



研究与开发

基于拟合型弱分类器的 AdaBoost 算法

宋鹏峰, 叶庆卫, 陆志华, 周宇

(宁波大学信息科学与工程学院, 浙江 宁波 315211)

摘要: 针对 AdaBoost 算法通过最小化训练错误率来选择弱分类器造成的精度不佳问题以及单阈值作为弱分类器训练过程较慢难以收敛问题, 提出了一种基于拟合型弱分类器的 AdaBoost 算法。首先针对每个特征, 在特征值与标记值之间建立映射关系, 引入最小二乘法求解拟合多项式函数, 并转换成离散分类值, 从而获得弱分类器。其次从获得的众多弱分类器中, 选择分类误差最小的弱分类器作为本轮迭代的最佳弱分类器, 构成新的 AdaBoost 强分类器。与传统训练算法相比, 极大地减少了待选弱分类器的个数。选取 UCI 数据集和 MIT 人脸图像数据库进行实验验证, 相较于传统 Discrete-AdaBoost 算法, 改进算法的训练速度提升了一个数量级, 人脸检测率可达 96.59%。

关键词: AdaBoost; 拟合型; 最小二乘法; 弱分类器

中图分类号: TP391.4

文献标识码: A

doi: 10.11959/j.issn.1000-0801.2019219

AdaBoost algorithm based on fitted weak classifier

SONG Pengfeng, YE Qingwei, LU Zhihua, ZHOU Yu

College of Information Science and Engineering, Ningbo University, Ningbo 315211, China

Abstract: AdaBoost algorithm was proposed to minimize the accuracy caused by weak classifiers by minimizing the training error rate, and the single threshold was weaker and difficult to converge. The AdaBoost algorithm based on the fitted weak classifier was proposed. Firstly, the mapping relationship between eigenvalues and marker values was established. The least squares method was introduced to solve the fitting polynomial function, and the continuous fitting values were converted into discrete categorical values, thereby obtaining a weak classifier. From the many classifiers obtained, the classifier with smaller fitting error was selected as the weak classifier to form a new AdaBoost strong classifier. The UCI dataset and the MIT face image database were selected for experimental verification. Compared with the traditional Discrete-AdaBoost algorithm, the training speed of the improved algorithm was increased by an order of magnitude. And the face detection rate can reach 96.59%.

Key words: AdaBoost, fitting type, least squares, weak classifier

收稿日期: 2019-03-09; 修回日期: 2019-09-12

基金项目: 国家自然科学基金资助项目 (No.51675286, No.61071198)

Foundation Items: The National Natural Science Foundation of China (No.51675286, No.61071198)



1 引言

模式分类是数据挖掘的重要研究方向, 多种算法应用于对事物的分类识别研究, 其主要由两个阶段构成: 学习和分类。学习已知类标记的数据构建分类器, 得到分类器后对未知类标记的数据进行分类。AdaBoost 算法是一种自适应的 Boosting 算法^[1], 主要思想是通过使用采样原始数据得到的不同训练集来构建一系列弱分类器, 通过加权投票, 弱分类器被组合起来构建强分类器来预测新样本的类标记, 其性能优秀, 已被较好地应用于人脸检测^[2]、文本分类^[3]、车辆检测^[4]等领域中。

对于 AdaBoost 算法存在的训练复杂度高、分类精度不佳等问题, 研究者们对算法进行了改进。参考文献[5]提出了 Real-AdaBoost 算法, 其将 AdaBoost 算法从处理二值判定推广到能够连续置信度输出。连续置信度能更精确地刻画分类边界, 分类结果更加精确。参考文献[6]提出了 Gentle AdaBoost 算法, 首先赋予各训练样本相同权重, 计算损失函数最优时的弱分类器参数, 再利用分类结果及其类别标签来更新样本权重。重复上述过程, 得到新的弱分类器参数及样本权重, 进行多次迭代之后, 累加所得多个弱分类器, 形成强分类器, 可以有效地提高分类器的泛化能力。Logitboost 算法^[7]是基于机器学习的判别分类算法, 它根据样本数据集构建弱分类器, 通过负对数似然函数计算样本权重, 调用分类器检测样本的分类, 并在下一轮的迭代过程中增加判错样本的权重, 经过反复调用该弱分类器, 赋予判错样本较大的权重, 增加其关注度, 最终使得弱分类器在迭代过程中变为强分类器。Logitboost 算法对于多因素、二分类及多分类数据的分析效果尤为明显, 但训练时间较长。Gao 等^[8]提出了一种新的 AdaBoost 算法, 在两个方面进行了改进。首先, 通过最小化新的距离相关标准 (误差度加权训练

误差度量以及泛化能力度量, 而不是仅训练误差率来选择最具辨别力的弱分类器)。其次, 在获得经验设置的系数之后, 通过使用基于内核的感知器调整系数来组合最佳弱分类器。经过不同数据集的验证, 证明其要比传统的 AdaBoost 在分类精度上更佳。Yang 等^[9]提出了一种动态自适应调整阈值的改进 Adaboost 算法, 集成多个 ESN (echo state network, 回波状态网络), 然后通过组合规则得到最终的 ESN-Adaboost 模型。仿真结果表明, 该方法在处理大样本和噪声问题时, 具有较高的预测精度和较强的泛化能力。

本文从构建弱分类器的结构和参数角度出发, 提出了一种基于拟合型弱分类器的 AdaBoost 算法, 将训练样本集与对应的标记集建立映射关系, 通过拟合的思路, 使得样本权重的更新更平滑, 可以有效地避免过拟合现象的发生, 与此同时, 避免了单阈值需遍历特征值寻找弱分类器过程, 有效降低了算法的训练复杂度, 提升了检测精度。

2 AdaBoost 算法

AdaBoost 是一种迭代型算法, 首先对同一训练样本集进行抽样获得不同的训练子集, 然后用这些不同的训练子集训练不同的弱分类器, 最后通过加权将分类器组合, 形成一个强分类器。AdaBoost 算法^[10]如下。

算法 1 AdaBoost 算法

步骤 1 输入: 训练集 $\{(x_1, y_1), \dots, (x_n, y_n)\}$, 对应的训练类标记 $y_i \in \{0, 1\}$, 迭代次数为 T 。

步骤 2 初始化样本权重 $w_{1,i} = 1/2m, 1/2l$, 其中, m, l 分别代表正、负样本的数目, 总样本数 $n = m + l$ 。

步骤 3 寻找弱分类器 For $t = 1, \dots, T$ 。

(1) 对所有的样本权重进行归一化:

$$w_{t,i} = w_{t,i} / \sum_{i=1}^n w_{t,i} \quad (1)$$

(2) 对于每个样本中的第 j 个特征, 可以得到一个弱分类器 h_j , 误差记为

$$\varepsilon_j = \sum_{i=1}^n w_{t,i} |h_j(x_i) - y_i|。$$

(3) 在确定的弱分类器中, 选择一个具有最小分类误差 ε_t 的弱分类器 h_t 。

(4) 更新所有样本的权重: $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$, 如果样本被正确分类, 则 $e_i = 0$, 反之 $e_i = 1$, 其中 $\beta_t = \varepsilon_t / (1 - \varepsilon_t)$ 。

步骤 4 最终得到的强分类器为:

$$H(x) = \begin{cases} \sum_{t=1}^T \alpha_t h_t(x) \geq 0.5 \sum_{t=1}^T \alpha_t \\ 0, \text{ 其他} \end{cases} \quad (2)$$

其中, $\alpha_t = \ln(1/\beta_t)$ 代表 t 轮得到的弱分类器 h_t 所占权重, 该数值越大则表示该弱分类器更重要。

由 AdaBoost 算法的描述过程可知, 该算法在实施步骤中依据训练集的大小来初始化样本权重, 使其服从均匀分布, 在后续的操作中通过计算式改变和规范化算法迭代后样本的权值^[11]。样本被错误分类使得权值增大, 反之权值相应减小, 说明被错误分类的训练集包含一个更高的权值。这样在下轮时训练样本集更侧重于难以分类的样本, 针对这些关键样本的不断学习来构建下一个弱分类器, 直至样本被正确分类。在达到规定的迭代次数或者预期的误分率时, 则完成了强分类器的构建。

在算法 1 中, 每个特征的弱分类器需通过搜索所有的样本找到。在样本数和特征维度都很大的情况下, 得到一个弱分类器的时间复杂度较高。本文从弱分类器的设计上对原算法进行改进, 提高了训练速度及算法的泛化能力。

3 拟合型 AdaBoost 算法

本文算法首先在特征值与标记值之间建立映射关系, 引入最小二乘法求解拟合多项式函数^[12], 从而获得众多分类器, 选择出拟合误差更小的分

类器作为弱分类器, 构成新的 AdaBoost 强分类器用于检测分类。

3.1 拟合型弱分类器设计

步骤 1 构建一个用于代表样本矩阵当前列中的每个特征与对应的样本标记之间映射关系的矛盾方程组。设当前 t 轮次样本矩阵为 $X^{(t)}$, $x_{i,j}^{(t)}$ 为当前 t 轮次第 i 个样本的第 j 个特征, g_i 为该样本对应的样本标记 $g_i \in \{-1, 1\}$ 。样本总个数为 M , 特征个数为 L 。第 j 个特征与样本标记的矛盾方程组可描述为:

$$\begin{cases} \alpha_0 + \alpha_1 x_{1,j}^{(t)} + \alpha_2 (x_{1,j}^{(t)})^2 + \cdots + \alpha_k (x_{1,j}^{(t)})^k + \cdots + \alpha_L (x_{1,j}^{(t)})^L = g_1 \\ \alpha_0 + \alpha_1 x_{2,j}^{(t)} + \alpha_2 (x_{2,j}^{(t)})^2 + \cdots + \alpha_k (x_{2,j}^{(t)})^k + \cdots + \alpha_L (x_{2,j}^{(t)})^L = g_2 \\ \vdots \\ \alpha_0 + \alpha_1 x_{i,j}^{(t)} + \alpha_2 (x_{i,j}^{(t)})^2 + \cdots + \alpha_k (x_{i,j}^{(t)})^k + \cdots + \alpha_L (x_{i,j}^{(t)})^L = g_i \\ \vdots \\ \alpha_0 + \alpha_1 x_{M,j}^{(t)} + \alpha_2 (x_{M,j}^{(t)})^2 + \cdots + \alpha_k (x_{M,j}^{(t)})^k + \cdots + \alpha_L (x_{M,j}^{(t)})^L = g_M \end{cases} \quad (3)$$

其中, k 为拟合阶次, $0 \leq k \leq L$ 。 α_0 、 α_1 、 α_2 、 α_k 、 α_L 均为系数。将该矛盾方程组转换为矩阵形式, 描述为:

$$\begin{bmatrix} 1 & x_{1,j}^{(t)} & (x_{1,j}^{(t)})^2 & \cdots & (x_{1,j}^{(t)})^k & \cdots & (x_{1,j}^{(t)})^L \\ 1 & x_{2,j}^{(t)} & (x_{2,j}^{(t)})^2 & \cdots & (x_{2,j}^{(t)})^k & \cdots & (x_{2,j}^{(t)})^L \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{i,j}^{(t)} & (x_{i,j}^{(t)})^2 & \cdots & (x_{i,j}^{(t)})^k & \cdots & (x_{i,j}^{(t)})^L \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{M,j}^{(t)} & (x_{M,j}^{(t)})^2 & \cdots & (x_{M,j}^{(t)})^k & \cdots & (x_{M,j}^{(t)})^L \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_k \\ \vdots \\ \alpha_L \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_i \\ \vdots \\ g_M \end{bmatrix} \quad (4)$$

接着令待定系数向量 $A = [\alpha_0, \alpha_1, \cdots, \alpha_k, \cdots, \alpha_L]^T$, 其中 $\dagger$ 代表转置标记。则有: $\bar{X}^{(t)} A = G$, 利用最小二乘法求解 $\bar{X}^{(t)} A = G$, 得到 A 的唯一解, 记为 $\bar{A}$, $\bar{A} = [\bar{\alpha}_0, \bar{\alpha}_1, \cdots, \bar{\alpha}_k, \cdots, \bar{\alpha}_L]^T$ 。

步骤 2 根据步骤 1 得到的待定系数向量 $\bar{A} = [\bar{\alpha}_0, \bar{\alpha}_1, \cdots, \bar{\alpha}_k, \cdots, \bar{\alpha}_L]^T$, 确定用于代表样本 $X^{(t)}$ 中的特征与对应的样本标记之间映射关系的拟合多项式函数, 描述为:



$$f(x_{i,j}^{(t)}) = \sum_{k=0}^L \bar{\alpha}_k (x_{i,j}^{(t)})^k \quad (5)$$

若 $L=1$, 则 $f(x_{i,j}^{(t)}) = \sum_{k=0}^L \bar{\alpha}_k (x_{i,j}^{(t)})^k$ 为线性拟合函数, 若 $L>1$ 则为非线性拟合函数。

步骤 3 将拟合多项式函数得到的连续拟合值转换为离散的分类值。首先将当前列中每个特征的值分别代入 $f(x_{i,j}^{(t)}) = \sum_{k=0}^L \bar{\alpha}_k (x_{i,j}^{(t)})^k$ 中进行求解, 若求解得到 $f(x_{i,j}^{(t)})$ 的值大于 0, 则将用于代表该特征与对应的样本标记之间映射关系的拟合评估值确定为+1; 若求解得到 $f(x_{i,j}^{(t)})$ 的值小于或等于 0, 则将用于代表该特征与对应的样本标记之间映射关系的拟合评估值确定为-1。然后将当前列对应的所有拟合评估值构成列向量, 记为 F , $F = [f_1, f_2, \dots, f_i, \dots, f_m]^T$, f_i 表示用于代表 $x_{i,j}^{(t)}$ 与 g_i 之间映射关系的拟合评估值。

步骤 4 按下标对比 F 中的每个拟合评估值与标记 G 中的每个样本标记, 统计出数值不相同的下标。然后根据下标对应样本的样本权重计算当前特征列的拟合误差, 记为 err_j 。

步骤 5 更新特征列。令 $j = j+1$, 将样本 $X^{(t)}$ 中待处理的下一列作为当前列, 然后返回步骤 1 继续执行, 直至 $X^{(t)}$ 中的所有特征列处理完毕, 得到 $X^{(t)}$ 中的每列的拟合误差。再将 L 个拟合误差中值最小的拟合误差记为 $\varepsilon^{(t)}$, 并将 $\varepsilon^{(t)}$ 获取过程中确定的拟合多项式函数作为第 t 次分类器寻优过程中的弱分类器, 记为 $h^{(t)}(x)$ 。至此得到了该轮次样本权重下的拟合型弱分类器。

图 1 给出了 UCI 数据集 Jain 第 1 列特征值与对应标记的线性、二阶、三阶多项式拟合关系。

由图 1 可得出, 相较传统的离散 AdaBoost 算法单阈值遍历搜索寻优, 拟合型寻优方法一次拟合就可得到寻优结果, 减少了训练的时间复杂度。随着拟合多项式阶次的升高, 拟合效果更为明显。本文实验部分采用一阶、二阶、三阶多项式拟合,

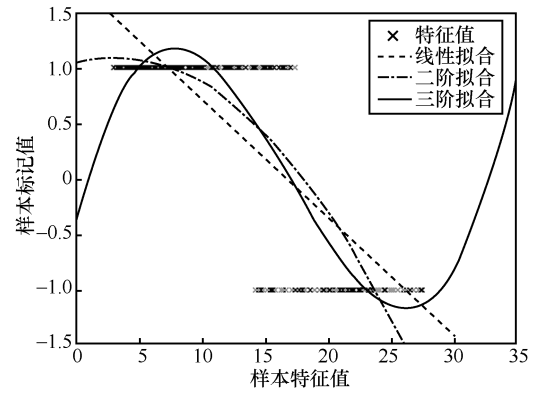


图 1 Jain 数据集第 1 列特征值与标记拟合

在保证拟合精度的情况下避免过拟合的现象发生, 保证算法的泛化能力。

3.2 算法描述

改进后的算法将算法 1 中步骤 3 的第 (2) 步改进为: 将样本中的第 j 个特征与对应的样本标记值之间建立映射关系, 得到该关系式的矛盾方程组, 最小二乘法求解得到拟合多项式函数。将算法 1 中的步骤 3 的第 (3) 步改进为: 得到的拟合多项式函数, 选择最小拟合误差代替最小分类误差作为选择弱分类器的评估标准。具体算法步骤如下。

算法 2 拟合型 AdaBoost 算法

步骤 1 输入: 训练集 $\{(x_1, y_1), \dots, (x_n, y_n)\}$,

对应的训练类标记 $y_i \in \{0, 1\}$, 迭代次数 T 。

步骤 2 初始化样本权重 $w_{1,i} = 1/2m, 1/2l$,

其中 m, l 分别代表正、负样本的数目, 总样本数 $n = m + l$ 。

步骤 3 寻找弱分类器 For $t = 1, \dots, T$ 。

(1) 对所有的样本权重进行归一化:

$$w_{t,i} = w_{1,i} / \sum_{i=1}^n w_{1,i} \quad (6)$$

(2) 对于每个样本中的第 j 列特征, 按式 (1) 建立矛盾方程组关系式, 引入最小二乘法求解, 得到 t 轮次 L 个拟合多项式函数。

(3) 在确定的拟合多项式函数中, 代入该列的特征值, 得到函数的连续拟合值。然后对拟合值进行离散化, 若拟合值大于 0, 则记为 1, 反之

则记为-1。将离散化的分类值与标记对比,求解拟合误差。选择一个具有最小拟合误差 ε_t 的弱分类器 h_t 。

(4) 更新所有样本的权重: $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$, 如果样本被正确分类, 则 $e_i = 0$, 反之 $e_i = 1$, 其中 $\beta_t = \varepsilon_t / (1 - \varepsilon_t)$ 。

步骤4 最终得到的强分类器为:

$$H(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq 0.5 \sum_{t=1}^T \alpha_t \\ 0, & \text{其他} \end{cases} \quad (7)$$

其中, $\alpha_t = \ln(1/\beta_t)$ 代表 t 轮得到的弱分类器 h_t 所占权重。

4 实验结果与分析

4.1 UCI 数据集仿真

为了验证本文算法的改进效果,在 UCI 数据集上选取 5 组实验数据集,分别为 Sonar、CMC、Mushroom、Credit-g、Cylinder-bands。在 Matlab 上应用传统 Discrete-AdaBoost 算法、Real-AdaBoost 算法和本文改进算法进行实验,采用交叉验证方式进行训练测试,随机取 4/5 的数据集用作训练,剩余 1/5 用作测试。设置算法训练的迭代次数分别为 30 次、60 次、100 次,改进算法采用一阶、二阶、三阶多项式拟合。样本实验数据集见表 1。

表 1 UCI 实验数据集

数据集	实例数	特征维数
Sonar	208	60
CMC	962	9
Mushroom	5 644	22
Credit-g	1 000	20
Cylinder-bands	277	39

在此给出了 Sonar 数据集中不同算法直观的训练结果对比,如图 2 所示。其中横坐标表示训练迭代的次数,纵坐标表示误检率。从该数据集

来看,3 种算法都随着训练迭代次数的增加,误检率有所下降。传统 Discrete-AdaBoost 在开始迭代时误检率处于 3 种算法中最小,但由于寻找最佳弱分类器性能不佳,精度较差,误检率下降速率小于 Real-AdaBoost 算法与本文拟合型算法,至 100 次迭代时,误检率为 4.2%,明显高于后两种算法。对于 Real-AdaBoost 算法,其将 Discrete-AdaBoost 算法从处理二值判定推广到能够连续置信度输出,在精度上有所提升,至 100 次迭代时,误检率为 2.6%,近似于线性拟合结果。对于拟合型算法,其在采用二阶拟合,到达 100 次迭代时,误检率为 1.6%,效果已明显优于前两种算法,当多项式阶次到达三阶时,效果更为明显,体现出了改进算法精度上的优势。

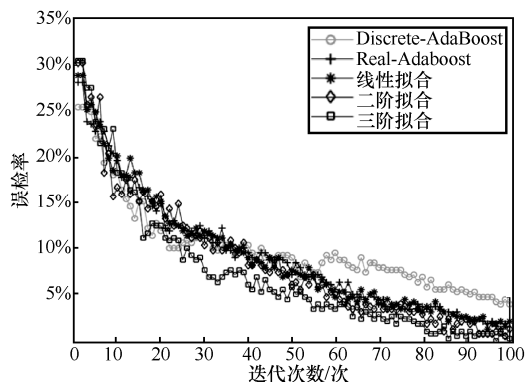


图 2 Sonar 数据集训练结果比较

表 2 显示了 Sonar、CMC、Mushroom、Credit-g、Cylinder-bands 数据集在不同算法、不同迭代次数下的误检率。跟传统 Discrete-AdaBoost 算法和 Real-AdaBoost 算法比较,二阶与三阶的拟合型 AdaBoost 算法误分率具有优势,意味着二阶与三阶的拟合型 AdaBoost 算法在精度上要高于传统 Discrete-AdaBoost 算法和 Real-AdaBoost 算法。从另外一个角度考虑,对于同等误分率的二阶与三阶的拟合型 AdaBoost 算法与传统 Discrete-AdaBoost 算法和 Real-AdaBoost 算法对比,前者的迭代轮次更少,也就是说组合成最终分类器的弱分类器个数更少,从而可以推断出前



表 2 UCI 实验数据集误检率

对比项	Discrete-AdaBoost	Real-AdaBoost	线性拟合	二阶拟合	三阶拟合	迭代次数
Sonar	0.116	0.113	0.121	0.105	0.079	30
	0.097	0.058	0.050	0.052	0.037	60
	0.042	0.026	0.024	0.016	0.005	100
CMC	0.413	0.391	0.386	0.381	0.364	30
	0.401	0.383	0.377	0.366	0.351	60
	0.390	0.371	0.368	0.357	0.326	100
Mushroom	0.328	0.331	0.320	0.321	0.315	30
	0.322	0.325	0.315	0.311	0.307	60
	0.315	0.313	0.309	0.302	0.298	100
Credit-g	0.211	0.205	0.186	0.188	0.186	30
	0.208	0.201	0.191	0.171	0.175	60
	0.199	0.193	0.176	0.161	0.160	100
Cylinder-bands	0.144	0.158	0.140	0.081	0.077	30
	0.112	0.117	0.094	0.050	0.031	60
	0.103	0.108	0.090	0.036	0.027	100

者的单个弱分类器的分类性能优于后两者。

为了验证算法的时间复杂度,表 3 给出了 UCI 数据集训练时间的对比,可见本文算法的训练时间远远小于传统 Discrete-AdaBoost 算法的训练时间。在线性拟合情况下,训练速度提升了约一个数量级,随着拟合阶次升高,运算更加复杂,训练速度有所下降。

上述分析可得,在 UCI 数据集仿真实验中,本文算法在相同迭代次数下,相比传统 Discrete-AdaBoost 算法和 Real-AdaBoost 算法误检

率更低,在二阶、三阶拟合情况下效果尤为明显。在算法训练的时间复杂度上,本文算法只需一次拟合即可得到弱分类器,而传统 Discrete-AdaBoost 算法在更新样本权重后每次都要遍历搜寻最佳阈值,随着样本增多,速度就会越慢,体现了本文算法在时间复杂度上的优势。

4.2 人脸检测

为验证改进算法的人脸检测效果,本文选择 Haar-Like 特征,选用美国麻省理工学院(Massachusetts Institute of Technology)人脸训练

表 3 UCI 实验数据集训练时间对比

对比项	Discrete-AdaBoost/s	Real-AdaBoost/s	线性拟合/s	二阶拟合/s	三阶拟合/s
Sonar	148	133	19	32	41
CMC	266	251	23	34	53
Mushroom	354	369	32	67	81
Credit-g	276	280	27	45	70
Cylinder-bands	136	127	12	21	43

库和非人脸训练库中的样本进行了实验。训练库中包含了 2 706 个分辨率为 19 dpi×19 dpi 的人脸样本和 4 381 个分辨率为 19 dpi×19 dpi 的非人脸样本, 包含各种肤色、遮挡、姿态、光照等情况。实验软件环境为 Windows 10, 主频为 3.5 GHz, 8 GB 内存配置, Matlab r2014a。部分训练样本如图 3 所示。



(a) 人脸样本



(b) 非人脸样本

图 3 部分训练样本

测试集选用美国卡耐基梅隆大学 (Carnegie Mellon University, CMU) 人脸库和自建单人脸测试集。CMU 人脸库选用其中 test 测试子集 42 幅图像, 包含 159 张人脸。CMU 测试集的结果见表 4, 在表 3 中, 检测率和误检率的定义分别为:

$$\text{检测率} = \frac{\text{检出人脸数}}{\text{测试人脸总数}} \times 100\% \quad (8)$$

$$\text{误检率} = \frac{\text{误检个数}}{(\text{测试人脸总数} + \text{误检个数})} \times 100\% \quad (9)$$

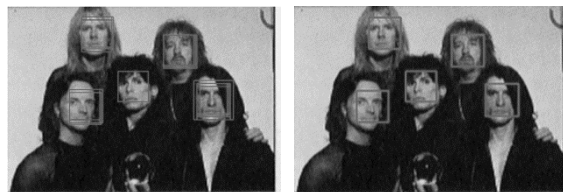
表 4 CMU 测试集的检测结果

对比项	Discrete-AdaBoost	Real-AdaBoost	本文算法
人脸数/张	159	159	159
检出人脸/张	133	137	139
漏检个数/张	26	25	20
误检个数/张	51	57	34
检测率	83.64%	86.16%	87.42%
误检率	24.29%	26.39%	17.62%

CMU 测试集部分检测结果如图 4 所示。

由于测试图中人脸所占比例不同, 采用多尺度策略使人脸检测分类器能够在图像中搜索不同

大小的人脸。如图 4 (a) 中共检测出 18 个人脸候选窗口, 采用并查集算法将检测到的人脸候选窗口进行合并, 使同一张人脸只被标记一次。具体方法为: 得到每个候选窗口的最左上角像素点坐标、窗口高度与宽度。比较两个窗口的重叠率, 若重叠率大于 80% 则判定此时两个窗口标记的为同一张人脸。依次进行两两比较直至遍历完所有的候选窗口, 将标记为同一张人脸的窗口左上角坐标点和窗口高度、宽度取平均得到该张人脸最终的检测窗口。图 4 (a) 最终合并为了 5 个人脸检测窗口, 如图 4 (b) 所示。本文人脸检测均采用该方法来显示结果。



(a) 多尺度人脸检测 1

(b) 合并检测窗口结果



(c) 多尺度人脸检测 2

(d) 多尺度人脸检测 3

图 4 CMU 测试集部分检测结果

图 4 均来自于 CMU 标准人脸测试库的 test 子集, 对于很多背景复杂图片中人脸存在漏检、误检的情况, 如图 4 (d) 所示。从检测率和误检率可得, 相比传统 Discrete-AdaBoost 算法和 Real-AdaBoost 算法, 本文算法在误检率上有明显的下降, 在人脸检测率上可达 87.42%, 高于传统 Discrete-AdaBoost 算法 4% 左右。

自建单人脸测试集采集了 7 人在不同角度、姿态、表情、光照、饰物下的单人脸图像。图片共 205 幅, 人脸图像大小均为 320 dpi×240 dpi。自建单人脸测试集的结果见表 5。



表 5 自建单人脸测试集的检测结果

对比项	Discrete-AdaBoost	Real-AdaBoost	本文算法
人脸数/张	205	205	205
检出人脸/张	192	197	198
漏检个数/张	13	8	7
误检个数/张	18	13	9
检测率	93.66%	96.09%	96.59%
误检率	8.07%	5.96%	4.21%

自建单人脸测试集检测结果如图 5 所示。

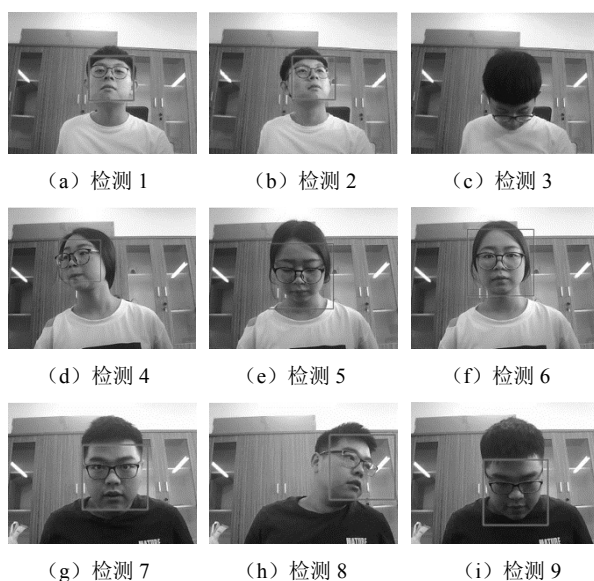


图 5 自建单人脸测试集检测结果

由于自建训练集人脸成像质量较高,采用单人脸图片,3种算法误检率较低,基本都能将正脸情况检测出来,但如果存在人脸倾斜,如低头、侧脸或有饰物遮挡的情况,则会出现漏检情况,如图 5(i)所示。从结果看本文算法的检测率可达 96.59%,误检率为 4.21%,优于传统 Discrete-AdaBoost 算法。

5 结束语

本文针对传统 AdaBoost 算法的训练时间复杂度、检测精度低问题,对弱分类器作了改进,提出了一种拟合型的 AdaBoost 训练算法,首先在

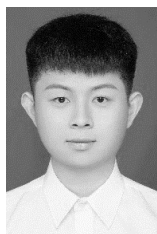
特征值与标记值之间建立映射关系,引入最小二乘法求解拟合多项式函数,从而获得众多分类器,选择出拟合误差更小的分类器作为弱分类器,构成新的 AdaBoost 强分类器。在 UCI 数据集和人脸数据库中进行了验证,实验结果表明,改进算法训练时间复杂度低,泛化能力强,分类精度也得到了提高。下一步的工作是将该算法的应用推广至多分类问题中,进一步提高算法的训练速度和测试精度。

参考文献:

- [1] 唐彪, 金炜, 符冉迪, 等. 多稀疏表示分类器决策融合的人脸识别[J]. 电信科学, 2018, 34(4): 31-40.
TANG B, JIN W, FU R D, et al. Face recognition using decision fusion of multiple sparse representation-based classifier[J]. Telecommunications Science, 2018, 34(3): 31-40.
- [2] LI X, WANG L, SUNG E. AdaBoost with SVM-based component classifiers[J]. Engineering Applications of Artificial Intelligence, 2008, 21(5): 785-795.
- [3] YI X, YING W, PENG J. An improved AdaBoost face detection algorithm based on the weighting parameters of weak classifier[C]//IEEE International Conference on Cognitive Informatics & Cognitive Computing, July 16-18, 2013, New York, NY, USA. Piscataway: IEEE Press, 2013.
- [4] 徐凯, 陈平华, 刘双印. 基于 AdaBoost-Bayes 算法的中文文本分类系统[J]. 微电子学与计算机, 2016, 33(6): 63-67.
XU K, CHEN P H, LIU S Y. A chinese text classification system based on AdaBoost-Bayes algorithm[J]. Microelectronics & Computer, 2016, 33(6): 63-67.
- [5] HUANG Z. Vehicle pedestrian detection algorithm based on AdaBoost[C]//International Conference on Intelligent Transportation, December 19-20, 2015, Halong Bay, Vietnam. Piscataway: IEEE Press, 2015.
- [6] WU S, NAGAHASHI H. Parameterized AdaBoost: introducing a parameter to speed up the training of real AdaBoost[J]. IEEE Signal Processing Letters, 2014, 21(6): 687-691.
- [7] ZHU J Q, CAI C H. Real-time face detection using gentle AdaBoost algorithm and nesting cascade structure[C]//International Symposium on Intelligent Signal Processing & Communications Systems, Nov 4-7, 2012, Taipei, China. Piscataway: IEEE Press, 2013.
- [8] KARLOS S, FAZAKIS N, KOTSIANTIS S, et al. Self-Train LogitBoost for semi-supervised learning[C]//EANN, Sep 25-28, 2015, Rhodes, Greece. Heidelberg: Springer, 2015.
- [9] GAO C, SANG N, TANG Q. On selection and combination of weak learners in AdaBoost[J]. Pattern Recognition Letters, 2010, 31(9): 991-1001.

- [10] YANG H, LIU S, LU R, et al. Prediction of component content in rare earth extraction process based on ESNs-Adaboost[J]. IFAC-Papers on Line, 2018, 51(24): 42-47.
- [11] CAO Y, MIAO Q G, LIU J C, et al. Advance and prospects of AdaBoost algorithm[J]. Acta Automatica Sinica, 2013, 39(6): 745-758.
- [12] CHENG W C, JHAN D M. A self-constructing cascade classifier with AdaBoost and SVM for pedestrian detection[J]. Engineering Applications of Artificial Intelligence, 2013, 26(3): 1016-1028.
- [13] 田垅, 刘宗田. 最小二乘法分段直线拟合[J]. 计算机科学, 2012, 39(S1): 482-484.
- TIAN L, LIU Z T. Least-squares method piecewise linear fitting[J]. Computer Science, 2012, 39(S1): 482-484.

[作者简介]



宋鹏峰 (1995-), 男, 宁波大学信息科学与工程学院硕士生, 主要研究方向为计算机视觉与图像处理。



叶庆卫 (1970-), 男, 博士, 宁波大学信息科学与工程学院教授、硕士生导师, 主要研究方向为信号检测、最优化搜索、视频识别与跟踪等。



陆志华 (1983-), 男, 博士, 宁波大学信息科学与工程学院讲师, 主要研究方向信号处理、多运动目标的实时跟踪、统计信号处理算法研究和应用等。



周宇 (1960-), 男, 宁波大学信息科学与工程学院教授、硕士生导师, 主要研究方向为信号处理、网络与信息安全、物联网技术等。