



(12)发明专利申请

(10)申请公布号 CN 108353174 A

(43)申请公布日 2018.07.31

(21)申请号 201680061530.4

(22)申请日 2016.08.23

(30)优先权数据

14/925,761 2015.10.28 US

(85)PCT国际申请进入国家阶段日

2018.04.20

(86)PCT国际申请的申请数据

PCT/US2016/048244 2016.08.23

(87)PCT国际申请的公布数据

W02017/074539 EN 2017.05.04

(71)申请人 高通股份有限公司

地址 美国加利福尼亚州

(72)发明人 帕维尔·辛哈 马克·托多罗维奇

(74)专利代理机构 北京律盟知识产权代理有限公司 11287

代理人 杨林勳

(51)Int.Cl.

H04N 19/13(2006.01)

H04N 19/91(2006.01)

H04N 19/436(2006.01)

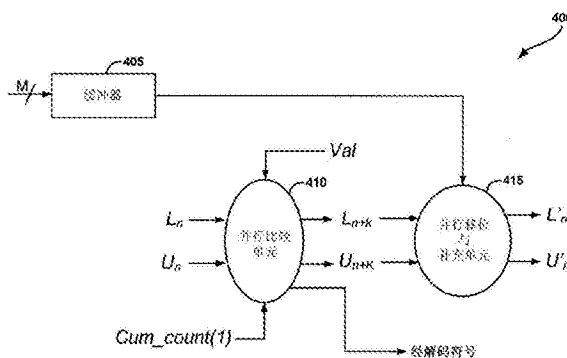
权利要求书2页 说明书21页 附图5页

(54)发明名称

并行算术译码技术

(57)摘要

本文中描述用于解码经压缩数据的系统和
方法。存储器可接收包括使用基于算术译码的译
码技术所编码的经编码符号字符串的数据流。所述
经编码符号字符串表示一系列一或多个经解码符
号。所述一或多个经解码符号中的每一个包括最
可能符号和最不可能符号中的一个。处理器可通
过基于所述经编码符号中的一或多个来执行多
个并行数学比较而确定所述系列一或多个经解
码符号中的连续最可能符号的数目。所述处理器
可进一步基于所述多个并行数学比较来解码所
述系列一或多个经解码符号中的符号群组。所述
符号群组包括所述数目的连续最可能符号。



1. 一种用于解码经压缩数据的方法,所述方法包括:

接收包括使用基于算术译码的译码技术编码的经编码符号字串的数据流,其中所述经编码符号字串表示一系列一或多个经解码符号,且其中所述一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个;

通过基于所述经编码符号中的一或多个来执行多个并行数学比较而确定所述系列一或多个经解码符号中的连续最可能符号的数目;和

基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组,其中所述符号群组包括所述数目的连续最可能符号。

2. 根据权利要求1所述的方法,其中所述解码所述数目的连续最可能符号是在单一时钟循环中发生。

3. 根据权利要求1所述的方法,其中所述符号群组进一步由在所述数目的连续最可能符号之后的一个最不可能符号组成。

4. 根据权利要求1所述的方法,其中在不使用位填充技术的情况下编码所述经编码符号字串。

5. 根据权利要求1所述的方法,其中基于算术译码的译码技术包括以下各种中的一种:算术译码和Q译码。

6. 根据权利要求1所述的方法,其中所述基于算术译码的译码技术为基于乘数的译码技术。

7. 根据权利要求1所述的方法,其中所述经解码符号群组指示定义图片序列的至少一部分的一或多个语法元素。

8. 一种用于解码经压缩数据的系统,其解码器包括:

存储器,其配置成接收包括使用基于算术译码的译码技术编码的经编码符号字串的数据流,其中所述经编码符号字串表示一系列一或多个经解码符号,且其中所述一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个;和

处理器,其配置成进行以下操作:

通过基于所述经编码符号中的一或多个来执行多个并行数学比较而确定所述系列一或多个经解码符号中的连续最可能符号的数目;和

基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组,其中所述符号群组包括所述数目的连续最可能符号。

9. 根据权利要求8所述的系统,其中所述解码所述数目的连续最可能符号是在单一时钟循环中发生。

10. 根据权利要求8所述的系统,其中所述符号群组进一步由在所述数目的连续最可能符号之后的一个最不可能符号组成。

11. 根据权利要求8所述的系统,其中在不使用位填充技术的情况下编码所述经编码符号字串。

12. 根据权利要求8所述的系统,其中基于算术译码的译码技术包括以下各种中的一种:算术译码和Q译码。

13. 根据权利要求8所述的系统,其中所述基于算术译码的译码技术为基于乘数的译码技术。

14. 根据权利要求8所述的系统,其中所述经解码符号群组指示定义图片序列的至少一部分的一或多个语法元素。

15. 一种非暂时性计算机可读媒体,其上存储有指令,所述指令在执行时致使用于解码数据的装置的处理器的处理器执行用于解码经压缩数据的方法,所述方法包括:

接收包括使用基于算术译码的译码技术编码的经编码符号字串的数据流,其中所述经编码符号字串表示一系列一或多个经解码符号,且其中所述一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个;

通过基于所述经编码符号中的一或多个来执行多个并行数学比较而确定所述系列一或多个经解码符号中的连续最可能符号的数目;和

基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组,其中所述符号群组包括所述数目的连续最可能符号。

16. 根据权利要求15所述的非暂时性计算机可读媒体,其中所述解码所述数目的连续最可能符号是在单一时钟循环中发生。

17. 根据权利要求15所述的非暂时性计算机可读媒体,其中所述符号群组进一步由在所述数目的连续最可能符号之后的一个最不可能符号组成。

18. 根据权利要求15所述的非暂时性计算机可读媒体,其中在不使用位填塞技术的情况下编码所述经编码符号字串。

19. 根据权利要求15所述的非暂时性计算机可读媒体,其中基于算术译码的译码技术包括以下各种中的一种:算术译码和Q译码。

20. 根据权利要求15所述的非暂时性计算机可读媒体,其中所述基于算术译码的译码技术为基于乘数的译码技术。

21. 一种用于解码经压缩数据的系统,其解码器包括:

用于接收包括使用基于算术译码的译码技术编码的经编码符号字串的数据流的装置,其中所述经编码符号字串表示一系列一或多个经解码符号,且其中所述一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个;

用于通过基于所述经编码符号中的一或多个来执行多个并行数学比较而确定所述系列一或多个经解码符号中的连续最可能符号的数目的装置;和

用于基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组的装置,其中所述符号群组包括所述数目的连续最可能符号。

22. 根据权利要求21所述的系统,其中所述解码所述数目的连续最可能符号是在单一时钟循环中发生。

23. 根据权利要求21所述的系统,其中所述符号群组进一步由在所述数目的连续最可能符号之后的一个最不可能符号组成。

24. 根据权利要求21所述的系统,其中在不使用位填塞技术的情况下编码所述经编码符号字串。

25. 根据权利要求21所述的系统,其中基于算术译码的译码技术包括以下各种中的一种:算术译码和Q译码。

26. 根据权利要求21所述的系统,其中所述基于算术译码的译码技术为基于乘数的译码技术。

并行算术译码技术

技术领域

[0001] 本发明大体上涉及译码,包含视频译码。更具体地说,本发明涉及用于通过使用并行处理技术来解码使用基于算术译码的译码技术所编码的数据(包含视频数据)的系统和方法。

背景技术

[0002] 数字视频能力可并入到广泛范围的装置中,所述装置包含数字电视、数字直播系统、无线广播系统、个人数字助理(PDA)、笔记本或台式电脑、平板电脑、电子书阅读器、数字摄像机、数字记录装置、数字媒体播放器、视频游戏装置、视频游戏主控台、蜂窝式或卫星无线电电话(所谓的“智能型电话”)、视频电传会议装置、视频流式传输装置及其类似者。数字视频装置实施视频译码技术,例如由MPEG-2、MPEG-4、ITU-T H.263、ITU-T H.264/MPEG-4(第10部分)、高级视频译码(AVC)、高效率视频译码(HEVC)标准和此类标准的延伸所定义的标准中所描述的技术。视频装置可通过实施此类视频译码技术来更高效地传输、接收、编码、解码和/或存储数字视频信息。

[0003] 视频译码技术包含空间(图片内)预测和/或时间(图片间)预测以缩减或去除为视频序列所固有的冗余。对于基于块的视频译码,可将视频切片(例如,视频帧或视频帧的部分)分割成视频块,对于一些技术,视频块也可被称作树型块、译码单元(CU)和/或译码节点。图片的经帧内译码(I)切片中的视频块是使用关于同一图片中的相邻块中的参考样本的空间预测予以编码。图片的经帧间译码(P或B)切片中的视频块可使用关于同一图片中的相邻块中的参考样本的空间预测,或关于其它参考图片中的参考样本的时间预测。图片可被称作帧,且参考图片可被称作参考帧。

[0004] 空间或时间预测得到用于待译码块的预测性块。残余数据表示原始待译码块与预测性块之间的像素差。根据指向形成预测性块的参考样本块的运动向量和指示经译码块与预测性块之间的差的残余数据来编码经帧间译码块。根据帧内译码模式和残余数据来编码经帧内译码块。出于进一步压缩起见,可将残余数据从像素域变换到变换域,从而得到残余变换系数,其接着可被量化。可扫描最初以二维阵列而布置的经量化变换系数以便产生变换系数的一维向量,且可应用熵译码以达成甚至更多的压缩。

[0005] 除了数字视频能力以外或替代数字视频能力,装置还可能能够以空中方式或通过有线链路来传输和接收任何类型的数字内容。可在传输之前压缩此类数字内容。

发明内容

[0006] 本发明的教示的一个方面涉及一种用于解码经压缩数据的方法。所述方法包括接收包括使用基于算术译码的译码技术所编码的经编码符号字串的数据流。所述经编码符号字串表示一系列一或多个经解码符号。所述一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个。所述方法进一步包括通过基于所述经编码符号中的一或多个来执行多个并行数学比较而确定所述系列一或多个经解码符号中的连续最可能符号的数

目。所述方法进一步包括基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组。所述符号群组包括所述数目个连续最可能符号。

[0007] 本发明的教示的另一方面涉及一种用于解码经压缩数据的系统。所述系统包括存储器,其配置成接收包括使用基于算术译码的译码技术所编码的经编码符号字串的数据流。所述经编码符号字串表示一系列一或多个经解码符号。所述一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个。所述系统进一步包括处理器,其配置成通过基于所述经编码符号中的一或多个来执行多个并行数学比较而确定所述系列一或多个经解码符号中的连续最可能符号的数目。所述处理器经进一步配置成基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组。所述符号群组包括所述数目个连续最可能符号。

[0008] 本发明的教示的另一方面涉及一种非暂时性计算机可读媒体,其上存储有指令,所述指令在执行时致使用于解码数据的装置的处理器执行用于解码经压缩数据的方法。所述方法包括接收包括使用基于算术译码的译码技术所编码的经编码符号字串的数据流。所述经编码符号字串表示一系列一或多个经解码符号。所述一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个。所述方法进一步包括通过基于所述经编码符号中的一或多个来执行多个并行数学比较而确定所述系列一或多个经解码符号中的连续最可能符号的数目。所述方法进一步包括基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组。所述符号群组包括所述数目个连续最可能符号。

[0009] 本发明的教示的另一方面涉及一种用于解码经压缩数据的系统。所述系统包括用于接收包括使用基于算术译码的译码技术所编码的经编码符号字串的数据流的装置。所述经编码符号字串表示一系列一或多个经解码符号。所述一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个。所述系统进一步包括用于通过基于所述经编码符号中的一或多个来执行多个并行数学比较而确定所述系列一或多个经解码符号中的连续最可能符号的数目的装置。所述系统进一步包括用于基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组的装置。所述符号群组包括所述数目个连续最可能符号。

附图说明

[0010] 图1为说明可实施本发明的技术的实例视频编码和解码系统的框图。

[0011] 图2为说明可实施本发明的技术的视频编码器的实例的框图。

[0012] 图3为说明可实施本发明的技术的视频解码器的实例的框图。

[0013] 图4为说明可实施本发明的技术的解码器的实例的框图。

[0014] 图5为用于解码经压缩数据的过程的流程图的说明确。

具体实施方式

[0015] 本发明的技术涉及用于通过使用并行处理技术以缩减解码潜时来解码使用基于算术译码的译码技术所编码的数据(包含视频数据)的系统和方法。具体地说,本文中所描述的系统和方法可关于同时处理经译码数据块(例如,在单一时钟循环中的若干经解码符号)的算术解码技术,其可被称作“块算术译码”。此类技术可有利地允许经译码数据的解码

速率大约与经译码数据的经编码(例如,经压缩)数据符号速率成比例,这意谓解码经编码数据符号字符串所需要的循环的数目大约等于经编码数据符号的数目。此类经缩减潜时可导致解码计算较快,从而潜在地节省时间和能量。所述技术可应用于现有视频编解码器中的任一种(例如高效视频译码(HEVC)),或可为用于任何未来视频译码标准的高效译码工具。此外,此类技术可经调适为无乘数译码技术(例如,Q译码器、MQ译码器等等)和基于乘数的译码技术。这允许使经缩减潜时解码器不对压缩比产生任何负面影响(例如,基于乘数的译码技术)或能够用压缩比交换计算复杂度(例如,基于无乘数的译码技术)的灵活性。此外,此类技术并不需要位填塞(bit-stuffing)以便执行并行处理。尽管关于视频译码来描述所述技术,但所属领域的技术人员应理解,此类技术可用于译码其它类型的数据。

[0016] 举例来说,通常可通过使用进一步压缩以增强低功率要求来改进数字数据的串列传输。根据本文中所论述的技术,可针对同步和/或非同步链路来改进此类串列传输。具体地说,本文中所描述的压缩技术可不仅缩减通过同步链路而传输所需要的位的总数目,而且缩减在装置之间传输所需要的同步时钟的数目。更一般化地说,本文中所论述的压缩技术可(例如)通过允许尽可能快速地使时钟关闭以节省电力来缩减压缩和解压缩数字数据所需要的时钟循环的数目,借此需要较低功率来实施。

[0017] 视频译码装置实施视频压缩技术以高效地编码和解码视频数据。视频压缩技术可包含应用空间预测(例如,帧内预测(intra-frame prediction/intra-prediction))、时间预测(例如,帧间预测(inter-frame prediction/inter-prediction))和/或其它预测技术以缩减或去除为视频序列所固有的冗余。视频编码器通常将原始视频序列中的每一图片分割成被称作视频块或译码单元的矩形区域。可使用特定预测模式来编码这些视频块。

[0018] 可以一或多种方式将这些视频块划分成一或多个较小视频块(例如,预测块(PB)、变换块(TB))群组,每一群组个别地代表整个视频块,如本文中进一步所描述。因此,除非另外指定,否则大体上对块的参考可指此类视频块(例如,译码树型块(CTB)、译码块等等)、预测块、变换块或其它适当块或子块,如将由所属领域的一般技术人员所理解。此外,这些块中的每一个也可互换地在本文中被称作“单元”(例如,译码树型单元(CTU)、译码单元、预测单元(PU)、变换单元(TU)等等)。所属领域的一般技术人员将认识到,单元可指示在位流中所编码的译码逻辑单元,而块可指示为过程的目标的视频帧缓冲器的部分。

[0019] 对于帧间预测模式,视频编码器通常在另一时间位置中的帧(被称作参考帧)中搜索相似于正被编码块的块。视频编码器可将搜索限定于从待编码块的某一空间位移。可使用包含水平位移分量和竖直位移分量的二维(2D)运动向量来定位最优匹配。对于帧内预测模式,视频编码器可基于来从同一图片内的经先前编码相邻块的数据而使用空间预测技术来形成经预测块。

[0020] 视频编码器可确定预测误差,即,正被编码块与经预测块中的像素值之间的差(也被称作残余)。视频编码器还可将例如离散余弦变换(DCT)的变换应用于预测误差以产生变换系数。在变换之后,视频编码器可量化变换系数。可使用语法元素来表示经量化变换系数和运动向量,且经量化变换系数和运动向量连同控制信息一起形成视频序列的经译码表示。在一些情况下,视频编码器可熵译码语法元素,借此进一步缩减其表示所需要的位的数目。

[0021] 举例来说,算术译码为可由视频编码器使用以熵译码语法元素和/或控制信息的

熵译码形式。具体地说,在熵译码中,运用较少位来编码和存储频繁使用的字符,而并非如运用较多位来编码和存储频繁使用的字符。经解码视频序列的表示(例如,语法元素和/或控制信息)可被表示为符号(例如,位)系列或字串。所述符号可具有不同值,包含最可能符号和最不可能符号。具体地说,一个符号相较于另一符号(即,最不可能符号)可更加有可能(即,最可能符号)位于经解码字串中。举例来说,最可能符号可为0,而最不可能符号可为1。视频编码器可配置成编码符号字串以产生表示经解码符号字串的经验编码符号字串。视频解码器可配置成解码经验编码符号以检索原始(即,经解码)符号字串。

[0022] 下文以伪代码展示使用区间按比例调整和整数算术的算术编码过程的一个实例。

[0023] 在一个实例中,待编码字串的长度为200个位(例如,符号),因此,变量Total_Count和Cum_count(x)表示长度为8个位(例如,用以能够表示高达200的值的位的最少数目)的寄存器。所属领域的技术人员应理解,伪代码也可调适用于其它适当寄存器大小和字串长度。此外,在所述实例中,L和U可表示长度为 $2 + \lceil \log_2(\text{Total_Count}) \rceil$ 的寄存器,在这个实例中,所述长度可为10个位。Scale3可为长度为 $\lceil \log_2(10-1) \rceil$ 的寄存器,在这个实例中,所述长度可为4个位。此外,可如下初始化所述寄存器: $L=0$; $\text{Scale3}=0$;将U设定为全1(即,在此实例中为1023); $\text{Total_Count}=200$ (即,其表示待编码位的数目); $\text{Cum_count}(0)=0$; $\text{Cum_count}(1)$ = 待编码字串中的最可能符号(例如,0)的总计数;且 $\text{Cum_count}(2)=200$ (即,其表示待编码位的数目。可根据编码伪代码A来如下执行算术编码:

[0024]

While (symbols to encode) {

/* Start Section A*/

Fetch next symbol to encode = 'x' // 'x' = {0,1}

//Cum_count(2) = Total_Count - Cum_count(0)

$$L \leftarrow L + \left\lfloor \frac{(U-L+1) * \text{Cum_count}(x)}{\text{Total_Count}} \right\rfloor$$

$$U \leftarrow L + \left\lfloor \frac{(U-L+1) * \text{Cum_count}(x+1)}{\text{Total_Count}} \right\rfloor - 1$$

/*End Section A*/

/*Start Section B*/

While (MSB (most significant bit) of U and L are equal OR the current interval

[L - U] is bounded by the interval $\left[\frac{1}{4} - \frac{3}{4} \right]$ of 1023 (i.e., the max value of U)) {

if (MSB of U and L are both equal to 'b') {

set 'b' as the next encoded symbol // 'b' = {0, 1}

Shift L to the left by 1 bit and shift 0 into LSB (least significant bit)

Shift U to the left by 1 bit and shift 1 into LSB

if (Scale3 > 0) {set the complement of 'b' Scale3 times as the next encoded

```

        symbol(s);
        setScale3 = 0 }
    }
    if (current interval  $[L - U]$  is bounded by the interval  $[\frac{1}{4} - \frac{3}{4}]$  of 1023) {
        shiftL to the left by 1 bit and shift 0 into LSB
        shiftU into left by 1 bit and shift 1 into LSB
[0025]    complement (new) MSB of L and U
        incrementScale3
    }
}
/*End Section B*/
}

```

[0026] 视频解码器可使用上文所论述的语法元素和控制信息来构建用于解码当前帧的预测性数据(例如,预测性块)。举例来说,视频解码器可将经预测块与经压缩预测误差相加。视频解码器可通过使用经量化系数来加权变换基底函数而确定经压缩预测误差。经重新构建帧与原始帧之间的差被称为重新构建误差。举例来说,视频解码器可解码从视频编码器接收的使用基于算术译码的译码技术(例如基于伪代码A)所编码的经编码符号字串以检索语法元素和/或控制信息。

[0027] 下文以伪代码B展示使用区间按比例调整和整数算术的算术解码过程的一个实例。

[0028] 在一个实例中,经编码字串的长度为200个位(例如,符号),因此,变量Total_Count、Cum_count(x)和Count_0表示长度为8个位(即,用以能够表示高达200的值)的寄存器。可从编码器接收经编码的经解码字串的长度的值和经编码字串中的最可能符号的数目的计数,例如,作为控制信息。所属领域的技术人员应理解,伪代码还可经调适用于其它适当寄存器大小和字串长度。此外,在所述实例中,L、U和T可表示长度为 $2 + \lceil \log_2(\text{Total_Count}) \rceil$ 的寄存器,在这个实例中,所述长度可为10个位。此外,可如下初始化所述寄存器:L=0;将U设定为全1(即,在这个实例中为1023);Total_Count=200(即,其表示待编码位的数目);将经编码字串之前 $2 + \lceil \log_2(\text{Total_Count}) \rceil$ 个位读取到T中(即,T表示输入经编码字串);Cum_count(0)=0;Cum_count(1)=Count_0;Cum_count(2)=200(即,其表示待编码位的数目);且Count_0=经编码字串中的最可能符号的总计数。可根据解码伪代码B来如下执行算术解码:

```

    While (allsymbols not decoded) {
[0029]    /*Start Section A/

```


[0030]

$$//Cum_count(2) = Total_Count - Cum_count(0)$$

$$Val = (T - L + 1) * Total_Count - 1$$

$$\text{if}(Val < Count_0 * (U - L + 1)) \{ \text{decode symbol 'x' as '0'} \}$$

$$\text{else} \{ \text{decode symbol 'x' as '1'} \}$$

$$L \leftarrow L + \left\lfloor \frac{(U-L+1) * Cum_count(x)}{Total_Count} \right\rfloor$$

$$U \leftarrow L + \left\lfloor \frac{(U-L+1) * Cum_count(x+1)}{Total_Count} \right\rfloor - 1$$

/*End Section A*/

/*Start Section B*/

While (MSB (most significant bit) of U and L are equal OR the current interval
$$[L - U] \text{ is bounded by the interval } \left[\frac{1}{4} - \frac{3}{4} \right] \text{ of } 1023 \text{ (i.e., the max value of } U)) \{$$

$$\text{if (MSB of } U \text{ and } L \text{ are both equal to 'b') } \{ //b' = \{ 0, 1 \}$$
Shift L to the left by 1 bit and shift 0 into LSB (least significant bit)Shift U to the left by 1 bit and shift 1 into LSBShift T to the left by 1 bit and read the next encoded bit of the string into

LSB

}

$$\text{if (current interval } [L - U] \text{ is bounded by the interval } \left[\frac{1}{4} - \frac{3}{4} \right] \text{ of } 1023) \{$$
Shift L to the left by 1 bit and shift 0 into LSBShift U into left by 1 bit and shift 1 into LSBShift T to the left by 1 bit and read the next encoded bit of the string into

LSB

Complement (new) MSB of L , U , and T

}

}

/*End Section B*/

}

[0031] 在将解码伪代码B的第一while循环(在第二while循环之前)的第一部分称作Section A且将嵌套于第一while循环中的第二while循环称作Section B的情况下,伪代码B表示解码算法。在这个算法中,每当执行Section A时,就解码单一输出符号且接着执行Section B。在Section B中,while循环可执行一或多次,之后返回到Section A。如可从伪代码B所见,Section A执行的次数等于经解码数据字串的长度(即,符号的数目)(其长于从编码器接收的经编码数据字串的长度)。因此,根据解码算法的经编码数据字串的解码速率高于经编码数据字串的速率。

[0032] 本文中所描述的技术注意到,如果不满足Section B的条件且不需要将新符号读取到T中以供解码,那么仅执行Section A以用于解码下一符号。仅在U与L相等或含于区间 $\left[0 - \frac{1}{2}\right]$ 、 $\left[\frac{1}{2} - 1\right]$ 或 $\left[\frac{1}{4} - \frac{3}{4}\right]$ 中的情况下才触发Section B。因此,本文中描述一种用于解码使用算术译码技术所编码的经编码字串的技术,所述技术使用并行处理技术以在触发算法的Section B之前解码尽可能多的符号。可大约以经编码符号速率(例如,每经编码符号一个时钟循环)解码这些符号。下文所描述的实例假定最可能符号为‘0’且因此经设计以同时解码若干‘0’。所属领域的技术人员应注意,相似技术可用以使用并行处理技术来解码另一值的若干最可能符号。

[0033] 在Section A中,L的值不会改变,只要经解码符号为‘0’即可。因此,可如下非递归地解码U的值:

$$[0034] \quad U_{n+1} = L + \left\lfloor \frac{(U_n - L + 1) * Cum_count(x + 1)}{Total_Count} \right\rfloor - 1$$

[0035] 假设, $P = \frac{Cum_count(1)}{Total_Count}$ // ‘0’的概率

[0036] 假定针对编码器算法和解码器算法两种去除底限算子:

$$[0037] \quad U_1 = (L-1) + \lfloor P (U_0 - L + 1) \rfloor$$

$$U_2 = (L-1) + \lfloor P ((L-1) + \lfloor P (U_0 - L + 1) \rfloor) - L + 1 \rfloor$$

$$[0038] \quad = (L-1) + \lfloor PL - P + P \lfloor P (U_0 - L + 1) \rfloor - PL + P \rfloor$$

$$= (L-1) + \lfloor P \lfloor P (U_0 - L + 1) \rfloor \rfloor$$

[0039]

$$U_N = (L-1) + \lfloor P \dots \lfloor P \lfloor P (U_0 - L + 1) \rfloor \rfloor \rfloor \rfloor \cong (L-1) + \text{floor}(P^N (U_0 - L + 1)) \quad (\text{方程式 1})$$

[0040] 因此,可运用以上方程式1来近似 U_N ,前提是同一方程式1用于编码器侧和解码器侧两处。因此,剩下要确定的仅有变量为‘0’的数目以解码或确定 U_N 的方程式1中的‘N’的值。

[0041] 假定L不变且正被解码的输入符号为0,那么Val的值不会改变。因此,用以确定待解码的0的数目以用于Section A的每一块解码的一组并行比较如下:

[0042] 方程式2.1: Val (1) < Count_0* (U₀-L+1) → 解码‘0’, 否则解码‘1’

[0043] 方程式2.2: Val (2) < Count_0* (U₁-L+1) → 解码‘00’, 否则解码‘01’

[0044] ...

[0045] 方程式2.N: Val (N) < Count_0* (U_{N-1}-L+1) → 解码‘00...0’, 否则解码‘00...1’

[0046] 可并行地评估以上方程式2.1到2.N中的每一个(例如,并行解码块),且第一无效条件(即,用于N的最低值的无效方程式)给出N的值。举例来说,如果方程式2.3为第一无效方程式,那么N=3。因此,取决于可并行地执行的方程式2.N的数目(例如,其可取决于所使用的解码硬件,所述解码硬件可为经设计和/或编程以执行所描述技术的任何合适形式的特殊化处理硬件),可在单一时钟循环中解码若干符号。具体地说,可在单一时钟循环中解码成一系列的一样多的连续最可能符号,随后可在单一时钟循环中解码单一最不可能符号。

替代地,如果可被解码的成一系列的连续最可能符号的数目受到硬件约束限制,且未达到最不可能符号(即,所述比较中的任一个皆有效),那么解码彼最大数目个连续最可能符号。因此,基于所描述技术,Section A的块反复的数目(例如,进行并行比较的次数)取决于经连续编码的最可能符号的数目,这意谓解码符号且执行Section A的速率可大约与传入经编码符号的速率成比例(例如,每经编码符号需要一个时钟循环以供解码)。

[0047] 此外,还可并行地评估伪代码B中所描述的解码器算法的Section B。具体地说,在Section B中执行的计算限于值的位移、补充和比较。因此,可实施组合逻辑(例如,经设计和/或编程以执行所描述技术的任何合适形式的特殊化处理硬件)以执行Section B的计算作为并行块(即,经并行计算逻辑)。

[0048] 举例来说,如果使用两个并行块以计算Section B,那么用以计算Section B的第一块将由组合逻辑组成以实施如所展示的Section B的算法以计算Section B的while循环的第一回合。用以计算Section B的第二块将包含预看逻辑(look ahead logic)且针对Section B的while循环的第二回合计算Section B的两个可能分支:1)根据while循环的第一回合而对U和L进行1个位的左移位;和2)根据while循环的第一回合而对U和L进行1个位的左移位且还补充U、L和T的MSB。用作最终结果的用以计算Section B的第二块的结果将基于用以计算Section B的第一块的结果(例如,相似于进位链逻辑)。因此,可在单一时钟循环中对Section B执行两个遍次的计算。

[0049] 相似地,可使用任何数目个额外并行块以计算Section B,以增加可在单一时钟循环中计算的Section B的遍次的数目。额外并行块将配置成在while循环的那个回合(即,第一、第二、第三、第四等等)处执行且计算所有可能分支。举例来说,用以计算Section B的第三块将需要计算对应于while循环的第三回合的4个可能分支(即,其中根据while循环的两个可能第二回合中的每一个而对U和L进行1个位的左移位;且其中根据while循环的两个可能第二回合中的每一个而对U和L进行1个位的左移位且还补充U、L和T的MSB)。将基于先前并行块的结果来选择所计算的最终并行块的结果(即,最终块是基于先前块,其是基于先前块,等等)。可看出,为了增加用以计算Section B的并行块的数目,对于每一连续块,逻辑呈指数地增加(即,假定第一块的逻辑要求被视为1,那么每一块的逻辑要求为 2^x ,其中x为块数目1、2、3、4等等)。

[0050] 此外,通过使用并行块以用于处理Section B,Section B的反复的总数目可大约等于正被解码的经编码符号的数目。

[0051] 应注意,可基于用以寻找硬件成本与性能之间的适当平衡的测试来选择用以针对Section A执行并行比较和/或对Section B中的块执行并行处理的逻辑的量(例如,硬件中的并行组件的量)。举例来说,逻辑的量可取决于用于实施的技术选择(例如,DSP、FPGA等等)和正被译码的数据类型而变化。

[0052] 在以上实例中,寄存器U的位宽度为10个位。因此,在任何时间所需要的Section B的反复的数目将小于10。因此,在此类实施方案中用于Section B的并行块的数目可限于不多于10,且更具体地说,可被设定为7,这是因为在使用高于7个并行块的情况下可不存在显著性能益处。

[0053] 基于针对解码所描述的块并行处理,Section B的反复的总数目等于待解码的经编码符号的总数目。因此,类似于Section A,执行Section B的速率可大约与传入经编码符

号的速率成比例(例如,每经编码符号需要一个时钟循环以供执行)。因此,用于所组合的Section A和Section B的经编码符号的解码速率大约与经编码符号速率(例如,正被解码的经编码符号的数目)成正比。在一些实施例中,可在单一时钟循环中一起计算Section A和Section B两个的反复的计算。

[0054] 针对基于乘数的译码技术描述用于通过使用上文所描述的并行处理技术来解码使用基于算术译码的译码技术所编码的数据(包含视频数据)的技术。然而,如本文中所论述,相似技术可应用于基于无倍数的译码技术。

[0055] 在一个实例中,本文中用于解码数据的技术可用于使用基于Q译码器的无乘数编码方案所编码的数据。具体地说,可如下计算上文所论述的伪代码B的解码算法的Section A。

[0056] 如上文所论述,在Section A中,L的值不会改变,只要经解码符号为‘0’即可。因此,可如下非递归地解码U的值:

$$[0057] \quad U_{n+1} = L + \left\lfloor \frac{(U_n - L + 1) * Cum_count(x+1)}{Total_Count} \right\rfloor - 1 \quad (\text{方程式 3})$$

[0058] 假设, $P = \frac{Cum_count(1)}{Total_Count}$ // ‘0’的概率

[0059] 因此,

[0060] $Q = 1 - P$

[0061] 因为将U和L按比例调整到用于整数表示的 2^k ,所以可重写方程式3,其中将U和L按比例调整到1,且接着在将所述方程式的整数形式转换为有理形式的情况下,可导出以下方程式4:

$$[0062] \quad u_{n+1} = 1 + P * (u_n - 1) \quad (\text{方程式 4})$$

[0063] 其中u和1指示U和L的未按比例调整值且被界限于0与1之间。

[0064] 在应用相似近似作为Q译码器的情况下,以下方程式保持成立:

$$[0065] \quad \text{如果 } 0 < K < 1; \text{ 则 } K * Q \approx Q \text{ 且 } K * P = K * (1 - Q) \approx K - Q \quad (\text{方程式 5})$$

[0066] 在将方程式5的以上条件应用于 u_{n+1} 的情况下,可导出以下方程式:

$$[0067] \quad u_{n+1} = 1 + P * (u_n - 1) \approx 1 + (u_n - 1) - Q = u_n - Q$$

$$[0068] \quad u_1 = u_0 - Q; \quad u_2 = u_1 - Q = u_0 - 2 * Q; \quad u_3 = u_2 - Q = u_0 - 3 * Q;$$

$$[0069] \quad \text{因此, } u_{n+1} = u_0 - (n+1) * Q \quad (\text{方程式 6})$$

[0070] 如果Q对于如在调适性Q译码器中的每一反复是不同的,那么方程式6变为:

$$[0071] \quad u_{n+1} = u_0 - \sum_{i=0}^n Q_i \quad (\text{方程式 7})$$

[0072] 将需要如下由Total_Count(例如,通过执行移位运算)按比例调整这些方程式6和7的整数等值:

$$[0073] \quad U_{n+1} = U_0 - [(n+1) * Q * Total_Count] \quad (\text{方程式 8})$$

$$[0074] \quad U_{n+1} = U_0 - \left\lfloor \sum_{i=0}^n Q_i * Total_Count \right\rfloor \quad (\text{对于变化的 } Q) \quad (\text{方程式 9})$$

$$[0075] \quad L_1 = U_0 - \left\lfloor (Q_0 * Total_Count) \right\rfloor \quad (\text{方程式 10})$$

[0076] 如运用基于乘数的途径所论述, U_n 的近似可仅在其应用于编码器和解码器两处以确保在编码器和解码器两个中的L和U的匹配跟踪的情况下才有效。

[0077] 假定L不变且正被解码的输入符号为0,那么Val的值不会改变。

[0078] 从Section A,进行以下比较:

[0079] $Val < Count_0 * (U-L+1)$;

[0080] 此外,

[0081] $Val = (T-L+1) * Total_Count - 1$

[0082] 因此:

[0083] $(T-L+1) * Total_Count - 1 < Count_0 * (U-L+1)$;

[0084] $(T-L+1) \leq P * (U-L+1)$;其中 $P = Count_0 / Total_Count$ 且 $Q = 1 - P$

[0085] $(T-L) < P * (U-L) + P$;其中 $0 \leq P \leq 1$ 且 $0 \leq Q \leq 1$

[0086] $(T-L+1) < P * (U-L)$;

[0087] $Val_{new} < P * (U-L)$;其中 $Val_{new} = (T-L+1)$

[0088] 另外,通过应用Q译码器样式近似,且因为U和L为用于整数表示的u和l的经按比例调整版本,所以下式保持成立:

[0089] $P * (U-L) = P * (u-l) * Total_Count = (u-l-Q) * Total_Count$

[0090] 因此,如下进行用以确定待解码的0的数目以用于Section A的每一块解码的一组并行比较:

[0091] 方程式11.1: $Val_{new}(0) < (u(0) - l - Q) * Total_Count \rightarrow$ 解码 '0', 否则解码 '1'

[0092] 方程式11.2: $Val_{new}(1) < (u(1) - l - Q) = (u(0) - l - 2 * Q) * Total_Count \rightarrow$ 解码 '00', 否则解码 '01'

[0093] ...

[0094] 方程式11.n: $Val_{new}(n) < (u(2) - l - Q) = (u(0) - l - n * Q) * Total_Count \rightarrow$ 解码 '00...0', 否则解码 '00...1'

[0095] 对于变量Q,所述比较变为:

[0096] 方程式11.n: $Val_{n-1} < Count_0 * (U_{n-1} - L - \sum_{i=0}^n Q_i) * Total_Count \rightarrow$ 解码 '00...0', 否则解码 '00...1'

[0097] 可并行地评估以上方程式11.1到11.n中的每一个(例如,并行解码块),且第一无效条件(即,用于n的最低值的无效方程式)给出n的值。举例来说,如果方程式11.3为第一无效方程式,那么 $n=3$ 。因此,取决于可并行地执行的方程式11.n的数目(例如,其可取决于所使用的解码硬件,所述解码硬件可为经设计和/或编程以执行所描述技术的任何合适形式的特殊化处理硬件),可在单一时钟循环中解码若干符号。具体地说,可在单一时钟循环中解码成一系列的一样多的连续最可能符号,随后可在单一时钟循环中解码单一最不可能符号。替代地,如果可被解码的成一系列的连续最可能符号的数目受到硬件约束限制,且未达到最不可能符号(即,所述比较中的任一个皆有效),那么解码彼最大数目个连续最可能符号。因此,基于所描述技术,Section A的块反复的数目(例如,进行并行比较的次数)取决于经连续编码的最可能符号的数目,这意谓解码符号且执行Section A的速率可大约与传入经编码符号的速率成比例(例如,每经编码符号需要一个时钟循环以供解码)。

[0098] 在另一实例中,本文中用于解码数据的技术可用于使用基于调适性Q译码器的无乘数编码方案所编码的数据。举例来说,JPEG Q译码器使用经预计算Q值的数据表,每当进行区间的重新正规化(例如,解码算法的Section B的执行)时就使用所述经预计算Q值。由JPEG使用的数据表可具有112个项目。

[0099] 根据本文中所描述的技术,在数据表中,Q可被初始化为0.5,且每当进行重新正规化(即,Section B的执行)时就被修改,而非针对待解码的每一新输入经编码符号被修改。在从经编码符号解码最可能符号之后,数据表通过缩减Q的值来改变下一经解码符号为最可能符号的可能性。在从经编码符号解码最不可能符号之后,数据表通过增加Q的值来改变下一经解码符号为最可能符号的可能性。

[0100] 因为Q的值仅在进入解码算法的Section B时才被更新,所以这暗示:在仅执行Section A时,Q的值不会改变。因此,以下方程式12和13可以整数形式用于解码基于调适性Q译码器的经编码符号(其中U和L是由Total_Count按比例调整):

$$[0101] \quad U_{n+1} = U_0 - [(n+1) * Q * Total_Count] \quad (\text{方程式 12})$$

$$[0102] \quad U_{n+1} = U_0 - \left\lfloor \sum_{i=0}^n Q_i * Total_Count \right\rfloor \quad (\text{对于变化的 } Q) \quad (\text{方程式 13})$$

[0103] 这些方程式相同于所论述的用于解码基于Q译码器的经编码符号的方程式8和9,且因此使用方程式11.n的相同比较可用以执行Section A以解码经编码符号。

[0104] 基于以下分析,用于关于基于调适性Q译码器和Q译码器的译码技术所描述的实例解码技术的Q译码器近似是有效的。

[0105] 假设q为经解码数据字符串中的最不可能符号(例如,1)的真概率,且假设Q为用于来自使用并行处理技术的算术解码的无乘数途径的实例的近似中的概率。因此,无乘数方程式6的未按比例调整版本如下:

$$[0106] \quad u_n = u_0 - n * Q \quad (\text{方程式 14})$$

[0107] 假设 A_i 为在进入Section A时的区间且假设 A_{ii} 为在退出Section A时的区间,那么如下基于方程式14来计算区间的改变:

$$[0108] \quad \Delta A = A_i - A_{ii} = (u_0 - 1) - (u_n - 1) = u_0 - u_n = n * Q; \text{ 因此}$$

$$[0109] \quad n = \frac{\Delta A}{Q} \quad (\text{方程式 15})$$

[0110] 因为q为最不可能符号的真概率,所以基于方程式14,可如下计算在Section A中具有成一序列的n个最可能符号0的概率Pmpsr:

$$[0111] \quad Pmpsr = (1 - q)^n \Rightarrow \ln(Pmpsr) = \frac{\Delta A}{Q} \cdot \ln(1 - q) \quad (\text{方程式 16})$$

[0112] 对于q的小值,可进行以下近似:

$$[0113] \quad \ln(Pmpsr) \approx -\frac{\Delta A}{Q} \cdot q; \text{ 因此} \quad (\text{方程式 17})$$

$$[0114] \quad Pmpsr = \exp\left(-\frac{\Delta A}{Q} \cdot q\right) \quad (\text{方程式 18})$$

[0115] 在存在相等数目个最可能符号和最不可能符号重新正规化的情况下,Pmpsr=0.5。因此:

$$[0116] \quad \ln(Pmpsr) = \ln(2) \approx -0.6391 \quad (\text{方程式 19})$$

[0117] 在使用方程式17和19且求解Q的情况下,导出下式:

$$[0118] \quad Q = \frac{\Delta A}{\ln(2)} \cdot q \quad (\text{方程式 20})$$

[0119] 假定 ΔA 通常小于0.75(即,约0.639或 $\ln(2)$),那么理论上, $Q \approx q$,且近似因此保持成立。

[0120] 因为 ΔA 的量值取决于重新正规化的类型, 所以如果最后发生最可能符号重新正规化, 那么 ΔA 接近于 0.75。此外, 如果最后发生最不可能符号, 那么 ΔA 通常略小于 0.75。此外, 如果 Q 过大, 那么 P_{mpsr} 也大且算法倾向于移动到 Q 的较小值。相反地, 如果 Q 过小, 那么 P_{mpsr} 小且算法倾向于移动到 Q 的较大值。因此, 算法调适为且平衡为大约接近于 $Q \approx q$ 的情况。

[0121] 应注意, 所属领域的一般技术人员基于本发明应理解, 编码和解码技术可实施于如本文中所描述的系统和方法的不同实施例中。此外, 所实施的编码和解码技术可基于本文中所描述的关于伪代码 A 和伪代码 B 的实例 (例如, 基于乘数的途径、基于无乘数的途径) 中的任一种, 或任何相似译码技术, 其中所述系统和方法通过基于一或多个经编码符号来执行多个并行数学比较而确定一系列一或多个经解码符号中的连续最可能符号的数目。

[0122] 图 1 为说明可通过使用如本文中所论述的并行处理技术来利用和实施基于算术译码的译码技术的实例视频编码和解码系统 10 的框图。如图 1 所展示, 系统 10 包含源装置 12, 其提供将在稍后时间由目的地装置 14 解码的经编码视频数据。可运用基于算术译码的译码技术 (例如根据伪代码 A 或相似译码技术) 且如本文中所描述来编码视频数据。可运用基于算术译码的译码技术 (例如根据伪代码 B 或相似译码技术) 且如本文中所描述来解码视频数据。具体地说, 源装置 12 通过计算机可读媒体 16 将视频数据提供到目的地装置 14。源装置 12 和目的地装置 14 可包括广泛范围的装置中的任一种, 包含台式电脑、笔记本 (即, 膝上型) 电脑、平板电脑、机顶盒、电话手机 (例如所谓的 “智能型” 电话、所谓的 “智能型” 板)、电视、摄像机、显示装置、数字媒体播放器、视频游戏主控台、视频流式传输装置或其类似者。在一些状况下, 源装置 12 和目的地装置 14 可配备成用于无线通信。

[0123] 目的地装置 14 可通过计算机可读媒体 16 来接收待解码的经编码视频数据。计算机可读媒体 16 可包括能够将经编码视频数据从源装置 12 移动到目的地装置 14 的任何类型的媒体或装置。在一个实例中, 计算机可读媒体 16 可包括通信媒体以使源装置 12 能够实时地将经编码视频数据直接传输到目的地装置 14。可根据通信标准 (例如无线通信协议) 来调制经编码视频数据, 且将其传输到目的地装置 14。通信媒体可包括任何无线或有线通信媒体, 例如射频 (RF) 频谱或一或多个物理传输线。通信媒体可形成基于包的网路 (例如局域网、广域网、或例如因特网的全球网路) 的部分。通信媒体可包含路由器、交换器、基站, 或可有用于促进从源装置 12 到目的地装置 14 的通信的任何其它设备。

[0124] 在一些实例中, 可将经编码数据从输出接口 22 输出到存储装置。相似地, 可由输入接口从存储装置存取经编码数据。存储装置可包含多种分散式或本地存取式数据存储媒体中的任一种, 例如硬盘机、蓝光 (Blu-ray) 光盘、DVD、CD-ROM、快闪存储器、易失性或非易失性存储器, 或用于存储经编码视频数据的任何其它合适的数字存储媒体。在另外实例中, 存储装置可对应于文件服务器或另一中间存储装置, 其可存储由源装置 12 产生的经编码视频。目的地装置 14 可通过流式传输或下载而从存储装置存取经存储视频数据。文件服务器可为能够存储经编码视频数据且将那些经编码视频数据传输到目的地装置 14 的任何类型的服务器。实例文件服务器包含网页服务器 (例如, 用于网站)、FTP 服务器、网络附接存储 (NAS) 装置或本地磁碟机。目的地装置 14 可通过任何标准数据连接 (包含因特网连接) 来存取经编码视频数据。这个数据连接可包含适合于存取存储于文件服务器上的经编码视频数据的无线信道 (例如, Wi-Fi 连接)、有线连接 (例如, DSL、缆线调制解调器等等) 或这两个的

组合。从存储装置的经编码视频数据传输可为流式传输传输、下载传输或其组合。

[0125] 本发明的技术未必限于无线应用或设定。所述技术可应用于视频译码以支持多种多媒体应用中的任一种，例如空中电视广播、有线电视传输、卫星电视传输、因特网流式传输视频传输（例如HTTP动态调适性流式传输（DASH））、编码到数据存储媒体上的数字视频、存储于数据存储媒体上的数字视频的解码，或其它应用。在一些实例中，系统10可配置成支持单向或双向视频传输以支持例如视频流式传输、视频播放、视频广播和/或视频电话的应用。

[0126] 在图1的实例中，源装置12包含视频源18、视频编码器20和输出接口22。目的地装置14包含输入接口28、视频解码器30和显示装置32。根据本发明，源装置12的视频编码器20可配置成应用用于本文中所论述的算术译码的技术，例如相似于关于伪代码A所描述的技术的编码技术。在其它实例中，源装置和目的地装置可包含其它组件或布置。举例来说，源装置12可从外部视频源18（例如外部摄影机）接收视频数据。同样地，目的地装置14可与外部显示装置介接，而非包含集成式显示装置。

[0127] 图1的所说明系统10仅仅为一个实例。用于利用本文中所论述的算术译码技术的技术可由任何数字视频编码和/或解码装置执行。尽管本发明的技术通常是由视频编码装置执行，但所述技术还可由视频编码器/解码器（通常被称作“CODEC”）执行。此外，本发明的技术还可由视频预处理器执行，或甚至由不限于视频数据的一般编码器/解码器执行。源装置12和目的地装置14仅仅为此类译码装置的实例，其中源装置12产生经译码视频数据以供传输到目的地装置14。在一些实例中，装置12、14可以大体上对称方式而操作，使得装置12、14中的每一个包含视频编码和解码组件。因此，系统10可支持视频装置12、14之间的单向或双向视频传输，例如，用于视频流式传输、视频播放、视频广播或视频电话。

[0128] 源装置12的视频源18可包含视频捕捉装置，例如摄像机、含有经先前捕捉视频的视频文件库和/或用以从视频内容提供者接收视频的视频馈送接口。作为另外替代例，视频源18可产生作为源视频的基于计算机图形的数据，或实况视频、经存档视频和经计算机产生视频的组合。在一些状况下，如果视频源18为摄像机，那么源装置12和目的地装置14可形成所谓的摄像机电话或视频电话。然而，如上文所提及，本发明中所描述的技术可大体上适用于视频译码，或甚至适用于译码，且可应用于无线和/或有线应用。在每一状况下，经捕捉、经预捕捉或经计算机产生视频可由视频编码器20编码。经编码视频信息可接着由输出接口22输出到计算机可读媒体16上。

[0129] 计算机可读媒体16可包含：瞬态媒体，例如无线广播或有线网络传输；或存储媒体（即，非暂时性存储媒体），例如硬盘、随身碟、紧密光盘、数字视频光盘、蓝光光盘或其它计算机可读媒体。在一些实例中，网络服务器（未图示）可从源装置12接收经编码视频数据，且（例如）通过网络传输将经编码视频数据提供到目的地装置14。相似地，媒体生产设施（例如光盘冲压设施）的计算装置可从源装置12接收经编码视频数据且生产含有经编码视频数据的光盘。因此，在各种实例中，计算机可读媒体16可被理解为包含各种形式的一或多个计算机可读媒体。

[0130] 目的地装置14的输入接口28从计算机可读媒体16接收信息。计算机可读媒体16的信息可包含由视频编码器20定义的语法信息，所述语法信息也是由视频解码器30使用，其包含描述块和其它经译码单元（例如，GOP）的特性和/或处理的语法元素。显示装置32将经

解码视频数据显示给用户,且可包括多种显示装置中的任一种,例如阴极射线管(CRT)、液晶显示器(LCD)、等离子体显示器、有机发光二极管(OLED)显示器或另一类型的显示装置。

[0131] 尽管图1中未图示,但在一些方面中,视频编码器20和视频解码器30可各自与音频编码器和解码器集成,且可包含适当MUX-DEMUX单元或其它硬件和软件,以处置共同数据流或单独数据流中的音频和视频两个的编码。适用时,MUX-DEMUX单元可符合ITU H.223多路复用器协议,或例如用户数据报协议(UDP)的其它协议。

[0132] 视频编码器20和视频解码器30(或实施本文中所论述的关于使用并行处理技术的基于算术译码的译码技术的技术的任何编码器或解码器)各自可被实施为多种合适编码器电路系统中的任一种,例如一或多个微处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)、离散逻辑、软件、硬件、固件或其任何组合。当所述技术是部分地以软件予以实施时,装置可将用于所述软件的指令存储于合适的非暂时性计算机可读媒体中,且使用一或多个处理器而以硬件执行所述指令以执行本发明的技术。视频编码器20和视频解码器30中的每一个可包含于一或多个编码器或解码器中,所述一或多个编码器或解码器中的任一个可被集成为各别装置中的组合式编码器/解码器(CODEC)的部分。

[0133] 本发明可大体上涉及视频编码器20将某些信息“传信”到例如视频解码器30的另一装置。术语“传信”可大体上指用以解码经压缩视频数据的语法元素和/或其它数据的通信。可实时地或近似实时地发生此类通信。替代地,此通信可发生在一时间跨度中,例如可能在编码时在经编码位流中将语法元素存储到计算机可读存储媒体时发生,接着可在存储到此媒体之后由解码装置在任何时间检索所述语法元素。

[0134] 视频编码器20和视频解码器30可根据视频译码标准操作。由ITU-T视频译码专家组(VCEG)和ISO/IEC动画专家组(MPEG)的视频译码联合协作小组(JCT-VC)以及3D视频译码延伸开发联合协作小组(JCT-3V)开发的实例视频译码标准包含高效率视频译码(HEVC)或ITU-T H.265,包含其范围延伸、多视图延伸(MV-HEVC)和可调式延伸(SHVC)。已定案的HEVC标准文件被公开为“ITU-T H.265,SERIES H:AUDIOVISUAL AND MULTIMEDIA SYSTEMS Infrastructure of audiovisual services-Coding of moving video-High efficiency video coding”,国际电信联合会(ITU)的电信标准化部门,2013年4月。替代地,视频编码器20和视频解码器30可根据其它专有或工业标准(例如ISO/IEC MPEG-4Visual和ITU-T H.264(也被称作ISO/IEC MPEG-4AVC),包含其可调式视频译码(SVC)延伸和多视图视频译码(MVC)延伸)而操作。然而,本发明的技术并不限于任何特定译码标准。举例来说,本发明的技术可与多种其它专有或非专有视频译码技术或后续标准(例如ITU-T H.266)一起使用。

[0135] 图2为说明可通过使用如本文中所论述的并行处理技术来利用和实施基于算术译码的译码技术的视频编码器20的实例的框图。视频编码器20可执行视频切片内的视频块的帧内译码和帧间译码。帧内译码依赖于空间预测以缩减或去除给定视频帧或图片内的视频中的空间冗余。帧间译码依赖于时间预测以缩减或去除视频序列的邻近帧或图片内的视频中的时间冗余。帧内模式(I模式)可指若干基于空间的译码模式中的任一种。帧间模式(例如单向预测(P模式)或双向预测(B模式))可指若干基于时间的译码模式中的任一种。

[0136] 如图2所展示,视频编码器20接收待编码视频帧内的当前视频块。在图2的实例中,视频编码器20包含视频数据存储单元38、模式选择单元40、参考图片存储器64、求和器50、变

换处理单元52、量化单元54和熵编码单元56。模式选择单元40又包含运动补偿单元44、运动估计单元42、帧内预测单元46和分割单元48。出于视频块重新构建起见,视频编码器20还包含反量化单元58、反变换单元60和求和器62。还可包含解块滤波器(图2中未图示)以滤波块边界以从经重新构建视频去除块效应伪影。视需要,解块滤波器将通常滤波求和器62的输出。除了解块滤波器以外,还可使用额外滤波器(循环内或循环后)。出于简洁起见而未展示此类滤波器,但视需要,此类滤波器可滤波求和器50的输出(作为循环内滤波器)。

[0137] 在编码过程期间,视频编码器20接收待译码视频帧或切片。可将所述帧或切片划分成多个视频块。视频数据存储器38可存储待由视频编码器20的组件编码的视频数据。可(例如)从视频源18获得存储于视频数据存储器38中的视频数据。参考图片存储器64可被称作存储参考视频数据以供视频编码器20(例如)在帧内或帧间译码模式中用来编码视频数据的DPB。视频数据存储器38和参考图片存储器64可由例如以下各种的多种存储器装置中的任一种形成:动态随机存取存储器(DRAM)(包含同步DRAM(SDRAM))、磁阻式RAM(MRAM)、电阻式RAM(RRAM)或其它类型的存储器装置。视频数据存储器38和参考图片存储器64可由同一存储器装置或单独存储器装置提供。在各种实例中,视频数据存储器38可与视频编码器20的其它组件一起在芯片上,或相对于那些组件在芯片外。

[0138] 运动估计单元42和运动补偿单元44执行经接收视频块相对于一或多个参考帧中的一或多个块的帧间预测性译码以提供时间预测。帧内预测单元46可替代地执行经接收视频块相对于与待译码块在同一帧或切片中的一或多个相邻块的帧内预测性译码以提供空间预测。视频编码器20可执行多个译码遍次,例如,以选择用于每一视频数据块的适当译码模式。

[0139] 此外,分割单元48可基于先前译码遍次中的先前分割方案的评估而将视频数据块分割成子块。举例来说,分割单元48可最初将帧或切片分割成LCU,且基于速率-失真分析(例如,速率-失真优化)而将LCU中的每一个分割成子CU。模式选择单元40可进一步产生指示将LCU分割成子CU的四元树数据结构。四元树的叶节点CU可包含一或多个PU和一或多个TU。

[0140] 模式选择单元40可(例如)基于误差结果来选择译码模式(帧内或帧间)中的一种,且将所得经帧内译码或经帧间译码块提供到求和器50以产生残余块数据,且将所得经帧内译码或经帧间译码块提供到求和器62以重新构建经编码块以供用作参考帧。模式选择单元40还将语法元素(例如运动向量、帧内模式指示符、分割区信息和其它此类语法信息)提供到熵编码单元56。

[0141] 运动估计单元42和运动补偿单元44可高度地集成,但出于概念目的而被单独地说明。由运动估计单元42执行的运动估计为产生运动向量的过程,所述运动向量估计用于视频块的运动。举例来说,运动向量可指示当前视频帧或图片内的视频块的PU相对于参考帧(或其它经译码单元)内的预测性块相对于当前帧(或其它经译码单元)内正被译码的当前块的位移。预测性块为在像素差方面被发现接近地匹配于待译码块的块,所述像素差可由绝对差和(SAD)、平方差和(SSD)或其它差度量确定。在一些实例中,视频编码器20可计算存储于参考图片存储器64中的参考图片的子整数像素位置的值。举例来说,视频编码器20可内插参考图片的四分之一像素位置、八分之一像素位置或其它分数像素位置的值。因此,运动估计单元42可执行关于全像素位置和分数像素位置的运动搜索且输出具有分数像素精

确度的运动向量。

[0142] 运动估计单元42通过比较PU的位置与参考图片的预测性块的位置来计算用于经帧间译码切片中的视频块的PU的运动向量。参考图片可选自第一参考图片列表(列表0)或第二参考图片列表(列表1),所述列表中的每一个识别存储于参考图片存储器64中的一或多个参考图片。运动估计单元42将经计算运动向量发送到熵编码单元56和运动补偿单元44。

[0143] 由运动补偿单元44执行的运动补偿可涉及基于由运动估计单元42确定的运动向量来提取或产生预测性块。再次,在一些实例中,运动估计单元42与运动补偿单元44可功能上集成。在接收到用于当前视频块的PU的运动向量后,运动补偿单元44就可在所述参考图片列表中的一项中定位所述运动向量所指向的预测性块。求和器50通过从正被译码的当前视频块的像素值减去预测性块的像素值来形成残余视频块,从而形成像素差值,如下文所论述。一般来说,运动估计单元42执行关于明度分量的运动估计,且运动补偿单元44将基于明度分量所计算的运动向量用于色度分量和明度分量两种。模式选择单元40也可产生与视频块和视频切片相关联的语法元素以供视频解码器30用来解码视频切片的视频块。

[0144] 作为由运动估计单元42和运动补偿单元44执行的帧间预测(如上文所描述)的替代例,帧内预测单元46可帧内预测当前块。具体地说,帧内预测单元46可确定待用以编码当前块的帧内预测模式。在一些实例中,帧内预测单元46可(例如)在单独编码遍次期间使用各种帧内预测模式来编码当前块,且帧内预测单元46(或在一些实例中为模式选择单元40)可从经测试模式选择待使用的适当帧内预测模式。

[0145] 举例来说,帧内预测单元46可使用用于各种经测试帧内预测模式的速率-失真分析来计算速率-失真值,且在所述经测试模式当中选择具有最优选速率-失真特性的帧内预测模式。速率-失真分析通常确定经编码块与原始未经编码块(其经编码以产生经编码块)之间的失真(或误差)的量,以及用以产生经编码块的位率(即,位的数目)。帧内预测单元46可从用于各种经编码块的失真和速率计算比率以确定哪一帧内预测模式展现块的最优选速率-失真值。

[0146] 在选择用于块的帧内预测模式之后,帧内预测单元46可将指示用于块的经选择帧内预测模式的信息提供到熵编码单元56。熵编码单元56可编码指示经选择帧内预测模式的信息。视频编码器20可在经传输位流中包含以下各种:配置数据,其可包含多个帧内预测模式索引数据表和多个经修改帧内预测模式索引数据表(也被称作码字映射数据表);用于各种块的编码上下文的定义;和待用于所述上下文中的每一个的最可能帧内预测模式、帧内预测模式索引数据表和经修改帧内预测模式索引数据表的指示。

[0147] 视频编码器20通过从正被译码的原始视频块减去来从模式选择单元40的预测数据来形成残余视频块。求和器50表示执行这个减去运算的一或多个组件。变换处理单元52将变换(例如离散余弦变换(DCT)或概念上相似变换)应用于残余块,从而产生包括残余变换系数值的视频块。变换处理单元52可执行概念上相似于DCT的其它变换。也可使用小波变换、整数变换、子频带变换,或其它类型的变换。在任何状况下,变换处理单元52将变换应用于残余块,从而产生残余变换系数块。变换可将残余信息从像素值域转换为变换域,例如频域。变换处理单元52可将所得变换系数发送到量化单元54。量化单元54量化变换系数以进一步缩减位率。量化过程可缩减与所述系数中的一些或全部相关联的位深度。可通过调整

量化参数来修改量化程度。在一些实例中,量化单元54可接着执行包含经量化变换系数的矩阵的扫描。替代地,熵编码单元56可执行所述扫描。

[0148] 在量化之后,熵编码单元56熵译码经量化变换系数。举例来说,熵编码单元56可执行上下文调适性可变长度译码(CAVLC)、上下文调适性二进制算术编码(CABAC)、基于语法的上下文调适性二进制算术译码(SBAC)、概率区间分割熵(PIPE)译码、Q译码、调适性Q译码、算术译码或另一熵译码技术。在基于上下文的熵译码的状况下,上下文可基于相邻块。具体地说,在一些实施例中,熵编码单元56可配置成通过使用如本文中所论述的并行处理技术而根据基于算术译码的译码技术来译码视频数据(例如,经量化变换系数)。举例来说,在一些实施例中,熵编码单元56配置成基于如本文中所描述的伪代码A来实施算术译码技术。熵编码单元56可配置成使用某些方程式以近似某些值或方程式(例如以伪代码A),以确保在熵编码单元56和对应解码器处的寄存器值的匹配跟踪。举例来说,熵编码单元56可配置成利用或实施方程式1、8、9、10、12和13中的一或多个者。

[0149] 在由熵编码单元56进行熵译码之后,可将包括一或多个经编码符号(例如,最不可能符号和最可能符号)的字串经编码位流传输到另一装置(例如,视频解码器30)或进行存档以供稍后传输或检索。

[0150] 反量化单元58和反变换单元60分别应用反量化和反变换以在像素域中重新构建残余块,例如,以供稍后用作参考块。运动补偿单元44可通过将残余块与参考图片存储器64的帧中的一个的预测性块相加来计算参考块。运动补偿单元44也可将一或多个内插滤波器应用于经重新构建残余块以计算用于运动估计的子整数像素值。求和器62将经重新构建残余块与由运动补偿单元44产生的经运动补偿预测块相加,以产生经重新构建视频块以供存储于参考图片存储器64中。经重新构建视频块可由运动估计单元42和运动补偿单元44用作参考块以帧间译码后续视频帧中的块。

[0151] 图3为说明可通过使用如本文中所论述的并行处理技术来利用和实施基于算术译码的译码技术的视频解码器30的实例的框图。在图3的实例中,视频解码器30包含视频数据存储器68、熵解码单元70、运动补偿单元72、帧内预测单元74、反量化单元76、反变换单元78、参考图片存储器82和求和器80。在一些实例中,视频解码器30可执行与关于视频编码器20(图2)所描述的编码遍次大体上互逆的解码遍次。运动补偿单元72可基于从熵解码单元70接收的运动向量来产生预测数据,而帧内预测单元74可基于从熵解码单元70接收的帧内预测模式指示符来产生预测数据。

[0152] 在解码过程期间,视频解码器30从视频编码器20接收表示经编码视频切片的视频块和关联语法元素的包括一或多个经编码符号(例如,最不可能符号和最可能符号)的字串经编码视频位流。可(例如)从计算机可读媒体(例如,从本地视频源,例如摄像机)通过视频数据的有线或无线网络通信或通过存取物理数据存储媒体来获得存储于视频数据存储器68中的视频数据。视频数据存储器68可形成存储来自经编码视频位流的经编码视频数据的经译码图片缓冲器(CPB)。参考图片存储器82可被称作存储参考视频数据以供视频解码器30(例如)在帧内或帧间译码模式中用来解码视频数据的DPB。视频数据存储器68和参考图片存储器82可由例如以下各种的多种存储器装置中的任一种形成:动态随机存取存储器(DRAM)(包含同步DRAM(SDRAM))、磁阻式RAM(MRAM)、电阻式RAM(RRAM)或其它类型的存储器装置。视频数据存储器68和参考图片存储器82可由同一存储器装置或个别存储器装置提

供。在各种实例中,视频数据存储器68可与视频解码器30的其它组件一起在芯片上,或相对于那些组件在芯片外。

[0153] 视频解码器30的熵解码单元70熵解码位流以产生经量化系数、运动向量或帧内预测模式指示符和其它语法元素。具体地说,在一些实施例中,熵解码单元70可配置成通过使用如本文中所论述的并行处理技术而根据基于算术译码的译码技术来解码位流。举例来说,在一些实施例中,熵解码单元70配置成基于如本文中所描述的伪代码B来实施算术解码技术。熵解码单元70可配置成使用某些方程式以近似某些值或方程式(例如以伪代码B),以便确保在熵解码单元70和对应熵编码单元56处的寄存器值的匹配跟踪。举例来说,熵解码单元70可配置成利用或实施方程式1、8、9、10、12和13中的一或多个。此外,熵解码单元70可配置成使用如本文中所描述的并行处理技术(例如关于伪代码B)以解码位流。举例来说,熵解码单元70可配置成执行多个并行数学比较(例如,方程式2.N和/或11.n)以便基于位流来确定待解码的连续最可能符号的数目。因此,熵解码单元70可配置成大约以位流的数据速率而根据伪代码B的Section A来解码符号。

[0154] 另外或替代地,熵解码单元70配置成实施并行块以用于选择符号以根据伪代码B的Section B而解码(例如,从位流读取额外符号以供解码),如本文中所论述。因此,熵解码单元70可配置成大约以位流的数据速率执行伪代码B的Section B。此外,熵解码单元70可配置成在单一时钟循环中对Section A和Section B执行并行处理的反复。

[0155] 熵解码单元70将运动向量和其它语法元素转递到运动补偿单元72。视频解码器30可在视频切片层级和/或视频块层级处接收语法元素。

[0156] 当将视频切片译码为经帧内译码(I)切片时,帧内预测单元74可基于经传信帧内预测模式和来自当前帧或图片的经先前解码块的数据来产生用于当前视频切片的视频块的预测数据。当将视频帧译码为经帧间译码(即,B、P或GPB)切片时,运动补偿单元72基于运动向量和从熵解码单元70接收的其它语法元素来产生用于当前视频切片的视频块的预测性块。可从参考图片列表中的一项内的参考图片中的一个产生预测性块。视频解码器30可基于存储于参考图片存储器82中的参考图片而使用默认构建技术来构建参考帧列表:列表0和列表1。

[0157] 运动补偿单元72通过剖析运动向量和其它语法元素来确定用于当前视频切片的视频块的预测信息,且使用所述预测信息以产生用于正被解码的当前视频块的预测性块。举例来说,运动补偿单元72使用经接收语法元素中的一些以确定用以译码视频切片的视频块的预测模式(例如,帧内或帧间预测)、帧间预测切片类型(例如,B切片、P切片或GPB切片)、用于所述切片的参考图片列表中的一或多项的构建信息、用于所述切片的每一经帧间编码视频块的运动向量、用于所述切片的每一经帧间译码视频块的帧间预测状态、以及用以解码当前视频切片中的视频块的其它信息。

[0158] 运动补偿单元72也可执行基于内插滤波器的内插。运动补偿单元72可使用如由视频编码器20在编码视频块期间使用的内插滤波器以计算参考块的子整数像素的经内插值。在这种状况下,运动补偿单元72可从经接收语法元素确定由视频编码器20使用的内插滤波器且使用所述内插滤波器以产生预测性块。

[0159] 反量化单元76反量化(即,解量化)位流中所提供且由熵解码单元70解码的经量化变换系数。反量化过程可包含使用由视频解码器30针对视频切片中的每一视频块所计算的

量化参数QPY以确定应被应用的量化程度且同样地确定应被应用的反量化程度。

[0160] 反变换单元78将反变换(例如,反DCT、反整数变换或概念上相似反变换过程)应用于变换系数,以便在像素域中产生残余块。

[0161] 在运动补偿单元72基于运动向量和其它语法元素来产生用于当前视频块的预测性块之后,视频解码器30通过求和来从反变换单元78的残余块与由运动补偿单元72产生的对应预测性块来形成经解码视频块。求和器80表示执行这项求和运算的一或多个组件。视需要,还可应用解块滤波器以滤波经解码块以便去除块效应伪影。还可使用其它循环滤波器(在译码循环中或在译码循环之后)以使像素转变平滑,或以其它方式改进视频质量。接着将给定帧或图片中的经解码视频块存储于参考图片存储器82中,参考图片存储器82存储用于后续运动补偿的参考图片。参考图片存储器82也存储经解码视频以供稍后呈现于显示装置(例如图1的显示装置32)上。

[0162] 图4为说明可通过使用如本文中所论述的并行处理技术来利用和实施基于算术译码的译码技术的解码器400的实例的框图。举例来说,解码器400可为熵解码单元70的实例。

[0163] 如所展示,解码器400包含缓冲器405(或其它类型的适当存储器)。如所展示,缓冲器405为先进先出(FIFO)缓冲器。缓冲器405可配置成在经编码数据流的每时钟循环接收M个符号(例如,位),所述经编码数据流包括使用基于算术译码的译码技术(例如,基于伪代码A;且缓冲器405可对应于寄存器T)所编码的经编码符号字串。经编码符号字串可表示一系列一或多个经解码符号,其中所述一或多个经解码符号中的每一个包括最可能符号(例如,0)和最不可能符号(例如,1)中的一个。

[0164] 解码器400可进一步包含并行比较单元410和并行移位与补充单元415。并行比较单元410可配置成实施用于根据伪代码B的Section A来解码数据的一或多种并行处理技术,或相似并行处理技术。并行移位与补充单元415可配置成实施用于根据伪代码B的Section B来解码符号的并行块(如本文中所论述),或相似并行处理技术。

[0165] 举例来说,并行比较单元410可配置成从缓冲器405接收经编码符号作为输入,以及从寄存器(例如,L、U、Cum_count()、Count_0、Total_Count等等;来自伪代码B且可如所论述而初始化)接收包含变量和值的输入。并行比较单元410可配置成使用经接收输入以通过执行多个并行数学比较(例如,例如以方程式2.N和/或11.n)来确定所述系列一或多个经解码符号中的连续最可能符号的数目。并行比较单元410可配置成基于所述多个并行数学比较来解码所述经编码符号中的一或多个以输出所述系列一或多个经解码符号中的符号群组,其中所述符号群组包含所述数目个连续最可能符号。如关于方程式2.N和11.n所论述,所述符号群组还可包含在所述数目个连续最可能符号之后的一个最不可能符号。

[0166] 并行比较单元410还可配置成执行其它计算以更新伪代码B的Section A的寄存器(例如,L和U)中的值。并行移位与补充单元415可配置成接收经更新寄存器值作为输入,且从缓冲器405接收输入,以实施并行块以用于根据伪代码B的Section B来选择符号以供解码(例如,从经编码数据流读取额外符号以供解码),如本文中所论述。并行比较单元410和并行移位与补充单元415可配置成皆在单一时钟循环中一起执行单一反复(包含执行并行比较和并行块移位与补充)。因此,解码器400配置成通过使用如本文中所论述的并行处理技术来解码使用基于算术译码的译码技术所编码的数据。

[0167] 图5为用于解码经压缩数据的过程500的流程图的说明。过程500可实施于任何合

适解码器(例如本文中所描述的解码器400)中。如所展示,在块505处,解码器在缓冲器中接收数据流的至少一部分,所述数据流包括使用基于算术译码的译码技术所编码的经编码符号字串。经编码符号字串表示一系列一或多个经解码符号。一或多个经解码符号中的每一个包括最可能符号和最不可能符号中的一个。

[0168] 此外,在块510处,解码器通过基于经编码符号中的一或多个来执行多个并行数学比较(例如,例如以方程式2.N和/或11.n且如关于伪代码B的Section A所论述)而确定所述系列一或多个经解码符号中的连续最可能符号的数目。

[0169] 在块515处继续,解码器基于所述多个并行数学比较来解码所述系列一或多个经解码符号中的符号群组。所述符号群组包括所述数目个连续最可能符号。如关于方程式2.N和11.n所论述,所述符号群组还可包含在所述数目个连续最可能符号之后的一个最不可能符号。

[0170] 此外,在块520处,解码器执行其它计算以更新寄存器中的值(例如,如关于伪代码B的Section A所论述)。

[0171] 在块525处,解码器确定数据流中是否存在任何更多的经编码符号以供解码。如果在块525处确定不存在额外经编码符号以供解码,那么过程结束。如果在块525处确定存在额外经编码符号以供解码,那么过程继续到块530。

[0172] 在块530处,解码器使用并行处理以更新寄存器中的值且将来自数据流的额外经编码符号读取到缓冲器中。举例来说,解码器可对某些寄存器值执行移位与补充,且基于寄存器值来检查是否满足某些准则以确定如何将来自数据流的额外经编码符号读取到缓冲器中(例如,如关于伪代码B的Section B所论述)。过程500可接着返回到块510。

[0173] 如上文所论述,对于实施并行处理技术以用于基于算术译码技术而解码的解码器,用于实施伪代码B的Section A和Section B的并行度为基于用以实施所述技术的成本、效率增益和硬件类型的设计选择。举例来说,在解码器400中,可进行关于缓冲器405的大小、并行比较单元410配置成并行地进行的比较的数目、并行移位与补充单元415配置成并行地进行的并行块反复的数目的选择。这些选择可影响解码数据所需要的时钟循环的数目。

[0174] 在一些实施例中,将并行比较单元410配置成并行地进行的比较的数目设定为6,将并行移位与补充单元415配置成并行地进行的并行块反复的数目设定为7,且并行比较单元410和并行移位与补充单元415配置成在单一时钟循环中操作。因此,可在仍达成良好解码速率的同时缩减缓冲器405的大小。此外,输入符号速率可为每时钟循环单一输入符号,或每时钟循环两个输入符号(例如,在双数据速率(DDR)系统中)。在一些实施例中,将并行比较单元410配置成并行地进行的比较的数目设定为4以缩减并行比较单元410的复杂度。此外,硬件复杂度和成本对性能可仍引起良好性能。所属领域的一般技术人员将认识到,可进行其它设计选择,然而,这些特定设计选择或相似设计选择可提供关于性能对复杂度的某些优点。

[0175] 应认识到,取决于实例,本文中所描述的技术中的任一项的某些动作或事件可以不同序列被执行,可被添加、合并或完全地省去(例如,并非所有所描述动作或事件为实践所述技术所必要)。此外,在某些实例中,可同时地(例如,经由多线程处理、中断处理或多个处理器)而非依序地执移动作或事件。

[0176] 在一或多个实例中,所描述功能可以硬件、软件、固件或其任何组合予以实施。如果以软件予以实施,那么所述功能可作为一或多个指令或代码而存储于计算机可读媒体上或经由计算机可读媒体进行传输,且由基于硬件的处理单元执行。计算机可读媒体可包含:计算机可读存储媒体,其对应于例如数据存储媒体的有形媒体;或通信媒体,其包含促进将计算机程序从一处传送到另一处(例如,根据通信协议)的任何媒体。以这种方式,计算机可读媒体通常可对应于(1)为非暂时性的有形计算机可读存储媒体,或(2)例如信号或载波的通信媒体。数据存储媒体可为可由一或多个计算机或一或多个处理器存取以检索用于实施本发明中所描述的技术的指令、代码和/或数据结构的任何可用媒体。计算机程序产品可包含计算机可读媒体。

[0177] 作为实例而非限制,此类计算机可读存储媒体可包括RAM、ROM、EEPROM、CD-ROM或其它光盘存储装置、磁碟存储装置或其它磁性存储装置、快闪存储器、或可用以存储呈指令或数据结构形式的所要代码且可由计算机存取的任何其它媒体。而且,任何连接被适当地称为计算机可读媒体。举例来说,如果使用同轴缆线、光缆、双绞线、数字用户线(DSL)或无线技术(例如红外线、无线电和微波)而从网站、服务器或其它远程源传输指令,那么同轴缆线、光缆、双绞线、DSL或无线技术(例如红外线、无线电和微波)包含于媒体的定义中。然而,应理解,计算机可读存储媒体和数据存储媒体并不包含连接、载波、信号或其它暂时性媒体,而是有关于非暂时性有形存储媒体。如本文中所使用,磁碟和光盘包含紧密光盘(CD)、激光光盘、光学光盘、数字多功能光盘(DVD)、软性磁碟和蓝光光盘,其中磁碟通常以磁性方式再现数据,而光盘运用激光以光学方式再现数据。以上各种的组合还应包含于计算机可读媒体的范围内。

[0178] 指令可由例如以下各种的一或多个处理器执行:一或多个数字信号处理器(DSP)、一般用途微处理器、专用集成电路(ASIC)、现场可编程门阵列(FPGA)、或其它等效集成式或离散逻辑电路系统。因此,如本文中所使用的术语“处理器”可指上述结构或适合于实施本文中所描述的技术的任何其它结构中的任一种。另外,在一些方面中,本文中所描述的功能性可提供于配置成用于编码和解码的专用硬件和/或软件模块内,或并入于组合式编解码器中。而且,所述技术可完全地实施于一或多个电路或逻辑元件中。

[0179] 本发明的技术可实施于广泛多种装置或设备中,所述装置或设备包含无线手机、集成电路(IC)或IC集合(例如,芯片集)。在本发明中描述各种组件、模块或单元以强调配置成执行所公开技术的装置的功能方面,但未必要求由不同硬件单元来实现。实情为,如上文所描述,各种单元可结合合适的软件和/或固件而组合于编解码器硬件单元中或由互操作性硬件单元的集合提供,硬件单元包含如上文所描述的一或多个处理器。已描述各种实例。这些和其它实例是在以下权利要求书的范围内。

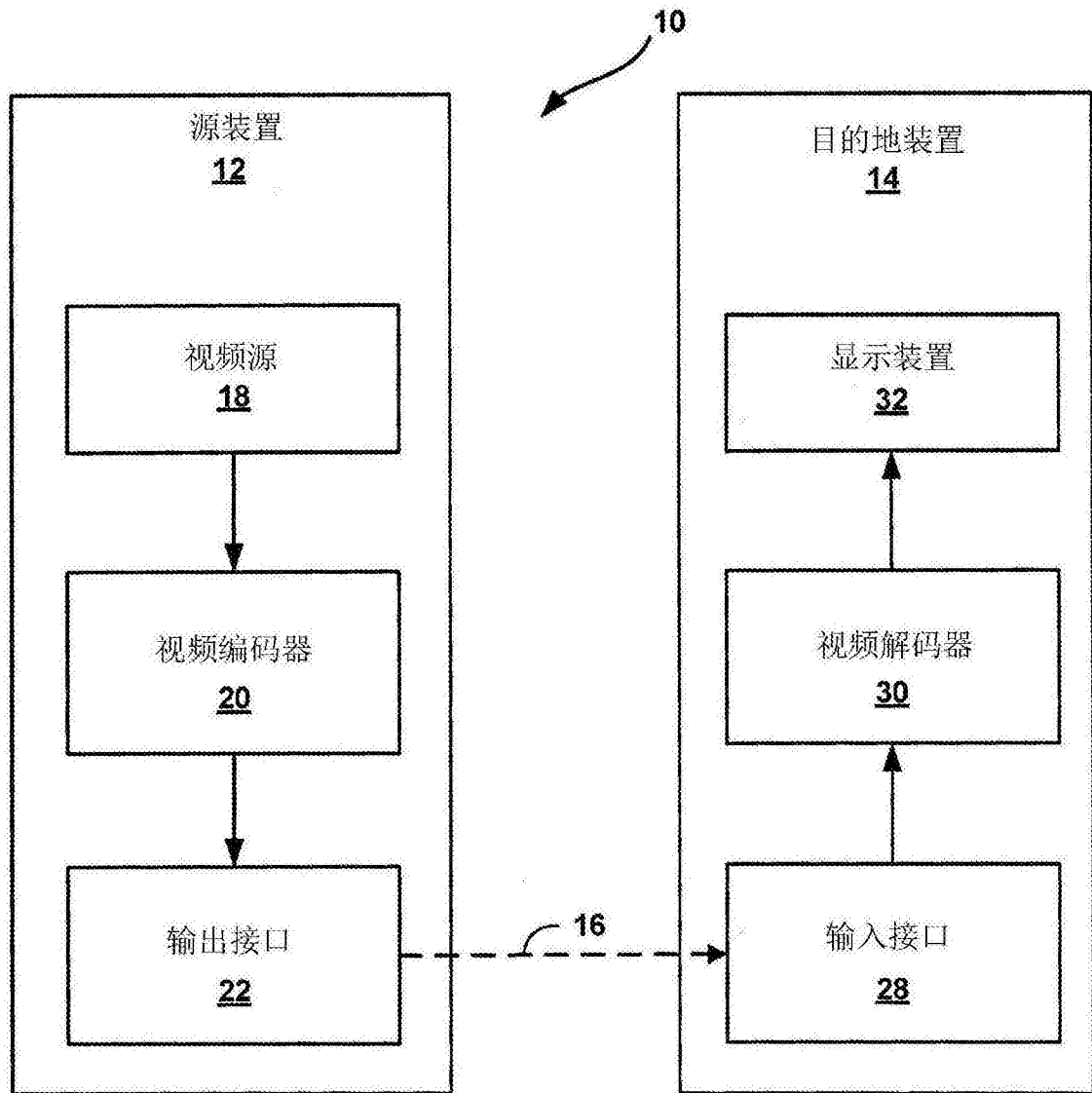


图1

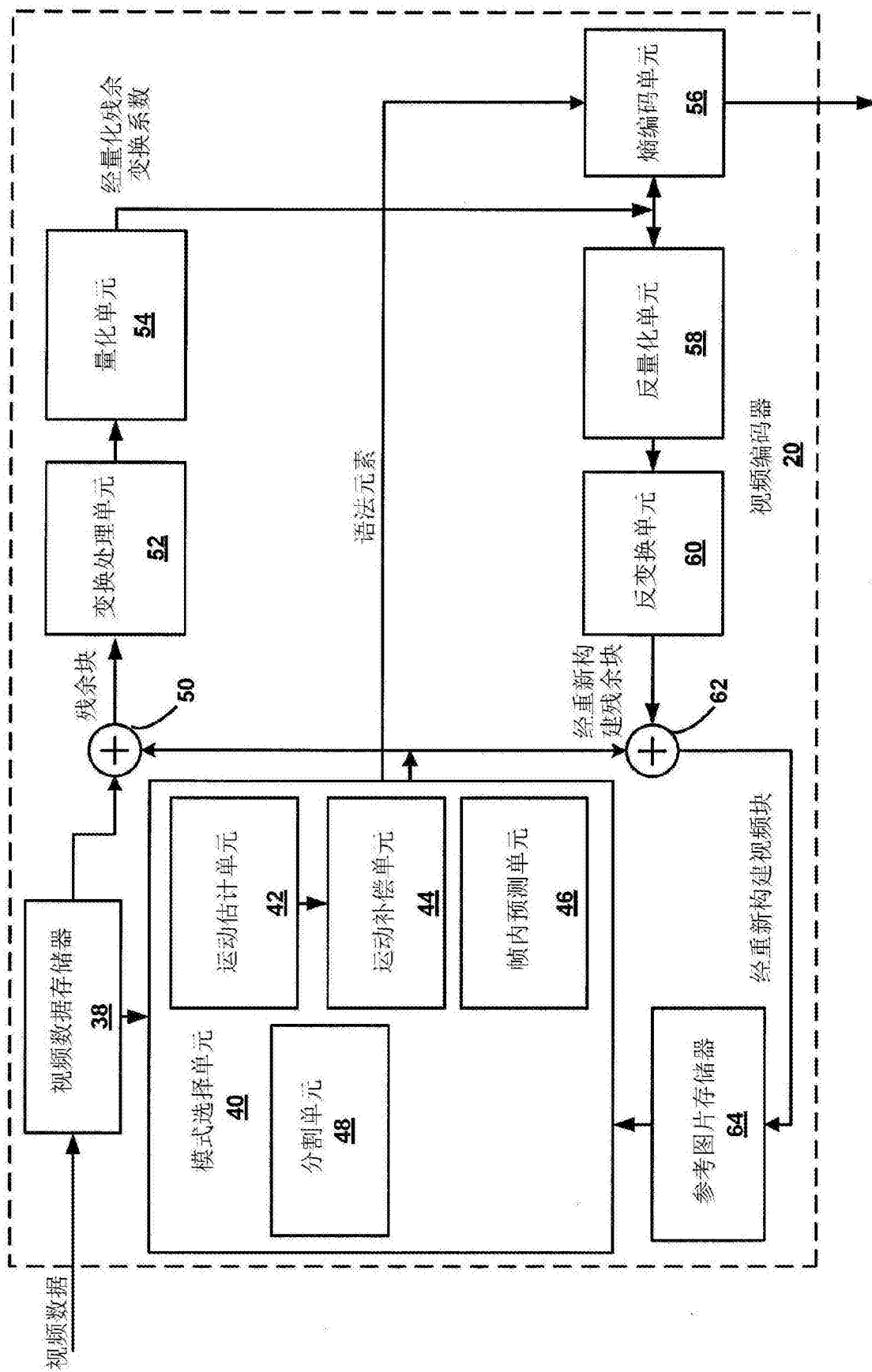


图2

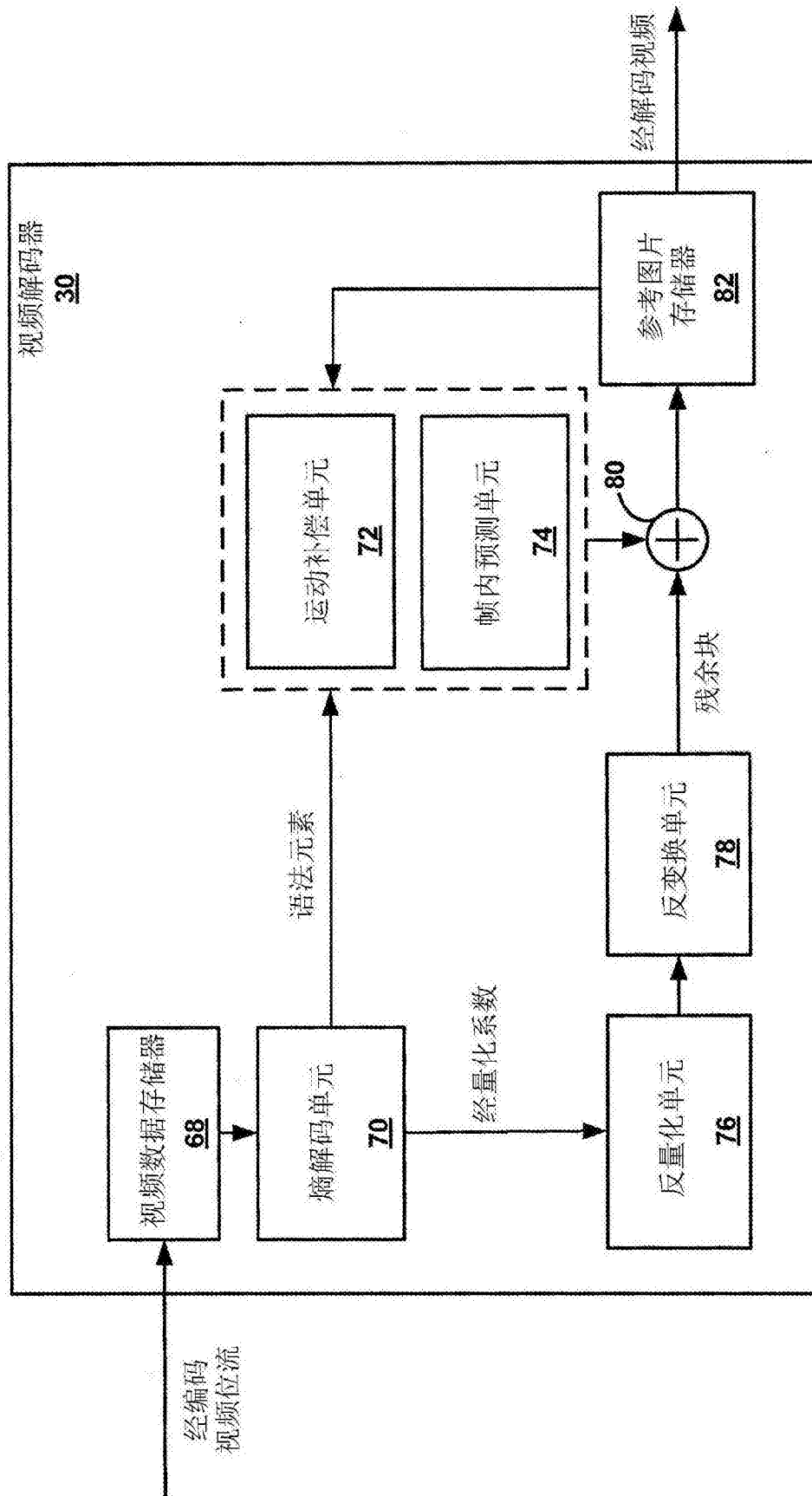


图3

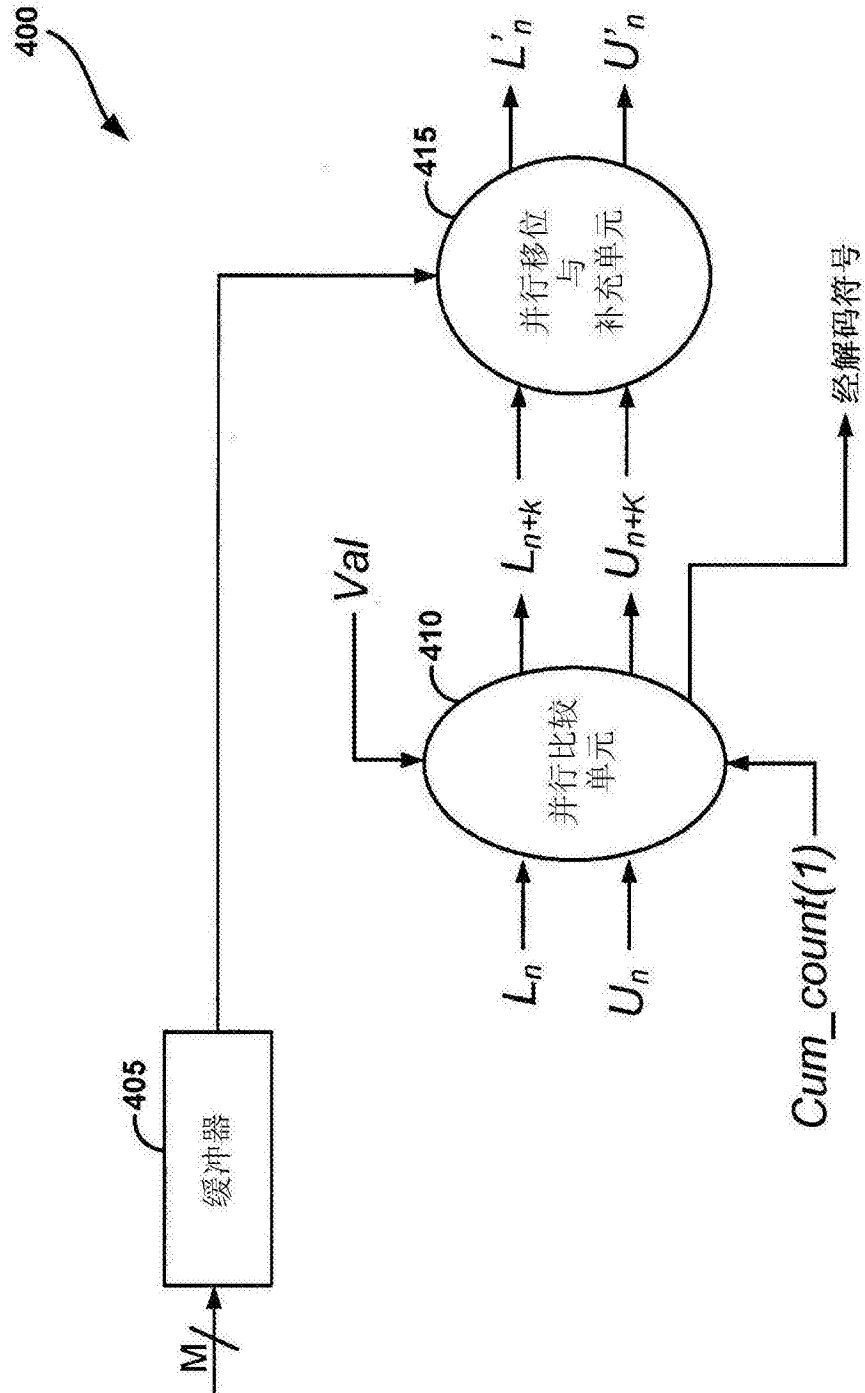


图4

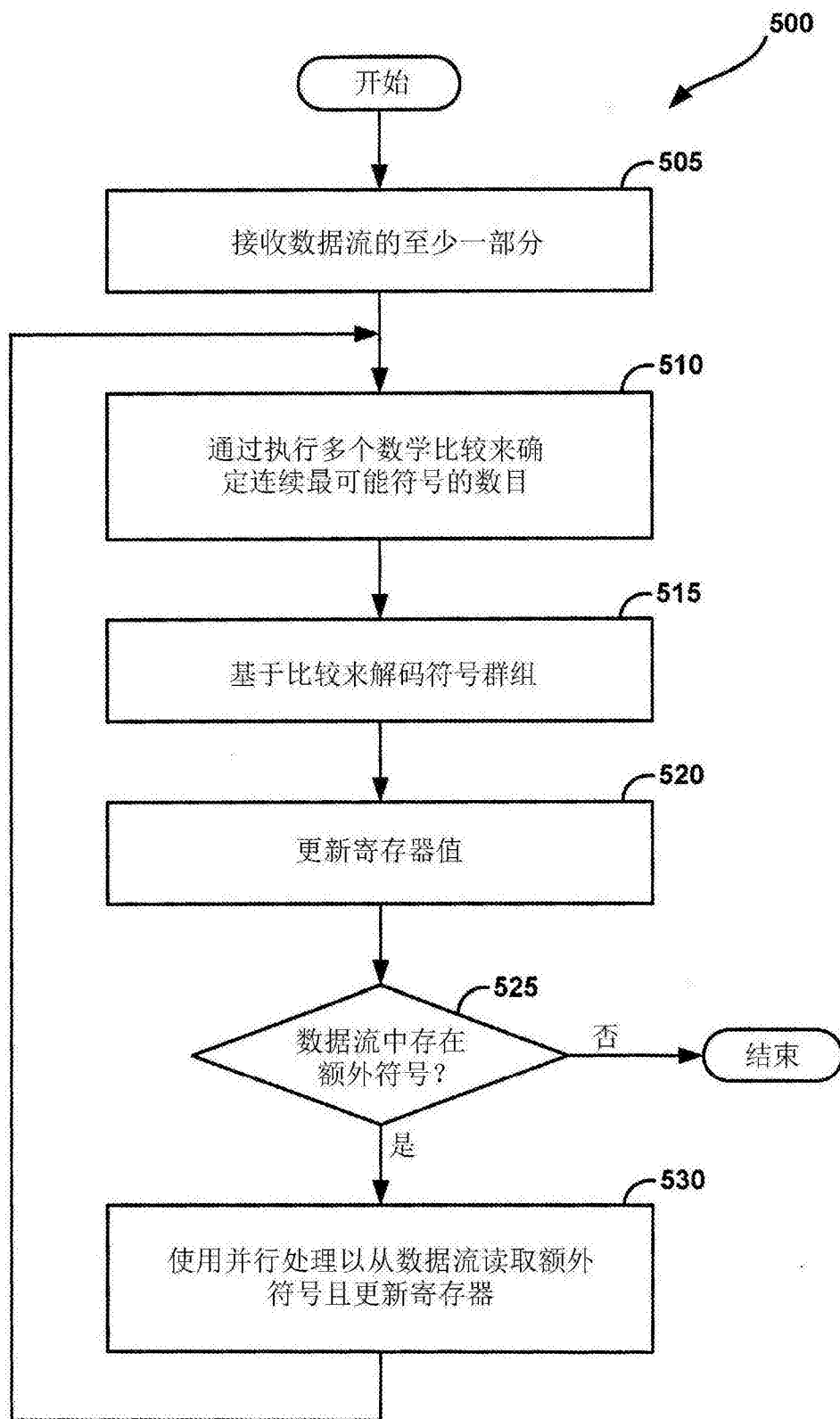


图5