

# 基于 resnext101 的手写汉字识别

李志远

(南京农业大学人工智能 201)

**摘要：**本文旨在探究在手写汉字识别任务中，使用不同的深度神经网络对性能的影响，并找到最优的网络结构。我在给定的数据集上进行了实验，并使用了常见的神经网络架构，包括 ResNet101、ResNeXt101 和 ResNet152。我们使用 9:1 的训练集-测试集比率进行数据集划分，并在预处理时将所有样本 resize 为 60\*60 大小的图片，并在训练时进行不大于 30 度的随机旋转。实验结果表明，使用 ResNeXt101 网络结构获得了最好的性能表现，准确率达到了 99.2%。在实验过程中，我们还尝试了使用数据增强、调整学习率等方法来优化模型性能，但结果表明这些方法并没有显著提高模型的准确率。综上所述，我们建议在手写汉字识别任务中使用 ResNeXt101 网络结构，以达到最佳性能。

**关键词：**深度学习，残差神经网络，手写汉字识别

**Abstract:**This paper aims to explore the impact of using different deep neural networks on the performance of handwritten Chinese character recognition task, and find the optimal network structure. I performed experiments on the given dataset and used common neural network architectures including ResNet101, ResNeXt101, and ResNet152. We use a 9:1 training-test set ratio for dataset partitioning, and resize all samples to 60\*60 sized images during preprocessing, and randomly rotate them by no more than 30 degrees during training. The experimental results show that the best performance is obtained by using the ResNeXt101 network structure, and the accuracy reaches 99.2%. During the experiment, we also tried using data augmentation, adjusting the learning rate and other methods to optimize the model performance, but the results showed that these methods did not significantly improve the accuracy of the model. In summary, we propose to use the ResNeXt101 network structure in the handwritten Chinese character recognition task to achieve the best performance.

**Keywords:**Deep Learning, Resnet, Chinese Handwriting Recognition

---

## 1 整体思路

在手写汉字识别任务中,神经网络已经被证明是一种有效的工具。我的计划是使用 ResNet101、ResNext101 和 ResNet152 等不同的网络结构进行对比实验,并针对是否使用预训练模型进行了对比实验,以找到最佳性能的网络结构。

实验的数据集已提前给出,图片偏小,因此不适合使用固定 224 大小输入的 AlexNet,该数据集包含了 500 类汉字、每个汉字有 80 张图片。我将数据集划分为了训练集、测试集,以进行模型训练和评估。

在训练神经网络时,我针对我的训练设备进行超参数的设置,例如学习率、批量大小、迭代次数等,以获得最佳性能。此外,考虑到汉字的绝对位置对其分类有较大影响,我仅使用了随机旋转进行数据增强,没有使用裁剪、翻转等手段, resnet 等网络结构本身使用了 batch normalization[4] 以防止过拟合,由于 dropout[6] 与 batch norm 同时使用会影响性能,因此不另外设置 dropout 层。

训练完成后,我对测试集进行了评估,并记录了各个网络结构的性能指标。由于分类数较大,混淆矩阵与 roc 曲线并不直观,因此此处没有使用。

最后,我对实验结果进行了分析和总结,并提出改进方案。例如,你可以尝试其他的网络结构、优化算法或数据增强技巧,以进一步提高手写汉字识别的性能。

## 2 数据集

### 2.1 数据集划分

我将整个数据集按照 9:1 的比率划分为训练集和测试集。训练集用于训练神经网络,测试集用于评估网络的性能。具体做法为将数据重新划分到 train 和 test 文件夹内,使用 imagefolder 类读取。

### 2.2 数据预处理

我将所有图像统一 resize 为了 60x60 像素的大小以使得所有图像具有相同的尺寸,方便输入到神经网络中进行训练和测试。同时,我使用了数据增强技术,在训练时进行不大于 30 度的随机旋转,可以使得神经网络更好地学习汉字的不同书写风格。

## 3 模型

### 3.1 ResNet

ResNet (Residual Network) 是一种深度卷积神经网络,它由何凯明等人于 2015 年提出 [2],并在 ImageNet[1] 图像识别和 COCO[5] 物体检测等任务中取得了非常好的结果。ResNet 主要的创新点是引入了残差连接 (residual connections) 来解决深度神经网络训练时的梯度消失问题。

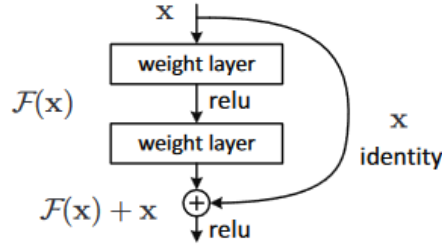


Figure 1: Residual learning:a building block[2]

传统的深度神经网络中，随着网络层数的增加，梯度会逐渐变得很小，导致无法正确地更新网络参数。为了解决这个问题，ResNet 提出了残差学习的思想：即对于每个层，不再学习其直接映射，而是学习其残差映射，即输入与输出之间的差异。这样，网络就可以直接学习输入和输出之间的差异，而不必担心梯度消失的问题。

在 ResNet 中，残差连接被添加到了卷积层中，以使得每个卷积层的输出不仅取决于该层的输入，还取决于该层之前所有层的输入。这种全局信息的融合可以帮助网络更好地学习特征，从而提高模型的准确率。ResNet 中还使用了批量归一化和池化层等技术，以进一步提高网络的性能。

ResNet 的网络结构非常深,可以达到 1000 多层,但其计算量并不比传统的浅层网络大。这使得 ResNet 成为了处理大规模图像数据的首选网络之一。同时，ResNet 的成功也启示了后续研究者提出了更加深入的残差网络，如 ResNeXt、Wide ResNet[8] 等。

### 3.2 ResNext

ResNeXt (Residual Next) 是一种基于 ResNet 的残差网络结构，由微软亚洲研究院的 Xie Saining 等人于 2016 年提出 [7]。ResNeXt 主要的创新点是使用了组卷积（group convolution）来进一步提升网络的性能和可扩展性。

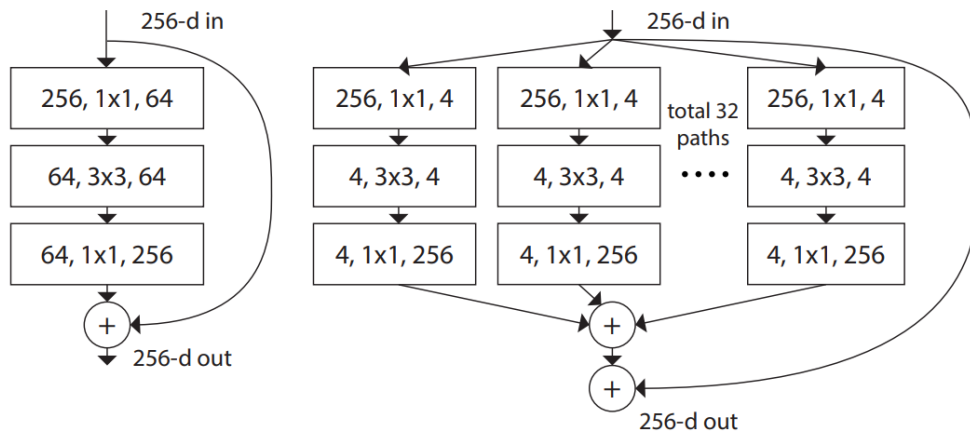


Figure 2: **Right:** A block of ResNeXt with cardinality = 32, with roughly the same complexity. A layer is shown as (# in channels, filter size, # out channels)[7]

组卷积是一种卷积操作，将输入的特征图分成若干个组，每个组之间进行卷积，最后将各组的卷积结

果拼接在一起。在 ResNeXt 中，组卷积被广泛应用于卷积层，可以有效减少参数数量和计算量，并提高了模型的准确率。组卷积的思想类似于 Inception 模块中的分支结构，但 ResNeXt 中的组卷积更加灵活，可以根据具体任务的需求进行组合。

ResNeXt 的结构与 ResNet 类似，但在卷积层中引入了组卷积，使得网络可以更好地学习特征。ResNeXt 还采用了批量归一化和池化层等技术来进一步提高网络的性能。与 ResNet 相比，ResNeXt 的计算量和参数数量都要小，但准确率却有显著的提升。

ResNeXt 的成功启示了后续研究者对残差网络的深入探索，如 DenseNet[3] 等。同时，ResNeXt 也被广泛应用于图像分类、目标检测和语义分割等领域，并在多项比赛中取得了优异的成绩。

## 4 实验

### 4.1 超参数

我针对每个网络都至少训练了 40 个周期，学习率设置为  $1e-3$ ，权重衰退值设置为  $1e-4$ ，批量大小为 64，momentum 为 0.9，其中 resnet101 训练时间约为 2min/epoch，resnext101\_32 $\times$ 4d 约为 4min/epoch。

### 4.2 网络比较

我使用了 resnet101、resnext101\_32 $\times$ 4d、resnet152 网络结构，使用 SGD、Adam 优化器进行训练并将结果进行了对比，并针对是否使用预训练参数进行了对比实验，最终选择了最优的结果作为本次课设的模型。

### 4.3 resnet101 表现

#### 4.3.1 模型结果

首先下图为 resnet101 的最优结果的训练过程中训练、验证的损失与准确率。resnet101 的预训练模型在实验条件下验证集的准确率可达 93.3%。

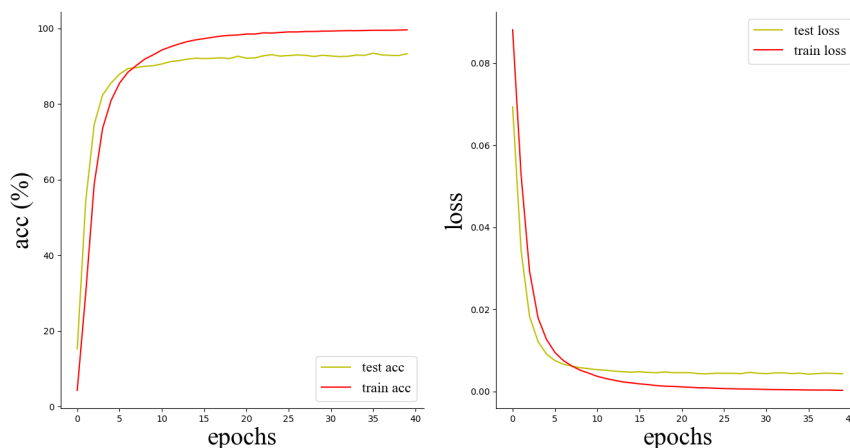


Figure 3: resnet101\_pretrained 训练过程及结果

### 4.3.2 预训练模型对比实验

我们在 SGD、resnet101 条件下设置了是否使用预训练模型进行训练进行对比，结果如下：

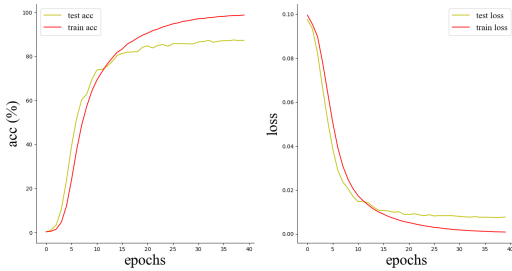


Figure 4: 不使用预训练模型

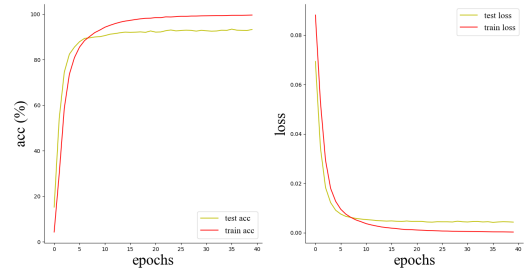


Figure 5: 使用预训练模型

明显可以看出使用预训练模型后模型泛化能力更强、更快收敛。下表为数据对比

模型	最高准确率	到达 80% 所需 epochs
resnet101	87.5%	15
resnet101_pretrained	93.46%	4

Table 1: 是否使用预训练模型进行的数据对比

### 4.3.3 优化器对比实验

我们在使用预训练模型的 resnet101 设置分别使用 SGD 与 Adam 优化器进行训练，并将结果进行对比：

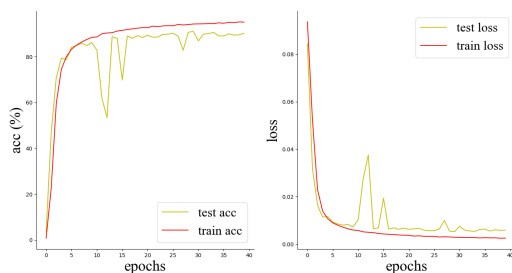


Figure 6: Adam

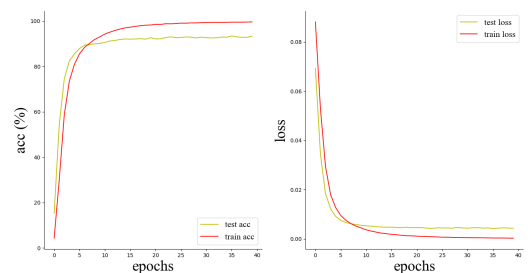


Figure 7: SGD

综合来看，在选择恰当的学习率后，SGD 的表现要优于 Adam，同时 SGD 的训练过程也更加稳定；Adam 的优势在于它对学习率不是很敏感，调参方面会比较方便。实验中 SGD 最高准确率为 93.46%，Adam 为 91.12%。

因此在后续两个网络的训练中，我们均使用预训练模型与 SGD，其它组合我也实验过，为避免重复介绍这里只展示最好结果。

#### 4.4 resnet152 表现

resnet152 的预训练模型在 SGD 条件下验证集的准确率可达 93.28%，下图为其训练过程中训练集、验证集的损失和准确率。

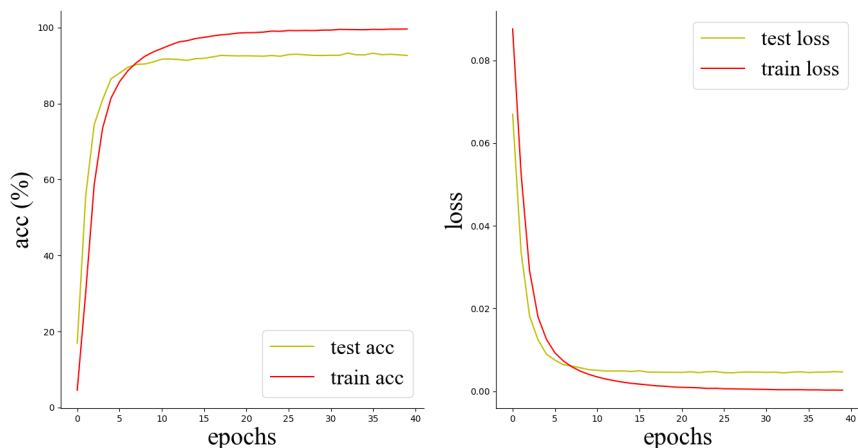


Figure 8: resnet152\_pretrained 训练过程及结果

#### 4.5 resnext101\_32×4d 表现

resnext101 的预训练模型在 SGD 条件下验证集的准确率可达 94.01%，下图为其训练过程中训练集、验证集的损失和准确率。

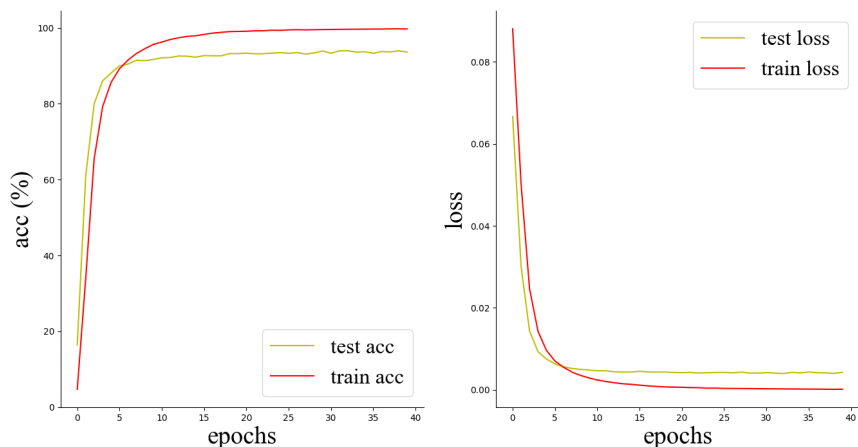


Figure 9: resnext101\_pretrained 训练过程及结果

#### 4.6 结果

综上所述，我选择 resnext101\_32×8d 作为本次课设的模型，其在实验条件下验证集的准确率最高，为 94.01%。其实验条件设置如下：

---

超参数	值
learning rate	1e-3
weight decay	1e-4
momentum	0.9
batch size	64
epochs	40

Table 2: 超参数设置

## 5 结语

在这次的手写汉字识别的神经网络设计报告中，我使用了多种不同的网络结构进行对比实验，并最终确定了最优性能的网络结构。通过这个过程，我深入了解了神经网络在图像分类领域的应用，掌握了如何选择神经网络模型以及优化器、调参使得模型性能变高。

在实验过程中，我也遇到了各种问题和挑战，但是通过不断尝试和调整，最终达到了预期的结果。这个过程让我更加深入地理解了神经网络的运作机制和优化方法，并增强了我解决实际问题的能力和信心。

尤其重要的一点是，我又学到了很多模型展示的技巧，或者说将模型结果展示在论文/报告当中的方式，这是我之前没有考虑到的。总的说来，这次的课程设计让我受益匪浅，提高了我的专业技能和学术素养。我相信这些经验和知识将在未来的学习和工作中发挥重要作用。

---

## 参考文献

- [1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.
- [2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [3] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.
- [4] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.
- [5] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.
- [6] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.
- [7] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.
- [8] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.