

# 旋律弹性匹配实现 Web 哼唱检索

李 鹏<sup>1</sup> 王小明<sup>1</sup> 王小凤<sup>2</sup> 王亚文<sup>3</sup>

(陕西师范大学计算机科学学院 西安 710062)<sup>1</sup> (西北大学信息科学与技术学院 西安 710069)<sup>2</sup>  
(西安工业大学计算机科学与工程学院 西安 710021)<sup>3</sup>

**摘 要** 面对日益增加的音乐作品及多样化的用户检索需求,采用基于内容的音乐检索方式,提出一种对音乐旋律的定义及表达方式,并在此基础上依照旋律轮廓的几何相似性实现音乐的相似度计算,具体地提出影响检索算法准确度的若干因素,通过实验的方法确定相应的阈值,最后实现检索系统的单机版开发和 Web 版发布。实验表明,所采用的方法正确、有效,能够取得较高的检索准确度。

**关键词** 音乐检索,旋律轮廓,弹性匹配,线性对齐

**中图分类号** TP391.42 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.12.042

## Melodies Elastic Matching for Humming Retrieval on Web

LI Peng<sup>1</sup> WANG Xiao-ming<sup>1</sup> WANG Xiao-feng<sup>2</sup> WANG Ya-wen<sup>3</sup>

(College of Computer Science, Shaanxi Normal University, Xi'an 710062, China)<sup>1</sup>

(College of Information Science and Technology, Northwest University, Xi'an 710069, China)<sup>2</sup>

(College of Computer Science and Technology, Xi'an Technological University, Xi'an 710021, China)<sup>3</sup>

**Abstract** With the increasing of musical works and variety user retrieval needs, using content-based music retrieval methods, we proposed a music melody definition and expression style, and then completed the similarity calculation on the melody contour geometric similarity. Specifically we proposed a series of factors affecting the accuracy of retrieval algorithms, and determined the appropriate threshold through the experimental method, and implemented the final retrieval system both on stand-alone version and Web version. The experiment results indicate that the solution is correct, effective and can achieve higher retrieval accuracy.

**Keywords** Music retrieval, Melodic contour, Elastic matching, Linear alignment

## 1 引言

数字音乐的急剧增加以及用户期望采用多种音乐检索手段的需求,使得人们在处理音乐作品时遇到了瓶颈:首先,互联网上海量的音乐数据通过人工方式实现分类和标注变得艰难;其次人工信息的介入,使得歌曲的某些标注如音乐情感类型、流派等加入了主观偏见,不具有普遍性;人们无法通过部分歌曲的旋律实现对未知歌曲信息的检索;歌曲的大量增加也对歌曲的版权保护提出要求;音乐爱好者的增加使人们对音乐类娱乐工具也有迫切需求。基于内容的音乐检索方式不依赖于标注信息,从乐曲内容本身出发来认定乐曲,根据音乐中的旋律、音高、节奏等信息来实现检索、分类,进而管理音乐信息。它在音乐数据库管理、Internet 音乐检索以及生活娱乐方面都具有非常重要的意义。在娱乐方面,一个典型应用就是 KTV 点歌系统和娱乐网站的歌唱搜索。通常人们在 KTV 唱歌时都是通过输入字符信息查找歌手名称或者歌曲名称,

以此获得所需歌曲。然而一个常见的情况就是用户对某首歌的某几句的旋律比较熟悉,对歌词记忆不准确,对歌手和歌名信息并不掌握,这时传统的歌曲检索方法就无法帮助用户找到其希望的歌曲,在网页上检索歌曲也面临同样的问题,网上歌曲的检索也同样以歌曲元数据作为关键字开展,少有的几个网站也仅以歌曲的节奏、声调和音色的宏观特征来检索某一类歌曲,在这种状况下,通过歌曲自身旋律变化的特征实现检索成为必要。

## 2 研究现状

目前国际上对基于内容音乐检索的研究主要集中在韩国、新西兰、新加坡、美国和法国,中国国内也积极开展相关领域的研究,中科院、上海交通大学、北京师范大学、西北大学、陕西师范大学、台湾“清华大学”等科研院所均在该课题的研究上取得了较大进展,在详细调研各方面的成果的基础上,绘制了音乐检索研究的主流体系结构图,如图 1 所示。

到稿日期:2013-06-25 返修日期:2013-08-16 本文受中央高校基本科研业务费专项资金(GK201302027),陕西省教育厅项目(12JK0730),陕西师范大学青年科技项目(201001001),陕西师范大学科研启动经费基金(999453)资助。

李 鹏(1981—),男,博士,讲师,主要研究领域为音频音乐检索、模式识别、图像视频处理、计算机网络等, E-mail: lpfai@mail.bnu.edu.cn; 王小明(1964—),男,博士,教授,主要研究领域为信息安全、访问控制、工作流安全;王小凤(1979—),女,博士,副教授,主要研究领域为音频处理、数据挖掘;王亚文(1981—),男,硕士,讲师,主要研究领域为计算机网络。

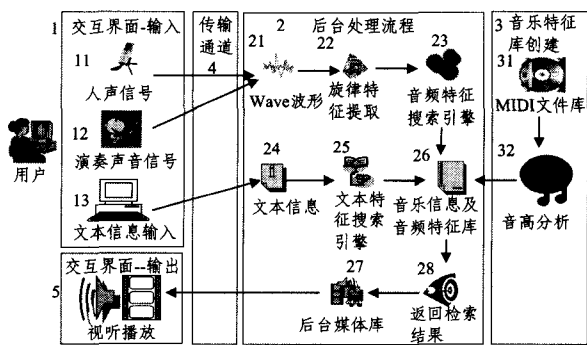


图1 音乐检索研究体系架构

音乐检索的交互过程可以简单描述为输入、处理和输出。输入是用户向计算机提交查询的内容，一种查询方式是直接采用人声输入，即进行歌唱、哼唱、口哨等的麦克风录入方式；

结构	概念级别		音乐内容特征				
语义级	高级 2	可表达	表达				
	高级 1	结构化	影响、经验				
非语义级	中级	概念级	旋律	谱音	节奏	音源	动态
	低级 2	感知级	连续、间断模式	同步音程模式	节拍	频谱封装	自动调整声高级别
	低级 1	声学级	基音	时间	音色	音量	
			频率	时长	频谱	音强	

图2 音乐感知级别分类体系

国外研究机构对音乐检索进行了多方面的研究。1995年, Ghias<sup>[3]</sup>对单音轨的MIDI音乐的哼唱检索进行了开创性研究, 采用时域自相关算法提取音高信息, 然后使用字符串近似匹配的方式实现对单声部音乐的检索。新西兰 Waikato 大学的 Rodger J. McNab<sup>[4]</sup>和 新西兰数字音乐图书馆合作开发了一套名为 MT(Melody Transcript)和 MELDEX 的系统, 但是 MELDEX 系统无法正确切割音符, 用户在哼唱时, 必须在音符与音符之间自行留下短暂的停顿或插入特定断句声音, 因此用户使用起来不方便, 也不自然。Tomonari 等人<sup>[5]</sup>提出同时使用音长和音高作为旋律特征, 然后采用基于“动态闭值调整”的匹配方法, 采用了先粗选后细选的匹配策略。Kosugi<sup>[6]</sup>提出同时使用音高变化和音高分布来提高系统的性能。他们开发出了一套名为 SoundComPass 的系统, 系统中有 10086 首歌曲, 在使用时需配合乐器的节拍哼唱。韩国的 Seungmin 等<sup>[7,8]</sup>改进了基音提取算法, 在传统检索算法的基础上增加了访问频度索引功能, 采用基于遗传算法的相关反馈机制, 使得算法的检索正确率有了显著提升。

目前代表 CBMR 最高学术水准的是国际音乐信息检索学会(The International Society of Music Information Retrieval, ISMIR)举办的年度会议, 该会议自 2000 年开办, 已在美国普利茅斯、伯明顿、法国巴黎、西班牙巴塞罗那、英国伦敦、日本神户等 13 个城市举办了 13 次, 最近的一次于 2012 年 10 月在葡萄牙波尔图举办, 这次会议依照惯例将 CBMR 的主要研究内容分为音乐文化、节奏节拍、相似度、旋律与基音、标记信息、用户学习、音频分类、检索评价、不可行方案汇集、光学音符识别、人声处理共计 11 个类别。会上, Maarten de Rijke<sup>[9]</sup>对 CBMR 近年的研究趋势做了综述分析。Philippe Hamel 等<sup>[10]</sup>分别探讨了在低级语义上分析音色、音量、节奏和在高级语义级别上分析旋律、和声、节奏等相关特征来实现音乐相关性的音频特征构建方法。Justin Salamon 等<sup>[11]</sup>提出

另一种查询方式是辅助乐器演奏方式, 可由擅长乐器演奏的用户用钢琴、提琴、口琴等各类有旋律高低的乐器实时弹奏录入, 还可采用键盘查询的方式, 比如用户使用 MIDI 键盘或者计算机键盘录入音乐乐符信息或者歌曲文本信息。用户的查询内容通过远程或者本地的传输通道提交给后台服务器处理, 分别按照不同的查询内容做不同的处理。

音乐是一种包含了丰富语义信息的音频, 而对于音频信息的表达, 其定义包括语义层、非语义层。从音乐物理意义到语义级别的定义方法上, Leman 等<sup>[1]</sup>从人类音乐学的角度将音乐的理解分为 5 个层次, 即信号、帧级别、参数级、事件级和概念级, 该体系根植于音乐声学特征, 上推到最高级的语义描述。基于这种描述, 文献[2]细化了这种思想, 提出了基于内容音乐检索的层次界定方法, 如图 2 所示。

了一种利用统计规律改进音频信号基音特征轮廓的 melodicness 方法, 以方便旋律轮廓进入计算环节, 避免繁冗的人工修正。Jos'e R. Zapata 等<sup>[12]</sup>通过设定置信度阈值来快速判定大规模音乐数据中的节奏变化特征, 统计拍数等相关信息, 为检索算法提供输入。此外, 由 ACM 主办的年度国际多媒体检索会议也汇集了 CBMR 研究领域的最新思想。

除了学术会议, 在高水平期刊上的文献也包含了该领域的新成果。Dmitry Bogdanov 等<sup>[13]</sup>通过统计分析用户音乐赏析行为来提取偏好的语义特征, 通过对音乐风格相似性度量来建立语义概率模型, 较好地实现了有针对性地向用户推荐音乐作品的算法。中国台湾学者 Wei-Ho Tsai 等<sup>[14]</sup>提出一种快速处理旋律相似性度量的算法, 该方法在实现时域到频域的傅里叶转化之后, 将不等长的音符序列转换为等长的同维特征向量, 通过欧几里德距离实现匹配, 该方法在时间复杂度上有良好的表现。

国内大陆地区有关 CBMR 的研究方兴未艾, 其中中科院声学所、上海交通大学、浙江大学、北京师范大学、西北大学等众多高校、研究机构都先后涉足此课题, 能反映近几年国内学术现状和研究趋势的文献也不断出现。刘亚多等<sup>[15]</sup>提出一种基于 MDCT 频谱熵的压缩域音频指纹算法, 该算法对各种常规频域和时间域的音频信号处理失真问题具有较强的鲁棒性, 在包含百首歌曲的中文音乐库上能获得较高识别率。冯嘉礼等<sup>[16]</sup>提出哼唱片段旋律特征提取的定性映射模型和算法, 包括哼唱旋律音符切分阈值的动态定性判定以及音符音高确定的动态赋权的定性映射, 实验表明该方法能有效得到哼唱片段的旋律轮廓线。王小凤等<sup>[17,18]</sup>针对音乐自身特征, 提出一个以句为单位的哼唱检索算法, 取得了较好检索效果。

### 3 查询信号处理及数据库构建

目前通过歌曲的名称、词曲作者、歌词等元数据信息实现

歌曲检索已经有广泛的应用,不再叙述。若是采用 MIDI 键盘、计算机鼠标键盘等输入设备采集音符信息、旋律轮廓或者节奏信息,都能够精确统计某个音符的音高和它的持续时间,也能较容易地获取输入旋律的音高变化曲线,但使用者需要有专业的音乐弹奏功底或者较长的培训。

使用麦克风采集人声输入或者乐器弹奏的输入获得的是数字化的音频信号,它是按照一定的采样率得到的声压大小序列,该离散的序列隐含了声音的音调、音强和音色变化信息。在音乐检索中,关注的不是哪个人在唱,或者哪个乐器在演奏,而是音频信号的音调变化状况,因此需要使用有效的时域-频域转换方法来计算查询音频片段的音调变化信息。对音频信号进行加窗分帧做基因提取和“小阁”后处理即可得到规整的旋律轮廓信息。数据库的创建方法有 3 种,即采用基音提取方法、乐谱录入方法和 MIDI 分析方法。MIDI 分析方法是实现快速、准确、大批量建库的最佳方法。上述的旋律轮廓获取方法和音乐特征库建库方法作者之前均已详述,分别参见文献[19,20]。

#### 4 弹性匹配检索方法

音频特征搜索引擎的构建首先须明确实施匹配的两组数据的格式,查询序列是经过旋律特征提取和规整的数值变化序列,为提高检索速率,在匹配之前,需要把该序列转化为一个 2 维标准形式,以(音高、音长)的形式描述其变化;音乐特征数据库中的数据是对 MIDI 文件进行分析得到的描述音高和持续时间的数据,同样也以(音高、音长)的形式表示,需要注意的是,二者的生成过程必须保证同样的旋律在同样的时间内生成的基音值数量基本相等,本研究规定两者均按照每秒生成约 100 个基音值的速度来生成,这样,同样的歌曲通过 WAVE 和 MIDI 两种文件格式的分析,均可得到长度相近、变化轮廓相似的音高曲线图。不同的是,通过 MIDI 获取旋律轮廓的时候已经按照音乐小结中短句与短句之间的较长静音进行断句划分,即将持续性的音乐旋律变化过程切分为若干个短句,切分的原则是该短句是用户哼唱的最小单位,用户哼唱检索时必然唱的是一个或者多个连续的短句,而不会哼唱半个短句,该过程融合了歌曲的特性,这为提高检索速率和准确度提供了先决条件。

确定了待匹配的二组数据之后,提出了一种基于几何相似性的弹性匹配方法。用这种方法找到查询序列在每首歌曲中最可能出现的位置及与该位置的相似程度,之后比较每首歌曲所得到的相似度并排序,即可得到检索结果。

处理流程如图 3 所示,检索详细过程如下:

- 1)按顺序从音乐特征数据库中获取一首事先已经实现短句划分的歌曲。
- 2)将待匹配指针指向该歌曲句首位置。
- 3)参照查询序列的长度,以短句为最小单位从库中该歌曲上指针位置截取与查询序列长度相当的序列,满足所截取的片断长度介于 0.7~1.3 倍于查询序列长度。
- 4)将所截片断与查询片断进行拉伸,使之等长、等幅。
- 5)采用线性对齐算法将二者相近的音高转折点拉伸对齐。

- 6)对二者进行差运算,计算差序列的均方差并保存。
- 7)将指向该歌曲的指针向后移动一个短句。
- 8)重复执行 3)~7)操作,直至该歌曲结束。
- 9)重复执行 1)~8)操作,直至该库歌曲取尽。

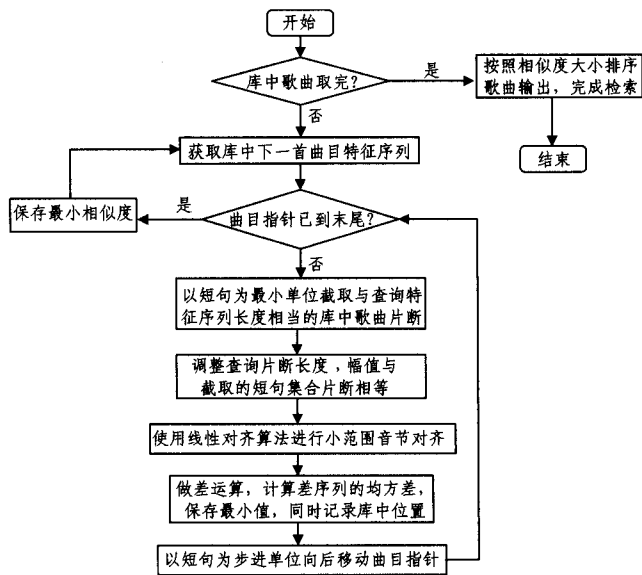


图 3 检索流程图

其中第 4)步将所截片段与查询片段进行拉伸使之等长、等幅,调整等幅必须以人声旋律片段为标准,因为特征库中的某个片段可能是持续水平的旋律曲线,这样,无论人声的旋律如何变化,最终的结果都被调整到一个很窄的范围内,进行查运算的结果必然将二者判断为很相似,而使用人声旋律片段为标准进行调幅,则可以避免这种状况。

查询片段与每首歌曲的每个裁剪片断都有一个均方差,每首歌曲取最小者作为描述该歌曲与查询片段相似程度的标准,按照库中均方差大小排序,即可得到相似度由高到低的歌曲排序,同时也能确定最相似的片断在歌曲中出现的位置。

由于该过程是以哼唱旋律轮廓曲线和特征库旋律轮廓曲线的几何相似性为考察内容,因此曲线上微观范围内增加或者减少若干个音高、延长或者缩小若干音长并不会对检索的结果有重大影响,这正符合人声哼唱的不确定性,容忍了人声在微观范围内并不规整的特性。同时使用均方差作为评定两个曲线相似性的标准,也能客观地评价处于不同音调范围内的两个旋律轮廓,比如同样的一个旋律,男声和女声得到的旋律轮廓曲线走势都相同,但女声的音调范围会比男声的高,该匹配方法恰能较好地解决这种差异。

该方法允许用户所唱的片段与标准旋律片段时长有变化,但一般应保持在一定的范围内,如果同样一段旋律改为较高的速率播放,我们不认为这是同一段旋律,比如哀乐如果加速播放,其中体现的将是一种欢快的气氛,人们很难认为这是之前听到过的哀乐。

该弹性匹配方法在实施中仍受多方因素的干扰,因此系统初步开发完成后我们需综合各方因素来对其进行优化。

#### 5 系统实现及优化

我们设计实现了一个可以通过哼唱/歌唱实现歌曲检索的系统,系统所使用的数据库为按 MIDI 分析构建的音乐特

征库。系统设计了一种便捷的交互界面,用户可以在单击按钮之后现场哼唱/歌唱,系统实时录音保存为文件、做预处理、提取基音、后处理、匹配检索等操作,最终按相似度返回若干首歌曲名称,单击歌曲即可实现播放。

在上述算法设计及系统实现中存在多个影响检索速率和准确率的因素,现将其进行总结并提出解决方案。

### 5.1 影响检索耗时的因素

用户发声录音参数为 44.1kHz、8 位、单声道。经测试,影响运行速率的因素主要有以下方面:

1) 基音提取过程,因每秒采样数据过多,在时域、频域转换运算中形成了大规模的数据量。解决方法是适量减少采样频率。经实验可得,使用 22.05kHz 的采样率能够较好地保持音高曲线并能将处理时间减少到原来的 1/10 以内。

2) 旋律特征库的存储方式。最直接的方法是将 MIDI 分析所得的旋律音高变化数据直接存储在数据库中,但这样会使每首歌保存大量的数值型数据,使数据库规模过大,并且在检索中读入读出均造成了大量的耗时。采用(音高、时长)的保存方式能大幅度缩减数据量,而且音高、时长的数值采用能够描述其变化的最小精度。本文将描述音高时长的数字定义在 0~255 之间,并采用将该数值对应的 ASCII 码存入数据库的方式,这大幅地缩减了数据量,加速了数据存取速率。

3) 数据库频繁存取耗时。解决办法是在程序加载时将已按上述方法缩减容量后的特征库数据库全部读入内存,使其在内存中操作。

### 5.2 影响检索准确度的因素

表 1 参数值确定表

	1	2	3	4	5	6	7	8	9	10	排序和 (前十位)
去刺滤波宽度(15) 弹性强度(0.8~1.2) 自动对齐阈值(8)	1	7	5	1	4	11	1	1	1	15	32
去刺滤波宽度(19) 弹性强度(0.8~1.2) 自动对齐阈值(8)	1	6	3	4	4	11	1	1	1	13	32
去刺滤波宽度(21) 弹性强度(0.8~1.2) 自动对齐阈值(8)	2	5	2	4	3	10	1	1	1	18	29
去刺滤波宽度(25) 弹性强度(0.8~1.2) 自动对齐阈值(8)	2	6	2	12	1	10	1	1	1	18	36
去刺滤波宽度(21) 弹性强度(0.9~1.1) 自动对齐阈值(8)	1	18	6	3	2	18	1	1	1	10	43
去刺滤波宽度(21) 弹性强度(0.7~1.3) 自动对齐阈值(8)	1	6	5	9	5	6	1	1	2	18	36
去刺滤波宽度(21) 弹性强度(0.8~1.2) 自动对齐阈值(4)	1	5	2	12	11	8	1	1	1	18	42
去刺滤波宽度(21) 弹性强度(0.8~1.2) 自动对齐阈值(10)	1	3	2	7	7	12	1	1	1	18	35

在算法设计和程序实现中,有较多影响检索准确率的因素,如为了去除基音提取后序列中存在的毛刺,增强不同音符之间的跳变,而采用“小窗”滤波的后处理方法,这个滤波宽度应该多长比较合适;人声旋律片段和特征库旋律片段的长度在怎样的弹性范围内进行匹配,即弹性强度怎样确定;自动对

齐算法将两个片段中音调跳变位置接近的两个点进行合并,这项操作能较好地解决因人声旋律变化速率的自然性而产生与标准旋律无法严格一致的状况,但两个跳变点距离在什么范围内进行合并。

以上影响检索准确率的 3 个因素需要通过实验来确定其最佳参数范围,因此本研究采用了台湾“清华大学”发布的 48 首标准歌曲进行测试,以确定参数阈值。确定方法为使用编号为 1~10 的 10 首歌曲,配置不同的参数,测试正确的检索结果在返回歌曲列表中所处的位置。由于排序越靠后,随机因素越大,因此在每组检索参数下,去除一个排序最靠后的歌曲,计算其余歌曲的排序累加和来评价参数配置效果。给出若干代表性的测试结果,如表 1 所列。

通过上述实验结果可以看出,选择去刺滤波宽度为 21、弹性强度为 0.8~1.2、自动对齐阈值为 8 的参数组能取得最好的检索效果。

## 6 Web 实现

为了实现多人在线使用,对已经实现的系统进行了 Web 开发,使用 PHP 和 Socket 技术实现用户音频、文字信息的获取和检索结果的反馈。Web 实现框架可分为 3 个部分:客户端、服务器端和后台处理系统。

客户端负责采集用户音频、文字输入,支持用户按照歌曲的多种分类方法浏览同时能够显示用户检索结果的列表。用户可以直接在网页上点击相关按钮实现实时录音,或者可将事先录好的文件上传到服务器。界面上也支持传统的通过关键字检索或者通过分类浏览的形式查找目标歌曲。界面如图 4 所示。



图 4 Web 实现用户界面

服务器端负责接收用户输入的音频文件或字符串,采用 Socket 通讯将从客户端获得的信息发送到后台处理系统,同时获得从后台返回的歌曲列表信息,之后将该信息或者用户浏览的响应发送到客户端。

后台处理系统负责处理输入的音频信息,获得特征序列后与库中数据匹配检索,或者按照用户输入的关键字进行检索,最后返回用户歌曲列表及相应的歌曲文件和元数据。

需要说明的是,单机使用时,数据库只对一个用户提供访问,可以允许用户有反复的读写操作,如果是多人同时使用,则无法提供服务。本文的解决办法是:每个用户提交检索请求时访问数据库,将保存音乐特征的表读取到内存之中,然后断开与数据库的连接,以后的每一次检索都使用保存在内存的特征库进行相似度计算、排序和输出,通过这个手段,可以

支持多个用户同时使用检索系统。

用户提交检索请求、实现检索、返回数据等传输过程如图 5 所示。

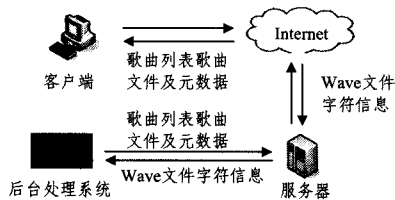


图 5 音乐检索 Web 实现框架

为了达到较高的准确度，系统要求用户输入的查询片段必须在 8 秒以上，最好控制在 15 秒以内，一般为 2~4 个乐句。从上边的测试样本中选择的符合时间要求并且声音洪亮、杂音较少的可用样本共计 177 个。使用的音乐特征库共计乐曲 515 首。准确度分为 3 个标准，即分别将检索成功认定为出现在序列的前 3 位、前 5 位和前 10 位。分别以本文提出的旋律曲线配准方法和字符式特征定义实现检索测试。实验结果如表 2 所列。

表 2 实验结果

准确度(%)	前 3	前 5	前 10
旋律配准	78.0	81.4	86.4
字符匹配	68.9	83.1	85.9

**结束语** 以实现基于内容音乐检索为目标，采用弹性匹配为核心检索方法实现了基于内容的哼唱检索系统，然后利用 PHP 及 Socket 技术实现了哼唱检索系统的 Web 发布。研究围绕怎样构建一种具有高容错的旋律比对方法展开论述，提出了一种具有较好检索准确度的旋律几何轮廓匹配方法，在系统实现及调试中，重点讨论了实现基于内容音乐检索所必须解决的若干问题，并给出了处理办法，最后实现了可以支持多人同时在线检索的 Web 哼唱检索系统。多次实验及对比表明，本算法能够获得较好的检索准确度。目前，系统音乐库容量仍然有限，今后的工作可着重在用户接口和音乐数据库检索机制上展开研究，以达到对大规模音乐数据库的高精度检索要求。

## 参 考 文 献

[1] Leman M, et al. Musical audio mining; In dealing with the data flood; mining data, text and multimedia [J]. STT Netherlands Study Centre for Technology Trends, Rotterdam, 2002; 440-456

[2] Lesaffre M, et al. User-dependent taxonomy of musical features as a conceptual framework for musical audio-mining technology [C] // Proceedings of the Stockholm Music Acoustics Conference. Stockholm, Sweden, 2003; 635-638

[3] Ghias A, Logan H, chamberlain D, et al. Query by humming: Musical Information Retrieval in an Audio Database [C] // Proceeding of Third ACM Informational Conference on Multimedia. San Francisco, 1995; 231-236

[4] McNab R J, Smith L A, Witten J H. Towards the Digital Music Library; Tune Retrieval from Acoustic In Put [C] // Proceeding of ACM Multimedia Conference. 1996; 11-18

[5] Snada T, Muraoka Y. A WWW based Melody Retrieval system- An Indexing Method for A Large Database [C] // Proceeding of International Computer Music Conference. 2010; 170-173

[6] Kosugi N, Nishihara Y, Sakata T, et al. A practical query-by-humming system for a large music database [C] // Proceeding of ACM Multimedia Conference. November 2000; 333-342

[7] Rho S, Hwang E. FMF; Query adaptive melody retrieval system [J]. The Journal of Systems and Software, 2006(79); 43-56

[8] Rho S, Han B J, Hwang B E, et al. MUSEMBLE; A novel music retrieval system with automatic voice query transcription and reformulation [J]. The Journal of Systems and Software, 2008 (81); 1065-1080

[9] de Rijke M. Current Trends in Information Retrieval [C] // Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012. October 2012

[10] Hamel P, Bengio Y, Eck D. Building Musically-Relevant Audio Features Through Multiple Timescale Representations [C] // Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012. October 2012; 553-558

[11] Salamon J, Peeters G, Robel A. Statistical Characterisation of Melodic Pitch Contours and its Application for Melody Extraction [C] // Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012. October 2012; 187-192

[12] Zapata J R, Holzapfel A, Davies M E P, et al. Assigning a Confidence Threshold on Automatic Beat Annotation in Large Datasets [C] // Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012. October 2012; 157-162

[13] Bogdanov D, Haro M, Fuhrmann F, et al. Semantic audio content-based music recommendation and visualization based on user preference examples [J]. Information Processing and Management, 2013, 49(1); 13-33

[14] Tsai W-H, Tu Yu-ming, Ma C-H. An FFT-based fast melody comparison method for query-by-singing/humming systems [J]. Pattern Recognition Letters, 2012, 33(16); 2285-2291

[15] 刘亚多, 李伟, 李晓强, 等. 压缩域鲁棒音乐指纹算法研究 [J]. 电子学报, 2010, 5(5); 1172-1176

[16] 冯嘉礼, 刘文娟. 基于定性映射的哼唱音乐旋律提取 [J]. 计算机工程, 2012, 38(18); 51-254

[17] 王小凤, 耿国华, 孙霞, 等. 一个以句为单位的音乐哼唱检索算法 [C] // 第五届和谐人机环境联合学术会议. 西安, 2009; 44-49

[18] 王小凤, 耿国华, 李鹏, 等. 一个基于相关反馈的神经网络音乐情感分类器构建算法 [J]. 西北大学学报: 自然科学版, 2012, 42(1); 30-35

[19] Li Peng, Zhou Ming-quan, Wang Xue-song, et al. An Effective Signal Processing Method to Musical Information Retrieval [C] // Proceedings of the 2009 International Conference on Computational Intelligence and Natural Computing. Wuhan, 2009; 280-283

[20] 李鹏, 周明全, 李娟, 等. 音乐检索特征库构建方法 [J]. 软件学报, 2009, 20(12); 213-220