|  |
| --- |
| **成绩** |
|  |



中国农业大学

实验报告

（2024 -2025 学年 秋 季学期）

**报告题目：** 语音数字信号识别

**课程名称：** 数字信号处理实验

**任课教师：** 安冬、孟岩

**班　　级：** 智能221

**学　　号：** 2022308130412

**姓　　名：** 何卓越

目录

[语音数字信号识别 2](#_Toc8443)

[一、 实验目的 2](#_Toc27230)

[二、 实验设计 2](#_Toc28942)

[2.1 MFCC特征提取 2](#_Toc31333)

[2.1.1梅尔刻度（Mel Scale） 2](#_Toc29791)

[2.1.2临界带（Critical Band）和 mel 滤波器 3](#_Toc14821)

[2.1.3 MFCC 的计算过程 3](#_Toc11942)

[2.2 DTW算法 3](#_Toc8535)

[三、 实验过程、结果与分析 4](#_Toc13205)

[四、 实验扩展：连续数字信号识别 6](#_Toc26203)

语音数字信号识别

1. **实验目的**

实现独立的语音数字信号识别，尝试连续的语音数字信号识别。

1. **实验设计**

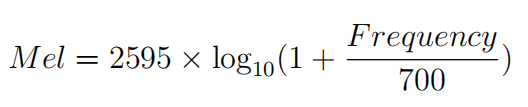
**2.1 MFCC特征提取**

梅尔频率倒谱系数（Mel Frequency Cepstral Coefficients，简称 MFCC）是语音信

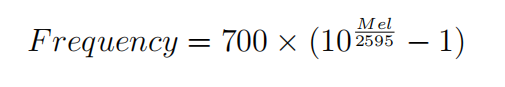
号处理中广泛应用的一种特征提取方法。它通过模拟人类听觉系统的感知方式，对语音信号进行分析和处理，从而获取更能反映语音本质特征的参数。以下简单介绍 MFCC 和实现 MFCC 特征提取过程。

**2.1.1梅尔刻度（Mel Scale）**

人耳对频率的感知是 ff 线性的，低频部分的分辨率较高，而高频部分的分辨率较低。梅尔刻度正是用来模拟这种非线性的感知方式。将实际频率转换为梅尔刻度的公式如下：



从梅尔刻度转换回实际频率的公式为：



梅尔刻度的引入，使得频率的变化更符合人耳的听觉特性，有效地模拟了听觉系统对声音的处理过程。

**2.1.2临界带（Critical Band）和 mel 滤波器**

人耳对声音的感知不仅与频率有关，还与频率的分布密切相关。临界带将频率划分为若干个频带，每个频带对应一个滤波器。这些滤波器的集合被称为 Mel 滤波器组（Melfilter bank）。研究表明，人耳对 200Hz 到 5kHz 范围内的语音信号最为敏感，这一频段对语音的清晰度有重要影响。

当两个不同响度的声音同时存在时，响度较大的声音会掩蔽响度较小的声音，这种现象被称为掩蔽效应。低频声音在内耳蜗基底膜上的行波传递距离大于高频声音，因此低频声音更容易掩蔽高频声音。基于这一特性，Mel 滤波器组设计为从低频到高频按临界带宽的大小由密到疏排列，对输入信号进行带通滤波。每个滤波器输出的信号能量作为基本特征，进一步处理后可作为语音识别的输入特征。

**2.1.3 MFCC 的计算过程**

MFCC 的计算过程包括以下几个步骤：

具体实现过程请查看仿真代码中的 mfcc.ipynb ，代码仅作为展示mfcc的具体实现过程，在主程序中使用的mfcc函数并非mfcc.ipynb中的。

1. 预加重（Pre-emphasis）：通过一个高通滤波器增强高频部分，补偿语音信号中高频成分的衰减。

2. 分帧（Framing）：将语音信号分成若干帧，每帧通常为 20-40 毫秒，并且相邻帧之间有一定的重叠。

3. 加窗（Windowing）：对每一帧信号乘以一个窗口函数（如汉明窗），以减少帧与

帧之间的不连续性。

4. 傅里叶变换（FFT）：将每一帧信号从时域转换到频域，得到每帧信号的频谱。

5. Mel 滤波器组（Mel Filter Bank）：将频谱通过 Mel 滤波器组，得到每个滤波器的能量。

6. 取对数（Logarithm）：对滤波器输出的能量取对数，模拟人耳对声音强度的感知。

7. 离散余弦变换（DCT）：对对数能量进行离散余弦变换，得到 MFCC 系数。

**2.2 DTW算法**

针对独立的语音数字信号识别，本实验采用DTW算法实现，接下来主要介绍DTW算法的主要实现步骤。

动态时间规整(DTW, Dyanmic Time Warping) 本质上是一种简单的动态规划算法，它可以分为两个步骤，一个是计算两种模式的帧之间的距离，即得到帧匹配距离矩阵；另一个是在帧匹配距离矩阵中找到最优路径。算法步骤如下：

首先，使用欧几里德距离 (Euclidean distance) 初始化成本矩阵 (cost matrix)。

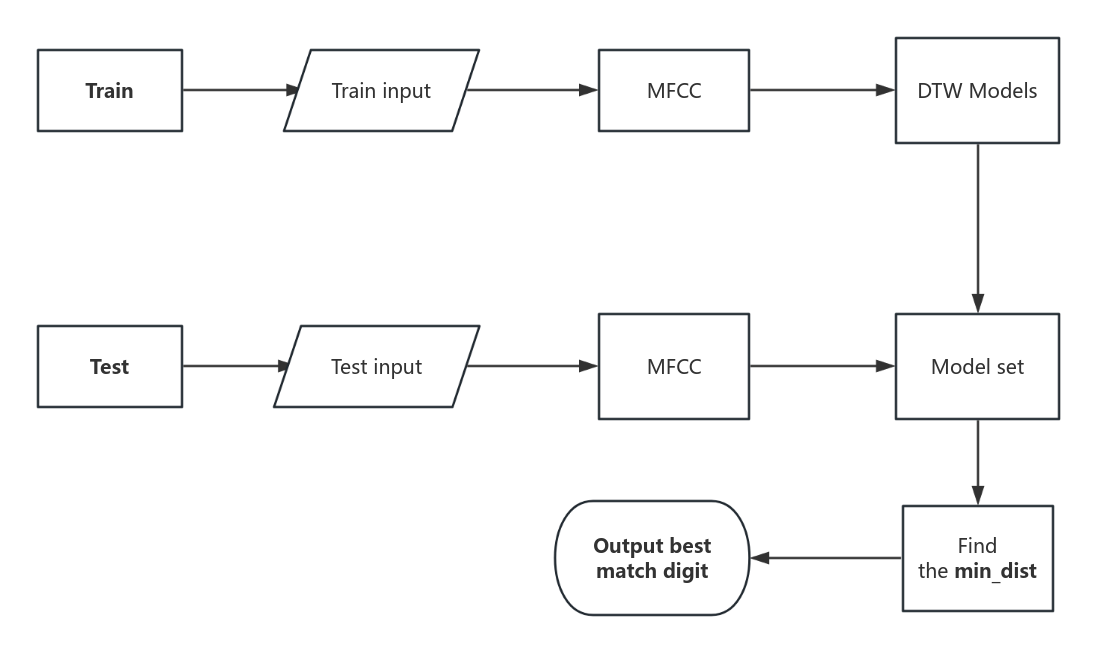
其次，使用动态规划计算成本矩阵，将每个数据的向量与模板向量进行比较，动态转移方程为



第三，计算规整路径 (warp path)，最小距离为标准化距离。将第一个训练序列设置为标准 (template)，合并所有训练序列并求其平均值，返回 0-9 位数字的模型。

测试过程中，使用 DTW 衡量每条音频到 0-9 这十个模板的”距离“，并选择模板距离最小的数字作为音频的预测值。

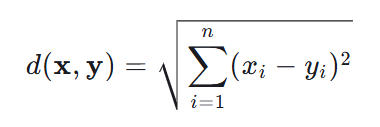
流程图如下：



具体的函数内容如下：

函数的关键在于两段 MFCC 序列的动态规划矩阵和规整路径的计算。令输入的两段 MFCC 序列分别为 M1 和 M2，M1\_len 和 M2\_len 表示各自的长度，则 cost matrix 的大小为 M1\_len \* M2\_len，先用欧式距离对其进行初始化，再根据转移式计算 cost matrix。

欧式距离的计算函数：



计算 warp path，MFCC 序列长度为 1 时的路径可以单独分情况讨论，最小代价的 path\_1 和 path\_2 即为所求，最后返回数组 path。

1. **实验过程、结果与分析**

本次实验共有两种数据集，分为中文和英文，实验结果如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 数据集 | Train | Test | Accuracy |
| 中文 | 280\*10 | 4\*10 | 0.15 |
| 英文 | 16\*10 | 4\*10 | 0.62 |

根据以上结果显示，模型在英文数据集上的表现明显优于中文数据集，可见中文数字的识别难度还是很大的，简单地利用现有的基本算法和数据集，很难识别出正确的数字。

除了DTW算法，实验还尝试了GMM和HMM算法，仅仅是尝试这两个算法，并未对这两个算法进行深度的理解分析，所以这里就直接附上实验结果：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模型 | 数据集 | Train | Test | Accuracy |
| GMM | 中文 | 280\*10 | 4\*10 | 0.100 |
| 英文 | 16\*10 | 4\*10 | 0.950 |
| HMM | 中文 | 280\*10 | 4\*10 | 0.200 |
| 英文 | 16\*10 | 4\*10 | 0.875 |

根据实验结果，我们可以进行以下分析与讨论：

1.模型在英文数据集上的表现明显优于中文数据集。这可能是由于英文数字的发音相对简单，且在语音识别领域已有较多的研究和优化，因此现有的算法和数据集能够更好地处理英文数字的识别。而中文数字的发音复杂多样，且受到方言、口音等因素的影响较大，导致识别难度增加。

2.DTW（动态时间规整）算法在中文数据集上的表现较差，准确率仅为0.15。这可能是因为DTW算法对于时间序列的对齐要求较高，而中文数字的发音变化较大，导致时间序列的对齐效果不佳。相比之下，GMM（高斯混合模型）和HMM（隐马尔可夫模型）在中文数据集上的表现略好一些，但仍然远低于英文数据集。

3.GMM和HMM算法在英文数据集上的表现较好，准确率分别为0.950和0.875。这表明这两种算法在处理英文数字识别时具有一定的优势。然而，由于实验中并未对这两种算法进行深度的理解分析，因此无法确定其具体的优势所在。

4.实验结果表明，简单地利用现有的基本算法和数据集，很难实现对中文数字的准确识别。为了提高中文数字识别的性能，可以尝试以下方法：

增加训练数据量：通过收集更多的中文数字语音样本，可以提高模型的训练效果。

特征提取优化：针对中文数字的特点，设计更合适的特征提取方法，以提高模型的区分能力。

算法改进：尝试使用更先进的语音识别算法，如深度学习方法，以提高模型的识别性能。

数据增强：通过对现有数据进行变换、合成等方式，生成更多的训练样本，以提高模型的泛化能力。

1. **实验扩展：连续数字信号识别**

借助网上数据集和代码，本实验搭建了以下内容所对应的模型：

1.音频信号特征提取

我们对音频信号提取 mfcc 特征，作为 VAD 和机器学习的输入

2.语音活动检测（VAD）

语音活动检测（VAD）用于从音频信号中区分语音和非语音段，以得到数字语音段

3.卷积神经网络建模与训练

卷积神经网络用于对提取的 MFCC 特征进行分类，使用数据集进行训练

4.输入信号识别

系统对输入的音频信号进行下列处理 ：

a. 使用 VAD 算法将输入音频分为多个语音段，每个段落对应一个数字语音

b. 对每段音频提取 mfcc 特征

c. 将每段音频的 mfcc 特征输入神经网络进行推理得到对应的结果

其中，

**语音活动检测（Voice Activity Detection，简称 VAD）**是一种用于识别语音信号中的语音片段和非语音片段的技术。其主要目的是在连续的音频信号中区分出哪些部分是 包含语音活动的，从而过滤掉背景噪声和静音部分，提高后续处理的效率和准确性。VAD 算法广泛应用于各种语音处理系统中，包括语音识别、语音增强、语音编码和语音合成等。通过准确地检测出语音活动段，可以减少处理不必要的无用信息，从而提高系统的性能和可靠性。

**基于卷积神经网络 (CNN) 的语音识别** ：

我们的语音识别系统采用基于卷积神经网络（CNN）的语音识别系统的实现。该系统主要包括数据预处理、特征提取、模型构建、训练和推理等步骤。以下是对每个步骤的具体描述。

数据集处理和特征提取

数据集来源于网络，将随机其分成训练集，验证集和测试集。音频文件通过 scipy.io.wavfile 库读取，并转换为适当的采样率。之后，使用python\_speech\_features 库提取 MFCC（Mel 频率倒谱系数）特征。这些特征能够有效地表征音频信号的频谱特性。

特征提取是将音频信号转换为模型能够处理的特征向量的过程。

在本系统中，我们采用了 MFCC、以及其一阶和二阶差分特征。具体步骤如下：

1. 从音频信号中提取 MFCC 特征。

2. 计算 MFCC 特征的一阶差分和二阶差分。

3. 将 MFCC 特征及其差分特征拼接在一起，形成最终的特征矩阵。

4. 对特征矩阵进行截取或填充，使其形状一致。

5. 将特征矩阵转置并添加空维度，以符合 CNN 的输入要求。

模型构建

在模型构建部分，我们设计了一个卷积神经网络来处理音频特征。模型主要包括以

下几层：

• 卷积层和批归一化层：网络包含多个卷积层，每层之后接一个批归一化层，以加

速收敛和提高模型稳定性。

• 激活函数：每个卷积层后应用 ReLU 激活函数，以引入非线性。

• 全连接层：卷积层输出的特征经过全连接层进行分类。

• Dropout 层：在全连接层后添加 Dropout 层，以防止过拟合。

模型训练

模型训练过程使用 AdamW 优化器和交叉熵损失函数。为了提高模型性能和防止过

拟合，训练过程中还采用了以下策略：

• 学习率调度：使用 ReduceLROnPlateau 调度器，根据验证集精度自动调整学习率。

• L2 正则化：在损失函数中添加 L2 正则化项。

• 早期停止：根据验证集精度保存最佳模型。

每个训练周期中，模型会在训练集上进行前向传播和反向传播，更新权重参数。之

后，在验证集上评估模型性能，并根据验证结果调整学习率和保存最佳模型。

模型推理

模型推理包括在测试集上评估模型和对新音频进行识别。推理过程如下：

1. 加载预训练模型权重。

2. 将测试数据通过 DataLoader 加载到模型中进行推理。

3. 计算模型的预测精度，并输出预测结果。

推理过程中，模型会根据输入特征进行前向传播，输出预测的类别标签，并与真实

标签进行比较，计算分类准确率。

实验结果：

输入“2\_5\_6\_7\_8\_9”,输出[2, 5, 8, 7, 8, 5]，准确度为66.7%。

关于连续数字信号识别的详细内容见代码和readme.md文件。