

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 17/30 (2006.01)



[12] 发明专利申请公布说明书

[21] 申请号 200880018885.0

[43] 公开日 2010年3月24日

[11] 公开号 CN 101681381A

[22] 申请日 2008.6.4

[21] 申请号 200880018885.0

[30] 优先权

[32] 2007.6.6 [33] US [31] 60/933,614

[86] 国际申请 PCT/US2008/007054 2008.6.4

[87] 国际公布 WO2008/150544 英 2008.12.11

[85] 进入国家阶段日期 2009.12.4

[71] 申请人 杜比实验室特许公司

地址 美国加利福尼亚

[72] 发明人 蒋文宇 C·鲍尔

[74] 专利代理机构 中国国际贸易促进委员会专利
商标事务所
代理人 魏小薇

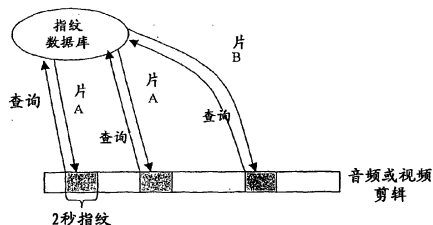
权利要求书2页 说明书13页 附图4页

[54] 发明名称

使用多搜索组合改善音频/视频指纹搜索精确度

[57] 摘要

一种用于识别音频和/或视频信息的分段的方法，包括：在所述分段中多个间隔开的时间位置的每一个时间位置处获得查询指纹；对于每一个这样的查询指纹，在数据库中搜索潜在匹配的指纹；对于每一个这样的查询指纹，获得与数据库中所找到的指纹的潜在匹配的置信级别；以及将搜索潜在匹配的结果组合，其中，对每一个潜在匹配结果以各自置信级别进行加权。置信级别可作为以下至少一项或两项的函数：(1) 查询指纹与所找到的指纹之间的差异量度，以及(2) 在查询指纹的时间位置与所找到的指纹的时间位置之间的相对定时关系。



1. 一种用于识别音频和/或视频信息的分段的方法，包括：
在所述分段中多个间隔开的时间位置中的每一个时间位置处，
获得查询指纹；
对于每一个这样的查询指纹，在数据库中搜索潜在匹配的指纹；
对于每一个这样的查询指纹，获得与数据库中所找到的指纹的潜在匹配的置信级别；以及
将搜索潜在匹配的结果组合，其中，对每一个潜在匹配结果以各自置信级别进行加权。
2. 根据权利要求 1 的方法，其中，置信级别为以下至少一项或两项的函数：（1）查询指纹与所找到的指纹之间的差异量度，以及（2）在查询指纹的时间位置与所找到的指纹的时间位置之间的相对定时关系。
3. 根据权利要求 2 的方法，其中，所述差异量度基于汉明距离。
4. 根据权利要求 3 的方法，其中，所述差异量度为误码率。
5. 根据权利要求 2 的方法，其中，所述差异量度为 L_p 范数，在该 L_p 范数中，获得在构成指纹的向量的每一个维度上的距离的绝对值的 p 次幂之和的 p 次根。
6. 根据权利要求 5 的方法，其中，所述差异量度为 L_2 范数距离量度，在该 L_2 范数距离量度中，获得距离的绝对值的平方之和的平方根。
7. 根据权利要求 2 的方法，其中，通过自相关获得所述差异量度，其中，相关值表示匹配的接近度。
8. 根据权利要求 2-7 中任一项的方法，其中，所述置信级别为以下至少一项或两项的函数中的每一个的各置信级别的组合：（1）查询指纹和所找到的指纹之间的差异量度，以及（2）在查询指纹的

时间位置与所找到的指纹的时间位置之间的相对定时关系。

9. 根据权利要求 8 的方法，其中，各置信级别的所述组合为各置信级别的总和。

10. 根据权利要求 8 的方法，其中，各置信级别的所述组合为各置信级别的乘积。

11. 根据权利要求 1-10 中任一项的方法，其中，所述时间位置是间隔开的，以便使查询指纹之中或之间搜索结果的相关性最小化。

12. 根据权利要求 1-11 中任一项的方法，其中，存在间隔开的时间位置的多个集合，关于每一个这样的集合，将搜索潜在匹配的结果组合。

13. 根据权利要求 12 的方法，其中，在每一个集合内将搜索结果组合，以及将这样组合的结果进一步进行组合。

14. 根据权利要求 12 的方法，其中，跨所有集合对搜索结果进行组合。

15. 根据权利要求 1-14 中任一项的方法，其中，在数据库中的指纹具有关于所述音频和/或视频信息的相关元数据。

16. 根据权利要求 15 的方法，其中，所述元数据至少包括音频和/或视频信息的片标题。

17. 根据权利要求 1-16 中任一项的方法，其中，以基本相同的方式，在所述多个间隔开的时间位置的每一个时间位置处，获得查询指纹。

18. 根据权利要求 2-17 中任一项的方法，其中，置信级别为至少在查询指纹的时间位置与在数据库中所找到的潜在匹配指纹的时间位置之间的相对定时关系的函数，且存在至少三个间隔开的时间位置。

19. 一种适于执行根据权利要求 1-18 中任一项的方法的装置。

20. 一种存储在计算机可读介质上的计算机程序，用于使计算机执行根据权利要求 1-18 中任一项的方法。

使用多搜索组合改善音频/视频指纹搜索精确度

技术领域

本发明涉及音频和视频指纹。更具体而言，本发明涉及改善指纹搜索精确性的精确度。

背景技术

随着数字内容的快速增长，越发需要对内容进行合适的管理和定位。一种未来应用是识别已知内容库内的音频或视频剪辑（**excerpt**）。这在监视因特网上内容的非法下载/传输中可以是有益的。它还可用于提供对用户所收听或观看以及感兴趣但不知道其标题或作者信息的内容的智能访问。例如，用户可将蜂窝电话放在当前播放歌曲的扬声器前，并且使蜂窝电话运营商的软件找到歌曲标题、它的艺术家和专辑信息等。

这样的识别功能通常通过这样的步骤来实现，即，首先生成被设计用于唯一地识别音频和视频信号的特征信息（称为**指纹**），然后，在来自库/数据库的指纹与来自搜索查询中的剪辑的指纹之间，执行某种形式的模式匹配检索。这样的数据库通常采用搜索树的形式来实现，不过其他数据结构也是可以的。一般而言，每一个指纹对应于音频或视频的某个分段（**segment**）。这样，2秒音频指纹将对应于2秒音频分段。一般将指纹实现为通常被称为签名的特征信息小块的串接（**concatenation**）。例如，2秒指纹可由200个10毫秒（**ms**）长的签名组成，其中，每一个签名由10ms的新音频或视频信息计算得出。因此，指纹的模式匹配是对相应签名进行比较的处理。这在图1中被示出，该图显示由签名创建指纹。

为了执行在来自查询剪辑的指纹与来自数据库的指纹之间的合适的模式匹配，关键是在比较期间在二者之间具有合适的时间对准。

为了确保这样，通常将开始于每一个可能时间偏移的所有指纹都添加到数据库，以便保证它们中的至少一个具有足够接近查询指纹的时间对准。如果签名为 10 ms 长，则在 2 秒滑动窗口上的每 10 ms 都对 2 秒指纹进行偏移，然后将其添加到数据库。在图 1 中也对此进行图示，在该情形中，它在连续指纹之间产生 99.5% 重叠，不过通常需要有这样的冗余以确保好的搜索性能。对于不足 10 ms（或通常为每一个签名的持续时间）的任何剩余时间对准，良好设计的指纹生成方法应选择定时方面更近的任何签名，以便更好与来自查询的相应签名相匹配。简而言之，指纹搜索系统的目标是发现与查询指纹的相应的配对物（counterpart）指纹，如果根本就存在时间不对准，那么该配对物应将其对于查询指纹的时间不对准最小化。

因为查询剪辑可能已经经历某些编辑或处理步骤（如使用蜂窝电话再次捕获由扬声器播放的声音），所以捕获的音频/视频信号中可存在某些失真。结果，得到的指纹也可相对于数据库中的它们的配对物（假设存在配对物）轻微地变化。

剪辑中的失真的可能性意味着在这样搜索中的最佳匹配通常不是完全相同的匹配，而是最接近的匹配的形式。为定义最接近的匹配，需要定义在两个指纹之间的差异量度（measure）。例如，普遍使用的差异量度为汉明（Hamming）距离，也就是，在来自查询剪辑的指纹与来自数据库的指纹之间的差异位数。利用该差异量度定义，从而匹配最接近的相应标准是来自数据库的指纹与来自查询剪辑的指纹具有最小汉明距离。通常将两个指纹之间的汉明距离除以指纹位数称为误码率（BER）。BER 是相对差异量度的例子。当来自剪辑的指纹与其配对物之间的 BER 小时，最小汉明距离标准工作良好。然而，当 BER 增大时，生成最小汉明距离的搜索结果愈发不能找到真正的配对物。幸运的是，在大多数指纹搜索应用中，仅有必要识别正确的音频/视频片，而不必识别相应的分段。但是当 BER 进一步增大时，搜索结果可能甚至找到错误的音频/视频片，更不用说该片内正确的分段了。BER 取决于查询剪辑中失真的级别和指纹提取

方法相对于这种失真的健壮性。

此外，剪辑有可能不属于数据库中的任何片。例如，剪辑可为新的合成音乐的录音。由于没有搜索算法能够事先（在不被告知的条件下）知道剪辑是否属于数据库，它能做的最佳方案仍然是应用最小汉明距离的相同标准，不过预期在该情形中找到的最小汉明距离与源自数据库的剪辑的最小汉明距离相比将会不同得更多（优选更高），并且使用一些阈值来确定该剪辑是否来自数据库。

因此，在单个搜索操作（其中，仅使用一个查询指纹搜索数据库）之后，在应用任何阈值（例如，根据 BER）之前，存在三种可能的结果：

- 1、该剪辑属于数据库，搜索返回正确的音频/视频片（找到正确的片就足够了，在此不必找到正确的配对物分段）。
- 2、该剪辑属于数据库，搜索返回错误的音频/视频片。
- 3、该剪辑不属于数据库，并且因为搜索总是返回一些音频/视频片，所以回答总是错误的。

图 2 显示单个搜索的三个不同可能结果的 BER 分布例子。这些结果中的每一个都将生成 BER 的相应概率-密度-函数（PDF）分布。如图 2 所示，对于良好设计的指纹提取算法，第一结果的 BER 通常会比第二和第三结果的 BER 小。

然而，如果第二和第三结果的 BER 具有非常相似的 PDF 分布，则将会难以区分属于数据库但具有错误搜索结果的剪辑与不属于数据库的剪辑。此外，对于源自数据库的片而言，在应用诸如 CODEC 压缩的普通音频/视频失真之后，在应用任何 BER 阈值之前，取决于指纹持续时间和失真类型，通常实施方式的搜索结果正确（就识别正确的片而言）的范围通常从 90 到 99%。其效果良好，不过当然希望获得更高级别的精确性，并且在应用 BER 阈值（如在图 2 中，BER=0.2）之后，正确搜索结果的比率仅轻微下降，因为将结果 1 的 BER 分布的尾部丢弃，以避免错误地过多拾取结果 2 的头部分布。这意味着，在单个搜索中，仅调节（tweak）BER 阈值不能导

致非常高（如 99.9%）的精确性。

附图说明

图 1 是可生成特定类型指纹的方式的示意性例子。

图 2 显示概率-密度-函数（PDF）的理想例子，所述概率-密度-函数（PDF）为对于单个指纹匹配搜索的三个可能结果中的每一个可得出的误码率（BER）的函数。

图 3 是关于三个搜索使用多数表决的指纹-匹配搜索结果组合的例子的示意性概念。

图 4a 是对于三个搜索的指纹-匹配搜索结果的例子的示意性概念，其中，对于三个潜在匹配的其中两个而言，时间位置是一致的。

图 4b 表示对于三个搜索的指纹-匹配搜索结果例子的示意性概念，其中，对于三个潜在匹配的其中两个而言，时间位置不一致。

图 5 是可将置信级别映射（map）成误码率（BER）的函数的方式的例子。

图 6 是可用于理解确定在指纹查询的时间与数据库中找到的指纹的时间之间的相对时间位置的一种使用线性回归的方式的理想例子。

图 7 是可将置信级别映射为在指纹查询的时间与数据库中找到的指纹的时间之间的相对时间位置的时间差的函数的方式的例子。

发明内容

根据本发明的方面，用于识别音频和/或视频信息的分段的方法包括，在所述分段中多个间隔开的时间位置中的每一个时间位置处，获得查询指纹；对于每一个这样的查询指纹，在数据库中搜索潜在匹配的指纹；对于每一个这样的查询指纹，获得与数据库中所找到的指纹潜在匹配的置信级别；以及将搜索潜在匹配的结果组合，其中，对每一个潜在匹配结果以各自的置信级别进行加权。

置信级别可作为以下至少一项或两项的函数：（1）查询指纹与

所找到的指纹之间的差异量度，以及（2）在查询指纹的时间位置与所找到的指纹的时间位置之间的相对定时关系。差异量度可基于汉明距离，或其相对差异变化量（如误码率），或者，它可基于其中获得在构成指纹的向量的每一个维度上的差异的绝对值的 p 次幂之和的 p 次根的 L_p 范数。 L_p 范数度量度可为其中获得差异的绝对值的平方之和的平方根的 L_2 范数差异量度。作为又一可选方案，可通过自相关获得差异量度，其中，相关值表示匹配的接近度。

置信级别可为以下至少一项或两项的函数中的每一个的各置信级别的组合：（1）查询指纹和所找到的指纹之间的差异量度，以及（2）在查询指纹的时间位置与所找到的指纹的时间位置之间的相对定时关系。各置信级别的组合例如可为各置信级别的总和、或者各置信级别的乘积。

置信级别可为至少在查询指纹的时间位置与在数据库中所找到的潜在匹配指纹的时间位置之间的相对定时关系的函数，在该情形中，存在至少三个间隔开的时间位置。

在音频和/或视频信息分段中的间隔开的时间位置可以被间隔开，以最小化查询指纹之中或之间的相关性。

可存在间隔开的时间位置的多个集合，关于每一个这样的集合，将搜索潜在匹配的结果组合。可在每一个集合内将搜索结果组合，然后将这样组合的结果进一步进行组合。备选地，可以跨所有集合对搜索结果进行组合。

在数据库中的指纹可具有关于所述音频和/或视频信息的相关元数据。这样的元数据可至少包括音频和/或视频信息的片标题。

以基本相同的方式，在所述多个间隔开的时间位置的每一个位置处，可获得查询指纹。

对来自多个搜索的结果进行组合的简单而又有用的方法是多数表决。例如，如图 3 所示，如果执行三个搜索，其中有两个返回片 A，一个返回片 B，则得出结论，最终搜索结果应该是片 A。如果获得三个不同的回答：片 A、片 B 和片 C，则不会有多数胜出者，回答

是不确定。在此情形中，最好是得出剪辑不属于数据库的结论，因为如果它真的属于，那么三个搜索将给出三个不同的结果是相对地不可能的。

使用后面所示伪代码所示出的多数表决（其中获得超过半数投票的片胜出），或实施者希望的任何表决方案（例如，具有最多投票且无并列者（tie）的片胜出），可将此推广到任意数量的搜索。应使用概率分析将表决方案设计成使得正确回答数量最大，错误回答数量最小。例如，对于源自数据库的片，如果单个搜索有 98% 精确度，则仅若三个搜索中至少有两个错误的情况下，利用多数表决组合三个搜索才会得出错误结论，其为 $3 \times 0.98 \times 0.02^2 + 0.02^3 \approx 1.2 \times 10^{-3}$ ，从而，组合精确度应几乎为 99.9%。该概率分析假设每一个搜索都独立于其他搜索，即，搜索之间不相关。

使用多数表决的搜索组合的示例性伪代码可表示如下：

```

N=3;
分别在剪辑的时间 t, t+d[1], ..., t+d[N-1]处进行 N 个指纹搜索;
If (大于 N/2 个搜索返回相同的片 A)
    返回 片 A
Else
    返回“剪辑不属于数据库”

```

在图 3 中的时间间隔 $d[1]$, $d[2]$, ..., 用于在不同位置处执行搜索，一般而言，将它们选择为使得其相应搜索分段之间的重叠最小，以便减小在搜索结果之间的任何不希望的相关。如果出现无结论的投票的情形，组合处理还可与多个搜索的附加迭代级连，或者仅考虑在新迭代中的投票，或者将先前投票中的一些或所有与实施者选择的投票方案进行组合。

像多数表决这样的简单方案看起来将能够给出非常好的结果。以上例子给出几乎 99.9% 的精确度，但当在搜索中存在强突发性

(burstiness) 或错误的相关性时, 实际精确度可能仅比最初的 98% 稍好一点: 如果一个搜索给出错误结果, 在附近时间位置处的搜索也同样可能给出错误的结果, 且往往给出相同的错误结果。例如由于音频/视频的某些区域 (如视频中的高运动分段或安静的音频信号) 对失真 (分别为低比特率压缩或背景噪音) 较为敏感, 则会出现这样的情形。这将误导多数判决算法在它没有发现正确的答案时认为它发现了正确的答案。另外, 如果片不属于数据库, 还必须处理多于一个搜索可能返回同样片的情形, 因为它将导致错误的多数胜出者。

通过采用本发明的方面, 即便存在这样的突发错误, 也可以提高精确度。例如通过不仅将来自每一个搜索结果的片的标题信息、并且将诸如所发现的分段在所发现的片内的时间位置之类的附属信息与搜索结果的 BER 进行组合, 就可实现它。例如, 在图 4 (a) 中, 如果三个搜索中的两个返回片 A, 这两个搜索分段的时间位置差 5 秒, 则在这两个搜索结果中所发现的分段的时间位置也应差 5 秒 (或在一定误差容限内)。如果这还不满意的话, 如图 4 (b) 中所示, 几乎肯定这是错误的多数胜出者。这是公平的假设, 因为如果搜索返回错误片, 它几乎肯定是由于它碰巧是生成最小汉明距离的结果的意外事件引起的。因此, 所发现分段的时间位置应该是相对随机的, 而非确定的。从而, 两个错误搜索导致返回相同错误片和一对始终接近的时间位置的概率是相当低的。通过对正确和错误搜索结果的时间位置的 PDF 分布进行分析, 能够估计出确切的概率。

将该比较推广到 N 个搜索, 它意味着在这些搜索的时间位置之间的差异对这些查询与搜索结果还应该是相同的或接近的。然而, 为了较易于在软件中实施该比较, 可计算在原始片和搜索结果中的剪辑的时间位置之间的偏移 (即, 差异), 在图 4 (a) 和 4 (b) 中该值被称为 t_{offset} 。第一搜索和第二搜索的偏移分别为 $t_{\text{offset},1}$ 、 $t_{\text{offset},2}$ 。对于每一对返回相同片的搜索结果, 可以计算它们的 t_{offset} 之间的差异。然后, 对于实现最小差异的对, 如果这样的最小差异小于某个阈值 (如, 0.2 秒), 则可将其片标题视为最终回答。这在以下用于使

用附属信息的搜索组合的伪代码中被示出，其中，`Min_t_offset_dist` 为最小差异，而候选者（`Candidate`）表示对于哪个（对）搜索实现了该最小化。具有 0.2 秒的示例值的阈值 `d_A` 被用于确定结果是否足以置信。

使用 `t_offset` 附属信息进行搜索组合的示例性伪代码可表示如下：

```
N=3;
```

```
分别在剪辑的时间 t, t+d[1], ..., t+d[N-1]处进行 N 个指纹搜索;
```

```
Min_t_offset_dist=min(t_offset,i- t_offset,j, 其中, search[i].piece==search[j].piece, i<j);
```

```
Candidate=arg,min(t_offset,i- t_offset,j, 其中, search[i].piece==search[j].piece, i<j)
```

```
;
```

```
d_A=0.2;
```

```
If ( Min_t_offset_dist< d_A && Candidate 有大于 N/2 的投票)
```

```
    返回 search[Candidate].piece
```

```
Else
```

```
    返回“剪辑不属于数据库”
```

在以上伪代码中，“`search[i]`”表示来自第 `i` 次搜索的结果，“`search[i].piece`”为找到的片的标识信息（如其标题）。如下所述，使用附属信息定时信息的更普遍方式采用时间量度 `t_distance` 而非 `t_offset`。

对于大的 `N` 次搜索，如果更多搜索返回相同片（倘若结果是正确的），则显然 `Min_t_offset_dist` 的值会较小。例如，其他所有都相同，五个采样之中的最小值肯定比三个采样中的最小值小。从而，除 `d_A` 外，我们可以定义更多阈值（例如，`d_B`，`d_C`），这取决于多少搜索返回相同的片。依此方式，可基于多数表决信息（有多少搜索返回相同的片）智能地使用附属信息（在该例子中为 `t_offset`），并且

实现更高的精确性。

如果查询材料已经经历一些速度提升或速度下降，在收音机节目中有时出现这样的情形，查询指纹之间的间隔与来自搜索结果的指纹之间的间隔将不再相同。而是它们将会相对于查询指纹之间的间隔慢慢漂移开。为处理该情形，可确定查询指纹之间的间隔决不会导致 t_{offset} 的偏差超过阈值（如 d_A ）。或者，如果投票数量足够的话，可观测查询指纹与来自搜索结果的指纹之间的定时相关，这应产生斜率（slope）稍高或低于 1 的线性趋势，这取决于它是经历速度提升或者速度下降（假设不会有任何其他失真会导致在来自搜索结果的指纹的定时中出现差错）。那么，可使用线性趋势的置信度量（如基于线性回归）确定组合结果的置信度。这样的线性回归的例子示出在后面所讨论的图 6 中。

类似于 t_{offset} 的概念，如果在搜索结果中的 BER 非常低，这表示非常接近的匹配，可以对结果的精确性高度置信。然而，如果 BER 高，则结果较不可信。此外，当以逻辑 AND 方式使用不止一片附属信息时，不限于对它们仅使用固定的阈值。例如，如果阈值 d_A 最初为 0.2 秒，且 BER 阈值为 12%，而当前搜索的 BER 极其低（如 0.1%），则可将阈值 d_A 从 0.2 秒放松到如具有强置信度的 0.3 秒。可将其称为动态阈值。可由实施者使用概率分析确定在这样的动态阈值中的确切的调整量，以确保因这样的动态阈值导致的错误匹配的数量可以忽略不计。

如果在将三个搜索组合之后，结果不确定，可以执行更多搜索并组合所有结果，而不是放弃和声明剪辑不属于数据库。例如，在图 4 (b) 的情形中，可执行两次额外搜索。如果剪辑真正属于片 A，但第二个搜索不知何故返回错误时间位置，则通过两个额外搜索，这两个搜索中的一个很有可能还将会返回具有正确时间位置的片 A。然而，当组合 5 个搜索时，表决方案的定义变得更加复杂。在简化实施方式例子中，按照在以上伪代码中使用“min”和“argmin”操作符的方式，可以将时间位置方面最一致、且返回相同的片的两个搜索

(5个搜索中的)定义为胜出者。

从而,根据本发明的方面,可通过将来自多个搜索的结果进行组合,来改善指纹搜索精确度。优选是,根据本发明的其他方面,智能地将这样的结果进行组合。

在上述的本发明的方面中,采用基于搜索结果中的可用信息的表决类型来改善指纹搜索精确度。主要可用信息为搜索片(查询剪辑)的(预料的)标题。可在组合结果中实质上改善精确度的附属可用信息包括诸如 t_{offset} (如上所述)或 t_{distance} 、以及距离量度(如 BER)的参数。参数 t_{distance} 表示在查询片和所找到的片之间的时间位置的线性度:时间位置越接近直线(可由线性回归获得),对结果的匹配就越应该置信。在后面还结合图 6 进一步描述参数 t_{distance} 。

类似地,由汉明距离得出的 BER 表示所找到的片多么不同于查询片。BER 越高,它们彼此就越不同,而 BER 越低,它们彼此就较少不同而较多相似。所有其他的都相同时,相比于具有较高 BER 的匹配,更应该置信具有较低 BER 的匹配。

因此,附属信息的值表现出对其相应搜索结果的置信度。在此概括性描述的智能组合它们的一种方法是:在组合结果的期间使用附属信息的各单个值的置信级别作为权重。图 5 显示置信级别作为 BER 的函数的例子。可将该函数称为 $f_B(\text{BER})$ 。

尽管在先前例子中将 BER 映射到置信级别,但是该差异量度对本发明而言并不关键的,而是还可使用其他形式的差异量度。可将指纹视为数字的向量。在汉明距离和 BER 的情形中,其被视为二进制数的向量,即,二进制向量。这种向量的每一个维度仅为一位,不过可以想到二进制之外的数字的指纹向量,并且甚至对于不同的维度,每一个维度的大小可以不同,不过这应该可被预见到。从而,更普遍而言,差异量度可为这样的 L_p 范数度量,其中,获得在构成指纹的向量的每一个维度上的距离的绝对值的 p 次幂之和的 p 次根。BER 基于汉明距离,其为 L_1 范数差异量度(在每一个维度处的差异的绝对值之和,其中,每一个维度为 1 位)。还可为 L_2 范数差异量度

(在此获得差异的绝对值的平方之和的根)，不过当每一个维度为 1 位时，L2 范数类似于 L1 范数。还可存在自动相关的差异量度，其中，较高的值表示较接近的匹配。

类似地，可测量在数据库中潜在匹配查询指纹的时间位置距其预测（例如，通过线性回归）的预料的位置有多远。如图 6 所示，其中， $time_{query}$ 是查询指纹的时间位置，而 $time_{found}$ 是在所找到的片中潜在地匹配指纹的时间位置，在图 6 中将这两个值的对显示为菱形。然后，对这些菱形点执行线性回归。每一个菱形点距其最终线性回归预测（直线）的偏差被称为时间位置距离或 $t_{distance}$ ，如图 6 中竖线所示。然后，如图 7 所示，可将该距离映射到置信级别。我们可将该函数称为 $f_i(t_{distance})$ 。可采用任何合适类型的线性回归。在执行线性回归之前，或作为线性回归处理的一部分，可采用用于去除非正常值（outlier）的各种技术中的任一种。

如上所述， $t_{distance}$ 参数是 t_{offset} 参数的更普遍化的形式。使用 t_{offset} 参数的例子被示出在图 4a 和图 4b 中。它测量查询剪辑中的时间位置与匹配中相应时间位置之间的偏移。如果连续指纹匹配正确，则它应该大致是恒定的，但是当不存在速度提升或速度下降时不一定非要接近 0 值（因为查询材料的绝对时间不确定）。参数 $t_{distance}$ 更一般化，如图 6 所示。测量采样点（其 X 和 Y 坐标分别为查询剪辑及匹配项中的时间位置）与通过各种合适形式的线性回归中的任一种获得的直线之间的垂直距离。如果 $t_{distance}$ 参数对应于好的匹配，那么预料 $t_{distance}$ 参数接近于 0。关于 $t_{distance}$ 的想法在即便存在速度提升或速度下降时也起作用，因为它使用线性回归，且不像 t_{offset} 参数那样假设线性回归的斜率为 1。还应注意，由于 t_{offset} 参数假设斜率为 1，有可能对仅两个时间位置检查到 t_{offset} 的一致性。然而，当使用 $t_{distance}$ 时，由于不假设斜率为 1，需要至少三个时间位置以获得非平凡的（non-trivial）线性回归（虽然如果需要，可使用至少两个时间位置来获得线性回归，然后丢弃离斜率 1 太远的任何斜率）。

注意，在图 7 中，置信级别与图 5 中相比会更快速下降。这是

由于在典型实施方式中预期 BER 的容限 (tolerance) 可以相对地高, 而 t_{distance} 的任何非平凡量是结果不可靠得多的指示。此外, 由于两个点总生成完美地通过它们的线, 所以仅当存在三个或更多个点时, 才应该考虑使用基于 t_{distance} 的置信级别。

作为例子, 根据本发明的方面, 在以下伪代码中显示用于智能地组合置信级别和确定搜索结果的过程 (procedure)。

可将用于组合置信级别和确定搜索结果的示例性伪代码表示如下:

```

For (在结果中的独特的片的集合中的 X) {
  For (其中 search[i].piece==X 的 i) {
    Confidence1= $f_B$ (search[i].BER);
    If (其中 results[i].piece==X 的集合具有>=3 个点)
      Confidence2= $f_i$ (search[i].t_distance);
    Else
      Confidence2=0;
    Total_score[X]+=Confidence1 + Confidence2;
  }
  挑选 Total_score[X]最高的 X;
  If (Total_score[X]>阈值)
    返回“查询剪辑为片 X”;
  Else
    返回“未知查询剪辑”;
}

```

如伪代码中所示, 首先将 BER 和 t_{distance} 映射到它们相应的置信级别, 然后将它们组合。在该例子中, 组合简单地求和操作, 不过只要组合操作对于每一个置信级别为单调不减小的函数, 那么也可使用其他操作, 如相乘 (则置信级别必须是非负的)。如果 $f_B()$ 总是返回 1, $f_i()$ 总是返回 0, 且可变“阈值”具有值 $N/2$ (其中, N 为进行的

搜索的次数)，则将伪代码中表示的程序简化为简单多数表决方案。实际上，应该选择函数 $f_B()$ 和 $f_i()$ 以使得它们将最终输出正确的概率最大化。

具体实施方式

可以硬件或软件、或二者的组合（例如，可编程逻辑阵列）实现本发明。除非特别指出，包括在本发明中作为其一部分的方法或算法并不内在地与任何特定计算机或其他装置相关。特别是，通过根据此处给出的教导所写出的程序可使用多种通用机，或者它可更便于构成更专用的装置（例如，集成电路）以执行需要的方法步骤。从而，可以在一个或多个可编程计算机系统（每一个包括至少一个处理器、至少一个数据存储系统（包括易失性和非易失性存储器和/或存储元件）、至少一个输入设备或端口，以及至少一个输出设备或端口）上执行的一个或多个计算机程序实现本发明。将程序代码应用到输入数据，以执行此处所述的功能，并生成输出信息。按照已知的方式，将输出信息应用到一个或多个输出设备。

可以以任何需要的计算机语言（包括机器、汇编，或高级的面向过程、逻辑，或对象的编程语言）实现每一个这样的程序，以与计算机系统通信。在任何情形中，语言可为编译或解释性语言。

每一个这样的计算机程序最好存储在或下载到通用或专用可编程计算机可读的存储介质或设备（例如，固态存储器或介质，或者磁或光介质），用于在计算机系统读取存储介质或设备时配置和操作计算机，以执行在此所描述的过程。还可考虑将本发明的系统实现为计算机可读存储介质，其配置有计算机程序，如此配置的存储介质使得计算机系统按照专用和预定义的方式操作，以执行此处所描述的功能。

在此描述了本发明的多个实施例。不过，应该理解，在不偏离本发明的精神和范围的条件下，可进行多种修改。例如，上述某些步骤可以是不依赖于顺序的，从而可按照不同于所描述的顺序执行这些步骤。因此，其他实施例也在下面权利要求的范围内。

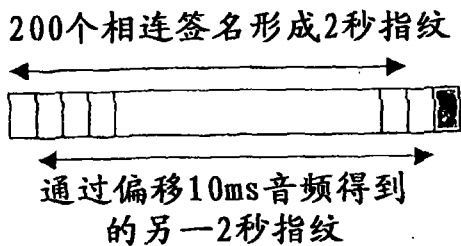


图1

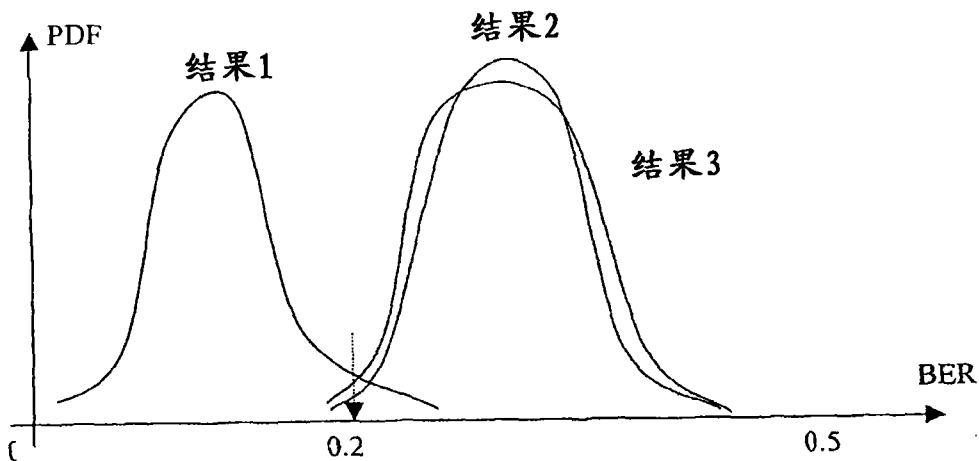


图2

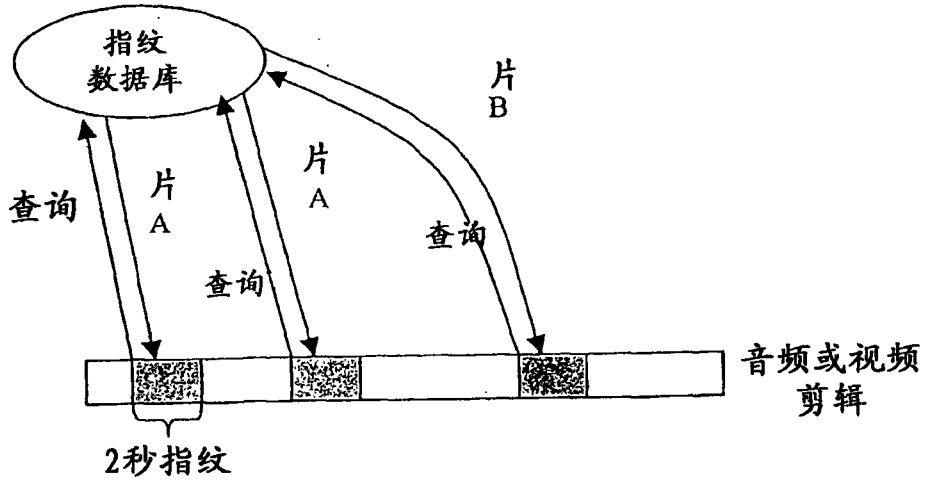


图 3

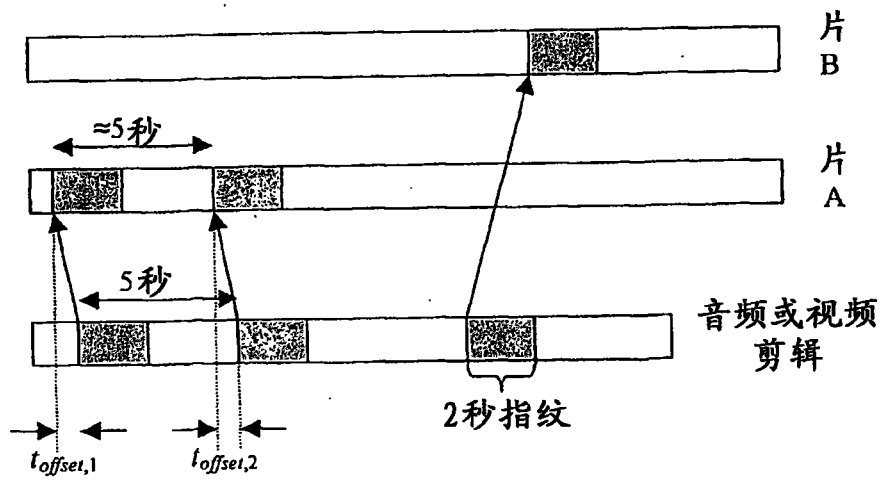


图 4a

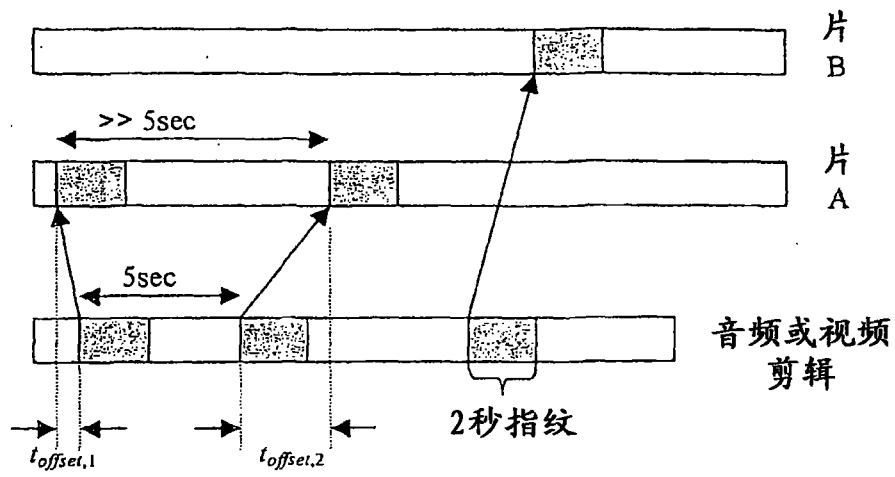


图4b

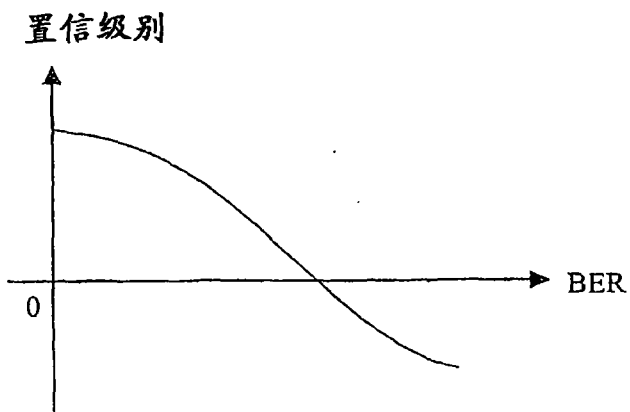


图5

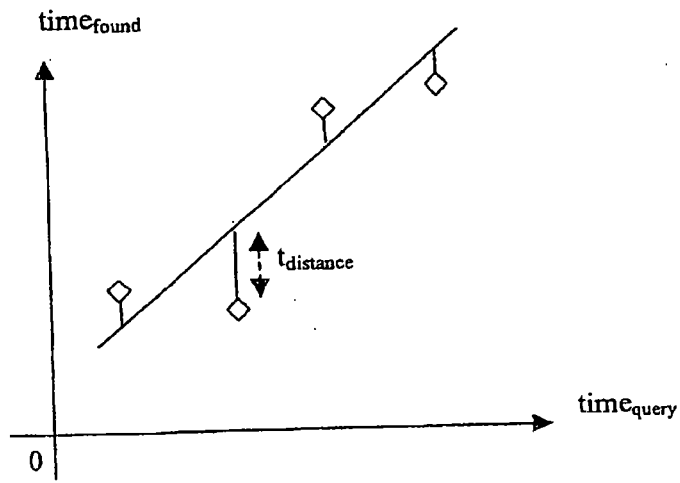


图6

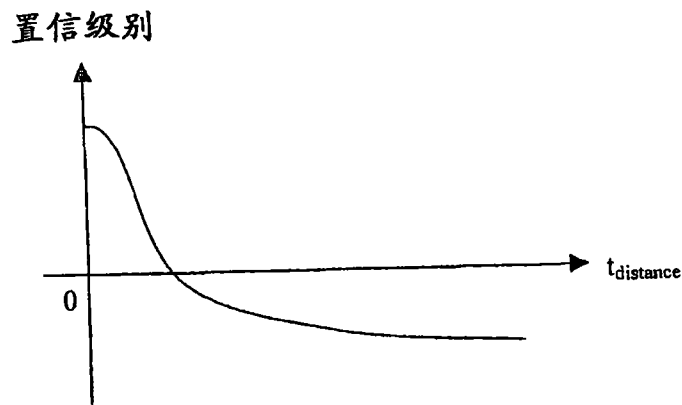


图7