



(12)发明专利

(10)授权公告号 CN 104041038 B

(45)授权公告日 2018.03.27

(21)申请号 201280066211.4

(22)申请日 2012.11.06

(65)同一申请的已公布的文献号

申请公布号 CN 104041038 A

(43)申请公布日 2014.09.10

(30)优先权数据

1160109 2011.11.07 FR

(85)PCT国际申请进入国家阶段日

2014.07.07

(86)PCT国际申请的申请数据

PCT/FR2012/052552 2012.11.06

(87)PCT国际申请的公布数据

W02013/068684 FR 2013.05.16

(73)专利权人 杜比国际公司

地址 荷兰阿姆斯特丹

(72)发明人 F.亨利 G.克莱尔

(74)专利代理机构 中国国际贸易促进委员会专

利商标事务所 11038

代理人 宿小猛

(51)Int.Cl.

H04N 19/46(2014.01)

H04N 19/463(2014.01)

H04N 19/467(2014.01)

H04N 19/176(2014.01)

H04N 19/147(2014.01)

(56)对比文件

CN 1878311 A, 2006.12.13,

US 2008246637 A1, 2008.10.09,

审查员 张琰

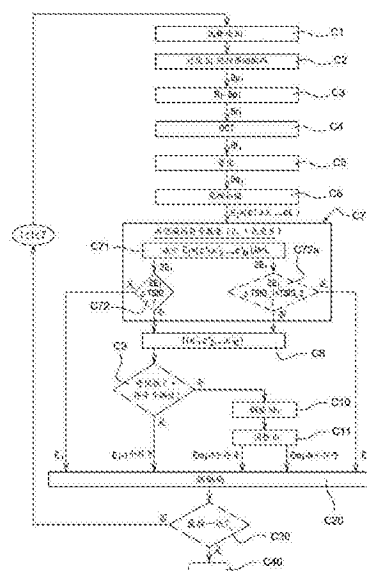
权利要求书2页 说明书13页 附图4页

(54)发明名称

用于编码和解码图像的方法、编码和解码设备

(57)摘要

本发明涉及一种用于对被划分为多个分区的至少一个图像进行编码的方法,所述方法实现下列步骤:基于至少一个已经被编码然后被解码的参考分区来预测(C2)当前分区的数据,生成已预测分区;通过将当前分区相关的数据与已预测分区进行比较来确定(C6)残留数据的集合,所述残留数据分别与要经历熵编码的各个数字数据项关联。在用于产生包含所述编码信息的信号的步骤之前,实现下列步骤:根据所述残留数据的所述预定集合来确定(C71)包含能被调整的残留数据的子集;计算(C8)表示所述预定子集的残留数据的函数的值;将所述已计算值与至少一个所述数字数据项的值进行比较(C9);基于所述比较的结果调整(C10)或不调整所述子集中的至少一个残留数据项;以及在调整的情况下,对所述至少一个已调整的残留数据项进行熵编码(C20)。



1. 一种计算机实现的用于对图像的符号数据隐藏允许分区进行解码的方法,包括:
接收来自编码器的一组经基于上下文自适应二进制算术编码CABAC编码的系数;
解码该组经CABAC编码的系数以生成代表符号数据隐藏允许分区的残留块的一组系数,所述一组系数包括未指定符号的第一非零系数;
将函数应用于所述一组系数以生成奇偶校验数据,其中,将函数应用于所述一组系数包括:
计算所述一组系数中的非零系数之和;以及
使用非零系数之和计算奇偶校验数据;以及
基于所述奇偶校验数据为所述第一非零系数指定符号。
2. 根据权利要求1所述的方法,其中,计算所述一组系数中的非零系数之和包括计算所述一组系数中的非零系数的绝对值之和。
3. 根据权利要求1所述的方法,其中,为所述第一非零系数指定符号包括:
确定所述奇偶校验数据的值等于被分配给正符号的特定值;以及
响应于确定所述奇偶校验数据的值等于被分配给正符号的特定值,为所述第一非零系数指定正符号。
4. 根据权利要求1所述的方法,其中,为所述第一非零系数指定符号包括:
确定所述奇偶校验数据的值等于被分配给负符号的特定值;以及
响应于确定所述奇偶校验数据的值等于被分配给负符号的特定值,为所述第一非零系数指定负符号。
5. 根据权利要求1所述的方法,包括:
确定所述一组系数中的非零系数的计数满足阈值;以及
响应于确定所述一组系数中的非零系数的计数满足阈值,基于该阈值确定未指定符号的非零系数的数量。
6. 根据权利要求1所述的方法,其中,确定对于所述一组系数允许符号数据隐藏编码。
7. 一种用于对图像的符号数据隐藏允许分区进行解码的解码器,包括:
一个或多个处理器;以及
耦接到所述一个或多个处理器的计算机可读介质,在所述计算机可读介质上存储指令,所述指令在由所述一个或多个处理器执行时使得所述一个或多个处理器执行包括以下的操作:
接收来自编码器的一组经基于上下文自适应二进制算术编码CABAC编码的系数;
解码该组经CABAC编码的系数以生成代表符号数据隐藏允许分区的残留块的一组系数,所述一组系数包括未指定符号的第一非零系数;
将函数应用于所述一组系数以生成奇偶校验数据,其中,将函数应用于所述一组系数包括:
计算所述一组系数中的非零系数之和;以及
使用非零系数之和计算奇偶校验数据;以及
基于所述奇偶校验数据为所述第一非零系数指定符号。
8. 根据权利要求7所述的解码器,其中,计算所述一组系数中的非零系数之和包括计算所述一组系数中的非零系数的绝对值之和。

9. 根据权利要求7所述的解码器, 其中, 为所述第一非零系数指定符号包括:
确定所述奇偶校验数据的值等于被分配给正符号的特定值; 以及
响应于确定所述奇偶校验数据的值等于被分配给正符号的特定值, 为所述第一非零系数指定正符号。
10. 根据权利要求7所述的解码器, 其中, 为所述第一非零系数指定符号包括:
确定所述奇偶校验数据的值等于被分配给负符号的特定值; 以及
响应于确定所述奇偶校验数据的值等于被分配给负符号的特定值, 为所述第一非零系数指定负符号。
11. 根据权利要求7所述的解码器, 其中, 确定对于所述一组系数允许符号数据隐藏编码。
12. 一种用于对图像的符号数据隐藏允许分区进行解码的装置, 包括:
用于接收来自编码器的一组经基于上下文自适应二进制算术编码CABAC编码的系数的部件;
用于解码该组经CABAC编码的系数以生成代表符号数据隐藏允许分区的残留块的一组系数的部件, 所述一组系数包括未指定符号的第一非零系数;
用于将函数应用于所述一组系数以生成奇偶校验数据的部件, 其中, 将函数应用于所述一组系数包括:
计算所述一组系数中的非零系数之和; 以及
使用非零系数之和计算奇偶校验数据; 以及
用于基于所述奇偶校验数据为所述第一非零系数指定符号的部件。
13. 根据权利要求12所述的装置, 其中, 计算所述一组系数中的非零系数之和包括计算所述一组系数中的非零系数的绝对值之和。
14. 根据权利要求12所述的装置, 其中, 为所述第一非零系数指定符号包括:
确定所述奇偶校验数据的值等于被分配给正符号的特定值; 以及
响应于确定所述奇偶校验数据的值等于被分配给正符号的特定值, 为所述第一非零系数指定正符号。
15. 根据权利要求12所述的装置, 其中, 为所述第一非零系数指定符号包括:
确定所述奇偶校验数据的值等于被分配给负符号的特定值; 以及
响应于确定所述奇偶校验数据的值等于被分配给负符号的特定值, 为所述第一非零系数指定负符号。

用于编码和解码图像的方法、编码和解码设备

技术领域

[0001] 本发明主要涉及图像处理的领域,且更具体地涉及数字图像和数字图像的序列的编码和解码。

[0002] 由此,本发明尤其可用于由目前的视频编码器(MPEG、H.264等)或未来的视频编码器(ITU-T/VCEG(H.265)或ISO/MPEG(HEVC))实现的视频编码。

背景技术

[0003] 目前的视频编码器(MPEG、H.264等)等使用视频序列的分块表示。图像被分为宏块,每个宏块本身被分为多个块,并且每个块通过图像内或图像间预测来编码。于是,某些图像通过空间预测(帧内预测)来编码,而其它图像借助于本领域技术人员已知的运动补偿通过相对于一个或多个编码-解码参考图像的时间预测(帧间预测)进行编码。

[0004] 对于每个块,存在与由预测所减掉的原始块相对应的、还被称为预测残留的已编码的残留块。残留块通过离散余弦变换(DCT)类型的变换来变换,然后使用例如标量类型的量化来量化。在量化步骤结束时得到系数,所述系数中一些是正的而其它的是负的。然后以通常为之字形的读顺序(和JPEG标准一样)来扫描系数,由此可利用高频中的大量零系数。在上述扫描结束时,得到系数的一维列表,其可被称为“量化残留”。然后通过熵编码来对该列表的系数进行编码。

[0005] 熵编码(例如算术编码或霍夫曼编码类型)以如下方式执行:

[0006] -信息项被熵编码,以表示列表中的最后的非零系数的位置,

[0007] -针对位于最后的非零系数之前的每个系数,信息项被熵编码以表示系数是否为零,

[0008] -针对之前所表示的每个非零系数,信息项被熵编码以表示系数是否等于1,

[0009] -针对不为零且不等于1的并且位于最后的非零系数之前的每个系数,幅度信息项(系数的绝对值,其值被减少2)被熵编码,

[0010] -针对每个非零系数,分配给该系数的符号由“0”(用于+符号)或“1”(用于-符号)编码。

[0011] 根据例如H.264技术,在宏块被分为多个块时,与每个块对应的数据信号被发送给解码器。该信号包括:

[0012] -在上述列表中包含的量化残留,

[0013] -表示所使用的编码模式的信息项,特别是:

[0014] • 预测模式(帧内预测、帧间预测、默认预测,其进行没有信息项被发送到解码器的预测(被称为“跳过”));

[0015] • 指定预测类型的信息(方向、参考图像等);

[0016] • 分区类型;

[0017] • 变换类型,例如 4×4 DCT、 8×8 DCT等;

[0018] • 运动信息,如果有必要的话;

[0019] • 等等。

[0020] 逐个图像以及针对每个图像逐个宏块地执行解码。针对宏块的每个分区,读取流的相应元素。执行块的系数的反量化和逆变换,以生成已解码的预测残留。然后,计算该分区的预测并通过将预测加到被解码的预测残留来重构该分区。

[0021] 由此,如H.264标准所实现的,通过竞争实现的帧内编码或帧间编码依赖于将例如上述那些编码信息项之类的各个编码信息项与选择最佳模式的目标进行竞争,该最佳模式即能根据预定的性能标准例如本领域技术人员众所周知的比率失真成本来使所讨论分区的编码最优化。

[0022] 表示被选中编码模式的信息项被包含在由编码器发送到解码器的数据信号中。解码器由此能够识别在编码器上被选中的编码模式,然后能够根据该模式应用预测。

[0023] 文档“Data Hiding of Motion Information in Chroma and Luma Samples for Video Compression”,J.-M.Thiesse,J.Jung and M.Antonini,International workshop on multimedia signal processing,2011介绍了一种在视频压缩期间实现的数据隐藏的方法。

[0024] 更具体而言,在至少一个竞争指数从要被发送的多个竞争指数中发出时,建议避免在要被发送到解码器的信号中包含所述至少一个竞争指数。这样的指数例如是指数MVComp,其表示用于标识运动矢量预测器的信息项,该预测器用于在帧间预测模式下预测的块。能够取值0或1的这样的指数未被直接包含在已编码的数据项的信号中,而是通过量化残留的系数之和的奇偶校验来传输。在量化残留的奇偶性和指数MVComp之间建立关联。通过实例,量化残留的偶数值与值为0的指数MVComp关联,而量化残留的奇数值与值为1的指数MVComp关联。会出现两种情形。在第一种情形中,如果量化残留的奇偶性已经对应于所期望发送的指数MVComp的奇偶性,则以传统方式对量化残留编码。在第二种情形中,如果量化残留的奇偶性与所期望发送的指数MVComp的奇偶性不同,则调整量化残留从而使其奇偶性与指数MVComp的奇偶性相同。该调整包含将量化残留的一个或多个系数增加或减少奇数值(例如+1、-1、+3、-3、+5、-5等),并且仅保留使预定标准最优化的调整,在该情形中,所述预定标准为前述比率失真成本。

[0025] 在解码器上,不从信号读取指数MVComp。通过传统地确定残留来简单地满足解码器。如果该残留的值是偶数,则指数MVComp被设置为0。如果该残留的值是奇数,指数MVComp被设置为1。

[0026] 根据刚才介绍的技术,不会总是以最优的方式选择已调整的系数,从而所应用的调整给要发送到解码器的信号带来了干扰。这样的干扰对于视频压缩的效率必然有害。

发明内容

[0027] 本发明的一个目标是纠正上述现有技术中的缺陷。

[0028] 为此,本发明的目标涉及一种用于对被划分为多个分区的至少一个图像进行编码的方法,该方法实现下列步骤:

[0029] -根据至少一个已经被编码然后被解码的参考分区来预测当前分区的数据项,生成已预测分区;

[0030] -通过将所述当前分区相关的数据项与所述已预测分区进行比较来确定残留数

据项的集合,所述残留数据项分别与要经历熵编码的各个数字信息项关联。

[0031] -生成包含所述已编码的信息项的信号。

[0032] 根据本发明的方法值得注意的是,它在所述信号生成步骤之前实现下列步骤:

[0033] -根据所述已确定的残留数据项的集合来确定包含适于被调整的残留数据项的子集,

[0034] -计算表示所述已确定子集的残留数据项的函数的值,

[0035] -将所述已计算的值与至少一个数字信息项的值进行比较,

[0036] -取决于比较的结果,调整或不调整所述子集中的至少一个残留数据项,

[0037] -在调整的情况下,对所述至少一个已调整的残留数据项进行熵编码。

[0038] 这样的布置能在残留数据项的缩减集合上应用数据隐藏技术,在该集合中残留数据项适于被调整。

[0039] 根据本发明,表达“残留数据项适合被调整”被理解为表示对其应用调整不会带来编码器和解码器之间的去同步化的数据项。

[0040] 于是,根据本发明,可以比前述现有技术更可靠地选择要经历调整的残留数据项,由此在解码器上可获得图像重构的更好质量。

[0041] 此外,对减少数量的残留数据项进行调整可能性能加速编码。

[0042] 在特定的实施例中,只有在满足依赖于适于被调整的所述残留数据项的预定标准时,才会执行在所述用于确定所述残留数据项的所述子集的步骤之后的所述步骤。

[0043] 这样的布置还使得编码器能以合理的方式来判断它是否能应用数据隐藏技术。这样的判断步骤具有仅应用于适合被调整的残留数据项的缩减集合的优势。由此确保了能比之前引用的现有技术更适合地应用数据隐藏技术,特别是针对更好选择的残留数据项的数量,就此确定一旦这些数据项被调整,则这样的调整所产生的信号干扰不会对解码器上的图像重构的质量带来负面影响。

[0044] 在另一特定实施例中,预定的判断标准依赖于适合被调整的残留数据项的数量和预定数量之间的比较结果。

[0045] 这样的布置能改善算术编码器的压缩性能同时有效降低了信令成本。特别地,该布置能准确地检测残留数据项的数量,由此合理地应用数据隐藏技术,而该数据隐藏技术不会引起要发送到解码器的信号的高干扰级别。

[0046] 在又一特定实施例中,如果在比较步骤中考虑多个数字信息项,则该比较步骤需要将表示确定子集的残留数据项的函数所计算的值与表示多个数字信息项的函数的值进行比较。

[0047] 该布置能优化算术编码器的压缩性能,同时优化信令成本的降低,因为它能在要发送到解码器的信号中隐藏若干个数字信息项。

[0048] 在又一特定实施例中,至少一个数字信息项对应于残留数据项的符号。

[0049] 该符号是要隐藏的特别相关的信息项,因为正或负符号出现的概率是相同的。这样,假设需要用一比特来对符号编码,则由此可能通过隐藏该信息来精简要发送到解码器的信号中的一个比特,由此显著降低信令成本。当根据本发明可隐藏多个符号且由此隐藏多个比特时,该成本的降低将会更多。

[0050] 相关联地,本发明还涉及一种用于对被划分为多个分区的至少一个图像进行编码

的设备,该设备包括:

[0051] -根据至少一个已经被编码然后被解码的参考分区来预测当前分区的数据项生成已预测分区的装置,

[0052] -确定适于将与当前分区相关的数据项与已预测分区进行比较的残留数据项的集合的装置,该残留数据项分别与要经历熵编码的各个数字信息项相关联,

[0053] -用于生成包含所述已编码的信息项的信号的装置。

[0054] 这样的编码设备值得注意的是,它在所述生成装置之前包括处理装置,该处理装置能够:

[0055] -根据所述已确定的残留数据项的集合来确定包含适于被调整的残留数据项的子集,

[0056] -计算表示所述已确定子集的残留数据项的函数的值,

[0057] -将所述已计算的值与至少一个所述数字信息项的值进行比较,

[0058] -取决于比较的结果,调整或不调整所述已确定子集中的至少一个残留数据项,

[0059] 以及在通过所述处理装置来调整的情况下,对所述至少一个已调整的残留数据项进行熵编码的装置。

[0060] 相应地,本发明涉及一种用于对表示之前已被编码的划分为多个分区的至少一个图像的数据信号进行解码的方法,包括通过所述信号的数据项的熵解码来获取与和至少一个之前已被编码的分区相关的残留数据项关联的数字信息项的步骤,

[0061] 该解码方法值得注意的是,它包括下列步骤:

[0062] -根据所述残留数据项来确定包含在之前的编码期间能被调整的残留数据项的子集,

[0063] -计算表示所述已确定子集的残留数据项的函数的值,

[0064] -根据所述已计算的值来获取与通过熵解码而获得的那些值所不同的至少一个数字信息项的值。

[0065] 在特定的实施例中,只有在满足依赖于能被调整的残留数据项的预定标准时,才会实现在所述用于确定残留数据项的子集的步骤之后的所述步骤。

[0066] 在另一特定的实施例中,预定判断标准依赖于能被调整的残留数据项的数量和预定数量之间的比较结果。

[0067] 在又一特定实施例中,根据所述已计算的值来获取与多个数字信息项分别关联的多个值,所述多个数字信息项与通过熵解码所获取的数字信息项不同。

[0068] 在又一特定实施例中,至少一个数字信息项对应于残留数据项的符号。

[0069] 相关地,本发明还涉及一种用于对表示之前已被编码的划分为多个分区的至少一个图像的数据信号进行解码的装置,包括用于通过所述信号的数据项的熵解码来获取与和至少一个之前被编码的分区相关的残留数据项关联的数字信息项的装置,

[0070] 该解码设备值得注意的是,它包括处理装置,其能够:

[0071] -根据所述残留数据项来确定包含在之前的编码期间能被调整的残留数据项的子集,

[0072] -计算表示所述已确定的子集的残留数据项的函数的值,

[0073] -根据所述已计算值来获取与通过熵解码所获取的那些数字信息项所不同的至少

一个数字信息项的值。

[0074] 本发明还涉及一种包含指令的计算机程序,在计算机上执行该程序时,该指令用于执行以上编码或解码方法的步骤。

[0075] 该程序可以使用任何编程语言并且可以是源代码、目标代码或介于源代码与目标代码之间的代码的形式,例如部分编译的形式或任何其它期望的形式。

[0076] 本发明的又一主题是记录介质,其可被计算机读取并包含如上所述的计算程序的指令。

[0077] 该记录介质可以是能够存储程序的任意实体或设备。例如,该介质可以包括诸如ROM之类的存储介质,例如CD-ROM或微电子电路ROM,或者 磁记录介质,例如软盘或硬盘。

[0078] 另一方面,该记录介质可以是诸如电信号或光信号之类的可传输介质,其可以经电缆或光缆通过无线电或通过其它方式来传递。根据本发明的程序可以特别地通过因特网类型的网络来下载。

[0079] 或者,该记录介质可以是其中包含有程序的集成电路,该电路适于执行所讨论的方法或适于在该方法执行时使用。

[0080] 上述编码设备、解码方法、解码设备和计算机程序至少表现出与根据本发明的编码方法所提供的优势相同的优势。

附图说明

[0081] 通过阅读参考附图来描述两个优选的实施例,其它特征和优点将变得明显,在附图中:

[0082] -图1表示根据本发明的编码方法的主要步骤,

[0083] -图2表示根据本发明的编码设备的实施例,

[0084] -图3表示根据本发明的解码方法的主要步骤,

[0085] -图4表示根据本发明的解码设备的实施例。

具体实施方式

[0086] 现在将描述本发明的实施例,其中,根据本发明的编码方法被用于根据二进制流来对图像序列进行编码,该二进制流与通过根据H.264/MPEG-4AVC标准来编码而获得的二进制流接近。在该实施例中,根据本发明的编码方法例如以软件或硬件形式通过对初始符合H.264/MPEG-4AVC标准的编码器的调整来实现。根据本发明的编码方法以图1所示的包含步骤C1到C40的算法的形式来表示。

[0087] 根据本发明的实施例,根据本发明的编码方法在图2所示的实施例中的编码设备或编码器C0中实现。

[0088] 根据本发明,如图2所示,在实际的编码步骤之前,以预定顺序将待编码的图像序列中的图像IE划分为多个(Z个)分区 B_1 、 B_2 、 $\dots$ 、 B_i 、 $\dots$ 、 B_z 。

[0089] 值得注意的是,在本发明的意义上,术语“分区”表示编码单元。该后一术语特别用于正在起草的HEVC/H.265标准,例如用在下列互联网地址可 访问的文档中:

[0090] http://phenix.int-evry.fr/jct/doc_end_user/current_document.php?id=3286

[0091] 特别地,这种编码单元将矩形或正方形的像素集合或表现出其它几何形状的像素集合分组到一起,所述矩形或正方形的像素集合被称为块、宏块。

[0092] 在图2所示的实例中,所示分区是具有正方形形状并且全部具有相同尺寸的块。根据图像的尺寸,该尺寸不一定是块尺寸的倍数,最左边的块以及最底部的块可以不是正方形。在替代的实施例中,块例如可以是矩形尺寸和/或可以不互相对齐。

[0093] 每个块或每个宏块本身可以进一步被分为子块,子块自己可以再细分。

[0094] 这样的划分是通过图2所示的分区模块PC0来执行的,该分区模块PC0使用例如像这样的分区模块所众所周知的分区算法。

[0095] 在所述划分步骤之后,对所述图像IE的每个当前分区 B_i (其中 i 是大于等于1且小于等于 Z 的整数)进行编码。

[0096] 在图2所示的实例中,对当前图像IE的块 B_1 到 B_Z 中的每个块相继实施这样的编码。例如根据诸如本领域技术人员众所周知的光栅扫描之类的扫描来对块进行编码。

[0097] 根据本发明的编码在如图2所示的编码器C0的编码软件模块MC_C0中实现。

[0098] 在图1所示的步骤C1中,图2的编码模块MC_C0选择当前图像IE的要被编码的第一块 B_1 作为当前块 B_i 。如图2所示,第一块 B_1 是图像IE左边的第一块。

[0099] 在图1所示的步骤C2中,通过已知的帧内预测技术和/或帧间预测技术来实现当前块 B_1 的预测编码,在此期间针对至少一个之前被编码和解码的块来对块 B_1 的预测编码进行预测。该预测通过图2所示的预测软件模块PRED_C0来实现。

[0100] 毋庸赘述,H.264标准中所建议的其它帧内预测模式是可能的。

[0101] 当前块 B_1 还可经过帧间模式的预测编码,在此期间针对来自之前被编码和解码的图像的块来预测当前块。当然可以想到其它类型的预测。在可能用于当前块的预测中,根据本领域众所周知的比率失真标准来选择最优预测。

[0102] 上述预测编码步骤对与当前块 B_1 近似的已预测的块 B_{p1} 进行构造。与该预测编码相关的信息项将被包含在要被发送到解码器的信号中。该信息项尤其包括预测的类型(帧内预测或帧间预测),并且如果必要的话,包括帧内预测模式、块或宏块(如果宏块已被细分)的分区类型、在帧间预测模式中使用的参考图像索引和位移矢量。这些信息项被编码器C0压缩。

[0103] 在图1所示的下一步骤C3中,预测模块PRED_C0将与当前块 B_1 相关的数据项与已预测的块 B_{p1} 的数据项进行比较。更具体而言,在该步骤中,传统上从当前块 B_1 减去已预测的块 B_{p1} 以产生残留块 B_{r1} 。

[0104] 在图1所示的下一步骤C4中,根据传统的直接变换操作例如DCT型离散余弦变换来变换残留块 B_{r1} ,以产生已变换的块 B_{t1} 。该操作由如图2所示的变换软件模块MT_C0来执行。

[0105] 在图1所示的下一步骤C5中,根据传统的量化操作例如标量量化来量化已变换的块 B_{t1} 。然后得到已量化的系数的块 B_{q1} 。该步骤通过如图2所示的量化软件模块MQ_C0来执行。

[0106] 在图1所示的下一步骤C6中,以预定顺序来扫描块 B_{q1} 的已量化的系数。在所示实例中,这是传统的之字形扫描。该步骤由如图2所示的读取软件模块ML_C0来执行。在步骤C6结束时,得到系数的一维列表 $E_1 = (\epsilon_1, \epsilon_2, \dots, \epsilon_L)$,更一般地被称为“量化残留”,其中 L 是大于或等于1的整数。列表 E_1 中的每个系数与要经历熵编码的不同数字信息项关联。下面通过

实例来描述这样的数字信息项。

[0107] 假设在所示实例中, $L=16$ 并且列表 E_1 包括下列 16 个系数: $E_1 = (0, +9, -7, 0, 0, +1, 0, -1, +2, 0, 0, +1, 0, 0, 0, 0)$ 。

[0108] 在该特定情形下:

[0109] -针对位于列表 E_1 中的最后的非零系数之前的每个系数, 数字信息项例如比特将被熵编码, 以表示系数是否为零。如果系数是零, 则它例如是将被编码的值为 0 的比特, 而如果系数不是零, 则它是将被编码的值为 1 的比特。

[0110] -针对每个非零系数 $+9, -7, +1, -1, +2, +1$, 数字信息项例如比特要被熵编码, 以表示系数的绝对值是否等于 1: 如果它等于 1, 则它例如是将被编码的值为 1 的比特, 而如果它不等于 1, 则它是将被编码的值为 0 的比特。

[0111] -针对每个非零系数, 并且其绝对值不等于 1 且位于最后的非零系数之前, 例如值为 $+9, -7, +2$ 的系数, 幅度信息项 (其系数的绝对值减去了值 2) 被熵编码;

[0112] -针对每个非零系数, 分配给它的符号被数字信息项编码, 所述数字信息项例如被设置为“0”的比特 (用于+符号) 或被设置为“1”的比特 (用于-符号)。

[0113] 参考图 1, 现在将描述根据本发明的特定编码步骤。

[0114] 根据本发明, 决定避免对至少一个上述信息项进行熵编码。由于之前描述中解释的原因, 在优选实施例中, 决定不对列表 E_1 中的所述系数中的一个的至少一个符号进行熵编码。

[0115] 通过替代的实例, 尤其可以确定对所述列表 E_1 中的第一个非零系数的幅度的二进制表示的最低有效比特进行熵编码。

[0116] 为此, 在图 1 所示的步骤 C7 中, 选择在之后的熵编码步骤中隐藏的符号的数量。该步骤由图 2 所示的处理软件模块 MTR_CO 来执行。

[0117] 在优选实施例中, 要隐藏的符号的数量是 1 个或 0 个。额外地, 根据所述优选实施例, 要被隐藏的是第一个非零系数的符号。在所示实例中, 因此隐藏的是系数 $\epsilon_2 = +9$ 的符号。

[0118] 在替代的实施例中, 要隐藏的符号的数量是 0、1、2、3 或更多。

[0119] 根据步骤 C7 的优选实施例, 在图 1 所示的第一子步骤 C71 中, 根据所述列表 E_1 来确定包含适合被调整的系数的子列表 $SE_1, \epsilon'_1, \epsilon'_2, \dots, \epsilon'_M$, 其中 $M < L$ 。在以下描述中该系数将被称为可调整系数。

[0120] 根据本发明, 如果其量化值的调整不会引起解码器的去同步化, 一旦该调整的系数被解码器处理, 则系数是可调整的。于是, 处理模块 MTR_CO 被初始配置为不调整:

[0121] -位于第一非零系数之前的零系数或多个零系数, 从而解码器不会影响在所述零系数或所述多个零系数处隐藏的符号的值,

[0122] -并且由于计算复杂度的原因, 位于最后的非零系数之后的零系数或多个零系数。

[0123] 在所实例中, 在子步骤 C7 结束时, 得到的子列表 SE_1 由此是 $SE_1 = (9, -7, 0, 0, 1, 0, -1, 2, 0, 0, 1)$ 。因此, 得到 11 个可调整的系数。

[0124] 在图 1 所示的下一子步骤 C72 中, 处理模块 MTR_CO 将可调整的系数的数量与预定阈值 TSIG 进行比较。在优选实施例中, TSIG 具有值 4。

[0125] 如果可调整系数的数量小于阈值 TSIG, 则在图 1 所示的步骤 C20 中, 对列表 E_1 中的

系数进行传统熵编码,例如由在图2中的标记CE_CO所表示的CABAC编码器来执行。为此,列表E₁中的每个非零系数的符号被熵编码。

[0126] 如果可调整系数的数量大于阈值TSIG,则在图1所示的步骤C8中,处理模块MTR_CO计算函数f的值,其表示子列表SE₁中的系数。

[0127] 在要发送到解码器的信号中仅要隐藏一个符号的优选实施例中,函数f是子列表SE₁中的系数之和的奇偶校验。

[0128] 在图1所示的步骤C9中,根据在编码器CO中事先定义的规则,处理模块MTR_CO检查要隐藏的符号的值的奇偶性是否对应于子列表SE₁中的系数之和的奇偶性。

[0129] 在所建议实例中,所述规则是:正符号与值等于零的比特相关联,而负符号与值等于1的比特相关联。

[0130] 根据在根据本发明的编码器CO中采用的规则,如果符号为正,其对应于值为零的编码比特,并且如果子列表SE₁中的系数之和是偶数,则对上述列表E₁中除了系数ε₂的符号以外的系数进行熵编码。

[0131] 仍然根据在根据本发明的编码器CO中采用的规则,如果符号为负,其对应于值为1的编码比特,并且如果子列表SE₁中的系数之和为奇数,则也对上述列表E₁中除了系数ε₂的符号以外的系数进行熵编码的步骤C20。

[0132] 根据在根据本发明的编码器CO中采用的规则,如果符号为正,其对应于值为零的编码比特,并且如果子列表SE₁的系数之和是奇数,则在图1所示的步骤C10中,子列表SE₁中的至少一个可调整的系数被调整。

[0133] 仍然根据在根据本发明的编码器CO中采用的规则,如果符号为负,其对应于值为1的编码比特,并且如果子列表SE₁中的系数之和是奇数,则也在步骤C10中,子列表SE₁中的至少一个可调整的系数被调整。

[0134] 该调整操作由图2中的处理模块MTR_CO执行。

[0135] 在其中SE₁ = (+9, -7, 0, 0, +1, 0, -1, +2, 0, 0, +1)的示例性实施例中,系数的总和f等于5且因此是奇数。为了使解码器能够重构被分配给第一非零系数ε₂ = +9的正符号而不用编码器CO将该系数发送到解码器,必须将总和的奇偶性变成偶数。因此,在所述步骤C10中,处理模块MTR_CO测试子列表SE₁中的系数的各种调整,所有的目标都是改变系数之和的奇偶性。在优选实施例中,+1或-1被加到每个可调整的系数并从被执行的那些调整中选择调整。

[0136] 在优选实施例中,这样的选择形成根据性能标准的最优预测,该性能标准例如是本领域技术人员众所周知的比率失真标准。该标准由下列等式(1)来表示:

[0137] (1) $J = D + \lambda R$

[0138] 其中D表示原始宏块和重构宏块之间的失真,R表示编码信息项的比特的编码成本,R表示编码信息项的比特的编码成本,并且λ表示拉格朗日倍数,其值在编码之前被固定。

[0139] 在所建议的实例中,根据上述比率失真标准带来最优预测的调整是将值1加到子列表SE₁中的第二系数-7。

[0140] 在步骤C10结束时,因此得到已调整的子列表SE_{m1} = (+9, -6, 0, 0, +1, 0, -1, +2, 0, 0, +1)。

[0141] 值得注意的是,在该步骤中,特定的调整是被禁止的。于是,在第一非零系数 ϵ_2 具有值+1的情形中,不可能将该系数加上-1,因为它将变成零,并且然后该系数会丢失列表 E_1 中的第一非零系数的特征。解码器然后将已解码的符号(通过对系数之和的奇偶性进行计算)归属于另一系数,且然后将会出现解码错误。

[0142] 在图1所示的步骤C11中,处理模块MTR_CO对列表 E_1 中进行相应的调整。然后得到下一已调整的列表 $Em_1 = (0, +9, -6, 0, 0, +1, 0, -1, +2, 0, 0, +1, 0, 0, 0, 0)$ 。

[0143] 然后进行对上述列表 Em_1 中除了系数 ϵ_2 的符号以外的系数的熵编码的步骤C20,在所建议的例子中系数9为+符号,该符号被隐藏在系数之和的奇偶校验中。

[0144] 值得注意的是,列表 E_1 或已调整的列表 Em_1 中的系数的幅度的集合在除了如上解释的未被编码的第一非零系数 ϵ_2 的符号以外的符号的集合之前被编码。

[0145] 在图1所示的下一步骤C30中,图2中的编码模块MC_CO测试已编码的当前块是否是图像IE中的最后一块。

[0146] 如果当前块是图像IE中的最后一块,则在图1所示的步骤C40中,编码方法结束。

[0147] 如果不是该情形下,选择下一块 B_i ,然后根据前述光栅扫描的顺序通过重复步骤C1到C20来对 $1 \leq i \leq Z$ 的块 B_i 进行编码。

[0148] 一旦实现所有块 B_1 到 B_Z 的熵编码,则构造以二进制形式来表示所述已编码的块的信号F。

[0149] 二进制信号F的构造在如图2所示的流构造软件模块CF中实现。

[0150] 然后通过通信网络(未示出)将流F发送到远端。该远端包括解码器,将在下文描述中进一步详细描述该解码器。

[0151] 现在将主要参考图1来描述本发明的另一实施例。

[0152] 该另一实施例与前一个实施例的区别仅在于所要隐藏的符号的数量为0或N,其中N是大于等于2的整数。

[0153] 为此,前述比较子步骤C72被图1中的虚线表示的子步骤C72a替换,在子步骤C72a中,将可调整的系数的数量与若干个预定阈值 $0 < TSIG_1 < TSIG_2 < TSIG_3 \dots$ 进行比较,通过这种方式,如果可调整的系数的数量在 $TSIG_N$ 与 $TSIG_N+1$ 之间,则N个符号要被隐藏。

[0154] 如果可调整系数的数量小于第一阈值 $TSIG_1$,则在上述步骤C20中,对列表 E_1 中的系数进行传统熵编码。为此,列表 E_1 中的每个非零系数的符号被熵编码。

[0155] 如果可调整系数的数量在阈值 $TSIG_N$ 与 $TSIG_N+1$ 之间,则在图1所示的步骤C8中,处理模块MTR_CO计算函数f的值,其表示子列表 E_1 中的系数。

[0156] 在该另一实施例中,由于在解码器上决定隐藏N个符号,函数f是子列表 SE_1 中的系数之和的模 2^N 的余数。假设在所建议的实例中 $N=2$,则要隐藏的两个符号分别是前两个非零系数即 ϵ_2 和 ϵ_3 的前两个符号。

[0157] 在图1所示的下一步骤C9中,处理模块MTR_CO验证N个符号的配置即 2^N 个可能的配置是否对应于子列表 SE_1 中的系数之和的模 2^N 的余数的值。

[0158] 在所建议的 $N=2$ 的实例中,存在 $2^2=4$ 个不同的符号配置。

[0159] 这四个配置符合编码器C0的规则,该规则例如以如下方式确定:

[0160] -等于零的余数对应于两个连续的正符号:+,+;

[0161] -等于1的余数对应于连续的正符号和负符号:+,-;

[0162] -等于2的余数对应于连续的负符号和正符号:-,+;

[0163] -等于3的余数对应于两个连续的负符号:-,-。

[0164] 如果N个符号的配置对应于子列表SE₁中的系数之和的模2^N的余数的值,则对上述列表E₁中除了系数ε₂和系数ε₃的符号以外的系数进行熵编码的步骤C20,所述符号被隐藏在系数之和的模2^N的奇偶校验中。

[0165] 如果不是该情形,则进行用于对子列表SE₁中的至少一个可调整的系数进行调整的步骤C10。该调整由图2中的处理模块MTR_CO以如下这样的方式执行:子列表SE₁中的可调整的系数之和的模2^N的余数达到要隐藏的两个符号中的每个符号的值。

[0166] 在前述步骤C11中,处理模块MTR_CO对列表E₁进行相应的调整。由此得到已调整的列表E_{m1}。

[0167] 然后进行对上述列表E_{m1}中除了系数ε₂的符号和系数ε₃的符号以外的系数的熵编码的步骤C20,该符号被隐藏在系数之和的模2^N的奇偶校验中。

[0168] 解码部分的详细描述

[0169] 现在将描述根据本发明的解码方法的实施例,其中,解码方法以软件或硬件形式通过对初始符合H.264/MPEG-4AVC标准的解码器的调整来实现。

[0170] 根据本发明的解码方法以图3所示的包含步骤D1到D12的算法的形式来表示。

[0171] 根据本发明的实施例,根据本发明的解码方法在如图4所示的解码设备或解码器D0中实现。

[0172] 在图3中未示出的预备步骤中,在接收到的数据信号F中,识别之前已由编码器C0编码的分区B₁到B_z。在优选实施例中,所示分区是具有正方形形状并且全部具有相同尺寸的块。根据图像的尺寸,该尺寸不一定是块尺寸的倍数,最左边的块以及最底部的块可以不是正方形。在替代的实施例中,块例如可以是矩形尺寸和/或可以不互相对齐。

[0173] 每个块或每个宏块本身可以进一步被分为子块,子块自己可以再细分。

[0174] 该识别由如图4所示的流分析软件模块EX_D0来执行。

[0175] 在图3所示的步骤D1中,图4中的模块EX_D0选择要解码的第一块B₁作为当前块B_i。该选择例如包括在第一块B₁的数据项开始时在信号F中放置读指针。

[0176] 然后对选中的已编码的块中每个块进行解码。

[0177] 在图3所示的实例中,对已编码的块的B₁到B_z中的每个块相继实施这样的解码。例如根据本领域技术人员众所周知的光栅扫描来对块进行解码。

[0178] 根据本发明的解码在如图4所示的解码器D0的解码软件模块MD_D0中实现。

[0179] 在图3所示的步骤D2中,首先对已选择的第一当前块B₁进行熵解码。该操作由图4所示的例如CABAC类型的熵解码模块DE_D0来执行。在该步骤中,模块DE_D0对与列表E₁或已调整的列表E_{m1}中每个被编码的系数的幅度相对应的数字信息项执行熵解码。在该阶段,只有列表E₁或已调整的列表E_{m1}中的系数的符号未被解码。

[0180] 在图3所示的步骤D3中,确定在之前的熵编码步骤C20中可能已被隐藏的符号的数量。该步骤D3由如图4所示的处理软件模块MTR_D0来执行。步骤D3类似于前述用于确定要被隐藏的符号的数量的步骤C7。

[0181] 在优选实施例中,隐藏符号的数量是1或0。此外,根据所述优选实施例,被隐藏的是第一非零系数的符号。因此在所示实例中,它需要系数ε₂=+9的正符号。

[0182] 在替代实施例中,隐藏符号的数量是0、1、2、3或更多。

[0183] 根据步骤D3的优选实施例,在图3所示的第一子步骤D31中,根据所述列表 E_1 或根据已调整的列表 Em_1 来确定包含可能在编码时已被调整的系数 $\epsilon'_1, \epsilon'_2, \dots, \epsilon'_M$ 的子列表,其中 $M < L$ 。

[0184] 该确定由与前述编码步骤C7相同的方式来执行。

[0185] 和前述处理模块MTR_CO类似,处理模块MTR_DO被初始配置以不调整:

[0186] -位于第一非零系数之前的零系数或多个零系数,

[0187] -以及由于计算复杂的原因,位于最后的非零系数之后的零系数或多个零系数。

[0188] 在所实例中,在子步骤D31结束时,存在子列表 SE_{m_1} 从而 $SE_{m_1} = (9, -6, 0, 0, 1, 0, -1, 2, 0, 0, 1)$ 。因此,得到可能已被调整的11个系数。

[0189] 在图3所示的下一子步骤D32中,处理模块MTR_DO将可能已被调整的系数的数量与预定阈值TSIG进行比较。在优选实施例中,TSIG具有值4。

[0190] 如果可能已被调整的系数数量小于阈值TSIG,则在图3所示的步骤D4中,对列表 E_1 中的系数的所有符号进行传统熵解码。该解码由图4中的标记DE_DO表示的CABAC解码器来执行。为此,列表 E_1 中的每个非零系数的符号被熵解码。

[0191] 如果可能已被调整的系数的数量大于阈值TSIG,则在所述步骤D4中,对列表 Em_1 中的系数的除了第一非零系数 ϵ_2 的符号以外的所有符号进行传统熵解码。

[0192] 在图3所示的步骤D5中,处理模块MTR_DO计算表示子列表 SE_{m_1} 的系数的函数 f 的值,以确定所计算的值是偶数还是奇数。

[0193] 在信号F中隐藏一个符号的优选实施例中,函数 f 是子列表 SE_{m_1} 中的系数之和的奇偶校验。

[0194] 根据在编码器C0上使用的与解码器D0上相同的规则,子列表 SE_{m_1} 中的系数之和的偶数值表示已调整的列表 Em_1 中的第一非零系数的符号是正的,而子列表 SE_{m_1} 中的系数之和的奇数值表示已调整的列表 Em_1 中的第一非零系数的符号是负的。

[0195] 在 $SE_{m_1} = (+9, -6, 0, 0, +1, 0, -1, +2, 0, 0, +1)$ 的示例性实施例中,系数的总和等于6,且因此是偶数。于是,在步骤D5结束时,处理模块MTR_DO由此推断第一非零系数 ϵ_2 的隐藏符号是正的。

[0196] 在图3所示的步骤D6中,并且借助于在步骤D2、D4和D5中所有重构的数字信息项,以预定的顺序来重构块 B_{q_1} 的量化系数。在所实例中,这是相对于前述编码步骤C6中所执行的之字形扫描相反的之字形扫描。该步骤由如图4所示的读取软件模块ML_DO执行。更具体而言,模块ML_DO使用所述相反的之字形的扫描顺序来包含块 B_{q_1} (二维)中的列表 E_1 (一维)的系数。

[0197] 在图3所示的步骤D7中,根据传统的反量化操作来对已量化的残留块 B_{q_1} 进行反量化,该操作是与前述编码步骤C5中执行的量化相反的操作,以生成已解码的反量化块 BD_{q_1} 。该步骤由如图4所示的反量化软件模块MDQ_DO来执行。

[0198] 在图3所示的步骤D8中,对反量化块 BD_{q_1} 进行逆变换,所述逆变换是与编码时的前述步骤C4中执行的直接变换相反的操作。然后得到已解码的残留块 BD_{r_1} 。该操作由如图4所示的逆变换软件模块MTI_DO来执行。

[0199] 在图3所示的步骤D9中,对当前块 B_1 进行预测解码。这样的预测解码传统上在针对

至少一个之前已解码的块来预测块 B_i 的过程中通过已知的帧内和/或帧间预测技术来执行。该操作由如图4所示的预测解码模块PRED_D0来实现。

[0200] 毋庸赘述,如H.264标准所建议的其它帧内预测模式是可能的。

[0201] 在该步骤中,使用前述步骤中的解码语义元素来执行预测解码,且尤其包括预测类型(帧间预测或帧内预测),并且如果必要的话,包括帧内预测模式、块或宏块(如果宏块已被细分)的分区类型、在帧间预测模式中使用的参考图像索引以及运动矢量。

[0202] 上述预测解码步骤构造已预测的块 B_{p1} 。

[0203] 在图3所示的步骤D10中,通过将已解码的残留块 BDr_1 加到已预测的块 B_{p1} 来构造已解码的块 BD_1 。该操作由图4所示的重构软件模块MR_D0来执行。

[0204] 在图3所示的步骤D11中,解码模块MD_D0测试已解码的当前块是否是信号F中识别的最后的块。

[0205] 如果当前块是信号F中最后的块,则在图3所示的步骤D12中,解码方法结束。

[0206] 如果不是该情形下,则选择下一块 B_i ,根据前述光栅扫描的顺序通过重复步骤D1到D10来对 $1 \leq i \leq Z$ 的块 B_i 进行解码。

[0207] 现在将主要参考图3来描述本发明的另一实施例。

[0208] 该另一实施例与前一个实施例的区别仅在于所隐藏的系数的数量为0或N,其中N是大于等于2的整数。

[0209] 为此,前述比较子步骤D32被图3中的虚线所示的子步骤D32a替换,在子步骤D32a中,将可能已被调整的系数的数量与若干个预定的阈值 $0 < TSIG_1 < TSIG_2 < TSIG_3 \dots$ 进行比较,通过这种方式,如果所述系数的数量在 $TSIG_N$ 与 $TSIG_N+1$ 之间,则N个信号已被隐藏。

[0210] 如果所述系数的数量小于第一阈值 $TSIG_1$,则在前述步骤D4中,对列表 E_1 中的系数的所有符号进行传统熵解码。为此,列表 E_1 中的每个非零系数的符号被熵解码。

[0211] 如果所述系数的数量在阈值 $TSIG_N$ 与 $TSIG_N+1$ 之间,则在前述步骤D4中,对列表 E_1 中的系数的除了所述调整列表 Em_1 中的第一非零系数的N个各个符号以外的所有符号进行传统熵解码,所述N个符号被隐藏。

[0212] 在该另一实施例中,处理模块MTR_D0在步骤D5中计算函数f的值,其是子列表 SEm_1 中的系数之和的模 2^N 的余数。假设在所建议的实例中 $N=2$ 。

[0213] 处理模块MTR_D0由此根据在解码时所使用的规则来推断被分别分配给两个第一非零系数 ϵ_2 和 ϵ_3 中的每个非零系数的两个隐藏符号的配置。

[0214] 一旦这两个符号已被重构,则执行上述步骤D6到D12。

[0215] 毋庸赘述,本文所描述的实施例仅通过指示的方式给出而不是要限制,并且本领域技术人员可以容易地实现多种调整而不偏离本发明的范围。

[0216] 于是例如,根据关于图1所示的简化实施例,编码器C0可被配置为隐藏至少 N' 个预定符号,其中, $N' \geq 1$,而不是0、1或N个预定符号。在该情形下,比较步骤C72或C72a将被移除。通过相应的方式,根据关于图3所示的简化实施例,解码器D0将被配置为重构 N' 个预定符号而不是0、1或N个预定符号。在该情形下,比较步骤D32或D32a将被移除。

[0217] 此外,在编码步骤C72和解码步骤D32中应用的判断标准可被另一类型的标准替换。为此,代替将可调整的系数的数量或可能已被调整的系数的数量与阈值进行比较,处理模块MTR_CO或MTR_D0可以分别应用判断标准,所述判断标准是可调整或可能已被调整的系

数的幅度之和的函数,或者可调整或可能已被调整的系数中存在的零的数量的函数。

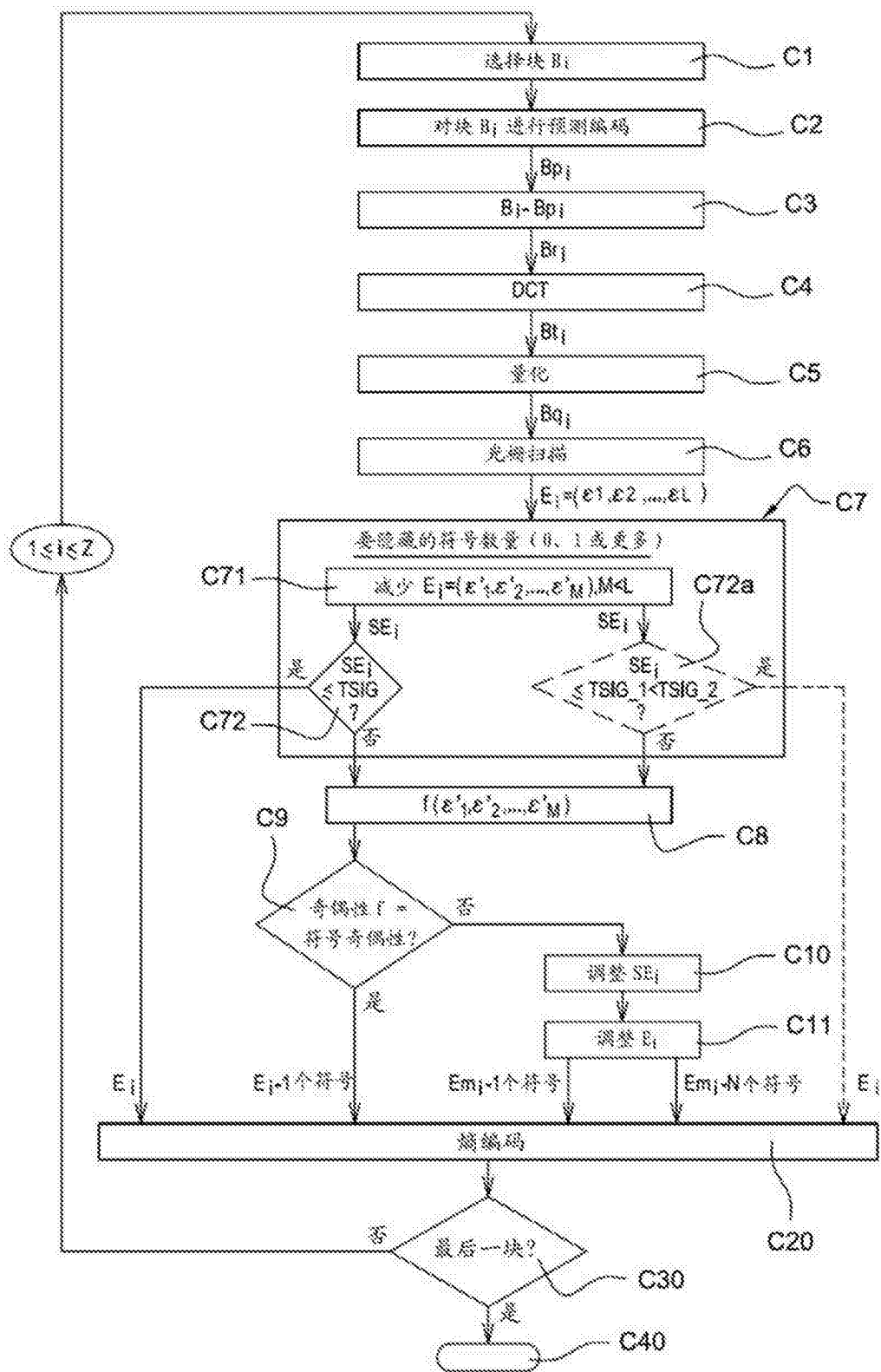


图1

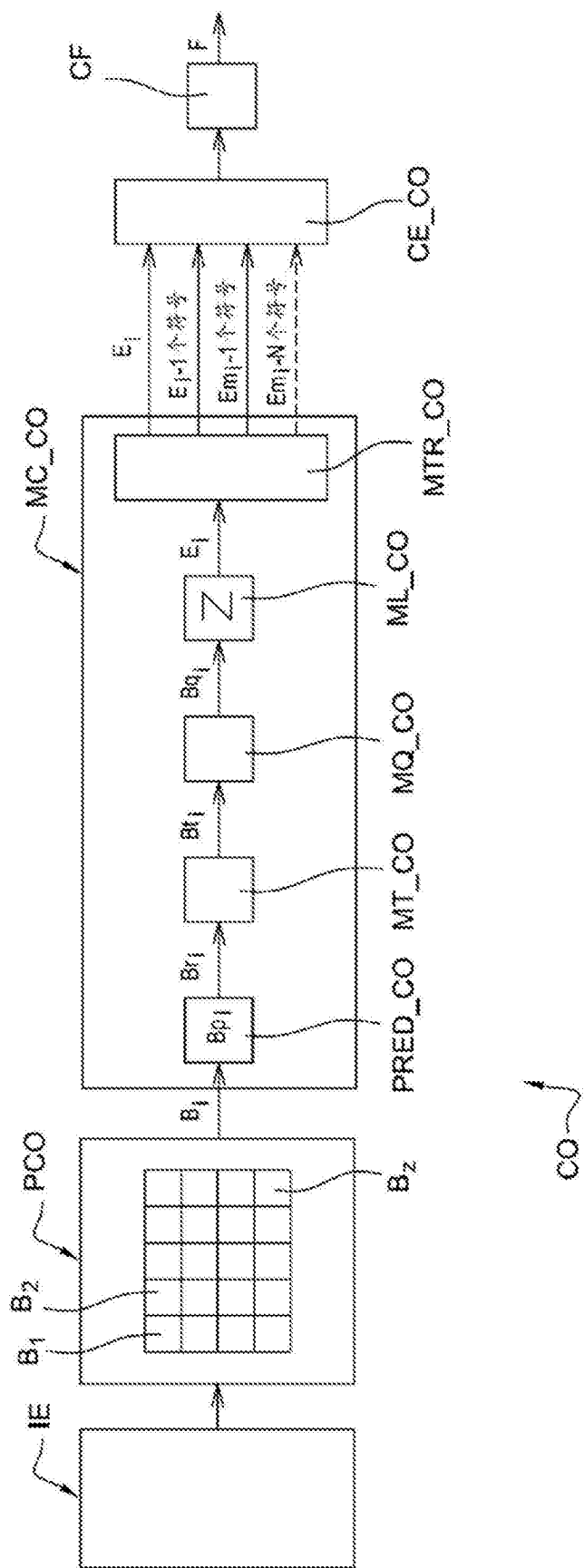


图2

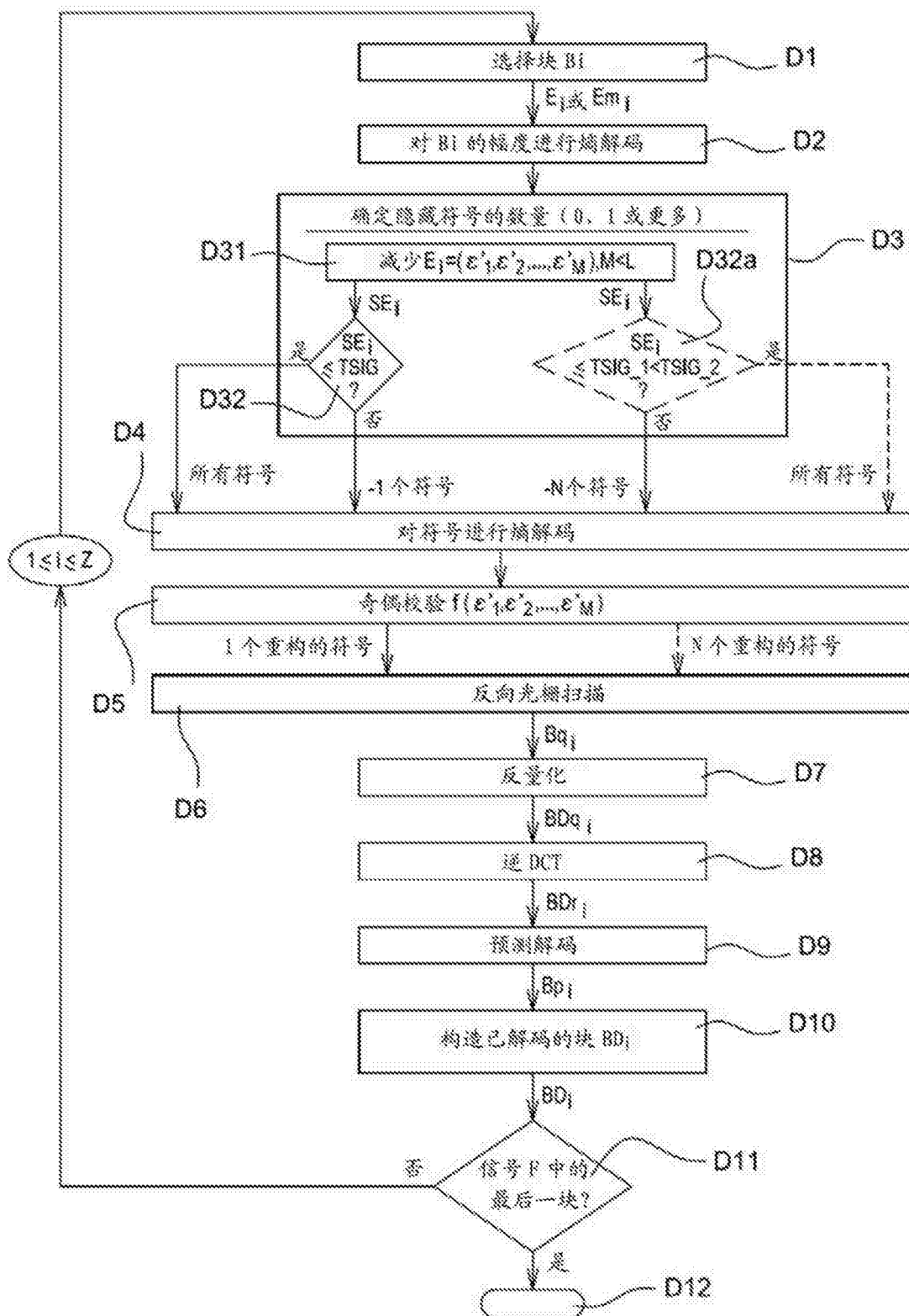


图3

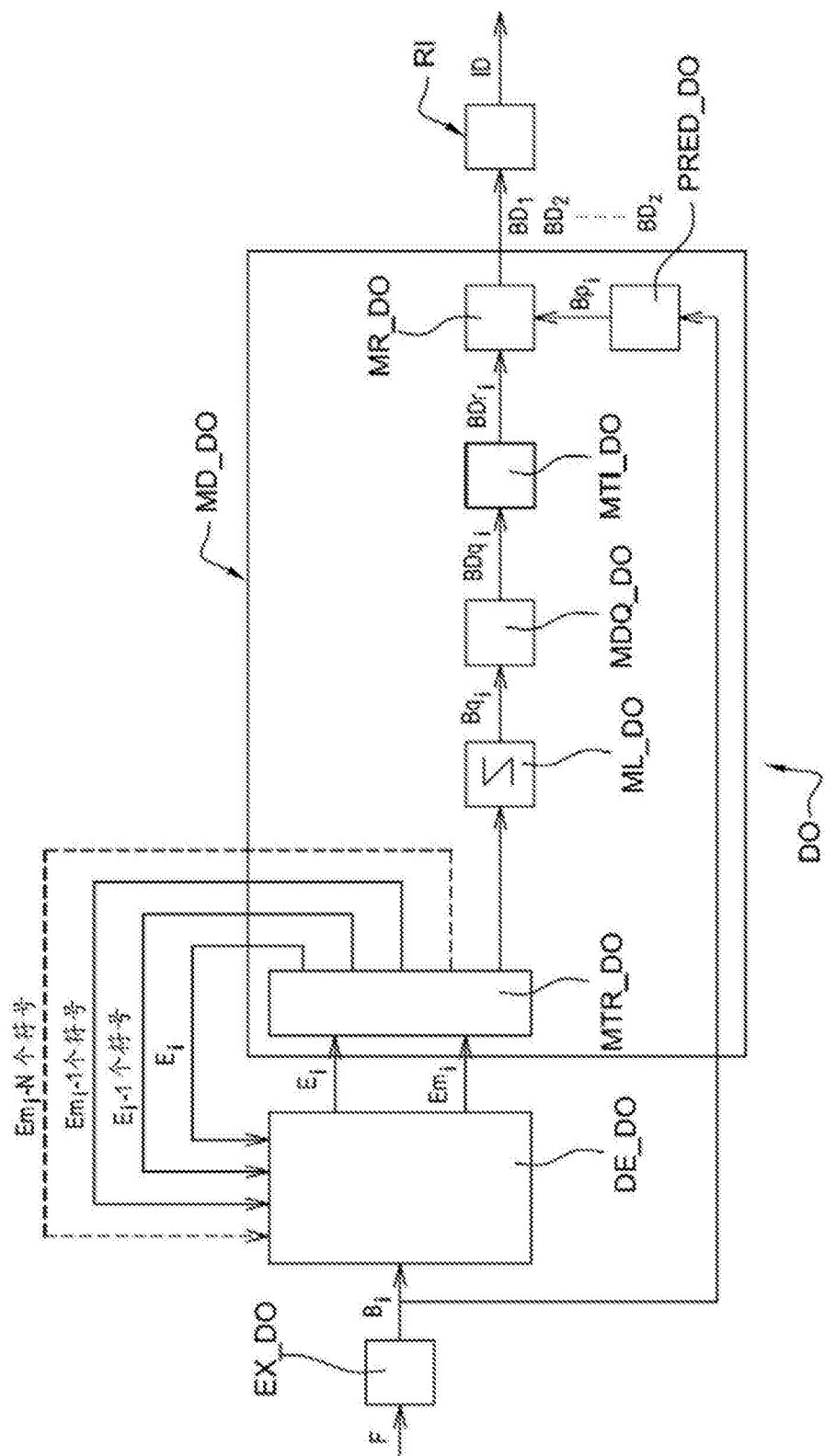


图4