

# 基于核函数的Fisher手写数字识别系统研究

丁建, 李宁谦

(西华大学 计算机与软件工程学院, 四川 成都 610039)

**摘要:**将费歇(Fisher)分类器中的线性函数替换成非线性的高斯核函数,使用MNIST数据集对Fisher分类器进行训练,研究了核函数对分类结果的影响以及将算法应用于手写数字识别。结果表明,替换了高斯核的Fisher分类器,能够有效地区分非线性数据的类别,同时又保持了线性数据的准确率。

**关键词:**Fisher; 核函数; 手写数字识别; MNIST; 模式识别

**中图分类号:**TP311 **文献标识码:**A

**文章编号:**1009-3044(2020)01-0176-04

**DOI:**10.14004/j.cnki.ckt.2020.0078

开放科学(资源服务)标识码(OSID):



## Kernel-based Fisher Handwritten Digital Number Recognition System

DING Jian, LI Ning-qian

(School of Computer and Software Engineering, Xihua University, Chengdu 610039, China)

**Abstract:** Using nonlinear Gaussian kernel function replaces the linear function in the Fisher classifier, and uses the MNIST dataset to train it. The influence of the kernel function on the classification result is systematically studied, and then applying algorithm to handwritten digit recognition. The results show that the Gaussian kernel-based Fisher classifier can effectively distinguish the categories of nonlinear data while maintaining the accuracy of linear data.

**Key words:** fisher; kernel function; handwritten digital number recognition; MNIST; pattern recognition

手写数字识别在很多地方都有应用,例如智能机中的手写输入法,公司的税务局报表、支票数字编码识别等。被识别对象可以分为手写体以及印刷体;印刷体相较于手写体而言,具有字体规则,大小基本一致等特点,识别相对简单;手写体,由于个人书写习惯不同,各具特点,识别相对困难。通过算法自动识别文字、数字、符号、字母,实现信息自动录入,可以为税务、金融、文学等工作节省大量重复不必要工作的时间,加快工作速度,提升工作效率,同时产生一定的经济效益。

数字的类型有多种,阿拉伯数字、罗马数字等,目前研究较多的是针对阿拉伯数字(以下简称数字)的识别。数字对比文字的优点在于字形简单,缺点在于所含特征信息少。同种数字的多种写法以及某些数字字形相似等多种特殊情况都给识别系统带来巨大挑战。

本文考虑10个类别的数字识别,已不再是简单的线性分类问题,所以采用带高斯核函数<sup>[1][2][3]</sup>的Fisher分类器,用MNIST<sup>[4]</sup>数据集训练和测试模型。MNIST数据集是属于美国国家标准与技术研究所(NIST)的子集,由250个不同人手写的数字构成,其中一半来源于高中学生,剩下一半来自人口普查局的工作人员。

本文组织结构如下,第2章介绍线性Fisher算法<sup>[5][6]</sup>,第3章介绍带核函数<sup>[7][8][9]</sup>的Fisher算法,第4章介绍手写识别系统的设计、实现与测试,第5章为全文总结。

## 1 Fisher算法

### 1.1 算法简介

在两类问题中,已知样本集合 $X$ 内有 $N$ 个 $d$ 维的样本,其中 $N_1$ 个属于 $c_1$ 类的样本记为子集 $X_1$ , $N_2$ 个属于 $c_2$ 类的样本记为子集 $X_2$ 。

标准线性判别式(决策函数):

$$y = w^T x + w_0$$

令:

$$y_k = w^T x_k, k = 1, 2, \dots, N$$

$y_k$ 由向量 $x_k$ 通过 $w$ 变换得到的标量。对于给定的 $w$ , $y_k$ 就是判决函数的值。

由于子集 $X_1$ 和 $X_2$ 的样本经过投影后得到的对应子集 $Y_1$ 和 $Y_2$ 。因为我们关心的是 $w$ 的方向,可以令 $\|w\| = 1$ ,那么 $y_k$ 就是 $x_k$ 在 $w$ 方向上的投影。使 $Y_1$ 和 $Y_2$ 最容易分开的 $w$ 方向正是区分超平面的法线方向。

### 1.2 算法步骤

#### 算法1 计算两类的分类平面(线性)

**输入:**子集 $X_1$ 和 $X_2$

**输出:**投影变换 $w$

收稿日期:2019-08-15

作者简介:丁建(1995—),男,四川安岳人,硕士,主要研究方向:视觉目标跟踪、人体行为检测。

```

01: function LinearFisher(X1,X2)
02:  $m_i \leftarrow 1/N_i \sum_{X \in c_i} X, i = 1,2$ 
03:  $S_i \leftarrow \sum_{X \in c_i} (X - m_i)(X - m_i)^T, i = 1,2, S_c \leftarrow S_1 + S_2$ 
04:  $S_b \leftarrow (m_1 - m_2)(m_1 - m_2)^T$ 
05:  $W^* = S_c^{-1}(m_1 - m_2)$ 
06: return  $W^*$ 
07: endfunction
08:
09:  $y \leftarrow (\text{LinearFisher}(X_1, X_2))^T X$ 
10:  $\tilde{m}_i \leftarrow 1/N_i \sum_{y \in c_i} y, i = 1,2$ 
11:  $\tilde{S}_i = \sum_{y \in c_i} (y - \tilde{m}_i)^2, i = 1,2, \tilde{S}_c \leftarrow \tilde{S}_1 + \tilde{S}_2$ 
12:  $y_0 = (N_1 \tilde{m}_1 + N_2 \tilde{m}_2)/N$ 
13: if  $y > y_0$  then
14:  $X \in c_1$  else  $X \in c_2$ 
15: end if

```

## 2 使用核函数的Fisher算法

### 2.1 常用核函数

核函数可以将线性空间中的非线性关系映射到非线性空间的变成线性关系,从而可以很好地克服线性方法无法处理非线性问题的不足。

以下介绍几种常用的核函数<sup>[10]</sup>:

#### (1) 线性核函数

$$k(x,y) = x^T y + w_0$$

其中,  $w_0$  为偏移量

#### (2) 多项式核

$$k(x,y) = (w_1 x^T y + w_0)^n$$

其中,  $w_0$  为偏移量,  $w_1$  为系数,  $n$  为多项式的最高次数。当  $w_1 = n = 1$  时, 多项式核可以转化为线性核。

#### (3) 径向基函数

$$k(x,y) = \exp\{-\alpha \|x - y\|^2\}$$

径向基函数是指由特定点之间距离的实值函数,即为  $\Phi(x,y) = \Phi(\|x - y\|)$ 。任何具体此特性的都叫作径向基函数,一般使用欧式距离来度量点之间距离,也可以其他距离代替,如马氏距离等。其中常用高斯径向基函数,其定义为

$$k(x,y) = \exp\{-\|x - y\|/(2\sigma^2)\}$$

$\sigma$  为常数。

#### (4) 多层感知器核函数(又称为Sigmoid核函数,S型核函数)

$$k(x,y) = \tanh(w_1 x^T y + w_0)$$

其中  $w_0$  为衰减系数,  $w_1$  为尺度。

### 2.2 算法步骤

<b>算法2</b> 计算两类的分类平面(以高斯核为例)
<b>输入:</b> 子集 $X_1$ 和 $X_2$
<b>输出:</b> 分类结果

```

01: function GaussianKernelBasedFisher(X1,X2)
02:  $(m_i)_j \leftarrow 1/N_i \sum_{k=1}^{N_i} \text{kernel}(X_j, X_k^{(c_i)}),$ 
     $i = 1,2; j = 1,2,\dots,M$ 
03:  $(K_i)_{p,q} \leftarrow \text{kernel}(X_p, X_q^{(w_i)}), H \leftarrow \sum_{i=1,2} K_i(I - L_i)K_i^T,$ 
     $H_\mu = H + \mu I$ 
04:  $\alpha \leftarrow H_\mu^{-1}(m_1 - m_2)$ 
05:  $y_i \leftarrow W^T \cdot \Phi(X_j) = \sum_{i=1}^N \alpha_i \text{kernel}(X_i, X_j), j = 1,2,\dots,N$ 
06:  $\tilde{m}_i^\Phi \leftarrow 1/N_i \sum_{y_j \in c_i} y_j$ 
07:  $y_0 = (N_1 \tilde{m}_1^\Phi + N_2 \tilde{m}_2^\Phi)/N$ 
08: if  $y > y_0$  then
09:  $x \in c_1$  else  $x \in c_2$ 
10: end if
11: endfunction

```

## 3 系统设计、实现与测试

### 3.1 系统流程

图1给出了本识别系统的详细流程。在用户完成一次识别后,系统会给出预测值,当预测值与用户实际手写数字不相符时,用户可以标注好此错误识别数字的正确标签,并将错误样本提交到该系统的数据库中,以扩充系统的样本数据,增加模型的识别率。

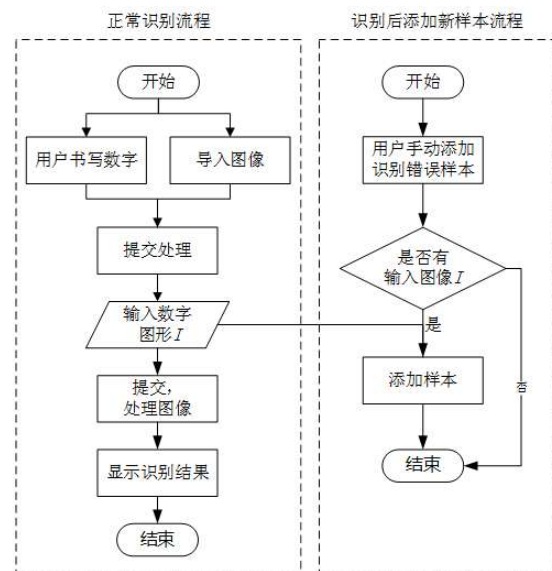


图1 手写数字识别系统流程图

### 3.2 模型训练

实验环境:操作系统 Windows1064 位,硬件平台: Intel-Corei7-6700 3.4GHz,内存 16GB,开发环境:MATLABR2016a。

用于训练的样本量为 60000 个,图片大小 28x28 像素,训练总共耗时 894 秒;用于测试的样本量为 10000 个,正确率为 92.89%。Fisher 属于二分类的分类器,一次只能训练 2 个类别,10 个类别总共需要训练 45 次。训练过程如图 2 所示。

```

>> main
Start training.....
Size of training dataset:60000
Digit 0 & Digit 1 ..... ok
Digit 0 & Digit 2 ..... ok
Digit 0 & Digit 3 ..... ok
Digit 0 & Digit 4 ..... ok

Digit 7 & Digit 9 ..... ok
Digit 8 & Digit 9 ..... ok
total training time:893.410988s
Size of t10k_images:10000
Using t10k_images, correct rate: 0.928900

```

图2 模型训练、测试截图

### 3.3 系统界面介绍

软件采用MATLABGUI<sup>[11]</sup>编程,系统界面主要由五大功能区域构成,区域一为显示功能区,能显示由用户通过手写窗口书写的数字;区域二为分类器选择区,可提供多种分类器(本文中只使用Fisher分类器);区域三为预测结果区,该区域不仅可以显示结果,还可以统计错误与正确的个数(需要用户按对应的按钮来操作识别正确的个数和识别错误的个数),并自动计算出对应的正确率;区域四为增加新样本区,用户在本次识别之后,可以选择将系统识别错误的样本添加到系统的样本库中,以增加样本库的数据量;区域五为按钮功能区,提供打开手写框、导入图片(支持PNG, JPEG, BMP格式)、启动识别、系统重置和退出的功能。

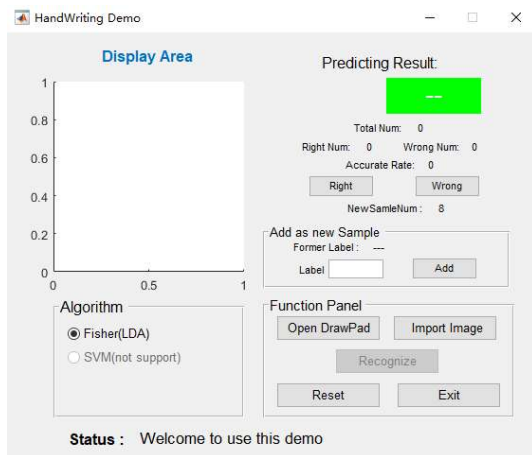


图3 系统界面

系统使用流程:点击界面上的“OpenDrawPad”按钮,打开画板,如图4所示,进行数字书写,完成后右击画板会自动保存所书写的数字图像,并处理为可以被识别的图像格式。然后再点击“Recognize”按钮,进行识别,结果会显示在界面的绿色区域内。如果预测结果与所写数字匹配,请点击“Right”按钮,否则点击“Wrong”按钮。你可以选择将本次识别的图像作为新样本,为训练下次模型提供样本。将正确的数字的标号输入“Label”右侧的输入框内,并点击“Add”按钮,添加成功会在界面底部状态栏进行提示,如图5所示。

### 3.4 系统测试

通过书写0至9任意数字,对本系统进行测试,测试部分结果如图6所示。测试共进行100次,每个数字进行10次测试,记录数字每次的测试结果(表1),并统计对应的正确率与总体的正确率。通过表分析,该系统算法对0、1、2、3、7能够准确识

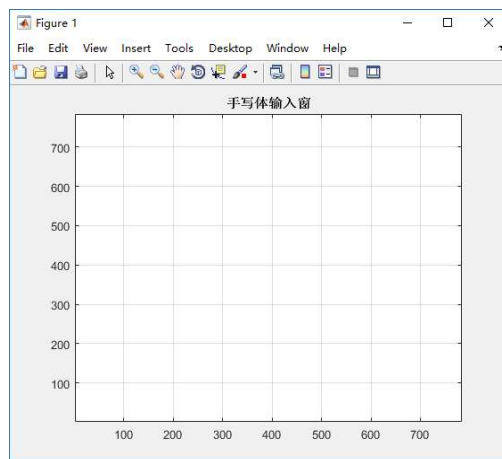


图4 手写输入窗口

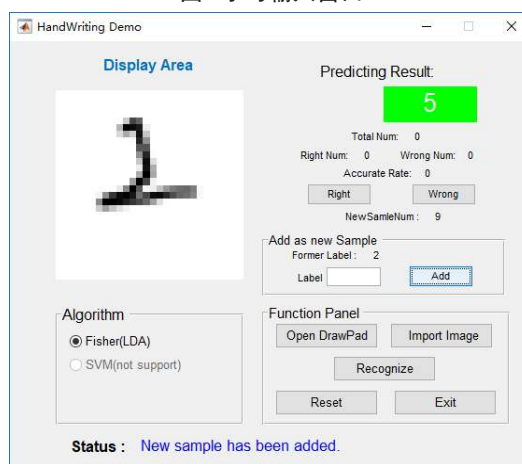


图5 手动添加新样本,状态提示

别,4、5、6、8、9能够识别,实际100次平均识别率为71%。

表1 数字0-9测试结果记录表

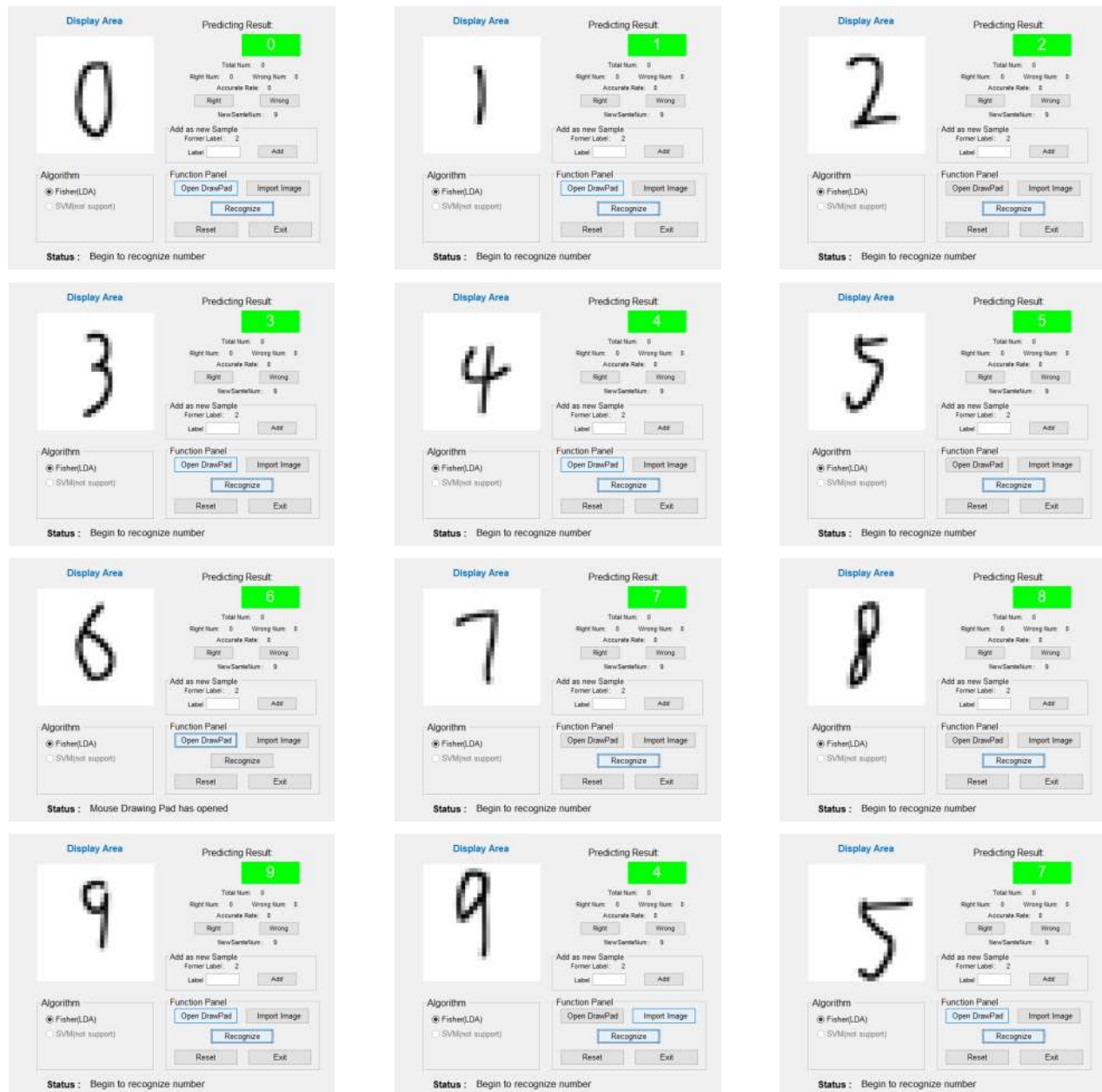
数字	0	1	2	3	4	5	6	7	8	9
正确次数(次)	9	9	9	9	5	5	4	9	7	5
准确率	90%	90%	90%	90%	50%	50%	40%	90%	70%	50%

## 4 结论

在测试集中识别正确率达92%,但仅限于没有旋转的数字,一旦数字发生形变或位置发生偏移,都不能正确的被识别,所以训练出来的模型局限性还是很大。待改进点一:可以对原始的数字图片进行平移、旋转、缩放等操作,以达到数据扩充的目的,增加识别系统的数据量;待改进点二:替换识别算法,利用现在比较先进的深度学习框架,例如GAN网络、带残差块的神经网络等;待改进点三:增加系统中分类算法的选择,此系统采用前后端分离的设计,方便扩展多种分类识别算法,可继续增加SVM、决策树、逻辑回归等算法,来增加此系统识别模块的鲁棒性和识别率。

## 参考文献:

- [1] 孙印祥.现代模式识别[M].2版.北京:高等教育出版社,2008:110-167.
- [2] 杨淑莹.模式识别与智能计算:Matlab技术实现[M].3版.北京:电子工业出版社,2011.
- [3] Shawe-Taylor J, Cristianini N. Kernel methods for pattern analysis.



(前7张为识别正确结果,后2张为识别错误结果)

图6 手写数字系统测试

sis[M].Cambridge University Press, 2004.

- [4] LeCun Y, Cortes C, Burges CJ. The MNIST database of handwritten digits[DB/OL]. <http://yann. lecun. com/exdb/mnist>, 1998:34.
- [5] Fisher R A. The use of multiple measurements in taxonomic problems[J]. Annals of Eugenics, 1936,7(2):179-188.
- [6] Randolph I f. Chromosome numbers in native american and introduced species and cultivated varieties of iris[J]. Bull. Amer. Iris Soc., 1934(52):61-66.
- [7] Mika S, Ratsch G, Weston J, et al. Fisher discriminant analysis with kernels[C]//Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (Cat. No. 98TH8468), Madison, WI, USA. IEEE, 10.1109/

nnspp.1999.788121:41-48.

- [8] Chen Y, Gupta M R. Learning kernels from indefinite similarities[J]. In: ICML, 2009:145-152.
- [9] Lu J W, Plataniotis K N, Venetsanopoulos A N. Face recognition using kernel direct discriminant analysis algorithms[J]. IEEE Transactions on Neural Networks, 2003,14(1):117-126.
- [10] Hofmann T, Schölkopf B, Smola A J. Kernel methods in machine learning[J]. The Annals of Statistics, 2008,36(3):1171-1220.
- [11] 余胜威, 吴婷, 罗建桥. MATLAB GUI设计入门与实战[M]. 北京:清华大学出版社, 2016:1-100.

【通联编辑:唐一东】