◎数据库、信号与信息处理◎

元音检测和最值点符号特征匹配的音乐检索

张 燕[1,2],钱 博[2],李燕萍[3]
ZHANG Yan[1,2],QIAN Bo[2],LI Yan-ping[3]

1.南京理工大学 计算机科学与技术学院,南京 210094
2.金陵科技学院 信息技术学院,南京 211169
3.南京邮电大学 通信与信息工程学院,南京 210003
1.Department of Computer Science & Technology,Nanjing University of Science and Technology,Nanjing 210094,China
2.Information Technology School of Jinling Institute of Technology,Nanjing 211169,China
3.College of Telecommunication & Information Engineering,Nanjing University of Posts and Telecommunications,Nanjing 210003,China
E-mail:sandson6@163.com

ZHANG Yan,QIAN Bo,LI Yan-ping.Music retrieval based on vowel detection and symbol feature representation. Computer Engineering and Applications,2009,45(36):126-128.

Abstract: In query-by-humming systems,uncertainty can arise in query formulation due to user-dependent variability and machine-based errors.The paper proposes a new system,which using vowel detection for note segmentation.After that,the segmented humming audio is transcribed to symbols with pitch and duration information.At last,the transcribed audio is compared with the database and finds the closest melodic fragments.Experiment shows the retrieval accuracy is higher than 70% of best candidate with faster retrieval speed than traditional system,which achieves the requirement of practice application.

Key words: audio retrieval;music retrieval;vowel detection;feature representation with symbol

摘 要:哼唱的随意性和音乐特征提取算法误差都会影响基于哼唱的音乐检索系统的性能。针对上述问题,利用元音帧检测获得较为精确的音符边界,实现音符分割;对分割后的音符提取相对音高和音长,实现符号描述;最后将哼唱片段中音高和音长最值点周围的符号描述作为特征与数据库中的数据进行匹配,得到最相似的候选音乐。实验表明该方法对未经训练的哼唱者的首位匹配正确率达到70%以上,匹配速度也大大优于传统方法,检索性能基本达到了实际应用的需求。

关键词:音频检索;音乐检索;元音检测;符号特征描述
DOI:10.3778/j.issn.1002-8331.2009.36.037 **文章编号:**1002-8331(2009)36-0126-03 **文献标识码:**A **中图分类号:**TP391

1 引言

音乐逐渐成为人们生活中不可或缺的部分,然而面对互联网上数以万亿计的音乐数据,如果不知道名称、歌手、歌词等信息,就无法实现基于文本查询。由于受到主观因素的影响,只能对音乐风格进行抽象描述,无法进行符号化和量化处理。因此,基于内容的音乐检索技术越来越受到重视:用户输入音乐片段(甚至非专业的哼唱片断),系统就可以自动提取旋律特征,并从音乐数据库中查询出类似的音乐数据。

一般来说,解决基于哼唱的音乐检索问题需要将哼唱音频和原始音乐数据表达成相同的模式,并在此基础上实现搜索和匹配。经典的哼唱检索系统采用音符级匹配策略[1-3],这些算法采用自相关基音频率跟踪算法和能量跟踪算法检测音符边界,利用音高音长的包络描述特征。动态规划技术(特别是动态时间规整算法(DTW)及其改进算法)被广泛应用于各种音乐检索系统中[4-6]。Pardo 等人采用 HMM 和 CHMM 等序列统计模型来描述音频元素[7],系统对音频数据产生一组序列统计模型,通过计算哼唱数据特征与各个模型的似然值判定匹配程度。这种方法的检索精度很高,但是计算量很大。

该文延续了上述算法思想,并针对其检索速度慢的问题进行了分析和改进:首先根据哼唱习惯(或输入要求),利用元音检测技术对音符进行分割, 这个步骤需要较高的处理精度,因此附加了后处理模块进行修正;其次将分割好的音符进行基音频率检测,获得音符的音高和音长,通过计算相邻音符的相对音高和相对音长实现符号描述;最后利用符号化描述中最大值和最小值周围的相对音高音长特征作为哼唱音频的特征与数据库中的候选项进行匹配。数据库中的原始音乐采用 MIDI 格式,因此可以通过符号处理得到精确的相对音高音长描述。匹配结果将列出相似度最高的前 n 个候选项供用户参考。实验数

基金项目:江苏省现代教育技术研究所课题(the Issue of Jiangsu Institute of Modern Educational Technology No.2007-R-4704)。
作者简介:张燕(1970-),女,博士,副教授,主要研究领域为音频处理,音乐检索;钱博(1981-),男,博士,讲师,主要研究领域为音频检索,语音处理;李燕萍(1983-),女,博士,讲师,主要研究领域为语音处理,语音合成。
收稿日期:2009-08-20 **修回日期:**2009-09-25

据表明，该系统的首位匹配正确率达到了 70%以上，检索速度大大加快。当数据库规模不大时，基本能达到用户查询的精度和速度需求。

2　音频检索原理

2.1　基于哼唱的音频检索系统结构

该文基于哼唱的音频检索系统如图 1 所示。

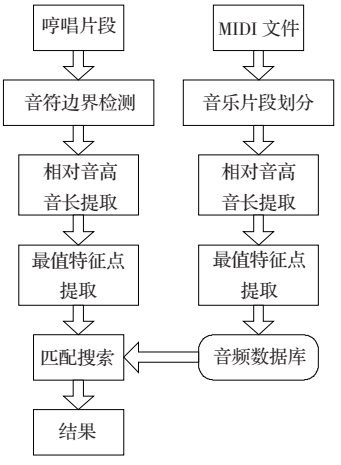


图 1　基于哼唱的音频检索系统示意图

数据库建设过程中，首先对大量的 MIDI 文件提取旋律音轨，并根据音乐旋律特征的变化对音乐数据进行分段。其次，提取每个音乐片段的相对音高和音长，实现符号化描述。选取符号描述中的最值点及其周边的符号描述作为该音乐段的特征存入音乐数据库中。为了加快检索速度，还可以建立索引或引入特殊存储结构，篇幅所限，该文未涉及相关技术。

用户通过话筒录入一段哼唱片断启动检索过程。系统要求用户都以“嗯”、“嗒”、“啦”等简单的发音进行旋律的描述，主要通过元音部分携带音高音长信息，利用元音检测技术可以提取音符边界，具体算法将在 2.2 节介绍。对于分割好的音符，通过基音频率检测和音长跟踪算法得到音高和音长，并计算相对音高音长实现哼唱片断的符号描述，2.3 节将介绍相关内容。之后采用最值特征点周边的符号描述作为音频特征，该方法与建立数据库时的算法相同，2.4 节给出了示例。最后利用 ED 方法或欧氏距离匹配哼唱音频特征与数据库中的音频特征，得到最相似的若干个实例提供给用户选择。算法的有效性将在第 3 章进行验证和分析。

2.2　音符检测算法

哼唱片断中的元音部分携带了音高音长信息，因此准确地检测元音帧就可以确定音符端点，方便后续处理。该文的检测算法主要分为两个步骤：第一步，根据元音特征（包括能量、过零率和频谱关系）检测元音帧；第二步，利用基音频率修正插入删除问题，获得更高的提取精度。

该文采用了作者之前提出的元音检测算法[8]，主要步骤如下：

步骤 1　预处理，通过选取无声帧（利用能量检测无声帧）计算频段分界点，计算各点贡献度 r_i。

步骤 2　依次取帧，若短时能量和短时过零率阈值存在，则计算当前帧的短时能量和短时过零率进行帧提取，转步骤 4；否则转步骤 3。

步骤 3　对本帧加窗、预加重、FFT 变换。计算低频段、高频段能量 E_{low} 和 E_{high}，若有 $E_{low}>2*E_{high}$，判定为元音帧。对本帧计算短时能量和短时平均过零率修改原有阈值。转步骤 2。

步骤 4　若当前帧被接受则依照步骤 3 按 1/P 概率抽样检测，若检测结果是元音帧转步骤 2；否则丢弃当前帧并按比例修改阈值。若当前帧被丢弃，当 P>1 时，全部进行检测；否则按 P 抽样检测。转步骤 3。

由于系统对音符检测的要求较高，根据检测结果，对各帧采用结合 HPS 和 LP 残差倒谱的方法[9]进行基音频率分析，减少插入删除错误（如图 2 所示）。因为基音频率分析是提取音高的必要操作，所以该步骤不会增加系统的总计算量。

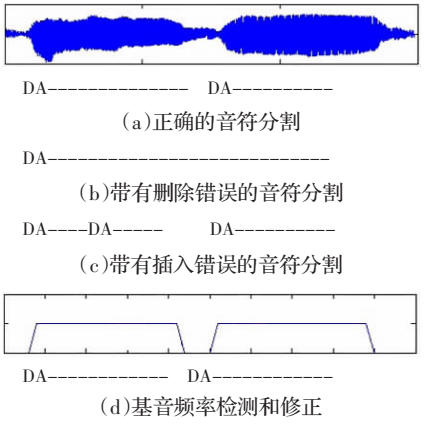


图 2　音符检测中的插入删除错误示意图及通过基音频率检测进行的修正

2.3　音高音长提取和符号化描述

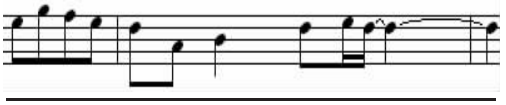
音频检索的难点主要在于无法直接描述音频内容，因此为了满足匹配和查询的需求，必须通过提取音频的特征实现符号化描述。对于音乐来说，传统的乐谱表达方式给研究人员提供了思路，提取音高和音长并进行符号化描述成为研究的主流。

一般来说，音乐的音高与基音频率有关，音长与音符持续的时间相关。因为音符分割时已经提取了各音符的基音频率，所以只需要通过对数变换实现频率和音高的转换即可，音长则可以直接用音符的持续时间表示。然而在实际应用中，比较绝对音高会受到用户哼唱能力的影响，因此需要通过相对音高、音频的提取实现符号化描述。提取方法如式(1)、(2)所示。

$$p(x)=\frac{\log f_x-\log f_{x-1}}{\log\sqrt[12]{2}} \tag{1}$$

$$d(x)=\frac{t_x}{t_{x-1}} \tag{2}$$

其中，$p(x)$表示音符 x 的相对音高，$f(x)$为音符 x 的绝对频率，$d(x)$表示音符 x 的相对音长，$t(x)$为音符 x 的绝对音长。图 3 给出了相对音长提取的示意图。



P	0	3	−2	−1	−2	−5	2	3	2	−2	…
D	1	1	1	1	1	1	2	1	0.5	4.5	…

P 表示相对音高，D 表示相对音长（注意区分全音阶和半音阶）

图 3　相对音高音长提取示意图

2.4　最值点符号特征提取策略

每一段音乐都有很多特征，如果直接对整段音乐的相对音高音长特征进行匹配必然会导致很大的计算量。此外，由于无法预知哼唱的起点，需要进行音频对齐，这就会进一步增大运

算时间。因此选择相对音高和相对音长的最大(最小)值点来表达一段音乐的特征,即用音高、音长变化最大/小点周边的符号描述作为音乐片段的特征描述。最值点周边提取多少个符号很敏感,数量太多会引入更大的不确定性,数量太少又无法表达音频段的特性,通过实验发现采用最值点前后各 2~3 个字符描述即可。具体操作中,若出现多个等值的最值点,则使用第一个带有完整标注的最值点作为特征点。因此一段音乐的最终描述如下式所示。

$$feature_{p\max}(p_i=p_{\max})=\{p_{i-2},p_{i-1},p_i,p_{i+1},p_{i+2}\}$$
$$feature_{p\min}(p_i=p_{\min})=\{p_{i-2},p_{i-1},p_i,p_{i+1},p_{i+2}\}$$
$$feature_{d\max}(d_i=d_{\max})=\{d_{i-2},d_{i-1},d_i,d_{i+1},d_{i+2}\}$$
$$feature_{d\min}(d_i=d_{\min})=\{d_{i-2},d_{i-1},d_i,d_{i+1},d_{i+2}\}$$

下面给出了一个示例描述,假设一段音乐的符号特征为:

No.	1	2	3	4	5	6	7	8	9	10
P	0	3	−2	−1	−2	−5	2	0	2	−2
D	1	2	0.5	1	4	0.5	0.5	0.5	2	3

则提取出的特征为:

$$feature_{p\max}(p_6)=\{-1,-2,-5,2,0\}$$
$$feature_{p\min}(p_8)=\{-5,2,0,2,-2\}$$
$$feature_{d\max}(d_5)=\{0.5,1,4,0.5,0.5\}$$
$$feature_{d\min}(d_3)=\{1,2,0.5,1,4\}$$

2.5 匹配策略

符号描述通常采用 ED(Edit Distance)算法来计算距离。然而该文的特征还可以看作 20 维的矢量特征,因此可以用欧氏距离来描述两个特征样本之间的相似度。即:

$$D=\sum_{i=1}^{5}[feature(X)-feature(Y)] \quad (3)$$

其中,$feature(X)$和$feature(Y)$指音乐片段 X 和 Y 的相对音高音长最值点特征。考虑到音乐的特性(如流行音乐中音高不变和音长不变的情况较多),通过加权调整最大值点和最小值点的作用,相似度计算变为:

$$D=\sum_{k=1}^{4}\sum_{i=1}^{5}w_k[feature(X)-feature(Y)] \quad (4)$$

其中,k=1,2,…,4 分别表示四种最值点,根据先验知识调节权值就可以实现不同的匹配策略。由于没有实验分析,该文没有采用加权策略,这将是后续工作之一。

将哼唱音频中提取的特征与音乐数据库中存储的特征进行相似度计算,按从小到大的顺序向用户提供候选音乐。并通过与用户的交互确定是否需要进一步的查询。

3 实验与结果分析

3.1 数据库描述和评价方法

实验中采用的音乐数据都是 MIDI 格式文件。根据 MIDI 规范,通过分析音频结构得到音高音长,实现音乐片段分割和片段特征的提取。数据库共采用了包括国内外不同音乐家不同风格的 300 个音乐作品,例如流行歌曲、古典音乐、舞曲、民族风格等。这些音乐作品通过人为干预被分割为 1 500 个片段以利于后续实验的安排。哼唱音频由乐感较好的非音乐专业大学生 28 人录制完成:每人选择熟悉的 5 首乐曲进行哼唱,并对 1 首随机乐曲进行模仿,哼唱和模仿输入不小于 30 s。因此哼唱曲目存在重叠,并且不能完全覆盖数据库中所有的作品,特别是民族风格类的作品没有涉及。

因为提交的结果按照相似度从小到大进行排列,所以评价系统检索精度的方法分为两类:第一类为首位匹配率,即第一个候选音乐即命中用户哼唱目标的成功率;第二类为前 n 位匹配率,即前 n 位候选音乐中包括目标音乐的成功率。

3.2 实验数据和分析

针对该文提出的系统,特别是针对最值点特征描述方法进行了仿真实验,实验主要从确认算法的精度和速度两方面入手。对比系统则采用了经典的 ED 算法和基于音符的音频检索系统。表 1 记录了在不同的数据库大小的情况下,两个检索系统首位匹配成功率和前 5 位匹配成功率的变化。

表 1 系统检索率随数据库大小的变化(%)

DB size	500	1 000	1 500
Max/Min(1)	75.6	71.3	70.5
Max/Min(5)	76.1	71.7	69.9
Note Based(1)	71.1	68.2	67.5
Note Based(5)	76.6	74.9	72.4

从数据上来看,提出的新算法比经典的基于音符的算法获得了更高的检索成功率。但是基于音符的算法前 5 位匹配率较首位匹配率有较大的提高,而该文算法相应的改变则并不明显。经过分析,主要原因是该文算法着重关注重要特征,丢弃了很多细节信息,无法实现首位匹配时往往在哼唱特征描述上出现了问题,因此前 n 位候选音乐仍然无法满足搜索需求。而经典算法采用全局匹配,即使存在误差导致总分竞争首位候选音乐失败,仍有较大可能保留在前 n 位。

图 4 给出了不同的数据库容量下,ED 算法、全局匹配算法和该文算法的时间复杂度对比。

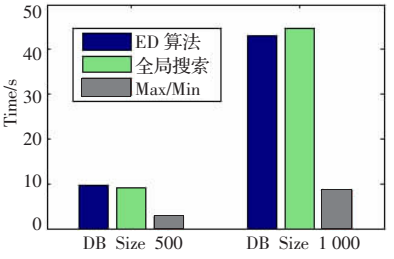


图 4 算法在不同数据库容量下的速度对比

图 4 中可以看出,由于 ED 算法和全局匹配算法都需要加入特征对齐计算,消耗了大量的时间。该文提出的算法避免了耗时的对齐操作,大大加快了搜索速度。在数据库不大的情况下,基本可以满足用户对查询速度的需要。

4 结论和展望

对已有的音频检索系统进行了改进,主要贡献在于利用元音检测方法确定哼唱音频的音符端点,并利用最值点相对音高音长特征的符号描述实现音频特征的表达,避免了由于哼唱不确定性引入的特征对齐操作,并因此降低了搜索匹配的计算量。从实验结果分析,该文提出的改进系统的首位检出率、前 5 位检出率和检索速度都超过了经典的基于音符的检索系统。并且在较小数据库上可以达到用户对查询速度的要求,具有一定的应用前景。后续的工作将在寻找更有代表性的特征点,提高音符端点检测精度等方面展开,并希望通过实验确定匹配计算的权值,进一步提高计算精度。

(下转 144 页)