改进批处理 RPEM 算法用于说话人识别*

项要杰[1,2]，杨俊安[1,2]，李晋徽[1,2]，杨瑞国[1]
(1. 电子工程学院，合肥 230037；2. 安徽省电子制约技术重点实验室，合肥 230037)

摘　要：针对传统 EM 算法训练 GMM 不能充分利用训练数据所属高斯分量信息，从而在一定程度上影响说话人识别性能的缺陷，采用 RPEM（竞争惩罚 EM）算法训练 GMM，并引入批处理 RPEM 算法解决 RPEM 算法运算量大、收敛速度慢的问题，同时针对 RPEM 和批处理 RPEM 算法训练时方差优化存在的问题进行了改进，提出了改进的批处理 RPEM 算法。在 Chains 说话人识别数据库上的实验表明，改进的批处理 RPEM 算法取得了相对于传统 EM、RPEM 以及批处理 RPEM 算法更好的性能，还极大地提高了训练效率，减小了运算量，说明了提出的改进批处理 RPEM 算法用于说话人识别时的有效性。

关键词：说话人识别；期望最大化算法；竞争惩罚 EM 算法；批处理竞争惩罚 EM 算法
中图分类号：TP391.4　　**文献标志码**：A　　**文章编号**：1001-3695(2013)12-3579-04
doi:10.3969/j.issn.1001-3695.2013.12.015

Improved batch RPEM algorithm for speaker recognition

XIANG Yao-jie[1,2], YANG Jun-an[1,2], LI Jin-hui[1,2], YANG Rui-guo[1]
(1. *Electronic Engineering Institute*, *Hefei* 230037, *China*; 2. *Key Laboratory of Electronic Restriction*, *Hefei* 230037, *China*)

Abstract: When the traditional EM algorithm was used to train GMM, it often failed to make full use of the information that the training data belongs to which Gaussian component. And it would influence the performance of speaker recognition to some extent. To solve this problem, this paper adopted the RPEM algorithm to train GMM. But the RPEM algorithm needed a large amount of computation, and its convergence speed was slow. So it introduced the batch RPEM algorithm to overcome the RPEM algorithm's above two shortcomings. However, there were also some problems with the optimization of the variance when it used RPEM or batch RPEM algorithm to train GMM. And it put forward the improved batch RPEM algorithm to solve these problems. The experiments that based on the chains speaker recognition database show that the improved batch RPEM algorithm not only achieves better recognition performance than other three algorithm's recognition performance, but also improves the training efficiency and reduces the amount of computation of the RPEM and batch RPEM.

Key words: speaker recognition; expectation maximization(EM) algorithm; rival penalized EM(RPEM) algorithm; batch rival penalized EM(RPEM) algorithm

0　引言

说话人识别技术属于生物认证技术的一种，是通过语音波形中反映说话人生理和行为特征的语音参数，利用计算机自动识别说话人身份的技术。随着人们对信息传递方便性、快捷性和安全性要求的不断提高，作为能够实现直接人机交互的说话人识别技术近年来得到了快速的发展，并且在银行证券系统、电话购物、国家和军队安全系统等需要身份认证的各种安全领域获得了广泛的应用。

说话人识别系统一般由特征提取模块、说话人模型的建立与训练模块以及评分判决模块组成。其中说话人模型的建立与训练模块是说话人识别系统的关键模块。目前常用的说话人模型包括矢量量化(vector quantification, VQ)模型[1]、高斯混合模型(Gaussian mixture model, GMM)[2]、支持向量机(support vector machine, SVM)模型[3]和神经网络(neural network, NN)模型[4]等。其中 GMM 是目前应用于文本无关说话人识别的主流模型，在进行说话人识别任务中能够取得相对于其他模型更好的识别效果，是现在说话人识别的基准方法。

GMM 参数通过传统的 EM[2]算法估计时，首先在完全数据集上求期望构造 Q 函数，然后对 Q 函数求各参数的偏导，得到各参数使似然函数取得局部极大值的表达式，根据各参数的表达式对初始模型不断迭代，更新模型参数，使似然函数值不断增大，最后估计出新的模型参数。由于高斯混合模型可以看成是一种软聚类方法，它假设数据集是由一个潜在的混合概率分布产生的，每个高斯分量表示一个不同的聚类[5]。用于训练的数据属于不同的高斯分量，具有特定的分布特性。传统的 EM 算法求得各参数的表达式对参数进行更新训练，使似然函数取得局部极大值，它在每次的迭代过程中没有充分利用训练数据的分布特性，所以最后训练的结果可能不能充分描述训练数据的分布，使得说话人识别系统达不到更好的识别效果。Cheung 等人[6,7]在研究高斯混合模型用于聚类时针对模型阶数的自动选择问题提出了竞争惩罚(rival penalized EM, RPEM)算法用于 GMM 的训练。该算法训练时根据当前训练数据在各高斯分量中的概率大小，对各高斯分量进行加权，权值与这些概率值成比例，使各高斯分量产生相互竞争来完成训练，最后产生的 GMM 参数能够使似然函数达到极大值。其在

收稿日期：2013-03-02；**修回日期**：2013-04-21　　**基金项目**：国家自然科学基金资助项目(60872113)
作者简介：项要杰(1987-)，男，安徽太和人，硕士研究生，主要研究方向为说话人识别(yixiang20062007@126.com)；杨俊安(1965-)，男，教授，博导，主要研究方向为信号处理、语音识别；李晋徽(1987-)，女，硕士研究生，主要研究方向为语种识别；杨瑞国，男，副教授.

训练时充分利用了数据的分布特征,所以能够很好地描述训练数据的分布,RPEM 算法的这种运算特点正好克服了传统 EM 算法的缺陷。Matza 等人[8]利用该算法自动选择 GMM 阶数的能力,将其应用到 TIMIT 语音库短语音说话人识别中,为每个人训练了合适阶数的 GMM,取得了较好的效果。在短语音的情况下,由于训练数据量较小,数据之间的混叠较小,GMM 能够收敛到合适的阶数,但是由于一般说话人识别训练时往往需要较大的数据量,并且数据的混叠较为严重,所以在大数据量的情况下,采用 RPEM 算法训练 GMM 其阶数难以稳定地收敛到特定的阶数,而且训练时间较长,识别效果也不太理想。

本文利用 RPEM 算法在训练 GMM 时能够充分利用数据分布特征的特点,针对 RPEM 算法在较大数据量下训练 GMM 时间长、效果不理想的问题,把批处理 RPEM (batch RPEM, BRPEM)[9]用于说话人 GMM 的训练。但是由于 BRPEM 方法在训练 GMM 时协方差的更新没有沿着最优化似然函数梯度上升的方向进行更新,这使得方差的更新较慢,同时还会影响似然函数极大值的寻找。针对这一问题,本文将方差的更新方向改进为梯度上升的方向,从而保证了协方差能够以最快的速度使似然函数达到最大值。

1 竞争惩罚 EM 算法

混合度为 M 的 GMM 可表示为:$\lambda = \{\omega_i, u_i, \Sigma_i \mid i = 1, 2, \cdots M\}$,对于 D 维的特征向量 X_t,其混合密度函数为

$$P(X_t \mid \lambda) = \sum_{i=1}^{M} \omega_i G(X_t \mid \mu_i, \Sigma_i) \tag{1}$$

其中:ω_i 为高斯分量的权重系数,$\omega_i > 0$ 且 $\sum_{i=1}^{M} \omega_i = 1$。

$$G(X_t \mid \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2} \mid \Sigma_i \mid^{1/2}} \exp\left[-\frac{1}{2} (X_t - \mu_k)^{\mathrm{T}} \Sigma_i^{-1} (X_t - \mu_k) \right]$$

对于给定的时间序列 $X = \{X_t \mid t = 1, 2, \cdots, T\}$,其中 X_t 是一帧的特征向量,利用 GMM 求得的对数似然度为

$$L(X/\lambda) = \log \prod_{t=1}^{T} P(X_t/\lambda) = \sum_{t=1}^{T} \log P(X_t/\lambda) \tag{2}$$

RPEM 算法在式(2)中构造一个权重函数,训练 GMM 时,每次迭代所得到的模型参数都使得式(2)的似然函数值逐渐增大,同时又充分利用训练数据的分布知识,直到最后得到稳定的模型参数。该算法具体如下:构造似然函数为

$$L(X/\lambda) = \sum_{t=1}^{T} \left[\sum_{i=1}^{M} g(i \mid X_t, \lambda) \right] \log P(X_t/\lambda) = \sum_{t=1}^{T} q_t(\Theta; X_t) \tag{3}$$

其中:$g(i \mid X_t, \lambda)$ 是权重函数,且 $\sum_{i=1}^{M} g(i \mid X_t, \lambda) = 1$。

$$g(i \mid X_t, \lambda) = 2I(i \mid X_t, \lambda) - h(i \mid X_t, \lambda) \tag{4}$$

定义:

$$h(i \mid X_t, \lambda) = \frac{\omega_i G(X_t \mid \mu_i, \Sigma_i)}{P(X_t \mid \lambda)} \tag{5}$$

$$I(i \mid X_t, \lambda) = \begin{cases} 1 & \text{若 } i = \underset{1 \leqslant i \leqslant M}{\operatorname{argmax}} h(i \mid X_t, \lambda) \\ 0 & \text{其他} \end{cases} \tag{6}$$

式(3)构造的似然函数,引入权重函数 $g(i \mid X_t, \lambda)$,使训练时若特征向量属于某一高斯分量的可能性最大,则该高斯分量的权重就会相应增加,而其他高斯分量的权重相应减小,并且增加和减小的比例与该特征向量属于各高斯分量的概率大小成比例。训练 GMM 的具体步骤如下:

a)参数初始化。设置初始模型 λ^o 各参数的初值,根据式(5)计算 $h(i \mid X_t, \lambda^o)$,然后由式(6)得出 $I(i \mid X_t, \lambda^o)$,最后根据式(4)计算 $g(i \mid X_t, \lambda^o)$。

b)更新模型 λ 的参数。定义参数 β_i 使其满足

$$\omega_i = \frac{\exp(\beta_i)}{\sum_{k=1}^{M} \exp(\beta_k)} \tag{7}$$

通过梯度上升法,更新 β_i。

$$\beta_i^n = \beta_i^o + \eta_\beta [g(i \mid X_t, \lambda^o) - \omega_i^o] \tag{8}$$

式中:β_i^n 和 β_i^o 分别是更新后参数和更新前参数。

同样,均值向量和方差矩阵的更新公式为

$$\mu_i^n = \mu_i^o + \eta g(i \mid X_t, \lambda^o) (\Sigma_i^o)^{-1} (X_t - \mu_i^o) \tag{9}$$

$$(\Sigma_i^n)^{-1} = [1 + \eta g(i \mid X_t, \lambda^o)] (\Sigma_i^o)^{-1} - \eta g(i \mid X_t, \lambda^o) U_{t,i} \tag{10}$$

其中:

$$U_{t,i} = (\Sigma_i^o)^{-1} (X_t - \mu_i^o) (X_t - \mu_i^o)^{\mathrm{T}} (\Sigma_i^o)^{-1} \tag{11}$$

以上公式中的 η_β 和 η 分别是权重和均值方差的学习率。

c)返回步骤 a),对输入的训练特征向量重复执行步骤 a)和 b),直到模型满足收敛条件。

在上面的训练过程中,协方差矩阵的更新是直接对其逆矩阵进行的更新,并且是沿着似然函数的 $\Sigma_i^{-1} \dfrac{\partial q_t(\Theta; X_t)}{\partial \Sigma_i^{-1}} \Sigma_i^{-1}$ 方向上进行的更新。也就是说,协方差参数更新的方向与似然函数的梯度方向 $\dfrac{\partial q_t(\Theta; X_t)}{\partial \Sigma_i^{-1}}$ 之间存在一个夹角。

从上面的过程可以看出,RPEM 算法在训练的过程中每次循环迭代只输入一个特征向量,当说话人模型训练的数据量较大时,计算量也相应地增加,使得模型训练时间很长。另外,由于 RPEM 算法在对协方差矩阵更新时没有沿着似然函数的梯度上升方向,这就使得方差的更新较慢,同时还会影响寻优过程,使训练达不到似然函数的极大值,这将在很大程度上影响训练模型的识别效果。

2 改进的批处理 RPEM 算法

为了减小 RPEM 算法的运算量,Zhang 等人[9]提出了 BRPEM 算法,并将其用于数据聚类,每次迭代时对一批数据进行计算,这样就很大程度地减少了模型训练运算量,同时也减小了迭代计算单个特征向量时模型收敛的不稳定性,使得训练过程中模型更加稳定地收敛到合适的高斯阶数。本文把 BRPEM 引入到说话人识别的模型训练中,同时针对 RPEM 及 BRPEM 算法方差更新存在的问题进行改进,使方差的更新方向沿着梯度 $\dfrac{\partial q_t(\Theta; X_t)}{\partial \Sigma_i^{-1}}$ 上升的方向,保证协方差能够以最快的速度使似然函数达到最大值。实验结果表明,改进的 BRPEM 算法不仅极大地减少了模型训练的时间,同时还提高了训练模型的识别效果。改进后的 BRPEM 算法步骤如下:

a)设置初始模型 λ^o 各参数、η_β 和 η 的初值;

b)把数据分成若干组,对于一组特征向量 $X = \{X_1, \cdots, X_t, \cdots, X_N\}$,根据式(5)计算 $h(i \mid X_t, \lambda^o)$,然后由式(6)得出 $I(i \mid X_t, \lambda^o)$,最后根据式(4)计算 $g(i \mid X_t, \lambda^o)$,其中 $i = 1, \cdots, M$。

c)使用特征向量组 $X = \{X_1, \cdots, X_t, \cdots, X_N\}$ 计算:

(a)权重更新增量

$$b_i = \frac{1}{N} \sum_{t=1}^{N} g(i \mid X_t, \lambda^o) - \omega_i^o \tag{12}$$

(b)均值更新增量向量

$$m_i = \frac{1}{N} \sum_{t=1}^{N} g(i \mid X_t, \lambda^o) (\Sigma_i^o)^{-1} (X_t - \mu_i^o) \tag{13}$$

(c)协方差更新矩阵

$$R_i = \frac{1}{N} \sum_{t=1}^{N} g(i \mid X_t, \lambda^o) U_{t,i} \tag{14}$$

$$S_i = \frac{1}{N}\sum_{t=1}^{N} g(i \mid X_t, \lambda^o)(\Sigma_i^o)^{-1} \tag{15}$$

$U_{t,i}$可由式(11)求得。

d)更新模型参数,对于 $i = 1, \cdots, M$,

$$\beta_i^n = \beta_i^o + \eta_\beta b_i \tag{16}$$

权重
$$\omega_i^n = \frac{\exp(\beta_i^n)}{\sum_{k=1}^{M}\exp(\beta_k^n)} \tag{17}$$

均值向量
$$\mu_i^n = \mu_i^o + \eta m_i \tag{18}$$

协方差矩阵
$$\Sigma_i^n = \Sigma_i^o + \eta(R_i - S_i) \tag{19}$$

e)返回步骤 b)反复迭代,直到模型收敛。

RPEM 算法以及本文提出的改进批处理 RPEM 算法中,初始参数 η_β、η 的设置对模型的训练有很大影响,参数设置得太大会使训练时出现收敛不稳定,参数设置得过小又会影响训练的收敛速度,所以根据训练数据的情况合理地设置初始参数也是很关键的。经过多次的实验,初始参数的设置在 0.001 ~ 0.1,模型一般都会稳定地收敛。

3　实验与分析

3.1　实验条件

实验在 MATLAB R2010a 环境下进行,所有的实验数据来自于 Chains corpus[10] 说话人识别数据库中的 Solo condition 标准语音数据,共 36 个说话人,包括 20 男 16 女,每个人分别有一段 1 min、两段 40 s、一段 30 s 的长录音和 33 段 3 s 左右的短录音,录音在两个月内分两次完成,实验中将一段 40 s 的录音分别切分成 20 s、10 s 和 7 s 左右的语音段。采样率经重采样后为 8 kHz,量化精度为 16 bit。

实验按照图 1 所示的流程建立说话人识别系统,完成基于上述语料库的文本无关闭集说话人识别实验测试。

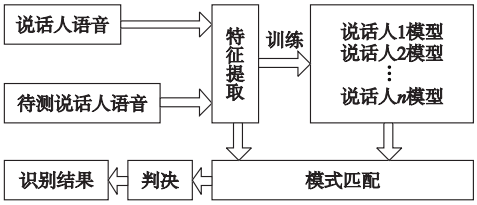

图1　说话人识别系统结构框图

采用 24 个三角 Mel 滤波器组经过离散余弦变换对说话人数据提取不同维数的 MFCC 特征进行实验,参数提取前进行的预处理包括:预加重 $1 - 0.95z^{-1}$,分帧帧长 256 个采样点,帧移 80 个采样点,加 Hamming 窗。

实验中把利用传统 EM 算法训练的高斯混合模型(GMM)作为说话人识别模型的基准模型,并分别采用 RPEM、BRPEM、改进的 BRPEM 算法为每个说话人训练一个 GMM(共 36 个 GMM),进行说话人识别实验,在训练时间和识别效果上进行分析比较。GMM 采用 32 阶的混合度,使用对角化协方差矩阵。

为充分比较不同方法训练 GMM 模型的识别效果,在训练时分别采用 1 min、40 s、30 s、20 s、10 s 以及 7 s 的语音进行训练,同时分别采用 19 ~ 23 维的 MFCC 参数进行多次实验比较。每个训练的模型都使用长约 3 s 的短语音进行测试,共 1 188 次测试,得出正确识别率。其中时间对比只在 1 min 训练时长情况下给出。正确识别率由式(20)进行计算:

$$\frac{\text{正确识别次数}}{\text{所有实验次数}} \times 100\% \tag{20}$$

3.2　实验结果与分析

a)传统 EM 算法在不同训练数据时长和训练特征维数下的测试效果如表 1 所示。

表 1　传统 EM 算法各种训练情况下识别率　/%

特征维数	数据时长					
	1 min	40 s	30 s	20 s	10 s	7 s
23	95.83	95.66	92.71	92.01	74.83	65.80
22	96.53	95.66	93.23	91.49	73.44	68.58
21	96.53	95.83	93.92	90.45	75.69	70.49
20	96.35	95.83	93.22	91.32	73.44	69.52
19	96.48	95.49	93.06	91.67	74.13	69.97

b)RPEM 算法训练时 η_β 和 η 均设置为 0.002,最大迭代次数设置为 1 500,其在不同训练数据时长和训练特征维数下的测试效果如表 2 所示。

表 2　RPEM 算法算法各种训练情况下识别率　/%

特征维数	数据时长					
	1 min	40 s	30 s	20 s	10 s	7 s
23	96.35	96.02	93.26	92.23	76.04	73.05
22	97.22	94.79	92.88	91.32	74.52	71.89
21	96.56	95.22	93.76	91.10	76.23	71.53
20	96.45	95.65	92.98	91.60	75.88	72.35
19	96.58	94.96	93.78	92.17	75.26	71.39

c)BRPEM 算法训练时 η_β 设置为 0.002,η 设置为 0.001,最大迭代次数设置为 1 500,批处理每组数据个数设置为 1 000,其在不同训练数据时长和训练特征维数下的测试效果如表 3 所示。

表 3　BRPEM 算法不同训练情况下识别率　/%

特征维数	数据时长					
	1 min	40 s	30 s	20 s	10 s	7 s
23	96.26	95.66	93.22	91.61	75.52	73.53
22	96.88	95.23	93.31	91.32	75.23	72.18
21	96.88	95.65	94.01	91.24	75.33	71.62
20	96.14	94.86	93.52	91.15	76.06	72.96
19	96.72	95.23	93.95	91.74	76.13	71.83

d)改进的 BRPEM 算法训练时 η_β 设置为 0.002,η 设置为 0.001,最大迭代次数设置为 1 500,批处理每组数据个数设置为 1 000,其在不同训练数据时长和训练特征维数下的测试效果如表 4 所示。

表 4　改进的批处理 RPEM 算法不同训练情况下识别率　/%

特征维数	数据时长					
	1 min	40 s	30 s	20 s	10 s	7 s
23	97.40	97.05	94.79	92.88	77.43	75.87
22	97.05	96.53	94.79	92.01	75.14	73.74
21	98.09	96.85	94.79	92.19	77.60	73.35
20	97.05	96.01	94.52	90.97	77.48	75.85
19	97.57	96.35	95.31	92.36	78.47	74.65

e)训练时各参数设置同实验 a) ~ d),1 min 训练数据的情况下,不同训练方法的时间开销如表 5 所示。

表 5　不同训练方法的单个 GMM 平均训练时间　/s

	训练方法			
	传统 EM	RPEM	BRPEM	改进 BRPEM
训练时间/s	369.14	4781.46	592.37	458.52

从上面的实验结果可以看出:

a)在识别效果上,传统的 EM 与 RPEM 算法相比,在训练

数据量较大时,两种算法在识别率上差别不是太大,在训练数据较小时 RPEM 算法的识别效果要明显比传统 EM 算法好,这主要是由于数据量较大时特征向量之间相互混叠,使得 RPEM 算法也不能够十分准确地描述数据的分布,所以其识别率并没有太大的提高;但当训练数据量较小时,数据间的混叠相对较小,易于 RPEM 算法的聚类运算,所以此时 RPEM 算法取得了相对传统 EM 算法更好的识别性能。BRPEM 算法虽然在训练时间上比 RPEM 算法减少了很多,但是由于它们在方差的更新上都没有沿着似然函数的梯度上升方向,所以训练的模型参数可能并非使似然函数达到最大值的最优参数,而改进的 BRPEM 算法则克服了前两种算法的这一缺点。从实验结果上看,其训练效果要好于前两种算法。

b)从模型的训练时间上看,传统 EM 算法训练单个 GMM 所用的平均时间最短,改进的批处理 RPEM 算法相对于 RPEM 算法极大地提高了训练效率,这是因为传统 EM 算法训练时采用使似然函数极大化各参数的直接迭代表达式,每次迭代只需计算该表达式的值就能完成参数的更新,而 RPEM 和改进的批处理 RPEM 算法训练时采用梯度上升的搜索算法使似然函数最大,初始参数的设置(如学习率、批处理数据的个数、高斯模型的初始参数等)往往会使 RPEM 算法在迭代过程中产生一些振荡,从而需要更多的迭代次数,对其收敛速度的影响很大。改进的 BRPEM 算法采用了数据的批处理方法,不仅极大地减少了运算量,同时把方差的更新改为沿着似然函数梯度上升的方向,更加快了算法的收敛速度,提高了算法收敛的稳定性,所以改进的批处理算法取得了比 BRPEM 更高的训练效率。

4 结束语

本文实验结果表明,改进后的 BRPEM 算法不仅提高了说话人识别的识别率,还极大地缩短了 RPEM 算法的运算时间;同时改进后的算法相对于传统 EM 算法在训练时间相差不大的情况下提高了说话人识别的识别率,说明了算法的有效性。但初始参数 η_β 和 η 在训练时不能根据训练情况实时调整也是本文工作的不足之处。笔者将在下一步把参数的自适应调整引入到本文的工作中来,以提高训练的效率和效果。

参考文献:

[1] KOMLEN D, LOMBAROVIC T, OGRIZEK-TOMAS M, *et al*. Text independent speaker recognition using LBG vector quantization[C]// Proc of the 34th International Convention. Croatia: MIPRO, 2011: 1652-1657.

[2] REYNODLS D, ROSE R. Robust text-independent speaker identification using Gaussian mixture speaker models[J]. IEEE Trans on Speech and Audio Processing, 1995, 3(1): 72-83.

[3] CAMPBELL W, CAMPBELL M, GLEASON T J P, *et al*. Speaker verification using support vector machines and high-level features[J]. IEEE Trans on Audio, Speech, and Language Processing, 2007, 15(7): 2085-2094

[4] TAN Jian-ding, TING Hua-nong. Malay speaker identification using neural networks[C]//Proc of International Conference on Information Science and Technology. 2011: 476-479.

[5] MINAEI-BIDGOLI B, TOPCHY A, PUNCH W F. A comparison of resampling methods for clustering ensembles[C]//Proc of International Conference on Artificial Intelligence. 2004: 939-945.

[6] CHEUNG Y M. Maximum weighted likelihood via rival penalized EM for density mixture clustering with automatic model selection[J]. IEEE Trans on Knowledge and Data Engineering, 2005, 17(6): 750-761.

[7] CHEUNG Y M, ZENG Hong. Feature weighted rival penalized EM for Gaussian mixture clustering: automatic feature and model selections in a single paradigm[C]//Proc of International Conference on computational Intelligence and Security. 2007: 633-638.

[8] MATZA A, BISTRITZ Y. Speaker recognition with rival penalized EM training[C]//Proc of IEEE International Workshop on Machine Learning for Signal Processing. 2011: 1-6.

[9] ZHANG Dan, CHEUNG Y M. A batch rival penalized EM algorithm for Gaussian mixture clustering with automatic model selection[C]// Proc of the 2nd International Conference on Rough Set and Knowledge Technology. Berlin: Springer, 2007: 252-259.

[10] CUMMINS F, GRIMALDI M, LEONARD T, *et al*. The CHAINS corpus: characterizing individual speakers[C]//Proc of SPECOM. 2006: 431-435.

(上接第 3578 页)

[7] CASTILLO E, CONEJO A J, MENENDEZ J M, *et al*. The observability problem in traffic network models[J]. Computer-Aided Civil and Infrastructure Engineering, 2008, 23(3): 208-222.

[8] CASTILLO E, MENENDEZ J M, SANCHEZ-CAMBRONERO S. Traffic estimation and optimal counting location with path enumeration using Bayesian networks [J]. Computer-Aided Civil and Infrastructure Engineering, 2008, 23(3): 189-207.

[9] CASTILLO E, JIMENEZ P, MENENDEZ J M, *et al*. The observability problem in traffic models: algebraic and topological methods [J]. IEEE Trans on Intelligent Transportation Systems, 2008, 9(2): 275-287.

[10] GENTILI M, MIRCHANDANI P B. Locating active sensors on traffic networks[J]. Annals of Operations Research, 2005, 136(1): 229-257.

[11] CASTILLO E, JIMENEZ P, MENENDEZ J M, *et al*. A ternary-arithmetic topological based algebraic method for networks traffic observability[J]. Applied Mathematical Modelling, 2011, 35(11): 5338-5354.

[12] LO H K, LUO X W, SIU B W Y. Degradable transport network: travel time budget of travelers with heterogeneous risk aversion[J]. Transportation Research Part B, 2006, 40(9): 792-806.

[13] CHEN A, YANG Hai, LO H K, *et al*. Capacity reliability of a road network: an assessment methodology and numerical results [J]. Transportation Research Part B, 2002, 36(3): 225-252.

[14] NG M W, WALLER S T. A computationally efficient methodology to characterize travel time reliability using the fast Fourier transform[J]. Transportation Research Part B, 2010, 44(10): 1202-1219.

[15] NG M W, SZETO W Y, WALLER S T. Distribution-free travel time reliability assessment with probability inequalities[J]. Transportation Research Part B, 2011, 45(6): 852-866.

[16] HU Shou-ren, PEETA S, CHU C. Identification of vehicle sensor locations for link-based network traffic applications[J]. Transportation Research Part B, 2009, 43(8-9): 873-894.

[17] NG M W. Synergistic sensor location for link flow inference without path enumeration: a node-based approach[J]. Transportation Research Part B, 2012, 46(6): 781-788.

[18] EPP S S. Discrete mathematics with applications[M]. London: Thomson Learning, 2004.