一般地指示像素级的平铺示例。下面进一步描述图22。

[0207] 多视图视频编码的应用是自由视点TV(或FTV)。该应用需要用户可以在两个或多个视图之间自由地移动。为了实现这一点,在两个视图之间的“虚拟”视图需要被内插或合成。存在若干执行视图内插的方法。该方法中的一个使用深度用于视图内插/合成。

[0208] 每个视图可以具有相关联的深度信号。由此,深度可以被视为另一种形式的视频信号。图9示出深度信号900的示例。为了使能诸如FTV之类的应用,深度信号随着视频信号被传送。在提出的平铺框架中,深度信号还可以被添加为平铺中的一个。图10示出添加为平

铺的深度信号的示例。在图10的右侧示出深度信号/平铺。

[0209] 一旦深度被编码为整个帧的一平铺,则高级语法应该指示哪个平铺是深度信号,使得呈现器可以恰当地使用深度信号。

[0210] 在当使用MPEG-4 AVC标准编码器(或对应不同视频编码标准和/或推荐标准的编码器)来编码输入序列(诸如图1中所示出的)的情形中,提出的高级语法可以出现在例如序列参数集(SPS)、画面参数集(PPS)、码片首标,和/或补充增强信息(SEI)消息中。提出的方法的实施例在表1中示出,其中语法出现在补充增强信息(SEI)消息中。

[0211] 在当使用MPEG-4 AVC标准的多视图视频编码(MVC)扩展的编码器(或对应关于不同视频编码标准和/或推荐标准的多视图视频编码标准的编码器)来编码伪视图的输入序列(诸如图1中所示出的)的情形中,提出的高级语法可以出现在SPS、PPS、码片首标、SEI消息或指定的简档(profile)中。提出的方法的实施例在表1中示出。表1示出出现在序列参数集(SPS)结构中的语法元素,包括根据本原理实现的所提出的语法元素。

[0212] 表1

	C	描述符
seq_parameter_set_mvc_extension() {		
num_views_minus_1		ue(v)
for(i = 0; i <= num_views_minus_1; i++)		
view_id[i]		ue(v)
for(j = 0; j <= num_views_minus_1; j++) {		
num_anchor_refs_10[i]		ue(v)
for(j = 0; j < num_anchor_refs_10[i]; j++)		
anchor_ref_10[i][j]		ue(v)
num_anchor_refs_11[i]		ue(v)
for(j = 0; j < num_anchor_refs_11[i]; j++)		
anchor_ref_11[i][j]		ue(v)
}		
for(i = 0; i <= num_views_minus_1; i++) {		
num_non_anchor_refs_10[i]		ue(v)
for(j = 0; j < num_non_anchor_refs_10[i]; j++)		
non_anchor_ref_10[i][j]		ue(v)
num_non_anchor_refs_11[i]		ue(v)
for(j = 0; j < num_non_anchor_refs_11[i]; j++)		
non_anchor_ref_11[i][j]		ue(v)
}		
pseudo_view_present_flag		u(1)
if(pseudo_view_present_flag) {		
tiling_mode		
org_pic_width_in_mbs_minus1		
org_pic_height_in_mbs_minus1		
for(i = 0; i < num_views_minus_1; i++)		
pseudo_view_info(i);		
}		
}		

[0214] 表2示出用于表1的pseudo_view_info语法元素的语法元素。

[0215] 表2

[0216]

	C	描述符
pseudo_view_info(pseudo_view_id) {		
num_sub_views_minus_1[pseudo_view_id]	5	ue(v)
if (num_sub_views_minus_1 != 0) {		
for (i = 0; i < num_sub_views_minus_1[pseudo_view_id]; i++) {		
sub_view_id[i]	5	ue(v)
num_parts_minus1[sub_view_id[i]]	5	ue(v)
for (j = 0; j <= num_parts_minus1[sub_view_id[i]]; j++) {		
loc_left_offset[sub_view_id[i]][j]	5	ue(v)
loc_top_offset[sub_view_id[i]][j]	5	ue(v)
frame_crop_left_offset[sub_view_id[i]][j]	5	ue(v)
frame_crop_right_offset[sub_view_id[i]][j]	5	ue(v)
frame_crop_top_offset[sub_view_id[i]][j]	5	ue(v)
frame_crop_bottom_offset[sub_view_id[i]][j]	5	ue(v)
}		
if (tiling_mode == 0) {		
flip_dir[sub_view_id[i]]	5	u(2)
upsample_view_flag[sub_view_id[i]]	5	u(1)
if (upsample_view_flag[sub_view_id[i]])		
upsample_filter[sub_view_id[i]]	5	u(2)
if (upsample_filter[sub_view_id[i]] == 3) {		
vert_dim[sub_view_id[i]]	5	ue(v)
hor_dim[sub_view_id[i]]	5	ue(v)
quantizer[sub_view_id[i]]	5	ue(v)
for (yuv = 0; yuv < 3; yuv++) {		
for (y = 0; y < vert_dim[sub_view_id[i]] - 1; y++) {		
for (x = 0; x < hor_dim[sub_view_id[i]] - 1; x++)		
filter_coeffs[sub_view_id[i]][yuv][y][x]	5	se(v)
}		
}		
}		
} // if (tiling_mode == 0)		
else if (tiling_mode == 1) {		
pixel_dist_x[sub_view_id[i]]		
pixel_dist_y[sub_view_id[i]]		
for (j = 0; j <= num_parts[sub_view_id[i]]; j++) {		
num_pixel_tiling_filter_coeffs_minus1[sub_view_id[i]][j]		
for (coeff_idx = 0; coeff_idx <=		
num_pixel_tiling_filter_coeffs_minus1[sub_view_id[i]][j]; j++)		
pixel_tiling_filter_coeffs[sub_view_id[i]][j][coeff_idx]		
} // for (j = 0; j <= num_parts[sub_view_id[i]]; j++)		
} // else if (tiling_mode == 1)		
} // for (i = 0; i < num_sub_views_minus_1; i++)		
} // if (num_sub_views_minus_1 != 0)		
}		

[0217] 下面描述表1和表2中出现的语法元素的语义：

[0218] pseudo_view_present_flag等于真指示某些视图是多个子视图的超级视图。

[0219] tiling_mode等于0指示在画面级平铺子视图。值1指示在像素级完成平铺。

[0220] 新SEI消息可以使用没有被用于MPEG-4 AVC标准或MPEG-4 AVC标准的扩展的SEI

载荷类型的值。新SEI消息包括具有下面语义的若干语法元素。

[0221] `num_coded_views_minus1`加1指示由比特流支持的编码视图的数量。`num_coded_views_minus1`的值在0到1023的范围内,包括0和1023。

[0222] `org_pic_width_in_mbs_minus1`加1指定宏块单位中的每个视图的画面的宽度。

[0223] 按以下导出宏块单位中的画面宽度的变量:

[0224] $PicWidthInMbs = org_pic_width_in_mbs_minus1 + 1$

[0225] 按以下导出亮度分量的画面宽度的变量:

[0226] $PicWidthInSamplesL = PicWidthInMbs * 16$

[0227] 按以下导出色度分量的画面宽度的变量:

[0228] $PicWidthInSamplesC = PicWidthInMbs * MbWidthC$

[0229] `org_pic_height_in_mbs_minus1`加1指定宏块单位中每个视图中的画面高度。

[0230] 按以下导出宏块单位中的画面高度的变量:

[0231] $PicHeightInMbs = org_pic_height_in_mbs_minus1 + 1$

[0232] 按以下导出亮度分量的画面高度的变量:

[0233] $PicHeightInSamplesL = PicHeightInMbs * 16$

[0234] 按以下导出色度分量的画面高度的变量:

[0235] $PicHeightInSamplesC = PicHeightInMbs * MbHeightC$

[0236] `num_sub_views_minus1`加1指示在当前视图中包括的编码子视图的数量。`num_coded_views_minus1`的值在0到1023的范围内,包括0和1023。

[0237] `sub_view_id[i]`指定具有由*i*指示的解码顺序的子视图的`sub_view_id`。

[0238] `num_parts[sub_view_id[i]]`指定`sub_view_id[i]`的画面被分成的部分的数量。

[0239] `loc_left_offset[sub_view_id[i]][j]`和`loc_top_offset[sub_view_id[i]][j]`分别指定在左侧像素偏移和顶部像素偏移中的位置,其中,当前部分*j*位于具有`sub_view_id`等于`sub_view_id[i]`的视图的最终重构画面中。

[0240] `view_id[i]`指定具有由*i*指示的编码顺序的视图的`view_id`。

[0241] `frame_crop_left_offset[sub_view_id[i]][j]`,`frame_crop_right_offset[sub_view_id[i]][j]`,`frame_crop_top_offset[sub_view_id[i]][j]`,以及`frame_crop_bottom_offset[sub_view_id[i]][j]`根据用于输出的帧坐标中指定的矩形区域来指定具有`sub_view_id`等于`sub_view_id[i]`的视图中部分*j*的编码视频序列中的画面的样本。

[0242] 按如下导出变量`CropUnitX`和`CropUnitY`:

[0243] 一如果`chroma_format_idc`等于0,则按如下导出`CropUnitX`和`CropUnitY`:

[0244] $CropUnitX = 1$

[0245] $CropUnitY = 2 - frame_mbs_only_flag$

[0246] 一否则(`chroma_format_idc`等于1、2、或3),按如下导出`CropUnitX`和`CropUnitY`:

[0247] $CropUnitX = SubWidthC$

[0248] $CropUnitY = SubHeightC * (2 - frame_mbs_only_flag)$

[0249] 该帧剪切矩形包括具有依据如下的水平帧坐标的亮度样本:

[0250] `CropUnitX*frame_crop_left_offset`至`PicWidthInSamplesL - (CropUnitX*frame_crop_right_offset+1)`以及垂直帧坐标从`CropUnitY*frame_crop_top_offset`至

$(16 * \text{FrameHeightInMbs}) - (\text{CropUnitY} * \text{frame_crop_bottom_offset} + 1)$, 包括端点。 $\text{frame_crop_left_offset}$ 的值应该在范围0至 $(\text{PicWidthInSamplesL} / \text{CropUnitX}) - (\text{frame_crop_right_offset} + 1)$ 内, 包括端点; 并且 $\text{frame_crop_top_offset}$ 的值应该在范围0至 $(16 * \text{FrameHeightInMbs} / \text{CropUnitY}) - (\text{frame_crop_bottom_offset} + 1)$ 内, 包括端点。

[0251] 当 chroma_format_idc 不等于0时, 两个色度矩阵的对应的指定样本是具有帧坐标 $(x / \text{SubWidthC}, y / \text{SubHeightC})$ 的样本, 其中 (x, y) 是指定的亮度样本的帧坐标。

[0252] 对于解码的场, 解码的场的指定样本是落入帧坐标中指定的矩形内的样本。

[0253] $\text{num_parts}[\text{view_id}[i]]$ 指定 $\text{view_id}[i]$ 的画面所分成的部分的数量。

[0254] $\text{depth_flag}[\text{view_id}[i]]$ 指定当前部分是否是深度信号。如果 depth_flag 等于零, 则当前部分不是深度信号。如果 depth_flag 等于1, 则当前部分是与由 $\text{view_id}[i]$ 标识的视图相关联的深度信号。

[0255] $\text{flip_dir}[\text{sub_view_id}[i]][j]$ 指定当前部分的翻转方向。 flip_dir 等于0指示没有翻转, flip_dir 等于1指示水平方向的翻转, flip_dir 等于2指示垂直方向的翻转, 以及 flip_dir 等于3指示水平和垂直方向的翻转。

[0256] $\text{flip_dir}[\text{view_id}[i]][j]$ 指定当前部分的翻转方向, flip_dir 等于0指示没有翻转, flip_dir 等于1指示水平方向的翻转, flip_dir 等于2指示垂直方向的翻转, 并且 flip_dir 等于3指示水平和垂直方向的翻转。

[0257] $\text{loc_left_offset}[\text{view_id}[i]][j]$ 、 $\text{loc_top_offset}[\text{view_id}[i]][j]$ 指定像素偏移中的位置, 其中当前部分 j 位于具有 view_id 等于 $\text{view_id}[i]$ 的视图的最终重构画面中。

[0258] $\text{upsample_view_flag}[\text{view_id}[i]]$ 指示属于由 $\text{view_id}[i]$ 指定的视图的该画面是否需要被上采样。 $\text{upsample_view_flag}[\text{view_id}[i]]$ 等于0指定具有 view_id 等于 $\text{view_id}[i]$ 的画面将不被上采样。 $\text{upsample_view_flag}[\text{view_id}[i]]$ 等于1指定具有 view_id 等于 $\text{view_id}[i]$ 的画面将被上采样。

[0259] $\text{upsample_filter}[\text{view_id}[i]]$ 指示要用于上采样的滤波器的类型。 $\text{upsample_filter}[\text{view_id}[i]]$ 等于0指示应该使用6抽头AVC滤波器, $\text{upsample_filter}[\text{view_id}[i]]$ 等于1指示应该使用4抽头SVC滤波器, $\text{upsample_filter}[\text{view_id}[i]]$ 等于2指示应该使用双线性滤波器, $\text{upsample_filter}[\text{view_id}[i]]$ 等于3指示传送定制滤波器系数。当 $\text{upsample_filter}[\text{view_id}[i]]$ 未出现时, 将其设置为0。在本实施例中, 我们使用2D定制滤波器。这可以容易地扩展至1D滤波器以及某些其他非线性滤波器。

[0260] $\text{vert_dim}[\text{view_id}[i]]$ 指定定制2D滤波器的垂直尺寸。

[0261] $\text{hor_dim}[\text{view_id}[i]]$ 指定定制2D滤波器的水平尺寸。

[0262] $\text{quantizer}[\text{view_id}[i]]$ 指定每个滤波器系数的量化因数。

[0263] $\text{filter_coeffs}[\text{view_id}[i]][\text{yuv}][\text{y}][\text{x}]$ 指定量化的滤波器系数, yuv 发信号通知应用滤波器系数的分量, yuv 等于0指定Y分量, yuv 等于1指定U分量, 以及 yuv 等于2指定V分量。

[0264] $\text{pixel_dist_x}[\text{sub_view_id}[i]]$ 以及 $\text{pixel_dist_y}[\text{sub_view_id}[i]]$ 分别指定在 sub_view_id 等于 $\text{sub_view_id}[i]$ 的视图中的相邻像素之间、最终重构伪视图中水平方向和垂直方向上的距离。

[0265] num_pixel_tiling_filter_coefifs_minus1[sub_view_id[i][j]加一指示当将平铺模式等于1时滤波器系数的数量。

[0266] pixel_tiling_filter_coefifs[sub_view_id[i][j]用信号通知代表可以被用于滤波平铺画面的滤波器所需的滤波器系数。

[0267] 转向图22,由参考标号2210和2220分别指示示出从四个视图平铺的像素的伪视图组成的两个示例。由参考标号2250共同地指示四个视图。在下面的表3中提供图22中的第一示例的语法值。

[0268] 表3

[0269]

pseudo_view_info (pseudo_view_id) {	值
num_sub_views_minus_1[pseudo_view_id]	3
sub_view_id[0]	0
num_parts_minus1[0]	0
loc_left_offset[0][0]	0
loc_top_offset[0][0]	0
pixel_dist_x[0][0]	0
pixel_dist_y[0][0]	0
sub_view_id[1]	0
num_parts_minus1[1]	0
loc_left_offset[1][0]	1
loc_top_offset[1][0]	0
pixel_dist_x[1][0]	0
pixel_dist_y[1][0]	0
sub_view_id[2]	0
num_parts_minus1[2]	0
loc_left_offset[2][0]	0
loc_top_offset[2][0]	1
pixel_dist_x[2][0]	0
pixel_dist_y[2][0]	0
sub_view_id[3]	0
num_parts_minus1[3]	0
loc_left_offset[3][0]	1
loc_top_offset[3][0]	1
pixel_dist_x[3][0]	0
pixel_dist_y[3][0]	0

[0270] 除了以下两个语法元素:loc_left_offset[3][0]等于5以及loc_top_offset[3][0]等于3之外,图22中的第二示例的语法值全部相同。

[0271] 偏移指示对应于视图的像素应该在某一偏移位置开始。这在图22中示出(2220)。例如当两个视图产生其中共同的对象表现出从一个视图到另一个视图被位移的图像时,可

能进行这一点。例如,如果第一和第二相机(代表第一和第二视图)拍摄对象的画面,与第一视图比较,该对象可能在第二视图中表现出向右位移五个像素。这意味着在第一视图中的像素 $(i-5, j)$ 对应第二视图中的像素 (i, j) 。如果两个视图中的像素被简单地逐像素平铺,则在平铺的相邻像素之间可能不存在多少相关性,并且空间编码获益(gain)可能是小的。相反地,通过位移平铺使得来自视图一的像素 $(i-5, j)$ 邻近来自视图二的像素 (i, j) 放置,可以增加空间相关性并且也可以增加空间编码获益。这是例如因为第一和第二视图中的对象的对应像素被相互邻近地平铺。

[0272] 由此,loc_left_offset和loc_top_offset的存在可能有益于编码效率。可以通过外部部件获得偏移信息。例如,相机的位置信息或视图之间的全局视差矢量可能被用于确定这样的偏移信息。

[0273] 作为偏移的结果,伪视图中的某些像素未被分配来自任何视图的像素值。继续上面的示例,当对于 $i=0 \dots 4$ 的值,沿着来自视图二的像素 (i, j) 平铺来自视图一的像素 $(i-5, j)$ 时,不存在用以平铺的来自视图一的像素 $(i-5, j)$,所以那些像素在平铺中是空的。对于在伪视图(平铺)中未被分配来自任何视图的像素值的那些像素,至少一种实现方式使用类似AVC中的运动补偿中的子像素内插例程的内插例程。即,空平铺像素可以依据相邻像素被内插。这样的内插可能造成平铺中更大的空间相关性和平铺的更高的编码获益。

[0274] 在视频编码中,可以为每个画面(诸如I、P和B画面之类)选取不同编码类型。此外,对于多视图视频编码,定义锚点和非锚点画面。在一实施例中,提出编组的判定可以基于画面类型进行。以高级语法用信号通知该编组的信息。

[0275] 转向图11,由参考标号1100一般地指示在单个帧上平铺的5个视图的示例。具体地,用单个帧上平铺的5个视图示出舞厅序列(ballroom sequence)。此外,可以看出第五个视图被分成两个部分,使得其可以被布置在矩形帧上。在此,每个视图是QVGA大小,所以全部帧尺寸是 640×600 。由于600不是16的倍数,所以其应该被扩展至608。

[0276] 对于本示例,可以在表4中示出可能的SEI消息。

[0277] 表4

[0278]

multiview_display_info(payloadSize) {	值
num_coded_views_minus1	5
org_pic_width_in_mbs_minus1	40
org_pic_height_in_mbs_minus1	30
view_id[0]	0
num_parts[view_id[0]]	1
depth_flag[view_id[0]][0]	0
flip_dir[view_id[0]][0]	0
loc_left_offset[view_id[0]][0]	0
loc_top_offset[view_id[0]][0]	0
frame_crop_left_offset[view_id[0]][0]	0
frame_crop_right_offset[view_id[0]][0]	320
frame_crop_top_offset[view_id[0]][0]	0
frame_crop_bottom_offset[view_id[0]][0]	240
upsample_view_flag[view_id[0]]	1
if(upsample_view_flag[view_id[0]]) {	
vert_dim[view_id[0]]	6
hor_dim[view_id[0]]	6
quantizer[view_id[0]]	32
for (yuv= 0; yuv< 3; yuv++) {	
for (y = 0; y < vert_dim[view_id[i]] - 1; y++) {	
for (x = 0; x < hor_dim[view_id[i]] - 1; x++)	
filter_coefs[view_id[i]] [yuv][y][x]	XX
view_id[1]	1
num_parts[view_id[1]]	1
depth_flag[view_id[0]][0]	0
flip_dir[view_id[1]][0]	0
loc_left_offset[view_id[1]][0]	0
loc_top_offset[view_id[1]][0]	0
frame_crop_left_offset[view_id[1]][0]	320
frame_crop_right_offset[view_id[1]][0]	640
frame_crop_top_offset[view_id[1]][0]	0
frame_crop_bottom_offset[view_id[1]][0]	320
upsample_view_flag[view_id[1]]	1

[0279]

if(upsample_view_flag[view_id[1]]) {	
vert_dim[view_id[1]]	6
hor_dim[view_id[1]]	6
quantizer[view_id[1]]	32
for (yuv= 0; yuv< 3; yuv++) {	
for (y = 0; y < vert_dim[view_id[i]] - 1; y++) {	
for (x = 0; x < hor_dim[view_id[i]] - 1; x++)	
filter_coeffs[view_id[i]][yuv][y][x]	XX
}	
}	
.....(similarly for view 2,3)	

[0280]

view_id[4]	4
num_parts[view_id[4]]	2
depth_flag[view_id[0]][0]	0
flip_dir[view_id[4]][0]	0
loc_left_offset[view_id[4]][0]	0
loc_top_offset[view_id[4]][0]	0
frame_crop_left_offset[view_id[4]][0]	0
frame_crop_right_offset[view_id[4]][0]	320
frame_crop_top_offset[view_id[4]][0]	480
frame_crop_bottom_offset[view_id[4]][0]	600
flip_dir[view_id[4]][1]	0
loc_left_offset[view_id[4]][1]	0
loc_top_offset[view_id[4]][1]	120
frame_crop_left_offset[view_id[4]][1]	320
frame_crop_right_offset[view_id[4]][1]	640
frame_crop_top_offset[view_id[4]][1]	480
frame_crop_bottom_offset[view_id[4]][1]	600
upsample_view_flag[view_id[4]]	1
if(upsample_view_flag[view_id[4]]) {	
vert_dim[view_id[4]]	6
hor_dim[view_id[4]]	6
quantizer[view_id[4]]	32
for (yuv= 0; yuv< 3; yuv++) {	
for (y = 0; y < vert_dim[view_id[i]] - 1; y++) {	
for (x = 0; x < hor_dim[view_id[i]] - 1; x++)	
filter_coeffs[view_id[i]][yuv][y][x]	XX
}	
}	

[0281]

表5示出用于传送表4中示出的示例的多视图信息的一般语法结构。

[0282]

表5

	multiview_display_info(payloadSize) {	C	标识符
	num_coded_views_minus1	5	ue(v)
	org_pic_width_in_mbs_minus1	5	ue(v)
	org_pic_height_in_mbs_minus1	5	ue(v)
[0283]	for(i = 0; i <= num_coded_views_minus1; i++) {		
	view_id[i]	5	ue(v)
	num_parts[view_id[i]]	5	ue(v)
	for(j = 0; j <= num_parts[i]; j++) {		
	depth_flag[view_id[i]][j]		
	flip_dir[view_id[i]][j]	5	u(2)
	loc_left_offset[view_id[i]][j]	5	ue(v)
	loc_top_offset[view_id[i]][j]	5	ue(v)
	frame_crop_left_offset[view_id[i]][j]	5	ue(v)
	frame_crop_right_offset[view_id[i]][j]	5	ue(v)
	frame_crop_top_offset[view_id[i]][j]	5	ue(v)
	frame_crop_bottom_offset[view_id[i]][j]	5	ue(v)
	}		
	upsample_view_flag[view_id[i]]	5	u(1)
	if(upsample_view_flag[view_id[i]])		
	upsample_filter[view_id[i]]	5	u(2)
[0284]	if(upsample_filter[view_id[i]] == 3) {		
	vert_dim[view_id[i]]	5	ue(v)
	hor_dim[view_id[i]]	5	ue(v)
	quantizer[view_id[i]]	5	ue(v)
	for (yuv= 0; yuv< 3; yuv++) {		
	for (y = 0; y < vert_dim[view_id[i]] - 1; y++) {		
	for (x = 0; x < hor_dim[view_id[i]] - 1; x++)		
	filter_coeffs[view_id[i]] [yuv][y][x]	5	se(v)
	}		
	}		
	}		
	}		
	}		

[0285] 参考图23, 示出视频处理设备2300。视频处理设备可以例如是机顶盒或接收编码视频并且向用户提供用于显示的解码视频或者用于存储的解码视频的其他设备。由此, 设备2300可以将其输出提供给电视机、计算机监视器或计算机或者其他处理设备。

[0286] 设备2300包括接收数据信号2320的解码器2310。数据信号2320可以包括例如AVC或MVC相容的流。解码器2310解码所接收的信号2320的全部或部分并且提供解码视频信号2330和平铺显示信息2340作为输出。向选择器2350提供解码视频2330和平铺信息2340。设备2300还包括接收用户输入2370的用户接口2360。用户接口2360基于用户输入2370向选择器2350提供画面选择信号2380。画面选择信号2380和用户输入2370指示用户期望显示多个画面中的哪一些。选择器2350提供所选择的(多个)画面作为输出2390。选择器2350使用画面选择信息2380以选择解码视频2330中的哪些画面提供作为输出2390。选择器2350使用平铺显示信息2340, 以在解码视频2330中放置所选择的(多个)画面。

[0287] 在各种实现方式中,选择器2350包括用户接口2360,而在其他实现方式中,不需要用户接口2360,这是因为选择器2350直接接收用户输入2370,不需要执行单独的接口功能。选择器2350可以例如以软件实现或者可以实现为集成电路。选择器2350还可以合并解码器2310。

[0288] 更一般地,在该应用中描述的各种实现方式的解码器可以提供包括全部平铺的解码输出。附加地或可替代地,解码器可以提供仅仅包括从平铺选择的一个或多个画面(例如,图像或深度信号)的解码输出。

[0289] 如上所述,根据本原理的一个或多个实施例,高级语法可以用于执行用信号通知。高级语法可以例如用于,但不限于用信号通知以下中的任一个:较大帧中呈现的编码视图的数量;所有视图的原始宽度和高度;对于每个编码视图,对应于该视图的视图标识符;对于每个编码视图,视图帧所分成的部分的数量;对于视图的每个部分,翻转方向(其可以是例如未翻转、仅水平翻转、仅垂直翻转,或水平且垂直翻转);对于视图的每个部分,像素中的左侧位置或宏块的数量,其中当前部分属于视图的最终帧;对于视图的每个部分,像素中该部分的顶部位置或宏块的数量,其中当前部分属于视图的最终帧;对于视图的每个部分,像素中剪切窗口的当前大的解码/编码帧的左侧位置或宏块的数量;对于视图的每个部分,像素中剪切窗口的当前大的解码/编码帧的右侧位置或宏块数量;对于视图的每个部分,像素中剪切窗口的当前大的解码/编码帧的顶部位置或宏块数量;以及,对于视图的每个部分,像素中剪切窗口的当前大的解码/编码帧的底部位置或宏块数量;对于每个编码视图,在输出之前视图是否需要被上采样(其中,如果需要执行上采样,则高级语法可以被用于指示用于上采样的方法(包括但不限于AVC 6-抽头滤波器、SVC 4-抽头滤波器、双线性滤波器或定制的1D、2D线性或非线性滤波器))。

[0290] 应注意,术语“编码器”以及“解码器”意味着一般结构,并且不限于任何特定功能或特征。例如,解码器可以接收携带编码比特流的调制载波,并且解调该编码比特流以及解码比特流。

[0291] 已经描述各种方法。这些方法中的许多被详述,以提供充分的公开。然而注意,设想可能改变这些方法描述的一个或许多特定特征的变型。另外,叙述的许多特征在本领域是已知的,并且因此不进行详细描述。

[0292] 另外,已经对在若干实施方式中使用高级语法用于发送某一信息进行引用。然而,应理解其他实现方式使用较低级语法或实际上与其他机制一起使用(诸如,例如发送信息作为编码数据的一部分),以提供相同的信息(或该信息的变型)。

[0293] 各种实现方式提供平铺和合适的信令,以允许多个视图(更一般地,画面)被平铺为单个画面、被编码为单个画面并且作为单个画面被发送。信令信息可能允许后处理器将视图/画面分离。此外,被平铺的多个画面可以是视图,但画面中的至少一个可以是深度信息。这些实现方式可以提供一个或多个优点。例如,用户可能希望以平铺的方式显示多个视图,并且这些各种实现方式提供通过在编码之前平铺这样的视图并且以平铺的方式传送/存储它们来提供编码并且传送或存储它们的高效方法。

[0294] 在AVC和/或MVC的背景中平铺多个视图的实现方式还提供附加的优点。AVC表面上仅用于单个画面,所以不期望附加的视图。然而,这样的基于AVC的实现方式可以在AVC环境中提供多个视图,这是因为平铺的视图可以被布置,使得例如解码器知道平铺的画面属于

不同的视图(例如,伪视图中的左上画面是视图1,右上画面是视图2,等等)。

[0295] 此外,MVC已经包括多个视图,所以不期望多个视图被包括在单个伪视图中。另外,MVC对可以被支持的视图的数量具有限制,并且这样的基于MVC的实现方式通过允许(如在基于AVC的实现方式中)附加的视图被平铺来有效地增加可以支持的视图的数量。例如,每个伪视图可能对应于MVC支持的视图中的一个,并且解码器可能知道每个“支持的视图”实际包括以预布置平铺顺序的四个视图。由此,在这样的实现方式中,可能的视图的数量是“支持的视图”的数量的四倍。

[0296] 在下面的描述中,诸如SEI消息之类的高级语法被扩展至各种实现方式,以包括关于多个空间交织模式中的哪个可归因于被平铺为单个画面的画面的信息。空间交织可以发生在多个模式中,诸如例如并排,从顶部到底部,垂直交织、水平交织和棋盘。此外,语法被扩展至包括关于平铺的画面中的内容的关系信息。关系信息可以包括立体图像中左右画面的指定或者当2D加深度用于画面时作为深度画面的标识,或者甚至画面形成分层的深度视频(LDV)信号的指示。

[0297] 如已经在本文中的上面所述,语法被设想为在除了SEI消息之外的任何高级语法,诸如,码片首标级、画面参数集(PPS)级、序列参数集(SPS)级、视图参数集(VPS)级和网络抽象层(NAL)单元首标级之类的语法,中实现。此外,设想低级语法可以被用于用信号通知该信息。甚至设想该信息可以以各种方式在波段之外(out of band)被用信号通知。

[0298] 如在前述的AVC草案修改中描述的棋盘空间交织的SEI消息语法被定义在如下的表6中:

[0299] 表6

	<code>spatially_interleaved_pictures(payloadSize) {</code>	C	描述符
	<code> spatially_interleaved_pictures_id</code>	5	ue(v)
	<code> spatially_interleaved_pictures_cancel_flag</code>	5	u(1)
	<code> if(!spatially_interleaved_pictures_cancel_flag) {</code>		
[0300]	<code> basic_spatial_interleaving_type_id</code>	5	u(8)
	<code> spatially_interleaved_pictures_repetition_period</code>	5	ue(v)
	<code> }</code>		
	<code> additional_extension_flag</code>	5	u(1)
	<code>}</code>		

[0301] 该SEI消息通知解码器使用指示的空间交织方案、由多个区别的画面的空间交织形成输出解码画面。解码器可以使用该SEI消息中的信息,以适当地去交织并且处理画面数据用于显示或其他目的。

[0302] 表6中的空间交织画面SEI消息的语法的定义值的语义如下:

[0303] `spatially_interleaved_pictures_id`包含可以用于标识对空间交织画面SEI消息的使用的标识数字。

[0304] `spatially_interleaved_pictures_cancel_flag`等于1指示空间交织画面SEI消息以输出顺序取消任何先前空间交织画面SEI消息的持久性(persistence)。`spatially_`

interleaved_pictures_cancel_flag等于0指示空间交织画面跟随。

[0305] basic_spatial_interleaving_type_id指示被包括在单个平铺画面中的多个画面的空间交织类型。

[0306] basic_spatial_interleaving_type_id等于0指示解码画面的每个分量平面包含对应于两个画面平面的基于“棋盘”的交织,如图26例示。

[0307] basic_spatial_interleaving_type_id等于1指示解码画面的每个分量平面包含对应于两个画面平面的基于“棋盘”的交织,如图26例示,并且附加地指示两个构成画面形成如图26例示的立体视图场景的左右视图。

[0308] spatially_interleaved_pictures_repetition_period指定空间交织画面SEI消息的持久性并且可以指定画面顺序计数间隔,具有相同的spatially_interleaved_pictures_id值的另一空间交织画面SEI消息或编码视频序列末尾将在该画面顺序计数间隔内呈现于比特流中。

[0309] spatially_interleaved_pictures_repetition_period等于0指定空间交织画面SEI消息仅应用于当前解码画面。

[0310] spatially_interleaved_pictures_repetition_period等于1指定空间交织画面SEI消息以输出顺序持续直至以下条件中的任一个为真:

[0311] -新编码视频顺序开始。

[0312] -访问单元中包含具有相同值的spatially_interleaved_pictures_id的空间交织画面SEI消息的画面是具有大于PicOrderCnt(CurrPic)的PicOrderCnt()的输出。

[0313] spatially_interleaved_pictures_repetition_period等于0或等于1指示具有相同值的spatially_interleaved_pictures_id的另一空间交织画面SEI消息可能呈现或可能不呈现。

[0314] spatially_interleaved_pictures_repetition_period大于1指定空间交织画面SEI消息持续直至以下条件中的任一个为真:

[0315] -新编码视频序列开始。

[0316] -访问单元中包含具有相同值的spatially_interleaved_pictures_id的空间交织画面SEI消息的画面是具有大于PicOrderCnt(CurrPic)并且小于或等于PicOrderCnt(CurrPic)+spatially_interleaved_pictures_repetition_period的PicOrderCnt()的输出。

[0317] spatially_interleaved_pictures_repetition_period大于1指示具有相同值的spatially_interleaved_pictures_id的另一空间交织画面SEI消息将呈现用于作为具有大于PicOrderCnt(CurrPic)并且小于或等于PicOrderCnt(CurrPic)+spatially_interleaved_pictures_repetition_period的PicOrderCnt()的输出的访问单元中的画面;除非比特流结束或者在不需要输出这样的画面的情况下开始新编码视频序列。

[0318] additional_extension_flag等于0指示在空间交织画面SEI消息中没有附加数据跟随。

[0319] 在不需要改变表6中示出的语法的情况下,本应用的实现方式提供示例性SEI消息内的关系信息以及空间交织信息。basic_spatial_interleaving_type_id的可能值范围在该实现方式中被修改并且被扩展,以指示多个空间交织方法,而不是只是一个棋盘方法。此

外,参数basic_spatial_interleaving_type_id被利用,以指示在画面中呈现空间交织的特定类型以及构成交织画面彼此相关。在该实现方式中,对于basic_spatial_interleaving_type_id,语义如下:

[0320] 值2或3意味着单个画面包含如图27中所例示的两个画面的对应平面的“并排”交织。值3附加地意味着两个构成画面形成立体视图场景的左视图和右视图。对于并排交织,将一个画面放置于另一画面的一侧,使得复合画面包括两个并排的图像。

[0321] 值4或5意味着单个画面包含两个画面的对应平面的“从顶部到底部”交织,如图28所例示。值5附加地意味着两个构成画面形成立体视图场景的左视图和右视图。对于从顶部至底部交织,一个画面放置于另一画面之上,以使出现一个画面在另一画面之上的复合画面。

[0322] 值6或7意味着单个画面包含两个画面的对应平面的“逐行”交织或简单行交织,如图29所例示。值7附加地意味着两个构成画面形成立体视图场景的左视图和右视图。对于逐行交织,单个画面的连续行从一个构成画面到另一构成画面交替。基本地,单个画面是构成图像的水平码片的交替。

[0323] 值8或9意味着单个画面包含两个画面的对应平面的“逐列”交织,如图30所例示。值9附加地意味着两个构成画面形成立体视图场景的左视图和右视图。对于逐列交织,单个画面的连续列从一个构成画面到另一个构成画面交替。基本地,单个画面是构成图像的垂直码片的交替。

[0324] 在用于示例性SEI消息以传递关于相关联的画面的信息的语法的另一个实施例中,若干附加语法元素已经被包括在表7中,以指示附加的信息。包括这样的附加语法元素,例如以指示一个或多个构成画面的方位(例如,翻转),以单独地指示对于图像是否存在左右立体成对关系,以指示任一构成画面是否需要被上采样,以及以指示上采样可能的度数以及方向。

[0325] 表7

	C	描述符
<code>spatially_interleaved_pictures(payloadSize) {</code>		
<code>spatially_interleaved_pictures_id</code>	5	ue(v)
<code>spatially_interleaved_pictures_cancel_flag</code>	5	U(1)
<code>if(!spatially_interleaved_pictures_cancel_flag) {</code>		
<code>basic_spatial_interleaving_type_id</code>	5	U(8)
<code>stereo_pair_flag</code>	5	U(1)
<code>upsample_conversion_horizontal_flag</code>	5	U(1)
[0326] <code>upsample_conversion_vertical_flag</code>	5	U(1)
<code>if(basic_spatial_interleaving_type_id == 1 or</code> <code>basic_spatial_interleaving_type_id == 2) {</code>		
<code>flip_flag</code>	5	U(1)
<code>}</code>		
<code>spatially_interleaved_pictures_repetition_period</code>	5	ue(v)
<code>}</code>		
<code>additional_extension_flag</code>	5	U(1)

[0327] 语义被定义如下：

[0328] `basic_spatial_interleaving_type_id`指示画面的空间交织类型。

[0329] `basic_spatial_interleaving_type_id`等于0指示解码画面的每个分量平面包含如早先提出的两个画面的对应平面的基于“棋盘”的交织(见图26)。

[0330] `basic_spatial_interleaving_type_id`等于1指示解码画面的每个分量平面包含两个画面的对应平面的基于“并排”的交织,如图27所例示。

[0331] `basic_spatial_interleaving_type_id`等于2指示解码画面的每个分量平面包含两个画面的对应平面的基于“从顶部到底部”的交织,如图28所例示。

[0332] `basic_spatial_interleaving_type_id`等于3指示解码画面的每个分量平面包含两个画面的对应平面的基于“逐行”的交织,如图29所例示。

[0333] `basic_spatial_interleaving_type_id`等于4指示解码画面的每个分量平面包含两个画面的对应平面的基于“逐列”的交织,如图30所例示。

[0334] `stereo_pair_flag`指示两个构成画面是否具有形成立体视图场景的左视图和右视图的关系。值0指示构成画面不形成左视图和右视图。值1指示它们有关于形成图像的左视图和右视图。

[0335] `upsample_conversion_horizontal_flag`指示解码期间将两个构成画面从单个画面中提取出之后,它们是否需要水平方向上的上采样。值0指示不需要上采样。这对应于采样因数为零。值1指示需要通过采样因数二进行上采样。

[0336] `upsample_conversion_vertical_flag`指示解码期间两个构成画面从单个画面中被提取出后,它们是否需要垂直方向上的上采样。值0指示不需要上采样。这对应于采样因数为零。值1指示需要通过采样因数二进行上采样。

[0337] 如下面更加详细描述,设想上采样操作的许多方面可以在SEI消息中递送,使得

上采样操作在画面解码期间被适当处理。例如,上采样的因数的附加范围可以被指示;上采样滤波器的类型也可以被指示;下采样滤波器也可以被指示使得解码器可以确定合适的或甚至最优的用于上采样的滤波器。还设想包括上采样滤波器系数的数量和值的滤波器系数信息还可以在SEI消息中被指示,使得接收器执行优选的上采样操作。

[0338] 采样因数指示视频画面的原始大小和采样大小之间的比率。例如,当采样因数是2时,原始画面大小是采样画面大小的两倍。画面大小通常是像素中分辨率的度量。所以,水平下采样的画面需要相同因数的对应水平上采样,以恢复原始视频画面的分辨率。如果原始画面具有例如1024像素的宽度,其可以由采样因数2而被水平地下采样,以变成具有宽度512像素的下采样画面。画面大小通常是像素中的分辨率的度量。可以示出用于垂直下采样或上采样的类似分析。可以将采样因数应用于依靠组合的水平和垂直途径的下采样或上采样。

[0339] `flip_flag`指示第二构成画面是否被翻转。值0指示在该画面中不呈现翻转。值1指示执行翻转。本领域的技术人员应理解,翻转涉及关于画面平面中的中心轴的180°旋转。在该实施例中,交织类型确定翻转方向。例如,当并排交织(`basic_spatial_interleaving_type_id`等于1)出现在构成画面中,如果由`flip_flag`的适当值指示,优选在水平方向(即,关于垂直轴)翻转右手画面(见图31)。当从顶部到底部空间交织(`basic_spatial_interleaving_type_id`等于2)出现在构成画面中,如果由`flip_flag`的合适值指示,优选在垂直方向(即,关于中心水平轴)翻转底部画面(见图32)。虽然在此已经描述翻转第二构成画面,但设想另一示例性实施例可以涉及翻转第一画面,诸如以顶部到底部交织的顶部画面或以并排交织中的左侧画面。当然,为了处理`flip_flag`的这些附加自由度,可能必须增加值的范围或引入关于其的另外语义。

[0340] 还设想附加实施利可能允许在水平和垂直方向二者中翻转画面。例如,当如图2所示每个象限一个视图(即,画面)、在单个画面中一起平铺四个视图时,能够使左上象限包括没有被翻转的第一视图,而右上象限包括仅水平翻转的第二视图,而左下象限包括仅垂直翻转的第三视图,而右下象限包括水平和垂直二者翻转的第四视图。通过以该方式的平铺或交织,可以看出在视图之间的接口的边界具有以下的大的似然性:画面中相邻视图的边界的两侧上具有共同的场景内容。该翻转类型可以提供附加的压缩效率。在本公开的范围设想该翻转类型的指示。

[0341] 还设想可以采用表7中的语法用于处理2D加深度交织。可以开发例如接受这样的格式作为输入的商业显示器。在表8中阐述这样的应用的示例性语法。很多语义已经被如上地定义。下面仅仅描述新引入的语义。

[0342] 表8

	描述符
<code>spatially_interleaved_pictures(payloadSize) {</code>	
<code> spatially_interleaved_pictures_id</code>	<code>ue(v)</code>
<code> spatially_interleaved_pictures_cancel_flag</code>	<code>u(1)</code>
<code> if(!spatially_interleaved_pictures_cancel_flag) {</code>	
<code> basic_spatial_interleaving_type_id</code>	<code>u(8)</code>
<code> semantics_id</code>	<code>u(1)</code>
<code> upsample_conversion_horizontal_flag</code>	<code>u(1)</code>
<code> upsample_conversion_vertical_flag</code>	<code>u(1)</code>
<code> spatially_interleaved_pictures_repetition_period</code>	<code>ue(v)</code>
[0343] <code> if(basic_spatial_interleaving_type_id == 1 or</code>	
<code> basic_spatial_interleaving_type_id == 2) {</code>	
<code> flip_flag</code>	<code>u(1)</code>
<code> }</code>	
<code> if(semantics_id == 1) {</code>	
<code> camera_parameter_set()</code>	
<code> }</code>	
<code> }</code>	
<code> additional_extension_flag</code>	<code>u(1)</code>
<code>}</code>	

[0344] `semantics_id`替代现有示例性实施例中的`stereo_pair_flag`并且被用于指示什么关系意图用于两个交织的画面,换言之,这两个画面在物理上意味着什么。值0指示两个构成画面形成图像的左视图和右视图。值1指示第二画面代表第一构成画面的对应深度图。图34示出示例性单个画面,其中,构成视频和深度图像(例如,2D+Z)被并排交织。应理解可以由`basic_spatial_interleaving_type_id`的特定值指示其他交织方法。值2指示两个交织画面之间的关系未被指定。大于2的值可以被用于指示附加关系。

[0345] `camera_parameter_set`指示与相机相关的场景的参数。参数可以出现在同一SEI消息中或者参数可以在另一SEI消息中递送,或者它们可以出现在诸如SPS、PPS、码片首标等之类的其他语法中。相机参数一般地包括至少焦距、基线距离、(多个)相机位置、`Znear`(场景和相机之间的最小距离)以及`Zfar`(场景和相机之间的最大距离)。相机参数集还可以包括每个相机的全部参数集,包括3x3本征(intrinsic)矩阵、3x3旋转矩阵以及3D变换矢量。例如在呈现时并且还可能在编码中使用该相机参数。

[0346] 当使用SEI消息或与表8中示出的语法相当的其他语法时,可能能够避免需要视频和深度之间系统级的同步的需要。该结果是因为视频和相关联的深度已经被平铺为一个单个帧。

[0347] 图33示出画面的层深度视频(LDV)格式。在该图中,四个画面被并排并且从顶部至底部地交织,以形成复合(composite)画面。左上象限画面代表中间视图层,而右上象限画面代表中间深度层。类似地,左下象限画面代表遮挡视图层,而右下象限画面代表遮挡深度层。图33中示出的格式可以用作商业自动立体显示器的输入格式

[0348] 使用类似前面实施利中示出的语法的语法来指示LDV的出现。首先,`semantics_id`

的语义被扩展,以引入如下的LDV选项:

[0349] 表9

	<code>spatially_interleaved_pictures(payloadSize) {</code>	C	描述符
	<code> spatially_interleaved_pictures_id</code>	5	uc(v)
	<code> spatially_interleaved_pictures_cancel_flag</code>	5	u(1)
[0350]	<code> if(!spatially_interleaved_pictures_cancel_flag) {</code>		
	<code> basic_spatial_interleaving_type_id</code>	5	u(8)
	<code> semantics_id</code>	5	u(1)
	<code> upsample_conversion_horizontal_flag</code>	5	u(1)
	<code> upsample_conversion_vertical_flag</code>	5	u(1)
	<code> spatially_interleaved_pictures_repetition_period</code>	5	uc(v)
	<code> if(basic_spatial_interleaving_type_id == 1 or basic_spatial_interleaving_type_id == 2) {</code>		
	<code> flip_flag</code>	5	u(1)
	<code> }</code>		
[0351]	<code> if(semantics_id == 1 semantics_id == 3) {</code>		
	<code> camera_parameter_set()</code>		
	<code> }</code>		
	<code> }</code>		
	<code> additional_extension_flag</code>	5	u(1)
	<code>}</code>		

[0352] semantics_id指示画面之间的关系,即,交织画面相互之间在物理上意味着什么。

[0353] 值从0至2意味着两个画面如上所定义地相关。值0指示两个画面形成左视图和右视图。该值还可以指示视图是场景的立体视图。值1指示第二视图代表第一画面的对应深度图,即,一组2D+Z画面。值2指示未指定两个交织画面之间的关系。

[0354] 值3指示四个分量画面被交织并且它们对应于例如图33所示的LDV代表的四个分量画面。可以利用大于3的值用于附加的关系指示。

[0355] 在此设想的附加关系指示包括但不限于:多个视图是多个2D+Z画面集(也已知为多视图加深度(MVD))的指示;多个视图代表两个LDV画面集(已知为深度增强立体(DES))中的图像的指示。

[0356] 当semantics_id等于3并且basic_spatial_interleaving_type_id等于1(并排)或2(从顶部到底部),四个分量画面被如图33所示地交织。

[0357] 当semantics_id等于3并且basic_spatial_interleaving_type_id等于0时,如果相对视图0的视图1、2和3的偏移附加地由特定的指示(诸如经由语法中的新语义)发信号通知,则如图22所例示地,如示例1(要素2210),或示例2(要素2220),交织四个画面。设想当semantics_id等于3时,如图22的示例1(要素2210)所示的,为LDV画面执行交织。还设想当semantics_id等于4时,如图22的示例2(要素2220)所示的,为LDV画面执行交织。以上已经关于图22描述了具有偏移和不具有偏移的LDV的交织。

[0371]
$$H = \begin{bmatrix} c_x & 0 & \dots & c_1 & 1 & c_1 & \dots & c_x & 0 & 0 & \dots \\ 0 & c_x & 0 & \dots & c_1 & 1 & c_1 & \dots & c_x & 0 & \dots \\ \dots & & & & & & & & & & \\ \dots & 0 & c_x & 0 & \dots & c_1 & 1 & c_1 & \dots & c_x & 0 \end{bmatrix}.$$

[0372] 水平上采样之后的输出图像矩阵Y然后表示为:

[0373] $Y^T = HX'^T.$

[0374] 以类似的方式,将示例性垂直上采样处理描述如下。新矩阵A'被定义为类似如下矩阵:

[0375]
$$A' = \begin{bmatrix} D & & 0_{(m-k)k} \\ 0_{(m-k)k} & I_{m,n} & \\ & & D \end{bmatrix}.$$

[0376] 然后经由下面所示张量积操作将图像Y转换为新图像矩阵Y':

[0377] $Y' = (YA') \otimes [1 \ 0].$

[0378] 垂直上采样之后的输出图像矩阵然后表示为:

[0379] $F^T = HY'^T.$

[0380] 矩阵F是水平和垂直上采样转换之后的最终图像矩阵。

[0381] 已经将上面示例中的上采样描绘为水平和垂直二者的操作。设想操作顺序可以逆向,使得在水平上采样之前进行垂直上采样。此外,设想在某些情形中仅仅可以进行一种类型的上采样。

[0382] 为了使用SEI消息将上采样信息传递到解码器,设想某些语义必须被包括在上面呈现的语法内。在下面表10中示出的语法包括用于视频解码器处的上采样的信息。下面的语法已经被删节,以仅仅示出传递上采样信息必需的参数。本领域的技术人员将理解下面示出的语法可以与现有语法的任何一个或多个组合来传递例如与关系指示、方位指示、上采样指示和空间交织指示有关的大量帧打包信息。

[0383] 表10

	Frame_packing_filter(payloadSize) {	描述符
	for (c = 0; c < 3; c++)	
	{	
	number_of_horizontal_filter_parameters[c]	ue(v)
	for (i=0; i < number_of_horizontal_filter_parameters[c]; i++)	
	{	
	h[c][i]	s(16)
[0384]	}	
	number_of_vertical_filer_parameters[c]	ue(v)
	for (i=0; i < number_of_horizontal_filter_parameters[c]; i++)	
	{	
	v[c][i]	s(16)
	}	
	}	
	}	

[0385] 下面将语法元素的语义定义如下：

[0386] number_of_horizontal_filter_parameters[c] 指示色彩分量c的 $(4k-1)$ 个抽头水平滤波器的阶 (order), 其中c是三个不同色彩分量之一。

[0387] h[c][i] 指定在 -2^{15} 到 $2^{15}-1$ 范围的16比特精度的色彩分量c的水平滤波器的第i个系数。滤波器的基准是 2^{16} 。

[0388] number_of_vertical_filter_parameter 指示色彩分量c的 $(4k-1)$ 个抽头垂直滤波器的阶。

[0389] v[c][i] 指定在 -2^{15} 到 $2^{15}-1$ 范围的16比特精度的色彩分量c的垂直滤波器的第i个系数。滤波器的基准是 2^{16} 。

[0390] 对于该语法, 假设已知滤波器类型呈现在解码器中。然而, 设想除了或替代上面描述的2维对称FIR滤波器, 可以使用各种不同类型的上采样滤波器。这样的其他上采样滤波器包括但不限于, 任何内插滤波器设计, 诸如双线性滤波器、三次滤波器、样条滤波器、维纳 (Wiener) 滤波器或最小平方滤波器。当其他滤波器类型可用时, 设想上面的语法应该扩展至包括一个或多个语义来指示关于上采样滤波器类型的信息。

[0391] 本领域的技术人员将理解可以基于诸如下采样滤波器信息等之类的各种输入、在编码器导出上采样滤波器参数用于包括在SEI消息中。此外, 还将理解这样的信息可以简单地提供编码器用于包括在SEI消息中。

[0392] 还设想表10中示出的语法可以被修改, 以替代上采样滤波器参数而传送下采样滤波器参数。在该实施例中, 向解码器指示下采样滤波器参数给予解码应用基于下采样滤波器参数来确定最优上采样滤波器参数的能力。类似的语法和语义可以用于用信号通知下采样滤波器。可以包括语义, 以指示参数表示下采样滤波器还是上采样滤波器。

[0393] 如果解码器要确定最优的上采样滤波器, 则解码器可以以本领域已知的任何方式

来进行这样的确定。此外,解码器可以确定不是最优的或者不一定是最优的上采样滤波器。在解码器处的确定可以基于处理限制或显示参数考虑等。作为进一步的示例,解码器可以初始地选择多个不同类型的上采样滤波器并且然后最终选择用于产生通过某些解码器标准确定的、对解码器最优的结果的一个滤波器。

[0394] 在图35中示出用于示出与上采样滤波器信息的指示有关的示例性编码技术的流程图并且简略描述如下。初始地,确定编码器配置并且创建高级语法。下采样滤波器下采样画面(例如,视图),用于打包为帧。下采样的画面在空间上交织为帧。上采样滤波器参数由编码器确定并且直接提供给编码器。在一个实施例中,从下采样滤波器导出上采样滤波器参数,诸如类型、大小和系数值。对于每个色彩分量(例如,YUV、RGB),确定水平和垂直上采样滤波器参数的数量和值。这一点在图35中用三个循环示出:第一循环用于分量、第二循环(嵌入在第一循环)用于分量的水平滤波器参数、以及第三循环(嵌入在第一循环)用于分量的垂直滤波器参数。上采样滤波器参数然后被写入SEI消息。SEI消息然后与视频图像比特流分开地(波段外)或者与视频图像比特流一起地(波段内)被发送至解码器。当需要时,编码器组合消息与打包帧序列。

[0395] 对比图35中描绘的编码方法,图37中示出的替代实施例示出了下采样滤波器参数替代上采样滤波器参数而包括在SEI消息中。在该实施例中,上采样滤波器参数不从编码器导出并且不随打包帧传递至解码器。

[0396] 在图36中示出了用于示出与上采样滤波器信息的指示有关的示例性解码技术的流程图并且简略描述如下。初始地,解码器接收SEI消息以及其他相关消息并且由解码器读取语法。分析SEI消息以确定在其中包括的信息,包括采样信息、交织信息等。在该示例中,确定每个色彩分量的上采样滤波器信息被包括在SEI消息中。提取垂直和水平上采样滤波器信息,以获得每个水平和垂直上采样滤波器系数的数量和值。在图36中用三个循环示出该提取:第一循环用于分量、第二循环(嵌入在第一循环)用于分量的水平滤波器参数、以及第三循环(嵌入在第一循环)用于分量的垂直滤波器参数。SEI消息然后被存储并且打包帧序列被解码以获得其中打包的画面。然后使用恢复的上采样滤波器参数来上采样画面,以将画面还原到它们原始的全分辨率。

[0397] 对比图36中描绘的解码方法,图38中示出的替代实施例示出了下采样滤波器参数替代上采样滤波器参数被包括在SEI消息中。在该实施例中,上采样滤波器参数不从编码器导出并且不随打包帧传递至解码器。因此,解码器使用接收的下采样滤波器参数来导出上采样滤波器参数,以用于将完全的原始分辨率还原到从打包视频帧的交织中提取的画面。

[0398] 图39示出根据本原理的实现方式、可以应用本原理的示例性视频传输系统2500。视频传输系统2500可以例如是首端(head-end)或者是传输系统,其使用诸如例如卫星、线缆、电话线或地面广播之类的多种介质中的任一个传送信号。传输可以通过因特网或某些其他网络提供。

[0399] 视频传输系统2500能够生成并且传递具有深度的压缩视频。通过生成(多个)编码信号来实现这一点,所述编码信号包括深度信息或者能够用于在例如可以具有解码器的接收器端合成深度信息的信息。

[0400] 视频传输系统2500包括编码器2510和能够传送编码信号的传送器2520。编码器2510接收视频信息并且生成具有深度的(多个)编码信号。编码器2510可以具有子模块,包

括组装单元,例如用于接收各条信息并且将其组装(assembly)为用于存储或者传送的结构化格式。各条信息可以包括例如编码或者未编码的视频、编码或未编码的深度信息以及编码或未编码的元素,诸如例如运动矢量、编码模式指示符和语法元素。

[0401] 传送器2520可以例如适配于传送具有一个或多个表示编码画面和/或与其有关的信息的比特流的节目信号。典型的传送器执行以下的一个或多个功能,诸如例如提供纠错编码、交织信号中的数据、随机化信号中的能量,和/或将信号调制到一个或多个载波上。传送器可以包括天线(未示出)或者与天线对接。因此,传送器2520的实现方式可以包括或限于调制器。

[0402] 图40示出根据本原理的实施例、可以应用本原理的示例性视频接收系统2600。视频接收系统2600可以被配置为通过诸如例如卫星、线缆、电话线或地面广播之类的多种介质来接收信号。该信号可以通过因特网或某些其他网络接收。

[0403] 视频接收系统2600可能例如是蜂窝电话、计算机、机顶盒、电视机或接收编码视频并且例如向用户提供用于显示的解码视频或者提供用于存储的解码视频的其他设备。由此,视频接收系统2600可以向例如电视机屏幕、计算机监视器、计算机(用于存储、处理或显示)或某些其他存储、处理或显示设备提供其输出。

[0404] 视频接收系统2600能够接收并且处理包括视频信息的视频内容。视频接收系统2600包括能够接收诸如例如在本申请的实现方式中描述的信号之类的编码信号的接收器2610,以及能够解码接收信号的解码器2620。

[0405] 接收器2610可以例如适配于接收具有表示编码画面的多个比特流的节目信号。典型的接收器执行以下一个或多个功能,诸如例如接收调制和编码的数据信号、从一个或多个载波解调数据信号、去随机化信号中的能量、去交织信号中的数据,和/或纠错解码信号。接收器2610可以包括天线(未示出)或者与天线接口。因此,接收器2610的实现方式可以包括或限于解调器。解码器2620输出包括视频信息和深度信息的视频信号。

[0406] 图41示出根据本原理的实施例、可以应用本原理的示例性视频处理设备2700。视频处理设备2700可能例如是机顶盒或者接收编码视频并且例如向用户提供用于显示的解码视频或者提供用于存储的解码视频的其他设备。由此,视频处理设备2700可以向电视机、计算机监视器或计算机或者其他处理设备提供其输出。

[0407] 视频处理设备2700包括前端(FE)设备2705和解码器2710。前端设备2705可以例如是接收器,其被适配以接收具有表示编码画面的多个比特流的节目信号,并且从多个比特流选择用于解码的一个或多个比特流。典型的接收器执行以下一个或多个功能,诸如例如接收调制和编码的数据信号、解调数据信号、解码数据信号的一个或多个编码(例如,信道编码和/或源编码)、和/或纠错数据信号。前端设备2705可以例如从天线(未示出)接收节目信号。前端设备2705向解码器2710提供接收的数据信号。

[0408] 解码器2710接收数据信号2720。数据信号2720可以包括例如一个或多个高级视频编码(AVC)、可分级视频编码(SVC)或多视图视频编码(MVC)相容的流。

[0409] 解码器2710解码接收的信号2720的全部或者部分并且通过作为输出的解码的视频信号2730。解码视频2730被提供至选择器2750。设备2700还包括接收用户输入2770的用户接口2760。用户接口2760基于用户输入2770向选择器2750提供画面选择信号2780。画面选择信号2780和用户输入2770指示用户期望显示可用的解码数据中的多个画面、序列、可

分级版本、视图或其他选择中的哪些。选择器2750提供(多个)选择画面作为输出2790。选择器2750使用画面选择信息2780来选择作为输出2790提供的解码视频2730中哪些画面。

[0410] 在各种实现方式中,选择器2750包括用户接口2760,而在其他实现方式中不需要用户接口2760,这是因为选择器2750直接接收用户输入2770,不需要执行单独的接口功能。选择器2750例如可以以软件实现或可以实现为集成电路。在一个实现方式中,选择器2750与解码器2710合并,并且在另一个实现方式中,解码器2710、选择器2750和用户接口2760全部被集成。

[0411] 在一种应用中,前端2705接收各种电视节目广播并且选择一个用于处理。选择一个节目是基于对于期望观看的频道的用户输入。虽然在图41中未示出用户对于前端设备2705的输入,但是前端设备2705接收用户输入2770。前端2705接收广播并且通过解调广播频谱的相关部分、并且解码解调节目的任何外部编码来处理期望的节目。前端2705向解码器2710提供解码节目。解码器2710是包括设备2760和2750的集成单元。由此解码器2710接收作为用户提供的、节目中期望观看的视图的指示的用户输入。解码器2710解码选择的视图,以及来自其他视图的所需任何参考画面,并且提供解码视图2790用于在电视上显示(未示出)。

[0412] 继续上面的应用,用户可能期望切换被显示的视图并且可以然后向解码器2710提供新输入。从用户接收“视图改变”之后,解码器2710解码旧视图和新视图二者,以及在旧视图和新视图中间的任何视图。即,解码器2710解码在物理上位于拍摄旧视图的相机和拍摄新视图的相机之间的相机拍摄的任何视图。前端设备2705还接收标识旧视图、新视图和之间的视图的信息。例如可以通过具有关于视图位置的信息的控制器(图41中未示出)或者解码器2710提供这样的信息。其他实现方式可以使用具有与前端设备集成的控制器的前端设备。

[0413] 解码器2710提供所有这些解码视图作为输出2790。后处理器(图41中未示出)在视图之间进行内插,以提供从旧视图到新视图的平滑过渡,并且向用户显示该过渡。过渡到新视图之后,后处理器向解码器2710以及前端装置2705通知(通过未示出的一个或多个通信链路)仅仅需要新视图。此后,解码器2710仅仅提供新视图作为输出2790。

[0414] 系统2700可以用于接收图像序列的多个视图,并且呈现单个视图用于显示,并且以平滑的方式在各个视图之间切换。平滑方式可以涉及视图之间的内插以移动到另一视图。此外,系统2700可以允许用户旋转对象或者场景或者以其它方式看到对象或者场景的三维表示。物体的旋转例如可以与视图到视图的移动、以及视图之间的内插对应,来获得视图之间的平滑过渡或者简单地获得三维表示。即,用户可以“选择”内插视图作为要显示的“视图”。

[0415] 如在此呈现的,本申请中描述的实现方式和特征可以用于编码视频、编码深度和/或编码其他类型数据的背景中。此外,这些实现方式和特征可以用于或适配用于以下背景中:H.264/MPEG-4 AVC(AVC)标准、具有MVC扩展的AVC标准、具有SVC扩展的AVC标准、3DV标准,和/或具有其他标准(现有或未来的),或者在不涉及标准的背景中。因此,应理解,根据AVC操作的本申请中描述的特定实现方式不意图限制于AVC,并且可以适配用于AVC之外。

[0416] 如上所述,实现方式可以使用各种技术发信号通知或者指示信息,包括但不限于以下:SEI消息、码片首标、其他高级语法、非高级语法、波段外信息、数据流数据和隐式信

令。虽然可以在具体背景中描述在此描述的实现方式,但是这样的描述绝不应该被认为将特征和构思限制于这样的实现方式或背景中。

[0417] 各种实现方式涉及解码。如在本申请中使用的“解码”可以包括例如在接收编码序列上执行的处理的全部或部分,以产生最终适合于显示的输出。这样的处理可以包括由解码器典型执行的处理,诸如例如熵解码、逆变换、逆量化、差分解码(differential decoding)之类。这样的处理还可以或者可替代地包括本申请中描述的各种实现方式的解码器执行的处理,诸如例如从平铺(打包)画面提取画面、确定使用的上采样滤波器以及然后上采样画面,并且将画面翻转回到其意图的方位。

[0418] 在本申请中描述的各种实现方式将多个画面组合成单个画面。组合画面的处理可能涉及例如基于例如画面之间的关系选择画面、空间交织、采样、以及改变画面的方位。因此,描述组合处理的信息可能包括例如关系信息、空间交织信息、采样信息以及方位信息。

[0419] 在描述各种实现方式中引用空间交织。如在本申请中使用的“交织”还称作例如“平铺”和“打包”。交织包括各种类型的交织,包括例如,画面的并排并置(juxtaposition)、画面的从顶部到底部并置、并排并且从顶部到底部地组合(例如,组合4个画面)、逐行交替、逐列交替以及各种像素级(也称作按像素方式或基于像素的)方案。

[0420] 此外,如在这里所使用的,词语“画面”和“图像”可互换地使用并且指代例如静止画面的全部或部分或来自视频序列的画面的全部或部分。如已知的,画面可以是帧或场。此外,如在这里使用的,画面还可以是帧的子集,诸如例如帧的上半部、单个宏块、交替列、交替行或周期性的像素。作为另一个实施例,深度画面例如可以是完全深度图或者仅仅包括例如对应视频帧的单个宏块的深度信息的部分深度图。

[0421] 本说明书中对本原理的“一个实施例”或“实施例”或者“一个实现方式”或“实现方式”以及其他变型的引用意味着在本原理的至少一个实施例中包括与该实施例相联系地描述的特定的特征、结构、特性等。因而,贯穿本说明书的多处出现的短语“在一个实施例中”或“在实施例中”或者“在一个实现方式中”或“在实现方式中”以及任何其他变型的出现不一定都指代相同的实施例。

[0422] 将要理解,如下“/”、“和/或”、和“至少一个”的任何一种的使用,例如,在“A/B”、“A和/或B”、和“A和B的至少一个”的情形下,意图在包含只选择第一列出选项(A),或只选择第二列出选项(B),或选择两个选项(A和B)。作为进一步的例子,在“A、B和/或C”和“A、B和C的至少一个”和“A、B或C的至少一个”的情形下,这样的措词意图包含只选择第一列出选项(A),或只选择第二列出选项(B),或只选择第三列出选项(C),或只选择第一和第二列出选项(A和B),或只选择第一和第三列出选项(A和C),或只选择第二和第三列出选项(B和C),或选择所有三个选项(A和B和C)。本领域及相关领域的普通技术人员容易明白,这可以扩展到许多列出项目。

[0423] 此外,本申请或其权利要求可以引用“确定”各种信息。确定信息可以包括以下的一个或多个:例如估计信息、计算信息、预测信息或从存储器检索信息的。

[0424] 类似地,“访问”意图是宽泛的术语。访问信息片断可以包括任何操作,例如使用、存储、发送、传送、接收、检索、修改或提供信息。

[0425] 应理解在图中示出的元素可以实现为各种形式的硬件、软件或其组合。优选地,在可能包括处理器、存储器和输入/输出接口的一个或多个合适的编程的通用设备上将这些

要素实现为硬件和软件的组合。此外,在此描述的实现方式可以实现为例如方法或处理、装置或者软件程序。即使仅仅在单个形式的实现方式的上下文中讨论(例如,仅仅讨论方法),讨论的特征的实现方式还可以实现为其他形式(例如,装置或程序)。可以如上提及地实现装置。在例如一般指代包括例如计算机、微处理器、集成电路或可编程逻辑器件的处理设备的诸如例如处理装置之类的装置中可以实现该方法。处理设备还包括通信设备,诸如例如计算机、蜂窝电话、便携式/个人数字助理(“PDA”)以及方便终端用户之间通信信息的其他设备。

[0426] 在此描述的各种处理和特征的实现方式可以体现在多种不同装备或应用中,特别是例如与数据编码和解码相关联的装备或应用中。装备的示例包括视频编码器、视频解码器、视频编解码器、万维网服务器、机顶盒、膝上型电脑、个人计算机、蜂窝电话、PDA、其他通信设备、个人记录设备(例如,PVR、计算机运行记录软件、VHS记录设备)、可携式摄像机、因特网或其他通信链路上的数据流以及视频点播。应该清楚,该装备可以是移动的,甚至安装在机动车中。

[0427] 另外,这些方法可以通过处理器执行指令来实现,这样的指令可以存储在处理器可读介质中,诸如例如集成电路、软件载体或者其他存储装置,诸如例如硬盘、致密盘、随机存取存储器(“RAM”)或者只读存储器(“ROM”)中。这些指令可以形成有形地体现在处理器可读介质上的应用程序。应该清楚,处理器可以包括具有例如执行处理的指令的处理器可读介质。可以将该应用程序上载到包括任何合适架构的机器,并由其执行。优选地,在具有诸如一个或多个中央处理单元(“CPU”)、随机存取存储器(“RAM”)、以及输入/输出(“I/O”)接口之类的硬件的计算机平台上实现该机器。该计算机平台还可以包括操作系统和微指令代码。在此描述的各种处理和功能可以是由CPU可以执行的微指令代码的一部分、或者应用程序的一部分、或者是其的任何组合。此外,诸如附加数据存储单元和打印单元之类的各种其它的外设单元可与该计算机平台连接。

[0428] 对于本领域的普通技术人员明显的是,实现方式还可以产生被格式化为携带可以例如被存储或发送的信息的信号。该信息可以包括,例如,执行一种方法的指令,或由描述的实现方式之一产生的数据。这样的信号可以被格式化为例如电磁波(例如,使用频谱的射频部分)或基带信号。格式化可以包括例如编码数据流、产生语法并且用编码数据流和语法调制载波。信号携带的信息可以是,例如,模拟信息或数字信息。信号可以在已知的多种不同有线或无线链路上传送。

[0429] 还要理解的是因为优选地以软件实现附图中描绘的某些构成系统组件和方法,所以取决于对本原理进行编程的方式,该系统组件或处理功能块之间的实际连接可能有所不同。给定在此的教导,相关领域的普通技术人员将能够设想到本原理的这些和类似的实现方式或配置。

[0430] 已经描述了许多实现方式。不过,应该明白,可以做出各种修改。例如,可以组合、补充、修改或删除不同实现方式的要素来产生其他实现方式。另外,本领域的普通技术人员应该明白,可以用其他结构和处理来替代所公开的那些,并且所得实现方式将以至少基本相同的(多个)方式执行至少基本相同的(多个)功能,以实现与所公开实现方式至少基本相同的(多个)结果。具体地,虽然在这里已经参考附图描述了例示性的实施例,但应当理解,本原理不限于这些精确的实施例,而相关领域普通技术人员可以在其中实现各种改变和修

改,而不脱离本原理的范围或精神。因此,这些和其他实现方式可以通过本申请来设想,并且在所附权利要求的范围之内。

100

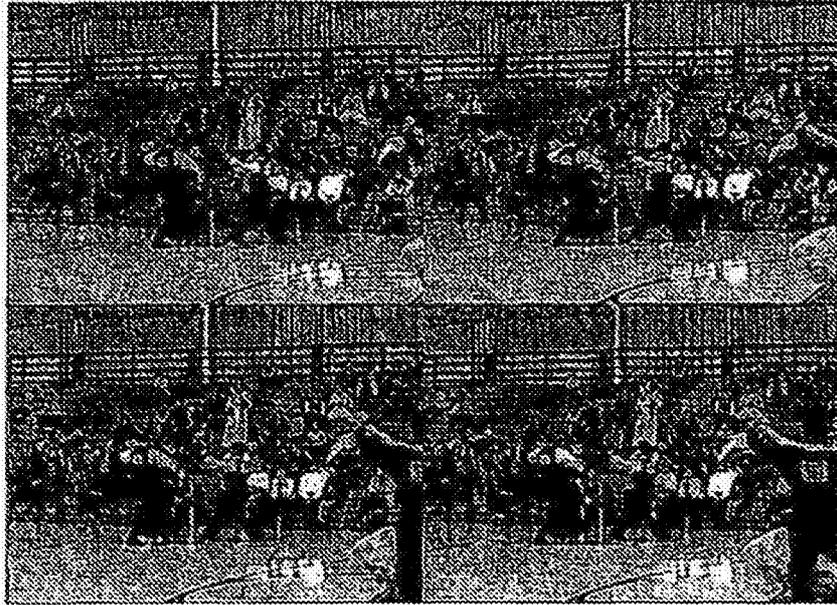


图1

200

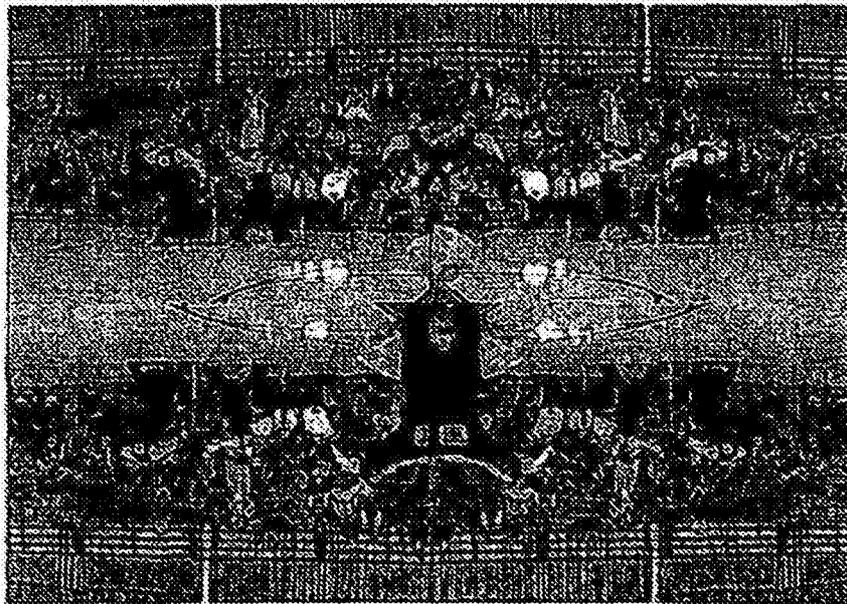


图2

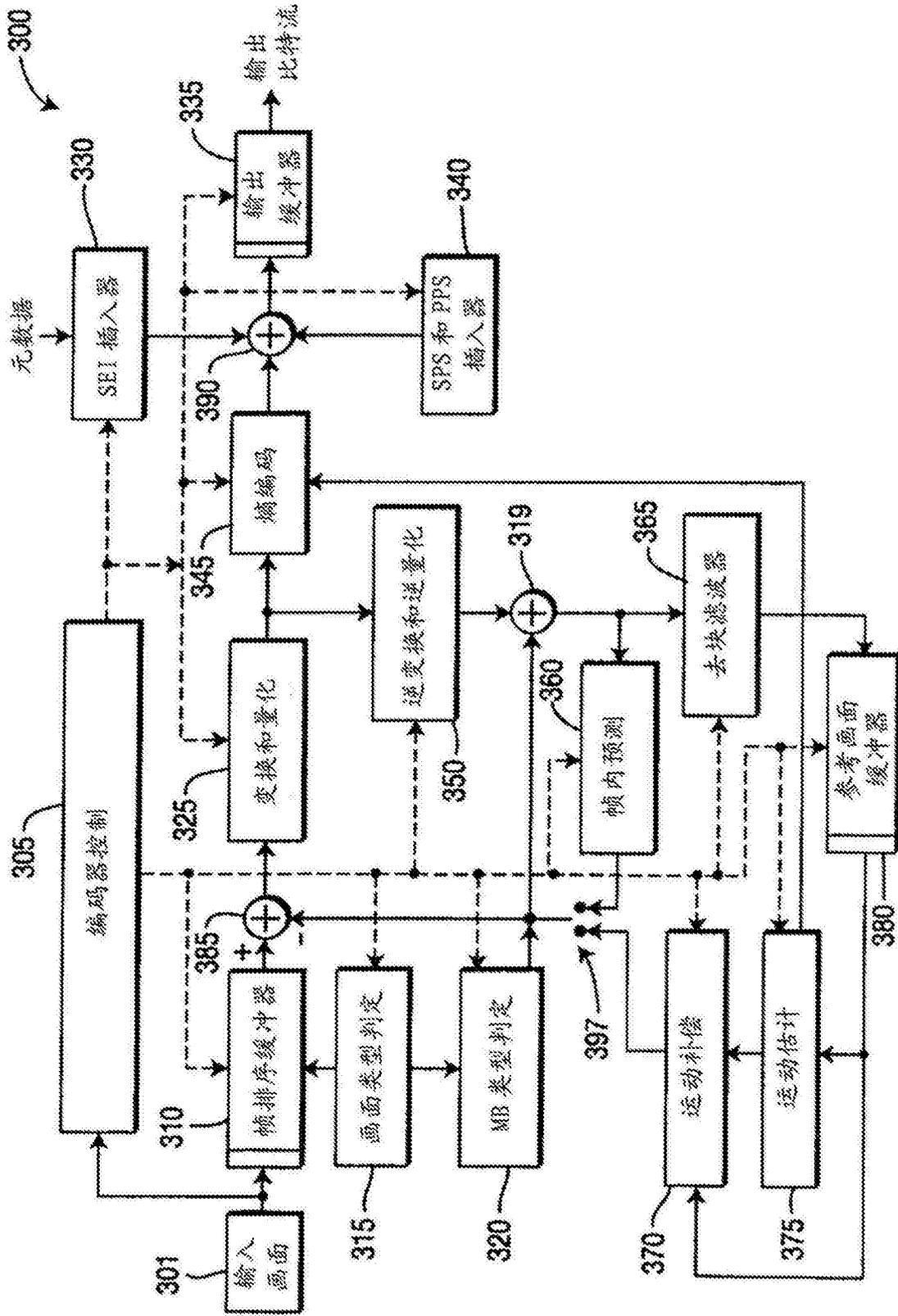


图3

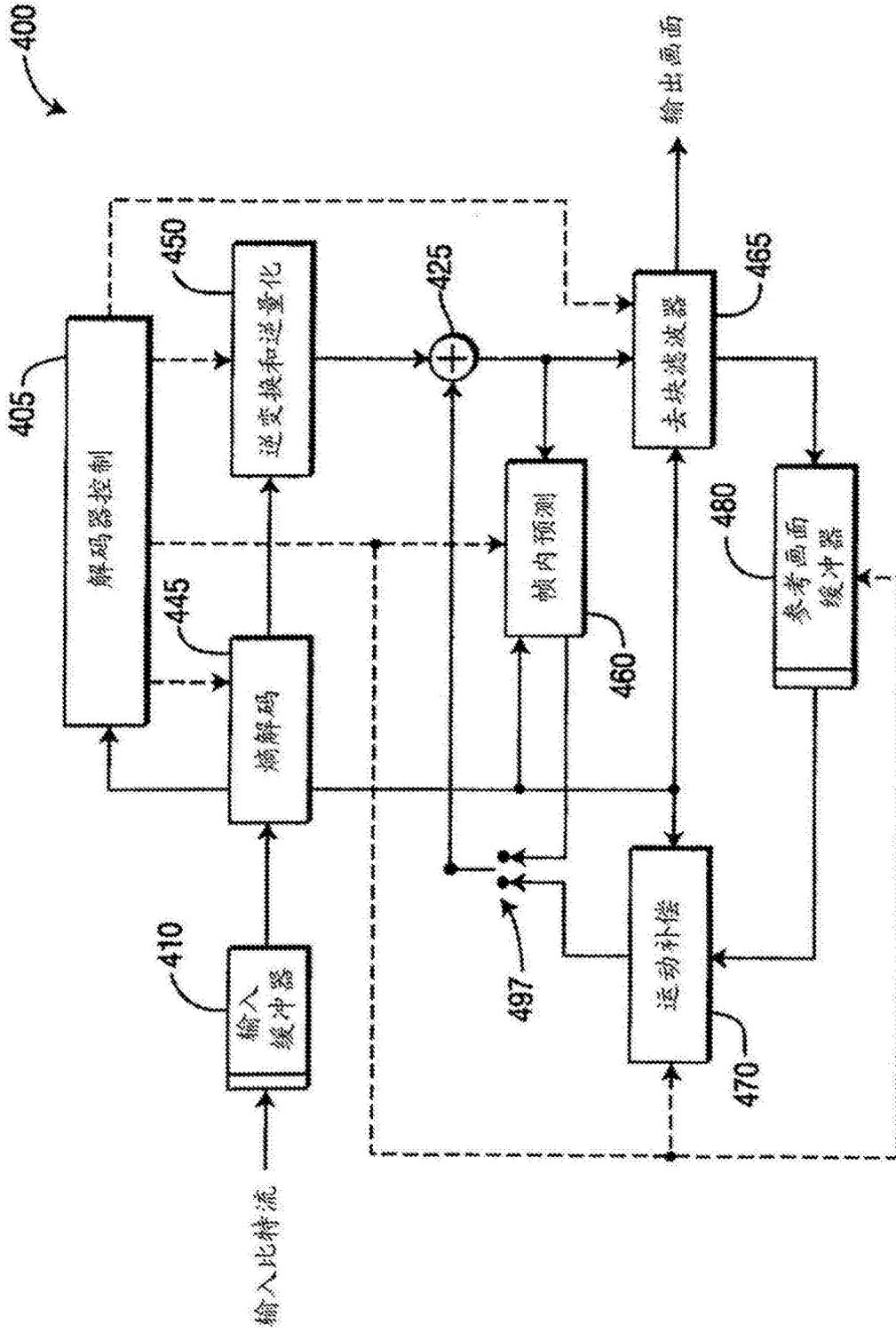


图4

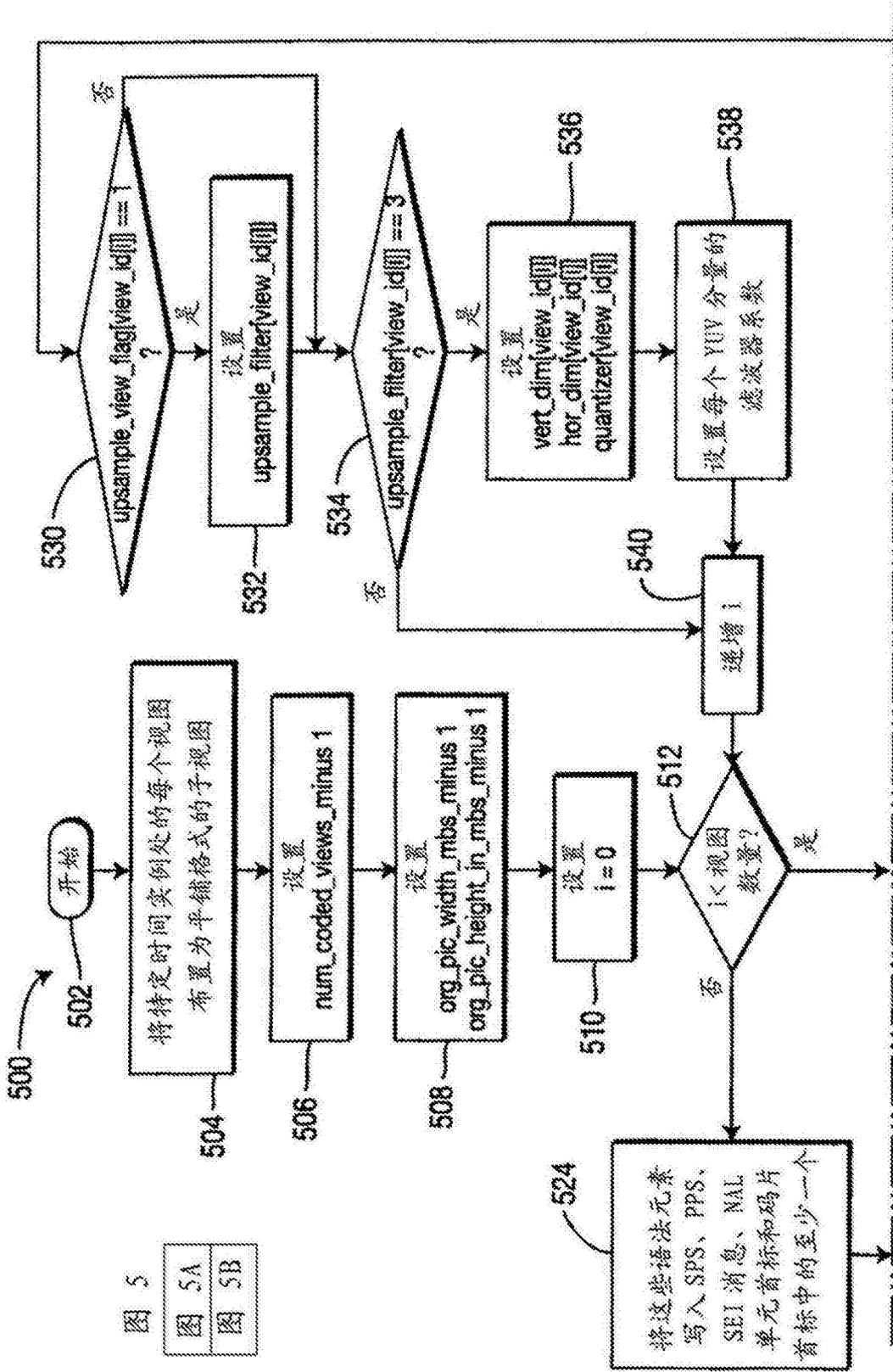


图 5
图 5A
图 5B

图 5A

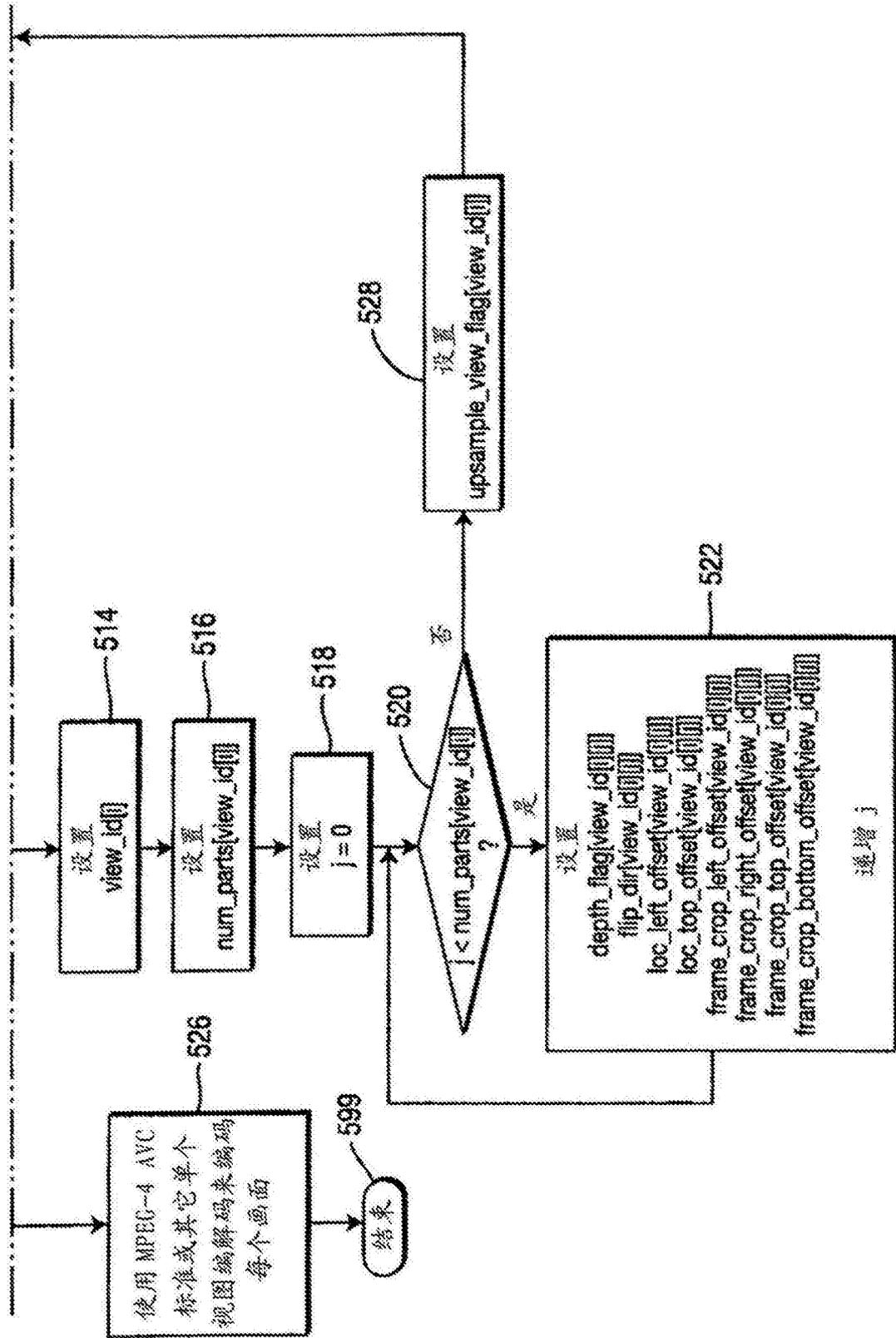


图5B

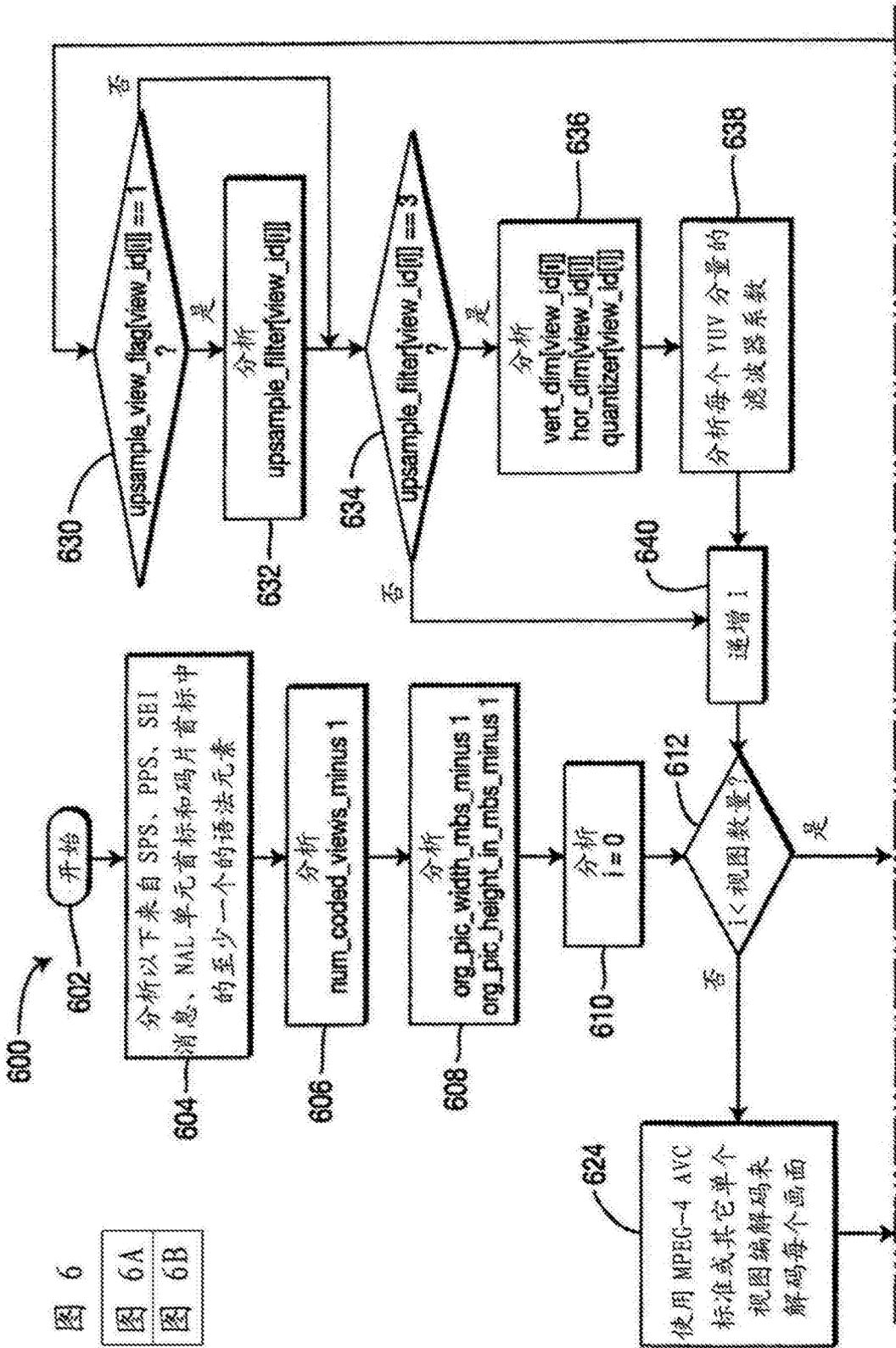


图 6

图 6A

图 6B

图 6A

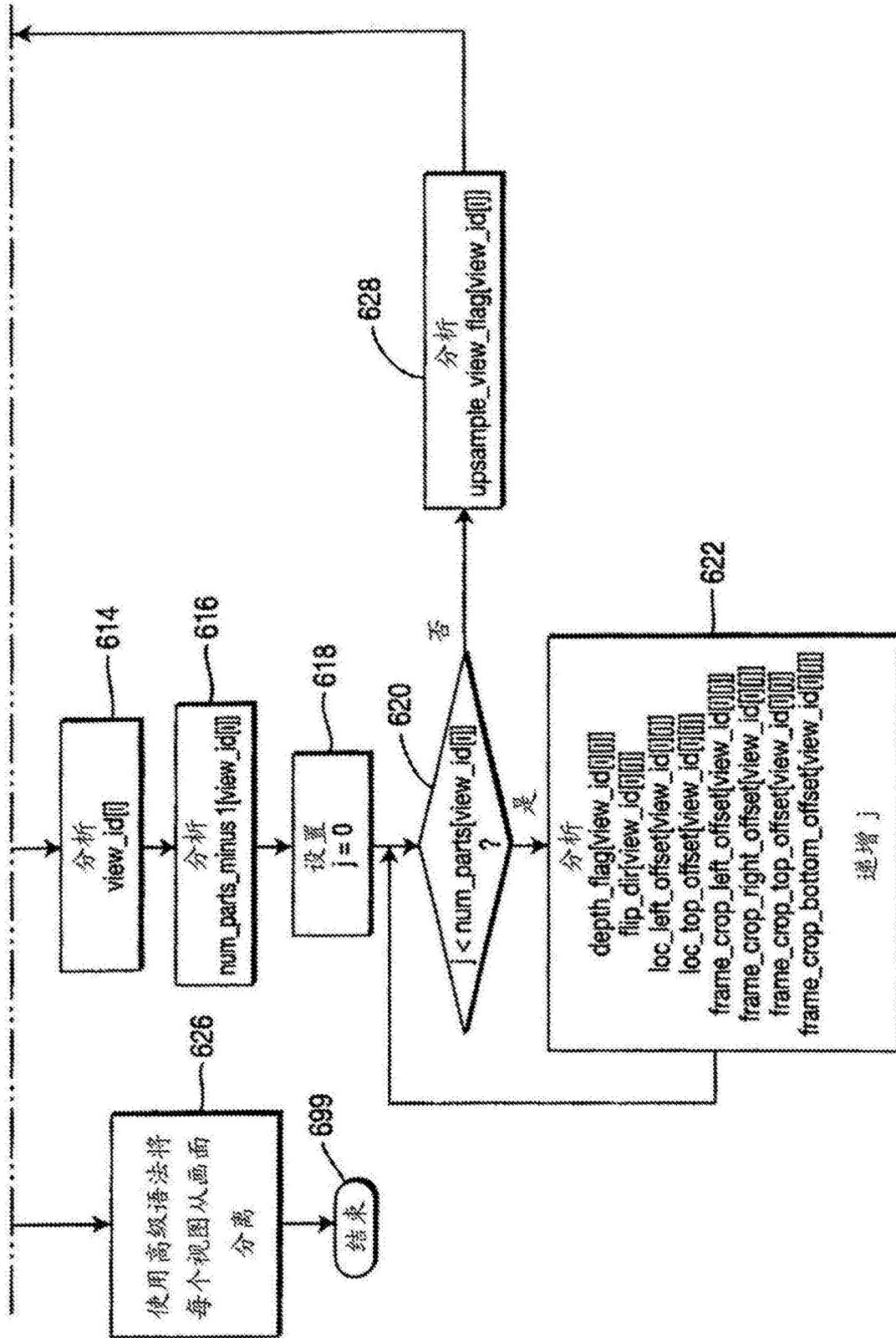


图6B

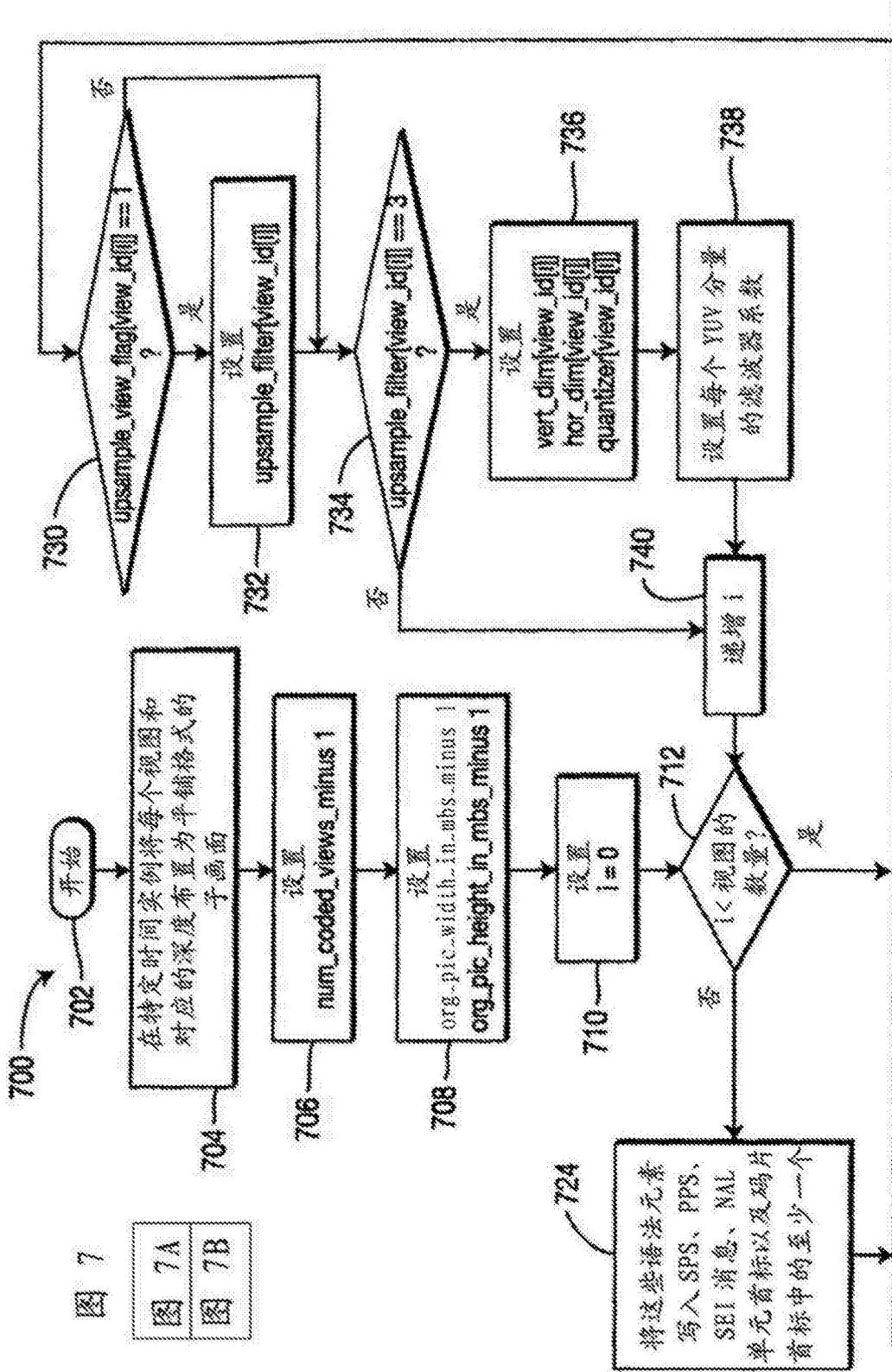


图 7A

图 7

图 7A
图 7B

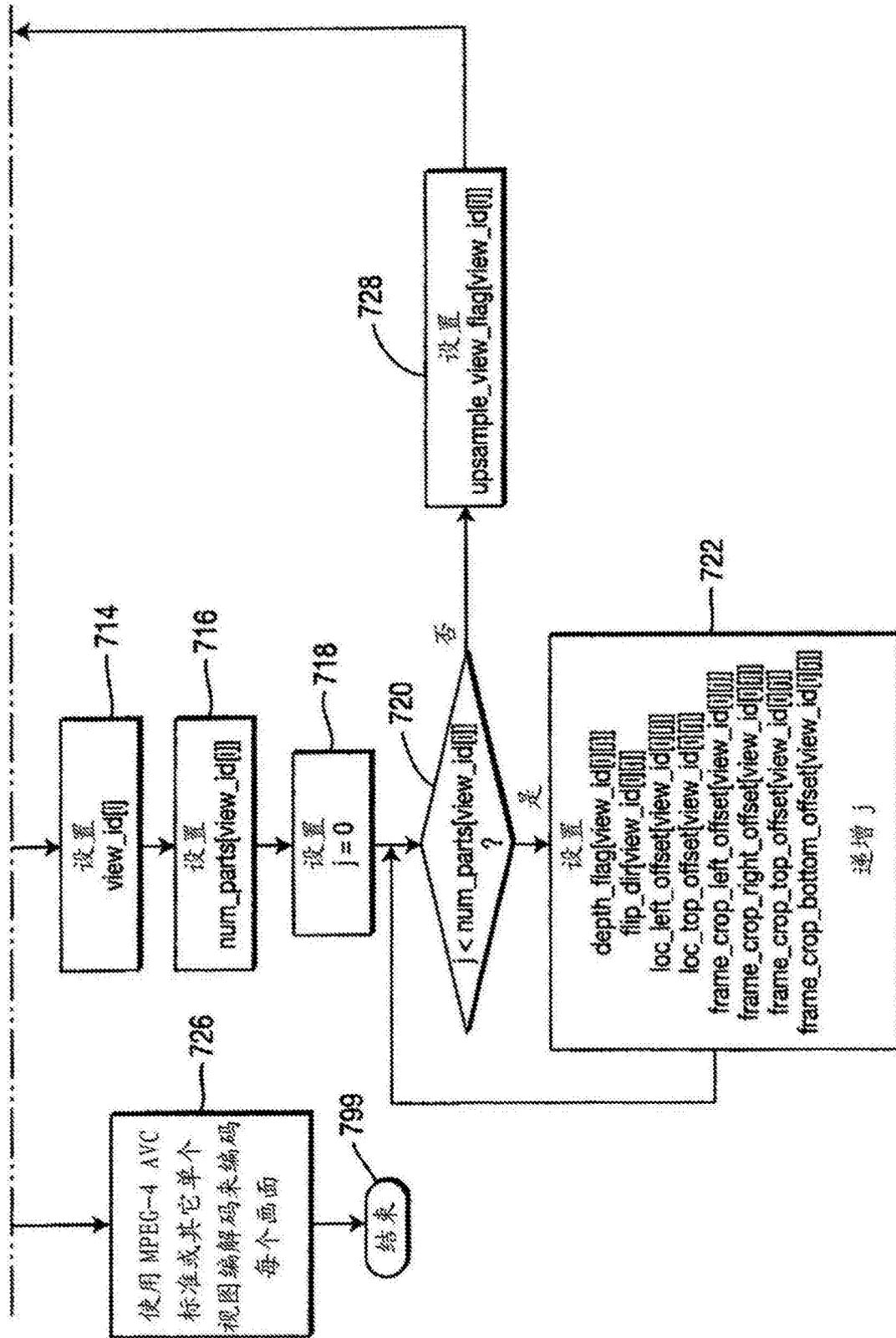


图7B

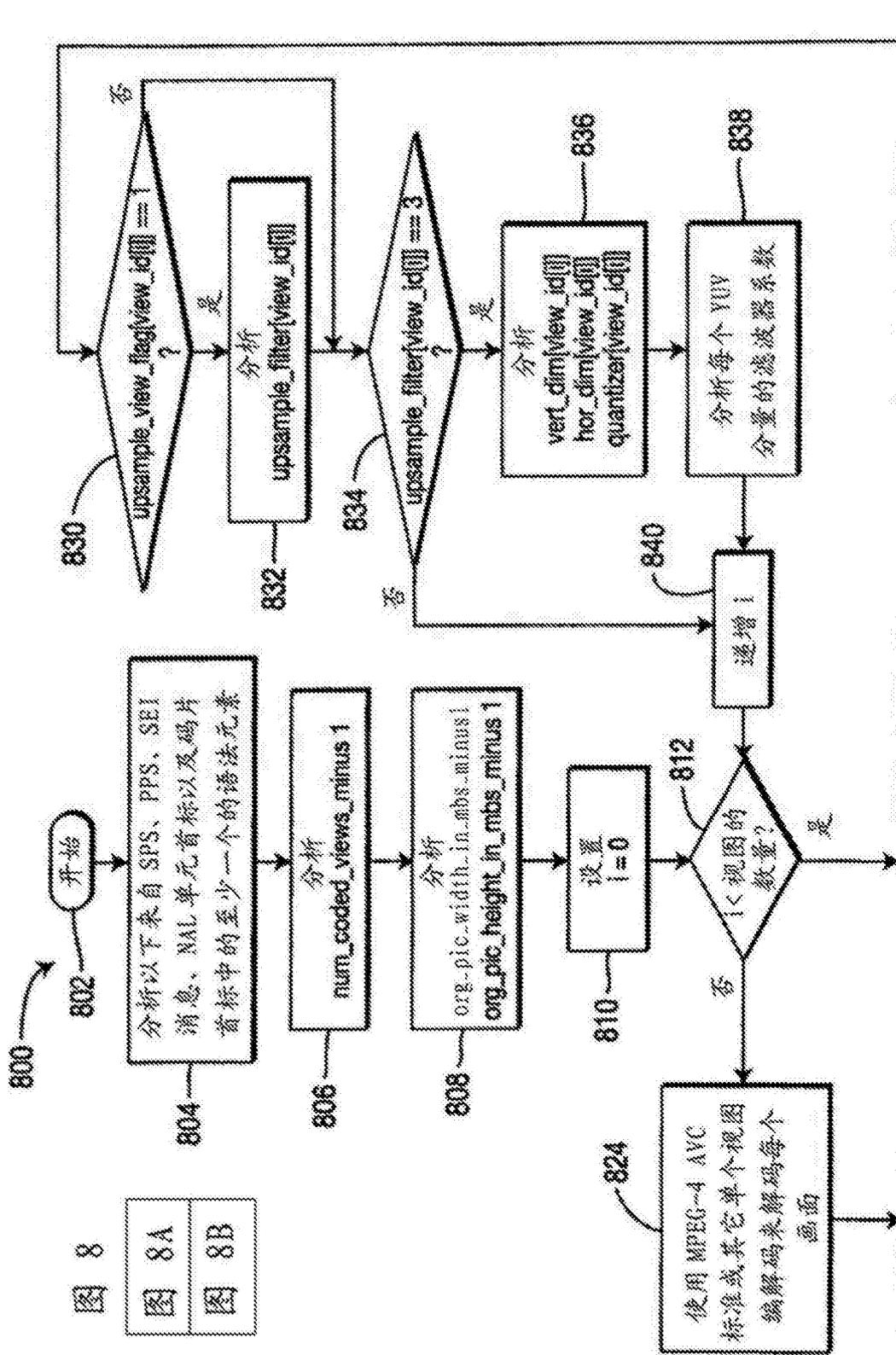


图 8

图 8A
图 8B

图 8A

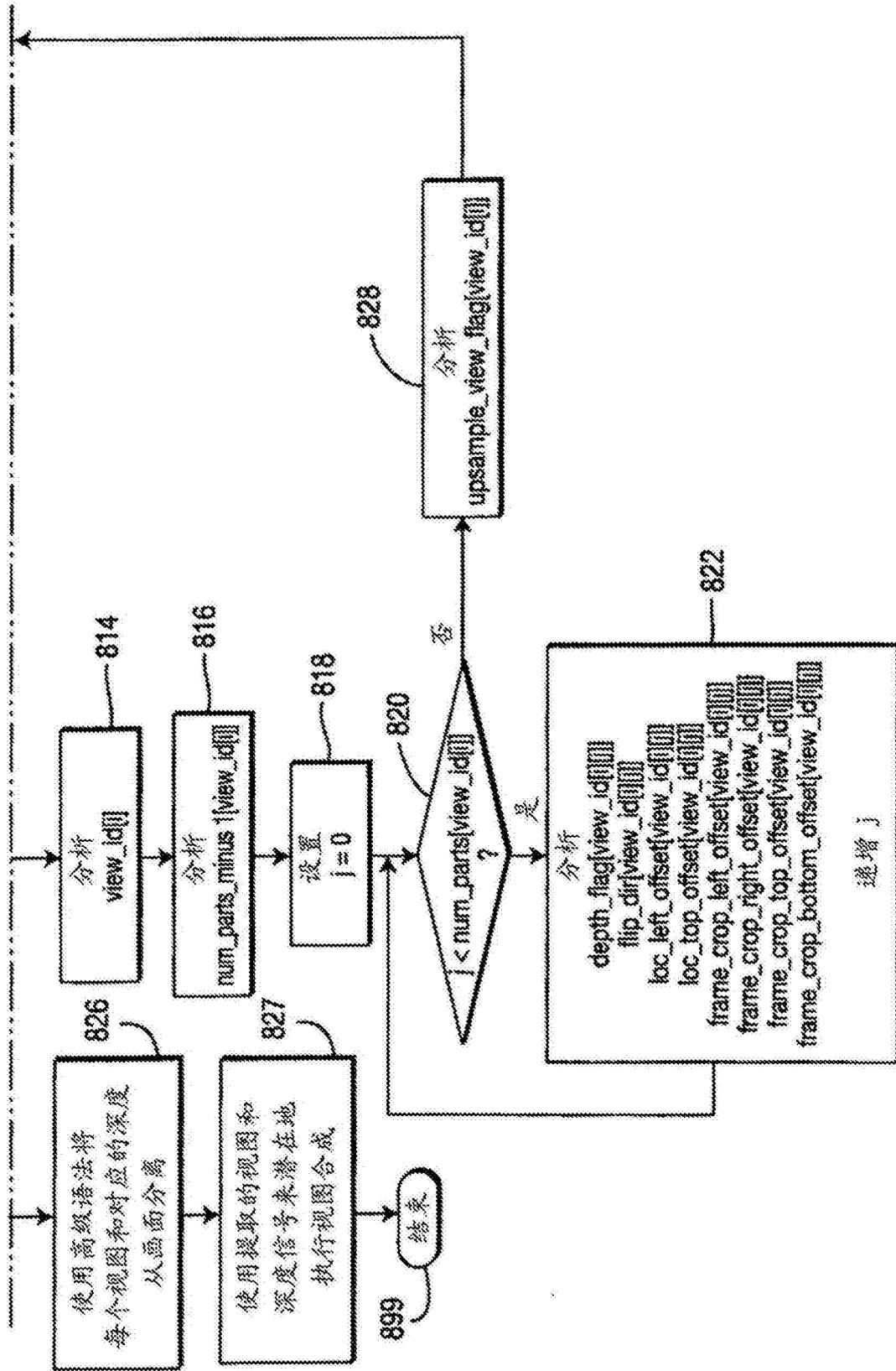


图8B

900



图9

1000

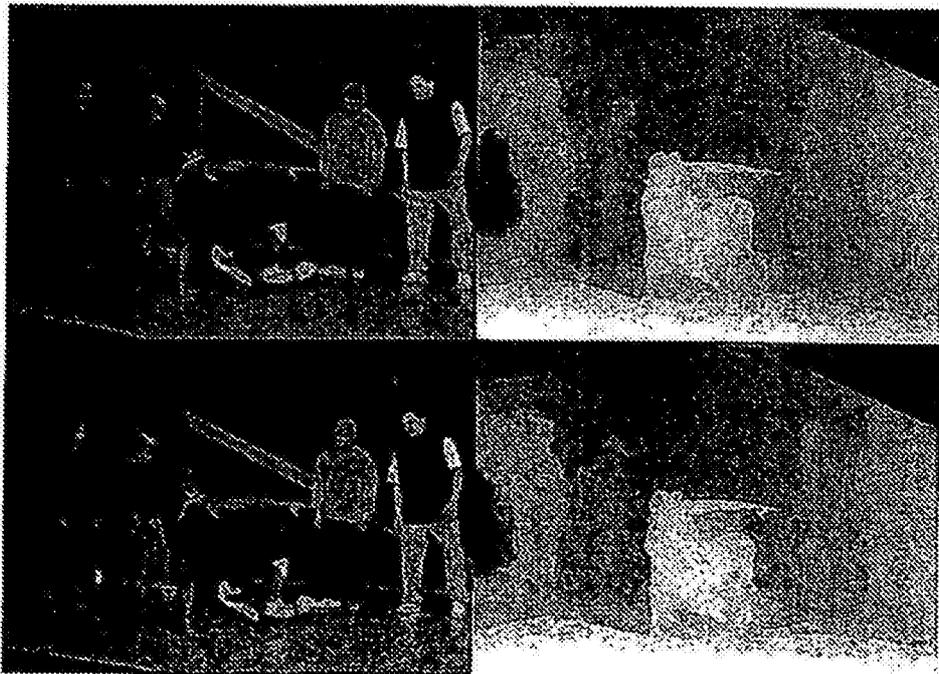


图10

1100

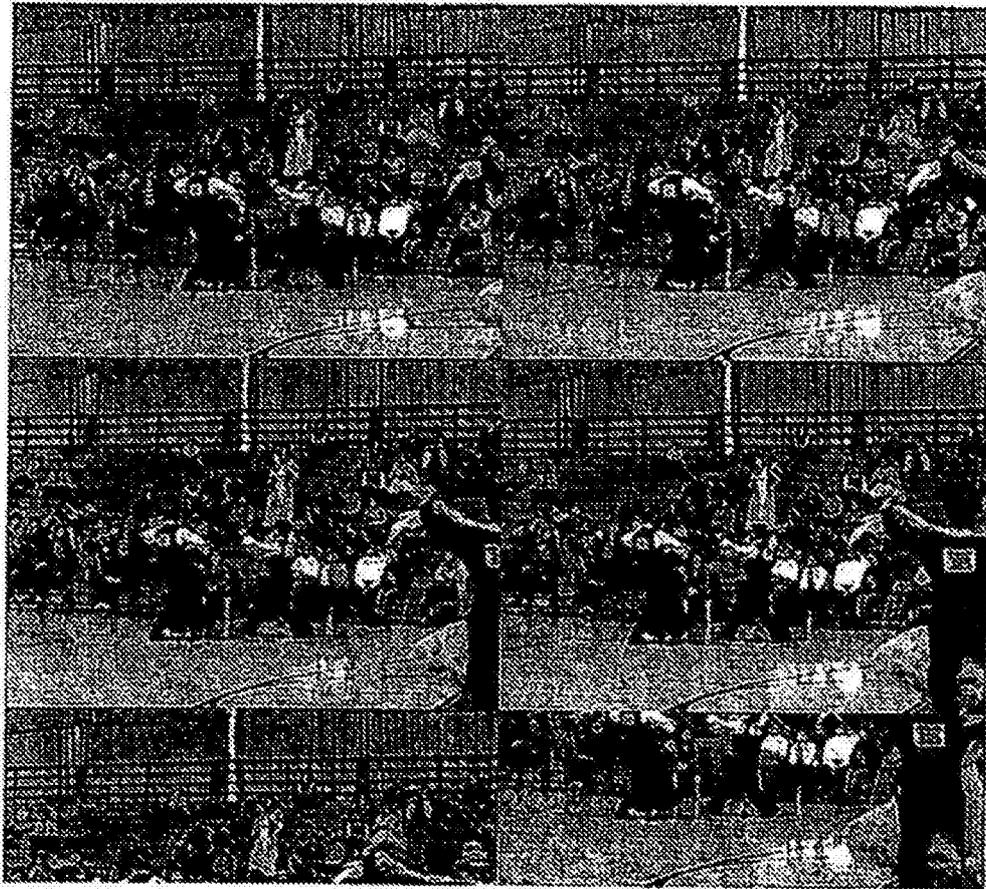


图11

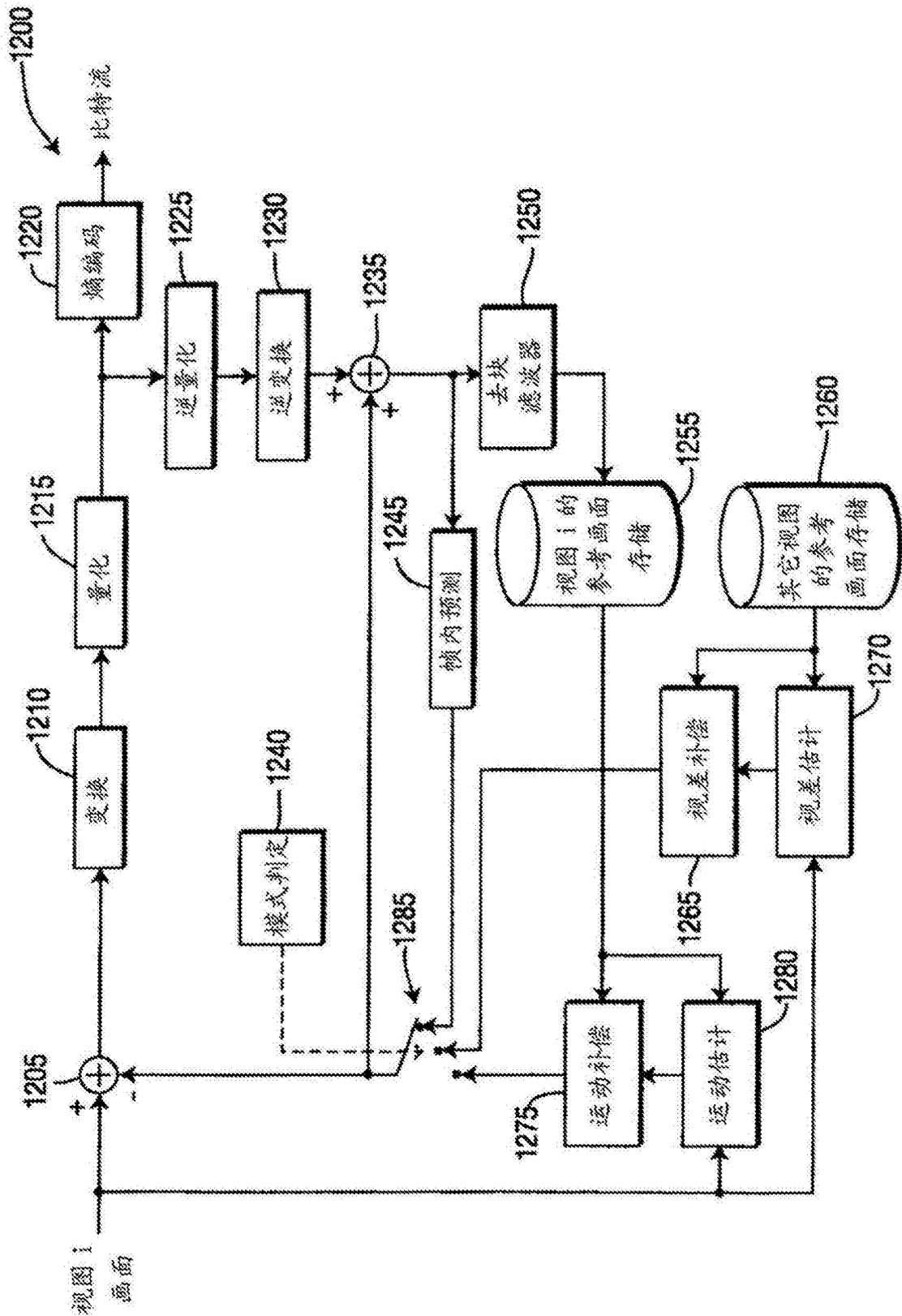


图12

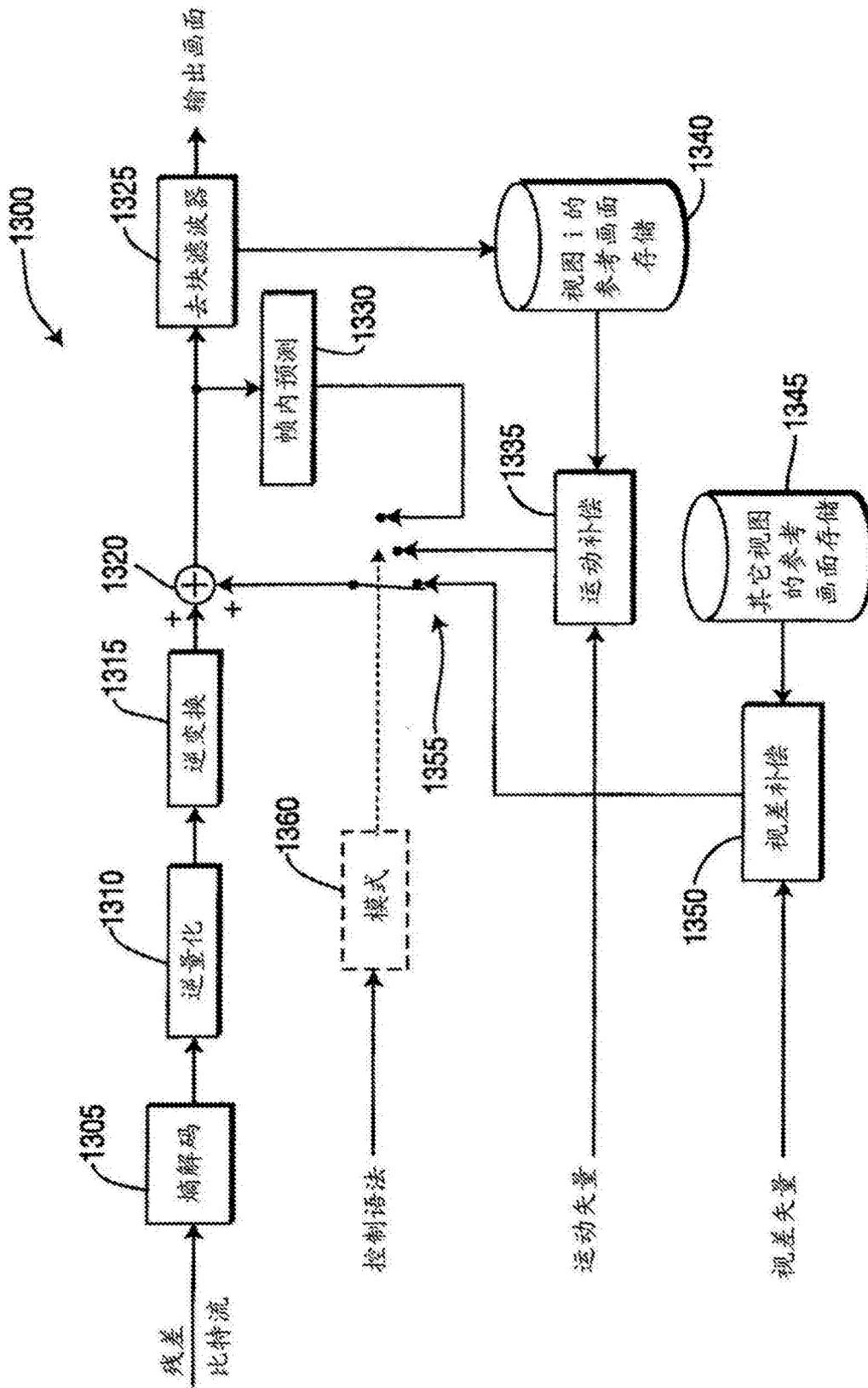


图13

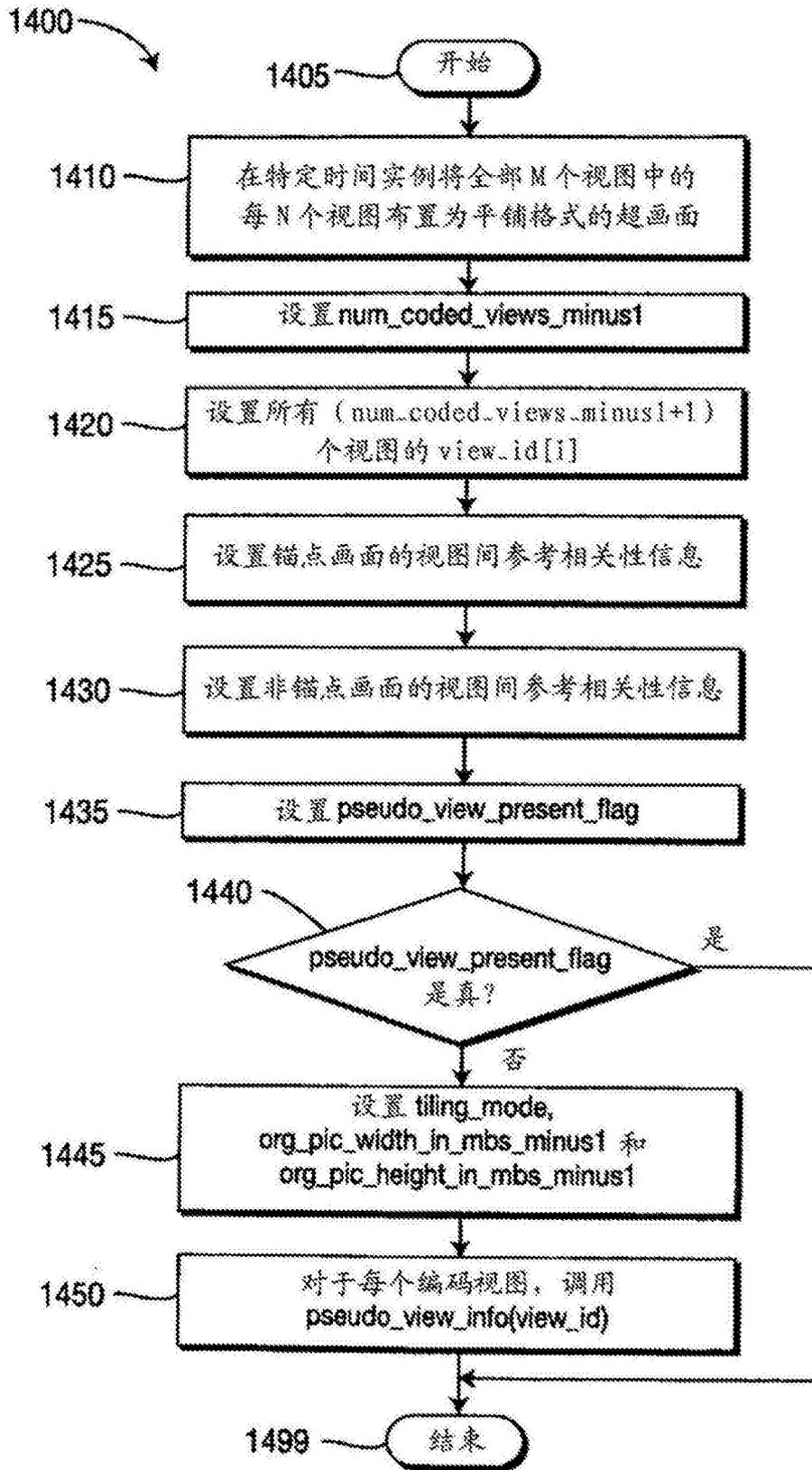


图14

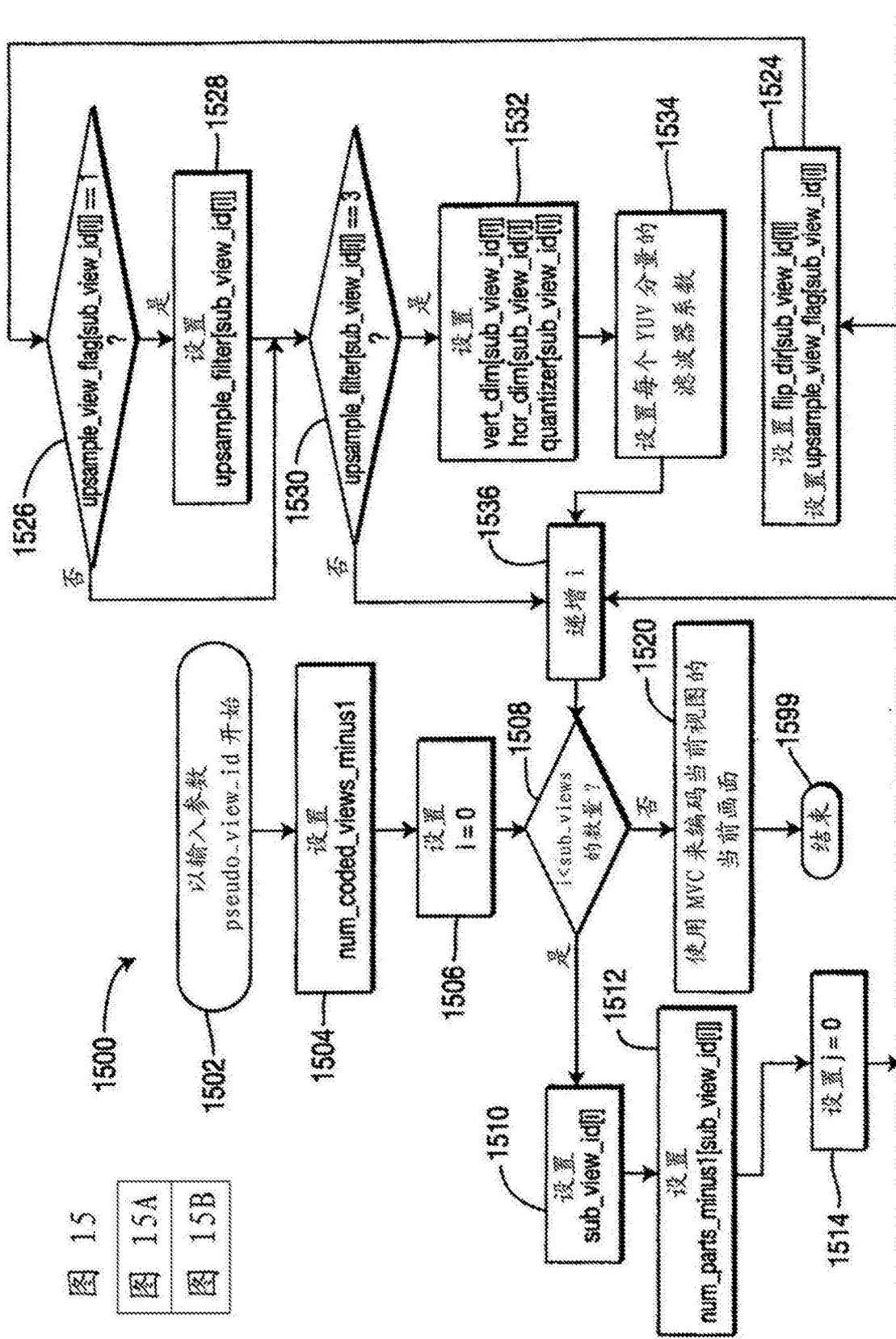


图 15A

图 15

图 15A

图 15B

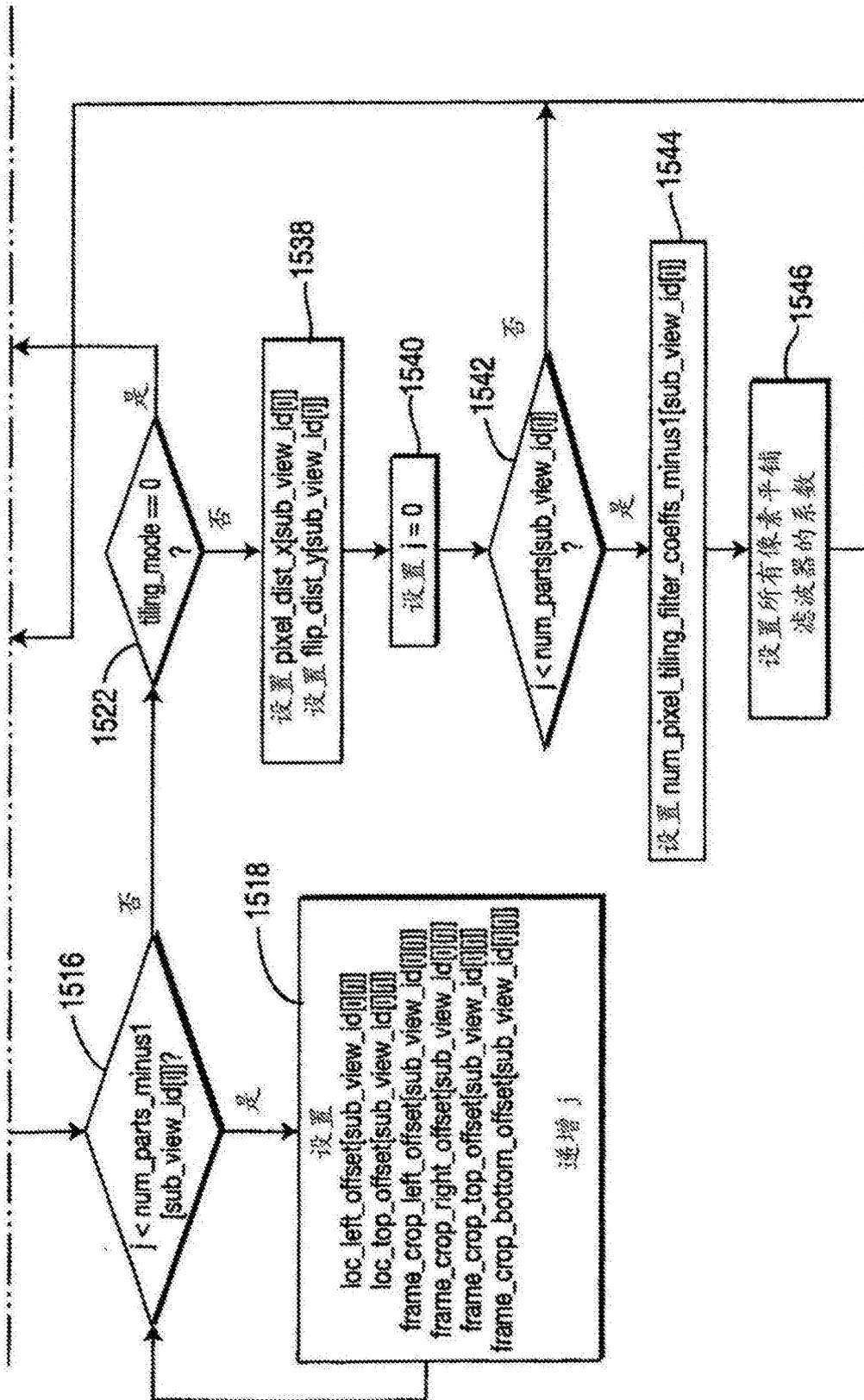


图15B

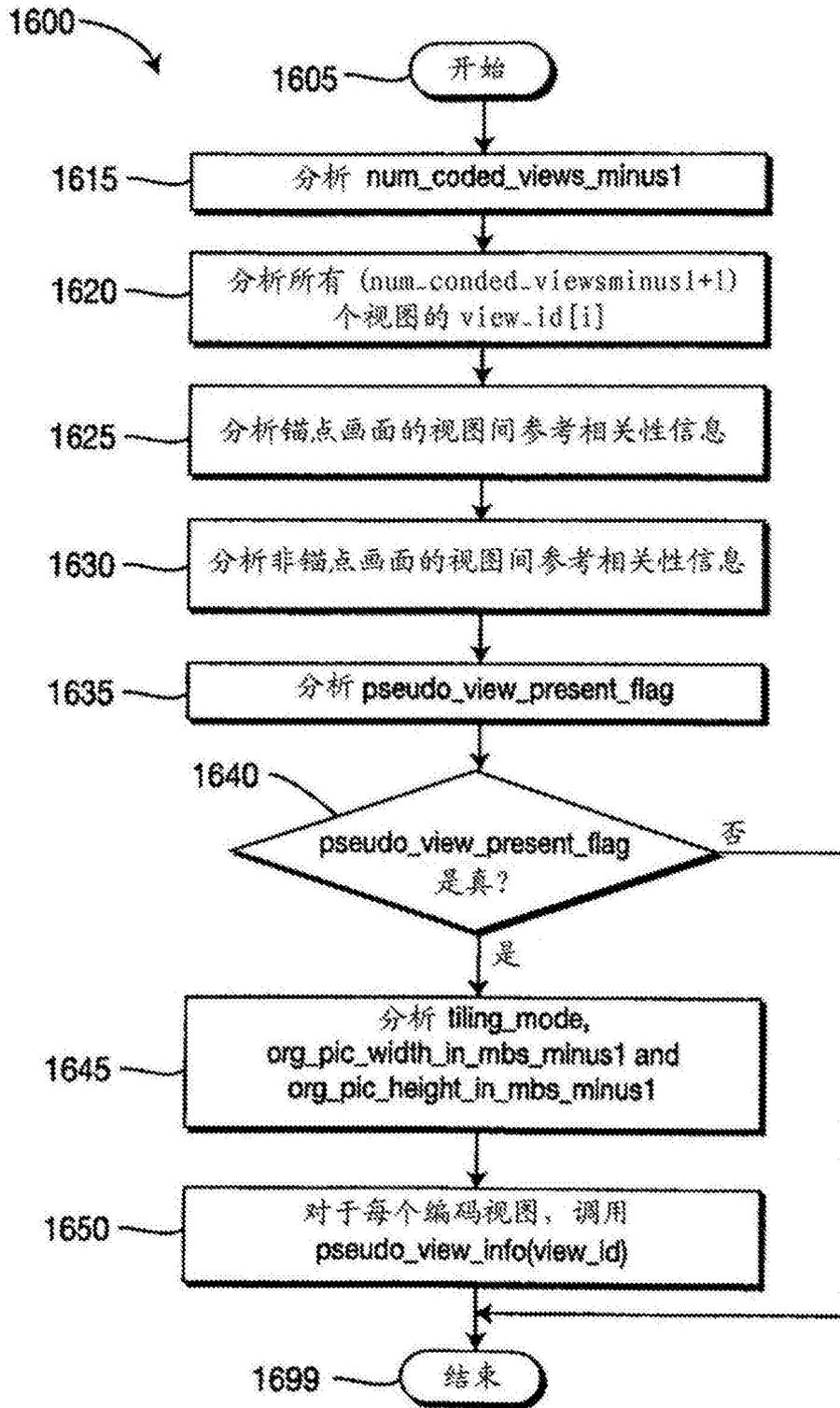


图16

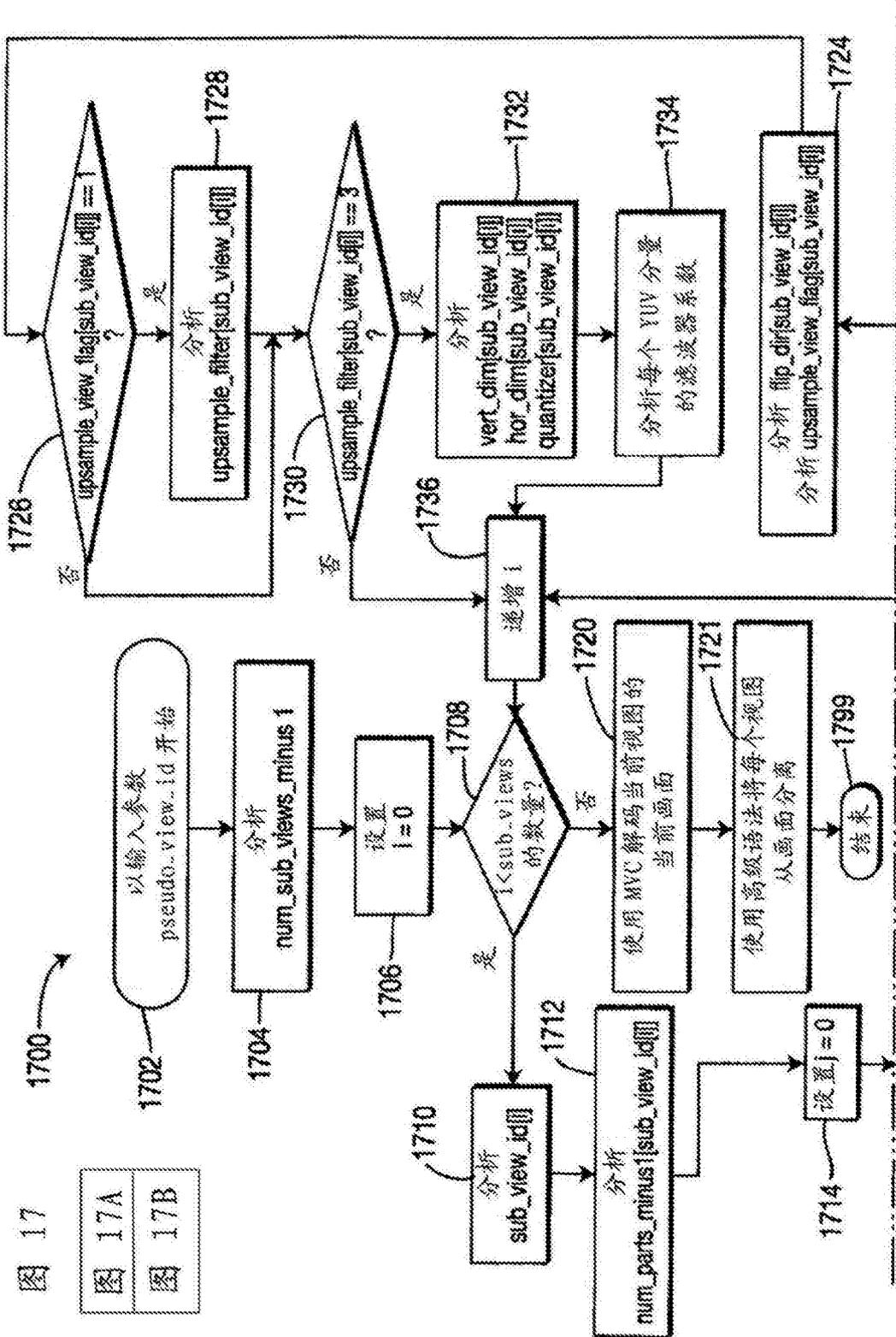


图 17A

图 17

图 17A
图 17B

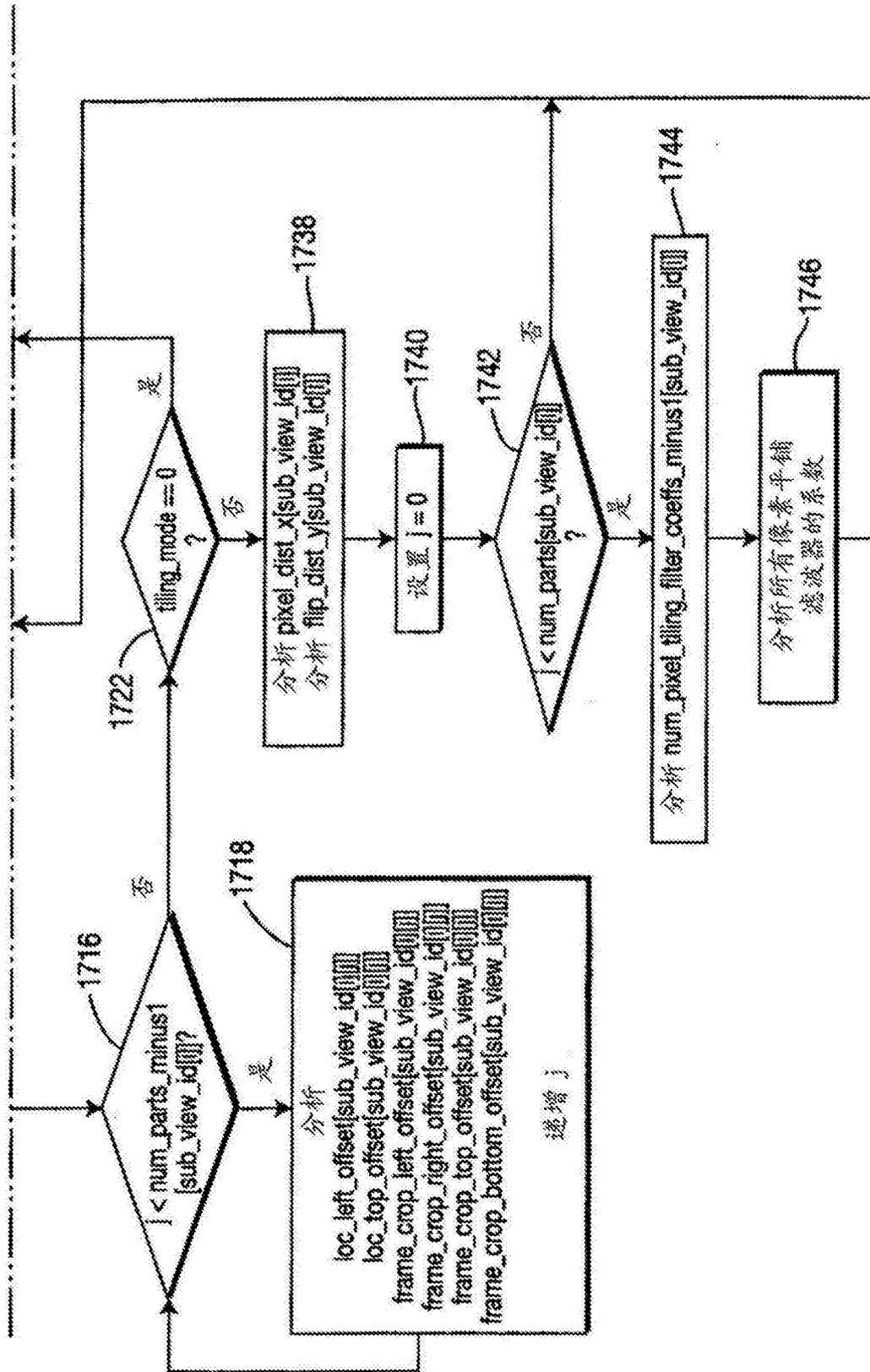


图17B

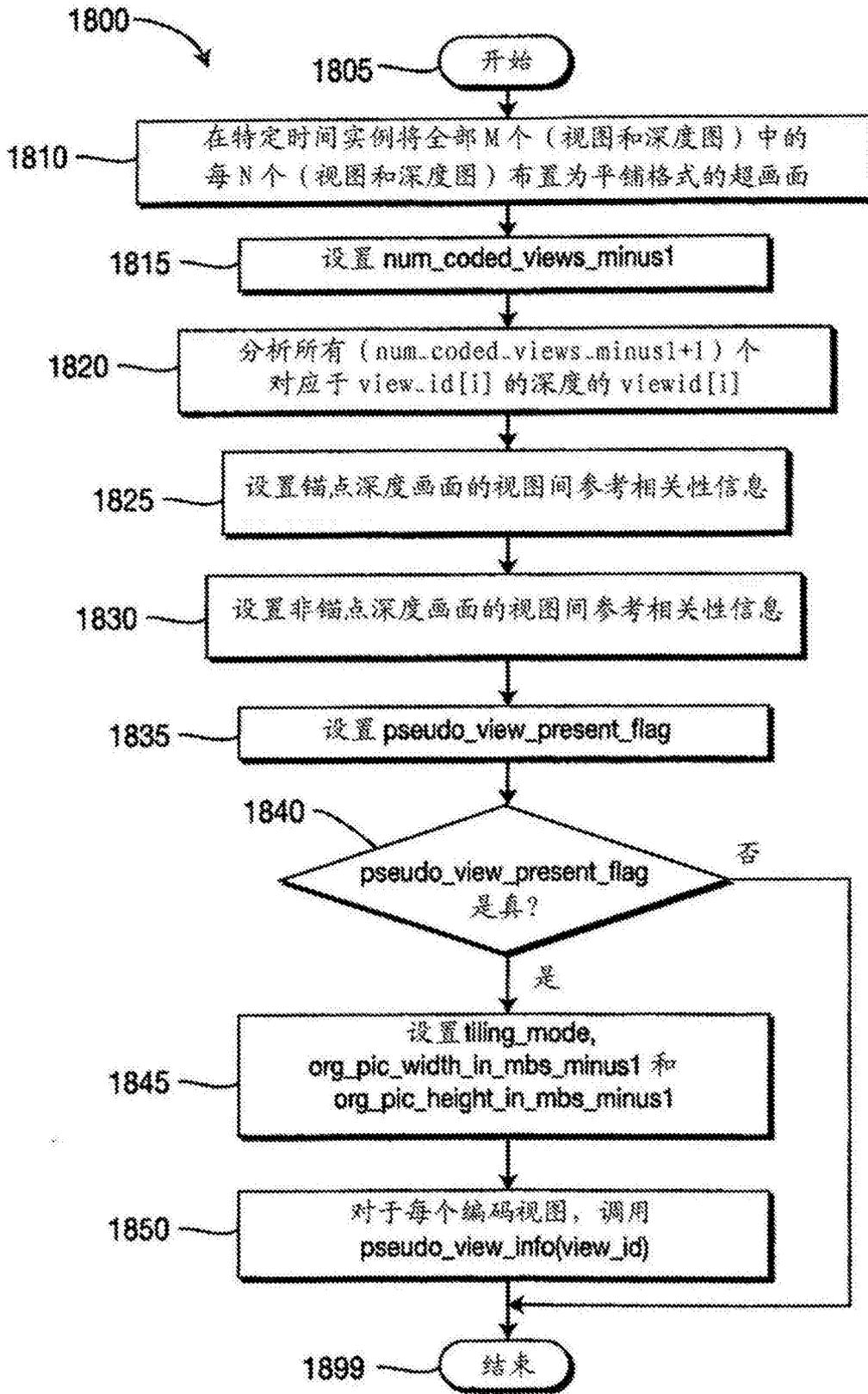


图18

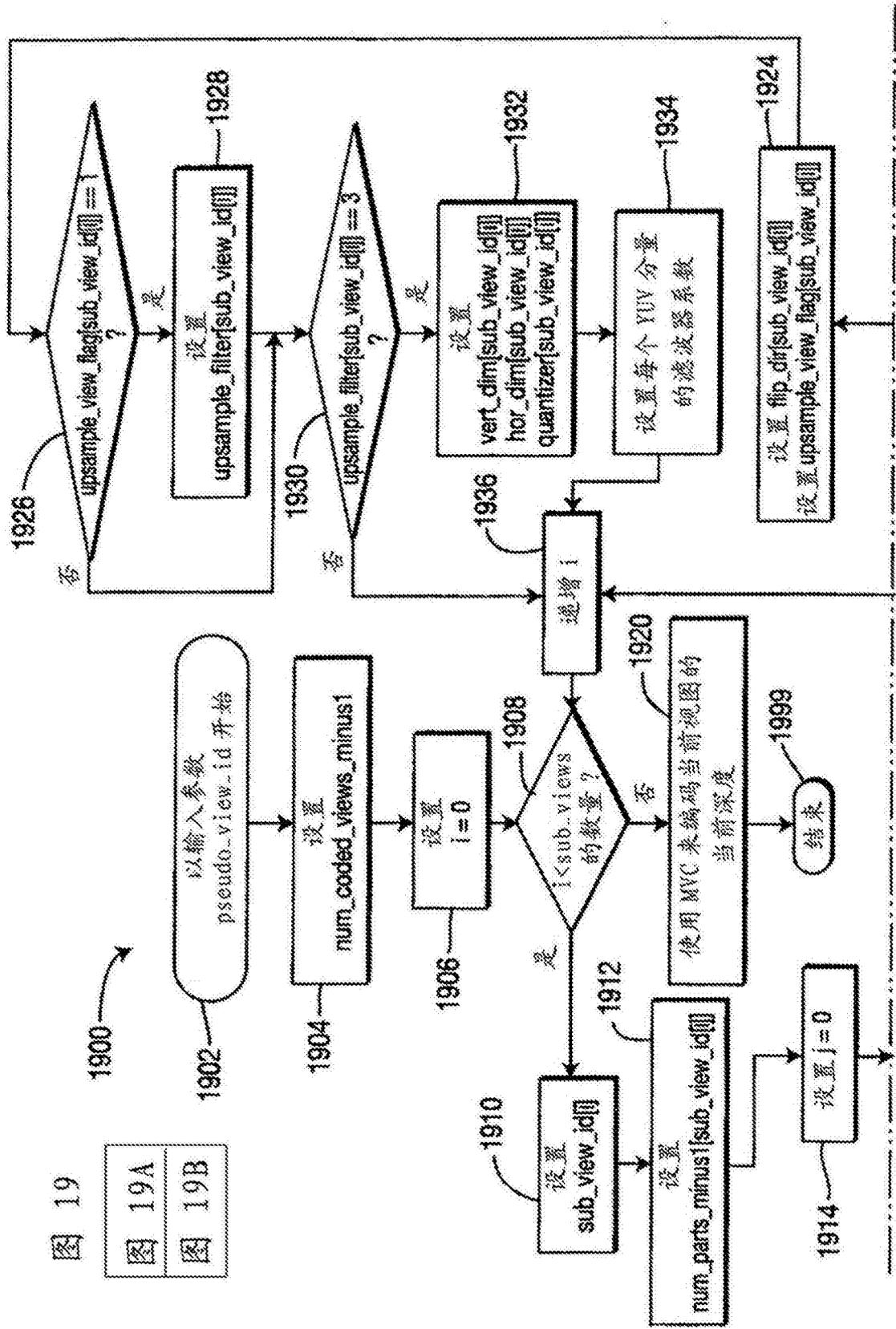


图 19A

图 19

图 19A

图 19B

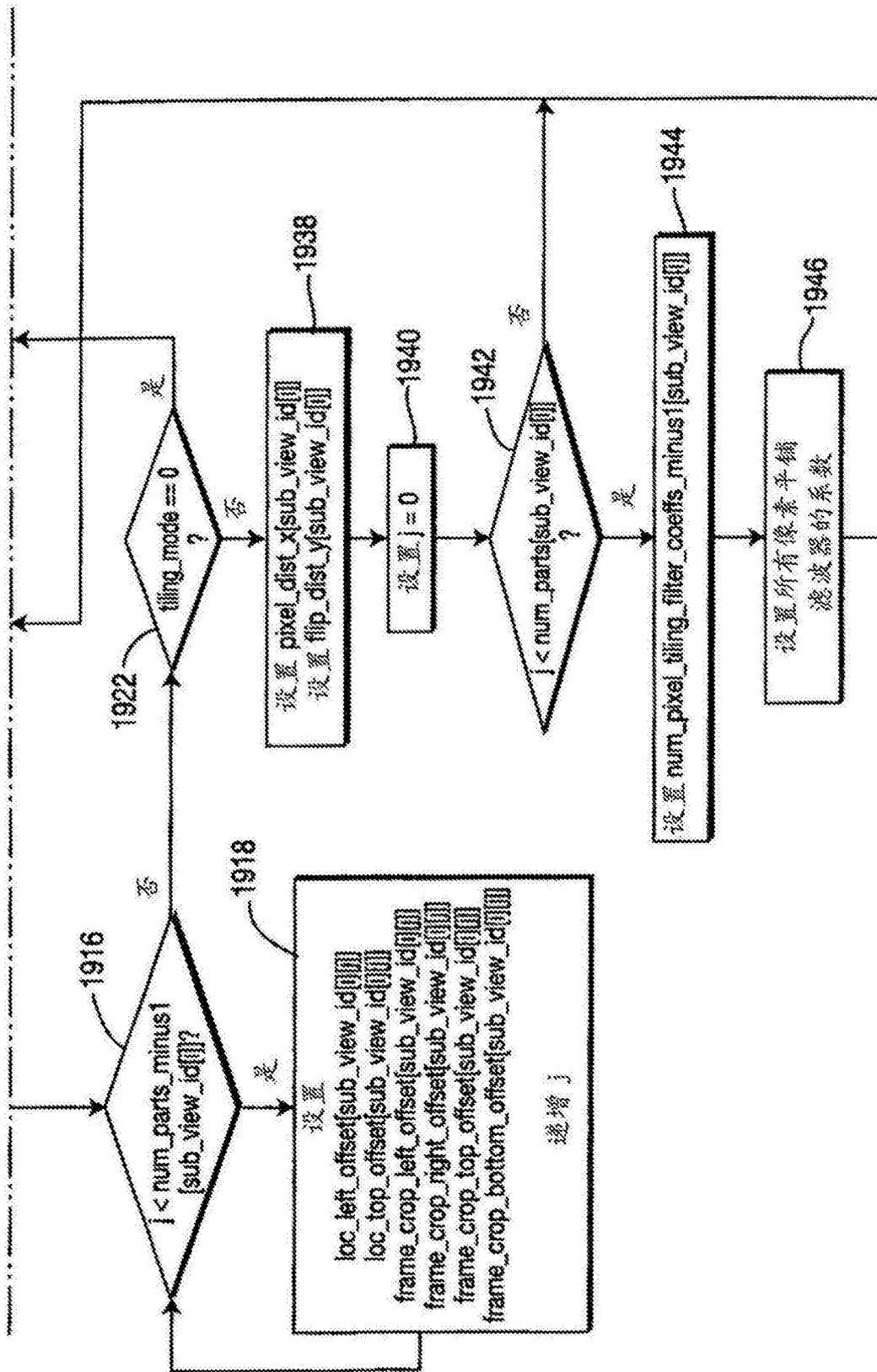


图19B

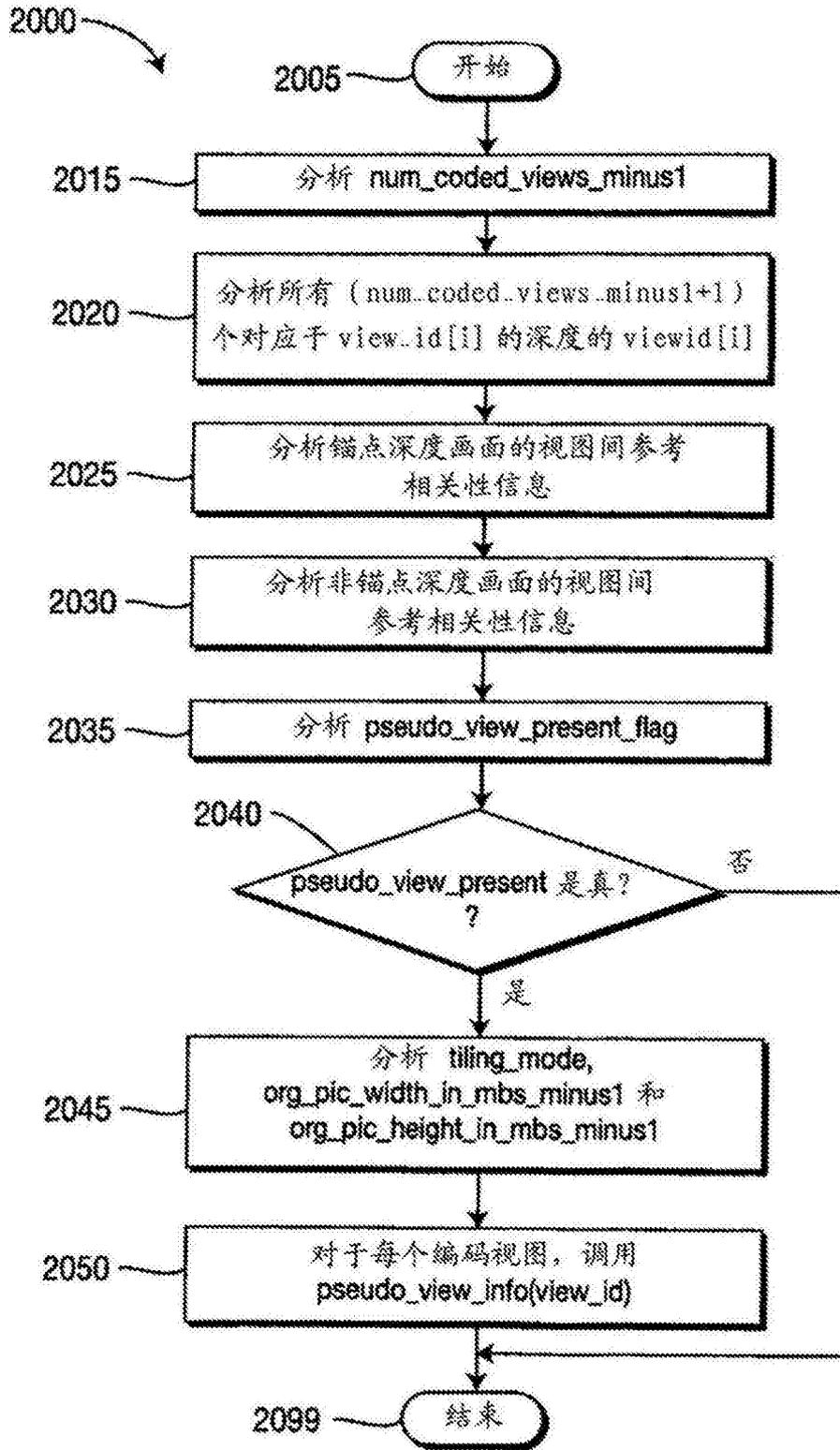


图20

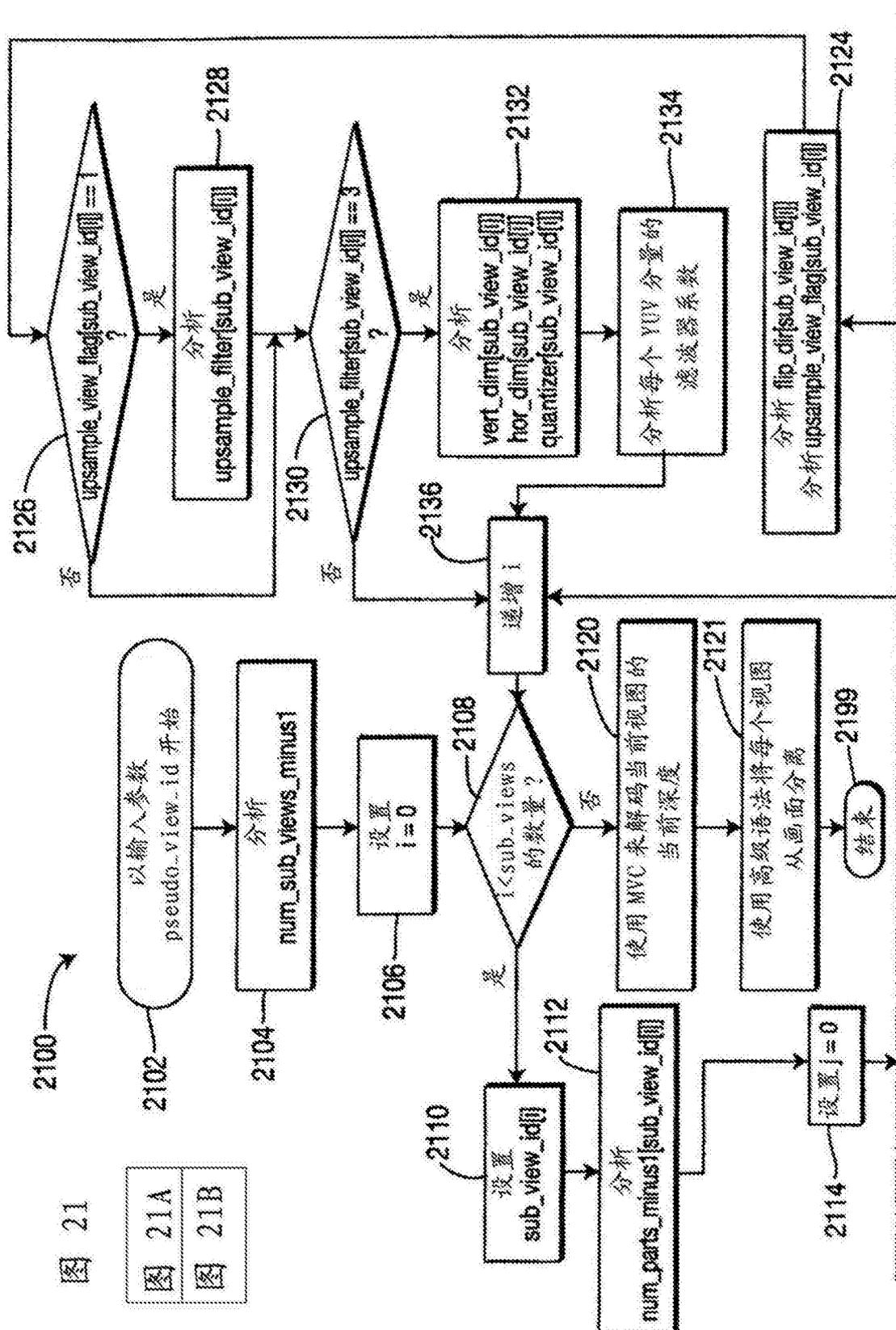


图 21A

图 21

图 21A

图 21B

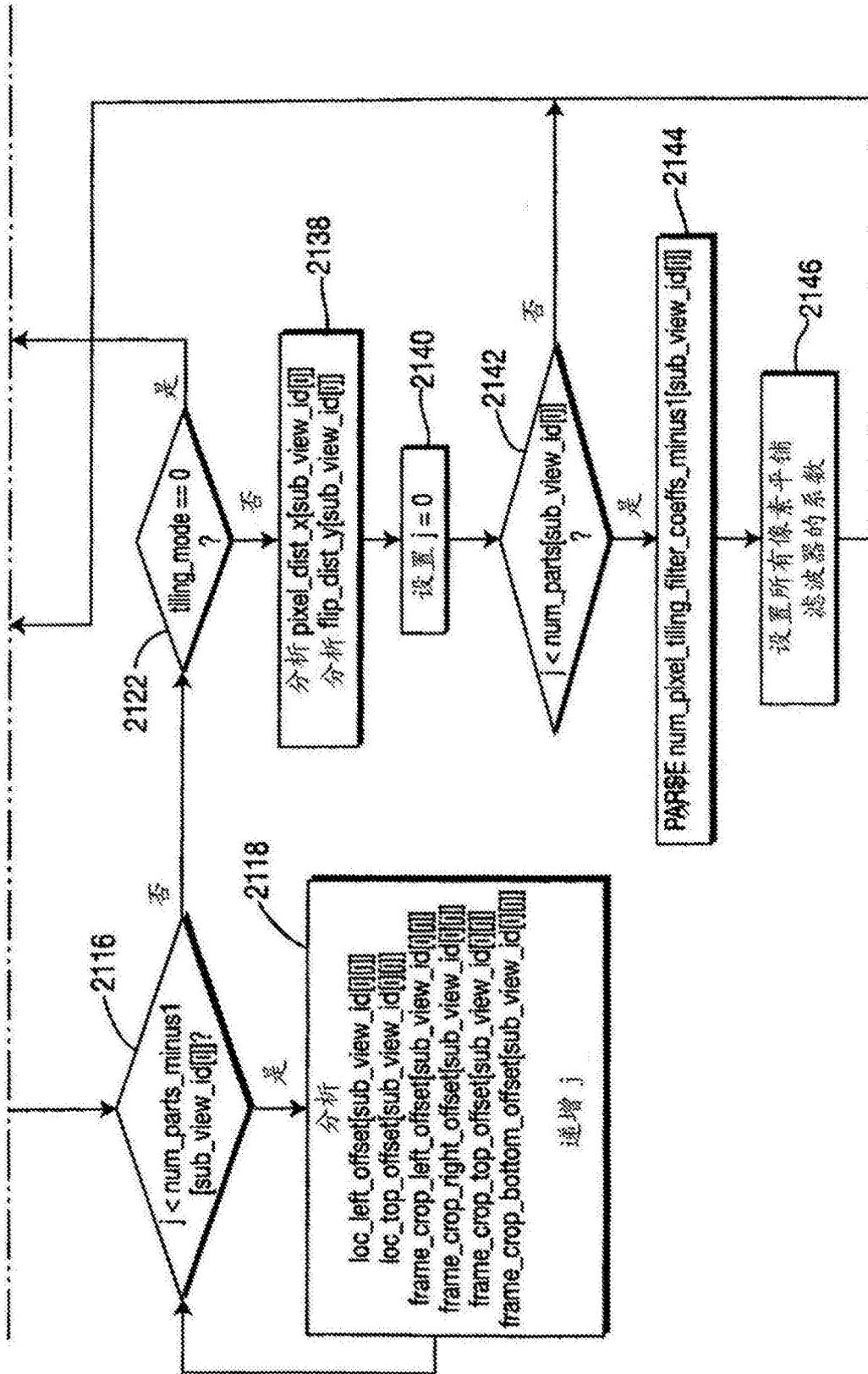


图21B

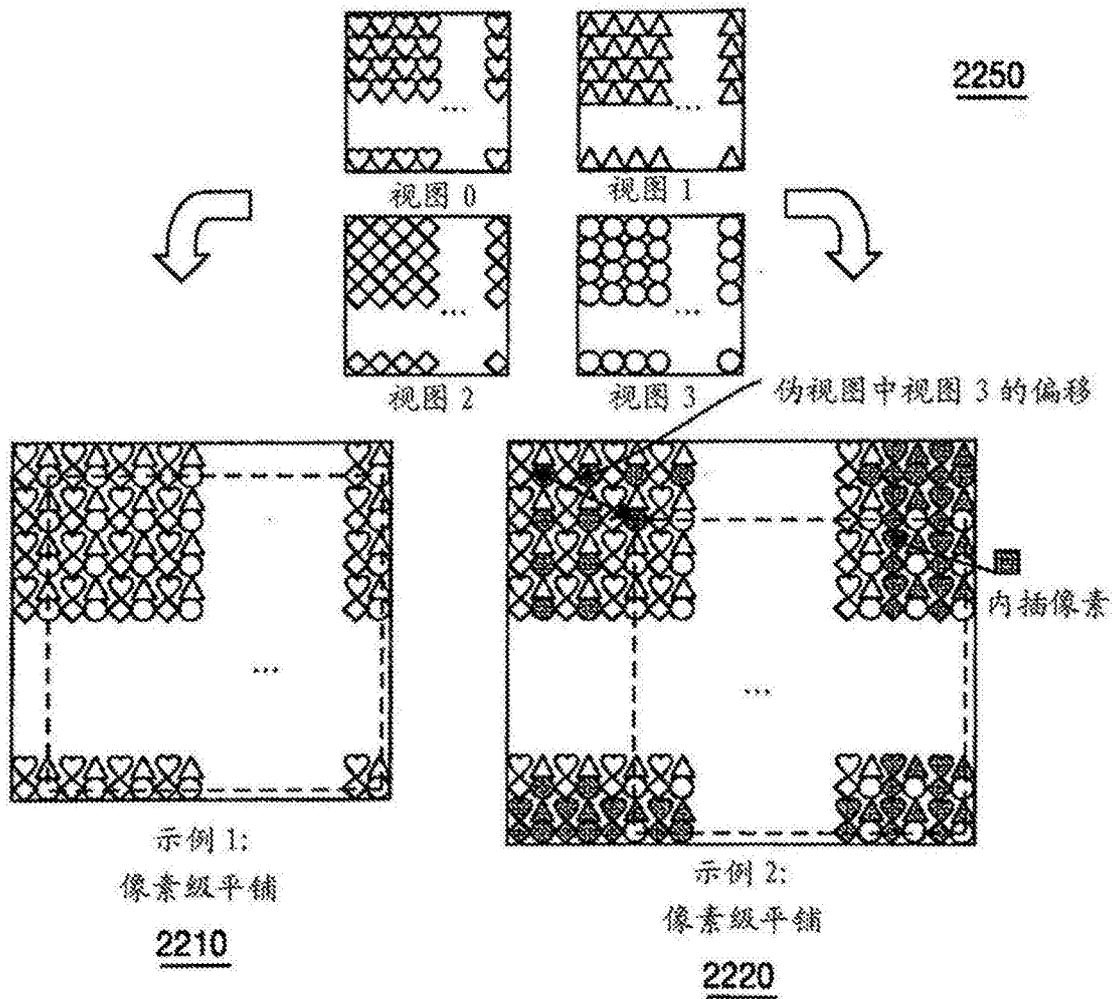


图22

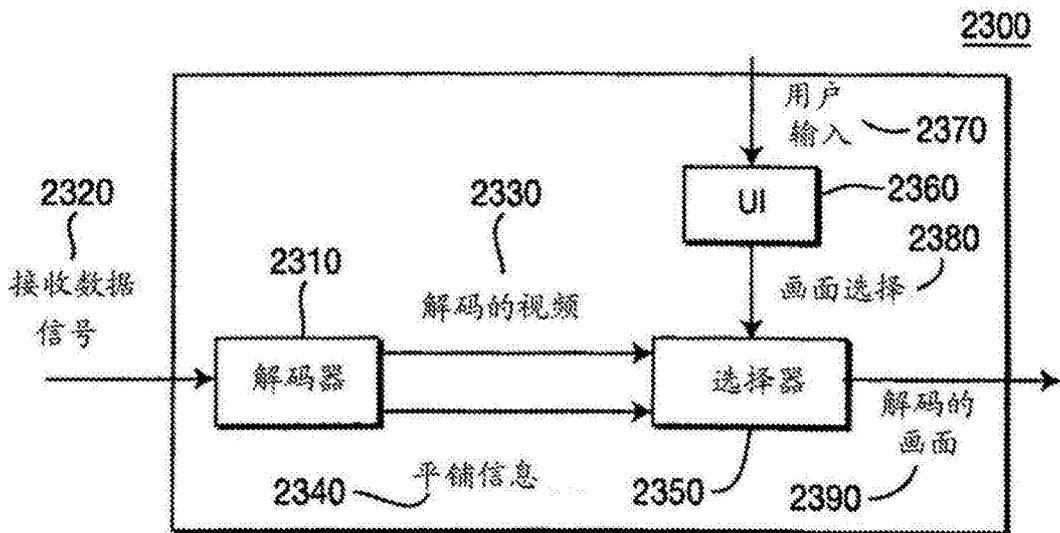


图23

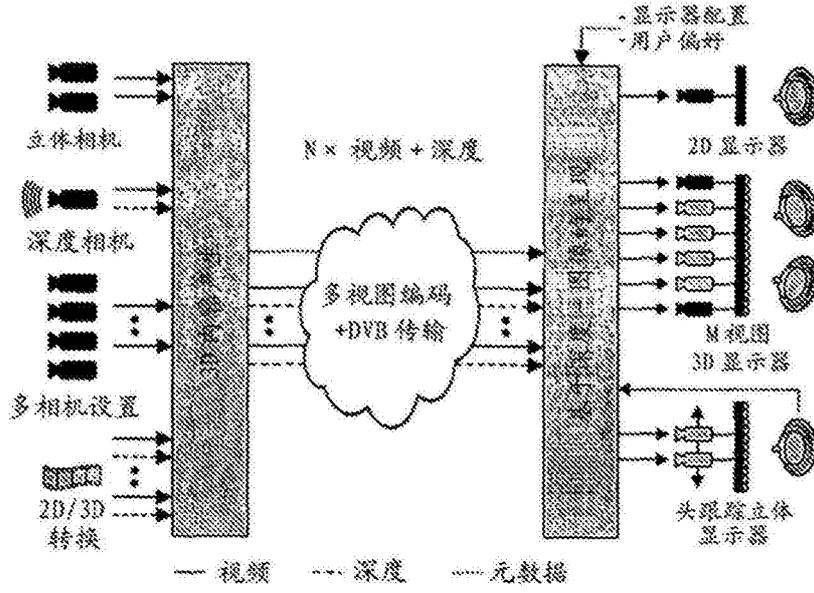
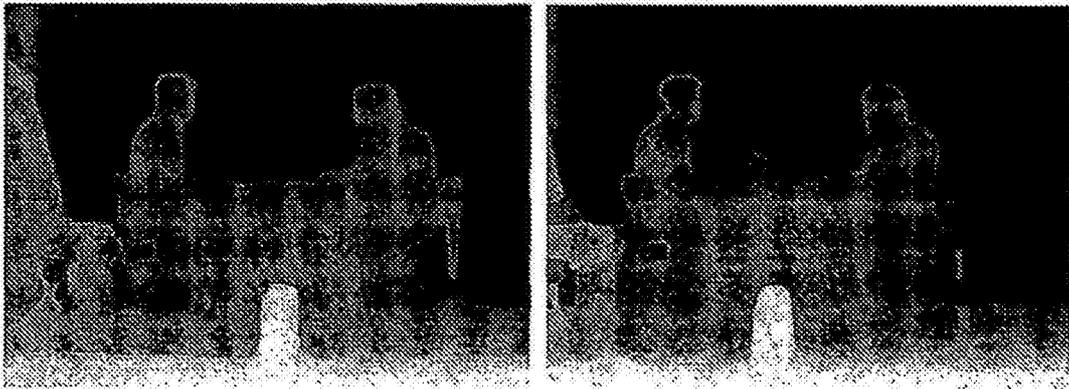


图24



(a) 左深度图

(b) 右深度图

图25

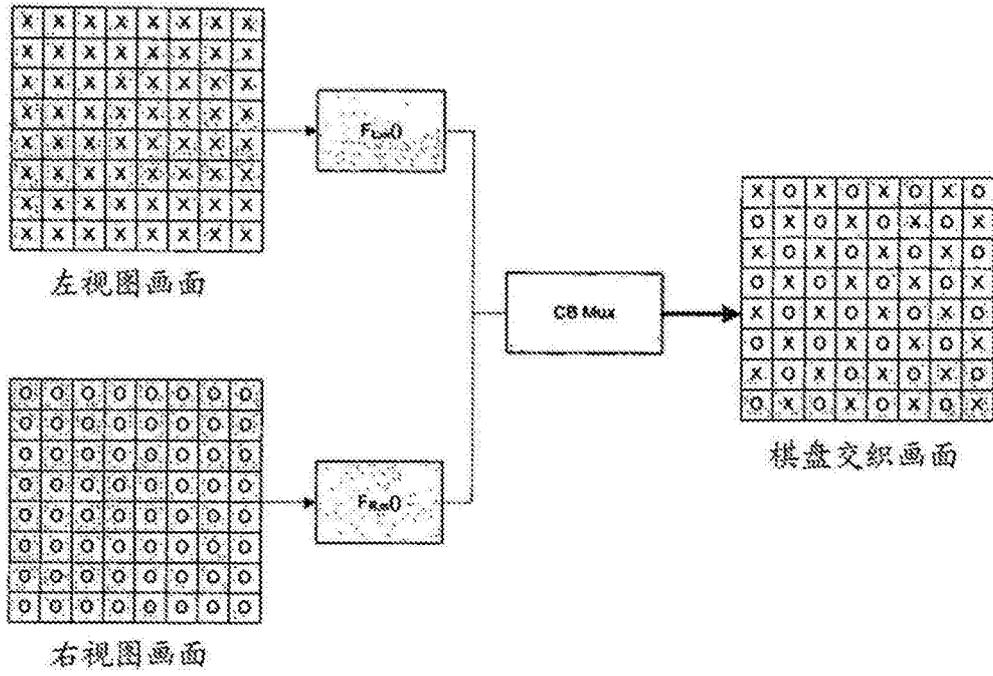


图26

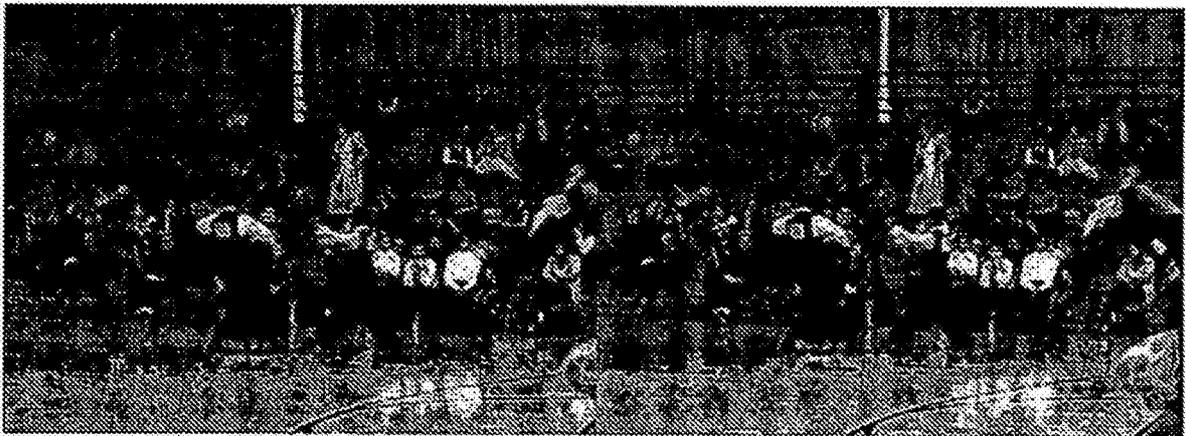


图27



图28

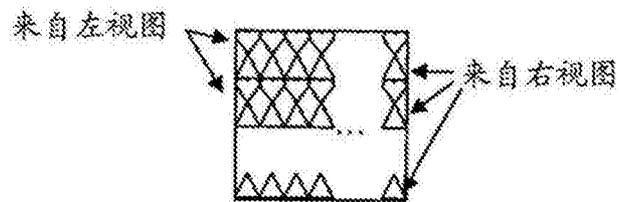
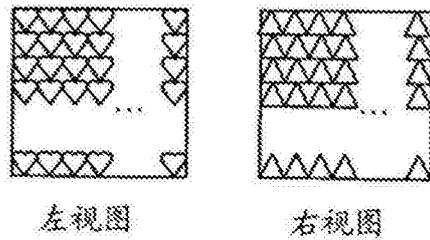


图29

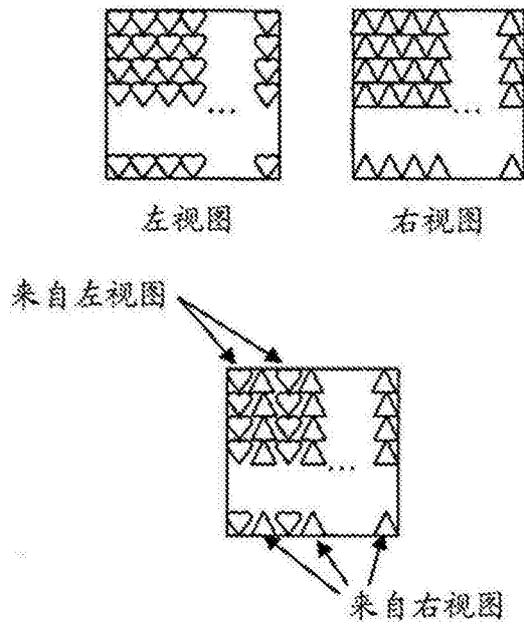


图30



图31

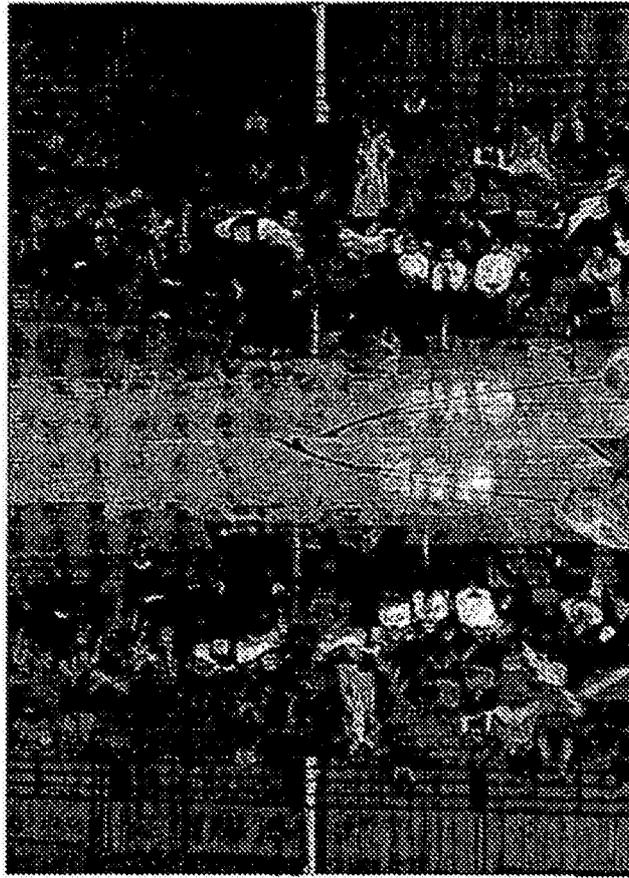


图32

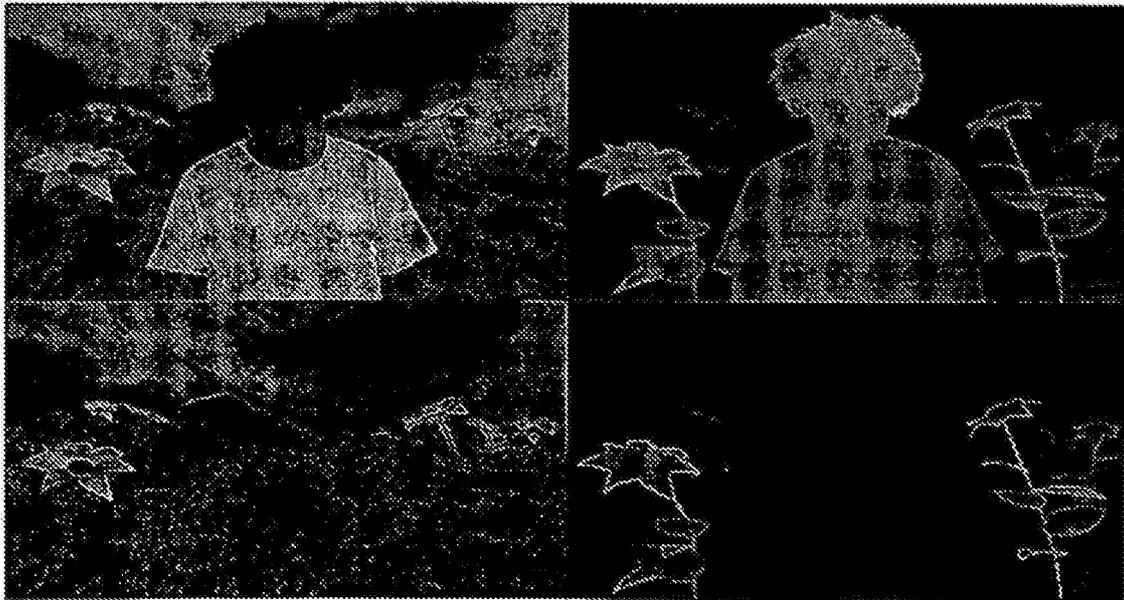
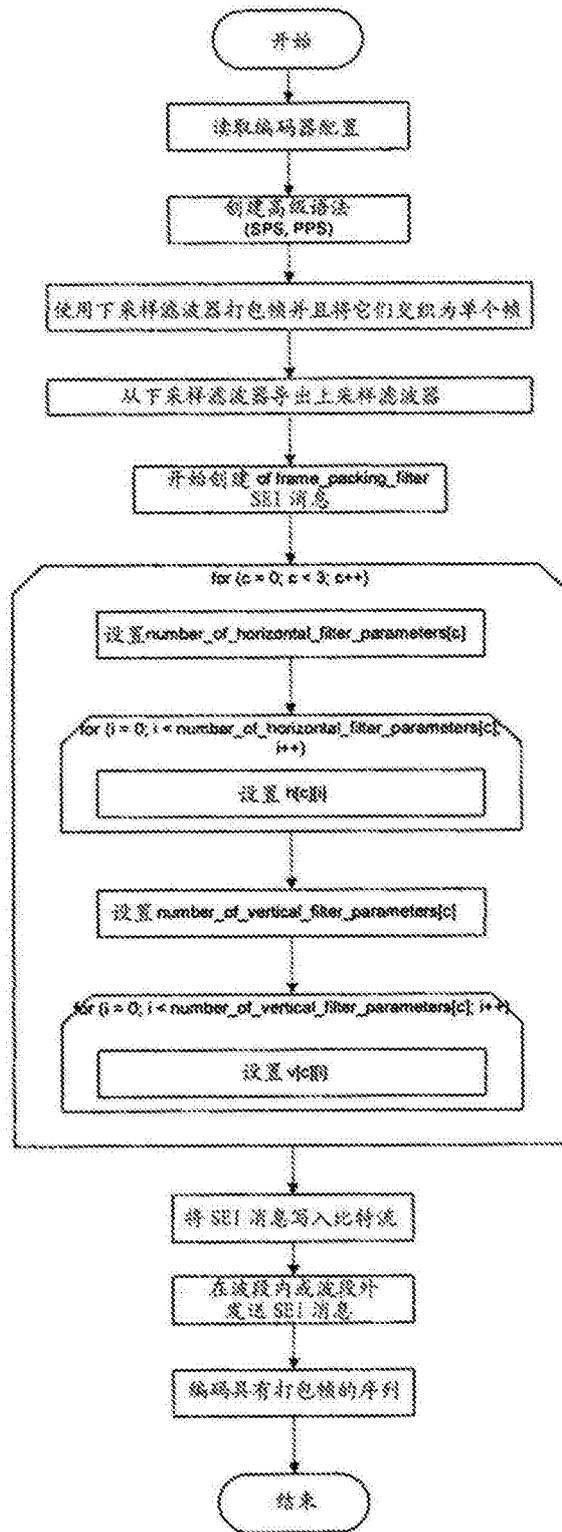


图33

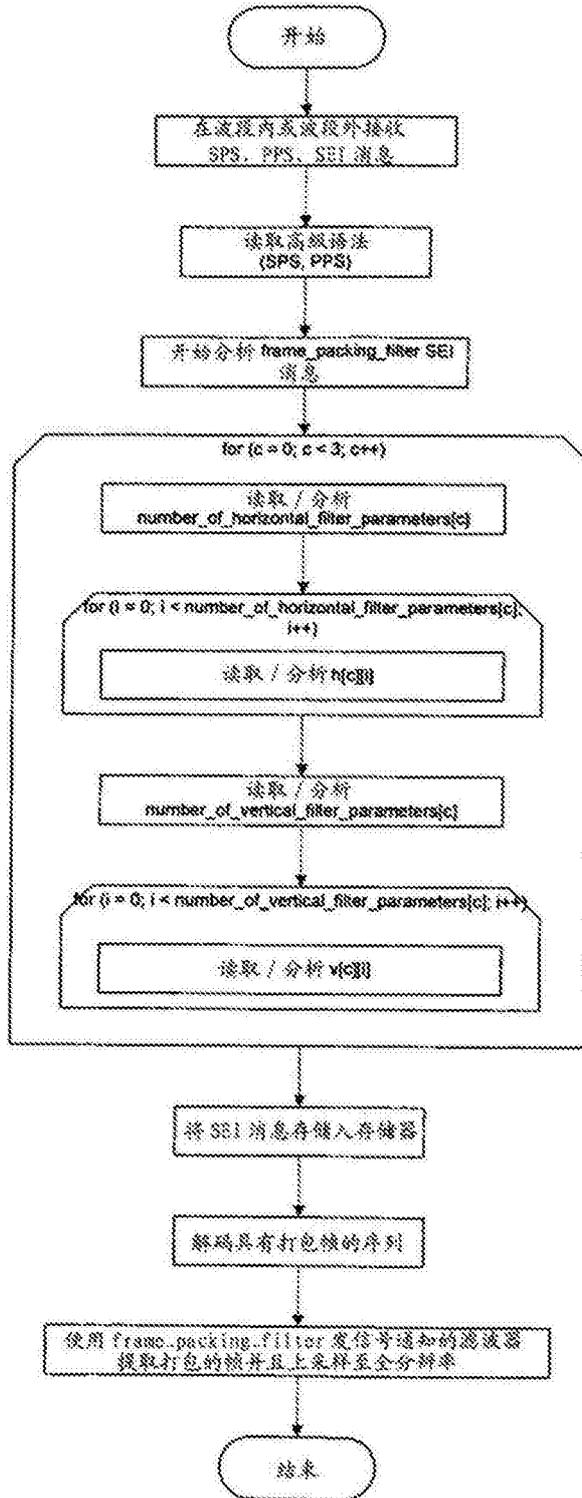


图34



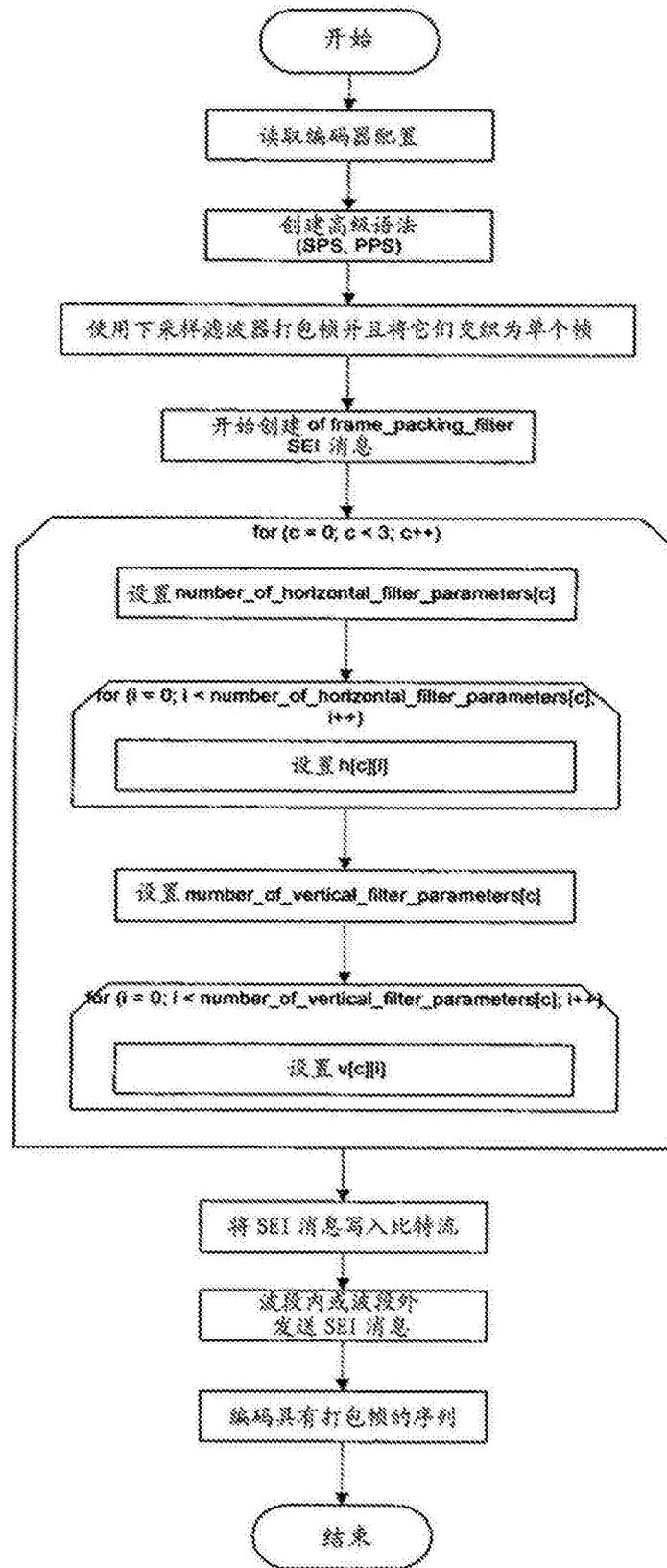
编码流程图 (实施例 1)

图35



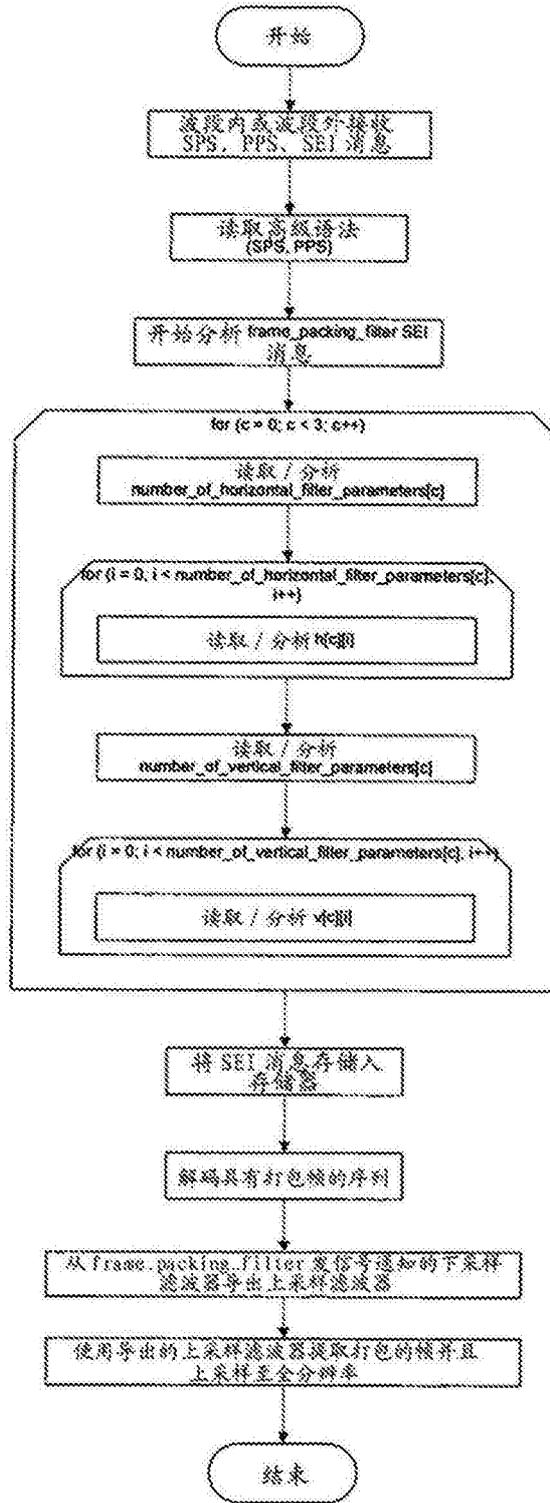
解码流程图 (示例 1)

图36



编码流程图 (实施例 2)

图37



解码流程图 (实施例 2)

图38

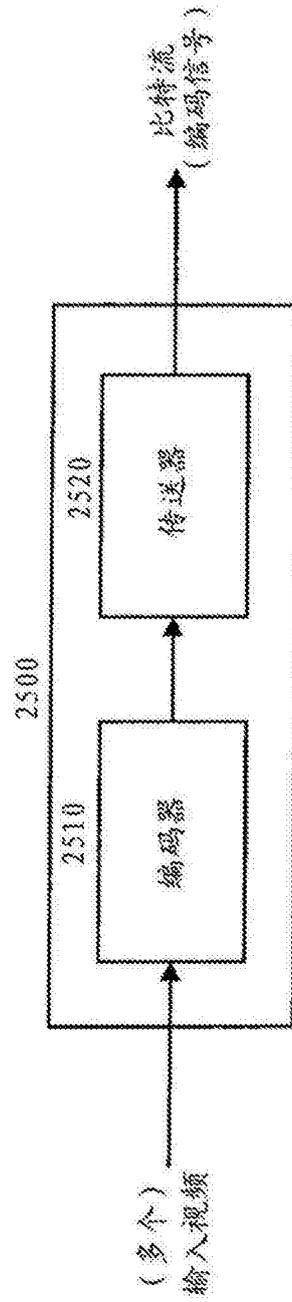


图39

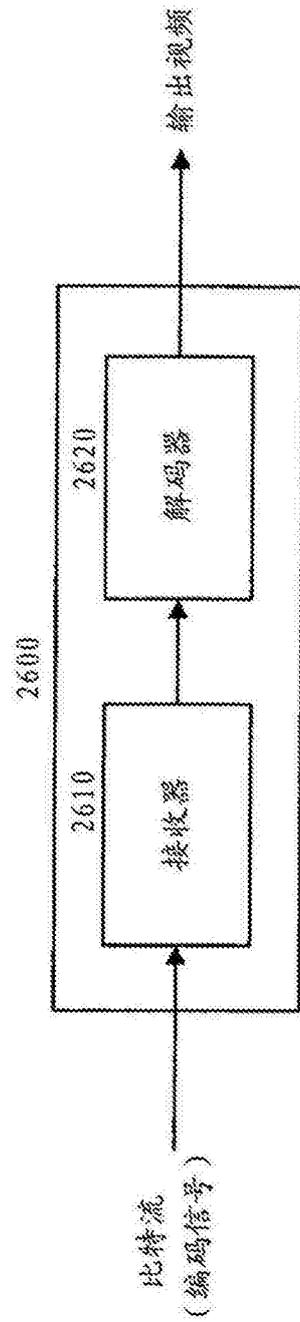


图40

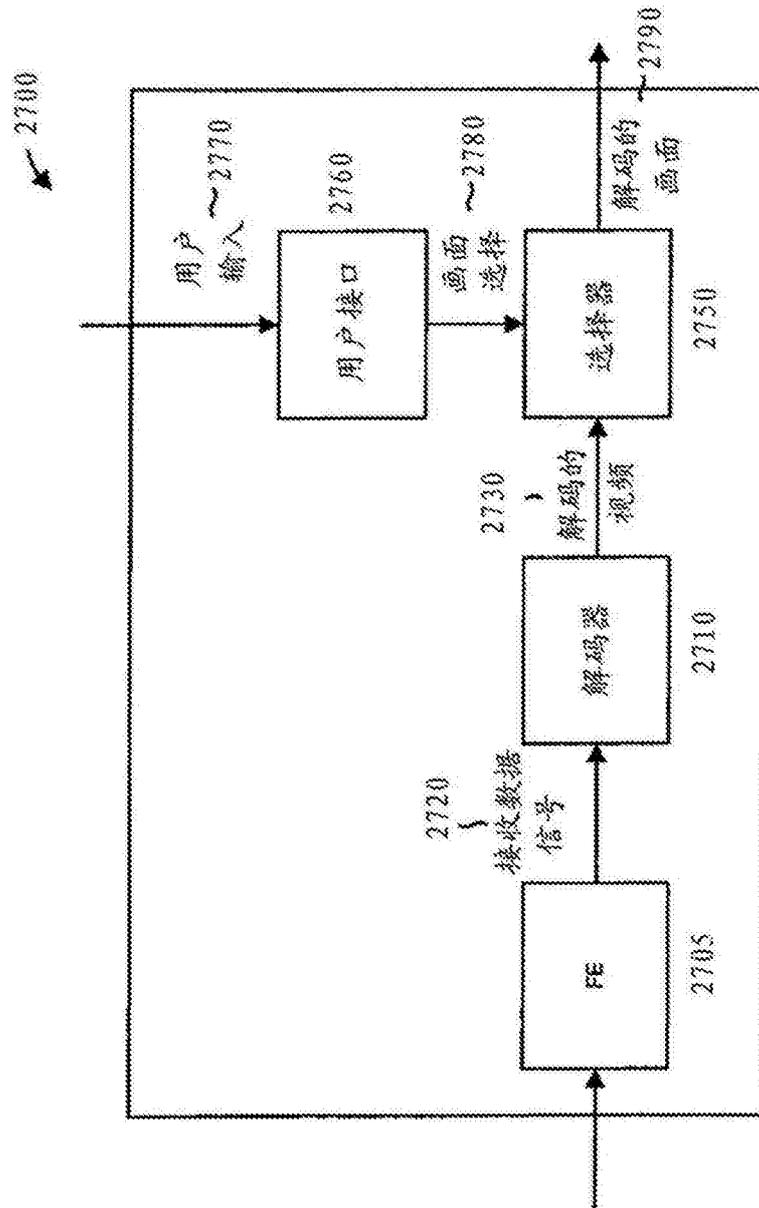


图41