基于顺序统计滤波的实时语音端点检测算法

郭丽惠 [1] 何 昕 [2] 张亚昕 [2] 吕 岳 [1]

摘 要 针对嵌入式语音识别系统, 提出了一种高效的实时语音端点检测算法. 算法以子带频谱熵为语音/噪声的区分特征, 首先将每帧语音的频谱划分成若干个子带, 计算出每个子带的频谱熵, 然后把相继若干帧的子带频谱熵经过一组顺序统计滤波器获得每帧的频谱熵, 根据频谱熵的值对输入的语音进行分类. 实验结果表明, 该算法能够有效地区分语音和噪声, 可以显著地提高语音识别系统的性能. 在不同的噪声环境和信噪比条件下具有鲁棒性. 此外, 本文提出的算法计算代价小, 简单易实现, 适合实时嵌入式语音识别系统的应用.

关键词 语音端点检测, 顺序统计滤波, 子带频谱熵, 语音识别
中图分类号 TP18

An Order Statistics Filtering-based Real-time Voice Activity Detection Algorithm

GUO Li-Hui[1] HE Xin[2] ZHANG Ya-Xin[2] LV Yue[1]

Abstract In this paper, we propose an effective real-time voice activity detection algorithm. It makes use of the subband spectral entropy as the speech/noise discrimination feature. The speech spectrum is divided into several subbands at first. Then, the spectral entropy of each subband is estimated. We apply order statistics filters (OSF) to a sequence of the subband entropies to obtain the spectral entropy of each frame. The speech/noise classification is based on the spectral entropy. The experimental results show that the proposed algorithm can distinguish speech from noise effectively and improve the performance of automatic speech recognition (ASR) system significantly. It is proved to be robust under various noisy environments and SNR conditions. Moreover, the proposed algorithm is of low computational complexity which is suitable for embedded ASR system application.

Key words Voice activity detection, order statistics filtering, subband spectrum entropy, speech recognition

语音端点检测算法也称语音/噪声分类算法是自动语音识别系统中的一个重要模块. 对语音识别系统而言, 语音中的非语音帧 (包括静音和噪声) 只携带着冗余和干扰信息. 在实际应用中, 一般将检测到的非语音帧丢弃, 这对嵌入式的语音识别系统有以下好处:

1) 将非语音帧丢弃而不送到后端的识别器, 可以减少后端识别器的计算量. 在嵌入式语音识别系统中, 如手机、PDA (Personal digital assistant) 等, 可以降低系统的响应时间, 提高系统的实时性.

2) 在分布式语音识别系统中, 只传送语音帧可以明显地减少传输的数据量.

3) 减少由于大量的非语音帧的特征被传送到后端识别器而引起的插入错误[1].

因此, 语音端点检测在嵌入式语音识别系统中起着重要的作用. 迄今为止, 大多数的算法在高信噪比的环境中可以获得较好的正确率, 但在低信噪比的条件下算法的性能会大幅度降低. 当然, 部分算法在低信噪比环境中可以保持稳定的性能[2]. 其缺点是计算复杂度太大, 不适合嵌入式语音识别系统的应用. 为了解决这一问题, 我们设计了一个高效的检测算法, 使其在各种环境和信噪比条件中都能保持较好的性能, 而且能够满足嵌入式系统的实时性和低功耗的要求.

通常, 我们可以将语音端点检测算法分为两大类: 1) 基于阈值的方法[3−5], 这类算法的特点是先提取出每帧语音信号的特征, 然后将语音特征值与预先设定的阈值比较, 从而得到语音/噪声的分类结果; 2) 基于模型的方法[6], 这类算法需要事先估计语音/噪声模型的参数. 语音/噪声的分类过程类似模型匹配过程. 基于模型的方法需要大量数据来训练模型的参数, 由于嵌入式语音识别系统工作在各种环境、不同信噪比条件下, 通常很难准确地估计模型的参数, 尤其是, 保存模型参数需要消耗系统大量的存储空间, 嵌入式系统的集成度高等特点限制了它的应用. 所以, 基于阈值的方法更适合嵌入式语音识别系统.

短时能量是语音端点检测算法中最常用的特征[3], 它在高信噪比环境中可以有效地区分出语音和噪声, 但是大量的实验结果显示, 基于短时能量的

收稿日期 2006-11-21 收修改稿日期 2007-05-09
Received November 21, 2006; in revised form May 9, 2007
国家自然科学基金 (60475006), 教育部新世纪优秀人才支持计划 (NCET-05-0430) 资助
Supported by National Natural Science Foundation of China (60475006), Program for New Century Excellent Talents in University (NCET-05-0430)
1. 华东师范大学计算机科学技术系 上海 200062 2. 摩托罗拉中国研究中心 上海 200041
1. Department of Computer Science and Technology, East China Normal University, Shanghai 200062 2. Motorola China Research Center, Shanghai 200041
DOI: 10.3724/SP.J.1004.2008.00419

方法在低信噪比和非平稳噪声环境中, 其性能明显下降. Shen[4] 最早提出将信息熵用于语音/噪声分类. 人的发音和噪声的差异可以从它们的频谱熵表现出来. 实验结果表明, 基于语音频谱熵的算法在低信噪比环境下胜过基于能量的方法. 随后许多学者改进了基于频谱熵的方法[5, 7].

本文提出了一种基于顺序统计滤波的语音端点检测算法, 先将每帧语音划分成若干子带, 根据 Xu[5] 设计的熵函数计算每个子带的频谱熵. 由于在非平稳的噪声环境中, 频谱熵轮廓曲线的波动比较大, 不利于阈值的选择. 因此, 我们将子带频谱熵经过一组顺序统计滤波器进行平滑处理. 在平滑滤波过程中, 用到了前后 L 帧的子带频谱熵的信息, 大大提高了算法的检测精度. 在实验中, 我们将本文提出的算法添加到 ETSI 发布的 Mel 倒谱系数标准前端[8] 中, 并将实验结果与没有语音端点检测算法的结果比较, 可以发现本文提出的算法显著地提高了系统的识别率. 我们还将该算法与最近发表的两个语音检测算法比较, 并给出比较结果. 这两个算法分别基于短时能量特征和频谱熵特征, 是语音端点检测算法中比较有代表性的.

1 传统频谱熵函数的定义

在传统的基于频谱信息熵的算法中, 首先将每帧的语音信号经过快速傅立叶变换 (FFT) 得到它在功率谱上的 N_{FFT} 个点 $Y_i\,(0 \le i \le N_{FFT})$, 则每个点在频谱域上的概率密度可用式 (1) 表示

$$p_i = \frac{Y_i}{\sum\limits_{k=0}^{N_{FFT}-1} Y_k} \tag{1}$$

相应信号在频谱域上的熵函数定义为

$$H = -\sum_{k=0}^{N_{FFT}-1} p_k \log_2(p_k) \tag{2}$$

频谱熵的值和 Y_i 的分布有关, Y_i 分布曲线的变化越平缓, 频谱熵的值就越大, 根据信息熵的原理, 其包含的信息量就越大. 图 1 比较了静音、语音和噪声的幅度谱分布, 从中我们可以看到图 1(b) 中语音帧的幅度谱曲线比较光滑, 幅度谱值的变化较小, 对应 H 的值就较大.

当 H 的值大于事先设定的阈值时, 则将这一帧判定为语音帧, 否则为非语音帧. 从上面两式可以发现, 基于信息熵的分类算法简单、计算复杂度低, 容易实现. 但是, 基于信息熵的传统算法只考虑当前帧的频谱信息, 在非平稳的噪声环境下噪声频谱信息熵波动很大, 这给阈值的选择带来了困难. 本文提出的语音/非语音分类算法将前后 L 帧的子带频谱熵通过顺序统计滤波器平滑处理, 克服了传统基于频谱熵算法的缺点.

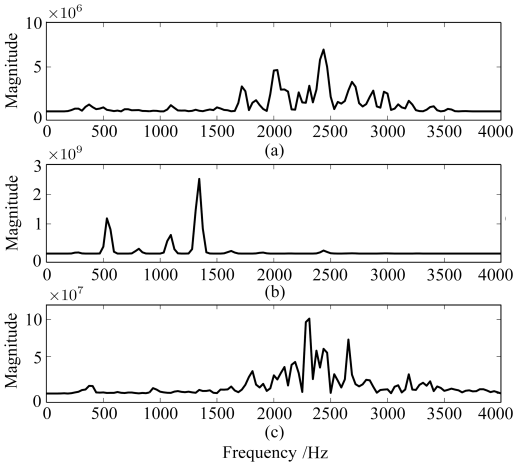


图 1 幅度谱 ((a) 静音; (b) 语音; (c) 噪声)

Fig. 1 Spectrum distributions

((a) Silence signal; (b) Speech signal; (c) Noise signal)

2 改进的语音端点检测算法

2.1 子带频谱熵估计

每帧信号经过快速傅立叶变换后得到功率谱 $Y_i\,(0 \le i \le N_{FFT})$, 把频域上的 N_{FFT} 个点划分成 K 个互不重叠的频段, 称为子带 (Subband). 因为某些环境中的噪声就集中在某个子带上, 子带方法在窄带噪声环境中可以提高算法的准确率. 根据式 (3) 计算出第 l 帧频谱域上每个点的概率

$$p_l[k,i] = \frac{(Y_i + Q)}{\sum\limits_{j=m_k}^{m_{k+1}-1} (Y_j + Q)} \tag{3}$$

$$m_k = \lfloor \frac{N_{FFT}}{K} k \rfloor (0 \le k \le K-1, m_k \le i \le m_{k+1}-1)$$

其中, Y_i 是第 k 个子带上的点, Q 是一个大的正数, 加上 Q 的目的是为了使相同信噪比环境中各种噪声信号的频谱熵值比较接近, 从而可以更容易区分出语音和噪声[5]. 实验中 Q 的值取 10^6.

根据信息熵的定义, 第 l 帧的第 k 个子带的频谱熵的值为

$$E_s[l,k] = \sum_{i=m_k}^{m_{k+1}-1} p_l[k,i] \log_2(p_l[k,i]) \tag{4}$$
$$(0 \le k \le K-1)$$

根据信息熵的原理, 当某些环境中的噪声信号比较有规律时, 分类器的准确性就会受到影响. 因此, 在计算当前帧的频谱熵时, 滤波器用到了前后 L 帧的信息. 算法中采用一组顺序统计滤波器分别对各个子带的频谱熵进行平滑处理.

2.2　顺序统计滤波

顺序统计滤波器最早用在图像处理中, 通常用来检测图像的边缘. 含有 n 个信号的序列 $X_1, X_2, \cdots, X_n$, 先将 X_i 按升序排序, 即 $X_{(1)} \le X_{(2)} \le \cdots \le X_{(n)}$. $X_{(i)}$ 就是 n 个输入信号的顺序统计, n 是滤波器的长度. 顺序统计滤波器的输出是 $X_{(i)}$ 的线性组合

$$Y = a_1 X_{(1)} + a_2 X_{(2)} + \cdots + a_n X_{(n)} \tag{5}$$

系数 a_i 决定了滤波器的特性. 比如, 中值滤波器的参数为

$$a_i = \begin{cases} 1 & i = (n+1)/2 \\ 0 & \text{否则} \end{cases} \tag{6}$$

而线性均值滤波的参数为 $a_i = 1/n$, 顺序统计滤波器的设计可参考文献 [3], Ramirze 等人将顺序统计滤波器用在语音端点检测中. 实验结果表明, 顺序统计滤波器可以提高算法的准确率. 由于 Ramirze 的算法是基于能量特征, 在实际应用中基于能量算法的缺点就不可避免会碰到. 本文提出的算法则可以克服这些缺点.

算法中每个子带的顺序统计滤波器对一组长度为 L 的子带信息熵 $E_s[l-N,k], \cdots, E_s[l,k], \cdots, E_s[l+N,k]$ 进行滤波. l 是当前分析的语音帧, 语音最初 N 帧假设是纯噪声, 用来估计噪声参数初始化阈值. 将这组子带信息熵按升序排序, $E_{s(h)}[l,k]$ 是 $E_s[l-N,k], \cdots, E_s[l,k], \cdots, E_s[l+N,k]$ 中的第 h 个最大值. 经过滤波平滑处理后的第 l 帧的第 k 个子带的信息熵定义如下

$$E_h[l,k] = (1-\lambda)E_{s(h)}[l,k] + \lambda E_{s(h+1)}[l,k] \tag{7}$$
$$(0 \le k \le K-1)$$

其中 $h = \lfloor \lambda L \rfloor (0 < \lambda < 1, L = 2N+1)$. λ 称为顺序统计滤波器的采样分位数, λ 满足高斯分布[3]. 为了提高算法语音检测的正确率, 根据实验结果 λ 取 0.9. 根据下式我们可以计算出第 l 帧的频谱信息熵:

$$H_l = -\frac{1}{K}\sum_{k=0}^{K-1} E_h[l,k] \tag{8}$$

2.3　语音/噪声分类

通过前面介绍的子带频谱熵估计和顺序统计滤波之后, 每帧的信号可以得到一个频谱熵 H_l. 当 H_l 的值大于事先设定的阈值时, 将第 l 帧判定为语音帧, 否则判为非语音帧. 阈值 T 的定义如下

$$Avg = -\frac{1}{K}\sum_{k=0}^{K-1} E_m[k] \tag{9}$$

$$T = \beta Avg + \theta$$

其中, $E_m[k]$ 是 $E_s[0,k], \cdots, E_s[N-1,k]$ 的中值, Avg 是输入信号最开始 N 帧的噪声估计. 根据实验结果选择 $\beta = 1.01, \theta = 0.1$.

图 2 给出了一段语音的波形以及它的短时能量特征曲线和顺序统计滤波之后的频谱熵曲线. 通过比较图 2(b) 和 (c), 可以看见经过顺序统计滤波之后的频谱熵的波形在非语音段时的变换比较平缓, 而当信号由非语音变化到语音时, 图 2(c) 的波形变化更为明显. 这使得阈值的选择更加容易, 而且在非语音段时不容易出现误判 (将噪声分类成语音) 的情况. 因此, 本文介绍的算法在性能上更加优越. 根据式 (8) 选择的阈值进行分类后的结果如图 3 所示. 我们可以看到该算法精确地标出了语音/噪声的分界点.

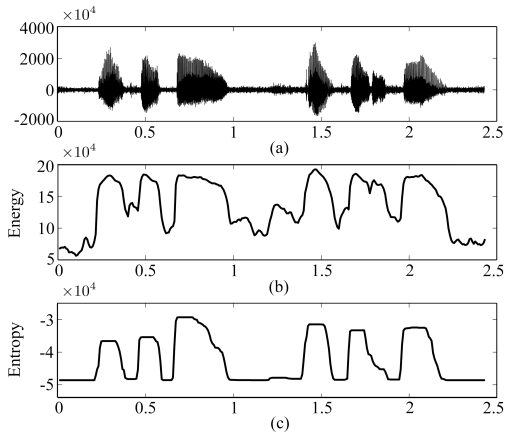


图 2　(a) 原始语音波形, 信噪比 10 dB; (b) 短时对数能量曲线; (c) 顺序统计滤波后的频谱熵曲线

Fig. 2　(a) Speech waveform; (b) Short-time energy; (c) Spectral entropy after order statistics filter processing

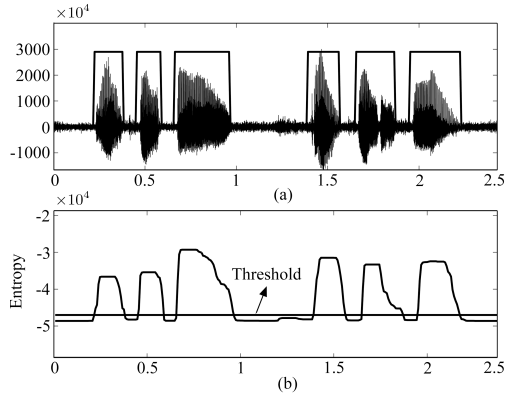


图 3　(a) 语音/噪声分类结果; (b) 语音信号的频谱熵曲线

Fig. 3　(a) Speech/noise classification; (b) The spectrum entropy

3 实验评估

评估语音端点检测算法性能的方法有很多, 如语音/噪声分类正确率、语音检测错误率和算法对语音识别系统性能的影响等. 评估算法的语音/噪声分类正确率或语音检测错误率, 需要预先人工标注定数量的语音, 将检测的结果和人工标注结果进行比较. 在实际应用中, 这两种方法一般用在因特网上的语音传输或者非连续的语音传输中. 由于语音识别系统的性能与识别器中静音模型也有很大关系, 语音/噪声分类正确率很高或语音检测错误率很低并不意味着算法用在语音识别系统中也会获得好的性能. 本节的实验结果都是在后端模型参数相同的前提下获得的. 本文的算法是针对嵌入式语音识别系统设计的, 提高系统的识别精度而不大幅度增加算法的复杂度是我们设计该算法的目的. 因此, 实验中我们主要考虑算法对语音识别系统性能的影响. 由于语音/噪声分类的正确率可以直观地评估语音端点检测算法的精度. 实验结果中也给出了算法在 Aurora 数据库上语音/噪声分类的正确率.

3.1 实验框架

实验采用 Aurora 工作组[9] 发布的标准语音数据库 Aurora2 和 Aurora3 来评估本文算法. Aurora2 是英文的数字串语音数据库, 每个数字串的长度为 1 ∼ 8, 其中包含机场、地铁、人群、车厢、展览馆、餐厅、街道和火车站 8 种环境模式. 每种环境下包括 7 种不同 SNR 的数据集 (无噪, 20, 15, 10, 5, 0, −5 dB), 每个 SNR 的数据集有 1 001 个语音数据. 所以实验中用到了 Aurora2 数据库的 56 056 个测试语音串, 这些数据由 52 个男性和 52 个女性录制. 我们取 Aurora2 数据库中多种环境的训练语音数据来训练隐马尔可夫声学模型. Aurora2 数据库采集过程的详细介绍可参考文献 [10].

Aurora3 是一个多语种的语音数据库, 语音内容全部是数字串, 长度 1 ∼ 10, 在不同行驶条件的车厢环境中录制, 包含了各种信噪比条件的数据. 实验中使用了 3 种语言的数据: 德语、西班牙语和丹麦语. 德语数据库有 3 118 个语音数据, 由 112 个人录制, 61 个女性和 51 个男性; 西班牙语数据库有 4 914 个语音数据, 由大约 160 人录制; 丹麦语有 2 457 个语音数据, 由 104 个男性和 104 个女性录制. 根据训练数据与测试数据录制条件不同, 每种语言都有三种不同的匹配条件: 高度匹配 (Well match, WM)、中度匹配 (Median match, MM) 和不匹配 (Highly mismatch, HM). Aurora3 数据库的实验框架和数据录制环境的详细定义可参考文献 [11]. 实验中所有的语音数据都是 8 kHz 采样频率, 16 位的量化精度.

语音识别前端采用 ETSI 发布的 Mel 倒谱系数的前端特征提取算法[8]. 为了更好地评估本文提出的算法, 我们在每种语言的数据库上进行了四组实验, 分别比较了四个算法的性能. 每组实验的前端算法说明如下:

1) 基准结果: 前端算法采用 Mel 倒谱系数标准前端, 也就是这组实验中前端的特征提取算法没有语音端点检测模块, 实验得到的是基准结果.

2) Ramirze: 在 Mel 倒谱系数标准前端的基础上增加了 Ramirze[3] 提出的语音端点检测算法, Ramirze 的算法也采用顺序统计滤波, 但是该算法基于短时能量. 利用两个顺序统计滤波器分别估计每个子带的信噪比和能量.

3) Xu: 在 Mel 倒谱系数标准前端的基础上增加了 Xu[5] 提出的语音端点检测算法, 该算法改进了传统基于频谱熵的语音端点检测算法.

4) 本文算法: 在 Mel 倒谱系数标准前端的基础上增加了本文提出的基于顺序统计滤波的子带频谱熵语音端点检测算法.

实验中我们将算法检测到的非语音帧直接丢弃, 而不送至后端的识别器. 前端提取出的特征是 39 维的特征参数, 由 12 维的倒谱参数和对数能量加上 13 维的一阶差分和二阶差分参数构成.

在所有实验中后端识别器的参数设置相同. 我们采用 HTK[1] 来训练隐马尔可夫声学模型, 后端识别器有 10 个数字的整词模型、一个短时停顿模型和一个静音模型, 隐马尔可夫声学模型参数的设置如下: 每个数字的整词模型由 16 个有效状态组成; 每个状态有 3 个高斯混合; 短暂停顿模型有一个有效状态, 静音模型有 3 个有效状态, 每个状态有 6 个高斯混合. 算法中的 3 个关键参数 $K = 4, \lambda = 0.9, N = 8$.

3.2 实验结果

为了更好地评估本文算法, 我们在实验过程中考虑了算法对语音识别系统性能的影响和算法的语音/噪声分类的正确率.

3.2.1 算法对语音识别系统性能的影响

表 1 给出了四个算法在 Aurora2 数据库各种信噪比条件下的语音识别系统的精度; 表 2 给出了算法在 Aurora2 数据库各种噪声环境下语音识别系统的精度. 表 3 给出了算法在德语、西班牙语和丹麦语数据库上的识别精度.

表 1　算法在 Aurora2 数据库各种信噪比条件下的平均识别精度

Table 1　Recognition accuracies under various SNR conditions on Aurora2 database

信噪比 (dB)	基准结果 (%)	Ramirze(%)	Xu(%)	**本文算法** (%)
无噪	98.48	95.67	96.01	**98.56**
20	97.51	95.72	95.10	**98.08**
15	96.34	95.24	94.66	**97.40**
10	93.86	93.29	93.40	**95.31**
5	85.06	87.36	86.32	**89.28**
0	57.49	68.39	69.66	**71.18**
−5	22.69	35.84	36.43	**36.79**

表 2　算法在 Aurora2 数据库各种噪声环境下的平均识别精度 (信噪比从 20 dB 到 0 dB)

Table 2　Recognition accuracies under various noisy environments on Aurora2 database (SNRs range from 20 dB to 0 dB)

环境	基准结果 (%)	Ramirze(%)	Xu(%)	**本文算法** (%)
机场	89.46	88.07	87.57	**92.03**
地铁	90.86	91.08	92.43	**92.21**
人群	85.91	86.63	87.09	**89.56**
车厢	87.41	90.05	86.72	**91.94**
展览馆	91.24	91.84	90.52	**92.16**
餐厅	87.54	88.91	89.34	**92.89**
街道	86.71	88.59	87.19	**89.75**
火车站	88.07	90.32	88.71	**90.67**

表 3　算法在 Aurora3 数据库上的识别精度

Table 3　Recognition accuracies on Aurora3 database

	条件	基准结果 (%)	Ramirze(%)	Xu(%)	**本文算法** (%)
	WM	90.62	92.86	91.27	**94.43**
德	MM	79.43	83.24	79.12	**86.53**
语	HM	74.1	74.97	79.82	**86.91**
	平均	81.38	83.69	83.40	**89.29**
西	WM	86.82	94.28	93.15	**96.61**
班	MM	73.45	84.52	84.10	**92.82**
牙	HM	41.08	75.66	66.53	**88.33**
语	平均	67.12	84.82	81.26	**92.59**
丹	WM	72.90	87.12	74.77	**92.10**
麦	MM	42.22	67.17	53.03	**78.43**
语	HM	26.06	61.75	37.95	**78.18**
	平均	47.06	72.01	55.25	**82.90**

实验结果表明, 本文提出的语音端点检测算法可以显著地提高语音识别系统的性能. 从表 1 我们可以看到, 添加本文算法后, 语音识别系统在不同信噪比的条件下, 系统的识别精度都得到了提高. 在信噪比越低的条件下, 识别精度提高得越显著. 在 0 dB 和 −5 dB 的条件下, 识别精度分别提高了 13.69% 和 14.1%. 表 2 中各种噪声环境下系统的识别精度都有不同程度的提高.

表 3 给出了算法在 Aurora3 数据库上的平均识别率. 三种语言数据库上分别提高了 7.91%、25.47% 和 35.84%, 系统性能在德语数据库提升的幅度比较小是因为该数据库的数据中非语音帧比较少, 而且信噪比相对其他两个数据库而言也比较高; 在同一个数据库中不匹配条件下, 与高度匹配和中度匹配条件下相比, 系统识别率提升的幅度大, 说明了语音端点检测算法可以改进训练环境与测试环境不匹配的问题. 在大多数的条件下, 引入语音端点检测算法可以提高语音识别系统的性能. 然而, 通过实验结果的比较, 本文算法优于其他两个算法.

比较表 2 与 3 的识别精度, 我们可以发现语音识别系统的识别精度在 Aurora2 数据库上提高的幅度比 Aurora3 数据库小. 因为 Aurora2 数据库中的语音, 发音比较连续, 语音串中的非语音帧数比较少.

3.2.2　语音/噪声分类的正确率

语音/噪声分类的正确率可以直观地评估语音端点检测算法的精度. 本节给出了三个算法在 Aurora2 数据库上, 各种信噪比条件下语音/噪声检测的正确率. 我们先把语音中的语音/噪声帧标注出来, 然后比较算法的检测结果. 图 4 和 5 给出了三个算法在 Aurora2 数据库的不同信噪比条件下, 语音/噪声检测的正确率. 语音/噪声检测正确率定义为, 正确检测到的语音/噪声的帧数与人工标注的语音/噪声总帧数的比例.

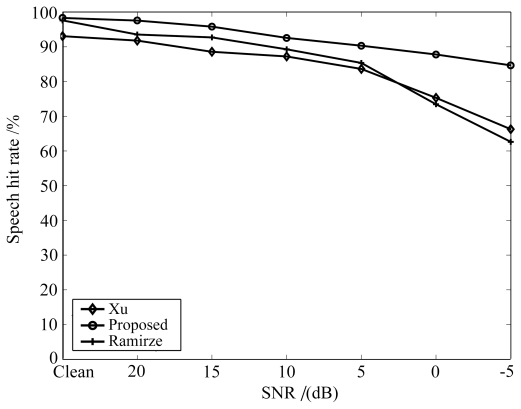


图 4　Aurora2 数据库上各种信噪比条件下语音检测正确率

Fig. 4　The speech hit rates under various SNR conditions on Aurora2 database

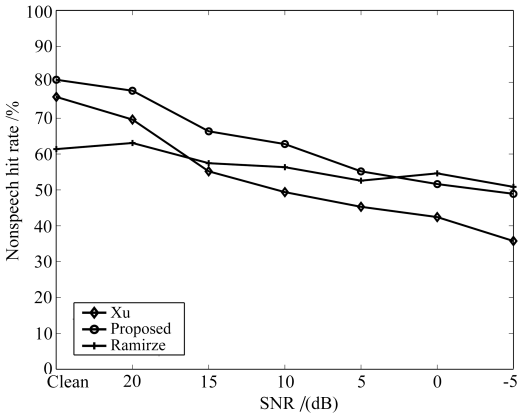


图 5 Aurora2 数据库上各种信噪比条件下噪声检测正确率

Fig. 5 The nonspeech hit rates under various SNR conditions on Aurora2 database

在语音分类的过程中会出现两种错误: 将语音误判为噪声和将噪声误判为语音. 由于系统将检测到的噪声帧丢弃, 不送至后端的识别器. 所以将语音误判为噪声, 必然会引起删除错误. 而将噪声误判为语音, 可能会引起插入错误, 但这不是必然的, 因为后端的识别器有静音模型. 因此, 语音检测算法应尽可能地避免将语音误判成噪声.

从图 4 和 5, 我们可以看到本文提出的算法比其他两个算法的性能更好. 大部分信噪比条件下, 本文提出算法的语音/噪声分类正确率比其他两个算法高. 各种信噪比条件下的语音检测的平均正确率为 92.7%, 即使在 -5 dB 的噪声条件下, 语音检测的正确率仍可达到 85%. 各种信噪比条件中, 平均 70% 的非语音帧可以被检测出, 显著地降低了系统的插入错误.

4 算法复杂度分析

由于本文算法是针对嵌入式语音识别系统设计的. 因此, 算法应具有较小的计算复杂度、延迟时间短、简单易实现等特点. 根据顺序统计滤波器的性质, 本文算法需要 N 帧的延迟. 在前端的特征提取算法中, 帧偏移量为 80 个采样点, $N = 8$, 实验中的语音数据采样频率 8 kHz. 因此, 算法的时间延迟为 80 ms, 这对于用户而言完全可以接受.

算法的计算量主要花费在子带频谱熵估计的对数运算和顺序统计滤波的排序运算, 除法也是比较耗时的运算, 而加法和乘法的运算代价不大. 在式 (3) 中, 对每帧的某个子带来说是一个常数, 在计算子带的频谱熵之前, 先计算出该式的值, 然后将除法运算转换为乘倒数运算, 所以每帧只需 K 次 ($K = 4$) 除法运算. 为了进一步减少对数运算的计算量, 在算法实现中对数运算采用了查表的方法. 每帧的子带熵计算完之后存储在缓冲区中, 得到第 L 帧的子带熵时, 根据顺序统计滤波器的原理对缓冲区中的数据进行排序. 此后, 缓冲区中的数据是有序的. 当新的频谱熵值可用时, 我们只需将新的值按算术顺序插入到合适的位置, 而不必重新排序. 本文算法计算量比 Ramirze 的算法小, 因为 Ramirze 的算法需要两个顺序统计滤波器分别估计子带的信噪比和噪声能量.

图 6 给出了 4 种算法在 Intel Xscale 处理器上处理 1 秒语音所需要的时钟周期数. 从图 6 可以看到, 本文提出的算法的计算代价仅占前端算法的 12%, 比 Ramirze 算法的计算量小, 与 Xu 的算法相近.

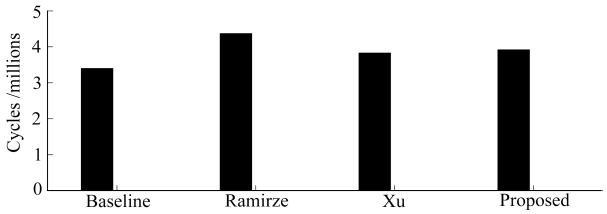


图 6 算法在 Xscale 处理器上所需的时钟周期数

Fig. 6 Algorithm computational complexity on Xscale processor

由于算法中无需存储大量的数据, 每个子带的顺序统计滤波器只需一个长度为 L 的缓冲区来存储子带熵, K 个子带需要 $K \times L$ 个缓冲单元. 假设每个缓冲单元占 4 个字节, 实验中 $K = 4, L = 17$, 那么计算过程中总共需要 272 个字节的存储空间. 因此, 算法的空间复杂度也不大, 适合嵌入式语音识别系统的应用.

5 总结

本文提出了一种基于顺序统计滤波的语音端点检测算法, 该算法以子带频谱熵为特征, 将每个子带的频谱熵经过顺序统计滤波, 改进了传统基于频谱熵算法的缺陷. 用相继若干帧的子带熵作为滤波器的输入, 滤波器的输出构成语音/非语音的区分特征. 实验表明该算法在各种噪声环境中能够准确地标出语音/非语音的边界, 在不同的信噪比条件和环境下具有良好的鲁棒性, 能够有效地改进训练环境与测试环境不匹配的问题. 本文提出的语音端点检测算法是针对嵌入式语音识别系统设计的, 其简单、易实现、计算代价小, 适合嵌入式语音识别系统的应用. 该算法已经集成进手机的语音识别系统, 对算法的鲁棒性和稳定性都进行了测试, 其性能和计算复杂度都能满足嵌入式系统的要求.

本文算法还可以进一步改进. 如果在非平稳噪声环境中式 (3) 中 Q 的值随着语音信号的信噪比变化, 在信噪比低的条件下取值比较大, 可以削弱噪声

对频谱熵的干扰; 在信噪比高的条件下取值比较小, 则可以避免辅音被误判成噪声, 从而可进一步提高语音识别系统的鲁棒性.

致谢

感谢摩托罗拉中国研究中心提供实验的所有数据库和测试平台.

References

1 Young S. *HTK BOOK - Version 3.3*. Cambridge: Entropic Cambridge Research Laboratory, 2005. 182−183

2 Nemer E, Goubran R, Mahmoud S. Robust voice activity detection using higher-order statistics in the LPC residual domain. *IEEE Transactions on Speech and Audio Processing*, 2001, **9**(3): 217−231

3 Ramirze J, Segura J C, Benitez C, de laTorre A, Rubio A. An effective subband OSF-based VAD with noise reduction for robust speech recognition. *IEEE Transactions on Speech and Audio Processing*, 2005, **13**(6): 1119−1129

4 Shen J, Hung J, Lee L. Robust entropy-based endpoint detection for speech recognition in noisy environments. In: Proceedings of International Conference on Spoken Language Processing. Sydney, Australia: 1998. 232−238

5 Jia C, Xu B. An improved entropy-based endpoint detection algorithm. In: Proceedings of International Symposium on Chinese Spoken Language Processing. Taibei, China: 2002. 285−288

6 Wu B, Ren X L, Liu C Q, Zhang Y X. A robust, real-time voice activity detection algorithm for embedded mobile devices. *Journal of Sol-Gel Science and Technology*, 1997, **8**(2): 133−146

7 Huang L S, Yang C H. A novel approach to robust speech endpoint detection in car environments. In: Proceedings of International Conference on Acoustics, Speech, and Signal Processing. Florida, USA: 2000. 1751−1754

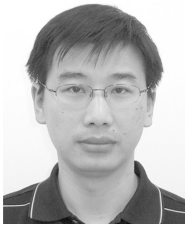
8 European Telecommunications Standards Institute for Speech Processing, Transmissions and Quality Aspects (STQ); Distributed Speech Recognition; Front-end Feature Extraction Algorithm; Compression Algorithms. European Telecommunications Standards Institute ES 201 108, 2003

9 Evaluation and language resources distribution agency [Online], available: http://www.elda.org/, April. 27, 2007

10 Pearce D, Hirsch H G. The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions. In: Proceedings of International Conference on Spoken Language Processing. Paris, France: 2000, 29−32

11 Netsch L. Description and Baseline Results for the Subset of the Speechdat-car German Database Used for ETSI STQ Aurora WI008 Advanced DSR Front-end Evaluation. STQ Aurora DSR Working Group Input Document, 2001

郭丽惠 华东师范大学计算机科学技术系硕士研究生. 主要研究方向为语音信号处理, 模式识别. 本文通信作者.
E-mail: guolihui@ecnu.cn
(**GUO Li-Hui** Master student at Department of Computer Science and Technology, East China Normal University. Her research interest covers speech signal processing and pattern recognition. Corresponding author of this paper.)



何 昕 博士, 摩托罗拉中国研究中心高级研究员. 1994 年和 1997 年在北京航空航天大学分别获得电子工程专业学士学位和通信与电子系统专业硕士学位, 2000 年在上海交通大学模式识别与智能系统专业获得博士学位. 主要研究方向为语音识别, 语音信号处理, 统计模式识别. E-mail: xin.he@motorola.com
(**HE Xin** Ph. D., senior researcher in Motorola China Research Center. He received his bachelor and master degrees from Beijing University of Aeronautics and Astronautics in 1994 and 1997, respectively, and his Ph. D. degree from Shanghai Jiao Tong University in 2000. His research interest covers speech recognition, speech signal processing, and statistic pattern recognition.)

张亚昕 博士. 1995 年在西澳大利亚大学电子工程系获得博士学位. 主要研究方向为语音识别和语音人机界面技术.
E-mail: yaxin.zhang@motorola.com
(**ZHANG Ya-Xin** Ph. D.. He received his Ph. D. degree from University of Western Australia, in 1995. His research interest covers speech signal processing, audio processing, automatic speech recognition, and speaker recognition.)

吕 岳 博士, 教授. 1990 年和 1993 年在浙江大学分别获得学士和硕士学位, 2000 年在上海交通大学获得博士学位. 主要研究方向为模式识别、图像处理、智能系统. E-mail: ylu@cs.ecnu.edu.cn
(**LV Yue** Ph. D., professor at Department of Computer Science and Technology, East China Normal University. He received his bachelor and master degrees from Zhejiang University, in 1990 and 1993, respectively, and his Ph. D. degree from Shanghai Jiao Tong University in 2000. His research interest covers pattern recognition, image processing, and intelligent system.)