[19] 中华人民共和国国家知识产权局

[51] Int. Cl[7]
H04N 7/26
H04H 9/00

[12] 发明专利申请公开说明书

[21] 申请号 02801152.X

[43] 公开日 2003 年 12 月 10 日

[11] 公开号 CN 1461565A

[22] 申请日 2002.2.7 [21] 申请号 02801152.X
[30] 优先权
[32] 2001. 2.12 [33] EP [31] 01200505.4
[32] 2001. 7.17 [33] EP [31] 01202720.7
[86] 国际申请 PCT/IB02/00379 2002.2.7
[87] 国际公布 WO02/065782 英 2002.8.22
[85] 进入国家阶段日期 2002.12.9
[71] 申请人 皇家菲利浦电子有限公司
地址 荷兰艾恩德霍芬
[72] 发明人 J·A·海特斯马
A·A·C·M·卡克
C·P·M·J·巴根
J·C·奥斯特维恩

[74] 专利代理机构 中国专利代理(香港)有限公司
代理人 杨 凯 罗 朋

权利要求书 4 页 说明书 14 页 附图 6 页

[54] 发明名称 生成和匹配多媒体内容的散列

[57] 摘要

散列是可用于标识文件的数据文件的简短摘要或签名。 对多媒体内容(音频、视频、图像)运用散列法较为困难，因为原始内容的散列和处理后(例如压缩后)的内容的散列可能显著不同。 本公开的方法为多媒体内容、例如音频剪辑生成健壮散列。 音频剪辑被分成(12)连续(最好交叠)的各帧。 对于每一帧，将频谱分成(15)频带。 计算(16)每个频带的健壮性质(例如能量)并通过各个散列比特表示(17)。 由此，音频剪辑就由一串二进制散列字(每帧对应一个)来表示。 为识别可能压缩的音频信号，计算机(20)将从其中推导出的散列字块与大型数据库(21)比较。 这些匹配策略也已公开。 在最佳实施例中，提取处理还提供关于散列比特中哪些最不可靠的信息(19)。 将这些比特翻转会显著地提高匹配处理的速度和性能。

ISSN1008-4274

1. 一种生成标识信息信号的散列信号的方法，所述方法包括如下步骤：

将所述信息信号分成各帧；

为每一帧计算散列字；以及

将相继的散列字连接而构成所述散列信号。

2. 如权利要求1所述的方法，其特征在于，所述计算步骤包括如下步骤：

将所述信息信号的每一帧分成不相交的带或块；

计算所述每个带或块中的信号的性质；

将所述带或块中的性质与各个阈值进行比较；

由散列字的各个比特表示所述比较的结果。

3. 如权利要求2所述的方法，其特征在于，相邻带或块的性质构成所述阈值。

4. 如权利要求2所述的方法，其特征在于，前一帧中相应的带或块的性质构成所述阈值。

5. 如权利要求2所述的方法，其特征在于，所述带或块是所述信息信号的各个帧的频谱的频带。

6. 如权利要求5所述的方法，其特征在于，所述频带具有作为频率的函数的递增带宽。

7. 如权利要求5所述的方法，其特征在于，所述性质是频带的能量。

8. 如权利要求5所述的方法，其特征在于，所述性质是频带的音调。

9. 如权利要求1所述的方法，其特征在于，所述信息信号被分成交叠的帧。

10. 如权利要求2所述的方法，其特征在于，所述信息信号是视

频信号，该信号的各帧被分成块，块的平均亮度构成所述块的性质。

11. 如权利要求 2 所述的方法，其特征在于还包括采用所述比较步骤的输入来生成指示所述散列字的比特的可靠性的信息的步骤。

12. 一种生成标识信息信号的散列信号的方法，其特征在于包括如下步骤:

将所述信息信号分成块;

对于每个块，提取所述块内的信息信号的特征;

将所提取的特征的值与阈值进行比较;

为每个块生成指示所述提取的特征的值是大于还是小于所述阈值的散列比特;

为每个块确定指示所述提取的特征的值是否与所述阈值相差很远的可靠性信息;

将这些块的所述散列比特与所述可靠性信息组合成具有如下比特的散列值: 可靠散列比特，对于这些比特，所提取的特征与所述阈值相差很大; 以及不可靠比特，对于这些比特，所提取的特征与所述阈值没有实质上的差别。

13. 一种根据如权利要求 1 至 12 中任一个所述的方法、生成标识信息信号的散列信号的装置。

14. 一种将表示信息信号的至少一部分的散列字输入块与存储在数据库中标识各个信息信号的散列信号进行匹配的方法，所述方法包括如下步骤:

(a)选择所述散列字输入块中的一个散列字;

(b)在所述数据库中搜索所述散列字;

(c)计算所述散列字输入块与存储的散列字块之间的差，其中在步骤(b)找到的散列字与输入块中所选的散列字的位置相同;

(d)对进一步选择的散列字重复步骤(a)至(c)，直到所述差低于预定的阈值为止。

15. 如权利要求 14 所述的方法，其特征在于，所述进一步选择的

散列字是所述散列字输入块中的另一个散列字。

16. 如权利要求 14 所述的方法，其特征在于，所述进一步选择的散列字是通过使先前选择的散列字的比特相反来获得的。

17. 如权利要求 16 所述的方法，其特征在于还包括如下步骤：接收表示所选的散列字的比特的可靠性的信息，并利用所述信息确定要使之相反的比特。

18. 一种将表示未识别的信息信号的散列值与存储在数据库中且标识相应的多个信息信号的多个散列值进行匹配的方法，该方法包括如下步骤：

(a)以多个可靠散列比特和不可靠散列比特的形式接收所述散列值；

(b)在所述数据库中搜索所存储的散列值，对于这些散列值，所运用的散列值的可靠比特与所存储的散列值的相应比特匹配；

(c)对于步骤(b)中找到的每个存储的散列值，计算表示所述未识别的信息信号的散列值的可靠比特与所述存储的散列值的相应比特之间的误码率；以及

(d)确定对于哪个存储的散列值、该误码率最小且足够小。

19. 一种将表示未识别的信息信号的散列信号与存储在数据库中且标识相应的多个信息信号的多个散列信号进行匹配的方法，该方法包括如下步骤：

(a)以一系列散列值的形式接收所述散列信号，各个散列值具有可靠散列比特和不可靠散列比特；

(b)将所述系列的散列值之一运用于所述数据库；

(c)在所述数据库中搜索满足如下条件的所存储的散列值：所运用的散列值的可靠比特与所存储的散列值的相应比特匹配；

(d)对于步骤(c)中找到的每个存储的散列值：

在所述数据库中选择存储的散列值的相应系列；

计算表示所述未识别的信息信号的所述一系列散列值中的可靠

比特与数据库中选择的散列值系列的相应比特之间的误码率；以及

(f)确定对于哪一个存储的散列值系列、该误码率最小且足够小。

20. 如权利要求 19 所述的方法，其特征在于还包括以下步骤：对所述未识别的信息信号的其它散列值重复步骤(b)-(f)，直到找到误码率最小且足够小的存储的散列值系列。

21. 一种装置，它根据如权利要求 14 至 20 中任一个所述的方法，将表示信息信号的至少一部分的散列字输入块与存储在数据库中标识各个信息信号的散列信号进行匹配。

22. 一种将信息信号的接收者重定向至因特网站点的方法，该方法包括如下步骤：从所述信息信号推导出散列信号，并将所述散列信号与存储在数据库中标识因特网站点的散列信号进行匹配。

23. 一种测量信息信号的质量的方法，该方法包括如下步骤：从所述信息信号推导出散列信号，将所述散列信号与存储在数据库中的标识所述信息信号的散列信号进行匹配，然后计算推导的散列信号与存储的散列信号之间的差。

24. 一种识别多媒体信号的方法，该方法包括如下步骤：接收和/或记录所述多媒体信号的至少一部分，从所述多媒体信号推导散列信号，将所述散列信号发送到数据库，用以将其与存储在所述数据库中的散列信号进行匹配，以及从所述数据库接收所述多媒体信号的标识符。

25. 如权利要求 24 所述的方法，其特征在于，所述接收和/或记录所述多媒体信号、推导和发送所述散列信号以及接收所述标识符的各步骤是由移动电话装置来执行的。

生成和匹配多媒体内容的散列

发明领域

本发明涉及用于生成标识信息信号的散列信号的方法和装置。本发明还涉及用于使这种散列信号与存储在数据库中的散列信号匹配的方法和装置。

发明背景

散列函数在密码技术领域中是广为人知的，其中它们被特别用于标识大量的数据。例如，为了验证大文件的正确接收，发送该文件的散列值(也称为签名)就足够了。如果返回的散列值与原始文件的散列值匹配，则几乎完全可以确定该文件已被接收方正确接收。余留的不确定性是由可能出现冲突的情况引入的：即，两个不同的文件可能具有相同的散列值。精心设计的散列函数将冲突的概率降至最低。

密码散列的一个特性是它的极端脆弱性。翻转源数据中单个比特通常就会导致完全不同的散列值。这使得密码散列法不适用于标识多媒体内容，多媒体内容中相同内容的不同质量的版本应产生相同的签名。在某种程度上对数据处理(只要处理保持内容可接受的质量)保持不变的多媒体内容的签名称为健壮签名，或者作为我们最佳命名约定，称为健壮散列。利用健壮散列和内容标识符的数据库，可以识别未知内容，即使它已被劣化(例如被压缩或经过 AD/DA 转换)。健壮散列捕获视听内容感觉上的基本部分。

对于就同一目的采用水印技术，采用健壮散列标识多媒体内容是一个可选方案。但是其间也有很大差别。加水印需要在释放之前对原始内容起作用(即水印嵌入)，存在对内容质量的潜在影响和逻辑问题，而健壮散列则不需要在释放之前起作用。散列技术的缺点是

需要访问数据库(即，散列法仅在连接的上下文中才可行)，而水印检测器可以在本地操作(例如未连接的 DVD 播放器)。

美国专利 4677466 公开了一种从电视信号中导出签名的已知方法，用于广播监控。在此现有技术的方法中，在指定事件、如空白帧出现之后从短的视频或音频序列中推导出签名。

发明目的和概述

本发明总的目的是提供一种健壮散列技术。更具体地说，本发明的第一目的是提供一种用于从多媒体内容中提取有限数量的散列比特的方法和装置。散列比特是健壮的，但不意味着误码的概率为零。众所周知，非精确模式匹配(即在数据库中搜索最相似的散列值)是 NP-完全的。对于外行来说，这意味着最佳搜索策略是穷举搜索，这在涉及大型数据库的许多应用中是行不通的。因此，本发明的第二目的是提供一种克服这种 NP-完全搜索的复杂性的方法和装置。

第一目的是通过如下步骤实现的：将信息信号划分成连续的(最好是交叠的)帧，为每帧计算散列字，并把相继的散列字连接而构成散列信号(或简称散列)。散列字是通过如下方式计算的：对信息信号的标量性质或性质矢量、如不相交频带的能量或图像块的平均亮度进行阈值处理。

第二目的是通过如下步骤实现的：选择散列字输入块中的单个散列字，在数据库中搜索所述散列字，计算散列字输入块与对应的存储的散列字块之间的差值。对于进一步选择的散列字重复这些步骤，直到所述差值低于预定阈值。

本发明的另一些特征在从属权利要求中定义。

附图简介

图 1 是根据本发明从音频信号中提取散列信号的装置的实施例的示意图。

图 2 是说明将音频信号频谱再划分成按对数间隔的带的示意图。

图 3 是说明从音频剪辑中提取的散列字的示意图。

图 4 是根据本发明从视频信号中提取散列信号的装置的实施例的示意图。

图 5 是说明从视频序列中提取的散列字的示意图。

图 6 是根据本发明、由图 1 所示计算机完成的操作的流程图。

图 7 是说明图 1 所示的计算机操作的示意图。

图 8 表示图 3 所示的构成提取散列块的散列字中的误码的数量的图表。

图 9 表示图 3 所示的散列块的散列字的最可靠比特的图表。

图 10 是根据本发明的另一个实施例由图 1 所示计算机完成的操作的流程图。

实施例的描述

在描述最佳实施例之前，先阐述本发明所基于的考虑的一般说明。

从信号理论的意义上来说，两个信号(音频、视频、图像)可以非常显著地不同(例如通过压缩)，而它们在感觉上又不能区分。理论上，散列函数模拟人类听觉系统(HAS)或人类视觉系统(HVS)的行为，即它生成被认为与 HAS/HVS 所生成的相同的内容的散列信号。但是，许多种类的处理(压缩、添加噪声、添加回波、D/A 和 A/D 转换、均衡等)可能用于信号，而且没有能够完美地模拟 HAS/HVS 的算法。一种复杂的因素是，甚至 HAS/HVS 对于不同的人以及不同的时间也不同，甚至单独一个 HAS/HVS 的概念也不是一成不变的。散列的经典定义未将时间考虑在内：健壮散列应该不仅能够标识内容，而且应该还能够标识时间(时间间隔)。为此，这里采用如下对健壮散列的定义:健壮散列是将多媒体内容的每个基本时间单元与对于HAS/HVS所感受的内容相似性而言是连续的半一致的比特序列相关联的函

数。

换言之，如果 HAS/HVS 标识两个非常相似的音频、视频或图像片断，则相关联的散列也应该非常相似。具体地说，原始内容和压缩内容的散列应该是相似的。再者，如果为交叠的各帧计算散列字，这些散列字应该是相似的，即散列应该具有低通特征。另一方面，如果两个信号真实地表示不同的内容，则健壮散列应该能够区分这两个信号(半一致)。这类似于经典密码散列的冲突要求。该散列函数所要求的健壮性是通过如下方式实现的：根据健壮特征(性质)、即对处理有很大程度的不变性的特征来推导散列函数。健壮性可以由误码率(BER)来表示，它被定义为错误比特数与总比特数之比。

健壮散列法可实现内容识别，这是许多感兴趣应用的基础。考虑识别多媒体数据库中的内容的实例。假定一个人正在观看电影中一个场景，想知道该镜头出自哪部电影。一个查找方法是将该场景与数据库中所有电影的尺寸相同的所有片断进行比较。显然，在大型数据库的情况中，此方法完全不可行：即使短的视频场景也是由大量字节表示的，可能需要与整个数据库进行比较。由此，为了实现此目的，需要存储大量易于访问的数据，所有这些数据都需要与要识别的视频场景进行比较。因此，存在存储量的问题(数据库)以及计算问题(配合大量数据)。健壮散列法通过减少表示视频场景所需的比特数来减轻这两个问题：需要存储的比特较少，需要用于比较的比特也较少。

首先说明音频信号的健壮散列。音频信号假定为单声道音频，它是以采样频率 44.1kHz(CD 质量)进行采样的。如果音频是立体声的，则有两种选择：或者散列信号是对左右声道分别提取的，或者在提取散列信号之前将左右声道相加。

即使只有较短音频片断(数秒的)，也想判断它是哪首歌。因为音频可以被看作无限的音频样本流，所以需要将音频信号再划分成时间间隔或帧，并对每一帧计算散列字。

往往在尝试匹配数据库中的散列时，无法确定帧边界。此同步问题特别适合于音频散列。此问题通过将信号分成交叠帧来解决。交叠还确保相邻帧的散列字具有一定的相关度。换言之，散列随时间缓慢改变。

图 1 表示根据本发明、用于生成音频散列信号的装置的实施例的示意图。首先在下降抽样器 11 对音频信号进行下降抽样，以便降低后续操作的复杂性且将该操作限制于 300-3000Hz 的频率范围，这个频率范围最切合人类听觉系统。

在成帧电路 12 中，音频信号被分成多个帧。这些帧被具有 16384 个样值的长度(≈0.4 秒)和 31/32 的重叠因子的汉宁窗进行加权。重叠是这样选择的：确保相继的帧之间的散列字具有高的相关性。每个帧的频谱表示由傅立叶变换电路 13 来计算。在下一个方框 14 中，计算(复数的)傅立叶系数的绝对值(量值)。

频带划分级 15 将频谱分成许多个(例如 33 个)频带。在图 1 中，此操作由选择器 151 来图示说明，它们各选择各频带的傅立叶系数。在装置的最佳实施例中，频带具有对数间距，因为 HAS 也在近似于对数的频带上工作。通过以此方式选择频带，散列将较小地受到处理变化(如压缩和滤波)的影响。在最佳实施例中，第一个频带起始于 300Hz 而每个频带具有一个乐音的带宽(即按每个频带 $2^{1/12}\approx1.06$ 的因子增加的带宽)。图 2 示出帧的频谱 201 及其划分成按对数规律间隔的频带 202 的实例。

随后，对每个频带，计算一定(不一定是标量)的特征性质。性质的实例是能量、音调以及功率谱密度的标准偏差。一般，所选的性质可以是傅立叶系数的任意函数。通过试验已经验证，每个频带的能量是对许多种类的处理最健壮的性质。此能量计算在能量计算级 16 进行。对于各个频带，它包括计算该频带内傅立叶系数的(平方)量值之和的级 161。

为了得到每帧的二进制散列字，随后将这些健壮的性质转换成

比特。这些比特可通过以下方式指定：计算可能不同的帧的健壮性质的任意函数，然后将其与阈值比较。阈值本身还可以是健壮性质值的另一个函数的结果。

在本装置中，比特推导电路 17 将频带的能级转换成二进制散列字。在一个简单实施例中，比特推导级为每个频带生成一比特，例如，如果能级在阈值之上，则为“1”；如果该能级在所述阈值之下，则为“0”。阈值对于不同频带可以不同。或者，如果其能级大于相邻频带的能级，则对该频带赋予散列比特“1”；否则该散列比特为“0”。本实施例采用后一方案的又改进的版本。为防止音频信号中的主要单个频率对于连续帧产生完全相同的散列字，还要将随时间的幅度变化考虑在内。更具体地说，如果其能级大于相邻频带的能级且前一帧也是如此，则为频带赋予散列比特“1”；否则该散列比特为“0”。如果以 EB(n,m)表示帧 n 的频带 m 的能量，以 H(n,m)表示帧 n 的散列字 H 的第 m 个比特，则比特推导电路 17 以如下方式生成散列字的比特：

$$\mathrm{H(n,m)}=\begin{cases}1 & \text{如果} EB(n,m)-EB(n,m+1)-(EB(n-1,m)-EB(n-1,m+1))>0\\ 0 & \text{如果} EB(n,m)-EB(n,m+1)-(EB(n-1,m)-EB(n-1,m+1))\le 0\end{cases}$$

至此，对于每个频带，比特推导电路 17 包括第一减法器 171、帧延迟 172、第二减法器 173 以及比较器 174。音频帧的频谱的 33 个能级由此被转换成 32 位散列字。最后将连续帧的散列字存储在可被计算机 20 访问的缓冲器 18 中。计算机将大量原始歌曲的健壮散列存储在数据库 21。

在后续操作中，采用相同装置计算未知音频剪辑的散列。图 3 中参考标号 31 表示存储在数据库 21 中的音频剪辑的 256 个连续交叠音频帧(≈3 秒)的散列字。图中，每行为一个 32 位散列字，白色像素表示散列字的“1”比特，而黑色像素表示“0”比特，时间由顶至底进行。参考标号 32 显示以 32 千比特/秒进行 MP3 压缩之后从同一音频剪辑中提取的散列字。理论上，两个散列块应该完全相同，

但是由于压缩的原因，某些比特会不同。此差异表示为图 3 中的 33。

现在说明图像或视频信号的健壮散列法。同样，健壮散列是由信息信号的特定特征推导出来的。要问的第一个问题是，要在哪个域中提取确定散列字的所述特征。与音频(其中频域最佳地表示了感觉特征)相比，要采用哪个域更不明确。由于复杂性的原因，最好避免诸如 DCT 或 DFT 变换之类的复杂操作。因此，计算时空域中的特征。再者，为了使从最大压缩视频流中的特征提取更容易，选择可以容易地由基于块的 DCT 系数来计算的特征。

基于这些考虑，最佳算法是以简单统计、如对相对较大图像区域计算的平均值和方差为基础的。这些区域以相当简单的方式来选择：图像帧被划分为 64×64 像素的方块。这些特征从亮度分量中提取。但是，这不是基本的选择：还可以采用色度分量。事实上，增加散列比特数的最容易的方法是以从亮度中提取的类似方式从色度分量中提取它们。

图 4 表示根据本发明、用于生成标识视频信号的散列信号的装置的方框图。该装置接收视频信号的连续帧。每帧被分成(41)M+1 个块。对这些块的每一个，计算(42)像素的亮度值的平均值。帧 p 中的块 k 的平均亮度表示为 F(p,k)，其中 k=0、...、M。

为了使散列独立于亮度的整体层次和标度，计算(43)两个相邻块之间的亮度差。此外，为了降低散列字在时间方向上的相关性，还计算(44,45)相邻帧中的空间不同的平均亮度值的差。换言之，将简单的时空 2×2 Haar 滤波器运用于平均亮度。结果的符号构成(46)帧 p 中块 k 的散列比特 H(p,k)。用数学方法表示如下：

$$H(p,k)=\begin{cases}1 & \text{如果}(F(p,k)-F(p,k-1))-(F(p-1,k)-F(p-1,k-1))\geq 0\\ 0 & \text{如果}(F(p,k)-F(p,k-1))-(F(p-1,k)-F(p-1,k-1))<0\end{cases}$$

在本实例中，每帧被分成尺寸为 64×64 的 33 个块(即 M=32)。完整的散列 H 由从 30 个连续帧提取的比特构成。包括 30 个各有 32

位的散列字的这种散列块(960 个比特)使错误正概率足够地小，如下所述。图 5 中的 51 表示典型的原始散列块，其中黑和白分别对应于“0”和“1”。参考标号 52 表示水平缩放为 94％的相同素材的对应散列块。标号 53 表示散列块 51 与 52 之间的差。在此情况中，误码率等于 11.3%。应当指出，错误比特在时间(垂直)方向上的确具有强相关性。

现在说明将所提取的散列块与大型数据库中的散列块匹配的处理。这是非平凡任务，因为众所周知不良匹配(切记所提取的散列字可能含有误码)是 NP 完全的。这一点通过下列(音频)实例来说明。在数据库中，存储约 5 分钟的 100000 首歌曲(每首歌曲≡25000 个散列字)。假定已经从未知的音频剪辑中提取了具有 256 个散列字的散列块(例如图 3 中的散列块 32)。现在确定这 100000 首存储的歌曲中哪一个与所提取的散列块最匹配。因此，需要找到这 100000 首歌曲之中与所提取的散列块最相似、即误码率(BER)最小或者 BER 低于一定阈值的一首歌中的散列块的位置。阈值直接决定了错误确认比率、即错误地从数据库中识别歌曲的比率。

如果两个推导的散列块 H_1 和 H_2 之间的汉明距离低于一定的阈值 T，则判断这两个 3 秒的音频剪辑(或两个 30 帧的视频序列)相似。该阈值 T 直接决定了错误确认比率 P_f，即错误判断两个音频剪辑/视频序列为相等(即在观看者眼中看来是不正确的)的比率：T 越小，概率 P_f 将越小。另一方面，小值的 T 会负面影响错误否认概率 P_n，即，两个信号“相等”、但未被识别为相等的概率。为了分析阈值 T 的选择，假定散列提取处理产生了随机 i.i.d.(独立且同样的分布)比特。则误码的数量将具有参数(n,p)的二项式分布，其中 n 等于提取的比特数而 p(=0.5)是提取“0”或“1”的概率。因为在此应用中 n(对于音频，$32 \times 256=8192$；对于视频，$32 \times 30=960$)是很大的，二项式分布可以由平均值为 $\mu = np$ 且标准偏差为 $\sigma = \sqrt{np(1-p)}$ 的正态分布来近似。给定散列块 H_1，随机选择的散列块 H_2 相对于 H_1 有少于 $T = \alpha n$ 个

误码的概率由下式给出：

$$P_f(\alpha)=\frac{1}{2\pi}\int_{(1-2\alpha)\sqrt{n}}^{\infty}e^{\frac{x^2}{2}}dx=\frac{1}{2}erfc\left(\frac{1-2\alpha}{\sqrt{2}}\sqrt{n}\right) \quad (1)$$

但是，在实际情况中，健壮散列沿时间轴具有高相关性。这由基础视频序列的大时间相关性或者音频帧的重叠所致。实验表明，错误比特的数量是正态分布的，但是标准偏差约为 i.i.d.情况的 3/2 倍。因此对公式(1)进行修改以包括因子 3/2。

$$P_f(\alpha)=\frac{1}{2}erfc\left(\frac{1-2\alpha}{3}\sqrt{2n}\right) \quad (2)$$

试验过程中采用的 BER 的阈值为 α=0.25。这意味着，8192 个比特中，要出现少于 2048 个误码，以便确认该散列块是取自同一首歌曲。在此情况中，这些误码具有 np=4096 的平均值 μ 和 $3\sqrt{(np(1-p))}$=135.76 的标准偏差 σ 的正态分布。所选阈值设置则对应于 15.2σ 的错误告警概率。因此，该错误告警概率等于 $1.8\cdot10^{-52}$。但是，应当指出，如果数据库中包括具有相似散列字的音乐(例如两个不同钢琴演奏家演奏一个莫扎特作品)，则错误告警概率实际上更高。

在数据库中搜索所提取的散列块的位置可以通过强制匹配来完成。这将进行约 25 亿(=25000 × 100000)次匹配。此外，匹配数目随数据库的大小而线性增加。

根据本发明的一个方面，计算机 20 采用更有效的策略以在数据库 21 中查找对应的歌曲。图 6 是由计算机完成的操作的流程图。当将原始歌曲存储在数据库中时，在步骤 60 计算机更新查找表(LUT)。LUT 在图 1 中表示成单独的存储器 22，但是应当明白，它实际上是大型数据库存储器 21 的一部分。如图 7 所示，LUT 22 含有对应于每个可能的 32 位散列字的入口。LUT 的每个入口指向一首或多首歌曲以及这些歌曲中各个散列字出现的位置。因为一个散列字可能出现在多首歌曲中的多个位置，所以歌曲指针被存储在链接的列表中。由此，LUT 可以生成多个候选歌曲。应当指出，当数据库中只有数量有限的歌曲时，包含 2^{32} 个入口的 LUT 可能是不切实际的。在此情

况中，最好用散列表和链接列表实现 LUT。图 7 中的参考标号 70 表示从未知音频剪辑中提取的 256 个散列字的块(例如图 3 中的散列块 32)。

在匹配方法的第一实施例中，假定常常单个散列字没有误码。在步骤 61，单个散列字 H(m)是从散列块中选择的并被发送到数据库。最初，这将是所提取的散列块的最后一个散列字 H(256)。在图 7 所示的实例中，这是散列字 0x00000001。数据库中的 LUT 指向歌曲 1 中的某个位置。假定此位置为位置 p。在步骤 62，计算机要计算所提取散列块与从歌曲 1 的位置 p-255 至位置 p 的散列字块(如图 7 中 71 所示)之间的 BER。在步骤 63，检查 BER 低(<0.25)还是高。如果 BER 低，则所提取的散列字源自歌曲 1 的概率较高。如果 BER 高，该歌曲不在数据库中或者单个散列字 H(m)含有差错。本实例中假定为后一种情况。然后在步骤 64 选择另一个单个散列字，并在 LUT 中查找。在图 7 中，现在正在查找最后且仅一个单个散列字 H(255)。此散列字看来似乎出现在歌曲 2 中。输入块 70 与存储块 72 之间的 BER 看上去是低于 0.25，这样歌曲 2 就被标识为该音频剪辑所源于的歌曲。应当指出，存储块 52 中的最后一个散列字是 0x00000000。显然，先前选择的散列字 0x0000001 有一个误码。

因此，计算机一次仅查看一个单个的散列字，而且假定时常此单个散列字没有误码。然后将提取的散列块的 BER 与候选歌曲的(时间轴上)对应的散列块进行比较。具有最低 BER 的候选歌曲的标题被选为所提取散列字所源于的歌曲，前提是最低 BER 低于阈值(步骤 65)。否则，数据库将报告找不到所提取的散列块。然后就尝试另一个单个散列字。如果任何一个单个散列字都没有获得成功(步骤 66)，则数据库将给予响应，报告数据库中不存在该候选歌曲(步骤 67)。

上述方法基于这样的假定：时常所提取的散列字没有误码，即它完全等于对应的存储散列字。广泛的实验表明，对于大多数音频来说，这会有规律地每秒出现几次。例如图 8 中说明了这一点，其

中表示构成图 3B 的提取块的 256 个散列字中的误码数。在此 3 秒的音频剪辑中出现十三个没有任何误码的散列字。

但是，当对音频进行严重处理时，出现没有任何误码的散列字是不可能的。在此情况中，通过先前的方法无法检索歌曲的标题。至此，将说明匹配方法的另一个实施例。此方法采用散列提取算法的软信息在数据库中查找所提取的散列字。软信息应理解为表示比特的可靠性或者已经正确检索出散列比特的概率。在本实施例中，用于提取散列字的装置包括比特可靠性判断电路。比特可靠性判断电路在图 1 所示的音频散列提取装置中表示为 19。此电路以实数的形式接收差分能带电平。如果实数非常接近于阈值(此实例中为 0)，此个别的散列比特是不可靠的。相反，如果此数与阈值差很远，则它是可靠的散列比特。该阈值可以是固定的或者受控的，使得可靠比特的数量为固定的。

比特可靠性判断电路 19 判断每个散列比特的可靠性，从而使此提取装置或计算机 20 可以为每个散列字生成最可能备选散列字的列表。通过再次假定备选散列字中至少一个是正确的，就可以正确和容易地接收歌曲标题。图 9 表示对于图 3 中的散列块 32 的所有 256 个散列字来说，散列字的哪个比特是最可靠的。

图 10 是在数据库中查找提取散列块的方法的这个实施例中、由计算机完成的操作的流程图。对于前面已描述的操作，采用相同的参考标号。同样地，最初选择散列块的最后一个提取的散列字(0x00000001，见图 7)，并将其发送到数据库(步骤 61)。数据库中的 LUT 指向歌曲 1 中的位置 p。计算提取的散列块和歌曲 1 中的对应块 71 之间的 BER(步骤 62)。同时，从先前实例已知，该 BER 较高。在步骤 101，计算机现在询问比特可靠性判断电路 19(图 1)，了解到比特 0 是此特定散列字的最不可靠比特。现在通过翻转所述比特来获得下一个最可能的候选散列字。在步骤 102，新的散列字(0x00000000)被发送到数据库。如图 7 所示，散列字 0x00000000 产生数据库中的

两个可能的候选歌曲：歌曲 1 和歌曲 2。例如，如果所提取的散列字现在相对于歌曲 2 的散列字具有低 BER，则歌曲 2 将被标识为该提取散列块所源于的歌曲。否则，将生成新候选散列字，或者将采用另一个散列字来尝试查找数据库中的各个歌曲。此策略将继续进行，直到在步骤 103 发现再没有可选的候选散列字为止。

应当指出，一旦音频片断被实际识别为取自某首歌曲，数据库可以在生成所有候选散列字之前，先尝试将所提取的散列字与该歌曲进行匹配。

一个非常简单的生成最可能的散列字的列表的方法是包括具有 N 个固定的最可靠比特和其余比特的每种可能组合的所有散列字。在每个散列 32 位且选择 N=23 的情况中，需要 512 个候选散列字的列表。此外，它意味着在再也不能识别音频选段之前，散列字的 9 个最不可靠比特可能是错误的。对于图 6 所示的情况，这意味着 117 个散列字(而不是先前方法的 13 个散列字)会产生指向数据库中歌曲的正确指针。

在匹配方法的一个可选实施例中，匹配只是基于被标记为可靠的散列比特进行的。此方法基于这样的理解：没必要将所接收散列的不可靠比特与数据库中的相应比特进行比较。虽然付出的代价是更复杂的搜索策略和将所有必需信息发送到数据库所需的更大带宽，但是得到了小得多的误码率。

现在说明健壮散列法的几个应用。

广播监控：广播监控系统包括两个部分：包含大量歌曲的散列的中央数据库；以及从例如广播电台所广播的音频中提取散列块的监控台。监控台将所提取的散列块发送到中央数据库，然后数据库就可以判断哪首歌曲已经广播过。

移动电话音频信息：想象一下，坐在酒吧里并倾听着想知道其名称的歌曲。然后只需拿起移动电话，拨打音频散列数据库。音频散列数据库就收听该歌曲并提取散列块。如果它在数据库中找到该

散列块，就会报告歌曲的名称。

连接的内容(“媒体桥”)：Digimarc 公司目前有一种称为“媒体桥”的应用，它是基于水印技术。其理念是多媒体片断中的水印把用户引导到因特网上某个 URL，在那里他可以获取某些额外信息。例如，对杂志中的广告进行水印处理。将此广告放置在网络摄相机(webcam)前方，水印检测器提取要发送到数据库的水印密钥。该数据库就包含了这样的 URL，用户要重定向到该 URL。同样的应用可以对健壮散列技术的使用行得通。将来，甚至可以设想，一个人将他的移动视频电话对准现实中的实物。音频散列数据库就会直接或通过因特网上的 URL 返回报告有关该实物的信息。

多媒体质量测量：如果高质量的原始内容的散列字被列于数据库中，则可以通过确定处理后的多媒体内容的提取散列字的 BER 来获得质量测量值。

从抽象的角度来看，健壮音频散列是从音频信号中通过比较不同频带和不同时间上的能量来推导的。此方法的广义形式是考虑 LTI 和非线性函数的任何级联。具体来说，健壮散列还可以通过以下方式获得：应用(二元)滤波器组(LTI 算子)，然后进行平方或取绝对字(非线性函数)，然后是时间和/或频带上的差异算子(LTI 算子)，最后是阈值处理算子。通过运用仔细设计的线性滤波器组作为初始算子，可以避免 FFT 的复杂性。此外，由于许多压缩引擎具有作为初始级的线性滤波器组，所以可以选择将特征提取与压缩集成。

还应指出，可以组合地采用健壮散列法和数字水印来识别内容。上述方法和某些水印检测算法有许多共同的初始处理步骤、即频谱表示的计算。这导致这样的想法：可以很容易地将水印检测与特征提取集成在一个应用中。然后可以同时将检索的水印和散列字发送到中央数据库以供进一步分析，从而实现内容识别。

总之，所公开的方法为多媒体内容、例如音频剪辑生成了健壮散列。音频剪辑被分成(12)连续(最好交叠的)帧。对于每一帧，又将

频谱分成(15)频带。计算(16)每个频带的健壮特性(例如能量)并通过各个散列比特来表示(17)。由此，音频剪辑就由一串二进制散列字(每帧一个散列字)来表示。要识别可能经过压缩的音频信号，计算机(20)将从中推导的散列字块与大型数据库(21)进行匹配。这些匹配策略也已公开。在最佳实施例中，提取处理还提供了关于散列比特中哪些最不可靠的信息(19)。将这些比特翻转会显著地提高匹配处理的速度和性能。

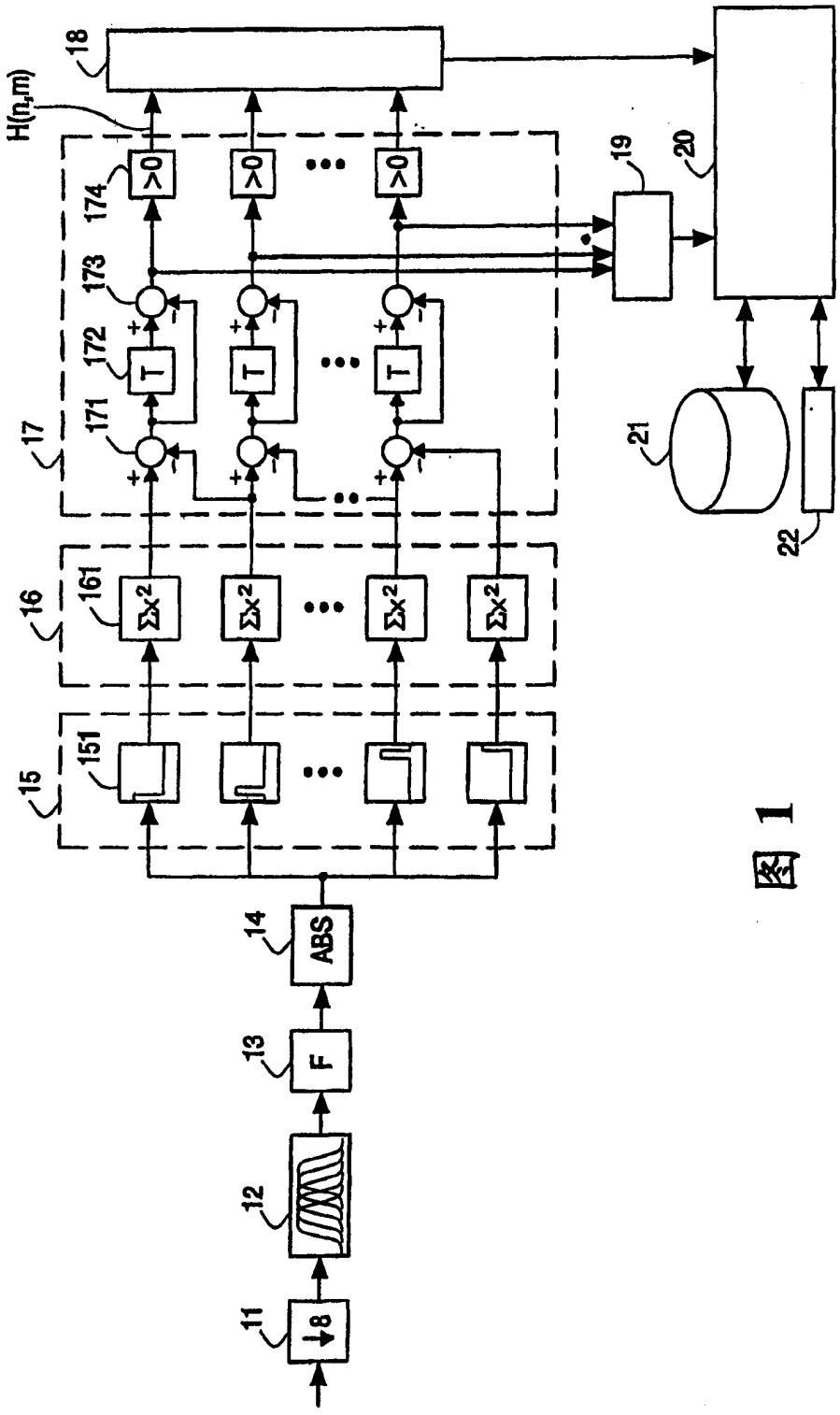
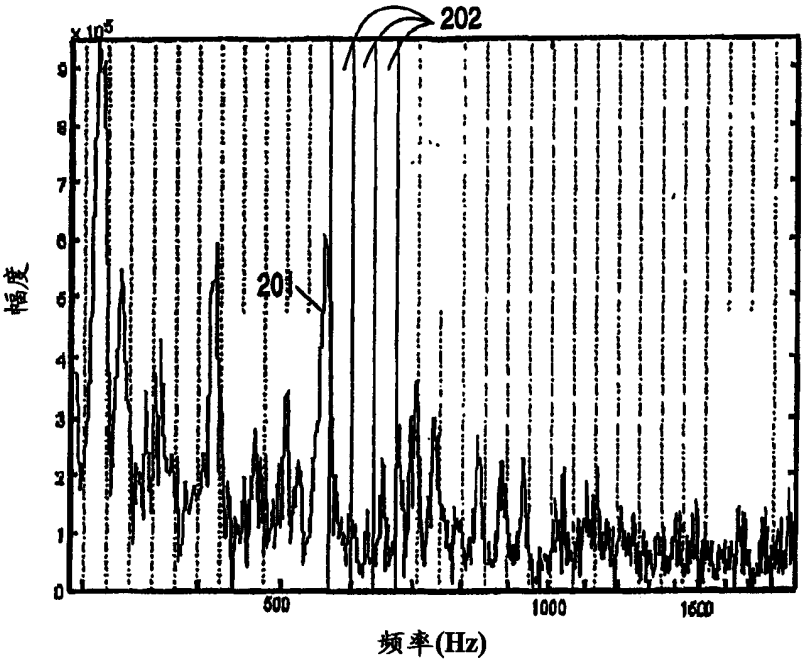

18
H(n,m)
174
173
172
171
17
16
161
15
151
14
ABS
13
F
12
11
↓8
19
20
21
22
T
Σx²
>0

图 1

图 2


x 10^5
幅度
202
203
500
1000
1500
频率(Hz)

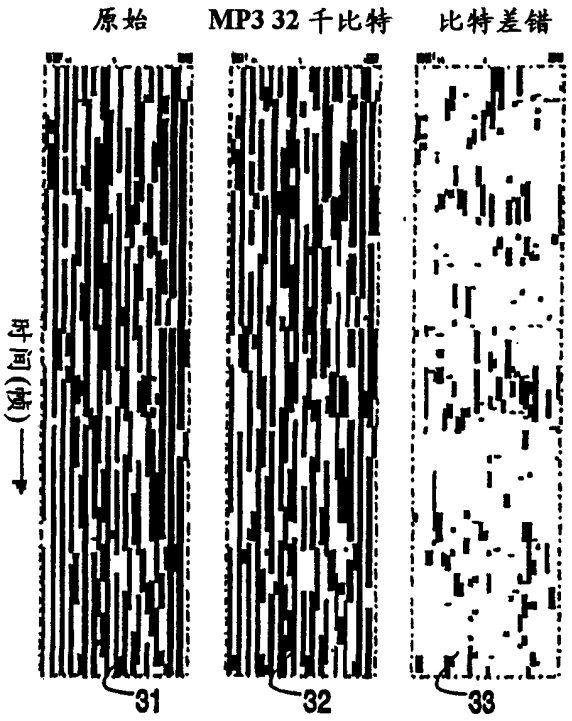
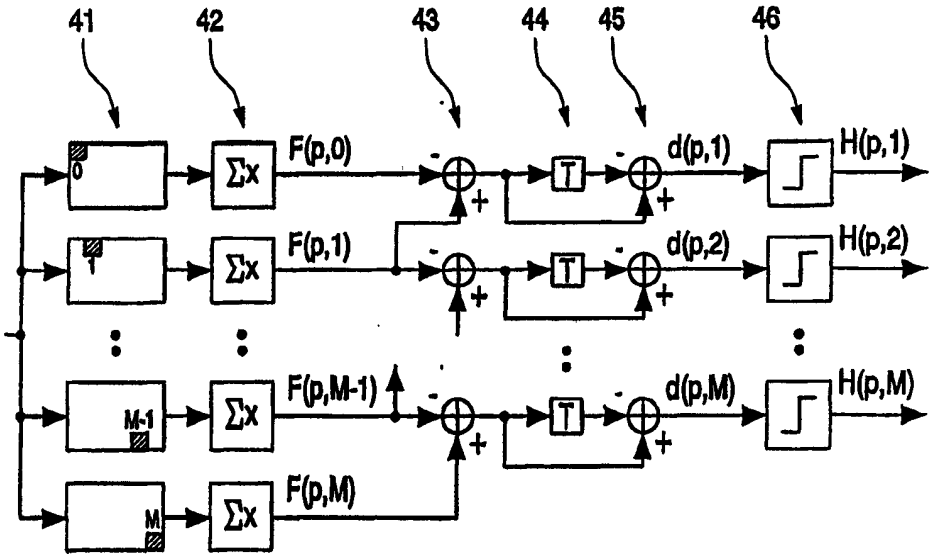

原始
MP3 32 千比特
比特差错
时间(秒)
31
32
33

图 3

图 4


41
42
43
44
45
46
0
1
M-1
M
ΣX
F(p,0)
F(p,1)
F(p,M-1)
F(p,M)
T
d(p,1)
d(p,2)
d(p,M)
H(p,1)
H(p,2)
H(p,M)

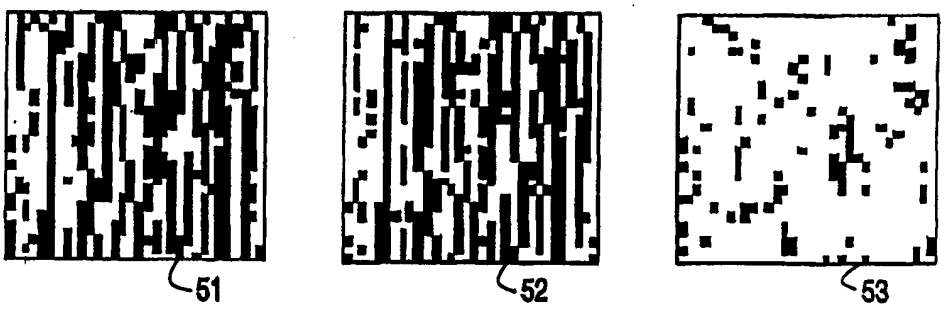
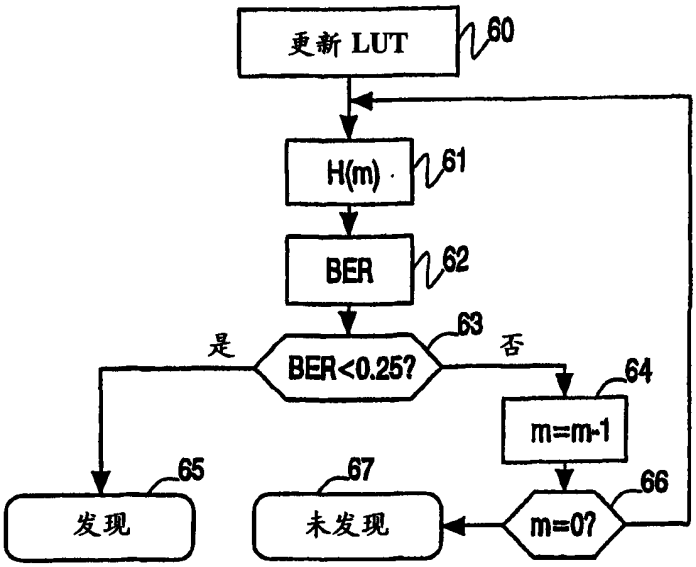

51
52
53

图 5

图 6


更新 LUT
60
H(m)
61
BER
62
BER<0.25?
63
是
否
m=m-1
64
m=0?
66
发现
65
未发现
67

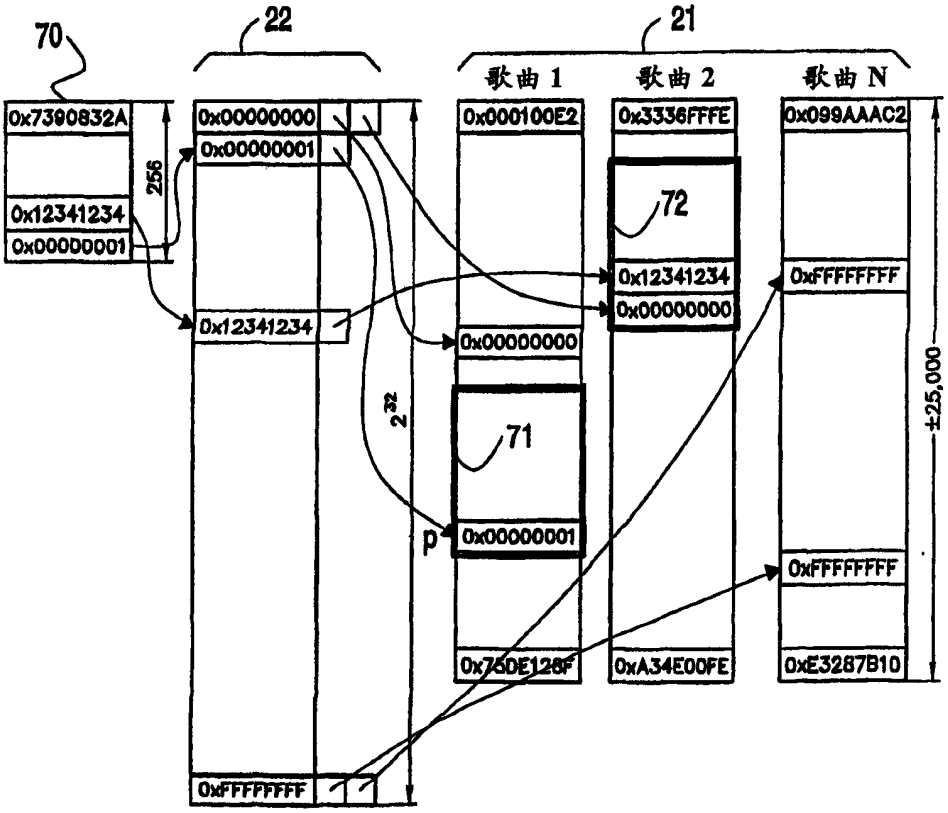
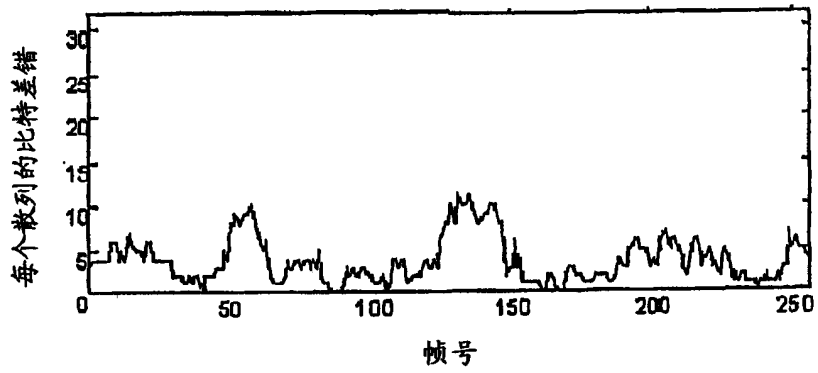

70
22
21
歌曲 1
歌曲 2
歌曲 N
0x7390832A
0x12341234
0x00000001
256
0x00000000
0x00000001
0x12341234
0xFFFFFFFF
0x000100E2
0x3336FFFE
0x099AAAC2
72
0x12341234
0x00000000
0xFFFFFFFF
0x00000000
71
p
0x00000001
0xFFFFFFFF
0x75DE128F
0xA34E00FE
0xE3287B10
2^{32}
±25,000

图 7

图 8


每个散列的比特差错
30
25
20
15
10
5
0
50
100
150
200
250
帧号

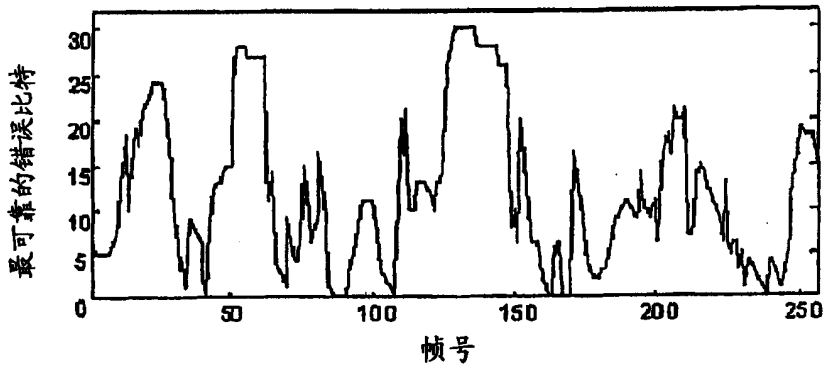

最可靠的错误比特
30
25
20
15
10
5
0
50
100
150
200
250
帧号

图 9

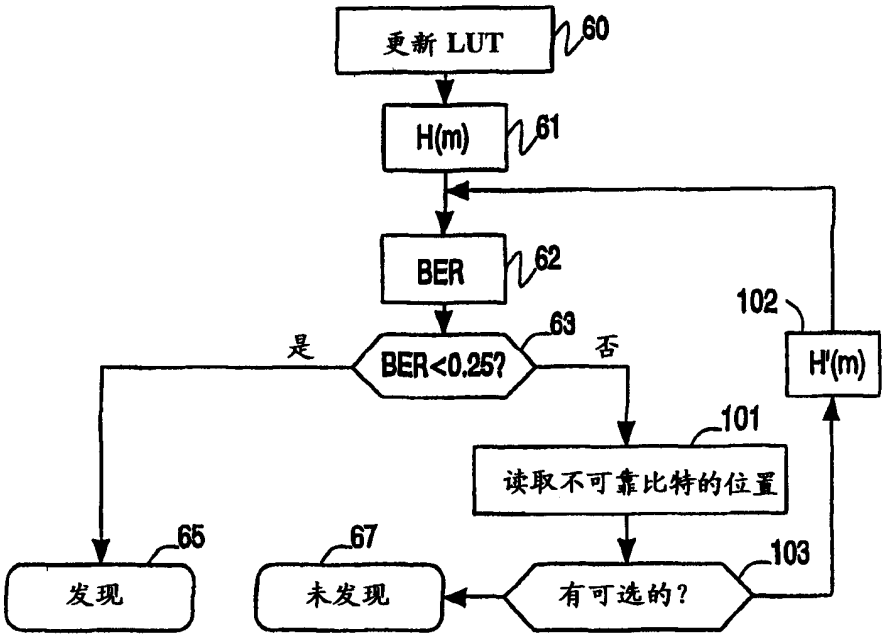

更新 LUT
60
H(m)
61
BER
62
BER<0.25?
63
是
否
H'(m)
102
101
读取不可靠比特的位置
发现
65
未发现
67
有可选的?
103

图 10