·学术探讨·

说话人识别中语音切分算法的研究

何致远　胡起秀　徐光祐
（清华大学计算机科学与技术系，北京 100084）
E-mail：zhiyuanhe99@mails.tsinghua.edu.cn

摘　要　论文针对说话人识别中语音能量变化和噪声对提取有效语音数据的影响，在传统时域语音切分算法[1,3]的基础上，提出了三种孤立词的精确切分算法和一种连续语音的非精确切分算法。实验表明，新算法较好地克服了语音能量变化对切分的影响，在原始语音具有较高信噪比（≥10dB）的情况下，能够切除某些短时噪声和白噪声[2]。

关键词　语音切分　说话人识别

文章编号 1002-8331-(2003)06-0055-04　**文献标识码** A　**中图分类号** TP391.42；TN912.34

Research on Speech Segmentation Algorithm in Speaker Recognition

He Zhiyuan　Hu Qixiu　Xu Guangyou
(Department of Computer Science & Technology, Tsinghua University, Beijing 100084)

Abstract: In this paper, three word-segmentation algorithm and one continual speech segmentation algorithm are presented to avoid the effect caused by speech energy shifting and noise in speech data. The novel methods can overcome the limitation of the conventional speech segmentation method, and also have some effect on cutting off short-time noise and white noise.

Keywords: speech segmentation, speaker recognition

1　引言

在说话人识别中，通常只根据帧幅度或帧能量筛选出有声帧用于训练和识别，对语音的精确切分并没有太高的要求。但是，当用于训练和识别的语音数据量较小时，如基于孤立词的文本提示与文本相关的说话人识别，为了保证数据的有效性，需要对输入的语音进行精确切分。即使在不需要精确切分的情况下，如基于连续语音的文本无关的说话人识别，也要对原始语音进行筛选，滤除那些无声帧和噪声帧，最大程度地保证用于训练和识别的语音数据的有效性。要达到上述目的，就需要较好的语音切分算法。论文提出了 4 种语音切分新算法，在不同程度上消除了能量变化和噪声对切分的影响，应用于若干说话人识别系统，取得了良好的效果。

2　传统的语音切分算法

一般说来，语音切分需要提供在安静环境中录制的静音数据、在噪声环境中录制的噪声数据以及说话人的语音数据。语音切分常用的参数有：帧能量 E、帧幅度 M、帧过零率 Z、帧幅度过零率乘积 MZ 等。计算公式如下：

$$E_n=\sum_{l=1}^{L}S_l^2 \tag{1}$$

$$M_n=\sum_{l=1}^{L}|S_l| \tag{2}$$

$$Z_n=\frac{1}{2}\sum_{l=2}^{L}|\mathrm{sgn}S_l-\mathrm{sgn}(S_{l-1})| \tag{3}$$

$$MZ_n=M_n\times Z_n \tag{4}$$

其中 E_n、M_n、Z_n 和 MZ_n 分别是第 n 帧的能量、幅度、过零率和幅度过零率乘积。L 为一帧语音中采样点数目。S_l 为第 l 个采样点值。

传统的语音切分方法[1]利用帧幅度上限阈值 M_H、帧幅度下限阈值 M_L 和帧过零率下限阈值 Z_L 找出一段语音的起始帧 F_s 和结束帧 F_e。如图 1 示。

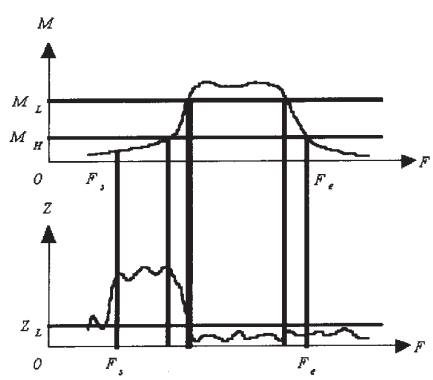


图 1　传统的语音切分方法

基金项目：国家 863 高技术研究发展计划项目（编号：863-306-ZT03-01-1）；清华大学 985 重点攻关项目（编号：985-校-22-攻关 06）
作者简介：何致远（1976-），男，清华大学计算机系硕士研究生。胡起秀，女，清华大学计算机系教授。徐光祐，男，清华大学计算机系责任教授、博士生导师。

帧幅度上限阈值 M_H、帧幅度下限阈值 M_L 和帧过零率下限阈值 Z_L 的计算方法如下：

$$M_H=\frac{1}{N_{snd}}\sum_{n=1}^{N_{snd}}M_{n,snd} \tag{5}$$

$$M_L=\frac{1}{N_{sil}}\sum_{n=1}^{N_{sil}}M_{n,sil} \tag{6}$$

$$Z_L=\frac{1}{N_{sil}}\sum_{n=1}^{N_{sil}}M_{n,sil} \tag{7}$$

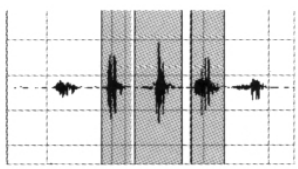
其中 $M_{n,snd}$ 为语音数据第 n 帧的帧幅度，$M_{n,sil}$ 为静音数据第 n 帧的帧幅度，$Z_{n,sil}$ 为静音数据第 n 帧的帧过零率。N_{snd} 和 N_{sil} 分别为语音数据的总帧数和静音数据的总帧数。

传统的语音切分算法的主要缺陷在于：

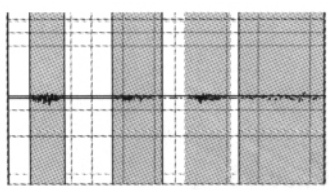
(1)用统计均值作为帧幅度上限阈值 M_H 准确性差，当一段语音中能量变化较大时，很可能由于 M_H 定得过高而误切掉某些能量较低的语音。图 2 (1)显示了用传统算法对汉语发音“7-8-3-9-4”(5 个孤立词)进行切分时，由于“7”和“4”的能量显著低于其他孤立词的能量而被误切的情况。

(2)当语音中存在噪声时，可能会连同噪声一同切进来。图 2 (2)显示了用传统切分算法对汉语发音“1-5-0”(3 个孤立词)进行切分时，由于语音能量较小，末尾的喘气(噪声)未被切除的情形。

由此可见，传统的语音切分算法不能适应声音能量的变化，受环境噪声影响较大，鲁棒性较差。



(1)能量变化对切分的影响



(2)噪声对切分的影响

图 2　实验结果：传统语音切分算法的缺陷

3　基于帧幅度统计阈值的孤立词切分算法

为了避免语音能量变化对切分的影响，在传统切分算法的基础上，提出了一种基于帧幅度统计阈值的孤立词切分算法。算法的主要思想是：统计各个能量水平上的帧数，根据每个能量水平上的帧的概率的高低确定帧幅度上限阈值。

大量实验表明，帧幅度 M 服从正态分布。找出原始语音数据中帧幅度的最小值 M_{min} 和最大值 M_{max}，将区间$[M_{min},M_{max}]$等分成 n 个长度为 d 的幅度子区间 $I_1,I_2,\cdots,I_n$，每个子区间 I_j 代表一个帧幅度量级 M_{Ij}：

$$I_j=[L_{I_j},R_{I_j}] \tag{8}$$

$$M_{I_j}=\frac{1}{2}(L_{I_j}+R_{I_j}),j=1,2,\cdots,n \tag{9}$$

扫描原始语音数据，若第 i 帧的帧幅度 $M_i\in I_x$，则用帧幅度量级 M_{Ix} 代替该帧的帧幅度 M_i。统计各个帧幅度量级的概率，找出概率最大的帧幅度量级作为帧幅度的上限阈值 M_H。如图 3 所示。

$$\exists M_{I_m},p_t[M_{I_m}]=\max_{1\le x\le n}p_t[M_{I_x}]\Rightarrow M_H=M_{I_m} \tag{10}$$

用统计得到的帧幅度上限阈值 M_H，结合帧幅度下限阈值 M_L、帧过零率下限阈值 Z_L，按前文中所述的方法进行语音切分。

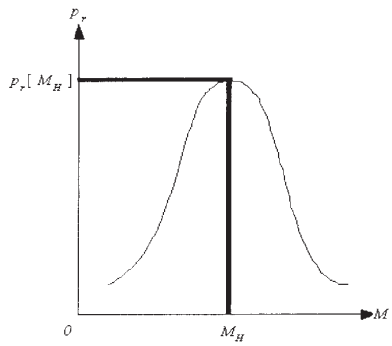


图 3　用统计方法确定帧幅度上限阈值

图 4 显示了用基于帧幅度统计阈值的孤立词切分算法对汉语发音“7-8-3-9-4”进行切分的情况，与图 2 (1)所示的传统算法的切分结果相比，采用新算法得到的统计上限阈值较准确地反映了大多数语音帧的能量水平，因而得到了正确的结果。

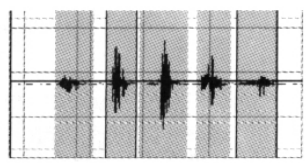


图 4　实验结果：基于帧幅度统计阈值的孤立词切分算法

4　基于动态搜索窗的孤立词切分算法

为了减少短时低能量噪声对切分的影响，提出了一种基于动态搜索窗的孤立词切分算法。算法的主要思想是：用一个动态窗口监视当前帧之前的所有无声帧的能量水平，用另一个动态窗口对可疑的有声段进行搜索，并根据该有声段的能量和过零率水平及其长度来区分短时噪声和语音。

自左向右扫描原始语音数据，设置两个动态窗口监视帧幅度水平的变化。设当前有声段为 A_c，当前帧为 F_C。定义左窗口 W_L 为前一个有声段 A_{c-1} 的结束帧 $F_e^{A_{c-1}}$ 到当前帧 F_C 之间的所有帧。若 F_C 为静音帧，左窗长 L_{W_L} 加 1。定义右窗口 W_R 为当前帧 F_C 到右窗口结束帧 $F_e^{W_R}$ 之间的所有帧。若 F_C 从静音帧变为有声帧，创建一个右窗口 W_R，然后向右搜索右窗口 W_R 的结束帧 $F_e^{W_R}$。若 W_R 的当前帧 F_{RC} 从有声帧变为静音帧，从 F_{RC} 再向前搜索 c_0 帧，若连续 c_0 帧都为静音帧，标记 F_{RC} 为右窗口 W_R 的结束帧 $F_e^{W_R}$ 并计算右窗长 L_{W_R}。

若公式 (1)成立，当前帧 F_C 被标记为当前有声段 A_c 的起始帧 $F_S^{A_C}$。

$$MZ_{W_R}\ge c_1\times MZ_{W_L},\ R\ge c_2 \tag{11}$$

其中 c_0,c_1,c_2 均为常数，一般令 $c_0=3,c_1=5,c_2=8$。

$$MZ_{W_R}=\frac{1}{L_{W_R}}\sum_{r=1}^{L_{W_R}}(M_r\times Z_r) \tag{12}$$

$$MZ_{W_L}=\frac{1}{L_{W_L}}\sum_{l=1}^{L_{W_L}}(M_l\times Z_l) \tag{3}$$

若公式(1)不成立,则当前帧 F_C 向右移动到右窗口 W_R 的结束帧 $F_e^{W_R}$ 处,同时左窗口 W_L 扩展到新的当前帧 F_C 处。重复上述过程直到找到当前有声段 A_c 的起始帧 $F_S^{A_c}$。寻找当前有声段 A_c 的结束帧 $F_e^{A_c}$ 的过程与前面叙述类似,只不过左、右两个窗口的定义以及相应判别条件相反而已。起始帧 $F_S^{A_c}$ 和结束帧 $F_e^{A_c}$ 之间的所有帧就是所需的当前有声段 A_c。

图5显示了用基于动态搜索窗的孤立词切分算法对汉语发音"1-5-0"进行切分的情况,与图2(1)所示的结果相比,新算法对末尾的喘气(噪声)进行了搜索并确定为非语音段,并将其正确地切除。

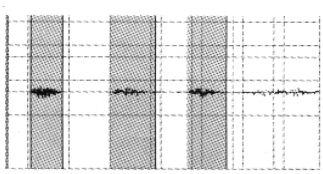


图5 实验结果:基于动态搜索窗的孤立词切分算法

这种算法动态地监视了帧能量和帧过零率水平的变化,能够在一定程度上切除那些能量和过零率水平与静音水平相差不大的短时噪声,对安静环境下录制语音进行切分效果较好,但是对于处理那些能量较高、时间较长的噪声没有太大的作用。

5 基于帧参数规一化的孤立词切分算法

基于帧幅度统计阈值的孤立词切分算法和基于动态搜索窗的孤立词切分算法从克服能量变化以及切除短时噪声两个不同的角度出发,在一定程度上避免了传统切分算法的缺陷,但是这两种算法有一个共同的缺点,即它们都规定了较多的加权系数和局部阈值,这使得算法本身的相关程度较高,因而不够鲁棒。针对这个问题,提出了一种新的基于帧参数规一化的孤立词切分算法,这种算法相关度较小,较前两种算法更鲁棒。该算法的主要思想是:对每一帧的参数用从静音(或噪声)数据中统计出来的帧水平进行规一,使之能够与静音(或噪声)水平相互比较,由于语音帧的规一化帧水平与静音(或噪声)水平有显著差异,因此能够较容易地将它们区分开来。

统计静音(或噪声)数据的帧幅度 M_{sil} 和帧幅度过零率乘积 MZ_{sil} 的均值 $\mu_{M,sil}$、$\mu_{MZ,sil}$ 和标准差 $\sigma_{M,sil}$、$\sigma_{MZ,sil}$,设 N_{sil} 为静音数据的帧数:

$$\mu_{M,sil}=\frac{1}{N_{sil}}\sum_{n=1}^{N_{sil}}M_{n,sil} \tag{4}$$

$$\mu_{MZ,sil}=\frac{1}{N_{sil}}\sum_{n=1}^{N_{sil}}MZ_{n,sil} \tag{5}$$

$$\sigma_{M,sil}=\sqrt{\frac{1}{N_{sil}}\sum_{n=1}^{N_{sil}}(M_{n,sil}-\mu_{M,sil})^2} \tag{6}$$

$$\sigma_{MZ,sil}=\sqrt{\frac{1}{N_{sil}}\sum_{n=1}^{N_{sil}}(MZ_{n,sil}-\mu_{MZ,sil})^2} \tag{7}$$

对语音数据的每一帧的帧幅度 M_{snd} 和帧幅度过零率乘积 MZ_{snd} 进行规一化,设 $1\leqslant i\leqslant N_{snd}$,$N_{snd}$ 为语音数据的总帧数:

$$M_{n,nor}=\frac{M_{n,snd}-\mu_{M,sil}}{\sigma_{M,sil}} \tag{8}$$

$$MZ_{n,nor}=\frac{MZ_{n,snd}-\mu_{MZ,sil}}{\sigma_{MZ,sil}} \tag{9}$$

设当前有声段为 A_c,当前帧为 F_C。从前一个有声段 A_{c-1} 的结束帧 $F_e^{A_{c-1}}$(帧标号为 e_{c-1})开始从左到右扫描,寻找当前有声段 A_c 的起始帧 $F_S^{A_c}$(帧标号为 s_c)。起始帧 $F_S^{A_c}$ 应满足公式(10)、(11)和(12)。

$$\exists s_1,M_{i_1,nor}<1\wedge M_{s_1,nor}\geqslant 1\wedge e_{c-1}\leqslant i_1\leqslant s_1<N_{snd} \tag{10}$$

$$\exists s_2,MZ_{i_2,nor}<1\wedge MZ_{s_2,nor}\geqslant 1\wedge e_{c-1}\leqslant i_2\leqslant s_2<N_{snd} \tag{11}$$

$$s_c=\min_{e_{c-1}\leqslant s_1,s_2<N_{snd}}(s_1,s_2) \tag{12}$$

然后从当前有声段 A_c 的起始帧 $F_S^{A_c}$(帧标号为 e_c)开始继续从左到右扫描,寻找当前有声段 A_c 的结束帧 $F_e^{A_c}$(帧标号为 e_c)。结束帧 $F_e^{A_c}$ 应满足公式(13)、(14)和(15)。

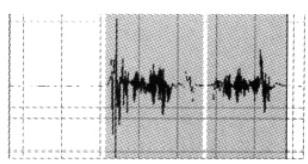
$$\exists e_1,M_{j_1,nor}<1\wedge M_{e_1,nor}<1\wedge s_c<j_1\leqslant e_1\leqslant N_{snd} \tag{13}$$

$$\exists e_2,MZ_{j_2,nor}\geqslant 1\wedge MZ_{e_2,nor}<1\wedge s_c<j_2\leqslant e_2\leqslant N_{snd} \tag{14}$$

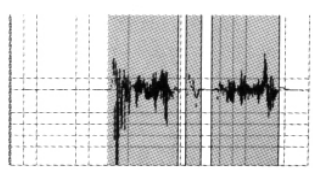
$$e_c=\min_{s_c\leqslant e_1,e_2\leqslant N_{snd}}(e_1,e_2) \tag{15}$$

若公式(16)成立,则起始帧 $F_S^{A_c}$ 到结束帧 $F_e^{A_c}$ 之间为有声段,且如果 $S_C=e_{c-1}+1^S$,要将前一个有声段 A_{c-1} 和当前有声段 A_c 合并;否则为无声段,令当前帧为 $F_C=F_{S_c}$。设 c_3 为常数(通常取 $c_3=10$):

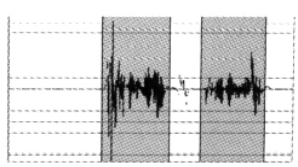
$$(e_c-s_c)\geqslant c_3 \tag{16}$$



(1)基于帧幅度统计阈值的孤立词切分算法



(2)基于动态搜索窗的孤立词切分算法



(3)基于帧参数规一化的孤立词切分算法

图6 实验结果:基于帧参数规一化的孤立词切分算法

图6分别显示了采用基于帧幅度统计阈值、基于动态搜索窗和基于帧参数规一化的孤立词切分算法对汉语发音"胡起秀-胡起秀"(2个孤立词)进行切分的结果。两次发音之间的喘气(噪声)的能量和过零率水平以及长度均超出了前两种算法的处理能力,而用帧参数规一化的方法得到了正确的切分结果,可见其鲁棒性确实优于前两种算法。

6 基于帧信噪比统计阈值的连续语音切分算法

前面所述三种方法都是孤立词的精确切分算法，在连续语音的说话人识别中，由于数据充足，而且不连贯语音对说话人识别影响不大，往往不需要进行精确切分，只要把那些无效的数据（如无声帧和噪声帧）滤除就可以了。基于帧信噪比统计阈值的连续语音切分算法的基本思想是：首先根据每一段需要处理的连续语音数据所附带的噪声数据（从现场采集而来或从语音中提取得到）和噪声基准数据（一般定义为安静环境）统计出适用于该连续语音数据的信噪比阈值，然后对每一帧语音数据统计其信噪比，将帧信噪比与阈值进行比较，低于阈值的就舍弃，反之则保留。

帧信噪比统计阈值 THD_{SNR} 和第 n 帧语音数据的帧信噪比 SNR_n 由下面公式得到：

$$THD_{SNR}=20\times\log_{10}\frac{\overline{M}_{noi}}{\overline{M}_{sil}} \tag{17}$$

$$SNR_n=20\times\log_{10}\frac{\overline{M}_{n,snd}}{\overline{M}_{noi}},1\leq n\leq N_{snd} \tag{18}$$

其中 N_{sil}，N_{noi}，N_{snd} 分别为静音数据、噪声数据和语音数据的总帧数，L 为语音数据每一帧的采样点总数，$\overline{M}_{sil}$ 和 $\overline{M}_{noi}$ 分别为静音数据和噪声数据的平均帧幅度，$M_{n,snd}$ 为第 n 帧语音数据的帧幅度。静音数据和噪声数据的平均帧幅度以及每一帧语音数据的帧幅度分别用下面的公式计算：

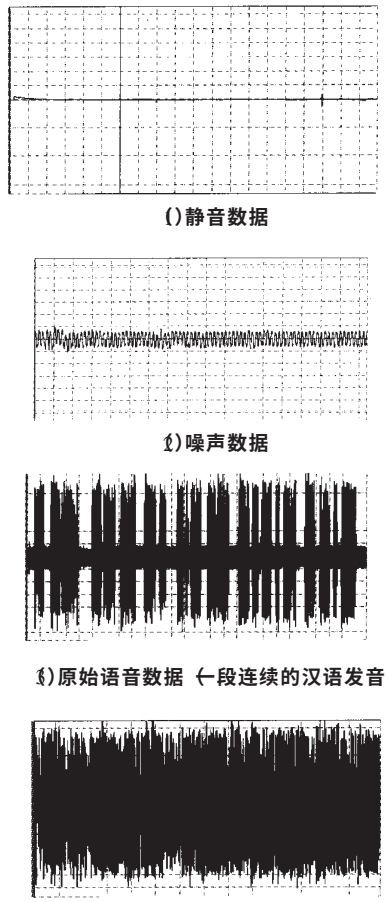
$$\overline{M}_{sil}=\frac{1}{N_{sil}}\sum_{n=1}^{N_{sil}}M_{n,sil} \tag{19}$$

$$\overline{M}_{noi}=\frac{1}{N_{noi}}\sum_{n=1}^{N_{noi}}M_{n,noi} \tag{20}$$

$$\overline{M}^{snd}=\frac{1}{K}\sum_{j=1}^{K}M_j^{snd} \tag{21}$$

若第 i 帧语音数据的 $SNR_i<THD_{SNR}$（$1\leq i\leq N_{snd}$），则该帧被滤除；否则被保留。

图 7 显示了对一段连续的汉语发音采用基于帧信噪比统计阈值的切分算法进行切分后的结果。



(a)静音数据

(b)噪声数据

(c)原始语音数据（一段连续的汉语发音）

(d)切分后的语音数据

图 7 实验结果：基于帧信噪比统计阈值的连续语音切分算法

7 结论

根据需要，将以上几种切分算法分别应用于不同的说话人识别系统，进行有效语音的提取，均取得了较好的效果，为训练和识别提供了准确有效的数据。结论可以归纳为以下 5 点：

(1)传统的语音切分算法不能适应声音能量的变化，受环境噪声影响较大，鲁棒性较差，其应用效果不理想，因而有必要根据说话人确认和识别的需要研究新的算法。

(2)基于帧幅度统计阈值的切分算法是对传统算法的直接改进，这种方法可以有效地避免由于语音能量变化而造成切分阈值选得过高或过低的问题，实际运用于一个基于随机数字串的文本提示的说话人确认系统。但该算法中需要确定的权重系数较多，算法的相关性较大，且如何确定这些权值是一个难点。

(3)基于动态搜索窗的切分算法利用左窗监视非语音帧的平均噪声水平，用右窗统计可疑的语音帧的能量水平及数量，依据右窗的长度及其能量和过零率水平可以判断右窗中的数据究竟是一个随机噪声还是一个语音。这种方法能够较准确地区分能量较低的短时噪声和一般孤立词语音，实际运用于一个基于姓名的文本相关的说话人确认系统。不足之处在于搜索窗的长度以及左端点能量的状态对算法有较大的影响，且算法相关性也较大。

(4)基于帧参数规一化的切分算法是这四种算法中最为鲁棒的一种，它实际上对每一帧的幅度、过零率以及二者的乘积做了规一，然后根据指定的阈值区分静音、噪声和语音。该算法考虑了长时语音的变化，相关性较低，实际运用于一个基于姓名的文本相关的说话人辨认系统。

(5)基于帧信噪比统计阈值的连续语音切分算法针对连续语音的说话人识别中不需要对语音进行精确切分的特点，舍弃那些信噪比较低的无声帧和噪声帧，达到数据选择的目的，实际运用于一个基于连续语音的文本无关的说话人识别系统。该算法一般只适用于连续语音，且精确度不高，但是对于说话人识别来说，在一段连续语音中丢弃一些语音帧对识别的影响不大，因此常用这种方法对一些数据进行自动切分。

（收稿日期：2002 年 8 月）

参考文献

1.杨行峻，迟惠生.语音信号数字处理[M].北京：电子工业出版社，1995

2.何致远.说话人确认和辨认的研究与实现[D].硕士学位论文.北京：清华大学计算机系，2002

3.Herbert Gish，Michael chmidt，Angela Mielke.A Robust，Segmental Method for Text Independent Speaker Identification[J].IEEE，1994

4.张继勇，郑方.连续汉语语音识别中基于归并的音节切分自动机[J].软件学报，1999；10（1）：1212~1215