

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H04N 1/41 (2006.01)

G06T 9/00 (2006.01)



[12] 发明专利申请公布说明书

[21] 申请号 200610100305.5

[43] 公开日 2007年2月21日

[11] 公开号 CN 1917561A

[22] 申请日 1996.7.2

[21] 申请号 200610100305.5

分案原申请号 200410078981.8

[30] 优先权

[32] 1995.7.3 [33] US [31] 498036

[71] 申请人 株式会社理光

地址 日本东京都

[72] 发明人 A·赞迪 E·L·许瓦兹

M·J·戈米什 M·波利克

[74] 专利代理机构 中国专利代理(香港)有限公司
代理人 刘宗杰

权利要求书4页 说明书66页 附图32页

[54] 发明名称

可逆子波变换和嵌入代码流处理

[57] 摘要

一种以用于在一通道上传送的压缩数据流对输入数据进行编码的编码器，其特征在于，所述熵编码器包括：一用来将输入数据变换为多个系数的可逆子波滤波器；连接到该可逆子波滤波器用来在诸如所产生的一比特流之类的多个系数上执行嵌入编码的一嵌入编码器；连接到该嵌入编码器用来在该比特流上执行熵编码以产生编码数据的一熵编码器；一用来控制在一固定尺寸存储器中存储的被压缩数据的通道管理器，该存储器管理器将被压缩数据分成多个重要级，并且进一步其中该通道管理器传送的所有充分的带宽里的数据是有用数据，否则较少或重要数据被丢弃并且仅传送较重要的数据。



1. 一种以用于在一通道上传送的压缩数据流对输入数据进行编码的编码器，其特征在于，所述熵编码器包括：

一用来将输入数据变换为多个系数的可逆子波滤波器；

连接到该可逆子波滤波器用来在诸如所产生的一比特流之类的多个系数上执行嵌入编码的一嵌入编码器；

连接到该嵌入编码器用来在该比特流上执行熵编码以产生编码数据的一熵编码器；

一用来控制在一固定尺寸存贮器中存贮的被压缩数据的通道管理器，该存贮器管理器将被压缩数据分成多个重要级，并且进一步其中该通道管理器传送的所有充分的带宽里的数据是有用数据，否则较少或重要数据被丢弃并且仅传送较重要的数据。

2. 如权利要求 1 的编码器，其特征在于，该通道管理器动态地确定该通道的带宽。

3. 如权利要求 1 的编码器，其特征在于，该数据被分割成多个尾部，并且每个尾部被分别地编码和由重要级分成固定尺寸页。

4. 如权利要求 1 的编码器，其特征在于，该通道管理器进一步包括一缓冲存贮器。

5. 如权利要求 4 的编码器，其特征在于，该编码器是具有多个固定尺寸段的环形缓冲存贮器，这样作为每个固定尺寸段输出的数据被完全地被分配。

6. 如权利要求 5 的编码器，其特征在于，该信息包尺寸与该通道带宽相匹配。

7. 一种以在一通道上被传送的压缩数据流对输入数据进行编码的编码器，其特征在于，所述的熵编码器包括：

一用来将输入数据变换成多个系数的可逆子波滤波器；

一连接到该可逆子波滤波器用来在诸如一所产生的比特流的多

个系数上执行嵌入编码的嵌入编码器；

一连接到该嵌入编码器用来在该比特流上执行熵编码以产生编码数据的熵编码器；和

连接到该熵编码器以将该数据分割成固定尺寸段的一通道管理器，其中每个固定尺寸段包括有在该段中的该最重要数据的一指示。

8. 如权利要求 7 的编码器，其特征在于，每一段包括被编码数据、一随意的指针和在这一段中的最重要数据的级。

9. 如权利要求 7 的编码器，其特征在于。每一区域的开始点是利用重新启动标记识别的。

10. 如权利要求 8 的编码器，其特征在于，该随意指针可包括用于固定尺寸存储器管理的下一段指针。

11. 如权利要求 8 的编码器，其特征在于，该随意的指针包括一用于该段的 ID。

12. 如权利要求 8 的编码器，其特征在于，该随意指针指明该段包括的若干不同区域。

13. 如权利要求 8 的编码器，其特征在于，该随意指针包括一个指明多于二段所含有的区域的比特。

14. 一种用来将输入数据编码为压缩数据流的编码器，其特征在于，所述编码器包括：

一用来利用具有一可变级数的角锥分解将该输入数据变为多个系数的可逆子波滤波器；

一连接到该可逆子波滤波器的具有一固定尺寸的存储器，其中在多个系数的分解期间该可逆子波滤波器使用该存储器存储信息；

一连接到该可逆子波滤波器和该固定尺寸存储器、基于该存储器的固定尺寸去选择该级数的控制机构；

一连接到该可逆子波滤波器和固定尺寸存储器的排序和造型机构，所述排序和造型机构响应于多个系数产生一嵌入数据流，其中该排序和造型对多个系数排序并对多个系数的每一个进行二进制值

排序；和一连接到该排序和造型机构可对嵌入代码流进行二进制熵编码以产生压缩数据的二进制熵编码器。

15. 一种使用一嵌入代码流的方法，其特征在于，包括：

识别一目标装置去接收在一嵌入代码流中的数据；

对每个比特平面译码以提供该目标装置数据，所述译码步骤包括为了维持该目标装置数据而在该嵌入代码流中舍位每一比特平面的步骤。

16. 如权利要求 15 的方法，其特征在于，每个比特平面基于在每个编码单元中表示舍位可能出现的位置的一指示而被舍位。

17. 如权利要求 16 的方法，其特征在于，该指示包括一标识符。

18. 如权利要求 16 的方法，其特征在于，该指示包括一指针。

19. 如权利要求 15 的方法，其特征在于，基于指明的每个编码单元中指明舍位可能出现的位置的多个指示中的一个来对每个比特平面舍位，所述舍位步骤进一步包括基于该目标装置选择多个指示中的一个的步骤。

20. 如权利要求 15 的方法，其特征在于，舍位步骤包括对来自嵌入代码流的在每个编码单元中被单独地编码的目标分辨率系数舍位。

21. 如权利要求 15 的方法，其特征在于，该目标装置包括一低分辨、高象素深度嵌入目标以便在需要实现该目标装置的全象素深度和低空间分辨时作为若干较高级系数的译码的评码步骤。

22. 如权利要求 15 的方法，其特征在于，该目标装置包括一高分辨力、低象素深度嵌入目标和在一比特平面数时舍位每个编码单元并且逆子波变换每个编码单元的未舍位数据以完成该目标装置的低象素深度和高空间分辨。

23. 如权利要求 15 的方法，其特征在于，进一步包括步骤：

基于在该目标装置中的有用缓冲的数而选择编码单元；和

用多于有用缓冲的数据舍位每个编码单元。

24. 如权利要求 15 的方法，其特征在于，舍位步骤进一步包括有步骤：

检测舍位每个编码单元的均匀总数；

舍位每个编码单元中的至少一最重要级的一部分。

25. 如权利要求 24 的方法，其特征在于，舍位步骤是使用在显示重要级信息的该代码的一标题的信息来执行的。

26. 如权利要求 24 的方法，其特征在于，舍位步骤是使用为了在该代码流中的每个编码单元而显示的重要级信息的该代码的一标题中的信息来执行的。

27. 如权利要求 24 的方法，其特征在于，各步骤是在编码时间之后执行的。

可逆子波变换和嵌入代码流处理

本申请是申请日为1994年9月20日，申请号为08/310,146，名称为“使用可逆嵌入子波进行压缩的装置”的申请的后续部分。

技术领域

本发明涉及数据压缩和解压系统领域，更详细地说，本发明涉及用来在压缩/解压系统中数据的无损耗和有损耗编码和解码的方法和装置。

背景技术

数据压缩是用于大量数据的存贮和传送的一种极为有效的方法。例如，当压缩被用来降低还原诸如一文件传真传送之类的图象所需的比特数时，传送一图象所需的时间急剧的减少。

在现有技术中存在有许多不同的数据压缩技术。压缩技术可分成二类主要的类型，有损耗编码和无损耗编码。有损耗编码包含导致信息损耗的编码，这就不能保证原始数据的理想的再现。有损耗压缩的目的是以这样一种方式来改变原始数据以使得它们不会不适宜或不会易被察觉。在无损耗压缩中，所有信息均被保留并且数据以一种能够理想再现的方式被压缩。

在无损耗压缩中，输入符号或亮度数据被变换成输出代码字。该输入可以包括图象、声频、一维(即，数据空间地或时间地变化的数据)、二维(即，在二个空间方向由(或一空间和一时间量纲)上数据变化)、或多维/多频谱数据。如果压缩成功，则该代码字由比未编码的输入符号(或亮度数据)所需的比特数要少的比特来表示。无损耗编码方法包括编码(即，Lempel-Ziv)、行程编码、枚举编码和熵编码的字典方法。在无损耗图象压缩中，压缩是基于预测或上下文、插值编码。用于传真通信的JBIG标准和用于连续色调图象的DPCM(差动脉冲编码调制—在JPEG标准中的一种选择)是用于图象的无损耗压缩的例子。在有损耗压缩中，输入符号或亮度数据在变换成输出代码字之前被量化。量化的意思是指保存数据的相关特征同时消除不重要的特征。在量化之前，有损耗压缩系统通常使用一变换以提供能量压缩。JPEG是用于图象数据的有损耗编码的例子。

近来图象信号处理的发展集中趋向于发展高效率和精确的数据压缩编码形式。提出了各种变换或角锥信号处理的方式，包括多分辨(multiresolution)

角锥处理和子波角锥处理。这些方式还称之为副频带处理和分层处理。图象数据的子波角锥处理是多分辨角锥处理的一特定形式，它可使用正交镜像滤波器(QMFS)以产生一原始图象的副频带分解。应注意的是存在有非QMF的其它类型。对于子波处理的信息，请见Antonint, M., 等“Image Coding Using Wavelet Transform”, IEEE Transaction on Image Processing, Vol. 1, No.2, 1992. 4. 4; Shapiro, J., “An Embedded Hierarchical Image Coder Using Zerotrees of Wavelet Coefficients”, Proc. IEEE Data Compression Conference, pgs 214-223, 1993.

与现有技术子波处理极为相关的一个问题是在进行处理时需要一个存贮所有数据的大存贮器。换句话说，在执行子波处理中，在该数据执行编码之前所有的数据必须被检验。在这种情况下，直至最少一个充分的通过全部数据为止没有数据输出。事实上，子波处理典型地包含多重数据传送。由于这个原因，通常需要一大存贮器。最好是利用子波处理，同时避免大存储器的要求。另外，还期望执行子波处理时仅一次通过该数据。

许多子波或副频带变换处理过程需要特殊范围形式的滤波器。例如，低通和高通滤波器必须具有相同的长度、该系数的平方和必须为1、高通滤波器的时间和频率必须与低通滤波相反等(见1991年5月公布的Lawton 等人的美国专利No 5,014, 134)。最好是允许滤波器的种类要宽一些。也就是，期望能提供使用的低通和高通滤波器不要求有相同的长度、其系数的平方和不需为1、高通滤波器不要求是低通滤波器相反的时间和频率的子波或副频带变换处理过程。

发明内容

本发明是提供使用一种提供了良好能量压缩变换的有损耗和无损耗压缩。

描述了一种压缩和减压系统。该压缩系统包括一将输入数据编码成一被压缩的数据流的编码器。该熵编码器、即所说编码器包括一可逆子波滤波器、一排序和造型机构和一二进制熵编码器、可逆子波滤波器将输入数据变换为使用角锥介压的系数。该排序和造型机构从通过在该系数内对该系数和二进制值排序的系数中产生一嵌入式代码流。该二进制熵编码器可使该嵌入式代码流二进制熵编码以产生被压缩的数据流。

通过下面结合附图对本发明的各种不同实施例的详细说明将对本发明有更为全面的了解，但本发明并不局限于这些特定的实施例，这些实施例仅供解释

和理解而已。

附图说明

图1是本发明的编码系统的编码部分的一实施例的方框图。

图2是本发明系数数据排序和造型的一实施例的方框图。

图3A是一子波分解合成系统的方框图。

图3B是对于具有非重叠最小限度长度可逆滤波器滤波的情况下变换系统的前向和反向表达。

图4是一2D可逆滤波器的另一可供选择的实施例的方框图。

图5示出了一四级分解的结果。

图6示出了在二个连续级之间的母体关系。

图7是一三级角锥变换的方框图。

图8是一二维、二级变换的方框图。

图9是表示执行一多分辨解压的一维滤波器的方框图。

图10是一使用本发明的可逆子波的一系统的方框图。

图11是使用本发明的可逆子波的放大和分析系统的方框图。

图12A示出了在本发明中的系数尺寸。

图12B是用于使用本发明的系数对准的频段的乘法器的一实施例。

图13是本发明的编码过程的一实施例的流程图。

图14是本发明的详码过程的一实施例的流程图。

图15是本发明的造型过程的一流程图。

图16是本发明的前向子波滤波器的一实施例。

图17是本发明的一反向子波滤波器的一实施例的方框图。

图18示出了不同变换的系数范围。

图19A和19B示出了使用查询表的上下文模式的两个实施例。

图20示出子波分解级的一个实施例。

图21示出了一编码单元。

图22示出了在TS-变换时的垂直通路。

图23示出了缓冲和系数计算。

图24A示出了一代码流结构的一个实施例。

图24B示出了用于低分辨目标的一代码流结构的一个实施例。

图25示出了在各系数(或象素)之间的相邻关系。

图26A-D示出了上下文模式的实施例。

图27是本发明的上下文模式的一实施例的方框图。

图28是本发明的符号/数量单元的一实施例的方框图。

图29示出了用于一通过操作的被编码数据存贮器的动态分配。

图30示出一通道管理器的一个实施例。

图31示出了在本发明中存贮器的利用。

图32示出了在本发明中的一比特流。

图33示出了一程序段的结构。

图34示出了目标器件相对一参量空间的关系曲线。

图35A和35B示出了本发明的分析程序的不同的实施例。

具体实施方式

现在介绍一种用来压缩和解压的方法和装置。为了提供对本发明的彻底的了解，而陈述了诸如编码器的类型、比特数、信号名称等大量特殊的细节。但是，本技术领域的普通技术人员很清楚的是没有这些特殊的细节也可以实施本发明。在另外的例子中，为了避免使本发明模糊不清，已知的结构和装置足以方框图形式表示而不详细说明。

被详细说明的某些部分在下面是在一计算机存贮器中的数据比特的操作的算法和符号表达式的形式来表示。数据处理技术领域中的某些技术人员利用这些算法说明和表达式作为手段来将他们的工作内容最有效地传送给该技术领域中的另外的技术人员。这里的一算法一般来说是想象为导致一希望结果的步骤的一自相容顺序。这些步骤是实际参量所需的实际的操作。通常，虽然并非必然的，这些电或磁信号形式的参量可以被存贮、转移、组合、比较以及其它的方式操作。主要为了共同使用的原因，有时将这些信号作为比特、数值、要素、符号、字符、术语、数等被证明是方便的。

但是，应记住，所有这些以及类似的项是与适当的实际量相关的并且仅仅是向这些量提供方便的标记而已、除非从下面的讨论中另有明显的特别地说明，在整个本发明中可了解到相对于计算机系统或类似的电子计算装置的动作和处理的讨论所利用的诸如“处理”或“估算”或“计算”或“测定”或“显示”等等，是将在计算机系统的寄存器或存贮器内表示为物理(电)量的数据操作和

转换成在计算机的存贮器或寄存器内或其它的这种信息存贮、传输或显示装置内的其它类似地表示为物理量的数据。

本发明还涉及用来在其内执行该操作的装置。这种装置可针对特定目的而构成，或者它可以包括由在该计算机内存贮的计算机程序来选择地激活或重新组合的一通用的计算机。在其内所呈现的算法和显示并不固有地涉及任何特定的计算机或其它的装置。可以使用具有根据其内所教导的程序的各一般用途的机器，还可以证实为执行所需的方法步骤构成更为专业化的装置则更为方便。在下面的说明中将会出现用于这些机器的各种所需的构型。另外，本发明对涉及的任何特定程序语言不作描述。将会了解到各种程序语言可用作在其中描述本发明的教导的工具。

如下的术语被用于下面的说明中。对于这些不同的术语包括有一个定义。但是所提供的定义并不认为限于在该技术领域已知术语的范围。这些所提供的定义有助于对本发明的理解。

比特有效位：一种数表示，类似于符号数值，如果有的话具有标题比特、跟着有符号比特，跟着有尾部比特。相应于这种表示以比特平面顺序嵌入编码。

编码单元：可按任意顺序被共同编码的系数单元。在一实施例中，一编码单元包括有一个或多个以一矩形安排的树。一编码单元可包含有一完整图象、一组图象或另外的数据集。该编码单元具有在缓冲器尺寸上为了计算一变换所必须的有效压缩。另外，在一实施例中，在与前编码单元的系数外侧可以没有任何内容。但是，在一编码单元之中或许多编码单元之后熵代码可被复位。该编码单元不必随机地可寻址。

上下文模式：相对于被编码的与前比特的可利用信息给出有关与前比特的历史地学习信息。这使得可以对熵编码条件概率进行估算。

树：该系数和象素是涉及在最高级子波分解的LL中的一单一系数。该系数是若干级数的量。

带：对于二维数据该系数和象素是涉及在最高级子波解压的LL中系数的单一行或线。带的规定类似于其它量纲数据的规定。

- 分解级:** 在该子波分解角锥中的一位置。
- 嵌入量化:** 该量化被代码流所蕴层。例如, 如果重要级是按序安置, 从最重要到最不重要, 则通过该代码流的简单舍位来执行量化。同样的函数性可利用标记、标志、指针或其它的信号来获得。
- 熵编码器:** 基于它的上下文来对当前比特编码的一种装置。对于当前比特(或多个比特)的最佳表达式该上下文允许概率估算。
- 固定速率:** 保持一定的象素速率并具有有限的带宽通路的一应用或系统。这需要获得局部的平均压缩而不是全局的平均压缩。例如: MPEC。
- 固定尺寸:** 具有一有限尺寸缓冲器的一应用或系统。在这种情况下, 获得一全局的平均压缩, 即, 一打印缓冲器。(一应用可为固定速率和固定尺寸或二者之一)。
- 固定长度:** 将数据的特定信息块转换为被压缩数据的特定信息块的一系统, 即, BTC。固定长度代码适用于固定速率和固定尺寸应用, 但是, 与可变速率系统比较其速率失真性能通常是低劣的。
- 水平线上下文模式:** 一种用于嵌入编码器(在一实施中)的上下文模式, 将其规定为本发明的一部分。
- 标题:** 在比特有效位表达式中, 该标题比特是自该最高有效位向上的数量比特并且包括第一非零比特。
- 重叠变换:** 一种将单源取样点分配给相同频率的多个系数的变换。例如包括许多子波和重叠正交变换。
- 累进:** 一种被排序的代码流, 使得来自可由多个数据提炼的被编码数据的部分的相关被压缩结果是可利用的。在某些实施例中, 一代码流由数据加深比特平面来排序, 在这种情况下, 它通常称之为子波系数数据。
- 角锥:** 分辨的顺序性, 每个较低的分辨是二个较大的一线性因素(在一面积中四个较大的因素)。
- 可逆变换:** 用整数运算所实施的有效变换, 能实现精确的重建。
- S变换:** 带有2抽头的低通和2抽头高通滤波器的一特殊可逆子波滤波器部分。

- 尾部:** 在比特有效位表达式中, 尾部比特是带有比最高有效非零比特低的有效位的幅值比特。
- 尾部信息:** 在一实施例中, 对于在比特有效位表达式中所表示的一系数有四种可能状态。它是系数和当前比特平面的函数, 并被用于水平线上下文模式。
- 尾部连接:** 在一实施例中, 取决于该尾部信息是否为零或非零的两种状态。它被用于水平线上下文模式。
- TS-变换:** 二—六变换, 带有2抽头低通和6抽头高通滤波器的一特殊子波滤波器部分。
- 接合无损耗/有损耗:** 相同压缩系统提供一可无损耗或有损耗重建的一编码数据流。如在以下所述的本发明的情况中, 这种代码流可以无须对该编码器置位或指示。
- 视觉重要级:** 通过规定一个特定系统, 输入数据(象素数据、系数、误差信号等)被逻辑地分成具有相同视觉压缩的组。例如, 最高有效比特平面或若干比特平面可比较低平面更为视觉重要。低频信息通常比高频更为重要。本发明如下所述包括的“视觉有效位”的多数操作定义是相同于某些误差量度而言的。但是, 在视觉重要的系统定义中可插入较好的视觉量度。另外的数据类型具有另外的重要级, 例如, 音频数据具有音频重要级。
- 子波滤波器:** 在子波变换中使用的高通和低通合成和分析滤波器。
- 子波变换:** 利用“频率”和“时间(或空间)”约束的变换。在所介绍的实施例中, 是一包含一高通滤波器和一低通滤波器的变换。其结果系数是 + 中取 = (临界地滤除)并且该滤波器被利用于低通系数。

本发明提供了一种具有一编码部分和一译码部分的压缩/解压系统。该编码部分负责时输入数据编码以产生压缩数据, 同时译码部分负责对被编码的数据译码以产生原始输入数据的重建的文本。该输入数据可包括各种数据类型, 诸如图象(静止的或视频的)、音频等。在一实施例中, 该数据是数字信号数据; 但数字化的模拟数据文本数据格式和其它格式也是可以的。对于编码部分和

/或译码部分来说, 该数据源可以是一存储器或通道。

在本发明中, 编码部分和/或译码部分的部件可由诸如在一计算机系统中被使用硬件或软件来实施, 本发明提供了一种无损耗压缩/解压系统。本发明还可被构成能执行有损耗压缩/解压的构型。

图1是该系统的编码部分的一实施例的方框图。应注意的是该系统的译码部分以和该数据流动的相反顺序工作。参见图1, 由于波变换部分102接收输入图象数据101。子波变换部分102的输出被送到系数数据排序和造型部分103。响应于自子波变换部分102的输出, 该排序/造型部分103产生至少一个由一熵编码器104所接收的比特流。响应于自排序/造型部分103的输出, 熵编码器104产生代码流107。

在一实施例中, 该排序/造型部分103包括一符号/量值格式化学单元201和一接合空间/频率上下文模式202, 如图2所示。在一实施例中, 该接合空间/频率上下文模式202包括一水平线上下文模式, 如下所述。符号/量值单元201的输入被送到子波变换代码部分102的输出。符号/量值单元201的输出被送到接合空间/频率上下文模式202。JSF上下文模式202的输出被送到熵编码器104的输入以产生输出代码流107。

再参见图1, 在本发明中, 如像下面所解释的, 接收图象数据101和在子波变换部分102中利用可逆子波的变换编码以产生表示该图象的多分辨压缩的一系列系数。本发明的可逆子波变换并不是复杂的计算。该变换可以用不带有系统的误差的软件或硬件来完成。另外, 对于能量压缩和压缩特性来说本发明的子波是极好的。这些系数由排序/造型部分103所接收。

该排序/造型部分103提供对系数的排序和造型。该系数排序提供一嵌入数据流。在编码时间、传送时间或译码时间该被嵌入的数据流允许一结果代码流被量化。在一实施例中, 排序/造型部分103将该系数排序并变换成符号-量值格式, 并基于它的有效位(如下面将要说明的)使该格式化的系数服从于一嵌入造型方法。在一实施例中, 该格式化系数服从于接合空间/频率造型。

排序和造型的结果包括由熵编码器所编码的决定(或符号)。在一实施例中, 所有决定均被送至一单一编码器。在另一实施例中, 通过有效位来标明这些决定, 并且对于每个有效位电平的决定通过不同的(实际的或虚拟的)多个编码器

来处理。

参见图2, 来自JSF上下文模式202结果的比特流以利用熵编码104的有效位的顺序被编码。在一实施例中, 熵编码器104包括一个或多个二进制熵编码器。

子波分解

本发明利用可逆子波来开始执行一图象(以图象数据方式)或另外的数据信的分解。在本发明中, 一可逆子波变换包括一精确重建系统以整数算法进行的实施过程, 这样一具有整数系数的信号可被无损耗地复原。通过使用可逆子波, 本发明能够提供具有有限精度算法的无损耗压缩。通过对图象数据利用可逆子波变换的结果是一系列的系数。

本发明的可逆子波变换可使用一组滤波器来实施。在一实施例中, 这些滤波器是二抽头低通滤波器和六抽头高通滤波器。在一实施例中, 这些滤波器是仅使用加和减操作(加以硬件的比特位移)来实施的。另外, 在本发明的一实施例中, 利用该低通滤波器的结果高通滤波器产生它的输出。所得结果的高通系数仅是大于象素深度的少数比特并且该低通系数与该象素深度相同。由于在一角锥分解中仅低通系数反复地被滤除, 可以在多级分解中系数分辨力没有增加。

在另外的实施例中, 低通滤波器输出系数的尺寸可增加, 而不是高通滤波器输出系数的尺寸增加。

一子波变换系统是由一对FIR分析滤波器 $h_0(n)$, $h_1(n)$ 和一对FIR综合滤波器 $g_0(n)$, $g_1(n)$ 所规定。在本发明中, h_0 和 g_0 是低通滤波器而 g_1 和 h_1 是高通滤波器。在图3A中示出了该子波系统的方框图。参见图3A, 一输入信号 $X(n)$ 被提供给分析滤波器 h_0 和 h_1 并且其输出被 $\downarrow 2$ 中取(临界地二次抽样)以产生被变换的信号 $y_0(n)$ 和 $y_1(n)$, 分别作为低通(平滑)和高通(细节)系数。该分析滤波器和它的相应的抽选、或二次抽样、字组构成该子波变换系统的分析部分。该编码器/译码器包括所有处理逻辑和在变换范围中的常规变换(即, 断定、量化、编码等)。如图3A所示的子波系统还包括一综合部分, 在该综合部分中该被变换信号被由 $\uparrow 2$ 向上取样(即每一项之后插入一个零)并且随后通过综合滤波器 $g_0(n)$ 和 $g_1(n)$ 。该低通(平滑)系数 $y_0(n)$ 通过低通综合滤波器 g_0 和该高通(细节)系数 $y_1(n)$ 通过高通滤波器 g_1 。滤波器 $g_0(n)$ 和 $g_1(n)$ 的输出被组合以产生 $X(n)$ 。

在某些实施例中执行向下取样和向上取样, 在另外的实施例中, 由于不执

行向下取样和向上取样所以无需计算所使用的滤波器。

该子波系统可由Z变换项来描述，其中的 $X(Z)$ 、 $X(Z)$ 分别表示输入和输出信号， $Y_0(Z)$ 、 $Y_1(Z)$ 是被低通和高通变换的信号， $H_0(Z)$ 、 $H_1(Z)$ 是低通和高通分析滤波器和 $G_0(Z)$ 、 $G_1(Z)$ 是低通和高通综合滤波器。如果在该变换范围内不存变更和量化，则在图3中的输出 $X(Z)$ 由下式给出

$$\hat{X}(Z) = \frac{1}{2} [H_0(Z)G_0(Z) + H_1(Z)G_1(Z)]X(Z) + \frac{1}{2} [H_0(-Z)G_0(Z) + H_1(-Z)G_1(Z)]X(-Z)$$

在本发明中，称之为“混叠”项的 $\hat{X}(Z)$ 的第二项被取消，这是因为该综合滤波器是被分析滤波器的正交镜所规定，也就是，

$$\begin{cases} G_0(Z) = H_1(-Z) \\ G_1(Z) = -H_0(-Z) \end{cases}$$

在该滤波器系数的项中，

$$\begin{cases} g_0(n) = (-1)^n h_1(n) \\ g_1(n) = -(-1)^n h_0(n) \end{cases}$$

因此，对于一正交镜滤波器对，在变换之后，该输出为：

$$\hat{X}(Z) = \frac{1}{2} [H_0(Z)H_1(-Z) - H_1(Z)H_0(-Z)]X(Z)$$

因此，在本发明的正交镜系统中，该输出仅由分析滤波器的项来决定。该子波变换是将在由用作输入的滤波器所产生的输出中递旧地变换的信号被直接或间接地被提供给该滤波器中。在所述的实施例中，仅仅低通变换分量 $y_0(n)$ 被递旧地变换使得该系统是角锥形的。在图6中示出了这样一种角锥系统的例子。

Z变换是用来表示数据的硬件和/或软件操作的一种简便表示法。乘以 Z^{-m} 模式则在硬件中延迟 m 时钟周期，而在软件中对第 m 5前面元件进行阵列存取。这种硬件的实施包括存贮器、管道级、移相器、寄存器等。

在本发明中，在乘以一常数一延迟项之后信号 $X(n)$ 和 $X(n)$ 是相同的，即Z变换的项中，

$$\hat{X}(Z) = cZ^{-m}X(Z)$$

这称之为—精确重建系统。因此，在本发明的一实施例中，最初应用于输入数据的子波变换是可精确地重建的。

使用哈达玛(Hadamard)变换的本发明的一实施例是一精确重建系统，它的标准形式如下面以Z范围的表达式：

$$\begin{cases} H_0(Z) = \frac{1}{\sqrt{2}}(1+Z^{-1}) \\ H_1(Z) = \frac{1}{\sqrt{2}}(1-Z^{-1}) \end{cases}$$

在置换后, 该输出为

$$X(Z) = Z^{-1}X(Z),$$

这是很清楚地一精确重建。对于哈达玛变换的更详细的情况请见Anil K. Jain, Fundamentals of Image Processing, pg. 155。

哈达玛变换的可逆变型这里称之为S变换。对于S变换的更详细的情况, 请参见 A. and Pearlman, W. "Reversible Image Compression via Multiresolution Representation and Predictive Coding," Dept. of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 1993。因为哈达玛变换是一精确重建变换, 下面的非标准变型(不同于利用固定系数的哈达玛变换)也是一精确重建变换:

$$\begin{cases} h_0(Z) = \frac{1}{2}(1+Z^{-1}) \\ h_1(Z) = 1-Z^{-1} \end{cases}$$

给定的输入信号的取样为 X_0, X_1 , 则这个系统S变换是实施一可逆变换, 如像,

$$\begin{cases} y_0(0) = [(x(0)+x(1))]/2 \\ y_1(0) = x(0) - x(1) \end{cases}$$

该S变换可由具有一普通标志 n 的输出来定义, 如下所示:

$$\begin{cases} s(n) = \left\lfloor \frac{x(2n) + x(2n+1)}{2} \right\rfloor \\ d(n) = x(2n) - x(2n+1) \end{cases}$$

应注意的是在该变换系数寻址中的两个系数是通过二次子取样的结果。可逆变换和变型为：

$$\begin{cases} x(2n) = s(n) + \left\lfloor \frac{d(n)+1}{2} \right\rfloor \\ x(2n+1) = s(n) - \left\lfloor \frac{d(n)}{2} \right\rfloor \end{cases}$$

符号 $\lfloor \cdot \rfloor$ 表示下舍入或截尾并且有时称之为底面函数。类似地, 天棚函数 $\lceil \cdot \rceil$ 对最接近整数上舍入。

可以证实以下面方式实施可逆实际上仅是在近似于 $X(0)+X(1)$ 的最低有效位中丢失信息。但由于 $X(0)+X(1)$ 和 $X(0)-X(-1)$ 的最低有效位是相同的, 所以可从该高通输出 $y_1(0)$ 复原。换句话说,

$$X(0) = y_0(0) + \left\lfloor (y_1(0)+1)/2 \right\rfloor$$

$$X(1) = y_0(0) - \left\lfloor (y_1(0)-1)/2 \right\rfloor$$

该S变换是使用最小长度可逆滤波器的非重叠变换。最小长度滤波器包括有一对滤波器，这两个滤波器具有二个抽头。最小长度变换不提供良好的能量压缩。由于该滤波器的长度等于滤波器数所以最小长度滤波器实施一非重叠变换。重叠变换使用至少一个其长度大于滤波器数的滤波器。使用较长（不是最小长度）滤波器的重叠变换可提供较好的能量压缩。本发明提供允许一重叠变换的非最小长度可逆滤波器。

精确重建系统的另一例子包括具有Z范围定义的二/六 (TS) 变换，

$$\begin{cases} H_0(Z) = \frac{1}{\sqrt{2}}(1+Z^{-1}) \\ H_1(Z) = \frac{1}{8\sqrt{2}}(-1-Z^{-1}+8Z^{-2}-8Z^{-3}+Z^{-4}+Z^{-5}) \end{cases}$$

在置换之后，该输出为

$$\hat{X}(Z) = 2Z^{-3}X(Z),$$

这是一精确重建变换。

该TS变换的合理的非标准变型包括：

$$\begin{cases} h_0(Z) = \frac{1}{2} (1+Z^{-1}) \\ h_1(Z) = \frac{1}{8} (-1-Z^{-1}+8Z^{-2}-8Z^{-3}+Z^{-4}+Z^{-5}) \end{cases}$$

如果该信号的六个取样为 $x(0)$ 、 $x(1)$ 、 \dots 、 $x(5)$ ，则由下式给出前三

个低通系数 $y_0(0)$ 、 $y_0(1)$ 、 $y_0(2)$ 和第一个高通系数 $y_1(0)$:

$$\begin{cases} y_0(0) = \left[(x(0)+x(1))/2 \right] \\ y_0(1) = \left[(x(2)+x(3))/2 \right] \\ y_0(2) = \left[(x(4)+x(5))/2 \right] \end{cases}$$

$$y_1(0) = \left[(-x(0)+x(1))+8(x(2)-x(3))+x(4)+x(5))/8 \right]$$

但是, 该TS变换的有理非标准变型的直接的实施是不可逆的。下面的例子表示该实施是局部地非可逆的。作为用于全向情况的一个例子可建立一较长的数列。由于完成对 $y_0(0)$ 和 $y_0(2)$ 的计算因为 $-(x(0)+x(1))+x(4)+x(5) \neq -y_0(0)+y_0(2)$, 这种使用局部信息的变换是不可逆的。

例如, 如果 $x(0)=1$, $x(1)=1$, $x(2)=3$, $x(3)=1$, $x(4)=1$, $x(5)=1$, 则

$$y_0(0) = \left[(1+1)/2 \right] = 1$$

$$y_0(1) = \left[(3+1)/2 \right] = 2$$

$$y_0(2) = \left[(1+1)/2 \right] = 1$$

$$y_1(0) = \left[[-(1+1)+8(3-1)+(1+1)]/8 \right] = \left[(-2+16+2)/8 \right] = 2$$

和, 如果 $x(0)=1, x(1)=2, x(2)=4, x(3)=1, x(4)=1, x(5)=1$, 则

$$\begin{cases} y_0(0) = \left\lfloor (1+1)/2 \right\rfloor = 1 \\ y_0(1) = \left\lfloor (4+1)/2 \right\rfloor = 2 \\ y_0(2) = \left\lfloor (1+1)/2 \right\rfloor = 1 \end{cases}$$

$$y_1(0) = \left\lfloor \left[-(1+2) + 8(4-1) + (1+1) \right] / 8 \right\rfloor = \left\lfloor (-3+24+2)/8 \right\rfloor = \left\lfloor 23/8 \right\rfloor = 2$$

因为对于输入 $x(0) \cdots x(5)$ 的两种不同设置 $y_0(0)$ 、 $y_0(1)$ 、 $y_0(2)$ 和 $y_1(0)$ 是相同的, 因为从输入的两种设置的局部信息中不可能确定给出的 $y_0(0)$ 、 \cdots 、 $y_1(0)$, 所以该变换是不可逆的(应注意的是使用来自所有系数的全局信息所提供的变换是不可逆的。)

现在来考虑称之为RTS变换的一可逆TS变换, 它提供了一种不同的高通滤波操作。

如果 $x(0)$ 、 $x(1)$ 、 $x(2)$ 、 $x(3)$ 、 $x(4)$ 、 $x(5)$ 是该信号的六个取样, 则由下式给出前三个低通系数 $y_0(0)$ 、 $y_0(1)$ 、 $y_0(2)$ 和第一个高通系数 $y_1(0)$,

$$y_0(0) = \left\lfloor (x(0)+x(1))/2 \right\rfloor$$

$$y_0(1) = \left\lfloor (x(2)+x(3))/2 \right\rfloor$$

$$y_0(2) = \left\lfloor (x(4)+x(5))/2 \right\rfloor$$

$$y_1(0) = \left\lfloor \left(- \left\lfloor (x(0)+x(1))/2 \right\rfloor + 4(x(2)-x(3)) + \left\lfloor (x(4)+x(5))/2 \right\rfloor + 2 \right) / 4 \right\rfloor$$

$$= \left\lfloor \frac{(-y_0(0) + 4(x(2) - x(3)) + y_0(2) + 2)}{4} \right\rfloor$$

因为

$$x(2) - x(3) = y_1(0) - \left\lfloor \frac{-(y_0(0) - y_0(2) + 2)}{4} \right\rfloor$$

所以 $x(2) - x(3)$ 是完全已知的。随着 $y_0(1) = \left\lfloor \frac{x(2) + x(3)}{2} \right\rfloor$ and $x(2) - x(3)$

和 $x(2) - x(3)$ 如上所规定，由于 $x(0) + x(1)$ 和 $x(0) - x(1)$ 的最低有效位是相同的所以 $x(2)$ 和 $x(3)$ 可被复原。明确地说，让

$$d(0) = x(2) - x(3) = y_1(0) - \left\lfloor \frac{(-Y_0(0) + y_0(2) + 2)}{4} \right\rfloor$$

$$x(2) = y_0(1) + \left\lfloor \frac{(d(0) + 1)}{2} \right\rfloor$$

$$x(3) = y_0(1) + \left\lfloor \frac{(d(0) - 1)}{2} \right\rfloor$$

在该RTS变换的一实施例，以及该S变换的一实施例中，为了提供准确的加法，一被实施的除以8的操作变为先除以2然后除以4。应注意数学式

$$\frac{1}{8} (-1 - Z^{-1} + 8Z^{-2} - 8Z^{-3} + Z^{-4} + Z^{-5} + 4)$$

和数学式

$$\frac{1}{4} \left(\frac{1}{2} (-1 - Z^{-1}) + 4(Z^{-2} - Z^{-3}) + \frac{1}{2} (Z^{-4} + Z^{-5}) + 2 \right)$$

当执行无限精确运算时是相同的。理由是当用整数运算实际实施时第二等式明显表示一可逆滤波器。结合图16和17描述了低通滤波器和高通滤波器的硬件实施的例子。

应注意的是S变换和RTS变换中该低通滤波器被实施以使输入信号x(n)的范围与输出信号y_o(n)是相同的。例如，如果该信号是-8比特图象，则低通滤波器的输出也是8比特的。由于在现有技术系统中输出信号的范围是大于输入信号范围的，所以对于低通滤波器被相继应用的一角锥系统来说是一重要的特性，从而使该滤波器难以连续应用。另外，该低通滤波器仅具有两个抽头，使得它为一非重叠滤波器。如像后面所述，这个特性对于硬件实施是重要的。

在一更普通的格式中，该可逆TS变换是由一低通和一高通的两个输出的表达式来确定的：

$$s(n) = \left[\frac{x(2n) + x(2n+1)}{2} \right]$$

$$d(n) = \left[\frac{\left[\frac{x(2n) + x(2n+1)}{2} \right] + 4(x(2n+2) - x(2n+3)) + \left[\frac{x(2n+4) + x(2n+5)}{2} \right]}{4} \right]$$

对于d(n)的表达式可以简化并利用S(n)来写出(由4的整除可通过向分子增加一个2来完成)其结果为:

$$\left\{ \begin{array}{l} s(n) = \left[\frac{x(2n) + x(2n+1)}{2} \right] \\ d(n) = x(2n+2) - (2n+3) + \left[\frac{-s(n) + s(n+2) + 2}{4} \right] \end{array} \right.$$

该可逆TS变换和反量为:

$$\left\{ \begin{array}{l} x(2n) = s(n) + \left[\frac{p(n)+1}{2} \right] \\ x(2n+1) = s(n) - \left[\frac{p(n)}{2} \right] \end{array} \right.$$

其中P(n)必须首先被计算为:

$$p(n) = d(n-1) - \left[\frac{-s(n-1) + s(n+1) + 2}{4} \right]$$

来自该低通滤波器的结果在高通滤波器中可被使用二次(在第一和第三项中)。因此,仅仅必须执行二次其它的加法以得到该高通滤波器的结果。

除了可逆变换之外,该TS变换还是有效的变换。该TS变换(类似S变换)具有不增长的平滑输出,即,如果该输入信号是b比特深度,则为平滑输出。这对于在下面部分中所定义的角维系统是有用的,在那里该平滑输出被进一步分解。由于该变换的实施足以整数完成的因而不存在系统误差,所以在一有损耗系统中所有误差均可通过量化来控制。

在参与一子波变换的四个滤波器中,由于该低通综合滤波器兼有被量化的系数并且还平滑人为因素所以该低通综合滤波器是最重要的。实际上在本发明中引导去选择用于低通综合滤波器的相对长(六抽头)和特别好特性的滤波器。应注意的是在-QMF系统中,仅存在有两个独立的滤波器。

在本发明本可使用许多被重叠的、非最小长度可逆滤波器。图3B示出了用于具有非重叠的最小长度可逆滤波器的变换系统的前向和反向的表示。例如,在本发明中可以使用下面的滤波器的类型。对于整数 $L \geq 2$,

$$d(0) = x(2 \left\lfloor \frac{L}{2} \right\rfloor + 1) - x(2 \left\lfloor \frac{L}{2} \right\rfloor + 1) + 1$$

和

$$y_0(0) = \left\lfloor \frac{(x(0) + x(1))}{2} \right\rfloor$$

$$y_0(1) = \left\lfloor \frac{(x(2) + x(3))}{2} \right\rfloor$$

$$y_0(L-1) = \left\lfloor \frac{(x(2 \left\lfloor \frac{L-1}{2} \right\rfloor) + x(2 \left\lfloor \frac{L-1}{2} \right\rfloor + 1))}{2} \right\rfloor$$

和

$$y_1(0) = \frac{\sum_{i=0}^{\lfloor L/2 \rfloor} a_i y_0(i) + b d(0) + \sum_{j=\lfloor L/2 \rfloor + 2}^{L-1} c_j y_0(j) + \left\lfloor \frac{k}{2} \right\rfloor}{k}$$

该高通滤波器的长度为 $2L$ 。如果 L 为奇数，则该滤波器可接近于一对称滤波器。如果 a_i 、 b 、 c_i 和 k 为整数并且 $K \leq b$ 时，则该滤波器是可逆的。如果 a_i 、 b 、 c_i 和 K 是2的幂(或2的幂的负数或计数)时，则该滤波器的实施可被简化。如果 $K=b$ (不管 a_i 和 c_i 的值)，则该高通滤波器 y_1 的输出范围最小。对于每个 a_i ，如果精确地存在一个 c_j ，这里 $a_i = -c_j$ ，则该高通滤波器将不响应于一固定输入。如果 $a_i = -c_j$ ，当 $j - (L-1) = i$ 时，则该滤波器可接近于一对称滤波器。

另外的有用的特性为

$$\sum_{i=0}^{\lfloor L/2 \rfloor} \left[(a_i)(2i)^m + (a_i)(2i+1)^m \right] + (b)(2(\lfloor L/2 \rfloor + 1))^m - (b)(2(\lfloor L/2 \rfloor + 1) + 1)^m + \sum_{j=\lfloor L/2 \rfloor + 2}^{L-1} \left[c_j(2j)^m + c_j(2j+1)^m \right] = 0$$

当 $m=1$ 时该高通滤波器不响于一线性地变化的输入而当 $m=2$ 时不响应于二次方变化的输入，其中 m 为力矩条件。这个特性是RTS变换具有比S变换更好的能量压缩的基本原因。

当滤波器必须满足用于可逆性的最小约束时，对于不同的应用，滤波器可被用于不满足，满足某些或全部满足其它的特性。在某些实施例中，使用以下例子高通滤波器中的一个，这些滤波器以符号列于表中，表中列出了该滤波器合理变型的整数系数，以避免使本发明难以理解。

```

1 1 -4 -4 16 -16 4 4 -1 -1
1 1 -3 -3 8 -8 3 3 -1 -1
-1 -1 0 0 16 -16 0 0 1 1
-1 -1 4 4 -16 -16 256 -256 16 16 -4 -4 1 1
3 3 -22 -22 128 -128 22 22 -3 -3

```

最后的滤波器称之为TT滤波器，它具有不响应于三次方增长函数的特性。应注意的是因为 $22=16+2\times 3$ 和 $3=2+1$ ，所以这个滤波器可以用总共7次相加和相减来实施。

在一实施例中，这些滤波器可被组合并应用于一组，这样以一种操作执行水平和垂直通过。图4示出了执行该组合操作的滤波器。配置(a)示出了两个分离的1-D可逆滤波器的使用，对于每个通过的1-D可逆滤波器包括有一个1-D滤波器和1-D舍入(rounding)。配置(b)示出了一个1-D滤波器401，后面跟着另一个1-D滤波器402，末尾带有一舍入操作403。这种配置产生允许更好的舍入的更为精确的结果。

对于滤波器的所需要的精确的可逆要求可通过下面的表明加以缓解。高通系数以某种顺序被编码和译码。相应于前面的被译码的高通系数的象素值是精确地已知，这样它们可利用在当前的高通滤波器中。例如，当一光栅顺序被使用时可利用下面的滤波器。

$$H_1(Z) = \left[\frac{1}{4} \left(\left[-\frac{1}{2}(1+Z^{-1}) \right] + \left[\frac{1}{2}(8(Z^{-2}-Z^{-3})+(Z^{-4}+Z^{-5})) \right] + 2 \right) \right]$$

不需使用一单一固定高通滤波器。可使用自适应滤波器或可使用多级滤波器。在多级滤波器之中采用或选择的所使用数据必须是限于在特殊的反向滤波器操作之前在该译码器中是可用的数据。

使用多级滤波器的一种方式是在渐进地处理该高通系数。另外的高通滤波操作($Y1(0)$, $Y1(2)$, $Y1(4)$, ...)可用一诸如RTS高通滤波器之类的一可逆滤波器首先被处理。因为输入到该滤波器的重叠部分的精确值是已知的, 所以剩余处理($Y1(1)$, $Y1(3)$, $Y1(5)$, ...)可以使用高达六抽头的非可逆滤波器。例如, 任何如下的滤波器都可被使用。

$$\begin{array}{cccccc} -1 & 3 & -3 & 1 & & \\ -1 & 4 & -4 & 1 & & \\ -3 & 8 & -8 & 3 & & \\ 1 & -5 & 10 & -10 & 5 & -1 \\ 1 & -4 & 8 & -8 & 4 & -1 \end{array}$$

应注意的是在某些实施例中不使用QMF滤波器。

在某些实施例中, 高通滤波器可由一预测/插入操作来替代。一预测器/插入器可预言在一特定的预测/插入操作之前使用在译码器中的任何有用数据的一对输入之间的差异。将该被预言的差异从输入的实际差异中减去并输出其结果。在一实施例中, 现有技术预言方法被使用在DPCM中, 渐进编码或空间范围编码被使用。

在一实施例中可使用诸如形态学的滤波器(即, 中点滤波器)之类的非线性滤波器。在一实施例中, 该1, 1滤用来连接一用于高通的不同的滤波器。这样一种滤波器系统必须能够传送两个象素之间的差异。基于任何数据该译码器具有可预测出其差异是什么的预言。一非线性形态学滤波器可用作估测。本发明计算在该窗口的因果例(Causal Side)所使用的实际象素的一象素的周围的中点, 并将其入该滤波器中。在该滤波器的非因果侧, 低通系数被用来代替象素值。

二维的子波分解

利用本发明的低通和高通滤波器, 执行多分辨分解。分解的级数是变量并且可为任何数, 但当前的分解级数为从2至5级。

在诸如图象的二维数据上执行变换的最普通方式是分别地应用一维滤波器来实现的,即,沿行然后沿列来变换。该分解的第一级导致系数的四个不同区域,称之为LL、HL、LH和HH。代表相应于应用平滑和细节滤波器的底(L)和高(H)的字母分别由以上规定。因此,LL区包括在行和列两个方向上来自该平滑滤波器的系数。以如图5A-5D的格式来安置子波系数是普通常事。

在子波分解中的每个子组可被进一步被分解。最普通的方式是仅仅进一步分解该LL子组,但这可进行若干次。这样一种多次分解被称之为角锥分解(图5A-5D)。名称LL、LH、HL、HH和分解电平数表示每一分解。应注意的是本发明的任一滤波器(S或TS)进行的角锥分解不增加该系数尺寸。

例如,如果可逆子波变换被递归地应用于一图象,则分解操作的第一级将使图象有良好的细节,或分辨率。在第一分解级,该图象被分解成四个子图象(或子区)。每个子区表示一空间频率的区域。第一级子区标为LLO、LHO、HLO和HHO。分解该原始图象的处理包括在水平和垂直方向上的二次子取样,这就如象该输入具有该图象的象素(或系数)的许多系数一样,如图5A所示那样第一组子区LLO、LHO、HLO和HHO的每一个具有四分之一。

子区LLO包括同时发生的低频水平和低频垂直信息。在这个子区中典型地集中有该图象能量的一大的部分。子区LHO包括低频水平和高频垂直信息(即,水平边缘信息)。子区HLO包括高频水平信息和低频垂直信息(即,垂直边缘信息)。子区HHO包括高频水平信息和高频垂直信息(即,结构或对角线边缘信息)。

由分解前述级的低频LL子区而产生随后的第二、第三和第四较低分解级的每一个。如图5B所示,第一级的子区LLO被分解以产生适中细节第二级的子区LL1、LH1、HL1和HH1。类似地,如图5C所示,子区LL1被分解以产生第三级的粗的细节的子区LL2、LH2、HL2和HH2。另外,如图5D所示,子区LL2被分解以产生第四级的较粗的细节的子区LL3、LH3、HL3和HH3。由于是二次子取样,所示每个第二子区是原始图象尺寸的十六分之一。在这个级上的每一子样(即,象素)表示在相同位置上的原始图象中的适中细节。类似地,每个第三电平子区是原始图象尺寸的1/64。在这个电平上的每一象素相应于在相同位置上在该原始图象中相对地粗的细节。另外,每个第四电平子区是该原始图象的尺寸的1/256。

由于子取样而使被分解的图象实际上小于原始图象,因而用来存贮原始图

象的相同存贮器可被用来存贮所有被分解的子区。换句话说,原始图象和被分解子区LL0和LL1被抛弃并且在一第三电平分解中不被存贮。

虽然只示出了四个子区域分解级,但可根据一实际系统的需要而发展出更多的级。另外,也可以定义诸如DCT之类的其它变换或线性被隔开的子区、不同的母体关系。

在一角锥分解中存在有一固有的和有用的用于子波系数的树结构。应注意的是相应于最后的分解级存在有一单一的LL子信息组。另一方面,随着级数存在有一许多LH、HL和HH区。树结构规定了在较低分辨力上在一频段上的一系数对在相同频段上一系数的母体,涉及相同空间位置。图6示出了两个连续级之间的母体关系。

参见图6,在A上的系数直接与B、C和D有母体关系,而且还与B、C和D有母体关系的系数也有母体关系。详细地说,B与围绕E的四个系数有母体关系和围绕H的16个系数有母体关系等等。

多分辨力分解的过程可利用如图7中所示的一滤波系统来执行。一表示具有长度L的一维信号的输入信号在通过单元703和705进行二次取样之前由滤波器单元701和702进行低通和高通滤除。来自单元703的被子取样的输出信号在分别通过单元707和708被二次子取样之前由单元705和706低通和高通滤除。在单元707和708的各自输出端上呈现子区分量L和H。类似地,来自单元705的输出信号在分别由单元711和712子取样之前由单元709和710低通和高通滤除。在单元711和712各自的输出端上呈现子区分量L和H。如上所述,使用于子区分解的本发明的一实施例的该滤波器是用来将水平和垂直频段分离成低频和高频段的数字正交镜滤波器。

图8示出二维、二级变换。图9还示出了使用诸如在图16和17中所示的一维滤波器来实施的二维、二级变换。该一维滤波器用于各种其它の場合以避免由于子取样所必须进行的计算。在一实施例中,一维滤波器共享低通和高通之间计算的计算结果。

因此,本发明提供了一种用于压缩和解压的系统,在该系统中使用了非最小长度、重叠的可逆滤波器。图10是这样一种系统的一实施例的方框图。参见图10,开始就执行分层分解。该分层分解的结果被送到用于压缩的压缩器。所执行的压缩包括向量量化、标量量化、零运行长度编码、哈夫曼(Huffman)编

码、Tunstall等。该压缩器压缩数据的输出表示该原始输入数据的一被压缩的文本。一解压器可在未来的某时接收该数据并解压该数据。本发明利用非最小类型、重叠可逆滤波器执行一仅向分解以产生该原始数据的重建文本。应注意的是该非最小长度、重叠可逆滤波器包括有非S变换滤波器。

本发明的可逆子波滤波器还可用于如图11所示例子的分析和增强系统。参见图11,在使用非最小长度、重叠可逆子波滤波器的输出数据上执行分层分解。该分析单元接收由该滤波器产生的系数并将它们分类到决定之中,即,仅提取相关的信息而不是编码所有的系数。例如,在一文件归档系统中,可能仅利用最粗的低通子区来识别空白页。另一个例子是仅使用来自一特定子区的高通信息在文本图象和自然景色图象之间进行识别。该分层分解可用于寄存多个图象,这样首先由粗的子区来进行粗的寄存。在另一实施例中,随着反向分解该系数经受放大或滤除。利用分层分解可执行削尖、边缘增温、噪声控制等。因此,本发明提供一种用于连接时间/空间和频频范围和滤除/增强系统的子波变换。

系数和比特平面的排序和造型

在本发明中,作为子波分解的结果所产生的系数是熵编码。在本发明中,该系数最初地经受嵌入编码,在嵌入编码中该系数按视觉地有效顺序排序,更为一般地,相应于某误差度量(即,失真度量)来排序。误差或失真度量包括峰值误差、和均方误差(MSE)。另外,相对于数据库查询和定向(垂直、水平、对角线等),可被执行的排序优先给出比特有效位的空间位置。

执行该数据的排序的产生代码流的嵌入量化。在本发明中,所使用的二个排序系统是:第一个用于对系数排序,第二个用于对在一系数之中的二进制值排序。本发明的排序产生一在由一二进制熵编码器编码之后的比特流。

在一实施例中,该系数排序和造型包括M-ary编码。在另一实施例中,它可仅由区域嵌入而替代由比特嵌入。另外,对于无损耗编码或单一品质有损耗编码(即,在该编码器中被特定的量化),在该系数排序和造型中可使用非嵌入编码。

编码单元

在本发明中,以树型设置为矩形的编码单元对图象等进行独自地编码。该

编码单元表示被编码数据的最小单元(虽然存在有允许被译码的局部编码单元的量化选择)。在一编码单元中的曾经在存储器中被缓冲的所有数据对该编码器是可用的。

编码单元的选择是依从的实施的。该编码单元可作为整个图象(或其它数据组)或发明的单一树或在它们之间的任何矩形而被规定。在一实施例中,一编码单元的选择可需在压缩效能和存贮器使用之间取折衷的方案。

在一实施例中,在一编码单元内的所有系数在随机存取存贮器中是可用的。因为在一编码单元内的所有系数在随机存取存贮器中是可用的,所以在一编码单元中在该系数之间的嵌入顺序可以是任何任意的顺序。这个顺序对编码器和译码器来说是已知的。但由于熵编码器是相对于这个顺序,在压缩中该顺序有一有效的碰撞,因而应慎重选择。下面描述了一特定排序的实施例。

造型

在本发明中,接合空间/频率造型压缩和嵌入编码系统用来对由本发明的子波变换所产生的系数编码。该接合空间/频率造型具有已知频率频段和邻接象素(或数据)二者的优点。该接合空间/频率造型的一实施例这里称之为水平线造型。

该数据开始以符号数值格式被格式化,然后该数据基于有效位而被存贮。在另一实施例中,为了进一步减小存贮器工作空间,该系数可以数量/假数方式而不足以符号/数量方式被存贮。

在以相对于给出的有效度量存贮该数据之后,对该数据编码。

假设一数字信号 $X(n)$,对于每个 $X(n)$ 用精度 R 比特表示,然后本发明的嵌入编码对该信号的每个 $X(n)$ 的最高有效位比特(或比特)编码,然后再对下一个有效位比特(或比特)编码等等。例如,在可视的规定顺序的情况中,一需要更好品质的象素比在沿拐角或接近该边缘(诸如某些医学图象)的中心处可受到编码,这样在该边界象素的高阶比特之前该中心象素的低阶比特可被编码。

比特有效位表示

在一实施例中,用于在一系数内的二进制值的嵌入顺序是利用比特平面。该系数以比特有效位表示来表达。比特有效位是一记号数量表示,其中该记号

比特而不是最高有效位比特(MSB)是由第一非零数量比特编码。

在以比特有效位形式表示的一数中存在有三种比特类型：标题、尾部和记号。标题比特是从MSB到第一非零数量比特加第一非零比特的所有零比特。在比特平面出现的第一非零数量比特规定该系数的有效位。在第一非零数量比特到LSB之后是尾部比特。该记号比特简单表示该记号。带有作为MSB的非零比特的数仅有一个标题比特。零系数没有尾部或记号比特。

在该值为非负整数的情况中，例如相对于象素的亮度所呈现的值的状况中，该顺序可被用作比特平面顺序(即，从最高有效位到最低有效位比特平面)。在允许两个解码负整数的实施例中该记号比特的嵌入顺序是与该整数的绝对值的第一非零比特相同的。因此，直至一非零比特被编码为止不考虑该记号比较。例如，利用记号数量标记法，数-7的16比特为：

1000000000000111

在一比特平面基底上，前十二个决策为“无意义”或零。第1个1比特出现在该第十三个决策上。接着，记号比特(“页”)将被编码。在该记号被编码之后，尾部比特被处理。第十五和第十六个决策均为“1”。

字数校直

类似于FFT或DCT在不同子组中的系数表示不同的频率。在该比特平面编码之前相对于相互之间通过校直系数来执行该量化。少量密集量化系数将朝向较近的比特平面(即向左位移)校直。因此，如果该流被舍住，则这些系数将具有比更多密集量化系数更多的比特来规定它们。

在一实施例中，为了在SNR或MSE项中的最佳比率失真特性而校直该系数。另外，该校直可允许该系数数据的-psychovisual量化。在该图象品质的演变上(或换句话说是在速率-失真曲线上)该校直具有有效的碰撞，而在无损耗系统的最后压缩比上具有可忽略不计的碰撞。

在二级TS变换分解中来自每个象素具有b比特的输入图象的各种系数的比特泽度由图12示出。为了校直该系数，将1HH系数尺寸用作参考，并且移位是相对这个尺寸给出的。表1示出了这种校直过程的一个例子。

表1 系数校正的例子, $b=8$

| | | | | | | | | | | | | | | |
|-----------|----|---|----|---|---|---|---|---|---|---|---|---|---|---|
| 1-HH | 参考 | | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| x-HL或x-LH | 左3 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | |
| x-HL或x-LH | 左2 | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| x-HL或x-LH | 左1 | | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| x-HL或x-LH | 无 | | | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| x-HL或x-LH | 右1 | | | | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

应注意的是符号比不是MSB并且由第1尾部比特编码。重要的是校正简单地控制该顺序, 该比特被送至熵编码器。不执行实际的填充、移位、存贮或额外的零比特的编码。

上下文模式

下面介绍本发明所使用的水平线上下文模式的一实施例。该模式基于该系数的空间和频谱的相关性而使用在一编码单元内的比特。该相邻系数的可用的二进制数值和母体系数可用于建立上下文。但是该上下文是用来可译码的原因并且为了适应系数是很小的数量。

熵编码

在一实施例中, 由本发明执行的熵编码是由二进制熵编码器执行的。在一实施例中, 熵编码器104包括一Q编码器、一有限状态机器编码器、一高速并行

编码器。一单一编码器可用来产生一单一输出代码流。另外，多个(实际的或虚拟的)编码器可用于产生多个(实际的或虚拟的)数据流。

在一实际例中,本发明的该二进制熵编码器包括-Q编码器。对于该Q编码器的更多的情况请见Pennebaker, W. B., et al "An Overview of the Basic Principles of the Q-Coder Adaptive Binary Arithmetic", IBM Journal of Research and Development, Vol. 32, pg. 717-26, 1988。在另一个实施例中,一二进制熵编码器使用一QM编码器(这种编码器是公知的)和高效二进制熵编码器。在具有很高概率变形的比特中它是特别有效的。该QM编码器是用于JPEG和JBIG标准中。

该二进制熵编码器包括一有限状态机器(FSM)编码器。这样一种编码器提供来自一概率的转换和提供一输出到被压缩的比特流。在一实施例中,一有限状态机器编码器利用表实施对译码器和编码器的查找。各种概率估算方法都可用于这种有限状态机器编码器。对于概率接近于0.5来说压缩是极好的。对于高地变形的概率的压缩取决于所使用的查找表的尺寸。如像该QM编码器,因为该决策是以出现的顺序编码的,所以与被嵌入的比特流一起它是有用的。因为该输出是由一查找表规定的,所以不会存在用于“转移”问题的可能性。实际上,与Q和QM编码器不同,在被压缩的输出比特的编码和形成之间存在有一最大的延迟。在一实施例中,本发明的有限状态机器编码器包括有一在美国专利NO. 5,272,478(名称为“用于熵编码的方法和装置”公开日为1993年12月21日)中所指出的B编码器。

在一实施例中,本发明的该二进制熵编码器包括一高速并行编码器。QM编码器和FSM编码器一次需要被编码或被译码的1比特。该高速并行编码器并行处理几个比特。在一实施例中,该高速并行编码器以VLSI硬件或多处理器计算机实施而不会牺牲压缩特性。在本发明中可被使用的高速并行编码器的一个实施例在美国专利NO 5,381,145(名称为“用于数据的并行译码和编码的方法和装置”,公开日为1995年1月10日)中被披露。

大多数的高效二进制熵编码器通过基本的反馈环路来限制速率。该编码器的输出是多重可变长度编码数据流。具有这种处理类型的一个问题是如何在一单一通道上传送该数据。在美国专利NO. 5,381,145中所披露的该高速并行编码器与交叉存取这些编码数据的方法一起解决这个问题。

在本发明中使用的上下文的大多数是固定概率的，构成一有限状态机器编码器，例如B编码器特别地有用。当一系统使用的概率接近0.5时，上述的高速并行编码器和有限状态机器编码器具有比Q编码的更高的效率。因此，它们在用本发明的上下文模式时具有潜在压缩优点。

本发明的编码和译码处理

下面结合图13-15的流程图说明本发明的编码和译码的一实施例。这些过程可由硬件或所带软件来完成。在任一种情况中，参照所制作的处理逻辑可表示另一种情况。

图13示出了本发明的编码过程的一实施例。参见图13，通过所具有的处理逻辑对于一编码单元获取一掺入数据开始编码处理(处理框1301)。接着，处理逻辑将可逆滤波器加到该编码单元的输入数据单元(处理框1302)。

判断是否另一分解级是所希望的(处理框1303)。如果是，则处理逻辑将该可逆滤波器加到LL系数上(处理框1304)，并且该处理环返回并在处理框1303中继续进行处理。如果另一个分解级不是所希望的，则处理在处理框1305中继续进行处理，在那里处理逻辑变换该系数为符号/数量形式。

在变换该系数为符号/数量形式之后，一比特平面变量S被设置为最高有效位比特平面(处理框1306)。然后，该处理逻辑随意地初始化该熵编码器(处理框1307)。

一旦该熵编码器被初始化，处理逻辑用该上下文模式和熵编码的该比特模拟每一系数的每一比特(处理框1308)。在熵编码该比特之后，该数据被传送或存贮(处理框1309)。

之后，判断在该图象中是否还存在有编码单元(处理框1310)。如果还有编码单元，则处理继续到处理框1301。另一方面，如果没有更多的处理单元，则处理结束。

图14示出了本发有译码处理的一实施例。参见图14，通过处理逻辑对于一编码单元复原编码数据而开始处理(处理框1401)。接着可逆S被置为最高有效位比特平面(处理框1402)。在设置该比特平面可逆S为最高有效位比特平面之后，处理逻辑随意地初始化该熵编码器(处理框1403)。

在该熵编码器被初始化后，处理逻辑将每一系数的初始值设置为零(处理

框1404)。然后处理逻辑用上下文模式和熵译码器模拟每一系数的每一比特(处理框1405)并且将系数转换成用于滤除的适当形式(处理框1406)。该转换可以从比特有效位转换为2的补码形式。之后,处理逻辑从分解的最高级将一反向滤波器加到该系数起始(处理框1407)。

判断是否所有的级被反向滤除(处理框1408)。如果所有的级未被反向滤波,则处理逻辑从下一个分解的最高级将反向滤波器加到该系数(处理框1409),并且在处理框1408继续处理。如果所有的级均被反向滤除,则在处理框1410上继续处理,在那里重建数据被存贮或被传送。在存贮该被传送的重建数据之后,判断是否存在有更多的编码单元(处理框1411)。如果存在有更多的编码单元,处理环路返回并在处理框1401上继续处理,在那里处理被反复进行。如果不存在有更多的编码单元,则处理结束。

图15示出了根据本发明用于造型的处理的一实施例。参见图15,通过向第一系数设置一系数变量而开始对于造型比特的处理(处理框1501)。然后判断是否 $1C1 > 2^S$ 。如果是,处理继续在处理框1503上进行,在那里利用对于尾部比特的该模式,处理逻辑编码系数C的比特S并且继续在处理框1508处理。用于尾部比特的模式可以是一固定的(非自适应的)模式。如果C不大于 2^S ,则处理继续在处理框1504进行处理,在那里处理逻辑提供一用于标题比特的样板(即,初始为0和第1为“1”比特)。在提供该样板之后,处理逻辑编码系数C的比特S(处理框1505)。可能的样板如图26A-C所示。如图19A和19B中所示,样板可用LUTs来实施。

接着,判断是否系数C的比特S是通(on)(处理框1506)。如果系数C的比特S没有on,则处理继续在处理框1508进行处理。另一方面,如果系数C的比特X是on,则处理继续在处理框1507进行处理,在那里处理逻辑编码该符号比特。此后,处理继续在处理框1508进行。

在处理框1508,判断是否系数C是最后的系数。如果系数C不是最后的系数,则处理继续在处理框1509进行,在那里系数变量C被置为下一个系数并且处理继续在处理框1502进行。另一方面,如果系数是最后系数,则处理在处理框1510进行处理,在那里判断是否S是最后比特平面:如果S不是最后比特平面,则比特平面变量S被递减1(处理框1511),并且处理继续在处理框1501进行。如果S是最后比特平面,则处理结束。

本发明另外的实施例

本发明可以由硬件和/或软件来实施。本发明的一硬件实施需要子波滤波器，存贮器/数据流管理以提供用于该滤波器的数据，一上下文模式以控制本发明的嵌入编码，存贮器/数据流管理以提供用于该上下文模式的数据，和一二进制熵编码器。

子波滤波器

图16示出了本发明的前向子波滤波器的一实施例。图16中所示的子波滤波器容纳有4个16比特²的补码输入象素，以 $x(2)$ - $x(5)$ 表示。

参看图16，两抽头“11”低通滤波器使用一个16比特加法器1601。其输出被分别称之为S和D。使同移位一个比特方框1603使加法器的输出(S)被舍位成16比特。通过将其17比特输入向右移位一个比特，移位一个比特方框1603执行除2功能。

6抽头高通滤波器需要对 $-S_0+4D_1+S_2$ 进行计算。利用接收移位一个比特方框1603的输出和 $Y_0(0)$ 的减法器1605计算函数 S_2-S_0 。利用减法器1602、移位2个比特方框1604和加法器1608计算 $4D_1$ 项。利用称位两个比特方框1604使由16比特减法器1602产生的输出向左移位两个位置，借此以使它的输出被乘以4。方框1604的输出由加法器1608加到2。注意，由于移位了两个比特，所以，加法器1608可以通过布线加以替换。把加法器的 $4D_1$ 输出加到减法器1605输出端上的工作是由20比特加法器1606执行的。使用移位2个比特方框1607将加法器1606的输出舍位为18比特，通过将其20比特输入向右移位两个比特，移位两个比特方框1607执行除4的功能。

因此，(用于存贮暂时结果的非计数寄存器)所需的总的计算硬件是：

- 1 @ 16比特加法器，
- 2 @ 16比特减法器，
- 3 @ 19比特加法器。

注意，移位(和加法器1608)是利用诸如不需要任何逻辑关系的布线执行的。

在另一个实施例中，对于输入尺寸N来讲，可以使用一个N比特加法器，两个N比特减法器和一个(N+3)比特加法器。

由于这些加法器和减法器的硬件成本极低，因此如果需要的话，还可以同

时使用一个滤波器。

注意, 取代减去 $x(3)$ 和 $x(2)$, 可以计算 $x(4)-x(5)$ 并保留其结果直到以后作为用于下一次移位或滤波器应用的 $x(2)-x(3)$ 需要时为止。两个正向滤波器(或下面将要描述的反相滤波器)可以形成流水线配置, 以实现较高的吞吐量。

反相子波滤波器示于图17。利用减法器1701减去 $Y_0(0)$ 和 $Y_0(2)$ 的输入。利用加法器1709将2加到减法器1701的输出上。并利用移位两个比特方框1702 将相加的结果向右移位两个比特。这就使减法器的输出被除以4。减法是在移位2个比特方框1704的输出和 $Y_1(6)$ 输入之间进行的。输入 $Y_0(1)$ 被移位一个比特方框1703向左移位一个比特, 借此使该输入乘以2。在 $Y_0(1)$ 被移位了一个比特(被乘以2)以后, 移位值的LSB是取自减法器1704输出的LSB并与来自移位一个比特方框1703的16比特输出相组合以形成加法器1705和减法器1706的输入。加法器1705和减法器1706的其它输入是减法器1704的输出。加法器1705和减法器1706的输出。加法器1705和减法器1706的输出接着被进行箝位。

可以选择两种箝位操作。在这两种情况下, 20比特值被移位一个比特(9除以2)成为一个19比特值。对于仅执行无损耗压缩的系统来讲, 可以输出最低有效16比特(其余3比特被忽略掉)。在有损耗(或有损耗/无损耗)系统中, 如果19比特值是负的, 那么它被置为零, 如果它大于 $2^{16}-1$, 则将被设置成 $2^{16}-1$, 否则可以输出最低有效16比特。

对于尺寸为N个比特的输入而言, 可以使用一个N比特减法器, 一个(N+1)比特加法器、一个(N+2)比特加法器、一个(N+3)比特加法器和一个(N+3)比特减法器, 且箝位单元输出N个比特。

在子波变换的一个实施例中, 在变换中使用了Monte Carlo除法, 其中使用了伪随机发生器, 并根据它的输出结果, 变换操作的结果或上或下来回滚动。只要译码器意识到这种滚动一直在执行(即使用相同的随机发生器, 在相同的点开始), 就可以使用这种工具。

用于与子波滤波器相关的数据流管理的存储器

就用于本发明子波滤波器的存储器和数据流管理而言, 对于满帧可以装入存储器的图象存储器/数据流管理不是一件困难的事情。即使是 1024×1024 的16比特医学图象(例如24兆字节规模), 对于许多应用来讲, 要求满帧缓冲器也

是合理的。对于较大的图象(例如A4, 400DPI4-彩色图象的规模大约是50兆字节),也希望使用有限量行缓冲存贮器的子波变换。

注意,对于本发明的执行一通过系统来讲,满帧缓冲器是不需要的。由于这一点,所需存贮器可以被减少100倍(和使用与大图象相关的满帧缓冲器相比)。下面将描述本发明的一通过系统。

存储在滤波缓冲器中的数据是一系列将被进行嵌入编码和二进制熵编码的系数。嵌入编码使用上下文模型去配合水平(norizon)编码的使用并提供适当序列的数据。所述上下文模型和存贮的管理模式一起工作。对于使用满帧缓冲器的系统来讲,以适当的序列提供数据是不困难的。

对于使用有限量工作区存贮器的系统来讲,使用不同高度的变换以减少用于存贮所需的存贮器工作区行的数量。因此,如果遇到较宽的图象,它就会被有效地压缩在所分配的工作区存贮器内。例如,可以垂直地使用S变换以减少行数。

存贮器需要对光栅(vaster)数据进行缓冲,从而能够执行子波变换。在某些应用中,使这个存贮器最小化对于减少成本来讲是非常重要的。下面来描述实现这一点的技术。

这里所描述的子波2-D变换的一个实施例是为一通执行和受限存贮器应用而设计的。在一个实施例中,为实现熵解压而使用的子波变换仅仅是TS和S变换。在这个实施例中,具有4级相互独立的熵解压。在一个实施例中,使用S和TS变换执行4级解压。在一个实施例中,在水平(行方向)解压过程中,完全使用TS变换,即形成TS-TS-TS-TS的水平解压。在垂直(列方向)解压过程中,以TS-TS-S-S形式执行S变换和TS变换。除对存贮器应用方面有重要影响以外,都使用低成本的S变换来取代两个TS变换。为通常情况一样,水平和垂直变换被交替使用(图20)。

注意,S和TS变换的任意组合都可被用于执行水平和垂直变换。注意,虽然变换的顺序可以是混合的,但解码器必须意识到这个顺序并必须以全可逆的反相顺序执行相反的操作。

系数树

在熵系统中,可以使用树结构将系数分成多个组。每个树的根是一个纯的

低通系数。图6示出了变换后图象一个纯低通系数的树结构。对于诸如一个图象的两维信号来讲，树根具有3个“孩子”，其余节点中的每一个具有4个“孩子”。该树不被分层地限定为两维信号。例如，对于一维信号来讲，树根具有一个“孩子”，而非树根节点中的每一个具有两个“孩子”。较高的维是由一维和二维情况得出的。

从示于图7-9的滤波器的工作情况来看，其树结构是很明显的。利用子取样的一对滤波器的工作使得在前所描述的系数被涉及。

在一个实施例中，系数是在一个比特有效或比特平面嵌入系统中被编码。由于该系数是从最高比特平面向最低比特平面进行编码的，所以必须确定在数据中比特平面的数量。在本发明中，这是通过获得根据数据计算的系数值的幅值上限或得自图象深度或滤波器系数的幅值上限而实现的。例如，如果上限是149，那么就有8个有效比特和8个比特平面。由于软件中的速度问题，可以不使用比特平面编码。在另外一个实施例中，只有当系数变得和二进制数一样有效时，才使用比特平面编码。

在一个实施例中，本发明的水平上下文模型包括被馈入给二进制熵编码器的子波系数比特有效嵌入编码。

上下文模型的替换

一旦解压被完成且数据系数被排序，本发明的上下文模型被用于对这些系数进行编码。有各种各样可以被使用的上下文模型，其决定以空间位置、级和/或比特位置为条件。该决定还可以在空间位置、级和/或比特位置方向与当前数据接近的先前译码的数据为条件。

下面是某些例子。最高有效尾部比特(和因此是最容易予侧的)不同于尾部比特其余位的上下文。标题比特可以由与在同一变换级中空间接近在前系数有拳同一比特决定。类似地，用于有效系数的符号可以由在同一级中空间接近在前系数的符号或本源系数的符号来决定。

当对具有空间或多个分解结构的图象进行压缩时，上下文模型的改进可能是非常重要的行绘图或文本的灰度图象是具有这两种结构的图象的例子。对于使用规定峰值误差去压缩那些已经被压缩和解压的文件来讲，这种改进也是非常重要的。

当利用软件执行本发明时，由于所述比特需要决定状态(例如每个首位)，所以需要大量的时间去获得与上下文相关的比特。在本发明的一个实施例中，通过使用查询表(LUT)可以加速软件的执行。这避免了与用作上下文的北(N)西北(NW)、西(W)和西南(SW)等象素相关的单独比特的提取操作。

图19A和19B示出了本发明的与标题比特状态相关的状态机构。参看图19A，用于新母体的LUT1901被耦合到编码/解码方框1902。LUT1901被耦合以接收表示母体的比特、表示以上(NE)系数，当前系数(E)和表示以下(S)系数的比特，在一个实施例中，本源输入的当前输入各自包括两个比特。到LUT1901的其它输入包括作为反馈的来自LUT1901所有或部份的上下文输出和编码/解码方框1902的输出。在一个实施例中，作为LUT1901上下文的10个比特输出中的8个被反馈到LUT1901的输入端上。

由于NE、E和S代表包括有与各种比特平面相关的信息的样板信息的前沿，所以它们要被使用。注意，除了南(S)系数以外，西南(SE)系数也可以使用。

在一个实施例中，如果样板位于编码单元以外，那么可以利用来自当前象素的比特代替外部状态比特。

图19B示出了使用与同一母体相关的LUT的状态机构的状态。在这种情况下，全部上下文被作为输入反馈给LUT1903。

在利用光栅顺序对数据进行处理的地方由于被用于产生最后上下文的同一个存贮器不再被重新装载，所以，使用LUT减少了存贮器存取的数量。

为了减少LUT存贮器的规模，还可以利用仅处理其它条件的LUT的输出通过ORing分别处理母体状态。

可以用一个稍大的LUT表来最大程度地提供与下一个比特平面相关的状态。另外一个较小的表用来提供来自当前上下文LUT的状态信息并使之与来自下一个比特平面的新得到的数据相结合。如果在此时对一个树进行编码的话，这将是非常有用的。

如上面就本发明所述，“系数”可以被规定成表示决定因素为1的变换。在这种情况下，通过当低概率是零时保留用于对低概率事件进行编码所使用的空间使得代码空间不会被浪费。但是，8比特系数仍被输入并且产生一个8比特系数一个9比特系数。因此仍可以改善效率。所增加的低效率是由于可通过系数空间的旋转而致。

应当注意，变换操作的某些结果唯一的识别在计算中以使用的量，这发生在这些结果接近这些可能结果的范围的边界时。这样的一个例子示于图18，其中， u 代表低通而 v 代表高通。由于 u 和 v 的值不是独立的，所以，在考虑到连接信息的情况下很容易对这些易进行熵编码。为图18所示，对于大多数低通值来讲，这是由于用于高通的某些代码空间未被使用的缘故。在很多应用中，由于指定给这些不可能对的概率极低，所以，这也有一点优异。但是，在某些应用中，这可能有相当大的好处。为了加速操作，在LH, HL和HH系数之前应当传送更多的LL系数。这可以使得划界更容易。

在某些实施例中，在每个编码单元被编码以后，一切都复位，当时第二个单元进行编码时，统计量和概率被复位。在一个实施例中，某些或所有的统计量都被保留。然后这些统计量被作为一个后边编码单元开始编码时的最初统计量在一个实施例中，该统计量可以被保留在第一或先前编码单元编码过程中的某一预定点处。例如，在第三比特平面编码之后，对当前编码单元进行编码所使用的统计量被保留并且用作对下一个编码单元或以后的编码单元开始编码时的统计量。在另一个实施例中，评价所有图象的分类并确定统计量的硬编码组。然后，使用这些硬编码统计量作为缺席规划执行编码。在另外一个实施例中统计量被保留以用于每一个比特平面，从而当对处于另一个之中的类似比特平面进行编码时使用这些统计量。

在一个实施例中，在第一个一比特之前没有编码。在系数中发生第一个一比特时，对符号进行编码。虽然首位是一个从属的图象/区域，但对于不同的图象和区域来讲，尾部比特是更加唯一的。根据尾部比特距离最初一个比特(右标题比特内)有多远，可以使用某些概率分类对尾部内的比特进行编码。在一个实施例中，利用包括0.7在内的概率分类对系数内的第一个尾部比特进行编码，利用包括0.6在内的概率分类对第二和第三尾部比特编码，最后，利用包括0.5在内的概率分类对第四和以后的尾部比特进行编码。

A: 执行子波变换

在一通过系统中，以执行的子波变换是压缩参数和所使用存贮器数量之间的折衷方案。选择编码单元尺寸以用于具有最少解缓冲器的最少存贮器(假定以光栅顺序传送图象)。子波变换的中间系数被存贮在同一个存贮器之内，以

在适当时候取代输入。

子波变换滤波器的选择

这里所描述的子波2-D变换被设计来用于一通完成和有限存贮器。共有四级单独解压。在水平解压过程中,仅使用TS变换,即使形成TS-TS-TS-TS水平解压。在垂直解压中,使用S和TS两种变换。且垂直解压形成为TS-TS-S-S解压。水平和垂直解压交替使用。图20示出了水平和垂直解压。

除了在存贮器应用方面的有效压缩以外,在无损耗压缩过程中使用低成本的S变换代替两个TS变换。在最后两次垂直通过中选择使用S变换仅用于少量的存贮器。S变换的使用大约保留了系数缓冲器的32行(例如来自48行的16行以下)。注意,对所有的解压使用TS变换并不是提供更好的压缩性能。

编码单元的规定

在一个实施例中,编码单元是由树的一行(LL系数或所有它们的下节点的一行)规定的。利用四级解压,这意味着在空间域中,编码单元是16行乘以图象宽度。图21示出了一个编码单元。注意,图21并没有按比例去画。级1方框是一个2-D解压之后的图象。为了重复,名称LL(低-低)、LH(低-高)、HL(高-低)和HH(高-高)被用于对子方框寻址并被提供给所有的级1-4方框。级2方框是在级1方框中子方框LL 2-D解压的结果。类似的,方框3和4分别是在级2方框和级3方框中子方框LL 2-D解压的结果。

编码单元在级1中对于HH、HL和LH系数是8行高、在第二级中是4行高,在第三级中是2行高,而在第4级和LL子方框中则是一行,注意,当分解在每个步骤减少时,行的数量以及长度减半。在级4方框LL中的每个系数是一个树的最高母体。

缓冲和系数计算

为了产生一个在图21中所述的编码单元,可以使用其尺寸为 $2 \cdot w \cdot m$ 的工作区缓冲器,其中, w 是图象的宽度, m 是一个比特内的最大系数尺寸。由于为垂直变换(即行方向上)通过所选择的子波滤波器的特性,工作区存贮器的要求大约是18-20行。每次一行的计算所有的都是TS变换的每一个水平变换(即行方向上)的通过并用新的系数取代老的系数或像素。

头两遍垂直变换使用TS变换滤波器。由于是6抽头高通滤波器,所以,在垂直通过中的每个高通系数取决于任一像素系数数据的6行。所产生的高通TS

系数涉及到其下面4行用于重叠的顶部两行。这示于图22。参看图22, 该图示出了一个编码单元的垂直图象段。该编码单元的垂直图象段是正在利用TS变换的水平通过进行变换的原始图象的结果。这里示出了一个编码单元垂直级1段并且是该图象的第一个级2-D变换。通过利用TS变换执行水平通过获得垂直级1段。还示出了一个编码单元的垂直级2段, 该段是通过在两维上对级1方框的LL子方框执行TS变换而产生的。

应当注意, 由于TS变换是被4个象素(或系数覆盖的, 所以4行数据在编码单元的末端处被保留下来, 以备对下一个编码单元的系数进行计算时加以使用。换言之, 为了建立级1系数, 在顶部和底部需要两个额外的象素行或在左底部需要4个额外行。为了建立级2系数, 在顶部或底部需要2个额外的级1系数行或在底部需要4个。为了建立这些额外的级1系数, 在顶部和底部需要另外的两个象素行或在底部需要4个。因此, 每个编码单元要跨跃28个垂直行。

但是重要的是, 由于这些“额外”的级1系数将被用于当前编码单元之下或之下的编码单元, 所以不需要额外的计算去产生它们。另外还应当注意, 由于只存贮级2系数, 所以只需要存贮器的20行。

最后两个垂直通过是S变换, 它在低通情况下没有重叠, 因此不需要额外的行。

用于变换计算的存贮器

有关尺寸方向给出的象素值或系数值是 b 比特(在 $-2^{b-1}, \dots, 0, \dots, 2^{b-1}-1$ 的范围之内), s 变换和TS变换的平滑输出 $S(\cdot)$ 也是 b 比特。换言之, 它们与输入具有相同范围。但是, S 变换和TS变换的一组细节输出 $d(\cdot)$ 将利用 $b+1, b+2$ 分别加以表示。

图23示出了除编码单元以外所需的某个行缓冲。灰色区域和箭头是部份当前编码单元的系数, 它们需要被保存在存贮器中以用于当前编辑。虚线箭头是在编码单元中计算系数所需的暂时系数。这此系数将被新的系数所改写、实线箭头表示这样的系数, 即它们是当前编码单元系数计算的付产品并被保留以作为下一个编码单元的一部份。

这些系数的最后一级(级4)是所有4个子方框(LL, LH, HL, HH)中的单一一行。仅参看垂直变换, 从级3到计算级4仅使用了S变换, 所以, 所有子方框在级3中仅需要系数的两行。类似地, 从级2到计算级3在计算级2中需要4行。所

有的这些系数都是当前编码单元的一部份。

为了计算级2和级1的垂直通过，使用TS变换。由于6抽头高通重叠滤波器的重叠特性，这些级需要来自下一个编码单元的数据。这些数据被用于计算当前编码单元中的系数并被存贮起来以用于下一个编码单元。

为了计算级2(LH, HH)中来自级1的高通子方框，需要12行(8行到4行来自下取样，和4个额外行用于重叠)。这些行在图23的级1(LL, HL)低通子方框中示出，其中8行是当前编码单元的一部份，4行是下一个的一部分。

为了计算在级1(LL, LH)的低通子方框中的12行，需要从级0开始的24行，0在级0处的这24行能够建立在级1的高通子方框中的10行(16行到8行来自下取样，4个额外行用于重叠)。即使仅有8行需要用于当前编码单元，但计算这所有10行并在级1处存贮它们是非常重要的。因此，只有4个用于重叠的额外行在级0处被保留下来。

对于可分离2-D变换来讲，从象素深度b的图象开始，在行和列二者的情况下，变换是TS，对于LL, HL, LH和HH子方框来讲，系数范围分别是b, b+2和b+4。在可分离2-D变换包括水平TS和垂直S变换的情况下，对于LL, HL, LH和HH来讲，系数范围分别是b, b+1, b+2和b+3。表2, 3, 4, 5和6示出了与每个方框所需存贮器的计算。注意，该计算是根据与宽度为W的图象相关的比特的尺寸而进行的，每个方框中的一个：

表2 一用于级0的存储器成本

| 子方框 | L | H |
|-----|-----------------------|---------------------------|
| 存贮器 | $4 \cdot b \cdot W/2$ | $4 \cdot (b+2) \cdot W/2$ |

表3 一用于级1的存储器成本

| 子方框 | LL | HL | LH | HH |
|-----|-----------------------|----------------------------|---------------------------|----------------------------|
| 存储器 | $4 \cdot b \cdot W/2$ | $10 \cdot (b+2) \cdot W/2$ | $4 \cdot (b+2) \cdot W/4$ | $10 \cdot (b+4) \cdot W/2$ |

表4 一用于级2的存储器成本

| 子方框 | LL | HL | LH | HH |
|-----|-----------------------|---------------------------|---------------------------|---------------------------|
| 存储器 | $0 \cdot b \cdot W/2$ | $4 \cdot (b+2) \cdot W/4$ | $4 \cdot (b+2) \cdot W/4$ | $4 \cdot (b+4) \cdot W/4$ |

表5 一用于级3的存储器成本

| 子方框 | LL | HL | LH | HH |
|-----|-----------------------|---------------------------|---------------------------|---------------------------|
| 存储器 | $0 \cdot b \cdot W/8$ | $2 \cdot (b+1) \cdot W/8$ | $2 \cdot (b+2) \cdot W/8$ | $2 \cdot (b+3) \cdot W/8$ |

表6 一用于级4的存贮器成本

| 子方框 | LL | HL | LH | HH |
|-----|------------------------|----------------------------|----------------------------|----------------------------|
| 存贮器 | $1 \cdot b \cdot W/16$ | $1 \cdot (b+1) \cdot W/16$ | $1 \cdot (b+2) \cdot W/16$ | $1 \cdot (b+3) \cdot W/16$ |

7

把上述所有的量相加等于 $(26b+55 \text{ ———}) \cdot W$ 比特, 约等于 $(26b+56) \cdot W$ 比

8

特。大尺寸的两行计算缓冲器, 即 $b+4$ 比特加上 $2 \cdot (b+4) \cdot W$ 比特导致了 $(28b+64) \cdot W$ 比特。例如, 对于一个 8 比特 512 象素宽的图象来讲, 需要 147.456 比特或约 18K 字节。

在一个实施例中, 变换的规模是根据图象的宽度和可以得到的存贮器的固定尺寸进行选择的。换言之, 一个特殊图象可以被输入到本发明的系统中, 并且由于可获得的变换存贮器的数量有限, 所以解压级的数量减少。如果可以获得更多的存贮器, 那么, 解压缩级的数量可以增加。注意, 当系统正在接收图象时, 这种情况可能不断发生。如果可以得到足够的存贮器, 则可以通过子波滤波器馈入 LL 系数, 以执行附加级的解压。注意, 减少或增加级的数量这样一个事实意味着分别减少或增加可以实现的压缩的数量。

B. 本发明的嵌入顺序

代码流的排序

图 24A 示出了代码流的排序以及在编码单元内的排序。参看图 24A, 在标题 2401 之后跟随有从顶部到底部的编码单元 2402。在编码单元内, LL 系数 2403 不以光栅(行)顺序进行编码。在 LL 系数之后是从最高有效比特平面到最低有效比特平面的一次一个比特平面进行熵编码的数据。然后在第二比特平面之前, 来自每个系数的第一比特平面被进行编码, 等。

系数的排列

在本发明的一个实施例中,上下文模型使用一个非规格化的 $1+z^{-1}$ 低通滤

$$1+z^{-1}$$

波器。但是,该上下文模型可以使用诸如

$$\frac{1+z^{-1}}{\sqrt{2}}$$

的规格化滤波器。为了使用规格化滤波器,可以在正向子波滤波器1600和上下文模型105之间使用一个排列单元以对来自非规格化滤波器的增加(或损失)的能量进行补偿,这改善了压缩。由于这种排列对于有损耗操作来讲允许不均匀量化,所以,这种排列可以增强有损耗图象重建的可视质量。在一维情况下,来自该树每一级的系数都将具有不同的排列(除数= $\sqrt{2}$ 、2、 $2\sqrt{2}$ 、4,乘数= $2\sqrt{2}$ 、2、 $\sqrt{2}$ 、1)。在两维情况下,除数将是2、4、8、16,乘数将是8、4、2、1。

由于该排列仅用于类似于与编码相关的二进制判定的分组,所以使用精确格式化的值不是很关键的。在解码期间,排列必须被反相,所以需要乘法和除法,使用被平方的乘数/除数允许硬件更有效地执行操作。当系数被平方时,所增加的减少有效零比特不必进行编码。

系数排列可被用于调节和用于更好的非均匀量化。在(两维信号)图象的情况下,RTS变换的一个实施例通过把频带乘以图28所示的数来排列系数。乘以这些数导致RTS变换非常接近TS变换的精确重建的子波。

该一通过实施例仅使用了就与滤波器配对有关的MSE而言是最佳的一个排列。表7示出了排列号。利用有效位对系数进行编码,其中,第一比特平面是所有系数左边最高量级的比特。在所述系数具有非零量级比特以前,用于每个系数的符号比特不被编码。换言之,在第一个非零比特被编码之后,立即对符号比特进行编码。它的优点在于对于任一具有零量级的系数而言,可以不必对符号比特编码,并且在嵌入代码流内的符号比特的相关点前不必对符号比特进行编码。对于一个象素深度为b的图象而言,最大可能的系数幅值为 $2^{b+3}-1$,即:数量为b+3比特。因此,每个系数都以b+3二进制判定进行编码,并在需要时加上一个用于符号比特的附加的一个。

表7 一系数排列

| | | | | | | | |
|------|-----------|------|-----------|-----|------------|------|------------|
| 1-HH | 1HL, 1-LH | 2-HH | 2HL, 2-LH | 3HH | 3-HL, 3-LH | 4-HH | 4-HL, 4-LH |
| 基准 | 左1 | 左1 | 左2 | 左2 | 左3 | 左3 | 左4 |

不同尺寸系数的排列被通知给编码器和解码器，并不在熵编码器上进行有效压缩。

还应当注意，编码单元每个方框的每个子方框都可能具有它自己可能的最大幅值范围，这个范围都要被通知给编码器和解码器。对于大多数子方框来讲，具有一系列完整的决定性的二进制零值，为提高效率起见，这些零值都被熵编码器忽略。

在每个比特平面期间的系数的处理顺序是从低分辨率到高分辨率，从低频到高频。每个比特平面内的系数编码器是从高级(低分辨率，低频)到低级(高分辨率，高频率)，如下：

4-LL、4-HL、4-LH、4-HH、3-HL、3-LH、3-HH、2-HL、2-LH、2-HH、1-HL、1-LH、1-HH。

在每个子方框内，编码按照光栅扫描顺序。

注意，同一数据组的编码单元可以具有不同的排列。在一个实施例中，排列可以被规定在诸如图24A所示的标题2401中的标题内。

水平上下文模型

图25示出了与编码单元每一系数相关的相系数。参看图25，该图利用非常明显的图表说明诸如：N=北，NE=东北等注释了相邻系数。

由于给定了诸如图25中的P这样一个系数和一个当前比特平面，所以，在给定位平面之前，上下文模型可以使用来自所有编码单元的任一信息。该信息的母体系数也被用于这个上下文模型。

不使用相邻的或母体的系数值去确定与该系数的该比特相关的上下文，信

一息被减少到这里被称之为尾部信息的两个比特。这个信息可以被存贮在存贮器中或根据邻近的或母体系数不断地进行计算。尾部信息涉及到第一个非零幅值比特是否被观测到(即第一个“on-bit”是否被观察到)以及在此之后大约有多少比特平面。表8描述了尾部信息比特:

表8 - 尾部信息的规定

| 尾部 | 规定 |
|----|------------------------|
| 0 | 还没有观察到“on-bit” |
| 1 | 第一个“on-bit”在最后一个比特平面上 |
| 2 | 第一个“on-bit”在两或三个比特平面之前 |
| 3 | 第一个“on-bit”在多于三个比特平面之前 |

根据2-比特尾部信息,尾部接通比特指出尾部信息是否为零。在一个实施例中,在2系数编码之后,立即更新尾部信息和尾部接通比特。在另一个实施例中,较晚发生更新,以允许并行上下文产生。

作为一个例子,表9示出了尾部接通比特。作为比特平面的函数,用于具有用二进制为下表示的幅值的系数(“*”的意思是无论它是0还是1”):

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | * | * | * | * |
|---|---|---|---|---|---|---|---|

表9 - 是否是所举例上下文系数相关的尾部信息的表

| 比特平面 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------|---|---|---|---|---|---|---|---|
| 举例系数发生之前 | 0 | 0 | 0 | 1 | 2 | 2 | 3 | 3 |
| 举例系数发生之后 | 0 | 0 | 0 | 0 | 1 | 2 | 2 | 3 |

第三种上下文信息是符号比特。由于符号比特是在第一个on-bit (非零比特)之后立即编码的。所以, 该尾部信息表示所述符号信息是否是已知的。因此, 除非该尾部信息不是零(符号重新调用有三种可能: 即正、负或未知), 否则该符号比特不是信息上下文。

系统的上下文模型使用高达11个比特去描述上下文。这个数量并没有完全被规定, 实际上只使用了包括符号比特上下文在内的1030或1031个上下文。每一个比特位置的意义取决于在前的二进制值。一个实施例遵循下述规则:

如果该系数的尾部接通比特是零(用于首比特), 那么来自母体和W系数的尾部信息比特的1024个上下文分别是NW、N、NE、E、SW和S系数的尾部非零比特。在一个实施例中, 适当的编码被用于首比特。在某些实施例中, 一个单一的上下文被用于提供首比特的“运行编码”。如果紧接着将被编码的16个比特全都是首比特并且它们的N、S、E和W相邻和所有的母体的尾部信息为零, 那么将对一个单一的判定进行编码。这个判定表明将被编码的16个比特中的任一个是否有一个比特位于当前比特平面处。如果没有这样一个比特, 那么可以忽略被一般编码的16个判定。如果下一个16个系数中的任一个包含有它们的第一个有效比特, 则对于每个比特使用一次16个判定。这种“向前看”导致了较少调用二进制熵编码器, 从而导致了较高的速度和较高的压缩。

如果该系数的尾部接通比特是1(用于尾部比特), 那么, 三个上下文来自该系数的尾部信息比特)。如在前面的讨论的, 可以使用固定概率编码。

如果该系数的该位是第一个非零幅值比特，那么该系数的符号比特在此后被立即编码。与符号比特相关的上下文是来自N尾部比特和N符号比特的3个上下文，其中，如果N尾部比特是零，则N符号比特是未知的。如果N符号比特是未知的，利用概率0.5对符号进行编码，否则该符号被自适应编码。

总之，建立了11个比特量以表示来自从同一编码单元中的当前，相邻和本源系数中可以获得的信息的上下文。

图26A-D示出了可以用于决定系数P状态的因果和非因果系数。所以标准框中的每一个包括尾部比特和尾部信息的使用。当每个系数的尾部比特提供一个比特时，每个系数的尾部信息包括2个比特。在图26A中，由标准框提供的比特总数是8。在图26B和26C中，由标准框提供的比特总数是10。

附加比特可被用于决定系数P首比特的状态。在一个实施例中，两个附加的比特可以规定比特位置如下：

- 00 第一比特 (MSB) 和第二比特
- 01 第三比特和第四比特
- 10 第五比特和第六比特
- 11 其它比特

应当注意，可以根据相邻和母体系数来设计其它的标准框。另外，在一个实施例中，如果不是由比特平面定位的，那么用于决定系数P的系数是因果系数。

在一个实施例中，使用S变换母体而不是使用TS变换母体去决定状态。这减少了通过在继续为下一个编码之前保留低通行去决定状态所需的缓冲，它的缺点在于熵编码的编码器是重要的，而编码器存储器是不重要的。

注意，作为在上下文内减少数据的结果，在具有更多的上下文去建立更多错位数据和自适应效率之间存在着折衷。

在一个实施例中，那些不需要决定状态的尾部比特不被进行(用于决定状态的)缓冲。它们可以在被得到之后立即进行编码。在这种情况下，通道管理器可以直接将所述比特传送给通道。

在一个实施例中，不以和其它系数相同的方式对处于最低解压级的系数进行编码或根本不对它们进行编码，这些系数可以使用诸如DPCM的预测编码进行编码。

对于尾部编码比特来讲,可以使用一次固定概率或自适应。

就状态决定而言,最后一个比特可以部份地根据第二到最后比特来决定状态。另外,在第一个非零比特之后的比特可以根据它们距离第一个非零比特有多远来决定状态。

在一个实施例中,某些尾部比特被自适应编码。例如,当在系数9(例如: $T=2$ 、 $T=3$)中的尾部比特少于 T 尾部比特时,就使用自适应编码。与这些比特相关的上下文包括在当前系数中的比特位置和任一先前编码的尾部比特。这类似于Langdon就DPCM数据以教导的 M 进制的中心编码(the M -ary coding of center)。

在另一个实施例中,某些或全部数据都利用 M 进制的熵编码器而不使用二进制熵编码器进行编码。 M 进制的编码器包括Tunstall编码器、固定霍夫曼编码器和自适应霍夫曼编码器等。例如,一个霍夫曼编码可被用于首比特。在另外一个不是同时对首比特和一个比特进行编码的实施例中,使用优先级编码器来确定第一个非零比特的位置,然后利用二进制熵编码器对用二进制表示位置的比特进行编码。

水平上下文模型。

本发明的上下文模型以方框图的形式示于图27。上下文模型2700包含有一个符号/幅值单元109(图2)和三个用于对系数中的不同比特进行处理的单元。根据正在被编码的比特,选择上述三个单元之一。可以包括一个开关以便于在硬件结构单元中的转换。这些单元包括首比特方框2701,符号比特方框2702和尾部比特方框2703。如上所述,首比特方框2701,符号比特方框2702和尾部比特方框2703分别构型首比特、符号和尾部比特。这三个单元的输出被传送给熵编码器104(图4)。

编码器包括可选择控制,用于保留状态(可选择的)、提供最初状态和复位编码器(例如,在编码单元结束处)。

除少数例外,上面以规定的上下文使用自适应二进制角熵编码器。当 N -tail-on=1时的首比特(该系数的尾部比特=0)和符号比特允许修改。

但是,tail-on=1之后的比特和 N -tail-on=0时的符号比特是由一个固定源来构型的。在这种情况下,不需要熵编码器的自适应特性。事实上,该自适应特性可能是一个效率很差的压缩源。对于下面的上下文,使用根据 Q -编码器状态所描述的固定(无自适应)状态。

统计模型

当 $N_{\text{taril-on}}=0$ (N 系数的符号是未知的)时, 上下文被用于对符号比特进行编码, 并且在固定 Q -编码器状态0-概率近似为0.5处对上下文进行编码。

在固定 Q -编码器状态4-概率近似0.7处对第一非零(位 P 信息=1)比特编码之后, 上下文被用于对第一个二进制值进行编码。

在固定 Q -编码器状态3-概率近似为0.6处对第一个非零比特(尾部信息=2)进行编码之后, 上下文被用于对第二和第三个二进制值进行编码。

在固定 Q -编码器状态0-概率近似为0.5处对第一个非零比特(尾部信息=3)进行编码之后, 上下文被用于对第四和以后的二进制值进行编码。

在某些实施例中, 在每个编码单元之后, 熵编码被复位, 所以与允许去修改(例如用于对第一个on-bit之前的二进制值进行编码的)上下文有关的修改容量是非常重要的。为了保持这个最低容量, 可以计算一组初始状态, 以用于来自例如某个数据串的这些上下文。

下面的叙述假设该系数是18个比特, 且数入数据必需经过4级解压。

符号/幅值单元109的一个实施例示于图28, 它被用于将输入系数转换成一个符号/幅值格式。符号/幅值单元109被耦合以接收该系数的18个比特, 它包括反相器2801和多路转换器(MUX)2802。符号/幅值单元109输出(例如是5比特值的)有效指示、(例如是17个比特的)输入系数的假设、输入系数1的符号和例如是7个比特的来自计数器2804的索引。

MUX2802被耦合, 以接收直接输入给符号/幅值单元109的系数的17个比特和来自两个反相器2801的17个比特的反相比特。根据在MUX2802选择输入端接收到的符号比特(系数比特17), 两个输入端的正值被输出以作为一个假数)。

编码取舍

二进制角锥编码器被用于上下文和将要被编码的比特。

对于利用比特平面编码的比特平面, 本发明使用了(以通用计算机为基础的)保留进位形式计算, 从而使得计算能够以适于由比特平面快速编码的数据格式进行。例如, 在这样一个执行过程中, 32个比特处理器可以同时计算在同一个比特平面内的32个系数中每一个的一个比特, 而不是全部系数中的一个。当利用比特平面进行编码时, 使用这样一个实施例导致速度增加。

由于一个编码单元同时被编码且所有编码单元内的系数都驻留在存储器内,

所以,除了需要修改二进制熵编码器以外,就不再需要利用存储器去存储上下文信息。例如,Q-编码器需要保持与有上下文有关的LPS(最低有效符号)二进制值以及用于被允许进行修改的每个上下文的当前状态。由于Q-编码器需要30种状态,所以,对于每个上下文来讲需要6个比特(一个比特用于LSB,5个比特用于状态)。因此存储器的存储容量是 $1024 \times 5 = 6150$ 个比特。

注意,对于上述一通嵌入实施例来讲不需要特殊的信号信息。如果压缩级数是可变的,那么就需要至少标题信息的3个比特。没有被计算在压缩比特之内的用于该实施例的标题如下:

- 宽度, 2字节
- 高度, 2字节
- 输入图象每个象素的比特, 1字节存储器管理

对于在存储器中存储所有数据的系统或左通道中发送数据的系统来讲,存在有关于在一通系统中编码数据的存储器管理问题。在一通系统中,编码数据必需被存储以使其能够以嵌入因果方式被访问,从而使得可以在不损失较高有效数据的情况下丢掉一些较低有效数据。由于编码数据的长度是可变的,所以可以使用动态存储器分配。

在本发明的一个实施例中,嵌入编码方式使用18个比特平面,由此给数据指定18个有效级。一通系统中的顺序是“嵌入因果关系”,即,对与一个比特平面相对应的事件进行介码不需要来自低顺序比特平面的信息。在一个实施例中,在下一个树的任一比特被编码之前,来自该树的所有比特被进行编码,所以,不同有效级的比特不会被分开。对于那些不使用内部状态的编码器而言,与霍夫曼编码器类似,这不是问题。但是许多具有更好压缩的复杂压缩使用内部状态。

解决与这些编码器相关的这个问题的途径之一是使用18种不同的编码器,设想是18个Q-编码器芯片。在1992年3月17日授权的、发明名称为“用于在记录媒体上进行记录的数据压缩”的美国专利5,097,261 (langdon)中描述了一种允许使用9个Q-编辑器芯片的技术。一种校好的方式是使用流水线编码器利用诸如在1995年1月10日授权的、发明名称为“用于数据并行介码和编码的方法和装置”的美国专利5,381,145中描述的那种单一的实际编码器执行不同的虚

拟编码。在这种编码器中，与每一个概率相关的多个比特产生器状态中的每一个被指定给一部份数据。例如，18个状态中的每一个被指定给与18个比特数据相关的特殊比特平面。在该编码器中移位器内的寄存器也被指定给每一部份数据。在该编码器中，不执行重叠，每一部份数据被简单地构成一个比特信息包。

在多个实际或虚拟编码器中任一种的实施例中，存贮器被分配给每一部份数据。当压缩完成时，得到一个表示所分配的存贮器以及所分配存贮内容的一个键接表。

如果存贮器过溢，则存贮器分配程序安排使得比较重要的数据去重写不太重要的数据。例如，一个数字数据的最低有效位可能被首先重写。除了编码数据以外，描述有多少存贮器被分配的信息也必须被存贮起来。

图29示出了与三种有效级相关的动态存贮器分配单元的例子。为了避免使本发明难于理解，仅描述了三级，通过使用较大数量的级，诸如8,16或18级。寄存器文件(或其它存贮器)保持一个与第一有效级相关的指针，并保持一个用于指示下一自由存贮区分配的指针。该存贮器被分成固定尺寸的页。

最初，指定给有效级的每一指针指示存贮器一页的开始，而自由指针指示可以获得的下一个存贮器页。利用有效级识别的编码数据被存贮在由相应指针寻址的存贮器单元内。然后指针增加到下一个存贮单元。

当指针到达与当前页相关的最大值时，作为一个键接利用当前页存贮在自由指针内的下一个自由页的起始地址。在一个实施例中，为此目的使用了一部份编码数据存贮器或隔离存贮器或寄存器文件。当前指针被置于下一个自由页。增加自由指针。这些步骤使得一个新的存贮器页被分配给一个特定的有效级并提供对含有与公共有效级相关的数据的存贮器页的键接，从而使得可以在解码期间确定分配顺序。

当存贮器内的所有页都处于使用之中，且存在有比存贮器内最低有效数据更加有效的数据时，执行存贮器的重新赋值。描述了三种这样的赋值技术。在所有这三种情况下，指定给最低有效数据的存贮器被重新指定给较多有效数据，并不再存贮更低的有效数据。

首先，最低有效数据当前正在使用的页被简单地指定给较高有效位的数据。由于最典型的熵编码器使用初始状态信息，所以在那一页事先存贮的所有最低有效数据都被丢失了。

其次，最低有效数据当前正在使用的页被指定给较高有效数据，和先前情况不同，指针被置于该页的末端，并且当较高有效数据被写入该页时，相应的指针被减少。这有一个优点，就是如果较高有效数据并不需要整个页的话，那么就在该页的开始处预先保留了最低有效数据。

第三，代替正在被重新指定最低有效数据的当前页，可以重新指定最低有效数据的任一页。这需要对与所有页相关的编码数据独立编码，这可以减少所实现的压缩。它还需要识别与所有页的起点相对应的未编码数据。由于最低有效数据的任一页都可以被丢掉，所以可以在量化过程中获得较大的灵活性。

在整个图象区域内实现固定压缩速率的系统中，第三种方法可能是：最具吸引力的。规定数量的存贮器页可以被分配给图象的一个区域。较低有效数据是否保留取决于在一个特定区域内实现的压缩。注意，如果无损压缩需要的存贮数量少于指定的存贮数量，那么指定给一个区域的存贮器不可能全部被利用。在图象区域上实现固定速率的压缩能够支持对图象区的随机存取。

当压缩完成时，如果需要的话，可以按照有效的顺序将数据传送给一个通道或存贮器设备。各种键接和指针都得将不再需要，并且执行多通译码。另外，对于一通译码来讲，可以保持到与每个有效级相关的数据的指针。

在某些应用中，不可能使用某些有效级。例如，一个16比特的压缩器可以被用于一个12比特的医学图象，这样，与比特平面 $15 \cdots 12$ 相对应有效级将不被使用。在具有大页和很多未使用有效级的情况下，由于不必将存贮器分配给它们，所以（当该系统事先不知道某些级未被使用时）将造成存贮器的浪费。另外一种解决这个存贮器浪费的方法是使用一个小的存贮器（或寄存器）去保持与每个有效级相关的计数。该计数将对在任何其它判定发生以前发生的“无效”判定的数量进行跟踪。相对于由未被使用有效级所使用的存贮器，存贮这些计数所需的存贮器必须被折衷选择。

把数据从两端写入到每一页的能力可被用于更好地利用在系统中可以获得的存贮器总量。当所有的页都被进行了分配时，在末端处具有足够自由空间的任一页被可以被分配以从其末端开始加以应用。对于保持跟踪能够满足两页数据的存贮单元的容量来讲，使用一页两末端的能力必须平衡。注意，这不同于数据类型之一是无效的并应简单地加以重写的情况。

改进

在一个实施例中，本发明提供了一种使用较小存贮缓冲器的无损压缩。本发明能够保留很多不同的应用和设备环境。下面描述的技术所完成的各种特性可以使本发明的系统对于不同的应用和目标设备变得更加灵活。注意，对于本发明来讲，不必在编码时对分解率、象素深度、随机存取和量化等作出选择。

数据排列

关于数据排列，就利用本发明的系统对图象和系数数据进行排列方面可有一定数量的选择。如在下面将要详细描述，这种选择包括（不作为对本发明的限制）：编码单元的拼接、解压级的数量、子波变换滤波器的选择以及系数的校准。照此，这些中的每一个都应当是用户或系统设计者控制的参数。

如上所述，一个参数可以是编码单元的拼装。根据本发明的树来规定所述编码单元的高度和宽度。对于随机存取而言，与每个编码单元有关的编码数据的起点可由代码流中的指针或标记符或标题中的指针来指定。这将允许对不是图象宽度的方框进行存取。

另外一个可以被控制的参数是解压级的数量。根据较多的解压级数可以导致较好压缩这样一个事实，改变解压级的数量就是改变了压缩性能。注意，当较多的级需要较多的行缓冲器时，解压级数的变化也会影响存贮器的要求。对于企图达到低于全分辨率的分辨率来讲，可能需要较多的级。例如，如果原始图象是2000dpi，那么就需要5级解压去实现的62dpi。这允许在没有解压和子取样的情况下在一个监视器上进行接近实际尺寸的将被显示的高分辨率扫描。

用于在每一级中水平和垂直通过的子波变换滤波器的类型也是不同的。这允许不同的存贮器要求和压缩性能。注意，系数的尺寸不随级数的增加而增加。另外，由于子波变换是一个顺序N变换，并且当级数增加时只有少量的数据被变换，所以，多于较多级来讲只有少量的计算量。

用于嵌入代码流的目标设备

对于一个特定的压缩代码流来讲，可能有很多应用目标。可以指望有这样一个代码流，它可以利用较低的分辨率和全象素深度被传送给监视器，还可以指望有这样一个指针，它具有全分辨率和较低的象素深度，另外，可以指望一

个具有有限通道的固定速率实时设备或固定尺寸的有限存贮器设备。可能需要相同的代码流以保留以有这些需要。图34概括地示出了一个单一应用可以服务的相关设备特性。

变换成译码代码流分析程序

编码和解码器具有足够速度并具有足够带宽的本发明的系统能够从解压后的图象中提取所需的数据。另外，编码器可以建立一个供上述设备中一个设备使用的代码流。在编码器处，一个图象可被以传统方式进行量化或下取样。

但是，本发明的优点在于对于上述在一设备来讲，不必解压就可以在变换和解码之前建立起一个可被分析的代码流。这种分析程序示于图35A和B。参看图35A，分析程序方框3501接收无损耗比特流量并产生有损耗比特流。参看图35B，分析程序方框3502接收有损耗比特流，并产生另一个有损耗比特流。但是，图35B所示输出和输入之间的关系表明本发明具有等幂关系，这在后面将要详述。注意，在分析程序方框3501和3502的情况下，作为输入所接收的数据比特速率大于正在被输出的数据比特速率。

低分辨率、高象素深度的嵌入目标

如果一个目标是一个低分辨率、高象素深度的嵌入目标，那么这种应用假设该目标设备所具有的空间分辨率低于可以获得但需要全象素深度的目标设备。监视器是低分辨率、高象素深度嵌入目标的例子。使用图24A所示的代码流，在需要时一比特平面被译码以用作较高级系数。这要求分析程序将每一个比特平面截短。为协助该分析程序，每个编码单元的每个比特平面都具有表明可以发生截短的位置的标记符或指针。在这种实施例中，如果需要一个以上的目标分辨率，那么就需要多个标记符或指针。比特平面被进行单独编码，所以，熵编码器可以被复位以用于下一个比特平面。

另一种途径是嵌入诸如图24B所示的不同数据。参看图24B，每个编码单元内的目标分辨率系数在剩余高分辨率系数之前被首先编码。在这种情况下，每个编码单元仅需要一个截断且熵编码器不再需复位。标记符或指针可以表示所希望的截断点。

高分辨率、低象素深度嵌入目标

如果一个目标是一个高分辨率、低象素深度的嵌入目标，那么这个应用假定该目标设备需要可以获得的全部分辨率，特别是它不能使用全象素深度。高

分辨率、低象素深度嵌入目标的例子包括低档打印机和标准的监视器(当图象是高于每个平面8比特时)。图24A所示的代码流是按着这个顺序嵌入的、每个编码单元在正确数量比特平面的一个点处被截断,并根据量化系数执行变换。在系数深度和象素深度之间具有直接的关系。标记符或指针可以表示所希望的截断点。

另外,如果代码流被如图24B所示的嵌入,那么就要使用两个标记符或指针去表示截断,一个用于表示低分辨率比特平面,一个用于表示高分辨率比特平面。两组比特平面被独立编码的允许熵编码器被复位。

另一种办法就是为根据低分辨率、高象素深度嵌入目标所述的对部份或所有低分辨率系数进行解码和对来自高分辨率系数的某些可能的数据进行介码。然后如下所述地执行插值子波变换。

固定速率嵌入目标

如果该目标是一个固定速率嵌入目标,这个应用假设在使用约束通道的同时,必需保持实时恒定象素输出。在这种情况下,在适当的时刻(最小压缩速率)就会局部地具有某个最大代码流数据速率。为了实现这个目的,首先根据在目标设备中可以得到的缓冲器的数量选择编码单元。这规定了一个局部区域,在此区域上可以实现平均压缩速率。然后具有多于允许数据的每一编码单元被截断。

注意,如果代码流数据速率不超过最大通道代宽,那么图象被无损耗的复原。这并不适用于任一其它固定速率系统。

固定尺寸嵌入目标

如果该目标是一个固定尺寸嵌入目标,那么这个应用假设压缩图象数据可以获得固定尺寸的帧缓冲器。与固定速率应用不同,这需要在整个图象上、而不是局部上平均的最小压缩速率。当然,这是可以使用固定速率方法,但所使用的平均这个概念是整体而不是局部的,由此可以得到更好的比特分配和图象质量。

在含有整个图象的编码单元中,截断缓冲器的过溢数据是不重要的。如果编码单元少于整个图象并且所有的编码单元都被截短成相同数量的比特,那么就不能保证这个截断唯一地删除了最不重要的级。一个简单的方法就是在记录时间(或后边的分析时间)处记录下编码比特的数量,每一个重要的级都将这些

数量分配给与每一编码单元、全局编址首部或二者有关的代码流。该记录可使用一简单的计数器进行。这些数量被记录在标题段并能被用于决定在变换或存贮时如何去截断每一个编码单元。标题含有一个重要级和它的相应比特数量。标题还可以包含与每个包含在所述代码流中的编码单元相关的这个信息。当决定在什么地方截断时，在每个编码单元上执行的效果是相同的。例如，如果由于存贮器限制，它决定丢弃一个或1/2个重要级，那么来自每个编码单元的一个或1/2重要级将被丢弃。这就允许以唯一的方式在编码单元的两端执行截断。

在编码时间实现固定尺寸压缩的图象。存贮器被合成用于多个重要级的多个段。如果存贮器将要过溢，那么利用较重要的数据重写不必太重要的数据。

注意，如果压缩数据没有溢出存贮缓冲器，那么将无损耗地复原所述图象。这不同于其它任何固定尺寸系统。

利用子波变换插入

子波可被用于将图象插入到较高的分辨率中。和双三次样条技术 (bi-Cubic Speie) 相比较其结果是相当明显的。如果压缩数据已处于子波系数的形式，那么，与插入相关的有效附加计算少于双三次样条。

可以获得N级所有系数都被解压的图象。通过填入零或某些其它的方法建立一个新的最低有效级，然后执行N+1级子波重新，可以获得原始图象2:1插入方案的一个新图象。

这种方法也可被该系统所使用，在该系统中，有多个目标设备，用于嵌入代码流，特别是用于高分辨率，低象素深度的目标设备。编码单元被截短，因此仅具有低分辨率系数 (或仅具有低分辨率系数或少数几个比特的某些或全部高分辨率系数)。该系数被装入较高分辨率并执行重建。

使用通道

在一个数据在通道内传送而不是存贮在存贮器中并且使用存贮器固定尺寸页 (但每个有效级仅需要一页) 的系统中，当存贮器的页是满的时，数据在通道内被传送，并且数据一旦被传送，存贮器单元就可被重新使用。在某些应用中，存贮器的页尺寸可以是通道中使用的数据信息包的尺寸，也可以是信息包的尺寸的倍数。

在某些通信系统中，例如在ATM(异步传输模式)的通信系统中，可以将优先权指定给信息包。ATM具有两个优先级：主要和次要。如果可以获得足够的带宽，仅传送次要信息包。一个阈值可被用于确定哪一个有效级是主要的，哪个是次要的。另外一种方法是使用一个位于解码器内的阈值不去传送其有效级低于该阈值的的有效级。

由此，本发明存贮管理器的一个实施例控制在固定尺寸存贮器内压缩数据的存贮。即：存贮管理器将压缩数据分成不重要程度的级。当存贮器被充满时，利用比较重要的数据去覆盖不太重要的数据。

为了管理使用有限量缓冲存贮器(例如，固定速率)的通道，在本发明的一个实施例中，如果可以获得足够的带宽，所有的数据都将被传送，否则丢掉不太重要的数据并仅传送比较重要的数据。

图30示出了一个使用通道管理器的系统。参看图30，子波变换3001产生多个系数。这些系数承受上下文模型3002的处理。上下文模型3002被耦合到包括缓冲存储器的通道管理器3003。通道管理器3003被耦合到有限带宽通道3004。

通道管理器3003控制数据向通道3004输出的速度。当其缓冲存贮器接收到数据时，通道管理器3003确定数据量是否大于通道3004的带宽。如果该数据量不大于所述带宽，则通道管理器3003输出所有的数据。另一方面，若缓冲存贮器接收的数据量大于所述通道带宽，那么通道管理器3003丢掉其缓冲存贮器内的信息，从而与通道3004的带宽相匹配。

通道3004将它的带宽表示给通道管理器3003。在另一个实施例中，通道管理器3003可以根据通过通道3004传送预定数据单元(例如信息包)时占据通道3004的时间量来动态地确定通道3004的带宽。即，如果需要，该带宽可作动态处理。

在一个实施例中，通道管理器3003基于被分成多个块(taile)(或频带)的一个图进行工作。即和其中分块(tiling)和重要性稍微相互有些独立的固定尺寸存贮管理器相比较，这是一个“在重要程度上块优先”的方案。每个块被单独编码，并被分成重要级和固定尺寸页。因此，与每个块相关的所有编码数据都被一起分组。每个块中的编码数据都被加有重要级的标记。

在一个实施例中，通道管理器3403中的缓冲存贮器是通道信息包尺寸的至少两倍(或约三倍)并且对于一个块来讲，该倍数(可能是4倍)大于所期望压缩

数据的尺寸。

缓冲存储器的固定的最大量被指定给一个块。该最大量与该通道的带宽相匹配。缓冲存储器被分成固定尺寸段，并在需要时加以分配。若存储器占用达到了能够允许的最大值，这些段如在固定尺寸存储器系统中的管理一样被重新指定。

图31示出了缓冲存储器占用的一个例子。参看图31，环形缓冲存储器具有多个被分成用于通道输入/输出的多个固定尺寸信息包3102的固定尺寸段3101。如所示，不同的数据块可以占用存储器的相同数据色。在一个实施例中，不同的块表示不同的重要级。当使用缓冲器空间的信息包尺寸量时，通道管理器3103将输出数据的上下文模型指示给通道3104(图30)。如所示，块N-2和块N-1的一部份将作为当前信息包而被输出。由此，存储器的信息包尺寸量被分配和保存，以和通道的带宽相匹配。

若缓冲器没有被充满，多余的存储器可被用于未来的块、在一个实施例中，为了在对于一个与下一信息组相对的压缩区域来讲是很困难的起点处避免明显的块边界，下一个块仅使用了剩余存储器的某些部份(1/2, 1/3等)。

在数据只能在某个时间周期内进行传送的场合可以使用本发明的通道管理器。使用这种通道管理器，由于数据是在其重要性的基础上被嵌入的，数据传送是在考虑到复杂性的时间周期内发生的。

通道管理器的另一实施例

本发明通道管理器的一个目的就是要使用最小存储器。在一个实施例中，通道管理器不包含缓冲存储器，下述内容可被使用：

- 与每个编码单元相关的，
- 与每个比特平面相关的，
- 与每个频率相关的，
- 与每个空间单元相关的。

在一个实施例中，在每个频带的起点，编码器被复位(或被置成公知状态)。在一个实施例中，如果需要减少频带存储器，那么该频带包括与四级介压相关的16行。

图32示出了使用上述方法的比特流。参看图32，比特流被分成作为通道信

息包的固定尺寸段，盘扇区等是用于该通道的缓冲器的适当量。注意，该判定可以不多于编码期间的逻辑判定。如果需要，编码器可以不使用缓冲而输出。每个固定尺寸段包括该段中最重要数据的一个指示。

图33示出了一个段的结构。参看图33，与每个段相关的比特位包括，编码数据3301，运行指针或ID3302以及该段中最重要数据3303。在一个实施例中，比特段3303包括2到5个比特。如果最重要级为0(最重要的一个)，那么，紧跟在该段最后M比特的一个指针，它告诉级0的数据在该段的哪个位置开始。注意，数据的第一段可能都是编码数据，因而就不再需要额外的开销。

类似于在JPEG标准中所使用的，可以利用再开始标记来识别与每个频带相关的起点。但是，所使用的标记是那些在编码期间最不经常发生的符号。

现在再来考虑图31。假设仅对那些重要级(恐怕仅是最重要级)进行介压。介压从第一段开始。对于第二段，检查“在段中最重要数据的级”并且恐怕在该段中包括的最重要级低于正在被介压的级的整个段都可以被跳过。对于第三段，指针被用于获得频带2的起点，并可以开始对频带2进行介压。

注意，为了保证能够获得该段中所有的最有效数据，可以要求对整个段进行介压。特别是当一个以上的频带落入在该段之中的时候。

通过有选择地仅对预定数量的有效频带进行介压，可以获得一个予展图象。当数据处于嵌入形式并需要无损耗数据的有损耗版本时，这可能是个优点。

根据所要求的存取和量化可能性以及对整个频带解压所需时间是否是重要的，在该段末端处可选指针或ID包含：

- 用于固定尺寸存储器管理的下一段指针。
- 与该段有关的一个ID或被包含频带的ID(支持通道量化，例如用于表示段2是否被忽略)。
- 与该段包含数据相关的不同频带的数量(或者用于表示含有多于两个段的频带的至少一个比特1)。

与 512×16 象素的频带相关的附加开销的例子是具有2:1无损耗压缩的8比特图象以及512个字节的段尺寸。注意，一个频带通常被压缩成8个段。对于重要级来讲，使用5个比特特征。假设指针处于字节边界上，则使用9比特指针。因此，共有占总数0.15%的49个加比特时(32K压缩位+49)。

近似无损耗

近似无损耗的一个概念就是在重建象素值的过程中以绝对误差为基础。因此，在具有绝对误差为1的近似无损耗压缩图象中，可以保证在该压缩图象中的象素值与原始值的差异不会多于象素值的一个单元。这是一个与图象的象素深度和动态范围无关的绝对限定。在某些适当的假设之下，对这样一个系统明显的和最佳的是保持压缩/介压部份的无损耗并使用一个预处理和后处理方案去实现近似无损耗：在这个执行方案中采取了这种方法。

通过把每个 $2e+1$ 连续整数变换成它们的中值整数这样一种量化方法来实现具有绝对误差为 e 的近似无损耗图象。假如，误差等于3，象素值被量化，0到6被变换成3，7到13被变换到10等等。量化图象本身不适宜以变换为基础的系统。因此，它被逐一地(无损耗)变换成被称之为浅图象的较低动态范围或深度的图象，这是通过把所述中值(代表值)变换成保持所述顺序的连续整数进行的。数学上给出的象素值 X 被量化成：

$$g(x) = \left\lfloor \frac{x}{2e+1} \right\rfloor + 1)e$$

把代表值逐一变换成浅图象值是：

$$P(x) = \left\lfloor \frac{x}{e} \right\rfloor - 1$$

把浅图象值变换成代表值的相反逐一变换是：

$$P^{-1}(x) = e \cdot (x+1)$$

在所浅图象值 $p(x)$ 进行变换之前的量化 $q(x)$ 是预处理操作, 这个操作进行无损耗压缩。从浅图象值到代表值的变换形成了后处理操作, 该操作遵循无损耗解压。

可以使用变换域量化。通过多级变换, 很多系数会受到峰值误差扩散的影响, 很容易确定峰值误差对于那些没有“孩子”的高通滤波系数的影响。

下面来考虑一个将要利用最大峰值误差 $\pm 2E$ 进行编码的一维信号。它可以通过把最细微的高通系数量化成 $\pm 2E$ 来实现。对于二维信号来讲, 由于应用于两个高通滤波器, 所以, 最细微的HH系数可以被量化成 $\pm 4E$ 。

使用输入图象量化的另一方面是控制对熵编码器的判定。一个例子如下: 对于每个系数来讲, 如果把该系数设置为零不会使在受那个系数影响的任一象素中的误差超过最大误差, 则该系数被设置为零。在某些执行过程中, 只有一些特定的系数, 恐怕只有那些没有“子”的AC系数才会被进行测试。系数可以被认为具有其中每次只考虑一个的渴望(greedy)策略。另外一些策略考虑把系数分成多个小组且将该组最大可能的子集选择为零。

如上所述, 通过嵌入功能实现量化并使其最佳化, 从而使其能根据诸如RMSE的量化量度标准使其性能最佳。在一个实施例中, 执行各种系数的量化, 以根据人类可视系统改进其结果。在这种情况下, 需要对嵌入方案作少量改动。例如, 通过乘以因数2使系数移位从而改变他们之间的关系(或表示在诸如灰度码的不同类型计数系统中的数量)。

本发明的压缩子波系统在图象编辑期间是非常有用的。在现有技术中, 针对全分辨率打印图象的应用图象处理功能是非常耗时的并使交互处理非常困难。

在一个实施例中, 如果一个图象编辑系统保存了一些被压缩的块, 那么它将非常迅速地对标度表示执行一个操作(非常低通)以使用户进行评价。由于只需要根据所显示的象素进行操作, 所以这可以很迅速地加以执行。由于实际上的全分辨率象素影响输出, 所以这只是一个近似的最后结果。因此, 用户应迅速将摄像机移到该图象的各个部份(恐怕是某些电文)。如果用户这样作了, 那么图象编辑系统将施加一个操作给图象的那一部份。为便于此, 要存贮一个树, 该树包含有压缩系数和信息, 将对该信息进行处理操作且该信息还将被使用。

在一个实施例中, 重新规定重要级以允许在规定的窗口内执行无损耗压缩, 而有损耗压缩用于其余的图象。所述窗口是由用户固定或选择的。应当有不同

重要程度的多个窗口。在一个实施例中，尽管即使是对于两象素级也可能具有更好的具体值，但所述窗口只有 48×48 个字区那么大。

有关该点的可能用途是卫星成象。其中，卫星使用数据方面的无损耗窗口，所以利用JPEG人工制品不会使统计研究陷入困难，但有损耗压缩允许比利用无损耗压缩所能达到的更宽的领域。

在一个实施例中，用户在一个图象上给出一个任意的逻辑框，并规定在该逻辑框内数据的相对重要程度，一旦这个逻辑框被绘出，软件就增加该逻辑框的尺寸，使其增大到满足所需约束的最小尺寸，且该最小尺寸包含用户逻辑框。文件标题将包含有关所使用逻辑框的信息和重复程度的信息。当开始进行编码和译码时，编码器和译码器将对重要逻辑框中的系数提供更多的分辨率。对于卫星成像的情况，重要窗口可能要被预先规定。

幂等运算

对于无损耗压缩器来讲，通常一个重要的运算是： $DCDCI=DCI$ ，其中I是图象，C是压缩运算指令，D是介压运算指令。在本发明中，当数据被压缩成X比特而后被介压时，它们将能够被重新压缩成X比特并且有原来的X比特。对于嵌入系统存在有一个非常好的幂等方案。在一个实施例中，一个被压缩成X比特而后被介压再被重新压缩成Y比特($Y < X$)的图象和该原始图象被压缩成Y比特相同。

由于压缩和处理使图象从原始图象漂移，可以这一点是非常重要的。如果压缩器是幂等的，那么，多个有损耗压缩的介压周期就不会影响数据。在本发明中，有多少次数据以相同的速率被压缩和介压是无关紧要的。另外，对需要进一步量化的分析程序的有损耗输入所产生的结果与使用无损耗输入的结果相同。因此本发明包括以变换为基础的等幂系统，该系统包括一个子波变换，一个上下文模型、一个熵编码器，所以，系数被按顺序描述和存贮，因此，删除信息不能改变与当前系数有关的描述。

应用

本发明具有一定数量的用途，其中的一些作为例子在下面描述。特别是一些具有高分辨率和深象素的尖端应用和某些人工所不能及的应用使用本发明。

本发明可以在具有更加有限的带宽、数据存贮和显示能力的应用也可以使用相同压缩数据的同时，使尖端应用保持高质量环境中的最高质量。这是一种独立存在、通常为诸如粗略浏览(web browser)等现代成像应用所需的设备。

对于医学成像来讲，本发明在深象素图象(每象素10个比特到16个比特)方面的优良的无损耗压缩性能是很理想的。除无损耗压缩以外，本发明是一个纯的有损耗压缩器，没有如以信息组为基础的压缩器那样很多的人工仿真。使用本发明得到的有损耗人工仿真往往沿着形状的边缘，在这里，它们经常被人关于视系统的可视掩膜现象掩蔽起来。

本发明可以被用于予压(pre-press)生产中，在这种生产中，图象往往具有非常高的分辨率和很高的象素深度。利用本发明的角锥解压，一个予压操作员很容易根据图象的较低分辨率有损耗方案(在监视器上)执行图象的处理操作。当满足某些条件时一可以根据无损耗方案执行某些操作。

本发明还应用于电报传真中，其中没有压缩情况下需要的变换时间太长。本发明允许利用不同的空间和象素分辨率从传真机中输出非常好的图象。由于在这种情况下传送时间是非常重要的，所以，本发明的插入特性是很有用的。

本发明可以用于需要压缩的图象档案系统中，以特别用于增加存贮容量。由于该系统可以被在带宽、存贮及显示方面具有不同资源的系统所访问，所以本发明的设备单独输出是很有用的。另外，本发明的顺序传送能力对于浏览是很有用的。最后，利用本发明可以提供图象档案系统中输出设备所需的无损耗压缩。

本发明无损耗或高质量有损耗数据流中分级顺序特性使其能够很理想地用于world wide web/特别是用于设备独立性、顺序传送、高质量和公开标准必不可少的场合。

本发明可用于卫星图象，特别是那些深象素高分辨率的图象。另外，卫星成像具有有限带宽的通道。本发明允许灵活性，并且当利用它的顺序传送质量时，它可以允许人去浏览和预先观看。

如果该有用频带溢出则诸如ATM网络的“固定速率”的被限的带宽应用需要减少数据的方式。但是，如果存在有足够的带宽(或压缩是足够地高)时则不存在质量恶化。反之，如果该存贮器已装满则在计算机和其它成像装置中所存贮的如象有限存贮器帧所应用的“固定尺寸”需要减少数据的方式。再一次，

对于可无损耗地被压缩进入正确存贮器数的图象将没有恶化。

本发明的被嵌入的代码流适用于这种应用。对于一有损耗图象的传送或存贮来说，该嵌入含蓄的允许该代码流被调整或被舍位。如果无需调整或舍位，则该图象达到无损耗。

总之，本发明提供了一种单一的连续色调图象压缩系统。本发明的系统是具有相同代码流的无损耗和有损耗的系统并且使用的量化被嵌入（不言而喻是通过代码流）。该系统还被角锥化、被累加、提供一用于内插的装置、和被简化的实施。因而，本发明提供了一种多方面均适应的“独立装置”压缩系统。

被统一的有损耗和无损耗的系统是非常有用的系统。不仅是相同系统能够具有目前技术水平的有损耗和无损耗的压缩性能，而且相同的代码也同样如此。在编码、代码流的存贮或传送期间、或译码时，该应用可判定保持一图象无损耗代码或将它舍位成为一有损耗文本。

通过本发明所提供的有损耗压缩是通过嵌入量化来实现的。也就是该代码流包括了该量化。该实际的量化（或视觉的重要）水平可以是该译码器或该传送通道的函数而未必是该编码器的函数。如果该带宽、存贮、和显示手段允许时，该图象被无损耗地复原。反之，该图象仅需要和由更多限制手段一样的量化。

在本发明中所使用的子波被角锥化，其中通过该图象的两个系数来执行分解而不需不同的图象来执行。这比分层分解更为特殊。对于用于在低分辨装置浏览或显示图象的必须略图的应用来说，本发明的角锥本质是理想的。

在本发明中使用的嵌入是累进的，特别是通过比特平面，即跟随出租人比特的MSB累进。该空间和子波范围二者均可被累加地压缩，虽然本发明的累加上具体为在子波范围内。对于诸如打印之类的具有空间分辨但较低象素分辨的应用来说，在本发明中该比特的累加排序是理想的。具有相同代码流的这些特征是有用的。

本发明的该结构的一个优点是它提供了一用于插入的计算地有效模式。如果要求较高的分辨，则来自该有用子波系数的高通过系数可被插入并可执行本发明的反向子波。这种方法与双重立方仿样(bi-cubic spline)可直观的竞争但这小于在本发明的变换情况下的计算的强度。

本发明等幂的意思是一图象可以一有损耗形式被解压并重新压缩为相同的代码流。这个优点允许在具有浏览、滤除、或编辑的应用中多次压缩和解压循

环。

本发明涉及在软件和硬件二方面的简单实施。该子波变换可用四种加/减操作计算机和用于每个高通、低通系数对的某些位移。用一简单的“上下文模式”和一二进制“熵编码器”执行该嵌入和编码。该熵编码器可用有限状态机构或并行编码器来实施。

虽然在阅读了前面的说明之后对于本领域的技术人员来说对本发明的许多变化和变型毫不怀疑地是显而易见的，但也很清楚的是由实例所表示和描述的特定实施例并不意味着仅区限于这种方式。因此，最佳实施例的详细说明并不意味着对该权利要求范围的限定，权利要求中所述的仅是有关本发明所必需的特征。

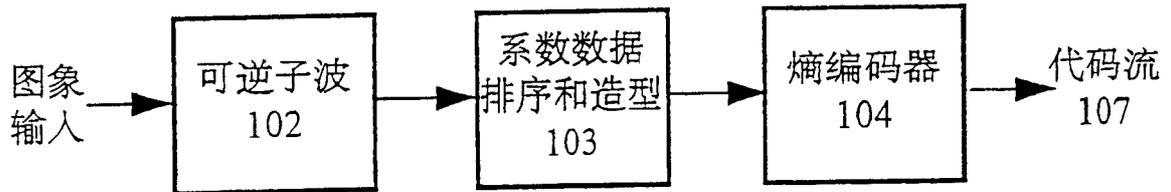


图 1

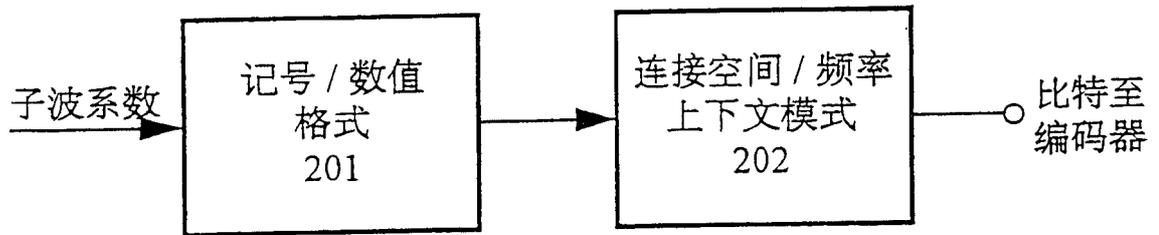


图 2

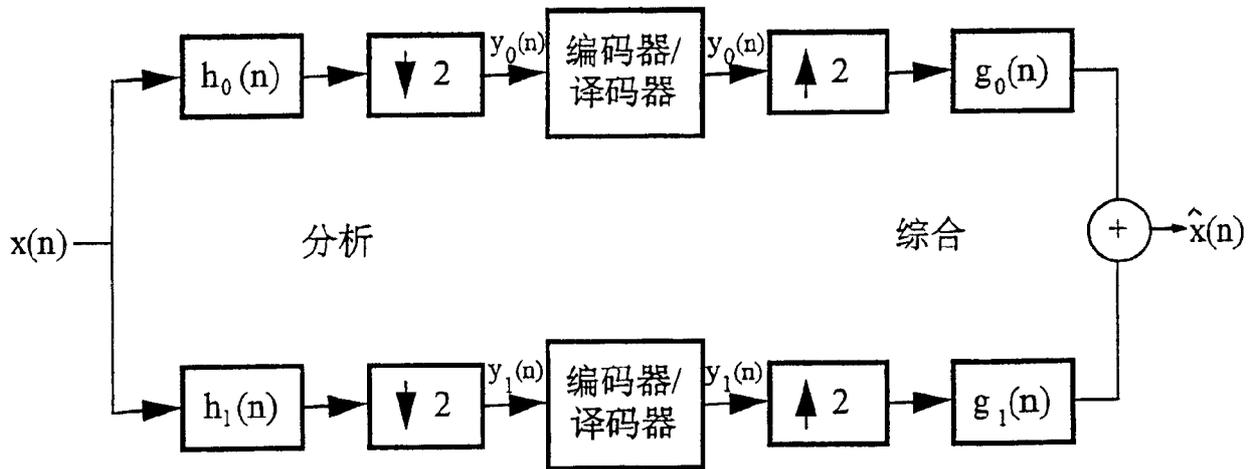


图 3A

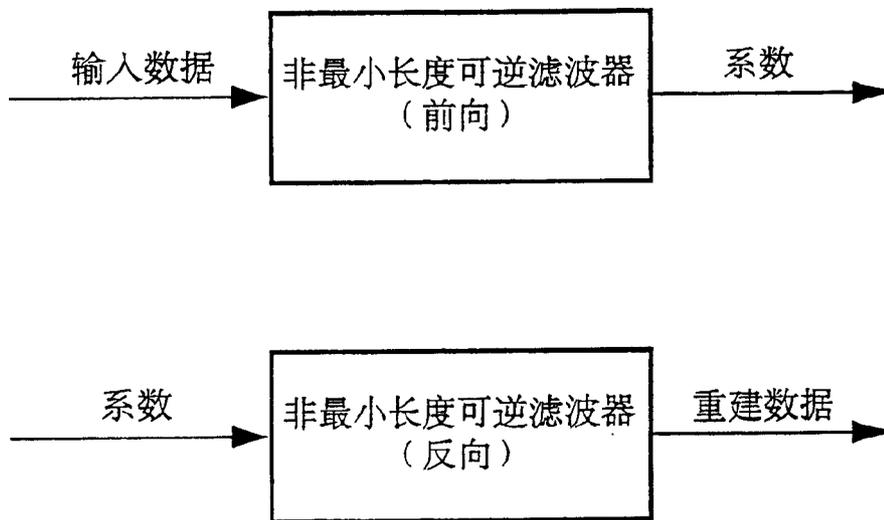


图 3B

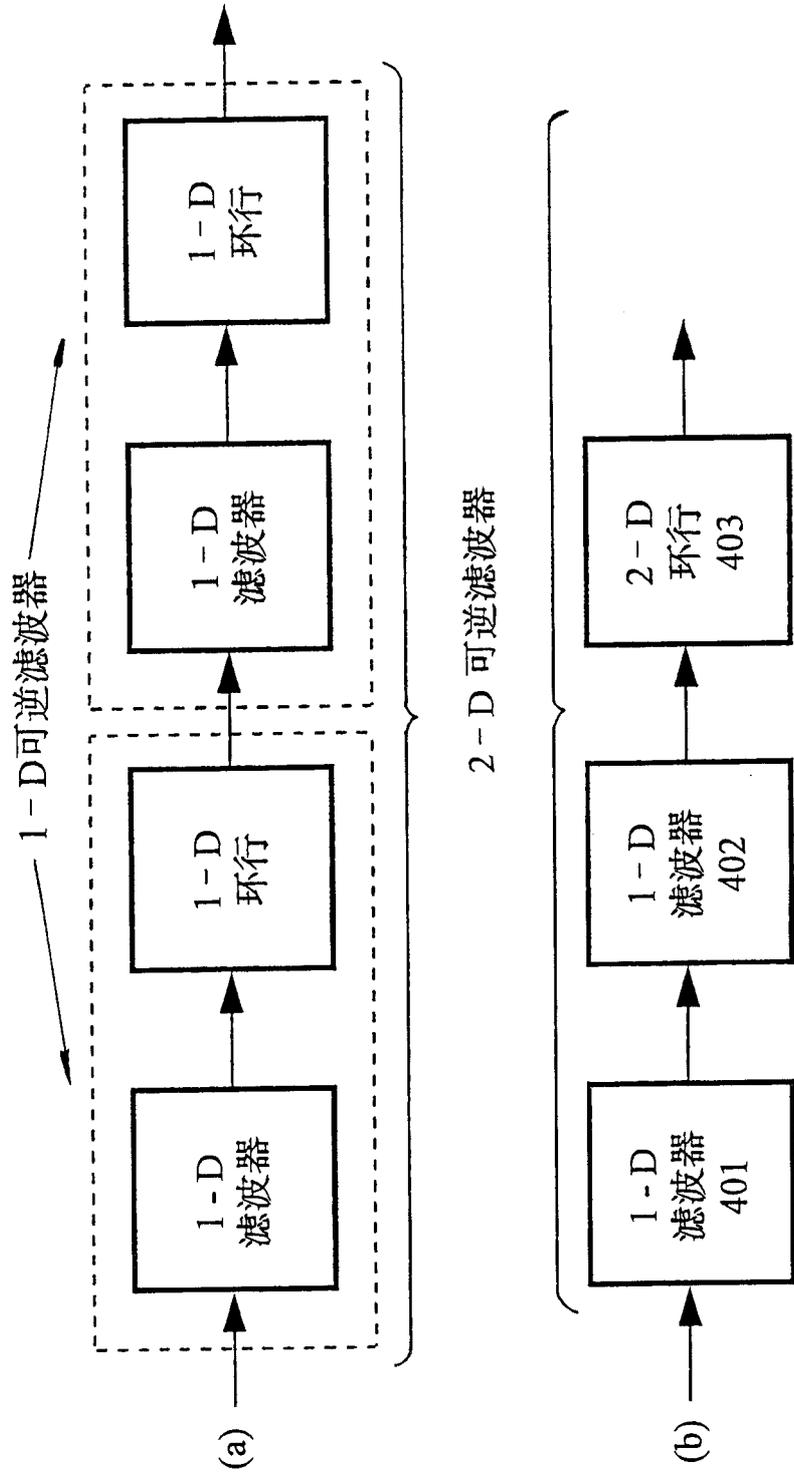


图 4

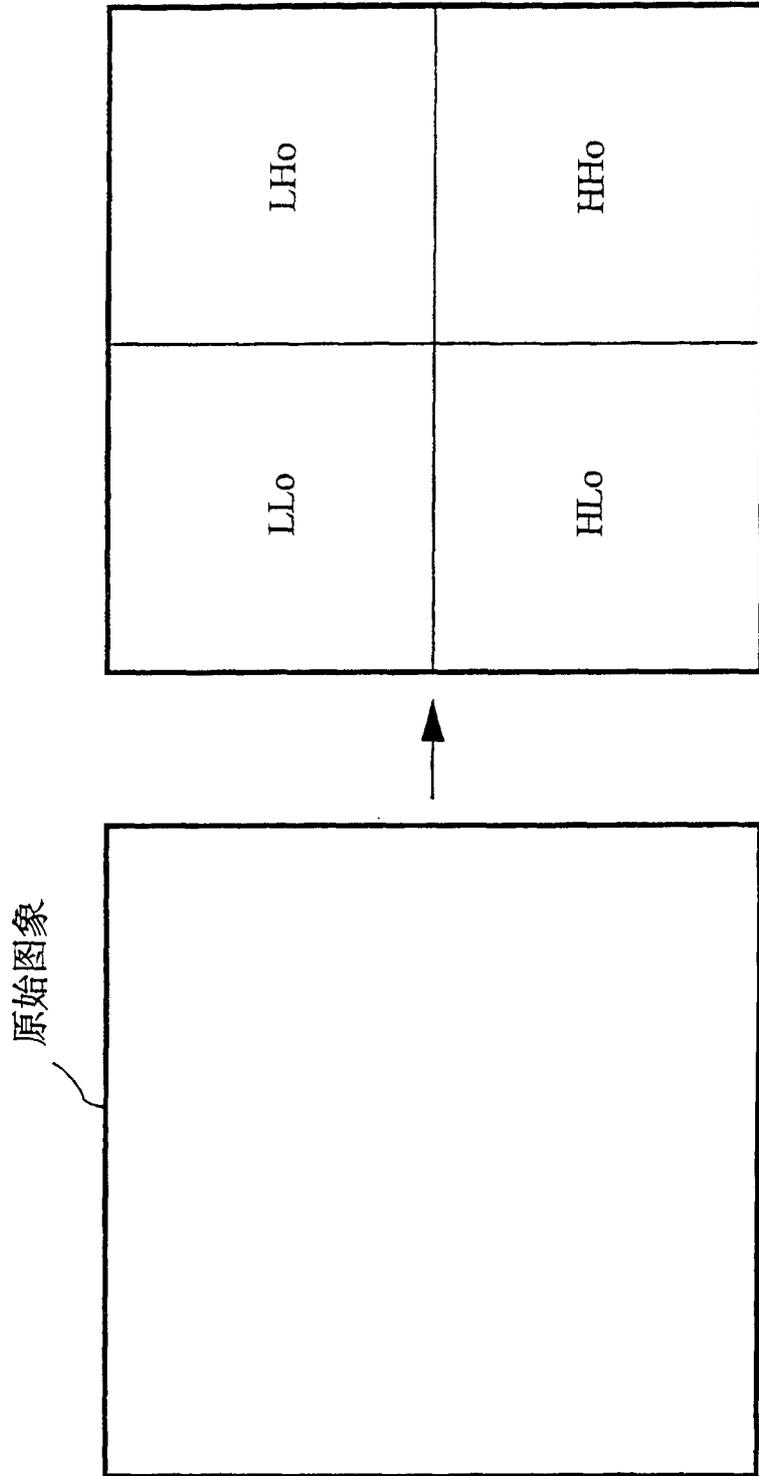


图 5A

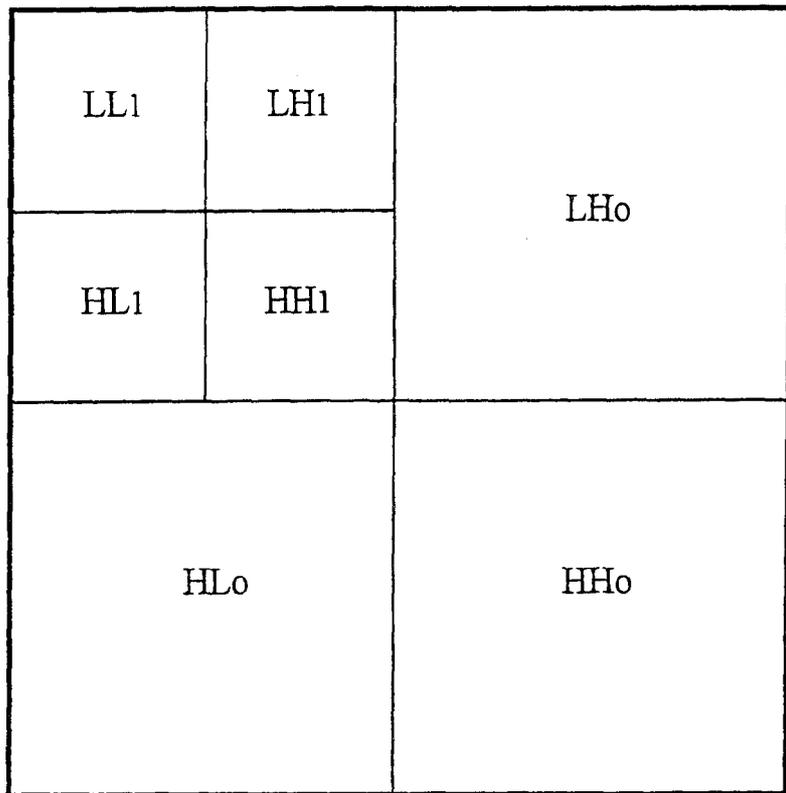


图 5B

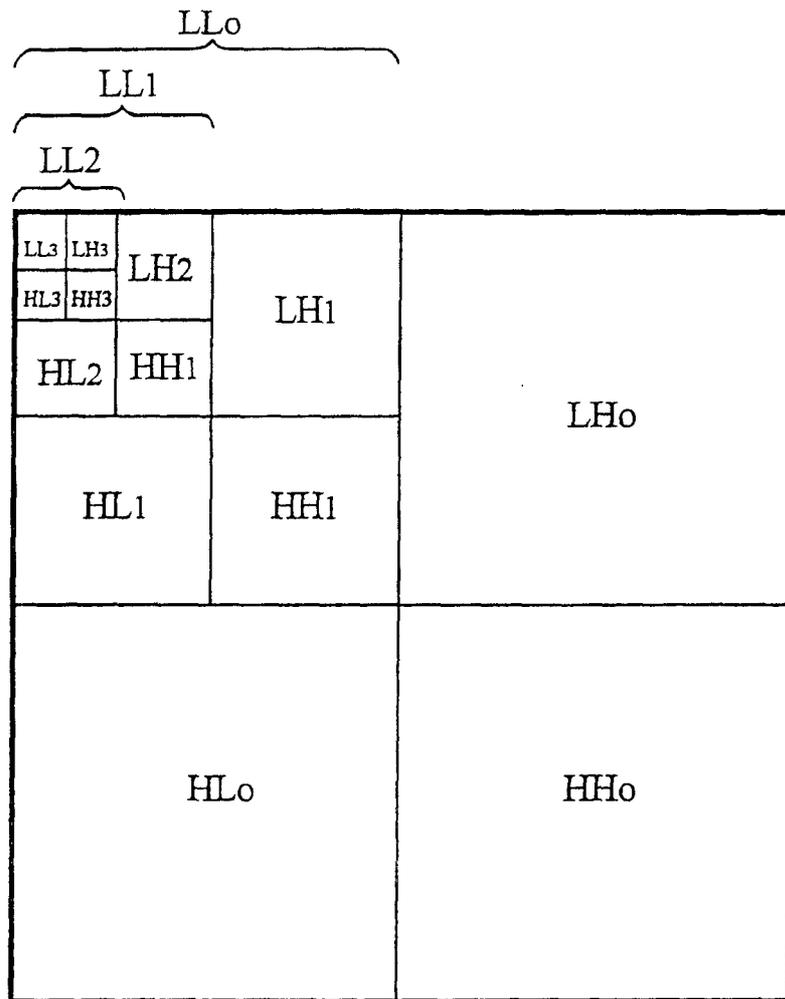


图 5D

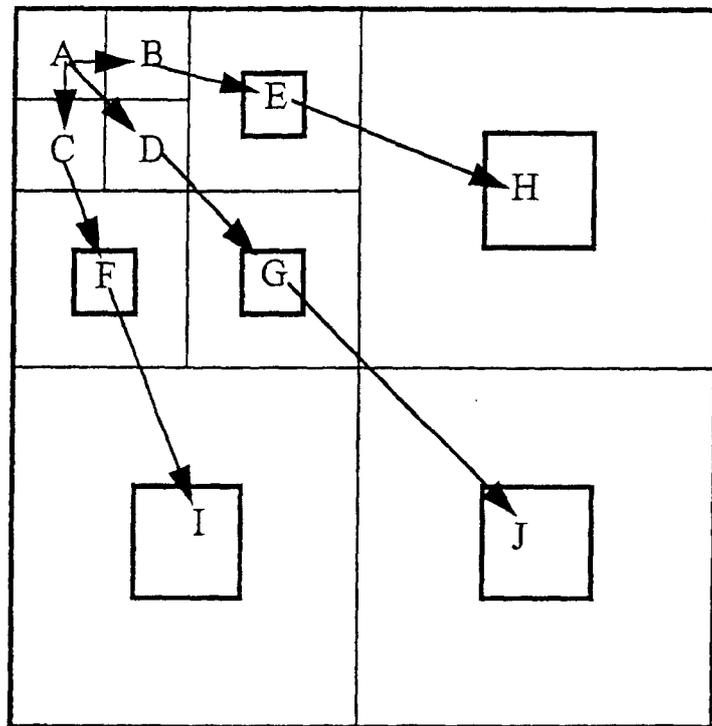


图 6

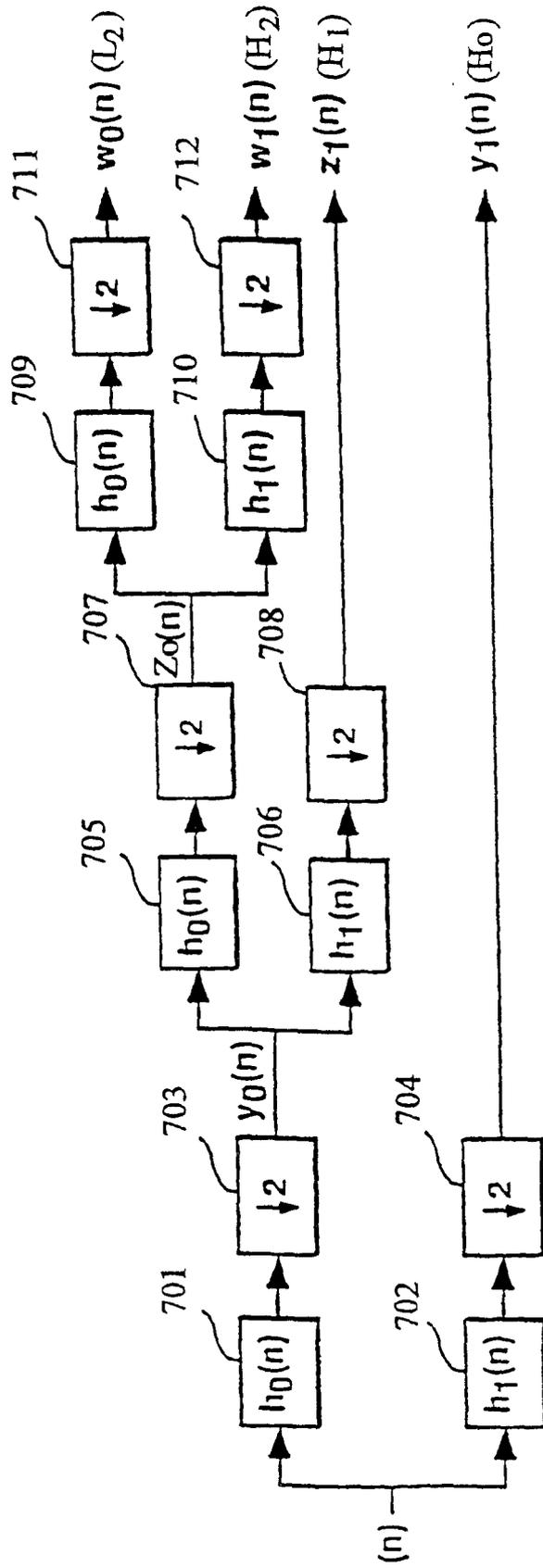


图 7

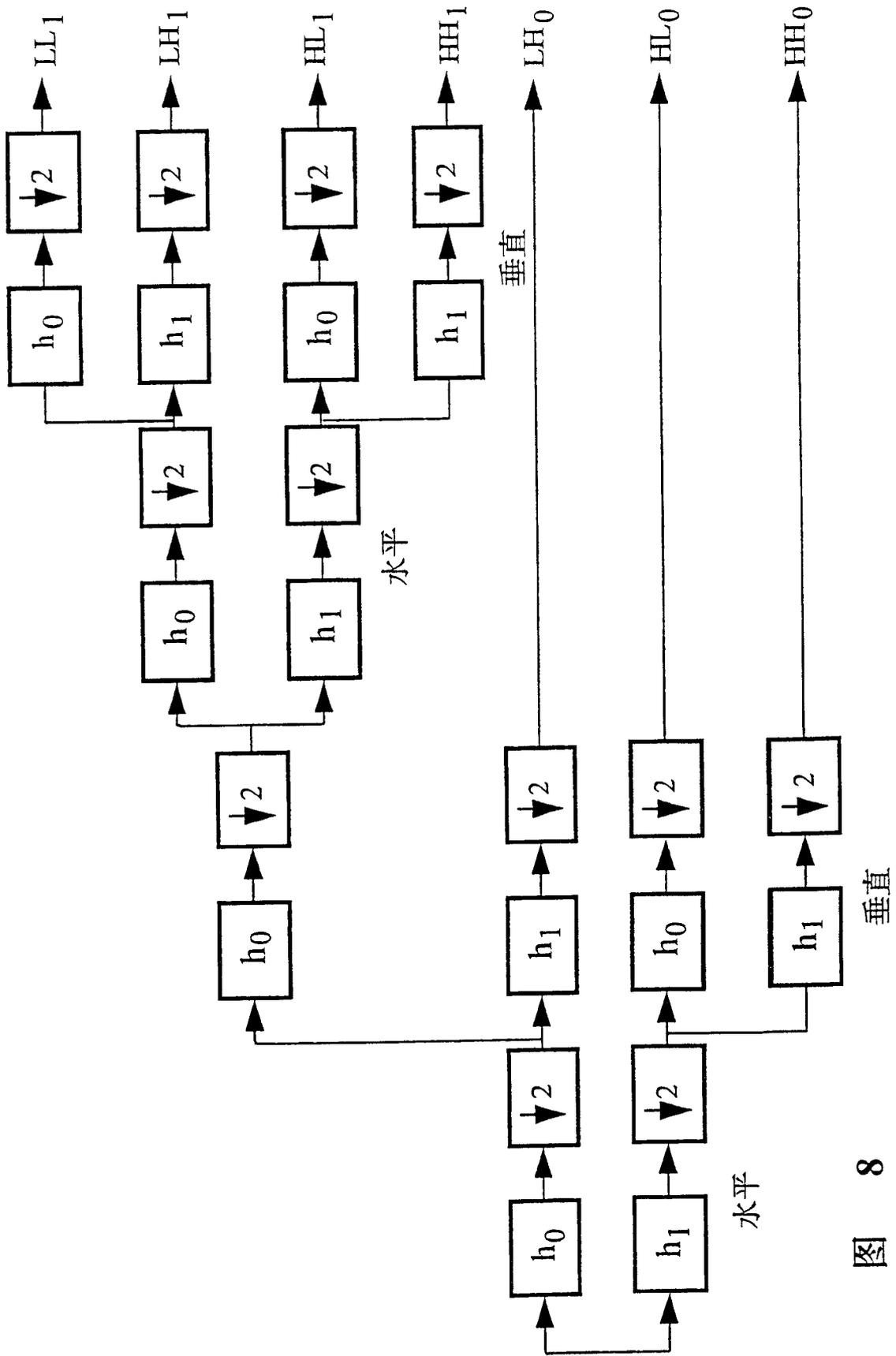


图 8

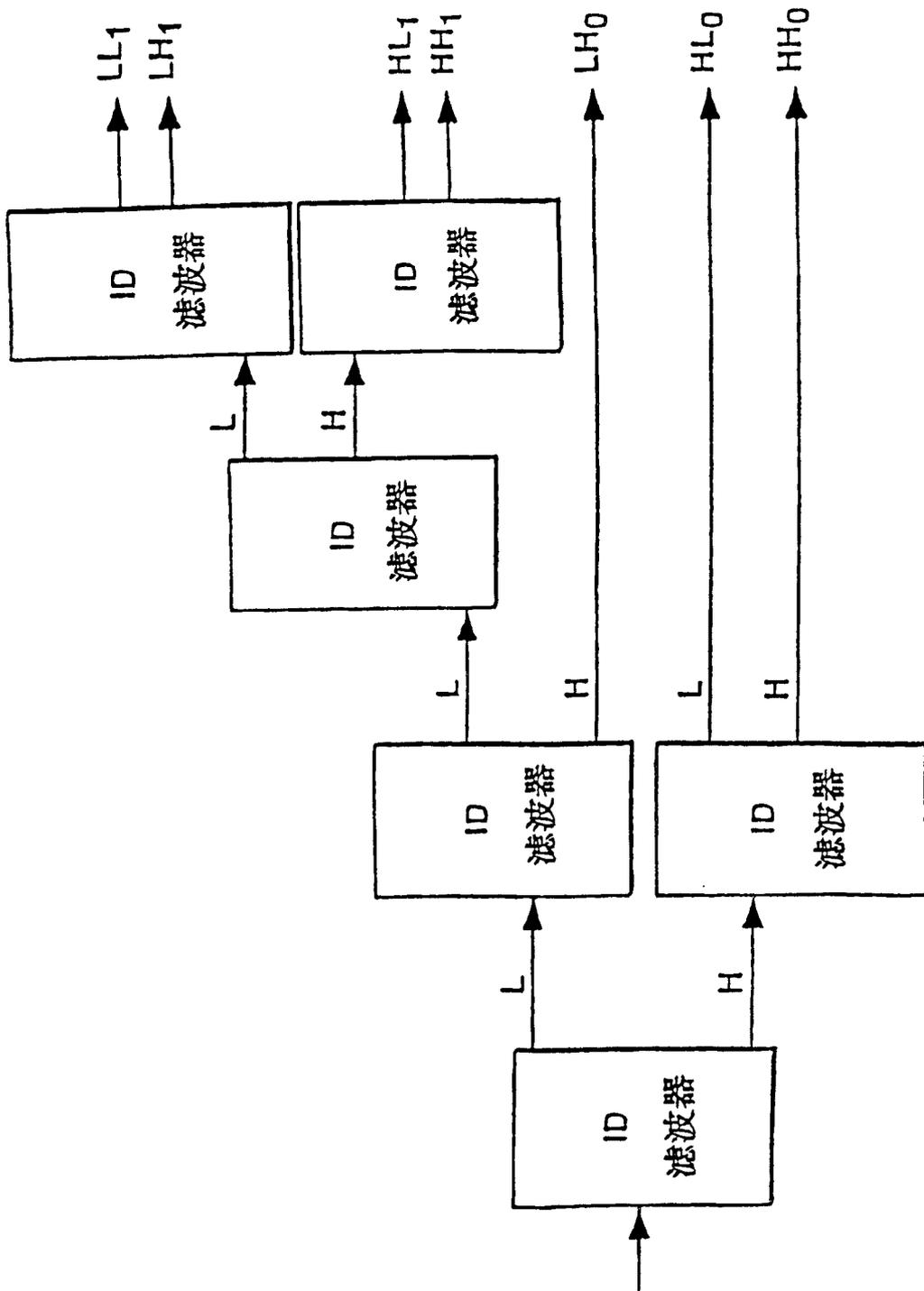


图 9

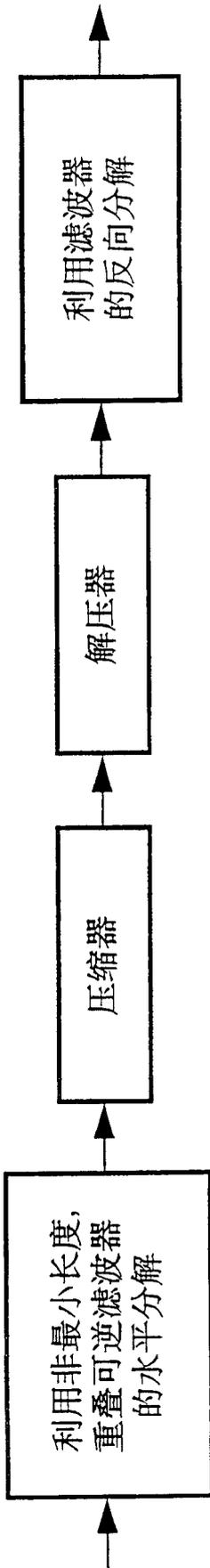


图 10

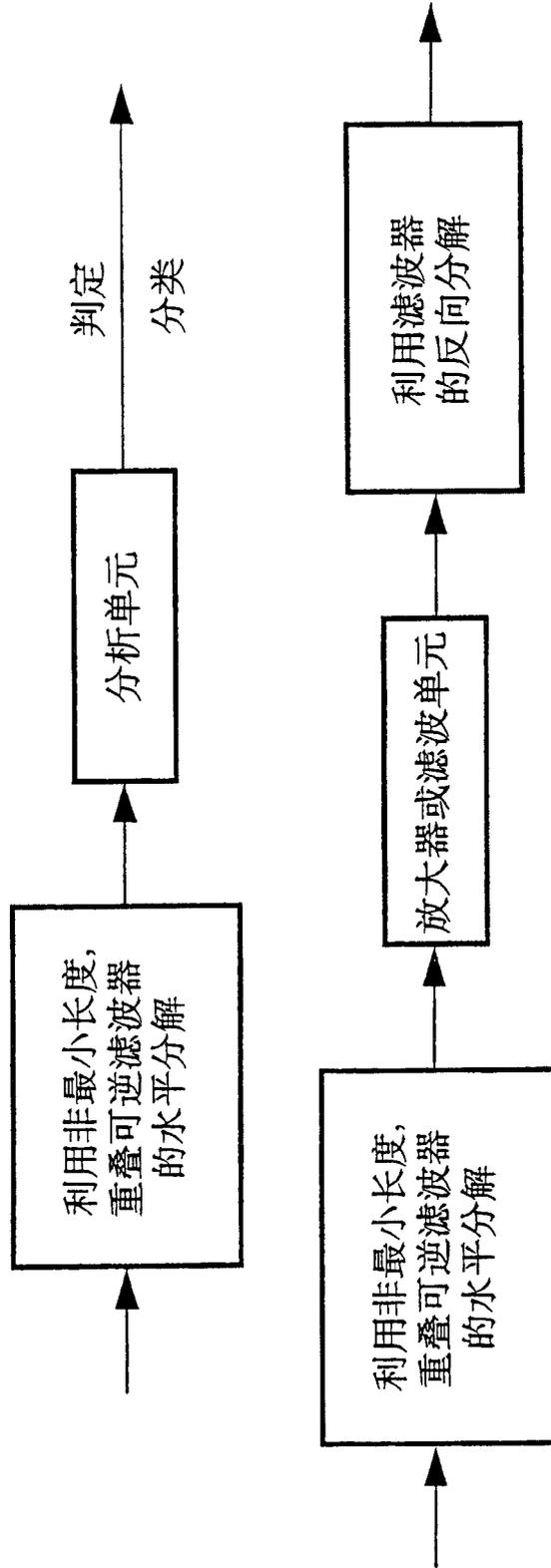


图 11

| | | |
|-----|-----|-----|
| b | b+2 | b+2 |
| b+2 | b+4 | |
| b+2 | | b+4 |

图 12A

| | | | | |
|----|----|---|---|---|
| | 16 | 8 | 4 | 2 |
| 16 | 8 | | | |
| 8 | 4 | | | |
| 4 | | 2 | | |
| 2 | | | 1 | |

图 12B

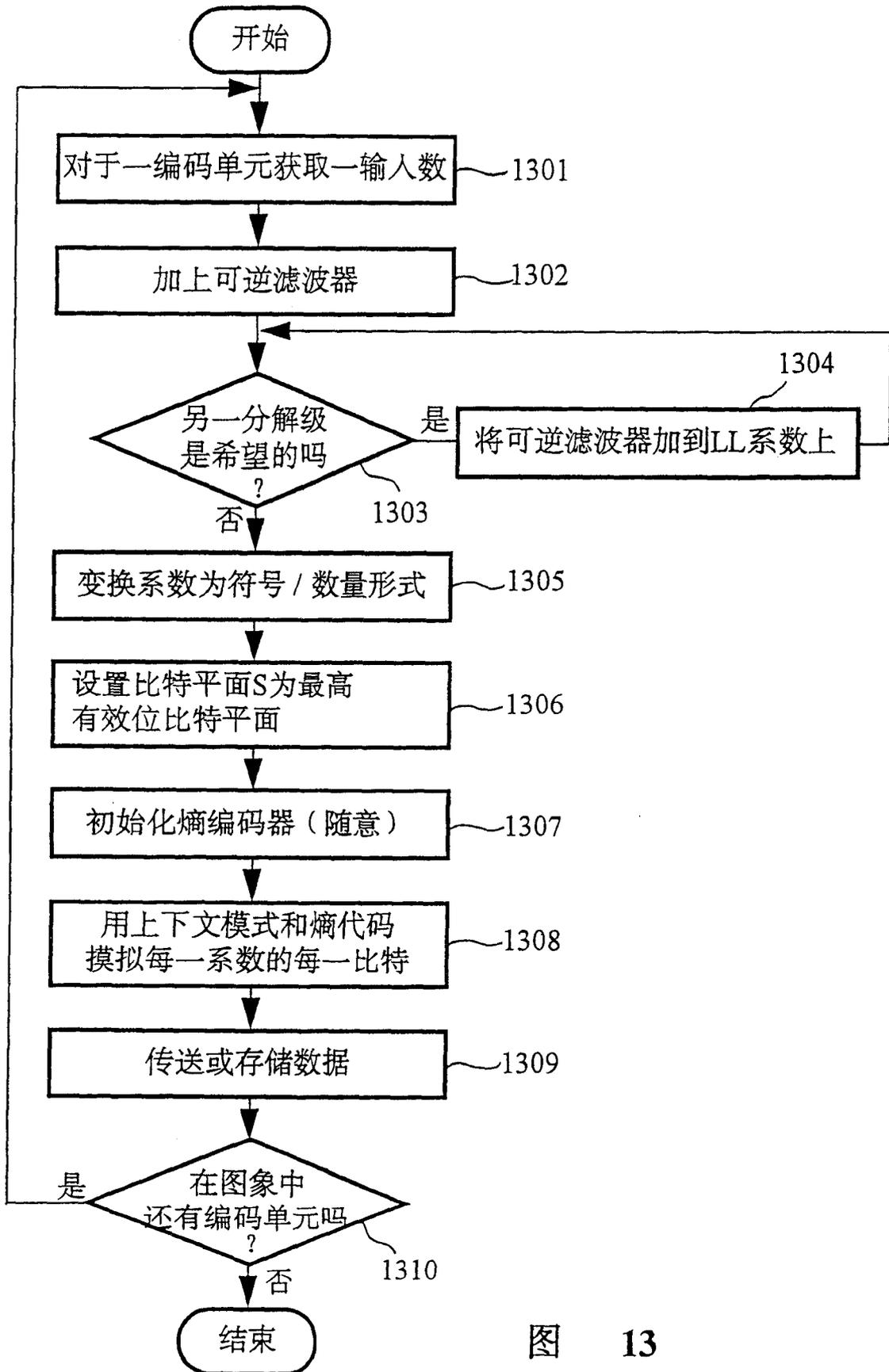


图 13

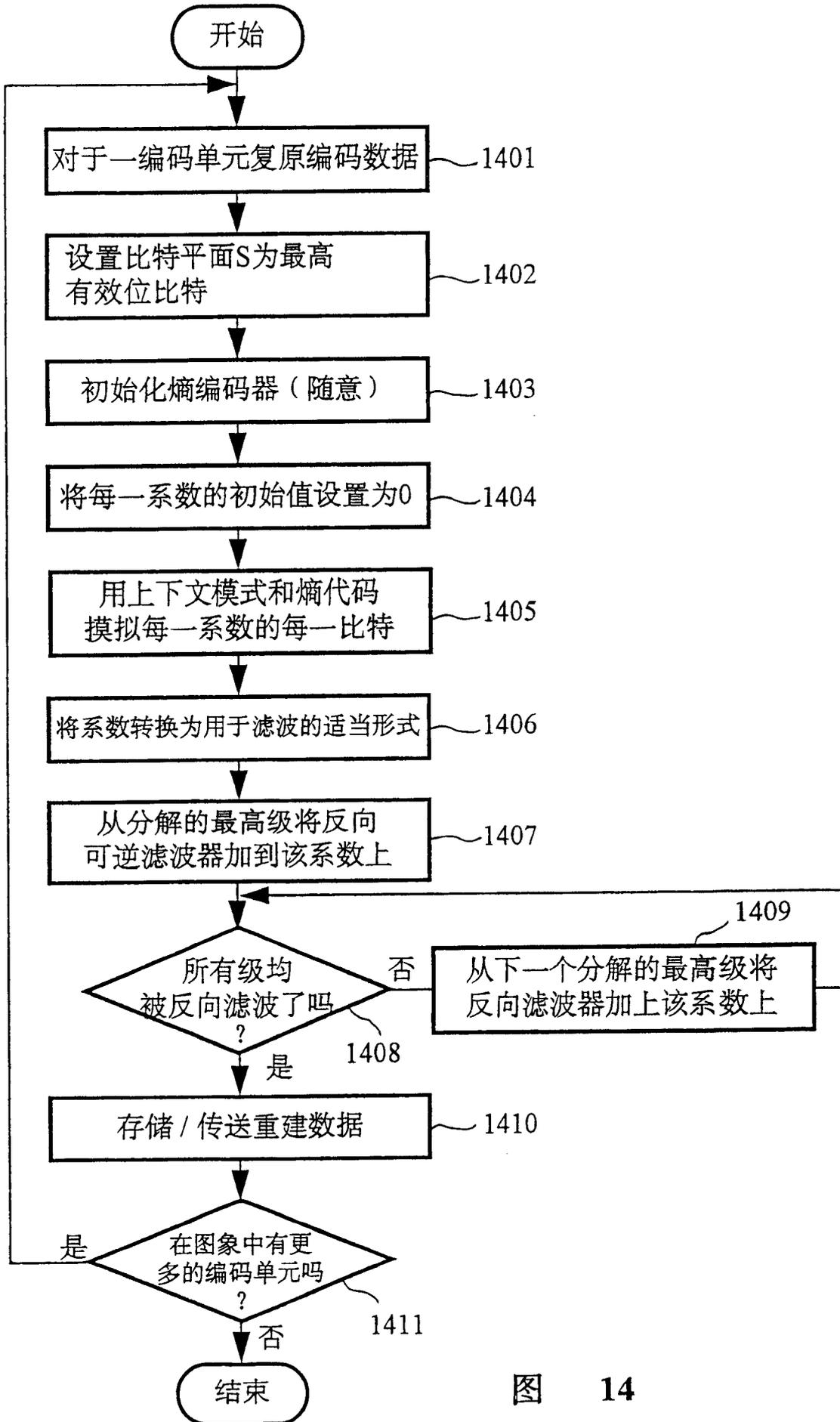


图 14

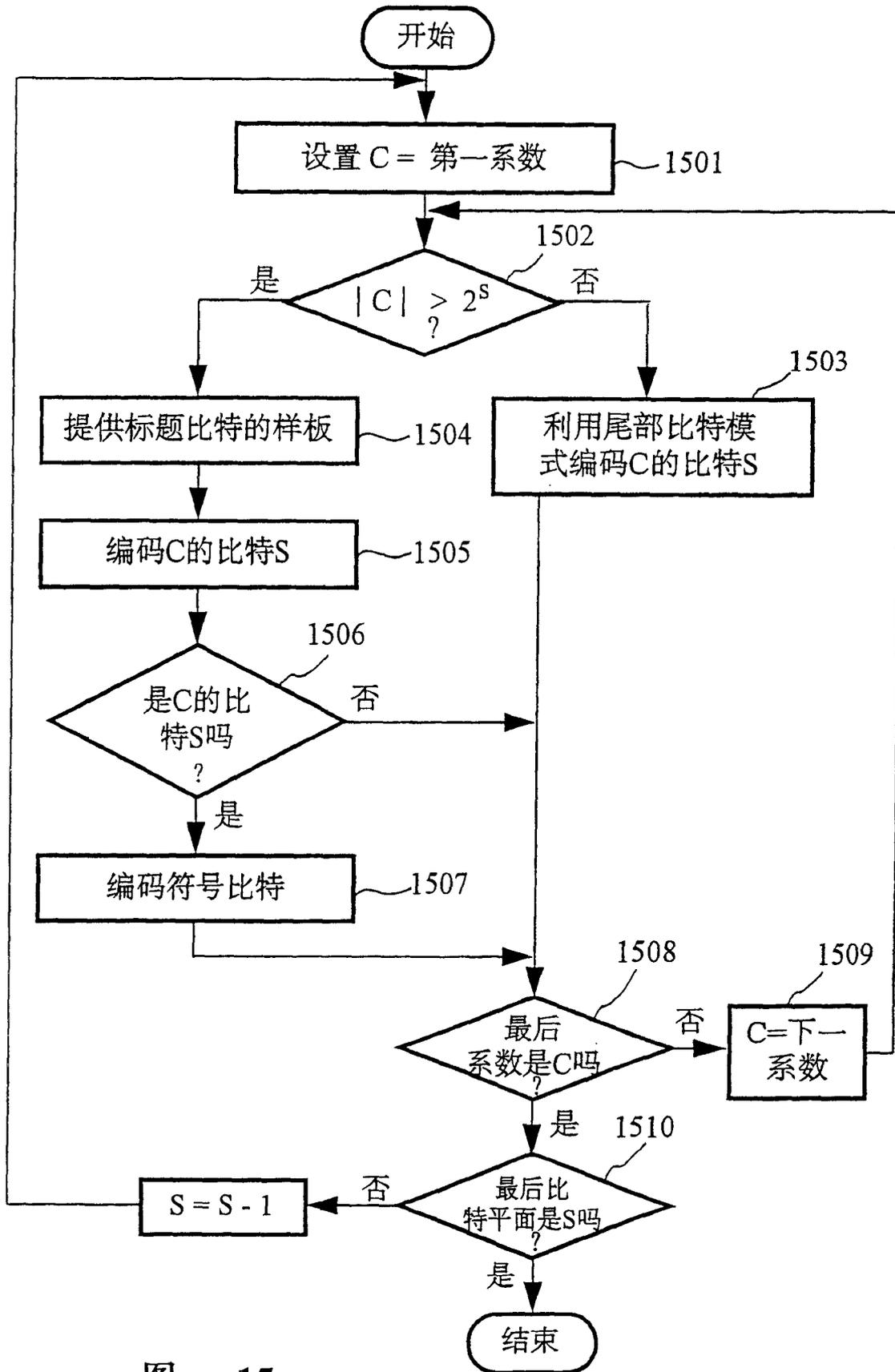


图 15

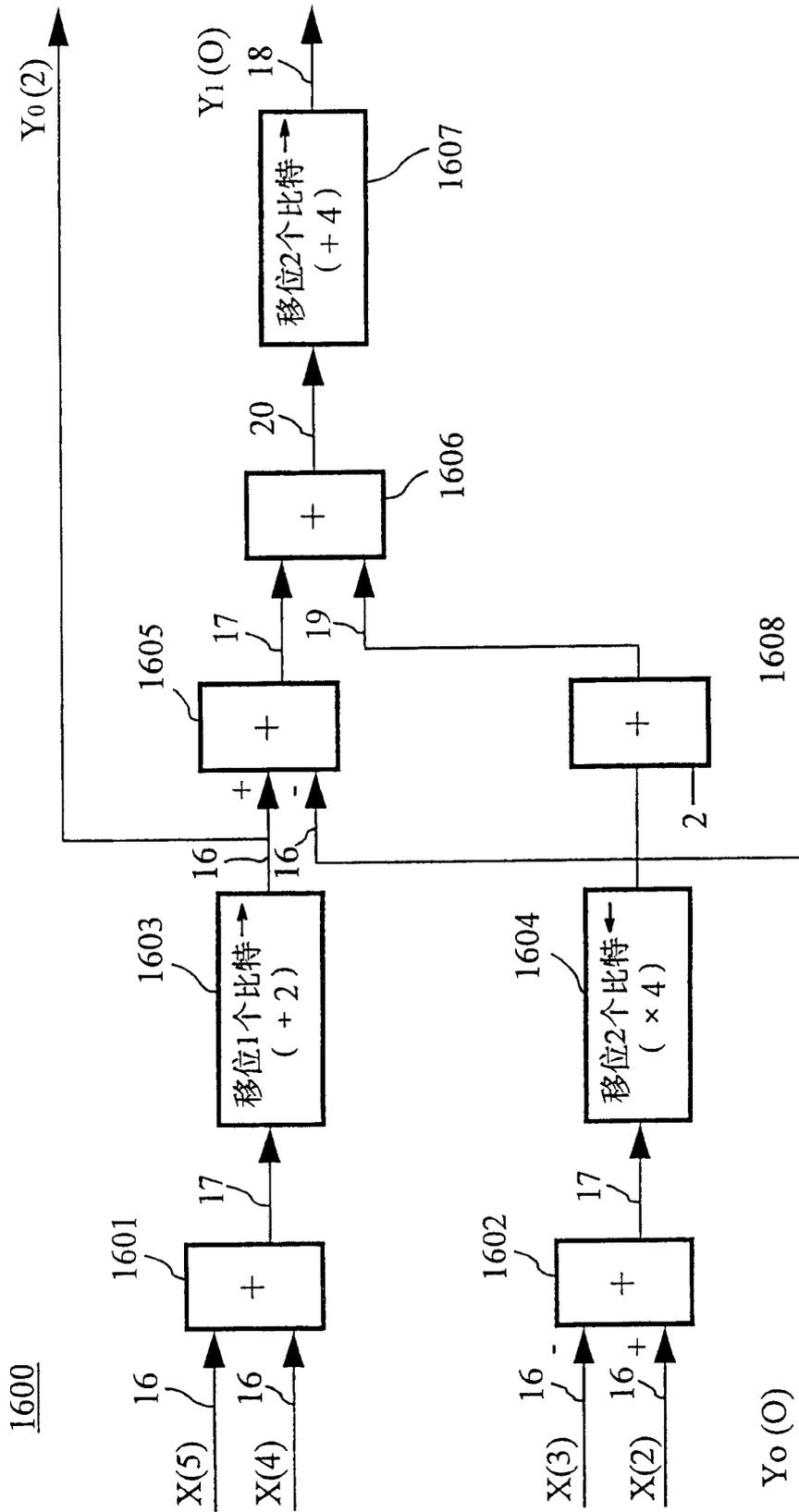


图 16

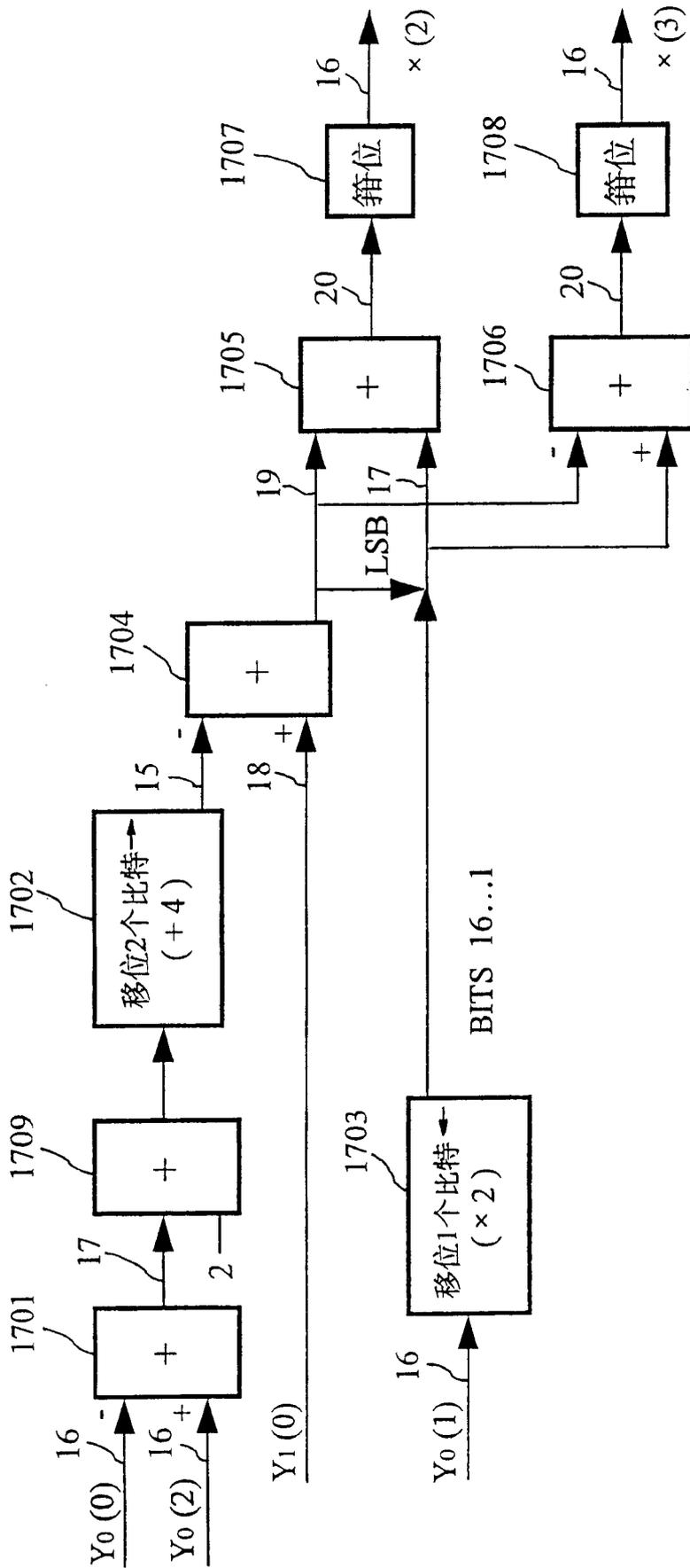


图 17

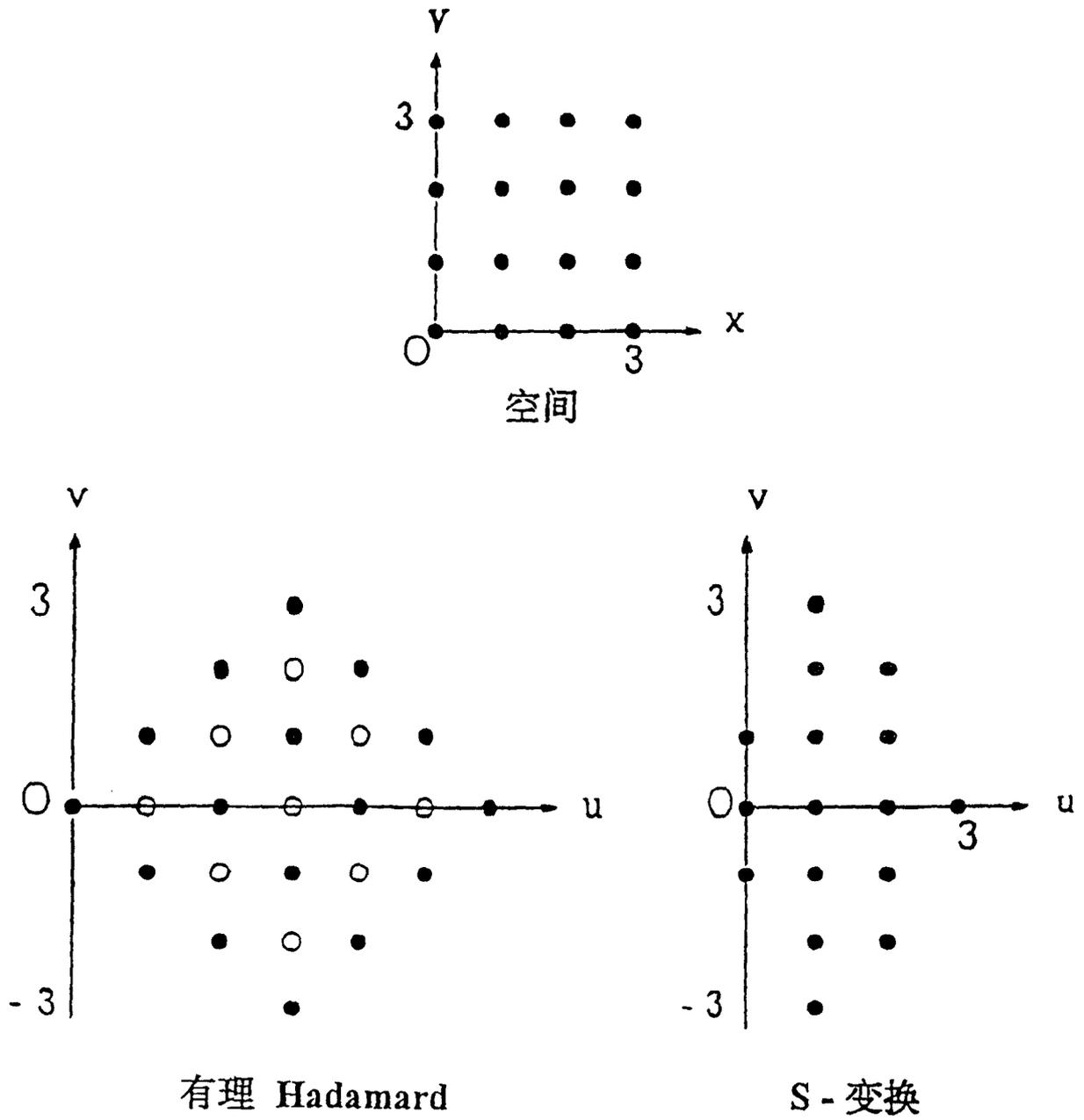


图 18

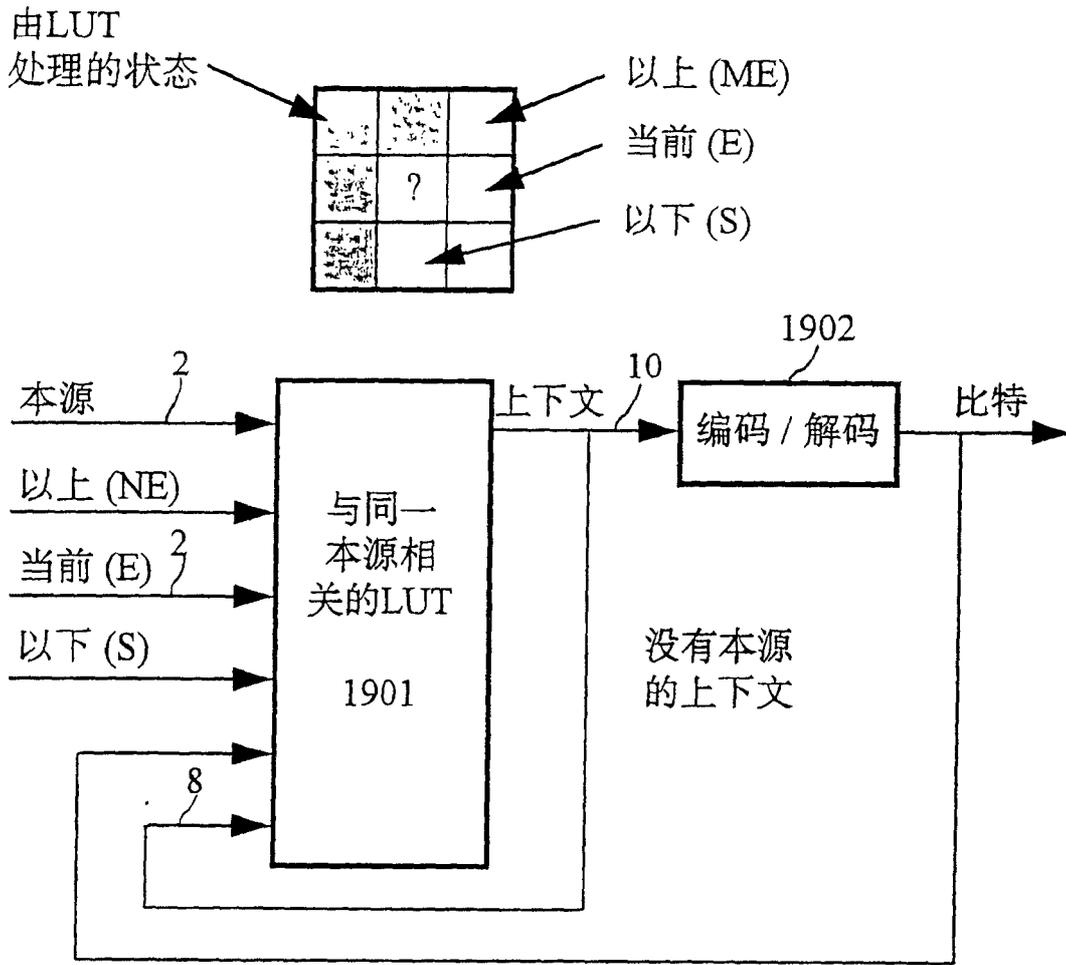


图 19A

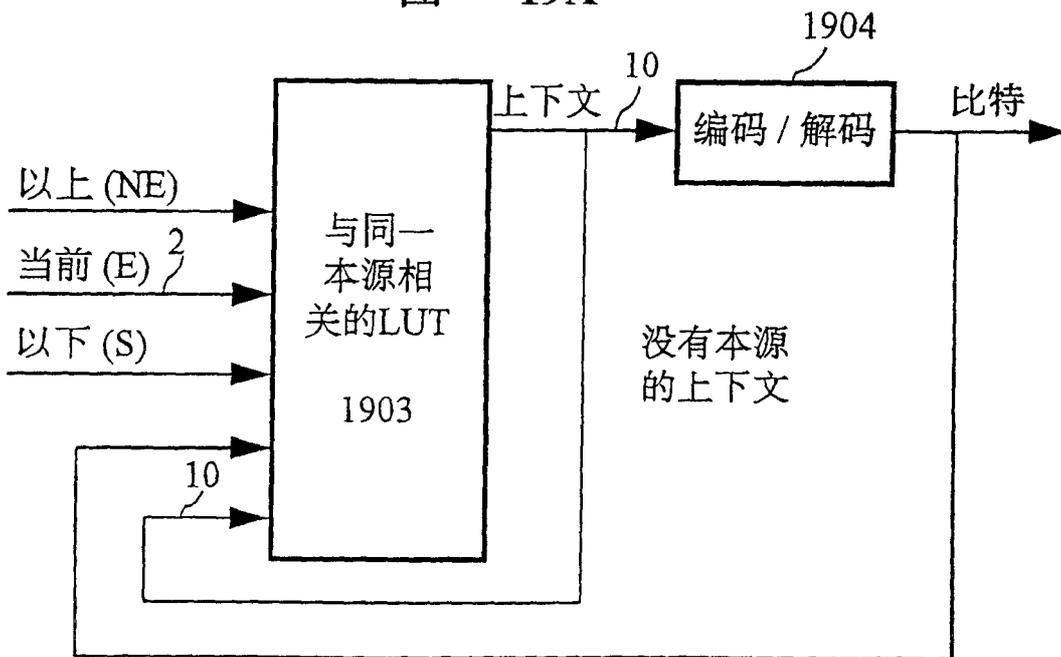


图 19B

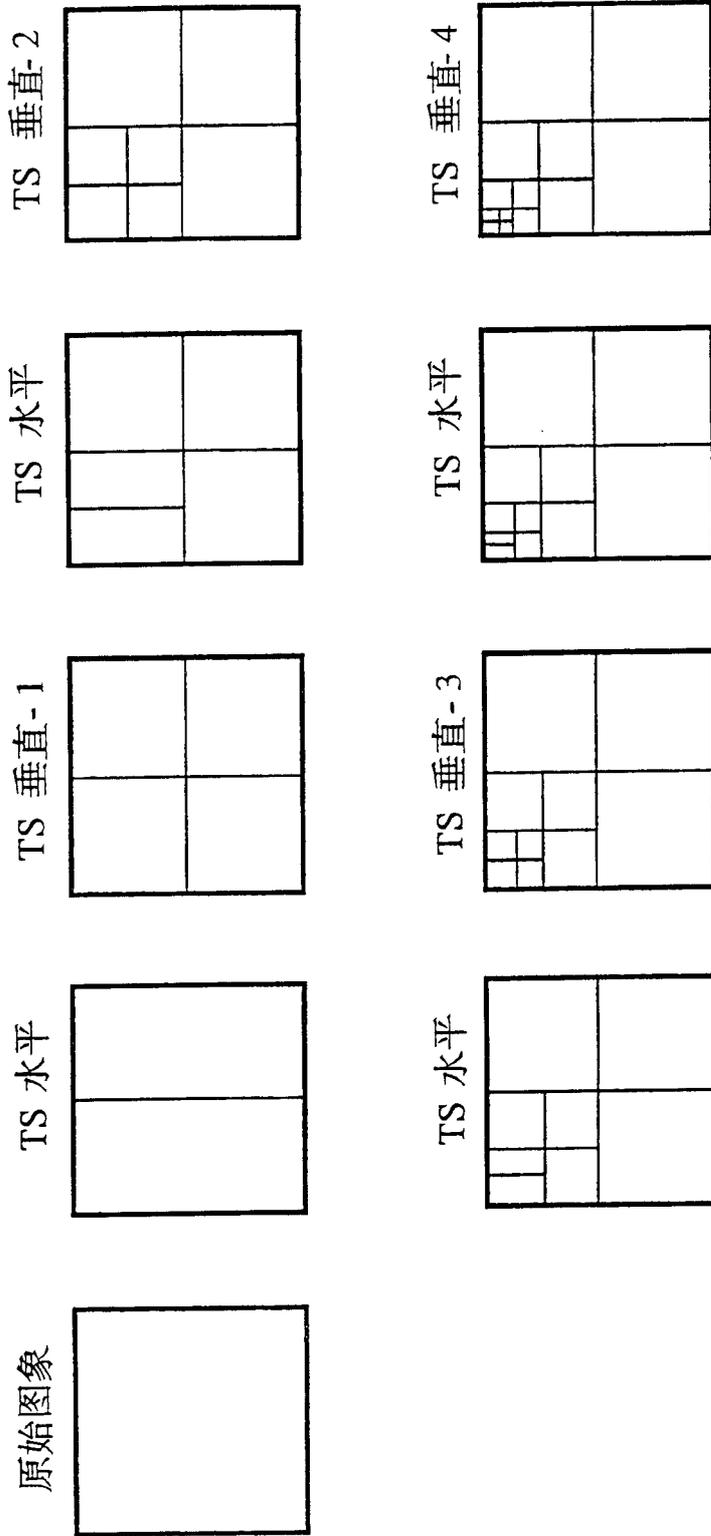


图 20

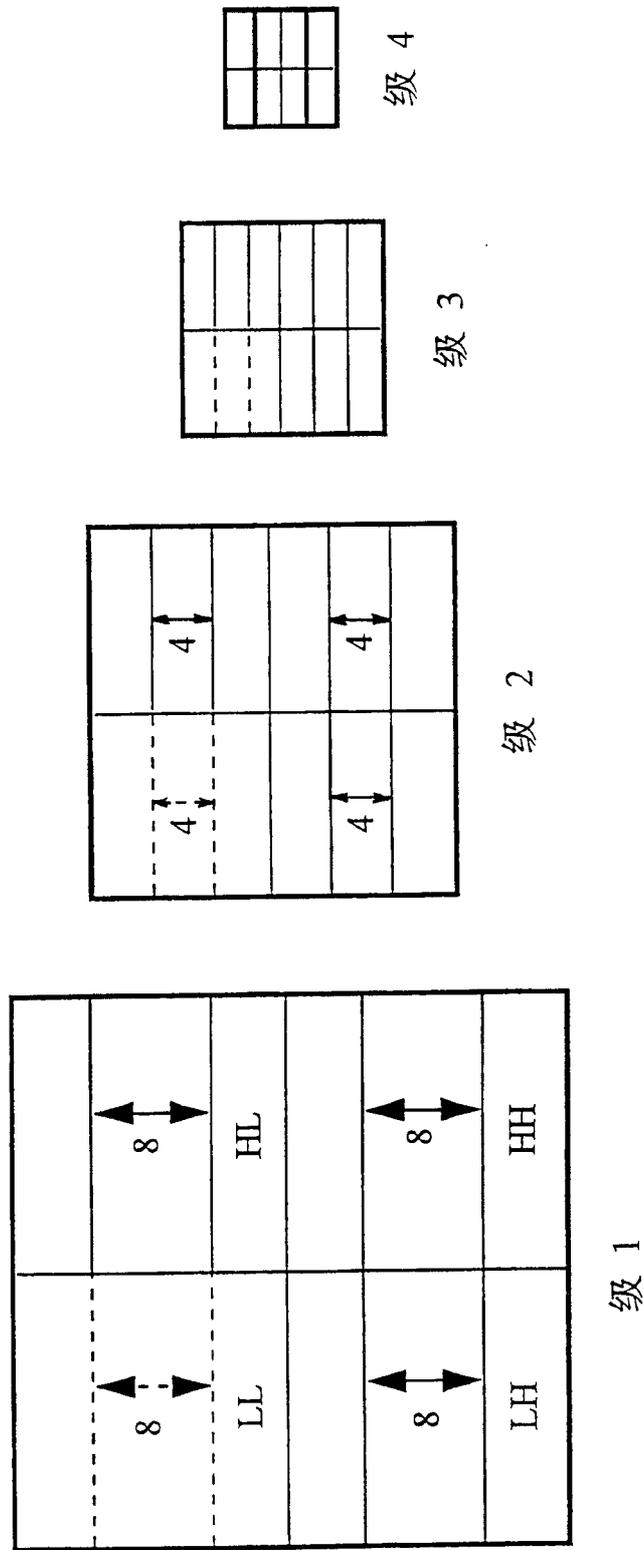


图 21

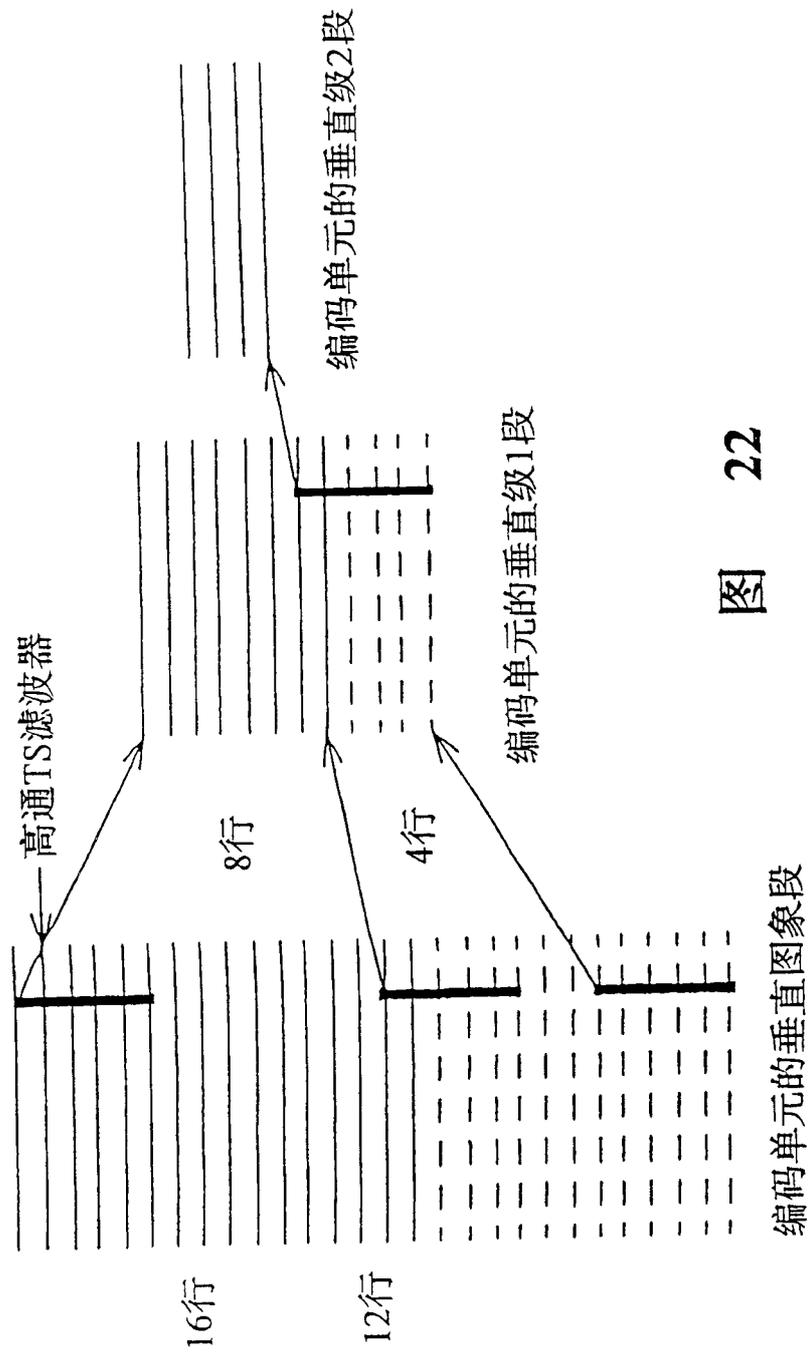


图 22

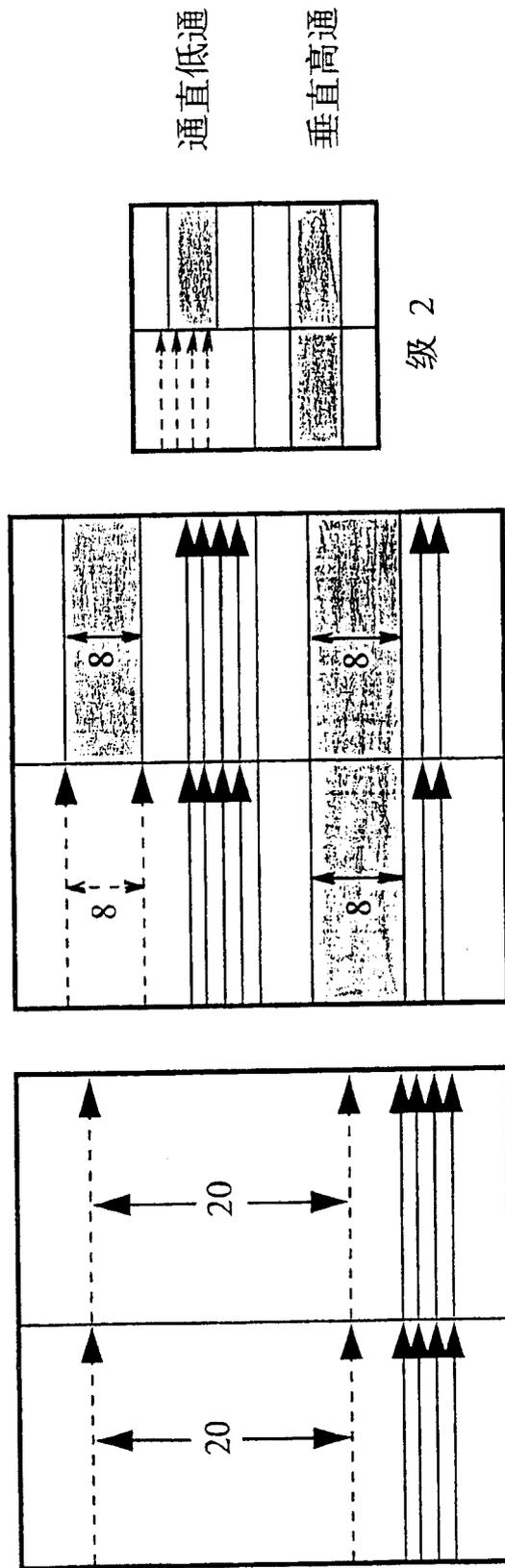


图 23

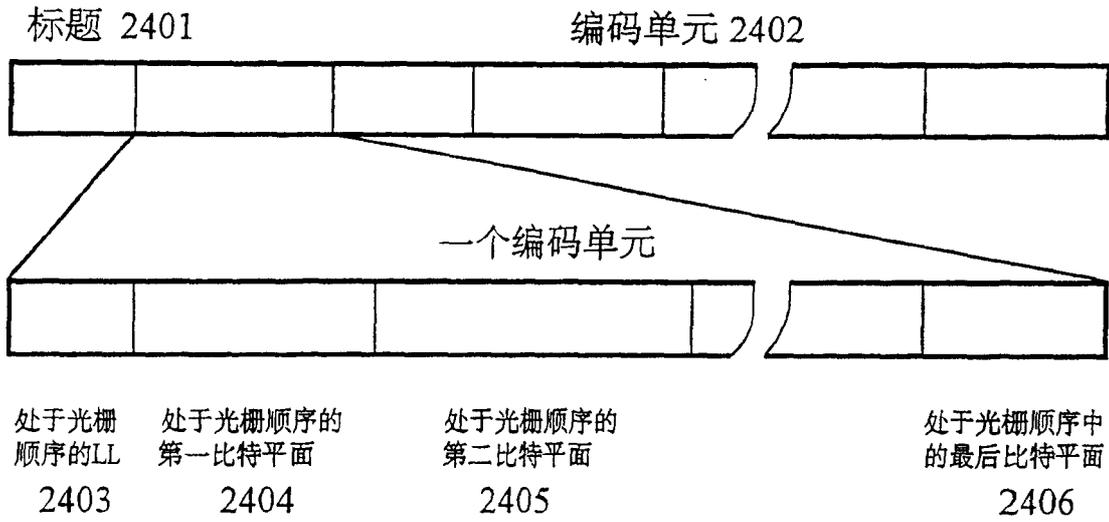


图 24A

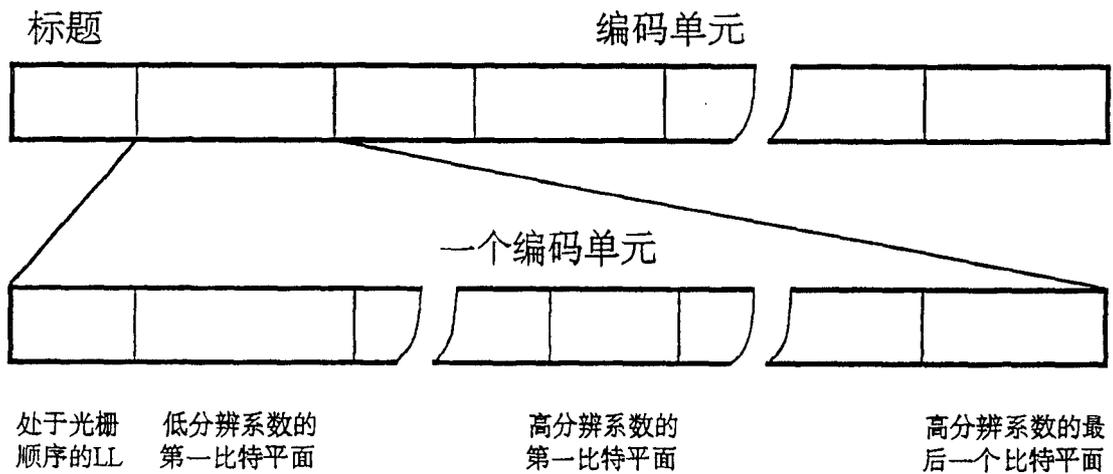


图 24B

| | | |
|----|---|----|
| NW | N | NE |
| W | P | E |
| SW | S | SE |

图 25

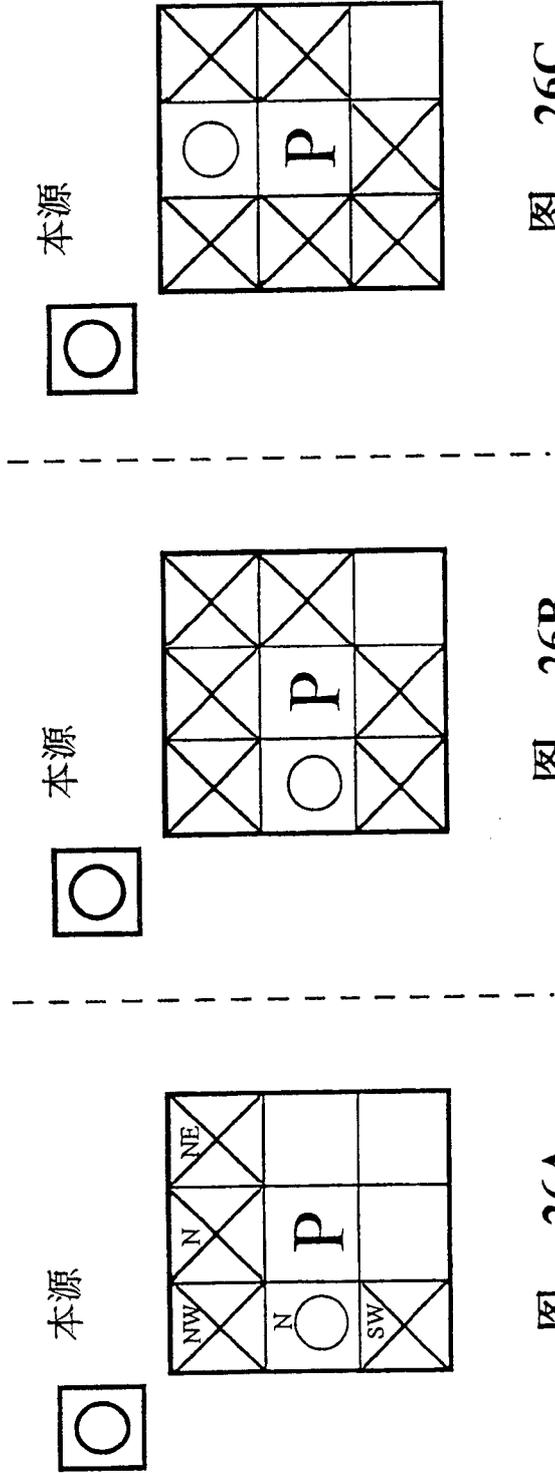


图 26A

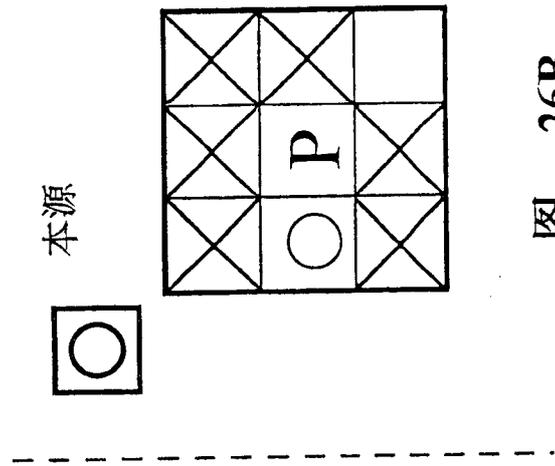


图 26B

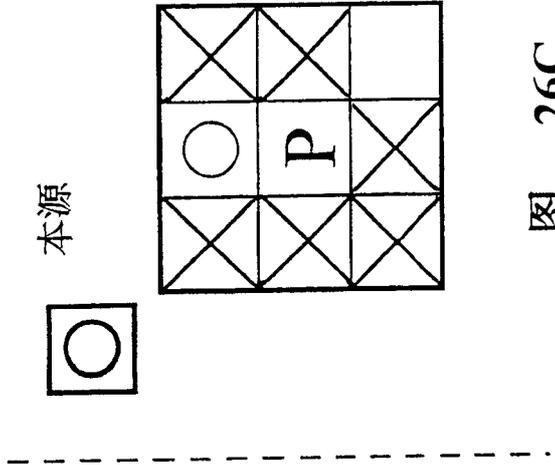


图 26C

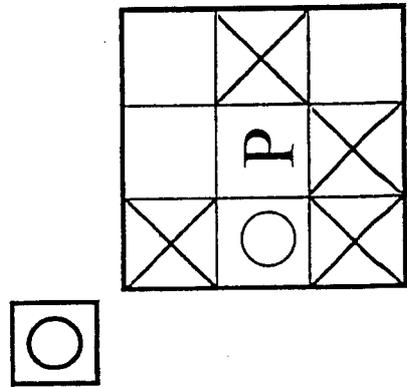


图 26D

X = 尾部接通比特 (1比特)
 O = 尾部接通信息 (2比特)

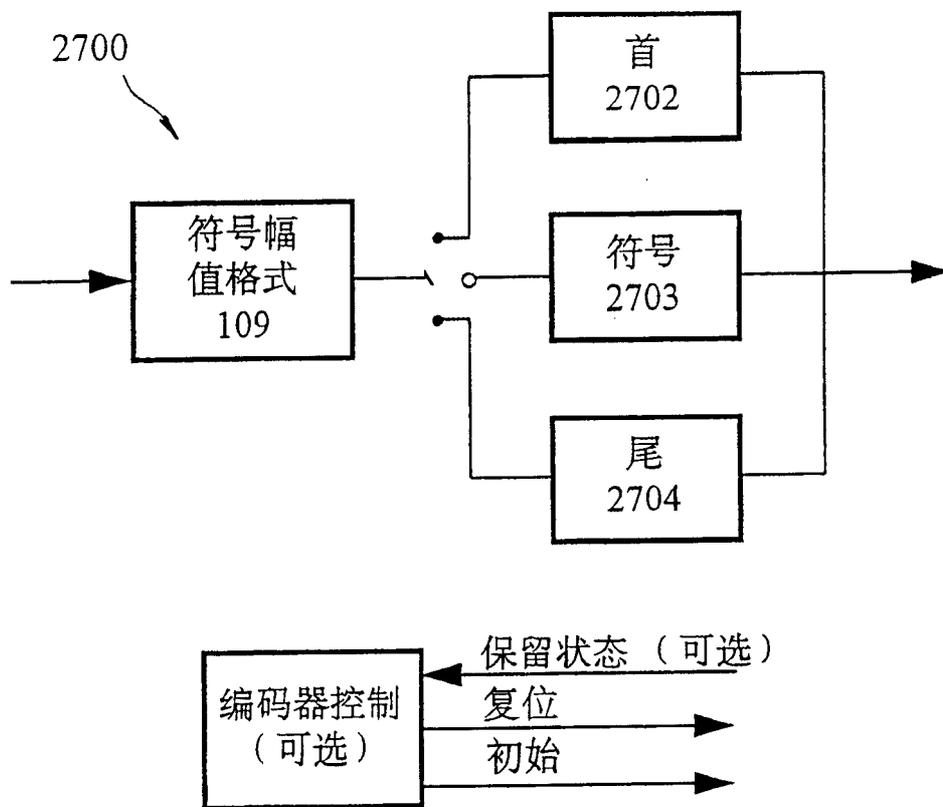


图 27

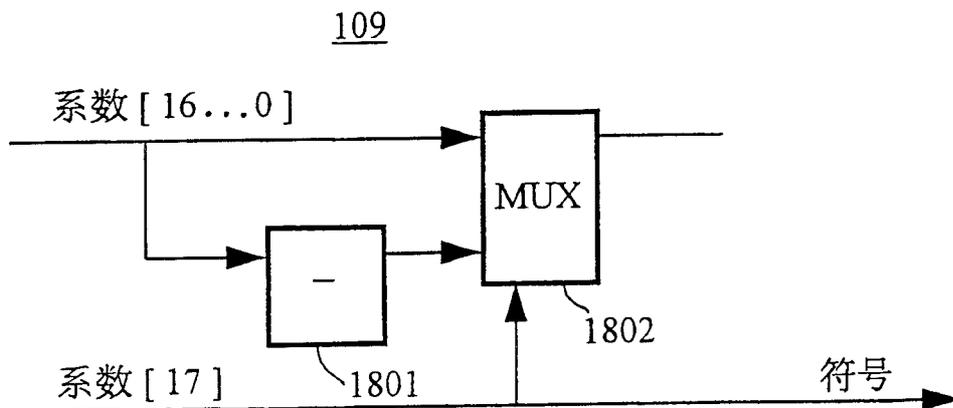


图 28

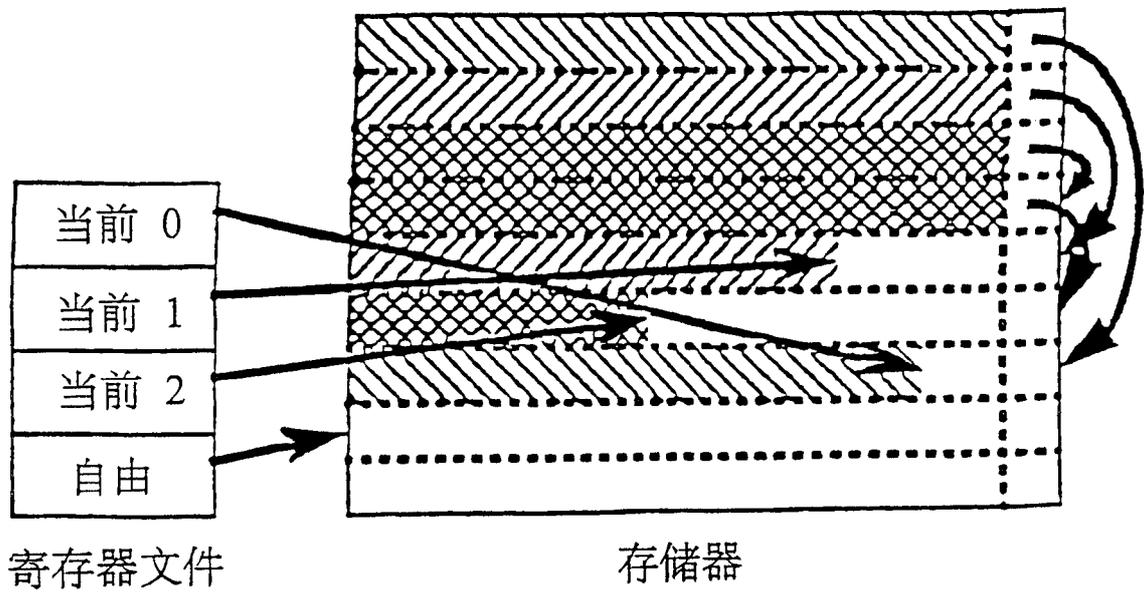


图 29

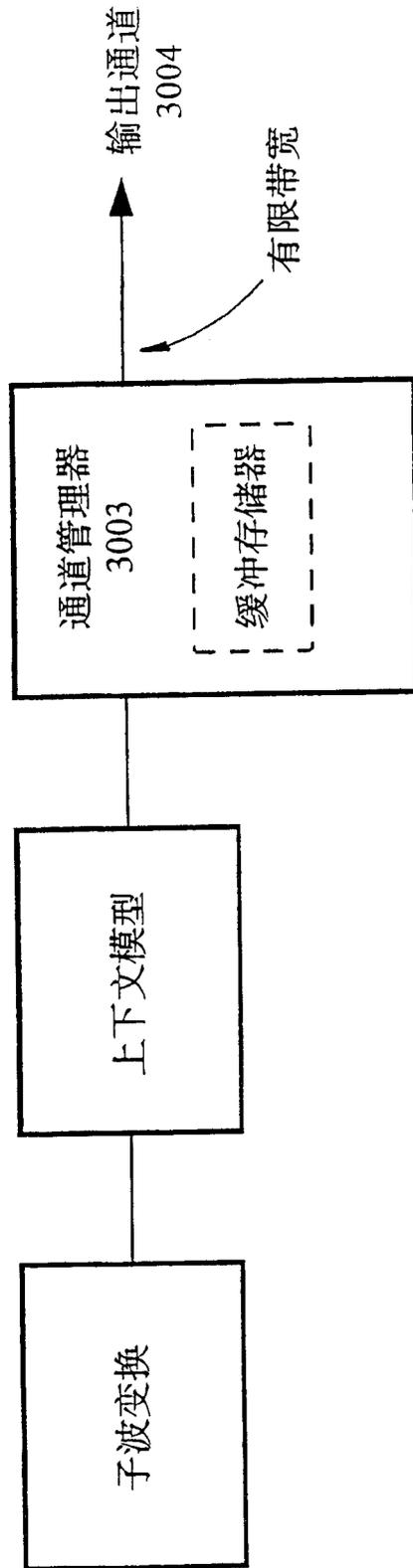


图 30

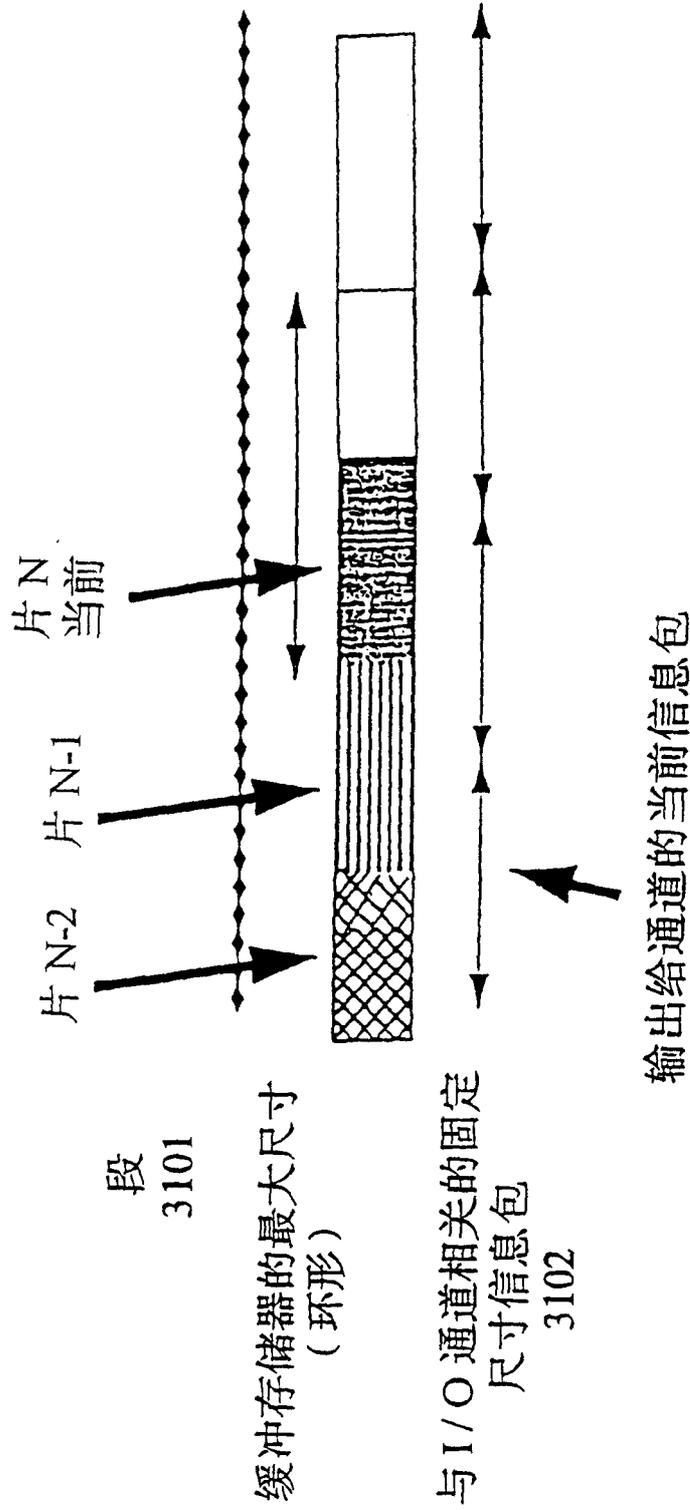


图 31

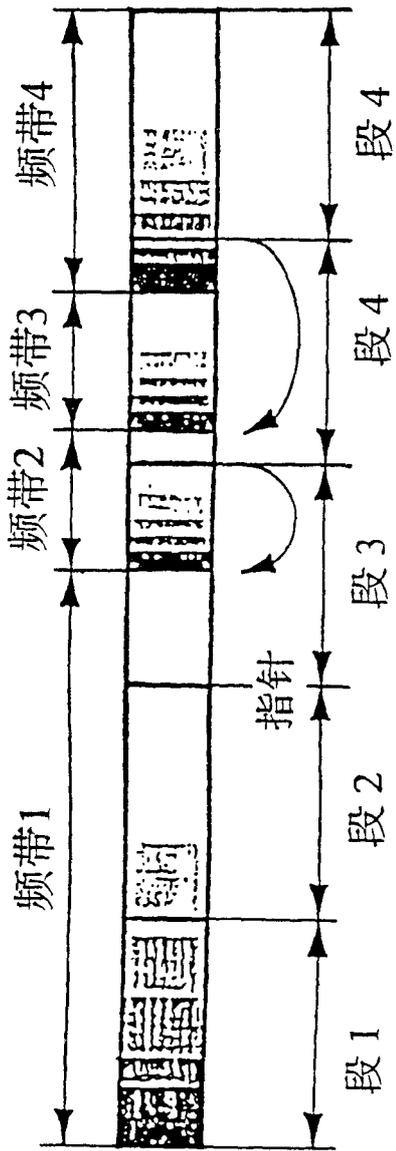


图 32

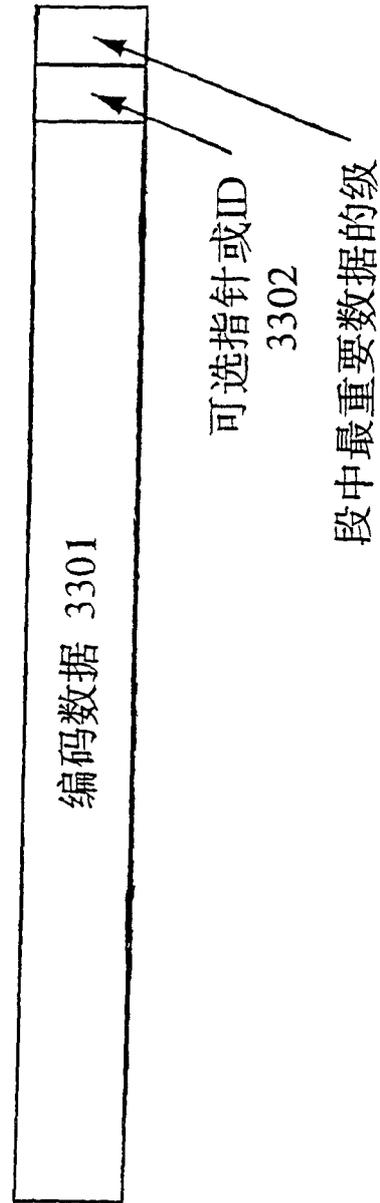


图 33

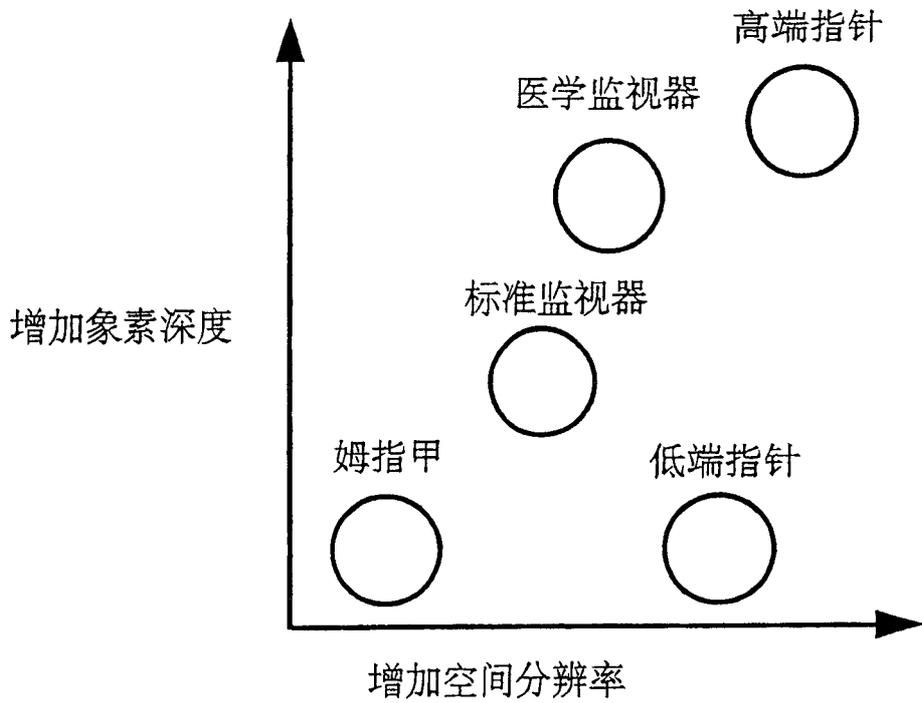


图 34

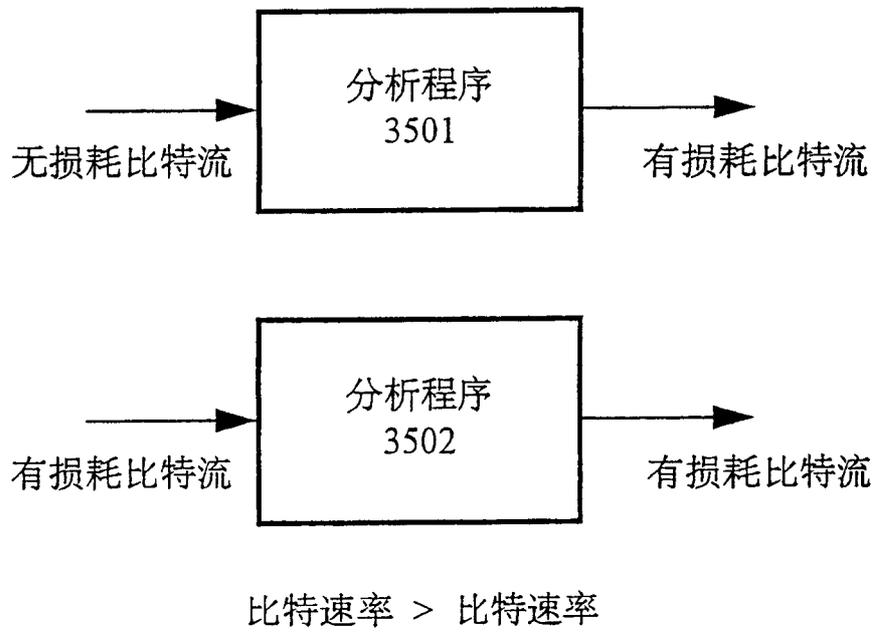


图 35