(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 108650510 B
(45)授权公告日 2023.03.31

(21)申请号 201810443709.7

(22)申请日 2018.05.10

(65)同一申请的已公布的文献号
申请公布号 CN 108650510 A

(43)申请公布日 2018.10.12

(73)专利权人 中南大学
地址 410083 湖南省长沙市岳麓区麓山南路932号
专利权人 腾讯科技(深圳)有限公司

(72)发明人 牟凡 廖念波 汪亮 翟海昌 马学睿 王剑光 张昊

(74)专利代理机构 北京康信知识产权代理有限责任公司 11240
专利代理师 周婷婷 江舟

(51)Int.Cl.
H04N 19/172(2014.01)
H04N 19/40(2014.01)
H04N 19/30(2014.01)
H04N 19/146(2014.01)
H04N 19/154(2014.01)
H04N 19/70(2014.01)

(56)对比文件
CN 103929655 A,2014.07.16
US 2017078676 A1,2017.03.16
US 2012263241 A1,2012.10.18
US 2005041099 A1,2005.02.24

审查员 尚琴

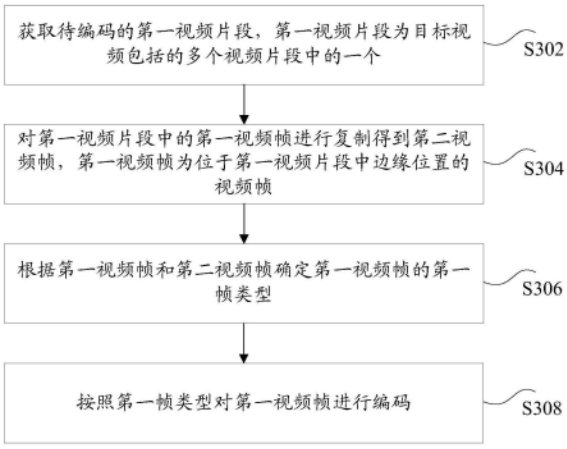
权利要求书3页 说明书20页 附图6页

(54)发明名称

视频的编码方法和装置、存储介质、电子装置

(57)摘要

本发明公开了一种视频的编码方法和装置、存储介质、电子装置。其中，该方法包括：获取待编码的第一视频片段，其中，第一视频片段为目标视频包括的多个视频片段中的一个，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；对第一视频片段中的第一视频帧进行复制得到第二视频帧，其中，第一视频帧为位于第一视频片段中边缘位置的视频帧；根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型；按照第一帧类型对第一视频帧进行编码。本发明解决了相关技术中导致视频画面的清晰度失去连贯性的技术问题。



CN 108650510 B

1.一种视频的编码方法,其特征在于,包括:

获取待编码的第一视频片段,其中,所述第一视频片段为目标视频包括的多个视频片段中的一个,所述多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码;

对所述第一视频片段中的第一视频帧进行复制得到第二视频帧,其中,所述第一视频帧为位于所述第一视频片段中边缘位置的视频帧;

根据所述第一视频帧和所述第二视频帧确定所述第一视频帧的第一帧类型,包括:根据关联关系确定第二帧类型,其中,所述关联关系为所述第一视频帧与所述第二视频帧之间、所述第一视频帧与所述第一视频帧之间以及所述第二视频帧与所述第二视频帧之间的关系,所述第二帧类型为所述第一视频帧或所述第二视频帧的帧类型;根据所述第二帧类型对所述第一视频帧和所述第二视频帧进行组合,得到至少一个帧集合,所述至少一个帧集合中的每个帧集合的集合类型包括以下至少之一:用于表示所包括的视频帧均为所述第一视频帧的第一类型、用于表示所包括的视频帧均为所述第二视频帧的第二类型以及用于表示包括所述第一视频帧和所述第二视频帧的第三类型;根据目标集合的集合类型确定所述目标集合中所述第一视频帧的所述第一帧类型,其中,所述目标集合为所述至少一个帧集合中的任意一个集合;

按照所述第一帧类型对所述第一视频帧进行编码。

2.根据权利要求1所述的方法,其特征在于,对所述第一视频片段中的第一视频帧进行复制得到第二视频帧包括:

对位于所述第一视频片段的播放结束位置的所述第一视频帧进行复制得到所述第二视频帧,其中,所述边缘位置包括所述播放结束位置;和/或,

对位于所述第一视频片段的播放起始位置的所述第一视频帧进行复制得到所述第二视频帧,其中,所述边缘位置包括所述播放起始位置。

3.根据权利要求2所述的方法,其特征在于,所述第一视频帧为多帧,

其中,对位于所述第一视频片段的播放结束位置的所述第一视频帧进行复制得到所述第二视频帧包括:

确定位于所述第一视频片段的播放结束位置的待复制的多帧所述第一视频帧;

按照在所述第一视频片段中的播放时间对多帧所述第一视频帧进行复制,得到多帧所述第二视频帧,其中,多帧所述第二视频帧按照播放时间从晚到早的顺序排列。

4.根据权利要求3所述的方法,其特征在于,确定位于所述第一视频片段的播放结束位置的待复制的多帧所述第一视频帧包括:

获取目标参数,其中,所述目标参数用于表示编码器已经从视频帧缓冲区数组读取的视频帧的帧数,所述视频帧缓冲区数组用于缓存待所述编码器进行预编码的视频帧;

在所述目标参数表示的帧数为第一阈值的情况下,确定位于所述第一视频片段的播放结束位置的N帧视频帧为多帧所述第一视频帧,其中,所述第一阈值用于表示所述第一视频片段中视频帧的帧数,N为所述视频帧缓冲区数组的缓冲容量,N为正整数。

5.根据权利要求3所述的方法,其特征在于,按照在所述第一视频片段中的播放时间对多帧所述第一视频帧进行复制,得到多帧所述第二视频帧包括:

将多帧所述第一视频帧写入视频帧缓冲区数组,并发送指示信息给编码器,其中,所述指示信息用于指示所述编码器按照播放时间从晚到早的顺序从所述视频帧缓冲区数组中

读取所述第一视频帧；

通过所述编码器，将按照播放时间从晚到早的顺序从所述视频帧缓冲区数组中读取的多帧所述第一视频帧作为多帧所述第二视频帧。

6.根据权利要求1所述的方法，其特征在于，根据目标集合的集合类型确定所述目标集合中所述第一视频帧的所述第一帧类型包括：

在所述目标集合的集合类型为所述第一类型的情况下，将所述目标集合中为所述第一视频帧确定的所述第二帧类型作为所述第一视频帧的所述第一帧类型；

在所述目标集合的集合类型为所述第三类型的情况下，根据所述目标集合的所述第一视频帧的数量确定所述第一视频帧的所述第一帧类型。

7.根据权利要求6所述的方法，其特征在于，所述目标集合中的所述第一视频帧和所述第二视频帧按照目标顺序排列，其中，根据所述目标集合的所述第一视频帧的数量确定所述第一视频帧的所述第一帧类型包括：

在所述目标集合的所述第一视频帧的帧数大于第二阈值的情况下，将第三视频帧的所述第二帧类型作为所述第三视频帧的所述第一帧类型，将第三帧类型作为第四视频帧的所述第一帧类型，将第四帧类型作为第五视频帧的所述第一帧类型，其中，所述第三视频帧为按照所述目标顺序排列时的最后一位的所述第一视频帧，所述第四视频帧是与所述第三视频帧相邻的所述第一视频帧，所述第五视频帧为多个所述第一视频帧中除所述第三视频帧和所述第四视频帧以外的视频帧，所述第二阈值为不小于2的整数；

在所述目标集合的所述第一视频帧的帧数等于所述第二阈值的情况下，将所述第三视频帧的所述第二帧类型作为所述第三视频帧的所述第一帧类型，将所述第四帧类型作为所述第四视频帧的所述第一帧类型；

在所述目标集合的所述第一视频帧的帧数小于所述第二阈值的情况下，将所述第一视频帧的所述第二帧类型作为所述第一视频帧的所述第一帧类型。

8.根据权利要求1所述的方法，其特征在于，

按照所述第一帧类型对所述第一视频帧进行编码包括：通过第一编码器，按照所述第一帧类型对所述第一视频帧进行编码，

在按照所述第一帧类型对所述第一视频帧进行编码的过程中，所述方法还包括：通过第二编码器，对第二视频片段中的视频帧进行编码，其中，所述多个视频片段包括所述第二视频片段，所述第一编码器和所述第二编码器为独立运行的编码器。

9.一种视频的编码装置，其特征在于，包括：

获取单元，用于获取待编码的第一视频片段，其中，所述第一视频片段为目标视频包括的多个视频片段中的一个，所述多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；

复制单元，用于对所述第一视频片段中的第一视频帧进行复制得到第二视频帧，其中，所述第一视频帧为位于所述第一视频片段中边缘位置的视频帧；

确定单元，用于根据所述第一视频帧和所述第二视频帧确定所述第一视频帧的第一帧类型，包括：根据关联关系确定第二帧类型，其中，所述关联关系为所述第一视频帧与所述第二视频帧之间、所述第一视频帧与所述第一视频帧之间以及所述第二视频帧与所述第二视频帧之间的关系，所述第二帧类型为所述第一视频帧或所述第二视频帧的帧类型；根据

所述第二帧类型对所述第一视频帧和所述第二视频帧进行组合，得到至少一个帧集合，所述至少一个帧集合中的每个帧集合的集合类型包括以下至少之一：用于表示所包括的视频帧均为所述第一视频帧的第一类型、用于表示所包括的视频帧均为所述第二视频帧的第二类型以及用于表示包括所述第一视频帧和所述第二视频帧的第三类型；根据目标集合的集合类型确定所述目标集合中所述第一视频帧的所述第一帧类型，其中，所述目标集合为所述至少一个帧集合中的任意一个集合；

编码单元，用于按照所述第一帧类型对所述第一视频帧进行编码。

10.根据权利要求9所述的装置，其特征在于，所述复制单元包括：

第一复制模块，用于对位于所述第一视频片段中的播放结束位置的所述第一视频帧进行复制得到所述第二视频帧，其中，所述边缘位置包括所述播放结束位置；

第二复制模块，用于对位于所述第一视频片段中的播放起始位置的所述第一视频帧进行复制得到所述第二视频帧，其中，所述边缘位置包括所述播放起始位置。

11.根据权利要求10所述的装置，其特征在于，所述第一视频帧为多帧，其中，所述第一复制模块还用于：

确定位于所述第一视频片段的播放结束位置的待复制的多帧所述第一视频帧；

按照在所述第一视频片段中的播放时间对多帧所述第一视频帧进行复制，得到多帧所述第二视频帧，其中，多帧所述第二视频帧按照播放时间从晚到早的顺序排列。

12.一种存储介质，其特征在于，所述存储介质包括存储的程序，其中，所述程序运行时执行上述权利要求1至8任一项中所述的方法。

13.一种电子装置，包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，其特征在于，所述处理器通过所述计算机程序执行上述权利要求1至8任一项中所述的方法。

视频的编码方法和装置、存储介质、电子装置

技术领域

[0001] 本发明涉及互联网领域，具体而言，涉及一种视频的编码方法和装置、存储介质、电子装置。

背景技术

[0002] 随着互联网的普及，多媒体，尤其视频成为内容的主要的承载媒介之一，而且视频正在朝着高清，甚至超清的方向发展，从而造成视频传输占据了网络传输大部分带宽，在给用户带来丰富体验的同时，为存储和传输带来了巨大压力，因此对视频的压缩很重要。

[0003] 相关技术中的视频编码标准普遍采用有损的率失真(Rate Distortion)压缩方法，采用基于块的预测一变换混合编码框架。要得到好的RD性能，往往带来编码计算复杂度的成倍增加，而分辨率越大，编码的复杂度也相应增大。

[0004] 服务器的CPU(Central Processing Unit，中央处理器)的运算资源有限，为了提高对高清视频的编码能力，大多数视频供应商采用将原始视频物理分割为多个片段，并把各个分片调度到不同的服务器上进行编码这一方案，在各个服务器编码完成之后进行合并，以提高编码速度。但是分割之后的视频片段由于复杂度的差异等原因导致各片段之间码率不一，进而导致最终合并视频的视频质量在分割点处产生波动，使得画面清晰度失去连贯性。

[0005] 针对上述的问题，目前尚未提出有效的解决方案。

发明内容

[0006] 本发明实施例提供了一种视频的编码方法和装置、存储介质、电子装置，以至少解决相关技术中导致视频画面的清晰度失去连贯性的技术问题。

[0007] 根据本发明实施例的一个方面，提供了一种视频的编码方法，包括：获取待编码的第一视频片段，其中，第一视频片段为目标视频包括的多个视频片段中的一个，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；对第一视频片段中的第一视频帧进行复制得到第二视频帧，其中，第一视频帧为位于第一视频片段中边缘位置的视频帧；根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型；按照第一帧类型对第一视频帧进行编码。

[0008] 根据本发明实施例的另一方面，还提供了一种视频的编码装置，包括：获取单元，用于获取待编码的第一视频片段，其中，第一视频片段为目标视频包括的多个视频片段中的一个，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；复制单元，用于对第一视频片段中的第一视频帧进行复制得到第二视频帧，其中，第一视频帧为位于第一视频片段中边缘位置的视频帧；确定单元，用于根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型；编码单元，用于按照第一帧类型对第一视频帧进行编码。

[0009] 根据本发明实施例的另一方面，还提供了一种存储介质，该存储介质包括存储的程序，程序运行时执行上述的方法。

[0010] 根据本发明实施例的另一方面，还提供了一种电子装置，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，处理器通过计算机程序执行上述的方法。

[0011] 在本发明实施例中，获取待编码的第一视频片段，第一视频片段为目标视频包括的多个视频片段中的一个，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；对第一视频片段中的第一视频帧进行复制得到第二视频帧，第一视频帧为位于第一视频片段中边缘位置的视频帧；根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型；按照第一帧类型对第一视频帧进行编码，采用复制视频序列边缘位置帧的方式近似模拟原视频帧进行预编码过程，利用视频序列在时间上的相关性(或称为时域相关性，是指视频序列中前后帧图像的相似性)以及预编码过程对码率分配的影响来减小分割点处的质量波动，利用时域相关性可以进行尾部帧复制模拟原视频帧情况，然后预编码的帧决策与原视频就比较相近，所以码率分配也与原视频较为相似，可以解决相关技术中导致视频画面的清晰度失去连贯性的技术问题，进而达到减少了视频帧决策对码率分配的影响、保持了分割点前的画面清晰度不变的技术效果。

附图说明

[0012] 此处所说明的附图用来提供对本发明的进一步理解，构成本申请的一部分，本发明的示意性实施例及其说明用于解释本发明，并不构成对本发明的不当限定。在附图中：

[0013] 图1是根据本发明实施例的一种可选的视频分片的示意图；

[0014] 图2是根据本发明实施例的视频的编码方法的硬件环境的示意图；

[0015] 图3是根据本发明实施例的一种可选的视频的编码方法的流程图；

[0016] 图4是根据本发明实施例的一种可选的视频的编码方法的示意图；

[0017] 图5是根据本发明实施例的一种可选的视频帧复制的示意图；

[0018] 图6是根据本发明实施例的一种可选的视频帧复制的示意图；

[0019] 图7是根据本发明实施例的一种可选的分布式视频转码的示意图；

[0020] 图8是根据本发明实施例的一种可选的视频帧的示意图；

[0021] 图9是根据本发明实施例的一种可选的视频的编码装置的示意图；

[0022] 以及

[0023] 图10是根据本发明实施例的一种终端的结构框图。

具体实施方式

[0024] 为了使本技术领域的人员更好地理解本发明方案，下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本发明一部分的实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都应当属于本发明保护的范围。

[0025] 需要说明的是，本发明的说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别类似的对象，而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换，以便这里描述的本发明的实施例能够以除了在这里图示或

描述的那些以外的顺序实施。此外，术语“包括”和“具有”以及他们的任何变形，意图在于覆盖不排他的包含，例如，包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元，而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

[0026] 首先，在对本发明实施例进行描述的过程中出现的部分名词或者术语适用于如下解释：

[0027] 码率控制：CRF模式(英文全称为Constant Ratefactor，是编码中根据片子质量自动分配码率的码率控制方式)和ABR(英文全称为Available Bit-Rate，中文名称为可用比特率)模式为码率控制的方式，如果用户不想要特定的比特率，而是指定质量，则CRF模式是一个最优的一次通过模式；而ABR模式是在码率控制不知道将来帧的情况下完成的，因此，ABR不能精准的控制目标文件大小。

[0028] pass 1、pass 2、pass 3分别指第一遍码率控制编码、第二遍码率控制编码、第二遍码率控制编码。

[0029] 图像组GOP(Group of Pictures)：视频序列由若干时间连续的图像构成，在对其进行压缩时，先将该视频序列分割为若干个小的图像组GOP，然后以GOP为单位进行编码。

[0030] 根据本发明实施例的一方面，提供了一种视频的编码方法的方法实施例。

[0031] 为了保持分布式视频转码中分割点处画面清晰度的连贯性，本申请提供了多种相应的解决方案，如使用pass3(即第三遍码率控制编码)进行平滑，但是存在编码时间较长的缺陷；再如，使用pass1中的码率数据调整pass2的码率分配过程，但是其效果仍然不够完美。

[0032] 可选地，除了上述方案以外，本申请还提供了一种基于重叠分片技术实现分布式视频转码的方案，如图1所示，该方案保证了分割点前视频帧的图像质量与未分片视频相应帧的图像质量的一致，但是存在需要传输额外的视频帧到编码服务器、而且需要等待后续延长的帧传输完成后才能完成编码的缺陷，会引起一定时延，在对原始视频进行物理分割时多切割预编码长度的帧(即分片)，通过增加输入的视频帧数(即额外帧)，以保证上一个分片尾部帧的预编码过程(预编码过程是为了确定后续GOP中每个帧的帧类型)和原始未切片的相应视频帧的预编码过程一致，减小两种转码过程中相对应帧的码率差别，保证分片转码的视频质量与原始未分片转码的视频质量有着相同的连贯性，在主观质量上无差别，除此之外，为了在分布式转码视频码流的准确拼接，需要在分割点所在的GOP处调整编码顺序，且将视频帧类型均修改为P帧，该技术方案需要改变原有分布式视频转码架构，根据不同的预编码长度调整物理分割时的视频帧长度，这种变动不利于相关技术中视频转码构架的稳定，而且将分割点所在GOP的视频帧类型全都修改为P帧的方法与视频原有的帧类型决策方案不同，不利于视频序列在分割点处的连贯性。

[0033] 从上述内容可知，使用pass 3的技术方案存在耗时较长，编码效率低下的缺陷；对于使用pass 1的码率数据调整pass 2码率分配的技术方案，由于pass 1的码率分配过程自身存在一定程度上的偏差，因此在pass 2码率分配的调整过程中容易出现误差，想要达到较好的效果难度较大，技术成本较高。对于基于重叠分片技术的分布式视频转码技术方法，该技术方案需要传输额外的帧，而且会产生时延，另外需要对整个分布式视频转码的构架进行变动，而且对分割点所在GOP的帧类型处理方式不利于视频序列的连贯性。

[0034] 在本申请的实施例中，还提供了一种视频的编码方法的方法实施例，以克服上述缺陷。

[0035] 可选地，在本实施例中，上述视频的编码方法可以应用于如图2所示的由编码服务器201和内容服务器203所构成的硬件环境中，如图2所示，编码服务器201通过网络与内容服务器203进行连接，可用于为内容服务器203提供服务(如游戏服务、应用服务、编码服务等)，内容服务器203用于为用户、应用、网站等提供多媒体内容服务，如提供视频内容、音频内容等，该硬件环境还可包括用于为编码服务器201提供数据存储服务的数据库205，上述网络包括但不限于：广域网、城域网或局域网。

[0036] 可选地，编码服务器201、内容服务器203以及数据库205之间的关系可以包括但不限于如下关系之一：1)编码服务器201、内容服务器203以及数据库205作为独立的硬件设备独立存在；2)编码服务器201、内容服务器203以及数据库205中的任意两个集成在一起，如将编码服务器201集成在内容服务器203上作为内容服务器203上的一个模块，或者将数据库205集成在编码服务器201作为编码服务器201上的一个模块等；3)将编码服务器201、内容服务器203以及数据库205三者集成在同一设备上。

[0037] 本发明实施例的视频的编码方法可以由编码服务器201来执行，图3是根据本发明实施例的一种可选的视频的编码方法的流程图，如图3所示，该方法可以包括以下步骤：

[0038] 步骤S302，编码服务器获取待编码的第一视频片段，第一视频片段为目标视频包括的多个视频片段中的一个。

[0039] 多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码，换言之，视频片段与编码器之间是一一对应的，有多少个视频片段那么就需要相应数量的编码器，这些编码器之间是独立运行的，此处的独立运行的方式包括但不局限于：1)在同一编码服务器上，通过多条进程独立运行这些编码器；2)通过多个编码服务器运行这些编码器，每个编码器在相应的一个编码服务器上执行编码服务；3)部分编码服务器中的每个编码服务器运行一个编码器，另一部分编码服务器中的每个编码服务器通过多条进程独立运行至少两个编码器。

[0040] 步骤S304，编码服务器对第一视频片段中的第一视频帧进行复制得到第二视频帧，第一视频帧为位于第一视频片段中边缘位置的视频帧。

[0041] 步骤S306，编码服务器根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型，也即通过对第一视频帧和第二视频帧进行预编码确定第一视频帧的第一帧类型。

[0042] 预编码过程，即编码器在实际编码每一帧之前对其进行的一项操作，通过综合评判当前帧(如上述的第一视频帧)及后续LookaheadDepth(预编码长度，后文缩写LaD)帧(如第二视频帧)的复杂度决定当前帧的帧类型sliceTpye(后文缩写为sT)及其码率。

[0043] 帧类型，可分为I帧、P帧、B_REF帧以及b帧等，其中，I帧中的编码单元CU使用帧内预测；P帧中CU可以使用帧内预测和前向帧间预测；B_REF帧中CU可以使用帧内预测和双向预测，且可以被其它帧参考；b帧中CU可以使用帧内预测和双向预测，但是不能被其他帧参考。

[0044] 步骤S308，编码服务器按照第一帧类型对第一视频帧进行编码。

[0045] 本申请的编码服务器为具备编码功能的服务器，若内容服务器本身也具备编码功能，那么内容服务器也可被视为编码服务器，换言之，本发明实施例的视频的编码方法可以

由内容服务器203来执行，还可以是由编码服务器201和内容服务器203共同执行。需要说明的是，编码服务器201或内容服务器203执行本发明实施例的视频的编码方法也可以是由安装在其上的编码器的客户端来执行。

[0046] 通过上述步骤S302至步骤S306，通过获取待编码的第一视频片段，第一视频片段为目标视频包括的多个视频片段中的一个，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；对第一视频片段中的第一视频帧进行复制得到第二视频帧，第一视频帧为位于第一视频片段中边缘位置的视频帧；根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型；按照第一帧类型对第一视频帧进行编码，采用复制视频序列边缘位置帧的方式近似模拟原视频帧进行预编码过程，利用视频序列在时间上的相关性(或称为时域相关性，是指视频序列中前后帧图像的相似性)以及预编码过程对码率分配的影响来减小分割点处的质量波动，利用时域相关性可以进行尾部帧复制模拟原视频帧情况，然后预编码的帧决策与原视频就比较相近，所以码率分配也与原视频较为相似，可以解决相关技术中导致视频画面的清晰度失去连贯性的技术问题，进而达到减少了视频帧决策对码率分配的影响、保持了分割点前的画面清晰度不变的技术效果。

[0047] 下面结合图3所示的步骤进一步详述本申请的实施例。

[0048] 随着日常生活节奏的加快以及网络信息量的增大，大众对于视频的要求也随之提高，为了满足这一要求，视频供应商大多按照编码服务器的处理能力将原始视频物理地分割为多个片段，并把各个分片调度到不同的服务器上进行编码，最后在各个编码服务器完成编码后对编码结果进行合并，以通过编码服务器的并行运行来提高编码速度，但是分割之后的视频片段由于复杂度的差异等原因会导致各片段之间码率不一，进而导致最终合并视频的视频质量在分割点处产生波动，使得画面清晰度失去连贯性。

[0049] 因此，本申请针对这一问题提出了一种基于视频尾部帧复制的分布式视频转码方法(可以在若干个编码服务器独立编码，也可以在一台编码服务器上面并行编码)，在需要进行编码时，即可执行本申请的方法，如在步骤S302提供的技术方案中，如图4所示，编码服务器获取待编码的第一视频片段，如内容服务器进行视频切片后分发的视频分片1，第一视频片段为内容服务器对目标视频进行视频切片后得到的多个视频片段中的一个，如视频分片1，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码，如视频分片1被分配到编码服务器1上进行编码，各个编码服务器上对相应的分片进行编码后得到的编码码流被拼接为完整码流。

[0050] 该方案主要利用视频序列在时间上的相关性以及预编码过程对码率分配的影响来减小分割点处的质量波动，本申请可采用镜面复制视频序列尾部帧的方式近似模拟原视频帧进行预编码过程，减少视频帧决策对码率分配的影响，尽量保持分割点前的画面清晰度不变。

[0051] 在步骤S304提供的技术方案中，编码服务器对第一视频片段中的第一视频帧进行复制得到第二视频帧，第一视频帧为位于第一视频片段中边缘位置的视频帧。

[0052] 在上述实施例中，对第一视频片段中的第一视频帧进行复制得到第二视频帧可包括以下两个方面中的至少之一：

[0053] 其一是，编码服务器对位于第一视频片段中的播放结束位置的第一视频帧进行复制得到第二视频帧，视频片段中的视频帧是与播放时刻具有对应关系的，如每个播放时刻

绑定对应的一个视频帧，换言之，每个视频帧均存在对应的播放时刻，若第一视频帧只有一帧，那么第一视频片段中最后一个播放时刻对应的视频帧即第一视频帧，若第一视频帧包括多帧，那么第一视频片段中最后多个播放时刻对应的视频帧即为第一视频帧，其中，边缘位置包括播放结束位置；其二是，编码服务器对位于第一视频片段中的播放起始位置的第一视频帧进行复制得到第二视频帧，其中，边缘位置包括播放起始位置；

[0054] 可选地，上述的第一视频帧可以为一帧视频帧，换言之，此时可以通过复制一个视频片段的尾部帧来进行预编码，在上述第一视频帧为多帧视频帧时就可使得预编码效果更为理想，编码服务器对位于第一视频片段中的播放结束位置的第一视频帧进行复制得到第二视频帧可以包括如下步骤1-步骤2：

[0055] 步骤1，确定位于第一视频片段的播放结束位置的待复制的多帧第一视频帧。

[0056] 可选地，确定位于第一视频片段的播放结束位置的待复制的多帧第一视频帧可包括：获取目标参数，目标参数用于表示编码器已经从视频帧缓冲区数组读取的视频帧的帧数，视频帧缓冲区数组用于缓存待编码器进行预编码的视频帧；在目标参数表示的帧数为第一阈值的情况下，确定位于第一视频片段的播放结束位置的N帧视频帧为多帧第一视频帧，第一阈值用于表示第一视频片段中视频帧的帧数，N为视频帧缓冲区数组的缓冲容量，N可与预编码长度的值相同。

[0057] 在本申请的实施例中，可以设置一个触发器，触发器中的计数器实时统计已经从视频帧缓冲区数组读取的视频帧数量，并判断是否读取完整个视频切片，若是，则将视频切片最后的N帧作为第一视频帧。

[0058] 步骤2，按照在第一视频片段中的播放时间对多帧第一视频帧进行复制，得到多帧第二视频帧，多帧第二视频帧按照播放时间从晚到早的顺序排列。

[0059] 可选地，按照在第一视频片段中的播放时间对多帧第一视频帧进行复制，得到多帧第二视频帧可包括：将多帧第一视频帧写入视频帧缓冲区数组，并发送指示信息给编码器(如通过后续的标志位btail告知编码器)，通过指示信息来指示编码器按照播放时间从晚到早的顺序从视频帧缓冲区数组中读取第一视频帧；编码器将按照播放时间从晚到早的顺序从视频帧缓冲区数组中读取的多帧第一视频帧作为多帧第二视频帧，如图5所示，可以对第一视频片段中的视频帧VFm+1至视频帧VFm+n进行复制，所采用的复制方式为镜面复制，也即按照播放时间从晚到早的顺序进行复制。

[0060] 上述实施例以编码服务器对位于第一视频片段中的播放结束位置的第一视频帧进行复制得到第二视频帧为例进行说明，类似地，编码服务器对位于第一视频片段中的播放起始位置的第一视频帧进行复制得到第二视频帧也可按照相似方式来实现，下面进行简单描述。

[0061] 可选地，编码服务器对位于第一视频片段中的播放起始位置的第一视频帧进行复制得到第二视频帧可包括如下步骤1-步骤2：

[0062] 步骤1，确定位于第一视频片段的播放起始位置的待复制的多帧第一视频帧。

[0063] 可选地，确定位于第一视频片段的播放结束位置的待复制的多帧第一视频帧可包括：确定位于第一视频片段的播放起始位置的N帧视频帧为多帧第一视频帧，N为视频帧缓冲区数组的缓冲容量。

[0064] 步骤2，按照在第一视频片段中的播放时间对多帧第一视频帧进行复制，得到多帧

第二视频帧，多帧第二视频帧按照播放时间从晚到早的顺序排列。

[0065] 可选地，按照在第一视频片段中的播放时间对多帧第一视频帧进行复制，得到多帧第二视频帧可包括：将多帧第一视频帧写入视频帧缓冲区数组，并发送指示信息给编码器，通过指示信息来指示编码器按照播放时间从晚到早的顺序从视频帧缓冲区数组中读取第一视频帧；编码器将按照播放时间从晚到早的顺序从视频帧缓冲区数组中读取的多帧第一视频帧作为多帧第二视频帧。

[0066] 如图6所示，可以对第一视频片段中的视频帧VF1至视频帧VFn进行复制，所采用的复制方式为镜面复制，也即按照播放时间从晚到早的顺序进行复制。

[0067] 在步骤S306提供的技术方案中，根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型。

[0068] 可选地，根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型可包括下述步骤1至步骤3：

[0069] 步骤1，根据关联关系确定第二帧类型，第一视频帧和第二视频帧为类指，换言之，第一视频帧可以为多帧，第二视频帧也可为多帧，关联关系为第一视频帧与第二视频帧之间、两个不同的第一视频帧之间以及两个不同的第二视频帧之间的关系，第二帧类型为第一视频帧或第二视频帧的帧类型。

[0070] 可选地，在预编码的帧类型决策中，编码器可读入n帧视频帧，将其1/2下采样图像存入数组中，然后利用下采样图像的复杂度决策未采样的前视频帧的帧类型，此处的复杂度即用于描述视频帧之间或视频帧内部的关联关系，如关键帧，I帧通常是每个GOP的第一个帧，经过适度地压缩，作为随机访问的参考点，可以当成图象。I帧可以看成是一个图像经过压缩后的产物，这一帧画面的完整保留，解码时只需要本帧数据就可以完成；再如，通过充分将低于图像序列中前面已编码帧的时间冗余信息来压缩传输数据量的编码图像，也叫预测帧，P帧表示的是这一帧跟之前的一个关键帧(或P帧)的差别，解码时需要用之前缓存的画面叠加上本帧定义的差别，生成最终画面，(也就是差别帧，P帧没有完整画面数据，只有与前一帧的画面差别的数据)。

[0071] 步骤2，根据第二帧类型对第一视频帧和第二视频帧进行组合，得到至少一个帧集合，至少一个帧集合的每个帧集合的集合类型包括以下至少之一：用于表示所包括的视频帧均为第一视频帧的第一类型、用于表示所包括的视频帧均为第二视频帧的第二类型以及用于表示包括第一视频帧和第二视频帧的第三类型。

[0072] 可选地，可根据率失真代价决策最优帧类型组合，将该最优帧类型组合成一个帧集合GOP。

[0073] 步骤3，根据目标集合的集合类型确定目标集合中第一视频帧的第一帧类型，目标集合为至少一个帧集合中的任意一个集合。

[0074] 可选地，根据目标集合的集合类型确定目标集合中第一视频帧的第一帧类型可包括：在目标集合的集合类型为第一类型的情况下，也即一个GOP中的图像帧均为第一视频帧，将目标集合中为第一视频帧确定的第二帧类型作为第一视频帧的第一帧类型；在目标集合的集合类型为第三类型的情况下，也即一个GOP中只要有复制帧(或称第二视频帧)，则可根据目标集合的第一视频帧的数量确定第一视频帧的第一帧类型。

[0075] 可选地，目标集合中的第一视频帧和第二视频帧按照目标顺序排列，根据目标集

合的第一视频帧的数量确定第一视频帧的第一帧类型包括：

[0076] 在目标集合的第一视频帧的帧数大于第二阈值的情况下，将第三视频帧的第二帧类型作为第三视频帧的第一帧类型，换言之，最后一个帧无需改变其帧类型，将第三帧类型作为第四视频帧的第一帧类型，例如，将倒数第二帧的帧类型修改为B帧，将第四帧类型作为第五视频帧的第一帧类型，例如，其余帧修改为b帧，第三视频帧为按照目标顺序排列时的最后一位的第一视频帧，第四视频帧是与第三视频帧相邻的第一视频帧，第五视频帧为多个第一视频帧中除第三视频帧和第四视频帧以外的视频帧，第二阈值为不小于2的整数；

[0077] 在目标集合的第一视频帧的帧数等于第二阈值的情况下，将第三视频帧的第二帧类型作为第三视频帧的第一帧类型，也即倒数最后一帧不用改变其帧类型，将第四帧类型作为第四视频帧的第一帧类型，如将倒数第二帧修改为b帧；

[0078] 在目标集合的第一视频帧的帧数小于第二阈值的情况下，将第一视频帧的第二帧类型作为第一视频帧的第一帧类型，也即，该视频帧帧可无需改变其帧类型。

[0079] 在步骤S308提供的技术方案中，按照第一帧类型对第一视频帧进行编码。

[0080] 可选地，按照第一帧类型对第一视频帧进行编码包括：通过第一编码器，按照第一帧类型对第一视频帧进行编码，在按照第一帧类型对第一视频帧进行编码的过程中，可通过第二编码器，对第二视频片段中的视频帧进行编码，多个视频片段包括第二视频片段，第一编码器和第二编码器为独立运行的编码器，第二视频片段中的视频帧的编码方式与第一视频片段中的类似。

[0081] 在上述实施例中，针对分布式转码中分割点过渡不平滑的缺陷，本申请提出了一种基于视频尾部帧或首部帧复制的解决方案，通过镜面复制视频序列尾部编码帧或首部编码帧，保证了分片转码过程中视频编码的预编码过程与未分片视频相对应的编码帧的预编码过程尽量保持一致，减小码率分配过程中分片视频与未分片视频之间的差别，使得主观感受上分片视频在分割点处与未分片视频相应点无差异。

[0082] 作为一种可选的实施例，下面结合具体的实施方式详述本申请的技术方案：

[0083] 本申请的技术方案主要适用于分布式视频转码场景，且不用改变整个分布式视频转码构架的设计架构和分发方案等，如图7所示，完整的分布式视频转码过程主要分为四个部分，视频序列物理分割、视频尾部帧复制、视频尾部帧类型决策、视频码流拼接，本申请的技术方案主要涉及其中的视频尾部帧复制和视频尾部帧类型决策，下文将在技术侧对整个发明方案进行详细说明，并对其中的专用词汇进行预先解释。

[0084] Buff[BufferSize]，视频帧缓冲区数组，若缓冲区为空或其中视频帧已全部被编码器读入并编码，则按顺序将yuv视频序列中的数据填入缓冲区中并将之前的数据完全覆盖，可缩写为Buff[BS]，BufferSize或BS表示视频帧缓冲区数组的缓冲容量。

[0085] Read，表示编码器已累计从视频帧缓冲区中读取的视频帧数。

[0086] Written，表示编码器已累计往视频帧缓冲区中写入的视频帧数。

[0087] 视频序列物理分割的流程如下：

[0088] 步骤1，获取原始视频序列VideoSeq.yuv(即目标视频的视频帧序列)，帧数为OrgFrames(后文缩写为OrgF)。

[0089] 步骤2，根据转码构架中设定的分片帧数SegFrames(后文缩写为SegF)将视频序列VideoSeq.yuv切割成SegF帧为一个片段，片段分别为VideoSeq_0.yuv(如第一视频片段)和

VideoSeq_1.yuv(如第二视频片段),为了方便说明,后文将以VideoSeq_0.yuv为例进行说明。

[0090] 步骤3,将片段VideoSeq_0.yuv和VideoSeq_1.yuv分发至不同的编码器(或编码器所在的服务器,如前述的编码服务器)上,分别对其进行编码。

[0091] 视频尾部帧复制的流程如下:

[0092] 步骤1,赋值视频帧缓冲区数组大小BS=LaD。

[0093] 步骤2,编码器读入原始视频VideoSeq_0.yuv,并根据视频序列大小和尺寸大小计算出输入视频帧数InputFrames(缩写为InF)。

[0094] 步骤3,利用步骤1获取的视频帧缓冲区数组Buff和步骤2获取的InF赋值总编码帧TotalFrames(缩写为TotF),TotF=InF+BS,即获取预编码的总帧数,包括实际需要编码的帧数InF和用于参数预编码的帧数BS。

[0095] 步骤4,获取Read和Written数值,并初始化标志位btail=false,标志位btail用于需要进行尾部帧复制操作。

[0096] 步骤5,利用步骤4获取的信息建立判断式子:

[0097] InF≤Read≤TotF。

[0098] 若满足上述式子,则转到步骤6,否则转到步骤7,也即判断是否已经将所有输入视频帧读取完,若是,则需要进行尾部帧复制了。

[0099] 步骤6,进行如下赋值操作,Written=Written+BS,btail=true(true表示为真)。

[0100] 第一个公式是认为编码器再次往视频帧缓冲区中写入了BS帧(实际依然是尾部的BS帧视频),第二个公式是为了标记是否需要进行尾部帧复制了(是否是尾巴的意思)。

[0101] 步骤7,判断公式Read==Written是否成立,若是则转到步骤8,否则转到步骤9。

[0102] 步骤8,重新往视频缓冲区中写入BS帧图像的数据并覆盖之前的数据。

[0103] 步骤9,判断公式Read<Written是否成立,若是则转到步骤10,否则转到步骤13。

[0104] 步骤10,判断btail是否为真,若是则转到步骤11,否则转到步骤12。

[0105] 步骤11,读取视频帧缓冲区数组Buff中的Buff[BS-1-Read%BS]帧,即实现尾部帧复制。

[0106] 在对视频尾部帧进行复制的方案上,除了对称镜面复制方式,还可以采取顺序复制的方式,视频帧尾部复制部分的步骤11修改为:读取视频帧缓冲区数组Buff中的Buff[Read%BS]帧,。

[0107] 步骤12,读取视频帧缓冲区数组Buff中的Buff[Read%BS]帧,%表示求余数。

[0108] 编码器是根据帧类型进行码率分配的,这一部分的主要作用是为了保证在帧类型决策的时候尾部帧不会将复制出来的帧作为参考帧,因为复制出来的帧只是用来在预编码过程中起作用,而不会被写入码流文件。视频尾部帧类型决策的流程如下:

[0109] 步骤1,在预编码的帧类型决策中,编码器读入LaD帧,将其1/2下采样图像存入frame[LaD]数组中,将未采样的前(bfs+2)帧存入list[bfs+2]数组中。

[0110] 步骤2,利用步骤1获得的frame数组中下采样图像的复杂度决策帧类型,如根据率失真代价决策最优帧类型组合,将该最优帧类型组合赋值给步骤1中的list数组,初始化list中每个视频帧的bBref_before标志位(该标志位用于记录视频帧是否为B_REF帧)为0,并获取list中第1次出现非B帧(包含b帧和B_REF帧)的位置并记序号为nonb_idx(标记list

数组中第一次出现非B帧,包含b帧和B_REF帧,的位置序号)。即可根据bfs的值遍历所有帧类型组合的情况,找到最优组合(率失真代价最小的为最优)。

[0111] 步骤3,利用步骤2中得到的存储了最优帧类型组合的list数组,生成视频帧标志位bpushed[8](标志着list数组中各个视频帧状态,-1表示不存在,0表示存在但未被推入输出列表,为编码器的编码列表,1表示存在且已被推入输出列表)和bfirst_half[8](用于标志着视频帧属于实际帧还是属于复制帧),表达式如下:

[0112] $$bpushed[i]=\begin{cases}0, & if\ list[i]\neq NULL\\ -1, & if\ list[i]=NULL\end{cases};$$

[0113] $$bfirst_half[i]=\begin{cases}0, & if\ list[i]\neq NULL\\ -1, & if\ list[i]=NULL\end{cases};$$

[0114] 其中,if表示判断条件,NULL表示为空,若满足if后的条件,则bpushed[i]或bfirst_half[i]的取值为相应条件下的取值。

[0115] 其中,list[i]表示list数组中第i个元素,判断list数组是否横跨了分割点视频帧,若是,则初始化标志位bsplit=true并转到步骤4,否则,标志位bsplit=false(false表示为假)并转到步骤6。

[0116] 步骤4,利用步骤2获得的list数组和nonb_idx获取list[nonb_idx]的视频帧序号POC_{nonb_idx},判断条件POC_{nonb_idx}<InF是否成立,若是则转到步骤5,是就记录为first half;否则转到步骤7,判断当前帧是否为原始帧,以确定是否为复制帧。

[0117] 步骤5,赋值bfirst_half[nonb_idx]=1,然后转到步骤7。

[0118] 步骤6,将list[nonb_idx]视频帧推入输出列表,按顺序设置编码序号PTS,同时设置bpushed[nonb_idx]=1,继续后续步骤。

[0119] 步骤7,判断nonb_idx>1是否成立,若是则初始化i=0,并转到步骤8,以遍历步骤3获得的list数组中序号为0～(nonb_idx-1)的视频帧,否则初始化i=0,并转到步骤15。

[0120] 步骤8,判断条件i<nonb_idx是否成立,若是则转到步骤9,否则初始化i=0并转到步骤15。

[0121] 步骤9,判断sT_i=B_REF是否成立,若是,则转到步骤10,否则转到步骤14,其中sT_i表示list中第i个元素的帧类型。

[0122] 步骤10,判断bsplit=true是否成立,若是则转到步骤11,否则转到步骤13。

[0123] 步骤11,判断POC_i<InF是否成立,若是则转到步骤12,否则转到步骤14。

[0124] 步骤12,进行以下赋值操作:

[0125] sT_i=P,$bBref_before_i$=1,bfirst_half[i]=1,然后转到步骤14。

[0126] 步骤13,将list[i]视频帧推入输出列表,按顺序设置编码序号PTS,同时赋值bpushed[i]=1,然后继续后续步骤。

[0127] 步骤14,操作i=i+1,然后转到步骤8。

[0128] 步骤15,判断条件i<nonb_idx是否成立,若是则转到步骤16,否则转到步骤22。

[0129] 步骤16,判断以下两个条件:

[0130] $sT_i\neq$B_REF;

[0131] $bBref_before_i\neq1$;

[0132] 若上述两个公式同时成立,则转到步骤17,否则转到步骤21。

[0133] 步骤17,判断bsplit＝true是否成立,若是则转到步骤18,否则转到步骤20。

[0134] 步骤18,判断POC_i＜InF是否成立,若是则转到步骤19,否则转到步骤21。

[0135] 步骤19,进行以下赋值操作:sT_i＝P,bfirst_half[i]＝1,然后转到步骤21。

[0136] 步骤20,将list[i]视频帧推入输出列表,按顺序设置编码序号PTS,同时赋值bpushed[i]＝1,然后继续后续步骤。

[0137] 步骤21,操作i＝i+1,然后转到步骤15。

[0138] 步骤22,判断bsplit＝true是否成立,若是则转到步骤23,否则结束帧类型决策过程。

[0139] 步骤23,初始化以下参数:i＝0、nfirst_half＝0、数组first_halfidx[8]＝{-1,-1,-1,-1,-1,-1,-1,-1},然后继续后续步骤。

[0140] 步骤24,判断条件i≤nonb_idx是否成立,若是则转到步骤25,否则转到步骤28。

[0141] 步骤25,判断bpushed[i]＝0和bfirst_half[i]＝1是否成立,若是则转到步骤26,否则转到步骤27。

[0142] 步骤26,进行以下赋值操作:first_halfidx[nfirst_half]＝i,nfirst_half＝nfirst_half+1,然后继续后续步骤。

[0143] 步骤27,操作i＝i+1,然后转到步骤24。

[0144] 步骤28,判断nfirst_half>2是否成立,若是则转到步骤29,否则转到步骤34。

[0145] 步骤29,将list[first_halfidx[nfirst_half-1]]视频帧推入输出列表,按顺序设置编码序号PTS,同时赋值bpushed[first_halfidx[nfirst_half-1]]＝1,然后继续后续步骤。

[0146] 步骤30,赋值$sT_{first_halfidx[nfirst_half-2]}$＝B_REF(即确定为B帧),然后将list[first_halfidx[nfirst_half-2]]视频帧推入输出列表,按顺序设置编码序号PTS,同时赋值bpushed[first_halfidx[nfirst_half-2]]＝1,然后初始化i＝nfirst_half-3并继续后续步骤。

[0147] 步骤31,判断i≥0是否成立,若是则转到步骤32,否则转到步骤34。

[0148] 步骤32,赋值$sT_{first_halfidx[i]}$＝b,然后将list[first_halfidx[i]]视频帧推入输出列表,按顺序设置编码序号PTS,同时赋值bpushed[first_halfidx[i]]＝1,然后继续后续步骤。

[0149] 步骤33,操作i＝i-1,然后转到步骤31。

[0150] 步骤34,判断nfirst_half＝2是否成立,若是则转到步骤35,否则转到步骤37。

[0151] 步骤35,将list[first_halfidx[nfirst_half-1]]视频帧推入输出列表,按顺序设置编码序号PTS,同时赋值bpushed[first_halfidx[nfirst_half-1]]＝1,然后继续后续步骤。

[0152] 步骤36,赋值$sT_{first_halfidx[nfirst_half-2]}$＝b,然后将list[first_halfidx[nfirst_half-2]]视频帧推入输出列表,按顺序设置编码序号PTS,同时赋值bpushed[first_halfidx[nfirst_half-2]]＝1,然后继续后续步骤。

[0153] 步骤37,判断nfirst_half＝1是否成立,若是则转到步骤38,否则初始化i＝0并转到步骤39。

[0154] 步骤38,将list[first_halfidx[nfirst_half-1]]视频帧推入输出列表,按顺序

设置编码序号PTS,同时赋值bpushed[first_halfidx[nfirst_half-1]]＝1,然后初始化i＝0并继续后续步骤。

[0155] 步骤39,判断i≤nonb_idx是否成立,若是则转到步骤40,否则初始化i＝0并转到步骤43。

[0156] 步骤40,判断以下3个条件:bpushed[i]＝0、sT_i＝P、sT_i＝I,

[0157] 若满足条件第一个和第二个,或满足第一个和第三个,则转到步骤41,否则转到步骤42。

[0158] 步骤41,将list[i]视频帧推入输出列表,按顺序设置编码序号PTS,同时设置bpushed[i]＝1,继续后续步骤。

[0159] 步骤42,操作i＝i+1,然后转到步骤39。

[0160] 步骤43,判断i≤nonb_idx是否成立,若是则转到步骤44,否则初始化i＝0并转到步骤47。

[0161] 步骤44,判断以下两个条件:bpushed[i]＝0、sT_i＝BREF,

[0162] 若同时满足上述两个条件,则转到步骤45,否则转到步骤46。

[0163] 步骤45,将list[i]视频帧推入输出列表,按顺序设置编码序号PTS,同时设置bpushed[i]＝1,继续后续步骤。

[0164] 步骤46,操作i＝i+1,然后转到步骤43。

[0165] 步骤47,判断i≤nonb_idx是否成立,若是则转到步骤48,否则结束帧类型决策过程。

[0166] 步骤48,判断以下两个条件:bpushed[i]＝0、sT_i＝b,

[0167] 若同时满足上述两个条件,则转到步骤49,否则转到步骤50。

[0168] 步骤49,将list[i]视频帧推入输出列表,按顺序设置编码序号PTS,同时设置bpushed[i]＝1,继续后续步骤。

[0169] 步骤50,操作i＝i+1,然后转到步骤47。

[0170] 视频码流拼接的流程如下:

[0171] 将片段VideoSeq_0.yuv和VideoSeq_1.yuv分别进行视频尾部帧复制和视频尾部帧类型决策,然后分别进行编码生成码流文件VideoSeq_0.bin和VideoSeq_1.bin,将上述两个码流文件进行拼接,生成最终码流文件。

[0172] 在一个更为具体的实施例中,若切片(第一视频片段)为11帧,预编码长度为3帧,每一帧的帧类型可由当前帧和后续3帧决定(表1中帧类型仅为示例,B为B_REF帧简写)。视频帧编码顺序与视频图像实际顺序不同,实际编码顺序PTS为单数的为复制所得到的帧。

[0173] 表1

[0174]

	未分片		分片		加了算法的分片	
实际编码顺序 PTS	帧类型	视频图像顺序 POC	帧类型	视频图像顺序 POC	帧类型	视频图像顺序 POC
0	I-SLICE	0	I-SLICE	0	I-SLICE	0
1	P-SLICE	4	P-SLICE	4	P-SLICE	4
2	B-SLICE	2	B-SLICE	2	B-SLICE	2
3	b-SLICE	1	b-SLICE	1	b-SLICE	1
4	b-SLICE	3	b-SLICE	3	b-SLICE	3
5	P-SLICE	8	P-SLICE	7	P-SLICE	8
6	B-SLICE	6	B-SLICE	5	B-SLICE	6
7	b-SLICE	5	b-SLICE	6	b-SLICE	5
8	b-SLICE	7	P-SLICE	10	b-SLICE	7
9	P-SLICE	13	B-SLICE	8	P-SLICE	10
10	B-SLICE	10	b-SLICE	9	b-SLICE	9
11	b-SLICE	9	I-SLICE	11	P-SLICE	13
12	b-SLICE	11	P-SLICE	13	b-SLICE	11
13	b-SLICE	12	B-SLICE	12	b-SLICE	12

[0175] 如图8所示，未分片的三个小节的帧类型为：IPBbb、PBbb、PBbbb；

[0176] 分片的三个小节的帧类型为：IPBbb、PBb、PBb；

[0177] 加了算法的分片的三个小节的帧类型为：IPBbb PBbb Pb(Pbb不进行编码)，预编码过程中实际编码顺序8、9、10帧，用11、12、13帧作为参考，因此帧类型分配与未分片时近似。

[0178] 表2

[0179]

	未分片	
视频图像顺序 POC	帧类型	实际编码顺序 PTS
0	I-SLICE	0
1	b-SLICE	3
2	B-SLICE	2
3	b-SLICE	4
4	P-SLICE	1
5	b-SLICE	8
6	B-SLICE	6
7	b-SLICE	5
8	P-SLICE	7

[0180]

9	b-SLICE	11
10	B-SLICE	10
11	b-SLICE	12
12	b-SLICE	13
13	P-SLICE	9

[0181] 以表2所示的未分片视频为例，按视频图像排序为：IbBbP、bBbP、bBbbP，按编码顺序排序：IPBbb PBbb PBbbb。

[0182] 编码优先级依次为I帧，P帧，B_REF帧，b帧，因此帧类型决策过程一个小节为单位，根据率失真优化决定，依次将I，P，B_REF，b帧推入输出列表(类似于通道)。

[0183] 加了算法的分片的视频尾部帧类型决策如下：

[0184] 步骤1，编码器利用率失真优化得到第一小节最优帧类型组合为IbBbP，每次得到最优帧类型组合都判断当前小节是否包含了分片点，若没有包含切片点，则按照I，P，B，b顺序推入输出列表为IPBbb(与本流程后续步骤6中的三次遍历相同)(具体实现方式参见视频尾部帧类型决策的流程的步骤4—步骤21)。

[0185] 步骤2，编码器利用率失真优化得到第二小节最优帧类型组合为bBbP，若没有包含切片点，按照I，P，B，b顺序推入输出列表为PBbb(具体实现方式参见视频尾部帧类型决策的流程的步骤4—步骤21)。

[0186] 步骤3，编码器利用率失真优化得到第三小节最优帧类型组合为bBbbP，判断得到当前小节包含了切片点(具体实现方式参见视频尾部帧类型决策的流程的步骤1—步骤3)。

[0187] 步骤4，处理bBbbP这一小节，首先标记出来哪些帧是需要编码的Need帧(POC<＝10，即属于当前小节的前段bfirst_half＝1)，并将这些帧的帧类型全部修改为P帧(PPbbP)，哪些不需要编码NoNeed帧(POC>10，即复制得到的帧，属于当前小节的后段bfirst_half＝0)(具体实现方式参见视频尾部帧类型决策的流程的步骤4—步骤21)。

[0188] 步骤5，将Need帧的帧数分为三种情况进行处理：

[0189] 其一，Need帧的帧数>2：若为PPP(具体实现方式参见视频尾部帧类型决策的流程的步骤28—步骤33)；

[0190] 1)先将PPP中最后一个P推入输出列表(具体实现方式参见视频尾部帧类型决策的流程的步骤29)；

[0191] 2)将倒数第二个P修改为B帧PBP并推入输出列表(具体实现方式参见视频尾部帧类型决策的流程的步骤30)；

[0192] 3)然后倒序将剩余的P修改为b帧bBP并推入输出列表(具体实现方式参见视频尾部帧类型决策的流程的步骤31—步骤33)；

[0193] 其二，Need帧的帧数＝2(具体实现方式参见视频尾部帧类型决策的流程的步骤34—步骤36)；

[0194] 1)先将PP中最后一个P推入输出列表(具体实现方式参见视频尾部帧类型决策的流程的步骤35)；

[0195] 2)然后将倒数第二个改为b帧bP并将其推入输出列表(具体实现方式参见视频尾部帧类型决策的流程的步骤36)。

[0196] 其三，Need帧的帧数＝1：将该P推入输出列表(具体实现方式参见视频尾部帧类型决策的流程的步骤37—步骤38)。

[0197] 步骤6，按IPBb顺序将NoNeed推入输出列表，首先遍历bPbbP，将NoNeed帧且为P的帧推入输出列表(bPP)(具体实现方式参见视频尾部帧类型决策的流程的步骤39—步骤42)；其次遍历bPbbP，将NoNeed帧且为B的帧推入输出列表(bPP)(具体实现方式参见视频尾部帧类型决策的流程的步骤43—步骤46)；然后遍历bPbbP，将NoNeed帧且为b的帧推入输出

列表(bPPbb)(具体实现方式参见视频尾部帧类型决策的流程的步骤47—步骤50)。

[0198] 为了验证算法性能的好坏,本申请采用实时编码器进行验证,采用$\Delta\sigma^2$来进行评估,σ^2表示编码分割点前10帧YUVPSNR值的方差,其中YUV PSNR为评价图像质量的客观标准,该值越大就代表失真越少;而方差表达了某组数值内部彼此波动的程度,越大波动越大,因此分割点前10帧YUVPSNR值的方差σ^2反映了分割点前视频质量的连贯性,其计算公式如下所示:

[0199] $$\sigma^2 = \frac{\sum(X-\mu)^2}{10}$$

[0200] 其中,X为每1帧的YUVPSNR值,μ为YUVPSNR均值,10表示帧数。σ^2越小,分割点之前的视频质量越连贯。$\Delta\sigma^2$则是反映当前方法对分割点前视频质量连贯性的提升,其计算公式如下所示:

[0201] $$\Delta\sigma^2 = \frac{\sigma^2_{org}-\sigma^2_{new}}{\sigma^2_{org}} \times 100\%,$$

[0202] 其中,σ^2_{org}代表分割的视频不加任何算法的分割点前10帧YUVPSNR值的方差,σ^2_{new}则代表分割的视频加重叠分片技术后分割点前10帧YUVPSNR值的方差,$\Delta\sigma^2$则代表加入重叠分片技术后分割视频的尾部质量连贯性提升的百分比。

[0203] 通过测试仿真,本申请的技术方案的测试结果如表3所示。

[0204] 表3

[0205]

测试序列	σ^2_{org}	σ^2_{new}	$\Delta\sigma^2$
测试样例 1	0.558867	0.324186	41.992%
测试样例 2	1.205720	0.541195	55.114%
测试样例 3	0.728196	0.281904	61.287%
平均			52.798%

[0206] 根据测试仿真结果表3可知,加入本技术方案后分割点前10帧YUV PSNR值的方差减小了52%,由此可以看出,本申请提升了分布式视频转码中分割点前视频质量的连贯性,达到了提升连贯性的目的。

[0207] 需要说明的是,对于前述的各方法实施例,为了简单描述,故将其都表述为一系列的动作组合,但是本领域技术人员应该知悉,本发明并不受所描述的动作顺序的限制,因为依据本发明,某些步骤可以采用其他顺序或者同时进行。其次,本领域技术人员也应该知悉,说明书中所描述的实施例均属于优选实施例,所涉及的动作和模块并不一定是本发明所必须的。

[0208] 通过以上的实施方式的描述,本领域的技术人员可以清楚地了解到根据上述实施例的方法可借助软件加必需的通用硬件平台的方式来实现,当然也可以通过硬件,但很多情况下前者是更佳的实施方式。基于这样的理解,本发明的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质(如ROM/RAM、磁碟、光盘)中,包括若干指令用以使得一台终端设备(可以是手机,计算

机,服务器,或者网络设备等)执行本发明各个实施例所述的方法。

[0209] 根据本发明实施例的另一个方面,还提供了一种用于实施上述视频的编码方法的视频的编码装置。图9是根据本发明实施例的一种可选的视频的编码装置的示意图,如图9所示,该装置可以包括:

[0210] 获取单元901,用于获取待编码的第一视频片段,其中,第一视频片段为目标视频包括的多个视频片段中的一个,多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码;

[0211] 复制单元903,用于对第一视频片段中的第一视频帧进行复制得到第二视频帧,其中,第一视频帧为位于第一视频片段中边缘位置的视频帧;

[0212] 确定单元905,用于根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型;

[0213] 编码单元907,用于按照第一帧类型对第一视频帧进行编码。

[0214] 需要说明的是,该实施例中的获取单元901可以用于执行本申请实施例中的步骤S302,该实施例中的复制单元903可以用于执行本申请实施例中的步骤S304,该实施例中的确定单元905可以用于执行本申请实施例中的步骤S306,该实施例中的编码单元907可以用于执行本申请实施例中的步骤S308。

[0215] 此处需要说明的是,上述模块与对应的步骤所实现的示例和应用场景相同,但不限于上述实施例所公开的内容。需要说明的是,上述模块作为装置的一部分可以运行在如图2所示的硬件环境中,可以通过软件实现,也可以通过硬件实现。

[0216] 通过上述模块,通过获取待编码的第一视频片段,第一视频片段为目标视频包括的多个视频片段中的一个,多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码;对第一视频片段中的第一视频帧进行复制得到第二视频帧,第一视频帧为位于第一视频片段中边缘位置的视频帧;根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型;按照第一帧类型对第一视频帧进行编码,采用复制视频序列边缘位置帧的方式近似模拟原视频帧进行预编码过程,利用视频序列在时间上的相关性(或称为时域相关性,是指视频序列中前后帧图像的相似性)以及预编码过程对码率分配的影响来减小分割点处的质量波动,利用时域相关性可以进行尾部帧复制模拟原视频帧情况,然后预编码的帧决策与原视频就比较相近,所以码率分配也与原视频较为相似,可以解决相关技术中导致视频画面的清晰度失去连贯性的技术问题,进而达到减少了视频帧决策对码率分配的影响、保持了分割点前的画面清晰度不变的技术效果。

[0217] 上述复制单元可包括:第一复制模块,用于对位于第一视频片段中的播放结束位置的第一视频帧进行复制得到第二视频帧,其中,边缘位置包括播放结束位置;第二复制模块,用于对位于第一视频片段中的播放起始位置的第一视频帧进行复制得到第二视频帧,其中,边缘位置包括播放起始位置。

[0218] 可选地,第一视频帧可为多帧,第一复制模块还可用于:确定位于第一视频片段的播放结束位置的待复制的多帧第一视频帧;按照在第一视频片段中的播放时间对多帧第一视频帧进行复制,得到多帧第二视频帧,其中,多帧第二视频帧按照播放时间从晚到早的顺序排列。

[0219] 可选地,第一复制模块还可用于:获取目标参数,其中,目标参数用于表示编码器已经从视频帧缓冲区数组读取的视频帧的帧数,视频帧缓冲区数组用于缓存待编码器进行

预编码的视频帧;在目标参数表示的帧数为第一阈值的情况下,确定位于第一视频片段的播放结束位置的N帧视频帧为多帧第一视频帧,其中,第一阈值用于表示第一视频片段中视频帧的帧数,N为视频帧缓冲区数组的缓冲容量(与预编码长度相同)。

[0220] 可选地,第一复制模块还可用于:将多帧第一视频帧写入视频帧缓冲区数组,并发送指示信息给编码器,其中,指示信息用于指示编码器按照播放时间从晚到早的顺序从视频帧缓冲区数组中读取第一视频帧;通过编码器,将按照播放时间从晚到早的顺序从视频帧缓冲区数组中读取的多帧第一视频帧作为多帧第二视频帧。

[0221] 在一个可选的实施例中,上述确定单元可包括:第一确定模块,用于根据关联关系确定第二帧类型,其中,关联关系为第一视频帧与第二视频帧之间、第一视频帧与第一视频帧之间以及第二视频帧与第二视频帧之间的关系,第二帧类型为第一视频帧或第二视频帧的帧类型;组合模块,用于根据第二帧类型对第一视频帧和第二视频帧进行组合,得到至少一个帧集合,至少一个帧集合的每个帧集合的集合类型包括以下至少之一:用于表示所包括的视频帧均为第一视频帧的第一类型、用于表示所包括的视频帧均为第二视频帧的第二类型以及用于表示包括第一视频帧和第二视频帧的第三类型;第二确定模块,用于根据目标集合的集合类型确定目标集合中第一视频帧的第一帧类型,其中,目标集合为至少一个帧集合中的任意一个集合。

[0222] 上述第二确定模块可用于:在目标集合的集合类型为第一类型的情况下,将目标集合中为第一视频帧确定的第二帧类型作为第一视频帧的第一帧类型;在目标集合的集合类型为第三类型的情况下,根据目标集合的第一视频帧的数量确定第一视频帧的第一帧类型。

[0223] 可选地,上述的目标集合中的第一视频帧和第二视频帧按照目标顺序排列,其中,上述第二确定模块还可用于:

[0224] 在目标集合的第一视频帧的帧数大于第二阈值的情况下,将第三视频帧的第二帧类型作为第三视频帧的第一帧类型,将第三帧类型作为第四视频帧的第一帧类型,将第四帧类型作为第五视频帧的第一帧类型,其中,第三视频帧为按照目标顺序排列时的最后一位的第一视频帧,第四视频帧是与第三视频帧相邻的第一视频帧,第五视频帧为多个第一视频帧中除第三视频帧和第四视频帧以外的视频帧,第二阈值为不小于2;

[0225] 在目标集合的第一视频帧的帧数等于第二阈值的情况下,将第三视频帧的第二帧类型作为第三视频帧的第一帧类型,将第四帧类型作为第四视频帧的第一帧类型;

[0226] 在目标集合的第一视频帧的帧数小于第二阈值的情况下,将第一视频帧的第二帧类型作为第一视频帧的第一帧类型。

[0227] 在又一个可选的实施例中,上述编码单元还可用于:通过第一编码器,按照第一帧类型对第一视频帧进行编码,其中,多个视频片段中的第二视频片段中的视频帧通过第二编码器进行编码,第一编码器和第二编码器为独立运行的编码器。

[0228] 在上述实施例中,针对分布式转码中分割点过渡不平滑的缺陷,本申请提出了一种基于视频尾部帧复制的解决方案,通过镜面复制视频序列尾部编码帧,保证了分片转码过程中视频编码的预编码过程与未分片视频相对应的编码帧的预编码过程尽量保持一致,减小码率分配过程中分片视频与未分片视频之间的差别,使得主观感受上分片视频在分割点处与未分片视频相应点无差异。

[0229] 此处需要说明的是，上述模块与对应的步骤所实现的示例和应用场景相同，但不限于上述实施例所公开的内容。需要说明的是，上述模块作为装置的一部分可以运行在如图2所示的硬件环境中，可以通过软件实现，也可以通过硬件实现，其中，硬件环境包括网络环境。

[0230] 根据本发明实施例的另一个方面，还提供了一种用于实施上述视频的编码方法的服务器或终端。

[0231] 图10是根据本发明实施例的一种终端的结构框图，如图10所示，该终端可以包括：一个或多个(图10中仅示出一个)处理器1001、存储器1003、以及传输装置1005(如上述实施例中的发送装置)，如图10所示，该终端还可以包括输入输出设备1007。

[0232] 其中，存储器1003可用于存储软件程序以及模块，如本发明实施例中的视频的编码方法和装置对应的程序指令/模块，处理器1001通过运行存储在存储器1003内的软件程序以及模块，从而执行各种功能应用以及数据处理，即实现上述的视频的编码方法。存储器1003可包括高速随机存储器，还可以包括非易失性存储器，如一个或者多个磁性存储装置、闪存、或者其他非易失性固态存储器。在一些实例中，存储器1003可进一步包括相对于处理器1001远程设置的存储器，这些远程存储器可以通过网络连接至终端。上述网络的实例包括但不限于互联网、企业内部网、局域网、移动通信网及其组合。

[0233] 上述的传输装置1005用于经由一个网络接收或者发送数据，还可以用于处理器与存储器之间的数据传输。上述的网络具体实例可包括有线网络及无线网络。在一个实例中，传输装置1005包括一个网络适配器(Network Interface Controller，NIC)，其可通过网线与其他网络设备与路由器相连从而可与互联网或局域网进行通讯。在一个实例中，传输装置1005为射频(Radio Frequency，RF)模块，其用于通过无线方式与互联网进行通讯。

[0234] 其中，具体地，存储器1003用于存储应用程序。

[0235] 处理器1001可以通过传输装置1005调用存储器1003存储的应用程序，以执行下述步骤：

[0236] 获取待编码的第一视频片段，其中，第一视频片段为目标视频包括的多个视频片段中的一个，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；

[0237] 对第一视频片段中的第一视频帧进行复制得到第二视频帧，其中，第一视频帧为位于第一视频片段中边缘位置的视频帧；

[0238] 根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型；

[0239] 按照第一帧类型对第一视频帧进行编码。

[0240] 处理器1001还用于执行下述步骤：

[0241] 在目标集合的第一视频帧的帧数大于第二阈值的情况下，将第三视频帧的第二帧类型作为第三视频帧的第一帧类型，将第三帧类型作为第四视频帧的第一帧类型，将第四帧类型作为第五视频帧的第一帧类型，其中，第三视频帧为按照目标顺序排列时的最后一位的第一视频帧，第四视频帧是与第三视频帧相邻的第一视频帧，第五视频帧为多个第一视频帧中除第三视频帧和第四视频帧以外的视频帧，第二阈值为不小于2；

[0242] 在目标集合的第一视频帧的帧数等于第二阈值的情况下，将第三视频帧的第二帧类型作为第三视频帧的第一帧类型，将第四帧类型作为第四视频帧的第一帧类型；

[0243] 在目标集合的第一视频帧的帧数小于第二阈值的情况下，将第一视频帧的第二帧

类型作为第一视频帧的第一帧类型。

[0244] 采用本发明实施例，通过获取待编码的第一视频片段，第一视频片段为目标视频包括的多个视频片段中的一个，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；对第一视频片段中的第一视频帧进行复制得到第二视频帧，第一视频帧为位于第一视频片段中边缘位置的视频帧；根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型；按照第一帧类型对第一视频帧进行编码，采用复制视频序列边缘位置帧的方式近似模拟原视频帧进行预编码过程，利用视频序列在时间上的相关性(或称为时域相关性，是指视频序列中前后帧图像的相似性)以及预编码过程对码率分配的影响来减小分割点处的质量波动，利用时域相关性可以进行尾部帧复制模拟原视频帧情况，然后预编码的帧决策与原视频就比较相近，所以码率分配也与原视频较为相似，可以解决相关技术中导致视频画面的清晰度失去连贯性的技术问题，进而达到减少了视频帧决策对码率分配的影响、保持了分割点前的画面清晰度不变的技术效果。

[0245] 可选地，本实施例中的具体示例可以参考上述实施例中所描述的示例，本实施例在此不再赘述。

[0246] 本领域普通技术人员可以理解，图10所示的结构仅为示意，终端可以是智能手机(如Android手机、iOS手机等)、平板电脑、掌上电脑以及移动互联网设备(Mobile Internet Devices，MID)、PAD等终端设备。图10其并不对上述电子装置的结构造成限定。例如，终端还可包括比图10中所示更多或者更少的组件(如网络接口、显示装置等)，或者具有与图10所示不同的配置。

[0247] 本领域普通技术人员可以理解上述实施例的各种方法中的全部或部分步骤是可以通过程序来指令终端设备相关的硬件来完成，该程序可以存储于一计算机可读存储介质中，存储介质可以包括：闪存盘、只读存储器(Read-Only Memory，ROM)、随机存取器(Random Access Memory，RAM)、磁盘或光盘等。

[0248] 本发明的实施例还提供了一种存储介质。可选地，在本实施例中，上述存储介质可以用于执行视频的编码方法的程序代码。

[0249] 可选地，在本实施例中，上述存储介质可以位于上述实施例所示的网络中的多个网络设备中的至少一个网络设备上。

[0250] 可选地，在本实施例中，存储介质被设置为存储用于执行以下步骤的程序代码：

[0251] S12，获取待编码的第一视频片段，其中，第一视频片段为目标视频包括的多个视频片段中的一个，多个视频片段中的每个视频片段被分配给相应的一个编码器进行编码；

[0252] S14，对第一视频片段中的第一视频帧进行复制得到第二视频帧，其中，第一视频帧为位于第一视频片段中边缘位置的视频帧；

[0253] S16，根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型；

[0254] S18，按照第一帧类型对第一视频帧进行编码。

[0255] 可选地，存储介质还被设置为存储用于执行以下步骤的程序代码：

[0256] S22，在目标集合的第一视频帧的帧数大于第二阈值的情况下，将第三视频帧的第二帧类型作为第三视频帧的第一帧类型，将第三帧类型作为第四视频帧的第一帧类型，将第四帧类型作为第五视频帧的第一帧类型，其中，第三视频帧为按照目标顺序排列时的最后一位的第一视频帧，第四视频帧是与第三视频帧相邻的第一视频帧，第五视频帧为多个

第一视频帧中除第三视频帧和第四视频帧以外的视频帧，第二阈值为不小于2；

[0257] S24，在目标集合的第一视频帧的帧数等于第二阈值的情况下，将第三视频帧的第二帧类型作为第三视频帧的第一帧类型，将第四帧类型作为第四视频帧的第一帧类型；

[0258] S26，在目标集合的第一视频帧的帧数小于第二阈值的情况下，将第一视频帧的第二帧类型作为第一视频帧的第一帧类型。

[0259] 可选地，本实施例中的具体示例可以参考上述实施例中所描述的示例，本实施例在此不再赘述。

[0260] 可选地，在本实施例中，上述存储介质可以包括但不限于：U盘、只读存储器(ROM，Read-Only Memory)、随机存取存储器(RAM，Random Access Memory)、移动硬盘、磁碟或者光盘等各种可以存储程序代码的介质。

[0261] 上述本发明实施例序号仅仅为了描述，不代表实施例的优劣。

[0262] 上述实施例中的集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用时，可以存储在上述计算机可读取的存储介质中。基于这样的理解，本发明的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的全部或部分可以以软件产品的形式体现出来，该计算机软件产品存储在存储介质中，包括若干指令用以使得一台或多台计算机设备(可为个人计算机、服务器或者网络设备等)执行本发明各个实施例所述方法的全部或部分步骤。

[0263] 在本发明的上述实施例中，对各个实施例的描述都各有侧重，某个实施例中没有详述的部分，可以参见其他实施例的相关描述。

[0264] 在本申请所提供的几个实施例中，应该理解到，所揭露的客户端，可通过其它的方式实现。其中，以上所描述的装置实施例仅仅是示意性的，例如所述单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，单元或模块的间接耦合或通信连接，可以是电性或其它的形式。

[0265] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者也可以不是物理单元，即可以位于一个地方，或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0266] 另外，在本发明各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

[0267] 以上所述仅是本发明的优选实施方式，应当指出，对于本技术领域的普通技术人员来说，在不脱离本发明原理的前提下，还可以做出若干改进和润饰，这些改进和润饰也应视为本发明的保护范围。

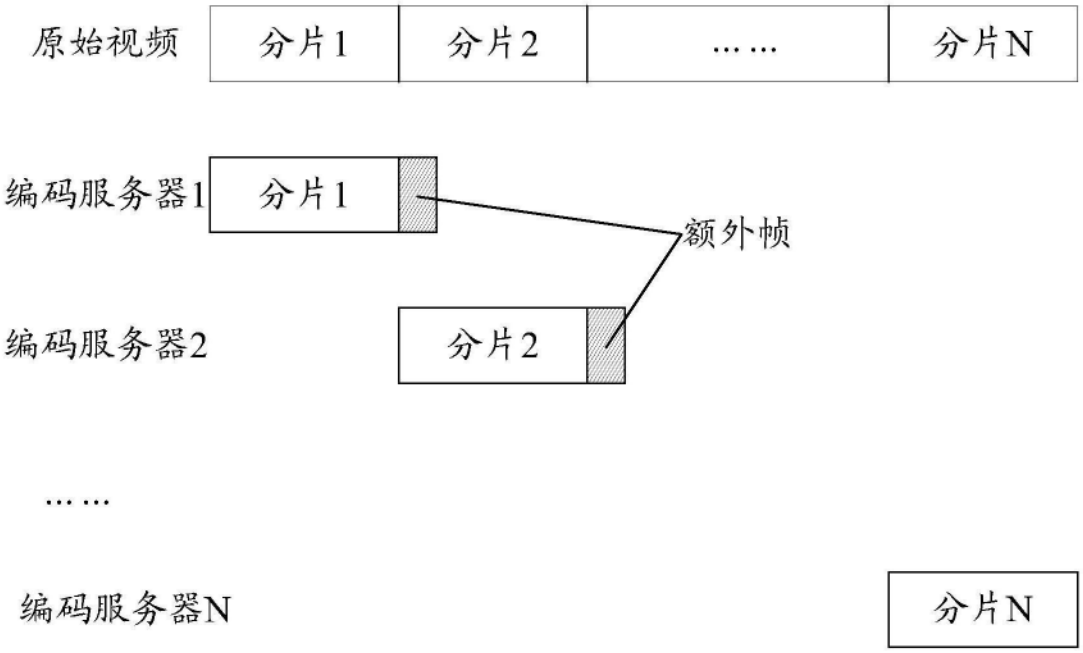

原始视频
分片1
分片2
……
分片N
编码服务器1
分片1
额外帧
编码服务器2
分片2
……
编码服务器N
分片N

图1

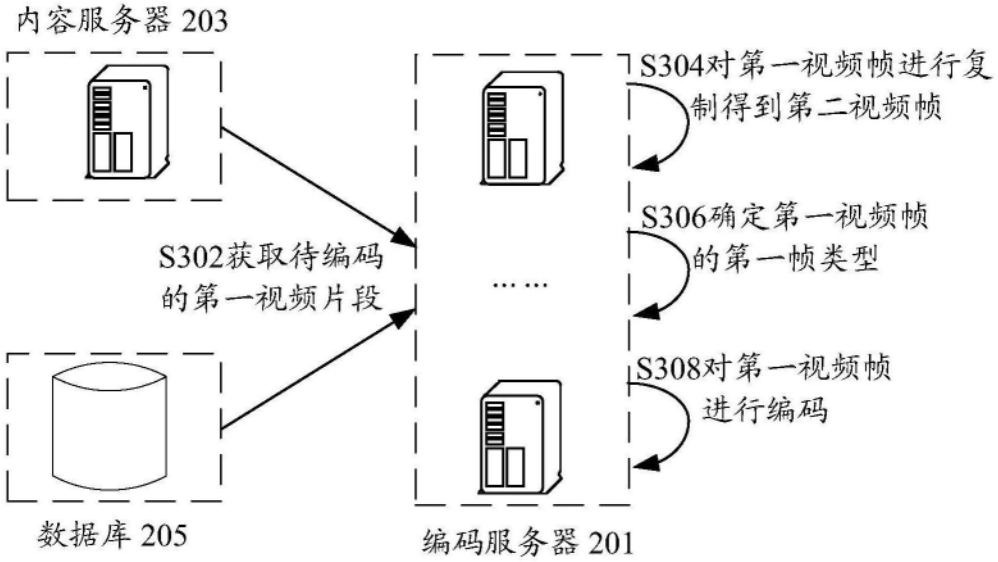

内容服务器 203
S304对第一视频帧进行复制得到第二视频帧
S306确定第一视频帧的第一帧类型
S302获取待编码的第一视频片段
……
S308对第一视频帧进行编码
数据库 205
编码服务器 201

图2

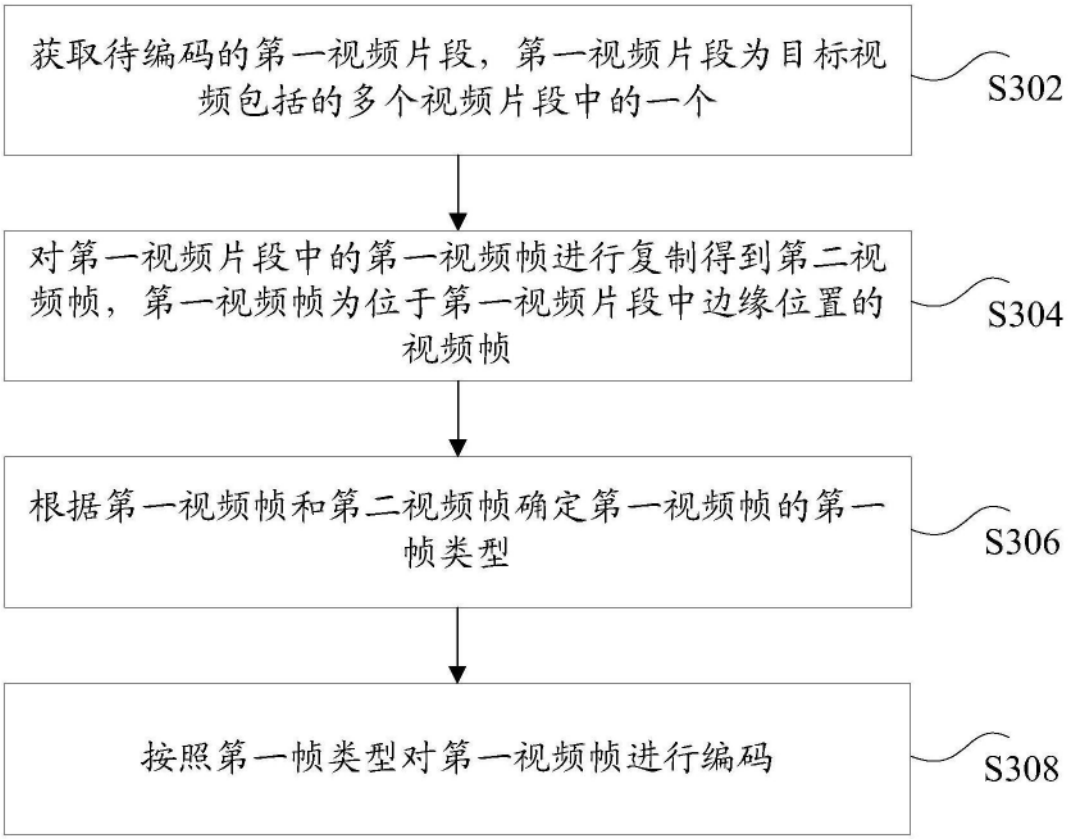

获取待编码的第一视频片段，第一视频片段为目标视频包括的多个视频片段中的一个
S302
对第一视频片段中的第一视频帧进行复制得到第二视频帧，第一视频帧为位于第一视频片段中边缘位置的视频帧
S304
根据第一视频帧和第二视频帧确定第一视频帧的第一帧类型
S306
按照第一帧类型对第一视频帧进行编码
S308

图3

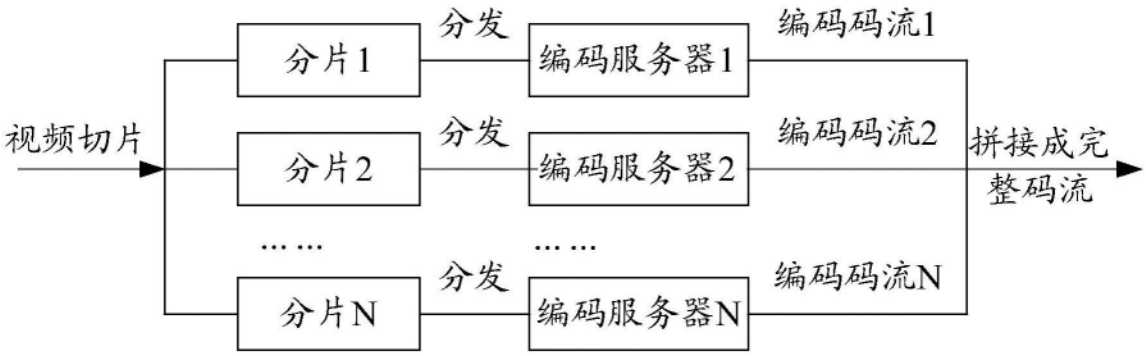

视频切片
分片1
分片2
……
分片N
分发
分发
分发
编码服务器1
编码服务器2
……
编码服务器N
编码码流1
编码码流2
编码码流N
拼接成完整码流

图4

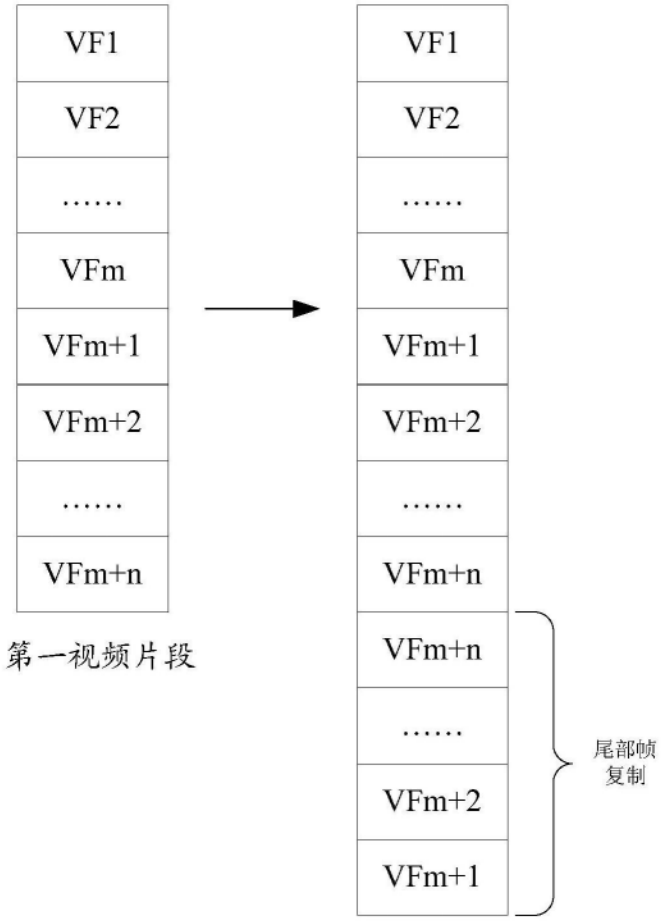

VF1
VF2
……
VFm
VFm+1
VFm+2
……
VFm+n
第一视频片段
VF1
VF2
……
VFm
VFm+1
VFm+2
……
VFm+n
VFm+n
……
VFm+2
VFm+1
尾部帧复制

图5

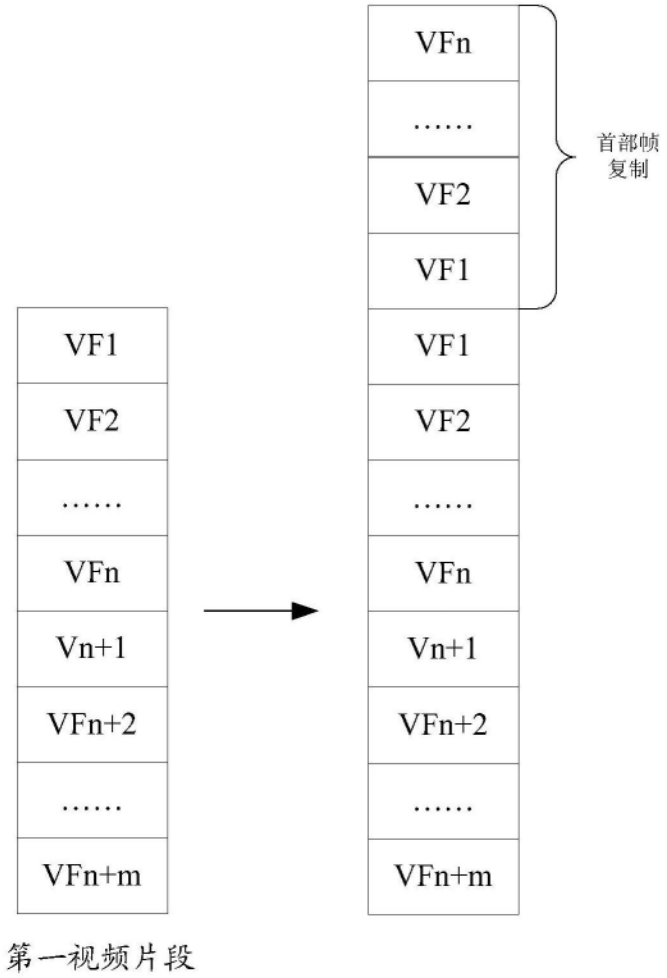

VFn
……
VF2
VF1
首部帧复制
VF1
VF2
……
VFn
Vn+1
VFn+2
……
VFn+m
VF1
VF2
……
VFn
Vn+1
VFn+2
……
VFn+m
第一视频片段

图6

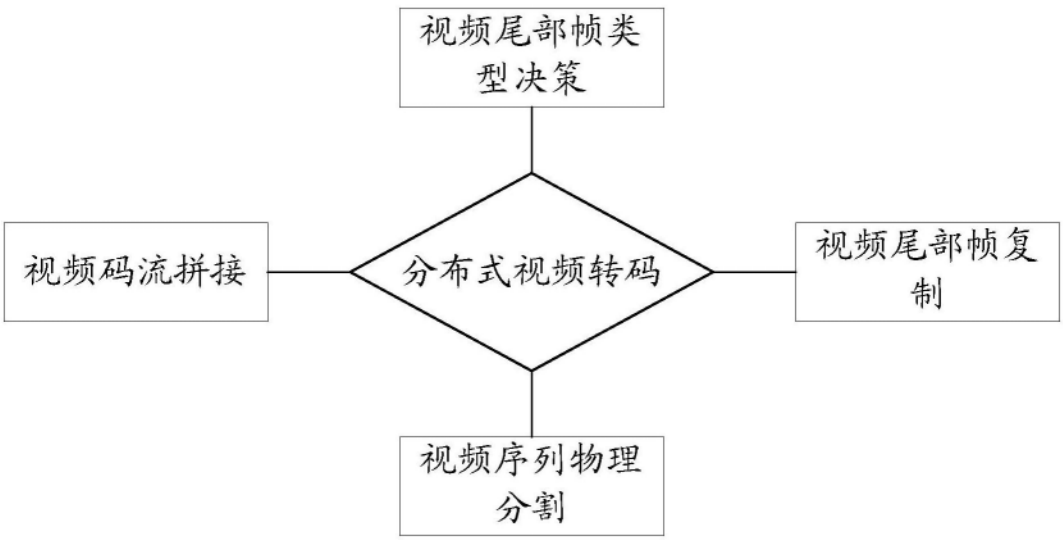

视频尾部帧类型决策
视频码流拼接
分布式视频转码
视频尾部帧复制
视频序列物理分割

图7

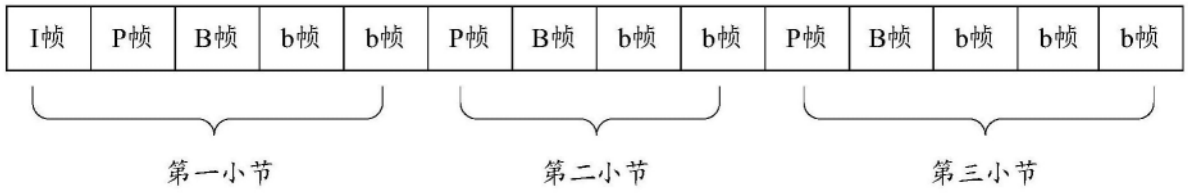

I帧
P帧
B帧
b帧
b帧
P帧
B帧
b帧
b帧
P帧
B帧
b帧
b帧
b帧
第一小节
第二小节
第三小节

图8

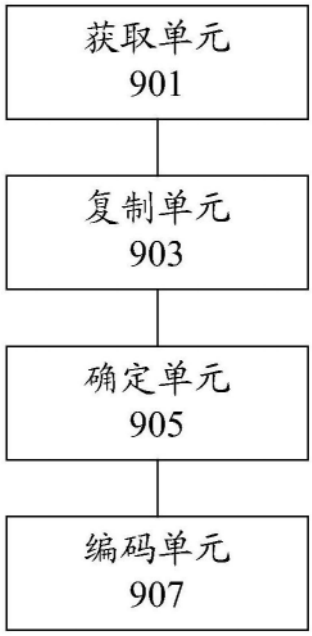

获取单元
901
复制单元
903
确定单元
905
编码单元
907

图9

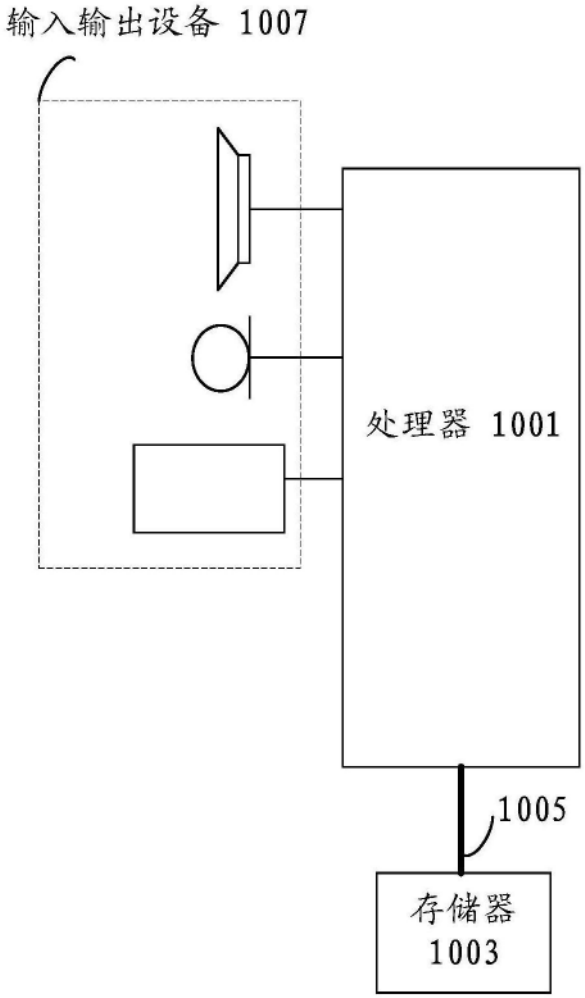


图10