

# Traffic Signal Recognition Based on Single-Order Neural Network and Feature Fusion Method

Zengrong Wang

School of Mechanical and Vehicle Engineering, Taiyuan University of Technology, Taiyuan Shanxi  
Email: wzrfiles@163.com

Received: Feb. 18<sup>th</sup>, 2020; accepted: Mar. 4<sup>th</sup>, 2020; published: Mar. 11<sup>th</sup>, 2020

---

## Abstract

Aiming at the problems of traditional traffic signal recognition methods, such as small target detection with low accuracy and unrecognizability under occlusion or reflection, this paper proposes convolutional neural networks. A single-stage neural network model is used as a classifier, and the color image captured by a single camera is used as the input of neural network after preprocessing. The features are automatically advanced in advance, and the recognition result is obtained by means of feature fusion. The feasibility of the method was verified by experiments.

## Keywords

Deep Learning, Convolutional Neural Network, Traffic Light Recognition, Feature Fusion

---

# 基于单阶神经网络与特征融合方法的交通信号灯识别

王增荣

太原理工大学, 机械与运载工程学院, 山西 太原  
Email: wzrfiles@163.com

收稿日期: 2020年2月18日; 录用日期: 2020年3月4日; 发布日期: 2020年3月11日

---

## 摘要

针对传统交通信号灯识别方法存在小目标检测准确率低、遮挡或反射情况下无法识别等问题, 本文提出

一种基于卷积神经网络的交通信号灯识别方法。采用单阶神经网络模型作为分类器，将单摄像头拍摄的彩色图像经过预处理后作为神经网络的输入，自动提取特征，通过特征融合的方法得出识别结果。通过实验验证了该方法的可行性。

## 关键词

深度学习，卷积神经网络，交通信号灯识别，特征融合

Copyright © 2020 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 概述

自动驾驶中对环境的感知是个技术难关，并且与道路、设施以及其它交通参与者的协调存在困难[1]。其中需要准确快速地识别交通信号灯，但实际的交通环境信息复杂多变以及天气等因素的影响使得对交通信号灯的识别难度大大增加，这给行车安全造成了安全隐患。孙学聪用形状和色彩来识别交通信号灯但模型简单，不能在复杂情况下进行准确识别[2]；田谨等人采用亮度分割和几何形态滤波对交通信号灯进行定位，但其对于信号灯区域面积过小的图像检测错误率较高且算法耗时[3]；潘卫国等人用 FasterRCNN 框架实现了对交通信号灯的识别，但其识别速率达不到实时性要求[4]。本文使用了基于 RCNN 的基础上发展出来的 SSD 算法在交通信号灯识别方面的应用。提出了基于 SSD 的检测方法，在其基础上添加了特征融合的方法，使得 SSD 对小物体的检测能力提高，同时在不大幅增加计算量的情况下使之可以实现实时检测识别，保证了实时性与准确性。

## 2. PyTorch 深度学习框架

PyTorch 用计算图来对神经网络具体实现形式的描述，包括每一个数据 Tensor 及 Tensor 直接的函数 function。计算图构建方法如图 1 所示。

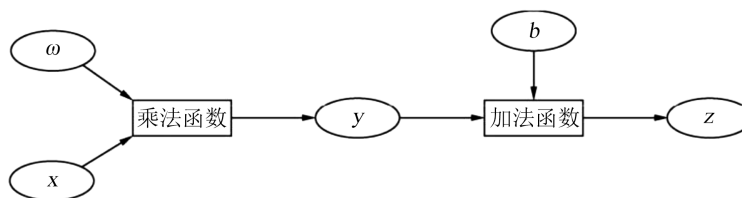


Figure 1. Calculation chart process

图 1. 计算图过程

在图 1 中， $x$ 、 $\omega$  和  $b$  都是用户自己创建的，因此都为叶节点， $\omega$ 、 $x$  首先经过乘法算子产生中间节点  $y$ ，然后与  $b$  经过加法算法产生最终输出  $z$ ，并作为根节点。而且 PyTorch 是自动求导，其梯度计算遵循链式求导法则，简化了编程的环节，更注重网络框架结构。与其他框架不同的是 PyTorch 的计算图是动态的，动态图是指程序运行时，每次前向传播时从头开始构建计算图，这样不同前向传播有不同的计算图，不需要事先把所有的图都构建出来，并且可以很方便的查看计算过程中的中间变量。在进行网络初始化，节点参数的更新，权重偏移量的更新采用反向传播的方法。每次选取小批量数据进行迭代计算，

不仅减少了计算资源的浪费更加快了迭代的速度。反向传播优化流程图见图 2。

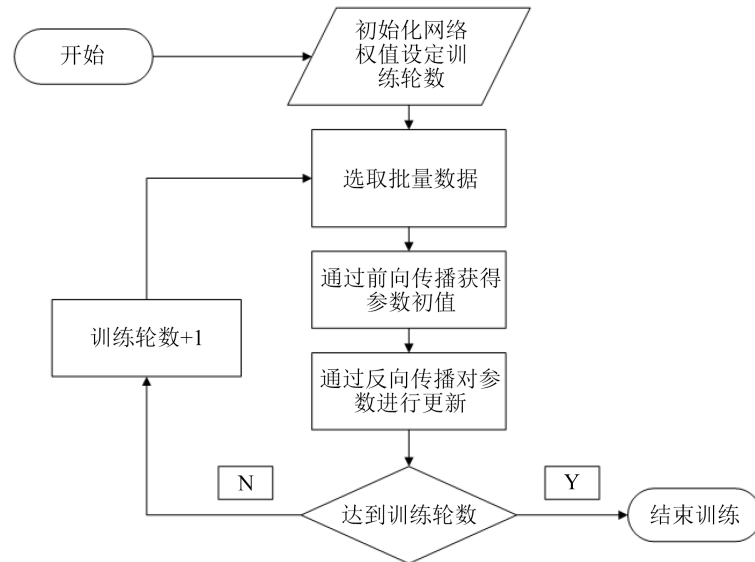


Figure 2. Neural network back propagation optimization process  
图 2. 神经网络反向传播优化过程

### 3. 交通信号灯的识别方法

本文提出的交通信号灯识别方法主要对数据图像的预处理，数据的增强，网络结构的搭建和反卷积模块的搭建。

#### 3.1. 图像预处理

为了提升数据集的多样性，满足训练的实际要求以及对小物体的识别度。总体上包括光学变换与几何变换两个过程[5]见图 3。光学变换是对亮度和对比度进行调整，可以改变像素值的大小但不对尺寸进行更改；几何变换是对图像进行扩展，裁剪和镜像的操作，先进行尺度上的变化在进行去均值的操作。所有操作过程都是随机的，目的是尽可能的保证数据的丰富性。

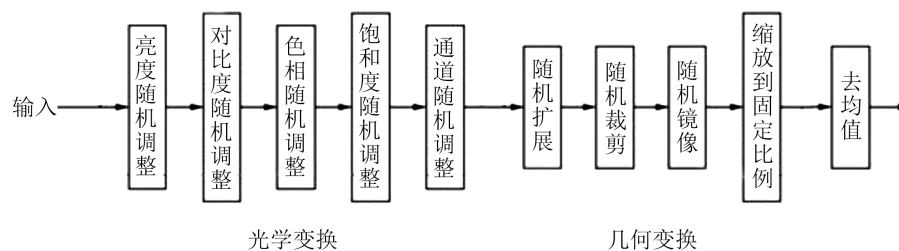


Figure 3. Data augmentation  
图 3. 数据增强

##### 3.1.1. 光学变换

首先是亮度的调节，以 0.5 概率进行对图中每一个点加一个实数，实数范围取 $[-32,32]$ 。然后是对比度，色相与饱和度随机调整，色相是随机加一个数而对比度与饱和度是随机的乘一个数。光学变换最后一项是增加光照的噪声，具体做法是随机交换 RGB 三个通道的值。

### 3.1.2. 几何变换

在几何变换中首先是对原图像进行随机扩展，扩展比例可以取[1,4]之间。

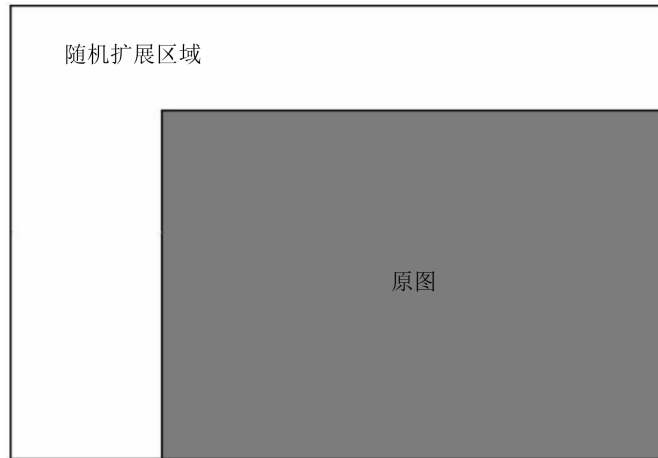


Figure 4. Random expansion method  
图 4. 随机扩展方法

随机扩展之后进行随机裁剪见图 4，随机裁剪的好处是可以滤掉不包含明显真实物体的图像，同时丰富看训练集尤其对于物体遮挡的情况。而镜像基本是对图像进行左右翻转。最后一步为了使输入为特定大小的图像，对图像进行固定缩放。最后一步是对图像进行去均值的处理。部分代码如下：

```
class SubtractMeans(object):  
    def __init__(self,mean):  
        self.mean = np.array(mean, dtype=np.float32)  
    def __call__(self, image, boxes=None,labels=None)  
        image = image.astype(np.float32)  
        image -= self.mean  
        return image.astype(np.float32),boxes,labels
```

### 3.2. 网络架构

#### 总框架搭建

首先是基于 VGGNet 为基础，使用 VGGNet 网络结构对图像进行特征的提取。总框架结构图见图 5。

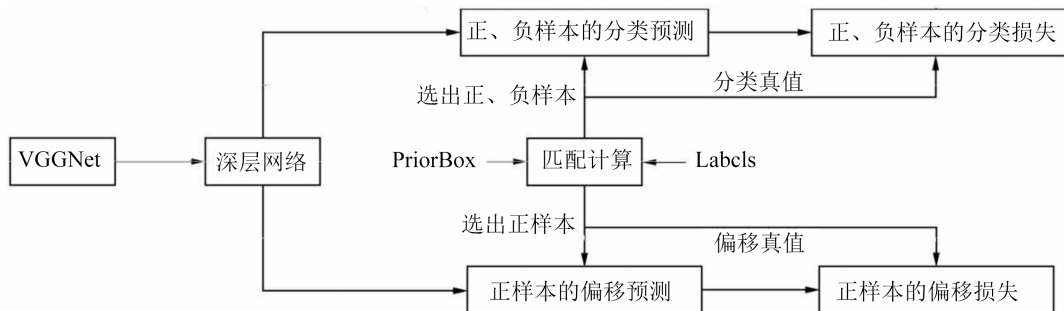


Figure 5. Frame structure diagram  
图 5. 框架结构图

VGGNet 采用了五组卷积层和三个全连接层，最后使用 Softmax 做分类。每次经过池化层后特征图的尺寸减小一倍，而通道数增加一倍。

本文构建的 VGGNet 卷积神经网络主要由五种结构组成：

1) 输入层。输入层是整个神经网络的输入，在处理图像的卷积神经网络中，它一般代表了一张图片的像素矩阵[6]。在输入层可以对较小的数据集进行数据增强，包括光学变换与几何变换等，极大地扩充了数据集的丰富性，从而有效的提升了模型的精度。

2) 卷积层。卷积层是整个网络最重要的部分，对图像进行特征提取，卷积核的大小通常有  $3 \times 3$  和  $5 \times 5$ ，卷积核越大运算量越大但精度会有所提高。VGG16 相比 AlexNet 中使用的卷积核基本都是  $3 \times 3$ ，这种结构优点在于首先从感受野来看，两个  $3 \times 3$  的卷积核的感受野与一个  $5 \times 5$  卷积核的感受野相同但参数量更少。更为重要的是两个  $3 \times 3$  卷积核可以使用两个激活函数，使其非线性能力大大增强，可大大提高网络的学习能力。

3) 池化层。神经网络池化可分为最大池化和均值池化，其首要作用是下采样，降维去除冗余信息，对特征进行压缩，简化网络的复杂度。其次可以扩大感受野以及实现非线性。

4) 全连接层。在经过之前卷积层池化层的数据处理后，将所有参数经过三个全连接层展开成为 1 维张量，便于 Softmax 函数进行分类[7]。

5) Softmax 层。该层主要用于分类问题。通过 softmax 层，可以得到当前样例属于不同种类的概率。

与 Faster RCNN 的 Anchor 类似 SSD 是采用 PriorBox 来进行区域生成，以固定大小宽高的 PriorBox 来进行对感兴趣的区域搜索，而本文交通信号灯的识别 PriorBox 可以集中在图片的中上区域。PriorBox 图见图 6。为了最优的选择尺度和高宽比，采用一种平铺的方法，对每个默认边框尺度进行计算：

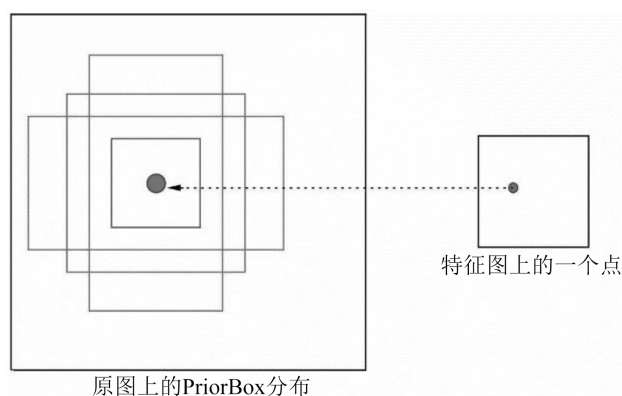


Figure 6. PriorBox diagram

图 6. PriorBox 图

$$S_k = S_{\min} + \frac{S_{\max} - S_{\min}}{5} (k-1) \quad k \in [1, 6] \quad (1)$$

其中  $k$  的取值为 1, 2, 3, 4, 5, 6 分别与 SSD 中第 4, 7, 8, 9, 10, 11 个卷积层。为了得到正确的边框预测值，还需进行匹配与损失计算，保证每一个框都有一定的 Recall 率。将 IoU 值大于 0.7 的视为正样本；IoU 值小于 0.3 的视为负样本，为了防止过拟合将正负样本比例维持在 1:3 附近。

$$\text{IoU}(A, M) = \frac{A \cap M}{A \cup M} \quad (2)$$

偏移值的计算根据下列公式：

$$t_x = (x - x_a) / w_a \tag{3}$$

$$t_y = \frac{y - y_a}{h_a} \tag{4}$$

$$t_w = \log\left(\frac{w}{w_a}\right) \tag{5}$$

$$t_h = \log\left(\frac{h}{h_a}\right) \tag{6}$$

**定位损失和 loss 值的定义**

本文使用了 smooth 函数作为定位损失函数且只对正样本进行计算。smooth 函数表达式如下：

$$smooth_{L1}(x) = \begin{cases} 0.5x^2, & |x| < 1 \\ |x| - 0.5, & |x| \geq 1 \end{cases} \tag{7}$$

其总体目标损失函数是位置损失与置信度损失的加权和：

$$L(x, c, l, g) = \frac{1}{N} (L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \tag{8}$$

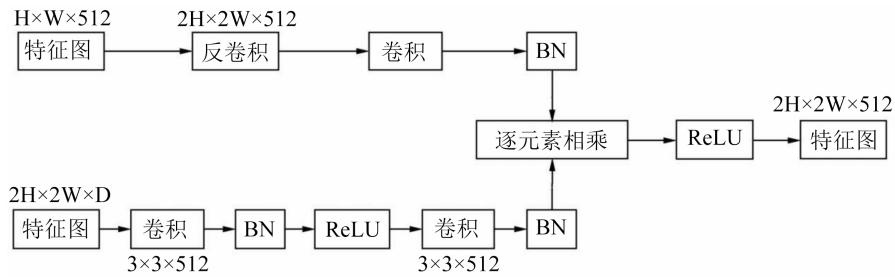
$$L_{Loc}(x, l, g) = \sum_{i \in Pos} \sum_{m \in \{cx, cy, w, h\}} x_{ij}^k smooth_{L1}(I_i^m - g_j^m) \tag{9}$$

$$L_{conf}(x, c) = -\sum_{i \in Pos} x_{ij}^p \log(c_i^p) - \sum_{i \in Neg} \log(c_i^0) \tag{10}$$

$$c_i^p = \frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)} \tag{11}$$

**SSD 特征融合**

将深层与浅层的每一个元素在对应位置相乘，在上述说明中 k 取六个值，将这六个值之后的特征图保留，进行融合处理，再将融合结果传递给后续分类与回归网络。



**Figure 7.** DSSD network structure diagram  
**图 7.** DSSD 网络结构图

下分支为 VGGNet 网络结构图，上分支是一个反卷积模块，深层特征图  $H \times W \times 512$  经过反卷积使之大小与浅层特征图  $2H \times 2W \times D$  一致。两者进行逐元素相乘最后经过 ReLU 模块得到最终所融合的特征图。见图 7。

**3.3. 模型训练与预测结果**

模型优化方法采用 SGD 随机梯度下降算法，又称最速下降算法，是在无约束条件下计算连续可微函

数极小值的基本方法。最后经过 softmax 函数进行伪概率分类，其表达式如下：

$$\text{softmax}(x_1, x_2, \dots, x_n) = \frac{1}{\sum_{i=1}^n \exp(x_i)} (\exp(x_i))_{n \times 1} \quad (12)$$

通过模型的搭建之后加载 Nxeer 红绿灯数据集使之模型准确率达到 86%。但在对雨天夜晚是由于环境因素干扰大，无法达到准确的预测和分类。在此可以添加辅助设备如可转动摄像头，进行角度变动使提高预测结果。

#### 4. 结语

本文构建了 SSD 的卷积神经网络图像识别方法，尝试了由 SSD 基础上加入特征融合的方式对交通信号灯进行分类预测，由输入层、卷积层、池化层、全连接层构成，准确率还有提升的空间，为了提高准确率可以采用随机森林的思想训练多种模型，然后取其均值或者对数据集进行扩充。其次不同数据集在同种模型下表现不同，数据量大的数据集可以使模型有着较充分的学习，使之提高准确率。本文尝试了将 SSD 与特征融合的方法应用于交通信号灯的识别上，之后可以进行网络压缩来进一步缩小网络模型的大小，使之更方便运用于移动端。

#### 参考文献

- [1] 和福建, 等. 国内外智能网联车汽车公共道路测试现状和发展趋势研究[J]. 中国汽车, 2018(12): 28-30.
- [2] 孙学聪, 邓煜. 一种简单的基于图像处理交通信号灯识别算法[J]. 汽车实用技术, 2019(1): 44-45.
- [3] 田谨, 应捷. 基于 Gabor 滤波和稀疏表示的信号灯类型识别[J]. 电子科技, 2018(9): 50-52.
- [4] 潘卫国, 等. 基于 Faster-RCNN 的交通信号灯检测与识别[J]. 传感器与微系统, 2019(38): 147-149.
- [5] 贾瑞明, 刘立强. 基于卷积神经网络的交通灯识别[J]. 北方工业大学学报, 2018(5): 30-38.
- [6] 吴丽华. 谷歌 Tensorflow 框架在自动驾驶汽车车辆识别功能中的应用[J]. 智能网联汽车, 2018(20): 38-40.
- [7] 李升波, 等. 深度神经网络的关键技术及其在自动驾驶领域的应用[J]. 汽车安全与节能学报, 2019(2): 119-145.