

# 交通标志识别算法模型的研究与实现

徐彬森<sup>1</sup>, 魏元周<sup>1</sup>, 毛光明<sup>1</sup>, 李曼曼<sup>2</sup>

(1. 北京航空航天大学软件学院, 北京 100191; 2. 河南财经政法大学计算机与信息工程学院, 河南 郑州 450046)

**摘 要:** 本文通过比较交通标志检测和分类算法, 利用阈值分割和神经网络算法思想构建了一个交通标志识别模型, 该模型对于 GTSRB 数据集上的交通标志识别图片识别错误率能控制在 5% 以内, 处理每帧的识别过程在 150ms 左右, 较好地实现了交通标志的实时检测与分类。为相关交通管理部门提供了一套方便的管理技术。

**关键词:** 交通标志识别; 阈值分割; 神经网络

**中图分类号:** TP391(TN911.73) **文献标识码:** A **DOI:** 10.3969/j.issn.1003-6970.2017.11.015

**本文著录格式:** 徐彬森, 魏元周, 毛光明, 等. 交通标志识别算法模型的研究与实现[J]. 软件, 2017, 38 (11): 74-81

## Research and Implementation of Traffic Sign Recognition Algorithm

XU Bin-sen<sup>1</sup>, WEI Yuan-zhou<sup>1</sup>, Mao Guang-ming<sup>1</sup>, LI Man-man<sup>2</sup>

(1. BeiHang University (Beijing) Software College, Beijing 100191, China; 2. Henan University of Finance and Economics (Zhengzhou) Computer and Information Engineering College, Zhengzhou 450046, China)

**【Abstract】:** In this paper, by comparing the traffic sign detection and classification algorithm, using the ideas of threshold segmentation algorithm and neural network, researchers constructed a traffic sign recognition model, the model for GTSRB data sets on traffic sign recognition image recognition error rate can be controlled within 5%, the processing of each frame recognition process is controlled in 150 ms, implement the real-time detection and classification of traffic signs well. The researchers provided a convenient management technology for related traffic management department.

**【Key words】:** Traffic sign detection; Threshold segmentation; Convolutional neural network (CNN)

## 0 引言

据统计, 中国每年均有超过 15 万以上数量的人因公路交通事故而丧失生命。这其中, 无道路标识等控制方式场景下发生的交通事故死伤人数占总死伤人数的 30.83%, 见图 1。道路交通标识设置的合理与否, 直接关系到道路交通状况<sup>[1]</sup>。现阶段对交通标识的统计与管理方法依然是通过交通管理部门对各条路段与路口查明并记录后上报, 工作量大, 效率有限。交通标识检测及分类属于交通标识识别的两个重要方面, 该类研究起源于上世纪 80 年代, 特别是近十几年来, 随着计算机视觉、机器学习等领域的进步, 以及计算机性能的提升, 相关技术领域的研究热度增加并涌现出许多好的研究成果<sup>[2]</sup>。

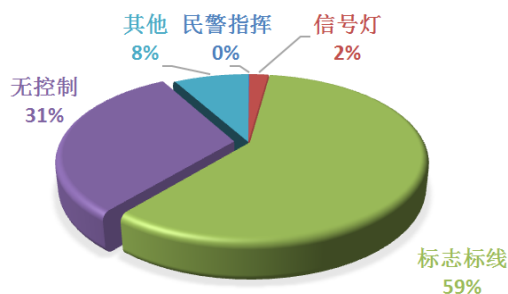


图 1 无道路标识等控制方式的事故死伤比率  
Fig.1 No road marking and other control methods of accident casualties ratio

本文第一部分回顾了交通标志识别研究状况及常用方法。文章第二部分是对交通识别流程和关键点的概述, 第三部分是交通识别算法模型的设计, 第四部分则是算法模型实现效果, 第五部分则是对

**作者简介:** 徐彬森(1992-), 男, 硕士研究生, 主要研究方向: 软件工程、图像识别与处理; 魏元周(1992-), 男, 硕士研究生, 主要研究方向: 软件工程、数据挖掘、图像识别与处理。毛光明(1991-), 男, 硕士研究生, 主要研究方向: 软件工程、深度学习。李曼曼(1992-), 女, 硕士研究生, 主要研究方向: 软件工程、数据挖掘、图像识别与处理。

全篇的回顾和总结。

## 1 交通标志识别介绍

### 1.1 相关研究回顾

交通标志识别算法根据内容也都大多采用两个处理步骤：交通标识检测以及交通标识分类<sup>[3]</sup>。一般来说，交通标识的检测有三种做法：基于颜色分割、基于形状信息和基于机器学习的方法<sup>[4]</sup>。基于颜色分割的算法简单、计算速度快、对几何形变不敏感，但在低光照或逆光环境，相似背景的场景等场景时缺点很明显，因为颜色是不可靠的信息，在不同时间，不同光照下采集到的颜色各不相同。而基于形状的算法通常计算代价大，并且在场景中出现相似形状的时候不能很好的处理。同时在实时场景下，还存在车辆抖动而造成的画面模糊等情况。因为交通标识一般具有醒目的颜色以及规则的形状，所以基于机器学习的算法一般也是针对这两个方面实现不同的分类器。一方面，因为对二维图像进行阈值分割本质上是对二维平面上的像素点进行分类的过程，机器学习中经典的分类算法，比如 SVM、K-Means 等算法均可以有所应用。另一方面，由于标识牌规则的形状，检测标识实际上也是形状识别算法的一种应用。Timofte 等人<sup>[5]</sup>将 Viola-Jones 提出的，一个级联类 Haar 分类器用到 6 种不同的标识类别上，实现了较好的实时效果。这些都是在应用中值得借鉴的地方。

将标识牌检测出来之后，关键目标是精确识别标识的具体类型，其本质上是一个分类的过程。对图像的分类，一些算法需要先对图像尺寸进行归一化、高斯模糊等预处理，然后进行分类，另一些基于机器学习的算法可以直接通过主动学习对图片进行分类。大体上常用的算法有：模版匹配、决策树、SVM、神经网络等几类。但是其对于分类的准确度还有提高的空间。

基于以上因素，本文选取 HSV 颜色模型，采用图像处理中的阈值分割等算法及 BP 神经网络等算法，构建检测分类模型，通过训练样本集，实现对测试集中交通标志的识别，并与该领域其他算法比较。

### 1.2 交通标志识别的基本流程

本文认为道路交通标识识别流程分为三个阶段：(1)预处理，将图像转化为易快速处理的“数据”；(2)交通标识的检测，包括对视频图像标识候选区域的定位、特征提取等；(3)交通标识的分类，主要是将前一步检测过程中提取的标识候选区域进行

精确分类。如图 2 所示。

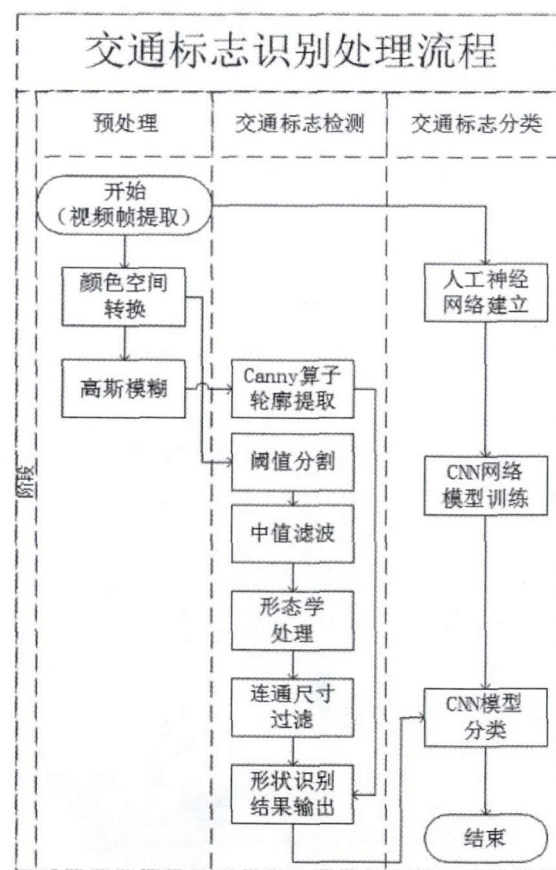


图 2 交通标志识别基本流程

Fig.2 The basic process of traffic signs recognition

## 2 交通标志识别算法模型设计

### 2.1 预处理

研究实验中道路视频数据采集设备采集的数据为 RGB 模型的视频数据，首先要对视频进行分解，获取每一帧的图像，然后对图像进行初步预处理：

#### 步骤 1 颜色空间转换：

颜色空间模型是颜色的各种数学表示方式。HSV 模型较 RGB、XYZ、CIE 等模型可以较为理想地进行颜色处理<sup>[6]</sup>。以 RGB 颜色模型为例，RGB 三个分量都是与光照相关的，这让 RGB 颜色模型在标识识别中的应用受到了限制，由于自然条件下光照条件变化多样，RGB 颜色模型相对来说不适合用在交通标识的检测中。

HSV 模型的 H 表示色相 (Hue)，通常取值范围为 [0, 360]，对应红橙黄绿青蓝紫-红这样顺序的颜色，S 分量表示饱和度 (Saturation)，即色彩的纯净程度，V 表示明度，即颜色明亮的程度。HSV 与 RGB 三个分量之间的转化关系如下：

$$h = \begin{cases} 0^\circ & \max = \min \\ 60^\circ \times \frac{g-b}{\max-\min} + 0^\circ & \max = r \text{ 并且 } g \geq b \\ 60^\circ \times \frac{g-b}{\max-\min} + 360^\circ & \max = r \text{ 并且 } g < b \\ 60^\circ \times \frac{b-r}{\max-\min} + 120^\circ & \max = g \\ 60^\circ \times \frac{r-g}{\max-\min} + 240^\circ & \max = b \end{cases} \quad (1)$$

$$s = \begin{cases} 0 & \max = 0 \\ \frac{\max - \min}{\min} & \text{其他情况} \end{cases} \quad (2)$$

$$v = \max \quad (3)$$

其中  $\max$  是 RGB 三个分量的最大值,  $\min$  是三个分量中的最小值。HSV 颜色空间是一个倒圆锥体, 一周表示色相 (Hue), 自中心到边缘表示饱和度 (Saturation), 圆锥体尖端表示明度为 0, 最上部分表示明度最大。如图 4 所示。

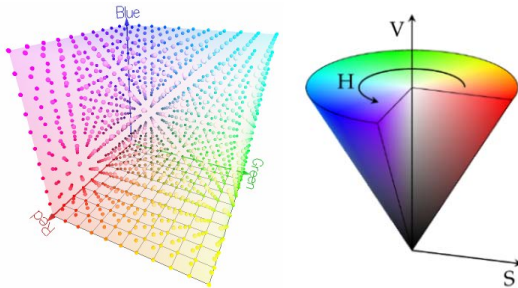


图3 RGB 颜色模型 (左) 和 HSV 颜色模型 (右)  
Fig.3 RGB color model (left) and HSV color model (right)

## 步骤2 高斯模糊:

高斯模糊 (Gaussian Blur), 也称为高斯平滑, 主要用来平滑图片, 减少图像中的噪点。数学角度来讲, 高斯模糊是一个卷积的过程, 所使用的的是一个呈正态分布的卷积核, 它用正态分布计算图像中每个像素的变换, 在二维空间中定义为:

$$G(u, v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}} \quad (4)$$

其中  $r$  是模糊半径 ( $r^2 = u^2 + v^2$ ),  $\sigma$  是正态分布的标准偏差, 实验中选取了  $\sigma=2$ 。图 4 为使用 (9, 9) 的卷积核对图像做高斯后的结果, 高斯模糊减少图像的噪点, 从而减少噪点对算法的影响, 增强算法的鲁棒性。

## 2.2 基于阈值分割算法的交通标志检测

### 步骤1 阈值分割:

阈值分割基本原理是: 通过设定不同的特征阈



图4 高斯模糊处理示意图  
Fig.4 Gaussian fuzzy processing diagram

表1 HSV 空间各颜色分割阈值  
Table 1 HSV space color separation threshold

颜色	H	S	V
红色	$h > 160$ or $h < 20$	$s > 100$	$v > 90$
蓝色	$90 < h < 126$	$s > 100$	$v > 100$
黄色	$90 < h < 33$	$s > 90$	$v > 100$

值, 把图像像素点分为若干类。研究中将经过 HSV 颜色模型转换的图像通过阈值分割为三类。通过实验测试得知, 表 1 的分割得到的分割效果最好。

### 步骤2 中值滤波:

中值滤波器将图像中每一点的像素值由对滤波区域  $R$  内的像素值的中值代替,

$$I'(u, v) \leftarrow \text{median} \{ I(u+i, v+j) | (i, j) \in R \} \quad (5)$$

$2K+1$  个像素点  $P_i$  的中值定义为:

$$\text{median}(P_0, P_1, \dots, P_k, \dots, P_{2k}) \triangleq P_k \quad (5-1)$$

即数列  $(P_0, P_1, \dots, P_k, \dots, P_{2k})$  以升序 ( $P_i \leq P_{i+1}$ )

排列, 那么其中值为  $P_k$ 。如果像素个数为偶数 ( $2K$ , 其中  $K > 0$ ), 那么对于一个顺序数列 ( $P_0, P_1, \dots, P_k, \dots, P_{2k-1}$ ), 那么其中值定义为中间两个元素的算术平均值,

$$\text{median}(P_0, P_1, \dots, P_k, \dots, P_{2k-1}) \triangleq (P_{k-1} + P_k) / 2 \quad (5-2)$$

### 步骤3 形态学处理:

形态学运算是数学中的形态学 集合论中的方法, 用于作二值图像处理。形态学 (Mathematical Morphology) 运算常见算法有腐蚀、膨胀以及基于腐蚀、膨胀的开运算、闭运算等, 给定输入的二值图像  $I(x, y)$  以及作为处理窗口模板  $T(i, j)$ , 则腐蚀运算可以表示为:

$$E(x, y) = (I \oplus T)(x, y)$$

$$\bigcap_{i,j=0}^m [I(x+i, y+j) \& T(i, j)] \quad (6)$$

膨胀可以表示为:

$$D(x, y) = (I \oplus T)(x, y)$$

$$\bigcup_{i,j=0}^m [I(x+i, y+j) \& T(i, j)] \quad (7)$$

开运算指先膨胀后腐蚀, 该运算可用来消除细



小区域以及纤细的连通区域，而闭运算正好相反，指先腐蚀后膨胀，主要用来清除物体内部的细小孔洞。利用开运算可以去除大量不相关的细小区域，不破坏感兴趣的标识区域，对于去除阈值分割之后的大量细小噪点的效果明显。

如图 5 所示，展示了原始图像（左上）经过阈值分割之后的结果（右上），然后通过腐蚀（左下）和膨胀（右下）操作的结果，可以看出，阈值分割之后细小的噪点被有效过滤。

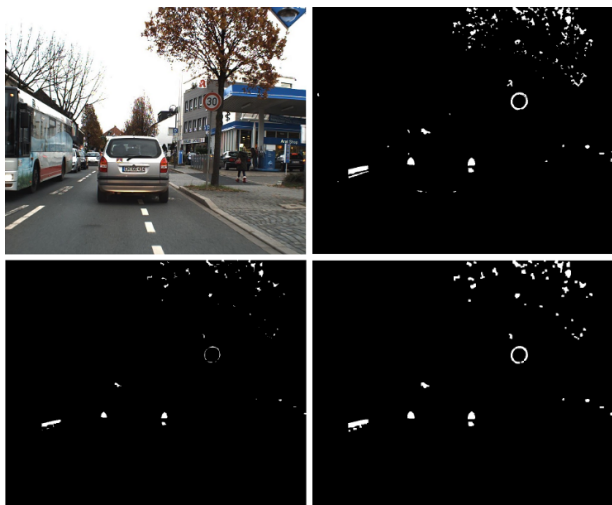


图 5 连通区域尺寸过滤  
Fig.5 Connectivity area size filter

#### 步骤 4 连通区域尺寸过滤

首先标记连通区域，即把结果中连在一起的像素分到同一类别，做上标记，不同的连通区域做不同的标记。在图像处理中，两个像素点判断为连通区域，当且仅当两个像素点相邻，并且像素值一致。二值图像中，处理像素值为 1 的像素点的连通性，一般有三个判断方法：4-邻域、8-邻域、混合邻域等。本文采用 8-邻域连通区域标记法对二值图像进行标记，并统计出各个连通区域的宽度和高度。之后应用以下规则进行过滤：

1. 由于视角影响，拍摄到的标识通常有一定程度的倾斜，为了程序的兼容性，标识宽高比范围大致  $\left(1.9, \frac{1}{1.9}\right)$  之间。

2. 由于交通标识在视频视野中一般较小，并且人眼视觉最多可以识别图片中大于 7 个像素点的标识牌，我们假设整体图像较短边的长度为 1，使用  $\left(\min(7, \frac{1}{20}), \frac{1}{2}\right)$  作为标识牌可能出现的尺寸。

#### 步骤 5 轮廓提取

边缘检测是一种重要的图像预处理手段，常见的手段有 Sobel、Roberts 等<sup>[7]</sup>，本文使用 Canny 算子进行边缘检测主要分为三步：首先使用高斯平滑对图像做预处理，有效抑制噪声，然后计算当前点的梯度值与梯度方向角，此处使用 Sobel 算子来计算，并且将同一方向上最大的梯度值保留，最后使用双阈值来过滤边缘点，只有梯度值大于高阈值或者大于低阈值且 3、5 邻域内存在大于高阈值的梯度值，当前点才被标记为边缘点。如图 6 所示。



图 6 Canny 算子轮廓提取示意图  
Fig.6 Canny operator contour extraction diagram

#### 步骤 6 形状识别

交通标识可能的形状只有可能是三角形，矩形以及圆形。在提取了闭合轮廓之后，通过计算闭合轮廓的曲率，即可判断形状类别。三角形，矩形有多个拐点，其余部分是直线，而圆形没有拐点，即三角形和矩形轮廓边界的切线方向存在突变，而圆形的轮廓边界不存在切线方向突变的情况，而衡量切线变化的量，正是曲率。如图 7（左）所示，选定逆时针为正方向，在 B 点的曲率可以表示为  $\overline{AB}$  与  $\overline{BC}$  的夹角。之后我们通过检测曲率序列 C 中的峰值点个数以及周期关系，即可得到轮廓的类型。图 7（右）为标准正三角形的曲率模板。

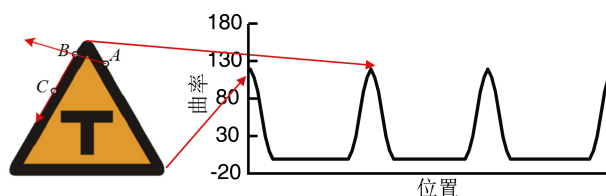


图 7 三角形的曲率模板  
Fig.7 Curve of curvature of the triangle

通过以上步骤，已基本完成了交通标志检测阶段的工作。

### 2.3 基于神经网络算法的交通标志分类

近年来，神经网络，特别是卷积神经网络在图片处理领域取得重要突破，比如人脸分类识别研究<sup>[8]</sup>。交通标识分类阶段工作通过载入训练好的模型，对

候选区域进行分类,即获取每个检测出来的标识的类别。它包含的内容为:(1)载入训练好的模型;(2)使用模型进行分类;(3)保存分类过后的类别信息。如图 8 所示。

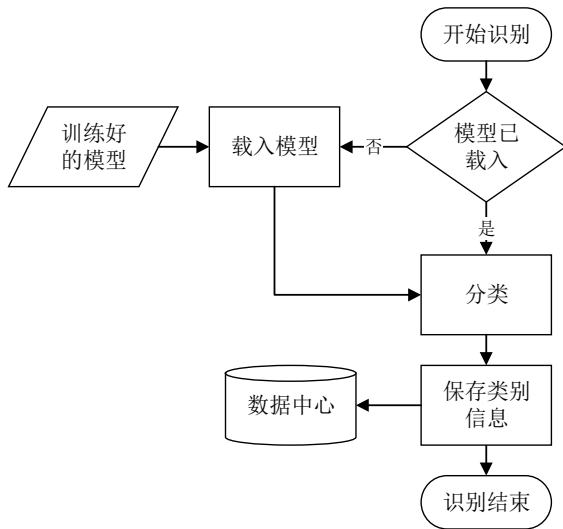


图 8 交通标识分类流程图

Fig.8 Traffic identification classification flow chart

采用人工神经网络模型算法是因为该算法结合了人类大脑的许多优良特征,如(1)信息处理并行化,效率极高。(2)具有很好的容错性能,对损伤冗余。(3)可以无监督的学习,善于归纳推广,这也是本文选取该方法的原因。

在现有神经网络模型中,简单及有效的两种神经网络分别是 LeCun 等人于上个世纪 90 年代提出的 LeNet-5<sup>[9]</sup>和 Alexander 在 2012 年提出的 AlexNet<sup>[10]</sup>,本文采用标识牌识别领域著名的数据集 GTSRB 来进行试验,验证本文网络与 LeNet-5 和 AlexNet 网络的性能。GTSRB 数据集由两部分组成,总共 43 类标识,第一部分包含 39209 个训练样本,第二部分含有 12630 个标记过的验证样本。

LeNet-5 网络由 7 层组成,如图 9 所示,使用输入 32x32 的图像,通过一系列卷积层和子采样之后,得到输出,由于原始的 LeNet 是用来分类手写数字的,输出只有 10 维,这里最后一步使用 SoftMax 层得到 43 维向量作为输出来分类标识。

AlexNet 网络则复杂得多,最开始适用于分类图片使用的,该网络使用 3 通道的彩色图片,尺寸为 224x224 的彩色图像,由于识别标识只有 43 类,最后一步采用 SoftMax 层得到 43 维向量。如图 10。

文章采用一个两个阶段的卷积神经网络,如图 11 所示,每个阶段都由一个卷积层和一个池采样层组成,池采样层减少了空间分辨率,让算法对几何形变和运动模糊等情况鲁棒性更好。

在传统的卷积神经网络里,最后一层的输出会全连接得到一个分类器。本文将所有阶段的输出都全连接到最后的分类器中,使得模型的分类器不仅可以利用高层次的全局的特征,同时还可以保留很

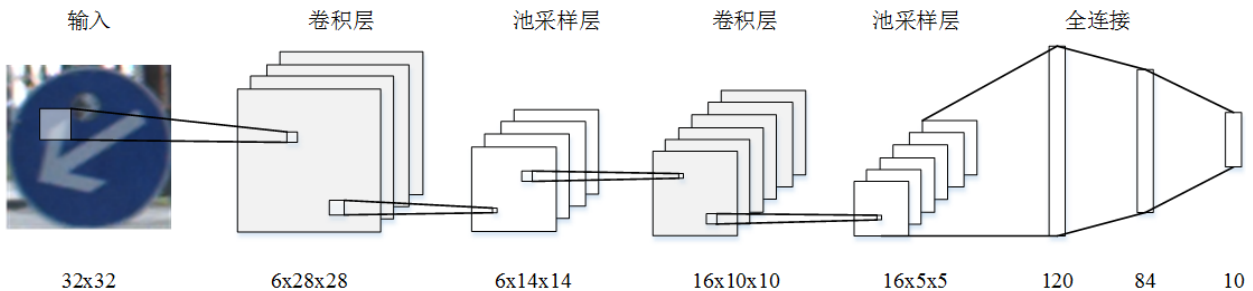


图 9 LeNet-5 的结构

Fig.9 LeNet-5 Architecture

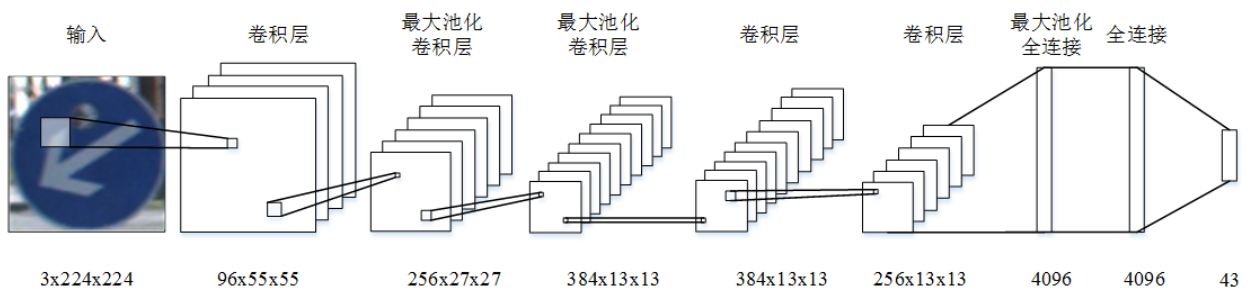


图 10 AlexNet 的结构

Fig.10 AlexNet Architecture

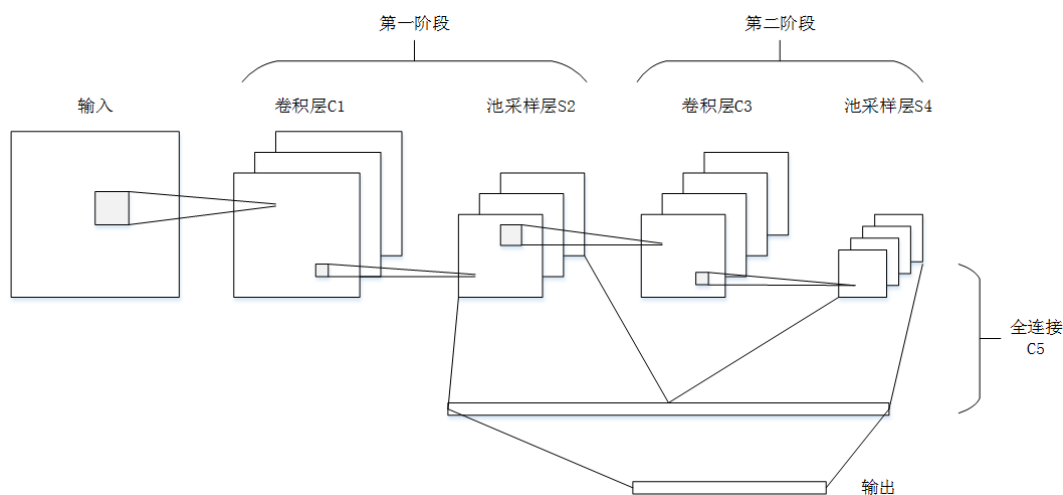


图 11 两个阶段的卷积神经网络

Fig.11 Two-stage convolution neural network

多池化过的低层次的特征，这些特征让算法对局部特征更加敏感，保留了很多细节信息。

具体参数如表 2 所示。该网络输入数据为  $48 \times 48$  的交通标识图像。卷积层 C1 和卷积层 C3 均采用  $5 \times 5$  的卷积核，使用 Relu 函数充当激励函数，采样层 S2 与 S4 均采用最大池化采样。最后全连接层 C5 采用全连接的方式，连接 S2 和 S4 的输出，产生一个 512 维的向量。最后一层输出层，采用全连接的形式使用 SoftMax 分类器生成一个 43 维的向量输出，该参数可以根据训练集中标识的类别数目进行相应的调整。

表 2 本文使用的网络模型参数  
Table 2 The network model parameters used in this article

层	类型	特征数	神经元个数	卷积核/采样尺寸
0	输入层		$48 \times 48$	
1	卷积层	96	$44 \times 44$	$5 \times 5$
2	最大池采样	96	$11 \times 11$	$3 \times 3$
3	卷积层	128	$7 \times 7$	$5 \times 5$
4	最大池采样	128	$2 \times 2$	$3 \times 3$
5	全连接		512	
6	输出层		43	

卷积神经网络训练所使用的数据均来自公开的 GTSRB 数据集<sup>[11]</sup>，训练过程先对图像尺寸进行归一化处理，由于模型主要学习的特征是图片之中细小的可重复的结构，这种特征对颜色不敏感，为了便于计算，将多通道的彩色图像转化为单通道的灰度图像，转化公式如下：

$$V_{\text{gray}} = 0.299R + 0.587G + 0.114B \quad (17)$$

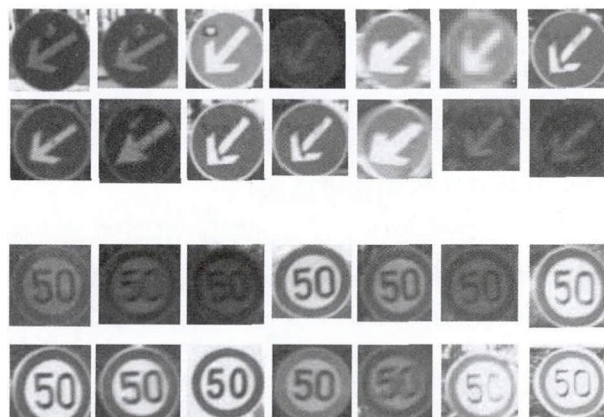


图 12 训练数据示例图

Fig.12 Examples of training data

训练好的模型参数可以保存下来，识别过程中检测到的候选区域作为输入，如图 13 所示，通过检测过程确定候选区域位置之后便可提取出标识所在的区域，通过裁剪、归一化、转化为灰度图像之后，输入训练好的模型，计算得到最终输出。

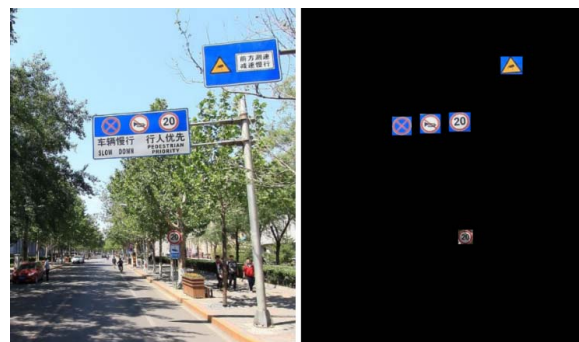


图 13 输入的候选区域示例

Fig.13 Examples of the candidate area entered



### 3 交通标志识别算法模型结果分析

#### 3.1 实验环境说明

操作系统为 Ubuntu14.04 版的 64 位 Linux。离线训练使用的实验环境如表 3 所示。

表 3 离线训练模块测试环境  
Table 3 Offline training module test environment

CPU	Intel Xeon® CPU E5-2690 2.9GHz 32 核
内存	64G
显卡	Tesla k20m

数据预处理、标识检测、标识识别模块使用的实验环境如表 4 所示。

表 4 数据预处理、标识检测识别模块测试环境  
Table 4 Data preprocessing, signs detection identification module test environment

CPU	Intel Xeon® CPU X5355 2.67GHz 2 核
内存	16G
显卡	Quadro FX3500

#### 3.2 交通标志检测阶段结果

标识检测主要使用了安徽和县 S105 道路视频以及作者在北航校园采集的部分视频作为测试数据集，如下图 14（第一行为原图，第二行为检测图）。



图 14 部分测试视频截图  
Fig.14 Partial test video screenshots

运行效果如下。

视频源	分辨率	平均每帧检测时间
安徽和县 S105 道路视频	1712×963	103 ms
北航校园	800×600	56 ms

由测试结果可知，接近 1080P 的视频可以达到 10 帧/s 的处理速度，已经达到了实时的效果。

#### 3.3 交通标志分类阶段结果

标识识别阶段，GTSRB<sup>[11]</sup>测试集中有 12630 张测试数据，尺寸从 20×20 到 100×100 不等，部分测试用例如图 15。



图 15 GTSRB 测试集示例  
Fig.15 Example of a GTSRB test set

训练机器如表 3 所示，训练集合为 GTSRB 数据集，其中包括 39209 张标记好的训练集以及 12630 张测试集。每次迭代使用 64 张图片做训练来更新参数，每 613 次迭代左右可以完全训练完一遍所有的图片，每次训练完一遍所有训练集，在测试集上做一次错误率统计。训练过程中错误率变化如图 16 所示。对该测试机进行一遍识别，平均耗时 630 s，平均每张图片识别耗时约 50 ms，完全可以满足实时识别的需求。

由上图可以得知，在迭代次数越来越多时，错误率快速下降并慢慢趋于平稳在 5% 左右，说明该模型有良好的收敛性，相关指标表现良好<sup>[12]</sup>。

本文模型与其他几种经典的神经网络算法模型在识别正确率方面的比较如表 5 所示。本文提出的算法模型相比之前两个模型在 GTSRB 上具有一定优势。

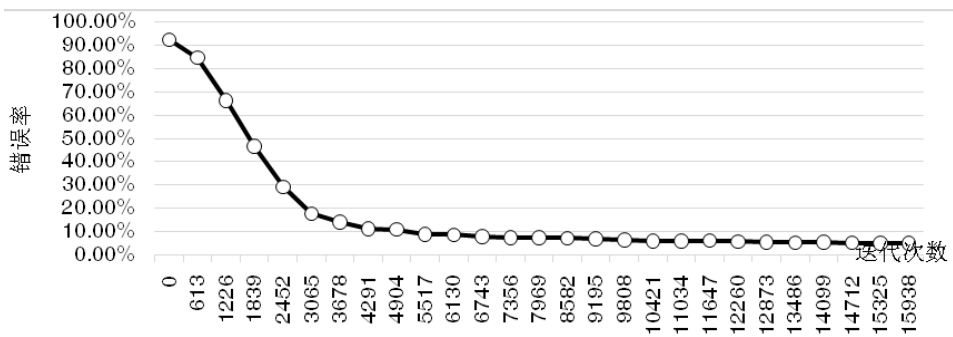


图 16 错误率与迭代次数关系  
Fig.16 The relationship between the error rate and the number of iterations

表 5 几种网络在验证数据集上识别正确率比较  
Table 5 Several networks' accuracy comparison on the verification dataset

网络	验证集识别正确率
LeNet-5	87%
AlexNet	91%
本文网络	95%

4 小结

鉴于交通标识牌在颜色和形状上都有着非常鲜明的特征，本文主要从颜色阈值分割和形状检测两个方面提取出图像中交通标识的候选区域。同时，随着近年来人工智能的发展，卷积神经网络最为一种成熟高效的方法，本文使用卷积神经网络对候选区域做进一步精确的分类。实验结果表明，本文提出的交通标志算法模型能够对交通标志实现较好地识别，多次迭代后保证 95% 的识别正确率，每帧视频检测阶段耗时 100 ms，分类阶段耗时 50 ms。达到了实时识别的要求。

参考文献

[1] 杨钧等, 中华人民共和国道路交通事故统计年报[M], 2012, 无锡: 公安部交通管理科学研究所.

[2] Loy, G. and N. Barnes. Fast shape-based road sign detection for a driver assistance system[J]. in Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on. 2004.

[3] Salti, S., et al., Traffic sign detection via interest region extraction[J]. Pattern Recognition, 2015. 48(4): p. 1039-1049.

[4] Brkic, K., An overview of traffic sign detection methods[J]. Department of Electronics, Microelectronics, Computer and Intelligent Systems Faculty of Electrical Engineering and Computing Unska, 2010. 3: p. 10000.

[5] Timofte, R., K. Zimmermann and L. Van Gool, Multi-view traffic sign detection, recognition, and 3d localization[J]. Machine Vision and Applications, 2014. 25(3): p. 633-647.

[6] 周明等, 基于HSV模型的运动目标提取与跟踪[J]. 指挥控制与仿真, 2010(02): 第93-96页.

[7] 钟彩, 边缘检测算法在图像预处理中的应用[J]. 软件, 2013, 34(1): 158-159.

[8] 杨燕, 刘刚, 张龙. 基于2DPCA和LDA的人脸图像预处理与RBF神经网络的人脸图像识别研究[J]. 软件, 2014(2): 115-118.

[9] Y Lecun, B Boser, JS Denker, etc. Backpropagation Applied to Handwritten Zip Code Recognition[J]. Neural Computation, 2014, 1(4): 541-551.

[10] A Krizhevsky, I Sutskever, GE Hinton. ImageNet classification with deep convolutional neural networks[J]. International Conference on Neural Information Processing Systems, 2012, 25(2): 1097-1105.

[11] Y Wu, Y Liu, J Li, H Liu, etc. Traffic sign detection based on convolutional neural networks[J]. International Joint Conference on Neural Networks, 2014: 1-7.

[12] 张明军, 俞文静, 袁志, 等. 视频中目标检测算法研究[J]. 软件, 2016, 37(4): 40-45.