

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H04N 7/26 (2006.01)

G06T 9/00 (2006.01)



[12] 发明专利说明书

专利号 ZL 200510083504.5

[45] 授权公告日 2008 年 10 月 1 日

[11] 授权公告号 CN 100423583C

[22] 申请日 1997.9.30

[21] 申请号 200510083504.5

分案原申请号 03108545.8

[30] 优先权

[32] 1996.12.18 [33] EP [31] 96402785.8

[73] 专利权人 汤姆森消费电子有限公司

地址 美国印第安纳州

[72] 发明人 G·A·克拉纳维特

M·A·舒尔茨

[56] 参考文献

EP0660612A2 1995.6.28

EP0692911A2 1996.1.17

CN1052761A 1991.7.3

审查员 吴 琼

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 王 岳 张志醒

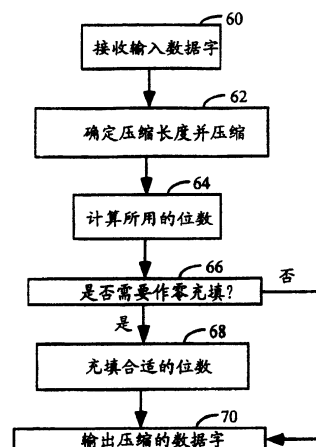
权利要求书 1 页 说明书 7 页 附图 2 页

[54] 发明名称

将数据压缩成固定长度数据块及解压的方法

[57] 摘要

本发明描述了以可变长度压缩产生固定长度压缩数据块的压缩和解压的装置和方法。压缩系统从一个数据块中接收 N 位数据字(60)并确定该数据字的可变压缩长度(62)。位计数器监视剩余的总位数(64)并确定是否已用足够的位数以确保可以填充固定长度压缩数据块(66)。如果已用足够的位数,则输出压缩的数据字(70)。如果没有,在输出之前给压缩的数据字填充一合适的位数(68),填充位可以是零位。



1、一种在用来对接收的表示提高分辨率视频图象信息的固定长度数据块的数字数据流作处理的系统中的信号处理方法，其中每一个数据块包括多个可变位长度数据字，该信号处理方法包括下述步骤：

对所述的视频信息进行译码和解压，以产生像素数据；

将所述的像素数据再压缩成固定长度数据块；

计数由再压缩上述像素数据所产生的位数；

按在上述计数步骤中计数的位数函数填充选取的再压缩像素数据来填充所述的固定长度数据块。

2、如权利要求1的方法，进一步包括下述步骤：

将在所述的再压缩步骤中压缩的再压缩数据存储于帧存储器中。

3、如权利要求1的方法，其中：

所述的数字数据流为高清晰度电视信号。

4、如权利要求1的方法，其中：

所述的再压缩步骤是在 MPEG 运动补偿网络中执行的，以产生最终的译码和解压的输出。

5、如权利要求1的方法，其中：

所述的再压缩步骤将所述的像素数据进行可变长度压缩。

6、如权利要求1的方法，其中：

所述的要求填充固定长度数据块的填充处理是分散在像素之中的。

7、如权利要求1的方法，其中：

所述的填充是零填充。

将数据压缩成固定长度数据块及解压的方法

本申请是申请号为 03108545.8 的分案专利申请的再分案申请。

技术领域

本发明涉及用固定长度输出块对数据进行有效编码的方法。

背景技术

数据压缩是数据传输和存储的一个重要特征。已知的压缩方法可以使数据在无数据丢失或丢失数据在允许范围的前提下占据较小的带宽或存储空间。有多种数据压缩方法，有不丢失数据的方法，也有丢失数据的方法，其中包括霍夫曼编码、量化和直接余弦变换。不同的压缩方法可实现不同的目的。

对于广播要求有高数据(传输)速率的图象数据来说，JPEG 和 MPEG 是两种众所周知的并且易被接受的数据压缩方法。在通常情况下，以这两种格式传输或存储的数据首先经过变换和量化处理，然后再作可变长度压缩以满足系统带宽或存储的要求。压缩的数据块并不局限于某一特定的长度。当出现数据块的末端时，在数据的尾部附加一个块结束(EOB)码或标志，以提醒系统当前数据块的结束和另一个数据块的开始。

然而在某些情况下，某些数据(例如图象数据)必须以固定长度数据块来储存或传输。使用可变长度压缩方法的已知系统保证对数据进行充分的压缩以使每一数据块适合于其分配的空间。使用小于分配的空间或带宽的压缩方法是很常见的。在这种情况下，系统(在数据的结尾)放置一个 EOB 码而不管剩余的空间，或用零或其他所期望的位模式充填剩余空间。但是，在处理剩余块空间时，(在数据的末尾)放置一个 EOB 码或充填零会使系统数据输入输出中断，从而降低了系统的效率。

当数据传输速率增加到系统输入输出的极限时，放置一个 EOB 码或作零充填处理会导致缓冲器中的数据溢出并导致数据跳行。解压时也可能存在类似的问题，从而会中止数据的输入输出并产生相同的非期望的结果。因此，提供一个能以可变长度压缩产生固定长度压缩数据块的系统是十分有益的，这种可变长度压缩方法能有效地利用存储

空间或输出带宽，而且放置 EOB 码或对剩余块空间充填零将不会中断数据流。

发明内容

根据本发明的原理，产生固定长度输出数据的压缩系统以可变长度压缩方法来压缩输入数据。有时候，用可变长度压缩方法来压缩输入数据块所用的位数小于固定长度输出块中所分配的位数。在这种情况下，压缩系统将在压缩过程中识别存在的剩余输出空间，并对所选取的压缩数据字作零充填，从而使最终压缩数据块的长度等于所分配的输出块的长度。

附图说明

图 1 示出了一个本发明压缩系统的示范性实施例。

图 2 是一个在图 1 系统中数据压缩的流程图。

图 3 是一个与图 1 和图 2 所示系统中执行的数据压缩相对应的数据解压流程图。

图 4 示出了一个本项发明的系统产生的数据流。

图 5 示出了一个本项发明所用的数据处理器。

具体实施方式

下文将描述在 MPEG 压缩系统中执行的操作，该系统要求固定长度再压缩数据块存储在帧存储器中。本发明并不仅仅局限于 MPEG 系统，它还可适用于任何能够将输入数据格式化成为固定长度压缩的输出数据的压缩系统。

图 1 示出了一个能输出预定的固定长度压缩数据块的压缩系统。可变长度压缩器 (VLC) 10 从数据源 (图中未示出) 中接收块数据。接收的数据可能已作过离散余弦变换处理而被压缩到某种程度。VLC 10 可用已知的方法压缩代表差分脉码调制 (DPCM) 值的预测数据。DPCM 处理技术是已知的，对该技术的说明已超出本文讨论的范围。

VLC 10 产生可变长度压缩数据。例如，VLC 10 可以将输入数据从 8 位转换成 3、4、5 或 6 位数据字。其压缩量的大小取决于各种变量，如输出网络的大小以及系统允许的数据丢失量等。例如，当提供给压缩的值相对较小时，可用 3 位压缩。而当提供给压缩的值相对较大时，可用 6 位压缩。提供给压缩的数值大小通常取决于在预测网络中输入数据值的相对变化。

例如，对于图象数据来说，其位置移动可忽略的图象帧和/或图象数据在压缩时通常会产生较小的预测误差，其位置移动较大的图象帧和/或多物体图象数据在压缩时容易产生较大的预测误差，但有时也可能产生小的预测误差。

当 VLC 10 产生可变长度压缩数据时，位计数器 20 跟踪所有的输入数据块所用的位数和每个数据字所用的位数。例如，如果在每个输出数据块中每个数据字平均用 4 位，并且在一个 8×8 的点阵中每个数据块由八位字组成，则固定长度压缩数据块输出必须为 256 位。输出数据块过长或过短均会使系统混乱，并在恢复时产生数据污染。设计参数为每个数据字 4 位，以此作为实例来说明在给定的压缩系统中可允许的数据丢失量。

位计数器 20 识别出当在输出块中的剩余位数不足以维持平均每个压缩字 4 位时，通知 VLC 10 将下一个输入的数据字以 3 位压缩。同样，位计数器 20 识别出当对给定数据块作 3 位压缩处理过于频繁时，则对给定数据块的每一个剩余数据字执行大于 4 位压缩处理。在这种情况下，当压缩系统确定以少于 5 位的压缩方法压缩后面的数据字提供的结果其数据丢失量在数据丢失参数所允许的范围之内时，位计数器给零填充器（Padder）30 发出一个控制信号，并以位计数器 20 确定的速率在压缩字的末端加零使平均每个数据字超过 4 位，以确保最终的压缩块为固定长度。在本文示范性系统中所充填的压缩字可以是 5 位或 6 位。位计数器 20 使用一种内部算法，并且在译码器中也使用同样的算法，来确定在数据块的末尾应充填多少位。

当然，填充压缩字的总长度不会超过输出寄存器、移位寄存器或其他将压缩数据从压缩器输出的硬件的长度。例如，通过利用一个桶形转换器将每一个压缩字的最大长度保持为输出寄存器的长度，这样，数据流就不会被零散的零充填所中断或终止。相反，使用数据末尾加零或在数据块的结尾附加一个 EOB 时，就会导致数据流中断或终止。当遇到 EOB 时，停止数据流动来移去数据块末尾的零充填或将指示针复位需要增加硬件和软件，而这在连续输入输出数据时是不必要的。

在译码器/解压器中处理进入数据的数据寄存器的长度与压缩器中输出寄存器的长度相同。在本实例中，由于最大压缩字长是 6 位，

因而这两种寄存器的长度也应该是 6 位。例如当最大压缩字长是 5 位时，则每一个处理压缩数据的寄存器也应是 5 位。通常使用最大字长可减小系统的复杂性。

压缩系统跟踪每一个压缩数据字的长度，并将数据字直接从位计算器 20 或从零填充器 30 直接传送到缓冲器/多路转换器 40 中。如有必要，在此给数据字添加附加信息并将数据字与附加信息串联在一起。在必要时，在数据压缩的同时位计数器 20 将通过保持所需的位数的方式来补偿附加信息。缓冲器/多路转换器 40 将压缩成固定长度的数据块传输到存储器 50 保存。另一方面，固定长度数据块也可以按要求传送或输出到其他允许固定数据长度或固定带宽的装置或数据路径中。

另一个实施例包括在 VLC 10 中执行零填充，如果压缩字不同，可将不需要作零填充的压缩字直接通过零填充器 30 而不加零，绕过图 1 所示的零填充器 30。

当需要重建驻存在存储器 50 中的压缩数据时，要执行与上文描述相反的解压处理。因此，对其相应的装置在此就不作进一步说明。但是，对于接收压缩数据的解压器来说，上述的零填充是适用的并且是必须的。解压器使用的算法是与跟踪所用的位数以及将控制信号传送给 VLC 10 和零填充器 30 所用的算法的逆算法。

图 2 和图 3 分别表示当将所选择的数据字进行零填充处理来产生固定长度压缩数据块时数据压缩和解压处理的流程图。首先参考图 1 来描述图 2 所示的流程。

在 60 步，VLC 10 接收 N 位数据字。62 步估算用来压缩接收的数据字的位数并压缩该数据字。位数是利用计算系统相关参数的算法和每个压缩字的平均位数以及当前数据块的数据字所用的位数来确定的。压缩长度一经确定，就可以压缩该数据字。位计数器 20 按照在 64 步的压缩算法所确定的位数将用来压缩的位数设置在各种计数器中。然后，如果数据字小于最大长度，在 66 步系统确定是否需要作零填充。如果需要作零填充，在 68 步在数据字末尾加上合适的位数并修改计数器。在 70 步输出压缩的数据字，输出的数据字可能作过零填充处理，也可能未作过零填充处理。

图 3 描述了对从上述压缩器中接收的作过零填充或未作零填充处理的数据字作解压处理的流程。在 80 步, 系统确定用来解压下一个固定长度数据块的数据字的位数, 包括填充的位数。在 82 步, 检索数据字, 并按照解压算法将所用的位数设置在各种跟踪解压处理的计数器中。在 84 步, 如果数据字作过填充处理, 则确定填充的位数, 如果有必要在 86 步执行去填充处理。在 88 步, 将去掉零填充的压缩数据字从去填充器传送到解压器, 如果数据字未作过零填充, 则将数据字从输入装置传输到解压器中。此时将数据字解压成原来的 N 位长。最后, 在 90 步, 将解压的数据字传输到输出网络。

根据本项发明的原理, 压缩/解压系统可以在不中断数据输入输出和处理的条件下, 连续产生固定长度的压缩数据块。如果在每个压缩块的末尾放置一个 EOB, 数据处理会停止来调整指示针的位置并执行其他的附加功能。如果在数据块的末尾充零以填充未用过的空间, 系统必须要用足够的时钟周期数来移去附加的零, 所用的时钟周期的数量取决于移位寄存器的长度。另一种方法是使移位寄存器有足够大的空间, 但这样做会增加系统的复杂性和花费。

图 4 示出了本项发明所解决的问题。图中所示的是两个从压缩网络输出到存储器中和从存储器输入到解压网络中的数据流。数据流 100 在其固定长度数据块的末尾填充有零。数据流 102 示出了根据上述发明的原理分散填充的零。例如, 压缩数据流可被压缩到百分之五十。对于一个 8×8 的数据块来说, 每个数据字有 8 位, 百分之五十压缩可将每一个固定长度数据块转换为 256 位。压缩的数据仍可由多路转换器以 8 位块移入或移出存储器。因此, 在百分之五十压缩网络中将数据从存储器中输出和输入到存储器中均需要 32 个时钟周期。数据流 100 和 102 包含界线指示位, 它表示单个压缩数据字, 而不是在某一个时钟周期中移位的位数。

在解压时, 数据必须从变位格式解压成原来的 8 位数据字。在数据流 100 中, 前 27 个时钟周期包含所有的数据信息, 剩余的 5 个时钟周期填充零, 以满足固定块长度的要求。解压器将利用来自 27 个输入时钟周期的数据在 64 个时钟周期内输出 64 个 8 位值。在压缩下一个数据块之前必须将 5 个剩余的含有零的输入时钟周期移出缓冲器。另一个去零的方法是将缓冲器指示针复位到下一个数据块。二者都要求

中止数据的输入输出，并且二者都需要用更复杂的硬件和软件来获取有效的数据。而且，如果缓冲器/存储器没有足够大的空间可供维持处理，一旦解压端的数据流停下来等待足够长的时间，压缩网络也会停止处理数据。

数据流 102 包含在选定的数据上分散充填的零。填充零使全部 32 个时钟周期的有效数据扩大。如上文所述，系统控制器知道有多少位表示当前正被解压的数据，包括多少位充填了零。在这一点，零充填会由于没有放慢系统的速度而不容易被觉察。这样可使固定长度的数据块连续流动而不中断数据通过压缩网络和解压网络。由于系统使用了现有的移位寄存器/桶形移位器来传送单个变位长度压缩数据字，因此降低了系统的复杂性。一旦解压了某一个压缩数据块的 256 位，下一步系统将准备检索在数据通道上的下一个压缩数据块的第 1 位。

图 5 是一个使用本项发明的实例性系统。该系统是一个高清晰度的电视接收器，能够以例如 MPEG 格式接收、译码和显示高清晰度广播信号。高清晰度电视（HDTV）信号输入装置 110 接收解调的信号并将它提供给装置 112 进行 MPEG 译码和解压。部分或全部的译码视频数据被传输到再压缩器 114，再压缩器 114 在空间域对数据进行再压缩来减小数据在帧存储器 116 中所占用的空间。在译码和解压下一个数据的同时，系统将部分译码的数据存储在帧存储器 116 中。然后从存储器 116 中检索部分被译码的数据，经解压器 118 解压、移动补偿装置 120 作补偿处理，并在加法器 122 中与来自装置 112 的数据进行组合，来重建最终的译码数据。移动补偿处理对上一个译码和解压的图象进行预测，产生最终的译码数据样品。移动补偿涉及到部分 I、P 和 B 帧的 MPEG 标准，因此，在此就不再作进一步的探讨。

再压缩器 114 将最终的译码数据进行再压缩，并将之存储在帧存储器 116 中，直到需要按上述方法重建和显示下一个帧信号为止。当需要显示再压缩的数据时，将再压缩的数据传送到解压器 124 并将之恢复成全位长度像素数据。然后将像素数据传送到显示处理器 126 作显示格式化处理并将处理后的数据输出给显示装置（图中未示出）。

按上述方法可将像素数据作 8×8 像素块处理。在压缩端，图 1 和图 2 所示的本项发明的系统和流程安装在再压缩器 114 中；而在解压

端,图 1 和图 2 所示的本项发明的系统和流程安装在解压器 118 和 124 (图 3) 中。在这两种情况下解压器的操作完全相同。



图 1

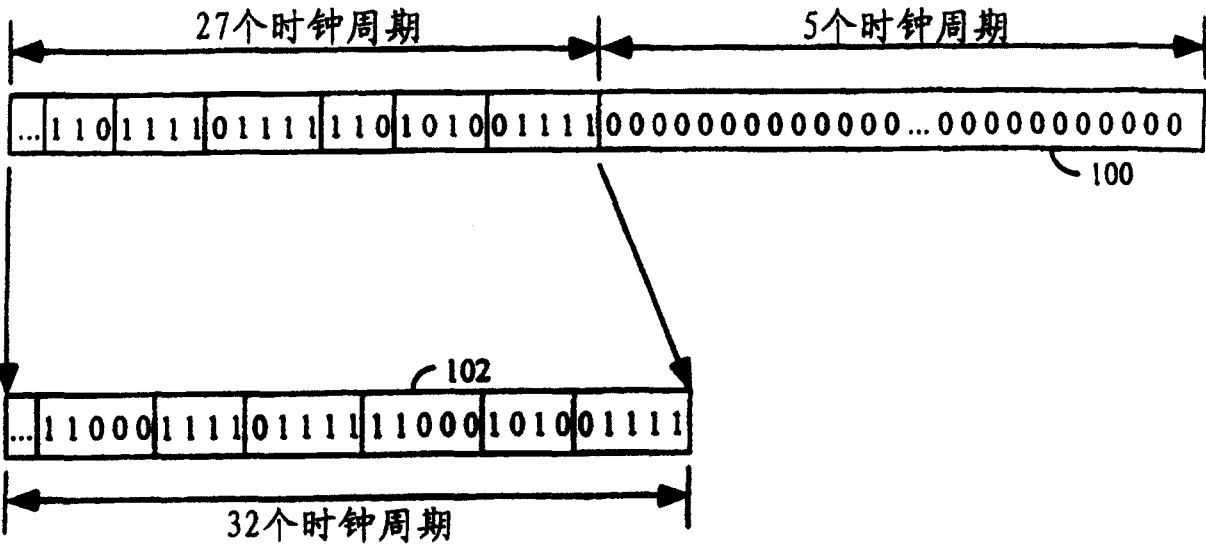


图 4

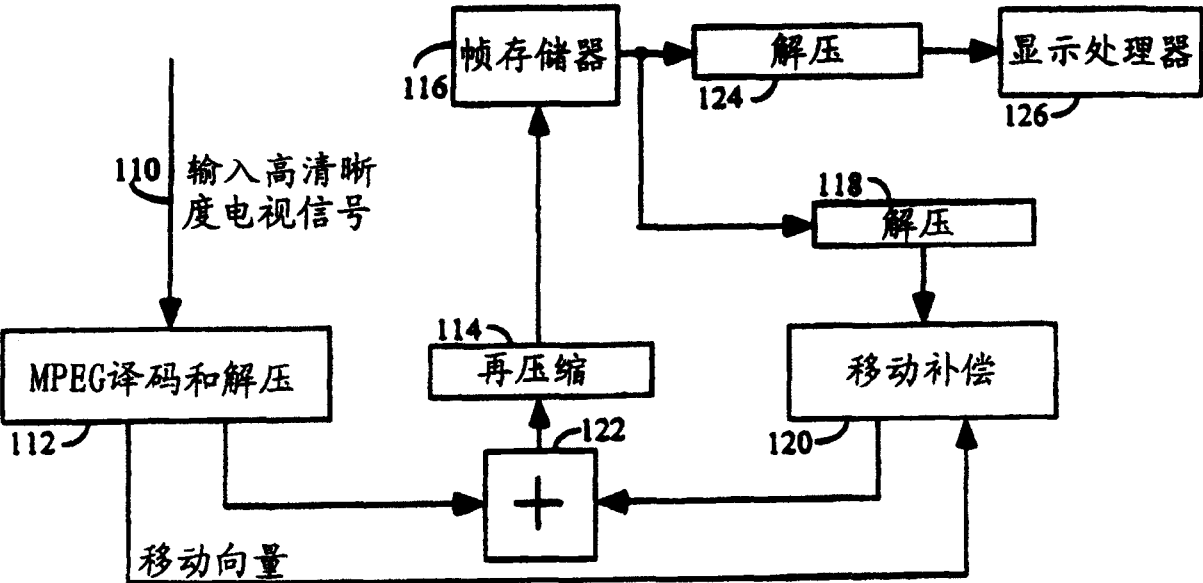


图 5

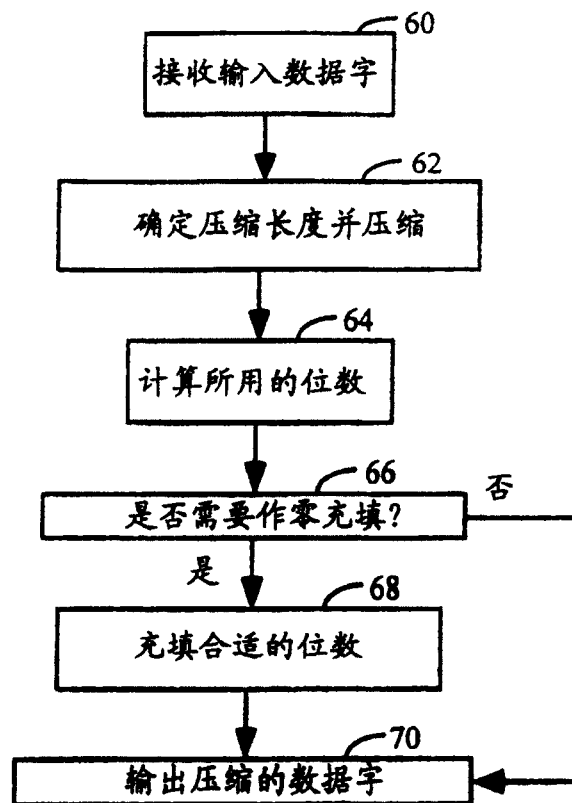


图 2

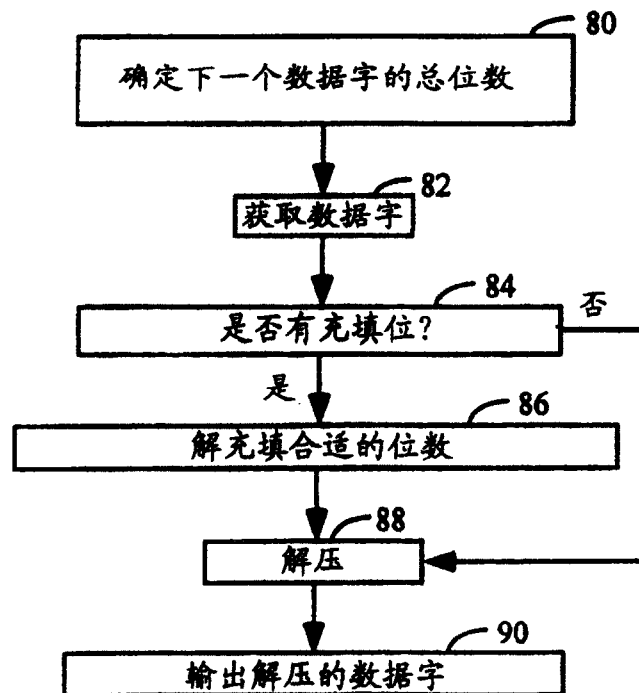


图 3