基于熵的音频指纹检索技术研究与实现

王 伟[1] 陈志高[2] 孟宪凯[2] 李 伟[2]
(海军医学研究所 上海 200433)[1] (复旦大学计算机科学技术学院 上海 201203)[2]

摘 要 介绍了一种基于熵的音频指纹检索技术,该技术采用音频的熵特征作为音频的指纹特征(AFP),在检索中,该指纹特征可以用多种串匹配算法进行信息比对。实验采用最大公共字串(LCS)、编辑距离(Levenshtein Distance)和动态时间规整(DTW)算法实现指纹特征匹配,并采用一定数量的歌曲文件作为实验的测试集。每首歌曲都有一个带有不同的较大失真的音频文件或由不同歌唱家演唱的不同版本,这些带有不同的较大失真的音频文件由原曲经过不同的严重音频处理得到,比如添加噪声、加快速度、剪辑等。实验结果显示,使用的 3 种匹配算法均可以将训练集中所有的歌曲正确地识别出来,从而证明了基于熵的音频指纹检索技术具有准确性、鲁棒性、区分性等优良性质。

关键词 音频指纹,检索,熵,最大公共子串,编辑距离,动态时间规整
中图法分类号 TP391 **文献标识码** A

Research and Implementation of Identifying Music through Performances Using Entropy Based Audio-fingerprint

WANG Wei[1] CHEN Zhi-gao[2] MENG Xian-kai[2] LI Wei[2]
(The Naval Medical Research Institute, Shanghai 200433, China)[1]
(School of Computer Science and Technology, Fudan University, Shanghai 201203, China)[2]

Abstract A technology of identifying music using entropy based audio-fingerprint was introduced, which takes the music's character of entropy as audio-fingerprint. In the domain of music identifying, the above audio-fingerprint enables to use flexible string matching algorithms. We adopted longest common subsequence (LCS), levenshtein distance and dynamic time warping (DTW) as the matching algorithms of this audio-fingerprint, and used a number of music as the test set. Every music has another performance which is generated from the original one, most of the other performances have been artificially changed, such as to be noise-accessed, accelerated, cut and so on, and some of them may even be paired of same music played by different orchestras. The obtained results are impressive, in which all the performances in the collection can be correctly identified either with LCS, levenhtein distance or the dynamic time warping (DTW) distances, proving the veracity, robustness and good distinguish ability.

Keywords Audio-fingerprint, Identifying, Entropy, LCS, Levenshtein distance, DTW

1 引言

在音频检索领域,数字指纹的概念使得庞大的音频数据可以被较小的指纹数据指代,只需比较代表音频文件的通常较小的数字指纹,因此出现了基于数字指纹的音频检索技术。在基于数字指纹的音频检索技术中,任何一个音频片段都可以作为检索的依据,这类似于基于文本的检索中的关键词。由于基于任何一个音频文件都可以提取出无数的音频片段,相比于基于文本关键词的检索,基于数字指纹的检索无疑给人们带来了更大空间的检索依据。

音频指纹检索系统通常包括两个部分,即一个计算听觉特征的指纹提取算法和一个在指纹数据库中进行有效搜索的比对算法[1]。良好的指纹提取算法应该具备以下特点:准确性、鲁棒性、区分性、可靠性和较小的指纹尺寸。指纹提取算法基本可以分为语义特征和非语义特征两大类。语义特征指基于感知的(如流派基调等)具有明确含义的音频特征,此类特征易被人为获取,但包含的信息量很少,难以无二义地表征一段音频。非语义特征指基于物理的(如能量谱特性等)音频特征,这类指纹具有明确的数学形式,不易于被人耳感知,但易于编程实现。已知的指纹特征有傅立叶系数 FFT[2]、梅尔倒谱系数 Mel Frequency Cepstral Coefficients (MFFC)[3]、频谱平滑度 Spectral Flatness[4]等。通常使用分类器技术将这些特征映射为一个更简洁的表示,如隐马尔可夫模型(Hidden Markov Models, HMM)[5]或量化技术[6]。

本文将介绍一种新型的音频数字指纹技术,即基于谱熵的音频指纹。这种指纹特征是在音频信号每帧中提取并通过复杂的计算得出来的,具有良好的准确性、鲁棒性、区分性、可靠性。通过 Matlab 编程实现,分别使用最大公共子串(LCS)、编辑距离(Levenshtein distance)和动态时间规整(DTW) 3 种算法作为指纹的比对算法,并采用单声道的 wav

本文受基金项目:基于人声检测及分离的多版本流行音乐检索关键技术研究(NSFC61171128)资助。
王 伟(1976—),男,博士,副研究员,主要研究方向为多媒体信息处理,E-mail:wwang_fd@fudan.edu.cn;陈志高(1994—),男,硕士,主要研究方向为音频信息处理;孟宪凯(1986—),男,主要研究方向为音频信息处理;李 伟(1970—),男,博士,教授,主要研究方向为音频信息处理及多媒体信息安全,E-mail:weili-fudan@fudan.edu.cn。

音频文件作为测试集建立指纹数据库。通过对指纹数据库中的特征进行比对，实现对这种音频指纹系统性能的检验。

2 基于信息熵的音频指纹提取

2.1 信息熵

作为一种表示信号中所含信息量大小的值，熵也可以被称作混乱程度或不确定度，直观的理解就是，信息的不确定度越高，它的熵值也就越高，人们对该信息就越感兴趣，那么它所包含的信息量也就越大。熵值可以利用 Boltzman 公式来计算，对于一个随机事件实验 A，设它有 n 个可能的(独立的)结局 $a_1 a_2 a_3 \cdots a_n$；每一个结局出现的概率分别为 $p_1 p_2 p_3 \cdots p_n$，则熵值为[7]：

$$H(x)=E[I(p)]=\sum_{i=1}^{n} p_i I(p)=-\sum_{i=1}^{n} p_i \ln(p_i) \tag{1}$$

在语音信号分析领域，熵已经被用作区分一个片段是环境的噪声或需要的语音片段的一个重要标准[8-9]，但是到目前为止，将熵作为主要的音频特征来匹配音频文件的研究还相当少。近期已经有学者提出了直接在时域范围内将基于熵的音频指纹用于判断音频信号的信息[10]，不久后又有研究者提出了一种谱熵指纹算法，这种指纹算法对噪声、失真等更具鲁棒性[11]。

2.2 基于熵的音频指纹提取算法

提取音频的熵指纹的过程如下：首先，以 1.5s 为一帧，相邻两个帧之间存在 1/2 的重叠；其次，对每一帧进行汉宁加窗操作，再对加窗后的数据进行快速傅立叶变换(FFT)，使得音频信号由时域转换到频域；然后，采用由 Zwicker 提出的将人易于感知的声频范围 20Hz～15.5kHz 分为 24 个临界频带的划分方法[12]，将结果划分为 24 个临界频带；最后，分别对每个临界频带中的数据利用式(2)计算求出熵值。

$$H=\ln(2\pi e)+\frac{1}{2}\ln(\sigma_{xx}\sigma_{yy}-\sigma_{xy}^2) \tag{2}$$

其中，σ_{xx} 和 σ_{yy} 分别是每个频带中样本点实部和虚部的方差，也可以记为 σ_x^2 和 σ_y^2；σ_{xy} 表示每个频带中样本点实部和虚部的协方差，也可以记为 σ_{yx}。

在对每个帧中 1～24 个临界频带分别求出熵值之后，可以得到一个熵矩阵，用 $H(n,b)$ 表示第 n 个帧的第 b 个频带的熵，并构造出一个包含了音频的熵信息图表，熵信息图表可以同时显示出熵信息在每一帧中不同的临界频带上的分布。图 1 和图 2 分别是用 Matlab 绘制的原版音频文件和加了较大失真处理后的音频文件的熵信息图。

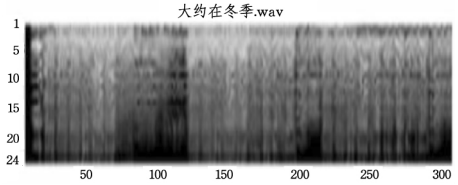

图 1 《大约在冬季》的原版音频文件的熵信息图

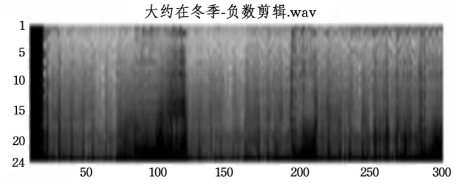

图 2 《大约在冬季》较大失真处理后的音频文件的熵信息图

以上两幅熵信息分布图中横坐标代表帧，纵坐标代表 24 个不同的频带，颜色越亮表示熵值越大，颜色越暗表示熵值越小。由于直接得到的熵信息值的范围较大，不利于直接用作指纹来进行比对，因此通过对上面得到的熵矩阵进行深度提取得到特征矩阵 B，公式如下：

$$B(n,b)=\begin{cases}1, & \forall\, H(n,b)-H(n-1,b)>0\\ 0, & \forall\, H(n,b)-H(n-1,b)\leqslant 0\end{cases} \tag{3}$$

其中，$H(n,b)$ 仍表示第 n 个帧的第 b 个频带的熵，这种计算方法类似于分别对各个频带求音频信号随着帧的变化而引起的熵值的增减性。图 3 和图 4 分别示出了图 1 和图 2 中的两首歌曲的熵谱经过深度提取特征处理后得到的图谱。

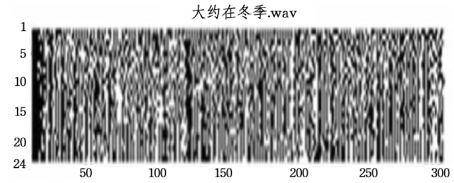

图 3 原始音频文件深度提取的熵指纹图谱

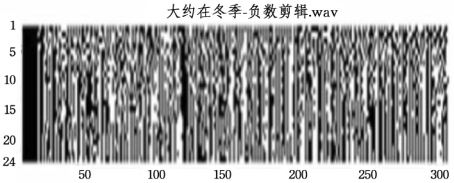

图 4 失真处理音频文件深度提取的熵指纹图谱

图 3 和图 4 中白色代表 1，黑色代表 0，将熵谱经过归一化后得到的图谱就是从音频文件中提取出来的指纹特征。

3 特征比对算法

3.1 汉明距离

汉明距离被定义为两个等长字符串之间对应位置的不同字符的个数。汉明距离本身也可以作为一种比对算法，但因其具有只能比较两个等长的串的局限性，本文不能直接使用。

3.2 DTW 算法

在指纹特征比对的过程中，由于语音信号具有极大的随机性，两个音频片段在检索中几乎不可能有完全相同的长度，因此相应的时间归正处理相当必要。动态时间规整(Dynamic Time Warping)就是一种把时间归正技术和距离测度计算结合起来的非线性的归正技术，并且已经很成功地被应用到了语音识别等研究领域。

DTW 算法采用动态规划技术(Dynamic Programming)，将复杂的全局最优化问题逐步转化为多个局部最优化问题。假设有两个特征向量 $R=\{r_1, r_2, r_3, \cdots, r_N\}$ 和 $T=\{t_1, t_2, t_3, \cdots, t_M\}$ 且 $M\neq N$，动态时间规整的过程可以看作是逐步寻找最佳的时间归正函数，使得特征向量 R 的时间轴 j 非线性地映射到特征向量 T 的时间轴 i 上，使总的累计失真量最小。

DTW 算法通过局部最优化的方式求出加权距离的最小总和[13]，即

$$D=\min\frac{\sum_{n=1}^{N}[d(R_{i(n)},T_{j(n)})\cdot Weight_n]}{\sum_{n=1}^{N} Weight_n} \tag{4}$$

其中，$d(R_{i(n)},T_{j(n)})$ 表示相应的匹配点对的局部距离，$Weight_n$ 表示加权函数，主要针对 $I\neq J$ 的路径进行惩罚。同时需要对其作出限制以保证匹配路径不违背时间顺序，约束如下：

1）**单调性**：$i(n)\geqslant i(n-1)$ **且** $j(n)\geqslant j(n-1)$。

2）**起始点约束**：$i(1)=j(1)=1$；$i(N)=I$，$j(N)=J$。

3）**连续性**：**一般规定不允许跳过任何点，即** $i(n)-i(n-1)\leqslant 1$ **和** $j(n)-j(n-1)\leqslant 1$。

4）**最大归正量的极限**：**最简单的情形为** $|i(n)-j(n)|<M$，**其中称** M **为窗宽**。

本文用 DTW **实现矩阵之间的比对，因此需要对矩阵间的匹配距离作出明确的定义。本文采用的是平均汉明距离：**

$$D(X,Y)=\frac{1}{K}\sum_{i=1}^{K}hamming\_distance(x_i,y_i) \tag{5}$$

利用动态规划算法来实现动态时间规整：

$$\begin{aligned}&D(i,0)=\sum_{k=0}^{i}d(i,0)\\&D(0,j)=\sum_{k=0}^{j}d(0,j)\\&D(i,j)=\min\begin{cases}D(i-1,j-1)+2\cdot d(i,j)\\D(i-1,j)+d(i,j)\\D(i,j-1)+d(i,j)\end{cases}\end{aligned} \tag{6}$$

最终得到的 $D(N,M)$**即为参考特征矩阵** $R(N\times K)$**和目标特征矩阵** $T(M\times K)$**的动态时间规整开销。**

3.3　编辑距离

求两个串之间的编辑距离即为找到将原串 s **转换到目标串** t **所需要的最少操作步骤，这些操作可以为插入、删除和替换，在某些时候也可以为调换原串中两个元素的位置。通常将每种操作的代价设为** 1，**但在某些特定情况下，对不同的操作定义不同的操作代价。如果将从原串** s **到目标串** t **的操作限定为替换，且操作代价为** 1，**那么对编辑距离的求解就退化成了求** s **和** t **的汉明距离；如果限定为插入和删除两种操作，且每种操作的代价都是** 1，**那么求解编辑距离问题就转化为求解最长公共子串**(LCS)**问题。**

假定原串 s **的长度为** M，**目标串** t **的长度为** N，**算法仅适用插入、删除和替换，且每种操作的代价都是** 1，**求解编辑距离的过程可以按照如下的算法进行：**

$$\begin{aligned}&C_{i,0}=i,\ \forall\ 0\leqslant i\leqslant N\\&C_{0,j}=j,\ \forall\ 0\leqslant j\leqslant M\\&C_{i,j}=\begin{cases}C_{i-1,j-1}, & t_i=s_j\\\min[C_{i-1,j-1},C_{i-1,j}]+1, & t_i\neq s_j\end{cases}\end{aligned} \tag{7}$$

实际对编辑距离的求解过程中，并不需要存储动态规划生成的全部的表，仅仅需要一行的存储量通过以下的算法来实现。

$$\begin{aligned}&C_i=i\\&C_i'=\begin{cases}C_{i-1}, & t_i=s_j\\\min[C_{i-1},C'_{i-1},C_i]+1, & t_i\neq s_j\end{cases}\end{aligned} \tag{8}$$

其中，C' **表示由上一次存储的** C **数组计算出来的新的** C **数组。**

3.4　最大公共子串

最大公共子串可以看作是对编辑距离问题加以限定，即限定从原串 s **到目标串** t **的操作只能是插入和删除两种操作，且每种操作的代价都是** 1。**由于添加了上述限定，因此需要修改求解编辑距离的算法，在求最大公共子串时，使用如下的递推公式：**

$$\begin{aligned}&C_{i,0}=i,\ \forall\ 0\leqslant i\leqslant N\\&C_{0,j}=j,\ \forall\ 0\leqslant j\leqslant M\\&C_{i,j}=\begin{cases}C_{i-1,j-1}, & t_i=s_j\\\min[C_{i,j-1},C_{i-1,j}]+1, & t_i\neq s_j\end{cases}\end{aligned} \tag{9}$$

以上讨论的求解编辑距离和最大公共子串的算法仍然只能够针对一维的串，而本文却要用它来实现矩阵的比对，因此需要对算法做出一定的修改，这一点将在后面比对算法的实现部分加以说明。

4　基于熵的音频指纹检索系统的实现

本文实验采用 Matlab7.5 **作为编程工具，基于** Matlab **语言完成程序的编写，并利用** goldwave4.26 **和** cooledit pro2.0 **作为预处理实验所需的音频文件的辅助软件。**

4.1　实验流程

实验的主要目标是测试前文所描述的基于熵谱的指纹特征分别在最大公共子串(LCS)**、编辑距离**((Levenshtein distance)**和动态时间规整**(DTW) 3 **种比对算法下能够达到怎样的效果。**

实验的总体设计流程如图 5 **所示，其中比对结果指在训练数据内部比对的结果，检索指用测试数据进行比对的结果。**

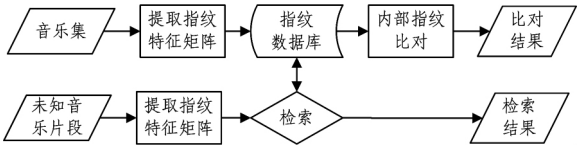

图 5　**实验总体流程图**

4.2　熵指纹提取过程

熵指纹的提取过程如图 6 **所示。**

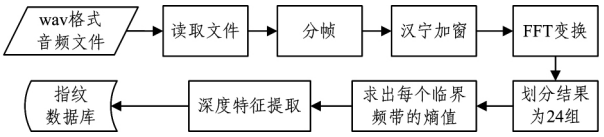

图 6　**指纹提取过程流程图**

4.3　比对算法的实现

对涉及的 3 **种比对算法做了一定的改进，在此之前需要分析作为比对内容的特征矩阵的特点。由上文介绍的指纹提取流程可知，本实验将音频在频域上分为** 24 **个临界频带，并对每帧的每个频带分别求熵值，最后提取的特征指纹也是在** 24 **个临界频带上分别计算得来的。结合指纹提取算法可知，作为比对内容的特征矩阵是二维的，第一维是音乐的帧数，第二维是临界频带数，整个矩阵由** 0 **或** 1 **组成。后文将在此基础上提出用到的最大公共子串**(LCS)**、编辑距离**(Levenshtein distance)**和动态时间规整**(DTW) 3 **种比对算法的改进方案。**

4.3.1　最大公共子串(LCS)算法的改进

最大公共子串算法的目的是找出两个串的最长子串长度，在处理音频特征的操作上，这种比对可以理解为寻求两个音频片段中同时出现过的声学特征的最大个数。在实现过程中，需要比对谱熵音频指纹矩阵，其不同于传统的一维串，因此需要对算法进行改进。假设需要比对的是参考特征矩阵 $R(N\times K)$**和目标特征矩阵** $T(M\times K)$，**按照** LCS **的要求，需要定义** K **位串的相等条件。本文中** $K=24$，**这个二进制串可能出现的情况有** 2^{24} **种，以完全相等作为判断依据显然是不现实的，因此本文采用折中的方法，将二进制串之间的汉明距离大于** 7 **作为判断不等的依据。**

这里不对 LCS **算法做大的改动，仍以动态规划的方法对其进行求解。**

4.3.2 编辑距离(Levenshtein distance)算法的改进

同理,用一个位上的不同区来判定整个 24 位串不同是不可取的,而如果按照 LCS 的方法来处理又失去了距离的特性。因此在实际应用中采用了更改操作代价的方法。本文规定在编辑距离算法中插入和删除的代价仍然是 1,而对于两个 K 位的串 X 和 Y,它们之间的修改代价 $d(X,Y)=hamming(X,Y)\ /\ K$。

基于以上的规定,将编辑距离算法改进为:

$$C_{i,j}=\min\begin{cases}C_{i-1,j-1}+d(t_i,s_j)\\C_{i,j-1}+1\\C_{i-1,j+1}+1\end{cases}\tag{10}$$

4.3.3 动态时间规整(DTW)算法的改进

在关于动态时间规整算法的讨论中,在匹配距离的度量上引入了平均汉明距离的度量方法:

$$D(X,Y)=\frac{1}{K}\sum_{i=1}^{K}hamming\_distance(x_i,y_i)\tag{11}$$

这种方法在 DTW 算法的实际应用中仍然可以继续使用。

4.3.4 距离的归一化

以上 3 种算法求得的值在某种程度上都可以理解为距离,由于这些距离会因特征矩阵的大小不同而对结果造成严重的影响,以编辑距离为例:"010"和"021"之间的编辑距离为 2,"0100000000"和"0210000000"之间的编辑距离也为 2,但后者的相似度显然是高于前者的,因此将距离的测度作为评判两个特征指纹相似度的指标时,需要对结果进行归一化。

在本实验中,对于参考特征矩阵 $R(N\times K)$和目标特征矩阵 $T(M\times K)$,在使用编辑距离进行比对时,结果永远不会大于 $\max(N,M)$;在使用最大公共子串进行比对时,结果永远不会大于 $N+M$;在使用动态时间规整作为比对算法时,结果永远不会大于 $N+M$。因此本文规定归一化的算法如下:

编辑距离归一化:$\dfrac{\text{编辑距离}}{\max(N,M)}$

最大公共子串归一化:$\dfrac{\text{最大公共子串长度}}{N+M}$

动态时间规整归一化:$\dfrac{\text{动态时间规整结果}}{N+M}$

5 实验结果与分析

5.1 测试集的选取和处理

在测试阶段之前,需要先建立音频指纹数据库。为便于实现且消除无关因素的影响,实验中的全部音乐集都是由普通的 mp3 格式音乐经过 cooledit pro2.0 处理得到的单声道的 wav 格式音频文件组成的,并且这些歌曲都有一个在原版本上加了较大失真的文件或另一版本。失真是经过 goldwave4.26 的一系列不同的音频处理(比如添加噪声、加快速度、剪辑等)完成的。本文共选取 13 对音频文件作为指纹数据库的训练集,分别使用了最大公共子串(LCS)、编辑距离(Levenshtein distance)和动态时间规整(DTW) 3 种算法对其中任意两对进行比对,实验结果就是相互比对所产生的归一化距离矩阵。

表 1 列出了本实验中指纹数据库采用的音频文件,这些音频文件被成对采用以证明基于熵的音频指纹检索具有较好的鲁棒性和区分性。其中第 1—9 对是通过用 goldwave4.26 软件人为地进行较为夸张的处理而得到的,而第 10—13 对是不同歌手所演唱的不同版本。实验结果表明,利用基于熵的音频指纹能够较好地识别出经过处理得到的版本,对不同歌唱家演唱的不同版本歌曲也能够做出正确的识别。

表 1 指纹数据库训练集的音频文件信息

音频文件名	时间长度	不同版本的音频文件名	时间长度
1 天使.wav	03:00	1_天使_嘘声噪音.wav	04:26
2 爱就一个字.wav	03:23	2_爱就一个字_失真.wav	03:23
3 大约在冬季.wav	03:49	3_大约在冬季_负数剪辑.wav	03:49
4 后来.wav	03:00	4_后来_回声.wav	03:10
5 过完冬季.wav	03:43	5_过完冬季_奏鸣.wav	03:42
6 练习.wav	02:53	6_练习_翻边.wav	02:44
7 水手.wav	03:03	7_水手_机械.wav	03:03
8 天黑黑.wav	02:56	8_天黑黑_时间.wav	02:28
9 一笑而过.wav	03:12	9_一笑而过_加速.wav	02:55
10 笑红尘_国语.wav	04:00	10 笑红尘_粤语.wav	04:10
11 张学友_一路上有你.wav	02:56	11 永邦_一路上有你.wav	02:28
12 许慧欣_七月七日晴.wav	03:12	12 香香_七月七日晴.wav	02:55
13 香香_我不是黄蓉.wav	04:00	13 王蓉_我不是黄蓉.wav	04:10

5.2 指纹数据库比对实验结果

本文使用灰度矩阵图像来表示经比对得到的相似度图示,矩阵的每一个点表示该点的横坐标和纵坐标所代表的音频文件指纹特征的相似程度,图中越暗的区域表示相似度越高,越亮的区域表示相似度越低(图 7—图 9)。该灰度矩阵的横坐标和纵坐标所代表的音频文件都是按照上面指纹数据库的音频文件信息表中的顺序依次排列的,并且为了便于观察,使同一首音乐的不同版本相邻,如'1'代表的音频文件是'1 天使.wav','1''代表'1'的不同版本。

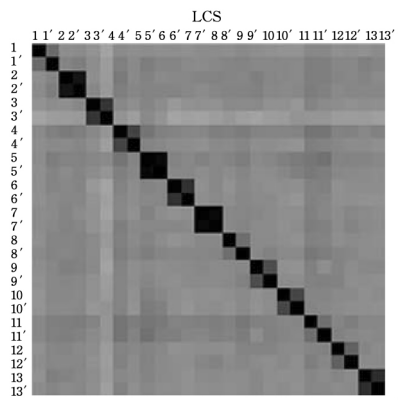

图 7 使用最大公共子串(LCS)算法得到的实验结果

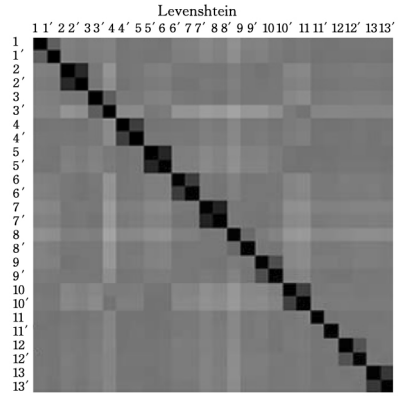

图 8 使用编辑距离(Levenshtein distance)算法得到的实验结果

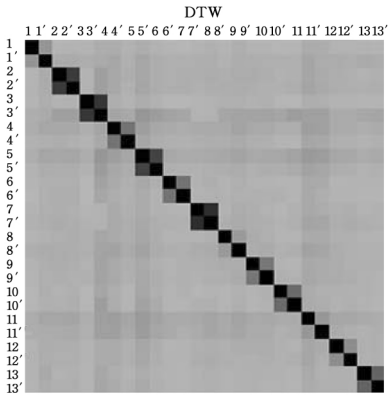

图 9　使用动态时间规整(DTW)算法得到的实验结果

通过观察可以发现，灰度矩阵图像中主对角线上的点都是最黑的，即自身与自身的相似度最高。事实上，3 种算法得到的自身比对距离都是 0。

从 3 个方面来分析实验结果：1)3 种算法的主对角线上的点的灰度明显暗于其余的点，可以明显地区分出完全不同的音乐。由此可以得到，动态规整最优，最大公共子串次之，编辑距离最差。这体现了算法的准确性和可靠性。2)3 种算法都可以将一首音乐的不同版本或严重音频处理后的音乐与完全不同的音乐明显地区分开，原音乐和其不同版本的相似度是比较高的。这体现了算法的区分性。3)同一首音乐不同版本之间也可以被识别出来，这体现了算法的鲁棒性。

5.3　未知数据比对实验结果

从上一节的结果分析中可以看出，基于谱熵的音频指纹能够成功地应用到音频文件的匹配中。后续实验将会对其在检索技术中的应用加以模拟实现，并分析其结果。

本文选用指纹库中原有的 3 个音频文件剪辑和指纹库中没有的 3 个音频文件剪辑，并将每个音频文件剪辑的时间长度定为 3 分钟。表 2 中，前三首音频文件取于指纹库，后三首未在指纹库中出现过。

表 2　检索实验测试集中音频文件信息表

音频文件名	时间长度
1 爱就一个字_检索. wav	03：00
2 大约在冬季_检索. wav	03：00
3 水手_检索. wav	03：00
1 爱过你_检索. wav	03：00
2 真爱_检索. wav	03：00
3 花心_检索. wav	03：00

图 10—图 15 是每个音频文件在音频指纹数据库中的匹配情况。测试结果分析如下。

(1)如果可以在指纹库中检测到未知音乐片段，匹配结果必然成对出现，因为指纹库中音乐片段都是成对出现的，两个版本的音乐都会与检测的音乐存在较大相似度。

(2)如果不能在指纹库中检测到未知音乐片段，那么检索结果中各区域就会表现得很平均，不存在突出的点，普遍偏亮。

基于以上结果的分析，实际应用中可以采用如下方法判断待检索的未知音乐片段是否存在于指纹库中，并对其明确定位。1)设定阈值，相似度高于此阈值的匹配点即为检索结果。这种方法的优点是简单明了，易于实现；缺点是阈值的选取较为困难，且对每种比对算法需要选择不同的阈值。该结论可以通过在上面的结果图得到印证，使用编辑距离算法得到的实验结果亮度普遍暗于使用动态时间规整算法得到的实验结果。2)比较检索过程中出现的最大相似度和整个比对中的平均相似度，如果最大相似度比平均相似度高出一个阈值或百分比，则认为最大相似度对应的点是检索到的音频文件，返回检索结果。这种方法在应用中的效果优于前面提到的单一阈值的方法，且减弱了音频文件长度导致的相似度变化的影响。

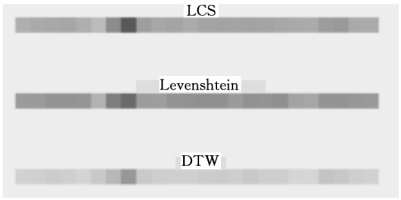

图 10　《1 爱就一个字_检索》的匹配结果

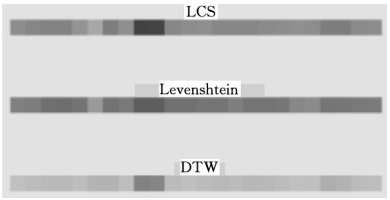

图 11　《2 大约在冬季_检索》的匹配结果

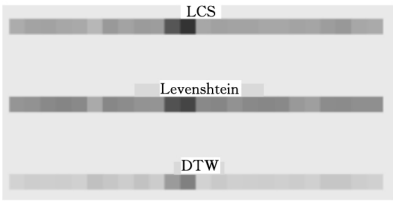

图 12　《3 水手_检索》的匹配结果

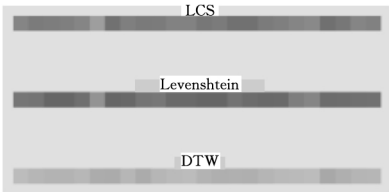

图 13　《1 爱过你_检索》在指纹数据库中的匹配结果

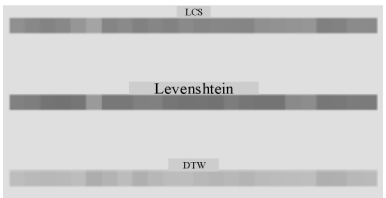

图 14　《2 真爱_检索》在指纹数据库中的匹配结果

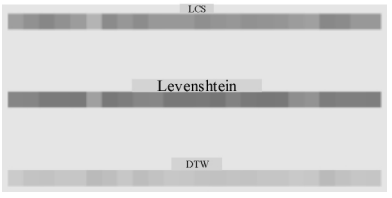

图 15　《3 花心_检索》在指纹数据库中的匹配结果

结束语　基于熵的音频指纹检索在准确性、鲁棒性、区分性和可靠性上取得了较好的效果，但也存在着一定的不足。

1)实验中遇到过如下情况,长音频片段总是难以匹配,远远短于它的音频片段,即使这个短的音频片段是长音频片段的一部分。这种检索在实际应用中确实有需求,然而由于比对算法的限制,本实验还无法满足这一要求。即便如此,本实验中采用的 3 种比对算法在匹配时间差异为一分钟以内的音频信息时也已足够。2)音频文件的指纹尺寸仍然偏大,如一首 3 分钟的音乐将被分为(3×60)/0.75−1=239 个帧,每个帧又分为 24 个频带,因此该音乐的指纹为一个 239×24 的矩阵,对如此庞大的指纹信息进行比对是低效的,不能满足在海量信息中得到实时检索结果的要求,该算法在这一方面还需进一步的改进。

参 考 文 献

[1] 李伟,李晓强,陈芳,等.数字音频指纹技术综述[J].小型微型计算机系统,2008,29(11):2124-2130.

[2] FRAGOULIS D,ROUSOPOULOS G,PANAGOPOULOS T,et al. On the automated recognition of seriously distorted musical recordings [J]. IEEE Transactions on Signal Processing,2001,49(4):898-908.

[3] LOGAN B. Mel Frequency Cepstral Coefficients for Music Modeling[C]// International Symposium on Music Information Retrieval. 2000.

[4] ALLAMANCHE E,HERRE J,HELMUTH O,et al. Audioid:Towards content-based identification of audio material [C]// Proc Aes Convention. 2001.

[5] NEUSCHMIED H,MAYER H,BATLLE E. Content-based identification of audio titles on the Internet[C]// First International Conference on Web Delivering of Music,2001. IEEE,2001:96-100.

[7] SHANNON C E. The mathematical theory of communication. [J]. The Quarterly Review of Biology,1951,60(26,3):306.

[8] SHEN J L,HUNG J W,LEE L S. Robust entropy-based endpoint detection for speech recognition in noisy environments[C]// Proc. Int. Conf. on Spoken Lang. Processing(ICSLP-98). 1998.

[9] YOU H,ZHU Q,ALWAN A. Entropy-based variable frame rate analysis of speech signals and its application to ASR[C]// IEEE International Conference on Acoustics,Speech,and Signal Processing,2004(ICASSP'04). IEEE,2004.

[10] IBARROLA A C,CHAVEZ E. A Robust Entropy-Based Audio-Fingerprint. [C]// 2012 IEEE International Conference on Multimedia and Expo. IEEE,2006:1729-1732.

[11] CAMARENA-IBARROLA A,CHÁVEZ E. Identifying Music by Performances Using an Entropy Based Audio-Fingerprint [OL]. http:/ict. udlap. mx/people/alfredo/tesis/MusicMatch DTW. pdf.

[12] 陈小平,胡泽.听觉临界频带及其在声频信号处理中的应用[J].中国传媒大学学报(自然科学版),2004,11(2):28-35.

[13] 王红.基于 DTW 和变帧率算法的广播语音关键词识别[D].昆明:云南大学,2007.

(上接第 525 页)

```
        border=4,
    )
    qr.add_data('http://…/detail.jsp?id=%s'%(s))
    qr.make(fit=True)
    img=qr.make_image()
    img.save('%s.png'%(s))
%>
```

结束语 运用手机二维码识别技术推进校园教学设备智能化管理进程,提高高校教学设备的管理效率,是持续研究的方向和目标。在手机二维码技术和数据库的支撑下,设计了基于二维码的高校教学设备管理系统。通过调阅数据库及时更新的信息,可以准确地获得设备的工作状态和存放地址,清楚地了解设备运行状态。该系统在我单位的部分教学设备管理中进行了试运行,使用效果良好,提高了设备的管理效率和质量;待对其进行进一步试用和完善后,期望为学校设备的智能化管理做出贡献。

参 考 文 献

[1] 白广梅,赵靖强.论高校固定资产清查与管理的加强与创新[J].实验科学与技术,2014,12(2):197-199.

[2] 钱鹏.二维码技术在高校多媒体教学中的应用[J].实验室研究与探索,2014,33(4):255-259.

[3] 林超.手机二维码在多媒体教室设备管理中的应用[J].计算机与现代化,2014(10):55-57.

[4] 王玉平,王文君.高校仪器设备档案信息资源共享平台的建设与管理[J].实验技术与管理,2015,32(9):247-249.

[5] 顾昕元,高磊.二维码在医疗设备管理和维护中的应用[J].中国医疗设备,2014,29(10):66-68.

[6] 王文俊,殷曦敏.手机二维码识别技术在大型仪器设备管理中的应用[J].实验室研究与探索,2015,34(5):278-281.

[7] 吴丹.基于手机二维码的高校实验室设备管理模式探讨[J].科教文汇,2014(3):77-81.

[8] 孙永和.手机二维码在医疗设备信息管理中的应用探索[J].医疗设备,2015,19(2):76-77.

[9] 钱鹏.二维码技术在高校多媒体教学中的应用[J].实验室研究与探索,2014,33(4):255-259.

[10] 戴军.基于二维码技术的装备管理系统的研究[C]// 2013 中国粮油测控技术研讨会论文集. 2013:37-38.

[11] 刘仁霖,钱大益,孟兆磊,等.高校仪器设备信息化管理系统的设计与实现[J].实验室研究与探索,2015,34(9):281-284.

[12] 赵一.基于二维码技术的健康证管理信息系统的设计与实现[D].西安:西北大学软件工程,2015.

[13] 庆民,闫鹏,雷洞婷飞,等.手机二维码技术在高校特种设备安全管理中的作用[J].安全科学技术,2014(6):7-10.

[14] 牟金进.基于手机平台的二维码物品信息管理系统的设计与实现[D].北京:北京交通大学,2012.

[15] 梁子乐.二维码技术在客户关系管理系统的应用研究[D].青岛:中国海洋大学,2013.

[16] 刘志成.JSP 程序设计实例教程[M].北京:清华大学出版社,2009.

[17] 张梦.基于 C/S 结构的中小企业人事管理系统的设计与开发[J].计算机科学,2016,43(6A):547-550.

[18] 孔陶茹.剖析二维码技术的访客管理系统设计与实现[J].科技风,2015(21):9-10.