

## 基于智能分类算法的音乐和弦识别与分析

**摘要：** 本文提出了一个对音频信号中所包含的和弦信息进行识别与分析的系统。该系统以音级轮廓图（PCP）为基础，结合了常量 Q 变换、KNN 等算法，实现了对钢琴和弦较为准确的识别。同时，本文也提出了一个对音乐信号进行曲速测算与节拍跟踪的算法，并通过实际的音乐对模型进行验证，实验结果证明了该算法具有较高的准确率与稳健性。

**关键词：** 音乐，和弦识别，智能算法，PCP，节拍跟踪

### Identification and Analysis of Musical Chords Based on Intelligent Classification Algorithm

**Abstract:** This paper presents a system for identifying and analyzing the chord information contained in audio signal. Based on the Pitch Class Profile (PCP) and the Constant-Q transform and KNN algorithm, the system can recognize piano chords accurately. At the same time, this paper also puts forward an algorithm to measure the tempo and track the beat of the music signal, and verifies the model through the actual music, experimental results show that the algorithm has high accuracy and robustness.

**Key words:** music, chord recognition, intelligent algorithms, PCP, beats tracking

# 目 录

1 绪论.....	1
1.1 选题的背景及研究意义.....	1
1.1.1 选题背景.....	1
1.1.2 研究意义.....	1
1.2 国内外研究现状.....	1
1.2.1 国外研究现状.....	1
1.2.2 国内研究现状.....	2
1.2.3 文献评述.....	2
1.3 研究方法与研究内容.....	3
1.3.1 研究方法.....	3
1.3.2 研究内容.....	3
1.4 本文创新点.....	4
2 和弦识别与节拍跟踪概念与相关音乐理论.....	5
2.1 和弦与节拍的定义.....	5
2.2.1 和弦的定义.....	5
2.2.2 节拍的定义.....	5
2.2 和弦识别的理论基础.....	6
2.2.1 律法.....	6
2.2.2 十二平均律.....	6
2.2.3 离散傅里叶变换.....	7
2.3 节拍跟踪的理论基础.....	8
2.3.1 音乐中音符的组织特点.....	8
2.3.2 音乐信号的响度特征.....	8
3 周期乘数法.....	10
3.1 用于曲速测算的周期乘数法.....	10
3.1.1 算法设计.....	10
3.1.2 算法评价.....	12
3.2 用于节拍跟踪的周期乘数法.....	13
3.2.1 低通滤波器与起音检测.....	13

3.2.2 用于节拍跟踪的周期乘法法 .....	14
3.3.3 算法评价 .....	14
4 基于离散傅里叶变换的音级轮廓图 .....	16
4.1 音级轮廓图的概念与求解流程 .....	16
4.2 和弦类型模式 .....	17
4.3 算法评价 .....	17
4.3.1 算法的识别率 .....	17
4.3.2 误差来源与分析 .....	18
5 常量 Q 宽带变换 .....	20
5.1 常量 Q 宽带变换的定义 .....	20
5.2 利用 CQT 结果进行 KNN 聚类 .....	21
5.2.1 KNN 聚类 .....	21
5.2.2 算法实现 .....	22
5.2.3 算法评价 .....	23
5.2.4 误差来源于分析 .....	23
5.3 基于常量 Q 变换的音级轮廓图 .....	24
5.3.1 算法实现 .....	24
5.3.2 算法评价 .....	25
6 引入节拍跟踪的乐段和弦分析 .....	26
6.1 算法流程 .....	26
6.2 基于拍同步的音频切分与平滑处理 .....	26
6.3 算法评价 .....	26
6.3.1 对钢琴柱式织体的响应 .....	26
6.3.2 对钢琴混合织体的响应 .....	27
6.4 误差来源与分析 .....	29
7 结论 .....	30
参考文献 .....	31
致 谢 .....	33

## 1 绪论

### 1.1 选题的背景及研究意义

#### 1.1.1 选题背景

近年来，随着我国经济的快速发展，人民群众对于艺术的追求也越加提升。而音乐艺术作为人类美学追求的重要组成部分，自然也受到人们的推崇。

在音乐鉴赏中，准确的听辨乐段中的和弦走向，是专业音乐人的一项必不可少的技能。然而音乐理论作为一项人文艺术类的学科，和弦的听辨需要大量的听辨练习与经验积累，对于刚踏入音乐鉴赏领域的新人，这是一段耗时耗力且枯燥的过程。

同时，随着互联网技术的发展，涌现出了例如网易云音乐、QQ 音乐、Spotify 与 Apple Music 等互联网音乐流媒体平台，为广大听众提供了更加方便快捷的欣赏音乐的方式。同时，各大平台的音乐推荐算法也成为了各大音乐流媒体平台提升自身市场竞争力的核心技术，如何提高音乐推荐算法的准确率，成为了各大音乐流媒体平台的重要议题。

和弦是音乐中最基本的元素之一，也是表达音乐情感的与风格的重要手段。对音乐中和弦准确的识别与分析，能为音乐的风格与情感分类提供有力的判据，进而应用与音乐流媒体平台的音乐推荐算法之中。

#### 1.1.2 研究意义

对音乐中的和弦进行识别与分析，本质上是一个信号处理与分析的过程。即对音频信号进行时域频域转换、滤波等处理后提取相关特征，再对特征向量进行聚类，来实现对音乐中和弦的识别。建立一个准确、稳健的和弦自动识别系统，可以辅助音乐爱好者或专业音乐制作人对乐曲进行分析，降低音乐鉴赏与教育门槛，提高音乐制作的工作效率。同时，计算机科学作为典型的理工科类学科代表，如何与其他学科交叉应用，成为当今世界的热门话题。在本文中，将以计算机科学为基础，通过结合信号处理与分析、音乐理论等诸多学科的专业知识，构建一个和弦识别系统，在为人们提供一个实用的音乐学习工具的同时，也探究跨学科专业知识交叉的方法与意义。

### 1.2 国内外研究现状

#### 1.2.1 国外研究现状

国外的专家学者已经对计算机技术在音乐和弦的识别与分析领域做了进行了相对

成熟的研究。1999年由斯坦福大学音乐和声学计算机研究中心的 Takuya FUJISHIMA 提出的基于离散傅里叶变化的音级轮廓图（Pitch Class Profile, PCP）<sup>[7]</sup>建立了计算机领域在和弦识别工作中的理论基础。音级轮廓图是一个十二维的向量，它代表了一段音频信号在十二平均律中每个音符的强度大小，通过建立音频信号的频率与音高的映射，计算出向量中每个元素的大小。然而，传统的离散傅里叶变换由于使用线性标度，在低频区域的分辨率较低，难以对信号中的低频部分进行有效的描述。Schoerhuber 等人提出的常量 Q 变换（Constant-Q Transform, CQT）<sup>[11]</sup>通过动态改变不同频率滤波器的窗口大小，实现了对数标度的频域变换，由于 CQT 的这种性质与律法中音符的频率分布特点十分吻合，这使得它十分适合于音乐信号的处理与分析。在信号的预处理方面，Nobutaka 等人提出的谐波-冲击分离（Harmonic-Percussive Source Separation, HPSS）算法<sup>[10]</sup>，使用支持向量机实现了音乐信号中和声乐器与节奏乐器的分离，降低了节奏组乐器在和弦识别工作中产生的噪音影响。Filip Korzeniowski 与 Siddharth Sigtia 等人尝试使用混合递归神经网络与深度神经网络等人工智能算法对音频信号中的和弦特征进行聚类，获得了较为准确的结果。<sup>[8,9]</sup>

### 1.2.2 国内研究现状

我国的音乐产业与相关的技术研究水平相对滞后，因此早期对于计算机技术在音乐领域的相关应用研究较少。然而，随着经济水平的提高与人民对于艺术追求的提升，我国的音乐市场发展迅速，同时也出现了一些相关的研究。如闫志勇提出的基于增强型 PCP 与 SVM 的和弦识别方法<sup>[16]</sup>，相较于传统模型提升了最高 9.7% 的准确率，为我国在相关领域的研究提供了重要参考。

### 1.2.3 文献评述

国内外的各种研究都指出，音级轮廓图是计算机进行和弦识别的主要方法，是连接音频信号与音乐理论概念的重要桥梁。而传统的基于离散傅里叶变换的音级轮廓图存在较大的识别误差，这些误差主要是由于信号中的噪音影响与傅里叶变化在低频区域的分辨率达不到对音符的准确分类造成的。应用不同的方法对音级轮廓图的求解方法进行改进，是提高模型识别率的关键。

谐波-冲击分离算法可以有效的解决由节奏组乐器对频谱造成噪声影响的问题。同时，CQT 算法的提出也为音乐信号的处理与分析提供了更加合适的理论基础。

然而，大多数的研究都忽略了和弦识别中的一个很重要的工作：曲速的测算与节拍跟踪。节拍是音乐在时间上的基本构成单位，通过对音乐的节拍进行跟踪，能更加有效

的检测和弦的转换与使用拍同步的方法对音频信号进行切分与平滑。

根据以上分析，本文结合 CQT 与 PCP 各自的优点，尝试提高和弦识别系统的准确率。同时，也提出一个对音乐信号进行曲速测算与节拍跟踪的算法：周期乘法法。

### 1.3 研究方法与研究内容

#### 1.3.1 研究方法

##### （1）文献研究法

研究国内外相关学者的相关文献，归纳了如今计算机和弦识别系统中的限制性，得出了引入曲速测算与节拍跟踪的重要性。在此基础上分析，设计出一个更加符合音乐理论的、人机交互更为友好的、数据显示更为直观的和弦识别系统。

##### （2）模型实证研究法

利用各种信号处理手段对音乐信号进行分析，采用不同的模型相互结合，实验各种模型对于不同风格的音乐的识别率。结果表明，周期乘法法在音乐的曲速测算与节拍跟踪任务中取得了较为准确的结果。采用 CQT 与 PCP 模型相结合的方法，对和弦的识别准确率高于基于离散傅里叶变换的 PCP 与仅利用 CQT 进行 KNN 聚类的准确性。

#### 1.3.2 研究内容

本文旨在构建一个加入了节拍跟踪算法的计算机自动和弦识别系统。首先，对计算机在音乐领域中的研究背景与国内外的相关研究进行了基本阐述。随后对音乐理论与各种信号处理方式的概念进行讨论，在此基础上，使用 Native Instruments 公司的 Alicas Keys 钢琴采样构建测试与训练所使用的和弦。首先使用基于离散傅里叶变换的 PCP 模型对和弦进行识别，对此领域的基础研究方式进行评价，实验 CQT、KNN、PCP 等不同的模型相结合的方式改进基础的 PCP 模型。最后，引入对音乐信号的曲速测算与节拍跟踪算法，对实际音乐进行和弦的识别与分析。

本文的结构主要分为下面八个部分：

第一部分：绪论

对和弦识别的理论与实际意义进行探讨，并综合国内外研究成果，明确本文的研究内容、方法与创新点。

第二部分：音乐相关概念及相关理论

简述和弦、节拍的概念，介绍信号处理方法与音乐理论中律法的关联理论，为后文的研究方法提供理论基础。

第三部分：周期乘法法

本章提出一个适用于测算歌曲曲速与对音乐信号进行节拍跟踪的算法，将测算的数据与实际结果进行比较，评判模型的性能。

#### 第四部分：基于离散傅里叶变换的音级轮廓图

实现计算机在和弦识别领域的基础研究方法，对模型进行误差分析，总结模型的优缺点与提出改进方法。

#### 第五部分：常量 $Q$ 变换

介绍常量  $Q$  变换的数学定义与实现，使用常量  $Q$  变换进行 KNN 聚类与常量  $Q$  变换与 PCP 模型相结合，对其与基础的 PCP 模型的性能。

#### 第六部分：引入节拍同步的乐段和弦分析方法

引入节拍同步算法对实际音乐片段进行分析，评价模型识别率。

#### 第七部分：结论

重新阐述本文的研究背景，根据实证分析得出结论，并指出研究的不足之处，并提出一些建议，以帮助他人更好的进行计算机和声学的讨论与研究。

## 1.4 本文创新点

本文的创新点主要体现在两个方面。首先是将 CQT 算法与 PCP 模型相结合，提高了传统 PCP 方法的识别率。其次是引入了节拍跟踪与拍同步音频帧平滑，为和弦的变换检测提供了更加符合音乐理论的方法，实现了更加友好的人机交互与更加直观的数据展示。

## 2 和弦识别与节拍跟踪概念与相关音乐理论

### 2.1 和弦与节拍的定义

#### 2.2.1 和弦的定义

使几个音组合成和音，并使这样的一些和音形成连续的进行，这就是和声。

和声对音乐作品的发展，对于加深和丰富音乐作品的表现力，有着非常重要的意义。它能使旋律获得即为多样的感情色调与色彩，特别是当同一个旋律采用了不同的和弦进行作为伴奏时，它的这种作用表现得尤为明显。



图 2.1 一个和声进行的例子

几个不同的音在同一时间的结合叫做和音。

三个、四个或五个音高和音名各不相同的音组成的和音就是和弦。<sup>[2]</sup>



图 2.2 最基本的和弦组成

如何利用计算机算法，对音乐中的和弦与和声进行识别与分析，是本文的主要研究内容。

#### 2.2.2 节拍的定义

有重音的几无重音的同样时间片段，按照一定的顺序循环重复，叫做节拍。节拍是用强弱关系来组织音乐的。用来构成节拍的每一时间片段，叫做一个单位拍。<sup>[1]</sup>

节拍的概念可以类比与计算机中的“时钟”的概念，它是一个周期性的标尺，为音



乐在时间上的组织提供基础参考。

## 2.2 和弦识别的理论基础

### 2.2.1 律法

音乐中的音符在频率上分布的规律叫做律法。律法规定了不同音高的音符频率的计算方式，是将音乐艺术与物理规律相关联的桥梁。历史上曾经出现过许多不同的律法，如纯律、五度相生律、十二平均律等<sup>[1]</sup>。现代的音乐理论一般采用十二平均律作为通用的律法，由于计算简单直观，本系统也使用十二平均律作为研究的对象。下一节将介绍十二平均律的数学表达。

### 2.2.2 十二平均律

在现代音乐理论中，各个音符的频率之间的关系，一般依照十二平均律。

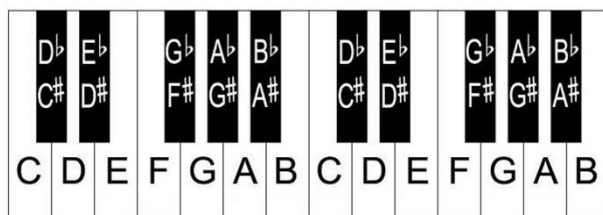


图 2.3 钢琴键盘

十二平均律基于以下规则制定：

- (1) 相隔八度的两个音频率比为 2:1
- (2) 将一个八度内的音平均分成 12 份，相邻两个音之间的频率比值相同。

基于以上规则，给出十二平均律中频率计算的数学表达式：

将一个八度内的音分为十二份，假设每个音是前一个音的  $x$  倍，标准音的频率为  $F_0$ ，易得方程：

$$x^{12}F_0 = 2F_0 \quad (2.1)$$

解得

$$x = \sqrt[12]{2} \quad (2.2)$$

设十二平均律中十二个音的下标分别  $F_0, F_1, \dots, F_{11}$ ，已知  $F_x$  的频率，由上述公式易得求  $x$  的公式为：

$$x = 12 \log_2(F_x/F_0) \quad (2.3)$$

例如，已知标准 A 的频率： $A_4 = 440\text{Hz}$ ，某个音符的频率为  $F_x = 523.25\text{Hz}$ ，求此音符：

$$x = 12 \log_2(523.25/440) \approx 3$$

比标准 A 高出三个半音，为 C<sub>4</sub>。

### 2.2.3 离散傅里叶变换

傅里叶变换在信号处理中的作用，一般是将一个时域函数转换为频域函数。一个周期性连续函数的傅里叶函数的复数形式如下所示<sup>[5]</sup>：

$$c_n = \frac{1}{2l} \int_{-l}^l f(x) e^{-\frac{n\pi x}{l}} dx \quad (2.4)$$

在现实生活中，声音由物体的振动产生。一个质点的简单受迫振动方程一般可以由一个连续的周期性函数表示。但是在数字音频领域中，一个连续的音频信号需要进行采样，转换成一个离散的函数进行储存。对音频的采样速率使用采样率表示，单位为 Hz。根据奈奎斯特-香浓采样定理，如果周期函数  $x(t)$  不包含高于  $B$  cps（次/秒）的频率，那么，一系列小于  $1/(2B)$  秒的  $x(t)$  函数值将会受到前一个周期的  $x(t)$  函数值影响。

因此  $2B$  样本/秒或更高的采样频率将能使函数不受干扰。相对的，对于一个给定的采样频率  $f_s$ ，完全重构的频带限制为  $B \leq f_s/2$ 。为了平衡计算量与信号失真，本次实验拟将音频的采样率统一为 44100Hz。

由于在计算机中存储的音频信号为离散形式，故对其进行的傅里叶变换也应该是离散的。下面给出离散傅里叶变化的数学表达式：

$$X(k) = \sum_{n=0}^{N-1} W(n)x(n)e^{-\frac{2\pi i k n}{N}} \quad (2.5)$$

其中，数组  $X(k)$  为 DFT 频谱，下标  $k$  为频率点位， $W(k)$  为窗口函数。 $N$  为 FFT 大小， $x(n)$  为输入信号。在  $x(n)$  为实数的情况下，由于每个频率点位存在一对共轭的复数结果， $X(0)$  至  $X(N/2 - 1)$  的结果即为完整的 DFT 频谱。由于仅对信号在各个频率点位的强度（振幅）进行考察，故 DFT 频谱的结果应对复数结果取模。

在对音乐信号进行处理时，通常不会将整个采样进行 DFT 处理。一般将信号切分成音频帧之后，在进行处理。一帧的长度即为 FFT 大小。为了保证每一个音频帧之间的连续性，分帧时通常会确保两帧之间存在交叉。同时，为了防止分帧之后产生能量泄露，通常对信号进行加窗。窗口函数是一个以 FFT 窗口大小为长度的函数，以本次实验中所使用的汉宁（Hanning）窗为例：

$$W(n) = \frac{1}{2} \left[ 1 + \cos \left( 2\pi \cdot \frac{n}{N-1} \right) \right] \quad (2.6)$$

汉宁（Hanning）窗可以看成是升余弦窗的一个特例，汉宁窗可以看作是 3 个矩形时间窗的频谱之和，或者说是 3 个 sinc（t）型函数之和，而括号中的两项相对于第一个谱窗向左、右各移动了  $\pi/T$ ，从而使旁瓣互相抵消，消去高频干扰和漏能。

## 2.3 节拍跟踪的理论基础

### 2.3.1 音乐中音符的组织特点

音乐中的音符一般严格按照节拍组织。在节拍中音符的不同组织方式构成了不同的节奏型。可能的音符组织的方式如图 2.4 所示：



图 2.4 音符的可能组织方式

**节拍层（Beat level）：**指示乐段中节拍的概念，在常见的 4/4 拍音乐中，通常以四分音符为一拍，每小节的第一拍为重拍，第三拍为次重拍，二、四拍为轻拍。

**分拍层（Division levels）：**将四分音符进行二次、四次或者更加多次的均分，构成分拍层。分拍层由八分音符、十六分音符等组成。

**合拍层（Multiple levels）：**节拍层的音符的组合称为合拍层。

在本实验中，将分拍层、合拍层的音符与节拍层的音符进行区分，尽可能减小分拍层与合拍层的音符对节拍层音符的判断是本次实验的关键所在。

拍按照一定的数量与形式组织成小节（Measure），组织方式通常以拍号（Signature）表示。以现代歌曲中常见的 4/4 拍为例子，它表示以四分音符为一拍，每小节由 4 拍构成。小节是乐句的基本组成。每小节的第一拍被称为下拍（Downbeat）。在本次实验中，下拍时间戳是判断和弦变换点位的主要参考标准。

曲速是指每分钟的拍数（Beats per Minute, BPM），是衡量音乐快慢的量化数据。

### 2.3.2 音乐信号的响度特征

一个音符在响度上的特征可以用 ADSR（起音：Attack，延迟：Decay，延音：Sustain，释放：Release）包络表示。以键盘乐器的演奏为例，起音是按下键盘琴键的瞬间，延迟是音符的响度增加值最大值的时间，延音是按住琴键，音符依然发出声音的时间，而释放则是音符在释放琴键之后，响度逐渐减小为零的时间。

在实际的音乐演奏之中，起音的时刻通常是谱面上所展示的音符的时刻。在音频信号的响度曲线中，一个音符或者几个音符同时发出声音的特点，通常表现为一个快速上

升的波峰，随后缓慢减小，直到下一个音符发出声音。

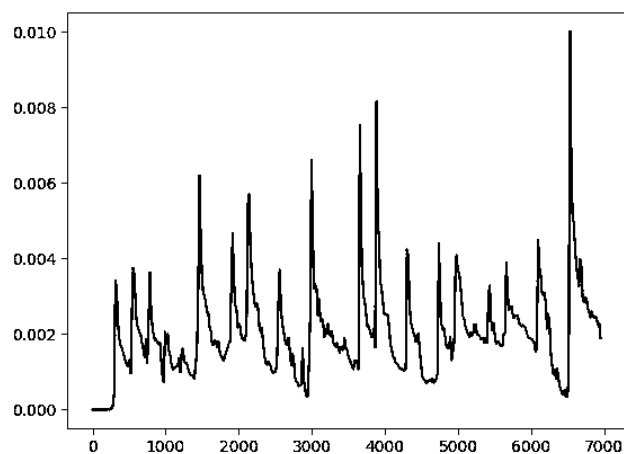


图 2.5 一个音乐片段随时间的响度变化

图 2.5 显示了一个音乐片段的响度变化。可以看出，音频的响度曲线符合上述的响度衰减特征。

### 3 周期乘数法

#### 3.1 用于曲速测算的周期乘数法

本节提出一个用于测算音乐信号曲速的算法。算法的流程如图 3.1 所示：

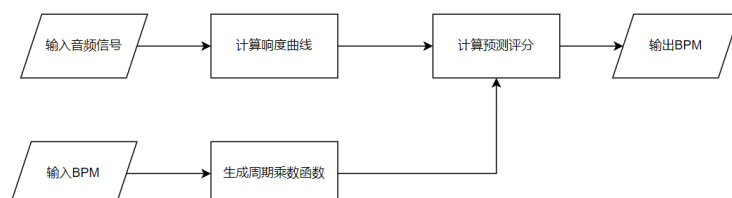


图 3.1 用于 BPM 测算的周期乘数法流程图

##### 3.1.1 算法设计

构建一个简单周期乘数函数：

$$T(t) = \cos\left(\frac{2\pi \cdot BPM}{6000(ms)} \cdot t\right) \quad (3.1)$$

音频采样的响度曲线

$$E(n) = \sum_{l=0}^{\frac{N}{2}-1} EQ(l) \cdot |X(l)| \quad (3.2)$$

其中 $EQ(l)$ 为一个均衡器（Equalizer, EQ）函数，它的数学意义是 DFT 变换结果的频率点位在能量计算中的权重，物理意义是能量谱线对某个频率点位的敏感程度。在实际音乐中，节奏组乐器（打击乐器）对节奏的指示性相比于旋律组乐器更加显著。而节奏组乐器的频率特征与旋律组乐器有着明显的区别：

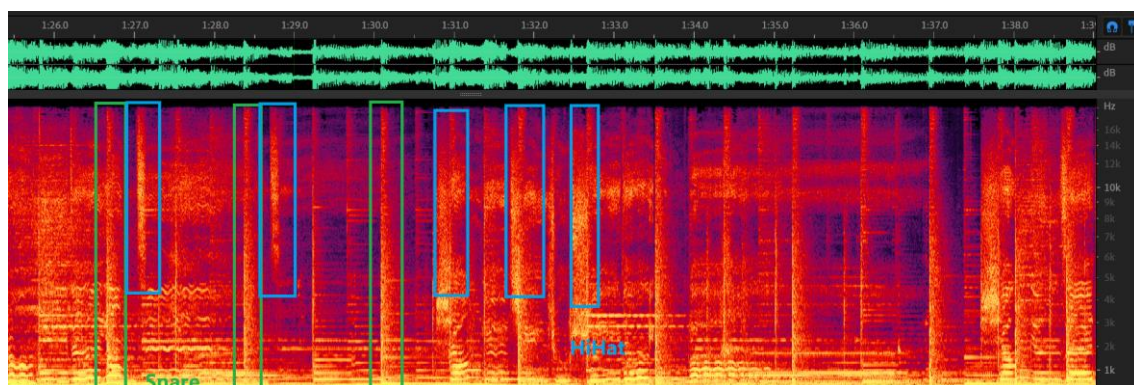


图 3.3 一个典型的节奏组乐器的频率响应特征

上图显示了一个典型的节奏组乐器（架子鼓）中各乐器的频率响应特征：

手击镲（Hi Hat）：图中蓝色框部分：集中在高频部分，中心频率 10KHz，频率范围从 5KHz 至 20KHz。

小军鼓（Snare）：图中绿色框部分，在中频与高频部分均对频谱有着较大的冲击。

合理设计均衡器函数，可以使得能量谱线对节奏组乐器的频率响应更为敏感，提高识别的准确率。

计算待测试乐段能量谱线 $E(n)$ 在当前预测 BPM 下的评分：

$$Score_{predict} = \frac{1}{F} \sum_{n=0}^F E(n) \cdot T(time(n)) \quad (3.3)$$

其中 $F$ 为乐段的帧总数。 $E(n) \cdot T(n)$ 即为周期乘数。

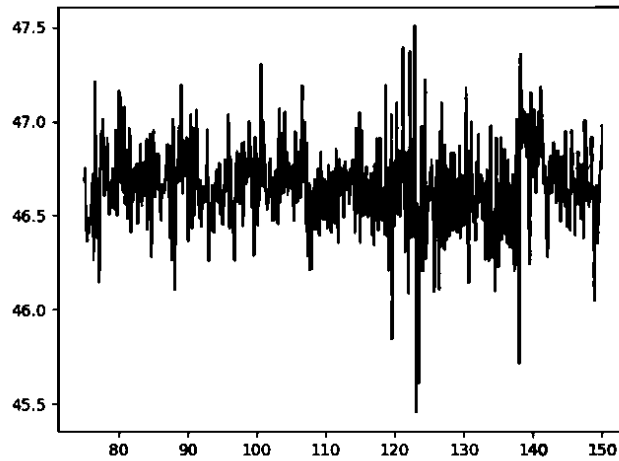


图 3.4 一个音乐片段在各 BPM 预测中的得分

由于在实际情况中，一段音乐并不是总严格按照节拍开始，也就是说，周期乘数函数的相位与音频采样的相位存在相位差。然而，忽略相位差的影响，一个周期与音频采样的节拍周期有着显著区别的周期乘数函数，预测 BPM 的评分接近于一个平均、无序的值，而周期与音频采样节拍相近或相等的周期乘数函数，预测得分总是明显大于或小于那个平均、无序的值（类比于物理中“共振”的现象），故在此取各 BPM 下的预测评分结果的方差，来反映当前音频在各预测 BPM 下的预测得分。

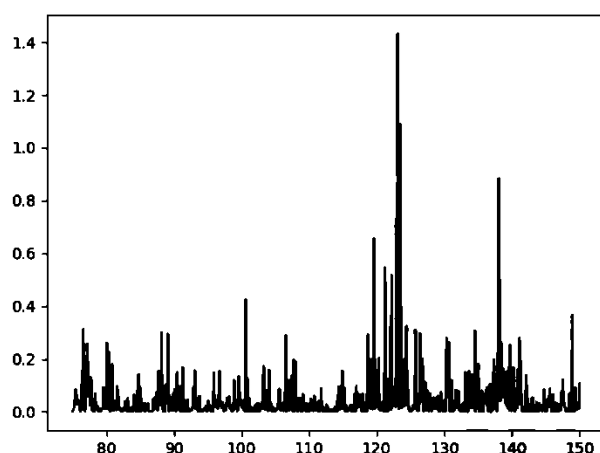


图 3.5 取方差后各预测 BPM 下的评分

### 3.1.2 算法评价

在本次实验中，使用不同风格与曲速的实际音乐对算法进行测试，部分实验结果如表 3.1 所示：

表 3.1 用于曲速检测的周期乘数法的预测结果

测试曲目	风格	实际值	预测值	偏差值
星座になればたら	日语摇滚	123	123	0
I Really Want to Stay at Your House	电子	125	125	0
言って。	日语摇滚	180	180	0
恋爱困难少女	国语流行	96	96	0
预言	国语摇滚	92	92	0
星球上的追溯诗	电子	130	130	0

在多次实验中，周期乘数法取得了基本准确的曲速预测结果。相比于利用拍点检测与动态规划的曲速检测，周期乘数法取得了更符合音乐特性与稳健性的结果。

但是，由于周期乘数法的算法特点，使得此算法的应用有着较大的限制性：

（1）无法检测变速的音乐曲速：由于对整个音乐信号采用了同一个周期乘数函数，使得此算法无法对音乐的曲速变化进行检测，也无法准确的测算变速音乐的曲速。

（2）结果标度的分辨率较低：由于此算法的原理是测算信号在不同曲速下的评分，得到最后的预测结果，故检测的曲速分度较低，对于精确到千分位或万分位的曲速值，较高的检测分度将导致计算量程指数级增长。

（3）计算量较大：由于此算法涉及到大量的迭代操作，算法的计算量十分巨大。相对于其他的曲速检测手段，此算法需要的计算时间更长。

## 3.2 用于节拍跟踪的周期乘法法

在本文 2.3.2 中，详细阐述了音乐信号的响度特征。这些特征可以用于起音检测，进而对音乐的节拍进行跟踪。本节提出一个利用音频的响度特征来对音乐进行节拍跟踪的算法，算法的流程如图 3.6 所示：

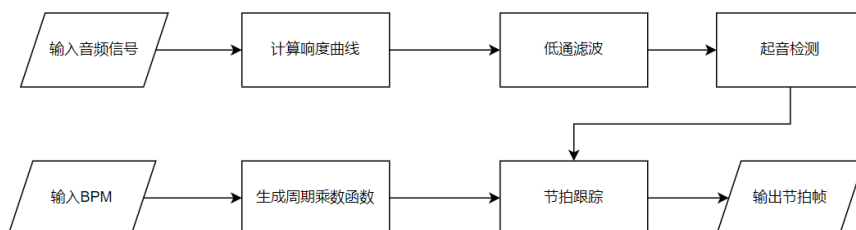


图 3.6 用于节拍跟踪的周期乘法法的算法流程图

### 3.2.1 低通滤波器与起音检测

低通滤波器的本质上是一个高斯模糊算法，高斯模糊算法在一维空间的表达式如下所示：

$$G(r) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{r^2}{2\sigma^2}} \quad (3.4)$$

由公式（3.2）得到音频的响度曲线 $E(n)$ ，对其进行低通滤波，过滤曲线中的高频信号在音符响度衰减过程中的噪音影响：

$$E'(n) = \sum_{i=0}^{F_n} E(i) \cdot G(i-n) \quad (3.5)$$

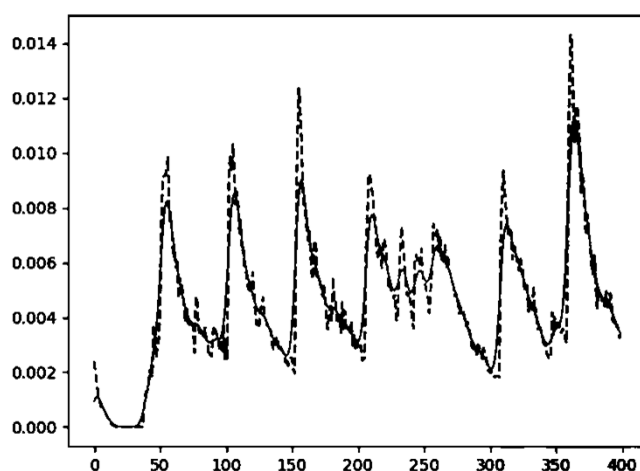


图 3.7 低通滤波后（实线）与滤波前（虚线）的响度曲线对比



滤波后的响度曲线中的极大值，可以认为是一个可能的起音点。记录所有可能的起音点得到集合 $\{A_0, A_1 \dots A_n\}$ 。

### 3.2.2 用于节拍跟踪的周期乘数法

对一段曲速固定的音乐来说，要对其进行节拍同步，只需要找出一个在拍上的起音。为了找到一个在拍上的起音，引入一个新的周期乘数函数：

$$T'(t, \varphi) = \cos\left(\frac{2\pi \cdot BPM}{6000(ms)} \cdot t + \varphi\right) \quad (3.6)$$

这个函数与函数（3.1）非常相似，唯一的不同是这个函数增加了一个参数 $\varphi$ ，它表示周期乘数函数的相位。

对与上一步求得的所有可能的起音点集合 $\{A_0, A_1 \dots A_n\}$ ，判断其中最有可能在拍上的起音点，选用的评分方法为：

$$Score_{predict} = \sum_{n=0}^F E'(n) \cdot T'((time(n), \varphi)) \quad (3.7)$$

其中，令 $\varphi = time(A_n)$ 。

此评分显示了起音点与拍点的偏差大小。值越大，则表示这个起音点与音乐节拍的偏差越小，负值的绝对值越大，则表示这个起音点与反拍的偏差越小。故此评分不取方差，认为评分最大的起音点是最有可能的一个拍点。

### 3.3.3 算法评价

此算法的误差用偏差率表示。令偏差率为：

$$\mathcal{L} = \frac{2|t_p - t|}{60000/BPM} \cdot 100\% \quad (3.8)$$

其中， $t_p$ 表示算法预测的拍点位置， $t$ 为离预测点最近的拍点位置。偏差率 0%表示预测完全准确，偏差率为 100%表示预测结果与实际节拍完全反相。

在本次实验中，使用与曲速检测算法相同的实际音乐对算法进行测试，部分实验结果如表 3.2 所示：

表 3.2 用于节拍跟踪的周期乘数法的预测结果

测试曲目	风格	$t_p$	$t$	$\mathcal{L}$
星座になれたら	日语摇滚	4,389ms	4,369ms	8.21%
I Really Want to Stay at Your House	电子	1,341ms	1,311ms	12.50%
言って。	日语摇滚	3,936ms	4,078ms	85.20%
恋爱困难少女	国语流行	10,252ms	10,201ms	17.92%
预言	国语摇滚	8,081ms	8,387ms	97.92%
星球上的追溯诗	电子	1,828ms	1,806ms	6.07%

总体来说，此算法的预测偏差接近于两个极端：接近准确与接近完全反相。在“言って。”与“预言”两首音乐中，算法受到了编曲中复杂的反拍鼓组的影响，给出了接近完全反相的结果。如何处理鼓组中反拍音符所带来的噪音影响，是优化算法的重要思路。

## 4 基于离散傅里叶变换的音级轮廓图

### 4.1 音级轮廓图的概念与求解流程

音级轮廓图由斯坦福大学的 Takuya FUJISHIMA 于 1999 年提出。这是一种根据律法原理对音频信号进行简单频率分析的方法。它是一个十二维的向量，每个维度表示当前音频帧在十二平均律中每个音的强度。<sup>[7]</sup>音级轮廓图的算法流程如图 4.1 所示：

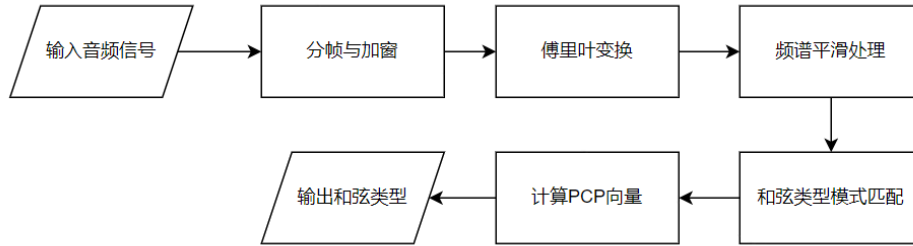


图 4.1 PCP 向量的计算与和弦类型匹配流程

在得到了 DFT 频谱 $X(k)$ 之后，可以对音频的 PCP 向量进行计算：

定义 PCP 向量

$$PCP(p) = \sum_{1 \text{ s.t. } M(l)=p} ||X(l)||^2 \quad (4.1)$$

其中， $M(l)$ 为一个将频率点位映射至 PCP 下标的矩阵，其定于如下：

$$M(l) = \begin{cases} -1 & l = 0 \\ \text{round}(12 \log_2((f_a \cdot \frac{1}{N})/f_{ref})) \text{Mod} 12 & l = 1, 2, 3, \dots, N/2 - 1 \end{cases} \quad (4.2)$$

其中， $f_{ref}$ 为标准音频率，记为 $PCP(0)$ 。 $f_a \cdot \frac{1}{N}$ 为 $X(l)$ 的频率。

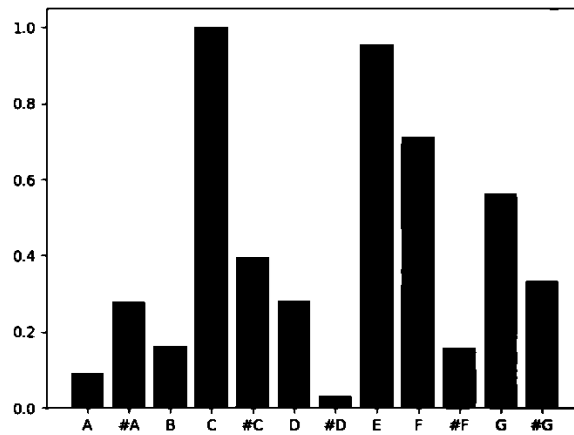


图 4.2 一个 C 大三和弦的 PCP 向量

## 4.2 和弦类型模式

PCP 向量用于进行模式匹配。和弦类型模板（Chord Type Templates, CTT）为一个十二维的向量，用于判断和弦的种类。若一个和弦是以 C 为根音的和弦，则

$$CTT_c(p) = 1$$

否则为 0.

一个 M7（大七和弦）和弦的 CTT 向量如下所示：

$$PCP_{CM7}(p) = (1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1)$$

最邻近算法下 PCP 向量在一个 CTT 中的预测得分为：

$$Score_{nearest,c} = \sum_{p=0}^{11} (T_c(p) - PCP(p))^2 \quad (4.3)$$

加权求和法下 PCP 向量在一个 CTT 中的预测得分为：

$$Score_{weighted,c} = \sum_{p=0}^{11} W_c(p) \cdot PCP(p) \quad (4.4)$$

## 4.3 算法评价

### 4.3.1 算法的识别率

本次实验使用 Native Instruments 公司的 Alicias Keys 钢琴采样构建的和弦样本对算法进行实验。将实验所使用的和弦分为两组：和弦音在相同八度内的（S1）与在不同八度内的（S2）。部分和弦的识别结果如表 4.1 所示：

表 4.1 基于 DFT 的 PCP 的和弦识别结果

输入和弦	组别	预测结果	是否准确
CMaj	S1	Cmaj	True
CMaj（低八度）		Fmaj	False
Dmaj		DMaj	True
DMaj（低八度）		Dmin	False
DMaj（低两个八度）		Fmaj	False
#FMaj7	S2	FMaj	False
#Cmin		FMaj	False
#DMaj		FMaj	False

实验数据显示，利用单纯的基于离散傅里叶变换的音级轮廓图进行和弦识别，准确率并不高，且音域越低，对模型的识别率影响就越大。实际上，在实验中，这个模型对

单一的钢琴和弦的识别率不足 10%。

这是由于傅里叶变换的特点决定的：离散傅里叶变换得到的频域离散化结果的标度是线性的，而十二平均律中，音符的频率分布却是对数的。

#### 4.3.2 误差来源与分析

基于离散傅里叶变换的音级轮廓图的识别误差主要是由于离散傅里叶变换的频域离散化结果与律法的特点不吻合造成的。

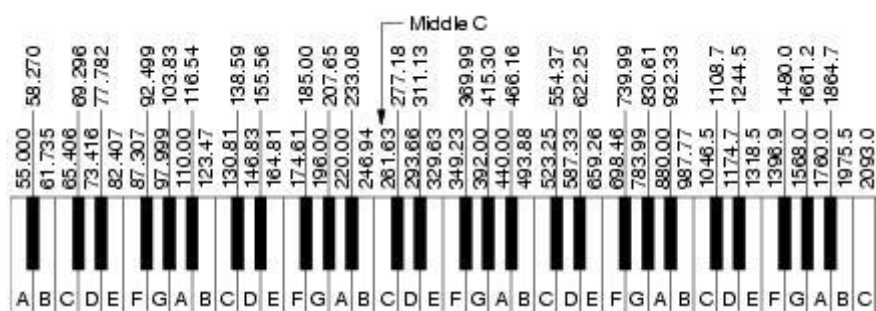


图 4.3 在十二平均律下各个部分音符的频率（A=440Hz）

以标准 A（440Hz）的频率举例，标准 A 的下半音（#G）的频率为 415.30Hz，上半音（#A）为 466.16Hz，在采样频率为 44100Hz、FFT 大小为 1024 的情况下，440Hz 周围的频率点位为 387.60Hz（第 9 点位）、430.66Hz（第 10 点位）、473.73Hz（第 11 点位）。

由于离散傅里叶变换的频域离散化结果的线性标度，导致基于离散傅里叶变换的音级轮廓图在低音区域的分辨率达不到准确识别音调高低的要求。

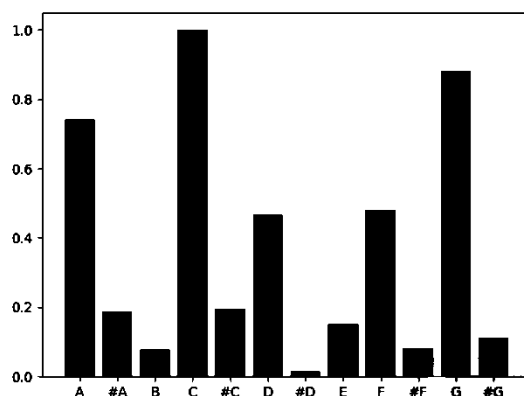


图 4.4 更低一个八度的 C 大三和弦的 PCP 向量

如图 4.4 所示，CMaj 和弦（C、E、G）中，三度音 E 的大小远小于一个和弦外音 A

的强度。而错误识别音高所带来的误差，导致一个 CMaj 和弦被错误的识别成了 FMaj 和弦。

## 5 常量 Q 宽带变换

### 5.1 常量 Q 宽带变换的定义

上文提到，在以 DFT 为频域转换方法的 PCP 中，存在着低频区域分辨率严重不足的问题，这是由于 DFT 的数学特性所决定。

在 DFT 中，由于窗口长度固定，DFT 结果中每个滤波器的频率差都是定值，为：

$$f_{res} = \frac{f_s}{N} \quad (5.1)$$

其中， $f_{res}$  是相邻滤波器之间的中心频率之差（滤波器的宽度），也就是频率的“分辨率”。 $f_s$  为音频采样的采样率， $N$  为窗口大小，或者说 FFT 大小。

再考虑到在十二平均律中乐音的频率的排列方式：

在低音区域，两个半音的频率相差非常小，如  $A1=55.0\text{Hz}$ ， $\#A1=58.27\text{Hz}$ 。

而在高音区域，两个半音之间的频率相差相对较大，如  $A6=1670.0\text{Hz}$ ， $\#A6=1864.66\text{Hz}$ 。

在传统的 DFT 中，由于窗口长度固定，无论在低频区域还是在高频区域，频率分辨率都是一致的。这就导致了频域谱在低频的分辨率严重不足，而在高频区的分辨率过高，导致音高识别产生误差。若要提高在低频区域的分辨率，就要提高 FFT 的窗口大小。但是仅仅提高窗口大小，会导致频域谱在时间轴上的分辨率的降低（每个 FFT 帧的长度增加了）。在后续的数据分析过程中，显然会降低数据的准确性。

再考虑到十二平均律中各个音符的频率特征：相邻音符的比值是一个固定值，也就是说，音高与频率的关系可以被抽象的理解为一个指数函数，在音域较高时，相邻两个音之间的频率只差较大，而在低音域，相邻两个音之间的频率相差较小。

根据律法的特点，可以考虑设计一个新的频域转换函数，令相邻两个滤波器之间的中心频率之差随着频率的提高动态变化。这样既能在低频区域获得较高的分辨率，也能在高频区域适当减小分辨率，减小在进行频率与音高之间进行转化时的误差。

这种算法就是下文将要提到的 CQT（Constant Q Transform，常量 Q 宽带变换）。

CQT 算法是 DFT 的一个改进形式，相比于 DFT 的线性标度，它使用对数标度对音频进行频域转换，这使得它更加适合于音乐信号的分析与处理。<sup>[11]</sup>

本质上，CQT 是一系列在不同频率上的滤波器。已知在十二平均律中，相邻的两个音之间的频率之比为一个定值，且两个相同音名且相差八度的音的频率比为 2 倍。根据此规则，设计一个滤波器序列：第  $k$  个滤波器的频谱宽度  $\delta f_k$  等于前一个滤波器宽度的倍数：

$$\delta f_k = 2^{1/n} \cdot \delta f_{k-1} = (2^{1/n})^k \cdot \delta f_{\min} \quad (5.2)$$

其中  $\delta f_k$  是第  $k$  个滤波器的带宽， $f_{\min}$  是最低滤波器的中心频率， $n$  是每个倍频程的滤波器数。

上文已经提到，在 DFT 中，滤波器的带宽为：

$$f_{res} = \frac{f_s}{N} \quad (5.3)$$

要改变每个滤波器的带宽，只需要改变 DFT 的窗口大小。  
定义滤波器宽度  $\delta f_k$ ， $Q$  因子

$$Q = \frac{f_k}{\delta f_k} \quad (5.4)$$

与第  $k$  个窗口的窗口长度：

$$N[k] = \frac{f_s}{\delta f_k} = \frac{f_s}{f_k} \cdot Q \quad (5.5)$$

于是，每个频率点位由上述离散傅里叶变换中的

$$\frac{2\pi k}{N}$$

变为 CQT 中的

$$\frac{2\pi Q}{N(k)}$$

同时，窗口函数也变化为：

$$W(k, n) = \frac{1}{2} \left[ 1 + \cos \left( 2\pi \cdot \frac{n}{N(k) - 1} \right) \right] \quad (5.6)$$

综合上述式子，将离散傅里叶变换的结果转换为 CQT 结果的公式成为：

$$X(k) = \frac{1}{N(k)} \sum_{n=0}^{N(k)-1} W(k, n) x(n) e^{-\frac{2i\pi Qn}{N(k)}} \quad (5.7)$$

在这里，求和之后要乘上  $1/(N(k))$  的原因是由于窗口长度发生了改变，要对每一个窗口的结果进行归一化。

经过这样的变换，传统的离散傅里叶变换中因为频率线性标度产生的低频区域分辨率不足的问题，可以由 CQT 变换的对数标度的滤波器组解决。

## 5.2 利用 CQT 结果进行 KNN 聚类

### 5.2.1 KNN 聚类

KNN (K-Nearest Neighbor, K-最临近) 算法是一种基本分类与回归方法。在本文中，



我们需要对一个音频片段的频率响应特征进行聚类，判断它属于哪个和弦类型，以及由哪一个音作为根音所构成。KNN 非常适合执行此类任务。

KNN 利用训练数据对特征向量空间进行划分，然后将划分后的空间标记为不同的类别。

对于实例  $X$ ，有给定的训练数据集：

$$T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

其中：

$$x_i \in x \subseteq R^n$$

是实例  $X_i$  的  $n$  维的特征向量。

$$y_i \in Y = \{c_1, c_2, \dots, c_k\}$$

为实例的类别。

根据给定的训练数据集，预测实例  $X$  在  $Y$  中的类别。

KNN 的执行步骤如下：

1. 确定距离测度方法。闵可夫斯基距离定义如下：

$$d_{xy} = \sqrt[p]{\sum_{k=1}^n (x_k - y_k)^p} \quad (5.8)$$

闵可夫斯基距离是一类距离的定义。其中， $p$  是可变参数。当  $p=1$  时，称为曼哈顿距离， $p=2$  时，称为欧氏距离，当  $p$  趋近于无穷时，称为切比雪夫距离。

在实际使用时，欧氏距离（ $L_2$  范数）是最常用的距离测度方法。根据以上定义，欧氏距离的定义如下：

$$d_{xy} = \sqrt{\sum_{k=1}^n (x_k - y_k)^2} \quad (5.9)$$

2. 计算实例  $X$  在给定距离测度方式下距离训练集  $T$  中所有的实例  $X_i$  的距离，并且将所有距离进行排序

3. 确定  $K$  值。在排序完成后的数据中选出  $K$  个数据，根据多数投票原则对实例  $X$  进行分类。将实例  $X$  归于离  $X$  最近的  $K$  个数据中最多的那个分类的类别。

### 5.2.2 算法实现

设 CQT 滤波器的最低中心频率  $F_{min}=32.70\text{Hz}(C_0)$ ，每个倍频程中的滤波器个数  $n = 12$ （半音），考虑 7 个八度音程内的频率范围，得到一个音频帧的 CQT 频谱：

$$CQT[k] = \frac{1}{N(k)} \sum_{n=0}^{N[k]-1} W[k, n] x[n] e^{-\frac{2i\pi Qn}{N[k]}} \quad (5.10)$$

其中：

$$k \in [0, 83], k \in \mathbb{Z}$$

CQT 频谱的结果是一个 84 维的向量，向量的每一个维度代表着待测试的音频帧在十二平均律中不同八度内各个音符的强度。将 CQT 结果作为训练集实例  $X_i$  与待聚类实例  $X$  的特征向量。计算待聚类实例  $X$  与训练集样本实例  $X_i$  的欧氏距离：

$$distance_{(X, X_i)} = \sqrt{\sum_{k=0}^{83} (CQT_{X_i}[k] - CQT_X[k])^2} \quad (5.11)$$

再按照上文提到的 KNN 聚类方法对待聚类实例  $X$  进行分类。

### 5.2.3 算法评价

使用与上文 4.3.1 中提到的和弦样本对 KNN 聚类进行实验。实验的部分结果如表 5.1 所示：

表 5.1 利用 CQT 结果进行 KNN 聚类的和弦识别结果

输入和弦	组别	预测结果	是否准确
CMaj	S1	EMaj	False
CMaj（低八度）		CMaj	True
Dmaj		DMaj	True
DMaj（低八度）		#Fmin	False
DMaj（低两个八度）		Fmin	False
#FMaj7	S2	Fmin	False
#Cmin		#Cmin	True
#DMaj		Emin	False

利用 CQT 结果进行 KNN 聚类的和弦识别率稍高于基于 DFT 的音级轮廓图的识别率。较低的音域，此算法也能正确的识别某些和弦。但是由于相邻两个滤波器之间存在频率泄露，会导致预测结果将音符的音高错误的识别为上下两个半音的音高。

### 5.2.4 误差来源于分析

常量  $Q$  变换的误差主要来自于两个相邻滤波器之间的频率泄露。

以一个由钢琴弹奏的标准 A 为例，这个音符的 CQT 频谱中，产生了#G、B 两个半音的错误识别结果：

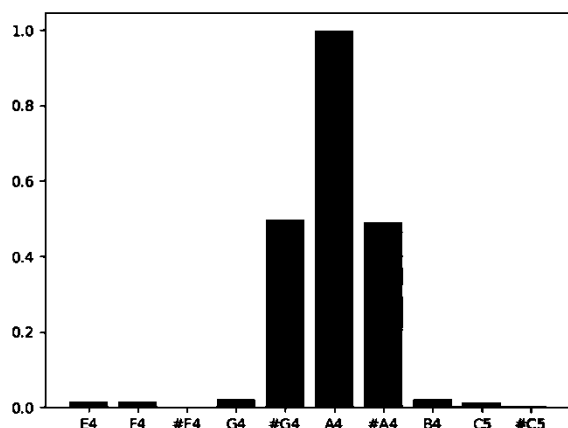


图 5.1 一个标准 A 的部分 CQT 频谱

这种错误的识别结果直接对 KNN 聚类的分类结果产生了影响。如表 5.1 所示，算法将一个 CMaj 和弦错误的识别为了 EMaj 和弦，这是由于 CMaj 和弦的三个组成音（C、E、G）与 EMaj 和弦的三个组成音（E、#G、B）存在着两个相差半音的音（G 与 #G、C 与 B）与一个相同的音（E）。这导致了大部分的错误识别结果。

### 5.3 基于常量 Q 变换的音级轮廓图

#### 5.3.1 算法实现

通过选择合适的常量 Q 变换的相关参数，可以令 CQT 频谱的频率离散化结果与十二平均律中的各音符的音高频率相吻合。这样，可以直接通过 CQT 频谱计算出音频的音级轮廓图：

$$PCP[p] = \sum_{n \bmod 12 = p} CQT[n] \quad (5.12)$$

再对计算出来的 PCP 向量进行和弦模式匹配。

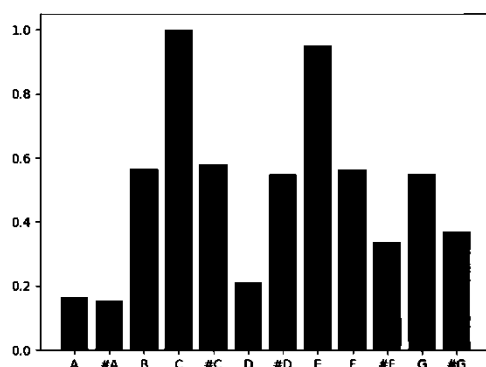


图 5.2 利用 CQT 算法得到的 PCP 向量

### 5.3.2 算法评价

使用与上文 4.3.1 中提到的和弦样本此算法进行实验。实验的部分结果如表 5.2 所示：

表 5.2 基于 CQT 的 PCP 的和弦识别结果

输入和弦	组别	预测结果	是否准确
CMaj	S1	CMaj	True
CMaj（低八度）		CMaj	True
Dmaj		DMaj	True
DMaj（低八度）		Dmaj	True
DMaj（低两个八度）		Dmaj	True
#FMaj7		#FMaj	False
#FMaj7	S2	#FMaj7	True
#Cmin		#Cmin	True
#DMaj		#Dmaj	True

基于 CQT 的 PCP，对单个钢琴的和弦识别的准确率相较于前文提到的两个算法有了明显的提升。将 CQT 降维成 PCP，可以有效的规避在 KNN 聚类中由于和弦音分布在不同的八度内所导致的识别准确率降低，在 S2 组的和弦识别之中取得了较好的识别率。而 CQT 在低频区域的分辨率提升，使得该算法在较低的音域内也能对和弦进行准确的识别。除了 S1 组中的一个七和弦被错误的识别成了三和弦之外，其余的和弦识别都取得了较为准确的结果。

## 6 引入节拍跟踪的乐段和弦识别方法

### 6.1 算法流程

结合上文提到的几种算法，可以对实际音乐片段进行和弦的识别与分析。算法的流程如图 6.1 所示：

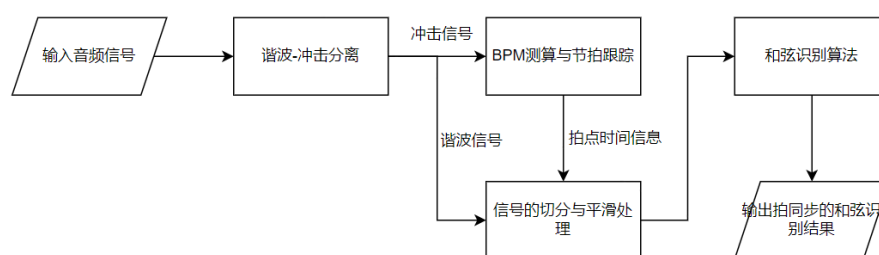


图 6.1 算法流程图

### 6.2 基于拍同步的音频切分与平滑处理

和弦的转换通常在拍上。我们可以认为一拍中的所有音符构成了这一拍的和弦。

以拍同步的方式，将一拍内的所有音频帧的频域谱信息进行平滑处理，可以显著提高信息的利用效率，一个最简单的平滑方式，就是用平均值，将一拍时间内的所有帧的频域谱平滑为一帧：

$$X[k] = \frac{1}{F_n} \sum_{n=0}^{F_n} x_n[k] \quad (6.1)$$

其中 $X[k]$ 为平滑后的帧， $F_n$ 为一拍中音频帧的总数， $x_n[k]$ 为一拍中第 $n$ 帧的频域谱。

### 6.3 算法评价

#### 6.3.1 对钢琴柱式织体的响应

使用一段钢琴的柱式和弦独奏来测试模型的性能，测试乐段的谱例如图 6.2 所示：



图 6.2 测试乐段的谱例

这是一段较为简单的谱例，但是也使用了和弦转位等手法。将音频信号输入模型，模型给出的预测结果如图 6.3 所示：

BPM = 59  
Smoothing Spectrum 1606 of 1607 (99.94 % Completed)

1	1	2	3	4	
	FMaj	-----	-----	-----	
2	1	2	3	4	
	GMaj	-----	-----	-----	
3	1	2	3	4	
	Emin	-----	-----	-----	
4	1	2	3	4	
	Amin	-----	-----	-----	
5	1	2	3	4	
	Dmin	-----	-----	-----	
6	1	2	3	4	
	GMaj	-----	-----	-----	
7	1	2	3	4	
	CMaj	-----	-----	-----	

图 6.3 模型的预测结果

在这个例子中，除了第三小节的 Emin7 和弦被错误的识别为了 Emin 和弦之外，模型给出了较为准确的识别结果。这个乐段的实际曲速为 60，模型的预测结果为 59。不过这个误差并未对模型最后的识别结果产生显著的误差。

### 6.3.2 对钢琴混合织体的响应

使用一段钢琴混合织体独奏对该算法进行测试。测试的音乐片段谱例如图 6.4 所示：



图 6.4 测试音乐片段谱例

这是一段流行音乐钢琴伴奏的谱例，和弦标记与乐谱的上方。其中的和弦组成非常复杂，在写作时大量使用了和弦的转位、装饰音等技巧，节奏型也非常复杂。对模型的性能要求非常高。

这一段谱例的和弦识别结果如图 6.5 所示：

BPM = 67  
Smoothing Spectrum 2313 of 2314 (99.96 % Completed)

1	1	2	3	4	
	Bmin	GMaj	-----	CMaj	
2	1	2	3	4	
	Cmin	Bmin	-----	-----	
3	1	2	3	4	
	DMaj	Amin	CMaj	Bmin	
4	1	2	3	4	
	-----	DMaj	Bmin	BMaj7	
5	1	2	3	4	
	BMaj	Cmin	GMaj	Cmin	
6	1	2	3	4	
	Gmin	Bmin	DMaj	-----	
7	1	2	3	4	
	EMaj	AMaj	Amin	BMaj	
8	1	2	3	4	
	-----	Emin	-----	-----	
9	1	2	3	4	
	DMaj	AMaj	DMaj7	AMaj	
10	1	2	3	4	
	EMaj7	Emin	GMaj	Emin	
11	1	2	3	4	
	DMaj	AMaj	-----	-----	
12	1	2	3	4	

图 6.5 对钢琴混合织体独奏乐段的和弦识别结果

模型对乐段进行了准确的曲速测算与节拍跟踪。然而，和弦的识别准确率却较低。

## 6.4 误差来源与分析

该模型在对十分复杂的钢琴混合织体独奏的和弦分析中，难以识别出准确的和弦信息。实际上，音乐的创作是一个十分复杂的过程，作曲家往往会在一个节拍中使用非常多的作曲手法来丰富乐曲的色彩，如使用和弦外音来对和弦进行丰富，也会采用不同的节奏型增强音乐的律动。这些变化使得要对复杂乐段中的和弦进行准确的识别变得异常困难。



## 7 结论

音乐作为人民艺术的一项重要组成部分，已经越来越受到人们的关注。越来越多的人也将音乐作为自己的爱好或者专业。在音乐鉴赏或音乐制作领域，和弦的听辨是一项非常重要的专业技能。然而，对和弦的听辨能力需要大量的时间进行训练，这是一项十分枯燥且艰难的过程。本文通过建立一个计算机自动和弦识别系统，辅助音乐爱好者或专业音乐制作人提供一个高效的辅助工具。

建立一个计算机和弦识别系统，涉及音乐理论、信号处理、计算机科学等不同领域的相关知识。通过对此问题的研究，本文也尝试探究跨专业学科交叉的方式。

本文对比分析了国内外学者的研究成果，对比分析了计算机和弦识别的多种方法，对这些方法的结果进行分析，并进行优化。通过这些实验与对比分析，本文主要取得了以下成果：

（1）提出了一个具有较高准确率与稳健性的音乐曲速测算与节拍跟踪算法，并且使用多种不同风格与曲速的实际音乐作品对算法进行测试，验证了算法性能。

（2）将常量  $Q$  变换与音级轮廓图相结合，通过结合各自的优点，使得模型取得了较好的识别率。

（3）将节拍跟踪算法用于后续对实际音乐的和弦识别工作中，使得模型更加符合音乐规律，实现了更加友好的人机交互与更加直观的数据展示。

本文在对复杂情况下的和弦识别工作上还有一些问题和不足需要进一步讨论，主要有以下几个方面：

（1）在和弦类别的识别方面，三和弦、七和弦与九和弦的识别率并不高，如何通过检测信号中频率的微小差异，来准确区分三和弦、七和弦与九和弦，是未来需要优化的方向。

（2）本文提出的模型虽然能正确判断转位和弦的原位和弦，但是却不能判断和弦是否经过转位以及如何转位。后续应通过加入低音检测的方式，识别和弦的低音与转位。

（3）周期乘数法虽然在实际音乐的曲速测算与节拍跟踪工作中取得了较高的准确率与稳健性，但是算法的计算量较大，也存在着诸多限制性。后续需要考虑对算法的优化方式。

（4）在对钢琴的复杂混合织体的和弦识别方面，本文的模型无法做到准确的识别。这是由于在实际的复杂乐段中，各种和弦外音与节拍的变化会导致大量的噪音数据输入模型，对最后的结果产生影响。如何检测这些噪音并消除，是后续需要研究的问题。

## 参考文献

- [1] 李重光. 音乐理论基础 [M].北京: 人民音乐出版社,1962:253.
- [2] 伊·杜波夫斯基, 斯·叶甫谢耶夫, 伊·斯波索宾, 符·索科洛夫. 和声学教程 [M].北京: 人民音乐出版社, 2008:435
- [3] 阿诺德·勋伯格. 作曲基本原理 [M]. 上海: 上海音乐出版社, 1984
- [4] 吴祖强. 曲式与作品分析[M]. 北京: 人民音乐出版社, 2003:408
- [5] 同济大学数学系. 高等数学 下册 第七版 [M]. 北京: 高等教育出版社, 2014: 325-326
- [6] 王万良. 人工智能导论 第五版[M]. 北京: 高等教育出版社, 2020: 211-250
- [7] Takuya FUJISHIMA, Realtime Chord Recognition of Music Sound: a System Using Common Lisp Music[J]. ICMC Proceedings,1999,464-467
- [8] Filip Korzeniowski and Gerhard Widmer. "Feature Learning for Chord Recognition: the Deep Chroma Extractor" [J]. 17th International Society for Music Information Retrieval Conference,2016
- [9] Siddharth Sigtia, Nicolas Boulanger-Lewandowski, Simon Dixon. "Audio Chord Recognition with a Hybrid Recurrent Neural Network" [J]. 16th International Society for Music Information Retrieval Conference, 2015.
- [10] Ono, Nobutaka, et al. "Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram." [J]. 2008 16th European Signal Processing Conference. IEEE, 2008.
- [11] Schoerhuber, Christian, and Anssi Klapuri. "Constant-Q transform toolbox for music processing." [J] 7th Sound and Music Computing Conference, Barcelona, Spain. 2010.
- [12] Chai, Wei, and Barry Vercoe. "Detection of Key Change in Classical Piano Music." [J]. ISMIR, pp. 468-473. 2005.
- [13] Allen, Paul E., and Roger B. Dannenberg. "Tracking musical beats in real time." [J]. ICMC. 1990.
- [14] Oliveira, Joao Lobato, Fabien Gouyon, Luis Gustavo Martins, and Luis Paulo Reis. "IBT: A Real-time Tempo and Beat Tracking System." [J]. ISMIR, pp. 291-296. 2010.
- [15] Cabral, Giordano, Jean-Pierre Briot, and François Pachet. "Impact of distance in pitch class profile computation." [J]. Proceedings of the Brazilian Symposium on Computer Music, pp. 319-324. 2005.
- [16] 闫志勇, 关欣, 李镡. 基于 SVM 和增强型 PCP 特征的和弦识别[J]. 计算机工程,

2014, 40(7):170-173.

## 致 谢