

基于 Android 平台的交通标志识别研究

强 杰 王舒憬 单振华

(上海大学上海 200072)

摘要: 通过前期调研和对基于颜色和形状的交通标志检测方法的 MATLAB 仿真实验,提出并实现了基于机器学习的交通标志检测方法。由于自然场景下的交通标志往往具有复杂的背景,基于机器学习的交通标志检测方法能够更准确地识别出自然场景下的交通标志。通过 Android NDK 技术,将 C++ 语言编写的图像处理程序交叉编译生成动态链接库,该动态链接库便可在 Android 端通过 Java 程序调用,解决了常用计算机视觉库在 Android 平台上提供的接口不全、资料不完整导致的无法进行复杂图像处理的问题。实现了将道路交通标志识别算法应用于 Android 系统平台上,并进行了相关功能测试。

关键词: 交通标志; Android NDK; 移动设备; 跨平台; 动态链接库; 仿真实验

中图分类号: TN014; TP391 **文献标识码:** A **国家标准学科分类代码:** 510.8040

Research on the traffic sign recognition based on Android terminal

Qiang Jie Wang Shujing Shan Zhenhua

(Shanghai University, Shanghai 200072, China)

Abstract: Through the preliminary simulation and the Matlab simulation experiment based on the color and shape of the traffic sign detection method, the traffic mark detection method based on machine learning is proposed and realized. Because the traffic signs in the natural scene often have complex background, the traffic detection method based on machine learning can more accurately identify the traffic signs under the natural scene. Through the Android NDK technology, the C++ language image processing program can be cross-compiled into the dynamic link library, which can be called in the Android side by the Java program to solve the problem that the commonly used computer vision library on the Android platform provides incomplete interfaces and incomplete data. The road traffic sign recognition algorithm is applied to the Android system platform, and the related function test is carried out.

Keywords: traffic sign; android NDK; mobile devices; cross-platform; dynamic link library; simulation experiment

0 引 言

现有的道路交通设施和日益增长的交通需求之间矛盾的日益突出,催生了大量关于智能交通的研究。道路交通标志识别的研究,已经成为了智能交通研究的重要组成部分^[1-2]。现有的对于道路交通标志识别算法和应用的研究,大都停留在基于 PC 平台的研究^[3-4],而基于移动设备的研究比较少见。本文通过使用 HOG 特征描述算子结合支持向量机(SVM),将交通标志识别算法应用于 Android 平台,有着很好的理论研究意义和实际应用价值。

1 交通标志检测和识别方法

1.1 基于颜色和形状的检测方法

由于交通标志具有显著的颜色和形状特征,现有的交

通标志检测方法大多基于交通标志的颜色和形状特征^[5-6]。基于颜色和形状检测流程如图 1 所示。

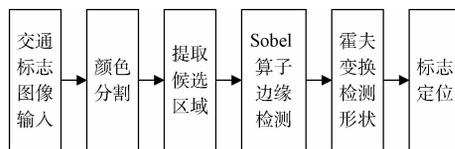


图 1 基于颜色和形状交通标志检测流程

首先进行颜色分割。在这一步中,选取合适的阈值至关重要,而参考大量的相关文献可得到适合禁止交通标志的阈值。颜色分割得到如图 2(b) 图像。

在颜色分割的基础上,对图像进行形状检测。为了有效获取边缘特征,利用边缘算子可提取图像边缘。常用边

边缘提取算子有 Canny 算子、Sobel 算子、Prewitt 算子、Kirsh 算子等。采用 Sobel 算子提取边缘后,得到图 2(c) 所示图像。

接着,进行 Hough 变换得到图像中的圆形,然后对找到的感兴趣区域进行目标判断。常用 Hough 变换来检测图像中的直线,为了检测出图像中的圆形标志,后来学者提出了很多改进的 Hough 变换方法,可以快速检测出图像中的圆形^[7-8]。此时得到两个含有交通标志的圆形区域,观察可知,限速标志和禁止临时或长时停放标志的一个区别是,图案中间是否有贯通的竖直白线。根据这一条件,得到如图 2(d) 所示感兴趣区域。



图 2 基于颜色和形状交通标志检测流程

1.2 交通标志识别方法

本文采用 HOG 特征描述算子和 SVM 分类器结合进行交通标志识别。选用 HOG 算子作为特征描述算子是因为 HOG 特征具有良好的光学不变性和几何形状不变性^[9-10],这在一定程度上减少了拍摄角度和光照对于识别结果的影响^[11-12]。

1.2.1 HOG 特征提取

以 20×20 的窗口为例,介绍 HOG 特征的提取方法^[13-15],其计算步骤如下。

1) 窗口由横竖线分为 5×5 个大小相等的小方块,最小的小方块叫做细胞单元(cell)。4 个相邻的 cell 构成一个较大的区间(block)。如图 3 所示,左上角 4 个 cell 组成一个 block。block 与 block 之间可以重叠,也可以不重叠。

2) 用梯度算子计算各个像素的纵向和横向梯度,分别设为 $H(x,y)$ 和 $v(x,y)$ 。

3) 计算每个像素点的梯度方向和梯度幅度,计算公式为:

$$\theta(x,y) = \tan^{-1}[V(x,y)/H(x,y)] \quad (1)$$

$$m(x,y) = [H(x,y)^2 + V(x,y)^2]^{1/2} \quad (2)$$

式中: $\theta(x,y)$ 为像素 (x,y) 的梯度方向, $m(x,y)$ 为它的梯度大小。

4) $\theta(x,y)$ 取值范围为 $-90^\circ \sim 90^\circ$ 。首先将其平均分成 9 份,然后每个细胞单元中按梯度方向投票,权重为 $m(x,y)$,这样每个 cell 就得到一个 9 维的向量。接着,把同一个块中 4 个细胞单元得到的向量连起来,就得到了一个块的 36 维向量,最后再把所有向量连起来,这就得到了该窗口图像的 HOG 特征向量。

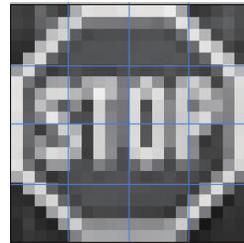


图 3 交通标志分块图像

1.2.2 SVM 分类器训练

使用 dlib 机器学习库进行训练流程如图 4 所示。

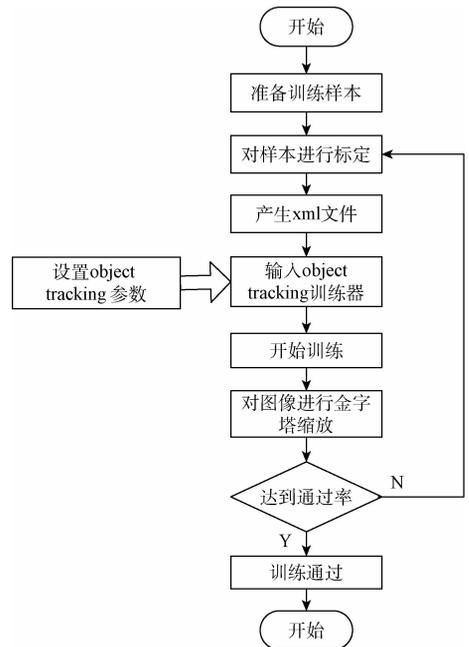


图 4 dlib 训练检测器流程

对 SVM 分类器进行训练时,采用核函数为线性核函数。在通过实验测试后发现,所训练的 SVM 分类器的检测率基本上能够达到实验要求,但制作分类器时间有些长,这将对开发效率带来影响。因此,本节通过 MATLAB 仿真实验测试不同核函数对于训练时间的影响。仿真界面如图 5 所示。

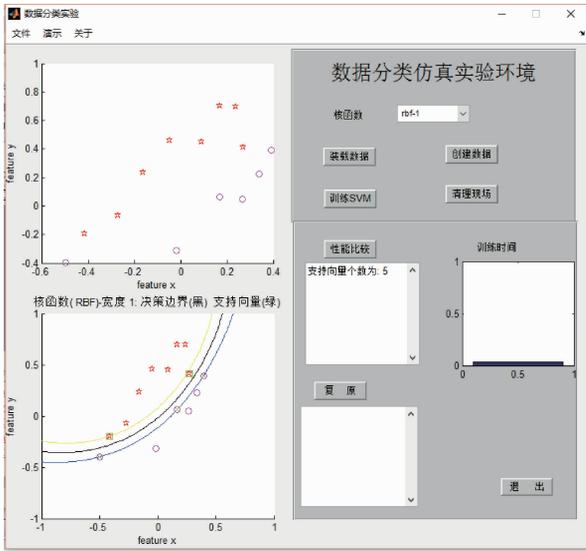


图 5 交通标志分块图像

1) 选用线性核函数,进行实验,得到如图 6 的实验结果,训练 SVM 分类器时间为 46.875 ms。

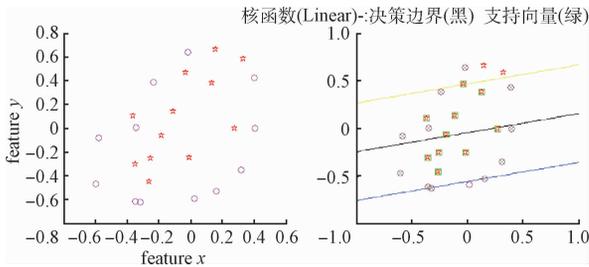


图 6 线性核函数分类器

2) 选用径向基核函数,宽度为 1,进行实验,得到如图 7 的实验结果,训练 SVM 分类器时间为 93.750 ms。

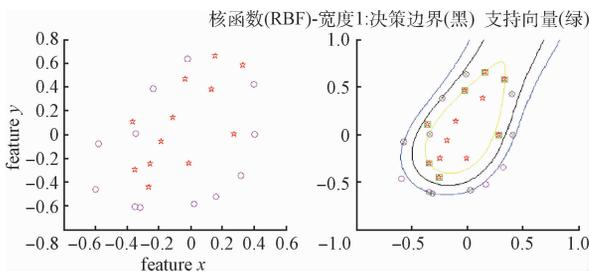


图 7 径向基核函数(宽度 1)分类器

3) 选用径向基核函数,宽度为 8,进行实验,得到如图 8 的实验结果,训练 SVM 分类器时间为 125.000 ms

4) 选用径向基核函数,宽度为 36,进行实验,得到如图 9 的实验结果,训练 SVM 分类器时间为 62.500 ms。

一般来说,训练 SVM 分类器常常需要很长时间。由上述实验结果可以看出,核函数的不同,对于同一样本集训练出的 SVM 分类器所需时间也不同。实验发现,线性核

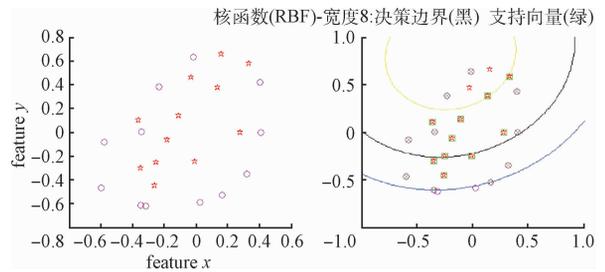


图 8 径向基核函数(宽度 8)分类器

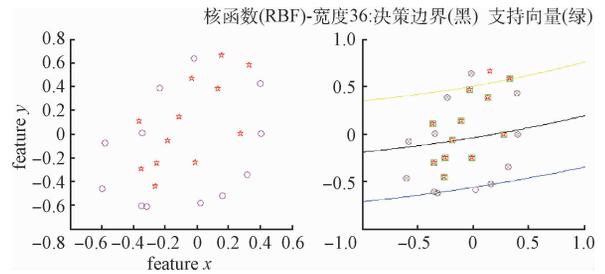


图 9 径向基核函数(宽度 36)分类器

函数相比径向基核函数训练时间较短,而改变径向基核函数的参数仍无法减少训练时间,因此在设计中将继续使用线性核函数作为训练 SVM 分类器的参数。

2 基于 Android 平台的交通标志识别实现

2.1 基于 Android 平台交通标志识别的工作流程

基于 Android 平台进行道路交通标志识别系统工作流程如图 10 所示,首先自行采集交通标志样本制作样本库,对样本库中全部交通标志进行特征提取,训练交通标志检测器以备图像识别时使用。在得到了交通标志检测器的基础上,选取手机系统图库中的任意一张图片,完成对图片的预处理,提取该图片特征,利用训练好的检测器去识别该图片中是否有满足条件的特征区域,若存在,则利用检测器得出该交通标志具体分类,以确定最终结果;若不存在,则不继续进行交通标志识别。

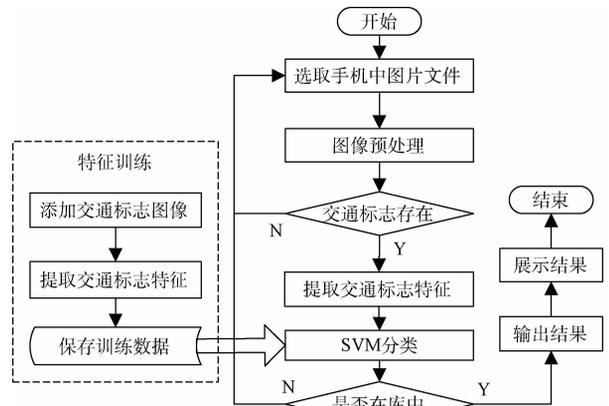


图 10 系统操作流程

2.2 Android 端交互界面

程序运行界面示意图如图 11 所示,当点击界面中的“选择图片”按钮,即可从手机图库中选择图片。当图片选择成功时,界面中主题展示区域即可展示从图库中选择的图片。当点击“识别标志”按钮时,即可检测当前图片中的交通标志,当成功检测到图片中的交通标志时,程序会用红色的方框框出交通标志,并在界面的底栏展示识别结果。



图 11 实际效果

在 Java 端实现了界面 UI、用户标签保存和采集到图像的序列保存,以及在 Native 层进行图像处理之后的图像数据的显示。在 UI 上可以看到,Java 端主要用于展示识别结果,主要图像处理逻辑均在 Native 层进行。刚进入到 App 时,用于看到的将会是默认图片,只有用户成功从图库中选择了图片,该图片才展示到界面上。

选取系统图库中的图片,并展示识别结果。当检测出图像有交通标志时,本设计会用一条红色粗线框框出交通标志所在位置,并把识别结果和识别时间展示在图片下方。

2.3 Android NDK 脚本文件配置

要想在 Android 程序中使用 Native 层代码,必须要通过链接库的形式,而不能通过 Java 代码直接执行 Native 层代码。而链接库有两种:一种是动态链接库,该类库的名字一般是 lib * *.so 形式;另一种是静态链接库,该类库的名字一般是 lib * *.a 形式。静态链接库编译成的文件一般比较大,因为整个函数库的所有数据都会被整合到目标代码中,而动态链接库在编译的时候并没有被编译到目标代码中。

```
static{System.loadLibrary("libdlib_android")}
```

这是程序加载动态链接库的标准用法。库的名字不带 .so,程序会自动寻找 lib 文件夹目录下的对应的库。

Android.mk 配置文件如图 12 所示;Application.mk 配置文件如图 13 所示。

```
dlib_android.cpp | detect.cpp | Android.mk
1 LOCAL_PATH := $(call my-dir)
2 include $(CLEAR_VARS)
3 OPENCV_LIB_TYPE := STATIC
4 include E:/other/OpenCV-2.4.9-android-sdk/sdk/native/jni/OpenCV.mk
5
6 # 编译后生成的模块名,在Java端使用System.loadLibrary加载模块的时候直接使用此名
7 LOCAL_MODULE := dlib_android
8 LOCAL_C_INCLUDES := \
9     E:/other/OpenCV-2.4.9-android-sdk/sdk/native/jni/include \
10    dlib/all/source.cpp \
11    $(LOCAL_PATH)
12
13 # LOCAL_SRC_FILES用于添加源文件(可以是.c,.cpp,.s等)
14 LOCAL_SRC_FILES := dlib/all/source.cpp \
15    dlib_android.cpp \
16    detect.cpp
17 LOCAL_LDLIBS := -lm -llog -ldl -lz -ljnigraphics
18 LOCAL_CPPFLAGS += -fexceptions -frtti -std=c++11
19 LOCAL_ARM_MODE := arm
20
21 # 条件预编译,如果当前架构为ARMv7以下的架构,执行以下代码
22 ifeq ($(TARGET_ARCH_ABI), armeabi-v7a)
23 LOCAL_ARM_NEON := true
24 endif
25 LOCAL_CFLAGS += -pie -fPIE
26 include $(BUILD_SHARED_LIBRARY)
```

图 12 Android.mk 配置文件

```
dlib_android.cpp | detect.cpp | Android.mk | Application.mk
1 APP_STL := gnustd_static
2 APP_CPPFLAGS := -frtti -fexceptions
3 # 使用当前NDK编译器所支持的所有处理器架构
4 APP_ABI := armeabi-v7a
```

图 13 Application.mk 配置文件

3 实验验证

本文已经详细阐述了介绍了基于 Android 平台上交通标志识别系统的实际应用。为了证明算法的有效性和可靠性,本节通过对不同情况下的图像进行试验验证,试验表明本文算法的有效性和可靠性。

本节实验验证是基于 MI-3 手机,该手机具体硬件参数如下:

版本号:Android OS 4.4.2

处理器:NVIDIA Tegra4

分辨率:1 920×1 080

RAM:2 GB

本节将试验验证图像大小、交通标志类型、拍摄角度等几个因素对识别结果的影响,以为进一步研究提供依据。

首先对不同大小的同一场景图像进行测试。通过表 1 可以看出,图像识别结果受图像大小影响不大,而识别时间随着图像大小近乎线性增长。然后对不同种类的交通标志进行测试。通过表 2 可以看出,交通标志类型对于识别结果和识别时间影响不大,而只受检测器的可靠度影响。检测器训练时收集的样本越全面,识别效果越好。接着对不同拍摄角度的图像进行测试。由表 3 可以看出,当拍摄角度大于一定角度时,系统无法检测出图像中交通标志,更无法进行交通标志识别。

最后将 327 张含有交通标志的自然场景下的图片进行测试,并且所有图片中含有的交通标志均在训练检测器时进行过训练。其中 3 张图片因拍摄过于模糊而识别出错,

表 1 图像大小对识别结果影响

| | | | | |
|---------|-------|-----|-----|-----|
| 测试结果/KB | 1 536 | 880 | 544 | 80 |
| 识别结果 | 正确 | 正确 | 正确 | 正确 |
| 识别时间/ms | 1511 | 805 | 453 | 221 |

表 2 交通标志类型对识别结果影响

| | | | | |
|---------|------|------|-----|-----|
| 测试结果 | 禁止左转 | 注意危险 | 直行 | 慢行 |
| 识别结果 | 正确 | 正确 | 正确 | 正确 |
| 识别时间/ms | 484 | 431 | 410 | 399 |

表 3 拍摄角度对识别结果影响

| | | | | |
|----------|-----|-----|-----|-----|
| 测试结果/(°) | 0 | 15 | 20 | 79 |
| 识别结果 | 正确 | 正确 | 正确 | 错误 |
| 识别时间/ms | 471 | 462 | 417 | 327 |

其余图片均识别正确。因此计算得到本设计的交通标志识别准确率为 $324/327 = 99.083\%$, 达到系统设计时的预期目标。

4 结 论

针对现有对于道路交通标志识别的研究没有基于移动设备上的应用问题, 创新性地基于 Android 平台进行道路交通标志识别, 利用 HOG 结合 SVM 进行交通标志特征提取和识别, 取得了较精确的识别结果。在实际的工业验证过程中, 拍摄图像过大、拍摄角度过大以及图像遮盖问题都给道路交通标志识别带来了挑战, 这也是下一步的研究重点。

参考文献

- [1] 张紫谦, 袁卫声, 梁凯. 基于 Android 终端的物联网家居环境监测系统设计[J]. 测控技术, 2016, 35(2): 108-111.
- [2] 黄青丹, 张亚茹, 裴利强, 等. 基于图像识别的仪表自动检定系统[J]. 测控技术, 2016, 35(10): 153-156.
- [3] 杨帆, 赵东东. 基于 Android 平台的 WiFi 定位[J]. 电子测量技术, 2012, 35(9): 116-119, 124.
- [4] 丛焕武, 郭福娟, 吕飞, 等. 基于 CCD 图像处理的焊缝

识别技术研究[J]. 电子测量技术, 2012, 35(3): 73-77.

- [5] 魏星. 基于 SVM 的山体滑坡灾害图像识别方法[J]. 电子测量技术, 2013, 36(8): 65-70.
- [6] 温浩, 刘力源. 基于小波包和 SVM 一类学习在电机异音检测中的应用[J]. 测控技术, 2015, 34(2): 35-38.
- [7] 张海娜, 马玉芳. 基于图像处理的机器播种参数检测方法研究[J]. 测控技术, 2015, 34(2): 44-47.
- [8] 纪蕾娟. 自然场景下交通标志检测算法的研究[D]. 上海: 华东师范大学, 2013.
- [9] 卢艳君. 交通标志自动检测与识别算法研究[D]. 武汉: 武汉科技大学, 2015.
- [10] 邵桂珠. 道路交通标志识别算法的研究[D]. 长春: 吉林大学, 2008.
- [11] 徐超, 高梦珠, 查宇锋, 等. 基于 HOG 和 SVM 的公交乘客人流量统计算法[J]. 仪器仪表学报, 2015, 36(2): 446-352.
- [12] 宋晓琳, 郑亚奇, 曹昊天. 基于 HMM-SVM 的驾驶员换道意图辨识研究[J]. 电子测量与仪器学报, 2016, 30(1): 58-65.
- [13] 戴雪梅, 郎朗, 陈孟元. 基于改进 ORB 的图像特征点匹配研究[J]. 电子测量与仪器学报, 2016, 30(2): 233-240.
- [14] 许佳佳, 张叶, 张赫. 基于改进 Harris-SIFT 算子的快速图像配准算法[J]. 电子测量与仪器学报, 2015, 29(1): 48-54.
- [15] 周美丽, 白宗文. 基于形状特征的图像检索系统的设计[J]. 国外电子测量技术, 2015, 34(6): 82-84.

作者简介

强杰, 硕士研究生, 主要研究方向为 Android 应用软件开发、嵌入式应用软件开发道路交通标志处理与识别等。

王舒憬, 硕士, 副教授, 主要研究方向为嵌入式软件的研究应用。

E-mail: wshujing@shu.edu.cn

单振华, 硕士研究生, 主要研究方向为 Android 应用软件开发、嵌入式应用软件开发道路交通标志处理与识别等。