



2018-7-20

# OCR 深度学习

## 实验室手册



编辑者：徐僖禧，马骋，褚倩云，张培尧，马建奇

单位：复旦大学

# 目录

第 1 章 图像字符生成.....	1
1.1 字符生成定义.....	1
1.2 数据集说明.....	1
1.2.1 数据集意义.....	1
1.2.2 常见数据集介绍.....	1
1.3 字符生成综述.....	2
1.3.1 对抗生成网络 GAN.....	2
1.3.2 VAE 模型.....	7
1.3.3 字符生成模型.....	8
1.4 主流算法介绍.....	16
1.4.1 zi2zi.....	16
1.4.2 SA-VAE .....	20
1.5 实验室现有算法介绍.....	24
1.6 zi2zi 中文字符生成实验.....	24
1.6.1 实验过程.....	24
1.6.2 实验结果.....	25
1.6.3 loss 图示及解释 .....	28
1.7 本章总结.....	30
参考文献.....	31
第 2 章 图像字符检测.....	32
2.1 字符检测定义.....	32
2.2 数据集说明.....	32
2.2.1 数据集意义.....	32
2.2.2 常见数据集介绍.....	32
2.3 字符检测综述.....	35
2.4 主流算法介绍.....	46
2.4.1 CTPN .....	46
2.4.2 DMPnet .....	47
2.4.3 Seglink .....	48
2.4.4 Textboxes & Textboxes++ .....	49
2.4.5 EAST.....	49
2.4.6 FOTS.....	50
2.4.7 IncepText.....	51
2.5 实验室现有算法介绍.....	52
2.5.1 算法介绍.....	52
2.5.2 模型结构.....	52
2.6 RRPN 字符检测实验.....	54
2.6.1 数据集介绍.....	54
2.6.2 评估标准.....	55
2.6.3 开发环境.....	55
2.6.4 实验细节.....	55
2.6.5 两次提交结果成绩.....	57

2.7 总结.....	59
参考文献.....	60
第3章 图像字符识别.....	61
3.1 字符识别定义.....	61
3.2 数据集说明.....	62
3.2.1 数据集意义.....	62
3.2.2 常见数据集介绍.....	62
3.3 字符识别综述.....	68
3.3.1 自然环境下采集的字符图像识别.....	68
3.3.2 手写体汉字识别.....	70
3.3.3 印刷体识别.....	71
3.4 主流算法介绍.....	72
3.4.1 TSCD model .....	72
3.4.2 AON model .....	73
3.4.3 DTRN model.....	74
3.4.4 R <sup>2</sup> AM model .....	76
3.4.5 DTRN model.....	79
3.4.6 CRF-CNN joint model.....	80
3.4.7 Embedded Attributes model.....	82
3.4.8 Text-deeplab model.....	85
3.5 实验室现有算法介绍.....	86
3.5.1 算法介绍.....	86
3.5.2 模型结构.....	86
3.6 CRNN 字符识别实验 .....	88
3.6.1 数据生成.....	88
3.6.2 模型训练.....	107
3.6.3 错误结果分析.....	114
3.7 本章总结.....	118
参考文献.....	119
附录A ICPR-MTWI2018 挑战赛一：网络图像的文本识别竞赛细则.....	122
附录B ICPR-MTWI2018 挑战赛二：网络图像的文本检测竞赛细则.....	125

---

# 第1章 图像字符生成

## 1.1 字符生成定义

字符生成是风格迁移任务在字符上的具体应用，简单来说，就是通过神经网络的方式，让字符在不同的字体风格间迁移，快速生成风格相同的字体。这一领域，尤其是中文字体生成有很大的市场需求。

在我们的生活实践中，随着科技的进步和人们精神文明的发展需要，个性化多样化的需要也成为了急需科技解决和面临的问题。在生活生产中，我们无时无刻不在与字体打交道，比如在街边看到的广告牌，上网浏览的网页，淘宝的广告字体，工业 logo 等。文字是平面设计中很突出的视觉元素，艺术家们投入大量的时间来设计和创造美观的中/英文字体。

一整套的字体设计通常包含两个方面：字型和字量。字型是字体的核心组成成分，一般的创新点来自字体设计师生活点点滴滴的灵感，字量是指字体讲求的均衡的美感，整体风格的统一，要保证整套字体里面所有文字的浑然天成。这需要耗费大量的劳动力

相比于英文的 26 个英文字母，中文字体的生成更加有难度。首先，中文汉字的数量庞大。GB2312 国标码中选入了常用的中文简体的中文字数就有 6763 个，分为两级，一级字库中有 3755 个常用汉字，二级字库中有 3008 个次常用汉字。中文简繁的中文字数有 9169 个，用于出版刊物的大字符集（包含很多生僻字）27533 个，包含绝大部分中文字体的超大的字符集的中文字数达 7 万字以上。对于中文汉字，生产生活中使用较多的是 GBK 标准（中国政府规定的字符集）的字体，这样的话，字体设计团队要完成对大部分汉字（26000 多字）的风格转换，需要很大的时间成本和人工成本。其次，中华文化中的汉字又千变万化，有具有明显风格属性的书法体和手写体，每个人书写的风格多样。

所以我们急需一种可以降低成本的方法来进行字体生成的工作。深度学习的快速发展带来了机会，设计师可以设计一小部分字符之后，通过神经网络强大的模仿学习能力，生成剩余字符，完成全部字体的风格转换，生成出我们预期的字体。因此深度生成模型得到了广泛的应用。近些年比较流行的主要有：生成对抗网络（Generative Adversarial Network,简称 GAN），变分自编码器（Variational Auto-Encoder,简称 VAE），及其一些混合模型。无论是在学术界还是在工业界，由于其深厚的理论基础，和良好的实验表现，这两个模型被大量的应用和创新发展。

## 1.2 数据集说明

### 1.2.1 数据集意义

对于字符生成任务，各种类型，各种风格的字符数据集是必需的。由于训练模型的复杂度很高，因此我们常常需要收集大规模的数据集进行模型训练。获取大规模数据集有两种方式：1.采用公开的数据集 2.自己创造制作数据集。目前，借助互联网，我们收集到了各大平台可供免费使用，且数据良好的数据集，这给我们训练模型节省了大量的精力，以下是我们收集到的一些优良的公开数据集。

### 1.2.2 常见数据集介绍

#### （1）手写数字数据集

该数据集，亦称 MNIST，总共包含 60000 个训练样本，和 10000 个测试样本，每个样本包括一个 28\*28 维的二值化图像（每个像素取值于 {0, 1}，和表示图片数字的标签（标签用 one-hot 嵌入表示）。

数据源：<http://yann.lecun.com/exdb/mnist/>

### (2) 手写英文字母数据集:

该数据集收集了 37 万余张大写的手写英文字母图像，所以共有 26 个类别，每个类别由 0-25 中的数字表示，每张图像尺寸为 28\*28，像素值是 0-255 的任意整数。数据集保存在 CSV 文件中，

数据源：<https://www.kaggle.com/sachinpatel21/az-handwritten-alphabets-in-csv-format>

### (3) 中文汉字字库

该字库，包含两部分，一部分是矢量图表示的手写中文字库，称为：CASIS-OLHWDB. 另一部分是位图表示的手写中文字库，称为：CASIS-HWDB。每一个字库可以分为 6 个数据集，三个是独立的中文字库，三个是手写文本库（连续脚本）。不论是矢量图字库还是位图字库，独立的中文字库包含 7356 个字符的大约 390 万个样本，手写文本库包含大约 5090 张共 135 万个字符的样本。而且每一个数据集都划分成了标准的训练和测试子集。样本的数据格式为\*.mpf，需要通过代码编写读取数据。

数据源：<http://www.nlpr.ia.ac.cn/databases/handwriting/Home.html>

## 1.3 字符生成综述

### 1.3.1 对抗生成网络 GAN

对抗生成网络 (GAN) [13] 是近年来兴起的一种深度学习的模型，是无监督学习中十分卓越的方法之一。一般模型由生成模型(Generative Model)和判别模型(Discriminative Model)两部分构成，Generator 和 Discriminator 相互博弈进而产生相当好的输出。

GAN 的工作原理如下：第一代生成模型 G 经过学习，从随机噪声或者潜在变量 (Latent Variable) 中生成逼真的样本，产生出一张图片（或是用户要求的元素），和真实的图片一起送入判别模型 D，检测判别模型 D 判断真伪的能力。如果骗过第一代判别模型 G，则第一代判别模型 D 升级至能识别出第一代生成模型 G 生成出的图片为假的第二代判别模型 D，然后第一代生成模型 G 再生成出能骗过第二代判别模型 D 的第二代生成模型 G，如此迭代，二者同时训练，相互博弈，相互促进。在这个过程中，生成模型 G 生成出质量越来越高的图片，判别模型 D 判断图片真伪的分类能力也越来越高，最后达到纳什均衡。

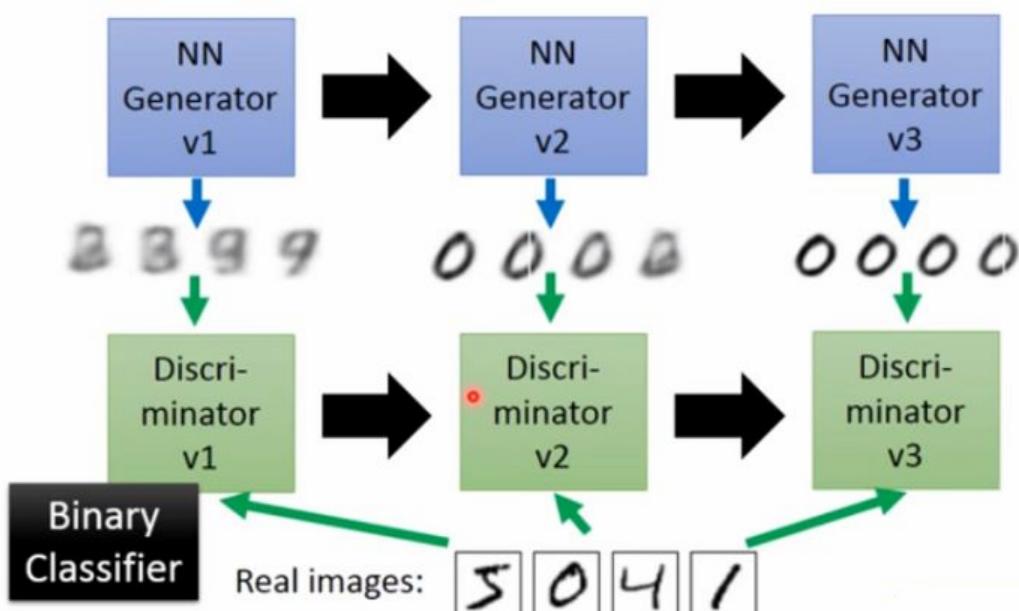


图 1-1 GAN 的工作原理

对抗生成网络(GAN)能够通过学习我们提供的现有的大样本数据，利用生成器和判别器的博弈过程，从而得出我们需求的目标数据的分布。除此之外，科研人员还发现，对抗生成网络(GAN)不受马尔科夫链的限制，而且只需梯度后向传播，就可以得到不错的生成效果。并且对抗生成网络(GAN)是无监督网络，在运行过程中，不需要干涉。无监督学习网络是没有条件限制的生成网络模型，对它的生成数据没有任何限制，因此要想控制这种网络有一种比较直接的方法：在网络中给予条件控制。

无监督学习[12]是一个大的方向，在方向的向前推进中，我们需要观察生成任务的优劣来判别。当然，在实际的研究中，生成式模型（generative models）会有两大类问题[15]。第一，我们在实际应用网络时，如何去做，怎么去安排节点，怎么去达到优化效果，先验知识如何给定，激励函数，优化函数如何选定等问题。第二，现实世界中存在的数据很复杂，我们拟合这些模型时，需要巨大的计算代价。

### (1) 深度卷积对抗生成网络 DCGAN[17]

卷积神经网络（CNN）在有监督学习任务上取得了成功，表现非常出色，而无监督学习领域中 CNN 的应用较少。DCGAN 算法让 CNN（有监督学习）和 GAN（无监督学习）两者相互融合，既可以称 DCGAN 是 CNN 到 GAN 领域的扩展，也可以称 DCGAN 是 GAN 到 CNN 领域的扩展。

GAN 不仅不需要特定的损失函数而且可以在学习过程中很好的抓取特征，但是因为是无监督学习，经常会生成无意义的输出。

DCGAN 对于此但不仅限于此做出了如下贡献：

- 1) 为 CNN 的结构设置很多的条件限制，从而达到稳定的训练效果。
  - 2) 得到特征表示后，进行图像分类，验证对图像特征的表达能力，从而得到较好的结果。
  - 3) 定性分析 GAN 学习的滤波（Filter）。
- DCGAN 在结构上也发生了变化：
- 1) 卷积层替换了 pooling 层，在 D 模块中使用跨距卷积，在 G 模块中用分数跨距卷积。
  - 2) 在 G 和 D 中都采用批量归一化(batch normalization)从而优化初始化效果不好的问题，梯度能够传递到所有层，防止样本收敛过快。
  - 3) 移除了全连接层。虽然全局的池化操作使模型更加稳定，但是降低了梯度收敛的速度。
  - 4) G 所有层（除去输出层）使用激励函数 ReLU, 输出层使用激励函数 Tanh。
  - 5) D 上所有层使用激励函数 LeakyReLU。

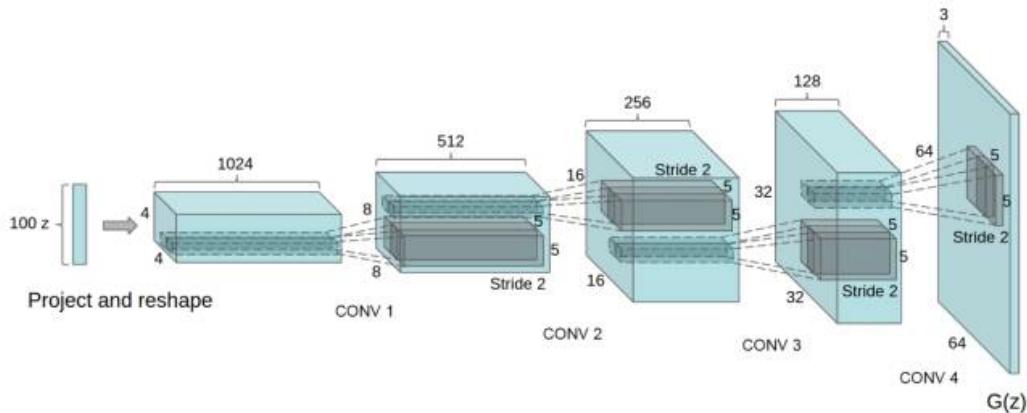


图 1-2 DCGAN 的生成器 G 网络结构

DCGAN 训练细节：

Step 1 预处理环节，将图像缩放到 Tanh 的[-1, 1]

Step 2 最小批训练，批尺寸是 128

Step 3 所有的参数初始化由 (0, 0.02) 的正态分布中随机得到

Step 4 LeakyReLU 的斜率是 0.2

Step 5 GAN 使用动量来加速训练，而 DCGAN 使用调好超参数的 Adam 优化器来加速训练

Step 6 学习率设置为 0.0002

Step 7 将动量参数 Beta 从 0.9 降为 0.5，以防止震荡和不稳定

## (2) 条件对抗生成网络 Conditional GANs[14, 16]

GAN 这种不需要预先建模，直接使用一种分布进行采样的方法，缺点就是太过自由，对于清晰度较高的图片，简单的 GAN 就很难训练。论文 2 首次提出条件对抗生成网络 Conditional GANs 的想法，给对抗生成网络 GAN 加上条件约束，其主要内容是在生成器 G 和判别器 D 的建模中均引入条件变量 y (conditional variable y)，使用额外信息 y 对模型增加条件约束，可以指导数据的生成过程。从模型架构上分析，此时生成器 G 的初始输入不再单单是具有一定维度的随机噪声 Z，而是 Z 和 y。对于判别器 D 来说，输入也不止是图像 x，还有 y。

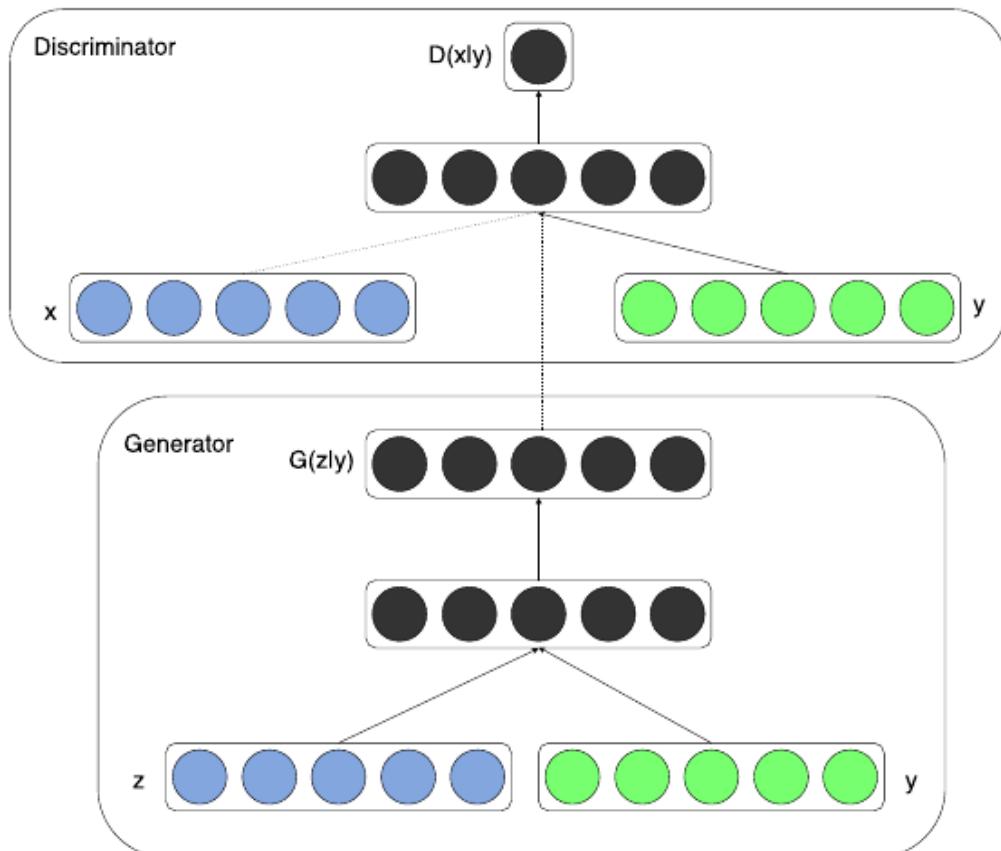


图 1-3 Conditional Adversarial net

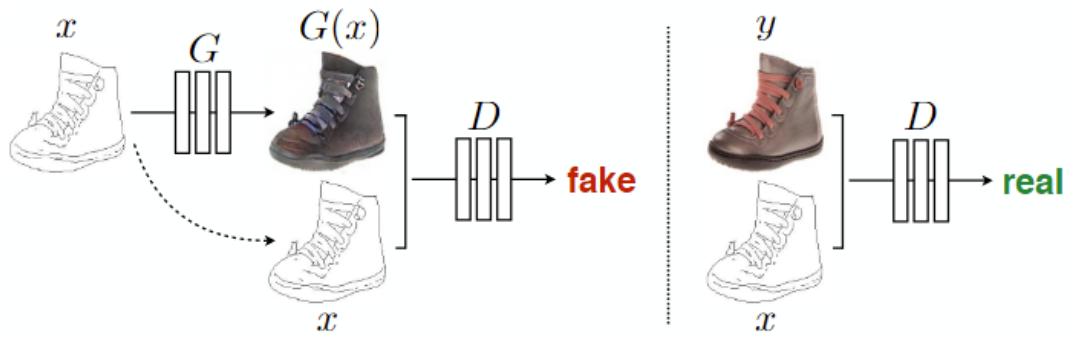


图 1-4 Conditional GANs 完成图片上色任务

当然，在 CGANs 的基础上，如果我们把网络的模型网络改为卷积网络，即成为了我们上述两种模式的结合：Conditional DCGAN。

### (3) Wasserstein GAN(WGAN)[18]

与 DCGAN 不同，WGAN 主要从损失函数的角度对 GAN 做了改进，论文认为 GAN 中交叉熵（JS 散度）不适合衡量生成数据分布和真实数据分布的距离，如果通过优化 JS 散度训练 GAN 会找不到正确的优化目标，所以，WGAN 提出使用 wasserstein 距离作为优化方式训练 GAN。损失函数改进之后的 WGAN 即使在全连接层上也能得到很好的表现结果。

WGAN 对 GAN 的改进主要有：

- 1) 判别器最后一层去掉 sigmoid
- 2) 生成器和判别器的 loss 不取 log
- 3) 对更新后的权重强制截断到一定范围内，比如[-0.01, 0.01]，以满足论文中提到的 lipschitz 连续性条件
- 4) 论文中也推荐使用 SGD, RMS prop 等优化器，不要基于使用动量的优化算法，比如 Adam

WGAN 做的贡献如下：

- 1) WGAN 理论上给出了 GAN 训练不稳定的原因，即交叉熵（JS 散度）不适合衡量具有不相交部分的分布之间的距离，转而使用 Wasserstein 距离去衡量生成数据分布和真实数据分布之间的距离，理论上解决了训练不稳定的问题。
- 2) 解决了模式崩溃的（collapse mode）问题，生成结果多样性更丰富。
- 3) 对 GAN 的训练提供了一个指标，此指标数值越小，表示 GAN 训练的越差，反之越好。

注：Lipschitz 限制是在样本空间中，要求判别器函数  $D(x)$  梯度值不大于一个有限的常数  $K$ ，通过权重值限制的方式保证了权重参数的有界性，间接限制了其梯度信息。

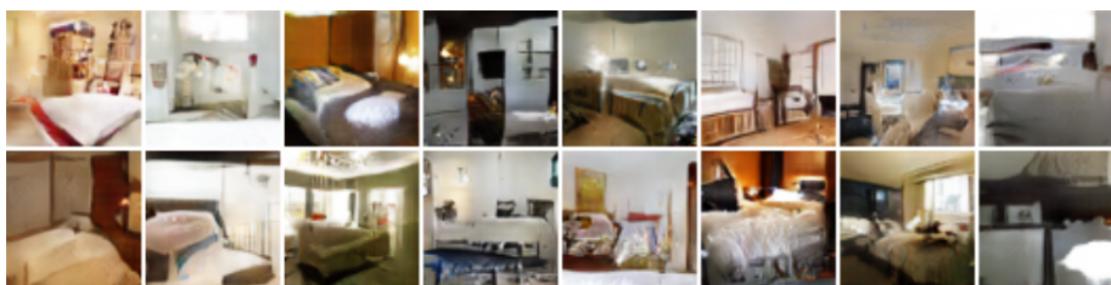


图 1-5 WGAN 生成效果

### (4) 最小二乘 GAN (Least Squares GAN) [19]

最小二乘 GAN (LSGAN) 针对标准的 GAN 生成的图片质量不高以及训练过程不稳定

这两个缺陷进行了改进，主要改进在于将 GAN 的目标函数由交叉熵损失函数改成了最小二乘损失函数，LSGAN 的损失函数要求比之前还要平滑和非饱和梯度。判别器 D 和生成器 G 共同作用生成的数据加入到真实图片中，使 G 同时也学习到类似 D 和 G 共同输出所得到的数据。

而判别器的激励函数没有选择  $\tanh$  而选择了  $\text{sigmoid}$ ，是因为  $\text{Sigmoid}$  函数，在数据输入时反应十分迅速，很容易饱和，所以即使输入的像素点很小， $\text{sigmoid}$  也会迅速忽略该点到判定边界的距离，降低因为距离决策边界的距离过大的惩罚。意味着该函数不会惩罚较远的点，随着输入点变得越来越大，判别器 D 的梯度也就随着该点迅速下降到 0。生成器 G 训练时使用判别器 D 的梯度，如果判别器 D 中的梯度归为 0，生成器 G 就不再能获得有效信息去学习了。

在 L2 loss 中，距离决策边界远的点，会遭到惩罚机制的干预。在模型优化过程中，生成器 G 的学习点要尽可能的避免 L2 loss 过大。

LSGAN 的训练的公式表如下所示：

$$\min_D V_{LSGAN}(D) = \frac{1}{2} E_{x-p_{data}(x)}[(D(x) - b)^2] + \frac{1}{2} E_{z-p_x(z)}[(D(G(z)) - a)^2] \quad (\text{公式 1-1})$$

$$\min_G V_{LSGAN}(G) = \frac{1}{2} E_{z-p_x(z)}[(D(G(z)) - c)^2] \quad (\text{公式 1-2})$$

当公式 1-1 中， $b=1$  时，代表其为现实的真实数据， $a=0$  是表示其为生成器 G 生成的伪造数据， $c=1$  代表生成器 G 成功欺骗了判别器 D。

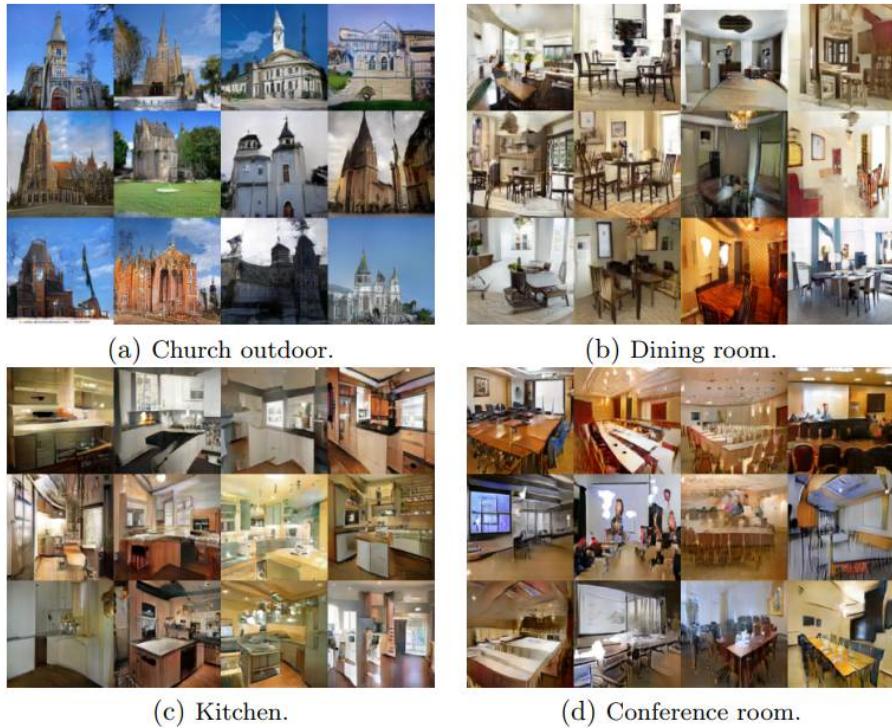


图 1-6 LSGAN 生成效果

**总结：**DCGAN 结合了典型的无监督学习 GAN 和有监督学习的 CNN，从而提升了实验结果的质量。CGANs 添加了有标签的图像的学习，加入了有监督的理念，也改善了原来的结果。WGAN 转用 wassertein 距离来衡量生成数据的分布和真实数据的分布之间的距离，LSGAN 用了最小二乘损失函数代替了 GAN 的损失函数。

### 1.3.2 VAE 模型

变分自编码器（简称 VAE）作为一种在生成方面表现良好的深度生成模型，是由有向概率模型的近似推断进一步演变而得，可以纯粹地使用基于梯度下降的方法进行训练。在概率统计理论方面，它将变分推断的识别模型 $q_\varphi(z|x)$ 和生成模型 $p_\theta(x|z)$ 用神经网络表示，所得产物。在模型方面，它是自编码器的一种变体，不同于普通的自编码器，它将编码器函数 $f(x)$ 和解码器函数 $g(z)$ 推广为概率映射 $p_{encoder}(z|x)$ 和 $p_{decoder}(x|z)$ 。这两个概率即为上述的识别模型和生成模型，损失函数则为变分贝叶斯的优化目标函数——变分下界：

$$L(\theta, \varphi) = -D_{KL}(q_\varphi(z|x)||p_\theta(z)) + \frac{1}{L} \sum_{l=1}^L [\log p_\theta(x|z^{(l)})] \quad (\text{公式 1-3})$$

简易模型如下图：

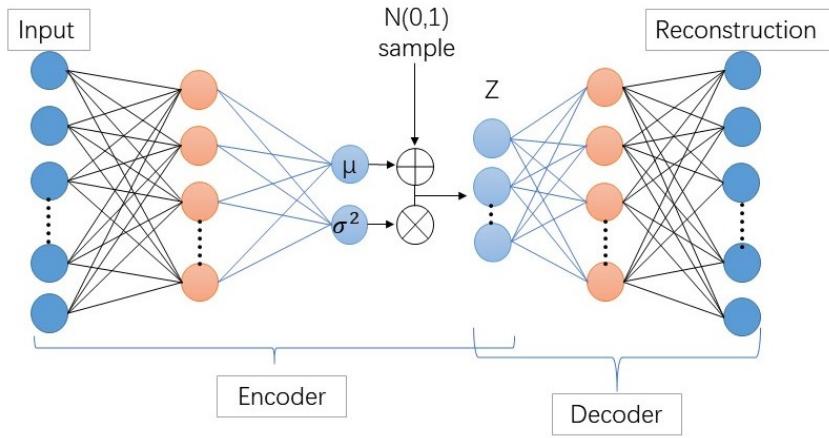


图 1-7 VAE 模型示意图

上图中编码器和解码器部分都用多层感知机（MLP）表示。VAE 的损失函数第一项表示正则化项，第二项表示负重构误差。它较于普通的编码器的重构误差，第一项的正则化项起到了输入对重构的一个监督作用，使得重构效果更好。

而且，需要说明的是，VAE 只提供了一个特殊的模型框架和训练损失函数，所以在编码器和解码器部分，我们可以根据需求或优化性能，搭建不同的网络模型，比如 CNN 模型，RNN 模型等等。这样 VAE 模型的生成效果会更优异。

当然，有关 VAE 模型，国内外的优秀学者已经做了大量的工作并取得了相当卓越的成果：

(1) Kingma D P 是首次提出了 Auto-Encoder Variational Bayes 这个模型[1]，他将变分推断应用到了求解最大后验概率问题（近似潜变量的分布），进而将这种方法推广到自编码器模型，形成了变分自解码器，具有较好的生成效果。

(2) Google DeepMind 实验室又研究出 DRAW 这个模型[2]，他将 RNN 模型和 Selective Attention model 结合起来生成图像，这种生成方法类似于人写字的时候，依笔画逐步生成。

而后又有一些研究者应用到了聚焦机制（Attention Model）[3-5]，对于图像生成，聚焦机制显得尤为重要。而且 DRAW 将 RNN 应用到了字符序列的生成，对于手写字体同样具有良好的效果。

(3) 清华一些研究者近期也做出了中文字符风格化迁移的生成模型[6]，他们提出的模型称为：SA-VAE，它能够将字符的风格和内容拆开，一部分用作中文字符的识别，另一部分用作学习字符的风格，然后输入特定的字符，然后将学习到的风格，融入到输入的字符中去，形成一个新的风格化字符。

(4) Salimans T 将 MCMC(Markov chain monte carlo) 算法[9]和 Variational Inference 做对比并且提出了一些新的观点[7]。

(5) George D 提出了一个图像识别和检测的比较好的模型[8]，他将一个物体的 Appearance 和 Texture 分别进行识别。

(6) Ha D 将 DRAW 这个模型做了拓展，提出 sketch-RNN 模型[11]，可以用来生成一张简笔画的草图。

### 1.3.3 字符生成模型

#### 1.3.3.1 普通字符的生成

对于普通字符的生成，实质上是图像的生成，因此模型对图片有好的生成效果，对应的对于字符图片效果同样优异。

一个比较优异的模型是基于 RNN 和 Attention 机制的 VAE：DRAW 模型。DRAW 模型的特点是在编码器和解码器都应用了 RNN 模型和 Attention 机制。

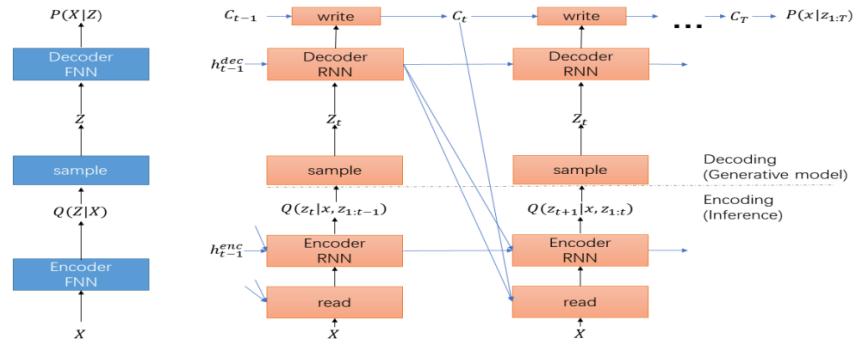


图 1-8 DRAW 模型示意图

Attention 机制保证在每一次训练时聚焦到字符的一个小区域，并绘制这些点的像素值。RNN 模型（具体采用 LSTM 模型）保证顺序访问，所以一个字符的生成，变成一个 sequence-to-sequence 的过程，类似人类作画的过程——连续地从局部到整体的生成，这也是 DRAW 模型的由来。

#### (1) Attention 机制

Attention 机制在自然语言处理 (NLP) 和图像处理领域都有重要的应用并且取得了不错的效果，它主要借鉴于人眼在观察物体时的特点：总是在全局中（比如人眼前的景象）聚焦于某个区域，使得对这个区域的识别更好，然后聚焦到另外一个部分，这样会更好的认识事物。所以该模型使用 Attention 机制，也是基于这个原因，Attention 机制的具体原理参考文献[3]。

对于该模型，原理如下：

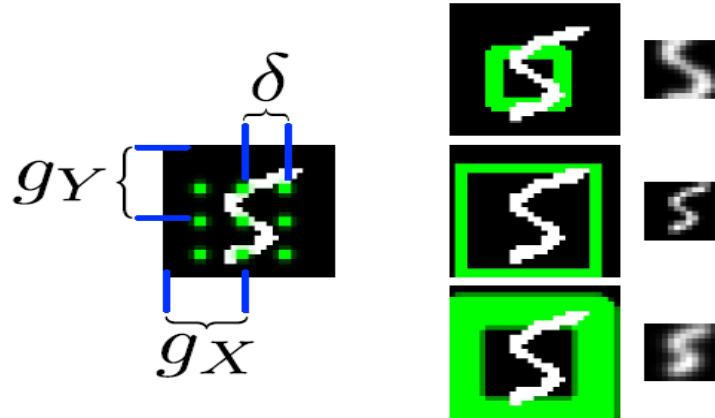


图 1-9 Attention 机制示意图

左图：一个  $3 \times 3$  网格的 filter (过滤器) 作用在图片上，步幅( $\delta$ )和中心位置( $gX, gY$ )如图，右图：从图中提取出一个  $N \times N$  的区域( $N=12$ )，绿色的方框表示的是边界和区域的精度( $\sigma$ )，提取出来的斑块如右边所示，最上面的区域含有一个小 $\delta$ 的和高 $\sigma$ ，呈现出来的是放大的但是模糊的图片的中心区域；中间的区域含有大的 $\delta$ 和低 $\sigma$ ，这是全图一个有效的下采样；下面区域含有大的 $\delta$ 和高 $\sigma$ 。

在该模型中，我们采用二维的高斯滤波器(filter)，这是一个低通滤波器，实质是二维高斯分布的离散形式。可以让提取出来的区域更加平滑，步幅控制着区域的缩放，也就是说，步幅越大，原图像越大的区域聚焦到，但是同时分辨率下降了，这个网络中心( $gX, gY$ )和步幅( $\delta$ )决定着在第  $i$  行，第  $j$  列的 filter 的平均位置  $\mu_X^i, \mu_Y^j$ ：

$$\mu_X^i = gX + (i - \frac{N}{2} - 0.5)\delta \quad (\text{公式 1-4})$$

$$\mu_Y^j = gY + (j - \frac{N}{2} - 0.5)\delta \quad (\text{公式 1-5})$$

对于 Attention 机制，另外还有两个参数需要计算：一个是高斯滤波器的各向同性的方差  $\sigma^2$  和滤波器响应的强度  $\gamma$ 。给定一个  $A \times B$  的输入图像，五个 Attention 的参数在每个时刻动态指定，这五个参数为  $(gX, gY, \sigma^2, \delta, \gamma)$ ，这五个参数作为解码器的输出  $h^{dec}$  得到，输出细节可参考文献[2]。这样就可以计算出水平和垂直高斯滤波（滤波由矩阵表示）， $F_X$  和  $F_Y$ （维数分别为  $N \times A$  和  $N \times B$ ），具体定义如下：

$$F_X[i, a] = \frac{1}{Z_X} \exp\left(-\frac{(a - \mu_X^i)^2}{2\sigma^2}\right) \quad (\text{公式 1-6})$$

$$F_Y[j, b] = \frac{1}{Z_Y} \exp\left(-\frac{(b - \mu_Y^j)^2}{2\sigma^2}\right) \quad (\text{公式 1-7})$$

其中  $(i, j)$  是 Attention 区域的坐标点， $(a, b)$  是输入图像的坐标点，而  $Z_X, Z_Y$  作为正规化的常数确保  $\sum_a F_X[i, a] = 1$  和  $\sum_b F_Y[j, b] = 1$ 。然后就可以得到对输入图像(x)高斯滤波之后的图片： $\text{read}(x) = \gamma[F_X x F_Y]$ ， $\text{read}$  表示将输入图像采用 Attention 机制读入。这样图片的维度就是  $N \times N$  维的。同样地， $\text{write}$  操作用类似的方法得到聚焦后的输出。

回到模型，同样以 MNIST 数据集作为生成样例，看看加入 Attention 机制具体的训练过程：

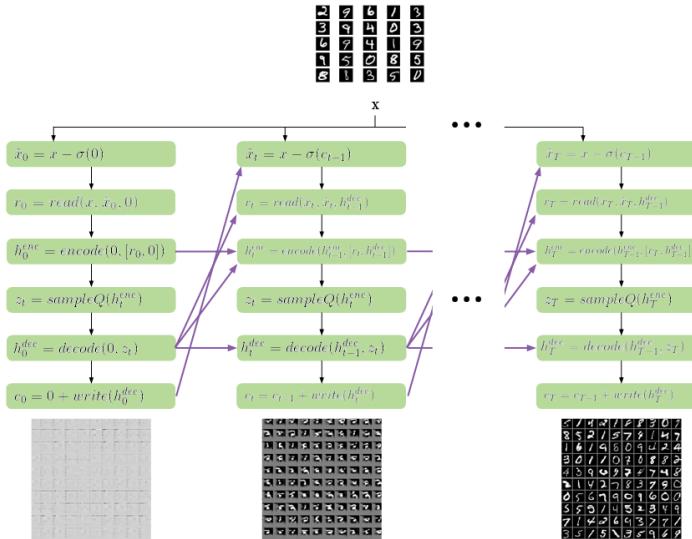


图 1-10 DRAW 模型训练过程

生成结果如图：

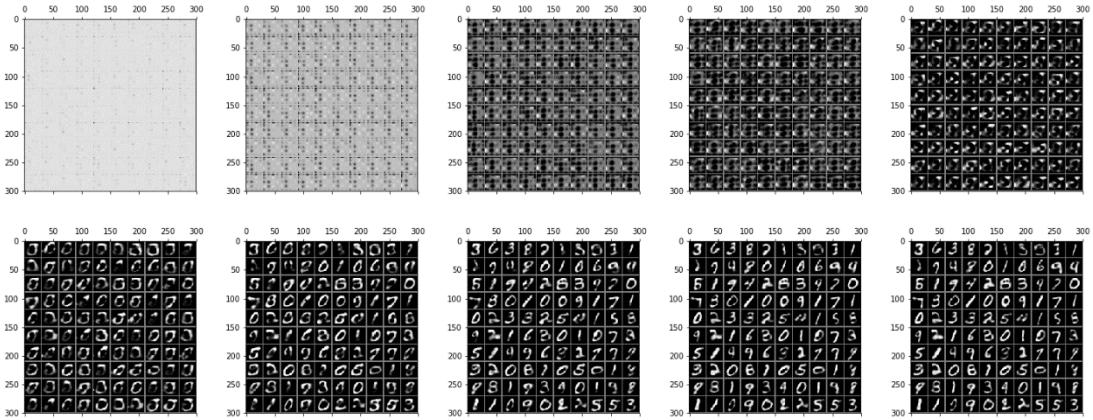


图 1-11 加入 Attention 机制生成结果图

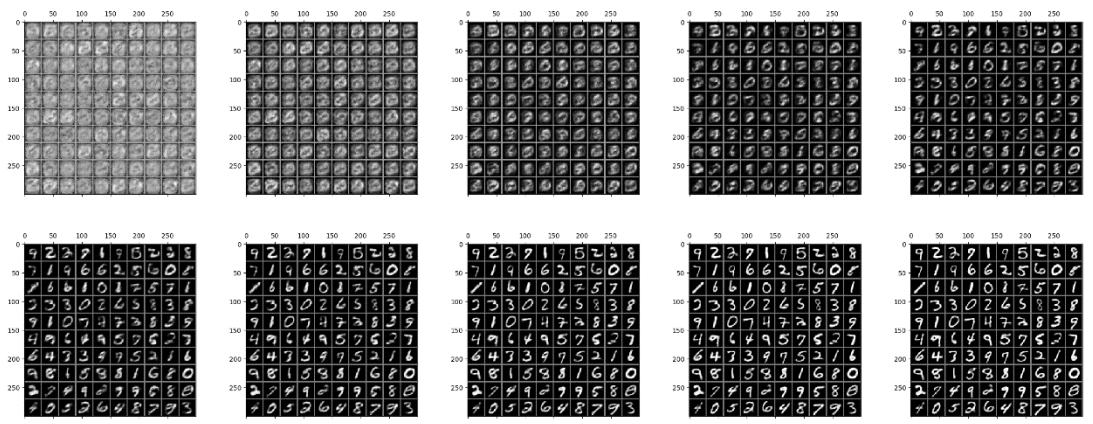


图 1-12 未加入 Attention 机制生成结果图

由于编码器，解码器均为 RNN 模型，设定整个生成过程共有 10 个时刻，从左至右，从上至下，逐渐生成出完整字符，但是仔细看，加入 Attention 机制和不加入 Attention 机制的生成细节是不同的。如上图 1-12，先生成局部，然后下个时刻生成另一个局部，逐渐完整的字符就生成出来了；图 1-13，是从一个模糊的整体逐渐变成一个清晰的整体，相比之

下，Attention机制下的字符生成，更像是人类写字的过程。

所以在该模型中加入Attention机制和不加入Attention机制在生成过程中有着很大的差别，其实在生成更加复杂的图像上，加入Attention机制的DRAW模型要更胜一筹。

如上文提到的，VAE的损失函数由两部分组成：KL散度（正则化项）+负重构误差，训练损失函数（loss）如图：

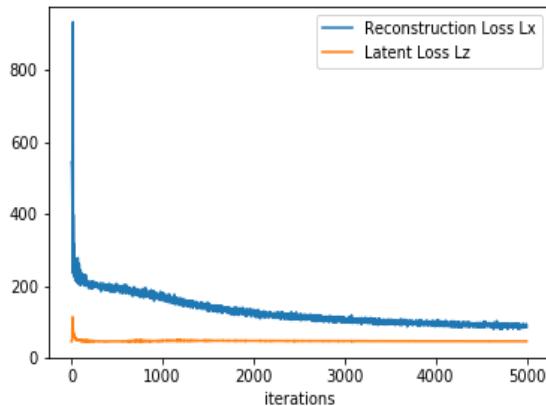


图 1-13 损失函数

### 1.3.3.2 字符风格迁移生成

#### (1) Rewrite: Neural Style Transfer For Chinese Fonts[20]

##### 1) 动机

该模型的作者认为字体设计的过程是一个风格迁移(style transfer)的问题，该模型使用已经设计好的某字体的一部分字符来训练神经网络，训练后的网络会自动将输入的剩余的汉字转化为另一种字体风格的汉字。



图 1-14 中文字符生成

##### 2) 网络

简单的 top-down 卷积神经网络结构：

## Network Structure

Input(size=160x160)
Conv(size=64x64, filters=8) x 2
Conv(size=32x32, filters=32) x n
Conv(size=16x16, filters=64) x n
Conv(size=7x7, filters=128) x n
Conv(size=3x3, filters=128) x 2
MaxPool(size=2x2)
Dropout
Sigmoid
Output(80x80)

图 1-15 Network Structure

- a. 每个 conv 层之后接 batch norm 层然后接 Relu 层，并且一直向下做 zero padding
  - b. 网络使用预测的输出值与 ground truth 之间的 MAE(Mean Absolute Error)
  - c. 不同的层上是不同大小的卷积核，网络可以捕获图像中不同级别的细节
- 3) 实验结果

下图所示是该模型最后的生成效果，每一组的左边一列是对应字符的 ground truth，右边一列是对应的生成的输出值。从生成效果来看，该模型存在的问题如下：

- a. 生成的图像通常是模糊的
- b. 在更多更复杂的艺术字体上生成效果不好
- c. 每次仅限于学习和输出一种目标字体样式，不能一对多



图 1-16 rewrite 生成效果

代码详见：<https://github.com/kaonashi-tyc/Rewrite>

可以看出这种简单的想法，对于风格迁移的生成工作并不能行得通，所以我们考虑更加复杂和有效的模型。

(2) Auto-Encoder Guided GAN for Chinese Calligraphy Synthesis (ICADR2017, 华中科技大学, 白翔)

1) 动机

这篇论文研究了中文书法的合成问题：从标准字体（如黑体）的图像直接生成具有指定风格的书法体的图像。

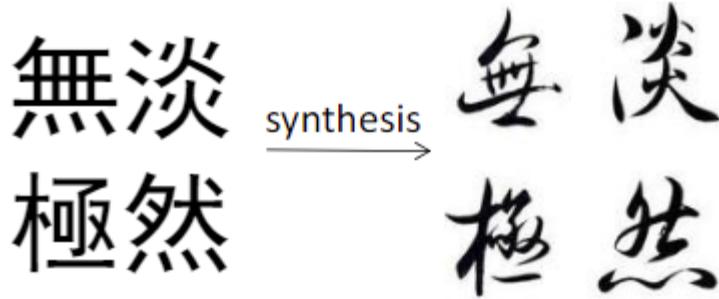


图 1-17 书法合成

2) 网络

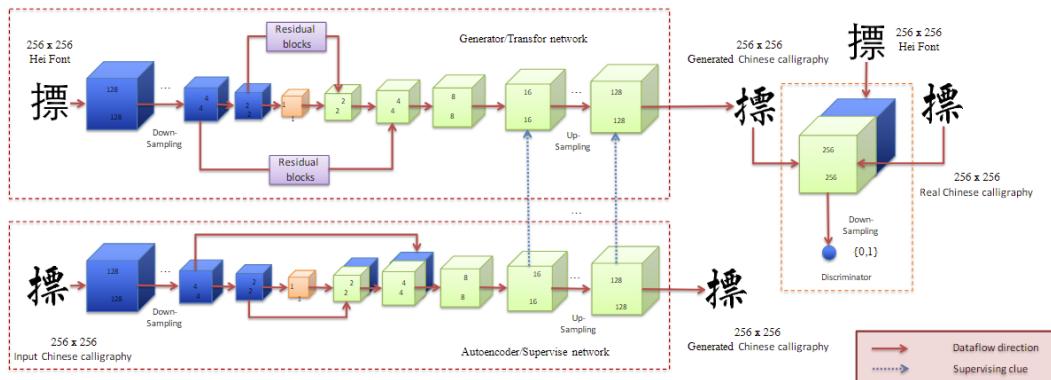


图 1-18 模型结构

整个模型由两部分组成：transfer network 和 supervise network。supervise network 是一个 auto-encoder 网络，encoder 是一系列的 Convolution-BatchNorm-LeakyReLU 块，decoder 是一系列的 Deconvolution-BatchNorm-ReLU 块。Transfer network 也是一个基于 CNN 的 encoder-decoder 网络，输入标准字体的图像，生成特定书法体的字符图像。这两部分网络以 end-to-end 的方式一起训练。

3) 实验结果

下面展示的这张图是论文提出的模型与现有方法的生成效果的比较图，可以看到该模型的生成效果明显超过现有方法，生成的字符图像风格属性上与目标书法字体的风格更像，也更清晰，模糊现象会改善很多。



图 1-19 生成效果

### (3) Separating Style and Content for Generalized Style Transfer (CVPR2018, 上海交通大学)

#### 1) 动机

这篇论文将中文汉字从风格和内容上分离表示，这样可以把训练好的模型泛化到新的字体风格上。

#### 2) 网络

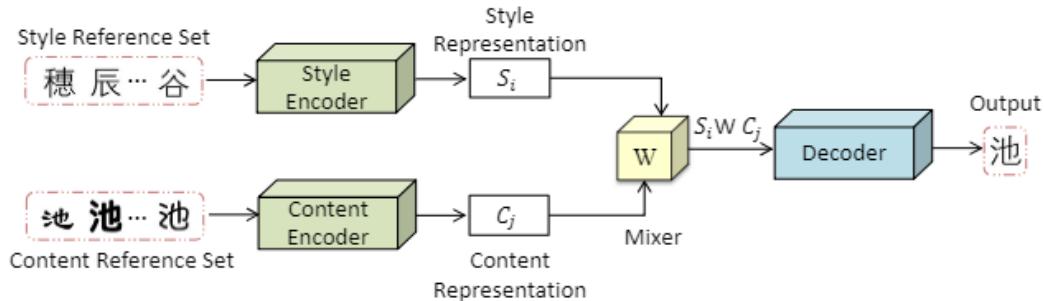


图 1-20 EMD model

该论文提出了 EMD 模型，整个模型是一个 encoder-decoder 网络，由 Style Encoder, Content Encoder, Mixer 和 Decoder 四部分组成：

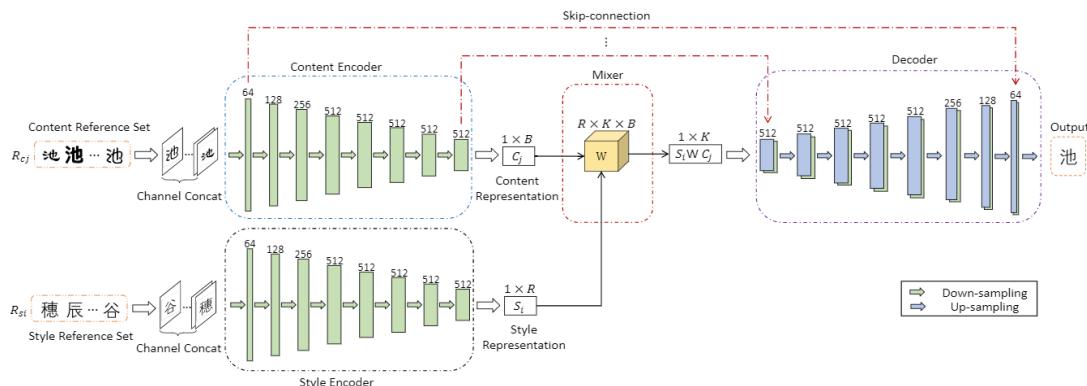


图 1-21 网络细节

Methods	Data format	Generalizable to new styles?	Requirements for new style transfer	What the model learned?
Pix2pix [10]	paired			
CoGAN [14]	unpaired			
CycleGAN [28]	unpaired			
Rewrite [1]	paired			
Zi-to-zj [2]	paired			
AEGN [16]	paired			
Perceptual [12]	unpaired			
StyleBank [5]	unpaired			
Patch-based [6]	unpaired			
Adain [9]	unpaired			
EMD	triplet	The learned model can be generalized to new styles.	One or a small set of style/content reference images.	The feature representation of style/content.

图 1-22 EMD 模型与现有方法的比较

#### 3) 实验结果

下图所示是文章中提出的 EMD 模型与现有其他方法的生成效果的比较，第一行是风格转换的源字体，最后一行是风格转换的目标字体。可以看到该模型无论是字符的生成效果还是 L1 loss, RMSE, PDAR 等评测指标都明显超过现有方法，生成的字符图像风格属性上与目标书法字体的风格更像，也更清晰。证明了把汉字分离成风格和内容这一思路的有效性。

	Source: 昂所挑直帽格梁朴朵酪	Pix2pix: 所朴昂沿格桑染挑直帽	AEGN: 扁膚挑直帽格梁朴朵酪	Zitozi: 昂所挑直帽格梁朴朵酪	C-GAN: 昂所挑直帽格梁朴朵酪	EMD: 昂所挑直帽格梁朴朵酪	Target: 昂所挑直帽格梁朴朵酪	L1 loss	RMSE	PDAR
	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	0.0105	0.0202	0.17
	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	0.0112	0.0202	0.3001
	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	0.0091	<b>0.0184</b>	0.1659
	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	0.0112	0.02	0.3685
	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	件捐娘找走挑期右克炒	<b>0.0087</b>	<b>0.0184</b>	<b>0.1332</b>

图 1-23 生成效果

## 1.4 主流算法介绍

### 1.4.1 zi2zi

该项目[21]是在前面介绍过的 Rewrite[20]项目基础上的改进，是 pix2pix[16]模型在中文字符生成方面的扩展。此神经网络面向中文字符，能够对输入的字符进行风格迁移，通过其内部的生成器和判别器，无监督的自学习，将源字体风格的字符图像生成目标字体风格的字符图像。生成器采用了 8 层 Unet 网络，判别器采用了 6 层卷积神经网络。该模型由 python 语言编写，tensorflow、CUDA、cudnn 等架构支撑，构建起了一个一对多的神经网络，来完成中文字体生成的任务。

神经网络框架如图 4-1 及改进如图 4-2 所示。

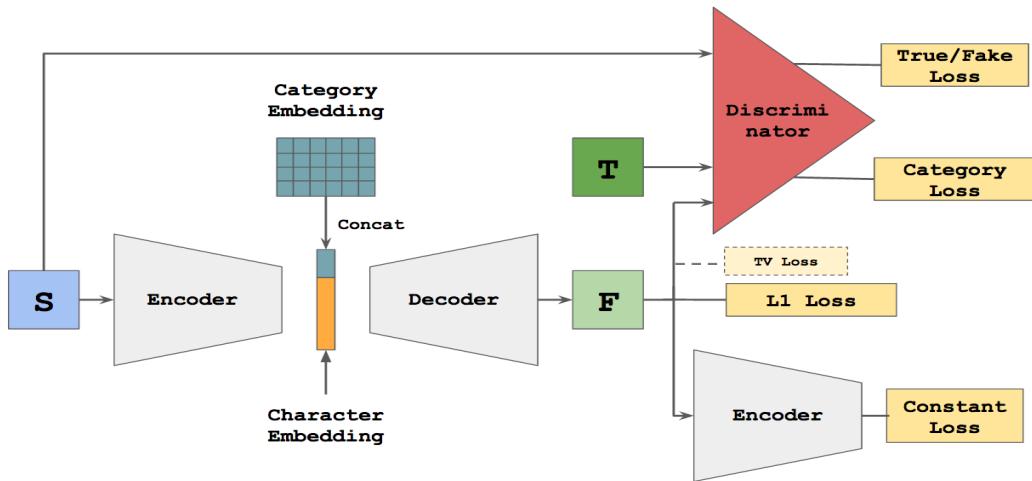


图 1-24 实验神经网络

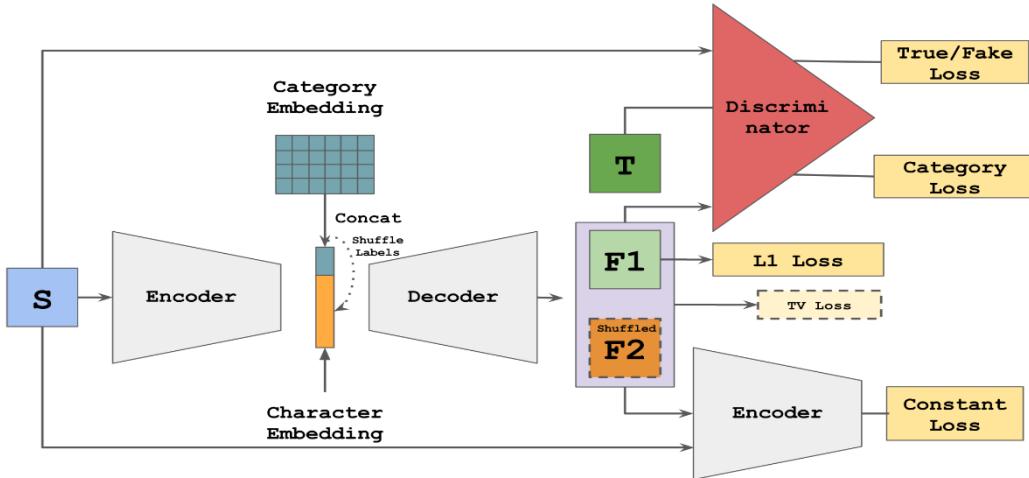


图 1-25 实验神经网络改进

整体流程：S 代表网络的型中，通过 encoder 来变为机器可以学习操作的向量，其中 category 采用一些措施，把字符向量分成了两个方面，一方面是风格部分，一方面是字符（字本身）部分，进行向量分割，随后生成模型开始学习，直至学习完毕，category 还要把风格和字符部分连接起来，作为一个完整的生成输出，再送入 decoder，输出后通过 decoder 变为图片，送入 decoder 后生成我们图中的 F，送入 D 进行判别。图一和图二的网络结构的区别在于 shuffle Labels 这个小算法，这是一个随机选取图片，使选择更具有常规分布的操作算法。Shuffle Labels 是随机在训练图片中挑选，而不是以往选择的顺序抽取训练，使图片的训练更加具有一般的分步性。

S 为原始字体 (source font)，T 是我们要生成的目标字体 (target font)，F 是生成器 G 生成的中文字符图片。

Encoder 和 decoder 属于 CNN 模型中的一部分，encoder 用来把 S 以某种编码形式变为机器识别的向量，decoder 则是反向过程。

Category 在分割学习后，重新组装函数，最后送入判别器 D 判断。

损失函数 (loss function) 的作用是估测网络模型的预测值与真实值的区别程度，它是非负值的实数函数，一般来说我们用  $L(Y, f(x))$  代表 loss function，loss function 的值越小代表网络模型的鲁棒性越好。Loss function 是经验风险函数的核心，同时它也是结构风险函数及极其重要的部分。网络模型中由经验风险项以及正则项组成结构风险函数，其公式表达为：

$$\theta^* = \arg \min_{\theta} \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i; \theta)) + \lambda \Phi(\theta) \quad (\text{公式 1-8})$$

前面的均值部分代表了经验风险函数，L 如上代表 loss function，最后的部分是正则化项，也叫惩罚项，可以是一些正则函数。

生成模型中 encoder 和 decoder 都为 8 层的 CNN 网络。

图 1-25 中 encoder 和 decoder 采用了论文 pix2pix 中的 Unet 模型，Unet 是深度学习的分割网络。在 Unet 的图像分割问题中，我们一般的目的的是检测出图片中物体的方框和轮廓。如图 4-4 所示。



图 1-26 图像边缘检测

当我们使用深度卷积网络[10]时，如果不加以处理，会出现输出的像素数低于输入，而我们要求的结果是想使输入输出一致。我们可以选择加入边缘像素或是上采样来使卷积神经网络得到一样输出大小的图片。

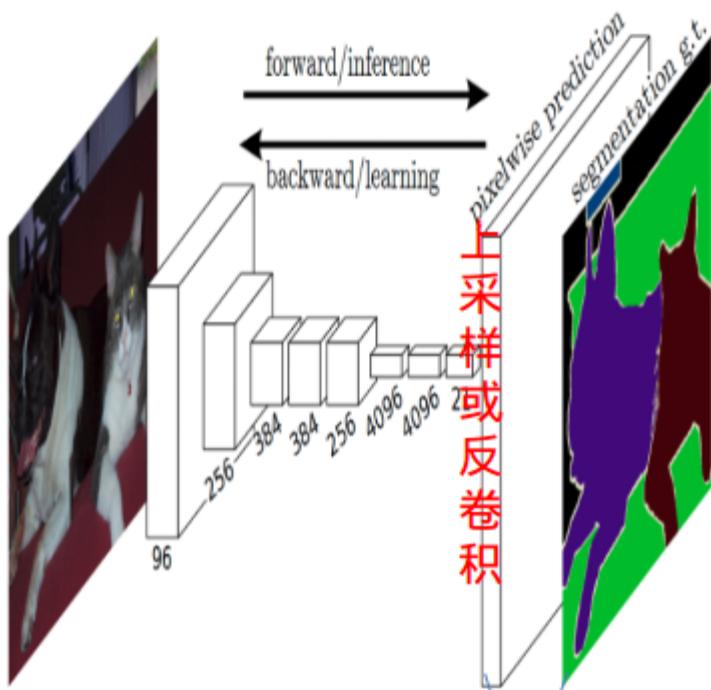


图 1-27 上采样过程

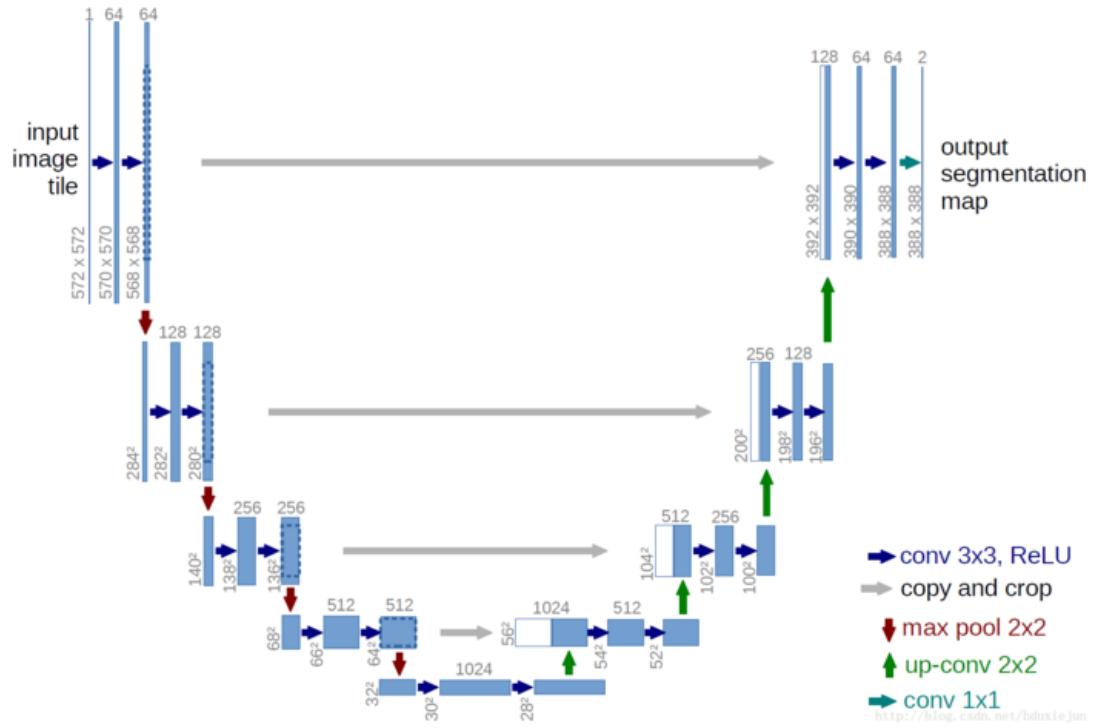


图 1-28 UNet 结构

Unet 的结构如图 4-6 所示，Unet 由两部分组成：

特征提取部分（左半边）：在此部分，用来学习特征，完成我们的学习动作。

上采样部分（右半边）：此部分如我们所说，进行输入和输出的规整化，得到一致的图像，也就是一定的拼接。

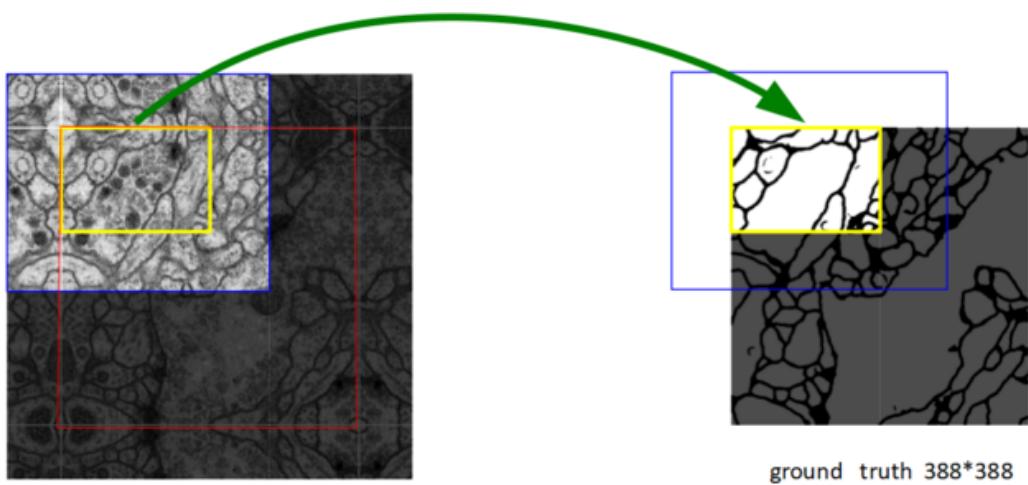


图 1-29 特征提取展示

特征提取时 Ground truth 提取了 388\*388，我们的输入为 572\*572

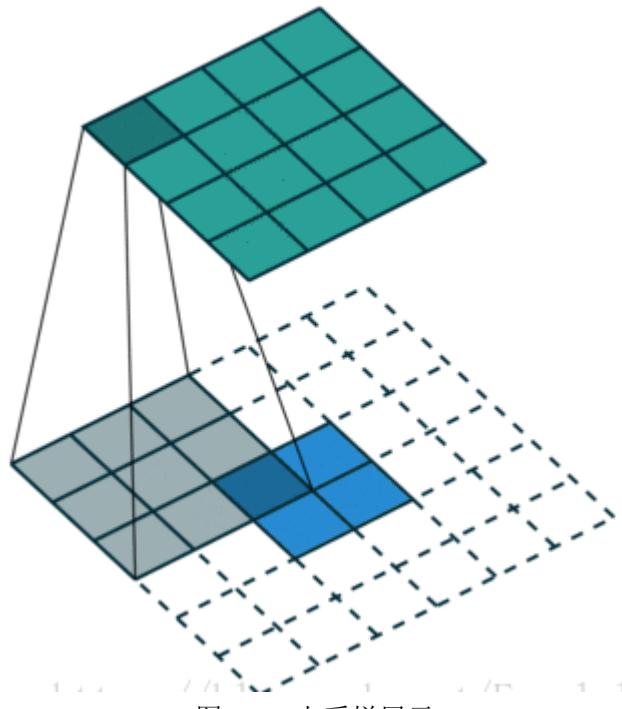


图 1-30 上采样展示

我们会发现，为了使输出与输入像素相同，选择了在中间层添加一些噪声或特征，来完成我们的任务。

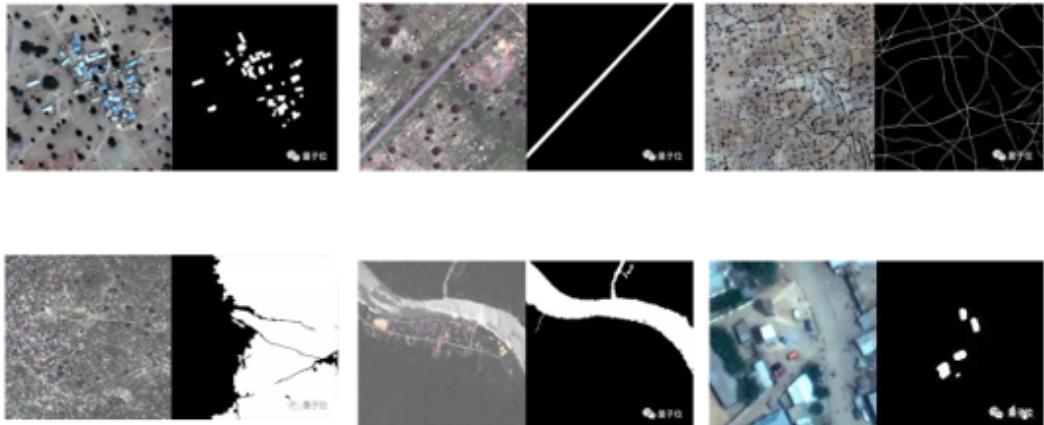


图 1-31 好的分割效果图

此外，encoder 和 decoder 之间添加了 category。Category 的出现是为了完成字体转换时，可以一对多转换。一对多转换就是源字体可以在此网络中转化成多种字体，而不是一般网络的只能一对一的转化，即特定的输出。

判别模型由一个 CNN 组成，结构为四层卷积层和两层全连接层。

#### 1.4.2 SA-VAE

##### (1) 动机

文章题目是《Learning to Write Stylized Chinese Characters by Reading a Handful of Examples》(IJCAI 2018, 清华大学, 朱军)。

整个生成任务分为两个部分：1.推断(Inference) 2.生成(generation)

在推断部分通常就是风格迁移的任务，该任务一般分为两个部分：1.学习风格(Style Inference Network) 2.学习内容(Content Recognition Network)。然后将学到的风格和内容

整合在一起，再通过解码器生成出风格迁移后的字符。

来自清华大学的研究者就基于这种思想，提出一个风格化迁移生成任务的 VAE 模型：  
Style-Aware Variational Auto-Encoder (简称 SA-VAE)，模型如图：

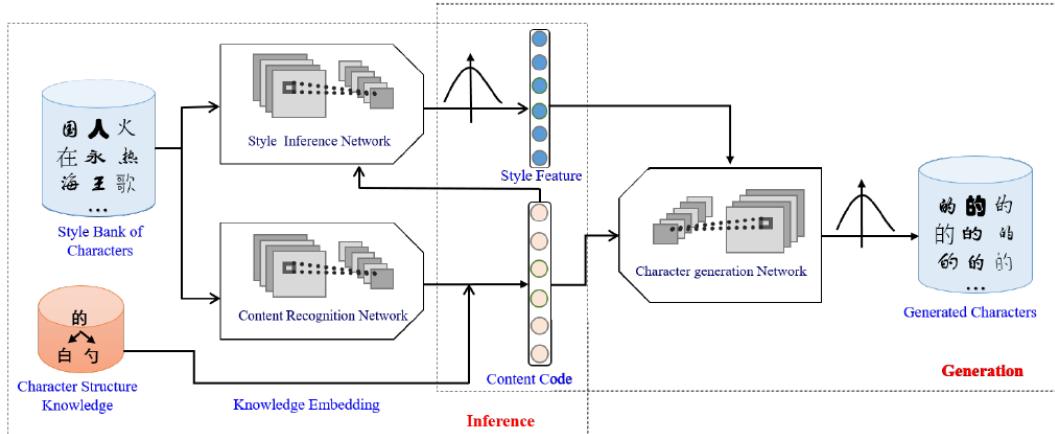


图 1-32 SA-VER 示意图

## (2) 贡献

1) 使用了 VAE 框架

a. 相比于 GAN, VAE 拥有后验推断的能力去揭示潜在的特征，例如从提供的观测字符得到的风格

b. 与对 GAN 的推断扩展相比，VAE 产生更好的推理效果

2) 交叉成对优化方法(Inter-cross pair-wise optimization method)

3) 结构信息作为先验得到更多信息内容表示

3) 网络

假设每个字符  $x_{i,j}$  都是由其内容和风格两部分构成：

$$x_{i,j} \leftarrow (S_j, C_j) \quad (\text{公式 1-9})$$

则建立一个 style bank:

$$X = \{x_{i,j}\}, i = 1, 2, \dots, M, j = 1, 2, \dots, N \quad (\text{公式 1-10})$$

模型提出三个子网络(如上图):

- Content Recognition Network  $C$
- Style Inference Network  $S$
- Character Generation Network  $G$

### a. Content Recognition Network

模型必要的部分：

$$y = f_\beta(x) \quad (\text{公式 1-11})$$

其中， $x$ : 表示字符图像  $y$ : 表示内容的标签  $\beta$ : 表示网络的参数

网络采用：VGG, ResNet 或者 DenseNet

### b. Character Structure Knowledge

相较于 one-hot 嵌入，该模型的编码方法可以利用存在于中文字符的结构信息和基础信息。这样内容标签  $y$  的 one-hot 编码，转化为如下表的独特的编码方式：

$$K: c \leftarrow y$$



汁	林	冷	…	乱	你	他	作	…	叶	…
华	志	泉	…	男	思	想	态	…	恋	…

图 1-33 编码方式

上图表表示了编码方式，其中包含了结构信息和基本信息

II	打、和	三	莫、意	厂	庆、房	口	凶、函
III	树、街	辶	边、建	匚	风、冈	匚	国、四
匚	昌、志	口	可、司	匚	区、匠	匚	不、大

图 1-34 字符结构示例

所有 12 种中文字符的结构信息和一些例子。

灬	烈、热	宀	宝、家	氵	海、洋	彑	彤、杉
辵	边、远	阝	阳、阴	刂	刚、刘	扌	提、打
忄	情、怀	夊	社、视	讠	训、说	…	

图 1-35 字符的一些基本结构

一些具有代表性的基础部分在中文字符中频繁的使用。

### c. Style Inference Network

这部分主要是基于卷积神经网络的，因为卷积核（filter）可以提取字符图像的局部特征。书写的风格充满着不确定性，因此我们使用一个随机网络建模，网络可以看作一个各向同性的高斯分布，参数由卷积神经网络拟合。

$$S : q_{\phi}(s | x, c) = N(s | \mu_{\phi}(x, c), diag(\sigma_{\phi(x, c)}^2)) \quad (\text{公式 1-12})$$

我们的目标是要使得下面等式成立：

$$q_{\phi}(s_i | x_{i,j}, c_j) = q_{\phi}(s_i | x_{i,k}, c_k) \quad (\text{公式 1-13})$$

上式等式的含义是，对于拥有同一风格的不同字符，输出的风格的分布是相同。值得注意的是，图像  $x$  和内容编码  $C$  都作为网络的输入

### d. Character Generation Network

内容编码  $C$  和风格特征  $S$  作为独立的输入。同样的，生成网络也是建立为随机网络，我们使用一个伯努利分布，参数由反卷积网络(deconvolutional neural network)来拟合，最终重建一个二值化图像，模型可以如下表述：

$$G : p_{\theta}(x | s, c) = Bern(x | \mu_{\theta(s, c)}) \quad (\text{公式 1-14})$$

那么整个模型由下面示意图表示：

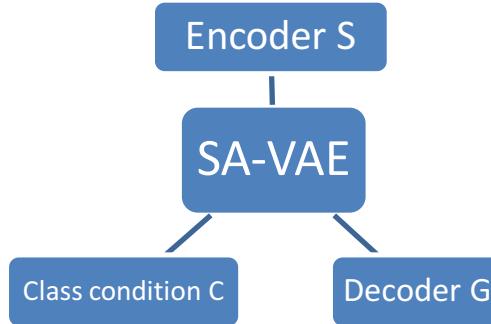


图 1-36 模型简图

e. 优化方法 (Intercross Pairwise Optimization)

$$\begin{aligned} \log p_\theta(x|c) &\geq \log p_\theta(x|c) - KL[q_\phi(s|x,c) \| p_\theta(s|x,c)] \\ &= L_{ELBO}(x;c,\theta,\phi) \end{aligned} \quad (\text{公式 1-15})$$

$$\tilde{L}(x,x',c,c';\theta,\phi) = \log p_\theta(x|c) - KL[q_\phi(s|x',c') \| p_\theta(s|x,c)] \quad (\text{公式 1-16})$$

由于数据的似然函数是不变的，因此最大化似然函数  $L$  等价于最小化 KL 散度：

$$KL[q_\phi(s|x',c') \| p_\theta(s|x,c)] \quad (\text{公式 1-17})$$

和：

$$KL[q_\phi(s|x,c) \| p_\theta(s|x',c')] \quad (\text{公式 1-18})$$

进一步，似然函数可以写作：

$$\tilde{L}(x,x',c,c';\theta,\phi) = E_{q_\phi(s|x',c')} [\log p_\theta(x|s,c) - KL[q_\phi(s|x',c') \| p_\theta(s)]] \quad (\text{公式 1-19})$$

下面是 Intercross Pairwise Optimization 的示意图：

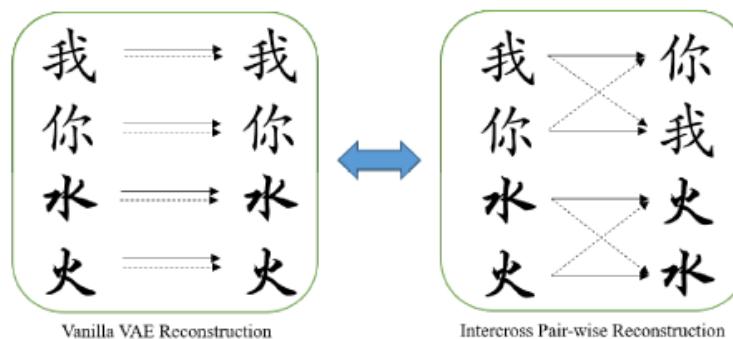


图 1-37 优化算法

上图是原生 VAE 模型与 SA-VAE 模型 intercross pairwise 训练方法的对比，其中实线和虚线分别表示要转化的风格和内容。

具体算法流程图：

---

**Algorithm 1:** Training Algorithm

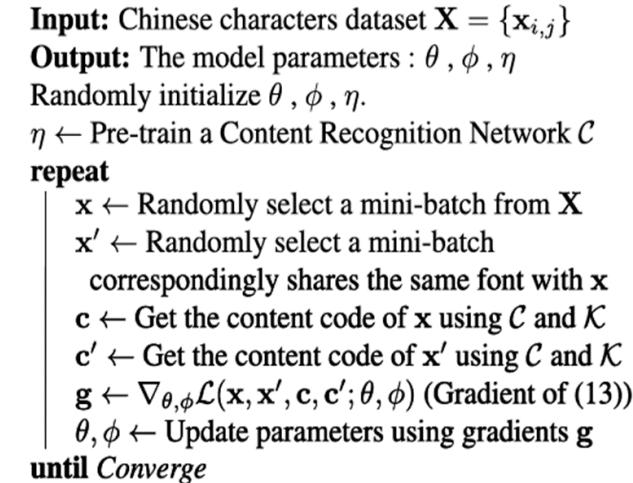
---

**Input:** Chinese characters dataset  $\mathbf{X} = \{\mathbf{x}_{i,j}\}$   
**Output:** The model parameters :  $\theta, \phi, \eta$   
Randomly initialize  $\theta, \phi, \eta$ .  
 $\eta \leftarrow$  Pre-train a Content Recognition Network  $\mathcal{C}$   
**repeat**  
     $\mathbf{x} \leftarrow$  Randomly select a mini-batch from  $\mathbf{X}$   
     $\mathbf{x}' \leftarrow$  Randomly select a mini-batch  
        correspondingly shares the same font with  $\mathbf{x}$   
     $\mathbf{c} \leftarrow$  Get the content code of  $\mathbf{x}$  using  $\mathcal{C}$  and  $\mathcal{K}$   
     $\mathbf{c}' \leftarrow$  Get the content code of  $\mathbf{x}'$  using  $\mathcal{C}$  and  $\mathcal{K}$   
     $\mathbf{g} \leftarrow \nabla_{\theta, \phi} \mathcal{L}(\mathbf{x}, \mathbf{x}', \mathbf{c}, \mathbf{c}'; \theta, \phi)$  (Gradient of (13))  
     $\theta, \phi \leftarrow$  Update parameters using gradients  $\mathbf{g}$   
**until** Converge

---

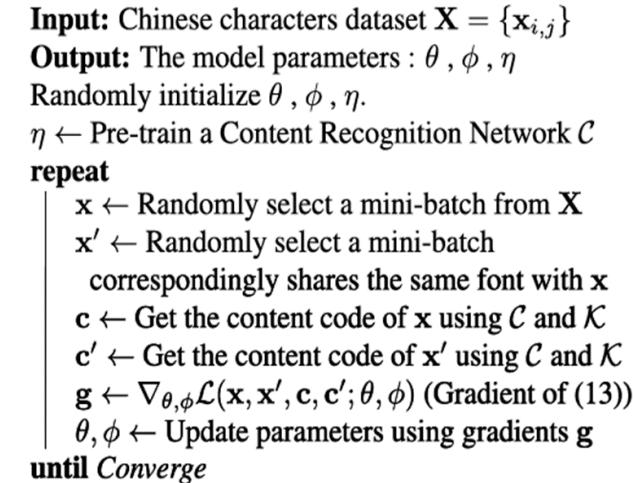
图 1-38 算法流程图

## f. 实验



刘山不在高有仙则鸣水不在深有龙则灵  
刘山不在高有仙则鸣水不在深有龙则灵  
刘山不在高有仙则鸣水不在深有龙则灵  
刘山不在高有仙则鸣水不在深有龙则灵

(a) An excerpt from a classical essay Lou Shi Ming written by the famous poet Yuxi Liu



孔丘学而时习之不亦悦乎有朋自远方来不亦乐乎  
孔丘学而时习之不亦悦乎有朋自远方来不亦乐乎  
孔丘学而时习之不亦悦乎有朋自远方来不亦乐乎  
孔丘学而时习之不亦悦乎有朋自远方来不亦乐乎

(b) An excerpt from the Analects of Confucius

图 1-39 实验结果

实验中，图 a 是实验设置为 one-shot 模式，从左边虚线框中生成得到一系列汉字字符。奇数行是 SA-VAE 模型生成结果，偶数行是 Ground truth. 上面是打印体，下面是手写体。图 b 是实验设置为 few-shot 模式。生成总体来说很好，但也可以看出手写字体的生成效果不如打印体的。

## 1.5 实验室现有算法介绍

### 1.6 zi2zi 中文字符生成实验

#### 1.6.1 实验过程

在实验之前我们需要准备好需要进行转换的字体文件，一般为.ttf 或者.otf 格式。第一步我们要将字体文件转换为图像，并且要将选中的源字体和目标字体的对应字符图像拼接在一起（如宋体的“薛”对应楷体的“薛”），组成样本。第二步要将图像和对应的 label 转换为 binary 格式，并按比例将样本分开，一部分用于 training，一部分用于 validation。第三步就可以开始训练网络了。

## 1.6.2 实验结果

总共四组实验，为了更好的比较字符的生成效果，每组实验的源字体都固定为楷体，目标字体两组实验选择了常规字体，两组实验选择了手写体。四组实验的目标字体分别为思源黑体，彩云简体，陈代明手写体，豆豆手写体。每组实验结果的示例图里包含目标字体的 ground truth 和生成结果的匹配对。

其中，第一组和第二组实验选择的目标字体是常规字体，第三组实验和第四组实验选择的目标字体是手写体。

第一组实验的生成结果：



图 1-40 字符生成结果

该组实验选取的源字体是楷体，目标字体是思源黑体，特点是笔画又粗又黑，从生成结果来看，模型很好的学会了该字体的书写风格，一些笔画很复杂的字也可以有很好的生成效果。

第二组实验的生成结果：



图 1-41 字符生成结果

该组实验选取的源字体是楷体，目标字体是彩云简体，该字体最大的特点是笔画是中空的，源字体和目标字体的差距很大，不过从生成结果来看，模型还是学到了中空的特点，不过也有一些生成效果不好的例子，一些笔画复杂的字的笔画不是很清晰。

第三组实验的生成结果：

通	通	常	常	雅	雅	杳	杳
例	例	法	法	鑫	鑫	金	金
羽	羽	斩	斩	若	若	从	从
究	究	臻	臻	魅	魅	帆	帆
胥	胥	瞧	瞧	宠	宠	烙	烙
欠	欠	强	强	舭	舭	浊	浊
也	也	垆	垆	圜	圜	亭	亭
采	采	狐	狐	妻	妻	执	执
好	好	劙	劙	狃	狃	械	械
咷	咷	士	士	跗	跗	涂	涂
畀	畀	偒	偒	嗬	嗬	蚜	蚜
廩	廩	解	解	捱	捱	籽	籽
节	节	那	那	豁	豁	碎	碎
父	父	稟	稟	冗	冗	卯	卯
泛	泛	钉	钉	骀	骀	揣	揣
炯	炯	咯	咯	擎	擎	拱	拱

图 1-42 字符生成结果

该组实验选取的源字体是楷体，目标字体是陈代明手写体，字体清秀隽永，是手写体的正楷，实用性很强。一般而言，手写体因为具有更强烈的风格属性，生成难度更大。从结果来看，生成效果整体不错，生成的字符跟 ground truth 的风格写法都很像，当然也存在少量笔画比较模糊的情况。

第四组实验的生成结果：



图 1-43 字符生成结果

该组实验选取的源字体是楷体，目标字体是豆豆手写体，从图片中可以看出，豆豆体的风格属性很强，字体活泼有趣，俏皮可爱，趣味性足，适合用于海报设计，儿童包装，幼儿园展板等广告设计使用。从最后的生成结果来看，字体整体生成效果很好，少量笔画比较复杂的字存在不清晰的现象。

### 1.6.3 loss 图示及解释

loss 函数是表示机器对训练的把握程度，表示机器学习到的表示与真实数据的差别情况。loss 在一般我们的训练过程中，是越来越小的，数据拟合情况越好，loss 越低。本节选取了第一组实验的 loss 变化曲线来量化的分析模型的学习情况。

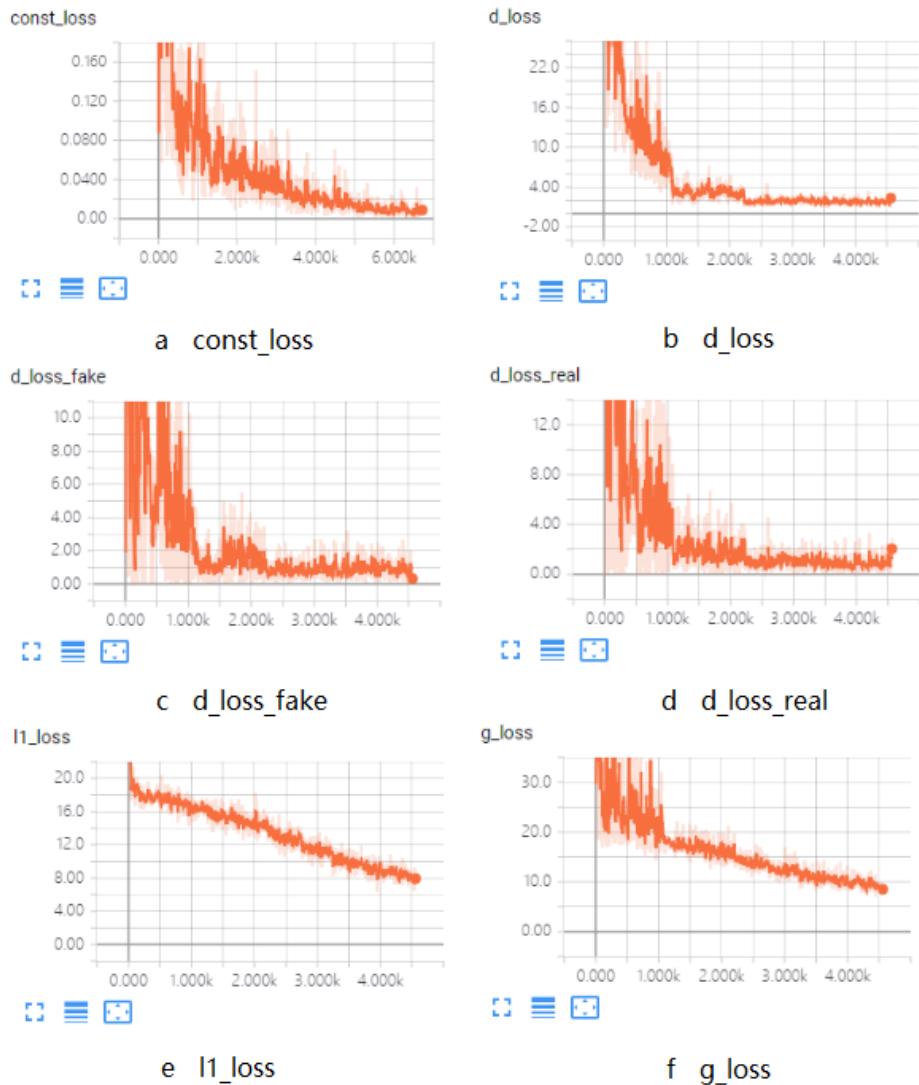


图 1-44 loss 变化曲线

图中的 Loss 函数的变化曲线有锯齿状，是因为随机选取数据，模型方法以及参数设置等原因，所以会出现了波动的曲线，loss 在降低之后又有些许上升。图中的 loss 曲线整体都是下降的趋势，这代表着随着训练的进行，生成器生成数据与真实数据的差距越来越小，判别器判别图片真假的能力越来越强。

其中 a 中的 `const_loss` 衡量的是源字体和目标字体的对应字符在嵌入空间中的距离大小，前期的波动，属于数据随机选取引起的。由于实验的转换字体相近，所以最终 loss 的曲线慢慢的变成了 0。

b 中的 `d_loss` 衡量的是判别器的判断能力。当判断一张图像时，和图片本身真正的属性与之判断出的结果对比。结果对错，加权至判断的 loss 函数中。可以发现在实验轮次的进行下，我们的 loss 函数也在慢慢的降低。

c 中的 `d_loss_fake` 表示的是判别器在判断图片为假的正确率。在我们的实际学习中，不能够笼统的去通过成功失败来判断，要知道在哪方面成功了，哪方面失败了。这个 loss 的意义就在于把综合判断分割了，当我们的效果不理想时，我们可以通过观察，来从某个弱项具体的来解决我们的模型问题。

d 中的 `d_loss_real` 表示的是判别器在判断图片为真是的正确率。

e 中的 `g_loss` 衡量的是生成器的生成能力。生成能力与判别器反馈的结果相关，当判

别器判别成功或失败时，会反馈给生成器以信息，生成器会以这个信息来优化自己以后的生成数据，最终在 7 左右的点，表示此模型在这个方面还是可以提高。

f 中的  $l1\_loss$  衡量的是图片的预测值与真实值的偏差程度。预测值是指生成模型的生成出的图片，而真实值是指我们真实提供的图片。在实验中，也就是说，生成器生成的图片不仅要骗过判别器，还要与真实的图片差别不大，这样才能以假乱真。

## 1.7 本章总结

首先，我们在这个实验进行的运行时，都采用的一对一的训练方式，而我们的模型已经能够支持一对多的训练方式，我们没有完全充分的利用模型优势，也是由于自己的实验器件的条件限制，所以这方面在以后的研究过程中我会努力去继续深刻研究这个方面。

其次，此模型对于形状差距不是很大的字体之间的转换，而当我们使用两个差距较大的字体时效果往往不是很好，需要训练很多轮，才能够达到一个我们勉强认出的状况。模型的鲁棒性和适应性还没有太好。

除此之外，在学习中，我们还会遇到一些学习的字体模糊的情况，这说明有些方面，模型的稳定性还不是很好，我们要在模型的代码和层数，以及每层的功能来改善此方面。

最后，我们的数据集学习时要千级别以上的样本，所以对于我们要求的小数量字体还是有些多，因为稍大的数据集对于字体设计师来说任务也有些不容易，我们要尽可能的降低需要人工操作的步骤，尽可能的缩减需要的样本数。

此次实验属于中文字符的风格转换，其实此类实验有很大的发展前景，除了对字符有很好的效果外，也可以在图片风格转移，真伪辨别等场景应用。对待不同的应用类型来说，我们的网络结构、层数、参数等要进行改动和重新设计，我们也可以选用新近的技术，参考比较新的论文思想，来进行我们此方向的深刻研究。

## 参考文献

- [1] Kingma D P, Welling M. Auto-encoding variational bayes. arXiv:1312.6114, 2013.
- [2] Gregor K, Danihelka I, Graves A, et al. DRAW: A recurrent neural network for image generation. arXiv:1502.04623, 2015.
- [3] Xu K, Ba J, Kiros R, et al. Show, attend and tell: Neural image caption generation with visual attention[C]//International Conference on Machine Learning. 2015: 2048-2057.
- [4] Graves A. Generating sequences with recurrent neural networks. arXiv:1308.0850, 2013.
- [5] Graves A, Wayne G, Danihelka I. Neural turing machines. arXiv:1410.5401, 2014.
- [6] Sun D, Ren T, Li C, et al. Learning to Write Stylized Chinese Characters by Reading a Handful of Examples. arXiv:1712.06424, 2017.
- [7] Salimans T, Kingma D, Welling M. Markov chain monte carlo and variational inference:Bridging the gap[C]//International Conference on Machine Learning. 2015: 1218-1226.
- [8] George D, Lehrach W, Kansky K, et al. A generative vision model that trains with high data efficiency and breaks text-based CAPTCHAs[J]. Science, 2017, 358(6368): eaag2612.
- [9] Michael Jordan, J.Kleinberg et al. Pattern recognition and Machine learning[M], Singapore:Springer,2007.10.
- [10] 伊恩.古德费洛, 约书亚, 本吉奥等.深度学习[M]. 赵申剑, 符天凡译. 北京: 人民邮电出版社, 2017.
- [11] Ha D, Eck D. A neural representation of sketch drawings[J]. arXiv preprint arXiv:1704.03477,2017.
- [12] Unsupervised Representations Learning With Deep Convolutional Generative Adversarial Networks
- [13] Generative Adversarial Networks Ian J. Goodfellow, Jean Pouget Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley
- [14] Mirza M , S. Conditional generative adversarial nets. arXiv preprint arXiv: 1411.1784, 2014
- [15] Salimans T, Goodfellow I, Zaremba W, et al. Improved techniques for training GANs. arXiv preprint arXiv, 2016.
- [16] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks .In CVPR, 2017.
- [17] Alec Radford, Luke Metz, Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv: 1511. 06434, 2015
- [18] Martin Arjovsky, Soumith Chintala, Léon Bottou. Wasserstein GAN. arXiv: 1701.07875
- [19] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, and Zhen Wang. Least squares generative adversarial networks .CoRR, abs/1611.04076, 2016b.
- [20]Yuchen Tian. Rewrite: Neural style transfer for Chinese fonts, 2016. <https://github.com/Kao-nashi-tyc/Rewrite>.
- [21]Yuchen Tian. zi2zi: Master chinese calligraphy with conditional adversarial networks, 2017. <https://github.com/kaonashi-tyc/zi2zi>.

## 第2章 图像字符检测

### 2.1 字符检测定义

光流字符识别（OCR）任务当中的一个重要子任务，任务主要是将图像上的字符区域以准确的坐标构成候选框（左，上，右，下）的形式表示出来。表示的越准确则说明算法的检测效果越好。当下的自然场景字符识别任务都以字符检测任务为前提，先将字符区域从整图中截取，之后输入到识别算法当中进行识别任务。

### 2.2 数据集说明

#### 2.2.1 数据集意义

近年来，深度检测算法在图像领域得到了长远的发展，字符检测任务也得到了长足的推进。因此作为需要学习的模型来说，足够的数据则是模型取得优秀效果的前提。并且，近年来提出的数据集也越来越难以通过简单的检测算法来取得很好的效果，因此数据集也是检验算法鲁棒性的一种最重要的手段。

#### 2.2.2 常见数据集介绍

##### (1) ICDAR2013

ICDAR2013 又称 Focused Scene Text Challenge，数据集在 2013 年发布，作为 2013 年的 Robust Reading Competition 的标准数据集。图片多为水平字符区域，区域较为居中。数据集中 229 张图片组成训练集，233 张图片组成测试集。示例如下，数据的标注为每个字符区域由 left, top, right, bottom 四个边界坐标组成。

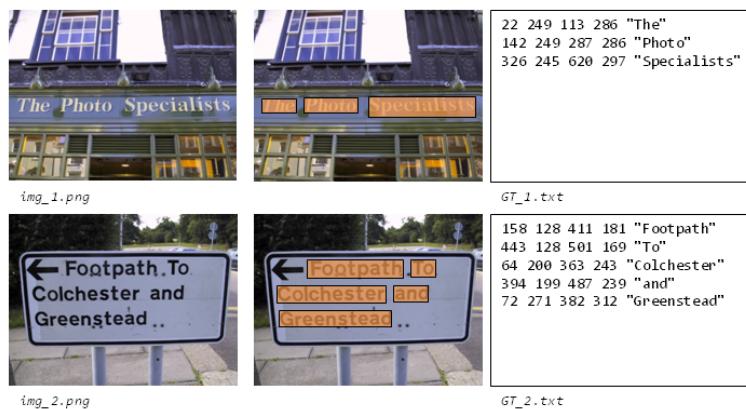


图 2-1 ICDAR2013 原图以及标注

##### (2) ICDAR2015

ICDAR2015 又称 Incidental Scene Text Challenge，数据集在 2015 年作为当年的 Robust Reading Competition 的标准数据集发布。图片中的字符区域相较 ICDAR2013 出现的更为不规则，出现的字符区域形状多呈任意四边形（因为相机视角问题而产生的透视变换）。整个数据集样本数量也远大于 ICDAR2013。训练集样本数量为 1000，测试集样本数量为 500。示意图如下，因为出现的字符区域并不是规整的长方形，因此标注的形式与 ICDAR2013 有所区别，标注的信息为字符区域不规则四边形的四个顶点横纵坐标(x1, y1, x2, y2, x3, y3, x4, y4)，所给的字符内容若标志为“####”，则该区域内的字符不需要做识别。

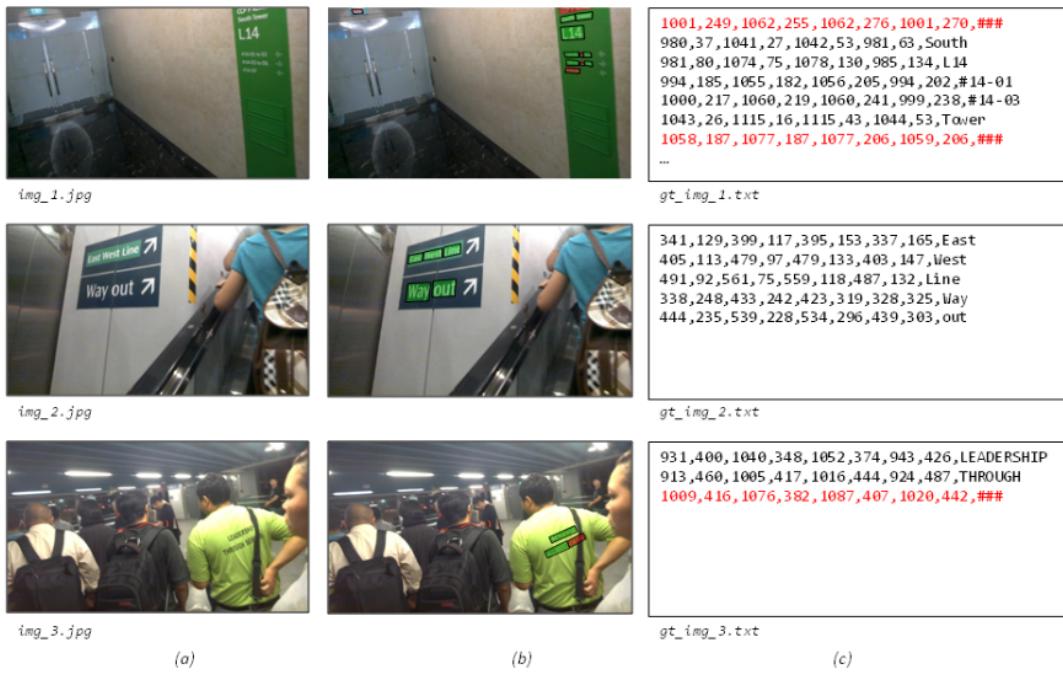


图 2-2 ICDAR2015 原图以及标注

### (3) ICDAR2017-MLT

ICDAR2017-MLT 又称作 Multi-lingual scene text detection。是 Robust Reading Competition 系列第一次发布的多语种的字符检测与识别数据集。该数据集涉及的语言有 6 种，总共为 "Arabic", "Latin", "Chinese", "Japanese", "Korean", "Bangla"（阿拉伯语，拉丁语，汉语，日语，韩语，孟加拉语）。数据集总共包括 9000（7200 张用于训练，1800 张用于验证）张训练图片，测试集需要向组织者 (nibal.nayef@univ-lr.fr) 发送邮件进行获取。标注的形式与 ICDAR2015 相近，标注为任意四边形的四个顶点坐标，以顺时针顺序排列 (x1, y1, x2, y2, x3, y3, x4, y4)，最后两项为语言种类和语言内容。标注文件编码为 utf-8。相较 ICDAR2015 数据集，数据样本数量再次增大。

### (4) ICDAR2017 COCO-Text

该数据集的数据样本来自著名的图片数据集 MS COCO，该数据集由康奈尔大学发布，数据集标注了数据集当中所有的字符区域，并对字符区域进行了较为细粒度的标注，包括，是否易读 (legible / illegible)，是否是英文 (English / non-English)，是否印刷体 (machine printed / handwritten / others)。训练集采用 43686 张图片作为训练样本，10000 张图片作为验证集，10000 张图片作为测试集。总共为 63686 张图片。标注格式为，左上角横纵坐标与长方形框的长宽 (x, y, w, h)。标注示意如下：



图 2-3 ICDAR2017 COCO-Text 数据集图片及标注

#### (5) RCTW-17

RCTW-17 由华中科技大学发布,数据样本特点主要为经过透视变换的多朝向字符区域,该数据集提供检测和识别两级标注。检测字符区域的标注为四个顶点坐标( $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$ )以及对应字符区域的内容。该数据集语言主要为中文和英文两种。中文为按行标注,英文为按词标注。训练集包括 8034 张图片,测试集图片若干(总数超过 12000 张)。图片样本如下图:



图 2-4 ICDAR RCTW 图片样例

#### (6) SCUT-CTW1500

由华南理工大学发布,数据样本主要是针对形变更为强烈的弯曲字符区域进行采集。该数据集包括 1000 张图片用于训练, 500 张图片用于测试。数据标注的格式则为 polygon(多边形)每个顶点的横纵坐标。样本图片示意如下(图 2-5):



图 2-5 SCUT-CTW1500 图片样本

#### (7) MSRA-TD500

该数据集由微软亚洲研究院的 Cong Yao 发布, 数据集内容主要针对多朝向字进行标注。数据集包括 300 个训练样本, 以及 200 个测试样本。标注的格式包括任意字符朝向的中心点  $x$ ,  $y$ , 任意朝向的矩形框的长宽, 最后是弧度制的角度  $\theta$ 。标注示意如下图 (图 2-6):

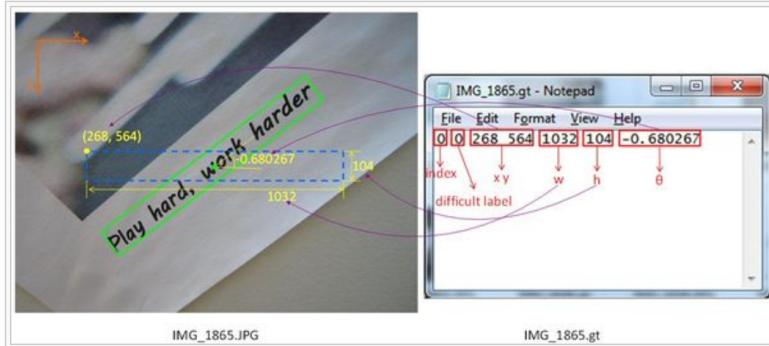


图 2-6 图片标注样例

### 2.3 字符检测综述

字符检测任务是光流字符识别总任务中的子任务, 也是字符识别任务的前置任务, 检测结果的好坏对随后识别结果的影响至关重要。如前所述, 字符检测任务需要算法将图片中的字符区域以坐标位置的形式预测出来。由于在实际的场景当中, 图片当中的字符区域因拍摄角度的关系, 所呈形状并不一定是水平或竖直的矩形形状, 实际出现的形状总是会经过一定程度的透视变换从而呈现的形式通常是不规则的四边形。如果模型预测产生的字符区域不够紧致, 则会对随后的识别模型预测的结果产生很大影响, 通常会使得整个预测结果质量下滑, 如下图 (图 2-7) 所示:



图 2-7 PhotoOCR 上的识别结果

图 2-7 为两种模型预测的检测区域在识别算法 PhotoOCR 上的识别结果, 第一种算法预

测的结果是水平矩形，而其中的字符呈现方式并不是水平的。第二种算法预测的是带字符区域朝向的矩形，可以预测出字符区域的朝向。明显看出后者预测出来的字符区域，明显比前者的预测结果要更为准确和紧致。从识别的结果来看，后者能够预测出准确的结果，而前者无法识别。检测任务的重要性由此可见一斑。

目前字符检测算法逐渐趋向于采用深度神经网络模型对字符区域位置进行预测。而目前所采用的深度神经网络字符检测算法主要分为两种，其一为基于检测框的通用物体检测（generic object detection）框架进行改进，用以预测更为紧致的字符区域，其二则是采用基于图像分割任务深度学习算法，预测出字符区域像素级的位置，再通过轮廓计算出最后的字符区域。如下图（图 2-8）所示：



图 2-8 当下两种主流字符检测算法结果：

左图为原图，中图为图像分割算法结果，右图为候选框检测结果。

因此为了衡量算法的性能，当下几大标准数据集（benchmark）都通过评估矩形框交并比的办法给出定量指标。

正确的检测结果：

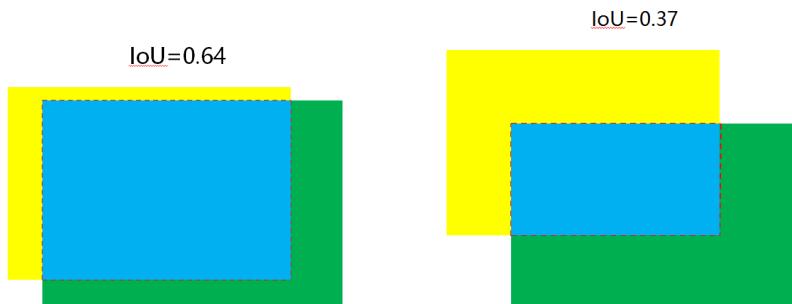


图 2-9 预测结果与标注区域交并比

我们通过计算两个矩形框的交并比的值，并与规定好的检测正样本阈值（IoU=0.5）作为对比，筛选出正确的检测结果与错误的检测结果。记标注区域为 GT，预测的结果为 P。则交并比的计算方式为： $(GT \cap P) / (GT \cup P)$ 。计算好的值取值范围在 (0, 1)。图 9 中蓝色区域为两结果交集的区域，黄色为算法预测候选框，绿色为真实的标注信息 GT。左边的交并比为 0.64，大于 0.5，因此是一个正确的预测结果；而图 9 右边的预测交并比为 0.37，小于 0.5。所以右边的检测结果是假阳性（false positive）结果。

(1) 准确率 (precision):

准确率的计算为所有正确的检测框 (True Positive) 比上所有算法预测出来的结果：

$$\text{precision} = TP / (TP + FP)$$

(2) 召回率 (recall):

召回率的计算为所有正确的检测框 (True Positive) 比上所有应该预测的正确结果：

$$\text{recall} = TP / (TP + TN)$$

(3) F-measure:

F-measure 是准确率与召回率的均衡值，F-measure 高表明一个算法对准确率和召回率的

兼顾较为到位。

$$F\text{-measure} = 2 * \text{precision} * \text{recall} / (\text{precision} + \text{recall})$$

补充：某一些数据集在评测时会根据检测结果自动将同一行上的多个检测框合并成为单独一行作为最后评比结果（如 ICDAR2013, ICDAR2015 和 ICDAR2017），这样结果会更为自由。

### 2.3.1 图像检测算法的发展

(1) 几种基本操作：

1) 卷积层操作：

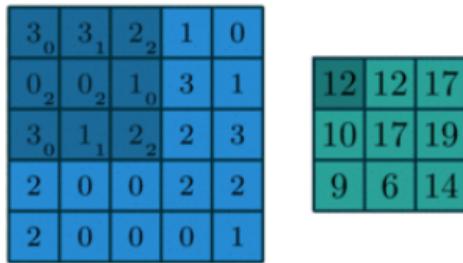


图 2-10 卷积操作示意图

已知：input size  $i$ , kernel size  $k$ , stride  $s$ , padding size  $p$ ,  
则输出的特征图大小为：

$$o = \left\lceil \frac{i + 2p - k}{s} \right\rceil + 1.$$

2) ReLU 非线性激活函数：

ReLU 的区分主要在负数端，根据负数端斜率的不同来进行区分，大致如下图所示：

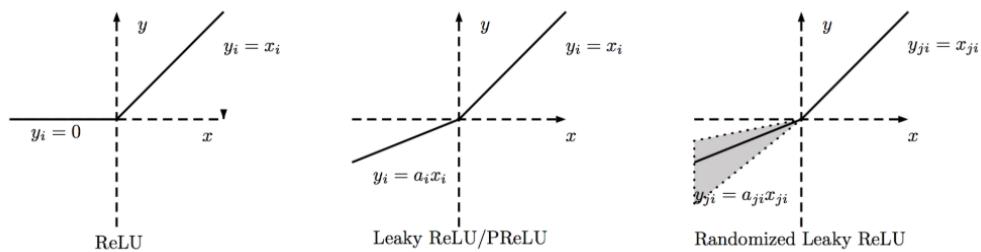


图 2-11 ReLU 图像

普通的 ReLU 负数端斜率是 0, Leaky ReLU 则是负数端有一个比较小的斜率, 而 PReLU 则是在后向传播中学习到斜率。而 Randomized Leaky ReLU 则是使用一个均匀分布在训练的时候随机生成斜率，在测试的时候使用均值斜率来计算。

3) 全连接层：

全连接层 (fully connected layers, FC) 在整个卷积神经网络中起到“分类器”的作用。如果说卷积层、池化层和激活函数层等操作是将原始数据映射到隐层特征空间的话，全连接层则起到将学到的“分布式特征表示”映射到样本标记空间的作用。在实际使用中，全连接层可由卷积操作实现：对前层是全连接的全连接层可以转化为卷积核为  $1 \times 1$  的卷积；而前层是卷积层的全连接层可以转化为卷积核为  $h \times w$  的全局卷积， $h$  和  $w$  分别为前层卷积结果的高和宽。

(1) 深度识别模型

从 1989 年 LeCun 提出第一个真正意义上的卷积神经网络到今天为止，它已经走过了 29 个年头。自 2012 年 AlexNet 网络出现之后，最近 6 年以来，卷积神经网络得到了急速发展，在很多问题上取得了当前最好的结果，是各种深度学习技术中用途最广泛的一种。在本文中将为大家回顾和总结卷积神经网络的整个发展过程。

### 1) Lenet

Lenet[13] 用于邮政编码的识别，在 9% 拒识率的条件下错误率为 1%。网络的输入为 28x28 的图像，输出为 0-9 这 10 个类。整个网络有 4 个隐含层，其中 H1 为 4 个 5x5 的卷积核，输出为 4 张 24x24 的特征图像。H2 为下采样层，对 H1 的输出结果进行 2x2 的下采样，得到 4 张 12x12 的图像。H3 有 12 个 5x5 的卷积核，输出为 12 张 8x8 的图像，这里输出图像每个通道的多通道卷积只作用于前一层输出图像的部分通道上，为什么采用这样方式？有两个原因：1.减少参数，2.这种不对称的组合连接的方式有利于提取多种组合特征。H2 和 H3 的连接关系如下图所示：

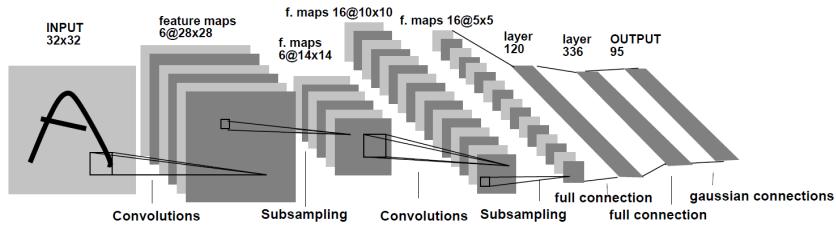


图 2-12 Lenet 网络架构图

### 2) AlexNet

现代意义上的深度卷积神经网络起源于 AlexNet 网络[14]，它是深度卷积神经网络的鼻祖。这个网络相比之前的卷积网络最显著的特点是层次加深，参数规模变大。网络结构如下图所示：

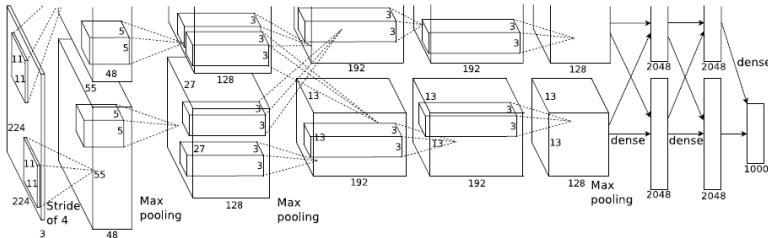


图 2-13 Alexnet 网络框架图

这个网络有 5 个卷积层，它们中的一部分后面接着 max-pooling 层进行下采样；最后跟 3 个全连接层。最后一层是 softmax 输出层，共有 1000 个节点，对应 ImageNet 图集中 1000 个图像分类。网络中部分卷基层分成 2 个 group 进行独立计算，有利于 GPU 并行化以及降低计算量。

这个网络有两个主要的创新点：1. 新的激活函数 ReLU，2. dropout 机制。dropout 的做法是在训练时随机的选择一部分神经元进行休眠，另外一些神经元参与网络的优化，起到了正则化的作用以减轻过拟合。

网络的输入图像为的彩色三通道图像。第 1 个卷积层有 96 组 11x11 大小的卷积核，卷积操作的步长为 4。这里的卷积核不是 2 维而是 3 维的，每个通道对应有 3 个卷积核（所以是一组卷积核），具体实现时是用 3 个 2 维的卷积核分别作用在 RGB 通道上，然后将三张结果图像相加。

### 3) GoogLeNet

文献[15]提出了一种称为 GoogLeNet 网络的结构( Inception-V1 )。在 AlexNet 出现之后，

针对图像类任务出现了大量改进的网络结构，总体来说改进的思路主要是增大网络的规模，包括深度和宽度。但是直接增加网络的规模将面临两个问题，首先，网络参数增加之后更容易出现过拟合，在训练样本有限的情况下这一问题更为突出。另一个问题是计算量的增加。GoogLeNet 致力于解决上面两个问题。

GoogLeNet 由 Google 在 2014 年提出，其主要创新是 Inception 机制，即对图像进行多尺度处理。这种机制带来的一个好处是大幅度减少了模型的参数数量，其做法是将多个不同尺度的卷积核，池化层进行整合，形成一个 Inception 模块。典型的 Inception 模块结构如下图所示：

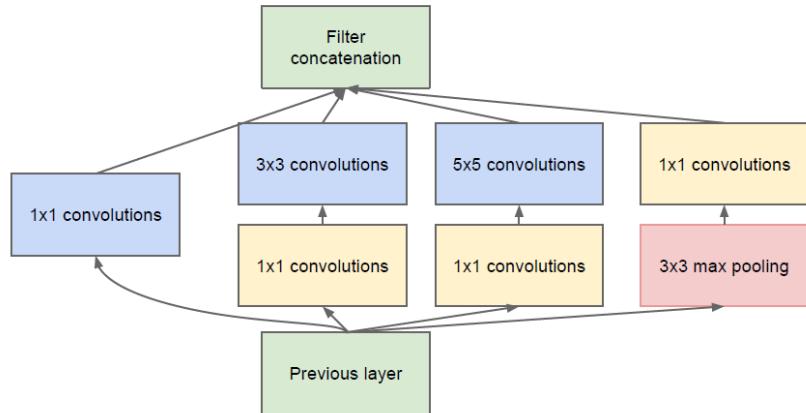


图 2-14 Inception 模块示意图

上图的模块由 3 组卷积核以及一个池化单元组成，它们共同接受来自前一层的输入图像，有三种尺寸的卷积核，以及一个 max pooling 操作，它们并行的对输入图像进行处理，然后将输出结果按照通道拼接起来。因为卷积操作接受的输入图像大小相等，而且卷积进行了 padding 操作，因此输出图像的大小也相同，可以直接按照通道进行拼接。

从理论上看，Inception 模块的目标是用尺寸更小的矩阵来替代大尺寸的稀疏矩阵。即用一系列小的卷积核来替代大的卷积核，而保证二者有近似的性能。

上图的卷积操作中，如果输入图像的通道数太多，则运算量太大，而且卷积核的参数太多，因此有必要进行数据降维。所有的卷积和池化操作都使用了 1x1 卷积进行降维，即降低图像的通道数。因为 1x1 卷积不会改变图像的高度和宽度，只会改变通道数。

GoogleNet 网络结构如下图所示：

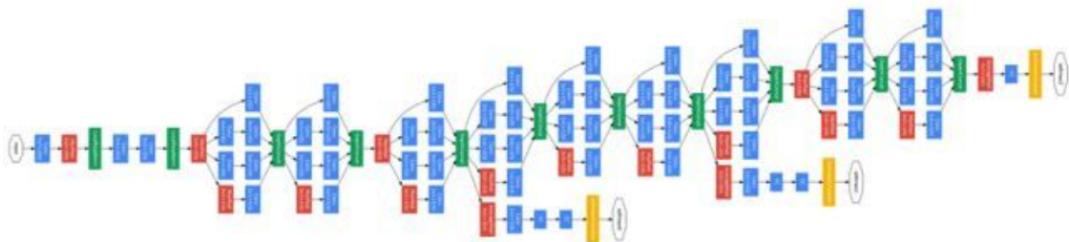


图 2-15 GoogleNet

#### 4) VGG 网络

VGG 网络[16]由著名的牛津大学视觉组(Visual Geometry Group)2014 年提出[14]，并取得了 ILSVRC 2014 比赛分类任务的第 2 名(GoogleNet 第一名)和定位任务的第 1 名。同时 VGGNet 的拓展性很强，迁移到其他图片数据上的泛化性非常好。VGGNet 的结构非常简洁，整个网络都使用了同样大小的卷积核尺寸(3x3)和池化尺寸(2x2)。

到目前为止，VGGNet 依然经常被用来提取图像特征，被广泛应用于视觉领域的各类任务。

VGG 网络的主要创新是采用了小尺寸的卷积核。所有卷积层都使用  $3 \times 3$  卷积核，并且卷积的步长为 1。为了保证卷积后的图像大小不变，对图像进行了填充，四周各填充 1 个像素。所有池化层都采用  $2 \times 2$  的核，步长为 2。全连接层有 3 层，分别包括 4096, 4096, 1000 个节点。除了最后一个全连接层之外，所有层都采用了 ReLU 激活函数。下图为 VGG16 结构图：

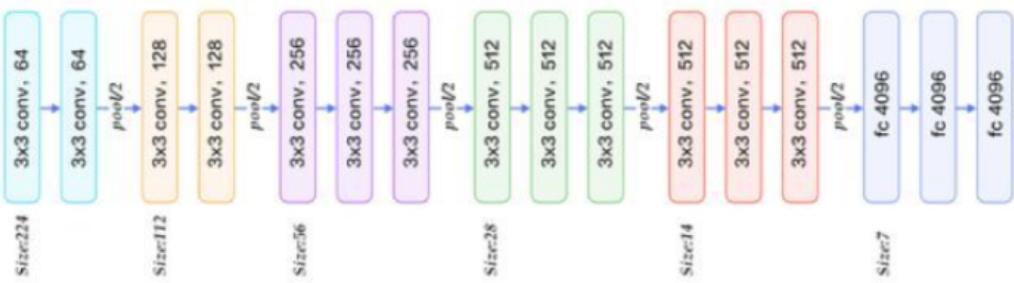


图 2-16 VGG16 网络框架图

VGG 与 Alexnet 相比，做了以下改进：

- a) 去掉了 LRN 层，作者实验中发现深度卷积网络中 LRN 的作用并不明显
- b) 采用更小的连续  $3 \times 3$  卷积核来模拟更大尺寸的卷积核，例如 2 层连续的  $3 \times 3$  卷积层可以达到一层  $5 \times 5$  卷积层的感受野，但是所需的参数量会更少，两个  $3 \times 3$  卷积核有 18 个参数（不考虑偏置项），而一个  $5 \times 5$  卷积核有 25 个参数。后续的残差网络等都延续了这一特点。

### 5) 残差网络

残差网络(Residual Network)[17]用跨层连接(Shortcut Connections)拟合残差项(Residual Representations)的手段来解决深层网络难以训练的问题，将网络的层数推广到了前所未有的规模，作者在 ImageNet 数据集上使用了一个 152 层的残差网络，深度是 VGG 网络的 8 倍但复杂度却更低，在 ImageNet 测试集上达到 3.57% 的 top-5 错误率，这个结果赢得了 ILSVRC2015 分类任务的第一名，另外作者还在 CIFAR-10 数据集上对 100 层和 1000 层的残差网络进行了分析。VGG19 网络和 ResNet34-plain 及 ResNet34-redidual 网络对比如下：

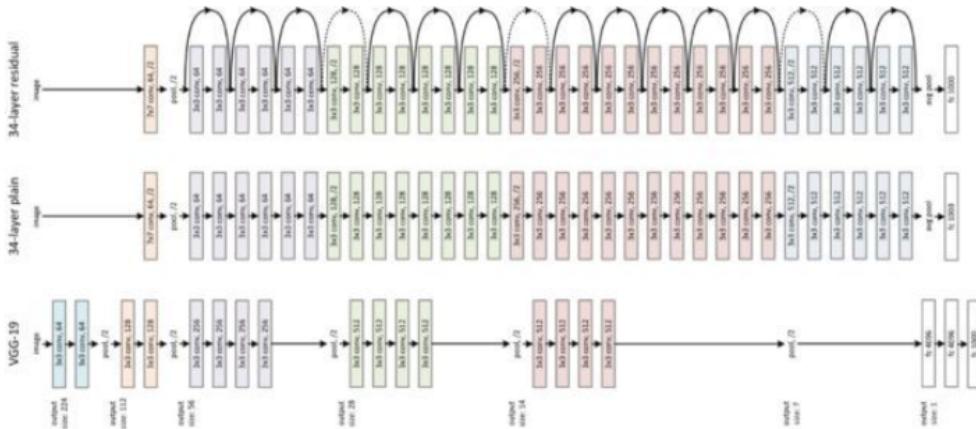


图 2-17 跳层连接与普通链接结构对比

之前的经验已经证明，增加网络的层数会提高网络的性能，但增加到一定程度之后，随着层次的增加，神经网络的训练误差和测试误差会增大，这和过拟合还不一样，过拟合只是

在测试集上的误差大，这个问题称为退化。

为了解决这个问题，作者设计了一种称为深度残差网络的结构，这种网络通过跳层连接和拟合残差来解决层次过多带来的问题，这种做法借鉴了高速公路网络(Highway Networks)的设计思想，与LSTM有异曲同工之妙。这一结构的原理如下图所示：

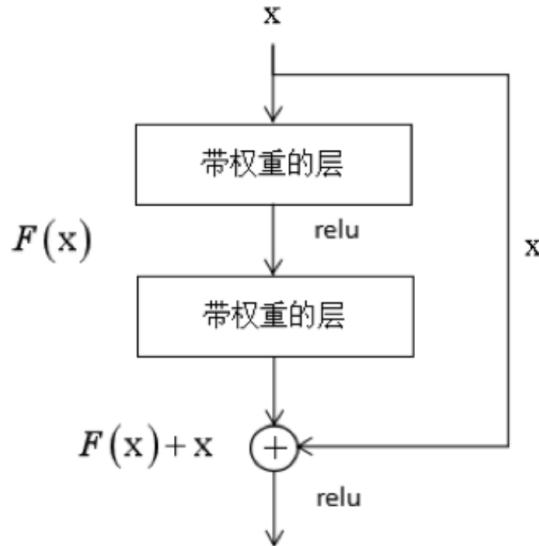


图 2-18 残差连接

后面有文献对残差网络的机制进行了分析。得出了以下结论：残差网络并不是一个单一的超深网络，而是多个网络指数级的隐式集成，由此引入了多样性的概念，它用来描述隐式集成的网络的数量；在预测时，残差网络的行为类似于集成学习；对训练时的梯度流向进行了分析，发现隐式集成大多由一些相对浅层的网络组成，因此，残差网络并不能解决梯度消失问题。

为了进一步证明残差网络的这种集成特性，并确定删除掉一部分跨层结构对网络精度的影响，作者进行了删除层的实验，在这里有两组实验，第一组是删除单个层，第二组是同时删除多个层。为了进行比较，作者使用了残差网络和VGG网络。实验结果证明，除了个别的层之外，删掉单个层对残差网络的精度影响非常小。相比之下，删掉VGG网络的单个层会导致精度的急剧下降。这个结果验证了残差网络是多个网络的集成这一结论。

第三组实验是对网络的结构进行变动，集调整层的顺序。在实验中，作者打乱某些层的顺序，这样会影响一部分路径。具体做法是，随机的交换多对层的位置，这些层接受的输入和产生的输出数据尺寸相同。同样的，随着调整的层的数量增加，错误率也平滑的上升，这和第二组实验的结果一致。

## (2) 深度模型检测算法

### 1) RCNN

RCNN[18]首先使用selective search算法[19]，从图片中提取出2000个可能包含有目标的区域，再将这2000个候选区(ROI: region of interest)压缩到统一大小(227\*227)送入卷积神经网络中进行特征提取，在最后一层将特征向量输入svm分类器，得到该候选区域的种类。整体上看R-CNN比较简单，与此同时也有两个重大缺陷：

- 1) selective search进行候选区域提取的过程在cpu内计算完成，占用了大量计算时间。
- 2) 对2000个候选框进行卷积计算，提取特征的时候，存在大量的重复计算，进一步增加了计算复杂度。针对以上两个缺点，R Girshick分别在Fast-RCNN和Faster-RCNN中进行了改进。

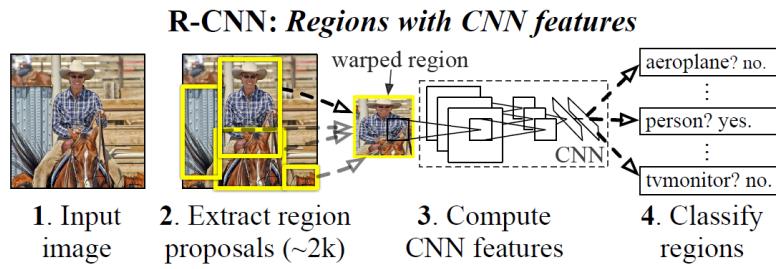


图 2-19 RCNN 检测网络结构图

## 2) Fast-RCNN [21]

### a) SPP-Net [20]

由于 Fast-RCNN 借鉴了 SPP-Net 的思想，所以先来了解一下 SPP-Net。

在 RCNN 种需要对 2000 个候选框进行卷积特征计算，而这 2000 个候选框是来自与同一张图片的，所以，作者考虑到先对整张图片进行一次卷积计算，得到整张图片的卷积特征，然后依据每个候选框在原始图片中的位置，在卷积特征图中取出对应的区域的卷积特征。再将卷积图中的到的特征向量送入分类器，在这里产生了一个问题，就是每个候选框的大小是不一样的，得到的卷积特征的维度也会不一样，无法送入全连接层，导致分类无法进行，为了将所有候选框的特征维度统一起来，作者就设计了 SPP-Net：

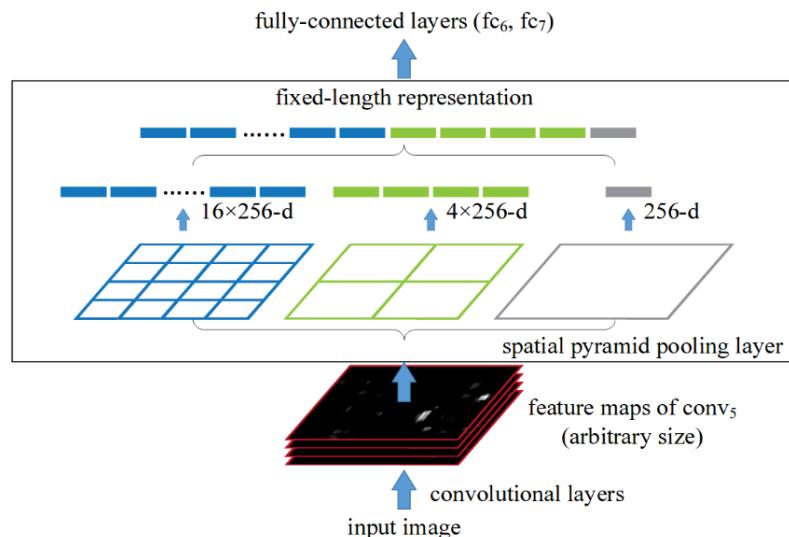


图 2-20 SPP-Net 网络结构图

### b) ROI pooling layer

在 Fast-RCNN 中作者采用了 SPP-Net 的简化版：只对 SPP-Net 进行了一种尺度的切分，之后直接下采样，得到特征向量。

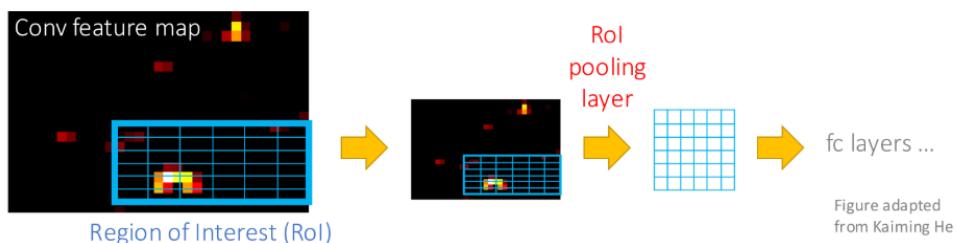


图 2-21 ROI Pooling 操作图

### c) Fast-RCNN 整体框架

在 rcnn 中进行卷积特征提取的时候，需要对图片中的 2000 个候选框进行卷积计算，其中很多计算是重复的，同时 spp-net 和 rcnn 都需要多阶段的训练包括特征提取、微调网络、训练 svm 分类器、边框回归等，不仅过程繁杂而且中间会产生大量的中间结果文件，占用大量内存。为此作者除了采用 roi-pooling layer 以外还设计了多任务损失函数(multi-task loss)，将分类任务和边框回归统一到了一个框架之内，整体思路如下：

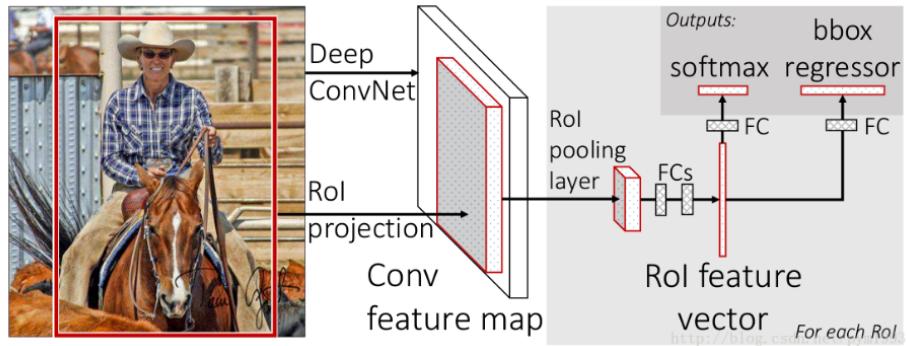


图 2-22 Fast-RCNN 整体框架图

用 selective search 方法从原始图片中提取 2000 个候选框 (ROI)，对整张图片进行卷积计算，得到卷积特征图 (conv feature map) ,然后利用 ROI pooling layer 从卷积特征图种提取每个候选框的特征向量，通过全连接层之后，特征向量进入两个输出层：一个进行分类，判断该候选框内的物体种类，另一个进行边框回归，判断目标在图中的准确位置。

Fast-RCNN 缺陷在于仍然没有解决 selective search 进行候选框选择的时候计算速度慢的问题。

### 3) Faster-RCNN[1]

针对 selective search 在 cpu 内进行计算速度慢等问题，作者创建了 RPN 网络替代 selective search 算法进行候选框选择，使得整个目标识别真正实现了端到端的计算，将所有的任务都统一在了深度学习的框架之下，所有计算都在 GPU 内进行，使得计算的速度和精度都有了大幅度提升。

#### a) RPN 网络

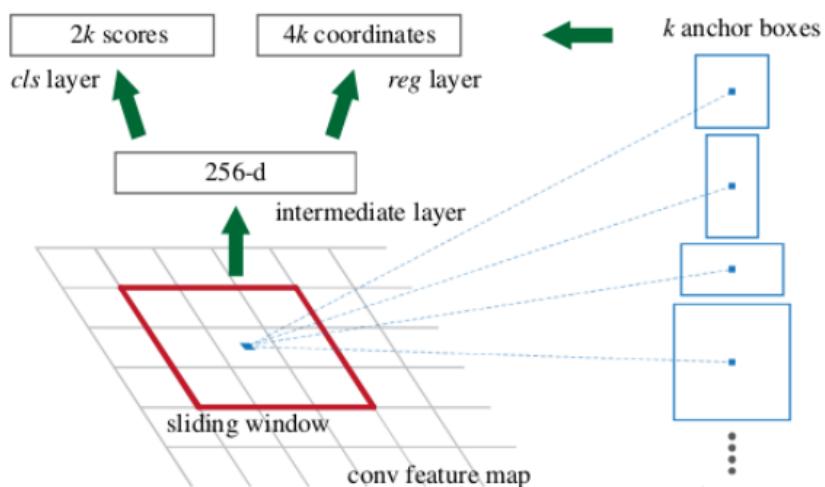


图 2-23 RPN 网络结构

RPN 网络的全称 region proposal network, 目的是利用神经网络进行候选框的选择，其实 RPN 也可以看做是一个分类网络，不过他的目标是分开前景（包含有 ROI 的部分）和背景（包含有 ROI 的部分），也就是一个二分类问题。

为了提取候选框，作者使用了一个小的神经网络也即就是一个  $n \times n$  的卷积核(文中采用了  $3 \times 3$  的网络)，在经过一系列卷积计算的特征图上进行滑移，进行卷积计算。每一个滑窗计算之后得到一个低维向量（例如 VGG net 最终有 512 张卷积特征图，每个滑窗进行卷积计算的时候可以得到 512 维的低维向量），得到的特征向量，送入两种层：一种是边框回归层进行定位，另一种是分类层判断该区域是前景还是背景。 $3 \times 3$  滑窗对应的每个特征区域同时预测输入图像 3 种尺度 ( $128, 256, 512$ )，3 种长宽比 ( $1:1, 1:2, 2:1$ ) 的 region proposal，这种映射的机制称为 anchor。所以对于  $40 \times 60$  图图，总共有约  $20000(40 \times 60 \times 9)$  个 anchor，也就是预测 20000 个 region proposal。

### b) Faster-RCNN 整体思路

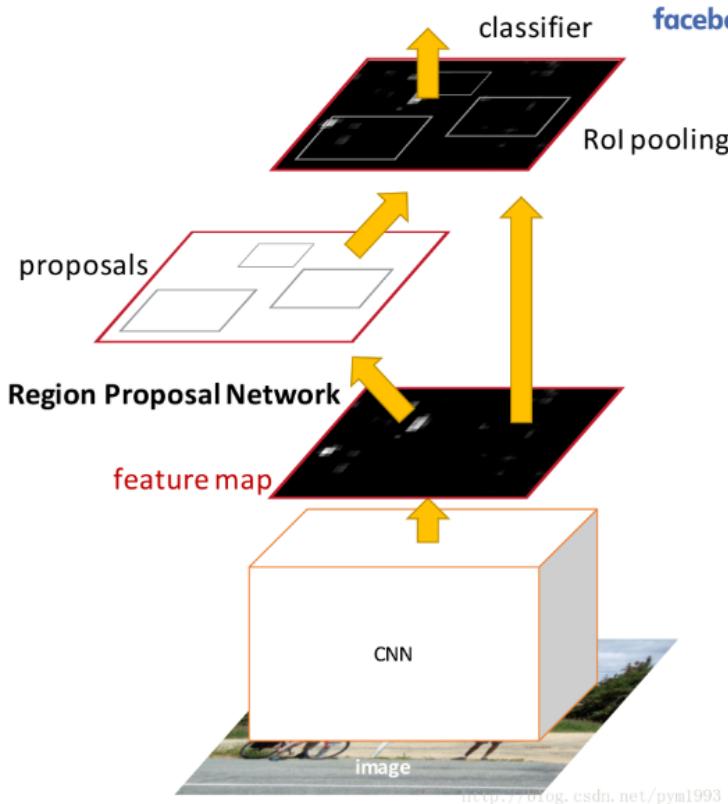


图 2-24 Faster-RCNN 整体架构图

首先对整张图片进行卷积计算，得到卷积特征，然后利用 RPN 进行候选框选择，再返回卷积特征图取出候选框内的卷积特征利用 ROI 提取特征向量最终送入全连接层进行精确定位和分类，总之：RPN+Fast-RCNN=Faster-RCNN。

### 4) YOLO[22]

尽管 faster-rcnn 在计算速度方面已经取得了很大进展，但是仍然无法满足实时检测的要求，因此有人提出力基于回归的方法直接从图片种回归的出目标物体的位置以及种类。具有代表性的两种方法是 YOLO 和 SSD。

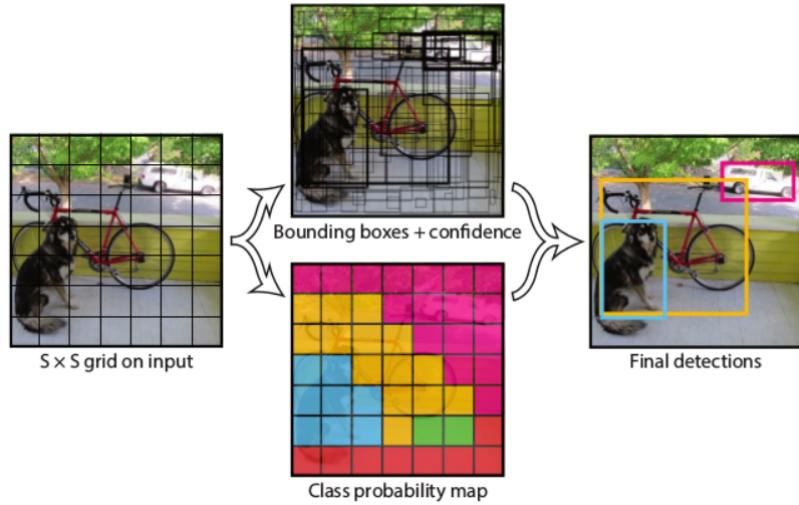


图 2-25 YOLO 整体方法

给一个输入图像，首先将图像划分成  $7 \times 7$  的网格。其次，对于每个网格，我们都预测 2 个边框（包括每个边框是目标的置信度以及每个边框区域在多个类别上的概率）。然后，根据上一步可以预测出  $7 \times 7 \times 2$  个目标窗口，然后根据阈值去除可能性比较低的目标窗口，最后非极大值抑制去除冗余窗口即可。可以看到整个过程非常简单，不需要中间的 region proposal 在找目标，直接回归便完成了位置和类别的判定。

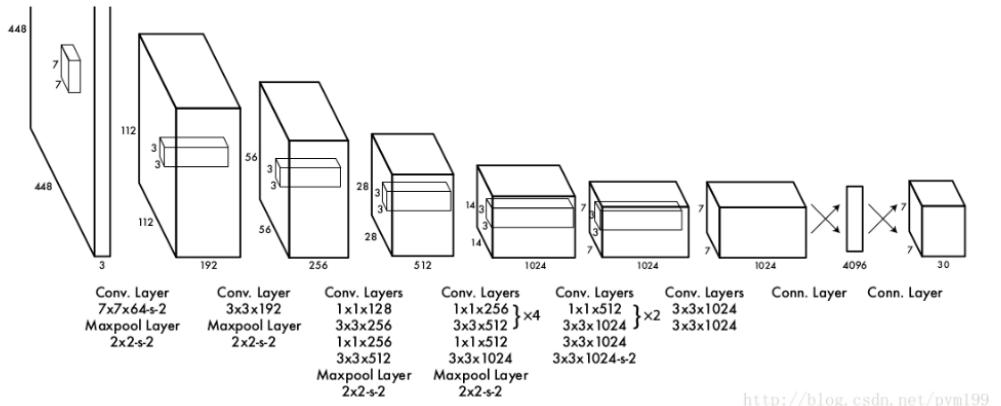


图 2-26 YOLO 网络结构

YOLO 的网络结构图，前边的网络结构跟 GoogLeNet 的模型比较类似，主要的是最后两层的结构，卷积层之后接了一个 4096 维的全连接层，然后后边又全连接到一个  $7 \times 7 \times 30$  维的张量上。实际上这  $7 \times 7$  就是划分的网格数，现在要在每个网格上预测目标两个可能的位置以及这个位置的目标置信度和类别，也就是每个网格预测两个目标，每个目标的信息有 4 维坐标信息(中心点坐标+长宽)，1 个是目标的置信度，还有类别数 20(VOC 上 20 个类别)，总共就是  $(4+1) \times 2 + 20 = 30$  维的向量。这样可以利用前边 4096 维的全图特征直接在每个网格上回归出目标检测需要的信息（边框信息加类别）。

Yolo 方法的缺点显而易见，虽然舍弃了 Region proposal 阶段，加快了速度，但是定位精度比较低，与此同时带来的问题是，分类的精度也比较低。

### 5) SSD[23]

鉴于 yolo 定位精度低的缺陷，SSD 结合 faster-rcnn 的 anchor 机制和 yolo 的回归思想进行目标检测，使得定位精度和分类精度相较与 yolo 都有了大幅度的提高。

#### a) The Single Shot Detector

SSD: Single Shot MultiBox Detector

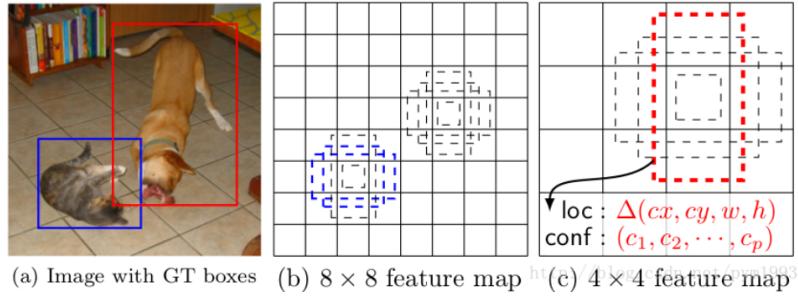


图 2-27 SSD 方法图

上图是 SSD 的一个框架图，首先 SSD 获取目标位置和类别的方法跟 YOLO 一样，都是使用回归，但是 YOLO 预测某个位置使用的是全图的特征，SSD 预测某个位置使用的是这个位置周围的特征（感觉更合理一些）。那么如何建立某个位置和其特征的对应关系呢？可能你已经想到了，使用 Faster R-CNN 的 anchor 机制。如 SSD 的框架图所示，假如某一层特征图大小是 8\*8，那么就使用 3\*3 的滑窗提取每个位置的特征，然后这个特征回归得到目标的坐标信息和类别信息(图 c)。不同于 Faster R-CNN，这个 anchor 是在多个 feature map 上，这样可以利用多层的特征并且自然的达到多尺度（不同层的 feature map 3\*3 滑窗感受野不同）。

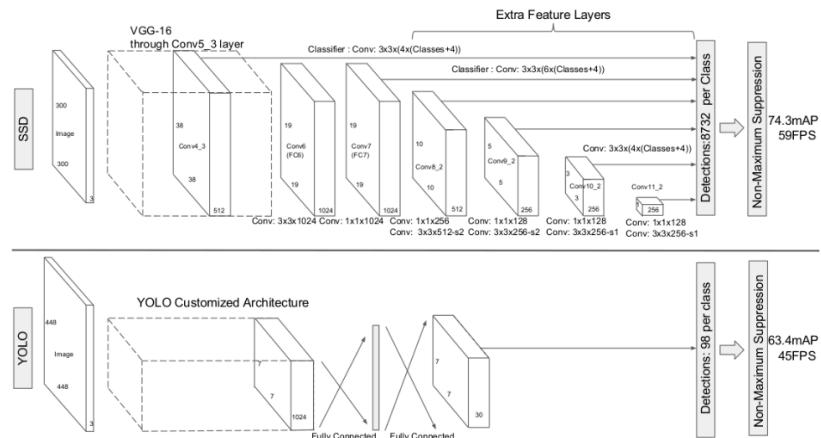


图 2-28 SSD 整体框架

首先第一步和其他方法一样利用卷积操作提取卷积特征，在最后级层卷积时候开始对与每一种尺度上的特征图运用 anchor 方法进行候选框提取，依据 anchor 在不同尺度上得到的候选框，进行目标种类和位置的判断。

## 2.4 主流算法介绍

### 2.4.1 CTPN

文章标题为 Detecting Text in Natural Image with Connectionist Text Proposal Network[3]，发表在 **ECCV 2016**（深圳先进研究院），框架图如下：

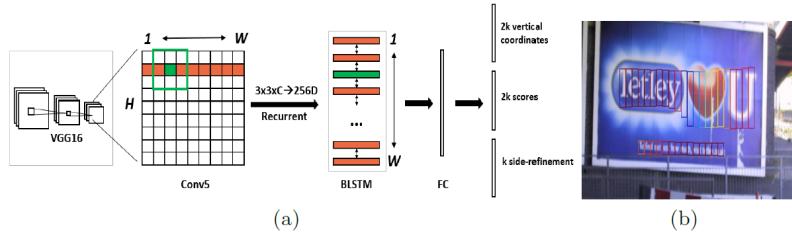


图 2-29 CTPN 框架图

本文主要改进了传统通用物体检测方法 Faster-RCNN[1]，并增加检测算法在文字检测上的鲁棒性。文章主要的网络架构为 RPN(Region Proposal Networks) + BLSTM，在 VGG16 的 conv5\_3 卷积特征上以  $3 \times 3$  的窗口进行滑窗操作并转换成 256 维特征为单位的序列，送入随后的 BLSTM 网络，最后通过一层全连接层输出三组预测结果： $2k$  个 vertical coordinate，因为一个 anchor 用的是中心位置的高（y 坐标）和矩形框的高度两个值表示的，所以一个用  $2k$  个输出。（注意这里输出的是相对 anchor 的偏移）。 $2k$  个 score，因为预测了  $k$  个 text proposal，所以有  $2k$  个分数，text 和 non-text 各有一个分数。 $k$  个 side-refinement，这部分主要是用来精修文本行的两个端点的，表示的是每个 proposal 的水平平移量。

#### 2.4.2 DMPnet

文章标题为 Deep Matching Prior Network: Toward Tighter Multi-oriented Text Detection[9]，发表于 CVPR 2017（华南理工大学），框架图如下：

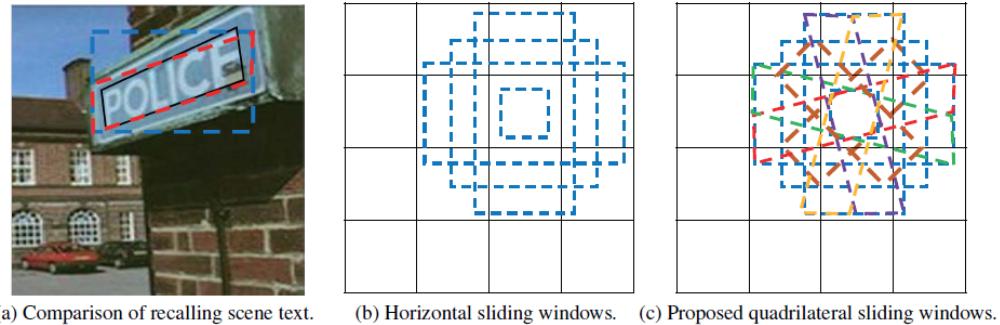


图 2-30 DMPnet 框架图

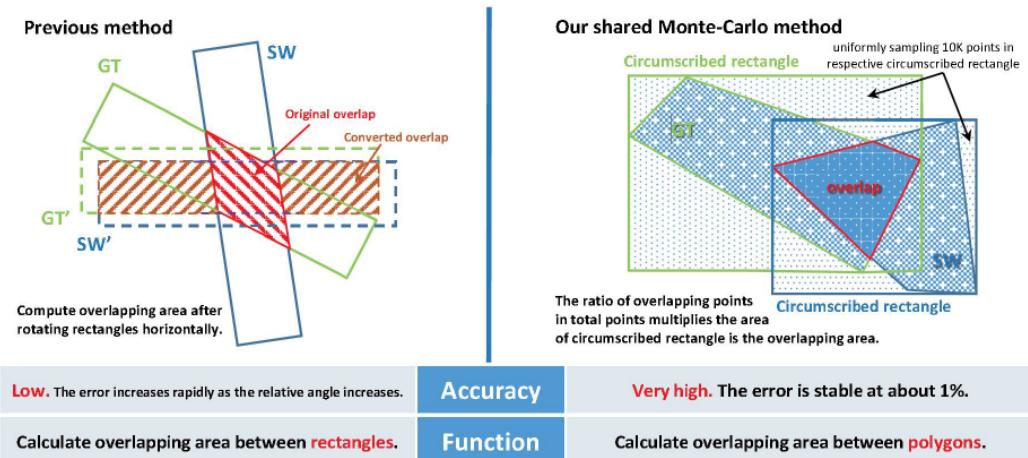


图 2-31 蒙特卡洛求不规则矩形 IoU

该检测算法基于 SSD[11] (Single Shot Detector)，不仅增加了多角度的 anchor (锚点)。并且增加了一些不规则矩形(平行四边形)锚点用以预测更为紧致的字符检测框。除此之外，

提出用蒙特卡洛方式求不规则矩形之间的 IoU。

### 2.4.3 Seglink

Seglink 算法，对应文章标题为 Detecting Oriented Text in Natural Images by Linking Segments[8]，发表于会议 CVPR 2017（华中科技大学 白翔）。

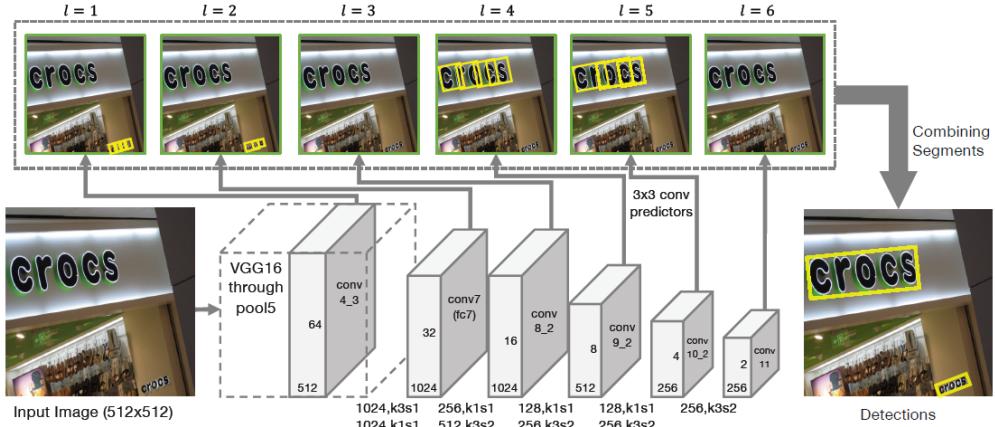


图 2-32 Seglink 框架图

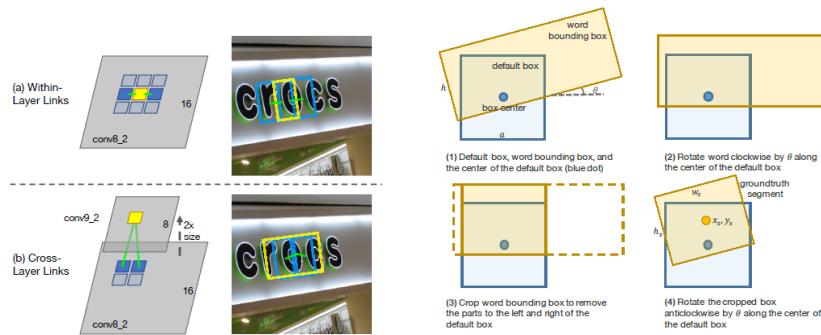


图 2-33 Seglink 跨层链接示意图

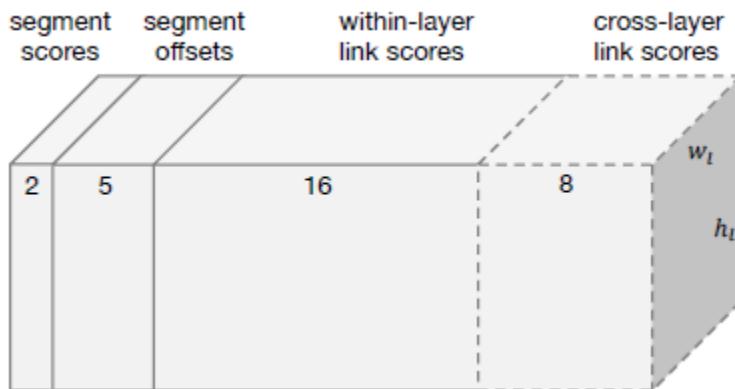


图 2-34 预测部分各个维度代表的含义

文章基于 SSD 方法并增加多朝向的 anchor 用以预测任意朝向的字符区域。在此之上，根据 SSD 算法能够在不同 anchor 层上产生预测候选框，因此在相同区域，同时会有多个层产生同区域的候选框，提出了跨层连接和同层链接两种方式用来链接字符区域的所有候选框，组成完整的字符区域。

#### 2.4.4 Textboxes & Textboxes++

算法 TextBoxes: A Fast Text Detector with a Single Deep Neural Network[4], 发表于 **AAAI 2017** (华中科技大学 白翔)。

算法 TextBoxes++ : A Single-Shot Oriented Scene Text Detector[5] , 发表于 arXiv:1801.02765v3 (华中科技大学 白翔)。

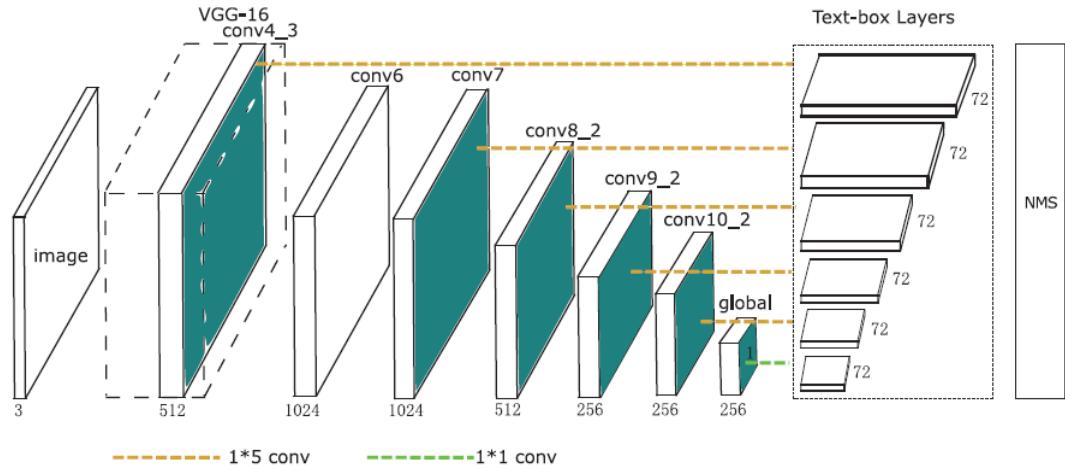


图 2-35 TextBoxes 框架图

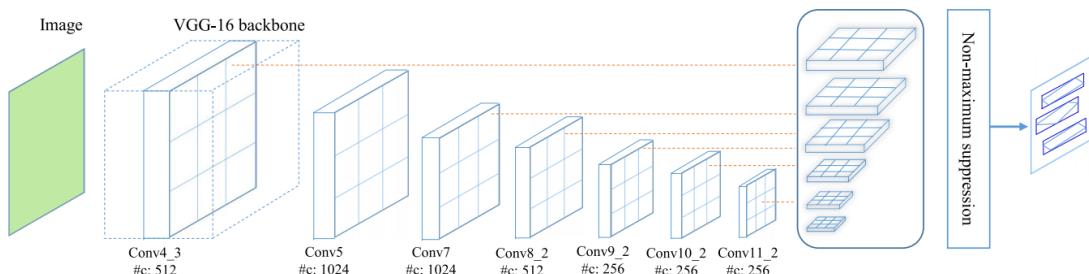


图 2-36 TextBoxes++框架图

以 SSD 为框架, 修改了 Anchor 的长宽比以更好地适应长字符检测。每一层 Anchor 层每个特征位置上预测 72-d 向量, 12 anchors \* (4 coordinates (Textboxes++里则为 5 或 8, 带角度或是 8 个坐标点) + 2 class scores)。

#### 2.4.5 EAST

算法 EAST: An Efficient and Accurate Scene Text Detector[12]发表于 **CVPR 2017** (旷世科技 姚聪)。

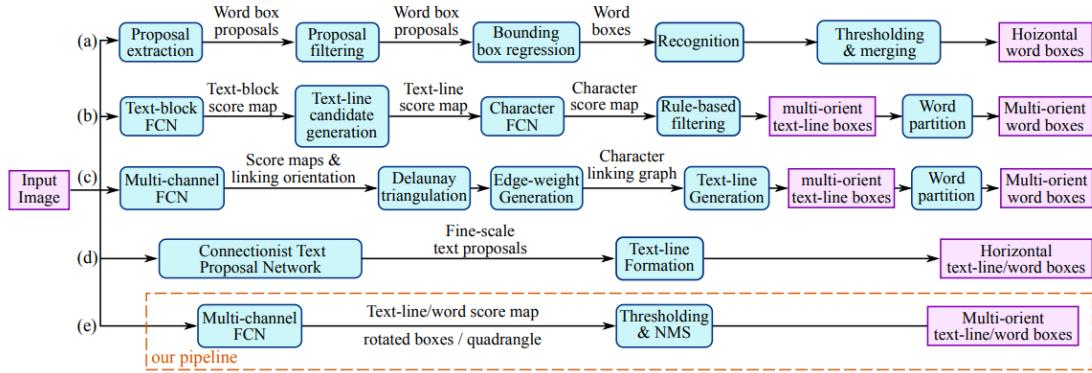


图 2-37 EAST 算法框架图

算法同时对字符区域进行分割与检测任务,检测部分提取自由度为 5 的旋转矩形或是自由度为 8 的任意四边形,并对每个字符区域预测一个分数图(score map),最后加入多朝向 NMS 步骤抑制冗余的候选框,最后产生任意朝向候选框的预测结果。

#### 2.4.6 FOTS

算法 FOTS: Fast Oriented Text Spotting with a Unified Network[6]发表于 CVPR 2018 (商汤科技&深圳先进研究院)。

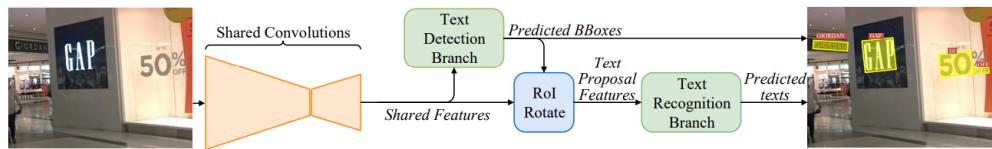


图 2-38 FOTS 算法框架图

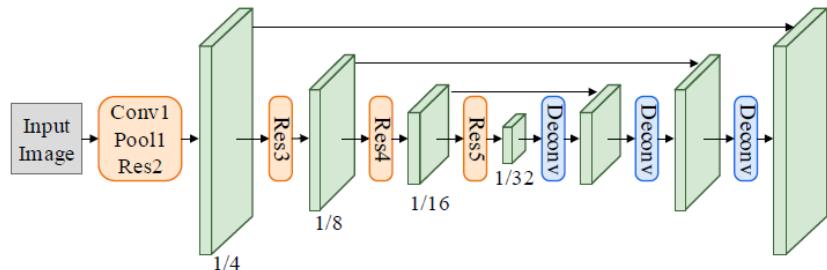


图 2-39 FOTS 检测部分多尺度特征

Type	Kernel [size, stride]	Out Channels
conv_bn_relu	[3, 1]	64
conv_bn_relu	[3, 1]	64
height-max-pool	[(2, 1), (2, 1)]	64
conv_bn_relu	[3, 1]	128
conv_bn_relu	[3, 1]	128
height-max-pool	[(2, 1), (2, 1)]	128
conv_bn_relu	[3, 1]	256
conv_bn_relu	[3, 1]	256
height-max-pool	[(2, 1), (2, 1)]	256
bi-directional_lstm		256
fully-connected		S

图 2-40 FOTS 识别部分网络架构参数

该算法为完整的自然场景字符识别框架。检测部分采用类似 FPN[10]+RRPN[2]的网络结构使得网络有能力预测带朝向的字符候选框。随后从原图 1/4 的特征图（Feature map）上截取候选框区域的特征（具体步骤为：先将特征图的字符区域转至水平，随后将相应区域截出，重采样为保证长宽比，且高为 8 的特征送入后端识别网络进行识别）。

#### 2.4.7 IncepText

算法 IncepText: A New Inception-Text Module with Deformable PSROI Pooling for Multi-Oriented Scene Text Detection[7] 发表于 IJCAI 2018 （阿里巴巴）。

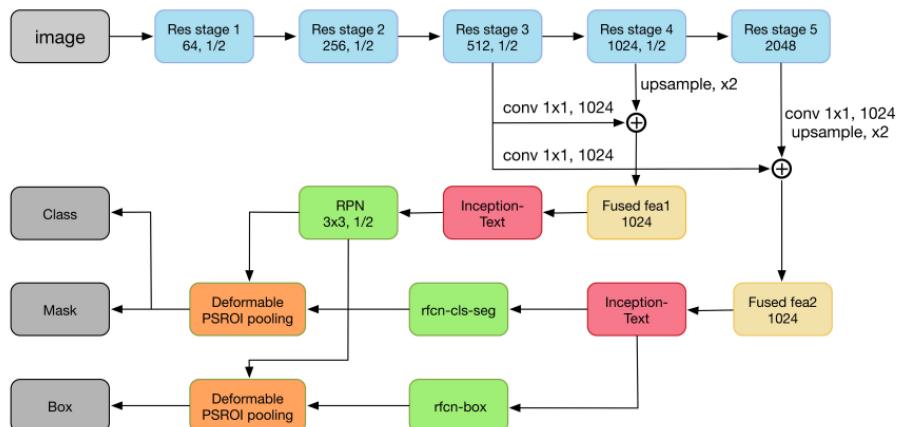


图 2-41 IncepText 算法网络框架图

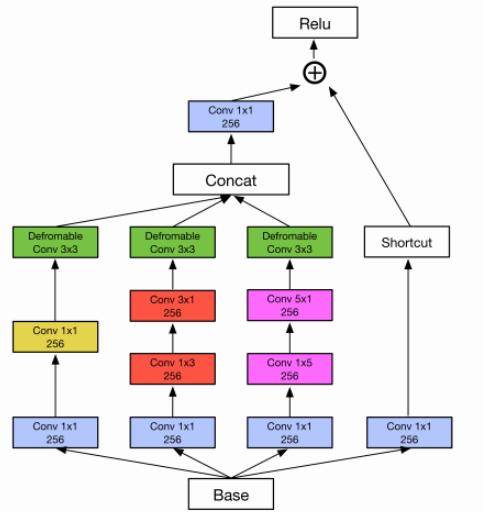


图 2-42 Inception 模块

本篇文章在网络特征选择部分提出 Inception 模块，并结合 Deformable PSROI pooling，即一种可以自动选择特征图上连续点位置，通过双线性插值（bilinear interpolation）的方式取得坐标位置上的特征的池化操作，组合成本篇文章的核心部分。图片输入 Inception-ResNet-V2 之后，形成两个网络分支，最后分别经过 Deformable PSROI pooling，其中一支产生字符区域的回归位置坐标，另一支则产生字符位置区域的分割轮廓结果。

## 2.5 实验室现有算法介绍

### 2.5.1 算法介绍

实验室现有算法为 Arbitrary-Oriented Scene Text Detection via Rotation Proposals。motivation 如下：自然场景中字符出现的并不只有水平和竖直方向，很有可能是任意朝向，因此直框无法准确的表达出字符的朝向，可能会使字符的阅读顺序丢失，导致识别算法失准或失效，从而使直框检测算法在多朝向任务上的实用度大打折扣。

采用检测网络+朝向信息，对现有检测网络进行扩展，使得检测网络能够适应多朝向的检测任务，并能够对自然场景中的字符阅读顺序（朝向）进行预测。

### 2.5.2 模型结构

#### (1) 架构图

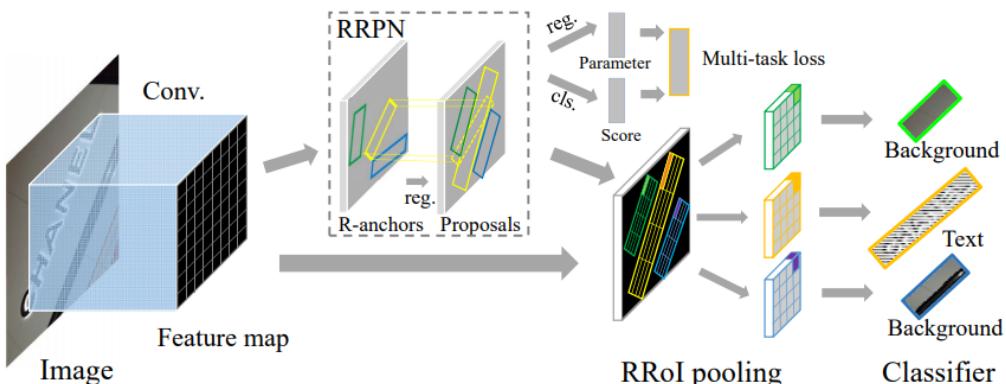


图 2-43 RRPN 算法框架图

#### (2) 方法细节 (Methodology)

方法基于 Faster-RCNN 框架，提出 RRPN 子网络，用于学习带角度的锚点（anchor）的分类（classification）与回归值（regression）。锚点则是在 Faster-RCNN 锚点设定的基础上，除了 Scale（尺寸大小）和 Ratio（长宽比例）之外，还增加了 Angle（初始角度），如下图所示：

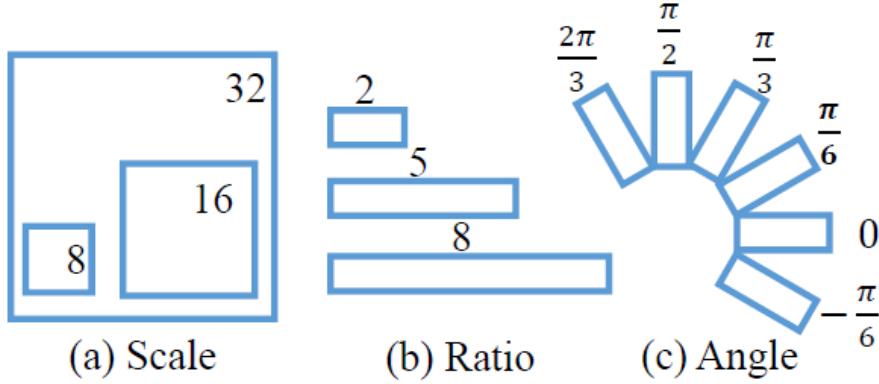


图 2-44 Roatake Anchor 对应三组参数

总共增加了锚点的 6 个初始化角度，均匀分布在 $[-45, 135]$ ，大小为 180 度的区间之内，因此锚点需要回归的角度被限制在 $[-15, 15]$ 度之间。这样就能够保证设定的锚点能够覆盖 $[-45, 135]$ 的区间。从而使得锚点经过微小的调整就能与 Ground Truth 对齐。

Faster-RCNN 后端部分因为需要将候选框对应的局部特征从全局特征图中截取，若采用原框架对应的感兴趣区域池化（RoI pooling）。由于对应的候选框并非水平或竖直方向，若保持取最小外接水平框的区域截取特征，则会带入太多背景信息，对最后的第二步分类与回归效果较差。示意图如下：

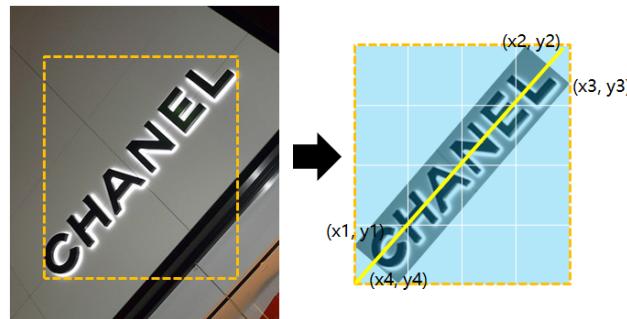


图 2-45 ROI Pooling 示意图

因此我们采取了一种对旋转矩形框更为友好的池化方式 --- RRoI Pooling，使得特征的截取能够按候选框的朝向进行，下图中展示了 RRoI Pooling 截取特征的过程：

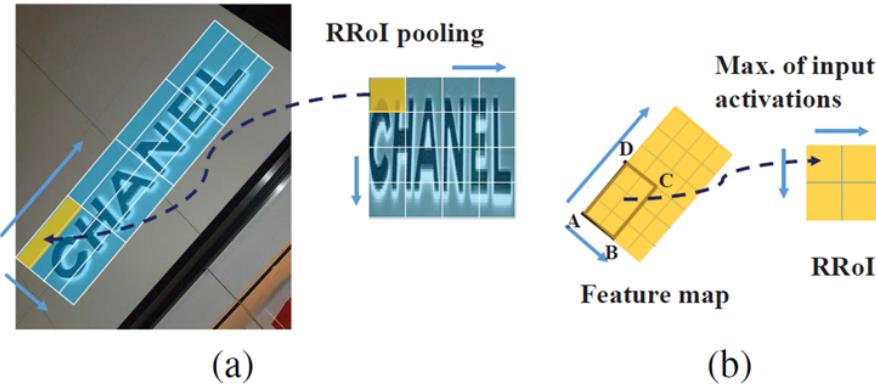


图 2-46 RRoI Pooling 示意图

RRoI Pooling 直接按候选框截取特征，随后经过最大值池化（max pooling）取得一定尺寸的特征，特征随后送入的 RCNN 部分进行第二步的分类与回归，从而取得更好的检测效果。

## 2.6 RPN 字符检测实验

### 2.6.1 数据集介绍

ICPR 官方提供了 20000 张图像作为本次比赛的数据集。其中 50% 用来作为训练集，50% 用来作为测试集。该数据集全部来源于网络图像，主要由合成图像，产品描述，网络广告构成，如下图所示：

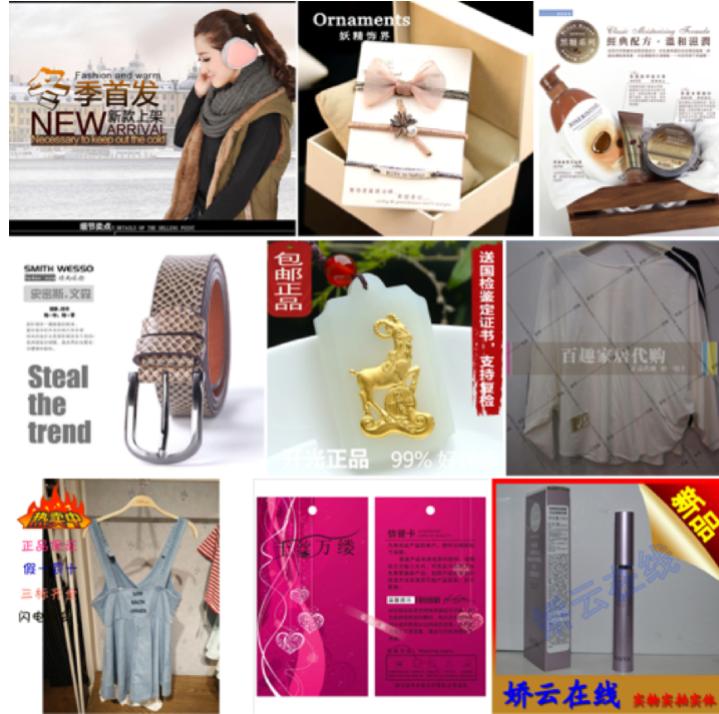


图 2-47 数据集样例

这些图像是网络上最常见的图像类型。每一张图像或者包含复杂排版，或者包含密集的小文本或多语言文本，或者包含水印，这对文本检测和识别均提出了挑战。

对于每一张图像，都会有一个相应的文本文件（.txt）（UTF-8 编码与名称：[图像文件

名] .txt)。文本文件是一个逗号分隔的文件，其中每行对应于图像中的一个文本串，并具有以下格式：

X1, Y1, X2, Y2, X3, Y3, X4, Y4, “文本”

其中 X1, Y1, Y2, X2, X3, X4, Y3, Y4 分别代表文本的外接四边形四个顶点坐标。而“文本”是四边形包含的实际文本内容。

## 2.6.2 评估标准

文本定位评测遵循 ICDAR2013 Born-Digital Image 的主体思路。本次竞赛数据集以中文为主，标注较细致，所以按照论文中“one to many”和“many to one”<sup>[1]</sup>的思路更为准确。其中一些阈值进行了调整：

第一，为“单个目标框”筛选合格“多个合格框”的阈值  $t_{many}$ 。“多”中的任意框与目标框交叉面积除以自身面积大于  $t_{many}=0.7$  时，视为合格候选。

第二，计算“多个合格框”覆盖“单目标框”的面积阈值  $t_{one}$ 。“多”中的所有框覆盖了目标框总面积大于  $t_{one}=0.7$  时，视单目标框可被召回或者属于正确检测范畴，视“多个合格框”为可被召回或者属于正确检测范畴。

第三，确定召回率和精度。计算“多”检测框对“单”标注框时，如果满足了  $t_{one}$ ，那么单标注框召回率为 1，多个检测框(个数为 k)的检测准确度为  $penal(K)$ 。计算“多”标注框对“单”检测框时，如果满足  $t_{one}$ ，那么单检测框精度为 1，多个标注框(个数为 k)中每一个召回率为  $penal(K)$ 。其中  $penal(K)$  为惩罚“分散”或者“合并”错误的函数，公式为：

$$penal(K) = 1 / (1 + \ln(K))$$

第四，处理“可忽略行”。对于行标注内容为“###”的文本行。“可忽略行”不计算召回率。当某个检测框被“可忽略行”覆盖的面积除以自身面积大于  $t_{ignore}=0.5$  时，视该检测框为“可忽略检测行”。可忽略的标注行和检测行不计入最终结果。

## 2.6.3 开发环境

- ◆ Python 环境：python2.7
- ◆ GPU 开发环境：Pascal 架构 Titan X 12G 显存 + nvidia 驱动程序 + GPU 开发环境 CUDA 8.0.61
- ◆ Deep learning 框架：Caffe

## 2.6.4 实验细节

验证集从训练集当中选取 1000 张图片组成。采用 RRPN 文章中基准方法的参数设置：

(1) baseline :

1) 训练：

    图片尺寸：短边 (800, 1000) 长边 1500

2) 训练轮数：

    100K for 1e-3

    100K for 1e-4

    20K for 1e-5

3) 验证集成绩：

    测试输出：短边 1000 长边 1700

    Precision: 0.747

    Recall: 0.337

    F-measure: 0.465 ---- baseline

测试输出: 短边 1200 长边 2000

Precision: 0.705

Recall: 0.375

F-measure: 0.490 + 2.5%

4) 结果展示:



图 2-48 实验 (1) 示例

(2) 更大的多尺度训练 1

1) 参数设置

训练输入: 短边长度 (800, 900, 1000, 1100, 1200) 长边 1600

测试图片输入: (1600, 1600)

训练轮数: 250K + 100K + 50K (三个阶段降低学习率)

框长宽比组合: 2:1, 5:1, 8:1 (同基准实验)

输出结果置信度阈值: 0.5

2) 实验结果:

Precision: 68.37%,

Recall: 47.87%,

f-measure: 56.31% + 9.81%

(3) 更大的多尺度训练 2

1) 参数设置

训练输入: 短边长度 (800, 900, 1000, 1100, 1200) 长边 1600

测试图片输入: (1600, 1600)

训练轮数: 250K + 100K + 50K (三个阶段降低学习率)

框长宽比组合: 1:1, 2:1, 5:1

输出结果置信度阈值: 0.5

2) 实验结果:

Precision: 65.75%,

Recall: 49.93%,

f-measure: 56.76% + 10.26%

(4) 更大的多尺度训练 3 (综合两种尺度)

1) 参数设置

训练输入: 短边长度 (800, 900, 1000, 1100, 1200) 长边 1600

测试图片输入: (1600, 1600)

训练轮数: (学习率 → 轮数)

$10^{-3} \rightarrow 100K$

$10^{-4} \rightarrow 100K$

$10^{-5} \rightarrow 40K$

输出结果置信度阈值: 0.5

框长宽比组合: 1:1, 2:1, 5:1, 8:1

## 2) 实验结果:

Precision: 0.651

Recall: 0.509

F-measure: 0.571 + 10.6%

增加更多数据训练 ICPR(8788) + RCTW(8034)

测试设置同 vi:

Precision: 0.710

Recall: 0.497

F-measure: 0.585 + 12.0%

降低分辨率至原来的一半，再扩大回原尺寸:

实验设置同 vi:

Precision: 0.69

Recall: 0.50

F-measure: 0.59 + 13%

## 2.6.5 两次提交结果成绩

(1) 第一次提交成绩采用实验 viii 的模型预测结果进行提交，成绩为:

F-measure: 0.677

Precision: 0.692

Recall: 0.663

成绩排名: 13 名

结果展示:



图 2-49 提交结果展示 1



图 2-50 提交结果展示 2

(2) 第二次结果提交方案: 多尺度预测结果融合 + 大分辨率对抗小物体检测问题

多尺度短边 (900, 1300, 1800), 最大长边为 1800, 选取第一次提交答案采用的模型作为基础模型。初始但模型非极大值抑制阈值为 IoU=0.3, 置信度阈值设置为 0.5。

900 为小尺寸, 1300 为中尺寸, 1800 为大尺寸。

小尺寸中选择候选框面积占全图 10%以上的候选框 (proposal) 作为预测结果, 中尺寸中面积占比在全图%1~%10 之间的候选框作为预测结果, 大尺寸则选取面积占比在 1%以下的。

### 1) 超大分辨率对抗小物体检测:

将图片均分成左上, 左下, 右上和右下四个部分, 每个部分以大分辨率送入网络产生预测结果, 参数设置与之前相同, 选取候选框面积比在 0.05%以下的候选框结果。

### 2) 结果融合:

将所有结果组合并进行 IoU 阈值为 0.1 的非极大值抑制, 取得最终结果。提交成绩如下:

F-measure: 0.655

Precision: 0.627

Recall: 0.685

排名 28

分析: 与第一次对比, 召回率有 2 个百分点的提升, 但是准确度下降较多, 因此 f-measure 总体下滑。原因是在大分辨率下, 较多小的错误 (不全) 检测框产生, 造成准确度的下降。效果展示:



(1) 红为实际检测, 黄为漏检部分 (2) Multi-Scale (3) Multi-Scale + Super resolution

图 2-51 Single-scale 效果展示

更多效果展示：



图 2-52 多种效果展示



图 2-53 缺陷：(漏检，错检)

## 2.7 总结

本章介绍了当下字符检测任务的定义以及当下主流的算法概况。由每年 ICDAR 官方推出的标准数据集来看，字符检测任务的难度也在逐年增加，从 ICDAR2013 的较为端正的字符区域，到往后的 ICDAR2015 和 ICDAR2017。任务的难度逐步转向透视变换更为严重的字符区域样本。与此同时，数据集规模也越来越大，对于模型的学习能力要求也越来越高。因此对经过透视变换的字符区域有更强的学习能力的模型，或是能够对不规则字符区域有更紧致预测结果的模型是当下热门的趋势。

## 参考文献

- [1] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems*. 2015.
- [2] Ma, Jianqi, et al. "Arbitrary-oriented scene text detection via rotation proposals." *IEEE Transactions on Multimedia* (2018).
- [3] Tian, Zhi, et al. "Detecting text in natural image with connectionist text proposal network." *European conference on computer vision*. Springer, Cham, 2016.
- [4] Liao, Minghui, et al. "TextBoxes: A Fast Text Detector with a Single Deep Neural Network." *AAAI*. 2017.
- [5] Liao, Minghui, Baoguang Shi, and Xiang Bai. "Textboxes++: A single-shot oriented scene text detector." *IEEE Transactions on Image Processing* 27.8 (2018): 3676-3690.
- [6] Liu, Xuebo, et al. "FOTS: Fast Oriented Text Spotting with a Unified Network." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.
- [7] Yang, Qiangpeng, et al. "IncepText: A New Inception-Text Module with Deformable PSROI Pooling for Multi-Oriented Scene Text Detection." *arXiv preprint arXiv:1805.01167* (2018).
- [8] Shi, Baoguang, Xiang Bai, and Serge Belongie. "Detecting oriented text in natural images by linking segments." *arXiv preprint arXiv:1703.06520* (2017).
- [9] Liu, Yuliang, and Lianwen Jin. "Deep matching prior network: Toward tighter multi-oriented text detection." *Proc. CVPR*. 2017.
- [10] Lin, Tsung-Yi, et al. "Feature pyramid networks for object detection." *CVPR*. Vol. 1. No. 2. 2017.
- [11] Liu, Wei, et al. "Ssd: Single shot multibox detector." *European conference on computer vision*. Springer, Cham, 2016.
- [12] Zhou, Xinyu, et al. "EAST: an efficient and accurate scene text detector." *Proc. CVPR*. 2017.
- [13] Bottou, Leon, Y. Bengio, and Y. L. Cun. "Global Training of Document Processing Systems Using Graph Transformer Networks." *Computer Vision and Pattern Recognition*, 1997. *Proceedings. 1997 IEEE Computer Society Conference on IEEE*, 1997:489-494.
- [14] Krizhevsky, Alex, I. Sutskever, and G. E. Hinton. "ImageNet classification with deep convolutional neural networks." *International Conference on Neural Information Processing Systems Curran Associates Inc.* 2012:1097-1105.
- [15] Szegedy, Christian, et al. "Going deeper with convolutions." (2014):1-9.
- [16] Simonyan, Karen, and A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *Computer Science* (2014).
- [17] He, Kaiming, et al. "Deep Residual Learning for Image Recognition." (2015):770-778.
- [18] Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.
- [19] Uijlings, J. R. R., et al. "Selective Search for Object Recognition." *International Journal of Computer Vision* 104.2(2013):154-171.
- [20] He, Kaiming, et al. "Spatial pyramid pooling in deep convolutional networks for visual recognition." *European conference on computer vision*. Springer, Cham, 2014.
- [21] Girshick, Ross. "Fast r-cnn." *Proceedings of the IEEE international conference on computer vision*. 2015.

- [22] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
- [23] Liu, Wei, et al. "Ssd: Single shot multibox detector." European conference on computer vision. Springer, Cham, 2016.

## 第3章 图像字符识别

### 3.1 字符识别定义

OCR (Optical Character Recognition), 中文又称光学字符识别, 是指电子设备(例如扫描仪或数码相机)检查纸上打印的字符, 通过检测暗、亮的模式确定其形状, 然后用字符识别方法将形状翻译成计算机文字的过程; 即, 针对印刷体字符, 采用光学的方式将纸质文档中的文字转换成为黑白点阵的图像文件, 并通过识别软件将图像中的文字转换成文本格式, 供文字处理软件进一步编辑加工的技术。目前的印刷及打印文字字符识别软件及设备能阅读各类中西文字符, 且准确率可达 90%以上。通过字符识别软件及设备可将书面上不可编辑的文档及图片转换为可编辑内容<sup>[1]</sup>。

字符识别是计算机视觉研究领域的分支之一, 也是模式识别应用的一个重要领域。目前该课题已经比较成熟, 并且在商业中已经成功应用, 比如汉王 OCR, 百度 OCR, 阿里 OCR 等等。我们的生活也因为 OCR 技术在改变着, 比如一个手机 APP 就能帮忙扫描名片、身份证, 并识别出里面的信息; 汽车进入停车场、收费站都不需要人工登记了, 都是用车牌识别技术等。可以说, 在日常工作、生活、学习中, OCR 技术百花齐放, 大放异彩。因此, 深入研究该项技术, 具有十分重要的意义。

## 3.2 数据集说明

### 3.2.1 数据集意义

字符数据集根据图像采集方式分为三个类别：自然环境下采集的字符图像数据集；手写字符图像数据集；计算机不同字体合成的字符图像数据集。每种数据集都对应不同的 OCR 识别处理方法，每种方法也都有适合自己领域的数据集。找到一个合适的数据集，并将自己的方法应用当中，是至关重要的。另一方面，许多数据集都充当一个 benchmark 的作用，在公开数据集上，测试自己的方法，将更具说服力和权威性。

### 3.2.2 常见数据集介绍

本小节主要介绍一些现今应用广泛且极具权威性的数据集。

#### (1) ICDAR2003

ICDAR2003 数据集于 2003 年发布，作为当年 Robust Reading Competition 的标准数据集，包括自然场景文本图片和人工合成图片。数据集有训练集图片 1157 张，测试集图片 1111 张以及用作样例的图片 171 张。该数据集同时包括单词级别的图片以及字符级别的图片，如下图所示，分别是单词图片和裁剪出来的单个字符图片。



图 3-1 ICDAR2003 示意图

#### (2) ICDAR2011

ICDAR2011 数据集于 2011 年发布，并作为当年 Robust Reading Competition 的标准数据集，包括自然场景文本图片和人工合成图片。它在原来 ICDAR2003 数据集的基础上扩大，并修改了 ICDAR2003 的一些不足之处，例如补全了丢失的 groundtruth 信息，bounding boxes 也与文本之间更加紧密等。数据集共包含 1564 张裁剪后的单词图片。如下图所示，图 (a) 是单词图片，图 (b) 是单词对应的 groundtruth。



图 3-2 ICDAR2011 示意图

### (3) ICDAR2013

ICDAR2013 又称 Focused Scene Text Challenge，数据集在 2013 年发布，作为 2013 年的 Robust Reading Competition 的标准数据集。图片样本多为水平字符，同时包含字符级图片和单词级图片。训练集共有 3567 张裁剪后的图片，测试集共有 1439 张裁剪后的图片。如下图所示，左边是单词图片，右边是单词对应的 groundtruth。

word_2.png	word_3.png	word_1.png, "flying" word_2.png, "today" word_3.png, "means" word_4.png, "vueling" word_5.png, "GET" word_6.png, "AWAY," word_7.png, "1.000.000" word_8.png, "SEATS"
		word_9.png, "FROM" word_10.png, "30€"
word_9.png	word_10.png	word_11.png, "Book" word_12.png, "now!" word_13.png, "SMI" word_14.png, "SensoMotoric"
		word_15.png, "Instruments" word_16.png, "NEWSLETTER" word_17.png, "click" word_18.png, "play"
word_16.png	word_17.png	word_19.png, "Politicians," word_20.png, "Cheating"
		word_21.png, "John" word_22.png, "How"
word_23.png	word_24.png	word_23.png, "Bill" word_24.png, "Clinton" word_25.png, "John" word_26.png, "How"
		word_27.png, "party" word_28.png, "lines"
word_30.png	word_31.png	word_27.png, "party" word_28.png, "lines"

图 3-3 ICDAR2013 示意图

### (4) IIIT-5K

IIIT 5K 单词数据集是从 Google 图像搜索中获取的，包括广告牌，商店招牌，门牌号，房屋铭牌，电影海报等。该数据集包含来自场景文本和人工合成的 5000 个裁剪之后的单词图像，每张图像关联一个 50 词的词典和一个 1000 词的词典。数据集分为训练和测试部分，可用于 large-lexicon 裁剪之后的单词识别。此外，该数据集还提供了一个包含超过 50 万个单词的词典。下图为数据集的部分样例。



图 3-4 IIIT-5K 示意图

#### (5) SVHN

SVHN (The Street View House Numbers) 是一个 real-world 的图像数据集，对数据预处理和格式化的要求很低，可以被看作与 MNIST 相似的数据集，但是该数据集有更多的标记数据（超过 600,000 个数字图像）。SVHN 是从 Google 街景图像中的门牌号码获得的，共分为 10 类，每类代表一个数字，例如数字“1”有标签 1，“9”有标签 9，“0”有标签 10。数据集可用于训练的样本共 73257 个，用于测试的共 26032 个，以及用作额外训练的有 531131 个难度稍低的样本。这些数据有两种格式：带有字符级边界框的原始图像；类 MNIST 的，图片大小为 32×32 的以单个字符为中心的图像。下图为数据集的部分样例。





图 3-5 SVHN 示意图

#### (6) MSRA TD-500

MSRA TD-500 是一个多方向文本的数据集，大部分文本都选自导向牌之类的标志。图片分辨率在 1296x864 到 1920x1280 之间，包含中英文，总共 500 张自然场景图片。其中，训练集为 300 张，测试集为 200 张，标注以行为单位，而不是单词，每张图片都完全标注，难以识别的有 difficult 标注。数据集的格式如下：



图 3-6 MSRA TD-500 示意图

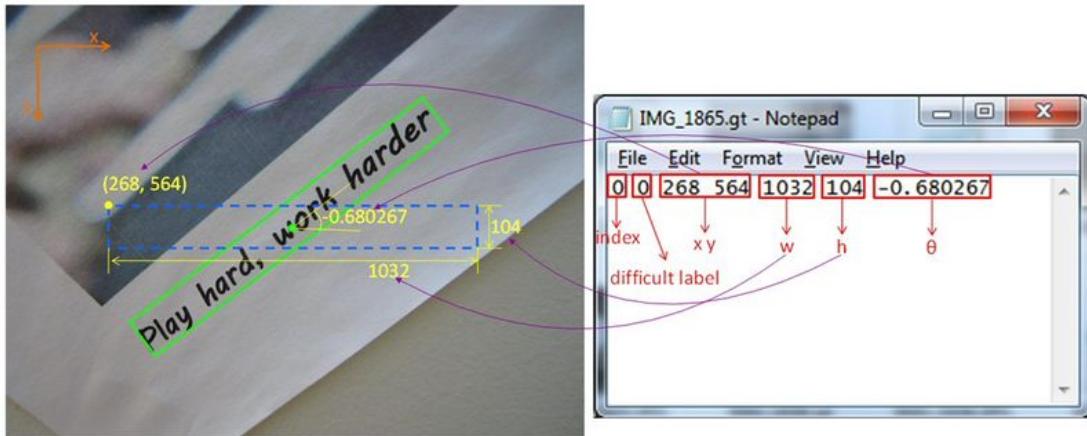


图 3-7 MSRA TD-500 格式示意图

#### (7) Chars74

Chars74K 数据集是一个经典的字符识别数据集，主要包括了英文字符与坎那达语 (Kannada) 字符。数据集一共有 74K 幅图像，所以叫 Chars74K。该数据集包含了 52 个字符类别 (A-Z, a-z) 和 10 个数字类别 (0-9) 一共 62 个类别，3410 副图像，由 55 个志愿者手写完成。图像按照 Samples001-Samples062 的命名方式存储在 62 个子文件夹下，每个子文件夹有 55 张图像，都为 PNG 格式，分辨率为 1200\*900 的三通道 RGB 图像。如下图所示，为数据集的一些样例。



图 3-8 Chars74 示意图

如下表所示，总结和整理了常见数据的基本信息及功能，方便查看。

表 1：数据集基本信息

数据集	图像数			标注级别				水平/ 倾斜	语言
	训练集	测试集	总计	字符	单词	按行 标注	分 割		
ICDAR2003	258	251	509	有	有	无	无	水平	英文

ICDAR2011	229	255	484	无	有	无	无	水平	英文
ICDAR2013 /ICDAR2015-Focus	229	233	462	有	有	无	有	水平	英文
ICDAR2015-Incidental	1000	500	1500	无	有	无	无	倾斜	英文
SVT	100	250	350	无	有	无	无	水平	英文
MSRA-TD500	200	300	500	无	无	有	无	倾斜	中英文
CASIA_Multilingual	248	239	487	无	无	有	无	水平	中英文
IIT5K-Word	2000	3000	5000	有	有	无	无	水平	英文
Synth800k	—	—	858750	无	有	无	无	水平	英文
Synth90k	—	—	约 90k	无	有	无	无	水平	英文
SCUT_FORU_English	1200	515	1715	有	有	无	无	水平	英文
SCUT_FORU_Chinese	1861	355	2216	有	无	无	无	水平	中文
COCO-Text	43686	20000	63686	无	有	无	无	水平	英文
KAIST	—	—	3000	有	有	无	有	水平	中英韩文
OSTD	—	—	89	有	无	无	有	倾斜	英文
USTB	500	500	1000	无	有	无	无	倾斜	英文
Chars74k	7705 个自然场景字符， 3410 个手写字符， 62992 个合成字符			有	无	无	无	水平	英文

表 2：数据集功能

数据集	水平/ 倾斜	语言	检测			识别		端到端	分割
			字符	单词	按行标注	字符	单词		
ICDAR2003	水平	英文	√	√		√	√	√	
ICDAR2011	水平	英文		√			√	√	
ICDAR2013 /ICDAR2015-Focus	水平	英文	√	√		√	√	√	√
ICDAR2015-Incidental	倾斜	英文		√			√	√	
SVT	水平	英文		√			√	√	
MSRA-TD500	倾斜	中英文			√				
CASIA_Multilingual	水平	中英文			√				
IIT5K-Word	水平	英文				√	√		
Synth800k	水平	英文		√			√		√
Synth90k	水平	英文					√		
SCUT_FORU_English	水平	英文	√	√		√	√	√	

SCUT_FORU_Chinese	水平	英文	√						
COCO-Text	水平	英文		√			√	√	
KAIST	水平	中英 韩文	√	√		√	√	√	√
OSTD	倾斜	英文	√						√
USTB	倾斜	英文		√			√	√	
Chars74k	水平	英文				√			

### 3.3 字符识别综述

1929 年，德国科学家 Tausheck 第一次提出 OCR（光学字符识别，Optical Character Recognition）这个概念，OCR 指采用图像处理和模式识别技术对光学字符进行识别。20 世纪 50 年代出现了商业光学字符阅读器（OCRs），过去的四十年的字符和文档识别的主要商业及工业应用一直在阅读方面，银行支票的阅读和邮政地址读取以及邮政号码、车牌号码、身份证号码及其他编号识别等<sup>[2]</sup>。80 年代初，我国科研人员开始对其进行研究。如今，人们不仅要求 OCR 产品具有很高的识别率，还要求 OCR 产品具有简便的操作、友好的用户界面以及稳定可靠易升级的特点<sup>[3]</sup>。

本部分将从自然环境下采集、手写、计算机不同字体合成三种字符图像数据集进行展开。

#### 3.3.1 自然环境下采集的字符图像识别

场景字符识别方法一般有两类。

##### 3.3.1.1 传统 OCR 方法识别

其一是传统的 OCR 方法，将字符从背景中分割出来，二值化后得到二值的连通域，提取连通域上的特征可进行分类。分类结果极大依赖于分割结果及特征的选择。

gllavata 等人[4]首先通过颜色量化的方法确定文本和背景颜色，然后采用 Kmeans 算法将每一个像素分为文本或背景。同样地，Song 等人[5]使用基于颜色聚类方法分割文字和背景。这种基于色彩聚类方法的性能依赖于颜色的一致性，而且对噪声和文本的分辨率也十分敏感。Chen[6]使用混合高斯模型对图像中的灰度分布进行建模，对每一个像素根据马尔科夫随机场产生的先验知识来指定一个高斯类别。Ye 等[7]提出利用采样点在 HIS 空间的强度和饱和度信息来训练一个混合高斯模型，然后结合空间连接性信息来分割文本区域像素。Li 等人[8]结合了局部视觉信息和上下文标注信息，利用 CRF 来分割字符。Field 等[9]采用双侧回归模型来平滑不受相邻区域影响的图像区域的色彩变化，然后利用识别系统的反馈来选择前景区域。

与传统扫描文档图像的二值化方法，如 Otsu[10]、Niblack[11]相比，一些近来提出的二值化或预处理的方法确实可以提高场景文本的识别率。然而，由于自然场景图像分辨率，光照条件，大小和字体样式不受约束，导致二值化结果并不理想。此外，二值化过程中的信息损失几乎是不可逆的，这意味着如果二值化效果差，正确识别的文本的机会很小。如图 3-9 所示，在二值化结果非常不理想的情况下，字符正确识别的概率几乎为零。



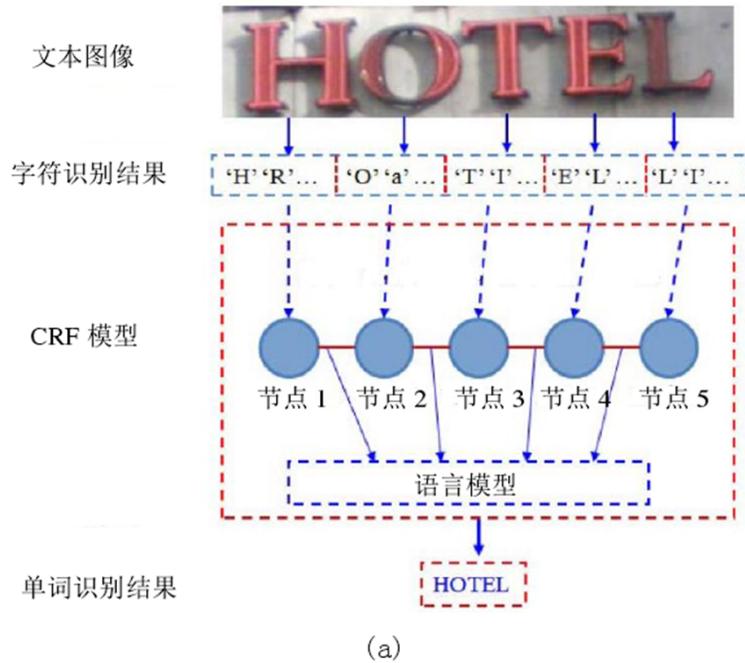
图 3-9 传统的 OCR 分割效果。 (a) 自然场景中文本区域。 (b) Ostu 的二值化效果

### 3.3.1.2 基于目标识别

其二为基于目标识别的方法，通过提取每类字符区域的特征表达来进行分类。基于目标识别的方法假设场景字符识别和具有高度的类内变化的目标识别是非常相似的。这类场景字符识别的方法，一般直接从原始图像中提取特征，然后利用不同的分类器完成识别。

Chen 等[10]提出了一种用局部强度归一化的方法来应对光照不均的情形，然后用 Gabor 变换获取局部特征，最后用线性判别分析来选择合适的特征进而分类。受到词袋模型 (bag-of-words)思想[12]的启发，De Campos 等[13]测试了不同特征对目标识别任务的表现，发现将基于几何模糊[14]和形状上下文[15]方式结合并利用最邻近分类器的识别结果表现最佳。Newell 和 Griffin[16]通过扩展两个 HOG 特征描述算子得到多尺度特征，他们的方法在多个公共测试数据库上都表现得十分突出。Weinman[17]提出将字符外观、两组字的频率、相似性和词汇信息融合到识别中。Shi 等人[18]提出一种基于字符局部部件的树形模型 (part-based tree-structured model, TSM)，可以同时定位和识别局部图像中的文字。

大多数现有的基于目标识别的场景文本识别方法，由于没有二值化分割阶段，均采用多尺度滑动窗口策略[19]得到候选字符检测结果。通过单词的识别策略，如图案结构或 CRF[20]，从候选字符检测结果中来得到最终的文本识别结果如图 3-10 (a) 所示。此外，借助语言模型或主题词管理系统[17]，校正错误识别的词进而获得有意义的词含义来实现基于单词级的文字识别，如图 3-10 (b) 所示。但是这类方法对字符的方向要求严格，且很难适应平移变换。



文本图像	直接识别结果	二元语法 (Bi-gram)	三元语法(Tri-gram)
jungle	JUN9IE	JUNGIE	JUNGLE
ship	SB1P	SBIP	SHIP
Better	BETTCR	BETROR	BETTOR
QUEEN	0UEEN	OLEEN	QUEEN

(b)

图 3-10 场景文本单词识别策略。 (a) 通过 CRF 进行候选字符组合。 (b) 通过语言模型进行单词识别结果校正。

### 3.3.2 手写体汉字识别

手写体汉字识别一直是模式识别研究领域中的难点。手写体汉字识别根据实时可分为联机手写体识别和脱机手写体识别。联机识别的技术比较成熟，识别率较高。脱机识别的难度很大，技术尚不成熟，尤其是脱机手写体汉字字符识别难度更大[21]。

#### 3.3.2.1 联机手写体汉字识别

目前大量的联机手写体汉字识别系统采用的都是结构识别方法。所谓结构识别方法，其基本思路是把复杂的汉字模式分解为简单的子模式直至基本字根、笔画、笔段等。通过对子模式的判定以及基本符号运算的匹配算法，达到对复杂模式的识别。

徐志明等提出了一种规则和统计相结合的计算语言模型应用于联机手写体汉字识别后处理的技术[22]，把基于统计的大词表 Markov 语言模型与语言规则量化模型，通过词网格技术集成在一个语言解码器。该项技术已应用于 HPC(手持机)手写电脑的联机手写体汉字识别系统中。俞庆英等设计了一种基于获取笔段序列的联机手写体汉字识别方法[23]，用可视化编程工具 VC ++6.0 实现了算法，平均识别率达 95.8%。张冬霞提出了人工神经网络(ANN)和隐马尔可夫模型(HMM)相结合的汉字识别法[24]，实验证明该汉字识别系统具有较高地识

别准确率。赵巍等为了对自由手写汉字进行有效地表征和识别，提出了一种基于部件 HMM 级联的联机手写体汉字识别方法[25]。该方法采用面向级联的 Viterbi 算法，无需做部件的分割和标注。实验训练与识别表明，该方法的第一候选识别率为 87.189%，而基于分段 HMM 识别方法的第一候选识别率为 86.117%。鲁湛等为了提高联机手写体汉字模型的空间结构描述能力和识别性能，从汉字的笔段关系出发，提出一种新的联机手写体汉字模型——ARHMM[26]。ARHMM 具有完整的参数训练方法和识别算法，实验结果表明 ARHMM 联机汉字模型与 THMM 联机汉字模型相比，平均错误率下降了 23.65%。

### 3.3.2.2 脱机手写体汉字识别

脱机手写体汉字识别基本上囊括了模式识别研究领域中的所有典型问题，脱机手写体汉字识别一直是模式识别研究领域中的难点[21]，存在方法各异的具体应用。龚才春等给出了一种从脱机手写体汉字字符中识别笔顺的法则[27]，从而将脱机识别问题转变为联机识别问题，简化了识别过程，大大提高了识别率和识别速度。

李元祥等利用上下文关系进行汉字识别后处理时，提出一种扩充候选字集的方法[28]。该方法利用单字识别给出的候选字来推测可能正确的字，文本平均识别率从扩充候选字之前的 93.93% 提高至 95.82%，错误率下降了 31.14%。张睿等在脱机手写体汉字识别中提出了最优采样特征法[29]。通过在 THCHR 样本集上进行实验，最优采样特征比均匀采样特征的识别率上升了 1.83%，错误率下降了 18.47%。童学锋等提出用模糊支持向量机解决了多类支持向量机中的不可分区域问题[30]。将模糊支持向量机引入到有限集脱机手写体汉字识别中，多组实验数据结果表明在相同的条件下可以达到比支持向量机更为理想的识别效果。高彦宇等提出的基于融合特征和 LS-SV 的脱机手写体汉字识别系统主要研究特征提取和分类识别 2 个模块[31]。由多组实验数据可知，对于相同的特征向量，LS-SVM 分类识别方法得到的识别率高于采用距离分类器得到的识别率。LS-SVM 所具有的泛化能力对小样本集类别有更好的分类识别能力。吴雪菁等采用了基于字符质心的层次特征对无约束手写体数字进行分类识别[32]，在一定程度上克服了无约束手写体数字字形变化引起的不稳定性。该算法应用于无约束手写体数字的信函分拣系统中，单字的平均识别率达 97% 以上。王革新等[33]提出了一个字符轮廓曲率计算的有效方法，该方法不是直接从轮廓点出发，而是进一步抽样得出新的轮廓点，最后得到了不受字符旋转、平移、大小、位置影响的字符特征，以及相应的识别算法。实验表明，与其他的多层识别算法比较，该方法的拒识率提高 10.7%，而识别率只降低 12.9%，误识率提高 9.1%。朱小燕等提出了一种基于反馈的手写体字符识别方法[34]。该方法将人工神经网络结构及学习算法运用于系统反馈机制中，多组试验数据证明该方法大大降低了高噪音手写体数字的识别率。龚才春等通过模拟人眼识别数字字符的过程，提出了一种基于字符整体特征(凹凸特征)的快速手写体数字字符识别方法[35]。该方法不需要对字符图像做复杂的细化处理，也不需要进行复杂的笔道特征分析，减少了细化形变可能带来的误识和拒识。在识别过程，平均每秒识别数字字符 235 个。从不同性别、不同文化层次的人群中采集了 50000 个样本数字字符，经多组试验数据显示该方法具有很高的识别率和很低的拒识率。

### 3.3.3 印刷体识别

最后一种是印刷体。印刷体字符识别指用扫描仪扫描或数码相机拍摄等光学方式录入印刷在纸张上的文字，提取得到的灰度图像或者二值图像的文字特征，进行识别。印刷体由于本身比较规整，识别难度相比手写体要低。

在印刷体汉字识别方面，我国有多种汉字识别产品已被广泛使用，如“汉王”、“尚书”、“清华紫光”等。其中“汉王”文字识别系列产品应用的最为广泛，“汉王全能阅读器”集成了多种模式识别方法，在技术上有重大突破，实现了印刷体、手写体汉字的扫描识别、汉王笔“三合一”功能。“汉王”系列文字识别产品从联机手写汉字输入识别系统，发展成为手写体、

印刷体、简体、繁体、数字识别及日文、韩文等多文种的识别系统，技术水平上一直处于国际领先地位[36]。任金昌等提出了一种面向“电子阅读笔”的快速文字识别算法[37]，从而使一种便携式的“电子阅读笔”的实现成为可能。熊军等提出了一种改进的手写印刷体汉字识别细化算法(I-FA 算法)并进行了计算机仿真[38]，实验结果表明 I-FA 无论在速度或细化质量上都有较明显的提高。唐亮等提出了在有相当强烈的干扰条件下，基于小波变换的印刷体汉字处理的一种新方法[39]，实验表明该方法具有良好的去干扰性。钟锐等采用特征识别的方法[40]，提取数字的垂直投影峰数和过线数等特征值对印刷体数字进行识别，并在 Visual Basic 编译环境下编程实现，该算法能准确快速地识别任意大小的宋体印刷体数字，其识别率达 98.125%。王维兰等以印刷体现代藏文白体、黑体、圆体、长体、竹体为字体样张，通过预处理、文本行字切分、特征选择和分类识别的初步研究，获得对 5 种字体文本的平均识别率为 89.582%，对其他字体的文本平均识别率为 93.867%，并形成藏文识别字典库[41]。郑朝晖等提出一种基于遗传算法的(0, 1, \*)——矩阵法的印刷体字符识别的新方法[42]。该方法在印刷体邮政编码识别实验中，在大大缩短识别时间的同时，识别率可达 98.1%；而相同实验条件下应用传统模板匹配法时的识别率为 92.1%。陈兆学等针对工业场景下印刷体字符的特点，提出了一种基于方波参数求取进行字符分割的方法[43]。通过对多种工业现场条件和背景下 60 幅灰度图像进行的实验结果表明，该系统有效识别率可以达到 95%以上，在特定工业场景下对印刷体字符的识别取得了满意的分割识别效果。如今印刷体字符的识别技术已经成为了一门相对成熟的技术[44]。该技术以清华文通、中国科学院计算机技术所智能中心、中字汉王、重庆大学关机所等研究单位作为国内代表性单位。其中清华大学电子工程系研制的 TH-OCR 产品知名度最高，它有效解决了模式样本差异显著、模式类数量极大的模式识别实用化问题。

## 3.4 主流算法介绍

### 3.4.1 TSCD model

文章标题《Robust Scene Text Recognition with Automatic Rectification》，白翔等[45]。[华中科技大学，CVPR2016。](#)

文章的 motivation 就是解决自然场景下不规则形状的文本识别。方法是借鉴 STN 网络的思想，先用 STN 网络对不规则文本图片进行矫正，再把矫正之后的图片送到一个基于注意力机制的 SEQ2SEQ 的网络去识别。



图 3-11 RARE model

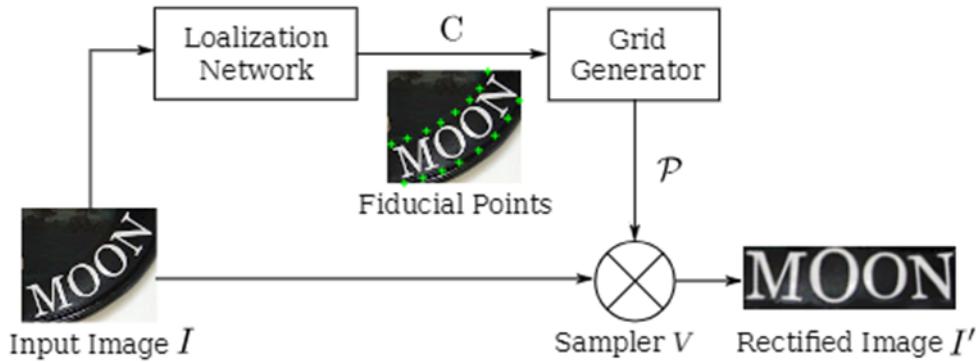


图 3-12 STN 网络结构

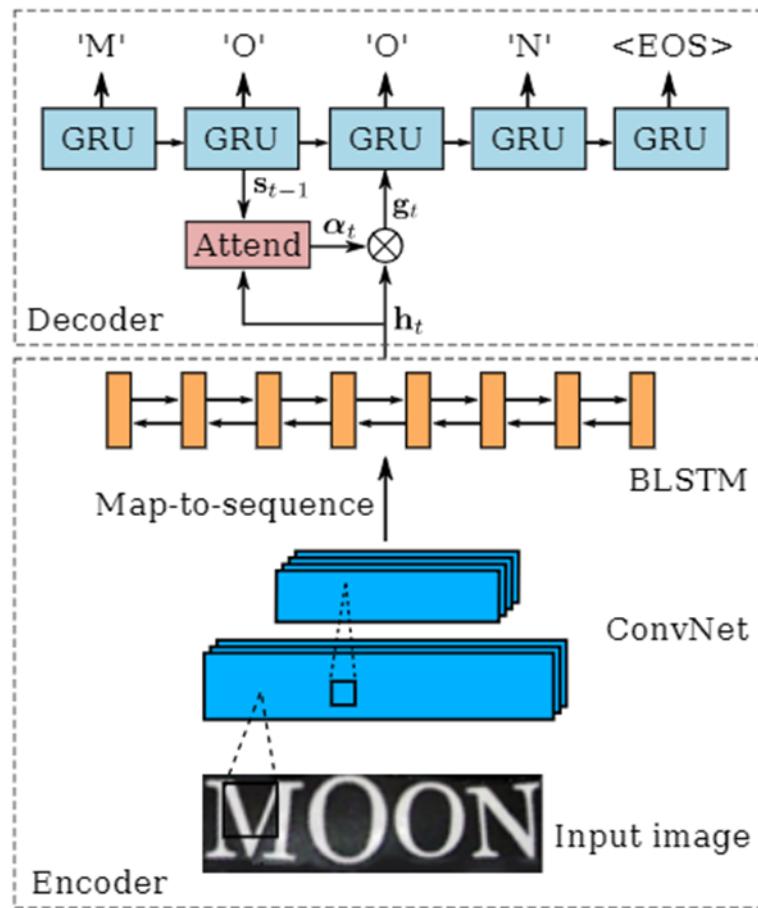


图 3-13 SRN 网络结构

从论文的实验结果来看，STN 网络对文中提到的不规则文本的矫正能力有限，尤其是当 perspective text 和 curved text 的形变角度很大时，STN 的矫正结果不是很理想。对此，我们可以做一下改进：首先，可以借鉴人脸的做法改进 STN 的 localization network 的基准点的坐标定位，把弱监督信息改为强监督信息。其次，透视变换文本中所有字符的形变角度都相同，弯曲的文本中不同的字符有不同的朝向，可以先把每个字符定位出来，算出每个字符的形变角度，在字符级别上做矫正。

### 3.4.2 AON model

文章题目《AON: Towards Arbitrarily-Oriented Text Recognition》,Zhanzhan Cheng 等[46]。  
Hikong Vision & Fudan, CVPR2018。

文章的 motivation 就是解决自然场景下任意朝向的不规则的文本识别。方法是将任意方向的字符编码为 4 个方向的 4 个特征序列表示：左→右，右→左，上→下，下→上。AON 提取 4 个方向的场景文字特征和位置信息，FG 融合 4 个方向的特征序列。

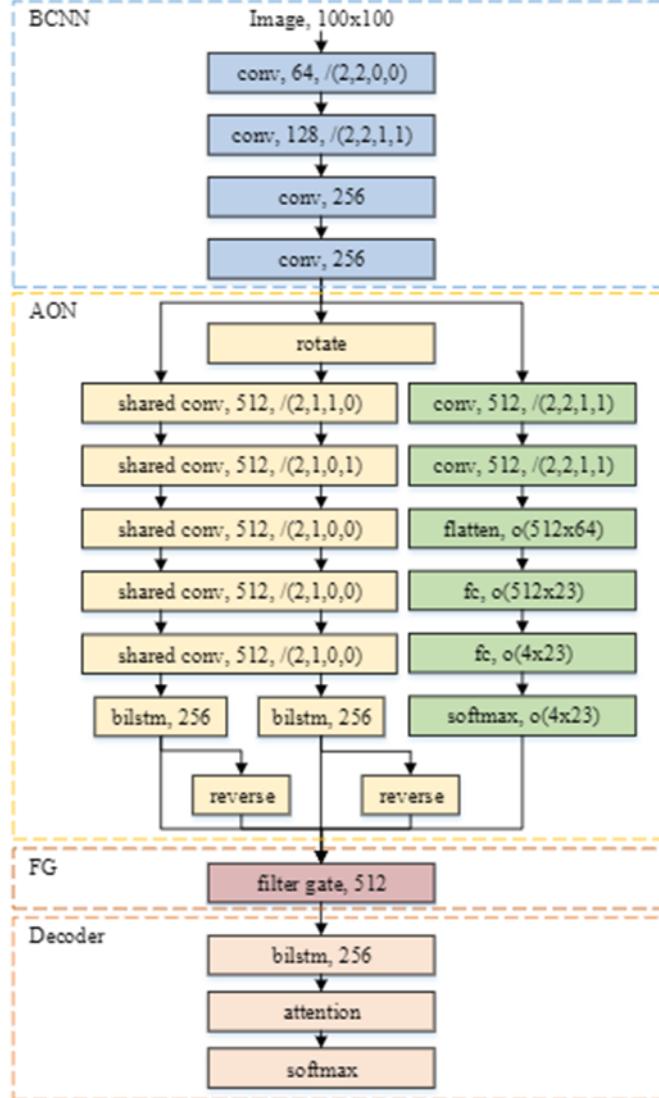


图 3-14 AON 网络结构

### 3.4.3 DTRN model

论文标题《Attention-based Extraction of Structured Information from Street View Imagery》，Wojna Z 等[47]。伦敦学院大学和谷歌合作论文，Arxiv.1704.03549。

这篇论文的主要贡献就是提出了一种新型的基于 attention 的文本阅读结构，并以 end-to-end 的方式，在 Street View Business Names dataset 和 French Street Name Signs Dataset (FSNS) 两个数据集上取得了相当好的效果。该模型的主要框架如图3-15所示。FSNS 数据集中每张图片有 4 个不同 view 的图片，将这四个视图中的每一个通过相同的 CNN 特征提取器，然后将结果连接成单个大的 feature map，如图中标有“f”的立方体所示。然后采用空间加权组合的方式，加入 attention 的权重之后，形成一个固定大小的特征矢量  $u_t$ ，再将其送入 RNN 中识别。

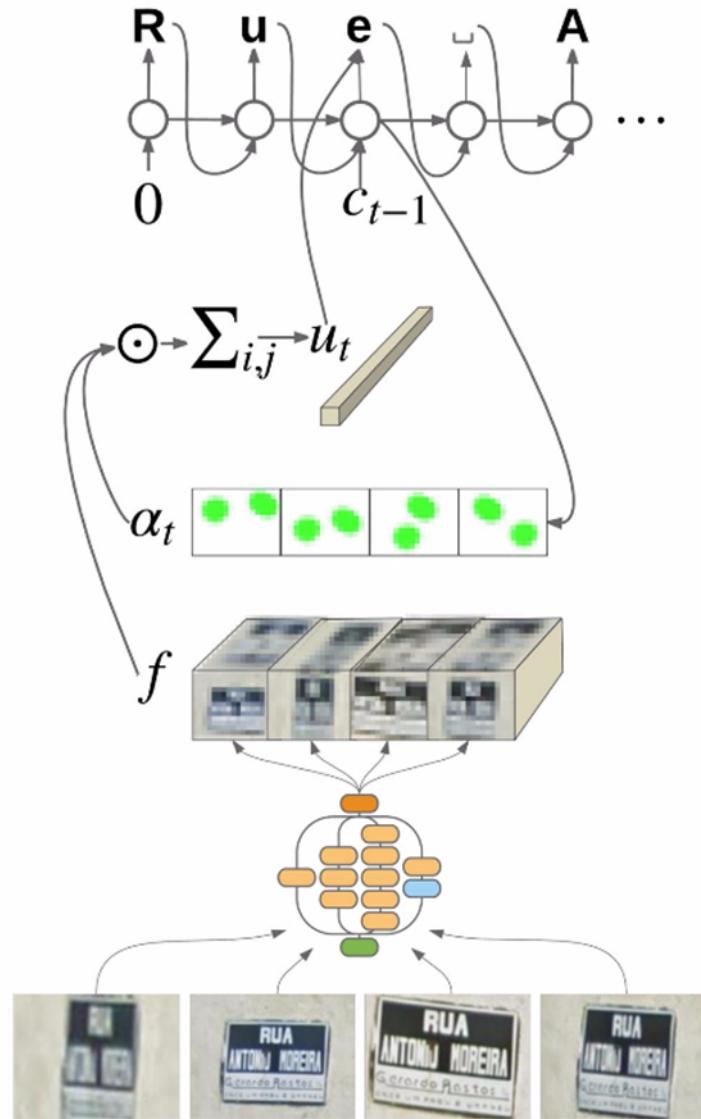


图 3-15 Architecture of attention model

在加入 attention 权重的时候，将公式 (2)  $\tanh$  函数中的参数换成公式 (4)，即用 one-hot 编码的形式，加入了坐标信息，如图 3-16 所示，也因而在能跳行读取竖排文本的同时，能够更准确的 focus 到相应的字符。

$$a_{t,i,j} = V_a^T \tanh(W_s s_t + W_f f_{i,j,:}) \quad (\text{公式 3-1})$$

$$\alpha_t = \text{softmax}_{i,j}(a_t) \quad (\text{公式 3-2})$$

$$W_s s_t + W_{f_1} f_{i,j,:} + W_{f_2} e_i + W_{f_3} e_j \quad (\text{公式 3-3})$$

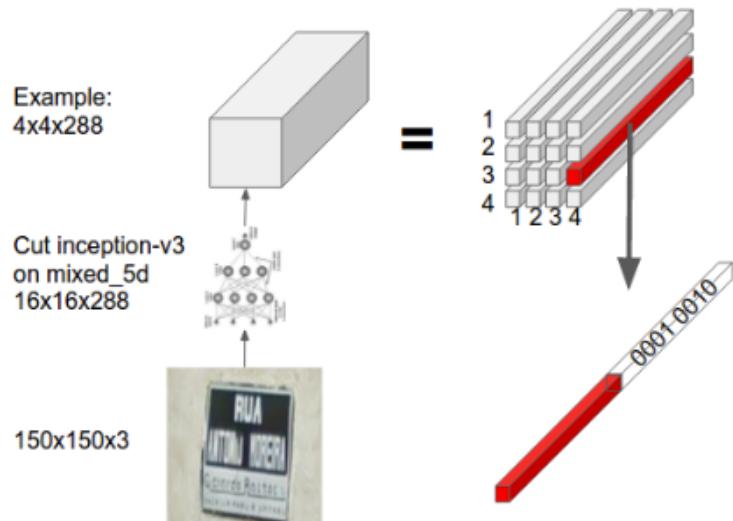


图 3-16 将像素坐标添加到图像特征

实验取得的效果如图 3-17 所示，可以准确的完成跳行读取文本的任务。

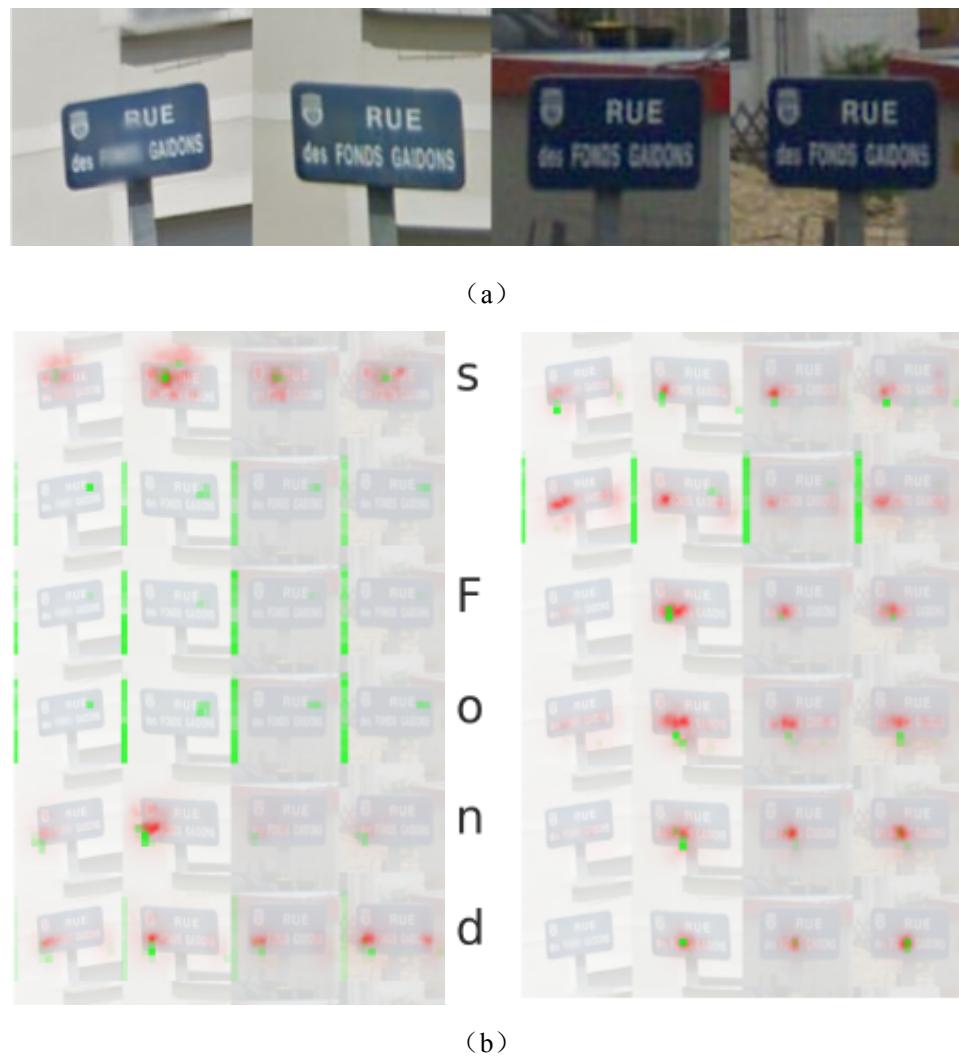


图 3-17 (a) 为数据集中的待测试图片；(b) 为识别结果

#### 3.4.4 R<sup>2</sup>AM model

文章标题《Recursive Recurrent Nets with Attention Modeling for OCR in the Wild》,

Chen-Yu Lee 等[48]。加利福尼亚大学圣迭戈分校，雅虎，CVPR 2016。

文章的 motivation 就是在 lexicon-free 的情况下，解决自然场景的文本识别。此文中的模型是基于参考文献《Deep structured output learning for unconstrained text recognition》[49] 所提出的模型的，这篇 paper 是用 CNN+CRF 结合的模型，对字符进行识别后，进行 N 元文法分析。本文的方法是使用带 Attention Model 的 recursive RNN。直接用图片进行词汇字符串（word string）学习，实现了对无约束（unconstrained，即 lexicon-free，未知 word 长度）自然场景文本进行识别。

整体系统结构如图 3-18 所示。首先将输入图片送到递归卷积层以提取 encoded 特征，然后通过基于隐式学习的字符级语言统计的循环神经网络进行 decode。Attention-based 机制使用的是 soft 特征选择，具有更好的特征运用效果。

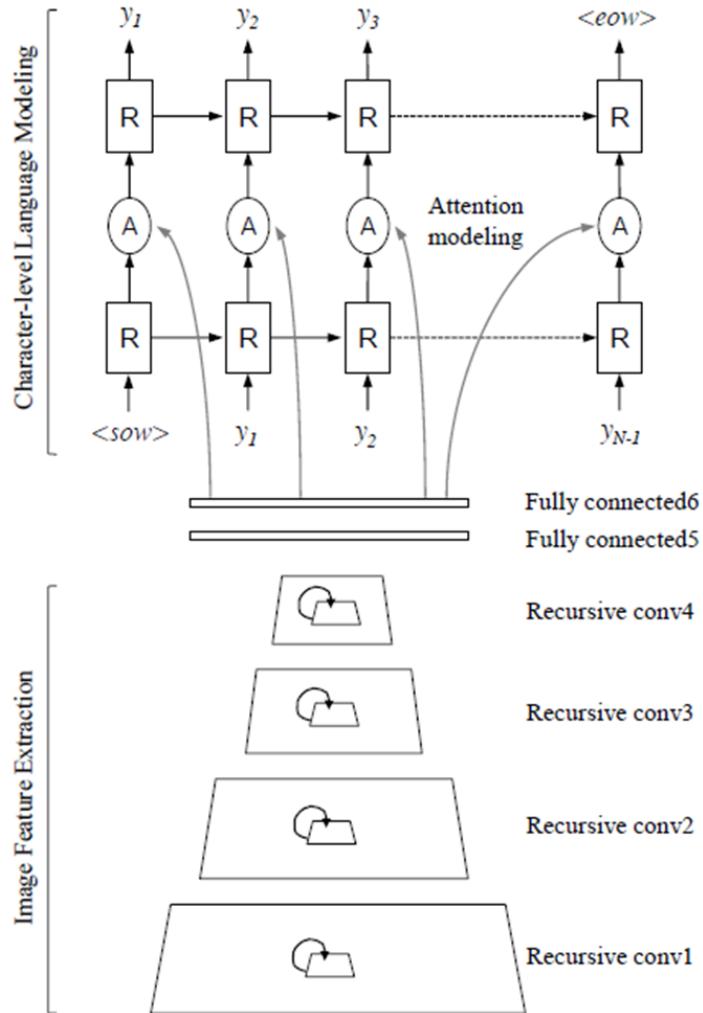


图 3-18 整体系统结构

本文中的方法使用了 recursive CNN：在时间  $t$  时，recursive 卷积层输入图像/特征的关系为：

$$h_{i,j,k}(t) = \begin{cases} \sigma((w_k^{hh})^T x_{i,j} + b_k), & t = 0 \\ \sigma((w_k^{hh})^T h_{i,j}(t-1) + b_k), & t > 0 \end{cases} \quad (\text{公式 3-4})$$

其中， $h_{i,j}(t-1)$  表示向量化前馈， $x_{i,j}$  表示 feature map 上以  $(i,j)$  为中心的输入 patches， $w_k^{hh}$  为输出 channel  $k$  的向量化前馈权值。 $b_k$  为输出 channel  $k$  的偏差。 $\sigma$  为确定的非线性转换函数。

recursive CNNs 在相同参数容量的情况下增加了传统 CNNs 的深度，同时也比 CNNs 产生更加紧凑的特征响应。recursive 相互作用也可以视为 feature map 中的一种“横向连接性”，使得给定层的表示更好捕捉到高层依赖。

上式约束所有的权值  $w_k^{hh}$  共享相同的内部值——他们“捆绑（tied）”一起。这种捆绑的一种结果就是所有层的 channels 数目将使一样的，因为共享权值总是将输入 feature maps 映射到相同维数（宽\*高\*channels 的数目）的输出 feature map。本文提出一种 recursive 卷积层的“非捆绑（untied）”变体，区别在于层间（inter-layer）前馈权值  $w_{untied,k}^{hh}$ ，后面的层内（intra-layer）recursive 权值  $w_{tied,k}^{hh}$  这种方法允许在不同层具有不同数目的 channel，并且时 recursive 权值可以更加自由特化。

通过在时间  $t=0$  的时候 untying 前馈权值，公式 3-4 变为：

$$h_{i,j,k}(t) = \begin{cases} \sigma((w_{untied,k}^{hh})^T x_{i,j} + b_k), & t = 0 \\ \sigma((w_{tied,k}^{hh})^T h_{i,j}(t-1) + b_k), & t > 0 \end{cases} \quad (\text{公式 3-5})$$

通过这种方法，任意 recursive 卷积层的 channels 数目可以由 untied 权值  $w_{untied,k}^{hh}$  来进行调整，控制整体的计算代价。可以使用相容的逻辑来 untie recurrent 卷积层，如图 3-19 所示。

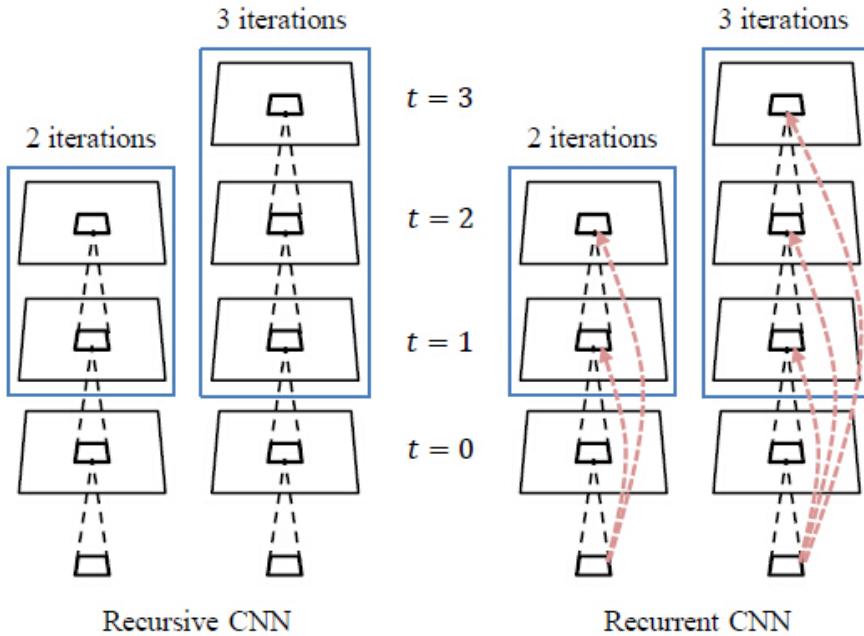


图 3-19 (a) Recursive CNN (b) Recurrent CNN

文中使用 RNNs (recurrent neural networks) 来对文本字符级统计进行建模。RNN 及其变体 LSTM (Long Short-Term Memory) 在处理序列数据的时候非常有效。识别图片中的字符可以将其视为解决 sequential dynamics 和学习从像素强度到自然字符级向量映射的问题。这个模型获取单张图片，并且生成一个字符序列，每个字符为 K 个编码字符中的一个。

Attention-based 机制使得模型专注于输入特征的最重要的分割，并且可能添加一个可解释性 (interpretability) 的级别。一般有两类 attention-based 图像理解：hard-attention 和 soft-attention。Hard-attention 模型学习选择一序列离散的 glimpse location，但是很难训练，因为损失梯度 (loss gradients) 很难处理。本文使用的是一个 soft-attention 模型，可以使用标准的反向传播来进行训练。

本文的主要贡献有：

(1) recursive CNNs：相同的参数容量下能够更加有效地提取图片特征；

(2) 隐式学习的字符级语言模型：嵌入了 RNN (recurrent neural network)，避免进行 N 元文法分析 (Ps: 个人理解就是用一个 RNN 代替了 N 元文法分析的功能);

(3) 使用了 soft-attention 机制：使模型能够有选择地利用图片特征，并且可以使用标准的反向传播来进行端到端训练。

### 3.4.5 DTRN model

文章题目《Reading Scene Text in Deep Convolutional Sequences》，Weilin Huang 等[50]。  
中国科学院深圳高等技术研究所，AAAI 2016。

文章的 motivation 就是解决不用字典进行单词识别，识别新词，任意没有语义的字符串以及有歧义、形变大的文字图像。方法是 maxout 版的 CNN 提取特征，RNN (LSTM) 进行分类，CTC 对结果进行调整。整个流程端到端训练和测试，和白翔的 CRNN 方法大体相同，除了（1）用 maxout 的 CNN 替换白翔的普通 CNN 以及（2）这篇文章的 CNN 是原图中的每个  $32 \times 32$  个大小的滑动窗口，直接得到第四层的一个卷积向量作为该窗口的特征（最后一层用的滤波器大小和该窗口对应的 feature map 大小一样，故得到  $1 \times 1$  的值），而白翔的 CNN 最后用了一个 Map-to-Sequence 把 CNN 最后一层的 feature map 上的每个滑动窗口直接拉成一列一列的特征输出到 RNN 中。图 3-20 流程图：

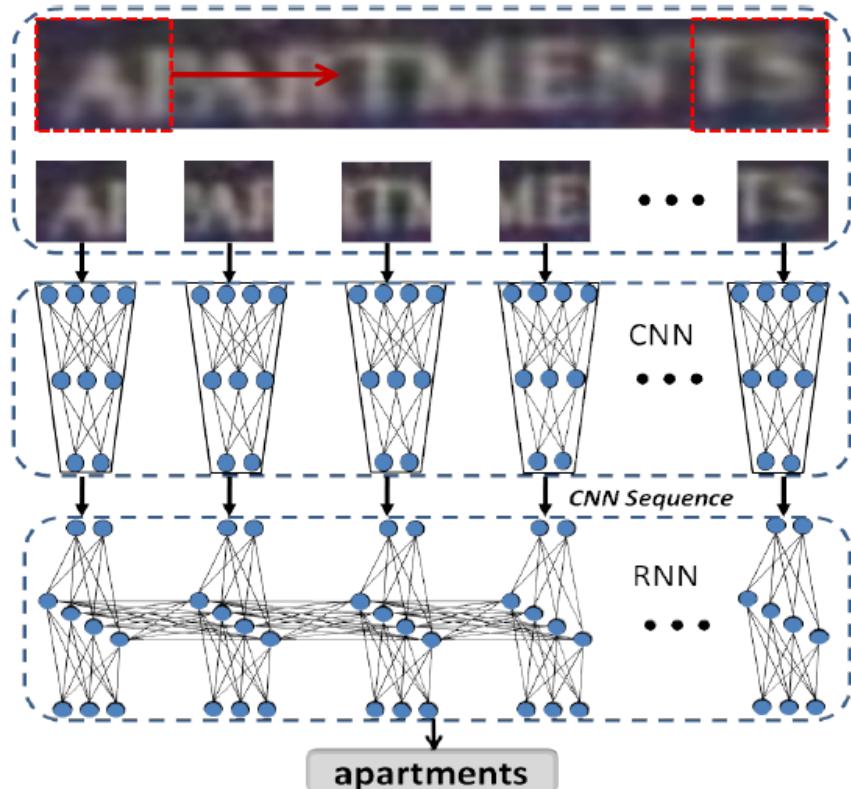


图 3-20 所提的 DTRN 模型的流程图

这篇文章的 maxout CNN 网络结构如图 3-21 所示：

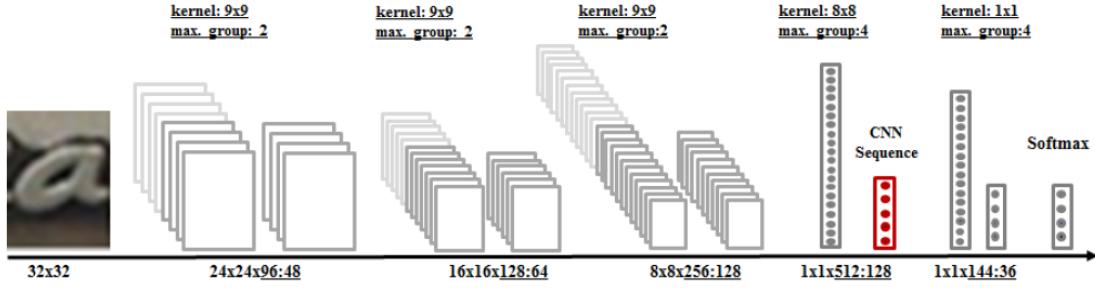


图 3-21 maxoutCNN 模型的结构图

这篇文章的识别（从 CNN 特征到最后的单词输出）流程如图 3-22 所示：

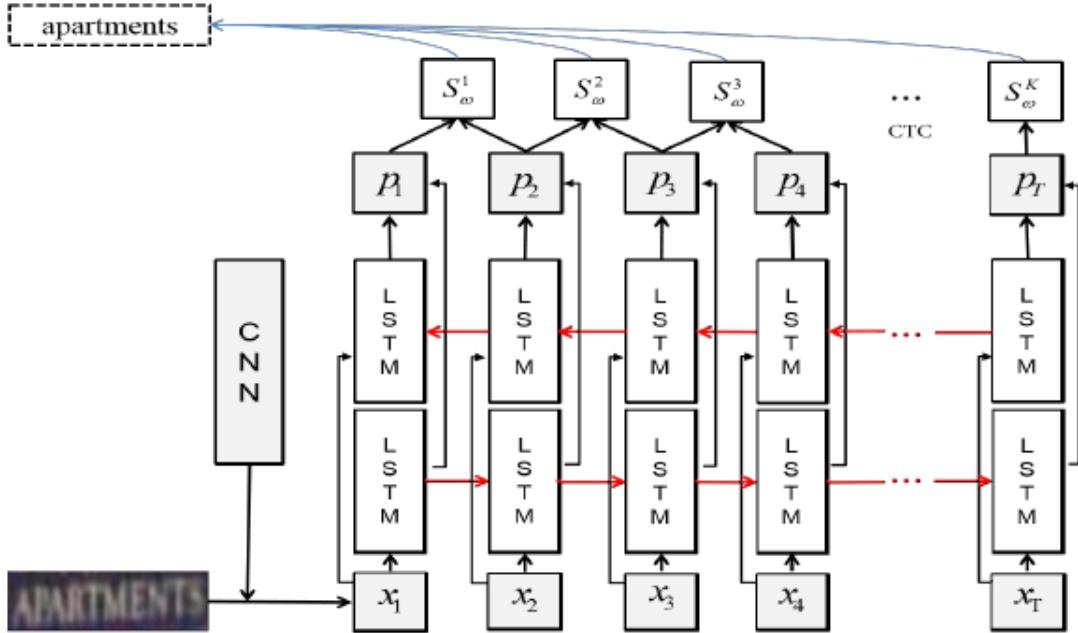


图 3-22 识别的流程图

贡献：idea 的出发点——把单词识别问题看成是 sequence labelling 的问题，把 CNN 和 RNN 放在一个网络中进行端到端训练。

### 3.4.6 CRF-CNN joint model

文章题目《Deep Structured Output Learning for Unconstrained Text Recognition》，Max Jaderberg 等[49]。牛津大学工程科学系几何视觉组，ICLR 2015。

文章的 motivation 就是解决自然场景图片中 free-lexicon 字符串的识别。方法是将 CNN 与 CRF 相结合，以整张 word 图片作为输入。CRF 中的一项由 CNN 提供来预测每个位置的字符，高阶项由另一个 CNN 提供来检测 N 元文法 (N-gram) 的存在。这个模型 (CRF, 字符 predictor, N 元文法 predictor) 可以通过整体反向传播结构化输出损失来优化，本质上要求系统进行多任务学习，而训练仅要求生成综合数据。其中，CNN 文本识别模型中的字符序列模型如图 3-23 所示：

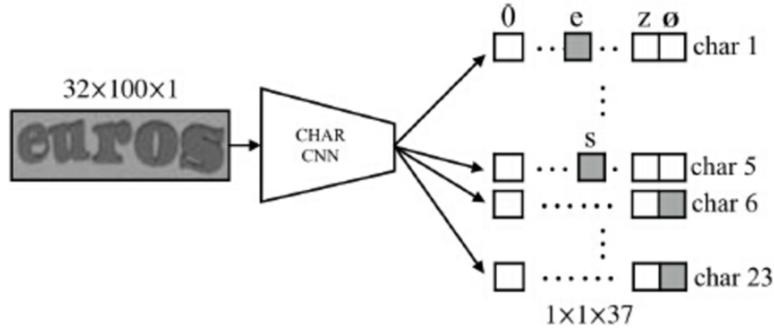


图 3-23 字符序列模型

其通过预测每个位置上的字符输出来识别一张 word image，一次拼写出字符。每个位置的分类器独自学习，但是共享一组联合优化的特征。

N 元文法编码模型示意图如图 3-24 所示：

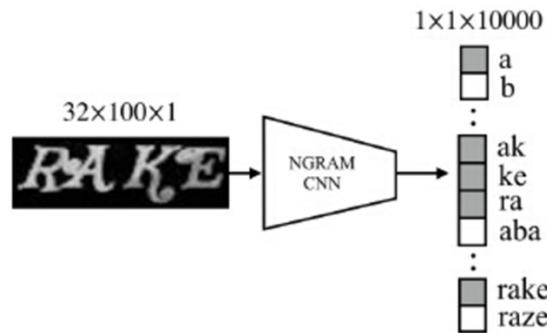


图 3-24 N 元文法编码模型

被识别的文本被表示为它的 N 元文法的组合 (bag-of-N-grams)。可以将其视为 10k 个使用共享联合学习特征集合的独立训练的二值分类器，训练用以检测某个特定 N 元文法的出现。

组合模型中 word camel 的路径 score  $S(\text{camel}, x)$  的构造说明示意图如图 3-25 所示：

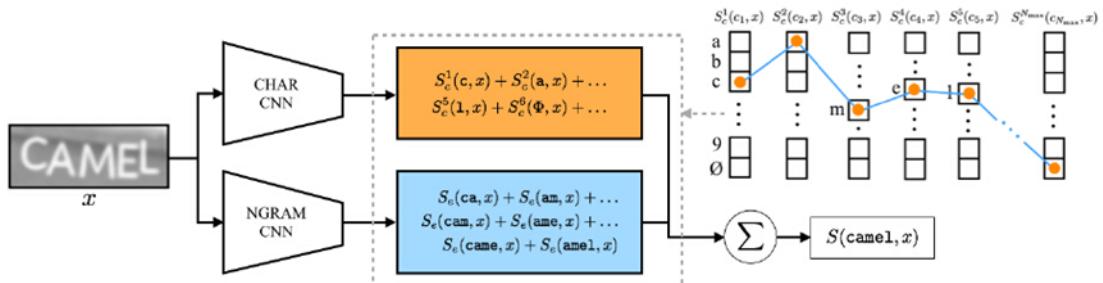


图 3-25 构造说明示意图

右上角所示为用于 score 的一元及边条目由穿过字符位置图的路径选择。这些条目的值， $S_c(c_i, x)$  和  $S_e(s, x)$ ，其中  $s \subset \omega$ ，由字符序列 CNN (CHAR CNN) 和 N 元文法编码 CNN (NGRAM CNN) 的输出给定。

联合模型的训练结构如图 3-26 所示：

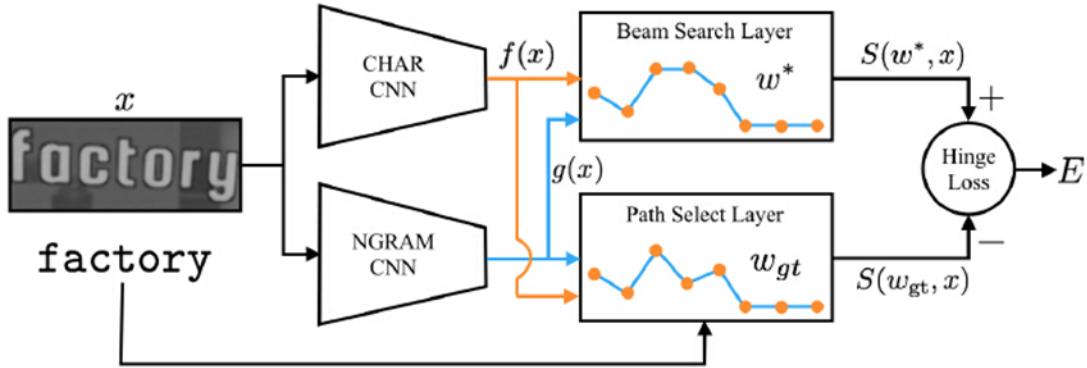


图 3-26 联合模型的训练结构

将字符序列模型 (CHAR) 和 N 元文法编码模型 (NGRAM) 与结构化输出损失相结合。路径选择层 (Path Select Layer) 通过对输入的 ground-truth word 求和来生成 score  $S(\omega_{gt}, x)$ 。定向搜索层 (Beam Search Layer) 通过定向搜索来从输入中选择最大的 score  $S(\omega^*, x)$ 。hinge loss 实现了一个 ranking loss, 限制最高得分路径为 ground-truth 路径, 可以反向传播到整个网络来联合学习所有参数。

贡献: (1) 提出了 CNN 和 CRF 的结合, 相对于仅仅对字符进行预测 (指不进行文法检测), 文中提出的模型在标准文本识别 benchmark 中更加准确。(2) 在 lexicon-constrained (有固定字典, 知道长度) 的情景中获得了 state-of-the-art 的准确率。

### 3.4.7 Embedded Attributes model

文章题目《Word Spotting and Recognition with Embedded Attributes》。Jon Almazan, Albert Gordo, Alicia Forn ‘ es, Ernest Valveny[51]。CVC, 法国国立计算机及自动化研究院, TPAMI2014。

文章的 motivation 就是解决 word spotting 和 word recognition。这里的 word spotting 是针对于剪切好的 word image 进行的, 可以理解为是 word retrieval (从字符图像集合中找到指定 word 的所有图像); word recognition 是基于带有 dictionary 或者 lexicon 的整个单词图像的识别 (不是那种一个字符一个字符识别的问题)。这里的 word spotting 和 recognition 都可以适用于扫描文档和自然图像。

方法: 提出 PHOC 描述子 (称为 label embedding) 来表征一个 word, 每个 word 都有一个唯一确定的 PHOC 描述子 (有 604 维度, 每一维度都是 {0,1})。基于 PHOC, 作者提出了一种通过分类器学习的方法, 用 PHOC 对图像进行编码, 让整张 word image 得到和类似 PHOC 的属性 (称为 attribute embedding)。这样的话就把 image 和 text string 映射到同一个空间上, 就可以进行距离运算了。

PHOC, 全称 pyramid histogram of characters 是对 text string 的字符统计, 示意图如图 3-27 所示:

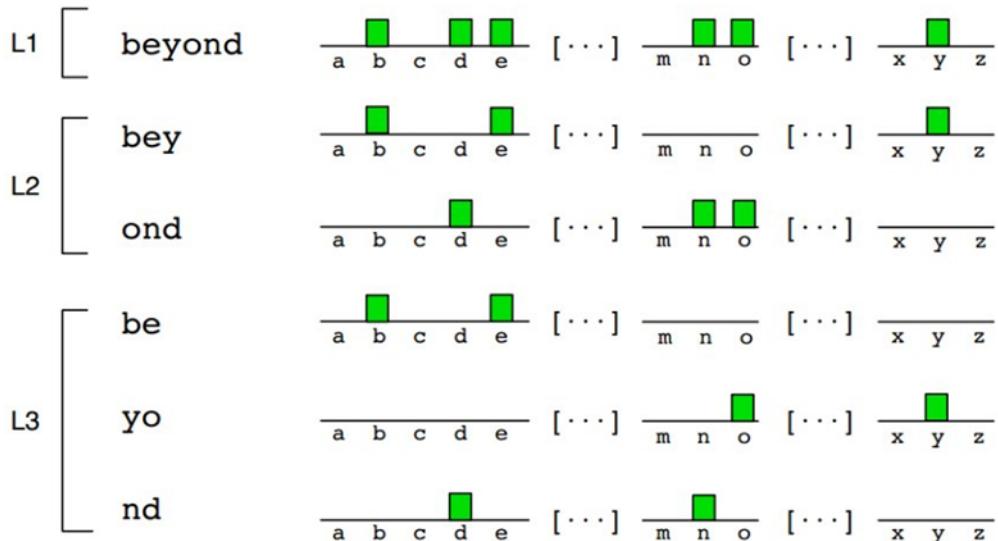


图 3-27 1、2、3 阶段的单词 PHOC 直方图

最后的 PHOC 直方图是这些局部直方图的连结。以 L-1 为例进行说明，给定一个 word (beyond)，统计其 36 个字符（英文字母 a~z，罗马数字 0~9）出现与否，若有出现则标记为 1 否则标记为 0，这样就得到了 36 维的字符描述。接着在 L-2 将 word 划分成两半 (bey 和 ond)，在这两半里像 L1 一样分别进行字符统计得到 2x36 的字符描述。L3, L4 依此类推。文章作者取 L2, L3, L4, L5，特征维度为  $(2+3+4+5) \times 36 = 504$ ，在 L2 基础上又增加了对 50 对双语出现与否的统计，所以最终的特征维度为  $504 + 2 \times 50 = 604$ 。

图像提取的特征是 Fisher Vector，所使用的分类器是 SVM（每个分类器学习一个维度的属性，共有 604 个 SVM）。由于 PHOC 的每一维度取值为 0 或者 1，表示某个字符出现与否。在学习图像嵌入属性时，我们可以将图像对应真实 word 的 PHOC 的每一维度看成嵌入属性相应维度上的 label。以下图 3-28 是第 i 维属性分类器的训练过程：

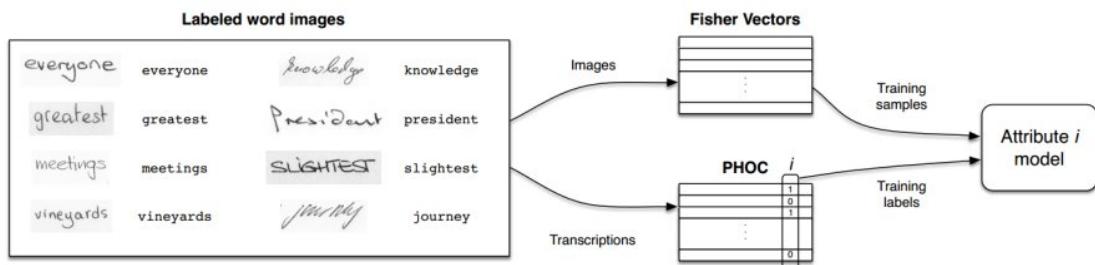


图 3-28 第 i 维属性分类器的训练过程。使用表征图像的 Fisher 矢量训练 SVM 分类器，以 PHOC 的 i 维值为标签

首先提取 word image 的 FV 作为分类器的输入样本，word image 对应的文本字符串所提取的 PHOC 描述子第 i 维上的值（0 或者 1）作为输入样本的 label 信息。用这些输入样本和 label 就可以训练得到一个 SVM 分类器，由这个分类器学习到的结果就是嵌入属性的第 i 维特征。

由于传统的 Fisher Vectors 是不包含空间信息的，作者在 sift 提取过程中还增加了空间信息。具体两种做法：

(1) Spatial pyramid：对图像进行空间划分成 k 个 regions，然后每个 region 提取 sift 特征，最终将 k 个 sift 链接在一起作为 fisher vector 的输入。

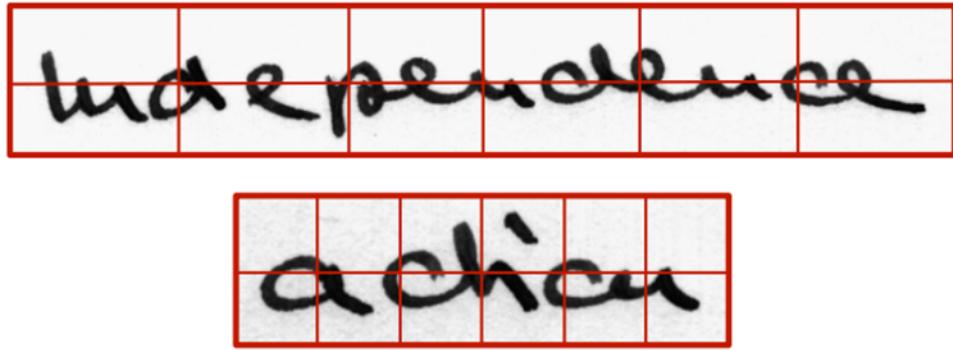


图 3-29 单词图像的空间金字塔。每个空间区域的尺寸和容量很大程度上取决于单词的长度

(2) 坐标定义法：给定一个 word image，先找到一个包含 95% 以上文字内容的最小 bounding box。然后将这个 bounding box 的坐标定义为(-0.5, 0.5)到(0.5, 0.5)之间，最后整张图片根据这个 bounding box 来调整坐标信息，提取 sift 的时候把空间坐标信息加进去。

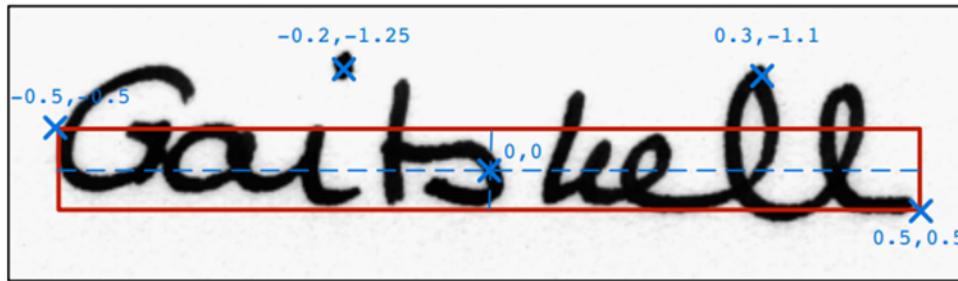


图 3-30 单词图像和定义坐标系的自动调整参考框

实验证明第二种做法好一些。

作者还提出了一系列归一化和属性降维方法。文章中将嵌入属性和 PHOC 都降维到一个共同的子空间上，大致流程如图 3-31：

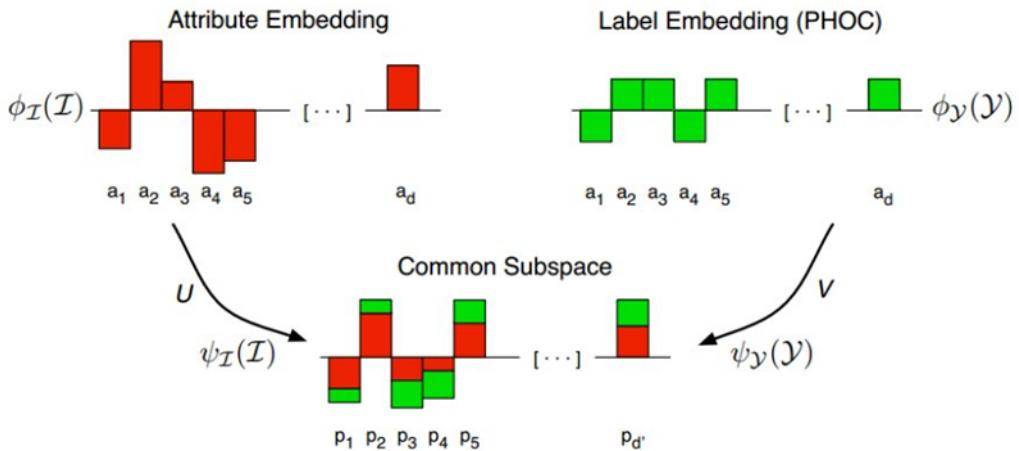


图 3-31 使用 CSR 将预测的属性分数和 ground truth 属性投影到更加相关的子空间  
综上所述，文章的整体思路如图 3-32 所示：

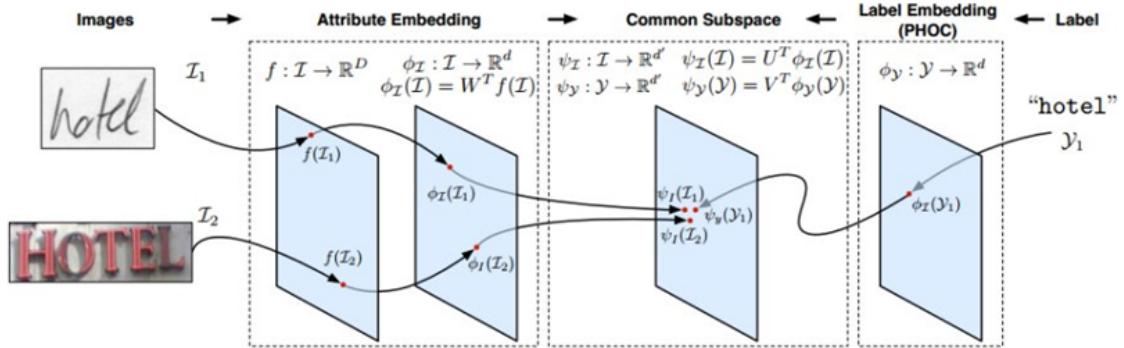


图 3-32 文章的整体框架

对于 word recognition 任务，从右到左表示，给定 lexicon，先算出每个 lexicon 上 word 的 PHOC 表示，然后映射到 common subspace 上。从左边到右边表示，给定一个待识别的 word image 先提取 feature，通过分类器学习到 attributes embedding，然后再把嵌入属性映射到 common subspace 上。最终在 common subspace 上找到和嵌入属性距离最近的 PHOC 对应的 word 就识别的结果。Word spotting 任务类似。

贡献：(1) 提出 PHOC 描述子（称为 label embedding）来表征一个 word 的方法。(2) 基于 PHOC，作者提出了一种通过分类器学习的方法。

### 3.4.8 Text-deeplab model

文章题目《基于语义分割技术的任意方向文字识别》。王涛，江加和[52]。[北京航空航天大学自动化科学与电气工程学院，《应用科技》2018。](#)

文章的 motivation 就是针对现有文本检测与定位方法只能处理单一方向文本行的缺点，提出了一种基于语义分割方法的用于自然图像中文本检测的新方法。

方法：首先通过对现有检测方法以及目前语义分割方法在文本行检测中的局限性分析。然后对加入矩形卷积核的全卷积网络模型进行训练，获得文本行区域的分类图。最后，通过全连接条件随机场(conditional random field, CRF) 的高精度分割能力将网络前端输出的文本行区域中的文字给区分出来。该框架用于处理任意方向、语言和字体中的文本。网络结构如图 3-33 所示：

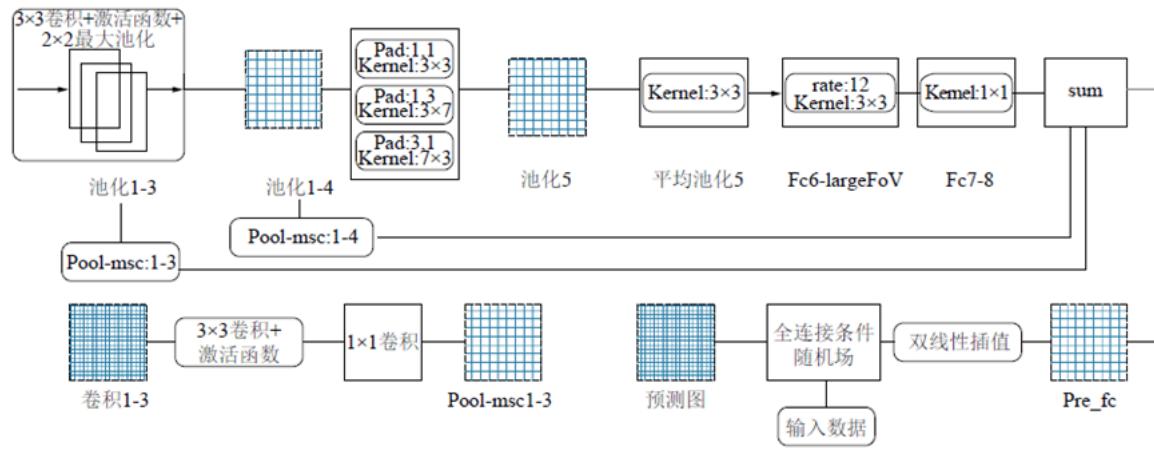


图 3-33 Text-deeplab 网络结构

贡献：(1) 提出了一种基于语义分割方法的用于自然图像中文本检测的新方法；(2) 所提出的方法在 MSRA-TD500 和 ICDAR2015 两个文本检测数据集上获得良好的分割结果且性能优越。

## 3.5 实验室现有算法介绍

### 3.5.1 算法介绍

本次比赛中，我们研究了网络图像的文本识别任务，采用的是 Convolutional Recurrent Neural Network(CRNN)[53]的方法，该方法将文本识别任务看成是 sequence recognition 问题，网络结构由三部分组成，包括 Convolutional Layers，Recurrent Layers，Transcription Layer。

其中，Convolutional Layers 的输入是 crop 好的文本图片，卷积神经网络 CNN 从图片中提取 feature maps，最后用 Map-to-Sequence 把 CNN 最后一层的 feature map 上的每个滑动窗口连接起来组成 feature sequence，这是 Convolutional Layers 的输出。Recurrent Layers 的输入是 Convolutional Layers 输出的 sequence of featre vectors，也就是  $X = x_1, x_2, \dots, x_t$ ，输出是 feature sequence 里每个  $x_t$  的 label 分布预测值  $y_t$ ，该层可以捕捉序列内的上下文信息。Transcription Layer 对结果进行调整，采用 CTC 方法[54]将 RNN 输出的每个  $x_t$  的预测值转换成 label sequence，也就是找到具有最高概率的标签序列作为最后的预测值。

该方法将文本识别当作是序列识别的问题，不需要对字符进行切割，可以识别任意长度的序列，并且充分利用了上下文信息。虽然 CRNN 由不同类型的网络架构(如 CNN 和 RNN)组成，但可以通过一个损失函数进行联合训练。

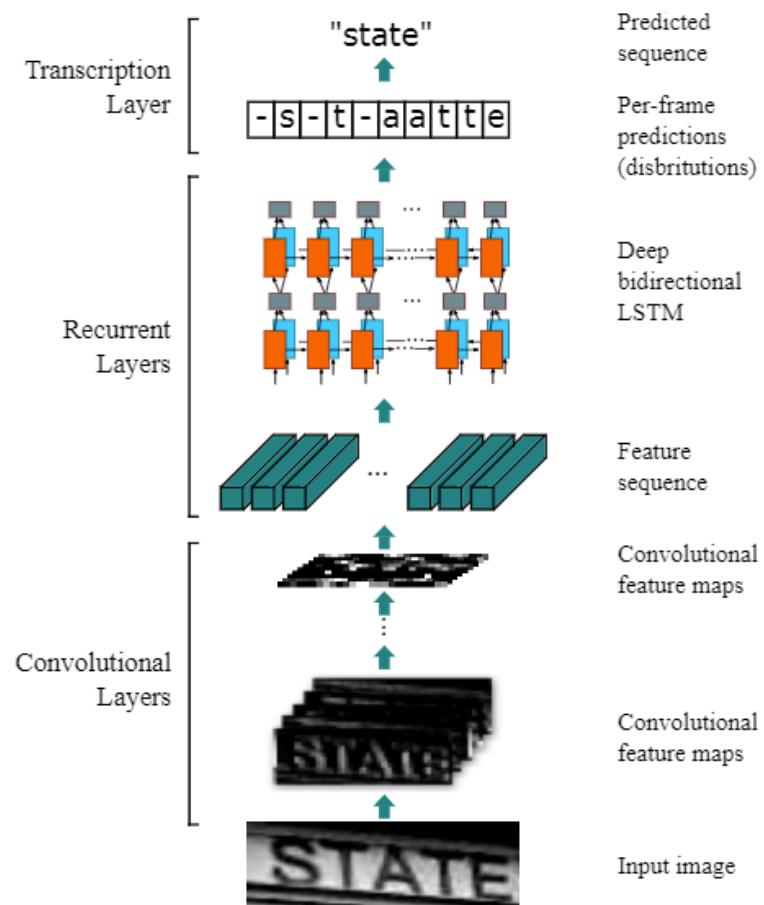


图 3-34 CRNN 模型架构。架构包括三部分：1) 卷积层，从输入图像中提取特征序列；2) 循环层，预测每一帧的标签分布；3) 转录层，将每一帧的预测变为最终的标签序列。

### 3.5.2 模型结构

本算法的总体模型架构为 CNN+RNN+CTC：

CNN 尝试了 defaultCNN, ResNet 和 DenseNet, 最后采用了实验效果最好的 ResNet50[55]模型, RNN 采用了 Bidirectional LSTM[56]模型, CTC 选用的 warpCTC。

### (1) 特征序列提取 (Feature Sequence Extraction)

在 CRNN 模型中, 通过采用标准 CNN 模型 (去除全连接层) 中的卷积层和最大池化层来构造卷积层的组件。这样的组件用于从输入图像中提取序列特征表示。在进入网络之前, 所有的图像需要缩放到相同的高度。然后从卷积层组件产生的特征图中提取特征向量序列, 这些特征向量序列将作为 RNN 层的输入。具体地, 特征序列的每一个特征向量在 feature map 上按列从左到右生成。这意味着第  $i$  个特征向量是所有 feature map 第  $i$  列的连接。在我们的设置中每列的宽度固定为单个像素。

由于卷积层、最大池化层和激活函数在局部区域上执行, 因此它们是平移不变的。因此, feature map 的每列对应于原始图像的一个矩形区域 (称为感受野)。这些矩形区域与 feature map 上从左到右的相应列具有相同的顺序。如图 3-35 所示, 特征序列中的每个向量关联一个感受野, 并且可以被认为是该区域的图像描述符。

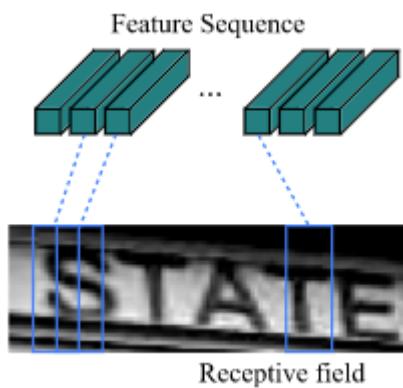


图 3-35 感受野

### (2) 序列标注 (Sequence Labeling)

一个深度双向循环神经网络建立在卷积层的顶部, 作为循环层, 即 RNN 层。RNN 预测特征序列  $x=x_1, \dots, x_T$  中每一帧  $x_t$  的标签分布  $y_t$ 。

使用 RNN 有三个优点。首先, RNN 具有很强的捕获序列内上下文信息的能力。对于基于图像的序列识别使用上下文提示比独立处理每个符号更稳定且更有帮助。以场景文本识别为例, 宽字符可能需要一些连续的帧来完整地描述 (参见图 3-35)。此外, 一些模糊的字符在观察其上下文时更容易区分, 例如, 通过对比字符高度更容易识别“il”而不是分别识别它们中的每一个。其次, RNN 可以将误差差值反向传播到其输入, 即卷积层, 从而允许我们在统一的网络中共同训练循环层和卷积层。第三, RNN 能够从头到尾对任意长度的序列进行操作。

传统的 RNN 单元在其输入和输出层之间具有自连接的隐藏层。每次接收到序列中的帧  $x_t$  时, 它将使用非线性函数来更新其内部状态  $h_t$ , 该非线性函数同时接收当前输入  $x_t$  和过去状态  $h_{t-1}$  作为其输入:  $h_t = g(x_t, h_{t-1})$ 。那么预测  $y_t$  是基于  $h_t$  的。以这种方式, 过去的上下文  $\{x_t'\}_{t'<t}$  被捕获并用于预测。然而, 传统的 RNN 单元有梯度消失的问题, 这限制了其可以存储的上下文范围, 并给其训练过程增加了负担。长短时记忆 (LSTM) 是一种专门设计用于解决这个问题的 RNN 单元。LSTM (图 3-36 所示) 由一个存储单元和三个多重门组成, 即输入, 输出和遗忘门。存储单元存储过去的上下文, 输入和输出门允许一个单元长时间地存储上下文。同时, 单元中的存储可以被遗忘门清除。LSTM 的特殊设计允许它捕获长距离依赖。

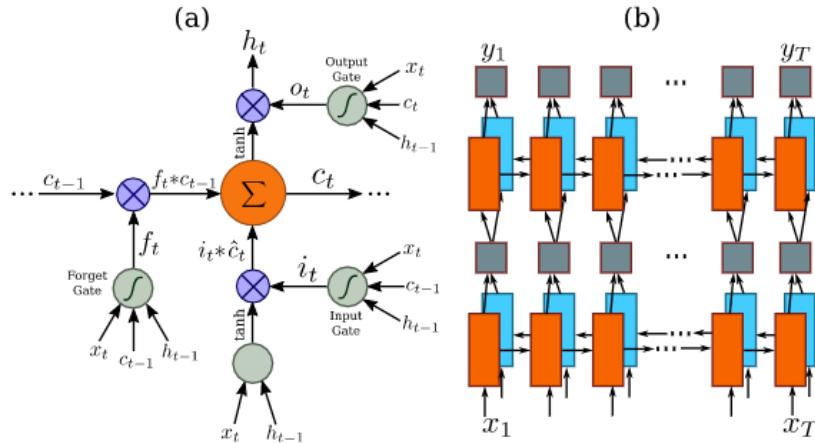


图 3-36 (a) 基本 LSTM 单元的结构。LSTM 由一个单元模块和三个门组成，即输入门，输出门和忘记门。(b) 本文中使用的深双向 LSTM 的结构。

### (3) 转录 (Transcription)

转录是将 RNN 所做的每帧预测转换成标签序列的过程。数学上，转录是根据每帧预测找到具有最高概率的标签序列。在实践中，存在两种转录模式，即 lexicon-free 转录和 lexicon-based 的转录。在 lexicon-free 模式中，预测时没有任何词典。在 lexicon-based 的模式中，通过选择具有最高概率的标签序列进行预测。

我们采用联接时间分类 (CTC) 层中定义的条件概率。按照每帧预测  $y=y_1, \dots, y_T$  对标签序列  $\mathbf{y}$  定义概率，并忽略  $\mathbf{y}$  中每个标签所在的位置。因此，当我们使用这种概率的负对数似然作为训练网络的目标函数时，我们只需要图像及其相应的标签序列，避免了标注单个字符位置的劳动。

条件概率的公式简要描述如下：输入是序列  $\mathbf{y}=y_1, \dots, y_T$ ，其中  $T$  是序列长度。这里，每个  $y_t \in \mathcal{R}|\mathcal{L}'|$  是在集合  $\mathcal{L}'=\mathcal{L} \cup \{\text{blank}\}$  上的概率分布，其中  $\mathcal{L}$  包含了任务中的所有标签（例如，所有英文字母），以及由“-”表示的“空白”标签。sequence-to-sequence 的映射函数  $B$  定义在序列  $\pi \in \mathcal{L}'^T$  上，其中  $T$  是长度。 $B$  将  $\pi$  映射到  $\mathbf{y}$  上，首先删除重复的标签，然后删除 blank。例如， $B$  将“-hh-e-l-ll-oo-” (-表示 blank) 映射到 “hello”。然后，条件概率被定义为由  $B$  映射到  $\mathbf{y}$  上的所有  $\pi$  的概率之和：

$$p(\mathbf{l}|\mathbf{y}) = \sum_{\pi: B(\pi)=\mathbf{y}} p(\pi) \quad (\text{公式 3-6})$$

CNN 网络结构的选取跟数据集的分布有关，实验结果表明，当数据集中的图片较少时（十万级），defauleCNN 的实验效果最好。当数据集较大时(百万级)，ResNet 的实验结果比较好，并且用在 ImageNet 数据集上预训练好的模型在数据集上 finetune，实验结果会有很好的提升。

## 3.6 CRNN 字符识别实验

### 3.6.1 数据生成

#### 3.6.1.1 实验原理

数据生成实验的理论支撑基于论文《Synthetic Data for Text Localisation in Natural Images》[57]和《Poisson Image Editing》[58]，目前该论文的代码已经开源。

基本原理即先确定前景和背景。前景即我们选定的文本，背景即需要贴上字的背景图片。首先确定文字的位置和方向，然后文本会被分配一种颜色。论文中，文本的调色是从 IIIT5K 单词数据集[59]中裁剪后的单词图像中学习的。使用 K-means 将每个裁剪后的单词图像中的像素划分为两组，其中一个颜色接近前景（文本）颜色，另一个接近背景颜色。渲染文字时，

选择背景色与目标图像区域最匹配的颜色对（使用 Labcolour 空间中的 L2 范数），并使用相应的前景色来渲染文本。大约 20% 的文本会被随机选择为有边框，这个参数是可以在代码中调整。边框颜色的选择为可以与前景色相同，也可以设置成前景色和背景色的平均值。

为了保持合成文本图像中的亮度梯度，我们使用 Poisson 图像编辑方法将文本混合到基本图像上，基本原理由公式 (3-7) 定义。

$$\text{for all } \mathbf{x} \in \Omega, v(\mathbf{x}) = \begin{cases} \nabla f^*(\mathbf{x}) & \text{if } |\nabla f^*(\mathbf{x})| > |\nabla g(\mathbf{x})|, \\ \nabla g(\mathbf{x}) & \text{otherwise.} \end{cases} \quad \text{公式 (3-7)}$$

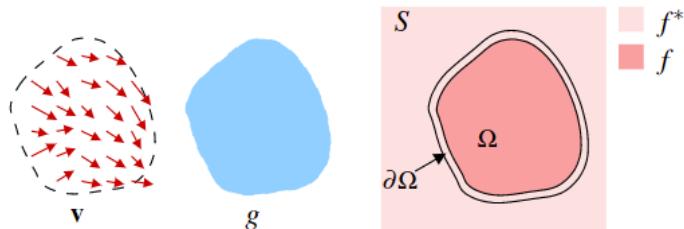
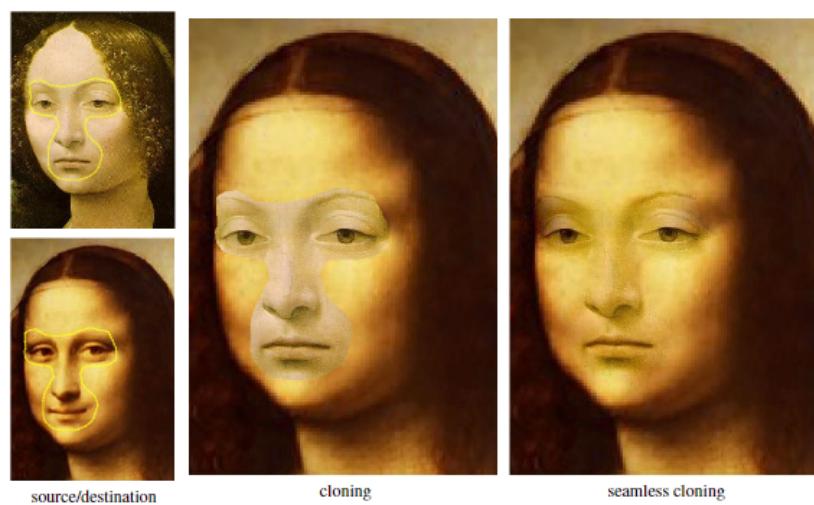


图 3-37 Poisson 融合原理图

通过 Poisson 融合处理之后的图像，前景能较好的与背景融合，达到十分逼真的效果。下图是一些论文中的实验示例图：



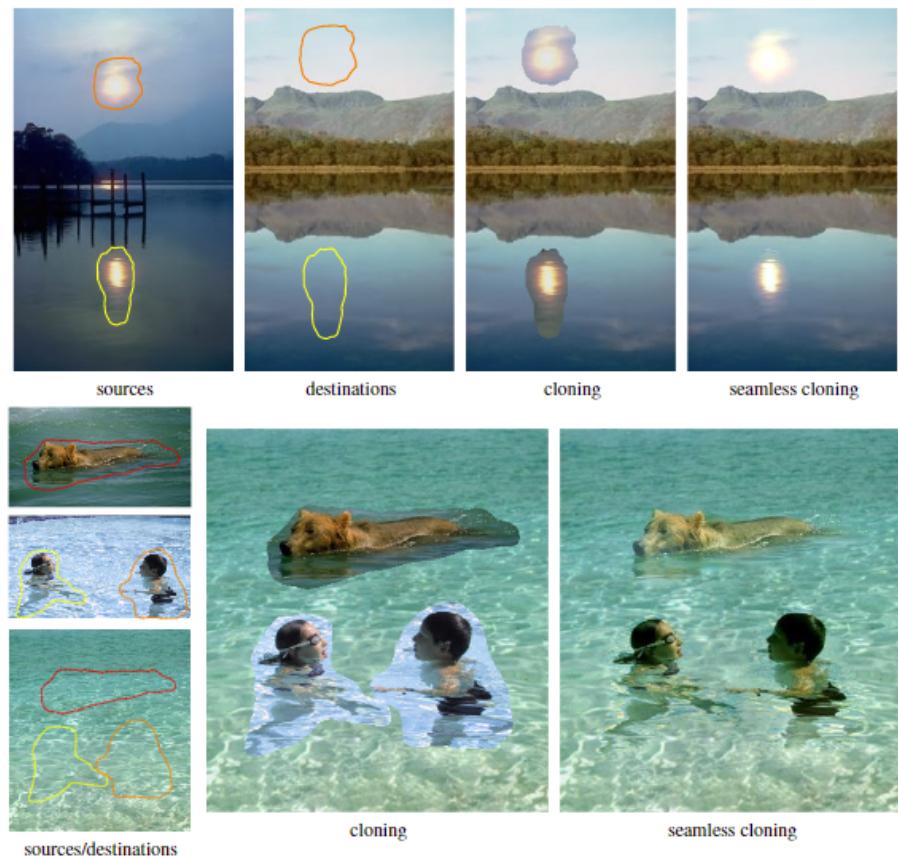


图 3-38 Poisson 融合实验效果

论文《Synthetic Data for Text Localisation in Natural Images》在 Poisson 图像编辑的基础上，进行了英文贴字的实验。效果如图所示：



图 3-39 英文文本贴图实验效果

由于该论文的代码仅仅限于英文文本的贴字实验，所以我们在该份代码的基础上修改，修改之后可进行中英文贴字，也因此可以用于本次竞赛的数据生成。

### 3.6.1.2 实验过程

针对 ICPR 竞赛，一共生成了 5 个版本的人造数据。每个版本都对上一版本生成结果分析统计之后有所改进。

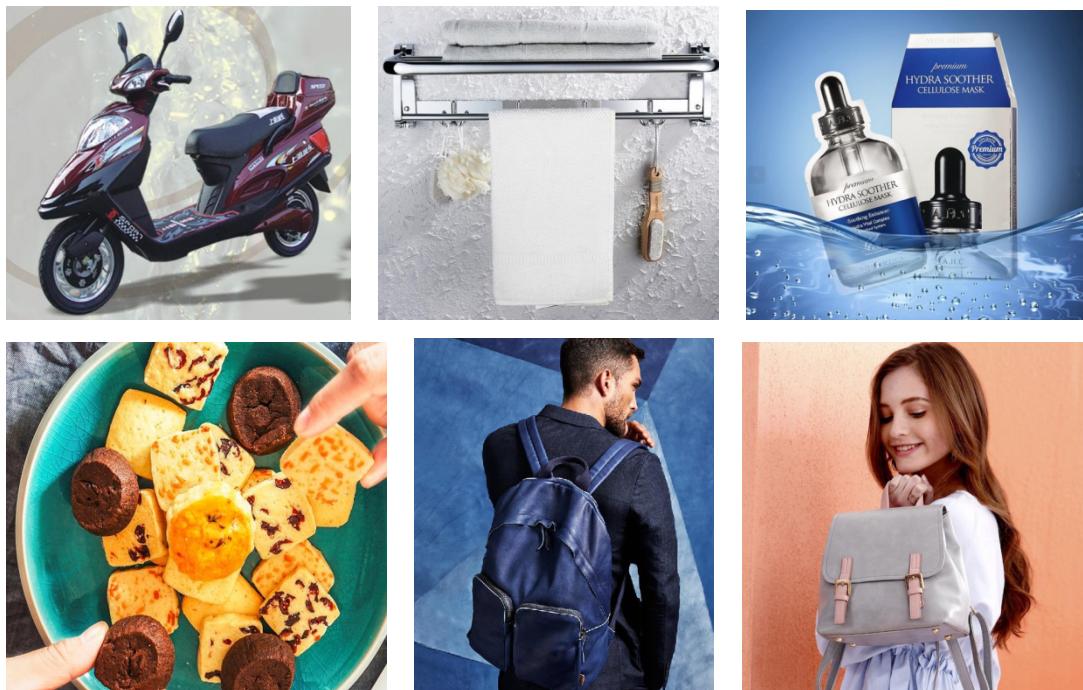
### (1) 实验 1

#### 1) 实验介绍

第一个版本的实验是初次尝试，在用 CRNN 训练了官方提供的 training dataset 之后，发现 baseline 的效果并不是很理想，所以有必要自己生成数据加进去训练。我们在统计了比赛官方所提供的 groundtruth 包含的所有类之后，生成了一份 key.py，这个版本的前景（文本），即要生成的 label 就是从中提取。

#### 2) 背景展示

背景一共挑选了 200 张来自淘宝网页的图片，涵盖了衣食住行各个方面，目的是尽量模拟官方数据集的图片形式。



#### 3) 字体示例

我们挑选了 10 种风格迥异的字体，有印刷体也有手写体，以此来模拟数据集中各种不同风格的字体形式。

**Innovation in China 中国智造**

# Innovation in China 中国智造

## Innovation in China 中国智造

### 4) txt 示例

对于要生成的前景（文本），一共生成了 10000 个 txt，每个 txt 包含 20 行记录，每个记录包含 10 个随机字符，即一共是 20 万的数据量。这些字符都是从统计的 groundtruth 的所有 5543 类中随机挑选的，不包括任何语义信息。

潔芭何鸣乌烹關蛻纪茂  
璇噜[君号惊輯患害k  
C歌炯际琦筝涤遠之R  
嬪追悲喔獮傻綫冰漸棚  
治换祁箱主意紊罕題\  
敲：根盲琴个壺】預蠟  
眠寡判聖注惡翁资片涨|  
于 g 旗窝訂某飘留掌掠  
k序菊周換湿卡抬蓝郑  
參呴緣冶咲I書泰娣晒  
v君 c 漑備兴囂拒玲@  
拿趣侦履泷嗓覆伪碑勿  
最俗燶閨沐產薪哲哺貢  
羽须兑咿笋膳啥弱剪联  
倾审蛻莽F架拽褂G夷  
沈本锵酱Ⅲ參話疲络逛  
1传補虫n镁植貌量n  
焰萃看沱蕴根總塔铠見  
宿賦种疑航膠涂您闻鸥  
!崖钥售丹饲翥岑对戰

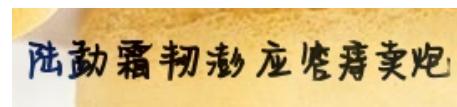
弱萊w板尖真鴻拆腋殊  
廓勝訣呛含玺钥煤輸3  
八坊①閉哄蚊巴肿笑終  
洲朱添衬娘烽傳毡洼碑  
唾摘船汙洛庙原鯨柚衆  
椎F女图辈旅湯熏窍錄  
F潮樊鄉待北钞穗“奢  
桐族尿过割n醚c 英A  
紐卵 r 终.郭集虹作段  
乱 I 放珀时弟逊极浮僧  
-洞}漆绚互 F 盆歷榜  
韵舵琢丢闯肺H御收頰  
荟+潔數天龍股摘檀證  
⑧段額和肚道父烹舍一  
瑰蕊诉途VII琦缥趋庚貝  
咧朦抱脖羸榔励豆禦覺  
笙凶憨宋压奪唤鹿瘤療  
死記拿宗箔俩咯蚊钦搖  
豐酱累氨伎饱镍塊測勢  
汪楞联啤廣卵锣器弯哇

說堪 ' 打勁錦淵查析娴  
塑誤辛腩烨南仇《灯販  
壁+兽咱盡流 ' 岳針冀  
S 祥鋼 II 断温鞭曉兆隨  
耻C虐璋時嗽寧絮樣澜  
賭b僵橹陆駕妈骆芪极  
狄ü爱火小鬼核节均慮  
=子坝司丙魔I礦碗豆  
瘤阱攻模舵江标颐瓶教  
唱療` 弗廉雾禪嘞药然  
屹殿娴鍾銀炖万软」 挠  
蒙凜際雀尿廠按锅跃巧  
鴨繁琴神柠暑臥吵秉瞞  
儻曇媳嫩泊村使夹两矶  
抢兰β猴獄匪家枯較}  
貉轻蛩濕卸讲皖琼國鸟  
硕各漯VII潮梧嘞大亡咯  
镂軒移榉塗殼g汕蝶錫  
璋近茧箔樟婆歲伽芝XI  
这師快攬摯众 =氣葫恍

### 5) 结果展示

实验 1 一共生成  $10000 \times 20 = 20$  万张 crop 后的图片，基本效果较差，存在大量的漏字、模糊、编码出错的情况。

#### ◆ 生成较好：



#### ◆ 生成残缺：



#### ◆ 模糊：



- ◆ 编码出错:



#### 6) 结果分析

从实验效果来看,由于大部分字体都是从网上下载的手写字体,存在着字体类别的缺失,字库不全,导致无法生成 groundtruth 中提取的 label。同时,该版本的生成数据还存在着图片 padding 过大的情况,不利于送入 CRNN 网络训练,需要调整。

#### (2) 实验 2.1

##### 1) 实验介绍

通过分析 CRNN 的模型结构,发现如果数据的 label 能够带语义,更有利于模型的训练,所以实验 2.1 直接选取了原数据集中的 groundtruth 作为生成数据集的 label,共生成 125798 张 crop 之后的图片。背景图片的选择同实验 1 中的设置。

##### 2) 字体选择

通过实验 1 的分析,发现一些手写体存在字库不全,类别缺失的情况,所以该实验的字体挑选了 5 种常见的广告体,这些字体广泛应用于各种广告、商标,而且字库类别比较全面,可以基本解决文本字体无法生成的情况。

黑体: **Innovation in China 中国智造**

隶书: *Innovation in China 中国智造*

楷体: *Innovation in China 中国智造*

宋体: Innovation in China 中国智造

汉仪菱心体: **Innovation in China 中国智造**

##### 3) txt 示例

label 取自 groundtruth,考虑到 CRNN 模型的 batchsize 大小,字符长度定义为<=10(后续实验均与此设置相同)。

薄利汽配  
五菱荣光前外拉手颜色齐全  
80ml  
\*小宅屋\*专用盗图可耻  
DERMATOLOGISTTESTED  
-臉部·身體均通用-  
100%物理性防曬適合敏感肌膚使用  
温和全護  
輕透防曬乳  
露得清  
<http://xiaozhaiwu.taobao.com>  
PA+++  
broadspectrumuva+uvb  
PUreSCREEN



薄利汽配  
五菱荣光前外拉手颜色  
80ml  
\*小宅屋\*专用盗图可  
DERMATOLOG  
-臉部·身體均通用-  
100%物理性防曬適  
温和全護  
輕透防曬乳  
露得清  
[http://xia  
PA+++  
broadspect  
PUreSCREEN](http://xiaozhaiwu.taobao.com)

#### 4) 结果展示:

通过以上实验设置的调整,取得了初步的生成效果。

◆ 汉字生成

◆ 数字生成

◆ 字母生成

5) 结果分析

由生成效果可以看出，彩色背景下的黑色字体可以大部分较好的生成，包括一些特殊字符，如“【】”、“③”等也可以生成。所以这一版本的 12 万数据很快用于我们 CRNN 的训练当中。

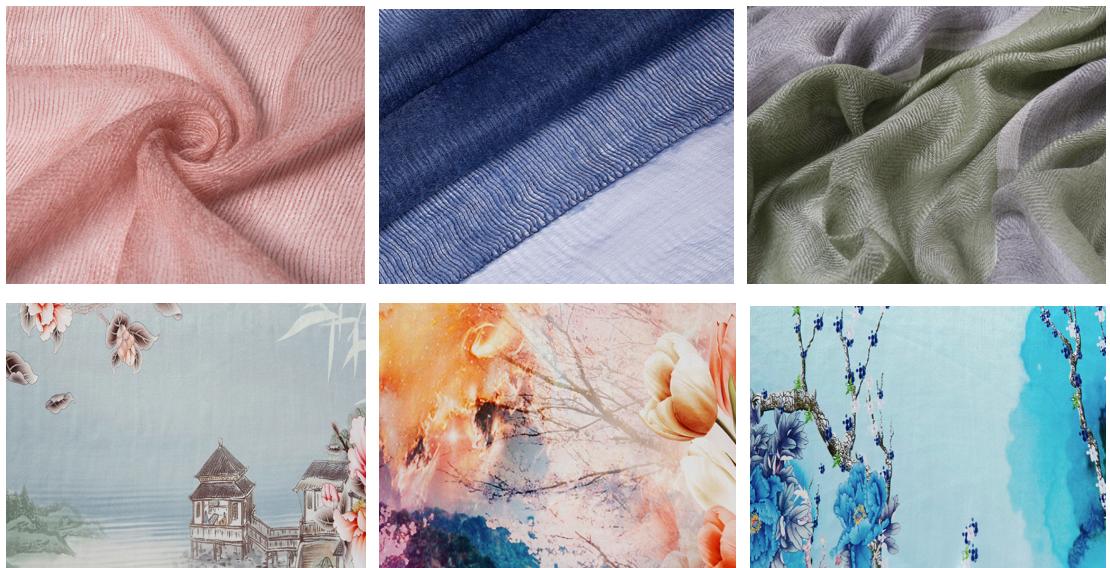
(3) 实验 2.2

1) 实验介绍

该版本实验的 label 依旧取自 groundtruth，字体样式也未改变。但是我们对背景做了重大的调整，选取的是带多种纹理的背景，共生成 125796 张 crop 之后的图片。

2) 背景展示

我们在淘宝网页上又挑选了 200 张具有特殊纹理的图片当作背景，目的是减少极度复杂背景下，背景干扰字体生成的问题，这种情况下的生成图片，往往难以区分背景和文本。而这些具有纹理的图片，能够更好的减少这种干扰。



3) 结果展示

◆ 汉字生成

**时尚女装**

2012女装部分款

**(1) 5CM加厚坐垫**

**批准文号】国药准字**

◆ 数字生成

**73/74/737**

**手机:1515852**

**73774 / 7377**

**QQ:2024672**

◆ 字母生成

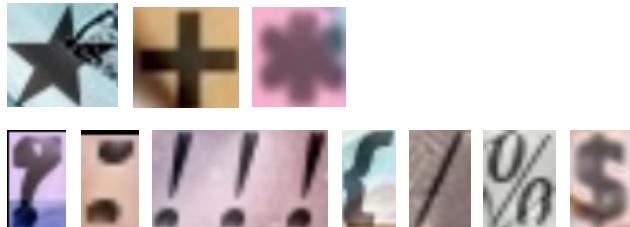
**ANALOG**

**Example**

**WELCOME**

**www.nfchom**

◆ 其他:



#### 4) 结果分析

这一版本的生成数据，对比上一个版本，大大减少了复杂背景对生成文本的干扰，同时文本与背景有较强的区分度，取得了不错的实验效果。一些特殊字符也可以很好的生成，如★、\*、\$等各种符号。

#### (4) 实验 3

##### 1) 实验介绍

实验 3 的背景、字体配置都同实验 2，但是对于生成的文本，我们做了一些改变。不再局限于原数据集的 groundtruth，而是选取了新闻及广告语的片段作为生成数据集的 label。

##### 2) txt 展示

通过分析 CRNN 的实验效果，发现我们需要更多具有语义信息的 label，以增强训练效果，也即不再让 CRNN 局限地学 groundtruth 的 label，而是扩充它的语料库。所以我们从网上搜集了几千条广告语，又拿到了一部分新闻素材，从这两部分中，我们随机截取了 100 万条字符长度 $\leq 10$  的 label（原因已经解释，不再赘述），最后实际生成 999482 张 crop 后的图片。

多一些润滑少一些摩擦(统一润滑油)
我的地盘听我的(中国移动通信动感地带)
我就喜欢(麦当劳)
只要你想(联想)
帕萨特，成就明天(帕萨特系列)
不同滋味不同心情(蒙牛心情二次方牛奶)
男人就应该对自己狠一点(柒牌服饰)
热爱生活冷静选择(奥克斯空调)
喝前摇一摇(农夫山泉农夫果园果汁)
煮酒论英雄才子赢天下(才子男装)
本世纪百佳广告策划
德国大众：“小即是好。”
可口可乐：“享受清新一刻。”
万宝路香烟：“万宝路的男人。”
耐克：“说做就做。”
麦当劳：“你理应休息一天。”
迪比尔斯：“钻石恒久远，一颗永留传。”
通用电气：“GE带来美好生活。”
米勒牌淡啤酒：“美妙口味不可言传。”
克莱罗染发水：“她用了？她没用？”
艾维斯：“我们正在努力。”
美国联邦快递公司：“快腿勤务员。”
苹果电脑：“1984年。”
阿尔卡-舒尔茨公司：“多种广告”。
百事可乐：“百事，正对口味。”
麦氏咖啡：“滴滴香浓，意犹未尽。”
象牙香皂：“99和44/100%纯粹。”
美国捷运公司：“你知道我吗？”
美国征兵署：“成为一个全材。”



中国移动通信动感地带
我就喜欢(麦当劳)
只要你想(联想)
成就明天(帕萨特系列)
心情(蒙牛心情二次方
本世纪百佳广告策划
可口可乐：“享受清新
耐克：“说做就做。”
勒牌淡啤酒：“美妙口
果电脑：“1984年
茨公司：“多种广告”
麦氏咖啡：“滴滴香浓
美国捷运公司：“你知
百事可乐：“新一代的
：“穿哈斯维的男人。
署：“头戴“冒烟”字
：“这百威是给你的。
FORM：“我梦想穿
机器公司：“大师级的
：“大拉拉米尔以西的
CUITS'BOY
“土星”系列：“不一
只溶在口，不溶在手。
我的Calvins之
烟草：“云丝顿，好烟
笑我，直到我开始弹起
驼香烟：“为了买这包
溶液：“永远是女嫔相

月 12 日电 特约记者  
三十二强。衡阳科维奇  
他单独谈话，问他临门  
完全是处于足球的原因  
季的第六枚金牌。赛后  
是秋衣），我一预感克  
京时间 3 月 14 日，W  
尔的比赛里我们缺失了  
市，7 岁开始学习花样  
尼日利亚国家队进行两  
夺战造势。李娜身披五  
蒂斯塔正式上任，并且  
亚军衡 1998 / 20  
是“最美奥运冠军”，  
牌，巧合的是，当时保  
薪合同加盟骑士队，在  
了特殊待遇。警官刘易  
后赛第 10 次三双，但  
后，卡佩罗下课，杰拉  
连胜的节点实在不是时  
幕，来自中国广州的冯  
博斯基有望首发，而卡  
80 生活播报汽车主编  
使消费者推迟购买。8  
滤的费用大概在 400



中广网唐山 6 月 12 日消息（记者汤一亮 庄胜春）据中国之声《新闻晚高峰》报道，今天（12 日）上午，公安机关 2012 年缉枪制爆专项行动“统一销毁非法枪爆物品活动”在河北唐山正式启动！10 万余支非法枪支、250 余吨炸药在全国 150 个城市被统一销毁。衡黄明：现在我宣布，全国缉枪制爆统一销毁行动开始！衡随着公安部副部长黄明一声令下，大量仿制式枪以及猎枪、火药枪、气枪在全国各指定场所，250 余吨炸药被分别销毁。公安部治安局局长刘绍武介绍，这次销毁的非法枪支来源于三个方面，衡刘绍武：打击破案包括涉黑、涉恶的团伙犯罪、毒品犯罪，还有从境外非法走私的枪支爆炸物。衡在销毁现场，记者看到了被追缴和上缴的各式各样的枪支。衡刘绍武：也包括制式枪，有的是军用枪，仿制的制式枪，还有猎枪、私制的火药枪等。按照我国的枪支管看法，这些都是严厉禁止个人非法持有的。中国是世界上持枪犯罪的犯罪率最低的国家之一。衡中美联手破获特大跨国走私武器弹药案近日，中美执法部门联手成功破获特大跨国走私武器弹药案，在中国抓获犯罪嫌疑人 23 名，缴获各类枪支 93 支、子弹 5 万余发及大量枪支配件。在美国抓获犯罪嫌疑人 3 名，缴获各类枪支 12 支。这是公安部与美国移民海关执法局通过联合调查方式侦破重大跨国案件的又一成功案例。衡 2011 年 8 月 25 日，上海浦东国际机场海关在对美国纽约发往浙江台州，申报品名为扩音器（音箱）的快件进行查验时，发现货物内藏有手枪 9 支，枪支配件 9 件，长枪部件 7 件。经检验，这些都是具有杀伤力的制式枪支及其配件。这引起了公安部和海关总署的高度重视。衡公安部刑侦局副局长刘安成：因为是从海关进口的货物中检查出来夹带，说明来源地是境外，或是说国外，这应该是一起特大跨国走私武器弹药的案件。衡上海市公安局和上海海关缉私局成立联合专案组，迅速开展案件侦查。专案组于 8 月 26 日在浙江台州 UPS 取件处将犯罪嫌疑人王庭（男，32 岁，台州市人）抓获。王庭交代，他通过一境外网站上认识了上家林志富，2009 年 11 月以来，林志富长期居住美国，他通过互联网组建了一个走私、贩卖、私藏枪支弹药的群体，通过网络在国内寻找枪支弹药买家，并通过美国 UPS 联邦速递公司将枪支弹药从纽约快递给多名类似王庭的中间人，再通过中间人发送给国内买家。衡此案中，犯罪分子依托虚拟网络进行犯罪交易，隐蔽性强，涉案人员使用的身份、地址、联系方式都是虚构的，侦查难度很大。刘安成说，此案体现了新型犯罪，特别是现代犯罪的新特点。衡刘安成：他不受距离的限制、经常是跨国跨境，甚至是跨一个、数个、甚至数十个国家。这种犯罪手法的改变和新型犯罪的特点，要求我们各国警方充分合作。衡作者：汤一亮 庄胜春

### 3) 结果展示



### 4) 结果分析

在随后的实验中，我们将这 100 万的新数据加入 CRNN 训练，发现效果并未有太大提升。考虑原因，分析认为字体变化的颜色不够丰富，导致 CRNN 只能局限的学习黑色字体，所以在下一个实验中，我们重新对字体做了设置。

#### （5）实验 4.1

实验 4 的主要目的有两个，一是为了验证我们生成的数据是否真的对模型的训练提升了效果，二是语义信息是否对模型精确度有影响。所以，我们做了 2 个对比实验。其中，我们做出的重大调整就是将原来的黑色字体转变为彩色字体。

#### 1) 实验介绍

为了验证语义信息的影响，实验 4.1 选取了原数据集中的 groundtruth 作为生成数据集的 label，字符长度 $\leq 10$ ，共生成 317412 张 crop 后的图片。

#### 2) 字体展示

字体的选择是在实验 3 中 5 种广告体的基础上，又增加了 5 种手写体，以模仿验证集中出现的手写艺术字体。

# Innovation in China 中国智造

Innovation in China 中国智造

*Innovation in China* 中国智造

*Innovation in China* 中国智造

**Innovation in China** 中国智造

*Innovation in China* 中国智造

*Innovation in China* 中国智造

*Innovation in China* 中国智造

*Innovation in China* 中国智造

**Innovation in China** 中国智造

3) 结果展示

broadSpect

SUNBLOCKLO

时尚袋袋

读者文摘

73774/73



4) 结果分析

在生成这个版本的字体时，出现了字体与背景融合的情况，因为彩色背景和彩色字体的选择都是随机的，无法避免字体颜色和背景相近的情况，所以我们进行了多次调参，并强制给生成的字体加了 border，最后生成的效果如图所示，基本上是可以满足我们的实验要求。

## (6) 实验 4.2

### 1) 实验介绍

实验 4.2 选取了新闻及广告语的片段作为生成数据集的 label，以此来与实验 4.1 形成语义信息是否有影响的对比实验。字符长度 $\leq 10$ ，共生成 317109 张 crop 后的图片，其中中文 label 数量为 304470，英文 label 的数量为 12639。字体样式设置同实验 4.1。

### 2) txt 示例

中文的 label 取自新闻及广告语，而英文的 label，我们分别生成了一份大写和一份小写

的 label (0-9 数字混在其中), 因而能够充分的涵盖原 groundtruth 包含的所有基本类别。

◆ 中文 label

空, 飞向世界 (东方航  
, 容声, 质量的保证 (千, 路路有航天 (航天  
平安回家来 (公益广  
先一步, 申花电器 (申  
江西五十铃 (江西五十  
一股浓香, 一缕温暖 (你一杯, 我一杯, 一杯  
人头马一开, 好事自然  
的地方 (亚细亚商场)  
, 当然亮泽 (潘婷洗发  
显健康本色 (太阳神口  
就有多少动人的故事 (京啤酒, 清爽怡人 (燕  
名车, 嘉陵摩托 (嘉陵  
思丽使我更美丽 (伊思  
爱人喜欢 (牡丹电视机  
(小霸王电脑学习机)  
省优, 部优, 葛优? (美的前程, 共度美的人  
皮肤好, 早晚用大宝 (全世界 (鄂尔多斯羊绒

● 英文 label

大写:

EYYOURTH  
4FEELTHE  
NTS.SHARE  
MAKEYOURS  
ERYWHERE.  
NEWGENERA  
EINTEGRAT  
LENTLESSP  
NGCLOSET  
THEREISA  
T'SMAKET  
EETH,GOOD  
THEREAL  
GOBETTER  
DE'SIN,D  
DIFFERENT.  
ISPOSSIB  
5THEGLOBE

小写:

squifobye  
iswhaty  
lLouisAr  
RhythmOf  
LifeBonJ  
YouEvian  
chaelBolt  
els.Thetr  
ythingyo  
d.Maybeth  
lsforthe  
oproblem  
suitofpe  
motion,d  
tsomeone  
atedequal  
allstart

3) 结果展示

◆ 中文

◆ 英文

4 FEEL THE

SNACK WE

THE ONES

R WORLD,

4) 结果分析

对于这个版本的生成实验，我们做了 8 组 CRNN 对比实验。效果如下：

10	defaultcnn_4.1_gt	train	txt (105614) + train_4.1_gt2.txt (105763) + train_4.1_gt3.txt (10603 validation.txt (原数据12880)	30	22.44%	561.4
11	defaultcnn_4.2_ad_news	train	i_news_Chinese.txt (304470) + ad_news_English.txt (12639) = 31710 validation.txt (原数据12880)	30	15.54%	685.9
12	defaultcnn_4.1_pretrain	train	train_4.1_gt.txt (317409) + train_new.txt (126387) = 443796 validation.txt (原数据12880)	30	58.49%	125.4
13	defaultcnn_4.2_pretrain	train	train_4.2_ad_news.txt (317109) + train_new.txt (126387) = 443495 validation.txt (原数据12880)	30	58.21%	119.3
6	resnet18_4.1_gt	train	4.1_gt1.txt (105614) + train_4.1_gt2.txt (105763) + train_4.1_gt3.txt validation.txt (原数据12880)	30	20.77%	577
7	resnet18_4.2_ad_news	train	i_news_Chinese.txt (304470) + ad_news_English.txt (12639) = 31710 validation.txt (原数据12880)	30	13.66%	696.4
8	resnet18_4.1_pretrain	train	train_4.1_gt.txt (317409) + train_new.txt (126387) = 443796 validation.txt (原数据12880)	30	55.82%	126.1
9	resnet18_4.2_pretrain	train	train_4.2_ad_news.txt (317109) + train_new.txt (126387) = 443495 validation.txt (原数据12880)	30	56.28%	121.8

备注：1、绿色部分是纯人工生成数据，不加原数据集数据训练的结果

2、红色部分是加上原数据集的训练数据的结果

可以看出，单独训练人工生成的准确率非常低，说明生成数据并未能很好的拟合官方数据集的分布，人工合成的数据无法达到原数据集的训练效果。同时，实验 12 和 13、实验 8 和 9 是两组对比实验，无论是用 groundtruth 生成的 label 训练还是我们选取的广告和新闻语生成的 label 训练，准确率都相差在 1 个点之内，说明语义信息的影响并不大。

(7) 实验 5.1

针对实验 4 的训练效果，发现生成的彩色背景下的彩色字体，并不能很好的拟合原来的验证集。所以，我们着重分析了 validation 集的数据分布。

一方面，通过分析发现，validation 集中黑色字体和白色字体居多，而彩色字体的数量相对于占比例较少。所以实验 5 中采用了黑色字体：白色字体：彩色字体=1:1:1 的分布来生成数据。



图 3-40 验证集中的白色字体较多



图 3-41 验证集中的黑色字体较多



图 3-42 验证集中彩色背景下的彩色字体比例少

另一方面，实验 4 中出现了严重的字体颜色与背景融合的情况，即使强制加了 border 也无法辨认 label，如图 3-42 所示。针对这种情况，我们有必要挑选特定高对比度的字体颜色和背景，以避免这种情况的出现。

**Bad :**

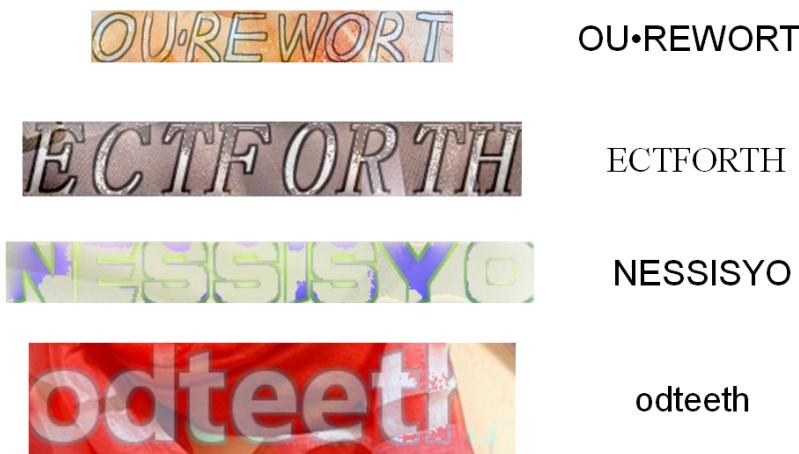


图 3-43 字体颜色与背景融合严重

### 1) 实验介绍

我们采用黑色字体：白色字体：彩色字体=1:1:1 配比。彩色字体从实验 4 中抽取，所以只需要分别生成黑色和白色字体即可，共生成 317541 条数据，label 来自 groundtruth。

### 2) 字体选择

黑体	<b>Innovation in China 中国智造</b>
宋体	Innovation in China 中国智造
隶书	<i>Innovation in China 中国智造</i>

楷体

# Innovation in China 中国智造

汉仪菱心体

# Innovation in China 中国智造

### 3) 背景示例

所有背景均采用纯色，并分为两部分，一部分为黑色字体对应的特殊颜色背景 50 张，另一部分是白色字体对应的特殊颜色背景 40 张。



图 3-44 background for black



图 3-45 background for white

### 4) 结果展示

- ◆ 黑色字体

**Neutrogena**

一件5折，两件4.5

乾爽、防水、防汗、不

73774/73

<http://xiao>

SUNBLOCK LO

◆ 白色字体

在生成白色字体的时候，又出现了严重的背景融合问题。分析原因是因为白色字体的像素（255, 255, 255）与相当一部分其通道像素值为255或接近255的背景像素太过接近，因而导致了融合问题。

**Bad example:**

73/74/737

Owner

DE

敏感肌用

兼容王

共好

**Groundtruth:**

73774/7377

Owner

MS6540B

敏感性肌用

兼容王

共好

图 3-46 白色字体与背景融合严重

针对这种背景融合问题，我们尝试了多种方法，最后发现，通过调低白色字体的像素，即将字体的三个通道像素值均由255调低至220，就可以解决这个问题。最后生成的结果如下。

上海僮憬专业五品假一

犹太人在中国

Example

T A O B A O

P O S T M O R

zhenyuping

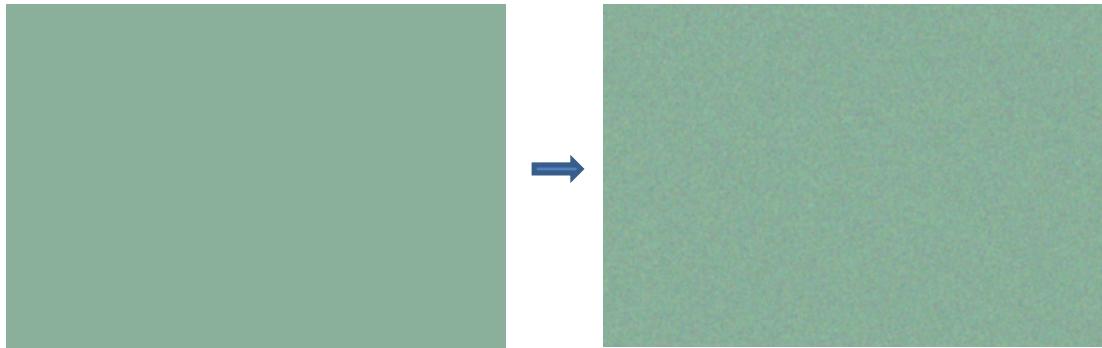
(8) 实验 5.2

### 1) 实验介绍

实验 5.2 在实验 5.1 的基础上对彩色字体的生成做了改变。黑色字体：白色字体：彩色字体仍然是 1:1:1，但是对所有背景加了高斯噪声和高斯模糊，另外还选取了特定的彩色字体和彩色背景，并将 12 万的 groundtruth 分成 6 份，每一份对应一种颜色的字体生成 label。

### 2) 背景展示

对所有背景均做了高斯噪声和高斯模糊的处理，以此来增加训练网络的鲁棒性。



另一方面，我们为字体选取了六种特定颜色，分别为 blue、cyan、green、red、yellow、purple，每种颜色选取特定高对比度的 10 张处理后的背景。如下图 12 所示，(a) 为 blue 字体对应的特定背景，(b) 为 cyan 字体对应的特定背景，(c) 为 green 字体对应的特定背景，(d) 为 purple 字体对应的特定背景，(e) 为 red 字体对应的特定背景，(f) 为 yellow 字体对应的特定背景。

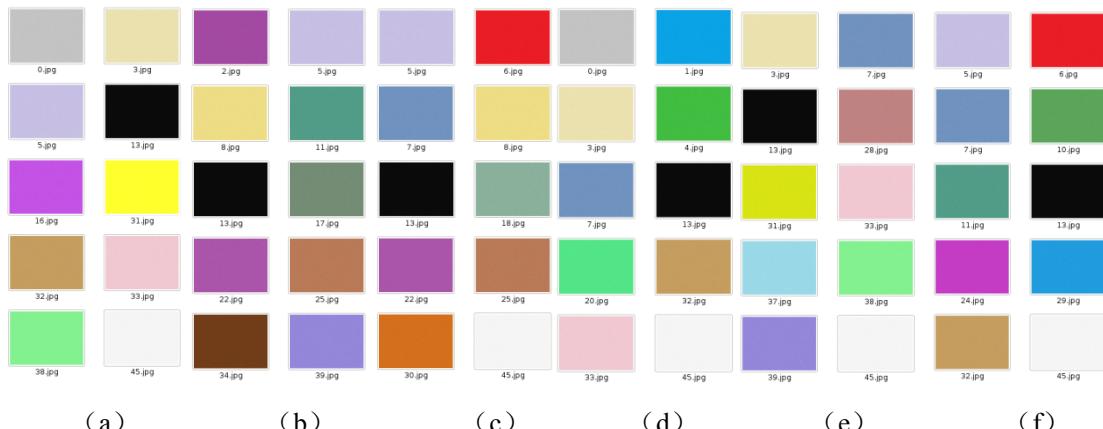


图 3-47 六种特定颜色字体对应的特定背景。

### 3) 结果展示

在生成彩色背景下彩色字体的数据时，仍然出现了相当一部分的背景融合问题。处理方法同处理白色字体生成的方法一样，通过适当调低这 6 种字体通道的像素值，可以很好地解决融合问题。

#### ◆ blue+特定背景

液压剪锁

2D.3D巨幕来袭

爆款！

维也纳的阳光

## 产品保修卡

广州 靓丽

◆ cyan+特定背景

就该这样吃

正版书籍

奇幻作家

台湾新锐

中文简体版

无纺布条形码

◆ green+特定背景

LIFE&

阿邦食品

福建名小吃

阿邦鱼卷正宗崇武特产

我们不是骄纵的孩子

1590070661

◆ purple+特定背景

Happy Birth

Happy Birth

Aftertesti

经检测开合50万次毫

COTO专营店

喝一口酒

◆ red+特定背景

数学四年级

**赠送草稿本**

**取出有害物质保留矿物**

**0120-32-04**

**515273863**

- ◆ yellow+特定背景

**配方升级**

**源自科技**

**浓密发丝**

**轻盈滋养**

**22.8162771**

**FUTURECOWB**

#### 4) 结果分析

实验 5 的数据随后被用于 CRNN 的 finetune 实验，模型从原来的 60.00% 上升了 2 个点，变成 62.27%，说明该版本的数据的生成效果较好。

##### 3.6.1.3 实验总结

人工生成数据的实验是 CRNN 模型训练中不可分割的重要组成部分，通过将这些生成的人工数据加入训练中，我们的模型从 baseline 实验的 58% 左右的准确率，增长了将近 2-3 个点。在数据生成的过程也遇到了很多困难，比如图片 padding 过大的问题，字体生成模糊的问题，一开始无法解决的字体与背景融合问题，都是通过不断的调参，修改代码解决的。另一方面，数据生成也要根据训练模型的实际需要，原数据的训练集及验证集的数据分布，及时修改生成模式，以便能更好的适应训练，提升训练效果。

后续该方向的工作也有很大的提升空间，例如如何通过修改代码自适应地选择前景色和背景色，以避免文本与背景融合的发生；如何能够更好地模拟真实场景的文本图片，达到更自然的效果等。这些都是亟待解决的问题。

## 3.6.2 模型训练

### 3.6.2.1 开发环境

CRNN 基础环境：

- ◆ CentOS7.4+ CUDA8.0
- ◆ Python3.6 +pytorch0.4.0+ opencv2.4+tensorflow1.4.0
- ◆ wrap\_ctc

### 3.6.2.2 数据预处理

#### (1) 裁剪

用于训练的数据集，包括比赛官方提供的原数据集，ICDAR2015-TRW, ICDAR2017-RCTW 在训练之前，需要根据图片对应的 groundtruth 把文字区域做相应的处理之后从图片中裁剪出来。

具体来说，如图 3-48 所示，数据集里的每一张图片对应一个文本文件，其中包含了图

片中的外接文本框（groundtruth）的四个顶点坐标和对应的文本内容。先根据四个顶点的坐标求边框的长和宽，确定文本框的中心点，计算好旋转角度，以文本框的中心点为中心旋转，将旋转好的文本框裁剪并保存到指定目录。为了便于后面的训练和测试，还需要将裁剪好的图片的路径和对应的文本内容（label）一一对应，写入文本文件。



图 3-48 图片与文本文件示例



图 3-49 原图片与裁剪后的图片



图 3-50 原图片与裁剪后的图片

```
./OCR_dataset/pre_1000/TB1.QRsLXXXXXb6aXXXunYpLFXX/2.jpg 购物狂在英国
./OCR_dataset/pre_1000/TB1.QRsLXXXXXb6aXXXunYpLFXX/3.jpg Gouwukuangzaiyingguo
./OCR_dataset/pre_1000/TB1.tFoLXXXXXXcapXXunYpLFXX/1.jpg You'reWorIt
./OCR_dataset/pre_1000/TB1.tFoLXXXXXXcapXXunYpLFXX/2.jpg Withthemostfashionableelement
./OCR_dataset/pre_1000/TB1.tFoLXXXXXXcapXXunYpLFXX/3.jpg Workhardtomakethefineleather
./OCR_dataset/pre_1000/TB1.tFoLXXXXXXcapXXunYpLFXX/4.jpg WHITEFOREST
./OCR_dataset/pre_1000/TB1.tFoLXXXXXXcapXXunYpLFXX/5.jpg Z
./OCR_dataset/pre_1000/TB1.tFoLXXXXXXcapXXunYpLFXX/6.jpg Z
./OCR_dataset/pre_1000/TB1.yxALXXXXXa3XVXXunYpLFXX/1.jpg 蜗牛
./OCR_dataset/pre_1000/TB1.yxALXXXXXa3XVXXunYpLFXX/2.jpg 保安堂药行
./OCR_dataset/pre_1000/TB1.yxALXXXXXa3XVXXunYpLFXX/3.jpg 手机淘宝：
./OCR_dataset/pre_1000/TB1.yxALXXXXXa3XVXXunYpLFXX/4.jpg 1248858
./OCR_dataset/pre_1000/TB1.yxALXXXXXa3XVXXunYpLFXX/5.jpg 100%
./OCR_dataset/pre_1000/TB1.yxALXXXXXa3XVXXunYpLFXX/6.jpg 实物拍摄
./OCR_dataset/pre_1000/TB1.yxALXXXXXa3XVXXunYpLFXX/7.jpg 盗图必究
```

图 3-51 存放图片路径和文本内容的文本文件示例

## (2) 坚排图片处理

本次评测选用的模型对坚排图片不太友好，需要先将坚排图片进行切割，将切割好的单字图片送进网络进行识别，再把识别结果拼接在一起。

对于评测数据集里面的坚排图片的切割，起初尝试了 canny 边缘检测的方法，实现步骤如下：①彩色图像转换为灰度图像；②对图像进行高斯模糊；③计算图像梯度，根据梯度计算图像边缘幅值与角度；④非最大信号压制处理；⑤双阈值边缘连接处理；⑥输出二值化图像的边缘检测结果；⑦将检测到的边缘切割保存。但由于中文字符结构的特殊性，利用检测到的边缘来对图形进行切割会出现“断裂”现象。

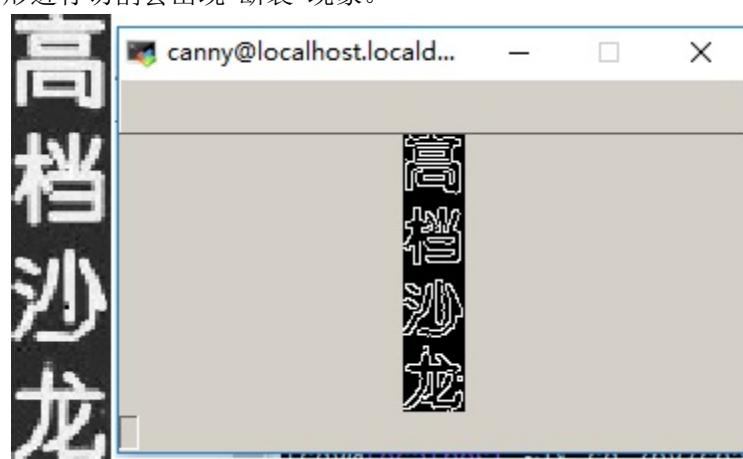


图 3-52 canny 算子边缘检测

之后对评测数据集里面的坚排图片用水平投影的方法进行切割。具体来说，先灰度化读

取竖排图片，再将竖排图片二值化，一开始采用了简单阈值，即只需要规定一个固定阈值，整个图像都是与该阈值比较大小，这种二值化方法效果不好。后来采用了自适应二值化，选取的是局部阈值，通过规定一个  $b \times b$  的区域大小，比较每个像素点与周围  $b \times b$  区域像素的算术平均值的大小来确定该像素点的黑白。



图 3-53 自适应二值化

之后为了去除背景图片的干扰，对二值化图片进行膨胀腐蚀处理。再对图像的每一行计算投影值，绘制水平投影图，根据水平投影值选定行分割线，将图片切割保存



图 3-54 水平投影

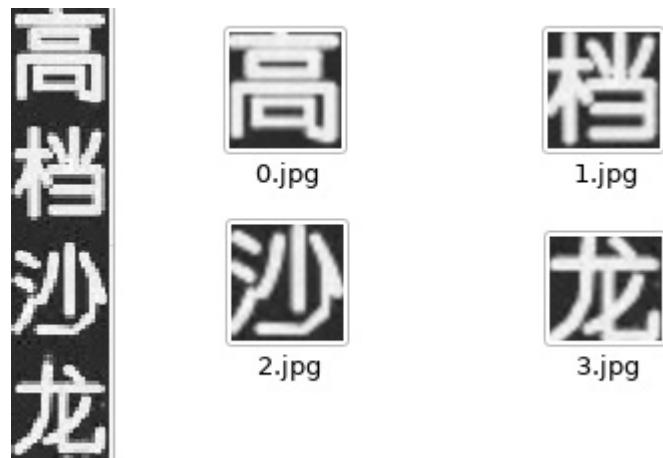


图 3-55 切割效果示例

### (3) 数据增强

为了减少网络的过拟合的现象，对训练数据做了 superpixel, invert, add(per channel), multiply, GaussianBlur 等十种变换，每个 batch 从十种变换操作中随机选择三种。测试数据尝试了 sharpen 的数据增强，可以锐化图像，使图像的边缘变得清晰。但实验效果不好，最后没有给测试图片加数据增强。



图 3-56 数据增强

#### 3.6.2.3 数据来源

本次比赛的数据集特点如下：

- (1) 数据集是 2 万张网络图片，其中 50% 用来作为训练集，50% 用来作为测试集
  - (2) 图片主要由合成图像，产品描述，网络广告构成
  - (3) 特点：复杂排版，多语言文本，水印，数十种字体，较多干扰背景
- 典型图片示例如下：



图 3-57 ICPR 典型图片

除了官方提供的训练数据集，我们还使用了以下数据集：ICDAR2015-TRW，ICDAR2017-RCTW 以及人工生成数据。其中 ICDAR2015-TRW 和 ICDAR2017-RCTW 是多语言中英文字符识别的竞赛数据集，主要由自然场景下的图片和部分网络图片组成，包括街景，海报，菜单，室内场景，屏幕截图等。这两个数据集的分布与 ICPR 竞赛数据集分布相似，可以很好地提升实验结果。



图 3-58 ICDAR 数据集典型图片

此外，考虑到 ImageNet 数据集中都是自然场景图片，与 ICPR 竞赛数据集的分布相似，且数据集规模较大，所以 CNN 模型使用了在 ImageNet 数据集上预训练好的模型来 finetune，实验结果显示这样可以提升分类准确率。

#### 3.6.2.4 CRNN 实验

##### (1) 原数据集实验

训练集：ICPR 竞赛原数据集，9000 张图片裁剪出 126k 张用于字符识别的文本图片。

验证集：ICPR 竞赛原数据集，1000 张图片裁剪出 15k 张用于字符识别的文本图片。

CRNN 实验结果：

Model	CNN	RNN	Training Dataset	Validation Dataset	Word Accuracy	Testing Loss
CRNN	Default_CNN	Bi-LSTM	9000(126k)	1000(13k)	58.04%	137.9
CRNN	ResNet18	Bi-LSTM	9000(126k)	1000(13k)	55.44%	143.1
CRNN	ResNet34	Bi-LSTM	9000(126k)	1000(13k)	53.66%	149.6
CRNN	ResNet50	Bi-LSTM	9000(126k)	1000(13k)	43.93%	166.2

表 1 CRNN 实验结果

##### (2) finetune 实验

37w 数据集的实验结果：

Model	CNN	RNN	Training Dataset	Validation Dataset	Word Accuracy	Testing Loss
CRNN	ResNet18	Bi-LSTM	37w	1000(13k)	58.52%	115.9
CRNN	ResNet34	Bi-LSTM	37w	1000(13k)	58.72%	117.5
CRNN	ResNet50	Bi-LSTM	37w	1000(13k)	59.60%	112.4

表 2 CRNN 实验结果

137w 数据集的实验结果：

Model	CNN	RNN	Training Dataset	Validation Dataset	Word Accuracy	Testing Loss
CRNN	ResNet18	Bi-LSTM	137w	1000(13k)	56.91%	117.0
CRNN	ResNet34	Bi-LSTM	137w	1000(13k)	58.35%	112.9.
CRNN	ResNet50	Bi-LSTM	137w	1000(13k)	60.00%	109.1

表 3 CRNN 实验结果

37w+RCTW+TRW 共 43w 数据集的实验结果：

Model	CNN	RNN	Training Dataset	Validation Dataset	Word Accuracy	Testing Loss	Average Distance
CRNN	ResNet50	Bi-LSTM	43w	1000(13k)	62.80%	100.4	0.8738
CRNN	ResNet50	Bi-LSTM	50w	1000(13k)	63.73%	96.91	0.8359
CRNN	ResNet50	Bi-LSTM	49w	1000(13k)	63.97%	96.73	0.8307

表 4 CRNN 实验结果

### 3.6.2.5 多模型融合

评测前有若干个训好的模型，在评测时，对各个模型的识别结果可视化以后可以发现，不同的模型的识别结果可以互补。一开始，我们将识别结果的概率值也写进识别结果的文本文件，对于某张图片，如果不同模型的预测值不同，就选择概率值大的结果作为最后的预测值，但是，对最终结果可视化以后，我们发现概率值大的结果不一定准确。最后，只把某些没有识别结果的图片，直接替换成其他模型的识别结果，将不同模型的识别结果结合起来。

Open	+	ver	~/zpy/icpr_val
line_10335.jpg/6 N 0.00000001744704469786			
line_10335.jpg/7 c 0.00000456928410130786			
line_10335.jpg/8 E 0.00000126906718378450			
line_10335.jpg/9 D 0.00001090478781407000			
line_10336.jpg/0 露 0.03791308775544166565			
line_10336.jpg/1 露 0.00003415658284211531			
line_10336.jpg/2			
line_10336.jpg/3 服 0.00252773286774754524			
line_10337.jpg/0 专 0.01745613478124141693			
line_10337.jpg/1 业 0.00005027241422794759			
line_10337.jpg/2 定 0.94192862510681152344			
line_10337.jpg/3 做 0.05433858186006546021			
line_10354.jpg/0 . 0.00000000354837204064			
line_10354.jpg/1 茶 0.00002212832441728096			
line_10356.jpg/0 上 0.0000004763533567598			
line_10356.jpg/1 * 0.00000001040665154051			
line_10357.jpg/0 H 0.00000000525658139239			
line_10357.jpg/1 拉 0.00000007635777876658			
line_10360.jpg/0 1 0.0000000059823640486			
line_10360.jpg/1 S 0.00000001058998577719			
line_10361.jpg/0 加 0.0000000105307629283			
line_10361.jpg/1 购 0.0000000131335395992			
line_10435.jpg/0 搭 0.94563913345336914062			
line_10435.jpg/1 配 0.00001979356056835968			

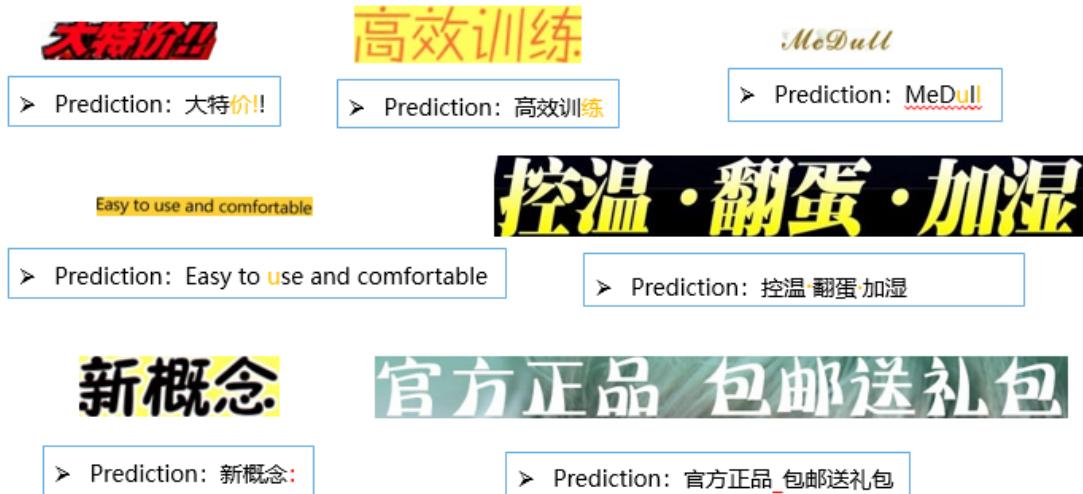
图 3-59 不同模型的识别结果

<input type="button" value="Open"/>	<input type="button" value="+"/>
line_10335.jpg/7 c	
line_10335.jpg/8 E	
line_10335.jpg/9 D	
line_10336.jpg/0 露	
line_10336.jpg/1 露	
line_10336.jpg/2 舞	
line_10336.jpg/3 服	
line_10337.jpg/0 专	
line_10337.jpg/1 业	
line_10337.jpg/2 定	
line_10337.jpg/3 做	
line_10354.jpg/0 .	
line_10354.jpg/1 泰	
line_10356.jpg/0 上	
line_10356.jpg/1 水	
line_10357.jpg/0 型	
line_10357.jpg/1 拉	
line_10360.jpg/0 2	
line_10360.jpg/1 点	
line_10361.jpg/0 加	
line_10361.jpg/1 热	
line_10435.jpg/0 搭	
line_10435.jpg/1 配	
	→
line_10066.jpg 水槽下门板出	
line_1017.jpg 正品专卖	
line_1018.jpg 盗图必究	
line_10204.jpg 全国二件包邮	
line_10205.jpg 新店开业	
line_10233.jpg 金刚版	
line_10335.jpg LULU DANcEDRESS	
line_10336.jpg 露露舞服	
line_10337.jpg 专业定做	
line_10354.jpg .泰	
line_10356.jpg 上水	
line_10357.jpg 型拉	
line_10360.jpg 2点	
line_10361.jpg 加热	
line_10435.jpg 搭配双吊绳	
line_10447.jpg 喜包邮	
line_10448.jpg 仅此一家	
line_10476.jpg 超经济型，超低价	
line_10548.jpg 进店有	
line_10555.jpg 非贴纸	
line_10706.jpg 活典	
line_10712.jpg 海宝乐耳知音	
line_10739.jpg 好东西介绍给大家	
line_10752.jpg 正品有友	
line_10753.jpg 还能美容	

图 3-60 多模型融合结果

### 3.6.3 错误结果分析

#### 3.6.3.1 漏检/多检



分析：这类漏检/多检问题可能与 CTC 的矫正有关，如第一张图，可能 RNN 输出的序列里由大特价，但 CTC 纠正之后把价字过滤掉了，就出现了漏检的情况。

#### 3.6.3.2 识别离谱

➤ Prediction: 富吉益智手工

➤ Prediction: 不孔奥

➤ Prediction: 澳瓶老特

➤ Prediction: 提0力

➤ Prediction: Moire

➤ Prediction: Girl龄

➤ Prediction: Blidle

➤ Prediction: 口5夕月拓力光业

分析：这类问题可能与数据训练不够有关，本次评测数据集较为复杂：多种语言，多种字体，复杂背景，当训练数据集与评测数据集的分布相差较大时，一些图片就识别不出来。最后一张图片是字符识别里的遮挡问题。

### 3.6.3.3 复杂背景

➤ Prediction: 业医疗888

➤ Prediction: 询电电话888

➤ Prediction: 德

➤ Prediction: 批发净D

➤ Prediction: 拍家手水

➤ Prediction: 下男

➤ Prediction: aHcovay

分析：复杂背景的图片可以考虑在识别前对图片做二值化处理，去除背景的干扰。

### 3.6.3.4 低分辨率



➤ Prediction: 话条美剂



➤ Prediction: 30



➤ Prediction: 三0



➤ Prediction: ulx

➤ Prediction: M3

➤ Prediction: 爱正活性刺类

分析：低分辨率的图片可以考虑用生成模型生成高分辨率图片再进行识别。

### 3.6.3.5 特殊字符/符号



➤ Prediction: HORD

➤ Prediction: R

➤ Prediction: None



➤ Prediction: 注音全彩手绘版

➤ Prediction: dbolo?

分析：特殊符号可以在数据集里加大训练。

### 3.6.3.6 数据集自身问题



➤ Prediction: ROCH

➤ Prediction: Healch

➤ Prediction: 金阁



➤ Prediction: T3VA

➤ Prediction: 三

➤ Prediction: 1-2807V93

分析：数据集自身存在一定的少割、错割、颠倒等问题，这部分数据只能舍去，目前无法处理。

### 3.6.3.7 形近字错检

复仇者联盟

➤ Prediction: 复优者联盟

香酥芝麻味

➤ Prediction: 香酥芯麻味

卷力高档

➤ Prediction: 专力高档

冷加工糕点

➤ Prediction: 令加工糕点

DEF

+

➤ Prediction: OEF

➤ Prediction: 七

分析：形近字的识别问题，一方面可以加大数据集，如果是识别和图片生成是一体的话，可以在训练的每一轮记录识别错误的字符与正确的文本内容（label），然后生成形近字的图片加入之后的训练。另一方面可以添加记录识别错误的形近字的列表，每一轮结束后更新列表，以此改善之后的形近字错检的情况。

### 3.6.3.8 艺术字/不规则形状

马上封侯

➤ Prediction: 金特好

AY

➤ Prediction: A尔

市

➤ Prediction: 代市

餐

➤ Prediction: 餐

RMS

➤ Prediction: RM

3

➤ Prediction: 3

世

➤ Prediction: 世

分析：不规则图片的识别可以参考 RARE[45]，在识别前利用 STN 网络对弯曲文字进行矫正，再识别矫正后的图片。

### 3.6.3.9 不同朝向



分析：任意朝向的文本图片的识别可以参考 AON[46]，将任意方向的字符编码为 4 个方向的 4 个特征序列表示：左→右，右→左，上→下，下→上。AON 提取 4 个方向的场景文字特征和位置信息，FG 集成 4 个方向的特征序列。

### 3.7 本章总结

本章主要介绍了图像字符识别相关内容，定义了什么叫做 OCR 及其主要任务。然后通过一篇简短的综述介绍了整个字符识别领域的发展历史以及未来的发展趋势。在主流算法一节中，着重介绍了目前达到 state-of-the-art 水平的几个主要算法模型。随后，介绍了现今比较权威和常用的多种数据集。最后介绍了实验室自己的算法以及 ICPR2018 比赛中详细的实验流程。

## 参考文献

- [1] 百度百科：<https://baike.baidu.com/item/%E5%85%89%E5%AD%A6%E5%AD%97%E7%AC%A6%E8%AF%86%E5%88%AB/4162921?fromtitle=OCR&fromid=25995&fr=Aladdin>
- [2] 安艳辉. 中英文混排字符切分方法研究[D].河北大学,2004.
- [3] 张秋月. 基于 ANDROID 平台的水表字符识别算法研究[D].杭州电子科技大学,2014.
- [4] Gllavata J, Ewerth R, Stefi T, Freisleben B. Unsupervised text segmentation using color and wavelet features. In Image and Video Retrieval. Springer-Berlin Heidelberg, 2004, 216-224.
- [5] Song Y, Liu A, Pang L, Lin S, Zhang Y, Tang, S. A novel image text extraction method based on K-means clustering. In: Proceedings of International Conference on In Computer and Information Science, 2008, 185–190.
- [6] Chen D, Olobez J M, Bourlard H. Text segmentation and recognition in complex background based on Markov random field. In: Proceedings of International Conference on Pattern Recognition (ICPR), 2002, 227–230.
- [7] Ye Q, Gao W, Huang Q. Automatic text segmentation from complex background. In: Proceedings of International Conference on Image Processing (ICIP), 2004, 2905–2908.
- [8] Li M, Bai M, Wang C, Xiao B. Conditional random field for text segmentation from images with complex background. Pattern Recognition Letters (PRL), 2012, 31(14): 2295–2308.
- [9] Feild J, Learned-Miller E G. Scene text recognition with bilateral regression. UMass Amherst Technical Report, 2012.
- [10] Chen X, Yang J, Zhang J, Waibel A. Automatic detection and recognition of signs from natural scenes. IEEE Transactions on Image Processing (TIP), 2004, 13(1): 87–99.
- [11] De Campos T E, Babu B R, Varma M. Character Recognition in Natural Images. In: VISAPP, 2009: 273-280.
- [12] Yang J, Jiang Y G, Hauptmann A G, Ngo C W. Evaluating bag-of-visual-words representations in scene classification. In: Proceedings of International Workshop on Multimedia Information Retrieval, ACM, 2007, 197-206.
- [13] De Campos T E, Babu B R, Varma M. Character Recognition in Natural Images. In: VISAPP, 2009: 273-280.
- [14] Berg A C, Berg T L, Malik J. Shape matching and object recognition using low distortion correspondences. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005, 26–33.
- [15] Belongie S, Malik J, Puzicha J. Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2002, 24(4): 509-522.
- [16] Newell A J, Griffin L D. Multiscale histogram of oriented gradient descriptors for robust character recognition. In: Proceedings of International Conference on Document Analysis and Recognition (ICDAR), 2011, 1085-1089.
- [17] Weinman J J, Butler Z, Knoll D, Feild J. Toward integrated scene text reading. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2014, 36(2): 375-387.
- [18] Shi C, Wang C, Xiao B, Zhang Y, Gao S, Zhang Z. Scene text recognition using part-based tree-structured character detection, In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, 2961-2968.
- [19] Mishra A, Alahari K, Jawahar C V. Scene text recognition using higher order language priors. In: Proceedings of British Machine Vision Conference (BMVC), 2012, 127.1-127.11.

- [20] Shi C, Wang C, Xiao B, Gao S, Hu J. End-to-end scene text recognition using tree-structured models. *Pattern Recognition (PR)*, 2014, 47(9): 2853-2866.
- [21] 高彦宇,杨扬.脱机手写体汉字识别研究综述[J].计算机工程与应用,2004,(7):74-77.
- [22] 徐志明,王晓龙,张凯,等.联机手写体汉字识别后处理技术的研究[J].计算机研究与发展,1999,36(5):608-612.
- [23] 俞庆英,吴建国.一种联机手写汉字识别算法的研究与实现[J].合肥学院学报(自然科学版),2004,14(1):37-39.
- [24] 张冬霞.基于(ANN)和(HMM)的联机手写体汉字识别系统[J].微计算机信息,2005,21(14):444-446 .
- [25] 赵巍,刘家锋,唐降龙.基于部件 HMM 级联的联机手写体汉字识别方法[J].哈尔滨工业大学学报,2004,36(5):570-573.
- [26] 鲁湛,丁晓青.基于笔段间关系的联机手写汉字 HMM 模型[J].清华大学学报(自然科学版),2004,44(7):913-916.
- [27] 龚才春,刘荣兴.脱机手写体汉字字符的笔顺信息恢复[J].山东大学学报(理学版),2004,39(1):73-75.
- [28] 李元祥,丁晓青,刘长松.一种基于噪声信道模型的汉字识别后处理新方法[J].清华大学学报(自然科学版),2001,41(1):24-28.
- [29] 张睿,丁晓青,方驰.脱机手写汉字识别的最优采样特征新方法[J].中国图象图形学报,2002,7(2):176-180.
- [30] 童学锋,石繁槐.FSVM 在有限集脱机手写体汉字识别中的应用[J].计算机工程,2003,29(13):109-111.
- [31] 高彦宇,杨扬,陈飞.基于融合特征和 LS-SVM 的脱机手写体汉字识别[J].北京科技大学学报, 2005,27(4):509-512.
- [32] 吴雪菁,施鹏飞.质心层次特征的无约束手写体数字识别[J].上海交通大学学报,1998,32(9):31-34.
- [33] 王贵新,刘建胜,居琰等.手写字符轮廓曲率的特征提取和识别[J].华中科技大学学报,2001,29(S1):83-86.
- [34] 朱小燕,史一凡.基于反馈的手写体字符识别方法的研究[J].计算机学报,2002,25(5):476-482.
- [35] 龚才春,刘荣兴.基于整体特征的快速手写体数字字符识别[J].计算机工程与应用,2004,(19):82-83.
- [36] 张文国.“汉王”多文种手写印刷体字符识别系统简介[J].中国产业科技,1997,(2):45-46.
- [37] 任金昌,赵荣椿,张炜.一种快速有效的印刷体文字识别算法[J].中国图象图形学报,2001,6(10):1011-1015.
- [38] 熊军,谢跃雷.手写印刷体汉字识别的细化算法研究[J].桂林电子工业学院学报,1997,17(4):45-48.
- [39] 唐亮,胡运发,张文龙.基于小波变换在强干扰条件下印刷体汉字处理研究[J].计算机应用与软件,2005,22(8):46-47.
- [40] 钟锐,黄华.一种用 VB 实现的印刷体数字识别方法[J].计算机工程,2003,29(7):106-107.
- [41] 王维兰,丁晓青,陈力等.印刷体现代藏文识别研究[J].计算机工程,2003,29(3):37-38.
- [42] 郑朝晖,裘聿皇,陈峻峰.一种印刷体字符识别的新方法:基于遗传算法( $0,1,\star$ )——矩阵法[J].控制与决策,2001,16(3):296-298.
- [43] 陈兆学,施鹏飞,周煦潼.一类特殊印刷体字符的分割和识别方法[J].微型电脑应用,2003,19(2):40-42.

- [44]胡守仁,沈清,胡德文等.神经网络应用技术[M].北京:国防科技大学出版社,1993.
- [45] Baoguang Shi, Xinggang Wang, Pengyuan Lv, Cong Yao, and Xiang Bai. Robust scene text recognition with automatic rectification. arXiv preprint arXiv:1603.03915, 2016.
- [46] Zhanzhan Cheng, Yangliu Xu, Fan Bai, Yi Niu, Shiliang Pu, Shuigeng Zhou. AON: Towards Arbitrarily-Oriented Text Recognition. arXiv preprint arXiv:1711.04226, 2017.
- [47] Wojna Z, Gorban A N, Lee D S, et al. Attention-Based Extraction of Structured Information from Street View Imagery[C]// Iapr International Conference on Document Analysis and Recognition. IEEE Computer Society, 2017:844-850.
- [48] Lee C Y, Osindero S. Recursive Recurrent Nets with Attention Modeling for OCR in the Wild. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2231-2239.
- [49] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, Andrew Zisserman. Deep structured output learning for unconstrained text recognition. arXiv:1412.5903v5 [cs.CV].
- [50] He Pan, et al. Reading scene text in deep convolutional sequences. arXiv preprint arXiv:1506.04395 (2015).
- [51] Almazán J, Gordo A, Fornés A, et al. Word spotting and recognition with embedded attributes[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2014, 36(12):2552-2566.
- [52] 王涛,江加和.基于语义分割技术的任意方向文字识别[J/OL].应用科技:1-6[2018-07-12]. <http://kns.cnki.net/kcms/detail/23.1191.U.20170704.1807.006.html>.
- [53] B. Shi, X. Bai, C. Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. CoRR abs/1507.05717 (2015).
- [54] A. Graves, S. Fernandez, F. J. Gomez, and J. Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In ICML, 2006.
- [55] K. He, X. Zhang, S. Ren, J. Sun, "Deep residual learning for image recognition", 2015.
- [56] A. Graves, A. Mohamed, and G. E. Hinton. Speech recognition with deep recurrent neural networks. In ICASSP, 2013.
- [57] Gupta A, Vedaldi A, Zisserman A. Synthetic Data for Text Localisation in Natural Images[J]. 2016:2315-2324.
- [58] P. Perez, M. Gangnet, and A. Blake. Poisson image editing. ACM Transactions on Graphics, 22(3):313–318, 2003.
- [59] A. Mishra, K. Alahari, and C. Jawahar. Scene text recognition using higher order language priors. Proc. BMVC., 2012.

## 附录 A ICPR-MTWI2018 挑战赛一：网络图像的文本识别竞赛细则

### 1. 赛题简介

在互联网世界中，图片是传递信息的重要媒介。特别是电子商务，社交，搜索等领域，每天都有数以亿兆级别的图像在传播。图片文字识别（OCR）在商业领域有重要的应用价值，是数据信息化和线上线下打通的基础，也是学术界的研究热点。然而，研究领域尚没有基于网络图片的、以中文为主的OCR数据集。本竞赛将公开基于网络图片的中英混合数据集，该数据集数据量充分，涵盖几十种字体，几个到几百像素字号，多种版式，较多干扰背景。期待学术界可以在本数据集上作深入的研究，工业界可以藉此发展基于OCR的图片管控，搜索，信息录入等AI领域的⼯作。

### 2. 数据集

我们提供 20000 张图像作为本次比赛的数据集。其中 50% 用来作为训练集，50% 用来作为测试集。该数据集全部来源于网络图像，主要由合成图像，产品描述，网络广告构成。典型的图片如图 1 所示：

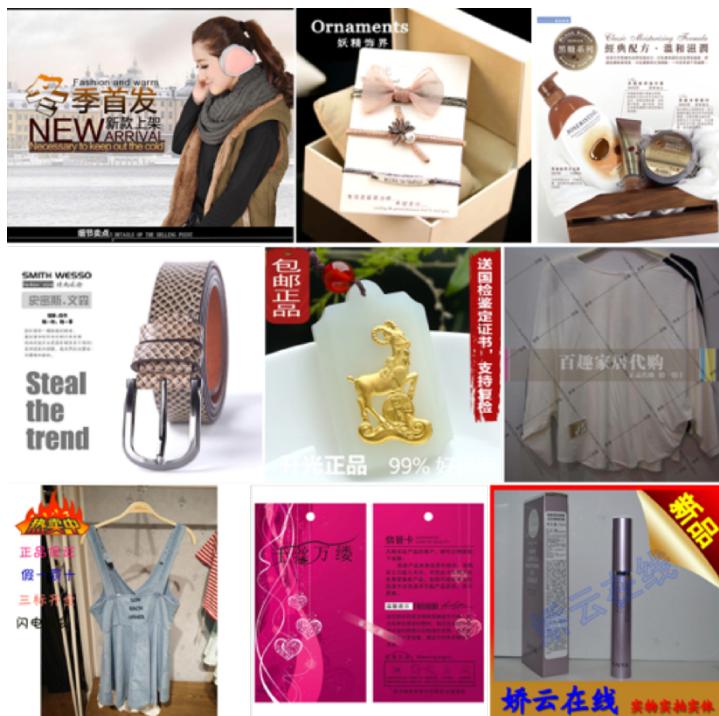


图 1：典型图片

这些图像是网络上最常见的图像类型。每一张图像或者包含复杂排版，或者包含密集的小文本或多语言文本，或者包含水印，这对文本检测和识别均提出了挑战。

对于每一张图像，都会有一个相应的文本文件（.txt）（UTF-8 编码与名称：[图像文件名].txt）。文本文件是一个逗号分隔的文件，其中每行对应于图像中的一个文本串，并具有以下格式：

X1, Y1, X2, Y2, X3, Y3, X4, Y4, “文本”

其中  $X_1, Y_1, Y_2, X_2, X_3, Y_3, X_4, Y_4$  分别代表文本的外接四边形四个顶点坐标。而“文本”是四边形包含的实际文本内容。

图 2 是标注的图片，红色的框代表标注的文本框。

图 3 是标注图片对应的文本文件。标注时我们对所有语言，所有看不清的文字串均标注了外接框（比如图 2 中的小字），但对于除了中文，英文以外的其它语言以及看不清的字符并未标注文本内容，而是以“###”代替。



图 2 : image.jpg

494. 91, 36. 36, 494. 91, 81. 45, 596. 0, 81. 45, 595. 27, 32. 73, 三星  
614. 91, 34. 91, 614. 91, 77. 82, 783. 64, 77. 82, 783. 64, 34. 91, N7100  
524. 73, 93. 82, 524. 73, 146. 18, 784. 36, 146. 18, 784. 36, 93. 82, 钢化玻璃膜  
164. 0, 143. 27, 164. 0, 157. 09, 251. 27, 157. 09, 251. 27, 143. 27, SNMSUNG  
316. 73, 174. 55, 316. 73, 189. 09, 353. 82, 189. 09, 353. 82, 174. 55, 17: 39  
117. 45, 236. 36, 117. 45, 284. 36, 298. 55, 284. 36, 298. 55, 236. 36, 17:39  
142. 91, 292. 36, 142. 91, 309. 82, 268. 73, 309. 82, 268. 73, 292. 36, 9月12日星期三  
262. 91, 354. 91, 262. 91, 367. 27, 345. 82, 367. 27, 345. 82, 354. 91, 小到中雨转阵雨  
321. 82, 338. 91, 321. 82, 350. 55, 345. 82, 350. 55, 345. 82, 338. 91, 北京  
70. 57, 378. 86, 70. 57, 384. 57, 87. 14, 384. 57, 87. 14, 378. 86, ###  
88. 86, 376. 57, 88. 86, 385. 71, 118. 57, 385. 71, 118. 57, 376. 57, 新浪天气  
206. 0, 371. 43, 206. 0, 381. 71, 324. 29, 381. 71, 324. 29, 371. 43, 已更新2012/09/1116:59  
76. 86, 545. 71, 76. 86, 556. 0, 106. 0, 556. 0, 104. 86, 546. 29, 沃 • 3G  
150. 57, 545. 71, 150. 57, 557. 71, 183. 71, 557. 71, 183. 71, 545. 71, 沃商店  
226. 57, 546. 86, 226. 57, 557. 71, 264. 29, 557. 71, 264. 29, 546. 86, 116114  
294. 57, 545. 14, 294. 57, 557. 71, 350. 0, 557. 71, 350. 0, 545. 14, 手机营业厅  
67. 71, 682. 86, 67. 71, 693. 14, 100. 29, 693. 14, 100. 29, 682. 86, 联系人  
135. 71, 682. 29, 134. 57, 694. 29, 156. 29, 694. 29, 156. 29, 681. 14, 手机  
195. 14, 682. 86, 195. 14, 693. 71, 218. 57, 693. 71, 218. 57, 682. 86, 信息  
251. 14, 682. 86, 251. 14, 692. 57, 282. 57, 692. 57, 282. 57, 682. 86, 互联网  
307. 14, 681. 71, 307. 14, 693. 14, 349. 43, 693. 14, 349. 43, 681. 71, 应用程序  
232. 29, 513. 14, 232. 29, 522. 86, 261. 43, 522. 86, 261. 43, 513. 14, 116114  
150. 0, 514. 29, 150. 0, 522. 86, 172. 86, 522. 86, 172. 86, 514. 29, W0  
153. 43, 523. 43, 153. 43, 532. 0, 180. 86, 532. 0, 180. 86, 523. 43, ###  
76. 29, 522. 29, 76. 29, 529. 14, 105. 43, 529. 14, 105. 43, 522. 29, ###  
76. 29, 504. 0, 76. 29, 514. 86, 107. 14, 514. 86, 107. 14, 504. 0, W0  
69. 43, 341. 14, 69. 43, 366. 86, 125. 43, 366. 86, 125. 43, 341. 14, 29° C  
141. 43, 340. 0, 141. 43, 366. 86, 195. 71, 366. 86, 195. 71, 340. 0, 17° C  
547. 71, 701. 14, 547. 71, 746. 86, 784. 29, 746. 86, 784. 29, 701. 14, 送贴膜工具  
568. 29, 554. 86, 588. 29, 608. 0, 792. 29, 534. 29, 775. 14, 481. 71, 防爆防刮  
551. 14, 449. 14, 577. 43, 510. 86, 795. 71, 434. 29, 785. 43, 372. 57, 智能贴合  
542. 0, 344. 0, 564. 29, 396. 57, 788. 86, 323. 43, 771. 71, 265. 71, 防指纹油  
659. 14, 543. 43, 659. 14, 544. 0, 659. 71, 544. 0, 659. 71, 543. 43, ###

图 3 : image.txt

### 3.任务描述

**网络图像的文本行（列）识别：**

识别单文本行（列）图片中的文字。模型训练中，允许使用其它数据集或者生成数据，允许 Fine-tuning 模型或者其他模型。入围团队提交报告中须对额外使用的数据集或非本数据集训练出的模型做出说明。

**训练集：**

选手需要抠出所公布数据集中的文本行（列）图片，去掉没有文本内容的图片（即忽略以‘###’为内容的文本行）。最终用文本行（列）图片和文本内容训练模型。

**测试集：**

输入：我们会提供切好的横排和竖排文本图片。

输出：选手将识别出的内容输出到对应的[图像文件名].txt 中。字符需以 UTF-8 编码。

**提交：**

将所有 [图像文件名].txt 文件放到一个 zip 压缩包中，然后提交。

### 4.评估标准

我们会分别给出全匹配和编辑距离两个结果的排名，但最终奖金的分配以编辑距离的排名为准。

## 附录 B ICPR-MTWI2018 挑战赛二：网络图像的文本检测竞赛细则

### 1. 赛题简介

在互联网世界中，图片是传递信息的重要媒介。特别是电子商务，社交，搜索等领域，每天都有数以亿兆级别的图像在传播。图片文字识别（OCR）在商业领域有重要的应用价值，是数据信息化和线上线下打通的基础，也是学术界的研究热点。然而，研究领域尚没有基于网络图片的、以中文为主的OCR数据集。本竞赛将公开基于网络图片的中英混合数据集，该数据集数据量充分，涵盖几十种字体，几个到几百像素字号，多种版式，较多干扰背景。期待学术界可以在本数据集上作深入的研究，工业界可以藉此发展基于OCR的图片管控，搜索，信息录入等AI领域的⼯作。

### 2. 数据集

我们提供 20000 张图像作为本次比赛的数据集。其中 50% 用来作为训练集，50% 用来作为测试集。该数据集全部来源于网络图像，主要由合成图像，产品描述，网络广告构成。典型的图片如图 1 所示：

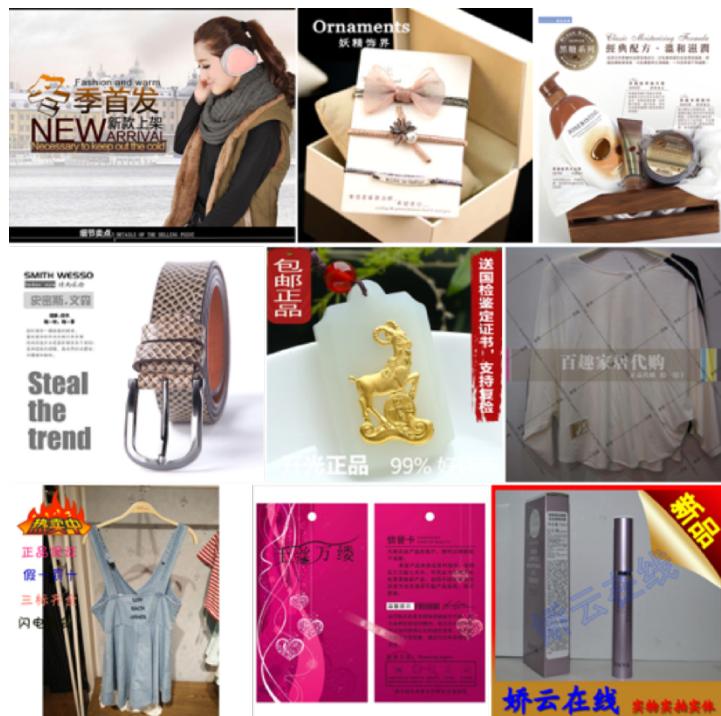


图 1：典型图片

这些图像是网络上最常见的图像类型。每一张图像或者包含复杂排版，或者包含密集的小文本或多语言文本，或者包含水印，这对文本检测和识别均提出了挑战。

对于每一张图像，都会有一个相应的文本文件（.txt）（UTF-8 编码与名称：[图像文件名].txt）。文本文件是一个逗号分隔的文件，其中每行对应于图像中的一个文本串，并具有以下格式：

X1, Y1, X2, Y2, X3, Y3, X4, Y4, “文本”

其中 X1, Y1, Y2, X2, X3, X4, Y3, Y4 分别代表文本的外接四边形四个顶点坐标。而“文本”是四边形包含的实际文本内容。

图 2 是标注的图片，红色的框代表标注的文本框。图 3 是标注图片对应的文本文件。标注时我们对所有语言，所有看不清的文字串均标注了外接框（比如图 2 中的小字），但对于除了中文，英文以外的其它语言以及看不清的字符并未标注文本内容，而是以“###”代替。



图 2 : image.jpg

```

494. 91, 36. 36, 494. 91, 81. 45, 596. 0, 81. 45, 595. 27, 32. 73, 三星
614. 91, 34. 91, 614. 91, 77. 82, 783. 64, 77. 82, 783. 64, 34. 91, N7100
524. 73, 93. 82, 524. 73, 146. 18, 784. 36, 146. 18, 784. 36, 93. 82, 钢化玻璃膜
164. 0, 143. 27, 164. 0, 157. 09, 251. 27, 157. 09, 251. 27, 143. 27, SNMSUNG
316. 73, 174. 55, 316. 73, 189. 09, 353. 82, 189. 09, 353. 82, 174. 55, 17: 39
117. 45, 236. 36, 117. 45, 284. 36, 298. 55, 284. 36, 298. 55, 236. 36, 17: 39
142. 91, 292. 36, 142. 91, 309. 82, 268. 73, 309. 82, 268. 73, 292. 36, 9月12日星期三
262. 91, 354. 91, 262. 91, 367. 27, 345. 82, 367. 27, 345. 82, 354. 91, 小到中雨转阵雨
321. 82, 338. 91, 321. 82, 350. 55, 345. 82, 350. 55, 345. 82, 338. 91, 北京
70. 57, 378. 86, 70. 57, 384. 57, 87. 14, 384. 57, 87. 14, 378. 86, ###
88. 86, 376. 57, 88. 86, 385. 71, 118. 57, 385. 71, 118. 57, 376. 57, 新浪天气
206. 0, 371. 43, 206. 0, 381. 71, 324. 29, 381. 71, 324. 29, 371. 43, 已更新2012/09/1116:59
76. 86, 545. 71, 76. 86, 556. 0, 106. 0, 556. 0, 104. 86, 546. 29, 沃·3G
150. 57, 545. 71, 150. 57, 557. 71, 183. 71, 557. 71, 183. 71, 545. 71, 沃商店
226. 57, 546. 86, 226. 57, 557. 71, 264. 29, 557. 71, 264. 29, 546. 86, 116114
294. 57, 545. 14, 294. 57, 557. 71, 350. 0, 557. 71, 350. 0, 545. 14, 手机营业厅
67. 71, 682. 86, 67. 71, 693. 14, 100. 29, 693. 14, 100. 29, 682. 86, 联系人
135. 71, 682. 29, 134. 57, 694. 29, 156. 29, 694. 29, 156. 29, 681. 14, 手机
195. 14, 682. 86, 195. 14, 693. 71, 218. 57, 693. 71, 218. 57, 682. 86, 信息
251. 14, 682. 86, 251. 14, 692. 57, 282. 57, 692. 57, 282. 57, 682. 86, 互联网
307. 14, 681. 71, 307. 14, 693. 14, 349. 43, 693. 14, 349. 43, 681. 71, 应用程序
232. 29, 513. 14, 232. 29, 522. 86, 261. 43, 522. 86, 261. 43, 513. 14, 116114
150. 0, 514. 29, 150. 0, 522. 86, 172. 86, 522. 86, 172. 86, 514. 29, W0
153. 43, 523. 43, 153. 43, 532. 0, 180. 86, 532. 0, 180. 86, 523. 43, ###
76. 29, 522. 29, 76. 29, 529. 14, 105. 43, 529. 14, 105. 43, 522. 29, ###
76. 29, 504. 0, 76. 29, 514. 86, 107. 14, 514. 86, 107. 14, 504. 0, W0
69. 43, 341. 14, 69. 43, 366. 86, 125. 43, 366. 86, 125. 43, 341. 14, 29° C
141. 43, 340. 0, 141. 43, 366. 86, 195. 71, 366. 86, 195. 71, 340. 0, 17° C
547. 71, 701. 14, 547. 71, 746. 86, 784. 29, 746. 86, 784. 29, 701. 14, 送贴膜工具
568. 29, 554. 86, 588. 29, 608. 0, 792. 29, 534. 29, 775. 14, 481. 71, 防爆防刮
551. 14, 449. 14, 577. 43, 510. 86, 795. 71, 434. 29, 785. 43, 372. 57, 智能贴合
542. 0, 344. 0, 564. 29, 396. 57, 788. 86, 323. 43, 771. 71, 265. 71, 防指纹油
659. 14, 543. 43, 659. 14, 544. 0, 659. 71, 544. 0, 659. 71, 543. 43, ###

```

图 3 : image.txt

### 3.任务描述

网络图像的文本检测：检测并定位图像中的文字行位置，允许使用其它数据集或者生成数据，允许 Fine-tuning 模型或者其他模型。入围团队提交报告中须对额外使用的数据集，或非本数据集训练出的模型做出说明。

**训练集：**

对于每个图像，只需要用[图像文件名].txt 里的坐标信息。即： $X_1, Y_1, X_2, Y_2, X_3, Y_3, X_4, Y_4$ 。

**测试集：**

输入：整图

输出：对于每一个检测到的文本框，按行将其顶点坐标输出到对应的[图像文件名].txt 中。

**提交：**

将所有图像对应的[图像文件名].txt 放到一个 zip 压缩包中，然后提交。

4. 评估标准

文本定位评测遵循 ICDAR2013 Born-Digital Image 的主体思路。本次竞赛数据集以中文为主，标注较细致，所以按照论文中“one to many”和“many to one”<sup>[1]</sup>的思路更为准确。其中一些阈值进行了调整：

**第一**，为“单个目标框”筛选合格“多个合格框”的阈值  $t_{many}$ 。“多”中的任意框与目标框交叉面积除以自身面积大于  $t_{many}=0.7$  时，视为合格候选。

**第二**，计算“多个合格框”覆盖“单目标框”的面积阈值  $t_{one}$ 。“多”中的所有框覆盖了目标框总面积大于  $t_{one}=0.7$  时，视单目标框可被召回或者属于正确检测范畴，视“多个合格框”为可被召回或者属于正确检测范畴。

**第三**，确定召回率和精度。计算“多”检测框对“单”标注框时，如果满足了  $t_{one}$ ，那么单标注框召回率为 1，多个检测框(个数为 k)的检测准确度为  $penal(K)$ 。计算“多”标注框对“单”检测框时，如果满足  $t_{one}$ ，那么单检测框精度为 1，多个标注框(个数为 k)中每一个召回率为  $penal(K)$ 。其中  $penal(K)$  为惩罚“分散”或者“合并”错误的函数，公式为：

$$penal(K) = 1 / (1 + \ln(K)) \quad (1)$$

**第四**，处理“可忽略行”。对于行标注内容为“###”的文本行。“可忽略行”不计算召回率。当某个检测框被“可忽略行”覆盖的面积除以自身面积大于  $t_{ignore}=0.5$  时，视该检测框为“可忽略检测行”。可忽略的标注行和检测行不计入最终结果。