利用 PCA 和 AdaBoost 建立基于贝叶斯的组合分类器

陈松峰 范 明
(郑州大学信息工程学院 郑州 450052)

摘 要 提出了一种使用基于贝叶斯的基分类器建立组合分类器的新方法 PCABoost。本方法在创建训练样本时,随机地将特征集划分成 K 个子集,使用 PCA 得到每个子集的主成分,形成新的特征空间,并将全部的训练数据映射到新的特征空间作为新的训练集。通过不同的变换生成不同的特征空间,从而产生若干个有差异的训练集。在每一个新的训练集上利用 AdaBoost 建立一组基于贝叶斯的逐渐提升的分类器(即一个分类器组),这样就建立了若干个有差异的分类器组,然后在每个分类器组内部通过加权投票产生一个预测,再把每个组的预测通过投票来产生组合分类器的分类结果,最终建立一个具有两层组合的组合分类器。从 UCI 标准数据集中随机选取 30 个数据集进行实验。结果表明,本算法不仅能够显著提高基于贝叶斯的分类器的分类性能,而且与 Rotation Forest 和 AdaBoost 等组合方法相比,在大部分数据集上都具有更高的分类准确率。

关键词 组合分类器,主成分分析,AdaBoost,贝叶斯
中图法分类号 TP181 **文献标识码** A

Construct Ensembles of Bayes-based Classifiers Using PCA and AdaBoost

CHEN Song-feng FAN Ming
(School of Information and Engineering, Zhengzhou University, Zhengzhou 450052, China)

Abstract We presented a novel method for constructing ensembles of Bayes-based classifiers called PCABoost. For creating a training data, our method splited the features set into K-subsets randomly, and applied principal component analysis to each of the feature subsets to get its corresponding principal components. And then all of principal components were put together to form a new feature space into which the total original dataset were mapped to create a new training set. Different process could generated different feature space and different training sets. On each of the new training data we generated a group of classifiers which were boosted one by one using AdaBoost, so we could generate several different classifiers groups in the several different feature spaces. In the classification phase we firstly got several predicts using weighted-voted inside each of the classifiers groups, and then voted on the several predicts to get the final result as the ensemble's predict. Experiments were carried on 30 benchmark datasets picked up randomly from the UCI Machine Learning Repository, the results indicate that our method not only improves the performance of Bayes-based classifiers significantly, but also get higher accuracy on most of data sets than other ensemble methods such as Rotation Forest and AdaBoost.

Keywords Classifier ensemble, Principal component analysis, AdaBoost, Bayes

1 引言

分类[1,2]是数据挖掘领域研究的核心课题之一,它通过具有类标号的训练数据集构建一个分类器,然后利用分类器对未知类标号的实例进行分类。如何有效地提高分类准确率,一直是研究者们努力的方向。作为提高弱分类器的有效方法,组合分类方法备受青睐。

组合分类方法[3-7]是从机器学习领域逐渐发展起来的用于提升弱分类器准确率的技术,被认为是近 10 年来提出的最有效的学习思想之一。目前组合分类器仍是机器学习和模式识别方面研究的活跃领域之一,自 2000 年以来先后举办过 7 届多分类器系统(MCS)国际研讨会。

组合分类方法的思想是由训练数据训练生成若干个基分类器,然后通过对每个基分类器的预测(加权)投票来进行分类,从而取得比单个分类器更好的结果。比较经典的组合分类方法有 Bagging[8],Boosting[9-11]等。但是它们使用的是随机或加权的自助抽样来产生训练样本,不能保证它们所建立的基分类器之间的差异性。旋转森林[12](Rotation Forest)作为最近提出的一种新的组合分类方法,采用的是一种新的产生训练样本的方法,即通过不同的特征提取产生不同的特征

到稿日期:2009-09-14 返修日期:2009-11-23 本文受国家自然科学基金(编号:60773048),国家"十一五"科技支撑计划课题(编号:2006BAF01A00)资助。
陈松峰(1984—),男,硕士生,主要研究方向为机器学习、数据挖掘,E-mail:chsff@163.com;范 明(1948—),男,教授,博士生导师,CCF 会员,主要研究方向为数据库、数据挖掘、机器学习。

空间，并将整个训练数据集映射到不同的新的特征空间。最后在新的特征空间建立决策树作为基分类器。通过实验结果知道，它的分类性能显著高于 Bagging 和 Boosting。

AdaBoost[11]是 Boosting 的一种，加入了自适应的成分。作为一种经典的组合分类方法，它可以把一个弱分类器提升为一个任意高精度的分类器，可以有效提高弱分类器的分类准确率。

综合旋转森林和 AdaBoost 两者的特点，并参照 RotBoost[13]一文中综合 Rotation Forest 和 AdaBoost 建立基于决策树的组合分类器的思想，本文使用不同的特征提取将整个训练数据集映射到不同的特征空间，在每一个新的特征空间中利用 AdaBoost 来生成一组基于 Bayes 的逐渐提升的分类器，这样就生成了若干个有差异的分类器组，然后在每个分类器组内部通过加权投票产生一个预测，再把每个组的预测通过投票来产生组合分类器的分类结果，从而产生一个具有两层组合的组合分类器。

本文第 2 节分析了算法的主要思想；第 3 节对算法进行了形式化的描述并给出了相应的伪代码；第 4 节是算法的实验结果和性能分析；最后是对算法的总结和对下一步工作的展望。

2 算法思想

我们的算法用 PCA 进行特征选择，将整个训练数据集映射到不同的、新的特征空间，在每一个新的、不同的特征空间利用 AdaBoost 建立一组基于贝叶斯的逐渐提升的分类器组。我们从产生各个分类器组的差异性和提高每个分类器组的分类准确率这两个方面来构造组合分类器。

要想产生各个分类器组的差异性，首先要产生有差异性的训练样本，而产生有差异性的训练样本有多种方法。Bagging 采用有放回的等容量随机抽样产生有差异性的训练样本；Boosting 根据上一个分类器的分类情况，提高未正确分类样本的抽样权重，采用加权抽样产生有差异性的训练样本。但是它们这样产生训练样本存在如下不足：产生的每个训练样本都只使用了整个训练集的部分样本，这样就无法保证在这上面训练的分类器的准确率。而 Rotation Forest 在产生每个训练样本时，是通过不同的变换将整个训练样本映射到不同的、新的特征空间，以此来得到不同的、有差异的训练样本。这样在这上面建立基分类器不仅保证了基分类器之间的差异性，还保证了每个基分类器的准确率。为此，我们借鉴旋转森林的做法，使用 PCA 进行特征提取。但是与旋转森林不同的是，我们在不同的、新的特征空间建立的不是单个分类器，而是利用 AdaBoost 建立的一组逐渐提升的分类器，即一个分类器组。

为了能够使用 PCA 进行特征提取，建立多个有差异的变换，在建立第 i 个分类器组时，将特征集 F 随机地划分为 K 个不相交的子集，将训练数据集投影到每个特征子集上，使用 PCA 得到每个特征子集的主成分。用 T_{ij} 表示到第 j 个特征子集的主成分的变换。这些 T_{ij} 形成一个如式(1)所示的矩阵。然后，按照特征集 F 中的特征次序调整该矩阵的诸列，得到一个如式(1)所示的特征变换矩阵。

不同分类器组所使用的特征变换矩阵之间的差异通过两种措施来保证：其一，每次都采用不同的随机划分将特征集 F 划分成 K 个不同的特征子集；其二，在使用 PCA 求每个特征子集的主成分时，使用训练数据集中随机抽取的 50%样本。

$$T_i=\begin{bmatrix} T_{i1} & & 0^* \\ & T_{i2} & \\ & \cdots & \\ 0^* & & T_{iK} \end{bmatrix} \tag{1}$$

不同的变换将整个训练数据集映射到不同的特征空间。当在新的特征空间建立分类器组时，这些分类器组之间是有一定差异的。由于特征子集上的主成分在这些特征上最能代表原数据，因此分类器组也具有较好的分类准确率。

每个分类器组的准确率还主要通过组内部每个分类器之间的逐渐提升、互相补充的关系来实现。当把训练数据映射到不同的特征空间之后，在新的特征空间利用 AdaBoost 建立一组基于贝叶斯的逐渐提升的分类器：首先在初始化权重的情况下训练，生成第一个贝叶斯分类器，然后根据它的分类情况，提高未正确分类样本的抽样权重。在训练下一个分类器时，更关注上一次分类错误的实例，从而达到对上一个分类器“提升”的效果。经过数次的提升，每次都产生一个贝叶斯分类器。这样就产生了一组贝叶斯分类器，这一组贝叶斯分类器之间是逐个提升、互相补充的。接着以同样的方法在其它不同的特征空间分别产生若干组不同的、逐渐提升的贝叶斯分类器。当进入分类阶段时，对于一个类标号未知的实例，先在每一组贝叶斯分类器内部进行加权投票，产生一个预测。这样就产生了若干个预测，然后再对这若干个预测进行投票，就产生了一个最终的分类结果，并以这个结果作为组合分类器的分类结果。

3 算法描述

为了进一步形式化地描述本文的算法，下面引入一些相关概念和记号。

设训练数据集 D 包含 N 个样本，每个样本具有 p 个属性(特征)值和一个类标号。设 F 为 p 个特征的有序集合，$\{c_1,\cdots,c_m\}$是类标号的集合。训练数据集 D 可以用一个 $N\times p$ 的数据矩阵 X 和一个向量 $y=[y_1,\cdots,y_N]^{\mathrm{T}}$ 表示。其中，X 的行向量 $X_i=[x_1,\cdots,x_p]$记录了第 i 个样本在 p 个属性上的取值，而 y 的分量 $y_i\in\{c_1,\cdots,c_m\}$是第 i 个样本的类标号。我们的目的是建立 L 组 Bayes 分类器 $G_1,G_2,\cdots,G_L$，然后再组合它们，得到最终的分类模型 M。

为建立基分类器组 G_i，首先把特征集 F 随机地划分为 K 个不相交的子集，记作 $F_{i1},F_{i2},\cdots,F_{iK}$，使得每个子集包含大致相等的特征数。然后，对于每个特征子集 $F_{ij}(1\leqslant j\leqslant K)$，

1) 将数据集 D 投影到 F_{ij} 中的属性上，并随机抽取 50%样本得到数据集 D_{ij}。

2) 在 D_{ij} 上使用 PCA 求特征子集 F_{ij} 的主成分，并得到 F_{ij} 到其主成分的变换矩阵 T_{ij}。

把 $T_{i1},T_{i2},\cdots,T_{iK}$ 放置在主对角线上，得到一个形如式(1)的矩阵 T_i。然后，按照特征集 F 中的特征次序调整 T_i 的诸列，得到一个特征变换矩阵，记作 T_i'。

以(XT_i',y)为训练数据，利用 AdaBoost 建立分类器组 G_i，设 AdaBoost 共进行 t 次提升，则生成的分类器组中有 t 个基分类器$(C_{i1},C_{i2},\cdots,C_{it})$。

用 S_i 表示转换后的训练集(XT_i',y)，则生成基分类器组

G_i 的步骤如下：

1) 初始化 S_i 的权重分布 $w(i)=1/N$ $(1\leqslant i\leqslant N)$

2) For $t=1,\cdots,T$

i) 利用加权抽样从训练集 S_i 中抽取样本

ii) 在抽取的样本上用 NaiveBayes 建立基分类器 C_{it}，并利用式(2)计算它的错误率 err_{it}：

$$err_{it}=\frac{1}{N}[\sum_{s=1}^{N}w_{it}(s)I(C_{it}(x_s)\neq y_s)] \tag{2}$$

式中，如果 $q=$TRUE，则 $I(q)=1$，否则为 0。

iii) 如果错误率大于 0.5，重新初始化训练集权重，进入下一次迭代；如果错误率为 0，将错误率设为 10^{-10}，进行 iv)

iv) 由式(3)计算 C_{it} 的权重：

$$\alpha_{it}=\frac{1}{2}\ln\left(\frac{1-err_{it}}{err_{it}}\right) \tag{3}$$

v) 利用式(4)更新训练集中实例的权重分布：

$$w_{i,t+1}(s)=\frac{w_{it}(s)}{Z}\times\begin{cases}e^{-\alpha_{it}}, & \text{if } C(x_s)=y_s\\ e^{\alpha_{it}}, & \text{else}\end{cases} \tag{4}$$

式中，Z 是正规因子，以保证 $\sum_i w_i=1$。

3) EndFor

以同样的方法建立 L 个基于贝叶斯的分类器组。训练 L 组基分类器的算法伪代码在图 1 中给出。

在分类阶段，给定待分类样本 x^*，先在每个基分类器组内部用式(5)组合产生 x^* 属于每个类 c_j 的可信度：

$$G_{ij}(x^*)=\frac{1}{T}\sum_{t}^{T}\alpha_{it}I(C_{it}(x^*)=y_j) \tag{5}$$

由于使用 AdaBoost 构造的分类器组 G_i 将返回 x^* 属于每个类 c_j 的可信度 $G_{ij}(x^*)$ $(1\leqslant j\leqslant m)$，组合分类器 M 将用式(6)组合每个基分类器组的预测，计算 x^* 属于每个类的可信度：

$$p_j(x^*)=\frac{1}{L}\sum_{i=1}^{L}G_{ij}(x^*),j=1,\cdots,m \tag{6}$$

并把 x^* 指派到具有最高可信度的类。

4 算法实验结果及其性能分析

实验是为了验证本文的组合分类方法是否能够有效提高基于贝叶斯的分类器的性能，并与以前的建立组合分类器的方法 Rotation Forest 和 AdaBoost 进行比较。

所有实验都在 Weka 数据挖掘平台[14]上进行。我们利用 Weka 提供的接口和函数实现了本文的算法。实验中使用本文方法 PCABoost 与 Weka 中的 Rotation Forest(旋转森林在 Weka 中的实现)和 AdaBoost. M1(AdaBoost 在 Weka 中的实现)进行比较。组合算法中基分类器都使用 Weka 中的 NaiveBayes(naivebayes 算法在 Weka 中的实现)。

算法：PCABoost

训练阶段

输入：

X：不含类标号的训练数据集($N\times p$ 矩阵)

Y：训练数据集的类标号向量(N 维向量)

L：L 组基分类器

F：属性(特征)集

K：属性子集的个数

输出：

L 组基分类器 $G_1,G_2,\cdots,G_L$。

方法：

(1) for ($i=1;i\leqslant L;i++$) {

(2) 把特征集 F 随机地划分成 K 个子集 F_{ij} $(1\leqslant j\leqslant K)$

(3) for($j=1;j\leqslant K;j++$) {

(4) 将数据集 D 投影到 F_{ij} 中的属性上，并随机抽取 50%的样本得到数据集 D_{ij}

(5) 在 D_{ij} 上使用 PCA 求特征子集 F_{ij} 的主成分，并得到 F_{ij} 到其主成分的变换矩阵 T_{ij}

}

(6) 组合 $T_{i1},T_{i2},\cdots,T_{iK}$ 得到形如式(1)的矩阵 T_i

(7) 按特征集 F 中的特征次序调整 T_i 的诸列，得到特征变换矩阵 T_i'

(8) 以 $S_i=(XT_i',Y)$ 为转换后的训练数据集

//使用 AdaBoost 构造基于 Bayes 的分类器组 G_i

(9) 初始化训练集 S_i 的权重分布 $w(i)=1/N(1\leqslant i\leqslant N)$

(10) for($t=1;t\leqslant T;t++$){

1) 利用加权抽样从 S_i 抽取样本

2) 在抽取的样本上用 NaiveBayes 建立基分类器 C_{it}，并利用式(2)计算它的错误率 err_{it}

3) 如果 $err_{it}>0.5$，重新初始化权重分布，进入下一次迭代，如果 $err_{it}=0$，则设置 $err_{it}=10^{-10}$，并进入 4)

4) 利用式(3)计算 C_{it} 的权重

5) 利用式(4)更新 S_i 中每个实例的权重

}

}

分类阶段

输入：待分类样本 x^*

输出：x^* 的类标号

方法：

(1) for ($i=1;i\leqslant L;i++$){

利用式(5)计算出第 i 个基分类器组预测 x^* 属于每个类 c_j 的可信度 $G_{ij}(x^*)(1\leqslant j\leqslant m)$

}

(2) 使用式(6)计算 x^* 属于每个类的可信度 $p_j(x^*)(1\leqslant j\leqslant m)$

(3) if ($p_k(x^*)=\max\{p_j(x^*)\}$)

(4) 将 x^* 指派到 c_k 类

图 1 组合分类方法 PCABoost 的伪代码

在实验过程中，本文的算法 PCABoost 使用 PCA 变换产生 10 个不同的特征空间，在每个特征空间中由 AdaBoost 产生 5 个逐渐提升的 NaiveBayes 分类器，这样总共是 10×5 个 NaiveBayes 分类器。为了便于比较，所有的组合分类方法都产生 50 个基分类器。

为了使实验结果更具有说服力，我们的实验使用 30 个数据集，这 30 个实验数据集均从 UCI 机器学习标准数据集[15]中随机选取。表 1 中分别列出了 30 个数据集的名字、实例个数、类个数、离散和连续属性的个数等详细信息。

表 1 实验使用的 30 个 UCI 数据集

数据集	实例个数	类个数	属性个数	离散属性	连续属性
anneal	898	6	38	32	6
audiology	226	24	69	69	0
autos	205	7	25	10	15
balance-scale	626	3	4	0	4
breast-cancer	286	2	9	9	0
cleverland-14-heart	307	5	13	7	6
credit-rating	690	2	15	9	6
german-credit	1000	2	20	13	7
glass	214	7	9	0	9

heart-statlog	270	2	13	0	13
hepatitis	155	2	19	13	6
horse-colic	368	2	23	16	7
hungarian-14-heart	294	5	13	7	6
hypothyroid	3772	4	29	22	7
ionosphere	351	2	34	0	34
iris	150	3	4	0	4
labor	57	2	16	8	8
lymphography	148	4	18	15	3
pendigits	10992	10	16	0	16
pima-diabetes	768	2	8	0	8
segment	2310	7	19	0	19
sonar	208	2	60	0	60
soybean	683	19	35	35	0
vehicle	846	4	18	0	18
vote	435	2	16	16	0
vowel-c	990	11	12	2	10
vowel-n	990	11	10	0	10
waveform	5000	3	40	0	40
wine	178	3	13	0	13
zoo	101	7	18	16	2

表 2 给出了在 30 个数据集上的实验结果，其中 PCABoost，Rotation，AdaBoost 和 NaiveBayes 分别代表本文算法、Rotation Forest、AdaBoost 和 NaiveBayes。分类准确率是 10×10 折交叉验证[16]的统计结果。从表 2 可以看出：(1)PCABoost 在 30 个数据集上的平均分类准确率比 NaiveBayes 高 3.11%，表明本文的集成方法显著地提高了基于 Bayes 的分类器的性能；(2)与使用 Rotation Forest 和 AdaBoost 建立的基于 Bayes 组合分类器相比，PCABoost 的平均分类准确率分别提高 2.62%和 1.11%，表明本文算法在整体上较 Rotation Forest 和 AdaBoost 的分类准确率有了显著的提高。

表 2 几种组合分类方法的分类准确率

数据集	PCABoost	Rotation	Adaboost	NaiveBayes
anneal	90.13	90.54	95.20	86.59
audiology	75.95	75.04	78.06	72.64
autos	65.38	64.44	57.12	57.41
balance-scale	90.79	88.80	92.13	90.53
breast-cancer	71.35	71.78	68.57	72.70
cleveland-14-heart	82.74	83.50	83.14	83.34
credit-rating	85.26	84.67	81.16	77.86
german_credit	74.40	72.46	75.09	75.16
glass	52.99	53.64	49.63	49.45
heart-statlog	83.59	83.41	82.30	83.59
hepatitis	84.80	83.04	84.23	83.81
horse-colic	82.27	79.38	77.30	78.05
hungarian-14-heart	83.92	82.41	84.70	83.95
hypothyroid	68.79	62.35	95.27	95.30
ionosphere	94.71	84.30	91.09	82.17
iris	96.13	96.67	95.07	95.53
labor	93.23	93.73	88.33	93.57
lymphography	85.56	84.59	80.86	83.13
pendigits	88.59	88.07	85.76	85.76
pima_diabetes	76.67	73.78	75.88	75.75
segment	90.58	86.52	80.17	80.17
sonar	82.63	73.12	81.26	67.71
soybean	93.60	93.66	92.05	92.94
vehicle	65.47	62.21	44.68	44.68
vote	95.38	90.55	95.19	90.02
vowel-c	73.06	50.72	81.31	62.90
vowel-n	74.92	71.21	74.52	66.70
wavefom	82.77	81.79	80.01	80.01
wine	98.03	98.20	96.52	97.46
zoo	93.39	93.97	97.23	94.97
平均	82.57	79.95	81.46	79.46

表 3 是几种方法的综合比较。我们通过综合比较每种算法与其它 3 种算法在 30 个数据集上的显著输赢对比来分析每种算法在这 30 个数据集上的分类性能。表中第 i 行和第 j 列的元素 $n_1(n_2)$ 表示第 j 列的方法在 n_1 个数据集上的分类准确率高于第 i 行的方法(胜出)，并且在其中的 n_2 个数据集上显著高于(显著水平 2%)第 i 行的方法(显著胜出)。例如，表中第 3 行、第 2 列的元素 22(14)表示，与 Rotation Forest 相比，PCABoost 在 22 个数据集上胜出，并且在其中的 14 个数据集上是显著的。第 j 列的合计 $s_1(s_2)$ 表示相对于其他方法，第 j 种方法胜出 s_1 次，其中 s_2 次是显著的。第 i 行的合计 $s_1(s_2)$ 表示相对于其他方法，第 i 种方法输 s_1 次，其中 s_2 次是显著的。例如，第 2 列的合计 66(41)表示 PCABoost 共胜出 66 次，其中 41 次是显著的；而第 5 行的合计 60(35)表示 NaiveBayes 共输 60 次，其中 35 次是显著的。

表 3 几种分类方法的胜负综合比较

	PCABoost	Rotation	AdaBoost	NaiveBayes	合计
PCABoost	—	8(0)	9(5)	7(1)	24(6)
Rotation	22(14)	—	14(13)	10(4)	46(31)
AdaBoost	21(11)	16(11)	—	13(4)	50(26)
NaiveBayes	23(16)	20(12)	17(9)	—	60(35)
合计	66(41)	44(23)	40(27)	30(9)	

每种方法的相对好坏可以用显著胜负的次数之差(优势度量)来表示，结果汇总在表 4 中。从表中可以看出，相对于 Rotation Forest，AdaBoost 和 NaiveBayes，本文算法 PCABoost 具有明显的优势。

表 4 几种分类方法显著胜负的优势度量

方法	胜	负	优势度量
PCABoost	41	6	35
Rotation	23	31	−8
AdaBoost	27	26	1
NaiveBayes	9	35	−26

结束语 本文提出了一种使用 PCA 和 AdaBoost 建立基于贝叶斯的组合分类器的新方法 PCABoost。实验表明，本文的集成方法显著地提高了基于贝叶斯的分类器的性能；并且与建立组合分类器的其它方法(Rotation Forest 和 AdaBoost)相比，本文的方法也具有明显的优势。RotBoost 和本文的结果表明，使用不同的特征变换将数据集映射到不同的特征空间，在新的不同的特征空间建立不同的分类器组，然后分两层集成它们，这是一种建立组合分类器的有效方法。

一个有趣的问题：其他的特征提取方法是否可以用来代替 PCA 生成新的特征空间？这样做对整个组合分类器性能有什么影响？我们下一步的工作就是要考察这些问题，相信不久可以报告我们的结果。

参 考 文 献

[1] Han J, Kamber M. Data Mining: Concepts and Techniques (2nd ed)[M]. Morgan Kaufmann, 2006

[2] Tan P, Steinbach M, Kumar V. Introduction to Data Mining [M]. Addison-Wesley, 2006

[3] Haindl M, Kittler J, Roli F. Multiple Classifier Systems[C]// Proc. of the 7th International Workshop on MCS. Springer, 2007

[4] Oza N C, et al. Multiple Classifier Systems[C]// Proc. of the 6th International Workshop on MCS. Springer 2005

[5] Roli F, et al. Multiple Classifier Systems[C]// Proc. of the 5th International Workshop on MCS. Springer, 2004

[6] Roli F, Kittler J. Multiple Classifier Systems[C]// Proc. of the First International Workshop on MCS. Springer 2001

(下转第 256 页)

5 实验与结论

对 OCFD(Fuzzy Edge Detection Algorithm Based on Object Cloud)算法进行实验并分析结果。为了体现算法的优劣,主要从检测质量、抗噪性能、运算时间等方面对算法进行评价,同时采用其它检测方法进行对比实验。分别采用包括 OCFD 算法、模糊 Sobel 算子、Pal. King 等经典算法对待检测图像进行边缘检测,结果如图 2 所示。可以看出,OCFD 算法、模糊 Sobel 算法效果较好,Pal. King 算法出现漏检现象。针对加入 10%椒盐噪声和 10%高斯噪声后的图像,对 3 种算法在不同噪声环境下的检测效果作对比,结果如图 3 和图 4 所示。

图 2 3 种算法对比测试

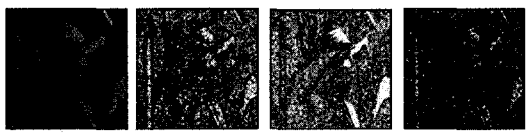

图 3 在 10%椒盐噪声下的对比测试

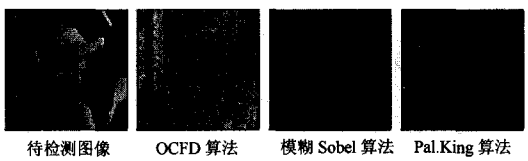

图 4 在 10%高斯噪声下的对比测试

从对比结果可以看出,3 种算法的检测结果都受到了明显的干扰,相对而言,OCFD 算法对椒盐噪声的抗噪能力要稍好于其余两种算法。对高斯噪声来说也同样如此,模糊 Sobel 算法对高斯噪声较为敏感,基本没有检测出边缘,而 Pal. King 算法稍好,OCFD 算法则效果更好一些。

利用上述 4 种算法对大小为 185×185 像素的待检测图像进行检测所花费的时间进行对比,结果如表 1 所列。模糊 Sobel 算法所需时间最短,其次是 Pal. King 算法;本文算法所需时间长于前两种算法。显然,如何进一步提高算法效率是本文算法尚需改善的地方。

表 1 检测算法耗时对比

Algorithm	Spending time(ms)	Order
OCFD	1264	3
Fuzzy Sobel	253	1
Pal. King	406	2

从上述分析可以看出,本文提出的基于对象云的图像模糊边缘检测方法(OCFD),同时考虑了图像模糊性以及随机性,弥补了传统模糊集理论处理模糊图像时的不足。算法利用云模型的特点,将图像空间映射到云空间中,并提取边界云所覆盖区域的模糊特征矩阵。利用模糊特征矩阵进行图像模糊划分熵的计算,将图像中的随机影响带入熵的求解,并多次利用模糊划分熵在服从一定概率分布的隶属度集合中寻求最优解。对比实验证明,OCFD 算法在检测质量、抗噪性方面优于对比算法,但运算速度仍需进一步提高。

参 考 文 献

[1] Fu X W, Fang K L, Li X. A improved and fast fuzzy algorithm on edge detection[J]. Journal of Wuhan University of science and technology: Natural science editiion, 2002, 25(1): 92-95

[2] Cheung K F, Chan WK. Fuzzy one-mean algorithm for edge detection[A]// IEEE International Conference on Fuzzy Systems [C]. Yokohama, Japan, 1995: 2039-2044

[3] Ho K H L. FEDGE-fuzzy edge detection by fuzzy categorization and classification of edges[A]// International Joint Conference on Artificial Intelligence Workshop [C]. Montreal, Canada, 1995: 182-196

[4] LiDY. UncertaintyinKnowledgeRepresentation[J]. Engineering Science, 2000, 2(10): 73-79

[5] Li DY . Liu C Y. Study on the Universality of the Normal Cloud Model[J]. Engineering Science, 2004, 6(8): 29-34

[6] Wang Shuliang, Shi Wenzhong, Li Deyi, et al. A method of spatial data mining dealing with randomness and fuzziness[C]// Proceedings of the Second International Symposium on spatial Data Quality. Hong Kong, 2003: 370-383

[7] Tian H, Lam S K, Srikanthan T. Area-time efficient between-class variance module for adaptive segmentation processes[J]. Vision, Image and Signal Processing, IEEE Proceedings, 2003, 150(4): 263-269

[8] 薛丽霞,王佐成,李永树,等. 基于云模型的模糊边缘检测[J]. 西南交通大学学报,2006,41(1):85-90

[9] Zhang Y J, Gerbrands J J. Transition region determination based thresholding[J]. Pattern Recognition Letters, 1991, 12: 13-23

[10] 乐宁,梁学军,翁世修. 图象过渡区算法及其改进[J]. 红外与毫米波学报,2001,20(3):211-214

[11] 王保平,刘升虎,张家田. 一种基于模糊熵和 FKCN 的边缘检测方法[J]. 计算机学报,2006,29(4):664-669

(上接第 239 页)

[7] Kuncheva L I. Combining Pattern Classifiers. Methods and Algorithms[M]. John Wiley and Sons, 2004

[8] Breiman L. Bagging Predictors[J]. Machine Learning, 1996, 24 (2): 123-140

[9] Schapire R E. The Strength of Weak Learnability[J]. Machine Learning, 1990, 5(2): 197-227

[10] Freund Y, Schapire R E. Experiments with a new boosting algorithm[C]// Proceedings of 13th International Conference on Machine Learning. Bari, Italy: Morgan Kaufmann, 1996: 148-156

[11] Freund Y, Schapire R E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting[J]. J. Computer and System Sciences, 1997, 55(1): 119-139

[12] Rodriguez J J, Kuncheva L I, Alonso C J. Rotation Forest: a new classifier ensemble method [J]. IEEE Trans. Pattern Anal. Mach. Intell, 2006, 28: 1619-1630

[13] Zhang Chun-xia, Zhang Jiang-she. RotBoost-A Technique for Combining Rotation Forest and AdaBoost[J]. Pattern Recognition Letters, 2008, 29(10): 1524-1536

[14] Witten, Frank E. Data Mining: Practical Machine Learning Tools and Techniques(Second ed)[M]. Morgan Kaufmann, 2005

[15] Blake C L, Merz C J. UCI repository of machine learning databases[EB/OL]. http://www. ics. uci. edu/~mlearn/MLRepository. html, 1998

[16] Nadeau C, Bengio Y. Inference for the Generalization Error[J]. Machine Learning, 2003, 62: 239-281