



東南大學

## 本科毕业设计（论文）报告

题 目： 面向低分辨率场景文本图像的增强与识  
别技术研究

学 号： 61520324

姓 名： 许睿

学 院： 吴健雄学院

专 业： 计算机科学与技术

指导教师： 薛晖

起止日期： 2023.12~2024.6

## 东南大学毕业（设计）论文独创性声明

本人声明所呈交的毕业（设计）论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得东南大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

论文作者签名：\_\_\_\_\_ 日期：\_\_\_\_\_年\_\_\_\_月\_\_\_\_日

## 东南大学毕业（设计）论文使用授权声明

东南大学有权保留本人所送交毕业（设计）论文的复印件和电子文档，可以采用影印、缩印或其他复制手段保存论文。本人电子文档的内容和纸质论文的内容相一致。除在保密期内的保密论文外，允许论文被查阅和借阅，可以公布（包括刊登）论文的全部或部分内容。论文的公布（包括刊登）授权东南大学教务处办理。

论文作者签名：\_\_\_\_\_ 导师签名：\_\_\_\_\_

日期：\_\_\_\_\_年\_\_\_\_月\_\_\_\_日 日期：\_\_\_\_\_年\_\_\_\_月\_\_\_\_日

# 摘 要

中文摘要

关键词：关键字 1，关键字 2

## ABSTRACT

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguere possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nollem me ab eo delectari, quod ista Platonis, Aristoteli, Theophrasti orationis ornamenta neglexerit. Nam illud quidem physici, credere aliquid esse minimum, quod profecto numquam putavisset, si a Polyaeo, familiari suo, geometrica discere maluisset quam illum etiam ipsum.

KEY WORDS: Keywords1, Keywords2

# 目 录

摘    要 .....	I
ABSTRACT .....	II
目    录 .....	III
第一章 绪论 .....	2
1.1 课题背景和意义 .....	2
1.2 本文研究内容 .....	3
1.3 论文各章节安排 .....	3
第二章 相关工作 .....	5
2.1 场景文本识别 .....	5
2.2 场景文本图像超分辨率 .....	7
2.3 数据集 .....	11
第三章 模型原理与设计 .....	13
3.1 模型总体架构 .....	13
3.2 编码器模块 .....	15
3.3 特征增强模块 .....	16
3.4 双任务解码器 .....	17
3.5 损失函数 .....	18
第四章 实验结果与分析 .....	20
4.1 评估指标 .....	20
4.2 实现细节 .....	21
4.3 模型性能 .....	22
4.4 消融实验 .....	25
4.5 推理结果 .....	30
第五章 总结与展望 .....	31
参考文献 .....	32
附录 A 损失函数 .....	36
A.1 识别损失 .....	36
A.2 多任务损失 .....	37
附录 B 模型性能 .....	40
B.1 最优模型 .....	40
致    谢 .....	43

表 0.1 中英术语对照表

术语	英文	中文
STR	Scene-Text Recognition	场景文本识别
STISR	Scene-Text Image Super-Resolution	场景文本图片超分
NLP	Natural Language Processing	自然语言处理
OCR	Optical Character Recognition	光学字符识别
LR	Low Resolution	低分辨率
HR	High Resolution	高分辨率
SR	Super-Resolution	超分辨率
PSNR	Peak Signal Noise Ratio	峰值信噪比
SSIM	Structural Similarity Index Measure	结构一致性指标
SRB	Sequential Residual Block	序列残差块
SOTA	State of the arts	现有的工作
ACC	Accuracy	准确率
FLOPs	Floating-point Operations	浮点运算数
Params	Parameters	参数量
FPS	Frequence per Second	每秒推理图片张数

# 第一章 绪论

## 1.1 课题背景和意义

从一般性的场景图片中识别文本信息不仅能帮助深度学习模型在训练时理解场景逻辑，还能在推理时给予使用者更多场景相关的信息，该类型的任务被称为场景文本识别（Scene-Text Recognition, STR）。场景文本识别能在诸多实际应用中有重要意义：例如在辅助驾驶中，汽车可以动态识别路上的交通符号，以便根据路况和交通规则做出安全的决策。近年来有许多研究人员躬身场景文本识别领域并取得了很多优秀成果。这些模型在目前广泛使用的基准数据集<sup>[1][2][3]</sup>中对于高分辨率（High-Resolution, HR）的图像取得了非常高的准确率。现实中的 STR 任务比基准数据集的形式更为复杂，被识别的字形可能因设计要求等因素，单个字符可以有多种字体和不同程度的扭曲，因此也有很多研究人员提出可以使用矫正模块<sup>[4]</sup>对场景文本进行空间上的矫正，使其便于识别。因此给一张比较清晰的场景文本图像，当前有许多方式可以以较高置信度对其进行识别。

然而，在场景文本图像中的特征受到大量影响时，识别器的性能也大幅下降。例如，受到拍摄环境或拍摄器材的制约时，拍摄的场景文本图像可能带有一定的模糊，这对识别器的鲁棒性带来了巨大的挑战。因此有研究人员<sup>[5]</sup>提出了更加一般的 STR 任务，即对于带有模糊等不良因素的低分辨率（Low-Resolution, LR）图像，可以先进行场景文本图像超分辨率（Scene-Text Image Super-Resolution, STISR）处理，得到较为清晰的超分辨率（Super-Resolution, SR）图像后，再进行常规的 STR。该研究人员构建了专门的模糊场景文本数据集<sup>[5]</sup>用于评判模型在不良条件下的超分辨率和识别能力。与传统的图像超分辨率（Image Super-Resolution）任务不同，STISR 任务更加注重于文本的恢复，由于不少文本包含一定的语义信息，这不仅给图像超分辨率模型带来了挑战，也使 STISR 模型得到更多新的思路。

对于 LR 场景文本的识别，当前广泛使用的模型架构是如图 1-1 (a) 所示的串行架构。这种架构虽然直观，但同时存在不少问题。串行架构中，两个任务的耦合程度太高，STR 任务非常依赖于 STISR 任务的性能，如果 STISR 得到的 SR 场景文本图像存在一定的问题，则会对识别任务造成很大的麻烦；同时，STISR 任务模型的监督信号含有一定的识别损失而不是专门的超分辨率损失，因此并不会生成比较完整的 SR 场景文本图像。而本文实现的模型是如图 1-1 (b) 所示的并行架构，该架构首先使用统一的编码器将输入的图片映射到一个统一的特征空间，再通过不同任务的解码器输出相应的预测目标。该架构

的优势在于多种任务使用之间的耦合度低，识别效果受到 SR 质量影响的程度较低，同时 SR 的质量也不受识别损失的干扰。

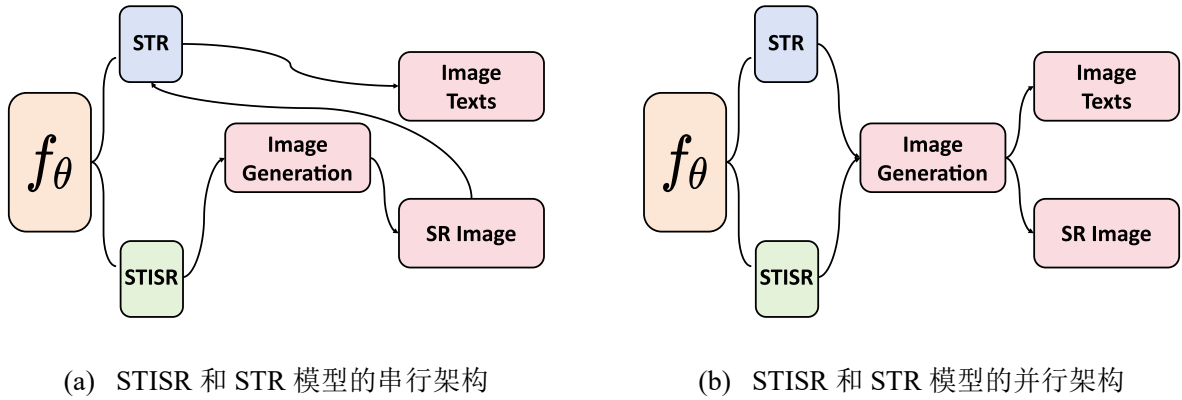


图 1-1 两种 STISR 和 STR 结合的架构

## 1.2 本文研究内容

本文主要使用如图 1-1 (b) 所示的多任务模型架构，分别设计了特征编码器、特征增强模块、识别解码器和超分辨率解码器四个部分，并使用了两阶段训练和多任务损失函数进行模型监督训练，最终实现了从 LR 场景文本图像到文本内容和 SR 图像的端到端任务。本文的主要贡献如下：

1. 提出了一般性的 LR 场景文本图像识别和超分辨率的模型架构；
2. 本文的模型使用了基于隐式扩散模型的特征增强模块，并在推理时对扩散模型进行有效的控制，实现在特征层面上增强 LR 图像；
3. 模型采用两阶段的训练模型，在第一阶段进行预训练，目的是让模型保存高分辨率特征信息，在第二阶段进行特征增强，目的是提升 LR 图像特征信息，提升多任务模型的泛化能力；
4. 本文使用基于不确定性权重的损失函数进行多任务模型训练，实现 STR 任务和 STISR 任务之间的平衡；

## 1.3 论文各章节安排

本文主要讨论 STR 任务和 STISR 任务的多任务模型和特征增强方法，以及本文所实现的模型在数据集上的性能。第二章介绍了目前现有的场景文本识别模型和场景文本图像超分辨率模型以及用于特征增强的扩散模型。第三章将会着重介绍本文提出的模型架构、模型设计思路的实现方式，并会从各个模块出发，分别介绍该模型实现特征增强的方



式和多任务解码方式。第四章主要评估第三章中模型的性能，并展示了多种对比结果。最后，第五章将会对本文提出的方法进行讨论和总结。

## 第二章 相关工作

本章将分别介绍场景文本识别（STR）模型和场景文本图像超分辨率（STISR）模型，并讨论其与本文模型的关系。除此之外，本章还将介绍本工作将要使用的 TextZoom 数据集，以便后文能顺利地引入本文模型。

### 2.1 场景文本识别

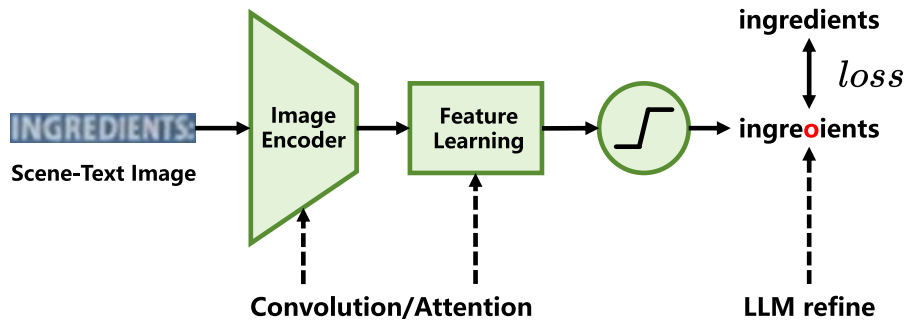


图 2-1 STR 识别器基本架构。其中虚线表示可选项

与传统的光学字符识别（OCR）不同，场景文本识别（STR）主要针对场景中的字符进行识别，OCR 面向的字符常常比较规整，而 STR 的识别对象经常由于拍摄问题，出现不同程度的透视或光影效果，有的甚至因为聚焦问题而出现低分辨率（LR）图像。对此问题，目前有很多研究人员提出了许多有效的方法，对高分辨率的场景文本进行识别，大部分模型的基本架构如图 2-1 所示，它们通常包含图像编码器用于将图像特征嵌入模型，再通过一系列的图像特征学习实现图像特征和文本特征的映射，最后使用激活函数将特征映射为标签，再经过损失函数将损失传播到模型参数，实现模型的训练。目前有研究人员在编码器和特征学习时使用卷积、循环神经网络以及注意力机制以便提取出更好的特征，除此之外，还有研究人员使用预训练的大语言模型对输出结果进行优化。基于相关工作中使用的不同特征提取方式，本节从基于卷积和循环网络、基于注意力机制和基于大语言模型三个角度介绍目前常用的 STR 模型，以及他们在基准数据集上的性能。

**卷积和循环网络。**卷积和循环网络在计算时拥有绝佳的性能，用该模块设计的网络常常具有训练和推理速度快、模型体积小等特点，因此仍有大量研究人员使用其作为图片特征提取和文本信息处理的主要模块。例如 SVTR<sup>[6]</sup> 模型使用不同步长的卷积运算模拟 ViT<sup>[7]</sup> 将图片分割为多个小块（patch）的操作，再对其进行后续处理，实验证明，这种图片处理方式能较为高效地将图片嵌入模型。CRNN<sup>[8]</sup> 模型则使用多个卷积层深度提取

图像特征，并使用双向的 LSTM (Bidirectional LSTM, BiLSTM)<sup>[9]</sup> 作为图片和文本之间的映射模块，最终使用 CTC 损失<sup>[10]</sup> 监督模型训练，其识别结果在当时取得了较高性能。而对于含有透视、扭曲结构等场景文本中，ASTER<sup>[4]</sup> 模型在其主干网络之前添加了矫正模块，将弯曲文本矫正为水平排布的文本，并最终使用基于循环网络 LSTM<sup>[9]</sup> 和 GRU<sup>[11]</sup> 的识别模块，最终在弯曲文本的识别中取得非常好的性能。目前，大部分的模型都使用卷积提取图片特征，这种提取方式的优势在于能够尽量保存原有的图像特征，对于文本识别有很大帮助。本文实现的模型同样使用卷积作为编码器的主干。

**注意力网络。**注意力机制在 NLP 领域中得到了广泛应用，使用注意力机制能大幅提升模型对较长序列处理能力，不会出现类似 LSTM 等 RNN 模型在使用较长序列进行训练时出现的梯度消失或梯度爆炸的现象。对于 STR 任务，存在两种使用注意力机制的情况，一是针对文本进行注意力学习，再与图像特征进行交叉注意力学习，二是先将图像的特征提取后嵌入模型，再直接对图像进行注意力学习，最后用损失约束输出结果。例如 PARSeq<sup>[12]</sup> 使用了多种注意力机制进行训练。该模型的创新点在于使用排列数作为注意力机制的掩码，在文本特征层面使用了自注意力机制，并与图像特征共同使用交叉注意力，学习文本特征和图像特征中的映射关系，在当时的数据集上取得了较高性能。由于 PARSeq 模型在多个合成数据集上进行了大规模预训练，因此最终得到的网络具有很强的先验知识，可以根据预训练网络进行微调。前文提到 SVTR<sup>[6]</sup> 模型利用带有不同步长的卷积运算模拟 ViT 将图像嵌入网络，同时，该模型还使用多个注意力机制用于提取局部和全局特征，该注意力机制使用不同大小的掩码块引导模型对图像注意力的学习。除此之外，MGP-STR<sup>[13]</sup> 以 ViT 为基础架构，将输入的文本图像分割为互不重叠小块。相较于 ViT 将不同的图像小块（即每个块的特征）聚合在一起作为整个图像的特征表示，在图像文本识别中，MGP-STR 使用了一种基于注意力机制模块对识别器进行多重约束，用于提取更加注重于文本的特征，将图像的小块有意义地聚合在一起，使得该模型可以预测字母、词组和字母个数等不同粒度的文本信息。DPAN<sup>[14]</sup> 模型在架构上进行了研究，并基于并行解耦的编码器-解码器架构 (Parallel-Decoupled Encoder Decoder, PDED)，改进了其注意力的查询输入矩阵 (query)，从而弥补了查询矩阵和键值矩阵 (key) 之间的图像信息，提高了模型的鲁棒性。尽管注意力机制的特征表示能力比卷积和循环模块强，但注意力机制可以让模型在全局特征的角度上进行计算，适合长度适中的特征序列，但

其计算复杂度随着序列长度的增加而成平方及增长，不仅如此，其模型参数规模也远大于卷积和循环模块。

**大语言模型辅助网络。**在大语言模型（Large-Language Model, LLM）兴起后，很多与文本相关的任务都可以借助大语言模型实现。由于大语言模型使用大量的数据集进行训练，而模型也拥有巨大的参数量，因此训练出来的大语言模型有着很强的特征表示能力。早期的大语言模型如 BERT<sup>[15]</sup> 基于 Transformer<sup>[16]</sup> 编码器设计了双向编码的无监督训练，其参数量达到了 1.1 亿（110 million<sup>[15]</sup>）。后续 GPT 系列<sup>[17]</sup> 模型则使用自监督的方式进行训练，其中 GPT-3<sup>[18]</sup> 的参数量达到了 1.75 千亿（175 billion<sup>[18]</sup>）。在研究图像和文本之间的关系中，CLIP<sup>[19]</sup> 多模态模型使用对比学习，同时得到了文本和图像的特征表示编码器。大语言模型中拥有的大量参数使其具有很强的特征的表示能力，因此目前有研究人员直接使用大语言模型的编码器对下游任务中的数据集进行编码表示，并设计模型直接学习编码后的特征。使用预训练的大模型可以在文本识别任务中取得较好性能。例如在 CLIP4STR<sup>[20]</sup> 工作中借助了视觉语言模型（Vision-Language Model, VLM），认为当前大部分识别器是基于单模态进行训练的，而 VLM 给予了文本和对应图像的信息，该模型通过视觉解码器和混合模态解码器的预测结果优化，在 11 个基准测试集中达到了较高的识别准确率。与此同时，由于大语言模型含有丰富的上下文语义信息，因此有研究人员使用大模型进行文本预测结果的优化。例如 ABINet<sup>[21]</sup> 设计了两个分支处理场景文本识别，其一是以残差网络和 Transformer 为基础架构，并且带有位置注意力的视觉分支，其二则是以 CLIP 为基础的，用于优化预测文本的语言模型分支。实验发现使用语言模型进行结果的迭代优化可以在基准数据集上达到较高识别准确率。正如前文所述，使用预训练的大语言模型作为特征编码器或预测优化器，可以达到较高的识别准确率，同时由于大语言模型的强语义性，识别的结果绝大多数依然带有语义性，即识别结果是正确的单词。但在场景文本中，存在许多无上下文语义的文本，因此在使用大语言模型时需要权衡语义信息和图像信息之间的关系。

## 2.2 场景文本图像超分辨率

场景文本图像超分辨率（STISR）的目的是提高低分辨率文本图像的质量，STISR 能大大提升前文提到的 STR 任务对低分辨率文本的识别率，因此对该领域的研究对自动驾驶等下游任务的发展有着重要意义。与单张图像的超分辨率略有不同，STISR 需要弥补图像特征和文本语义信息之间的模态差异，STISR 的工作包含了文本的语义信息，在增

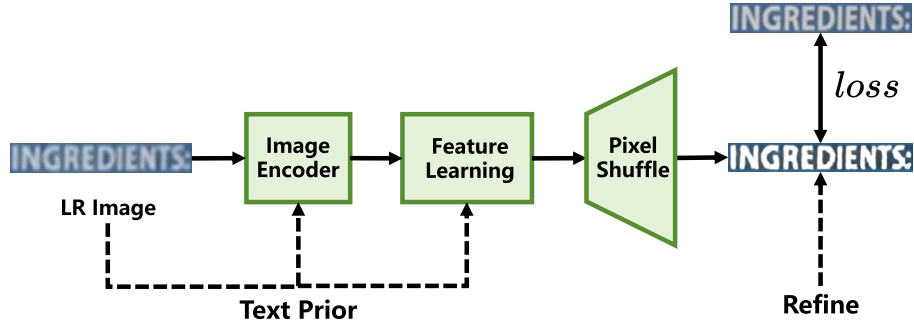


图 2-2 STISR 超分器基本架构。其中虚线表示可选项

强图像分辨率的同时不能破坏文本结构，因此 STISR 的约束条件比普通单张图像超分辨率更强。如图 2-2 所示，目前常用的模型框架通常包含一系列的编码器和特征学习模块，末尾通常使用像素重排模块<sup>[22]</sup>提高图像分辨率。目前，已经有很多研究人员在该领域做出了巨大贡献，例如有研究人员将文本先验知识引入图像的特征学习模块中，用先验特征引导超分辨率，还有研究人员在 SR 图像的基础上进一步做结果的优化。本节先介绍传统插值的图像超分辨率方法，接着以该方法为基础，引出当前该领域的相关工作。

**插值。**插值主要目的是根据给定数据点，估计给定点附近的值。用于图像超分辨率的插值方式包括最近邻插值、双线性插值和双三次插值等方式。如图 2-3 假设原始图像大小为  $(h, w)$ ，以将图片分辨率放大到原来的**两倍**为例，即超分辨率后的图像大小为  $(2h, 2w)$ ，分别讨论三种超分辨率方式的效果。对于一个像素点，最近邻插值忽略其周围像素值，直接在周围扩展该像素的值，其效果如图 2-3（c）所示，可以发现其锯齿化比较严重。而双线性插值会考虑到以该像素点为顶点的周围像素值，假设四个顶点分别为  $A(x_1, y_1), B(x_1, y_2), C(x_2, y_2), D(x_2, y_1)$ ，这些点所代表的像素值分别由  $f(A), f(B), f(C), f(D)$  给出，则该四个顶点间任意一个点的像素值  $P(x, y)$  由公式公式 (2.1) 公式 (2.2) 和公式 (2.3) 给出。其插值效果如图 2-3（d）所示。相较于最近邻插值，双线性插值在一定程度上减轻了锯齿化。而双三次插值插值一次使用的像素点个数是双线性插值的 4 倍，即使用 16 个像素值估计一个像素值，类似于双线性插值其基本思想是使用三次多项式函数拟合给定的四个像素值，并通过一阶导数和二阶混合偏导保证连续性，从而得到 16 个方程组成的方程组，解出 16 个顶点对应的权重。双三次插值的效果如图 2-3（e）所示，可以看出其效果比双线性插值多了更多的图像细节。

$$f(P_{y_1}) = \frac{x_2 - x}{x_2 - x_1} f(A) + \frac{x - x_1}{x_2 - x_1} f(B) \quad (2.1)$$

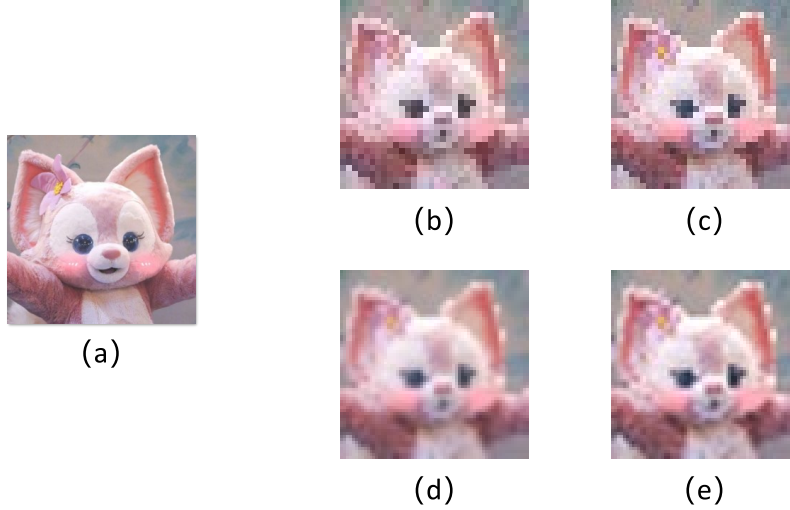


图 2-3 (a) 表示原始高分辨率图像  $(2h, 2w)$ , (b) 表示压缩后大小为  $(h, w)$  的图像等比放大后的低分辨率图像  $(2h, 2w)$ , (c) 表示使用最近邻插值方式的超分图像  $(2h, 2w)$ , (d) 表示使用双二次插值的超分图像  $(2h, 2w)$ , (e) 表示使用双三次插值的超分图像  $(2h, 2w)$

$$f(P_{y_2}) = \frac{x_2 - x}{x_2 - x_1} f(D) + \frac{x - x_1}{x_2 - x_1} f(C) \quad (2.2)$$

$$f(P) = \frac{x_2 - x}{x_2 - x_1} f(P_{y_1}) + \frac{x - x_1}{x_2 - x_1} f(P_{y_2}) \quad (2.3)$$

使用插值的方式进行图像超分辨率，是在图像的宽和高两个维度上进行像素扩展，在使用深度学习模型进行超分辨率时，需要先将 LR 图像插值到 SR 图像，再使用卷积、注意力机制等模块增强其分辨率。而近年来提出的像素重排（PixelShuffle）模块<sup>[22]</sup>可以直接在 LR 图像上提取特征，它的主要思想是将通道维度特征重组到宽和高维度上，该方式既能提高图像分辨率，又能在通道维度上提取特征，是当前网络增强图像分辨率的主要方式。如图 2-4 所示，像素重排主要基于网络提取的特征图，特征图的通道维度上是具有超分辨率像素信息的特征，通过重排后，通道维度的像素信息补全到宽和高两个维度，从而增强图像分辨率。与传统的插值方式相比，这种在通道维度上提取特征的方式可以减少网络传播中特征图的大小，同时也更适合具有卷积模块的网络在通道层面上提取特征的方式。基于该模块，有很多研究人员提出了新的超分辨率模型，并在基准数据集上取得了较好的结果。

**模型。**使用卷积和循环网络搭建模型的代表性工作为 TSRN<sup>[5]</sup>，同时该模型的作者也是 STISR 任务的提出者，并创建了新的数据集用于训练和测试。该模型的主要思路是残

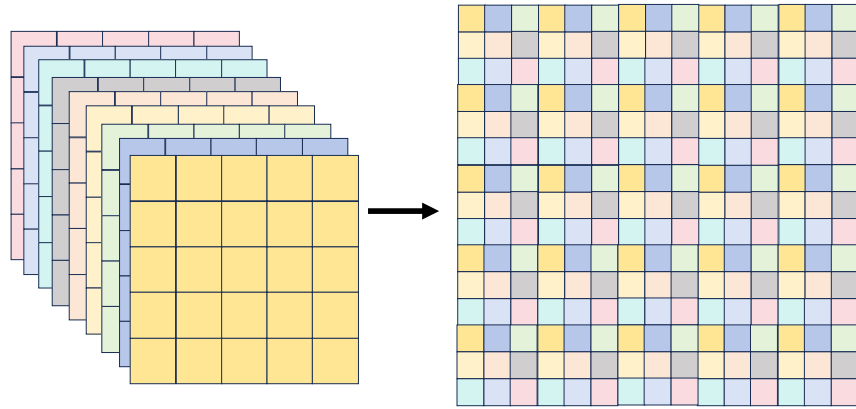


图 2-4 像素重排（PixelShuffle）模块的主要操作原理。原特征图的通道维度被重排至宽和高两个维度，从而提高图像的分辨率

差学习，模型使用了重复的超分子模块 **SRB** [5] 对超分图像的残差进行学习后，将残差与浅层特征进行加和，最后使用像素重排输出较高分辨率的图像。在整个网络的特征传播中，特征图的大小保持一致且与 LR 图像大小相同，模型仅在通道维度上进行特征学习，与前文介绍的像素重排相互印证。TSRN 是 STISR 领域的开山之作，因而有研究人员延续其特点，提出了新的架构。例如 TATT 模型 [23] 在此基础上增加了注意力机制，目的是混合先验文本信息和图像特征，在一定程度上用文本的先验特征引导超分辨率的进行。TATT 模型在 LR 图像输入时，通过预训练识别器对 LR 图像先进行一次识别得到预测的文本标签，同时用多个卷积模块提取 LR 图像的图像特征，再将二者输入到带有交叉注意力的特征融合模块，最后通过 TSRN 的主干网络输出超分辨率图像。与此思想类似的工作还包括 TPGSR [24]，该模型同样使用了文本先验模块对超分辨率模块进行引导，并同时使用由高分辨率图像产生的文本先验约束 LR 图像的文本先验，以便生成更可靠的文本先验。由于大部分超分网络具有一定的缺陷，有研究人员基于这些超分网络，设计了即插即用的超分辨率模块用于提高预训练超分模型的性能。DPMN [25] 则从优化的角度出发，在原有的超分辨率模型的基础上添加即插即用的性能提升模块，将现有的超分辨率模型输出的 SR 图像作为 DPMN 模块的输入，输入 PGRM（Prior-Guided Refinement Module）后使用预训练的 ViT 提取先验信息，并对其做像素重排，将多个 PGRM 的输出再进行组合，最终得到分辨率更高的 SR 图像。PCAN [26] 则训练出用于提取超分语义信息的注意力模块，并提出了新的损失函数用于模型的监督训练。STT [27] 模型则使用注意力机制关



注单个文本的字形的细节特征，并在特征图层面上用损失函数进行监督。而 C3-STISR<sup>[28]</sup> 则从不同特征理解的角度出发，利用注意力机制权衡识别、图像和语义三种理解，在基准数据集上超过同期模型。而随着 DDPM<sup>[29]</sup> 等扩散模型（Diffusion Model）的提出，研究人员发现该模型在图像恢复上面有一定的效果，因此部分研究人员将 DDPM 及其衍生模型引入 STISR 领域，扩充了文本图像的超分辨率模型类型。DDPM 在原图上进行多步加噪和多步去噪过程，用模型预测加的噪声，最终推理时用纯噪声逐步减去预测的噪声，实现图像的生成。由于原始的 DDPM 生成效果带有随机性，因此 SR3<sup>[30]</sup> 基于隐式扩散模型（Stable Diffusion）将 LR 图像作为条件概率中的条件，引导生成图像的效果，最终在单张图像超分上取得了很好的效果。在 STISR 任务上，TextDiff<sup>[31]</sup> 首次使用扩散模型进行残差学习，在基准数据集上取得良好效果。

## 2.3 数据集

对于单独的 STR 任务而言，目前绝大部分识别器需要在大规模的人造数据集和真实数据集上进行训练，得到的模型通常包含了与文本相关的语义信息，可以在诸多下游任务的数据集上进行微调。用于大规模训练的人造数据集有包含了 900K 样本量的 MJSynth<sup>[32]</sup> 和包含了 800K 样本量的 SynthText<sup>[33]</sup>。当前已经有很多 STR 工作在这些数据集上进行训练，用于后续模型的微调。而用于 STISR 和 STR 双任务的数据集目前常用的是 TextZoom<sup>[5]</sup>，该数据集由两个单张图片超分辨率的基准数据集 RealSR<sup>[34]</sup> 和 SR-RAW<sup>[35]</sup> 筛选和截取而成，其中标签的分布如图 2-5 所示，图像的分布如图 2-6 所示。TextZoom 分为训练集（包含验证集）、简单难度的测试集、中等难度的测试集和困难难度的测试集，各个数据集的样本量如图 2-5 (a) 所示。如图 2-5 (b) 所示，在文本字符角度，数据集的标签文本类型包含了数字和大小写英文字母，每个图像的标签不仅包含数字，

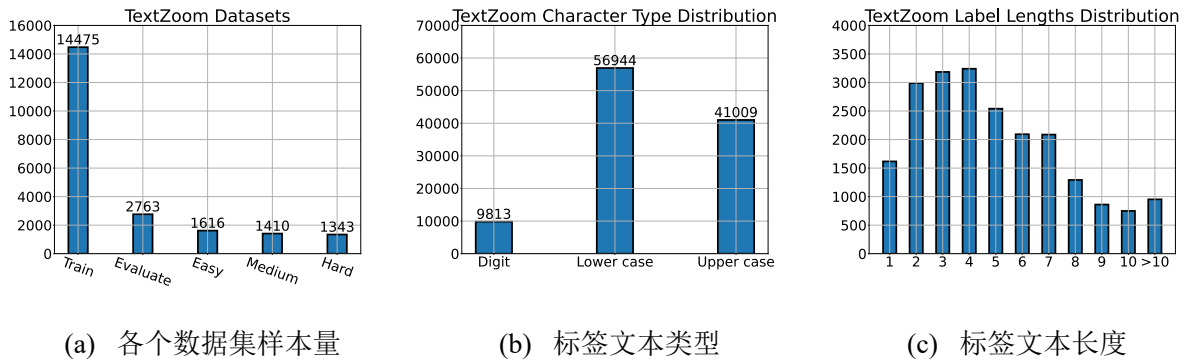
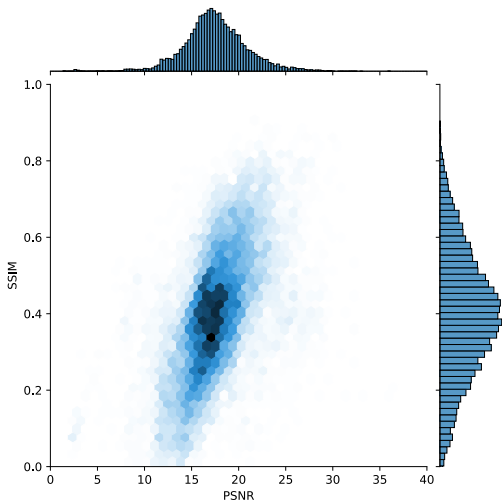
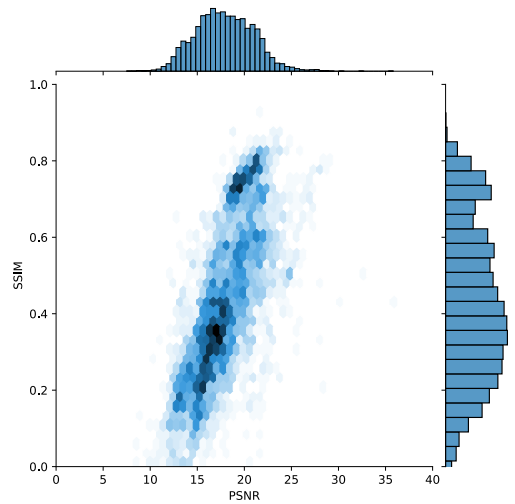


图 2-5 TextZoom 数据集标签特征统计





(a) 训练集 PSNR 与 SSIM 分布图



(b) 测试集 PSNR 与 SSIM 分布图

图 2-6 TextZoom 数据集图像特征统计。PSNR 和 SSIM 越高表示 LR 图像质量越高

也有可能同时包含大小写的英文字母。如图 2-5 (c) 所示，数据集大部分的标签长度在 10 以内，且各个长度的分布比较均匀。在图像方面，图像的质量由 PSNR (Peak Signal Noise Ratio) 和 SSIM (Structural Similarity Index Measure) 反映，指标的值越高，则两张图像越接近。在图 2-6 中，本文统计了 LR 图像相对于 HR 图像的 PSNR 和 SSIM，可以发现训练集和测试集在分布上有一定的相似性，因此使用 TextZoom 训练集作为模型的训练数据。而由于测试集图像质量比较分散，因此使用不同难度的 LR 图像作为模型性能的评估数据集也是合理的。

同前文所述，使用在大规模人造数据集和真实数据集上预训练的识别器进行微调可以给模型带来一定的先验语义特征，TextZoom 数据集和 MJSynth 数据集在标签和图像分布上存在一定差异，适合用作微调数据集。因此本文识别器采用一种在 MJSynth 数据集上预训练后的识别器作为主干，并研究了相应的微调方法，最终取得较好的识别性能。

### 第三章 模型原理与设计

本章首先介绍模型的总体架构及其形式化描述，再分别详细介绍模型各个模块的设计和实现效果，最后给出模型训练所使用的损失函数以及训练和推理算法。

#### 3.1 模型总体架构

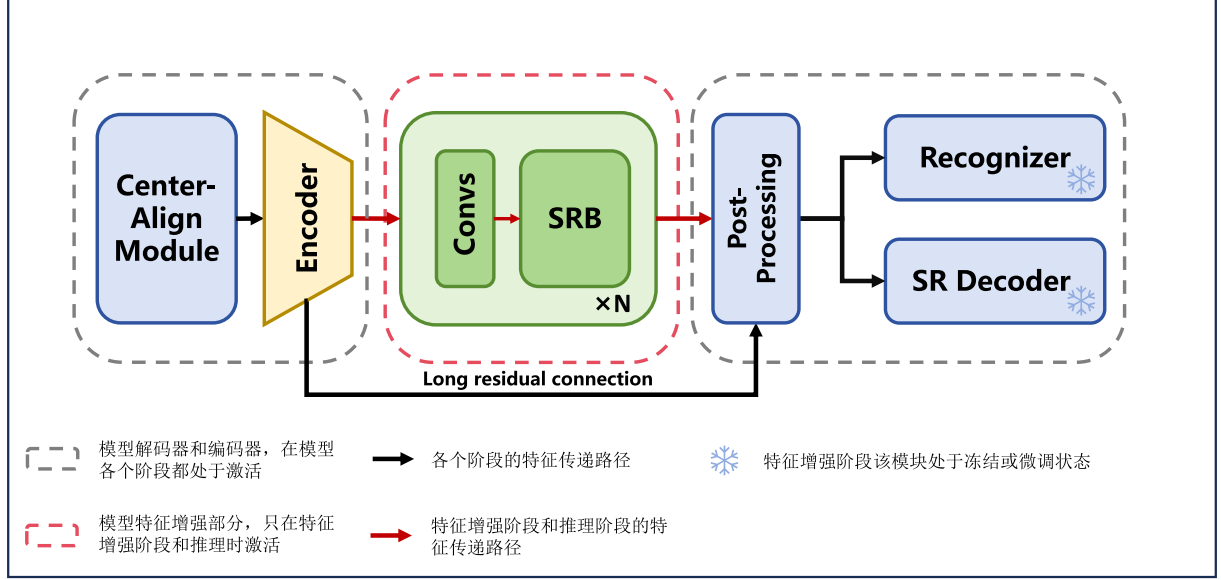


图 3-1 模型整体架构。模型编码器和解码器（灰色虚线框内部）在各个阶段都处于激活状态，特征增强部分（红色虚线框内部）只在特征增强阶段被激活和训练。在编码器和后处理模块中使用了额外的长残差连接

不失一般性，模型的输入和输出都是以一个批量大小进行，超分辨率的倍数为 2，假设输入的 LR 低分辨率图像用  $I_l$  表示，对应的 HR 高分辨率图像用  $I_h$  表示，而超分辨率模型的输出 SR 为  $I_s$ ，图像对应的真实标签和识别器预测标签在嵌入模型后，分别为  $S_t, S_p$ 。图像和嵌入标签维度分别满足： $I_l \in \mathbb{R}^{B \times C \times H \times W}$ ,  $I_h \in \mathbb{R}^{B \times C \times 2H \times 2W}$ ,  $I_s \in \mathbb{R}^{B \times C \times 2H \times 2W}$ ,  $S_t \in \mathbb{R}^{B \times L_t}$ ,  $S_p \in \mathbb{R}^{B \times L_p}$  其中  $B$  表示输入和输出数据的批量大小， $C$  表示输入和输出图像的通道数，通常为 3，而  $H, W$  分别表示 LR 图像的高和宽，由于放大倍数是 2，因此 SR 图像和 HR 图像的高和宽分别是  $2H, 2W$ ，对于标签， $L_t, L_p$  分别表示了真实标签的长度和预测标签的长度。

该模型的目标是通过训练集  $\mathcal{S} = \{(L_t, I_h, I_l)_i\}_{i=0}^N$  训练模型  $\mathcal{F}_\theta$ ，使其能够以  $I_l$  为输入，预测对应的标签  $L_p$  和重建 SR 图像  $I_s$ ，即

$$\mathcal{F}_\theta(I_l) = (L_p, I_s) \quad (3.1)$$

从目前研究中发现，现有的模型多是基于串行结构搭建，即先对 LR 图像进行超分辨率，提高文本图像的分辨率，再用识别器进行文本预测。类似这一类的串行结构理解简单、训练更为方便，同时有很多研究人员达到了部分预期效果。然而这种框架经常会带来一些问题：

1. 首先，由于识别结果的准确率大多基于 SR 图像的质量，因此如果超分辨率模块的输出结果并不可靠，从而生成错误的超分辨率图像，就极有可能导致后续识别器预测出错误的文本。
2. 其次，两个任务是串行关系，因此如果不使用梯度截止等方法，后一个模块的损失在进行传播时会影响到前一个模块参数的更新，从而导致超分辨率模块被识别损失监督，原有的超分损失受到影响，从而生成的 SR 图像并不完全有着较高的保真度。

为解决以上问题，本文使用了一种新型的多任务并行架构，如图 3-1 所示，该模型的识别模块和超分模块处于并行关系，二者共享图像的编码特征，但解码特征相互独立。在使用多个损失函数进行监督训练时，两个解码器会更加注重于更新参数使得与其对应的损失降低，而不会受到另一个损失的影响。其次由于识别器从特征层抽取特征，受到超分辨率效果的影响较小，两个任务的耦合度大大降低。为了进一步提升图像特征的质量，该模型还设计了特征增强模块，提高 LR 图像的特征质量。设编码器为  $\mathcal{E}_\theta$ ，识别器为  $\mathcal{D}_\theta^r$ ，超分器为  $\mathcal{D}_\theta^s$ ，特征增强模块为  $\mathcal{A}_\theta$ ，编码器输出特征为  $F_e \in \mathbb{R}^{B \times C_e \times H_e \times W_e}$ ，特征增强模块输出为  $F_a \in \mathbb{R}^{B \times C_a \times H_a \times W_a}$ （其中  $C_x, H_x, W_x$  分别表示中间特征的通道数、高度和宽度）该架构可以由公式 (3.2) 表示：在推理阶段，模型的输入为 LR 低分辨率图像，先经过编码器进行初步的特征提取，再由特征增强模块提取 LR 特征潜在的超分辨率图像特征和文本特征，最后将含有超分辨率和文本识别混合特征的张量图输入不同任务的解码器，从而输出识别文本和 SR 超分辨率图像，实现模型对 LR 图像的识别和超分辨率。

$$\begin{aligned}\mathcal{F}_\theta(I_l) &= (\mathcal{D}_\theta^r(F_a), \mathcal{D}_\theta^s(F_a)), \\ F_a &= \mathcal{A}_\theta(F_e), \\ F_e &= \mathcal{E}_\theta(I_l)\end{aligned}\tag{3.2}$$

整个模型的训练方式可以分为两个阶段：预训练阶段和特征增强阶段。

**预训练阶段。**该阶段特征增强模块不接入模型。模型主要使用降采样后的 HR 图像训练识别器分支和超分辨率分支。该阶段目的是将 HR 图像的编码特征保留在编码器中，

让编码器学习到 HR 图像的特征解码方式，以便后续能以类似的方式提取 LR 图像的特征。其次是预训练超分器和识别器，提高超分器的重建效果和识别器对编码特征的识别性能。该阶段训练主要为后续使用 LR 图像训练提供模型的先验初始化参数。

**特征增强阶段。**该阶段主要训练特征增强模块，需要将其接入模型。特征增强思路是使用 LR 图像和 HR 图像共同进行训练，使得该模块能输出与 HR 编码后特征相近的超分辨率特征。同时，该阶段需要微调解码器的两个分支，以便弥补 SR 特征和降采样 HR 特征之间的分布差异。

### 3.2 编码器模块

编码器主要负责对齐图像和混合特征。

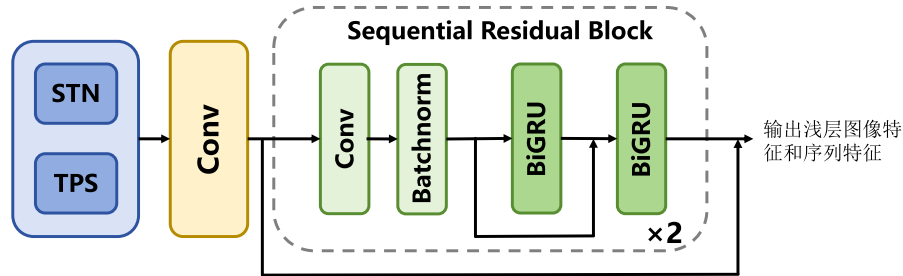


图 3-2 编码器模块，虚线部分为 SRB

**图像中心对齐。**由于数据集通过失焦采样真实的场景文本，LR 图像中心和 HR 图像中心会出现一定的偏移，因此模型在编码器部分设计了空间变换网络（Spatial Transformer Network, STN）<sup>[36]</sup> 用于对图像做空间尺度的变换，将 LR 图像与 HR 图像中心对齐。同时编码器还使用薄板样条插值（Thin Plate Spline, TPS）<sup>[37]</sup> 扩大 STN 变换的灵活性。STN 模块实现对图像变换的预测，增强卷积对图像剪切、旋转和仿射等变换的鲁棒性，TPS 模块使用插值的方式，通过关注图像控制点的变换，实现估计图像除控制点以外其余点的位置。该部分的基本流程为：先通过 STN 预测输入图像变换后控制点的位置，再将控制点和图像输入 TPS 实现图像的变换。

**混合特征。**编码器的定位特殊，作为不同任务的前置特征提取模块之一，编码器的设计和训练需要多次实验测试。由第二章图 2-1 可知，通常的识别器包含图像特征编码器和后续的序列特征学习模块，而由第二章图 2-2 可知，通常的超分器同样包含图像特征编码器和后续的超分图像特征学习模块。因此本文设想通过构造一种以卷积为主的特征提取编码器，同时实现文本特征的提取和超分特征的提取。预训练阶段，编码器输入

的是 HR 高分辨率图像，超分模块的重建任务较为简单，该阶段需要注重文本特征的提取；特征增强阶段，需要在编码器后接入特征增强模块，此时模型的输入是 LR 低分辨率图像，此时预训练完成的编码器会先对 LR 低分辨率图像进行特征提取，接着将 LR 低分辨率特征进行增强，输出适应预训练解码器的超分特征，该超分特征含有文本信息。因而在编码器设计部分，本文主要利用编码器实现图像文本信息的浅层提取，而超分变率特征主要由后续的特征增强模块实现。

该模块的设计如图 3-2 所示，输入的图像先统一调整大小，使其与推理时输入的 LR 图像大小一致，具体而言，如果输入的是大小为  $(2h, 2w)$  的 HR 图像，则使用最近邻插值调整大小到  $(h, w)$ ；而如果输入是 LR 图像，则不进行大小调整。调整大小后的图像输入中心对齐模块，经过 STN 和 TPS 的处理后再进入卷积网络映射到特征层，接着使用 SRB [5] 整合图像特征和文本特征。其中 SRB 的设计如图 3-2 中间虚线部分所示，进入 BiGRU 模块前的特征图进行维度调整：先切割为纵向排列的条状特征（设该操作为  $f$ ），再将这些条状特征依次连接（设该操作为  $g$ ），即

$$\begin{aligned} g(f(F)) &: \mathbb{R}^{B \times C \times H \times W} \rightarrow \mathbb{R}^{(BH) \times W \times C} \\ f(F) &: \mathbb{R}^{B \times C \times H \times W} \rightarrow \mathbb{R}^{B \times H \times W \times C} \\ g(F) &: \mathbb{R}^{B \times H \times W \times C} \rightarrow \mathbb{R}^{(BH) \times W \times C} \end{aligned} \quad (3.3)$$

维度调整能够实现图像特征和文本特征的融合，相邻的条状特征更有可能对应于一个文本。处理后的特征图送入 BiGRU 进行特征提取，输出的特征映射回原维度。最后将输出特征与卷积特征通过残差进行连接，输入到下一个模块。为了提高编码器特征提取能力，编码器内部堆叠了两个 SRB 模块。预训练阶段，编码器输出的特征包括两部分，第一部分的特征会经过像素重排输入识别器分支，第二部分不经过像素重排，送入超分辨率分支继续处理超分辨率特征。特征增强阶段和推理阶段，编码器输出的特征是 LR 特征，不会经过像素重排，直接输入特征增强模块做进一步的特征提升。编码器的训练和解码器的训练方式一致，使用多个损失函数和不确定性加权，这一点将在本章最后一节进行讨论。

### 3.3 特征增强模块

特征增强模块主要负责增强的编码器输出的 LR 特征。

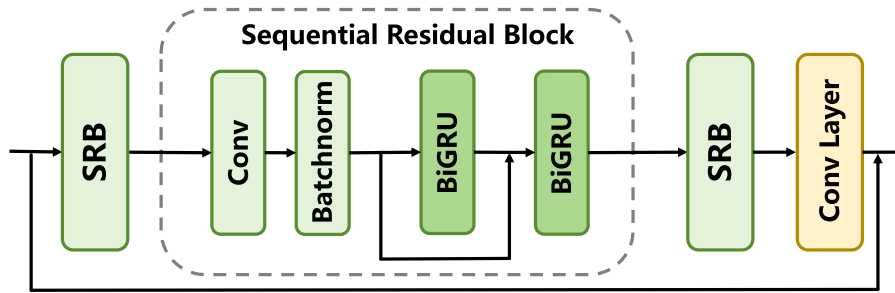


图 3-3 特征增强模块

与编码器类似，特征增强模块的主干网络使用 SRB 进行堆叠，但在整个模块中使用了更长的残差连接，并且该模块训练方式与编码器有所不同。如图 3-3 所示，特征图输入后先保留一份备份用于残差连接，再经过堆叠的  $N$  个 SRB 模块提取特征，接着通过卷积层融合图像和文本特征，最后使用残差连接将输出的特征图与进入网络的特征图相加，同 3.2 节所述，模块分别输出大小不同的特征图分别输入识别器和超分器。为了使用合适个数的 SRB 模块，本文在第四章对增强模块的 SRB 个数做了消融实验。

### 3.4 双任务解码器

双任务解码器分为识别器和超分器。识别器负责将输入的特征图解码为文本。超分器负责将特征图重建为 SR 超分辨率图像。

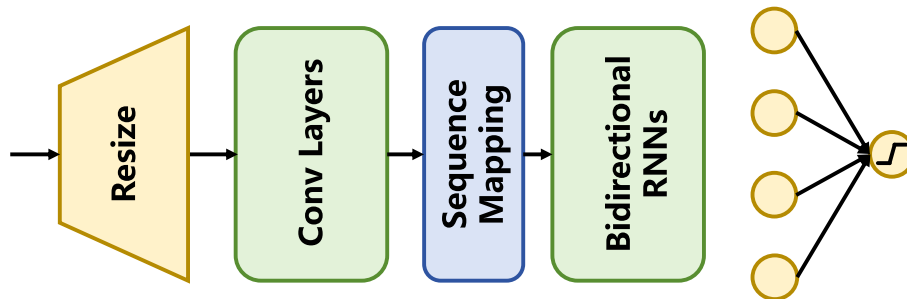


图 3-4 识别器模块

**识别器。**常见的工作中可以使用任何预训练识别器作为主干，只需要微调其分类头即可在相应数据集上实现较高的准确率。预训练模型中的编码器部分是用于提取图像特征的关键部分，该部分能提取多量、浅层的细节特征，而解码器部分用于处理这些浅层特征，并逐步整合为完整的深层特征，最后分类头和激活函数将深层特征映射为几率 (logits)。本文的识别器受到 CRNN<sup>[8]</sup> 的启发，在 CNN 和 LSTM 之间加入了 MLP 和残差连接，同时增强了模型分类头，如图 3-4 所示。在消融实验中，本文将讨论微调预训练识别器和重新训练识别器之间的差异。

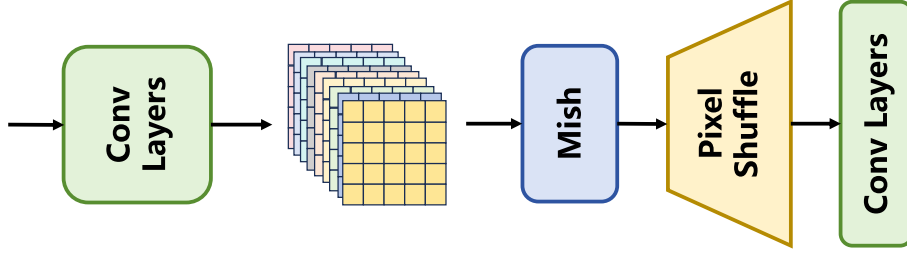


图 3-5 超分器模块

**超分器**。如图 3-5 所示，超分器先将输入的特征通过卷积层进行升维

$$\mathcal{D}_{\text{conv}} : \mathbb{R}^{B \times C \times H \times W} \rightarrow \mathbb{R}^{B \times (Cs^2) \times H \times W} \quad (3.4)$$

其中  $s$  代表超分辨率的倍数。经过一层 Mish 激活函数<sup>[38]</sup>后进入像素重排模块，按照  $s$  倍数放大特征图的高和宽

$$\mathcal{D}_{\text{pixel\_shuffle}} : \mathbb{R}^{B \times (Cs^2) \times H \times W} \rightarrow \mathbb{R}^{B \times C \times Hs \times Ws} \quad (3.5)$$

最终再由卷积层映射到原始图像的通道数。

### 3.5 损失函数

模型使用多个损失函数进行监督训练，并学习不确定性权重（Uncertainty Weight Loss）<sup>[39]</sup>来权衡不同任务之间的比重。

**识别损失 $\mathcal{L}_r$** 。识别器使用 CTC 损失函数<sup>[10]</sup>进行监督训练，该损失常用于文本序列预测的监督任务。在序列模型中，采用交叉熵损失或均方误差损失时，经常需要将输出的预测值与真实值标签对齐，这种对齐方式工作量大；除此之外，预测结果中可能存在空白字符和连续字符，空白字符可能由图像中两个文本间的空白区域预测得到，而连续字符则可能是图像中同一字符被多次识别，也有可能是文本中原本就存在连续字符；因此解码器的工作量较大。CTC 损失通过在标签的相邻字符中添加空白字符，在解码时先合并相同字符，再删去空白符得到预测标签，由于两个连续字符之间必然会存在空白字符，因而不会造成预测歧义。根据隐马尔可夫的前向后向算法可以得到该损失函数反向传播时的更新梯度为

$$g_t = \frac{\partial P_{\theta_t}(\hat{Y} | X)}{\partial \theta_t} = \sum_{i,j,c} \frac{\partial p_{i,j}(c)}{\partial \theta_t} \alpha_{i,j} \beta_{i,j+1} \quad (3.6)$$

该梯度的具体推导见附录。

**超分损失 $\mathcal{L}_s$** 。超分器使用两类损失共同监督训练。第一部分为 MSE 均方误差损失

$$\mathcal{L}_2 = \frac{1}{n} \sum_{i=0}^n (I_h(y_i) - I_s(y_i))^2 \quad (3.7)$$

该损失在图像像素空间对超分图像  $I_s(y) = \mathcal{F}_\theta(y)$  的输出进行了约束，监督图像在颜色上进行恢复。

第二部分基于梯度剖面先验（Gradient Profile Prior Loss, GPP）<sup>[40]</sup> 在图像的梯度场上设计了损失函数

$$\mathcal{L}_{GP} = \frac{1}{n} \sum_{i=0}^n \|\nabla I_h(y_i) - \nabla I_s(y_i)\|_1 \quad (3.8)$$

其中  $\nabla I$  为图像的梯度，即  $\nabla I = \left( \frac{\partial_x}{\partial x} I, \frac{\partial_y}{\partial y} I \right) = \sqrt{\left( \frac{\partial_x}{\partial x} I \right)^2 + \left( \frac{\partial_y}{\partial y} I \right)^2} \vec{n}$ （ $\vec{n}$  是同方向单位向量）。

图像梯度反映了图像的锐化程度，梯度场中的梯度剖面越小，说明该图像的边界越明显<sup>[40]</sup> 尖锐，在一定程度上说明该图像越清晰。最终的超分损失为二者的加权和

$$\mathcal{L}_s = \lambda_1 \mathcal{L}_2 + \lambda_2 \mathcal{L}_{GP} \quad (3.9)$$

其中  $\lambda_1, \lambda_2 \in \mathbb{R}$ ，在后续的消融实验中会进行该值的讨论。

**多任务损失权重。** 本文的模型涉及到多个损失函数，因此需要一种权衡任务规模、梯度大小的加权方式，实现模型使用多任务损失进行端到端训练。一种常用的加权方式是各个梯度间的线性组合

$$\mathcal{L} = \alpha_1 \mathcal{L}_r + \alpha_2 \mathcal{L}_s \quad (3.10)$$

这种加权方式简单，并且能够自行根据任务规模设置权重系数，但缺点是不能根据训练任务进行的状态实时更新权重，忽略了训练时各个梯度更新快慢不一致的特点。

而使用不确定性加权<sup>[39]</sup> 的方式则可以动态学习损失间的加权系数，加权系数中含有带学习的参数  $\sigma$ ，可以随着模型参数的更新同不更新，从而调整不同任务的损失函数权重，使用不确定性加权后的总损失可以写为

$$\mathcal{L} = \frac{1}{2\sigma_r^2} \mathcal{L}_r + \frac{1}{2\sigma_s^2} \mathcal{L}_s + \alpha \log(1 + \sigma_r^2) + \alpha \log(1 + \sigma_s^2) \quad (3.11)$$

其中  $\sigma_1, \sigma_2$  是待学习的参数， $\alpha$  是惩罚项，防止  $\sigma_1, \sigma_2$  过大，在训练前进行指定。该损失的具体推导可以参考附录。



## 第四章 实验结果与分析

本章主要对本文搭建的模型进行性能分析。本章首先从模型总体推理性能出发，将模型与部分现有模型（State of the art, SOTA）进行对比，着重分析每个模型推理性能的差异。其次本章设计了多个消融实验（Ablation Study），对模型中使用到的模块和训练方法进行了研究。

### 4.1 评估指标

本文使用了多个评估指标进行实验性能对比。

**峰值信噪比（PSNR）**。PSNR 主要用于评估超分辨率任务的性能，该指标一般用于衡量图像经有损压缩后的重建效果。其计算方式如公式 (4.1) 所示。

$$\text{PSNR}(I_s, I_h) = 20 \cdot \lg \left( \frac{\text{MAX}(I_h)}{\sqrt{\text{MSE}(I_h - I_s)}} \right) \quad (4.1)$$

$$\text{MSE}(I_h - I_s) = \frac{1}{rc} \sum_{i=0}^r \sum_{j=0}^c \|I_h(i, j) - I_s(i, j)\|^2$$

其中  $\text{MAX}(I_h)$  表示原图像  $I_h$  的最大像素值， $\text{MSE}(I_h - I_s)$  表示原高分辨率图像  $I_h$  和压缩后重建的超分辨率图像  $I_s$  之间的均方误差。不同于普通信噪比（SNR）的计算方式，PSNR 的分子使用图像信号的最大值而不是整个图像的模方，表示图像的重建效果更加关注图像的高强度区域的抗噪能力。

**结构一致性指标（SSIM）** [41]。SSIM 主要用于衡量超分辨率任务的性能，该指标通过比较超分辨率图像  $I_s$  和原始高分辨率图像  $I_h$  间的亮度  $l(I_s, I_h)$ 、对比度  $c(I_s, I_h)$  和结构  $s(I_s, I_h)$ ，对图像间的一致性进行衡量。其主要用像素值的统计量衡量图像的视觉特征，即图像像素值的均值可以反映图像的明暗程度，像素间的方差可以反映图像的对比度，而两张图像之间的协方差则可以反映像素值之间的关联程度。三个子指标的计算方式如公式 (4.2) 所示。

$$l(I_s, I_h) = \frac{2\mu_s\mu_h + c_1}{\mu_s^2 + \mu_h^2 + c_1}$$

$$c(I_s, I_h) = \frac{2\sigma_s\sigma_h + c_2}{\sigma_s^2 + \sigma_h^2 + c_2} \quad (4.2)$$

$$s(I_s, I_h) = \frac{\sigma_{sh} + \frac{c_2}{2}}{\sigma_s\sigma_h + \frac{c_2}{2}}$$

其中  $\mu_s, \mu_h$  分别表示超分辨率图像  $I_s$  和高分辨率图像  $I_h$  像素值的均值,  $\sigma_x, \sigma_y$  分别表示  $I_s$  和  $I_h$  像素值的标准差, 而  $\sigma_{sh}$  表示  $I_s$  和  $I_h$  之间的协方差, 为了防止分母出现 0 项, SSIM 的计算中添加了  $c_1, c_2$  两个变量。

因此总的SSIM指标可以由公式 (4.3) 计算得到。

$$\text{SSIM}(I_s, I_h) = l(I_s, I_h) \cdot c(I_s, I_h) \cdot s(I_s, I_h) \quad (4.3)$$

**准确率 (ACC)**。准确率主要用于衡量识别任务的性能, 计算方式比较简单, 如公式 (4.4) 所示, 即识别正确的标签数占总标签数的比例。

$$\text{ACC} = \frac{N_{\text{correct}}}{N_{\text{total}}} \quad (4.4)$$

值得注意的是, 衡量准确率时, 模型以标签文本整体而非单个字符作为识别统计的基本单位。

**浮点运算数 (FLOPs)**。用于衡量一个模型进行一次实例计算时需要进行的浮点运算次数。该指标可以反映一个模型的计算复杂度, FLOPs 越大, 表示模型进行一次推理所需要的浮点运算数越多, 模型的计算复杂度也就越大。

**参数量 (Params)**。模型参数量用于反映一个模型的大小。

**每秒推理图片张数 (FPS)**。FPS 能够衡量模型推理速度, FPS 越大表示推理速度越快。

## 4.2 实现细节

本文所实现的模型使用单张显存大小为24G 的 NVIDIA RTX 3090 显卡完成训练和推理。批量大小设置为 512 组数据, 其中一组数据包含0~9和a~z以及空白字符组成的文本标签、 $32 \times 128$  像素大小的高分辨率图像和  $16 \times 64$  像素大小的低分辨率图像。本文在3通道图像的基础上添加了一层灰度掩码, 不同分辨率图像的灰度掩码如图@所示。整个模型的输入是 $4 \times 16 \times 64$  大小的低分辨率图像。整个训练过程一共遍历300次训练集。模型使用 AdamW<sup>[42]</sup> 优化器进行参数优化, 并设置  $\beta \in [0.5, 0.999]$ <sup>[5]</sup>。由于模型使用了多个包含 BiLSTM 和 BiGRU 的模块, 为了防止梯度过大, 在训练时加入了梯度裁剪, 超分器分支的梯度裁剪为0.25, 识别器分支梯度裁剪为5。学习率更新策略采用 StepLR, 即每隔固定的步长, 将学习率按一定比例进行衰减。对于模型参数而言, 初始学习率设置为  $5 \times 10^{-4}$ , 学习率衰减比例设置为 0.5, 每遍历30次训练集进行一次衰减; 而对于可训练的损失权重而言, 与模型参数一致, 初始学习率和衰减比例分别设置为  $1 \times 10^{-5}$  和 0.8,

但衰减步长为模型参数的 1.5 倍,以便在模型训练初期以较大学习率调整任务权重,同时避免两组学习率以较大幅度衰减时模型出现的不稳定现象,学习率衰减曲线如图 4-1 所示。为了保证训练开始时两种损失策略对不同任务有相同的初始权重,在公式 (3.10) 中,设置  $\alpha_1 = \alpha_2 = 1.0$ , 在公式 (3.11) 中,设置  $\sigma_r = \sigma_s = \frac{\sqrt{2}}{2} \approx 0.7, \alpha = 0.1$ 。超分辨率任务损失函数公式 (3.9) 中的  $\lambda_1 = 20, \lambda_2 = 1 \times 10^{-4}$  [5], 本章后续将对该值进行消融实验。

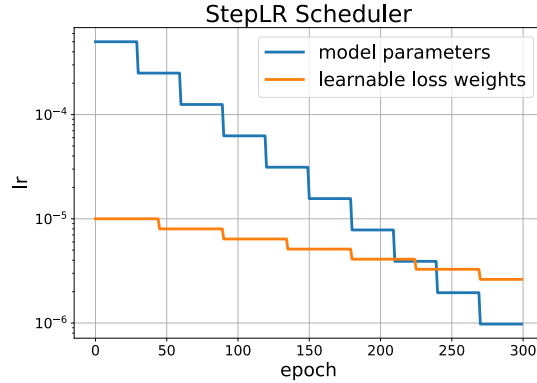


图 4-1 StepLR 学习率衰减策略。横轴为训练过程,纵轴(对数刻度)为学习率大小

由于是多任务训练,因此本文的 PSRec 模型在训练期间分别保存了准确率最优和 PSNR 最优的模型参数,在进行模型层面的对比时,如无特殊说明,本文的 PSRec 模型使用的是 PSNR 最优的模型参数。两种模型参数的指标对比差异可以查阅附录。除此之外,PSRec 的识别器采用第三章提到的添加旁路 BiLSTM 模块和 MLP 的方式进行微调,增强模块中包含 3 个 SRB,训练的超分辨率损失函数权重为  $\lambda_1 = 20, \lambda_2 = 1 \times 10^{-4}$ ,整个模型使用不确定性加权的方式进行损失之间的比例调控,权重参数初始值为  $\sigma_r = \sigma_s = \frac{\sqrt{2}}{2} \approx 0.7$ ,权重的惩罚值为  $\alpha = 0.1$ ,此时两个任务的权重为 1。

### 4.3 模型性能

本节主要将本模型 PSRec 与 TSRN [5] 的超分性能和 CRNN [8] 的识别性能以及它们串行连接组成的模型进行对比。在本实验中主要使用第二章提到的 TextZoom [5] 数据集衡量模型性能,并通过本章第一节提到的指标多方面评估模型效果。本节所使用的 TextZoom 测试集分为三种难度,如表 4.1 所示,Easy (E) 表示简单难度,Medium (M) 表示中等难度,Hard (H) 表示较困难难度,Avg (A) 表示在三种难度测试集上的加权平均,权值由三种难度数据集的数量给出,分别为[0.3699, 0.3227, 0.3074]。

表 4.1 TextZoom 数据集上三种难度测试集的 PSNR/SSIM 指标对比。其中  $L_2$  表示 MSE 均方误差损失,  $L_{GP}$  表示第三章提到的梯度剖面损失。二者的权重分别为  $\lambda_1 = 20, \lambda_2 = 1 \times 10^{-4}$ 。由于只比较超分辨率任务, 该表格省略了 PSRec 模型损失函数中的识别损失  $L_r = L_{CTC}$ 。双三次插值不需要损失函数进行训练。

模型	损失函数	PSNR				SSIM			
		E	M	H	A	E	M	H	A
双三次插值	无	-	-	-	-	-	-	-	-
TSRN [5]	$L_2$	22.33	18.37	19.46	20.17	0.8441	0.6366	0.7046	0.7343
TSRN [5]	$L_2 + L_{GP}$	23.27	18.67	19.84	20.73	<b>0.8605</b>	<b>0.6597</b>	<b>0.7284</b>	<b>0.7551</b>
PSRec(Ours)	$L_2$	23.50	19.86	20.59	21.43	0.7817	0.5241	0.6172	0.6480
PSRec(Ours)	$L_2 + L_{GP}$	<b>24.05</b>	<b>20.01</b>	<b>20.87</b>	<b>21.77</b>	0.8049	0.5377	0.6334	0.6660

在超分辨率方面, 本文的 PSRec 模型的超分辨率任务分别使用公式 (3.9) 提到的  $L_s = \lambda_1 L_2 + \lambda_2 L_{GP}$  超分损失和单独的均方误差损失  $L_s = L_2$  进行训练, 并与相同损失训练的 TSRN [5] 模型进行 TextZoom 测试集中低分辨率图像重建效果的对比, 衡量指标选择 PSNR 和 SSIM。

由表 4.1 可以看出, 本文搭建的模型 PSRec 使用与 TSRN 相同的如第三章所示的超分损失  $L_s = \lambda_1 L_2 + \lambda_2 L_{GP}$  进行训练后, PSRec 在 PSNR 上的指标超过了 TSRN, 表明本文设计的模型对于低分辨率图像的重建效果高于 TSRN。与 TSRN 模型不同的是, PSRec 在训练时超分器时同时训练了识别器, 而它们共用一个编码器特征, 因此编码器可以提供部分的识别信息传递给超分器, 对超分器性能有一定的提升。而相比于只使用  $L_s = L_2$  损失进行训练的模型, 在损失函数中添加梯度剖面损失可以增强模型的超分辨率性能。这是由于梯度剖面损失提供了图像的锐度信息, 将其作为损失函数一部分可以在一定程度上强化图像的边缘特征, 因此用含有该信息的损失函数训练的模型也能获得更高的 PSNR。

准确率方面, 本文的 PSRec 模型使用 CTC 损失进行训练, 并用多任务权重调整损失比例。其中识别器分别采用不同的微调策略: (1) **dense**: 表示重新训练最后一层分类头, (2) **mlp**: 表示将最后一层分类头更换为多层感知机并重新训练, (3) **rnn**: 在原有 BiLSTM 基础上增加 BiLSTM 旁路, 其结构与原有 BiLSTM 一致, 输出采用原有结构和旁路结构

表 4.2 TextZoom 数据集上三种难度测试集的识别准确率。模型后的“-”表示使用的微调策略,若无,则表示不使用任何微调策略,直接在测试集上进行推理。

模型和微调策略	Easy	Medium	Hard	Avg
CRNN	0.2110	0.1284	0.1236	0.1575
CRNN-dense	0.5860	0.4525	0.3150	0.4596
CRNN-mlp	0.4158	0.2957	0.2137	0.3149
CRNN-rnn	0.5990	0.4589	0.3142	0.4662
CRNN-rnn-mlp	0.5842	0.4631	0.3105	0.4610
TSRN+CRNN-dense	-	-	-	-
TSRN+CRNN-mlp	-	-	-	-
TSRN+CRNN-rnn	-	-	-	-
TSRN+CRNN-rnn-mlp	-	-	-	-
PSRec-dense(Ours)	0.7079	0.5532	0.3879	0.5596
PSRec-mlp(Ours)	0.6906	0.5418	0.3939	0.5514
PSRec-rnn(Ours)	0.7004	0.5631	0.3842	0.5589
PSRec-rnn-mlp(Ours)	0.7172	0.5447	0.3887	0.5605

输出的平均值, (4) rnn-mlp: 结合 (2) 和 (3) 的方式。作为对比, 本文同时在 TextZoom 训练集上微调了 CRNN 模型以及 TSRN 与 CRNN 的串行连接模型, 除此之外本文还使用了不经过任何微调的 CRNN 模型在测试集上进行 zero-shot 推理, 并将该推理结果作为各个模型的比较基准。表 4.2 展现了本文并行连接的模型 PSRec 与 CRNN 以及串行连接的 TSRN+CRNN 在 TextZoom 测试集上比较的结果。

由表 4.2 可以看出, 不进行微调的 CRNN 在各个测试集上进行推理的效果远远低于微调后的效果, 表明本文所使用的微调方式可以带来性能提升。其中在原有模型的基础上增加 BiLSTM 旁路和多层感知机 MLP 的微调效果最好。对于串行方式连接的模型 TSRN+CRNN, 它先将输入的低分辨率图像进行超分辨率处理, 得到超分辨率图像, 再使用超分辨率图像对 CRNN 识别器进行微调。由表 4.2 所示, 这种方式在测试集的准确率高出使用 CRNN 进行识别的准确率, 表明超分辨率模型的结果能对识别器性能带来提升。但通过串行方式连接超分器和识别器的识别准确率低于本文并行连接的模型 PSRec, 表明串行模型处理多任务时, 识别器会受到超分器输出的影响, 不同任务的耦合度较大,

表 4.3 串行连接的模型 TSRN+CRNN 和本文并行连接的模型在单张图像推理时间 FPS、参数量和浮点运算数上的比较。其中微调策略都为 rnn-mlp。

模型	FPS(Hz)	Params(M)	FLOPs(G)
TSRN+CRNN	-	-	-
PSRec(Ours)	108.87	11.95	1.88

而本文搭建的模型 PSRec 实现的是多任务并行架构，识别器与超分器共用编码器特征，并分别进入不同的解码器进行处理，在解码过程中识别器不会受到超分器最后结果的影响，两个任务的耦合度较低，因而识别效果高于前两种方式。

对于串行连接的多任务模型和并行连接的多任务模型，在模型规模、推理速度上也存在一定的差异。如表 4.3 所示，在 FPS、参数量和浮点运算数方面，同等微调策略下，并行连接的模型 PSRec 由于不同任务可以共用编码器特征，因此本文模型的参数量少于同等配置的 TSRN+CRNN 串行模型；与此同时，对于串行连接的模型架构而言，模型的结构较深，因此数据流的传播速度较慢，推理速度较慢，而对于并行连接的模型而言，多个任务的推理过程在编码器可以同时进行，推理时间少于串行连接的模型。

#### 4.4 消融实验

本节主要对本文搭建的 PSRec 模型进行模块层面和训练方式层面的消融实验。在特征增强模块中，本节将讨论 SRB 模块的个数对图像重建效果和识别效果的影响。在训练方式中，本节将讨论超分损失的组成方式以及是否使用多任务损失权重对模型在 TextZoom 测试集上性能的影响。

**SRB 数目的影响。**特征增强模块中 SRB 个数对模型推理性能有一定的影响。SRB 是特征增强模块的主要组成部分，而特征增强模块的输出特征会被识别器和超分器共享，不仅需要加强编码器的输出特征，还需要进一步融合文本特征和超分特征，因此有必要探究 SRB 模块对整个模型的性能的影响。主要的比较指标包括 TextZoom 测试集上体现对图像的重建能力的 PSNR 和 SSIM，以及识别器的识别准确率，在主要指标之外，本节还讨论了使用不同数量 SRB 的 PSRec 模型的单位参数准确率和 PSNR，以此作为模型的选择标准。不同数量的 SRB 模块对整个模型性能的影响结果如表 4.4 和表 4.5 所示，其中表 4.4 展现了模型的超分辨率性能，而表 4.5 展现了模型的识别性能。

表 4.4 使用不同数目的 SRB ( $N_{\text{SRB}}$ ) 对 PSRec 模型在 TextZoom 测试集上超分辨率性能的影响。其中 E 表示 Easy 简单测试集, M 表示 Medium 中等测试集, H 表示 Hard 困难测试集, A 表示三个测试集的加权平均。

$N_{\text{SRB}}$	PSNR				SSIM			
	E	M	H	A	E	M	H	A
0	23.50	19.86	20.59	21.43	0.7817	0.5241	0.6172	0.6480
1	23.85	19.85	20.81	21.62	0.7916	0.5271	0.6207	0.6537
2	23.52	19.97	20.53	21.45	0.7748	0.5183	0.6060	0.6401
3	24.05	<b>20.01</b>	20.87	21.77	0.8049	0.5377	0.6334	0.6660
4	<b>24.29</b>	19.93	<b>20.98</b>	<b>21.86</b>	<b>0.8161</b>	<b>0.5414</b>	<b>0.6397</b>	<b>0.6732</b>

表 4.5 使用不同数目的 SRB ( $N_{\text{SRB}}$ ) 对 PSRec 模型在 TextZoom 测试集上识别准确率的影响。其中 E 表示 Easy 简单测试集, M 表示 Medium 中等测试集, H 表示 Hard 困难测试集, A 表示三个测试集的加权平均。

$N_{\text{SRB}}$	E	M	H	A
0	0.6677	0.5106	0.3768	0.5276
1	0.6485	0.5355	0.3708	0.5227
2	0.6547	0.5050	0.3604	0.5159
3	<b>0.7172</b>	0.5447	<b>0.3887</b>	<b>0.5605</b>
4	0.6974	<b>0.5532</b>	0.3797	0.5532

如表 4.4 所示, 总体而言, SRB 的数目越多, 模型的超分性能越好。在特征增强模块中使用 4 个 SRB 的效果最好, 在 Easy 测试集上的 PSNR 达到了 24.29, 并且三个测试集的平均 PSNR 也更加接近 22.0。但对于使用 3 个 SRB 模块的模型而言, 其性能与使用更多的 SRB 模块差距并不大, 甚至在 Medium 测试集上的 PSNR 超过了 4 个 SRB 的模型, 表明在一定的范围内, 增加 SRB 的数量可以增强模型的超分性能。而仅使用编码器进行特征编码时的 PSNR 最低, 但总体而言相差不大, 例如其在 Easy 测试集上的 PSNR 比使用 4 个 SRB 模块的模型只低了 0.79, 表明使用高分辨率降采样图像预训练的参数初始化模型, 后续在低分辨率图像上进行微调的方式可以带来较高的模型鲁棒性。

而在准确率方面，结论与超分任务的结论略有不同。表 4.5 展现了使用不同数目的 SRB 对模型在不同测试集上的识别准确率的影响。从表中可以看出，在 PSRec 的特征增强模块中使用 4 个 SRB 时的识别准确率低于使用 3 个 SRB，表明增加 SRB 在一定范围内对于识别任务的性能有一定的提升，但过多的 SRB 会造成性能的饱和。

对于拥有多任务评估结果的 PSRec 模型，本文还探究了增加 SRB 个数在模型的参数量以及运算能力的影响。如表 4.6 所示，在特征增强模块上使用 3 个 SRB 能够带来最高的单位参数准确率，即平均每 1M 参数可以贡献 0.047 准确率。而对于 PSNR 而言，特征增强模块中不使用任何 SRB 模块可以达到最高的单位参数 PSNR，平均每 1M 参数可以提升 1.8492 的 PSNR 指标，但由表 4.4 和表 4.5 发现，不使用任何 SRB 模块的模型的整体性能远低于使用 3 个 SRB 的模型性能，因此本文最终选择增强模块中带有 3 个 SRB 的模型进行实验。

表 4.6 使用不同数目的 SRB 模块对模型规模和推理速度的影响。

$N_{\text{SRB}}$	FPS(Hz)	Params(M)	FLOPs(G)	PSNR/Params(M)	ACC/Params(M)
0	186.85	11.59	1.50	<b>1.8492</b>	0.0458
1	153.13	11.71	1.63	1.8466	0.0465
2	125.95	11.83	1.75	1.8134	0.0456
3	108.87	11.95	1.88	1.8216	<b>0.0470</b>
4	98.43	12.07	2.00	1.8114	0.0464

**梯度剖面损失。**超分辨率任务使用不同权重的梯度剖面损失能够给模型性能带来不同程度的影响。表 4.7 展示了使用和不使用梯度剖面损失以及不同权重的梯度剖面损失对模型在 TextZoom 测试集上的性能影响。从表中可以发现，当  $\lambda_2 = 1 \times 10^{-4}$  时，训练的模型性能相比于其它权重更好。

为了更加具体地展现梯度剖面损失对图像超分辨率重建的效果，@展示了对于同一张文本图像的超分辨率重建效果。



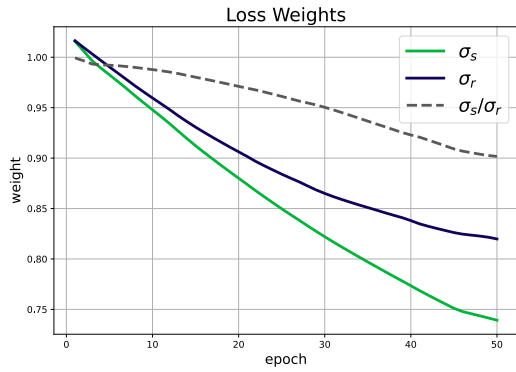
表 4.8 使用不同权重 $\lambda_2$ 的梯度剖面损失 $L_{GP}$ 对模型超分辨率性能的影响。其中超分损失中 $L_2$ 的权重为固定为20。E 表示 Easy 简单测试集，M 表示 Medium 中等测试集，H 表示 Hard 困难测试集，A 表示三个测试集的加权平均。

$\lambda_2(\times 10^{-4})$	PSNR				SSIM			
	E	M	H	A	E	M	H	A
0	21.46	19.26	19.57	20.17	0.7301	0.4960	0.5661	0.6041
0.5	23.86	<b>20.04</b>	20.84	21.70	0.8041	0.5373	<b>0.6344</b>	0.6658
1	<b>24.05</b>	20.01	<b>20.87</b>	<b>21.77</b>	<b>0.8049</b>	<b>0.5377</b>	0.6334	<b>0.6660</b>
5	23.81	19.98	<b>20.87</b>	21.67	0.8009	0.5332	0.6325	0.6627

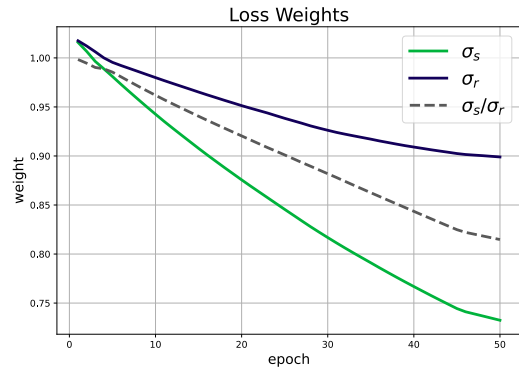
表 4.8 使用不同权重 $\lambda_2$ 的梯度剖面损失 $L_{GP}$ 对模型识别准确率的影响。其中超分损失中 $L_2$ 的权重为固定为20。E 表示 Easy 简单测试集，M 表示 Medium 中等测试集，H 表示 Hard 困难测试集，A 表示三个测试集的加权平均。

$\lambda_2(\times 10^{-4})$	E	M	H	A
0	0.0006	0.0	0.0	0.0002
0.5	0.6714	0.5440	0.3872	0.5429
1	<b>0.7172</b>	0.5447	0.3753	<b>0.5605</b>
5	0.6980	<b>0.5496</b>	<b>0.3902</b>	0.5555

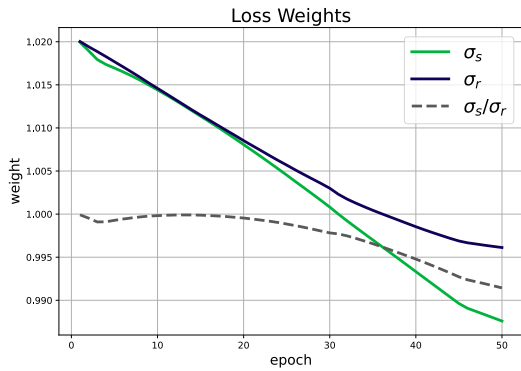
**不确定性权重。**在多任务学习中，使用可学习的权重参数训练模型可以给模型带来一定的性能提升。如表 4.9 所示，在训练使用多任务损失函数训练模型时，将损失函数的权重设置为可学习参数的一部分，可以在一定程度上提高模型的超分辨率和识别性能。其中，使用了可学习损失权重的 PSRec 模型在 PSNR 上达到了最高水平，即21.77，并由表 4.1 所示，使用该损失训练的 PSRec 超过了 TSRN 模型的超分辨率性能。在准确率方面，使用可学习的权重训练的模型的平均识别准确率，比未使用可学习权重训练的模型高了2%。因此可以看出对于多任务学习，不同任务之间的损失权重设置会对模型性能带来一定影响。



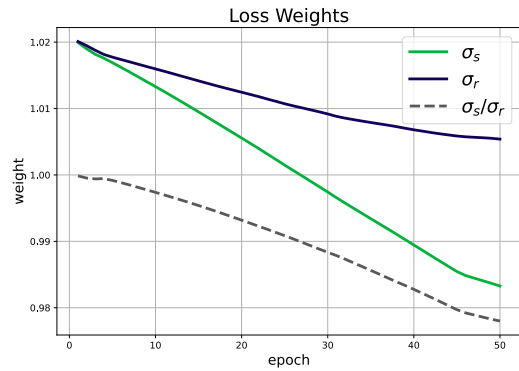
(a) 使用 dense 方式进行微调的模型



(b) 使用 mlp 方式进行微调的模型



(c) 使用 rnn 方式进行微调的模型



(d) 使用 rnn-mlp 方式微调的模型

图 4-2 前 50 个 epoch 中不确定性加权对不同微调方式模型的影响。其中绿色线条表示超分辨率任务损失权重中的参数  $\sigma_s$ ，紫色线条表示识别任务损失权重中的参数  $\sigma_r$ ，灰色虚线表示两个参数的比值。

表 4.9 使用可学习的损失权重对模型性能的影响。三个指标分别表示在 TextZoom 的三个测试集上的加权平均值。

是否使用可学习的损失权重	PSNR	SSIM	ACC
×	21.52	0.6628	0.5397
✓	<b>21.77</b>	<b>0.6660</b>	<b>0.5605</b>

为了展示不确定性权重对训练过程中实际的调控过程，图 4-2 展示了不同微调方式的模型在使用了不确定性加权后的各个任务对应损失函数的权重，以及任务之间的比值随着训练过程的变化。值得注意的是，图中展示的曲线并非实际的损失权重，其关系式由公式 (3.11) 给出。总体而言，二者的权重参数  $\sigma_s, \sigma_r$  呈现下降趋势，表明随着训练的进

行，超分辨率任务的损失和识别任务的损失分别在下降，因而需要加大权重。对于由灰色虚线所代表的权重参数比值而言，表明随着训练的进行，两个任务的权重并非一成不变，而是需要不断地调整。由图中可以看出，识别损失的权重参数大于超分损失，因而其权重小于超分损失，而这是可学习的权重参数在模型训练时，根据识别损失的值大于超分损失的值自行调控的结果。

## 4.5 推理结果

本节主要展示不同模型的推理可视化结果，并分析不同模型在图像超分辨率和识别任务上结果的异同。

## **第五章 总结与展望**

本文主要搭建了一种并行结构的多任务模型。

## 参考文献

- [1] 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.
- [2] KARATZAS D, SHAFAIT F, UCHIDA S, et al. ICDAR 2013 robust reading competition[C]//2013 12th international conference on document analysis and recognition. 2013: 1484-1493.
- [3] VEIT A, MATERA T, NEUMANN L, et al. COCO-Text: Dataset and Benchmark for Text Detection and Recognition in Natural Images[Z]. 2016.
- [4] SHI B, YANG M, WANG X, et al. Aster: An attentional scene text recognizer with flexible rectification[J]. IEEE transactions on pattern analysis and machine intelligence, 2018, 41(9): 2035-2048.
- [5] WANG W, XIE E, LIU X, et al. Scene text image super-resolution in the wild[C]//Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16. 2020: 650-666.
- [6] DU Y, CHEN Z, JIA C, et al. SVTR: Scene Text Recognition with a Single Visual Model[Z]. 2022.
- [7] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale[Z]. 2021.
- [8] 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[Z]. 2015.
- [9] HOCHREITER S, SCHMIDHUBER J. Long Short-Term Memory[J/OL]. Neural Comput., 1997, 9(8): 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>. DOI:10.1162/neco.1997.9.8.1735.
- [10] GRAVES A, FERNÁNDEZ S, GOMEZ F, et al. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks[C/OL]//Proceedings of the 23rd International Conference on Machine Learning. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 2006: 369-376. <https://doi.org/10.1145/1143844.1143891>. DOI:10.1145/1143844.1143891.
- [11] CHO K, VAN MERRIËNBOER B, GULCEHRE C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.
- [12] BAUTISTA D, ATIENZA R. Scene text recognition with permuted autoregressive sequence models[C]//European conference on computer vision. 2022: 178-196.
- [13] WANG P, DA C, YAO C. Multi-granularity prediction for scene text recognition[C]//European Conference on Computer Vision. 2022: 339-355.
- [14] FU Z, XIE H, JIN G, et al. Look back again: Dual parallel attention network for accurate and robust scene text recognition[C]//Proceedings of the 2021 International Conference on Multimedia Retrieval. 2021: 638-644.

- [15] DEVLIN J, CHANG M W, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
- [16] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
- [17] RADFORD A, NARASIMHAN K, SALIMANS T, et al. Improving language understanding by generative pre-training[J]. 2018.
- [18] BROWN T, MANN B, RYDER N, et al. Language models are few-shot learners[J]. Advances in neural information processing systems, 2020, 33: 1877-1901.
- [19] RADFORD A, KIM J W, HALLACY C, et al. Learning transferable visual models from natural language supervision[C]//International conference on machine learning. 2021: 8748-8763.
- [20] ZHAO S, WANG X, ZHU L, et al. Clip4str: A simple baseline for scene text recognition with pre-trained vision-language model[J]. arXiv preprint arXiv:2305.14014, 2023.
- [21] FANG S, XIE H, WANG Y, et al. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 7098-7107.
- [22] SHI W, CABALLERO J, HUSZÁR F, et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 1874-1883.
- [23] MA J, LIANG Z, ZHANG L. A text attention network for spatial deformation robust scene text image super-resolution[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 5911-5920.
- [24] MA J, GUO S, ZHANG L. Text Prior Guided Scene Text Image Super-Resolution[J/OL]. IEEE Transactions on Image Processing, 2023, 32: 1341-1353. DOI:10.1109/TIP.2023.3237002.
- [25] ZHU S, ZHAO Z, FANG P, et al. Improving scene text image super-resolution via dual prior modulation network[C]//Proceedings of the AAAI Conference on Artificial Intelligence: Vol. 37. 2023: 3843-3851.
- [26] ZHAO C, FENG S, ZHAO B N, et al. Scene Text Image Super-Resolution via Parallely Contextual Attention Network[C/OL]//Proceedings of the 29th ACM International Conference on Multimedia. Virtual Event, China: Association for Computing Machinery, 2021: 2908-2917. <https://doi.org/10.1145/3474085.3475469>. DOI:10.1145/3474085.3475469.
- [27] CHEN J, YU H, MA J, et al. Text gestalt: Stroke-aware scene text image super-resolution[C]//Proceedings of the AAAI Conference on Artificial Intelligence: Vol. 36. 2022: 285-293.

- [28] ZHAO M, WANG M, BAI F, et al. C3-stsr: Scene text image super-resolution with triple clues[J]. arXiv preprint arXiv:2204.14044, 2022.
- [29] HO J, JAIN A, ABBEEL P. Denoising diffusion probabilistic models[J]. Advances in neural information processing systems, 2020, 33: 6840-6851.
- [30] SAHARIA C, HO J, CHAN W, et al. Image super-resolution via iterative refinement[J]. IEEE transactions on pattern analysis and machine intelligence, 2022, 45(4): 4713-4726.
- [31] LIU B, YANG Z, WANG P, et al. TextDiff: Mask-Guided Residual Diffusion Models for Scene Text Image Super-Resolution[J]. arXiv preprint arXiv:2308.06743, 2023.
- [32] 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: 1-20.
- [33] GUPTA A, VEDALDI A, ZISSERMAN A. Synthetic data for text localisation in natural images[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2315-2324.
- [34] JI X, CAO Y, TAI Y, et al. Real-World Super-Resolution via Kernel Estimation and Noise Injection[C/OL]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW): Vol. 0. 2020: 1914-1923. DOI:10.1109/CVPRW50498.2020.00241.
- [35] ZHANG X, CHEN Q, NG R, et al. Zoom to learn, learn to zoom[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 3762-3770.
- [36] JADERBERG M, SIMONYAN K, ZISSERMAN A, et al. Spatial transformer networks[J]. Advances in neural information processing systems, 2015, 28.
- [37] BOOKSTEIN F L. Principal warps: Thin-plate splines and the decomposition of deformations[J]. IEEE Transactions on pattern analysis and machine intelligence, 1989, 11(6): 567-585.
- [38] MISRA D. Mish: A self regularized non-monotonic activation function[J]. arXiv preprint arXiv:1908.08681, 2019.
- [39] KENDALL A, GAL Y, CIPOLLA R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7482-7491.
- [40] SUN J, XU Z, SHUM H Y. Image super-resolution using gradient profile prior[C]//2008 IEEE conference on computer vision and pattern recognition. 2008: 1-8.
- [41] WANG Z, BOVIK A C, SHEIKH H R, et al. Image quality assessment: from error visibility to structural similarity[J]. IEEE transactions on image processing, 2004, 13(4): 600-612.
- [42] LOSHCHILOV I, HUTTER F. Decoupled weight decay regularization[J]. arXiv preprint arXiv:1711.05101, 2017.

- [43] LIEBEL L, KÖRNER M. Auxiliary tasks in multi-task learning[J]. arXiv preprint arXiv:1805.06334, 2018.



## 附录 A 损失函数

### A.1 识别损失

设输入 CTC 损失的特征图  $F$  的宽为  $T$ （即时间序列长度），目标单词表大小为  $S_c$ ，则分类的类别一共有  $S = S_c + 1$  类（包括空白字符  $\varepsilon$ ）。CTC 损失的标签对齐方式为：对于特征图  $F \in \mathbb{R}^{T \times S}$ ，按照时间步顺序依次预测字符，若预测出的字符长度满足空白字符个数与单词表字符个数和为  $T$ ，且单词表字符必须大于 0，即

$$T_\varepsilon + T_c = T, T_\varepsilon \geq 0, T_c > 0 \quad (\text{A.1})$$

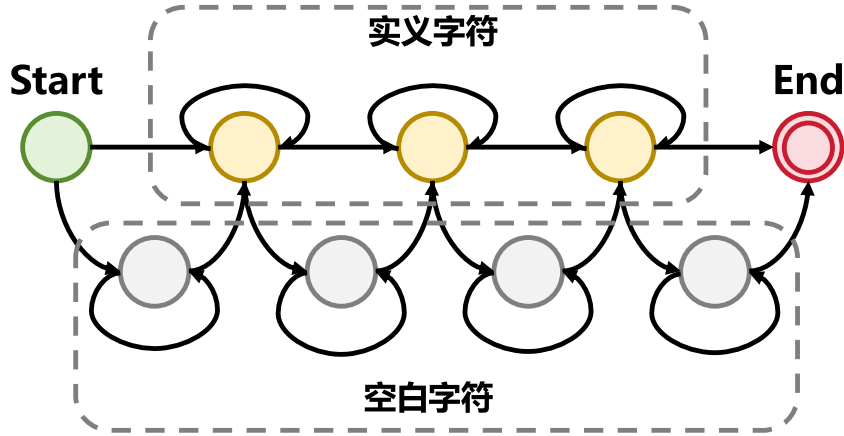


图 A-1 CTC 损失对应的隐马尔可夫链

则该预测标签为一个合法的对齐方式，设所有合法的对齐标签为集合  $\mathcal{S}_a$ 。CTC 损失满足如图 A-1 所示的隐马尔可夫过程：包括空白字符在内的字符表示一个隐状态，约束条件由公式 (A.1) 给出。设从预测开始，到时间步  $i$  时，已经产生  $j$  个单词表字符的概率为前向概率  $\alpha_{i,j}$ ，则由马尔可夫性，

$$\alpha_{i,j} = \alpha_{i-1,j} p_{i-1,j}(\varepsilon) + \alpha_{i,j-1} p_{i,j-1}(c_j) \quad (\text{A.2})$$

其中  $p_{i,j-1}(c_j)$  表示在时间步为  $i$  预测为第  $j$  个单词表字符的概率， $p_{i-1,j}(\varepsilon)$  表示在时间步为  $i-1$  预测出空白字符的概率，显然由递归关系可以得到公式 (A.2)。同理，设从时间步  $i$ ，位于第  $j$  个单词表字符开始，从  $i+1$  到  $T$  会产生合法对齐标签的概率为后向概率  $\beta_{i,j}$ ，则由马尔可夫性，

$$\beta_{i,j} = \beta_{i+1,j} p_{i+1,j}(\varepsilon) + \beta_{i,j+1} p_{i,j+1}(c_{j+1}) \quad (\text{A.3})$$

因此给定输入特征，预测标签的概率为

$$p_{\theta}(L_p | F) = \sum_{a \in \mathcal{S}_a} p_{\theta}(a | F) \quad (\text{A.4})$$

其中  $p_{\theta}(a | F) = \prod_{p \in \text{path}} p$  表示一个合法对齐标签路径上所有节点概率的乘积，整个标签预测的概率即所有合法路径概率的和。每一个节点概率  $p \in \text{path}$  是由模型网络产生的。以在时间步  $i$  已经预测出  $j$  个字符，且下一个预测字符为  $c$  的节点概率  $p_{i,j}(c)$  为例，在反向传播时，由链式法则得到

$$\frac{\partial p_{\theta}(L_p | F)}{\partial \theta} = \sum_{a \in \mathcal{S}_a} \frac{\partial p_{i,j}(c)}{\partial \theta} \frac{\partial p_{\theta}(L_p | F)}{\partial p_{i,j}(c)} \quad (\text{A.5})$$

$\frac{\partial p_{i,j}(c)}{\partial \theta}$  与整个模型有关，后一项  $\frac{\partial p_{\theta}(L_p | F)}{\partial p_{i,j}(c)}$  由 CTC 损失路径确定。将所有路径  $\mathcal{S}_a$  分为包含节点  $p_{i,j}(c)$  的部分  $\mathcal{S}_a^i$  和不包含节点  $p_{i,j}(c)$  的部分  $\mathcal{S}_a^e$ ，则对  $p_{i,j}(c)$  求偏导后，与其无关的路径导数为 0，其余部分可以写为

$$\frac{\partial p_{\theta}(L_p | F)}{\partial p_{i,j}(c)} = \sum_{a \in \mathcal{S}_a^i} \frac{p_{\theta}(a | F)}{p_{i,j}(c)} = \frac{1}{p_{i,j}(c)} \sum_{a \in \mathcal{S}_a^i} p_{\theta}(a | F) \quad (\text{A.6})$$

求和号中的式子表示在时间步  $i$  时，已经产生  $j$  个单词表字符、且在当前状态产生下一个单词表字符为  $c$ 、且从  $i+1$  到  $T$  将产生  $T-i$  会产生合法对齐标签的概率，即在时间步为  $i$ 、已经产生  $j$  个单词表字符的前向概率、转移概率和后向概率之积：

$$\frac{1}{p_{i,j}(c)} \sum_{a \in \mathcal{S}_a^i} (p_{\theta}(a | F)) = \frac{1}{p_{i,j}(c)} \alpha_{i,j} p_{i,j}(c) \beta_{i,j+1} = \alpha_{i,j} \beta_{i,j+1} \quad (\text{A.7})$$

即最终 CTC 损失反向传播时更新的梯度为

$$g_t = \frac{\partial P_{\theta_t}(\hat{Y} | X)}{\partial \theta_t} = \sum_{i,j,c} \frac{\partial p_{i,j}(c)}{\partial \theta_t} \alpha_{i,j} \beta_{i,j+1} \quad (\text{A.8})$$

## A.2 多任务损失

一般而言，识别任务为分类任务，超分任务为回归任务，其似然函数可以分别表示为

$$p_r(\mathcal{F}_{\theta}(I_l) = c | I_l) = \text{softmax} \left( \frac{1}{\sigma_r^2} \mathcal{F}_{\theta}(I_l) \right) \quad (\text{A.9})$$

$$p_s(I_s | I_l, I_h) = \mathcal{N}(\mathcal{F}_{\theta}(I_l), \sigma_s)$$

其中  $\sigma_r > 0, \sigma_s > 0$  分别表示识别任务和超分辨率任务中的不确定性（识别任务表达式类似于玻尔兹曼分布，则该不确定性参数表示分布中的温度；而超分任务表达式中不确定性参数表示方差）。模型训练的目的即通过优化参数  $\theta$  最大化对数似然分布

$$\begin{aligned}\theta^* &= \operatorname{argmax}_{\theta} \log p(c, I_s \mid I_l, I_h) \\ &= \operatorname{argmax}_{\theta} \log p_r(c \mid I_l) + \log p_s(I_s \mid I_l)\end{aligned}\quad (\text{A.10})$$

其中  $p, p_r, p_s$  都是带有模型参数  $\theta$  的似然函数。对于识别任务  $\log p_r(c \mid I_l)$ ，由公式 (A.9) 可以写为

$$\begin{aligned}\log p_r(\mathcal{F}_{\theta}(I_l) = c \mid I_l) &= \log \operatorname{softmax} \left( \frac{1}{\sigma_r^2} \mathcal{F}_{\theta}(I_l) \right) \\ &= \log \frac{\exp \left( \frac{1}{\sigma_r^2} \mathcal{F}_{\theta}(I_l) \right)}{\sum_c \exp \left( \frac{1}{\sigma_r^2} \mathcal{F}_{\theta}(I_l) \right)} \\ &= \frac{1}{\sigma_r^2} \mathcal{F}_{\theta}(I_l) - \log \sum_c \exp \left( \frac{1}{\sigma_r^2} \mathcal{F}_{\theta}(I_l) \right)\end{aligned}\quad (\text{A.11})$$

而对于超分任务  $\log p_s(I_s \mid I_l)$  可以写为

$$\begin{aligned}\log p_s(I_s \mid I_l, I_h) &= \log \mathcal{N}(\mathcal{F}_{\theta}(I_l), \sigma_s) \\ &\propto -\frac{1}{2\sigma_s^2} \|I_h - \mathcal{F}_{\theta}(I_l)\|^2 - \log \sigma_s\end{aligned}\quad (\text{A.12})$$

损失函数定义为最小化形式，可以表示为（近似过程参考文献<sup>[39]</sup>）

$$\begin{aligned}\mathcal{L} &= -\log p(c, I_s \mid I_l, I_h) \\ &= \frac{1}{2\sigma_s^2} \|I_h - \mathcal{F}_{\theta}(I_l)\|^2 + \log \sigma_s - \frac{1}{\sigma_r^2} \mathcal{F}_{\theta}(I_l) + \log \sum_c \exp \left( \frac{1}{\sigma_r^2} \mathcal{F}_{\theta}(I_l) \right) \\ &\approx \frac{1}{2\sigma_s^2} \mathcal{L}_s + \frac{1}{\sigma_r^2} \mathcal{L}_r + \log \sigma_s + \log \sigma_r\end{aligned}\quad (\text{A.13})$$

为了防止正则项出现负数，同时方便参数学习，可以采用如下方式<sup>[43]</sup>进行损失的计算

$$\mathcal{L} = \frac{1}{2\sigma_s^2} \mathcal{L}_s + \frac{1}{2\sigma_r^2} \mathcal{L}_r + \log(1 + \sigma_s^2) + \log(1 + \sigma_r^2)\quad (\text{A.14})$$

本文模型最终的多任务损失函数即由 公式 (A.14) 给出。其中  $\sigma_r, \sigma_s$  可以在训练前指定特定值，也将其加入模型中的可学习的参数中，随着模型参数一同更新，实现动态调整梯度传播时的权重。

## 附录 B 模型性能

### B.1 最优模型

本文实现的 PSRec 模型为多任务模型，因此对于每个任务而言，在模型训练时都分别保存不同任务的最优模型，即最优准确率模型和最优 PSNR 模型。在正文中，若无特殊说明，一般使用的是最优 PSNR 模型。本附录将进行这两类最优模型之间的对比。

表 B.1 和表 B.2 展示了用固定权重的损失公式 (3.10) 训练的模型分别使用最优准确率参数和最优 PSNR 参数的推理性能。而表 B.3 和表 B.4 展示了用不确定性权重参数公式 (3.11) 训练的模型分别使用最优准确率和最优 PSNR 参数的推理性能。

表 B.1 使用固定加权损失方式训练的模型在 TextZoom 数据集上三种难度测试集的超分辨率性能。模型后的“-”表示使用的微调策略。参数类型中“A”表示使用最优准确率模型参数，“P”表示使用最优 PSNR 模型参数

微调策略	参数	PSNR				SSIM			
		E	M	H	A	E	M	H	A
PSRec-dense	A	22.86	19.31	20.66	21.04	0.8078	0.5359	0.6372	0.6677
PSRec-mlp	A	23.69	19.97	20.63	21.55	0.7963	0.5318	0.6195	0.6566
PSRec-rnn	A	23.42	19.49	20.76	21.34	0.8100	0.5339	0.6369	0.6676
PSRec-rnn-mlp	A	22.89	19.66	20.57	21.14	0.7866	0.5272	0.6122	0.6493
PSRec-dense	P	23.40	19.65	20.72	21.36	0.8004	0.5333	0.6289	0.6615
PSRec-mlp	P	23.98	19.92	20.93	21.73	0.8101	0.5385	0.6370	0.6692
PSRec-rnn	P	23.41	19.61	20.79	21.38	0.8086	0.5362	0.6374	0.6680
PSRec-rnn-mlp	P	23.66	19.78	20.77	21.52	0.8029	0.5336	0.6297	0.6628

表 B.2 使用固定加权损失方式训练的模型在 TextZoom 数据集上三种难度测试集的识别准确率。模型后的“-”表示使用的微调策略。参数类型中“A”表示使用最优准确率模型参数，“P”表示使用最优 PSNR 模型参数

微调策略	参数类型	Easy	Medium	Hard	Avg
PSRec-dense	A	0.7067	0.5446	0.3961	0.5589
PSRec-mlp	A	0.6918	0.5461	0.3961	0.5539
PSRec-rnn	A	0.7054	0.5581	0.3917	0.5615
PSRec-rnn-mlp	A	0.6900	0.5567	0.3969	0.5569
PSRec-dense	P	0.6775	0.5319	0.3968	0.5443
PSRec-mlp	P	0.6825	0.5468	0.3946	0.5502
PSRec-rnn	P	0.6986	0.5426	0.3931	0.5544
PSRec-rnn-mlp	P	0.6770	0.5298	0.3850	0.5397

表 B.3 使用可学习权重损失方式训练的模型在 TextZoom 数据集上三种难度测试集的超分辨率性能。模型后的“-”表示使用的微调策略。参数类型中“A”表示使用最优准确率模型参数，“P”表示使用最优 PSNR 模型参数

		PSNR				SSIM			
微调策略	参数	E	M	H	A	E	M	H	A
PSRec-dense	A	23.56	19.52	20.76	21.39	0.8159	0.5390	0.6431	0.6734
PSRec-mlp	A	23.67	19.82	20.84	21.56	0.7980	0.5350	0.6299	0.6615
PSRec-rnn	A	23.05	19.64	20.78	21.25	0.8043	0.5359	0.6351	0.6657
PSRec-rnn-mlp	A	24.05	20.01	20.87	21.77	0.8049	0.5377	0.6334	0.6660
PSRec-dense	P	23.50	19.70	20.78	21.44	0.8047	0.5382	0.6322	0.6657
PSRec-mlp	P	23.67	19.82	20.84	21.56	0.7980	0.5350	0.6299	0.6615
PSRec-rnn	P	23.54	19.86	20.74	21.49	0.8061	0.5362	0.6328	0.6657
PSRec-rnn-mlp	P	24.05	20.01	20.87	21.77	0.8049	0.5377	0.6334	0.6660

表 B.4 使用使用可学习权重损失方式训练的模型在 TextZoom 数据集上三种难度测试集的识别准确率。模型后的“-”表示使用的微调策略。参数类型中“A”表示使用最优准确率模型参数，“P”表示使用最优 PSNR 模型参数

微调策略	参数类型	Easy	Medium	Hard	Avg
PSRec-dense	A	0.7197	0.5532	0.4028	0.5686
PSRec-mlp	A	0.6906	0.5418	0.3939	0.5514
PSRec-rnn	A	0.7110	0.5667	0.3931	0.5667
PSRec-rnn-mlp	A	0.7172	0.5447	0.3887	0.5605
PSRec-dense	P	0.7079	0.5532	0.3879	0.5596
PSRec-mlp	P	0.6906	0.5418	0.3939	0.5514
PSRec-rnn	P	0.7004	0.5631	0.3842	0.5589
PSRec-rnn-mlp	P	0.7172	0.5447	0.3887	0.5605

## 致 谢

论文写就，立夏已过。若无他助，难成本文。笔者思索，在此为谢。

首先，感谢我的导师薛晖教授，是她带给了我模式识别课程上的启蒙，带领我走进人工智能领域。若没有她的教导，我或许无法完成此文。接着要感谢同组的师兄师姐，学习路上若无你们给我树立的榜样，我或许无法完成此文。此外，特别感谢朱士鹏师兄对本文的悉心指导，并为本文的实验提供了充足的算力支持。另外，感谢吴健雄学院提供的优质资源，为我的求学生活提供了良好的学习环境。感谢健雄书院提供的多彩活动和人文理念，让我成为管理团队的一员，帮助我在心理上获得成长。

其次，感谢本科四年间所有在成长和学习上帮助过我的朋友和老师。感谢辅导员菜菜对我大三大四生活的关照。感谢瓜群和立交桥中的每一位成员，你们为我的本科生活带来了众多难忘的瞬间。感谢十只鸡，Jerry，然然，PC，T 老师等学长学姐为我的成长和大学生活带来诸多快乐和思考，你们永远是我的榜样。感谢 Zan, JenniferWu, Henny, 421, TtuHamg, King, Pepper, SYM, LYL 等朋友为我的本科生活带来了风、故事、大海和岛屿，没有你们的陪伴，我或许无法体验到真正的大学生活。

除此之外，感谢一直努力的自己，在幽暗昏惑之时依然能坚持。

最后，特别感谢一路以来一直陪伴着我的父母，在我缺席本应陪伴他们的无数的时光里，他们一如既往地给予我无限的支持、关心和爱。正是他们对生活的热爱和我对的照顾，让我能幸福成长并顺利完成学业。

感谢大学期间的所有相遇，与一草一木，与一朝一夕。