# 基于 FRCNN 以及合成数据的车标 LOGO 识别

蒋子航 姚沛恩

**摘要:**

本文在 Keras 框架下基于 RetinaNet 构造的基本识别网络，经过多层的特征学习，由神经网络直接完成汽车标志的定位回归以及识别分类。训练集共计 6000 张左右图片，包括比赛官方提供的 1132 张标注数据的图片以及 5000 张合成的图片。合成图片由 5000 张左右不含车标的背景图片与共 30 类的车标 LOGO 图片由算法合成，合成过程中采用了各种变换以求提高识别度。

复赛采用 50000 张验证集，采用常用的 mAP（mean average precision）进行评价，达到 0.35 的分类加定位准确率。并且识别速度为 300ms 每张（2G gpu）理论上在更多的计算资源条件下还有很大的提升空间。

**关键词:**

数据合成；定位；识别；FRCNN

## 0. 引言

随着互联网的发展，相比于文本数据，图像和视频数据的占比越来越高。如何处理图片数据，对图像的分析识别也变得越来越重要。在企业服务领域，广告监测和舆情监测是企业重点关注的方向，而识别图像中的品牌 LOGO 则是对图像内容做监测的基础。

最近几年，深度学习在物体识别领域取得了巨大的进步，在网络上有大量公开的数据集和分类比赛，例如 PASCAL VOC（20 类）、COCO（80 类）、ImageNet（200 类）等。同时有大量开源的相关算法和训练成熟的网络。目前，深度学习算法的主要缺点是对标注数据的需求量很大。而品牌 LOGO 的数量又比较多，超过千万级别。全部依赖对真实照片人工标注 LOGO 的位置，成本过高，不现实。况且，相比于通用物体，获取大量具有 LOGO 的真实场景图像也需要大量的人力成本。

从另一方面考虑，LOGO 有自身的特点，相比于人脸识别、车辆识别等只能用通用的特征来判断，每一类 LOGO 是在仿射变换等价的意义下完全一致的图像。而且在场景文字识别领域，人工合成场景文字的数据是比较普遍的增加数据集的方法。在 LOGO 识别中，能否通过合成数据，对标注数据进行扩充，从而降低标注成本，提高 LOGO 识别准确率，成为一个值得研究的课题。

赛题中，限定 LOGO 为汽车行业品牌，目标是在已有少量标注数据的情况下，通过自身合成含有 LOGO 的图片，提升 LOGO 识别的准确率。题目提供了 30 个 LOGO，每个 LOGO 提供少量标注数据，同时，提供了 5000 张背景图，用于与 LOGO 合成。

本题目的难点在于：

1. 直接合成的图片由于背景多样性、嵌入效果的限制，难以对于测试集合有很有效的拟合，需要针对上述问题专门设计算法进行有效优化来提高合成数据的质量。
2. 由于神经网络中间步骤的过程不可知，难以直接判断合成图片是否适合网络的训练，而且不同品牌的 LOGO 大小、颜色、嵌入图像的效果和形式等等具有多样性，为合成图片算法的改进带来困难。

# 1. 算法

## 1.1. 数据合成算法

### 1.1.1. 数据合成的目标

在 LOGO 识别中，由于标注数据的缺乏，我们希望通过合成数据，对标注数据进行扩充，从而降低标注成本，提高 LOGO 识别准确率。这要求我们合成的数据有训练的价值，但由于网络类似“黑箱”的特性，我们只能使合成的 LOGO 图像接近真实的汽车 LOGO 情况。一方面需要考虑 LOGO 在图像中的占比，另一方面需要保证变换后 LOGO 的稳定性，也就是仍然能够被合理地识别。

### 1.1.2. 尝试过的算法

首先的想法就是单纯的把 LOGO 的图像从每个图片中截取出来，然后做一些放缩，水平、竖直翻转等仿射变换后，粘贴到背景图像上。这样的好处是简单易于实现，生成 5000 张合成数据只需要几分钟。但是这样做的缺点也是明显的，由于车标 LOGO 本身的特殊性，有的图片中 LOGO 图样相对较小，而背景图片相对较大，导致粘贴后 LOGO 变得人眼都难以识别，这明显是不合理的。一个解决方案是人工筛选出不合适的 LOGO 图样，但是这耗时耗力，也不符合本次实验的初衷。经过初步尝试，我们只达到了 10%的识别准确率，甚至要低于仅使用不合成数据训练的模型，所以我们放弃了这一方案。

### 1.1.3. 最终确定的方法

在上一小节中提到，直接将汽车 LOGO 部分截取粘贴到背景图片的合成数据比较生硬，不能有效地提高网络拟合程度，所以我们希望找出包含 LOGO 的物体，在本题中，我们需要将 LOGO 周围的汽车部分一起截取，再将其嵌入背景图，这样既将背景作为部分负样本进行训练，也保留了 LOGO 以及其附近的一些图像特征。可以达到更好的训练效果。但是这也埋下了过拟合的隐患，这点我们会通过其他变化来弥补。

本模型中我们设计了一种新颖的算法通过 LOGO 的位置及其周围的图像特征计算出车的大致范围。为了找出 LOGO 附近车的位置，我们首先对全图进行 k-means 方法 3 聚类，得到了只有三色的图片，这时我们只要取定上(或下左右)一个方向，通过计算从 LOGO 位置出发，求出颜色的梯度变化，而后取一个恰当的自适应阈值，就可以找到大致的颜色第一次剧烈变化的位置，也就是我们需要的车和背景的分界位置。

四个方向都如此扩充，就得到了我们需要的车的位置，而后只需要适当的仿射变换，就可以得到各种各样的 LOGO 和车的图像了。并且值得一提的是，这样的插入使得训练数据也不显得突兀。

但是这一方法带来了一个问题，即每张图处理的时间较长，在算法刚成型是，生成一张训练数据就需要接近 2 秒，这在大量训练数据的合成里是难以容忍的。但是我们并没有很好的办法提高运算效率，所以在尽可能的情况下改进了算法，在不降低效果的基础上，每张图片平均耗时 500ms。尽管如此，合成 6000 张训练数据也耗费了大量时间。

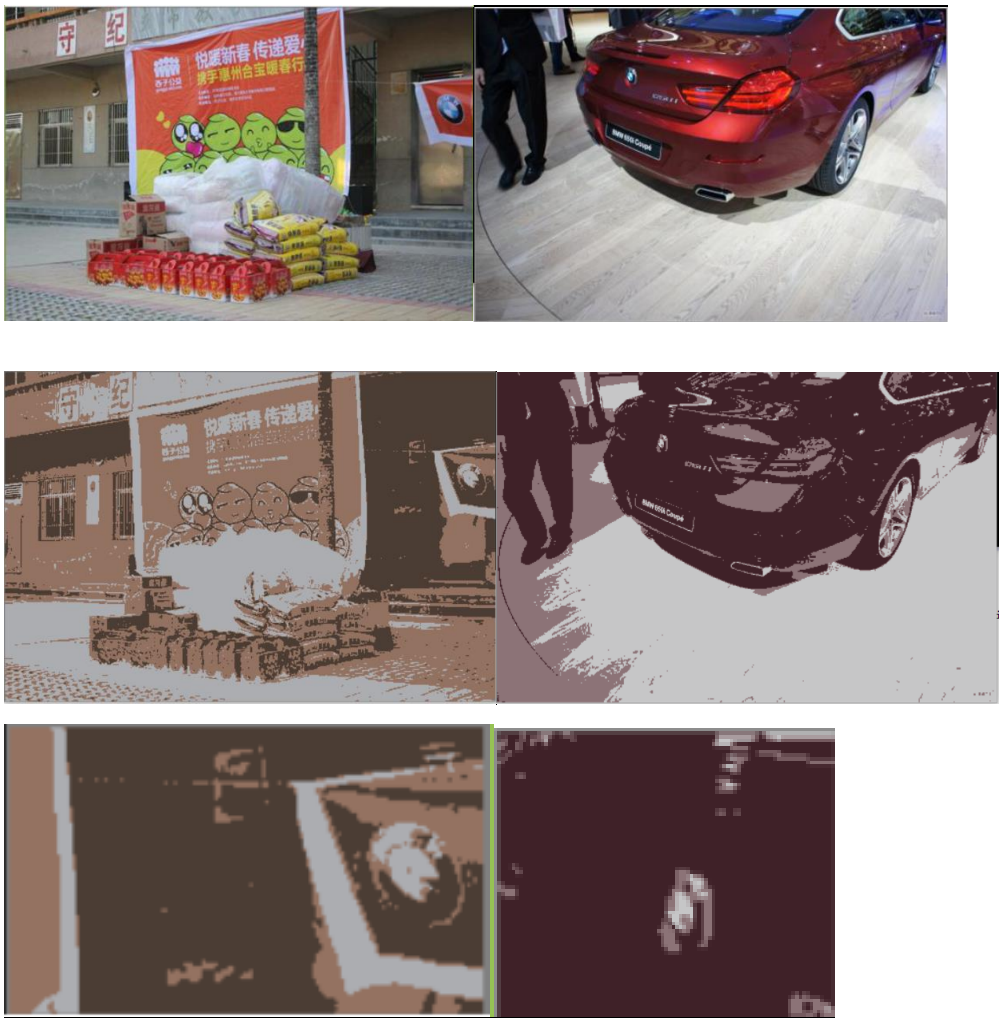


图 1 原始图片、k-means 处理后图片、算法截取后图片

### 1.1.4. 其他方法

复赛结束后一些排名靠前的参赛队分享了他们的思路，用 GAN 对抗神经网络生成 LOGO 数据，并取得了较好的效果。但是由于能力和时间有限，我们并没有尝试，并且一般的 GAN 神经网络生成的只是 LOGO 部分的图像，并不能很好的给出 LOGO 周围区域的情况。不适用于一些关注 LOGO 周边图像的神经网络，与我们准备使用的 FRCNN 网络有一些矛盾。

## 1.2. 网络训练算法

### 1.2.1. LOGO 定位算法

由于最开始我们基于的网络是简略版本的 FRCNN，LOGO 分类部分的 loss 始终很高，难以收敛，但是定位 LOGO 部分的网络却收敛的很好，在测试集上表现也比较稳定，所以我们先想到的是额外训练一个 LOGO 分类的模型。由于我们在图像分类算法上曾有过一些研究，而且我们有更多的方法增加 LOGO 图的数量，诸如颜色变换，剪切变换，倾斜等操作，这势必会比直接在和背景数据结合的时候进行变换更加方便。但是相对的，由于数据量有限，基本每个 LOGO 只有 30 张左右的原始数据，即使使用了许多变换方法也难以避免过拟合以及模型泛化能力差的发生。

所以我们还是决定将识别和定位进行糅合，并且更新了网络模型，采用了 Retina Net 作为框架，很好地解决了之前的问题。

### 1.2.2. LOGO 识别算法

我们使用的 baseline 是 RESNET，整个结构是 Fast-RCNN 的框架，代码在附录中。网络结构，训练的参数以及参考的 loss 函数来自参考文献[1]，，训练了 8 个 epoch 后再训练集上的 loss 降到了 0.03423 和 0.03824。前者是 LOGO 定位的 loss function，后者是 LOGO 分类的 loss function。此时测试集上的准确率已经不再增加，为了防止过拟合现象的产生，我们停止了训练。

最终这一算法在单纯的非合成数据下达到了 0.2mAP 左右，而在数据提升后的合成数据训练下，在测试集的表现上升高到了 0.3mAP 左右，可以看出合成数据对识别的提升帮助是显著的。

但是比赛后，据了解，SSD 在不使用任何合成数据的情况下就可以达到 0.6mAP，这是 FRCNN 在小数据量情况下似乎难以企及的。（与我们类似方法的队伍最高用不合成数据的 FRCNN 达到过 0.4mAP）。

## 2. 实验结果

以下是我们的识别结果中较好的部分。我们对一些质量比较高的图片的识别率基本上都在 90%以上，所以说模型训练的方向没有问题，但是对于一些特殊的比较小比较容易混淆的车标，抗干扰性比较差，特别是比赛后期，官方测试集加上了水印之后我们的模型变得更加不稳定了，很容易将水印误识别，这也是需要考虑的一部分。实验结果如下图所示，绿色框内即为识别出的汽车 LOGO。

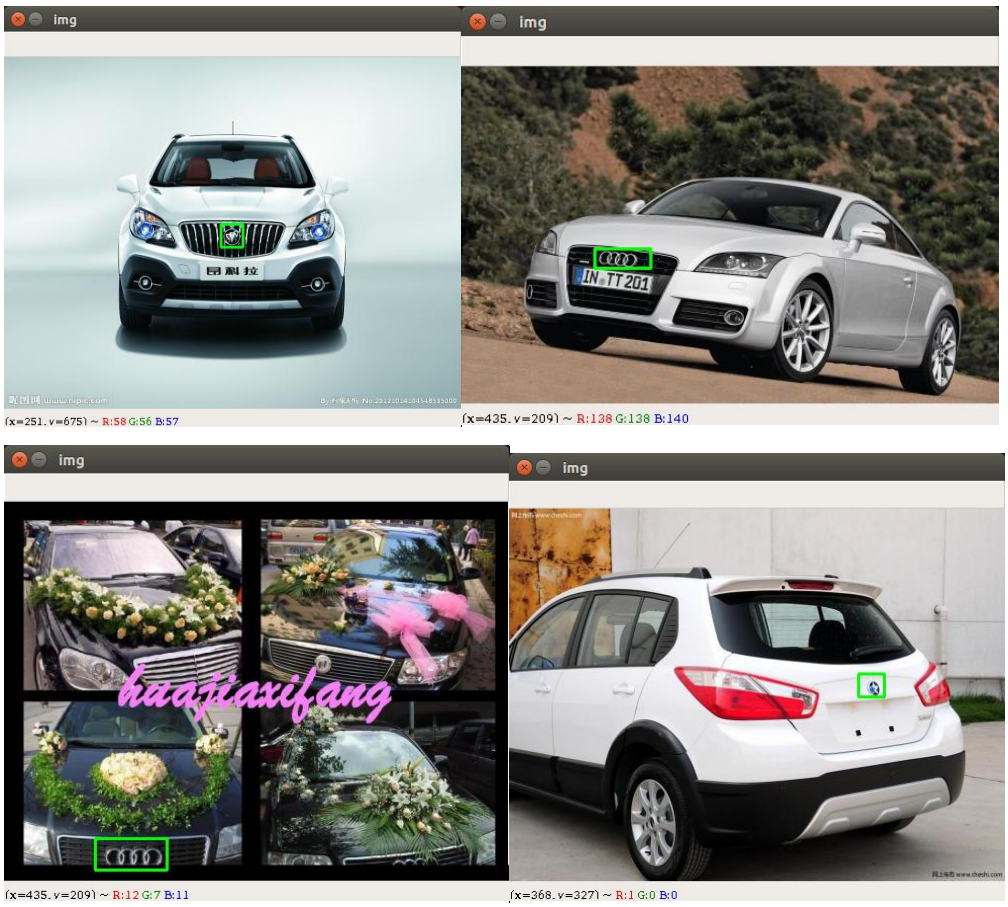
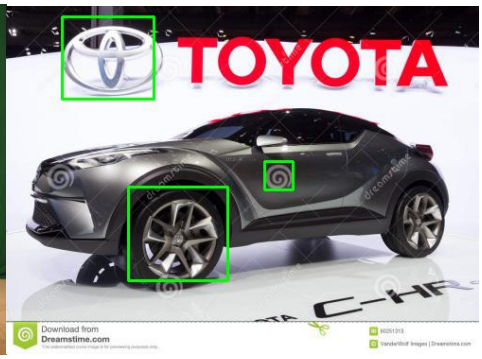



图 2 识别结果



## 3. 总结与不足分析

对官方给出的测试集的特征观察不足是本次实验的结果并不算特别好的一个重要原因，由于我们将所有图片归一化成 500*600 的像素，默认 RGB 的三通道的图片。而官方测试集中有很多图片普遍存在只是灰度图像，以及一张图片是由 3-4 张图片拼接而成的长图片。这也大大影响了我们的识别。单纯从正常的测试集上观察，我们的识别准确率应该至少可以达到 0.5mAP。特别是比较清晰的图片，基本上都能很好地识别。但是由于我们并没有在训练集上做颜色以及透明度上的变换，我们对光线较暗或者很强的图片还是没办法处理，这可能也应该是需要慎重考虑的问题。还有之后的水印，除了我们也添加水印这一类，或者在预处理的时候就去掉水印来抗干扰，否则很难排除干扰。

总的来说，本次实验取得的结果还是令人满意的，我们对神经网络在物体定位及识别上的应用也有了更进一步的了解。

## 4. 组员分工情况

蒋子航：神经网络的搭建与测试，数据的预处理，合成算法的设计，神经网络的训练，结果文件的调试，分析与提交。

姚沛恩：获取部分汽车 LOGO、调试神经网络的训练过程、确定模型的改进策略、设计部分算法。

## 5. 实际结果及工作量

我们在复赛 A 榜中排名第 38，最终 mAP 如下图所示。

| | | | | | |
|---|---|---|---|---|---|
| 36 | | LongGOURN... | 0.3485 | 1 | 2017-11-30 12:50:45 |
| 37 | | 榴莲与牛奶 | 0.34839 | 1 | 2017-11-28 12:20:57 |
| 38 | ▲ 1 | upup | 0.30201 | 3 | 2017-11-30 22:47:35 |
| 39 | | 欧米伽小分队 | 0.26584 | 2 | 2017-11-27 13:56:14 |
| 40 | | NeverMore | 0.23222 | 9 | 2017-12-03 15:50:13 |

## 6. 参考文献


[1] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollar and Facebook AI Research (FAIR) *Focal Loss for Dense Object Detection* arxiv:1708.02002,2017

[2] Wei Liu , Dragomir Anguelov , Dumitru Erhan , Christian Szegedy , Scott Reed , Cheng-Yang Fu , Alexander C. Berg *SSD: Single Shot MultiBox Detector* arxiv: 1512.02325