



[12] 发明专利说明书

专利号 ZL 02816523.3

[45] 授权公告日 2007 年 5 月 30 日

[11] 授权公告号 CN 1319017C

[22] 申请日 2002.7.2 [21] 申请号 02816523.3

[30] 优先权

[32] 2001. 7. 2 [33] US [31] 09/898,284

[86] 国际申请 PCT/US2002/021157 2002.7.2

[87] 国际公布 WO2003/005700 英 2003.1.16

[85] 进入国家阶段日期 2004.2.23

[73] 专利权人 高通股份有限公司

地址 美国加利福尼亚州

[72] 发明人 K·斯亚加拉严

[56] 参考文献

US5727068A 1998.3.10

审查员 王 艳

[74] 专利代理机构 上海专利商标事务所有限公司

代理人 包于俊

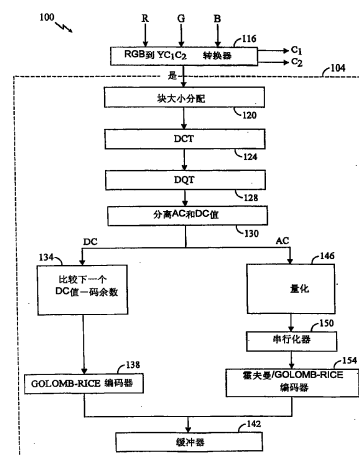
权利要求书 5 页 说明书 15 页 附图 11 页

[54] 发明名称

使用 GOLOMB - RICE 的无损失帧内编码的设备和方法

[57] 摘要

数字频域数据的帧内编码的一种设备和方法。数据包括多个片，每片具有多个块(120)。每个块具有 DC 分量和多个 AC 分量(130)。对每片的第一 DC 分量值进行编码。每片的每个后续的 DC 分量值表示为该值和它前面的 DC 分量值之间的差值。为了无损失编码，使用 Golomb - Rice (138) 对每片的初始 DC 分量值和差值进行编码。



1. 一种对数字频域数据进行无损失、帧内编码的方法，所述数据包括每片具有多个块的多个片，每块具有 DC 分量和多个 AC 分量，其特征在于，所述方法包括下列步骤：

接收第一片的第一 DC 分量值；

对第一片的第一 DC 分量值进行编码；

接收第一片的第二 DC 分量值；

对第一片的第二 DC 分量值进行编码作为余数值，其中余数值是第一片的第一 DC 分量值和第一片的第二 DC 分量值之间的差值以及第一片的第一 DC 分量值和第一片的第二 DC 分量值之间差值的符号的函数。

2. 如权利要求 1 所述的方法，其特征在于，所述数字频域数据是图像数据。

3. 如权利要求 1 所述的方法，其特征在于，所述数字频域数据是音频数据。

4. 如权利要求 1 所述的方法，其特征在于，余数值的编码使用 Golomb-Rice 编码。

5. 如权利要求 1 所述的方法，其特征在于，所述对第一片的第二 DC 分量值进行编码作为余数值包括使用下述公式：

$$y = \begin{cases} 2x, & \text{如果 } x \geq 0, \text{ 否则} \\ 2|x|-1 \end{cases}$$

其中 x 是待编码的余数值。

6. 在数字影院系统中，一种压缩数字图像的方法，所述图像包括像素数据，所述像素数据分成彩色分量，其特征在于，所述方法包括下列动作：

读出像素数据的彩色分量的组，每个组包括 DC 值和多个 AC 值；

产生块大小分配，以把像素的彩色分量组分割成像素数据的子块；
把像素数据的子块变换成相应的 DC 和 AC 频域表示；
把 AC 频域表示定标成 AC 数据流，其中定标的动作是基于与图像的质量相关的质量量度；
使 AC 数据流串行化；
使用指数分布对串行化的 AC 数据进行行程编码；
对第一 DC 频域表示进行编码；以及
对第二 DC 频域表示进行编码作为余数值，其中所述余数值是第二 DC 频域表示和第一 DC 频域表示之间的差值。

7. 如权利要求 6 所述的方法，其特征在于，所述定标的动作进一步包括把频率加权的屏蔽提供给所述像素数据的子块，使得频率加权屏蔽对于人类视觉系统更敏感的图像部分给予突出，并且对于人类视觉系统不很敏感图像部分给予较少的突出。

8. 如权利要求 6 所述的方法，其特征在于，所述变换的动作执行离散余弦变换。

9. 如权利要求 8 所述的方法，其特征在于，所述变换的动作执行离散余弦变换后，接着执行微分四叉树变换。

10. 如权利要求 6 所述的方法，其特征在于，所述对第二 DC 频域表示进行编码还是余数值的符号的函数。

11. 如权利要求 10 所述的方法，其特征在于，所述对第二 DC 频域表示进行编码作为余数值包括使用下述公式：

$$y = \begin{cases} 2x, & \text{如果 } x \geq 0, \text{ 否则} \\ 2|x| - 1 \end{cases}$$

其中 x 是待编码的余数值。

12. 一种对数字频域数据进行无损失、帧内编码的设备，所述数据包括每

片具有多个块的多个片，每块具有 DC 分量和多个 AC 分量，其特征在于，所述设备包括：

接收第一片的第一 DC 分量值的装置；

对第一片的第一 DC 分量值进行编码的装置；

接收第一片的第二 DC 分量值的装置；

对第一片的第二 DC 分量值进行编码作为余数值的装置，其中余数值是第一片的第一 DC 分量值和第一片的第二 DC 分量值之间的差值以及第一片的第一 DC 分量值和第一片的第二 DC 分量值之间差值的符号的函数。

13. 如权利要求 12 所述的设备，其特征在于，所述数字频域数据是图像数据。

14. 如权利要求 12 所述的设备，其特征在于，所述数字频域数据是音频数据。

15. 如权利要求 12 所述的设备，其特征在于，余数值的编码使用 Golomb-Rice 编码。

16. 如权利要求 12 所述的设备，其特征在于，所述对第一片的第二 DC 分量值进行编码作为余数值的装置使用下述公式：

$$y = \begin{cases} 2x, & \text{如果 } x \geq 0, \text{ 否则} \\ 2|x|-1 \end{cases}$$

其中 x 是待编码的余数值。

17. 在数字影院系统中，一种压缩数字图像的设备，所述图像包括像素数据，所述像素数据分成彩色分量，其特征在于，所述设备包括：

读出像素数据的彩色分量的组的装置，每个组包括 DC 值和多个 AC 值；

产生块大小分配，以把像素的彩色分量组分割成像素数据的子块的装置；

把所述像素数据的子块变换成相应的 DC 和 AC 频域表示的装置；

把 AC 频域表示定标成 AC 数据流的装置，其中定标的动作是基于与图像的质量相关的质量量度；

使 AC 数据流串行化的装置；

使用指数分布对串行化的 AC 数据进行行程编码的装置；

对第一 DC 频域表示进行编码的装置；以及

对第二 DC 频域表示进行编码作为余数值的装置，其中所述余数值是第二 DC 频域表示和第一 DC 频域表示之间的差值。

18. 如权利要求 17 所述的设备，其特征在于，所述定标的动作进一步包括把频率加权的屏蔽提供给所述像素数据的子块，使得频率加权屏蔽对于人类视觉系统更敏感的图像部分给予突出，并且对于人类视觉系统不很敏感图像部分给予较少的突出。

19. 如权利要求 17 所述的设备，其特征在于，所述变换的动作执行离散余弦变换。

20. 如权利要求 19 所述的设备，其特征在于，所述变换的动作执行离散余弦变换后，接着执行微分四叉树变换。

21. 如权利要求 17 所述的设备，其特征在于，所述对第二 DC 频域表示进行编码的装置还是余数值的符号的函数。

22. 如权利要求 21 所述的设备，其特征在于，所述对第二 DC 频域表示进行编码作为余数值的装置使用下述公式：

$$y = \begin{cases} 2x, & \text{如果 } x \geq 0, \text{ 否则} \\ 2|x| - 1 \end{cases}$$

其中 x 是待编码的余数值。

23. 一种对数字频域数据进行无损失、帧内编码的设备，其特征在于，所述数据包括多个块的，每块具有 DC 分量和多个 AC 分量，所述设备包括：

接收机，配置成接收连续片的连续的 DC 分量值；

编码器，配置成对每片的第一 DC 分量值进行编码，并对连续的 DC 分量值进行编码作为余数值，其中每个余数值是第一片的当前 DC 分量值和第一片

的第二 DC 分量值之间的差值以及第一片的第一 DC 分量值和第一片的第二 DC 分量值之间差值的符号的函数。

24. 如权利要求 23 所述的设备, 其特征在于, 所述数字频域数据是图像数据。

25. 如权利要求 23 所述的设备, 其特征在于, 所述数字频域数据是音频数据。

26. 如权利要求 23 所述的设备, 其特征在于, 余数值的编码使用 Golomb-Rice 编码。

27. 如权利要求 23 所述的设备, 其特征在于, 使用下述公式, 将第一片的第二 DC 分量值编码为一余数值:

$$y = \begin{cases} 2x, & \text{如果 } x \geq 0, \text{ 否则} \\ 2|x|-1 \end{cases}$$

其中 x 是待编码的余数值。

使用 GOLOMB-RICE 的无损失帧内编码的设备和方法

发明背景

I. 发明领域

本发明涉及图像处理和压缩。具体地，本发明涉及在使用 Golomb-Rice 的频域中的 DC 系数的无损失编码。

II. 现有技术的描述

在数字信号处理的一般学科中，数字图像处理具有突出的位置。在数字图像处理的技术和科学中，人类视觉感知的重要性已经大大鼓动了利害关系和进步。在诸如那些用于放映电影胶片或电影的视频信号的发送和接收领域中，正在作出对于图像压缩技术的各种改进。许多当前的和推荐的视频系统都采用数字编码技术。这些领域的一些方面包括图像编码、图像再现、以及图像特征选择。图像编码表示试图按有效方式来发送数字通信信道的图像，尽量使用最少比特数以使所需要的带宽最小，而同时使失真保持在某个极限之内。图像再现表示再现目标的真实图像的尝试。在通信信道上发送的编码图像可能会因各种因素而造成失真。降质的根源最初来源于从目标创建图像。特征选择是指图像某些属性的选择。在较广的情况中，在识别、分类和判定中可能需要这些属性。

诸如在数字影院中的那些视频的数字编码是从改进的图像压缩技术得益的领域。通常可以把数字图像压缩分成两类：无损失方法和有损失方法。再现无损失图像而没有任何信息的损失。有损失方法包括一些信息的不可恢复的损失，取决于压缩比、压缩算法的质量以及算法的实施。通常，有损失压缩方法被看作为获得用于性能价格比合算的数字影院方法所要求的压缩比。为了达到数字影院质量水平，压缩方法应该提供视觉上无损失的性能水平。因此，虽然作为压缩过程的结果信息存在数学的损失，但是对于正常观看条件下的观看者，这种损失造成的图像失真应该是看不出的。

现有数字图像压缩技术已被研制成用于其它应用，即，用于电视系统。这种技术已经使设计综合考虑而适合于所需的应用，但是不符合影院显示必

需的质量要求。

数字影院压缩技术应该提供看电影的人以前感受过的视觉质量。理想地，数字影院的视觉质量应该试图超过高质量发行录印影片的视觉质量。同时，压缩技术应该具有可实现的高编码效率。如这里所定义，编码效率是指压缩的图像质量符合某个质量水平所要求的比特率。此外，系统和编码技术应该具有内部的灵活性以适合不同格式以及应该是性能价格比合算的；即，小尺寸和有效的解码器或编码器过程。

现有的许多压缩技术提供有效的压缩等级，但是导致视频信号的质量降低。一般，传送经压缩信息的技术要求按恒定的比特率传送经压缩的信息。

能够提供有效的压缩等级而同时保持视频信号所要求的质量水平的一种压缩技术，利用经编码的离散余弦变换（DCT）系数数据的具有自适应大小的块和子块。下面将把这种技术称为自适应块大小离散余弦变换（ABSDCT）方法。在题为“自适应块大小图像压缩方法和系统”的美国专利第 5,021,891 号中揭示了这种技术，该专利已转让给本发明的受让人，并在此引用作为参考。还在题为“自适应块大小图像压缩方法和系统”的美国专利第 5,107,345 号中揭示了 DCT 技术，该专利已转让给本发明的受让人，并在此引用作为参考。此外，在题为“自适应块大小图像压缩方法和系统”的美国专利第 5,452,104 号中讨论了结合微分四叉树变换技术（Differential Quadtree Transform Technique）的 ABSDCT 技术的应用，该专利已转让给本发明的受让人，并在此引用作为参考。在这些专利中揭示的系统利用被称为“帧内”编码的技术，其中对图像数据的每个帧进行编码而不管任何其它帧的内容。使用 ABSDCT 技术，可达到的数据速率可以从约每秒 15 亿比特减少到约每秒 5 千万比特而没有可辨别的图像质量降质。

可以使用 ABSDCT 技术来压缩黑白图像或彩色图像或表示图像的信号。彩色输入信号可以按 YIQ 格式，对于每个 4: 4: 4 的取样，Y 是亮度取样或辉度取样，而 I 和 Q 是色度或彩色取样，或者可以按其它格式。也可以使用诸如 YUV、YCbCr 或 RGB 格式之类的其它已知格式。因为眼睛对于彩色的较低空间敏感度，所以大多数研究已经表示在水平和垂直方向上通过四的因子对颜色分量的子取样是合理的。因此，可以通过四个亮度分量和两个色度分量来表示视频信号。

使用 ABSDCT，一般把视频信号分段成用于处理的像素块。对于每个块，

把亮度就色度分量传送到块交错器(interleaver)。例如,可以把 16×16 (像素)块提供给块交错器,块交错器在每个 16×16 块内对图像取样进行排序或组织以显示并组合用于离散余弦变换(DCT)分析的数据子块。DCT运算符是把时间和空间取样的信号转换成相同信号的频率表示的一种方法。通过转换成频率表示,已经表明DCT技术允许进行极高等级的压缩,因为可以设计出利用图像的频率分布特性的量化器。在较佳实施例,把一个 16×16 DCT应用于第一排序,四个 8×8 DCT应用于第二排序,16个 4×4 DCT应用于第三排序,以及64个 2×2 DCT应用于第四排序。

DCT运算减少了视频源中固有的空间冗余。在执行DCT之后,大多数视频信号能量趋向于集中在几个DCT系数中。可以使用附加的变换,微分二叉树变换(DQT),以减少DCT系数中间的冗余。

对于 16×16 块和每个子块,分析DCT系数值和DQT值(如果使用DQT的话),以确定对块或子块进行编码所需要的比特数。然后,选择需要最少比特数进行编码的块或子块的组合来表示图像分段。例如,可以选择两个 8×8 子块、6个 4×4 子块以及8个 2×2 子块来表示图像分段。

然后按次序恰当地排列所选择的块或子块的组合成为 16×16 块。然后,在发送的准备中,DCT/DQT系数值可以进行频率加权、量化和编码(诸如变量长度编码)。虽然上述ABSDCT技术执行得非常好,但是计算强度较大。

此外,虽然ABSDCT的使用在视觉上无损失,但是有时要求按对数据进行编码的确切方式来再现数据。例如,录像原版和档案库的场合需要如此压缩数据,以致能确切地在它本域(native domain)中再现。

发明概要

本发明是压缩数据的一种设备和方法,它允许按对数据进行编码的确切方式来再现数据。实施例包括基于质量的图像压缩的系统和方法,它利用离散余弦变换系数数据的自适应大小块和子块进行图像压缩。把像素数据的块输入到编码器。编码器包括块大小分配(BSA)单元,它对用于处理的输入像素块分段。块大小分配是基于输入块和进一步子分割的块的方差(variance)的。一般,把具有较大方差的区域子分割成较小的块,而对具有较小方差的区域不进行子分割,只要块和子块平均值落在不同的预定范围中。因此,首先修改块的方差阈值,根据它的平均值从它的额定值进行修改,然后对块的方差和阈值进行比较,如果方差大于阈值,则对块进行子分割。

把块大小分配提供给变换单元，该单元把像素数据变换成频域数据。只在通过块大小分配选择的块和子块上执行变换。对于 AC 单元，然后通过量化和串行化进行变换数据的定标。根据图像质量量度（诸如相对于对比度、系数计数、速率失真、块大小分配的密度和/或过去定标因子进行调节的定标因子）对变换数据的量化进行量化。诸如锯齿形扫描之类的串行化是基于创建相同值的最长可能行程（run length）的。然后在发送的准备中，通过可变长度编码器对数据流编码。编码可以是霍夫曼编码、或可以是基于指数分布的编码，诸如 Golomb-Rice 编码。

对于 DC 分量，对每片的第一 DC 分量值进行编码。把每片的每个后续 DC 分量值表示为余数值，是当前值和当前值之前的 DC 分量值之间的差。使用 Golomb-Rice 对每片的初始 DC 分量值和剩余值进行编码。

因此，实施例的一个方面是提供一种设备和方法，以有效地执行无损失压缩。

实施例的另一个方面是按有益于录像原版和档案库的场合的方式对数字图像和音频信息无损失地进行压缩。

实施例的另一个方面是对无损失地压缩的 DC 分量值执行一种编码方案。

实施例的另一个方面是产生具有低方差的余数值。

实施例的另一个方面是使用利用指数分布式数据的一种编码方案。

附图简述

从下面结合附图的详细描述中，对本发明的特性、目的和优点将更为明了，在所有的附图中，用相同的标记作相应的识别，其中：

图 1 是图像压缩和处理系统的编码器部分的方框图；

图 2 是图像压缩和处理系统的解码器部分的方框图；

图 3 是流程图，说明包括在基于方差的块大小分配中的处理步骤；

图 4a 说明 DCT 系数矩阵中零行程的 Y 分量的指数分布；

图 4b 说明 DCT 系数矩阵中零行程的 C_0 分量的指数分布；

图 4c 说明 DCT 系数矩阵中零行程的 C_1 分量的指数分布；

图 5a 说明 DCT 系数矩阵中幅度大小的 Y 分量的指数分布；

图 5b 说明 DCT 系数矩阵中幅度大小的 C_0 分量的指数分布；

图 5c 说明 DCT 系数矩阵中幅度大小的 C_1 分量的指数分布；

图 6 说明 Golomb-Rice 编码过程;

图 7 说明用于 Golomb-Rice 编码的设备; 以及

图 8 说明对 DC 分量值进行编码的过程。

较佳实施例的详述

为了便于数字信号的数字发送以及享受相应的益处, 通常必需采用某种形式的信号压缩。为了获得所产生图像的高压缩, 保持图像的高质量也是重要的。此外, 对于小型化的硬件实施, 对计算效率有所要求, 这在许多应用中是很重要的。

在详细说明本发明的一个实施例之前, 要理解, 本发明的应用并不限于下述说明中规定的或在附图中示出的部件的构成和安排的细则。本发明能够是其它实施例以及可按各种方法执行。还有, 要理解, 这里使用的短语和术语是为了说明的目的, 不得认为是限制。

在实施例的一个方面中采用的图像压缩是基于离散余弦变换 (DCT) 技术的, 诸如在 1999 年 11 月 8 日申请的待批美国专利申请第 09/436,085 号“基于对比度敏感方差的自适应块大小 DCT 图像压缩”中所揭示的那样, 该专利已转让给本发明的受让人, 并在此引用作为参考。在 2000 年 1 月 28 日申请的“基于质量的图像压缩”的待批美国专利申请第 09/494,192 号中描述了利用 DCT 的图像压缩和解除压缩系统, 该专利已转让给本发明的受让人, 并在此引用作为参考。通常, 在数字域中待处理的图像由分成大小为 $N \times N$ 的非重叠块的阵列的像素数据构成。可以在每个块上执行二维 DCT。通过下列关系定义二维 DCT:

$$X(k, l) = \frac{\alpha(k)\beta(l)}{\sqrt{N \cdot M}} \sum_{m=0}^{N-1} \sum_{n=0}^{M-1} x(m, n) \cos\left[\frac{(2m+1)\pi k}{2N}\right] \cos\left[\frac{(2n+1)\pi l}{2M}\right], \quad 0 \leq k, l \leq N-1$$

其中 $\alpha(k), \beta(l) = \begin{cases} 1, & \text{如果 } k=0 \\ 2, & \text{如果 } k \neq 0 \end{cases}$, 以及,

$x(m, n)$ 是在 $N \times M$ 块中位置 (m, n) 处的像素, 以及

$X(k, l)$ 是相应的 DCT 系数。

由于像素值是非负的, 所以 DCT 分量 $X(0,0)$ 始终是正的, 并且通常具有最多能量。事实上, 对典型的图像来说, 大部分变换能量集中在分量 $X(0,0)$ 的四周。就是这种能量压缩特性使 DCT 系统成为如此吸引人的一种

压缩方法的。

图像压缩技术采用对比度自适应编码 (contrast adaptive coding) 来获得进一步的比特率减少。已经观察到, 大多数自然图像是由相当慢变化的平坦区域和诸如目标边界和高对比度纹理之类的繁忙区域构成的。对比度自适应编码方案通过把更多比特分配给繁忙区域和把较少比特分配给较不繁忙区域来利用这种因素。

对比度自适应方法利用帧内编码 (空间处理) 来代替帧间编码 (时空处理)。帧间编码除了要求更复杂的处理电路之外还固有地需要多个帧缓冲器。在许多应用中, 对于实际实施需要降低复杂性。在能使时空编码方案失败和较差地执行的情况中, 帧内编码也是有用的。例如, 每秒 24 帧的电影可以属于这种类别, 由于由机械快门引起的集成时间 (integration time) 是相当短的。短集成时间允许较高程度的瞬时混淆。快速运动变成蠕动时, 帧到帧相关的假设破坏。当包括 50 Hz 和 60 Hz 两种电源线频率时, 帧内编码也较易标准化。当前电视按 50 Hz 或 60 Hz 发送信号。作为数字方法的帧内方案的使用可以适用于 50 Hz 和 60 Hz 两种操作, 或甚至通过使帧速率对空间分辨率折衷而适用于每秒 24 帧的电影。

为了图像处理的目的, 在分成非重叠块的阵列的像素数据上执行 DCT 运算。注意, 虽然这里讨论的块大小是 $N \times N$ 的大小, 但是可以想象, 可以使用各种块大小。例如, 可以利用 $N \times M$ 块大小, 其中 N 和 M 两者都是整数, M 大于或小于 N 。另一个重要方面是可把块分割成至少一级子块, 诸如 $N/i \times N/i$ 、 $N/i \times N/j$ 、 $N/i \times M/j$ 等等, 其中 i 和 j 是整数。此外, 这里讨论的示例块大小是具有相应的 DCT 系数的块和子块的 16×16 像素块。可以进一步想象, 可以使用各种其它整数, 诸如两者都是偶数或奇数的整数值, 例如, 9×9 。

图 1 和 2 说明包括可配置的串行化器的概念的图像处理系统 100。图像处理系统 100 包括压缩所接收视频信号的编码器 104。使用发送信道或物理媒体 108 发送经压缩的信号, 并由解码器 112 接收。解码器 112 对所接收的经编码的数据解码成图像取样, 然后可以显示这些取样。

一般, 把图像分割成用于处理的像素的块。可以使用 RGB 到 $Y C_1 C_2$ 转换器 116 把彩色信号从 RGB 空间转换到 $Y C_1 C_2$ 空间, 其中 Y 是亮度分量或辉度分量, 而 C_1 和 C_2 是色度或彩色分量。由于眼睛对于颜色的空间敏感性较

低，所以许多系统在水平和垂直方向上通过四的因子来对 C_1 和 C_2 进行子取样。然而，子取样不是必需的。已知为 4:4:4 格式的全分辨率图像在诸如那些被称为包括“数字影院”的某些应用中可能很有用或必需的。两种可能的 YC_1C_2 显示 (representation) 是 YIQ 显示和 YUV 显示，本技术领域众知这两种显示。还可以使用称为 $YCbCr$ 的 YUV 显示的变型。因此，在一个实施例中，使用显示 Y -even (Y -偶数)、 Y -odd (Y -奇数)、 Cb -even (Cb -偶数)、 Cb -odd (Cb -奇数)、 Cr -even (Cr -偶数)、 Cr -odd (Cr -奇数)。

在较佳实施例中，对每个偶数和奇数的 Y 、 Cb 和 Cr 分量处理而无需子取样。因此，把像素的 16×16 块的 6 个分量中的每一个的输入提供给编码器 104。为了举例说明的目的，以 Y -even 分量的编码器 104 为例进行说明。对于 Y -odd 分量和偶数和奇数的 Cb 和 Cr 分量，使用相似的编码器。编码器 104 包括块大小分配单元 120，它在视频压缩的准备中执行块大小分配。块大小分配单元 120 根据块中图像的感知特征确定 16×16 块的块分解。块大小分配单元 120 根据 16×16 块中的活动按二叉树方式把每个 16×16 块子分割成较小的块，诸如 8×8 、 4×4 和 2×2 。块大小分配单元 120 产生被称为 PQR 数据的二叉树数据，该数据的长度可以在 1 和 21 比特之间。因此，如果块大小分配确定要分割 16×16 块，则设置 PQR 数据的 R 比特，后面跟随的是对应于四个经分割的 8×8 块的 Q 数据的四个附加比特。如果块大小分配确定要对任何 8×8 块进行子分割，则对每个经子分割的 8×8 块添加 P 数据的四个附加比特。

现在参考图 3，图中提供示出块大小分配单元 120 的操作细节的流程图。使用块的方差作为判定子分割块的量度。在步骤 202 处开始，读出像素的 16×16 块。在步骤 204 处，计算 16×16 块的方差 v_{16} 。计算方差如下：

$$\text{var} = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x_{i,j}^2 - \left[\frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x_{i,j} \right]^2$$

其中 $N=16$ ，而 x_{ij} 是 $N \times N$ 块中第 i 行第 j 列中的像素。在步骤 206 处，如果块的平均值在两个预定值之间，则首先修改方差阈值 T_{16} 以提供新阈值 T'_{16} ，然后比较块方差和新阈值 T'_{16} 。

如果 v_{16} 不大于阈值 T_{16} ，则在步骤 208 处，把 16×16 块的开始地址写入暂存存储器，并把 PQR 数据的 R 比特设置成 0，以表示不对 16×16 块进行子分割。然后算法读出像素的下一个 16×16 块。如果 v_{16} 大于阈值 T_{16} ，

则在步骤 210 处, 把 PQR 数据的 R 比特设置成 1, 以表示要把 16×16 块子分割成四个 8×8 块。

顺序地考虑四个 8×8 块, 即, $i=1:4$, 进行进一步的子分割, 如在步骤 212 中所示。在步骤 214 处, 计算对于每个 8×8 块的方差 $v8_i$ 。在步骤 216 处, 如果块的平均值在两个预定值之间, 则首先修改方差阈值 T8 以提供新阈值 $T'8$, 然后比较块方差和这个新阈值。

如果方差 $v8_i$ 不大于阈值 T8, 则在步骤 218 处, 把 8×8 块的开始地址写入暂存存储器, 并把对应的 Q 比特, Q_i , 设置成 0。然后处理下一个 8×8 块。如果方差 $v8_i$ 大于阈值 T8, 则在步骤 220 处, 把对应的 Q 比特, Q_i , 设置成 1, 以表示要把 8×8 块子分割成四个 4×4 块。

顺序地考虑四个 4×4 块, 即, $j_i=1:4$, 进行进一步的子分割, 如在步骤 222 中所示。在步骤 224 处, 计算对于每个 4×4 块的方差 $v4_{ij}$ 。在步骤 226 处, 如果块的平均值在两个预定值之间, 则首先修改方差阈值 T4 以提供新阈值 $T'4$, 然后比较块方差和这个新阈值。

如果方差 $v4_{ij}$ 不大于阈值 T4, 则在步骤 228 处, 写入 4×4 块的地址, 并把对应的 P 比特, P_{ij} , 设置成 0。然后处理下一个 4×4 块。如果方差 $v4_{ij}$ 大于阈值 T4, 则在步骤 230 处, 把对应的 P 比特, P_{ij} , 设置成 1, 以表示要把 4×4 块子分割成四个 2×2 块。

阈值 T16、T8 和 T4 可以是预定的常数。这称之为硬判定。另一方面, 可以实施自适应或软判定。例如, 软判定根据 $2N \times 2N$ 块的平均象素值来改变方差的阈值, 其中 N 可以是 8、4 或 2。因此可以使用平均象素值的函数作为阈值。

为了举例说明的目的, 考虑下列例子。设 Y 分量的预定方差阈值对于 16×16 块、 8×8 块和 4×4 块分别为 50、1100 和 800。换言之, $T16=50$ 、 $T8=1100$ 以及 $T4=880$ 。设平均值的范围为 80 和 100。假定所计算的 16×16 块的方差是 60。由于 60 大于 T16, 而平均值 90 是在 80 和 100 之间, 所以把 16×16 块子分割成四个 8×8 子块。假定所计算的 8×8 块的方差是 1180、935、980 和 1210。由于两个 8×8 块具有超过 T8 的方差, 所以进一步子分割这两个块, 以产生总数为八个的 4×4 子块。最后, 假定八个 4×4 块的方差是 620、630、670、610、590、525、930 和 690, 对应的平均值为 90、120、110、115。由于第一个 4×4 块的平均值落在范围 (80, 100) 中, 所以它的阈值要降低

到小于 880 的 $T'4=200$ 。所以，将子分割这个 4×4 块以及第七个 4×4 块。

注意，使用相似的过程来分配亮度分量 Y-odd 以及彩色分量 $C_{b\text{-even}}$ 、 $C_{b\text{-odd}}$ 、 $C_{r\text{-even}}$ 、 $C_{r\text{-odd}}$ 的块大小。可以在水平方向上或垂直方向上，或此两方向上，均分（decimate）彩色分量。

另外，注意，虽然已经描述了块大小分配作为自上而下的方法，其中首先估计最大的块（在本例中是 16×16 ），但是也可以使用自下而上的方法来代替。自下而上的方法将首先估计最小的块（在本例中是 2×2 ）。

返回图 1，把 PQR 数据和所选择的块的地址一起提供给 DCT 单元 124。DCT 单元 124 使用 PQR 数据在所选择的块上执行适当大小的离散余弦变换。只有所选择的块才需要进行 DCT 处理。

图像处理系统 100 还包括 DQT 单元 128，用于减少 DCT 的 DC 系数中间的冗余。在每个 DCT 块的左上角处碰到 DC 系数。一般，DC 系数比 AC 系数大。尺寸大小的差异使设计有效的可变长度编码器有困难。因此，减小 DC 系数中间的冗余是有利的。

DQT 单元 128 在 DC 系数上执行两维 DCT，每次取 2×2 。从 4×4 块中的 2×2 块开始，在四个 DC 系数上执行两维 DCT。把这种 2×2 DCT 称为四个 DC 系数的微分二叉树变换，或 DQT。接着，DQT 的 DC 系数和在 8×8 块中的三个相邻 DC 系数一起使用来计算下一级 DQT。最后，使用 16×16 块中的四个 8×8 块的 DC 系数来计算 DQT。因此，在 16×16 块中，存在一个真实的 DC 系数，而其余的是对应于 DCT 和 DQT 的 AC 系数。

把变换系数（DCT 和 DQT 两者）提供给量化器进行量化。在一个较佳实施例中，使用频率加权屏蔽（FWM）和量化定标因子（quantization scale factor）对 DCT 系数量化。FWM 是维数与输入 DCT 系数的块相同的频率权重表格。频率权重把不同的权重施加于不同的 DCT 系数。权重的设计，突出具有对人类视觉或光学系统更敏感的频率内容的输入取样，而弱化具有对人类视觉或光学系统不很敏感的频率内容的输入取样。还可以根据诸如观看距离等因素来进行权重的设计。

根据经验数据来选择权重。在国际标准组织，1994 年的 ISO/IEC JTC1 CD 10918 “连续色调静止图像的数字压缩和编码—第一部分：要求和准则”中揭示了设计 8×8 DCT 系数的加权屏蔽的一种方法，这里引用该文作为参考。通常，设计两个 FWM，一个用于亮度分量，而一个用于色度分量。通过对于

8×8 块的均分 (decimation) 得到块大小 2×2、4×4 的 FWM 表格, 以及通过对于 8×8 块的内插得到块大小 16×16 的 FWM 表格。定标因子控制经量化的系数的质量和比特率。

因此, 量化每个 DCT 系数所根据的关系式为:

$$DCT_q(i,j) = \left\lfloor \frac{8 * DCT(i,j)}{fwm(i,j) * q} \pm \frac{1}{2} \right\rfloor$$

其中, DCT (i,j) 是输入 DCT 系数, fwm (i,j) 是频率加权屏蔽, q 是定标因子以及 DCT_q (i,j) 是经量化的系数。注意, 根据 DCT 系数的符号, 在括号内的第一项四舍五入。还使用合适的加权屏蔽量化 DQT 系数。然而, 可以使用多种表格或屏蔽, 并施加于 Y、Cb 和 Cr 中的每一个。

然后从 DC 值分离出 AC 值 (130), 并分别处理。对于 DC 元素, 对每片的第一 DC 分量值进行编码。然后把每片的每个后续 DC 分量值表示为后续 DC 分量值和它前面的 DC 分量值之间的差值, 并进行编码 (134)。对于无损编码, 使用 Golomb-Rice 对每片的初始 DC 分量值和差值进行编码 (138), 如相对于图 6 和 8 所描述。使用 Golomb-Rice 对连续的 DC 分量值之间的差值进行编码的有利之处在于该 DC 分量值的微分趋向于具有双面 (two-sided) 的指数分布。然后可以使用缓冲器暂时存储数据 (142)

, 然后通过发送信道 108 传送和发送到解码器 112。

图 8 说明 DC 分量值的编码过程。该过程可同等地应用于静止图像、视频图像 (诸如, 但是不限于, 运动图像或高分辨率电视) 以及音频。对于给定的数据片 (804), 检索片的第一 DC 分量值 (808)。然后对第一 DC 分量值编码 (812)。不象 AC 分量值, 不需要对 DC 分量值进行量化。在一个实施例中, 使用 16×16 块的单个 DC 值而不管块大小分配故障。设想可以使用任何固定大小的块, 诸如 8×8 或 4×4, 或通过块大小分配定义的任何可变块大小。然后检索给定片的第二或下一个 DC 分量值 (816)。然后对第二 DC 分量值和第一 DC 分量值进行比较, 并且对差值或余数进行编码。因此只需要把第二 DC 分量值表示为它与第一值之间的差值。对于片的每个 DC 分量值重复这个过程。因此, 对于是否到达片 (最后的块和因而为最后的 DC 值) 的终点作出询问 (824)。如果不是 (828), 则检索片的下一个 DC 值 (816), 并且重复所述过程。如果是 (832), 则检索下一片 (804), 重复所述过程直到处理了帧的所有片和文件的所有帧。

DC 分量值无损编码的目标是产生趋向具有较小方差的余数值。在使

用 DCT 中, DC 系数分量值对最大象素能量作出贡献。因此, 通过不对 DC 分量值量化而减小余数的方差。

然后, 对于 AC 元素, 通过量化器 146, 或定标因子单元, 对数据块和频率加权屏蔽进行定标。DCT 系数的量化使大量 DCT 系数都减小到导致压缩的零。在一个较佳实施例中, 有 32 个定标因子对应于平均比特率。与诸如 MPEG2 之类的压缩方法不同, 根据所处理的图像的质量来控制平均比特率, 来代替目标比特率和缓冲器状态。

为了进一步增加压缩, 把经量化的系数提供给扫描串行化器 150。串行化器 150 对经量化的系数的块进行扫描, 以产生经量化的系数的串行化流。可以使用锯齿形扫描、列扫描或行扫描。还可以选择许多不同的锯齿形扫描模式以及其它不同于锯齿形的模式。一种较佳技术使用锯齿形扫描的 8×8 块大小。经量化的系数的锯齿形扫描改进了碰到大量零值的运行 (a large run of zero values) 的可能性。这种零运行 (zero run) 固有地具有减小的概率, 可以使用霍夫曼码有效地编码。

把经串行化的、经量化的 AC 系数的流提供给可变长度编码器 154。可以使用霍夫曼编码或 Golomb-Rice 编码对 AC 分量值进行编码。对于 DC 分量值, 利用 Golomb-Rice 编码。行程编码器使零之间的系数从非零系数分开, 并且结合图 6 详细描述。在一个实施例中, 利用 Golomb-Rice 编码。在对具有指数分布的非负整数的编码中, Golomb-Rice 编码是有效的。对于为指数分布变量提供长度较短的码的压缩中, 使用 Golomb 码是更佳的。

在 Golomb 编码行程中, 通过非负整数 m 使 Golomb 码参数化。例如, 给定参数 m , 正整数 n 的 Golomb 编码, 用一元码的 n/m 的商后跟随着用修改的二进制码表示的余数 (如果余数小于 $2^{\lfloor \log_2 m \rfloor} - m$, 则它是 $\lfloor \log_2 m \rfloor$ 比特长, 否则, 它是 $\lfloor \log_2 m \rfloor$ 比特长) 来表示。Golomb-Rice 编码是把参数 m 表示为 $m=2^k$ 的 Golomb 编码的特殊情况。在这种情况下, 通过把整数 n 的二进制表示向右偏移 k 比特而得到 n/m 的商, 并用 n 的最低 k 比特来表示 n/m 的余数。因此, Golomb-Rice 码是 2 的并置 (concatenation)。可以使用 Golomb-Rice 码对具有由公式 (1) 给定的双面几何 (指数) 分布的正和负两种整数编码。

$$P_a(x) = c\alpha^{|x|} \quad (1)$$

在 (1) 中, α 是给出 x 的概率的衰减特征参数, 而 c 是归一化常数。由于 $P_a(x)$ 是单调的, 所以可以看到整数值的序列应该满足

$$P_{\alpha}(x_i - 0) \geq P_{\alpha}(x_i = -1) \geq P_{\alpha}(x_i = +1) \geq P_{\alpha}(x_i = -2) \geq \dots \quad (2)$$

如在图 4a、4b、4c 和 5a、5b、5c 中所示，在经量化的 DCT 系数矩阵中的零运行和幅度两者都具有指数分布。在这些附图中说明的分布是根据来自真实图像的数据的。图 4a 说明零行程对于相对频率的 Y 分量分布 400。相似地，图 4b 和 4c 分别说明零行程对于相对频率的 Cb 和 Cr 分量分布 410 和 420。图 5a 说明幅度大小对于相对频率的 Y 分量分布 500。相似地，图 5b 和 5c 分别说明幅度大小对于相对频率的 Cb 和 Cr 分量分布 510 和 520。注意，在图 5a、5b 和 5c 中，图表表示 DCT 系数的大小的分布。每个大小表示系数值的范围。例如，为 4 的大小值具有范围 $\{-15, -14, \dots, -8, 8, \dots, 14, 15\}$ ，总共 16 个值。相似地，为 10 的大小值具有范围 $\{-1023, -1022, \dots, -512, 512, \dots, 1022, 1023\}$ ，总共 1024 个值。可以从图 4a、4b、4c、5a、5b 和 5c 看到，行程和幅度大小两者都具有指数分布。幅度的实际分布可证明符合下列公式 (3)：

$$p(X_{k,l}) = \frac{\sqrt{2\lambda}}{2} \exp\{-\sqrt{2\lambda}|X_{k,l}|\}, k, l \neq 0 \quad (3)$$

在 (3) 中， $X_{k,l}$ 表示分别在垂直和水平方向上对应于频率 k 和 l 的 DCT 系数，而平均值 $\mu_x = \frac{1}{\sqrt{2\lambda}}$ ，方差 $\sigma^2_x = \frac{1}{2\lambda}$ 。因此，在处理 DCT 中的数据时，按所述的方式使用 Golomb-Rice 编码是更佳的。

虽然下面是相对于图像数据的压缩来描述的，但是可以把实施例等同地应用于压缩音频数据的实施例。在压缩图像数据中，例如，图像或视频信号可以是具有线性或对数编码的像素值的 RGB、YIQ 或 YUV 或 Y Cb Cr 分量。

图 6 说明对零和非零系数进行编码的过程(600)。当对 DCT 矩阵进行扫描时，分别处理零和非零系数并加以分离(604)。对于零数据，确定零运行的长度(608)。注意，行程是正整数。例如，如果发现行程是 n，则确定 Golomb 参数 m。在一个实施例中，确定 Golomb 参数作为行程的函数。在另一个实施例中，通过下面公式 (4) 确定 Golomb 参数 (m)

$$m = \lceil \log_2 n \rceil \quad (4)$$

任选地，通过计数器或寄存器来计算行程的长度和相关联的 Golomb 参数。为了对为零的行程 n 编码，对商进行编码(620)。在一个实施例中，商作为零的行程和 Golomb 参数的函数被确定。在另一个实施例中，通过下面公式 (5) 确定商 (Q)：

$$Q = \lfloor N/2^m \rfloor \quad (5)$$

在一个实施例中，以一元码对商 Q 进行编码，它需要 $Q+1$ 比特。接着，对余数进行编码 624。在一个实施例中，对余数进行编码作为行程和商的函数，在另一个实施例中，使用下面公式 (6) 确定余数 (R)：

$$R = n - 2^m Q \quad (6)$$

在一个实施例中以 m 比特二进制码对余数 R 进行编码。在确定商 Q 和余数 R 之后，并置 Q 和 R 的代码 (628) 以表示为零的行程 n 的总代码。

还使用 Golomb-Rice 对非零系数进行编码。由于系数幅度可以是正的或负的，所以需要使用符号位，并对所给出的幅度的绝对值进行编码。若非零系数的幅度是 x ，则可以把幅度表示为幅度的绝对值和符号的函数。因此，可以使用下面公式 (7) 把幅度表示为 y ：

$$y = \begin{cases} 2x, & \text{如果 } x \geq 0, \text{ 否则} \\ 2|x|-1 \end{cases} \quad (7)$$

因此，任选地通过计数器或寄存器 (632) 对非零系数的值进行计数。然后判定 (636) 幅度是否大于或等于零。如果是，则对该值编码为给定值的两倍 (640)。如果不是，则对该值编码为绝对值两倍少 1 (644)。设想还可以使用其它映射方案。关键是不需要区分值的符号的额外的位。

按公式 (7) 的表达对幅度编码导致 x 的正值成为偶整数而负值成为奇整数。此外，这个映射保留了 (2) 中的概率分配。公式 (7) 中示出的编码的优点允许避免使用符号位来表示正数和负数。在完成映射之后，按零行程所进行的方式对 y 编码。继续该过程直到已经对当前块中的所有系数进行了扫描。

重要的是要承认，虽然本发明的实施例确定系数和行程的值作为公式 (1) — (7) 的函数，但是不需要使用精确的公式 (1) — (7)。是 Golomb-Rice 编码和 DCT 系数的指数分布的利用允许了更有效的图像和音频数据的压缩。

由于在编码之后零运行不能与非零幅度区分，所以可能必需使用固定长度的特定前缀码来标出第一零运行的发生。在已经碰到非零幅度之后碰到在块中的全零是很普通的。在这种情况下，使用称之为块结束 (EOB) 码的代码而不是使用 Golomb-Rice 码可能会更有效。任选地，EOB 码也是特定的固定长度码。

根据公式 (1) 或 (3)，通过 α 或 λ 对在 DCT 系数矩阵中的幅度或行程的概率分布进行参数化。实质是，如果在发生特定 DCT 系数块的情况中，则编码效率可以得到提高。然后可以使用合适的 Golomb-Rice 参数对感兴趣的量进行编码。在一个实施例中，对于每个行程和幅度大小值使用计数器或寄存器以计算各个累积的值以及发生这种值的相应次数。例如，如果存储累积值的寄存器和所累积的元素数量分别是 R_{rl} 和 N_{rl} ，则可以使用下面公式 (6) 作为 Golomb-Rice 参数对行程进行编码：

$$\left\lceil \log_2 \frac{R_{rl}}{N_{rl}} \right\rceil \quad (6)$$

对于幅度可使用相似的过程。

回到图 1，可以使用缓冲器 142 暂存编码器 104 产生的经压缩的图像信号，然后使用发送信道 108 发送给解码器 112。发送信道 108 可以是诸如磁性或光学存储装置之类的物理媒体，或有线或无线传输过程或设备。把包括块大小分配信息的 PQR 数据也提供给解码器 112（图 2）。解码器 112 包括缓冲器 164 和对行程值和非零值进行解码的可变长度解码器 168。可变长度解码器 168 的操作与图 6 描述的方式相似但相反。

把可变长度解码器 168 的输出提供给反串行化器 172，它根据所使用的扫描方案对系数进行排序。例如，如果使用锯齿形扫描、垂直扫描和水平扫描的混合，则反串行化器 172 用所使用的扫描类型的知识，会恰当地对系数进行再排序。反串行化器 172 接收 PQR 数据，有助于把系数恰当地排序成组合系数块。

把组合块提供给反量化器 174，使使用量化器定标因子和频率加权屏蔽的处理恢复原状（undoing）。

如果已经应用微分二叉树变换，则然后就把系数块提供给 IDQT 单元 186，接着提供给 IDCT 单元 190。否则，把系数块直接提供给 IDCT 单元 190。IDQT 单元 186 和 IDCT 单元 190 对系数进行反变换以产生像素数据的块。然后可以内插像素数据，转换成 RGB 形式，然后进行存储而供将来显示。

图 7 说明用于 Golomb-Rice 编码的设备 700。在图 7 中的设备最好实施相对于图 6 描述的过程。判定器 704 判定行程 (n) 和 Golomb 参数 (m)。任选地，对于每个行程和幅度大小值使用计数器或寄存器 708，以计算各个累积值和发生这种值的相应次数。编码器 712 对商 (Q) 进行编码作为行程

和 Golomb 参数的函数。编码器 712 还对余数 (R) 进行编码作为行程、Golomb 参数和商的函数。在另外的实施例中，编码器 712 还对非零数据编码作为非零数据值和非零数据值的符号的函数。使用并置器 716 来并置 Q 值和 R 值。

作为例子，可以在硬件或软件中用专用集成电路 (ASIC)、可编程逻辑器件、分立门或晶体管逻辑、例如，诸如寄存器和 FIFO 之类的分立硬件部件、执行一组固件指令的处理器、任何传统的可编程软件和处理器、或任何它们的组合来实施或执行结合这里揭示的实施例描述的各种例示性的逻辑块、流程图和步骤。处理器可以有利地是微处理器、但是另一方面，处理器可以是任何传统处理器、控制器或状态机。软件可以驻留在 RAM 存储器、快闪存储器、ROM 存储器、寄存器、硬盘、可拆卸盘、CD-ROM、DVD-ROM 或本技术领域众知的任何其它形式的存储媒体中。

提供较佳实施例的上述描述，以使熟悉本领域技术的人员可以制造或使用本发明。这些实施例的各种修改对熟悉本领域技术的人员将是显而易见的，可以把这里所定义的一般原理应用到其它的实施例而不需要用发明创造。因此，不打算把本发明限于这里所示出的实施例，而是和这里所揭示的原理和新颖特征相符的最宽广的范围相一致的。

在下面的权利要求书中规定本发明的其它特征和优点。

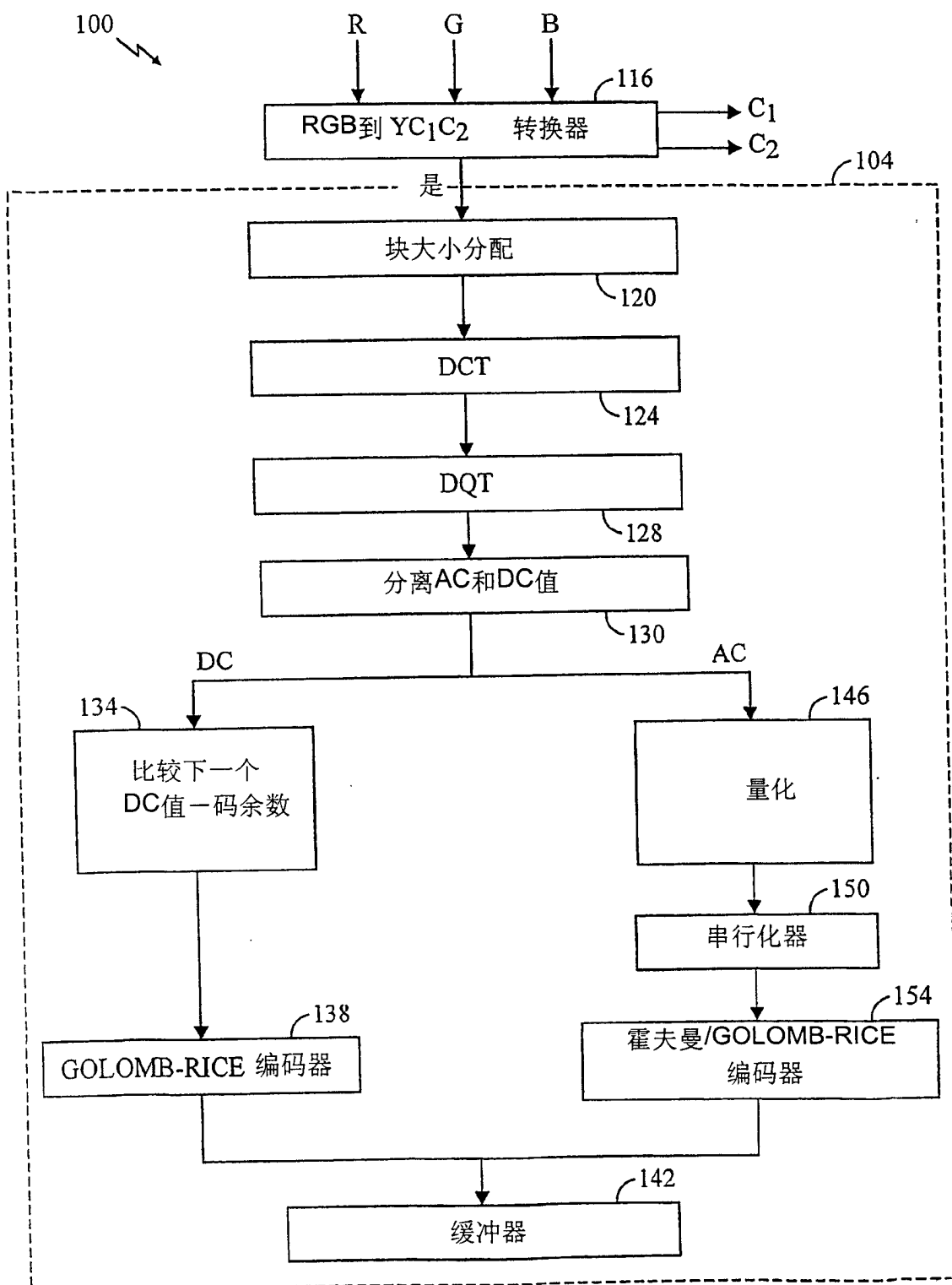


图 1

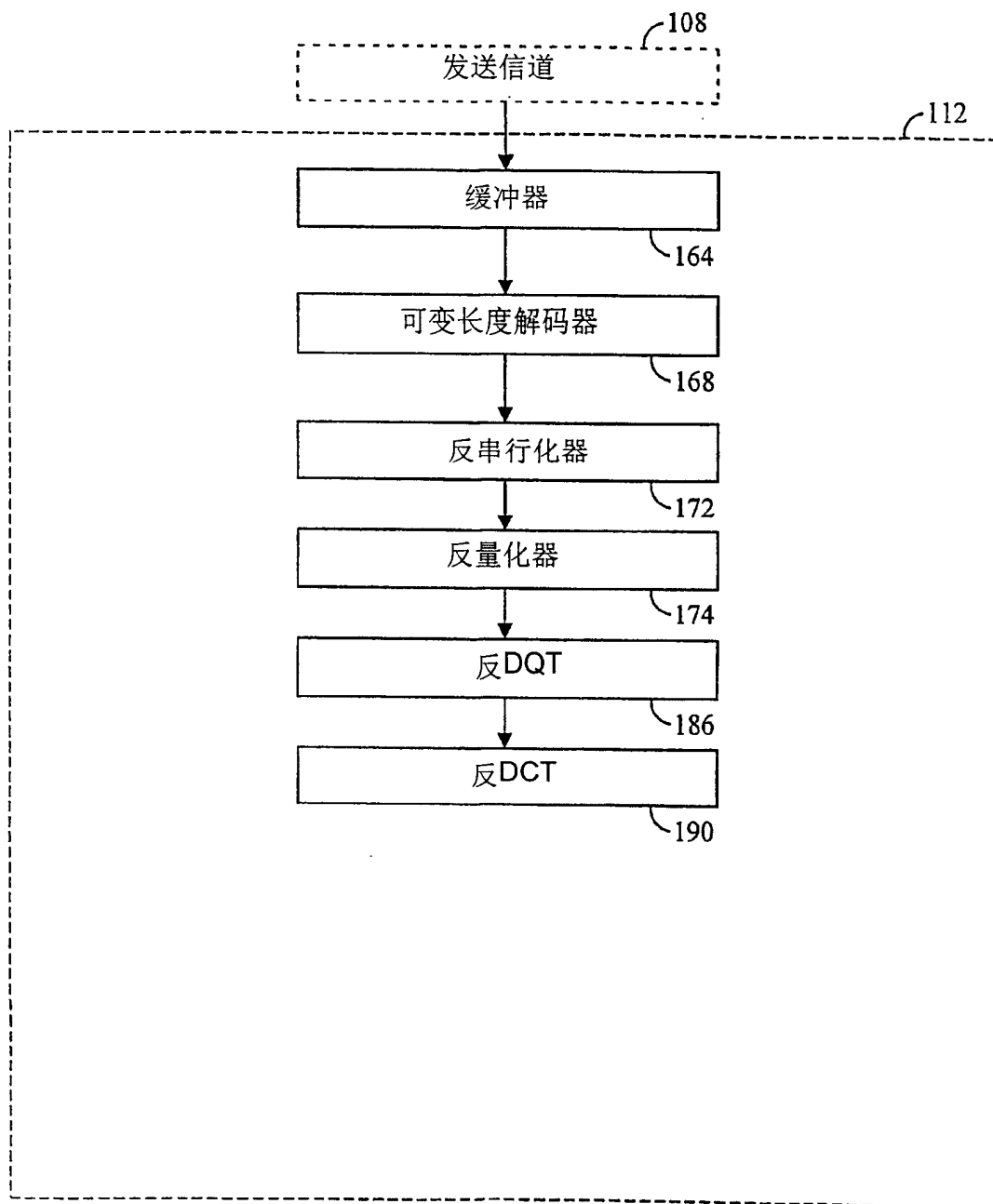


图 2

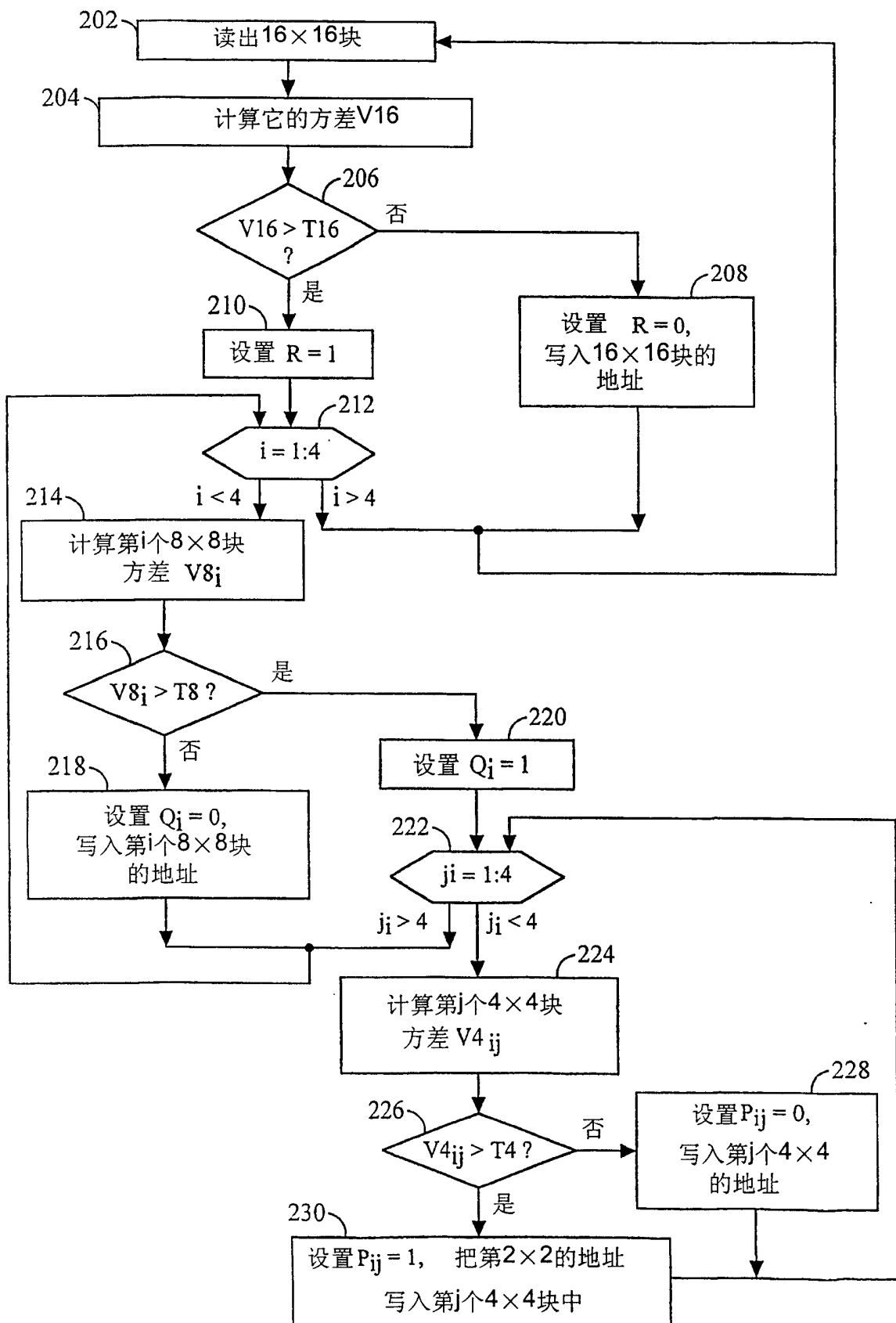


图 3

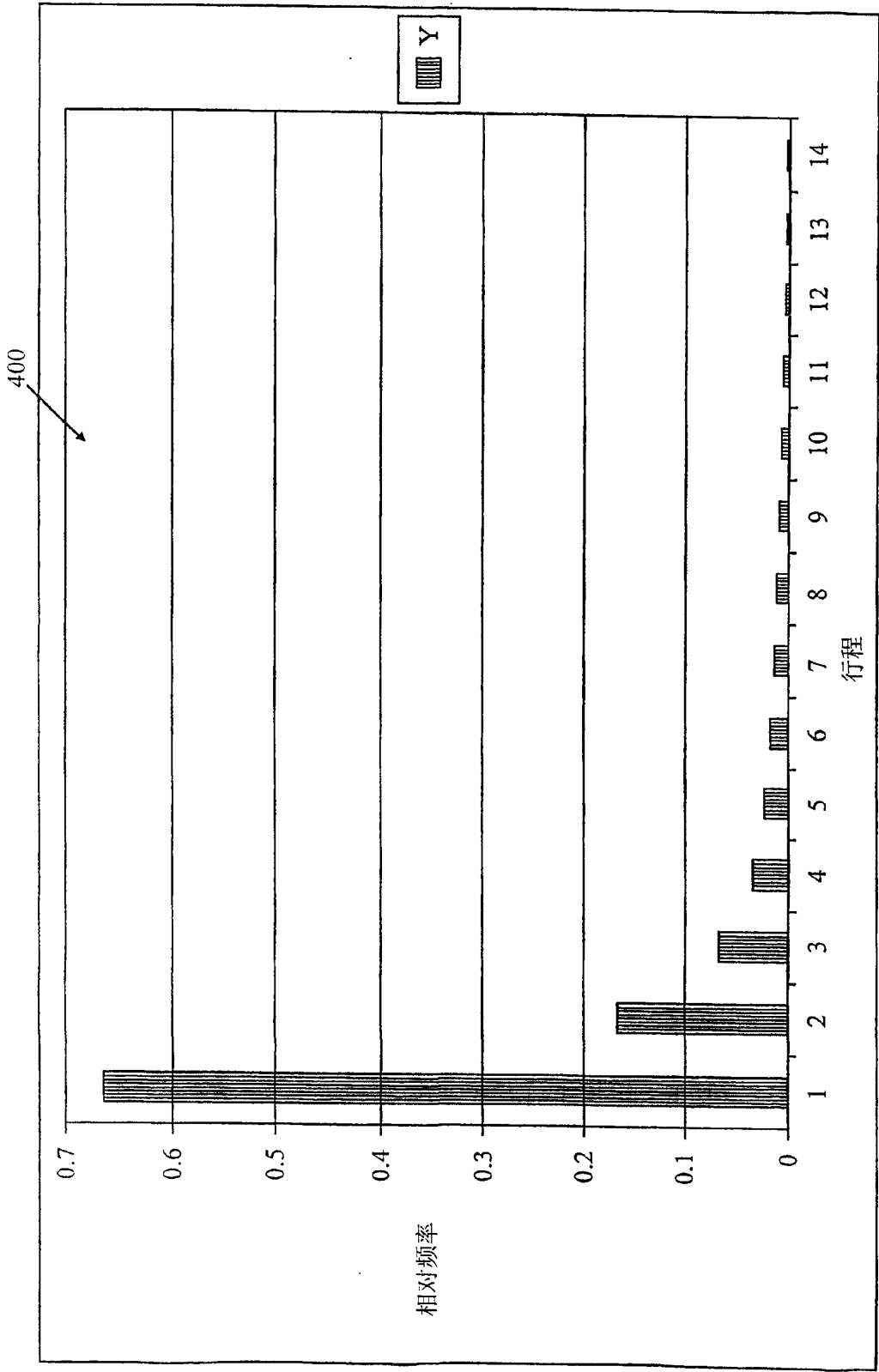


图 4A

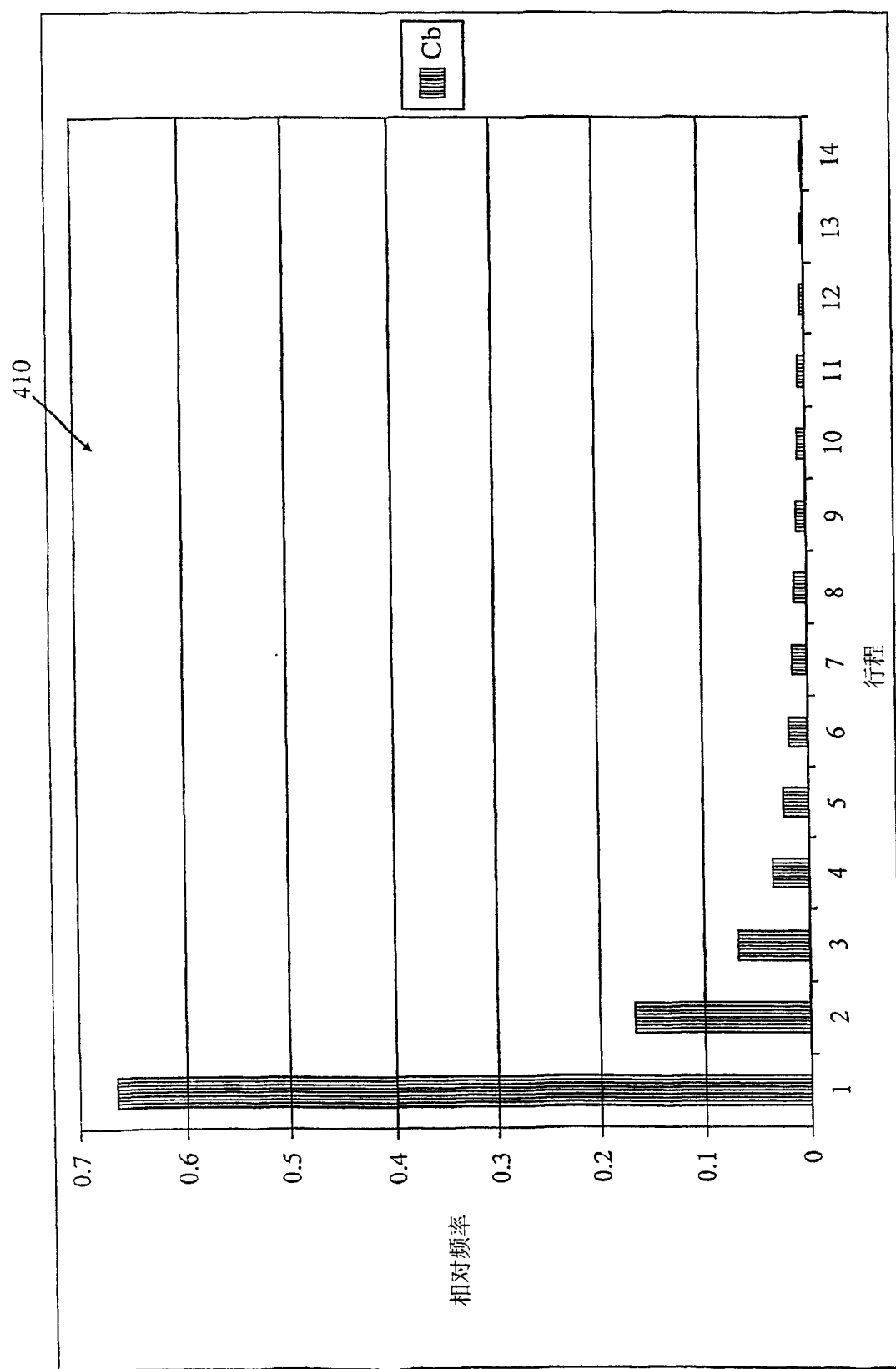


图 4B

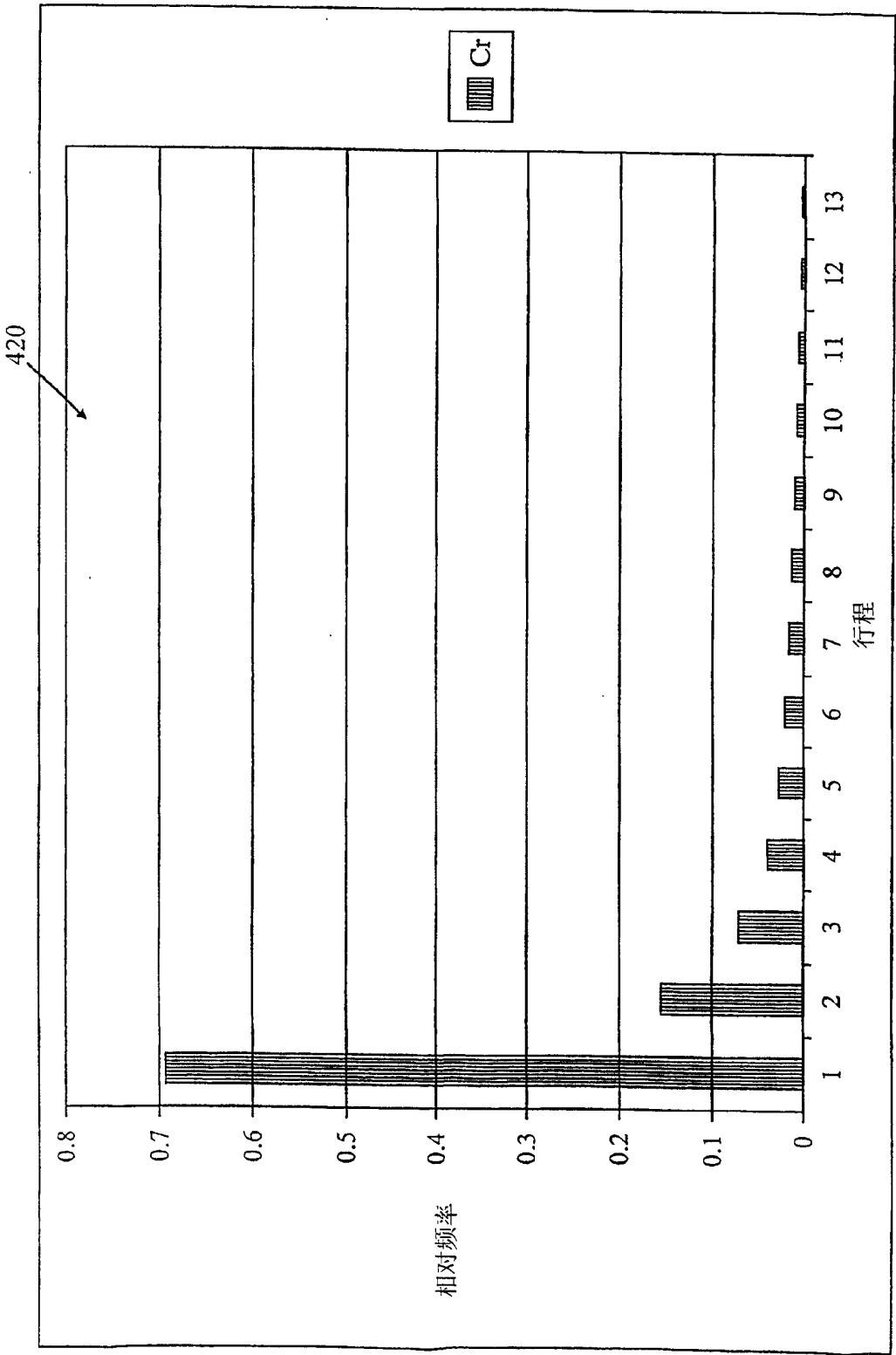


图 4C

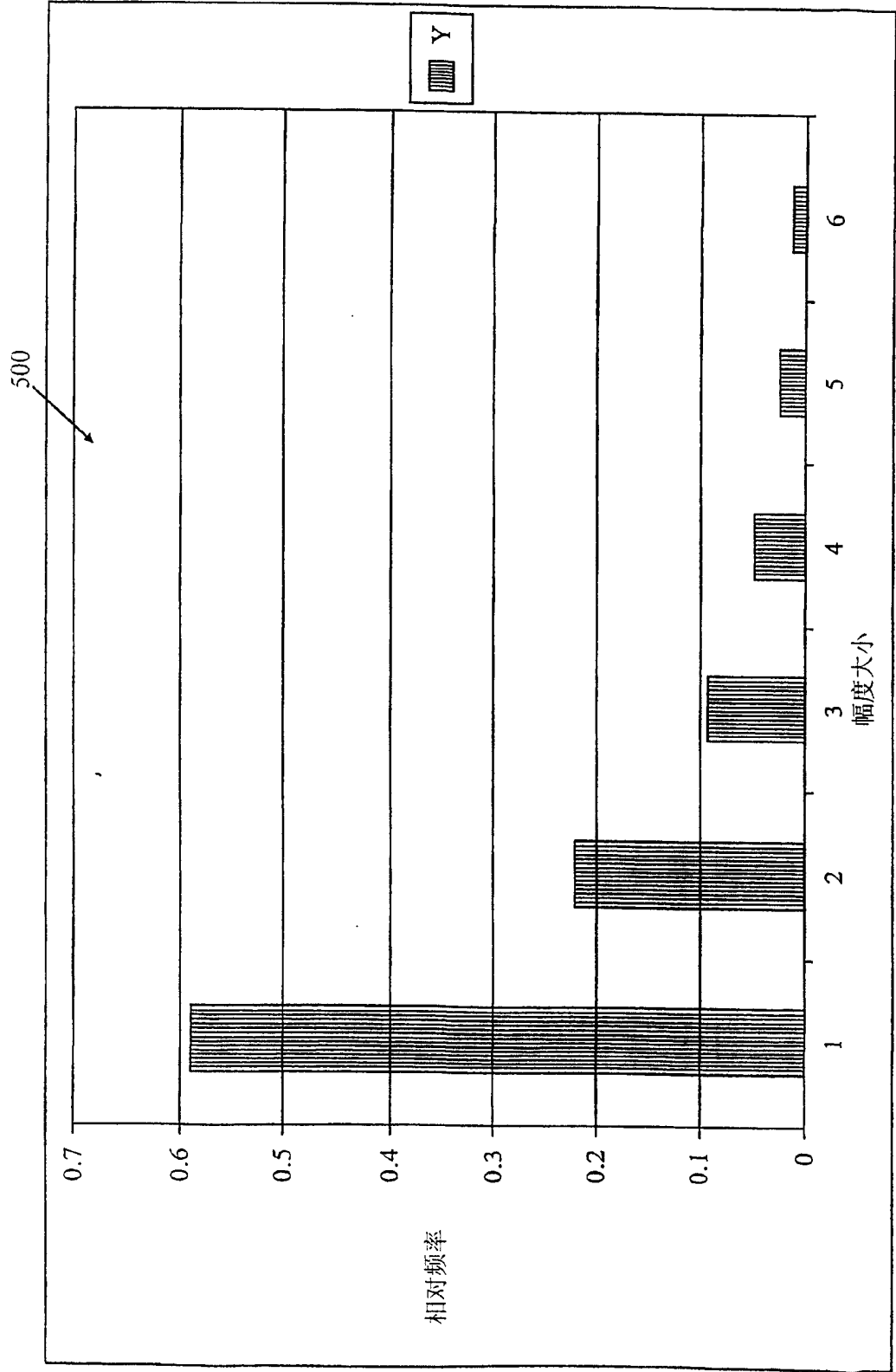


图 5A

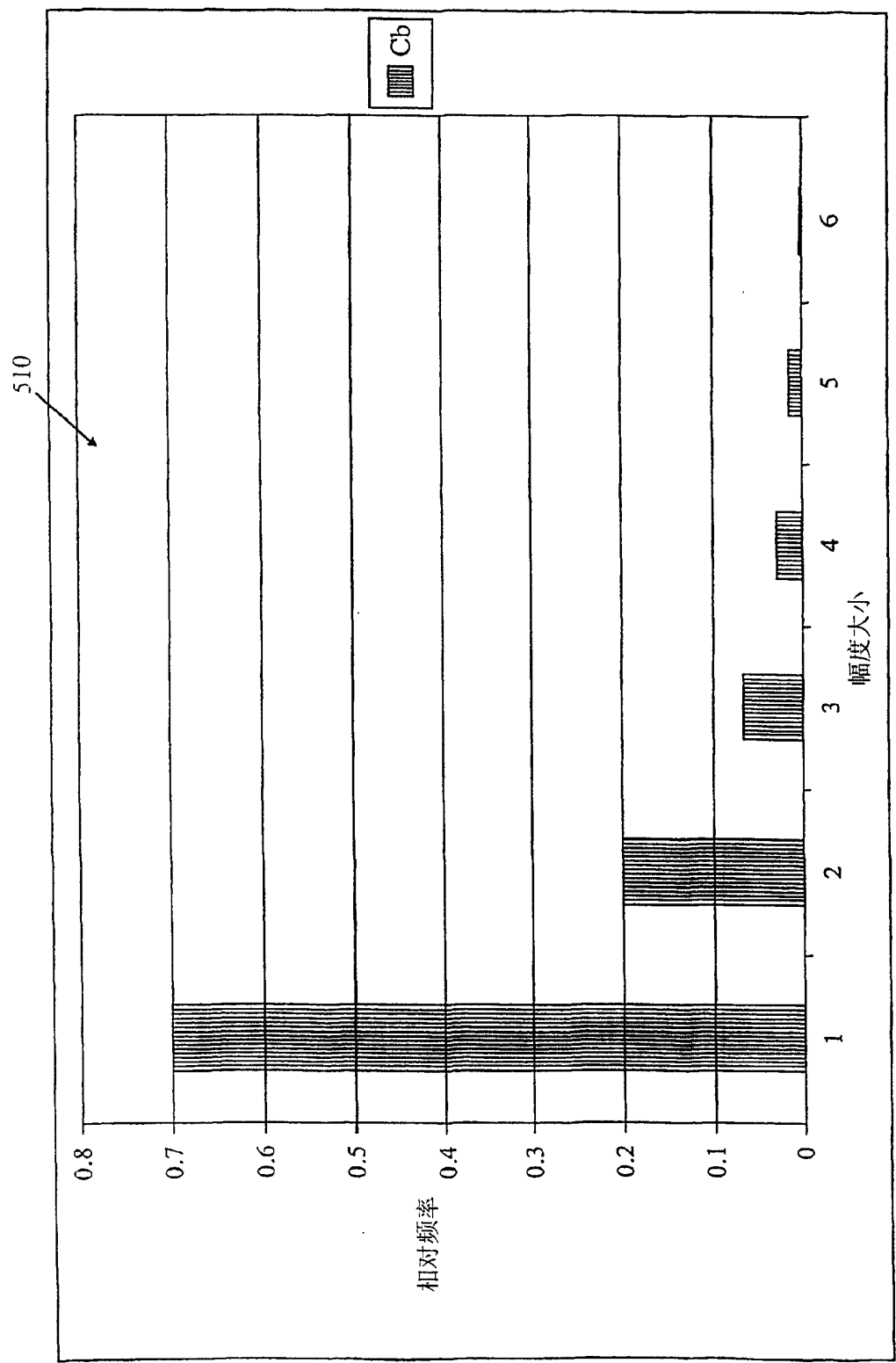


图 5B

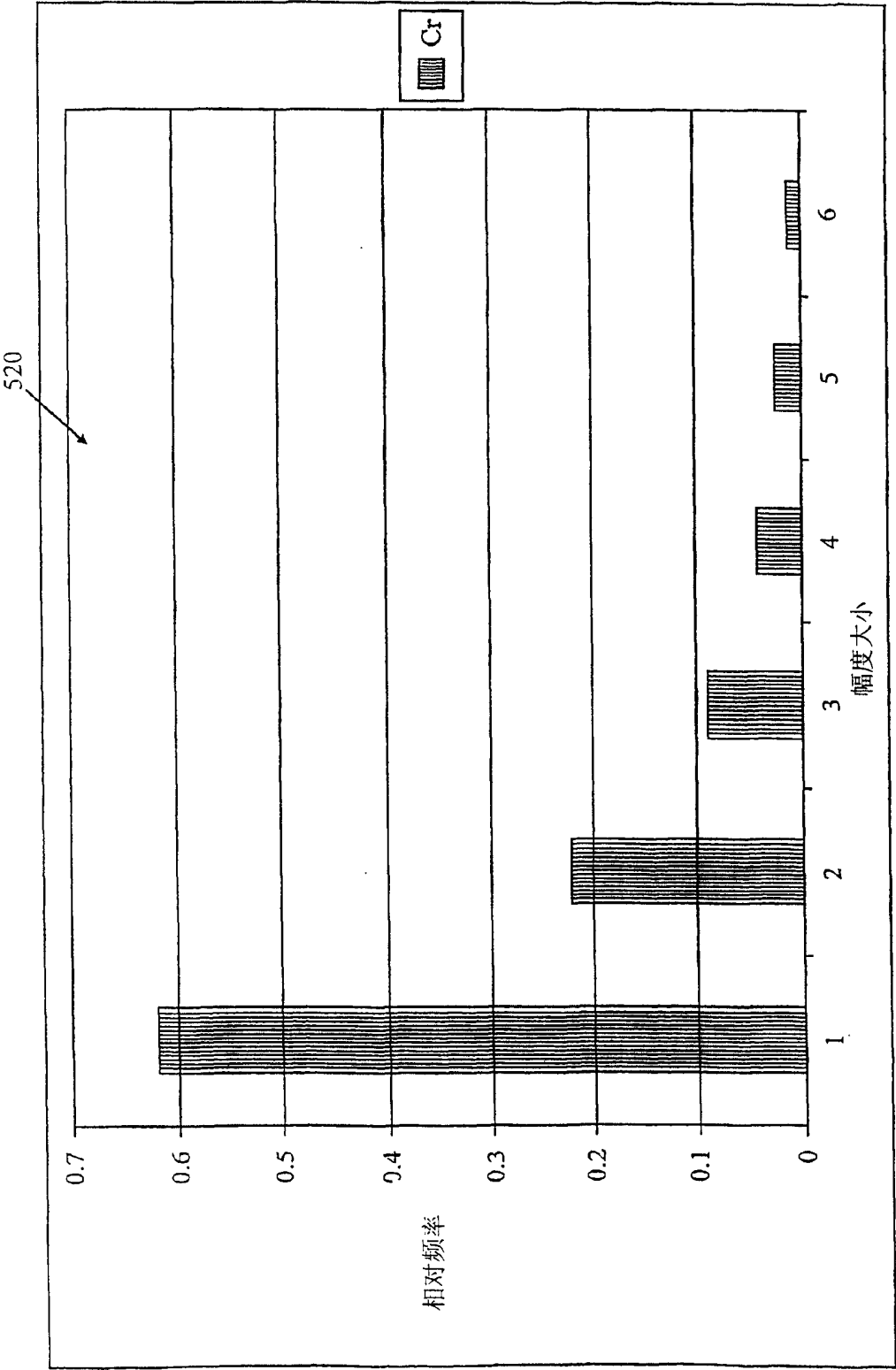


图 5C

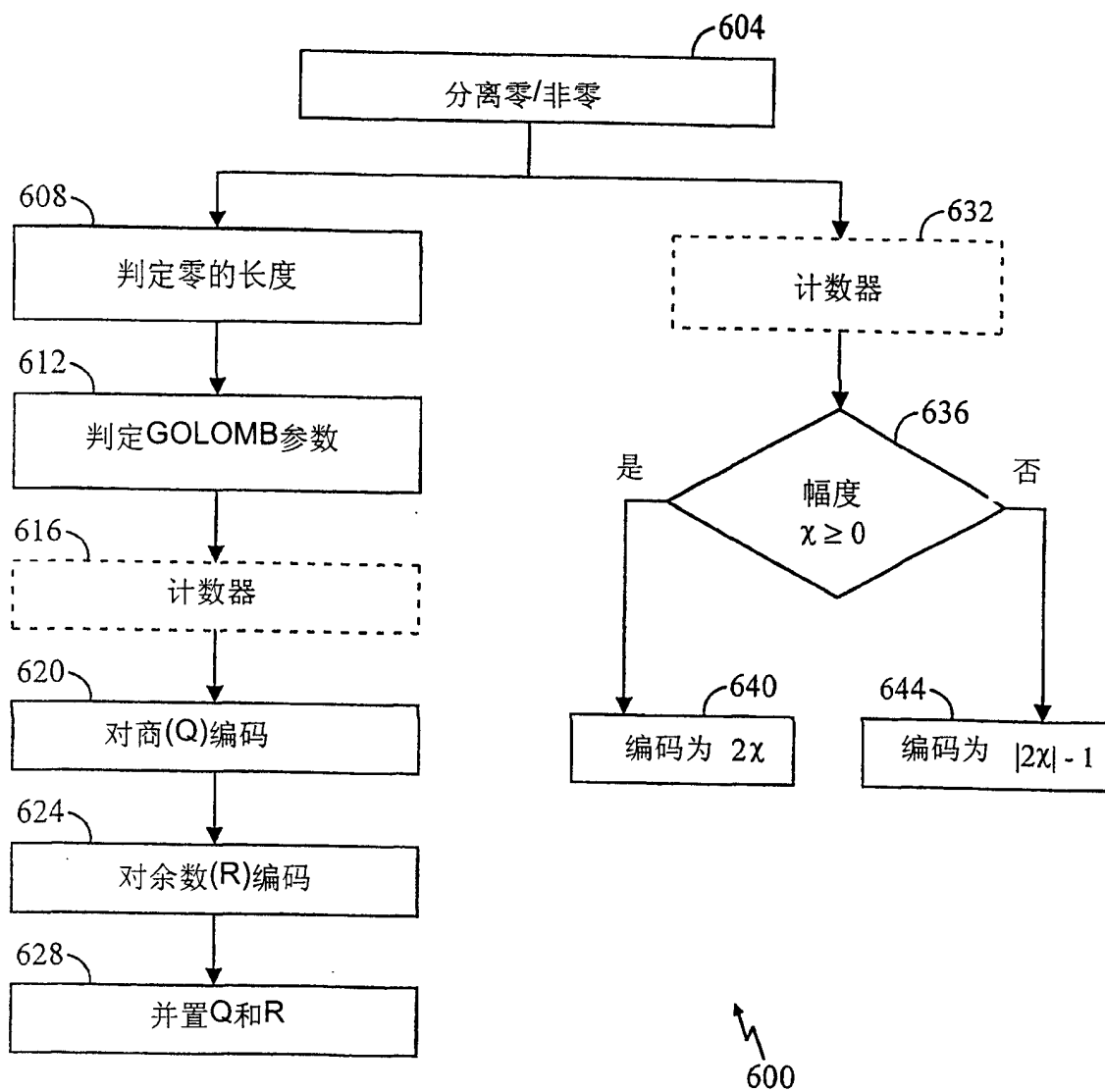


图 6

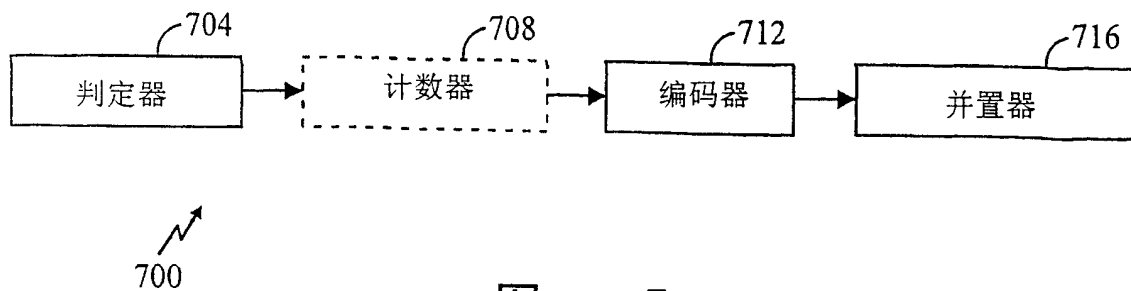
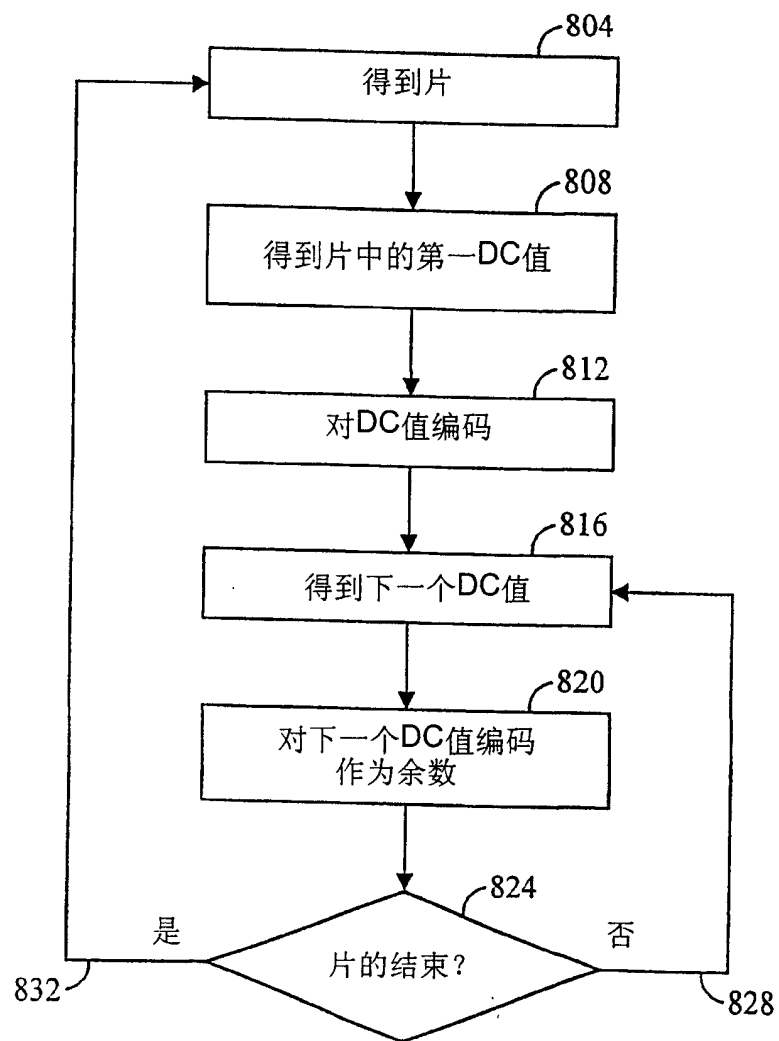


图 7



800

图

8