硕士学位论文

基于 2D 注意力机制的
不规则场景文本识别算法

2D ATTENTION SCHEME BASED IRREGULAR SCENE TEXT RECOGNIZER

杨志成

哈尔滨工业大学

2019 年 7 月

国内图书分类号：TP391.4　　学校代码：10213
国际图书分类号：681.3　　密级：公开

工程硕士学位论文

基于 2D 注意力机制的不规则场景文本识别算法

硕 士 研 究 生：杨志成

导　　师：吴晓军 副教授

申 请 学 位：工程硕士

学　　科：控制工程

所 在 单 位：哈尔滨工业大学（深圳）

答 辩 日 期：2019 年 6 月

授予学位单位：哈尔滨工业大学

Classified Index: TP391.4
U.D.C: 681.3

A dissertation submitted in partial fulfillment of
the requirements for the professional degree of
Master of Engineering

2D ATTENTION SCHEME BASED IRREGULAR SCENE TEXT RECOGNIZER

Candidate: Zhicheng Yang
Supervisor: Associate Prof. Xiaojun Wu
Academic Degree Applied for: Master of Engineering
Speciality: Control Engineering
Affiliation: Harbin Institute of Technology, Shenzhen
Date of Defence: June, 2019
Degree-Conferring-Institution: Harbin Institute of Technology

摘 要

识别不规则场景文本是光学字符识别（OCR）问题中较为困难的子问题，该问题对学术界所提出的字符识别算法非常具有挑战性。目前，工业上实际应用的算法分为三类：将不规则场景文本通过薄板样条函数插值（thin plate splines）成规则场景文本后，再进行识别，即由 2D 布局转成 1D 布局；提取 2D 图像特征，通过卷积神经网络和带有循环注意力机制神经网络，降维成 1D 特征序列，再进行识别；将 2D 图像转换为 1D 特征序列，然后通过从自然语言处理领域借鉴过来的联结主义时间分类器（connectionist temporal classification）算法，进行识别。尽管上述方法取得了较好的表现，但是准确率和鲁棒性仍然受限于 2D 到 1D 转换过程中空间信息的丢失。本文将 2D 布局的不规则场景文本通过 2D 注意力机制，直接预测字符序列。

本文提出将不规则场景文本识别分为 2D 特征提取模块、关系注意力模块和并行注意力模块，共计三个模块。其中，对于 2D 特征提取模块，本文将在现有文本图像 2D 特征提取算法的基础上，进行算法改进，在获取上下文语义信息的同时，保留 2D 空间信息，避免了 2D 到 1D 转换过程中，空间信息的丢失，该部分作为网络的编码器；关系注意力模块用于将 2D 特征提取器所输出的特征图，进行更进一步的上下文信息提取，获取更高维的语义信息。并行注意力模块用于将关系注意力模块的输出，进行注意力加权，并将加权后的特征图送入后续的同步解码器，预测所有的字符序列，上述模块是并行计算结构，将会提高算法的效率和准确率。

本文将针对提出的不规则场景文本识别算法，在公开数据集和本文提出的多行车牌文本识别数据上进行扩展实验，同时进行可视化分析和必要性讨论。实验证明，本文提出的算法在规则和不规则文本识别问题上是高效的。在速度上，比之前所提出的文本识别算法快 2.1 倍；在精度上，针对不规则场景文本数据集，准确率超出之前所提出的算法高达 7.3%。

关键词：场景文本识别；并行计算；注意力机制

Abstract

Irregular scene text, a difficult subproblem of Optical Character Recognition(OCR), which has complex layout in 2D space, is challenging to most previous scene text recognizers. Recently, all methods in academia and industry will be splited into three categories. Some irregular scene text recognizers rectify the irregular text to regular text image with approximate 1D layout via thin plate splines(TPS). Some transform the 2D image feature map to 1D feature sequence, and others cite the notion of connectionist temporal classification(CTC), which is from natural language processing. Though these methods have achieved good performance, the robustness and accuracy are still limited due to the loss of spatial information in the process of 2D to 1D transformation. In this paper, we will predict characters of 2D layout irregular scene text directly via 2D attention mechanism.

In this paper, we propose three modules to recnogize irregular scene text, which are 2D feature extraction module, relation attention module and parallel attention module, respetively. Different from all of previous, we propose a framework which transforms the irregular text with 2D layout to character sequence directly via 2D attentional scheme.To address the problem of irregular scene text recognizing. In this paper, three moduls are proposed to deal with the information loss due to the process of 2D to 1D transformation. First, 2D feature extraction module is proposed. This module is used to extract high-level feature of 2D image. And to avoid information loss, this module modifies the backbone of powerful residual network. Wath's more, this paper utilizes a relation attention module to capture the dependencies of feature maps and a parallel attention module to weight the output of relation attention module, and parrallel attention module will give attentional weighted output to decoder, which will decode all characters in parallel. To sum up, parallel module will make our method more effective and efficient.

Extensive experiments on several public benchmarks as well as our collected multi-line text dataset show that our approach is effective to recognize regular and irregular scene text and outperforms previous methods both in accuracy and speed. In terms of speed, our proposed method is 2.1 times faster than other algorithms proposed by academia . As for accuracy, ours is 7.3% higher than others in terms of irregular scene text dataset.

Keywords: irregular scene text recognition, parallel computing, attention scheme

目 录

第 1 章 绪 论

1.1 课题背景及研究的目的和意义

近几年，从图像中获取文本信息，尤其在自然场景下，已经引起学术界和工业界的广泛关注，因为在大量场景下都有实际应用的需求。文本识别作为光学字符识别（OCR）系统不可分割的一部分，对最终识别准确率和速度产生重要影响[1]-[9]。

尽管之前提出了很多方法[3]-[16]，场景文字识别仍然是一个非常具有挑战性的问题。识别场景文字不仅面临着复杂的背景和多变的字体字形变化。同时，在 2D 空间上不规则的布局也增加了识别难度。更进一步的是，前几年提出的大量识别算法是用来识别近似 1D 空间布局的规则文本，这些算法没有能力去处理不规则的文本图片，比如文本分布是弯曲或者多行文本情况，即文本是分布在 2D 图像空间。

本文旨在基于 2D 注意力机制，设计一个高效率和高精度的不规则场景文本识别算法。本文提出了 2D 关系注意力模块和并行注意力模块，以此提高框架的可用性、鲁棒性和高效性。提出了一个新的包含多行文本的数据集，这是学术界第一次提出并在多行文本上进行性能测试和比较的算法，以此验证本文提出算法的有效性。

1.2 国内外研究现状

本文针对的问题是不规则场景文本布局识别任务。不同的文本图片特征有较大差异，同时，复杂的背景干扰和不规则的字形、字体等也会对识别产生很大影响。本文算法主要目标是识别不规则场景文本，算法主要创新点在于通过 2D 注意力机制，直接预测字符序列，避免了之前算法中空间信息的丢失。

1.2.1 普通场景文本识别

最近几年，提出了大量的算法来识别自然场景中的文本，基于这些方法的特征，可以大致分为三类：基于单个字符的识别方法，基于单词分类的识别方法，基于文本序列的识别方法。早期的识别算法大量[6]-[12]都是基于单个字符识别，具体方法是先检测和识别单个字符，然后把单个字符的识别结果聚合成字符序列，即文本内容。在很多类似的方法中，字符的候选框是通过连通域或者滑窗算法[11][12][16]生成的，然后再通过一些手动设置的特征，比如 HOG 算子[16]或者可学习的

特征[8][10]，最后单个字符通过一些启发式算法聚合成字符序列，如图 1-1 所示，先找到每个字符的位置并进行预测，最后再通过连通域聚合成字符序列。该算法面临手动设置特征的局限性，复杂场景下表现较差，鲁棒性和有效性不足的缺点是该类算法面临的问题。

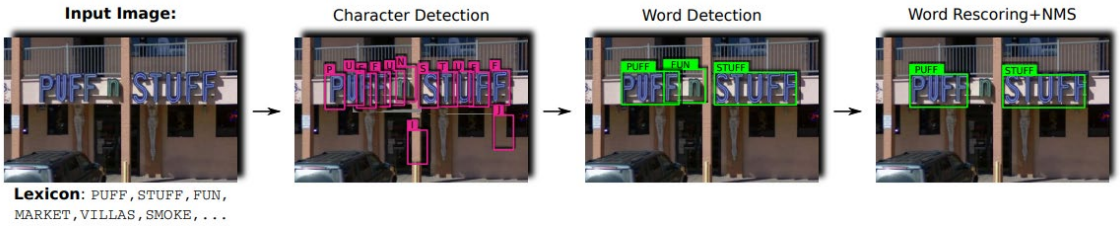

图 1-1 基于单个字符的识别方法

随着深度学习的发展，卷积神经网络[17]（CNN）和循环神经网络[18][19] (RNN)相继提出，并取得了重大的发展。Jaderberg[3][9]等人通过训练大规模的基于单词的分类器来解决字符识别问题。基于 800M 计算机合成图片，训练了一个强有力的多标签分类器，分类的标签为 90K 的单词表。如图 1-2 所示，一共是三个基于分类器的算法，图 1-2-a 是对单词直接进行分类，这样后续会存在一个很大的全连接层，效率不高且参数量巨大。图 1-2-b 是最后接 23 个字符分类的分类器，每个分类器预测一个字符，综合 23 个分类器的结果可以得到最终预测字符序列。图 1-2-c 是进行字符袋预测，原理是表达一个字符序列并不需要预测该字符序列的所有字符，可以通过预测字符序列中的某些特征部分，通过该特征部分直接映射会原字符序列，借鉴了哈希表的思想。然而该方法面临一个难以解决的问题，就是最终预测字符类别局限于一个字典里面，如果出现不在字典中的文本，最终将会错误预测。比如出现一个非英语单词或者是字母和数字的组合情况。

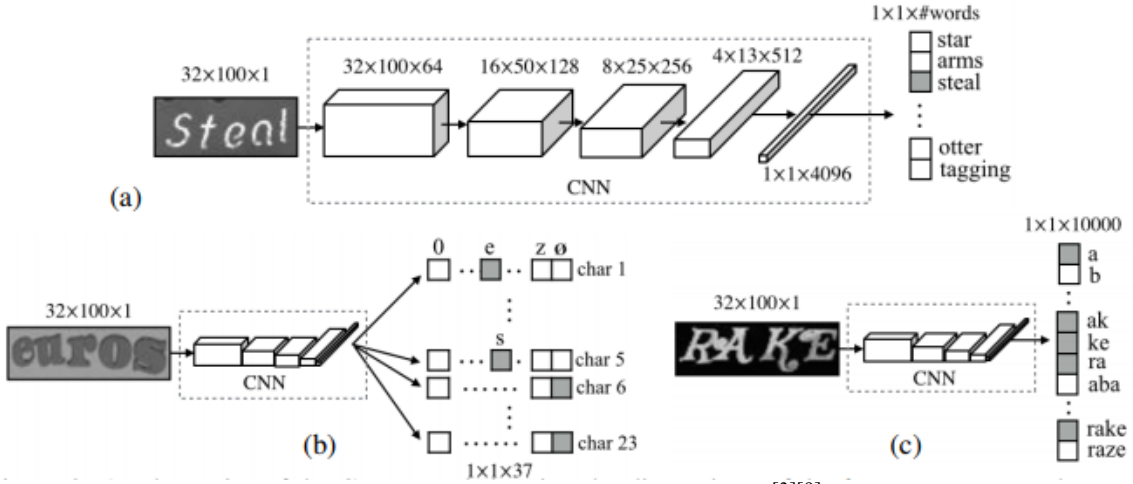

图 1-2 基于单词分类的识别方法[3][9]

后续又出现了通过序列到序列的识别算法[13][14][15]，Shi[14]等人用 CNN 和 RNN 提取图片特征，并将特征序列化，最后通过 CTC[20]来获取字符序列。

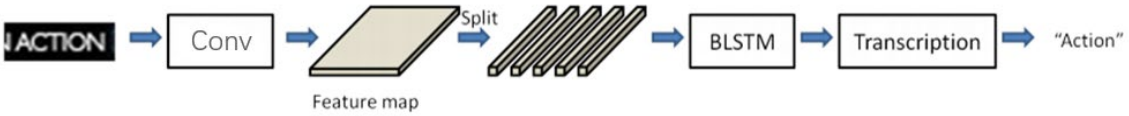

图 1-3 基于字符序列的识别方法[14]

如图 1-3 所示，该类方法首先将场景文本图像输入到特征提取网络中，然后对特征提取，将 2D 图像降维成 1D 特征序列。接下来将特征序列串行的输入循环神经网络，获取上下文信息。最终又将包含着上下文信息的特征序列，通过 CTC 进行“软对齐”，并计算损失函数，进行网络参数优化更新。该类方法面临着的问题是不同位置的特征序列在预测时具有相同的权重。而实际情况是在预测后面的字符序列时，需要对当前位置的特征序列给予较大的注意力，来自当前位置前面或者后面的特征序列，并不应该给予同样的注意力。

随着注意力机制[17][21]在自然语言处理领域的成功，有学者迁移了注意力机制到场景文本识别领域，Lee[13]和 Shi[14]等人提出了用注意力机制生成序列信息，再对序列信息进行逐步解码。如图 1-4 所示，将 2D 特征图降维 1D 成特征序列后，输入特征循环注意力网络中，对每个位置都学习一个注意力参数，方便对提取的当前位置的特征序列加权。该类方法可以很好的解决规则场景文本识别问题，然而针对不规则场景文本识别问题，该类方法缺少鲁棒性和有效性，分析其本质原因是在 2D 特征图转换成 1D 特征序列这个操作，导致空间信息不可避免的损失。空间信息损失后，针对识别不规则场景文本的问题时，就很难用注意力机制给对应位置较大的权值。后续的该类算法识别效果的提升，更多是依赖于前面编码网络部分，出现的强有力特征提取器，而非整体算法设计上的改进。

同时，从图 1-4 中可以明显看出循环注意力机制当前时刻的计算值，取决于上一时刻的计算结果，这样的串行计算极其消耗运算资源，也降低了运算效率。由于这样的网络结构无法并行运算，硬件的更新换代对该类算法运行速度的改进就不明显。制约该类串行预测文本的算法速度的原因将在第二章予以公式推导和分析，在第三章进行针对性改进。

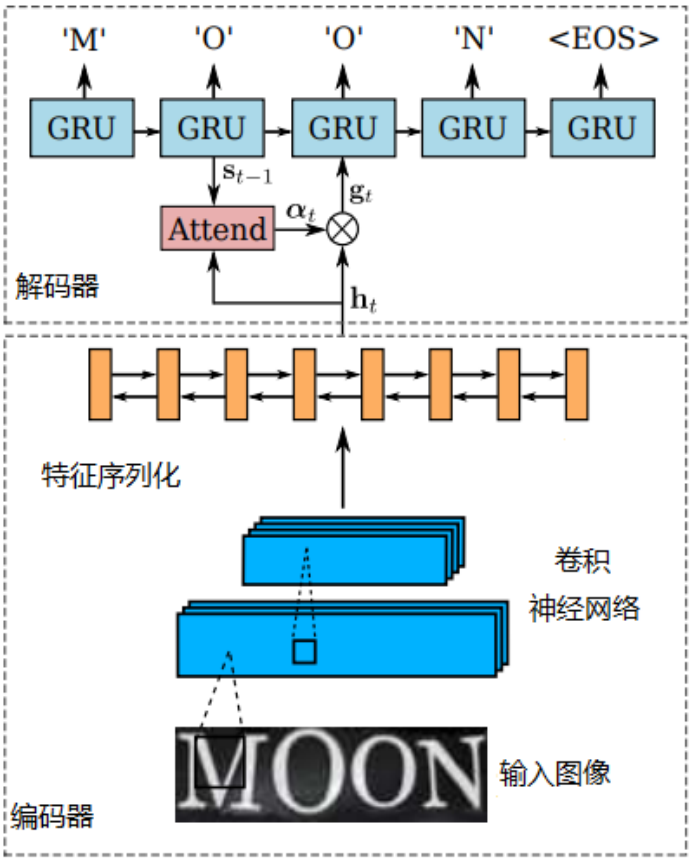

图 1-4 基于注意力机制的识别方法[14]

1.2.2 不规则场景文本识别

最近几年，不规则场景文本识别也吸引了很多注意力，成为研究的热点方向。Shi[14]和 Luo[23]等人提出用联合的网络去识别不规则场景文本。首先，先用一个校正网络校正不规则文本图像，得到规则文本图像，然后再用识别网络识别校正后的图像。这类方法的校正网络主要是用基于空间转换网络[26]（STN），通过该网络校正不规则场景文本，将不规则场景文本校正成规则场景文本。规则场景文本较易识别，后续识别主要是通过基于序列的识别方法。空间转换网络主要由两部分组成。其中一部分是位置回归网络（Localization Network），该网络主要是为了回归出薄板样条插值函数所需要的原图控制点坐标。另一部分是网格生成器（Grid Generator），通过网格生成器将目标图中的各个像素坐标，通过矩阵变换，对应到原图中的位置，再通过采样器对原图中的相应位置采样，得到校正后的图像。再对校正后的图像进行后续识别，识别算法可以是基于循环注意力神经网络，也可以是基于 CTC 损失函数的方法。该方法可以识别一些倾斜或者是弯曲程度不大的不规则场景文本。如果是弯曲幅度较大的极其不规则场景文本，或者是多行场景

文本，该类方法还是不能很好的解决。原因是薄板样条函数插值的能力有限，最终采样出来的校正图像，会有不同程度的缺失或者是扭曲。为后续识别算法增加了难度。

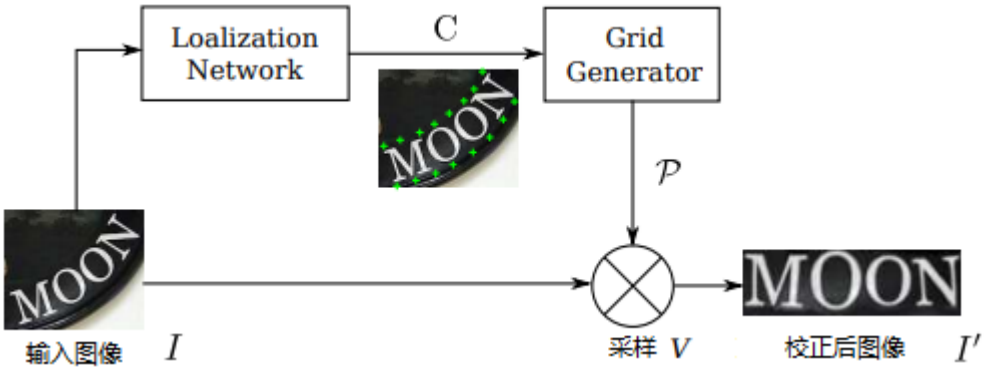

图 1-5 空间转换网络示意图[26]

不同于上述方法，Liu[24]等人提出循环的校正单个字符。同时，一些通过 2D 透视变换识别不规则文本图像的算法[29][30]相继提出。为了识别任意方法的不规则文本，Cheng 等人从四个方向获取特征信息，以此获取 2D 空间信息，改进了基于序列的方法。后续又提出了基于 2D 注意力机制的识别算法，通过将注意力机制应用到特征图上来识别不规则场景文本。Lyu[5]和 Liao[30]通过字符级别的标注，应用像素级分割的方法识别不规则场景文本。如图 1-6 所示，图 1-6-a 展示的是序列到序列方法的示意图，图 1-6-b 是利用分割网络对单个字符位置进行分割，分割之后判定是否存在字符，然后对存在字符的位置预测字符类别，并通过简单的几何关系判断字符序列顺序，得到最终的预测字符序列。该方法可以很好的解决不规则空间文本识别问题，但是面临两个难以避免的问题。其中一个问题是，因为是基于字符分割的方法，所以需要精确到字符级的标注，然而这样的数据集标注起来非常耗时，成本较高。另一个问题是，采用的较大的分割网络，参数和计算量爆炸式的增长，导致算法效率降低。

同时，如图 1-6-b 所示，基于像素级分割这类算法需要对字符位置有明确的划分，如果出现字符过于接近，就会导致分割时，不同字符连通域分不开，这样后续的预测难度会极大增加。另外，该类算法还面临一个不可避免的问题，分割出来的连通域是无序的，在聚合成字符序列这步后处理操作时，只是简单采用了从左向右的 1D 空间位置顺序进行排序，这样针对 2D 布局分布的多行文本，该类方法将会失效。

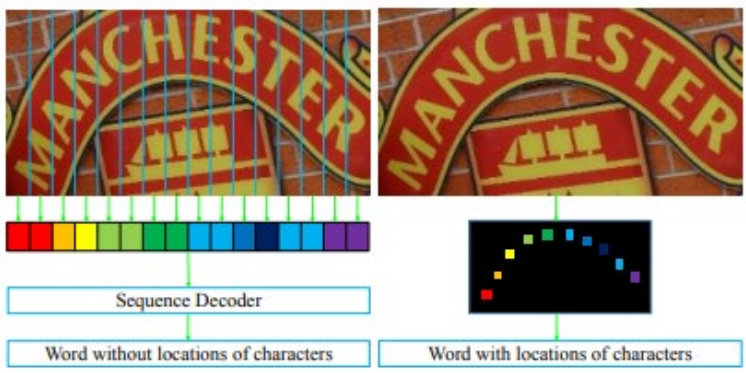

图 1-6 (a)基于序列到序列模型，(b)单个字符掩膜分割识别[30]

1.2.3 研究现状总结分析

如图 1-7 所示，近几年识别不规则场景文本的框架，第一分支是基于校正网络和识别网络算法，第二分支是基于 2D 透视变换算法框架，第三分支是进行字符级分割，并根据连通域聚合成字符序列。第四分支是通过对特征图应用 2D 注意力机制进行编码，再通过循环神经网络进行依次解码。第四分支是本文提出的基于关系注意力模块和并行注意力模块，直接同时并行预测所有字符，不需要循环神经网络，保留了 2D 空间信息，也避免循环神经网络串行算法效率较低的问题。

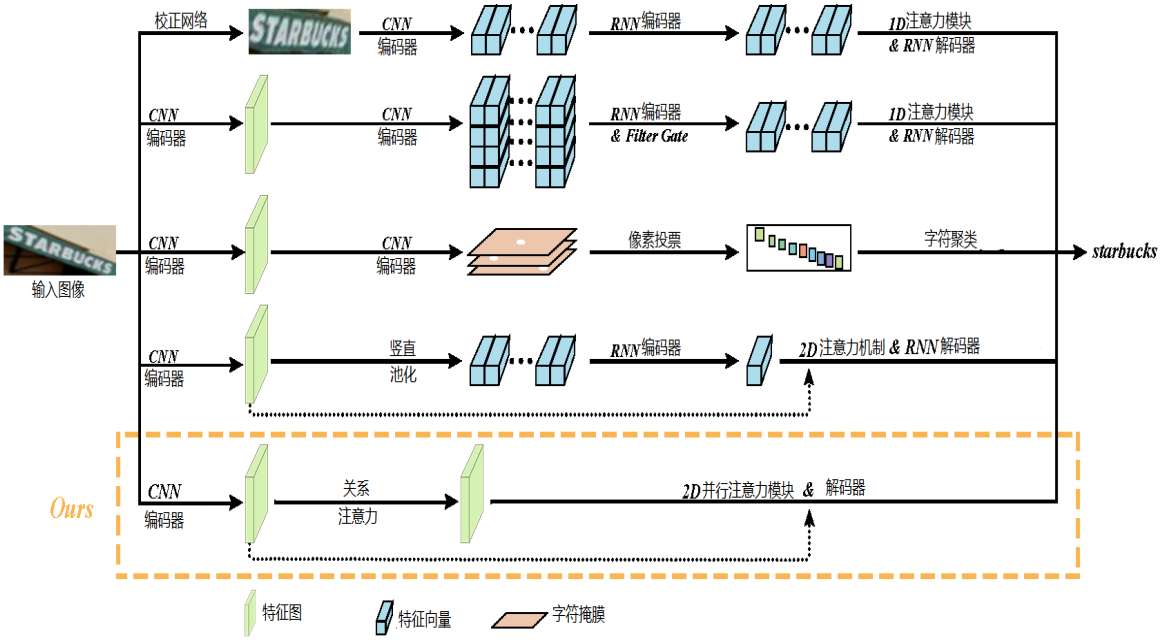

图 1-7 近几年提出的不规则文本识别算法流程

1.3 本文主要研究内容

本文主要在不规则场景文本识别问题上开展研究。设计一个端到端的并行识别网络，无需复杂的后处理，也避免了之前算法串行识别低效率的问题。本文的主要研究内容：

（1）介绍不规则场景文本识别的背景以及研究意义。综述目前场景文本识别算法。对现有文本识别算法的特点进行探究，同时说明本文提出的不规则文本识别算法的主要优势。

（2）提出应用到不规则场景文本识别算法，算法主要分为三个模块：2D 图像特征提取模块、关系注意力模块和并行注意力模块。在 2D 图像特征提取模块，改进现有场景文本识别方法，保留 2D 空间信息；在关系注意力算法上，根据不规则场景文本识别面临的实际问题，不同于之前的方法，提出了关系注意力模块替换循环神经网络，避免了串行计算；在并行注意力算法方面，提出用并行注意力机制来实现上下文通信，对关系注意力模块输出的特征图进行加权，获取全局文本信息。由此，本文提出的三个模块都是并行计算，从而避免了串行计算效率较低的问题，可以同时将不规则场景文本中的所有字符进行同时预测。

（3）提出含有多行文本的数据集，并在多行文本上进行实验，这是学术界首次在多行文本数据集上进行算法验证。

（4）实验验证本文所提出算法的性能。本文在 7 个包括规则和不规则的场景文本公开数据集上进行比较，本文提出的算法在绝大多数数据集上取得了最优的结果，证实本文提出的算法是具有优势。其中，在弯曲场景文本数据集 CUTE80，本文提出的算法超过之前最好的算法[22]7.3%的准确率。同时，为了验证本文提出算法在复杂场景的有效性，在含有多行和单行文本的数据集进行测试和算法比对。实验结果表明，本文提出的算法超出基于校正的识别算法[22]18.2%，超出基于循环 2D 注意力机制算法[27]29.6%，证明了本文提出算法的鲁棒性。更进一步，本文提出的算法比上述算法分别快 2.1 倍和 4.4 倍。

第 2 章 CTC 与循环注意力机制

当前，学术界所提出的算法框架可以根据损失函数不同，分为两类。其中一类是通过联结主义时间分类器[20]（CTC），对预测结果与标签之间进行软对齐，并计算损失函数。另一类是通过循环注意力神经网络求取交叉熵，以此作为损失函数，进行反向传播。在提出第三章的创新性算法框架之前，对上述两类算法进行了深入的研究和思考，做了很多前期工作。第三章所提出的算法，避免了上述算法的缺点，也保留住了其优点。同时，因为基于 CTC 损失函数和循环注意力机制的方法，在场景文本识别中具有重要地位，也是后续研究的算法和理论基础。

2.1 CTC 机制

通过场景文本图像，直接预测出图像内的本文内容。这样的实际问题，可以抽象成从一个未进行标签分割的数据中，进行模型参数学习，从而得到所需标签序列。循环神经网络由于需要预先分割好的数据，同时也需要后处理将输出转化为标签序列，并不是解决这类问题比较好的方法。CTC 机制的出现，解决了未分割数据映射到标签序列的问题。受益于 CTC 机制，可以端到端的直接从未分割的图像特征对应的标签序列。

2.1.1 算法推导

首先定义目标函数，不妨设输入的场景文本图像是 $\mathbf{x}$，对应的字符标签概率 $\mathbf{z}$，则 $p(\mathbf{z}\,|\,\mathbf{x})$ 表示，当输入文本图像为 x 时，输出是 z 的概率。则待优化的目标函数可以定义为：

$$O^{ML}(S,\mathcal{N}_w)=-\sum_{(x,z)\in S}\ln(p(\mathbf{z}\,|\,\mathbf{x})) \tag{2-1}$$

其中，S—— 整个训练集；

$\mathcal{N}_w$—— 最终网络预测。

为了实现反向传播，故对公式 2-1 进行求导，因为每个训练样例是互相独立的，故可以单独进行求导，

$$\frac{\partial O^{ML}(\{(\mathbf{x},\mathbf{z}\},\mathcal{N}_w)}{\partial y_k^t}=-\frac{\partial\ln(p(\mathbf{z}\,|\,\mathbf{x}))}{\partial y_k^t}=-\frac{1}{p(\mathbf{z}\,|\,\mathbf{x})}\frac{\partial p(\mathbf{z}\,|\,\mathbf{x})}{\partial y_k^t} \tag{2-2}$$

其中，y_k^t—— 第 k 个节点 t 时刻的输出。

同时，对网络预测的概率输出映射到对应的标签序列方法进行研究。用 β 来定义一个映射关系，

$$\beta(a-ab-)=\beta(-aa-abb)=aab \tag{2-3}$$

定义一个条件概率如式 2-4 所示，π 指的是所有可能的路径

$$p(l\,|\,\mathbf{x})=\sum_{\pi\in\mathcal{B}^{-1}(\mathbf{l})}p(\pi\,|\,\mathbf{x}) \tag{2-4}$$

其中，$p(\pi\,|\,\mathbf{x})$指的是给定一个图像$\mathbf{x}$，输出预测路径为π的概率，等于每个时刻每个节点预测为π_k的概率乘积。

$$p(\pi\,|\,\mathbf{x})=\prod_{t=1}^{T}y_{\pi_t}^{t},\forall\pi\in L^{\prime T} \tag{2-5}$$

由公式（2-4）和公式（2-5）可知，CTC 从预测概率 y 结合路径 π 推导出预测标签 l 的方法。

定义前向变量与后向变量如公式 2-6 和公式 2-7 所示，

$$\alpha_t(s)=\sum_{\pi\in L^{\prime T}:\mathcal{B}(\pi_{1:t})=\mathbf{l}_{1:\frac{s}{2}}}\prod_{t'=1}^{t}y_{\pi_{t'}}^{t'} \tag{2-6}$$

$$\beta_t(s)=\sum_{\pi\in L^{\prime T}:\mathcal{B}(\pi_{t:T})=\mathbf{l}_{\frac{s}{2}:|\mathbf{l}|}}\prod_{t'=t}^{T}y_{\pi_{t'}}^{t'} \tag{2-7}$$

其中，$\alpha_t(s)$表征的是前向变量，指的是到t时刻时，预测前s个标签的概率。

同理，$\beta_t(s)$表征的是后向变量，指的是t时刻时，预测s到$|l'|$标签的概率。

则t时刻时，预测标签l的全概率为：

$$\alpha_t(s)\beta_t(s)=\sum_{\pi\in\mathcal{B}^{-1}(\mathbf{l}):\pi_t=\mathbf{l}'_s}y_{\mathbf{l}'_s}^{t}\prod_{t=1}^{T}y_{\pi_t}^{t} \tag{2-8}$$

将公式（2-8）带入公式（2-5）可得，

$$\frac{\alpha_t(s)\beta_t(s)}{y_{\mathbf{l}'_s}^{t}}=\sum_{\pi\in\mathcal{B}^{-1}(\mathbf{l}):\pi_t=\mathbf{l}'_s}p(\pi\,|\,\mathbf{x}) \tag{2-9}$$

将公式（2-9）带入公式（2-4）可得，

$$p(l\,|\,\mathbf{x})=\sum_{s=1}^{|\mathbf{l}'|}\frac{\alpha_t(s)\beta_t(s)}{y_{\mathbf{l}'_s}^{t}} \tag{2-10}$$

对上述结果进行整理可得，

$$\frac{\partial O^{ML}(\{(\mathbf{x},\mathbf{z}\},\mathcal{N}_w)}{\partial y_k^t}=-\frac{1}{p(\mathbf{z}\,|\,\mathbf{x})}\frac{1}{y_k^{t\,2}}\sum_{s\in lab(\mathbf{l},k)}\alpha_t(s)\beta_t(s) \tag{2-11}$$

2.1.2 算法分析

联结主义时间分类器[20]（CTC）对输入未分割的序列和对应的标注进行对齐。

该对齐机制，如图 2-1 所示。输入是序列图片，输出是对应的序列标签，但是两者并无直接的对齐关系，因为输入是连续的图像，输出是连续的标注序列，通过 CTC 机制可以将两者进行对齐。

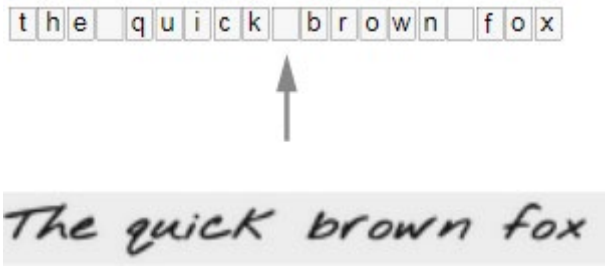

图 2-1 CTC 对齐机制

本文在研究过程中，观察到 CTC 机制在识别场景文本时，会遇到重复识别或者漏识别的问题，深入分析原因是因为 CTC 机制假设不同输出之间，条件独立。而实际在识别场景文本时，上下文并非严格独立，很多文本内容是自然语言，存在语言模型。同时，也因为 CTC 机制最终预测的不是标签序列，而是可能的所有路径，对最优路径的搜索是复杂度极高的问题，只能用贪心搜索（Greedy Search）或者束搜索(Beam Search)近似代替。

2.2 循环注意力机制

注意力机制最开始出现在自然语言处理领域，特别是神经机器翻译[21]。机器翻译问题是由一门语言外另外一门语言之间的转换，本质上是序列到序列的过程。类似地，场景文本识别是将图像中的像素序列转换成文本序列，也是序列到序列的过程。所以，场景文本识别任务可以借鉴自然语言处理的方法。本文第三章提出的算法，也是受到了自然语言处理算法的启发。注意力机制本质上是对所关注位置的特征向量给予较大的权值，比如机器翻译一句话时，着重关注的是翻译到对应位置的单词，而不是整个待翻译的句子。同样地，在场景文本识别过程中，识别到出于图像后半部分的字符时，主要关注的是来自后半部分的特征信息，而不是前半部分，故引入注意力机制可以提高识别准确率。

2.2.1 注意力编解码器

传统的编码-解码算法流程，会将所有的信息都编码成一个固定维度特征，该特征难以表征所有信息，缺乏对上下文位置的感知，公式推导过程如下，

$$h_t = f(x_t, h_{t-1}) \tag{2-12}$$

$$c = q(\{h_1, \cdots, h_{T_x}\}) \tag{2-13}$$

其中，h_i指的是编码器编码后的隐向量，t 指的是当前时刻，T_x 指的是最后时刻，f 和q是非线性函数。从公式 2-13 可以看出，所有的特征都被编码为一个上下文向量c。如图 2-2 所示，由公式 2-13 编码得到的特征向量c，解码时每个位置预测的结果都是取决于同一个上下文特征向量。为了提高解码后的准确率，实际需要的待解码特征是位置感知的，即需要结合自身位置信息进行加权，而不是所有位置都依赖同一个上下文向量。

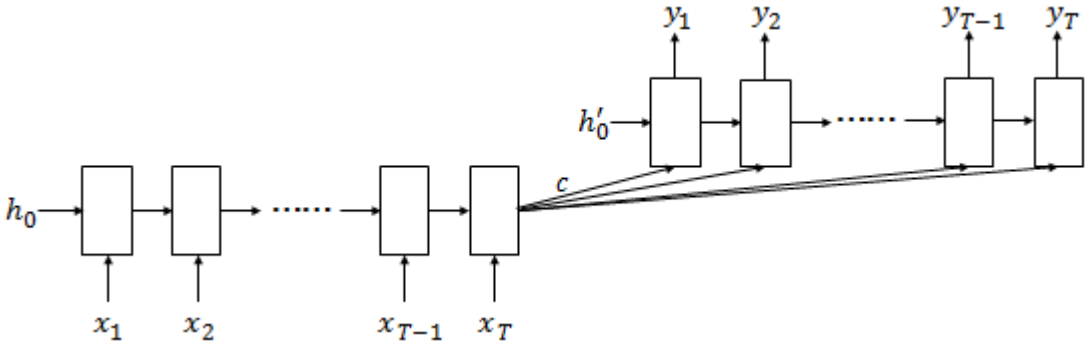

图 2-2 传统注意力机制的编码-解码流程

定义解码器如公式 2-14 所示，

$$p(y)=\prod_{t=1}^{T} p(y_t \mid \{y_1,\cdots,y_{t-1}\},c) \tag{2-14}$$

其中 $y=(y_1,\cdots,y_{T_y})$，指的是每个时刻对应输出预测的概率。条件概率如公式（2-15）式所示，

$$p(y_t \mid \{y_1,\cdots,y_{t-1}\},c)=g(y_{t-1},s_t,c) \tag{2-15}$$

其中 g 是非线性函数，s_t 是隐藏层的输出。

考虑到传统编码-解码的局限性，重新定义公式 2-15 中的条件概率，

$$p(y_i \mid y_1,\cdots,y_{i-1},x)=g(y_{i-1},s_i,c_i) \tag{2-16}$$

$$s_i=f(s_{i-1},y_{i-1},c_i) \tag{2-17}$$

其中，s_i是 RNN 第i时刻的隐藏状态，可由公式（2-17）计算得到。

注意力机制不同于之前的算法的地方，在于对上下文向量进行加权，

$$c_i=\sum_{j=1}^{T_x}\alpha_{ij}h_j \tag{2-18}$$

$$\alpha_{ij}=\frac{\exp(e_{ij})}{\sum_{k=1}^{T_x}\exp(e_{ik})} \tag{2-19}$$

$$e_{ij}=a(s_{i-1},h_j) \tag{2-20}$$

其中，α_{ij}是解码时$i-1$时刻的隐状态和编码时i时刻的隐状态通过非线性函数a组

合，再进过 softmax 函数计算得到的系数。通过公式（2-18）、（2-19）和（2-20），实现了对上下文向量加权，如图 2-3 所示。

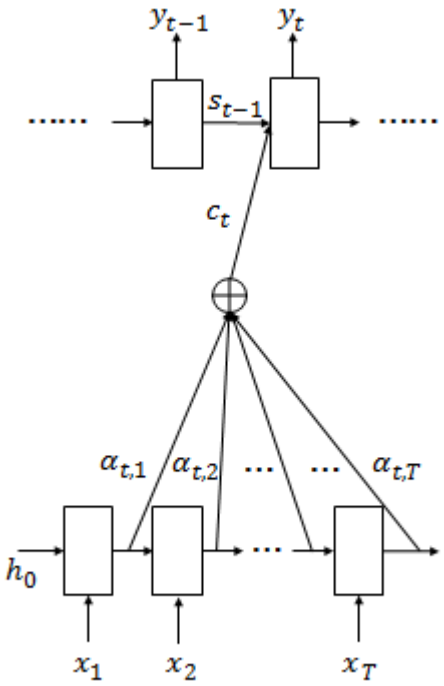

图 2-3 循环注意力机制

2.2.2 算法分析

基于循环注意力机制的算法，可以很好的处理序列到序列的问题，然而也面临着不可避免的局限性。从上述的公式可以看出，每个时刻的计算，都受制于上一个时刻的计算结果，这也是成为循环注意力机制的原因。由于计算的原理，导致该类方法不能并行，只能逐步串行执行，计算效率低。为了解决这个问题，第三章提出了关系注意力模块和并行注意力模块，避免了现有算法只能串行执行，不能并行运算的问题。

2.3 本章小结

本章对基于 CTC 机制的序列模型和基于循环注意力机制的序列模型，进行介绍。分别对算法的原理公式进行推导与算法存在的不足进行分析，上述两类序列模型是目前学术界场景文本识别领域的主流范式，但是存在的问题是十分明显，为了克服这两类方法遇到的问题，本文将在第三章介绍全新的算法框架，并在第四章进行实验与对比分析。

第 3 章 2D 注意力识别算法

本文提出了统一的网络，能够进行端到端的训练和测试。给定一张截取好文本内容的图片，网络可以直接预测识别结果，无需提前获取单个字符位置。网络结构如图 3-1 所示，我们首先用 CNN 编码图像，图像转换成具有高级语义信息的特征图。然后，关系注意力模块应用到 2D 特征提取模块所获取特征图的每个像素，捕获全局上下文信息。接下来，通过并行注意力模块处理关系注意力模块的输出，对输出特征进行加权，并输出固定维数的特征。最后字符解码器解码特征，产生预测的字符序列。

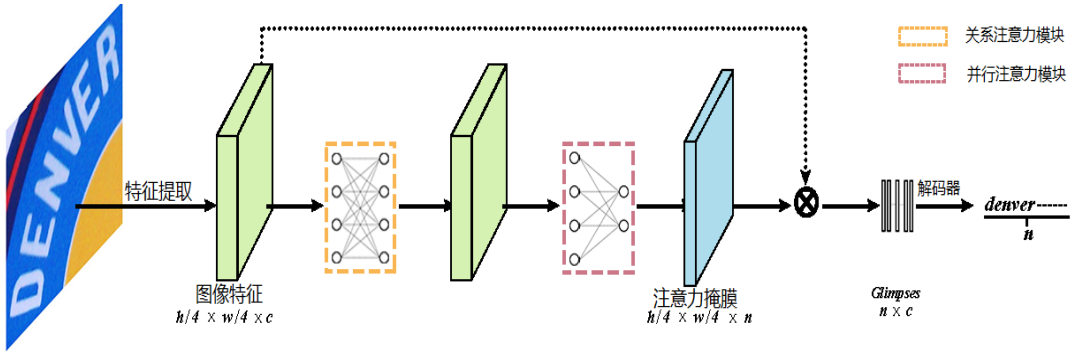

图 3-1 网络结构示意图

3.1 2D 特征提取模块

之前提出的算法，通常都是将 2D 图像经过特征提取网络后，降维成一维序列，这样不可避免地会损失空间信息，而空间信息对识别不规则场景文本图像是至关重要的。显而易见的，如果待识别的文本不规则地分布在 2D 图像空间上，将图像特征降维成一维的方法并不是很好的解决方案。同时，类似 Aster[22]之类的算法，在识别网络前面加一个薄板样条插值函数，先对图片进行校正。然而校正的方法也面临着图像倾斜或者弯曲过度时，校正后的图片会缺失一些边缘或者弯曲角度过大的部分。根据上述分析，本文提出 2D 特征提取模块，保留图像 2D 空间特征。

本文提出的 2D 特征提取模块，由五个子模块构成。其中，每个子模块是受到 Aster[22]采用残差网络结构 Residual Network 的启发，如图 3-2 所示，

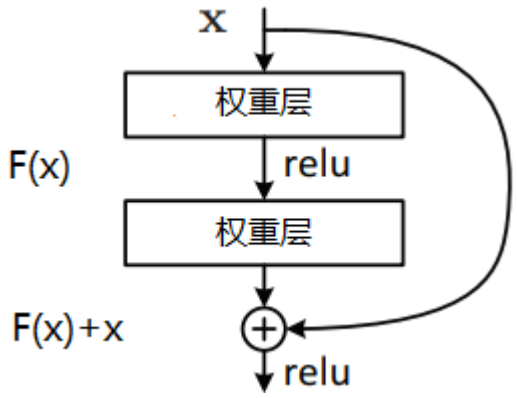

图 3-2 Residual Network 示意图

通过图 3-2 所示，可以看出残差网络结构最主要的特征是为了保留住梯度信息，从较低层传输的特征图与上层的特征图相加，融合成新的特征图。这样的结构优势在于，既能保留低层空间信息，同时又能保留高层语义信息。本文提出的 2D 特征提取网络中，采用了 5 个上述结构。本文的 2D 特征提取模块网络结构和参数配置如表 3-1 所示。

表 3-1 特征提取模块参数表

	网络层结构	输出维度	配置参数
编码器	模块 0	32×100	3×3 conv，s 1×1
	模块 1	16×50	$\begin{bmatrix}1\times1\ \text{conv},\ 32\\3\times3\ \text{conv},\ 32\end{bmatrix}\times3,\ \text{s}\ 2\times2$
	模块 2	8×25	$\begin{bmatrix}1\times1\ \text{conv},\ 64\\3\times3\ \text{conv},\ 64\end{bmatrix}\times4,\ \text{s}\ 2\times2$
	模块 3	8×25	$\begin{bmatrix}1\times1\ \text{conv},\ 128\\3\times3\ \text{conv},\ 128\end{bmatrix}\times6,\ \text{s}\ 1\times1$
	模块 4	8×25	$\begin{bmatrix}1\times1\ \text{conv},\ 256\\3\times3\ \text{conv},\ 256\end{bmatrix}\times3,\ \text{s}\ 1\times1$
	模块 5	8×25	$\begin{bmatrix}1\times1\ \text{conv},\ 512\\3\times3\ \text{conv},\ 512\end{bmatrix}\times3,\ \text{s}\ 1\times1$

本文提出的 2D 特征提取方法，主要是移除了 Aster 网络结构中 BiLSTM 和 Attention LSTM，为了保留了 2D 空间信息，本文把 Block 3 到 Block 5 中的结构进行改进，2×2 的步长卷积换为1×1 的卷积。通过这项改进，最终网络输出维度为 8×25，而不是传统卷积特征提取网络中输出1×25 的维度。从输出维度上，可以观察到 2D 空间信息没有被压缩成 1D，这也就是为后续不规则文本识别提供了充足

的特征信息。从表 3-1 中所列参数可以看出，模块 5 输出维度是8×25×512（$H \times W \times C$），为了降低计算量，本文在模块 5 后面，增加了一个1×1卷积，通过该卷积实现对输出特征图进行降维。

3.2 关系注意力模块

之前的算法[14][15][22]总是用 RNN 去获取经过 CNN 编码后的 1D 特征序列的上下文信息。考虑到计算效率，直接将 RNN 应用到 2D 特征图并不是很好的选择。Cheng[25]等人将 RNN 应用到从特征图四个方向提取出来的 1D 特征序列。Li[27]等人通过竖直方向的最大值池化，将 2D 特征图转换成 1D 特征序列。上述算法采用的策略某种程度上能减少计算量，但同时也丢失了空间信息。

受到一些捕捉输入和输出之前所有元素相互依赖关系和全局信息算法的启发，本文提出了关系注意力模块，其中包括了 transformer 结构[28]。关系注意力模块并行地捕获全局上下文信息，该策略比上述算法更高效。特别地，仿照 BERT 结构[53]，本文提出具有多层双向 transformer 编码器的关系注意力模块。

如图 3-3 所示，为了应用关系注意力网络到任意形状，本文提出把输入的特征图先转换成特征序列 I，其中 I 的形状为 $k \times c$。k 指的是展开成特征序列后，序列的长度。c 指的是特征序列中，每个特征向量的空间维度。针对每个特征向量 I_i，本文采用嵌入式操作，用位置向量 E_i 编码每个位置索引 i。其中，E_i 和 I_i 具有相同的维度。接下来，展开的特征序列 I 与嵌入位置信息的特征 E 相加，得到融合特征 F，此时融合特征 F 是对空间位置敏感的特征。

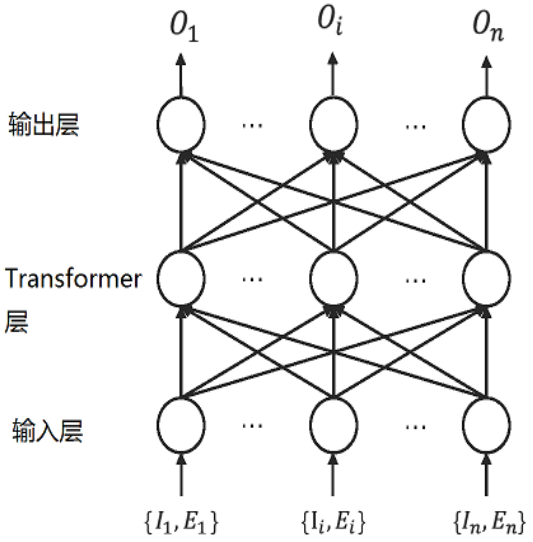

图 3-3 关系注意力模块示意图

每个 transformer 层都包括 k 个 transformer 单元，针对每个 transformer 单元，

如图 3-4 所示都包括 *query*， *keys* 和 *values*[28]，并且由下面的公式计算得到，

$$Q_l^i = \begin{cases} F_i & l=1 \\ O_{l-1}^i & l>1 \end{cases} \tag{3-1}$$

$$K_l^i = \begin{cases} F & l=1 \\ O_{l-1} & l>1 \end{cases} \tag{3-2}$$

$$V_l^i = \begin{cases} F & l=1 \\ O_{l-1} & l>1 \end{cases} \tag{3-3}$$

其中，Q_l^i 是查询向量 *query* 的第 l 层中第 i 个输入单元，其维度是 $1\times c$。K_l^i 和 V_l^i 是键向量 *keys* 和值向量 *values*，二者维度都是 $k\times c$。O_{l-1} 是上一层 transformer 结构的输出，其维度是 $k\times c$。

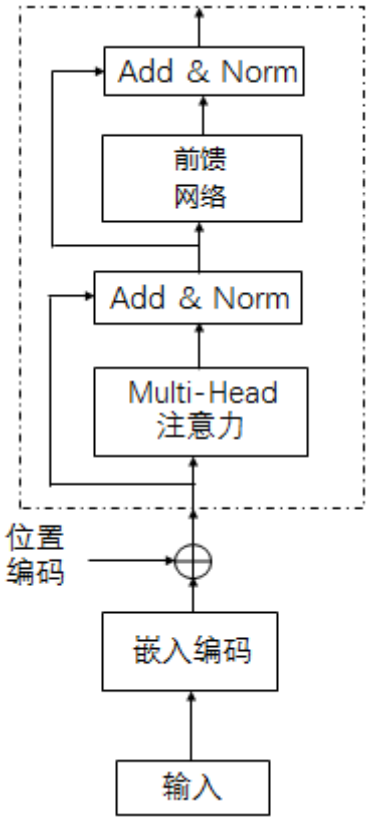

图 3-4 Transformer 网络结构[28]

以 *query*， *keys* 和 *values* 作为输入，transformer 结构的输出是通过计算一个加权求和操作，并把权值应用到 *values*。同时，每个 *values* 的权值通过下面公式计算，

$$\alpha_l^{ij} = \frac{e^{W_l^q Q_l^i \cdot W_l^k K_l^{ij}}}{\sum_{j'=1}^{k} e^{W_l^q Q_l^i \cdot W_l^k K_l^{ij'}}} \tag{3-4}$$

其中，W_l^q 和 W_l^k 是可学习的参数，α 是根据公式 3-4 计算出来的系数。每个 transformer 结构的输出可以通过下面加权求和公式得到，

$$O_l^i = Func(\sum_{j=1}^{k} \alpha_l^{ij} W_l^v V_l^{ij}) \tag{3-5}$$

其中，W_l^v 是可学习参数，$Func$ 是非线性函数，具体实现如图 3-5 所示。图 3-5-a 描述了整体结构，已经带有位置信息的特征 F_i，作为关系注意力模块的初始输入，维度是 $1\times c$。当输入关系注意模块后，具体网络运算如图 3-5-b 所示，Q_l^i 与 K_l^i 和 V_l^i 进行缩放点乘注意力计算（scaled dot-product attention）,计算方法如公式 2-4 所示。缩放点乘注意力计算过程是：Q_l^i 与 K_l^i 会得到一个加权系数 α，α 与输入的特征 V_l^i 进行加权，从而得到自注意力阶段的输出特征。需要明确的是 Q_l^i 与 K_l^i 和 V_l^i 三个输入特征，输入时是来自相同的特征图，可以理解为上述计算是自身进行的运算对应的加权系数，并不涉及其他特征，故该计算机制也称为自注意力机制（self-Attention）。

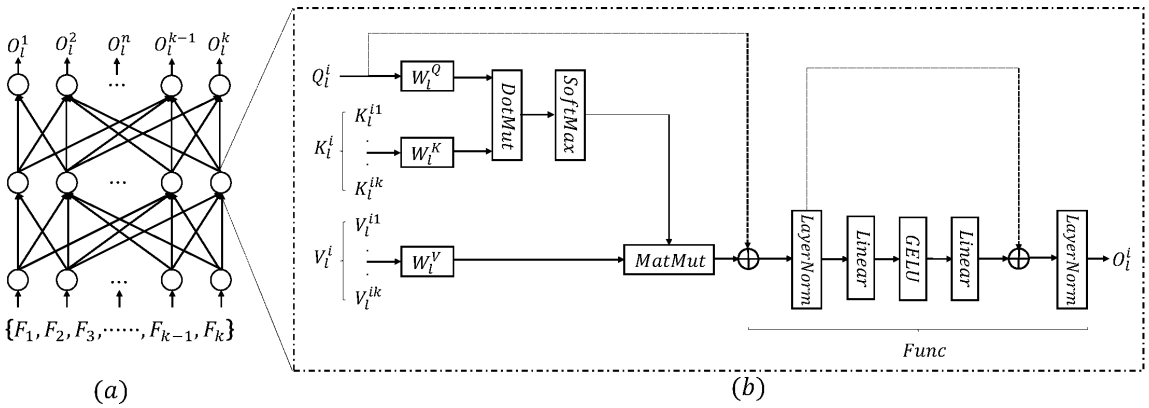

图 3-5 关系注意力网络结构

Func 如图 3-5-b 中的右半部分，是由层级归一化（LayerNorm）、线性层（Linear）和 GELU 层组成的一个非线性网络结构。上述非线性部分是为了增强网络的鲁棒性，避免过拟合，更好的提取高维语义信息。

本文提出的算法采用最后一层 transformer 结构的输出作为关系注意力模块的输出。

3.3 并行注意力模块

应用在算法中的基本注意力模块都是串行执行且集成在 RNN 中，如公式 3-6 所示，

$$\alpha_t = Attention(h_{t-1}, \alpha_{t-1}, I) \tag{3-6}$$

其中，h_{t-1}和α_{t-1}是上一时刻的 RNN 解码器的隐藏层和注意力权值，I 是编码后的图像特征序列。正如公式 3-6 所示，当前时刻的参数依赖上一时刻的计算结果，这样执行起来是低效率的。

与循环注意力机制不同，本文提出了并行注意力模块，这样输出节点之间的相互依赖关系被移除了，即每个输出节点的注意力计算是不相关的，并且能够容易并行实现和优化。

特别地，本文指定输出节点为n，给定一个特征序列O，其维度是$k \times c$，并行注意力机制输出的权值系数α采用下面公式进行计算，

$$\alpha = soft\max(W_2 \tanh(W_1 O^T)) \tag{3-7}$$

其中，W_1和W_2都是可学习参数，维度分别是$n \times c$和$c \times c$。

基于权值系数α和编码图像特征序列I，每个节点的输出为，

$$G_i = \sum_{j=1}^{k} \alpha_{ij} I_j \tag{3-8}$$

3.4 两阶段解码器

本文提出将并行注意力模块输出的结果，作为解码器的输入。对于每个输出节点，输出字符概率预测为公式 3-9 所示，

$$P_i = soft\max(WG_i + b) \tag{3-9}$$

其中，W和b是可学习权值和偏置。

尽管之前提出的并行注意力机制已经比基本的循环注意力机制高效，输出节点之前的依赖关系仍然会因为并行不相关计算，导致上下文关系的丢失。为了捕获上下文的关系，本文提出了第二阶段的解码器包含关系注意力模块和字符解码模块，如图 3-6 所示。第一阶段解码器指的是上半部分，直接将 2D 特征提取模块与关系注意模块和并行注意力模块计算后，得到的输出结果进行解码输出。第二阶段的解码输出指的是下半部分，对编码器和注意力模块加权后的输出，又经过一个关系注意力模块，进一步提取上下文信息，然后再用解码器进行解码。

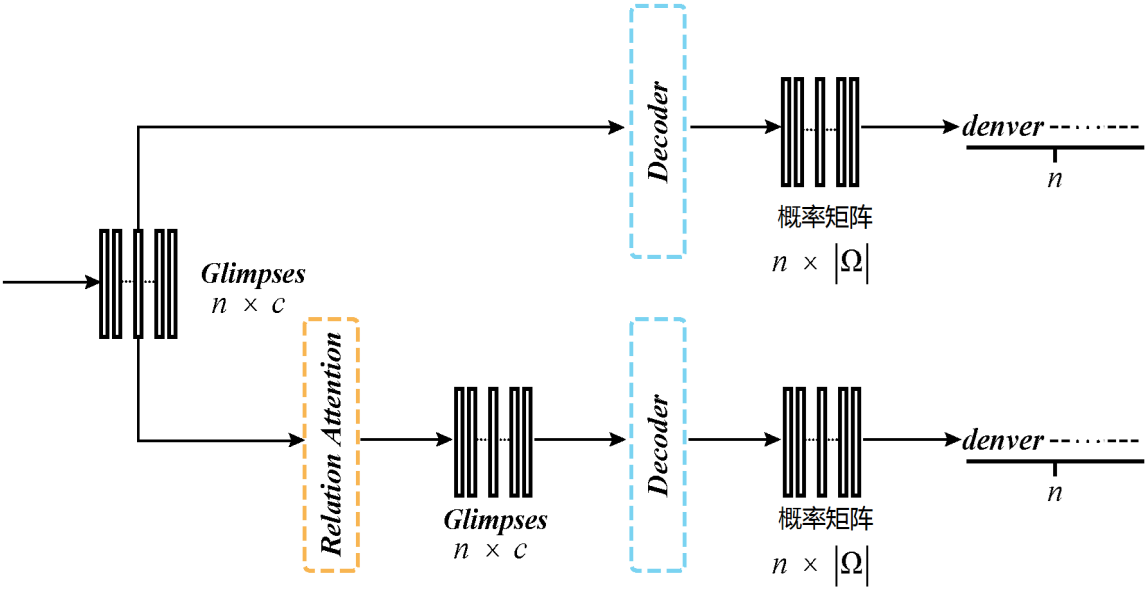

图 3-6 两阶段解码器

3.5 损失函数优化

本文采用端到端的策略优化损失函数，两个解码器用多任务的损失同时优化，

$$L=\sum_{i=1}^{2}\sum_{j=1}^{n}(-\log P_{ij}(y_j)) \tag{3-10}$$

其中，y 是文字序列的标注，i 和 j 分别是解码器和输出节点的索引。在训练阶段，如果标注长度小于 n，每张训练图片的标注将会被符号“EOS”填充。反之，如果长度大于 n，超出部分将会被直接丢弃。具体优化时的参数配置和优化方法，将在第四章进行介绍。

3.6 网络训练策略

本文实现整体算法时，运用了一些现在深度学习领域的训练策略，来提升最终算法的性能和效率。本文主要运用了三个策略，其中一个是对训练数据进行扩增（Data Augmentation），另一个是在训练过程中，采用在线困难样本挖掘（Online Hard Example Mining）的策略，最后一个策略是对损失函数的优化（Optimization），采用 ADADELTA[30]优化算法，对模型参数进行更新。

3.6.1 数据扩增

考虑到本文采用的训练数据全部都是计算机合成数据，故与真实场景会存在一定的偏差。为了尽量减少与真实场景之间数据域的距离，本文采用数据扩增来

弥补合成数据遇到的问题。主要的方法是透视变换、颜色抖动和随机模糊。

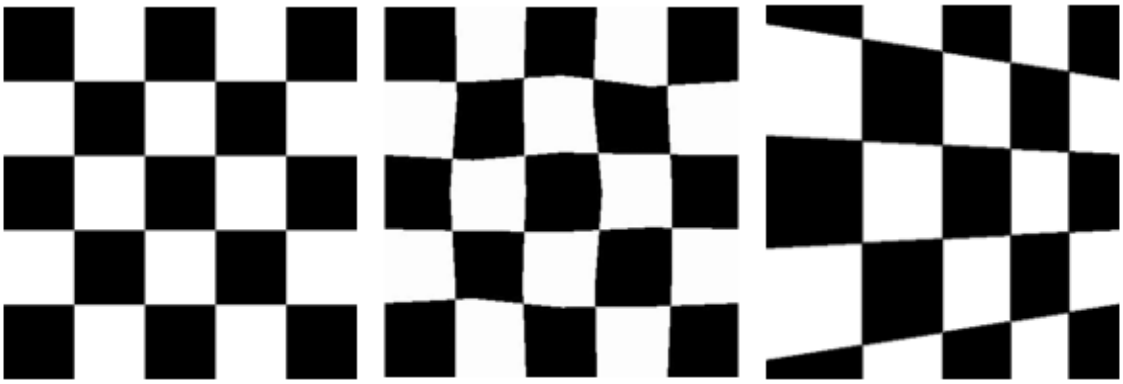

图 3-7 透视变换

如图 3-7 所示，图 3-7-a 是原图，图 3-7-b 和 3-7-c 是通过透视变换后，产生了畸变的图，从后两张图可以看出，通过透视变换后的数据，更贴进真实场景中相机拍摄到的图像效果和拍摄的角度导致的图像倾斜。

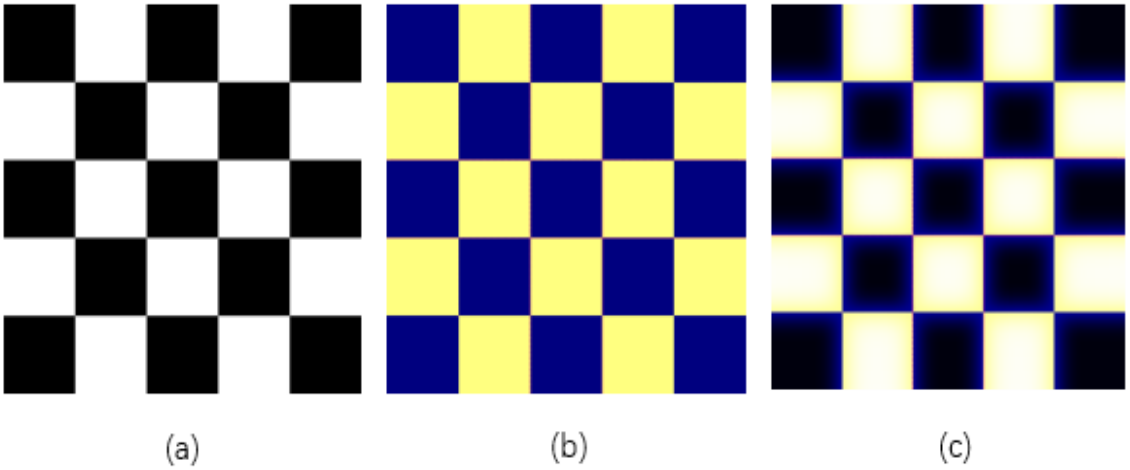

图 3-8 颜色抖动

另外一个数据扩增的方法是颜色抖动，具体做法是引入一个随机值，对输入图像的颜色通道（RGB），亮度和饱和度进行变化调整。如图 3-8 所示，图 3-8-是原图，图 3-8-b 和图 3-8-c 是加入颜色抖动后的图像。

同时，本文还引入了随机模糊。为了让训练数据更贴合真实场景，本文对训练集随机加入模糊噪声，通过随机修改模糊的卷积核大小，来控制最终模糊程度，达到扩增数据集的目的。如图 3-9 所示，图 3-9-a 是原图，图 3-9-b 和图 3-9-c 是不同卷积核大小导致的不同模糊程度。

图 3-9 随机模糊

3.6.2 在线困难样本挖掘

在训练过程中，为了充分利用训练集，本文采用了在线困难样本挖掘的训练策略（Online Hard Example Mining）。由于本文所采用的训练数据都是计算机按照一定规则合成的，这样会不可避免的存在困难样本和简单样本数据分配不均匀，当网络的训练数据大量都为简单样本时，面对困难样本，模型就很难进行精确的识别。为了解决该问题，本文采用对困难样本进行反复训练，增强网络对困难样本的识别能力。具体策略是通过对每个训练批次数据中，对每个样本都统计训练时所对应的损失函数计算出来的数值。根据损失函数公式（3-10）的特点可知，如果是预测错误的负样本，则对应的损失值会比较大。

由此，只需要按照每个批次数据对应产生的损失值，进行从大到小的排序，损失值较大的则是定义为负样本，即容易识别错误的困难样本。通过，对连续多个训练批次进行统计，可以得到一个较大的按损失值排序的列表，此时，只需要将列表中排序在前面的样本，重新选取出来，再送入网络模型中进行在线训练，即可以通过挖掘困难样本，实现对网络参数的在线优化与更新。

具体实现示意图如图 3-10 所示，输入训练的样本，送入网络中提取特征，并得到最终预测结果与实际标签之间的损失。通过对计算出来损失数值的大小进行排序，并筛选出损失值最大的前 k 个样本，作为困难训练样本，再加入训练数据中，对网络进行新的迭代和训练。这样通过上述训练策略，在保证对简单样本鲁棒性的同时，也可以提高最终算法模型对困难样本的识别率，整体提升网络模型的效率。

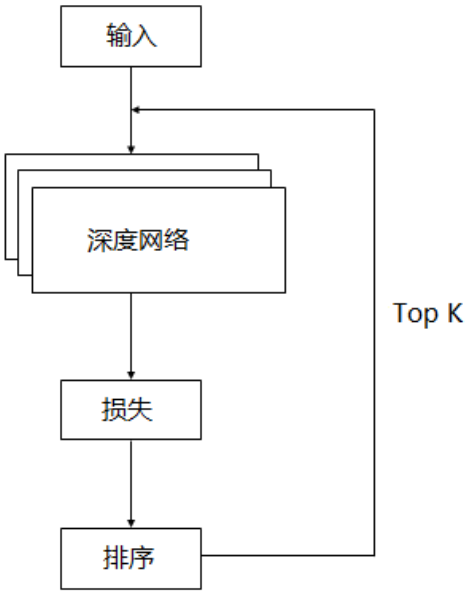

图 3-10 在线困难样本挖掘

3.6.3 模型优化算法

针对本文所提出的损失函数公式（3-10），采用自适应学习率方法（ADADELTA）进行参数模型更新。该方法只用一阶信息，并随着时间动态调整。同时，比之前的随机梯度下降方法，具有更小的计算量。正如该方法的名称，可以动态调整学习率，而不需要手动设置，具有很好的鲁棒性和稳定性，尤其是对超参数、大梯度信息和算法架构等不敏感。

很多机器学习或者深度学习问题，本质上是在优化损失函数，该过程是优化一个参数集合 x，为了让损失函数 $f(x)$ 最小，即公式（3-10）最小。

$$x_{t+1} = x_t + \triangle x_t \tag{3-11}$$

其中，x_t 是 t 时刻的参数，$\triangle x_t$ 是 t 时刻优化算法计算出来的更新量。

$$\triangle x_t = -\eta g_t \tag{3-12}$$

$$g_t = \frac{\partial f(x_t)}{\partial x_t} \tag{3-13}$$

其中，η 是学习率，决定着在负梯度方向更新多大的步长，因为样本的多样性，提前预设一个固定的学习率，显然不是很好的方法，最终训练的参数，极有可能陷入局部最优解。

本文采用上述方法训练网络收敛速度很慢，同时因为本文所采用的训练集有近 1600 万张图片，训练周期长，后续又尝试采用了两个模型优化算法 Adagrad 与 Adadelta 进行参数更新实验。

其中，Adagrad 是具有自适应学习率调整能力的优化算法。标记t时刻对应的第i个参数为$x_{t,i}$，每个参数如公式 3-14 和公式 3-15 进行更新。

$$\triangle x_{t,i} = -\frac{\eta}{\sqrt{\sum_{\tau=1}^{t}(g_{\tau,i})^2 + \varepsilon}} g_{t,i} \tag{3-15}$$

$$x_{t+1,i} = x_{t,i} + \triangle x_{t,i} \tag{3-16}$$

其中，ε是一个避免分母为 0 的极小值。从上述公式可以看出，随着训练的进行，到了训练后期，随着时间积累的分母会越来越大，导致参数无法更新。因为本文训练数据接近 1600 万，训练迭代次数达到 100 万次，故用上述算法在训练后期并不能很好的提升模型效果。

为了解决 Adagrad 在迭代次数比较大的时候，面临的学习率降低，在训练快结束时，参数无法更新的问题，Adadelta 采用了不同的学习率更新策略，如公式 3-17 和公式 3-18 所示，

$$E\left[g^2\right]_t = \gamma E\left[g^2\right]_{t-1} + (1-\gamma) g_t^2 \tag{3-17}$$

$$\triangle x_t = -\frac{\eta}{\sqrt{E\left[g^2\right]_t + \varepsilon}} \odot g_t = -\frac{\eta}{RMS[g]_t} \odot g_t \tag{3-18}$$

其中，RMS指的是梯度的均方差，$\odot$指对应元素位置相乘。为了让更新参数的“单位”匹配，最终定义更新公式为，

$$E[\triangle x^2]_t = \gamma E\left[\triangle x^2\right]_{t-1} + (1-\gamma)\triangle x_t^2 \tag{3-19}$$

$$RMS[\triangle x]_t = \sqrt{E\left[\triangle x^2\right]_t + \varepsilon} \tag{3-20}$$

$$\triangle x_t = -\frac{RMS[\triangle x]_{t-1}}{RMS[g]_t} \odot g_t \tag{3-21}$$

可以看出 Adadelta 优化算法不需要提前预设学习率，也避免了 Adagrad 算法在训练后期参数更新缓慢，以至于不更新的问题，故本文采用 Adadelta 作为优化损失函数（3-10）的方法，该算法较快的优化了本文所提出的算法，收敛速度快。

3.6.4 参数降维

经过 2D 特征提取模块得到的特征，参数量较大，在将该特征输入后续的关系注意力模块和并行注意力模块时，因为上述两个网络采用了大量全连接的结构，最终导致网络模型过大，训练效率降低，也增加了网络过拟合的概率。为了解决训练缓慢，且容易过拟合的问题。本文采用了1×1卷积进行降维。

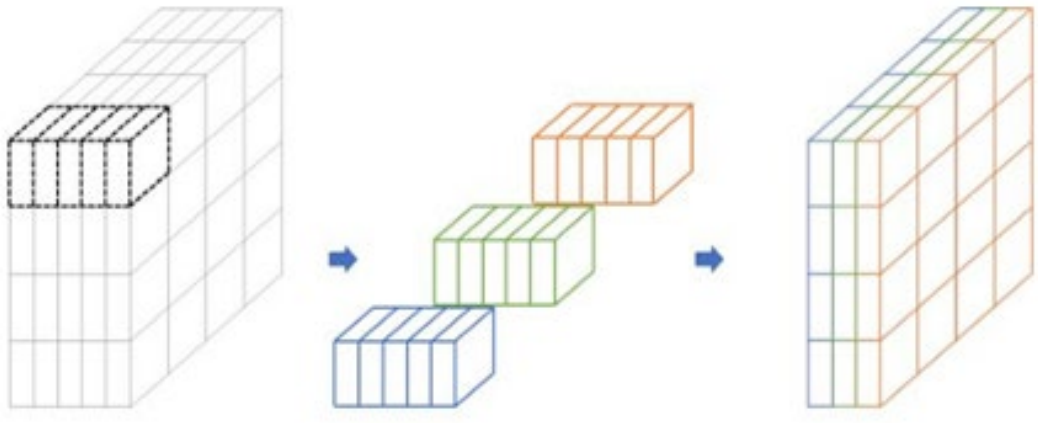

图 3-11 1×1 卷积示意图

如图 3-11 所示，在高维特征图的输入后面，通过1×1卷积，对特征图里的每个元素都通过该卷积核进行加权求和的运算（element-wise），该1×1卷积核输入维度是原特征图的通道维度，输出维度是所需要的降维维度。对 2D 特征提取模块的输出进行降维，可以减小计算量，提高网络模型的效率。同时，因为采用了1×1卷积，可以很好的进行通道之间的信息交换，也具有在保证特征图大小不变的前提下，大幅提升网络的非线性能力，能有效的应用在本文提出的关系注意力模块和并行注意力模块中，为后续计算注意力掩膜，提供了具有丰富语义和空间信息的特征图。

3.7 本章小结

本章针对不规则场景文本识别问题，创新性的提出了通过 2D 关系注意力网络和并行注意力网络，直接并行预测字符序列，既保留了 2D 空间信息，又得到高级语义信息。本文提出的方法不同于之前的算法，既不用附加一个校正网络，也不用注意力循环神经网络，最大限度保留了 2D 空间信息，加快了网络运行速度。

本文提出的 2D 特征提取网络，对之前已有的算法进行改进，移除了运算较为缓慢的串行部分，同时也为了保留空间信息，减少参数量，将之前 1D 特征序列输出改进为 2D 特征图输出。本文提出的关系注意力网络，应用 BERT 和 transformer 的思想，将不能并行的循环神经网络部分移除，改进为双向通信的神经网络结构，既保证了并行计算，也保证了上下文信息不被丢失。本文提出的并行注意力模块，通过对传统注意力模块的改进，可以同时输出每个节点对应的注意力权值，比较于之前的算法，需要用上一时刻的变量计算当前时刻的值，极大限制了网络运行的速度。通过本文的改进，网络预测速度被极大提升。本文提出了两阶段解码器，为了避免本文提出的互相独立的注意力模块，导致不可避免的上下文信息的缺失，在一个字符解码器的基础上，又提出了增加一个关系注意力机制，进一步提取上

下文信息，增强算法鲁棒性。因为是并行网络，在公开数据集测试中效率和准确率，超出了之前提出串行算法。具体实验环节将在第三章予以介绍。

本章最后阐述了网络训练策略，采用了数据扩增、在线困难样本挖掘、模型优化算法和参数降维四个增强网络鲁棒性，提高最终识别准确率的简单可行的训练策略。

第 4 章 实验结果与分析

在上述章节中，本文具体展开了研究不规则场景文本识别课题的意义，为了研究该课题，本文提出了 2D 特征提取网络，关系注意力网络和并行注意力网络，同时为了优化网络参数，本文提出了两阶段解码器。本章主要针对之前提出的算法进行科学实验验证，与其他算法进行算法评估与对比。对本文提出算法的优异性进行合理的解释与分析。

4.1 数据集

本文在数个公开数据集上进行实验，对本文提出算法的有效性和可行性进行了充分的验证。同时，本文提出算法是在两个计算机合成数据集 Synth90K[31]和 SynthText[33]进行训练。同时，本文还在规则场景文字和不规则场景文字数据集上，都进行了算法验证。并且，本文还提出了一个包含单行和多行文本的车牌数据集，目的是为了验证本文提出算法在复杂场景下的鲁棒性和有效性。

4.1.1 训练集

Synth90K[31]是一个合成文本数据集，由 Jadererg[33]等人提供合成数据。该数据集是通过把九万个常见英语单词随机贴合到场景图像块中。该数据集生成了近九百万张图片，里面所有的图片都用来作为预训练的数据。

图 4-1 Synth90K 训练数据示例

Synth90K 是通过下述步骤进行合成的，

(1) 字体渲染：渲染的字体是通过 1400 类从谷歌字体库得到的字体、字距、权值、下划线和其他属性，是从任意定义分布里面随机选取合适的渲染属性。

(2) 随机噪声：每张被渲染的图片中，都会被随机加入高斯噪声，椒盐噪声等。

(3) 随机标签：合成图片用来替换真实数据，同时标注标签的语料库是来自常用单词字典。

SynthText[33]是一个由 Gupta[34]等人提出的合成数据集，用于文字检测和文字识别的预训练任务。里面含有近八百万字符序列图片与 Synth90K 不同的是，该数据集是渲染到一张完整的图片上，并非是图片块，故也可以用于训练字符定位网络。

SynthText 中所有的数据都被裁剪成字符序列，并用于网络模型预训练。

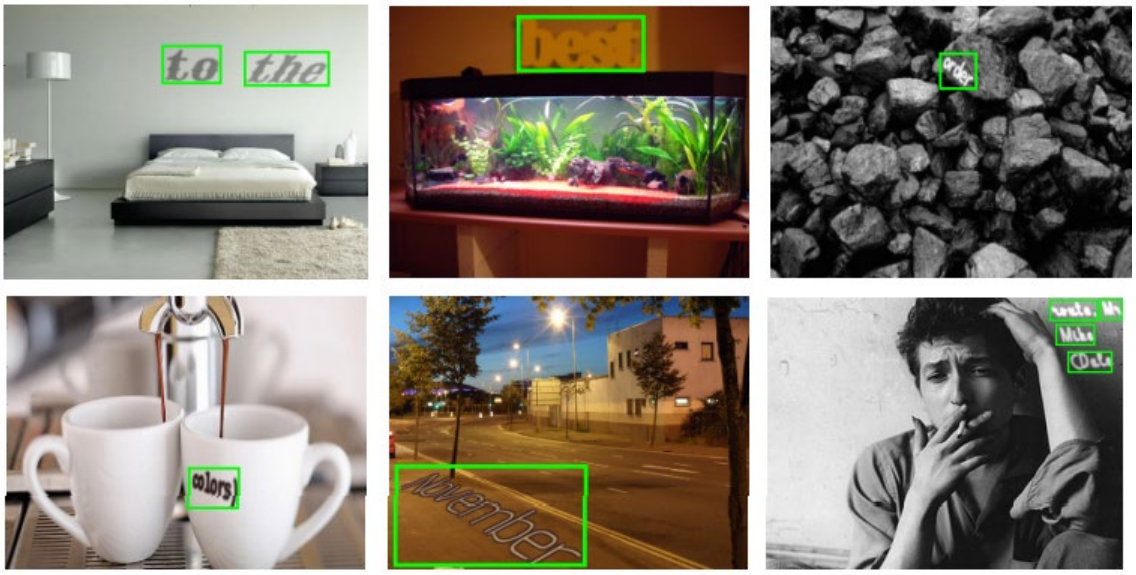

图 4-2 SynthText 训练数据示例

4.1.2 测试集

ICDAR2003[35]是一个从真实场景图片中截取下来的规则场景文本图像数据集。在过滤掉一些包含非数字和字母的特殊字符，还有长度小于三个字符的规则场景文本后，还包含 860 个规则场景文本的测试集，用于算法性能测试。同时，针对每张待测试图片，都提供 50 个单词的查询表。

图 4-3 ICDAR2003 测试样例示意图

ICDAR2013[36]数据集大部分图片从 ICDAR2003 中继承而来。同时，又加入了新的数据，与之前的过滤方法类似，本文过滤掉一些包含着非字母或者数字的图像，最后留下 1015 张符合条件的测试数据。在这个测试数据中，没有字典提供。

图 4-4 ICDAR2013 测试样例示意图

IIIT5K-Words[37]包含 5000 张文本图像，其中 3000 张用于测试。绝大多数测试图像是规则的。3000 张测试集中的每张图像，都提供了 50 个单词和 1000 个单词的单词表。

图 4-5 IIIT5K 测试样例示意图

Street View Text[7]数据集包含从谷歌街景中截取的 647 张图像。在这个数据集中，每张图像有 50 个单词作为字典。

图 4-6 SVT 测试样例示意图

SVT-Perspective[38]也来自谷歌街景图片，因为该数据集很多图像都是从侧边拍摄的，所以会存在透视畸变的现象。很多截取下来的场景文本图片都是不规则的。于是，该数据集被更多的用来测试对不规则场景文本的识别能力。

图 4-7 SVTP 测试样例示意图

ICDAR 2015 Incidental Text[36]数据集是来自谷歌眼镜拍摄的自然场景画面。里面有 2077 张测试集，并且大多数的测试图像都是多角度的。所以，该测试集可以用来测试算法处理不规则场景文本的性能。

图 4-8 ICDAR2015 测试样例示意图

CUTE80[40]是一个测试弯曲文本检测的数据集，截取里面的图像，可用作检测模型对弯曲文本的识别能力。一共有 288 张图像截取下来作为测试集。同时，没有字典提供。

图 4-9 CUTE80 测试样例示意图

4.1.3 多行不规则场景文本

Multi-Line Text 280（MLT280）是本文提出的含有单行和多行文本识别的数据集。本数据集是收集和筛选自互联网，共计有 280 张车牌图像

图 4-10 Multi-Line Text 280 测试样例示意图

本文为了验证所提出算法在多行文本上的表现，通过计算机合成了一百万车辆牌照数据，其中包括白底和黄底的车牌，也包含单行和双行车牌，目的是尽量与真实场景中的数据保持一致。所以，合成车牌时，参考车牌号码规则进行合成。按照车牌管理机构公布的合成规则，为了避免和数字 1,0 混淆，去掉了字母 I、O 和 Q。具体合成数据如图 3-11 所示，

图 4-11 SynthLicense 测试样例示意图

4.1.4 数据分析

本文提出算法一共涉及 11 个数据集，如图 4-12 所示。其中，Synth90K 和 SynthText 是作为训练数据，对网络的模型参数进行调整与优化。ICDAR2003、ICDAR2013、IIIT5K 和 SVT，上述四个数据集是用于测试算法识别规则场景文本的能力。ICDAR2015、SVTP 和 CUTE80 这三个数据集，是用于测试算法对不规则场景文本的鲁棒性。Multi-Line Text 280 数据集，是作为扩展实验部分，验证本文所提出算法具备识别多行文本的能力。SynthLicense 数据是针对多行文本识别的扩展实验，特意合成的数据集，用于对网络进行训练与模型参数优化。

图 4-12 中，第一分支是规则场景文本图像和不规则场景文本图像，从图中可以看出不规则场景文本会出现弯曲、倾斜和模糊等，识别起来难度很大。第二分支是本文所提出的多行文本数据集，里面包含了单行和多行车辆牌照。第三分支是本文为了识别多行不规则场景文本，按照待测数据集的数据分布特征采用计算机视觉算法库合成的文本图像数据。

图 4-12 数据集示意图

4.2 实现细节

4.2.1 网络参数设置

本文提出 2D 特征提取网络，本质上是卷积神经网络。特别地，本文只保留了最开始的两个2×2步长卷积，用来对特征图进行降采样，同时把最后三个2×2步长卷积，替换成1×1步长卷积。

为了减少第一个关系注意力模块的计算量，本文也提出对输入的卷积特征进行降维，具体方法是把1×1的卷积神经网络层对特征图进行卷积，维数可以降维至128维。同时，本文默认设置 transformer 层的层数为 2。同时，对关系注意力模块中的每个 transformer 单元，自注意力模块隐藏层的大小设置为 512，个数设置为 4。

本文也提出了并行注意力网络，针对并行注意力网络，输出的节点 n 设置为25，因为很多常见单词的长度是比 25 要短的。同时，本文设置解码器所解码的字符类别 $|\Omega|$ 为 38，具体就是 0 到 9 的阿拉伯数字，不区分大小写的 A 到 Z 共计 26 个英文字母，还有一个代表着序列结束的标识符“EOS”，最后还有一个代表其他不在上述字符中的“UNK”字符。

4.2.2 网络训练

为了与之前的方法进行公平的比较，本文提出的算法也按照之前的提出的算法，基于 Synth90K 与 SynthText 两个计算机合成数据集进行训练，并且是网络从随机初始化开始进行训练。两个数据集的采样比例设置为1:1。在训练阶段，输入图像的大小统一调整为32×100的尺度。同时，本文也采用了数据扩增，例如颜色偏移，模糊等方法。同时，本文提出采用 ADADELTA 优化算法，对模型进行优化，训练时的批次数量设置为 128。按照经验配置，本文提出将初始的学习率设置为 1，经过 60 万次训练迭代后，学习率设置为 0.1。经过 80 万次训练迭代后，学习率设置为 0.01。同时，训练将在 100 万次时停止。

4.2.3 网络推断

在网络推断阶段，本文依然把图像尺度设置为32×100，然后再通过一下准则将概率矩阵解码成对应的字符序列。

(1) 对于输出节点，每个节点对应概率最大的字符将视为输出

(2) 所有“UNK”标识符将会被删除

(3) 解码过程将在遇到“EOS”标识符时终止

本文默认设置从第二阶段的解码器获取字符预测结果，与之前算法的评测标准一致，本文将用合成数据训练的模型，在所有真实场景数据集中进行测试。

4.2.4 网络实现

本文提出的算法基于 PyTorch 框架实现，同时，所有的实验都是在常规的工作站上完成的。默认设置情况下，本文训练用两块英伟达 P40 显卡进行训练，并在一块显卡上逐图像进行测试，即测试的批量数据大小设置为 1。

4.3 算法性能比较

为了验证本文所提出算法的有效性，本文将在上述提及的公开数据集上，与当前学术界提出的最优算法的性能进行，与之前所提出算法评测标准一致，计算识别准确率是通过比较预测结果与真实标注的一致性，作为计算准则。如果完全一致，才算为预测正确，否则就预测错误。同时，当提供字典时，预测的结果可以根据字典进行校正，校正的依据是预测结果与字典之间的最小编辑距离。

$$LER(h,S^{'}) = \frac{1}{Z}\sum_{(x,z)\in S^{'}} ED(h(x)) \tag{3-1}$$

其中，LER 指的是字符错误率，表征的是预测字符序列 h 与目标字符序列 $S^{'}$ 归

一化后的编辑距离。Z 指的是目标字符序列长度，$ED(p,q)$ 指的是两个不同序列 p 和 q 的编辑距离，即从序列 p 变化成序列 q，需要的删除、插入、替换等操作的最小操作次数。

4.3.1 规则场景文本识别

本文首先四个规则场景文本数据集上进行比较。这四个数据集分别是：IIIT5k、SVT、IC03 和 IC13。预测结果准确率和指标对比都如表 4-1 所示，结果显示本文提出的算法在大部分数据集上，都取了当前最好的结果，超过之前学术界提出的准确率最高的算法。其中，表 4-1 中的“50”和“1000”指的是字典的大小，“FULL”指的是测试集中出现的所有单词都作为字典，“0”表示没有字典提供。被“*”标记的算法是指训练集为 Synth90K 和 SynthText，与本文所提出算法的训练集保持一致，对比公平。针对 IIIT5K 与 SVT，因为有的文本实例是弯曲或者有方向的，文本所提出算法的结果，在这两个数据集上比 Aster 略高一些。同时，以很大的优势超过其他算法。在 IC03 数据集上，本文提出的算法结果在没有字典的前提下，是与 MORAN[23]相比，是有竞争力的。在有字典的前提下，是超出其他所有算法的。本文提出的算法在 IC13 数据集上，比 Bai[44]和 Cheng[25]提出的算法表现略低，然而上述两个算法只能解决规则场景文本识别问题。在规则场景文本识别上的表现，证明了本文所提出算法的泛化性和鲁棒性。

表 4-1 对现有学术界算法进行了整理和统计，在不同测试数据集上的结果都被列入对比的范畴。所以该表格是科学全面有效的，同时，本文所提出的方法在该表格中所列举的方法中，算法表现最优异，因此可以认定目前本文所提出的算法与之前提出的算法进行比较，是最优算法。

4.3.2 不规则场景文本识别

为了验证本文所提出算法，针对不规则场景文本数据集的识别能力。本文在三个不规则场景文本数据集 IC15、SVTP、CUTE 上进行实验。实验结果同样被记录在表 4-1 中，本文提出的方法，以较大的优势超越之前学术界所提出最好的算法[22][26]别达到 0.2%，3.8%和 7.3%。特别地，本文提出的算法在 CUTE 数据集上，超出基于校正的方法至少是 7.3%。超过基于 2D 透视变换的方法高达 10%。本文所提出的算法相比较之前的算法，有了非常明显的提升。这证明了该算法比其他算法更具鲁棒性和有效性。在识别不规则场景文本时，表现更为明显。特别地，当识别具有复杂布局的不规则场景文本时（比如 CUTE80），优势更为明显。

表 4-1 算法准确率对比

算法	IIIT5K			SVT		IC03			IC13	IC15	SVTP	CUTE
	50	1000	0	50	0	50	Full	0	0	0	0	0
Wang *et al.* [7]	-	-	-	57.0	-	76.0	62.0	-	-	-	-	-
Mishra *et al.* [33]	64.1	57.5	-	73.2	-	81.8	67.8	-	-	-	-	-
Wang *et al.* [8]	-	-	-	70.0	-	90.0	84.0	-	-	-	-	-
Bissacco *et al.* [46]	-	-	-	-	-	90.4	78.0	-	87.6	-	-	-
Almaza´n *et al.* [43]	91.2	82.1	-	89.2	-	-	-	-	-	-	-	-
Yao *et al.* [16]	80.2	69.3	-	75.9	-	88.5	80.3	-	-	-	-	-
Rodr´ıguez-Serrano *et al.* [50]	76.1	57.4	-	70.0	-	-	-	-	-	-	-	-
Jaderberg *et al.* [11]	-	-	-	86.1	-	96.2	91.5	-	-	-	-	-
Su and Lu [51]	-	-	-	83.0	-	92.0	82.0	-	-	-	-	-
Gordo [47]	93.3	86.6	-	91.8	-	-	-	-	-	-	-	-
Jaderberg *et al.* [10]	95.5	89.6	-	93.2	71.7	97.8	97.0	89.6	81.8	-	-	-
Jaderberg *et al.* [11]	97.1	92.7	-	95.4	80.7	98.7	98.6	93.1	90.8	-	-	-
Shi *et al.* [14]	97.8	95.0	81.2	97.5	82.7	98.7	98.0	91.9	89.6	-	-	-
Shi *et al.* [15]	96.2	93.8	81.9	95.5	81.9	98.3	96.2	90.1	88.6	-	71.8	59.2
Lee *et al.* [13]	96.8	94.4	78.4	96.3	80.7	97.9	97.0	88.7	90.0	-	-	-
Liu *et al.* [24]	-	-	83.6	-	84.4	-	-	91.5	90.8	-	73.5	-
Yang *et al.* [29]	97.8	96.1	-	95.2	-	97.7	-	-	-	-	75.8	69.3
Liao *et al.* [30]	99.8	98.8	91.9	98.8	86.4	-	-	-	91.5	-	-	79.9
Li *et al.* [27] *	99.4	98.2	91.5	98.5	84.5	-	-	-	91.0	69.2	76.4	83.3
Cheng *et al.* [52]*	99.3	97.5	87.4	97.1	85.9	99.2	97.3	94.2	93.3	70.6	-	-
Cheng *et al.* [25]*	99.6	98.1	87.0	96.0	82.8	98.5	97.1	91.5	-	68.2	73.0	76.8
Bai *et al.* [44]*	99.5	97.9	88.3	96.6	87.5	98.7	97.9	94.6	**94.4**	73.9	-	-
Shi *et al.* [22]*	99.6	98.8	93.4	**97.4**	89.5	98.8	98.0	94.5	91.8	76.1	78.5	79.5
Luo *et al.* [23]*	97.9	96.2	91.2	96.6	88.3	98.7	97.8	**95.0**	92.4	68.8	76.1	77.4
Ours	**99.8**	**99.1**	**94.0**	97.2	**90.1**	**99.4**	**98.1**	94.3	92.7	**76.3**	**82.3**	**86.8**

4.3.3 多行场景文本识别

之前不规则场景数据集仅包括角度倾斜、透视变换和弯曲文本，但是缺乏在多行场景文本上的实验和比较。多行场景文本一般会出现在车辆牌照，数学公式和验证码等常见自然场景。为了验证本文所提出的算法识别多行场景文本的能力，本文提出一个多行场景文本数据集 MLT280，里面包含单行和多行车辆牌照图片，共计 280 张图像。同时，本文对应制作了一个计算机合成数据集，包含大约一百

万张合成车辆牌照图像，作为模型训练的数据集。

本文针对多行场景文本识别问题，一共训练了两个模型。一个模型是从随机初始化开始训练，训练数据只有计算机合成数据，标记为“random init”。另一个模型是基于 Synth90K 和 SynthText 两个数据集预训练的基础上，然后用计算机合成数据对模型进行优化调整，标记为“fine-tuned”。针对随机初始化的模型，本文采用同识别不规则场景文本时，一样的训练策略，即采用 ADADELTA 优化算法，初始学习率设置为 1， 在训练迭代次数为 60 万次时，学习率设置为 0.1，训练迭代次数 100 万次数，学习率设置为 0.01。针对参数调优的模型，本文在原训练模型的基础上，又训练迭代 150 万次。为了与之前不同算法流程的网络模型进行对比，本文在训练数据一致的前提下，又训练了基于先校正图片，再识别文本的方法 Aster 与基于循环注意力流程的方法 SAR。本文针对两个算法采用对应的官法代码。类似地，两个算法都针对随机初始化和参数优化两个模型，都分别进行训练，与本文提出的算法进行比较。比较结果如表 3-2 所示，其中，“Original”表示在原论文对应测试平台上统计的速度。“*”指的是在 NVIDIA TianX GPU 中测试结果，同时，测试批次是不明确的。

由表 4-2 可知，Aster 在随机初始化模型与优化调整参数模型上的准确率分别是 40%和 62.5%。特别地，本文观察到基于校正的算法并不能处理多行文本，原因是校正图像网络所采用的薄板样条插值函数，只对倾斜或者弯曲文本有效果，多行场景文本几乎都识别错误了。这也暗示着基于先校正图像再识别的算法流程，并不能解决多行文本识别问题。

对于 SAR 算法，在随机初始化模型和优化调整参数模型上的准确率分别是 43.9%和 51.1%。该实验结果说明基于循环注意力网络的串行算法，是不能很好解决多行文本识别问题的，后续实验结论分析中，将会具体阐述这类算法的局限性。

本文提出的算法均在两组实验中，均取得了最优的结果，在随机初始化模型和优化调优模型上准确率分别为 61.4%和 80.7%。实验结果表明，本文提出的算法大幅度的超过了 ASTER 和 SAR，这表明该算法具备识别多行场景文本的能力。

表 4-2 不同算法在 MTL280 数据集上对应准确率与速度

算法	准确率		速度	
	随机初始化	参数优化	本文平台	原始平台
ASTER [22]	40.0	62.5	32.4ms	20ms
SAR [27]	43.9	51.1	66.7ms	15ms*
Ours	**61.4**	**80.7**	**15.1ms**	-

4.3.4 算法运行速度

为了验证本文所提出方法的效率，本文对算法速度进行了测试，同时也与其他识别不规则场景文本的算法进行了对比。为了公平的对比，本文所有的实验都是在相同的硬件平台上进行的，并且测试时的批次大小都设置为 1。每个都算法都在 MTL280 数据集上进行了 5 次测试，取平均测试时间作为最终算法速度的测试结果。如表 4-2 所示，受益于本文所提出的关系注意力模块和并行注意力模块，本文所提出的算法比基于先校正后识别的算法快 2.1 倍，比基于循环 2D 注意力的方法快 4.4 倍。

4.4 实验分析

本文对上述对比实验都进了可视化分析，因为现在的深度学习模型近似一个“黑箱”，很难看到内部运算和输出结果的过程，为了方便对结果进行科学合理的分析，针对基于先校正后识别的方法，本文可视化出校正后的图像。针对基于循环 2D 注意力的方法，本文可视化生成的 2D 注意力掩膜。同时，本文所提出的算法也进行注意力掩膜可视化，便于与上述算法进行比较，分析具有优势的原因。

4.4.1 可视化分析

如图 4-13 所示，本文提出的算法能识别不规则场景文本，主要原因是 2D 注意力模块成功学习到了字符文本的空间位置信息，针对倾斜，弯曲和透视变换的图像，都能很好的感知对应的文本位置，找到了文本的位置，后续的识别问题就相对简化了。2D 注意力掩膜的作用是在预测文本时，对应“高亮”区域的掩膜会提供 2D 特征提取网络输出的特征图较大的权值。以此方法，在预测网络结果时，对应位置的文本区域会表现出较高的“响应”，避免了其他位置字符或者背景噪声的干扰。比如，图 4-13 中的第一列和第二列，可以看出针对弯曲文本，本文所提出的算法可以很好的将对待识别的区域进行注意力加权，从提高该区域的识别进度。同时，从图 4-13 中的第八列可以看出，针对倾斜文本，本文所提出算法也可以很好的进行识别，从可视化后的注意力掩膜可以看出，倾斜文本也可以很好的进行注意力加权。上述可视化结果证明了本文所提出算法针对弯曲和倾斜文本具有很好的鲁棒性，也证明了该方法的可解释性。

图 4-13 本文提出算法的并行注意力模块输出的 2D 注意力掩膜

4.4.2 可视化对比

如图 4-14 所示的注意力掩膜和校正后的图像，针对单行不规则场景文本，SAR 可以相对较好识别到文本区域，并有正确的预测。但是，针对双行文本识别时，SAR 就不能很好的找到文本的位置，可以明显看到，SAR 在双行文本预测的掩膜，几乎都是错误的。分析其错误的原因，主要是因为在提取特征图时，尽管 SAR 是保留了 2D 特征图，但是后续为了将特征图传入双向长短期记忆网络中，又沿着竖直方向对特征图进行压缩，最终转换为 1D 特征向量。即从 2D 特征图转换为 1D 特征向量这一步骤，空间信息不可避免的损失了。随着空间信息的损失，在预测多行文本的掩膜时，就会产生错误，强行将分布在 2D 空间的不规则文本，空间信息都压缩到 1D 维度。

针对 Aster 的可视化结果，可以明显看出，识别倾斜的单行文本时，校正网络确实可以对不规则场景文本进行校正，然后识别校正后的图片就是相对简单的任务。但是，本文也观察到在识别多行场景文本图像的时候，校正网络就无法发挥作用。显而易见的，薄板样条函数插值只能针对倾斜，弯曲的不规则文本图像，将其进行校正，而多行文本场景图像，即使进行了校正，还是多行文本，而校正后的识别算法，只局限于识别规则场景文本。深入分析原因是，后续的识别网络会在特征提取时，就已经将 2D 特征图直接降维成 1D 特征向量，这样损失了 2D

空间信息，后续也就无法对多行文本进行识别，所以在多行文本识别的准确率远低于本文所提出的算法。

从图 4-14 的第三部分，可以明显看出本文所提出的算法，能够非常好的针对不规则场景文字区域位置进行预测，针对多行文本图像，也能很好的进行识别，相对于上述两个算法具有非常大的优势。而具有优势的原因是本文所提出的关系注意力模块和并行注意力模块，保留了 2D 空间上下文信息，也就可以对多行文本进行识别，同时针对倾斜和弯曲等不规则场景文本识别任务，该算法也具备很好的鲁棒性。

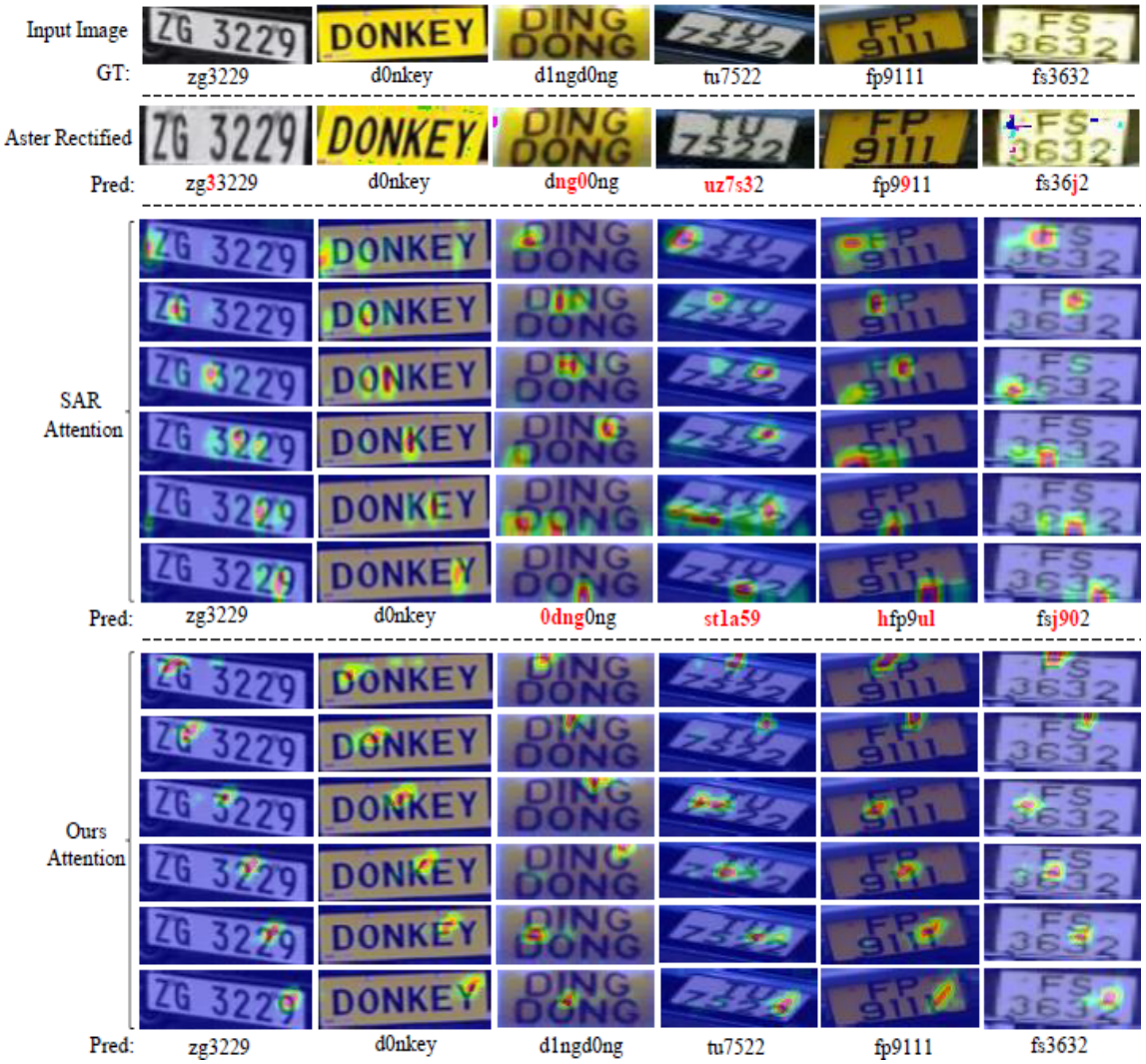

图 4-14 三个算法在 MTL280 数据集上可视化结果对比

4.5 网络结构分析

本文提出了关系注意力网络和两阶段的字符解码器结构，为了验证上述两个结构是否是性能本文所提出算法性能提升的原因，本文又进行了多组实验，通过实验结果来验证本文所提出算法网络结构的合理性。

4.5.1 两阶段解码器有效性分析

为了验证文本所提出的两阶段解码器的有效性，本文对第一阶段解码器输出结果和第二阶段解码器输出结果进行了量化对比和分析，结果如表 4-3 所示。为了表述简洁，本文将第一阶段解码器结果标记为 Ours(d1)，第二阶段解码器结果标记为 Ours(d2)。可以看出，第二阶段解码器的结果在所有数据集上，都是比第一阶段解码器输出的结果表现要好。这也说明了经过第一个关系注意力模块后输出结果的上下文关系，可以被第二阶段中的关系注意力模块很好的获取，从而提升了最终的预测准确率。

表 4-3 两阶段解码器在不同数据集上的准确率

算法	IIIT5K	SVT	IC03	IC13	IC15	SVTP	CUTE
Ours(d1)	93.3	89.3	94	92.5	75.4	81.9	86.5
Ours(d2)	**94.0**	**90.1**	**94.3**	**92.7**	**76.3**	**82.3**	**86.8**

4.5.2 关系注意力模块有效性分析

为了验证本文所提出的关系注意力模块对最终结果的提升，产生了影响。本文设计了一组实验来验证关系注意力模块有效性，具体方案是对关系注意力模块的 transformer 层的数量进行调整。为了表述上的简洁，本文将设计的算法结构标记为 Ours(a,b)，其中 a 和 b 分别指的是第一阶段和第二阶段中的关系注意力模块所对应的 transformer 层的层数。Ours(0,0)代表第一和第二阶段关系注意力模块中的 transformer 层数都为 0，即此时模型没有关系注意力模块。实验结果如表 3-4 所示。从表 4-4 中，可以明显看出 Ours(2,0)的结果始终高于 Ours(0,0)，说明第一阶段的关系注意力模块，是可以获取 2D 特征提取网络输出的 2D 特征图中的上下文语义信息。同时，Ours(2,2)的结果优于 Ours(2,0)的结果，这也说明了第二阶段关系注意力模型在对第一阶段的输出结果基础上，再次获取了上下文信息，增强了算法的最终预测表现。同时，本文也进行了其他对比实验，增加了两个阶段对应 transformer 层的层数，最终发现结果与 Ours(2,2)类似，并没有明显的提升。为了实现的简单与运算效率的平衡，本文最终采用（2,2）作为第一阶段和第二阶段中关系注意力模块 transformer 层对应层数的默认设置参数。

表 4-4 不同关系注意力模块层数的预测结果

算法	IIIT5K	SVT	IC03	IC13	IC15	SVTP	CUTE
Ours(0,0)	92.1	88.1	94.1	91.5	73.4	80	85.4
Ours(2,0)	93.5	90.3	**94.3**	92.2	74	80.9	85.1
Ours(2,2)	**94.0**	**90.1**	**94.3**	**92.7**	**76.3**	**82.3**	**86.8**

4.6 局限性分析

本文为了全面分析所有的测试图像，对错误预测的图像也进行了分析。如图 4-15 所示，本文提出的算法在一些特殊场景下，面临着漏识别和误识别的问题，尤其是当文本是竖直的情况下，整体图像被调整大小至32×100，这样竖直文本会被严重压缩失真，导致难以准确预测。同时，也因为训练数据集中，缺乏竖直的文本图像，更进一步加大了识别的难度。另外，当图片具有复杂背景干扰或者是特殊艺术字等情况“G”、“C”、“D”、“0”等字母和数字会难以区分。本文所提出算法面临的问题，当前学术界提出的表现优秀的算法[22][27][55]等也存在相同的局限性，亟待解决。

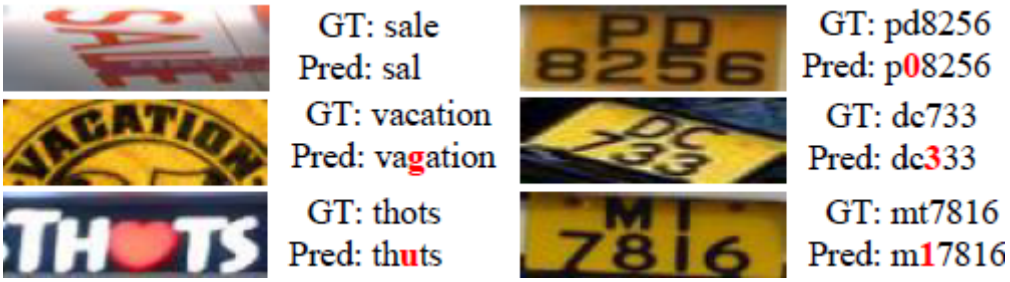

图 4-15 错误样例，预测错误的字符用红色字母标记

4.7 本章小结

本章首先对采用的数据集进行介绍与特点分析，然后对实验中的参数配置与具体实现方法进行了详细的描述，针对规则场景文本识别和不规则场景文本识别进行了与其他学术界所提出的算法进行比较，并得出本文所提出的算法超过了之前所提出算法的性能。同时，针对多行场景文本的情况，本文进行了进一步分析和实验，对算法的效率进行了对比，对实验结果进行可视化分析，得到本文所提出算法具有优越性的原因。接下来，对本文所提出的网络结构，进行了有效性分析，得出提出上述结构的必要性。最后本章针对本文所提出算法的局限性进行总结和分析。

结　论

之前学术界提出的不规则场景文本识别方法，一般是通过卷积神经网络提取1D 特征序列，输入到循环注意力网络中，进行串行预测解码。本文创造性的提出应用卷积神经网络提取 2D 特征图，输入到关系注意力模块和并行注意力模块中，进行并行预测解码。本文主要贡献如下：

（1）本文针对之前方法中普遍存在的 2D 图像特征，转为 1D 特征序列时，不可避免的空间信息丢失的问题，本文采用将卷积神经网络提取的 2D 空间特征直接输入到关系注意力模块中，而不是采用传统的最大值池化降维方法，从而保留住了 2D 图像中的空间信息。同时，通过本文所提出的关系注意力模块，可以更好的提取 2D 特征图中的高级语义信息，作为后续并行注意力模块和解码器部分的输入。

（2）本文提出的关系注意力模块和并行注意力模块，最大的优势是既能很好的获取 2D 特征图的高级语义信息，提高了不规则场景文本预测的准确率，又能避开了循环神经网络串行执行的缺点，可以并行同时预测所有的不规则场景文本，速度上有了很大提升。本文在 7 个公开数据集与首次提出的多行文本数据集上，都取得了超过之前所提出方法的实验结果，这也证明了本文所提出算法的有效性与优异性。

本文提出的识别不规则场景文本算法，在具有极大优势的同时，也面临着当前学术界遇到的问题，比如竖直文本识别和背景复杂文本识别。在识别极其不规则场景文本时，后续研究方向是通过卷积神经网络预测出字符文本位置，对单个字符进行分割，同时对每个字符进行位置编码。采用这样的算法思路，即使是竖直或者是复杂布局的文本，也可以很好的进行预测。不过因为引入了字符分割的思想，运算量会大量增加，网络推测的速度会不可避免的降低，找到一种快速分割字符的方法，也是以后研究的重点方向。

参考文献

[1] Busta M , Neumann L , Matas J . Deep TextSpotter: An End-to-End Trainable Scene Text Localization and Recognition Framework[C]// 2017 IEEE International Conference on Computer Vision (ICCV). IEEE, 2017:2223-2231.

[2] He T , Tian Z , Huang W , et al. An end-to-end TextSpotter with Explicit Alignment and Attention[J]. 2018:5020-5029.

[3] Jaderberg M , Simonyan K , Vedaldi A , et al. Reading Text in the Wild with Convolutional Neural Networks[J]. International Journal of Computer Vision, 2016, 116(1):1-20.

[4] Liu X , Liang D , Yan S , et al. FOTS: Fast Oriented Text Spotting with a Unified Network[J]. 2018:5676-5685.

[5] Lyu P , Liao M , Yao C , et al. Mask TextSpotter: An End-to-End Trainable Neural Network for Spotting Text with Arbitrary Shapes[J]. 2018:71-88.

[6] Neumann L , Matas J . Real-time Scene Text Localization and Recognition[C]// Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012:3538-3545.

[7] Wang K , Babenko B , Belongie S . End-to-end Scene Text Recognition[C]// 2011 International Conference on Computer Vision. IEEE, 2012:1457-1464.

[8] Wang T, Wu D, et al. End-to-end Text Recognition with Convolutional Neural Networks. [C]//In Proceedings of the 21st International Conference on Pattern Recognition, 2012:3304-3308.

[9] Yao C, Bai X, Liu W. A Unified Framework for Multioriented Text Detection and Recognition[J]. Image Processing IEEE Transactions on, 2014, 23(11):4737-4749.

[10] Jaderberg M, Simonyan K, Vedaldi A, et al. Synthetic Data and Artificial Neural Networks for Natural Scene Text Recognition[J]. Eprint Arxiv, 2014.

[11] Jaderberg M , Vedaldi A , Zisserman A . Deep Features for Text Spotting[C]// European Conference on Computer Vision. Springer, Cham, 2014:512-528.

[12] Lee C Y , Bhardwaj A , Di W , et al. Region-Based Discriminative Feature Pooling for Scene Text Recognition[C]// IEEE Conference on Computer Vision & Pattern Recognition. IEEE Computer Society, 2014:4050-4057.

[13] Lee C Y , Osindero S . Recursive Recurrent Nets with Attention Modeling for OCR in the Wild[J]. 2016:2231-2239.

[14] Shi B , Bai X , Yao C . An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Application to Scene Text Recognition[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2015, 39(11):2298-2304.

[15] Shi B , Wang X , Lyu P , et al. Robust Scene Text Recognition with Automatic Rectification[J]. 2016:4168-4176.

[16] Bai X , Yao C , Liu W . Strokelets: A Learned Multi-Scale Representation for Scene Text Recognition[C]// 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2014:4042-4049.

[17] Lecun Y , Bottou L , Bengio Y , et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.

[18] Fan R , Zhou P , Chen W , et al. An Online Attention-based Model for Speech Recognition[J]. 2018:577-585.

[19] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735-1780.

[20] Graves A , Santiago Fernández, Gomez F . Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks[C]// International Conference on Machine Learning. ACM, 2006:369-376.

[21] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate[J].2014.

[22] Shi B, Yang M, Wang X, et al. ASTER: An Attentional Scene Text Recognizer with Flexible Rectification[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2018, PP(99):1-1.

[23] Luo C , Jin L , Sun Z . A Multi-Object Rectified Attention Network for Scene Text Recognition[J]. 2019:109-118.

[24] Liu W, Chen C, et al. Char-net: A characteraware neural network for distorted scene text recognition[C]// In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18),2018:7154-7161.

[25] Cheng Z , Xu Y , Bai F , et al. AON: Towards Arbitrarily-Oriented Text Recognition[J]. 2017:5571-5579.

[26] Jaderberg M, Simonyan K, Zisserman A, et al. Spatial Transformer Networks[J]. 2015:2017-2025.

[27] Li H , Wang P , Shen C , et al. Show, Attend and Read: A Simple and Strong Baseline for Irregular Text Recognition[C]//In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, (AAAI-19), 2019.

[28] Vaswani A , Shazeer N , Parmar N , et al. Attention Is All You Need[J]. 2017: 6000-6010.

[29] Yang X, He D, Zhou Z, Kifer D, and Giles C. L. Learning to read irregular text with attention mechanisms [C]//In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence(IJCAI 2017), 2017: 3280-3286.

[30] Liao M , Zhang J , Wan Z , et al. Scene Text Recognition from Two-Dimensional Perspective[J]. 2018.

[31] Zeiler M D. ADADELTA: An adaptive learning rate method[J]. Computer Science, 2012.

[32] Synth90k. http://www.robots.ox.ac.uk/˜vgg/data/text/.

[33] Synthtext. http://www.robots.ox.ac.uk/˜vgg/data/scenetext/.

[34] Gupta A , Vedaldi A , Zisserman A . [IEEE 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) - Las Vegas, NV, USA (2016.6.27-2016.6.30)] 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) - Synthetic Data for Text Localisation in Natural Images[J]. 2016:2315-2324.

[35] Lucas S M , Panaretos A , Sosa L , et al. ICDAR 2003 Robust Reading Competitions: entries, results, and future directions[J]. International Journal on Document Analysis and Recognition, 2005, 7(2-3):105-122.

[36] Karatzas D , Shafait F , Uchida S , et al. ICDAR 2013 Robust Reading Competition[C]// 2013 12th International Conference on Document Analysis and Recognition. IEEE Computer Society, 2013:1484-1493.

[37] Mishra A, Alahari K and Jawahar C. V. Scene Text Recognition Using Higher Order Language Priors[C]// In British Machine Vision Conference(BMVC2012), 2012:1-11.

[38] Phan T Q , Shivakumara P , Tian S , et al. Recognizing Text with Perspective Distortion in Natural Scenes[C]// Proceedings of the 2013 IEEE International Conference on Computer Vision. IEEE, 2013:569-576.

[39]Karatzas D , Lu S , Shafait F , et al. ICDAR 2015 Competition on Robust Reading[C]// 13th International Conference on Document Analysis and Recognition (ICDAR). IEEE Computer Society, 2015:1156-1160.

[40]Risnumawan A , Shivakumara P , Chan C S , et al. A Robust Arbitrary Text Detection System for Natural Scene Images[J]. Expert Systems with Applications, 2014, 41(18):8027-8048.

[41]Mishra A , Alahari K , Jawahar C V . Top-Down and Bottom-Up Cues for Scene Text Recognition[J]. Proceedings CVPR. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2012.

[42]ASTER. https://github.com/bgshih/aster.

[43]Almazan J , Gordo A , Fornes A , et al. Word Spotting and Recognition with Embedded Attributes[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 36(12):2552-2566.

[44]Bai F , Cheng Z , Niu Y , et al. Edit Probability for Scene Text Recognition[J]. 2018:1508-1516.

[45]PyTorch. https://pytorch.org/.

[46]Bissacco A , Cummins M , Netzer Y , et al. PhotoOCR: Reading Text in Uncontrolled Conditions[C]// 2013 IEEE International Conference on Computer Vision (ICCV). IEEE, 2013:785-792.

[47]Gordo A . Supervised mid-level features for word image representation[C]// IEEE Conference on Computer Vision & Pattern Recognition. IEEE, 2015:2956–2964.

[48]Sar.https://github.com/wangpengnorman/SAR-Strong-Baseline-for-Text-Recognition.

[49]Jaderberg M , Simonyan K , Vedaldi A , et al. Deep Structured Output Learning for Unconstrained Text Recognition[J]. Eprint Arxiv, 2014.

[50]Rodriguez-Serrano J A , Gordo A , Perronnin F . Label Embedding: A Frugal Baseline for Text Recognition[J]. International Journal of Computer Vision, 2015, 113(3):193-207.

[51]Su B , Lu S . Accurate Scene Text Recognition Based on Recurrent Neural Network[J]. 2014:35-48.

[52]Cheng Z , Bai F , Xu Y , et al. Focusing Attention: Towards Accurate Text Recognition in Natural Images[C]. IEEE International Conference on Computer Vision (ICCV), 2017:5086-5094.

[53]Devlin J , Chang M W , Lee K , et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. 2018.

[54]Hu H , Gu J , Zhang Z , et al. Relation Networks for Object Detection[J]. 2017:3588-3597.

[55]吕鹏原. 自然场景文字检测与端到端识别算法研究[D].华中科技大学,2018.

攻读硕士学位期间发表的论文及其它成果

（一）发表的学术论文

[1] Yang Z, Wu X, Zhou J. Learning from synthetic data for automatic license plate detection and recognition[C]// Tenth International Conference on Digital Image Processing (ICDIP 2018). International Society for Optics and Photonics, 2018, 10806: 1080624.

[2] Zhang R, Wu X, Qiu L, Yang Z. OCR with a convolutional neural networks integration model in machine vision[C]// Tenth International Conference on Digital Image Processing (ICDIP 2018). International Society for Optics and Photonics, 2018, 10806: 1080624.

哈尔滨工业大学学位论文原创性声明和使用权限

学位论文原创性声明

本人郑重声明：此处所提交的学位论文《基于 2D 注意力机制的不规则场景文本识别算法》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：杨志成 日期：2019 年 7 月 11 日

学位论文使用权限

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

（1）学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；（2）学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；（3）研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。

本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：杨志成 日期：2019 年 7 月 11 日

导师签名： 日期：2019 年 7 月 11 日

致　谢

这是第二次在论文上致谢，忽然发现距离本科毕业已经过去三年。三年时间过的非常仓促，很多事情都如过眼云烟，转瞬即逝。以至于，我已经忘记了三年前致谢的内容。为了让极有可能是最后一次论文致谢，记忆更长久一些。我决定说一些发自内心的话。

感谢我的导师吴晓军老师，如果没有吴老师的收留，我的余生也许就会在一个小县城的工厂里面度过，碌碌无为。

感谢实验室的各位同学，包容我的偏执。特别感谢静辉，听了我三年的故事，不厌其烦的安慰那无知的少年。台风前的那个夜晚，我现在想起来都热泪盈眶。

感谢我的父母和姐姐，他们是我前行的动力与人生的港湾。

感谢你，虽然我不敢说出你的名字，但是你第一次让我想成为更好的自己。希望你能精彩地度过自己追求的人生。

最后，向各位参与我的毕业论文评审和答辩的老师，表达我衷心的感谢。

突然想起来了，三年前致谢的最后一句话：感谢我遇到的所有人，感谢有你。