

深圳大学考试答题纸

(以论文、报告等形式考核专用)

二〇二一~二〇二二 学年度第 2 学期

课程编号	1901990006	课序号	02	课程名称	机器学习	主讲教师	王保华	评分	
学 号	2019193009	姓名	邹杰	专业年级	数学与应用数学（数金）	2019 级			

教师评语：见评分详表

题目：基于微调卷积神经网络的掌静脉与掌纹匹配方法

基于微调卷积神经网络的掌静脉与掌纹匹配方法

邹杰 2019193009

摘要

目前,掌静脉、掌识别技术领域受到了研究人员的高度关注,因而在该领域取得了飞速的发展与较大的突破。本文使用了基于微调卷积神经网络的方法,对掌静脉、掌纹进行识别与匹配。

具体而言,使用预训练的 ResNet18,并冻结 layer4 之前的参数进行掌静脉、掌纹识别的训练和测试。最后进行预测时,将模型输出的特征矩阵转化为特征向量,两两计算余弦相似度,认为与匹配对象有最大相似度的图片为同一个人。

本文依次使用了 TJU-P 掌纹数据集和 PolyU M_N 掌静脉数据集进行训练和测试,分别取得了训练准确率 100%、top-1 识别率为 90.2%、top-5 识别率为 95.78%;训练准确率 100%、top-1 识别率 97%、top-5 识别率为 99% 的较好结果。预测方面,在 PolyU M_N 掌静脉数据集上取 495 个类别进行训练和测试,并取 5 个类别共 60 个样本两两匹配,经过多次调试比较后,通过以准确率优先,兼顾 AUC 的最优值策略,选取最优相似度临界值 $\delta=0.9718$,最终匹配准确率达到 96%,此时 $AUC=0.93$ 。再使用 PolyU M_N 掌静脉数据集上所有类别 (500) 进行训练和测试,用 30 个验证集进行预测时,经过数十次实验后,选取最优相似度临界值 $\delta=0.99465$,最终匹配准确率达到 99%,此时 $AUC=0.87$ 。

接着,分别对图像增强方式、批处理大小与是否选取 ROI 区域进行了最优参数选取的讨论及可视化分析。图像增强方式方面:直方图均衡、CLAHE、Laplacian 和 log 变换中,直方图均衡 (hist) 效果最好;批处理大小方面:batch size=8, 16, 32, 64, 128 中 batch size=8 时模型的准确率收敛速度最快;是否选取 ROI 区域方面:相对简单地对原图像进行切割而言,不选取 ROI 区域更有利于模型对特征的识别与提取,最终准确率也更高。

关键字: 掌静脉匹配、掌纹匹配、迁移学习、卷积神经网络、ResNet18

目录

一、引言	3
二、文献综述	3
三、数据集介绍	5
四、基于迁移学习与 ResNet 的掌静脉与掌纹匹配方法	5
4.1 处理思路	5
4.2 图像预处理	6
4.2.1 ROI 区域切割	6
4.2.2 掌静脉、掌纹图像增强	7
4.2.3 其他预处理	8
4.3 数据集处理	8
4.3.1 实验使用的数据集	8
4.3.2 处理方法	8
4.4 迁移学习	9
4.5 卷积神经网络：以 ResNet18 为例	9
4.5.1 ResNet18 的独特机制	10
4.5.2 ResNet18 的架构	11
4.6 预测：掌静脉、掌纹特征匹配	12
五、实验结果与分析	13
5.1 实验环境	14
5.2 实验结果	14
5.2.1 图片增强实验结果	14
5.2.2 批处理大小	16
5.2.3 ROI 区域切割前后对比	17
5.2.4 验证集最优临界值寻找	18
5.3 实验分析	18
5.3.1 图片预处理实验分析	18
5.3.2 批处理大小实验分析	19
5.3.3 ROI 区域选取实验分析	19
5.3.4 验证集最优临界值寻找	20
六、结论	20
参考文献	22

一、引言

在当今网络信息化社会，我们越来越需要在不同的公众场合证明身份。之前，我们会采用密码、身份证、户口本和驾驶证之类传统的识别方法，但这些方法易遗忘、易丢失、易盗失，存在着若干安全隐患。大数据时代下，我们拥有了高性能的计算机，它们具备超强的算力，能够采取并处理生物的独特特征作为身份识别的“密码”，比如人脸、指纹、掌纹、虹膜、声音等生物特征。

其中，掌纹、掌静脉识别已经成为生物特征识别领域中一种非常有效的技术。相比于其他生物特征，手掌区域有许多可用的纹线，如主线、乳突纹、皱纹等。同时，掌纹还具有区分度高、稳定度高、侵犯度低等优点。目前，掌纹、掌静脉识别得到了研究人员的大量关注，广泛应用于刑侦及公共安全方面。

然而，从掌纹、掌静脉的采集到匹配的过程中，存在着许多干扰：手掌的形变（平移、变换和旋转），采集图像中存在的噪声、光线，采集图像的分辨率的高低等，这在很大程度上影响了识别的效率。因此，如何快速、准确地识别掌纹则是本文关注的问题。同时，本文不停留于掌静脉、掌纹识别层面（相当于是多分类任务），而是更进一步，通过训练出能够提取掌纹、掌静脉的特征的模型，直接进行特征比对，从而判断是否是同一个人。同时，由于数据集的特征十分类似，因此在研究掌静脉匹配方法时，可以迁移目前相对更为成熟的掌纹识别领域的方法。

二、文献综述

目前，掌静脉、掌纹识别技术领域受到了研究人员的高度关注，因而在该领域取得了飞速的发展与较大的突破。在对掌纹识别算法进行研究时，通常人们会将该过程分解成若干个子任务^[1]：数据集（掌纹）的采集、图片预处理、特征提取、特征匹配与决策，如图 1 所示：

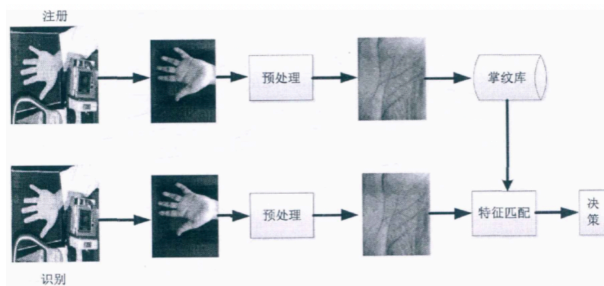


图 1 掌静脉、掌纹识别系统框架

掌静脉、掌纹的预处理目前主要分为两个任务：一是对掌静脉、掌纹感兴趣区域 (ROI) 的获取，林森、苑玮琦^[2] 给出了 ROI 区域的定位选择方法；二是图像增强，主要分为空域图像增强和频域图像增强算法^[3]。

掌静脉、掌纹的特征提取与特征匹配对应掌纹识别算法的设计，目前主流算法共有五类：基于子空间的方法、基于结构的方法、基于统计的方法、基于编码的方法和基于特征融合的方法。

基于子空间的方法：主成分分析（PCA）、独立成分分析（ICA）、线性判别分析（LDA）。因此，核心是对掌纹图像进行降维处理。

基于结构的方法：Shu^[4], Zhang^[5], Wu^[6] 分别提出了用不同的掌纹纹线检测算子提取四个方向上的特征，然后再合并。Rotinawa-Akinbile^[7] 运用了 Sobel 算子提取掌纹主线，特征匹配运用了欧氏距离。Nibouche^[8] 则运用了 SIFT 对提取的掌纹线和褶皱的特征点进行计算。

基于统计的方法：Liu^[9] 提出了中值滤波和小波变换，Pan^[10] 提出了微尺度不变的 Gabor 滤波器。

基于编码的方法：最早的有 Zhang^[11] 提出的 PalmCode 编码，到较新的 Tamrakar D^[12] 提出的异或和编码。

基于特征融合的方法：Pan X^[13] 采用 Gabor、PCA 等提取掌纹特征，并进行特征融合。Lu^[14] 将小波分解和 2DPCA 进行融合，提高了识别率。

近年来，随着机器学习、深度学习算法的流行以及高性能计算机的发展，许多复杂的分类任务都可以用神经网络来完成。其中，在特征提取方面，卷积神经网络尤其有效。贾伟等^[15] 则将用于分类任务的经典卷积神经网络（CNN）发展史绘制如图 2 所示：

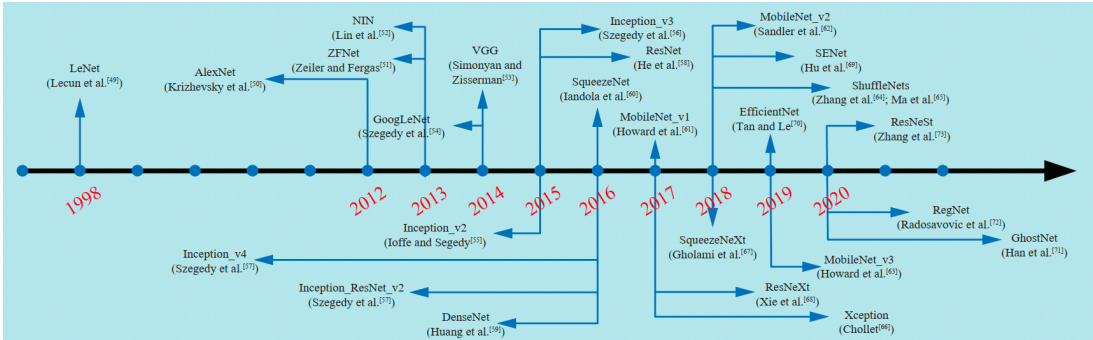


图 2 用于分类任务的经典 CNN 发展史

从手写体识别 MNIST 训练集上训练出的 LeNet-5，到第一个现代深度卷积网络模型的 AlexNet，后来出现了 VGG, Inception, ResNet, EfficientNet 等，模型的分类能力越来越强。贾伟教授等人是第一个将 CNN 运用到掌纹识别任务中的团队，并评估了经典 CNN 在该任务重的性能^[15]。

因此，本文在掌静脉、掌纹识别算法的设计上将参考贾伟团队的做法，使用 CNN 进行特征提取与特征匹配。

三、数据集介绍

贾伟团队将选取的 CNN 在 5 个 2D 掌纹数据库、1 个 3D 掌纹数据库和 2 个掌静脉数据库上进行评估，这些数据库在 2D 和 3D 掌纹、掌静脉识别领域都是具有代表性的。

其中，五个 2D 掌纹数据库包括了香港理工大学的掌纹 II 数据库 (PolyU II)、香港理工大学蓝带多光谱数据库 (PolyU M_B)、合肥工业大学掌纹数据库 (HFUT)、合肥工业大学交叉传感器掌纹数据库 (HFUT CS) 和同济大学掌纹数据库 (TJU-P)。使用的 3D 掌纹数据库是香港理工大学的 3D 掌纹数据库 (PolyU 3D)。两个掌静脉数据库分别是香港理工大学红外波段多光谱掌纹数据库 (PolyU M_N) 和同济大学手掌静脉数据库 (TJU-PV)。

这些数据库一般含有两个部分，来源于相同的样本，前后间隔一段时间再进行掌纹或掌静脉采样。在用深度学习的方法进行掌纹、掌静脉识别时，若训练集包含了这两个部分的样本，则很容易提高识别精度，这种实验模式称为混合数据模式。若训练集只来自第一个部分，测试样本来自第二个部分，则将该训练模式称为分离数据模式。^[15] 在本次实验中，采用了分离数据模型进行该任务的学习，以便获取更加准确、真实的识别率。

四、基于迁移学习与 ResNet 的掌静脉与掌纹匹配方法

4.1 处理思路

由于使用了具有代表性的数据集，因此不需要考虑数据集的采集过程，从而整体解决思路包括了：图片的预处理、特征提取、特征匹配与决策。基于此，绘制出本文处理思路如图 3 所示：

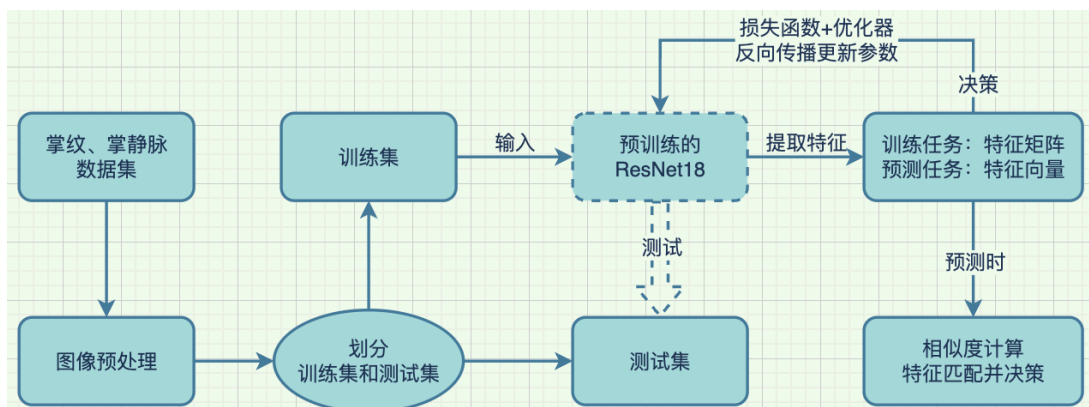


图 3 处理思路流程图

在设计掌静脉、掌纹匹配方法时，主要分成两个部分：模型的训练、测试和模型的预测。

模型的训练与测试则是用目前掌纹识别领域做的较多的方法，基于迁移学习的方式，通过调用 Pytorch 官方给出的经过充分调试、在 ImageNet 上预训练后的 ResNet18 模型，对 TJU-P 数据集进行训练和测试。在这个多分类任务上，模型输出的是一个 (2, 600) 的特征矩阵，通过交叉熵函数 Cross Entropy 计算损失，在 Adam 优化器的助力下通过反向传播计算梯度从而更新参数。在设定的训练轮次结束后，得到了有效提取掌纹特征的 ResNet 模型。

模型进行预测时，输入的数据同样是掌纹或掌静脉的图片，但此时模型的输出结果变成 (1, 1200) 的特征向量，将目标掌纹与其余所有掌纹进行相似度计算后进行比对，从而一一判断是否是同一个人。

4.2 图像预处理

4.2.1 ROI 区域切割

在处理 PolyU M_N 掌静脉数据库时，每一张原始图像的形状是 (288, 352, 3)，表示宽 288、长 352、通道数为 3。同时，原始图像中含有大量的无效信息，比如图像边缘的黑色背景、采集到的手指以及手掌的边缘，这些无效信息会加大模型的工作量，从而消耗有限的算力。因此，有必要先获取掌纹感兴趣区域 (ROI)。这部分区域含有丰富信息的掌静脉，能够作为生物特征有效区分不同身份。

由于采集的图片尺寸相同以及手掌位置基本相同，因此直接对图片的特定区域进行划分。在经过多次实验后，得出 ROI 区域为每张图大小中 (45 : 250, 75 : 300) 的部分，相当于给出四点坐标进行切割。

随机选择两个样本进行 ROI 区域的选取，结果如图 4 所示：

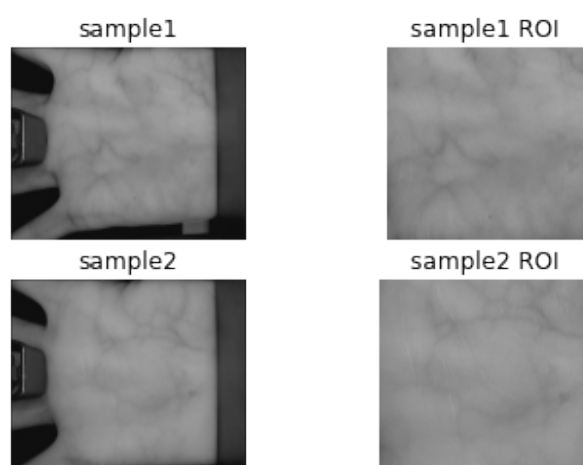


图 4 ROI 区域的选取

4.2.2 掌静脉、掌纹图像增强

图像增强主要是为了突出某些信息或者满足特定的需求。图像增强的技术一般分为两大类：空域图像增强和频域图像增强算法，此外还有结合模糊理论和遗传学的图像增强算法。

空域图像增强是对图像像素的直接处理，基本上以灰度映射变换为基础。常用方法有直方图均衡、空间滤波、灰度变换等。

频域图像增强的本质是卷积。它主要采用傅立叶变换或小波变换将图像变换到频域，然后进行频域增强，最后进行反变换回到空域，从而获得需要的图像。常用方法有低通滤波、高通滤波等。

对掌纹图像增强可以消除原始图像中的噪声，平衡光照影响以及增强对比度。本文选取了 4 种图像增强方法：直方图均衡、拉普拉斯算子、限制对比度自适应直方图均衡算法（CLAHE）和 log 变换。分别在两个数据集上随机选取两个样本，处理如图 5 和 6：

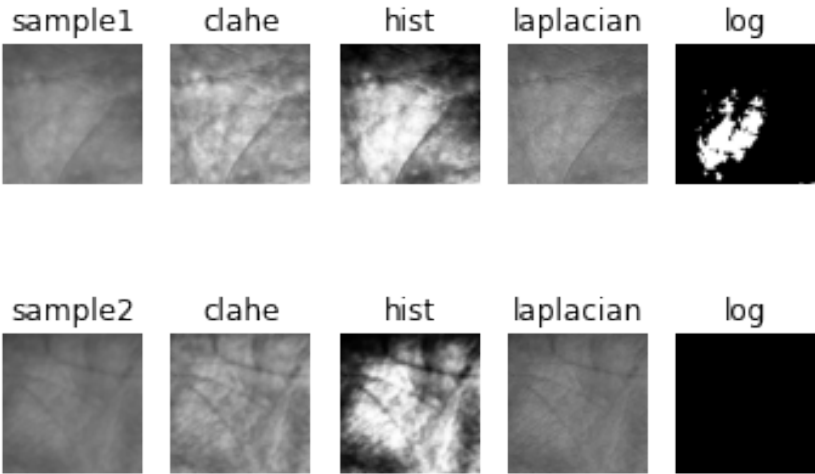


图 5 4 种对 TJU-P 的图像增强对比

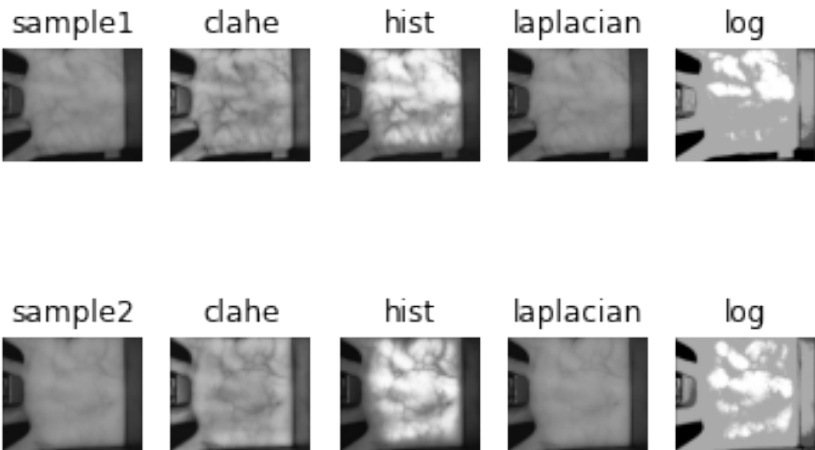


图 6 4 种对 PolyU M_N 的图像增强对比

其中，在 TJU-P 数据集上，log 变换效果最差，laplacian、clahe 和 hist 则有效去除了原始图像的噪声。在 PolyU M_N 数据集上，laplacian 算子处理后的结果和原始图像差别不大，其余方法增强后各自凸显出明显的特征。

4.2.3 其他预处理

这部分预处理主要是在使用 Pytorch 将数据集转换成数值型时需要进行的部分，由于使用的是预训练后的 ResNet18 模型，模型输入的图片尺寸为 (224, 224)，以及通道数为 3，因此首先需要使用 torchvision.transforms 中的 Resize 和 Grayscale 方法调整为标准大小，以便和预训练模型更匹配，从而有效快速地提高识别率。

其次，将图片处理为 Tensor 对象后（转化为张量进行运算），对张量进行正则化。正则化的参数为：($mean = [0.485, 0.456, 0.406]$, $std = [0.229, 0.224, 0.225]$)，即 ImageNet 的正则化参数。

4.3 数据集处理

4.3.1 实验使用的数据集

本文使用了 2 个 2D 数据库进行实验，分别是同济大学掌纹数据库 (TJU-P) 和香港理工大学红外波段多光谱掌静脉数据库 (PolyU M_N)。具体信息如表 1：

表 1 实验使用数据集介绍

数据库	种类	是否接触	个体数量	掌纹、静脉数量	状态数量	状态间隔	每个掌纹图片数量	总数量
TJU-P	2D 掌纹	否	300	600	2	61 天	10×2	12000
PolyU M_N	掌静脉	是	250	500	2	9 天	6×2	6000

4.3.2 处理方法

在 TJU-P 数据库上，采用分离数据模式^[15]进行训练，以 session1 的 6000 张作为训练集，并划分为 0-599（共 600）类，以 session2 的 6000 张作为测试集，并划分为 0-599（共 600）类。处理完后，将顺序随机打乱，即开启 Shuffle。同时，训练时采用批处理，设定 Batch Size=64（默认参数）。

在 PolyU M_N 数据库上，也采用分离数据模式进行训练，以各个掌纹文件夹（共 500 个）中数字为类别，取每个掌纹类别中前 6 张作为训练集，后 6 张作为测试集，划分为 0-499（共 500）类。处理完后，将顺序随机打乱，即开启 Shuffle。同时，训练时采用批处理，设定 Batch Size=64（默认参数）。

4.4 迁移学习

深度学习能够通过学习有效的特征并进行识别，但对训练数据的数量和质量要求较高。目前掌纹数据库大多数数据量小且图像尺寸大，因此直接在深度学习网络中使用难以有较好的识别效果。但迁移学习可以快速、有效地处理该任务。因此，本文提出了基于迁移学习的卷积神经网络的掌纹匹配方法。

迁移学习的目的是将域源学习到的知识应用推广到目标域中^[15]，一般目标域的数据量远远小于域源的样本数据量。迁移学习中有两个概念非常重要：域（Domain）和任务（Task）。域由空间特征 X 和特征分布 $P(X)$ 组成，是学习的主体。域又分为源域和目标域，源域是已有知识的域，目标域是即将要进行学习的域。迁移学习和传统机器学习的区别如图 7：

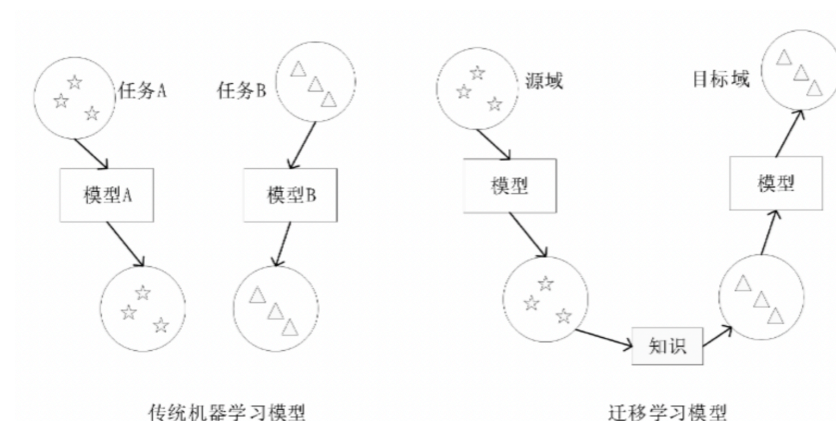


图 7 迁移学习和传统机器学习的比较

从图 7 可看出，传统的机器学习模型使用任务 A 的数据训练和测试模型，对于不同的任务 B 则要重新建模、训练、测试。迁移学习则是把在源域上学得的知识，有选择地应用到目标域上，从而更快更好地完成目标域任务。

本文采用了微调的深度迁移学习方式，即使用在 ImageNet 上训练好的 ResNet18 模型的权重参数作为模型的初始化权重参数，并有选择地冻结一部分参数，并对其余层进行参数的更新和训练。用 ImageNet 训练的神经网络对纹理的识别能力更强，因此适用于掌纹、掌静脉的识别。

使用 `models.resnet18(pretrained=True)` 调用 ResNet18 模型后，需要对冻结的参数设定 `requires_grad=False`，并使 `layer4` 和 `fc` 层的参数可以训练。同时，将 `fc` 层最后的输出特征数由 1000 改成 600（TJU-P 数据集中的分类数量）。

4.5 卷积神经网络：以 ResNet18 为例

ResNet 是何恺明等在 2015 年首创的基于深度残差学习的图像识别模型^[17]。它最大的贡献是解决了随着卷积神经网络的增加，模型的识别效果却更差的问题。ResNet 拥

有独特的残差学习机制，即引入了残差网络，从而使得随着网络深度的增加，模型能够表征的最优点至少是不比浅层所表征的最优点差。然而，在没有残差网络时，网络的表征能力虽然能随着层数的加深而增强，但此时表征的最优点会离实际需要学习的最优点越来越远，从而导致模型效果变弱。从数学角度上描述，则是解决了网络加深后梯度变小训练失效的问题。

4.5.1 ResNet18 的独特机制

1. **Residual Learning(残差学习)**. 何恺明等将残差学习形象地表示成了一块一块积木^[17]，如图 8：

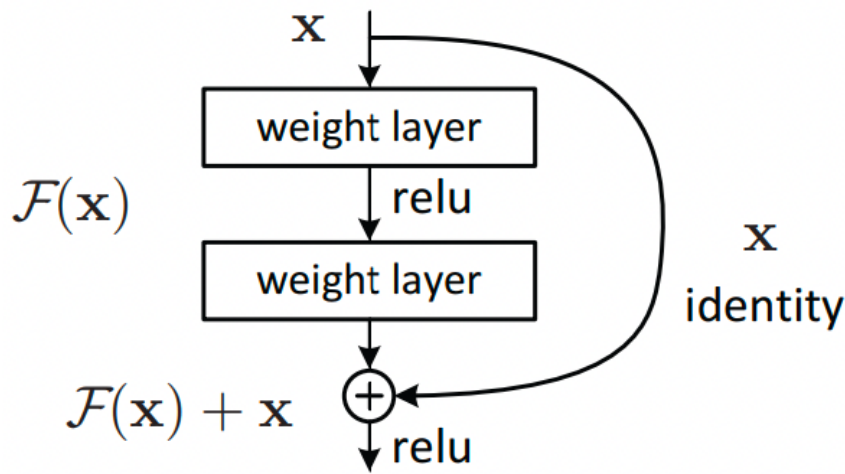


图 8 残差学习：一块积木

图中 X 为浅层网络的输出，右边连接一个 X 的 identity ，表示 identity mapping （一种输入为 X ，输出也为 X 的机制）。假设我们想得到的映射为 $H(X)$ ，则可让非线性网络层去拟合残差映射 $F(X) := H(X) - X$ ，那么目标映射则可写成 $F(X) + X$ 。

2. **Shortcut Connection(快捷连接)**. 在图 8 中便展示了快捷连接的作用：它能使浅层网络的输出值跳过若干层直接和当前网络输出值相连接，从而得到 $F(X) + X$ 。

前边提到，ResNet 最大的贡献是解决了随着卷积神经网络的增加，模型的识别效果却更差的问题。从数学角度上描述，则是解决了网络加深后梯度变小训练失效的问题。这里从数学角度给出解释：

假设原始网络为 $g(x)$ ，网络加深后为 $f(g(x))$ ，则在没有引入残差学习机制时，反向传播求梯度时：

$$\frac{\partial f(g(x))}{\partial x} = \frac{\partial f(x)}{\partial g(x)} \times \frac{\partial g(x)}{\partial x} \quad (4.1)$$

当梯度很小趋于 0 时，可理解为 0 附近的高斯分布，此时相乘的操作可能会使加深后模型的梯度更小，从而增大了训练难度。

引入残差机制后，反向传播求梯度时：

$$\frac{\partial[f(g(x) + g(x))]}{\partial x} = \frac{\partial f(x)}{\partial g(x)} \times \frac{\partial g(x)}{\partial x} + \frac{\partial g(x)}{\partial x} \quad (4.2)$$

此时，得到的映射是 $F(X) + X$ 的形式，即残差学习的标准输出形式。在求梯度时，得到了加深部分的梯度与未加深前的梯度的和，两者之间保持线性关系，互不影响，因此减少了加深后梯度训练不出来的问题的影响，因为在这之前的映射可以保留并传递下去。

4.5.2 ResNet18 的架构

ResNet 通过 **Shortcut Connection** 将非线性网络层分为一块一块的残差网络，就像搭建的一块块积木，这些残差块只专注于训练自己内部的映射，不需要考虑其他网络层的映射，这充分发挥了卷积神经网络提取特征识别特征的优势，并大大减少了模型的训练参数，降低了模型的复杂程度。何恺明等也给出了 18、34、50、101 和 152 层的 ResNet 网络架构^[17]，如图 9：

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

图 9 ResNet 的网络架构

注意到，以 50 层为分界线，小于 50 层的 ResNet 模型使用的是 BasicBlock 的卷积层结构（即 3×3 的卷积核），大于 50 层的 ResNet 模型使用的是 BasicNeck 的卷积层结构（即 $1 \times 1, 3 \times 3$ 的卷积核）。同时，conv1*1 的卷积核，在 padding=0, stride=1 的情况下，只会影响图片的通道数，不会改变图片的大小。conv3*3 的卷积核在 padding=1, stride=1 的情况下，只会影响图片的通道数，不会改变图像大小；而在 padding=1, stride=2 的情况下，则会使图像缩小一倍。本文主要对 18 层的模型进行分析。

如上图所示，18 层的 ResNet 模型除了一开始预处理层（将通道数由 3 变为 64，图片尺寸由 224*224 变为 112*112）和最后的全连接层（输入特征 512，输出特征在掌纹识别任务中为 600）外，主体部分由 4 层组成，每一层中包含 2 个残差块即 Basic Block。如图 10：

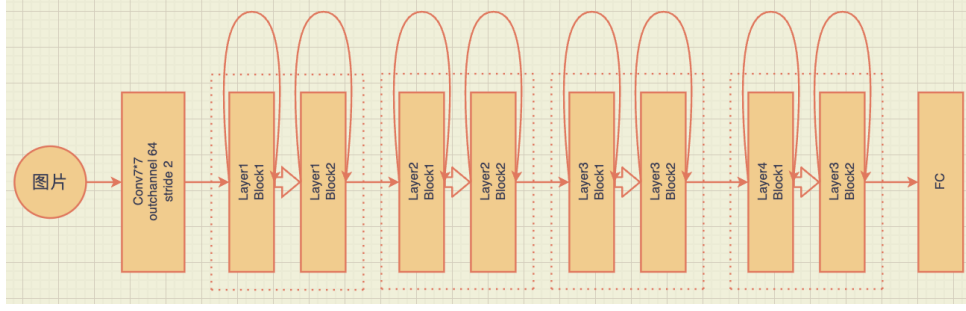


图 10 ResNet18 的架构

第一层的 2 个 Basic Block 中：使用的都是的 3*3 卷积核，通道数不变仍是 64，图片尺寸由 112 变为 56；第一个 block 的结构是 conv, batchnorm, relu, conv, batchnorm；第二个 block 的结构是 conv, batchnorm, relu, conv, batchnorm.

第二层的 2 个 Basic Block 中：使用的都是的 3*3 卷积核，通道数由 64 变为 128，图片尺寸由 56 变为 28；第一个 block 的结构是 conv, batchnorm, relu, conv, batchnorm, downsample；第二个 block 的结构是 conv, batchnorm, relu, conv, batchnorm.

第三层的 2 个 Basic Block 中：使用的都是的 3*3 卷积核，通道数由 128 变为 256，图片尺寸由 28 变为 14；第一个 block 的结构是 conv, batchnorm, relu, conv, batchnorm, downsample；第二个 block 的结构是 conv, batchnorm, relu, conv, batchnorm.

第四层的 2 个 Basic Block 中：使用的都是的 3*3 卷积核，通道数由 256 变为 512，图片尺寸由 14 变为 7；第一个 block 的结构是 conv, batchnorm, relu, conv, batchnorm, downsample；第二个 block 的结构是 conv, batchnorm, relu, conv, batchnorm.

卷积层结束后，经过平均池化层和全连接层，并使图片大小变为 1*1，输入特征为 512，输出特征为 600（PolyU M_N 掌静脉数据集上是 500）。

4.6 预测：掌静脉、掌纹特征匹配

在进行预测时，模型输出的不再是特征矩阵，而是 1*1200(PolyU M_N 掌静脉数据集上是 1*1000) 的特征向量。特征向量有助于计算图形的相似度，从而实现不需要注册的掌静脉、掌纹匹配方法。

具体方法是：将训练完的模型保存下来后，在需要进行预测时再将模型已训练好的参数加载到定义模型中。将预测样本一对一对进行匹配，计算余弦相似度，通过设定许多临界值，依次计算在若干临界值下预测集的准确率和 AUC 值，从而选取最优临界值 $\hat{\delta}$ 。只要余弦相似度大于等于 δ ，则判定为同一人，输出 1，反之输出 0.

$$\begin{cases} \cos(X_1, X_2) \geq \delta, result = 1 \\ \cos(X_1, X_2) < \delta, result = 0 \end{cases} \quad (4.3)$$

五、实验结果与分析

在具体实验过程中，本文先在 TJU-P 数据集上进行了实验，最终获得了训练集准确率 100%，测试集上（此时 loss=0.005）Top-1 识别率达到 90.2%、Top-5 准确率达到 95.78% 的结果。接着，再在 PolyU M_N 数据集上进行实验，最终获得了训练集准确率 100%，测试集上（此时 loss=0.002）Top-1 识别率达到 97%、Top-5 识别率达到了 99% 的结果。在 TJU-P 数据库上，采用分离数据模式^[15]进行训练，以 session1 的 6000 张作为训练集，并划分为 0-599（共 600）类，以 session2 的 6000 张作为测试集，并划分为 0-599（共 600）类。处理完后，将顺序随机打乱，即开启 Shuffle。在 PolyU M_N 数据集上，同样采用分离数据模式进行训练，划分为 0-499（共 500）类，以每个类别中第一个 session 的 6 张掌静脉作为训练集，第二个 session 的 6 张作为测试集。在实验过程中，在 TJU-P 数据集上，通过调整了图片预处理的方式和批处理大小的超参数后，得到了一些有趣的结论。在 PolyU M_N 数据集上，则比较了切割 ROI 前后的模型训练、测试效果。在两种数据集上模型最终训练、测试结果的数据如表 2：

在实验过程中均使用 Adam 优化器进行参数更新，并使用最优学习率^[15] (5×10^{-5}) 进行训练与测试。

表 2 在两种数据集上模型 epoch=30 时训练、测试结果

数据集	loss	训练集准确率 / %	top1 准确率 / %	top5 准确率 / %
PolyU M_N	0.005	100.00	97	99
TJU-P	0.002	100.00	90.2	95.78

在进行预测时，使用 PolyU M_N 的 500 个类中 495 个类别进行训练及测试，留 5 个类进行预测。训练及测试后，将模型保存下来，并在预测时加载。5 个类中共有 60 个样本，一个类有 12 张掌静脉图。本文共采用了两种预测方法：

方法一：取最大值。依次取每一张图片分别与 60 张图片计算余弦相似度，再取每一次的比较后列表中相似度最大的 12 张，若这 12 张来自同一类，则说明准确率为 100%。这样的操作重复 60 次，再取平均准确率为最终的准确率。经过测试后，匹配准确率为 94.58%。

方法二：设定临界值。当验证集以一对的方式给出时，此时无法和其他候选目标依次计算相似度再取最大值进行比较，只能进行一次比较，并判断是否是同一类。此时，则可根据临界值法，设定许多临界值，依次计算在若干临界值下预测集的准确率和 AUC 值，从而选取最优临界值。经过数十次实验后，得到最优临界值 δ 为 0.9718，该临界值下预测集准确率为 96%，AUC=0.93，说明具有较好的分类效果。

在第二个验证集中，使用 PolyU M_N 的 500 个类进行训练及测试，并将模型保存，在预测时加载。验证集中有 30 个类，每类含 3 张掌静脉，共 90 张，因此可以设置成

90*90 共 8100 次两两匹配的比较。使用设定临界值的方法，如方法二中介绍。经过数十次实验后，得到最优临界值 δ 为 0.99465，该临界值下准确率为 99%，AUC=0.87。

5.1 实验环境

在进行实验时，一共用到了 2 个环境。

1. Macbook Air M1 2020

- Python 3.7.13
- Pytorch 1.7.1 + torchvision 0.8.2
- OpenCV 3.4.2

2. MIST GPU 深度学习平台

- GPU: RTX A6000 (32G)
- Python 3.9.12
- Pytorch 1.11.0
- OpenCV 4
- CUDA 11.6 + cudnn 8.4.0

5.2 实验结果

5.2.1 图片增强实验结果

在 TJU-P 数据集上，将通过三种不同方式进行图片增强后的掌纹输入模型后，与未经增强的图像训练 15 个 epochs 后的结果进行对比。迭代次数为 15，批大小为 64，学习率为 5×10^{-5} ，优化器为 Adam。损失、top1 准确率和 top5 准确率的结果如下图 11，12 和 13：

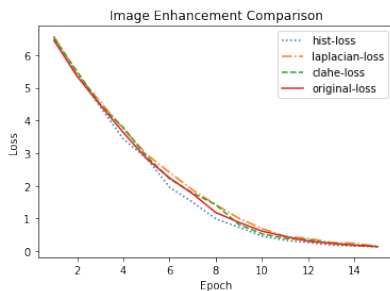


图 11 loss

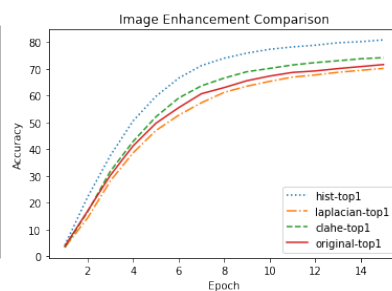


图 12 top-1 accuracy

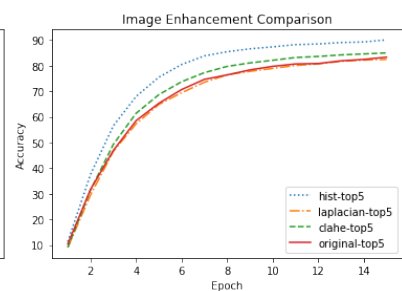


图 13 top-5 accuracy

由实验可知：经过直方图均衡方式图像增强后，loss 收敛最快、识别率提升最高；CLAHE 的效果其次；拉普拉斯算子处理的效果最不显著，甚至在 top1 上的准确率低于未处理前的图像的识别率。同时，本文并没有展现模型在训练集上的准确率，是因为

ResNet 的特征提取与识别能力过于出色，以至于在使用数据增强后基本都在 4 到 5 个 epoch 后准确率便能达到 100%，不使用数据增强时也大概能在 7 个左右 epoch 时达到 100%. 进一步地，分别取以上四种情况在最后一个 epoch 的结果进行对比，如表 3：

表 3 3 种不同图片增强方式 epoch=15 时结果对比

增强方式	loss	训练集准确率 / %	top1 准确率 / %	top5 准确率 / %
original	0.15	100.00	71.62	84.32
laplacian	0.15	100.00	70.20	82.45
clahe	0.15	100.00	74.2	84.98
hist	0.14	100.00	80.85	90.07

由表可知，训练结果最好的是使用直方图均衡进行图片增强的模型，top1 识别率提升了 12.89%.

在 PolyU M_N 掌静脉数据集上，由于通过上述实验证明 hist 和 clahe 取得的效果更好，因此只采用这两种图像增强方式进行对比。参数选取：迭代次数为 15，批处理大小为 8，学习率与优化器保持不变。对比结果如图 14，15 和 16：

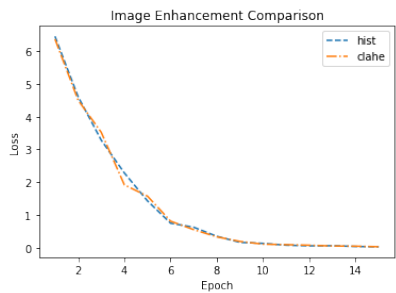


图 14 loss

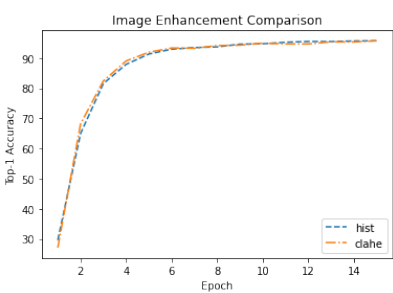


图 15 top-1 accuracy

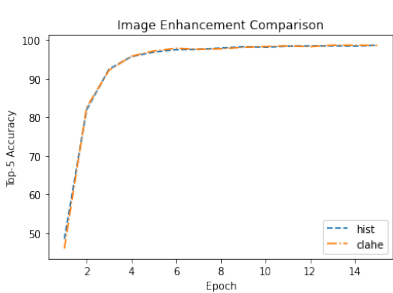


图 16 top-5 accuracy

由图可知：在 PolyU M_N 掌静脉数据集上，使用 hist 和 clahe 进行图像加强的提升效果几乎相同，细微观察后发现，loss 的收敛速度和 top1 准确率上 hist 略胜一筹。进一步地，分别取该两种情况在最后一个 epoch 的结果进行对比，如 4：

表 4 2 种不同图片增强方式 epoch=15 时结果对比

增强方式	loss	训练集准确率 / %	top1 准确率 / %	top5 准确率 / %
clahe	0.028	100.00	95.70	98.57
hist	0.027	100.00	95.73	98.60

由表可知：hist 和 clahe 在处理 PolyU M_N 掌静脉数据集时难分伯仲。

5.2.2 批处理大小

在 TJU-P 数据集上，设定批处理大小 Batch Size 分别为 128, 64, 32, 16, 8，依次对模型进行训练和测试。迭代次数为 15，图像增强方式为直方图均衡，学习率为 5×10^{-5} ，优化器为 Adam。损失、top1 准确率和 top5 准确率的结果如下图 17，18 和 19：

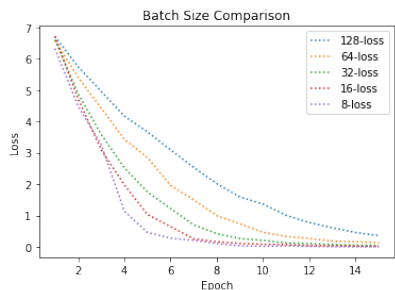


图 17 loss

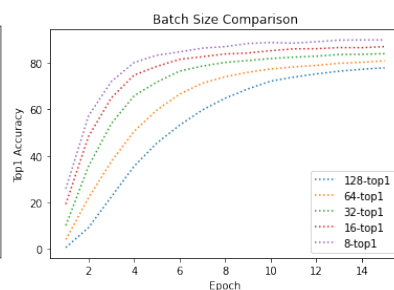


图 18 top-1 accuracy

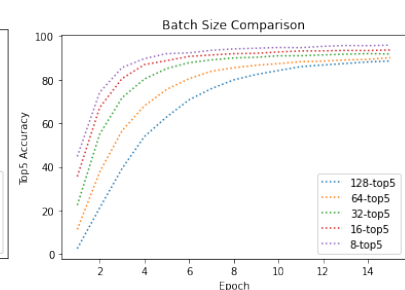


图 19 top-5 accuracy

由图可知：随着批处理大小的减小，模型的损失函数收敛速度加快，初始的和最终的 top1 和 top5 识别准确率递增，并且准确率提升速度也递增。经过观察，batch size=32 时前后区分较为明显。最优秀的模型是使用 batch size=8 时的模型。值得注意的是，小批次训练时 top5 准确率相较于 top1 准确率却提升地较少，并且随着迭代次数的增加，准确率提升的速度趋缓。进一步地，分别取以上五种不同批处理大小在最后一个 epoch 的结果进行对比，如表 5：

表 5 5 种不同批处理大小 epoch=15 时结果对比

批处理大小	loss	训练集准确率 / %	top1 准确率 / %	top5 准确率 / %
128	0.36	100.00	77.82	88.53
64	0.14	100.00	80.85	90.07
32	0.05	100.00	83.88	91.75
16	0.02	100.00	86.93	93.60
8	0.005	100.00	90.20	95.78

由表可知，训练结果最好的是使用 batch size=8 的模型，top1 识别率最终达到了 90.20%，较 batch size=64（初始批处理大小）提升了将近 10%。

在 PolyU M_N 掌静脉数据集上，由于上述实验证明批处理大小越小，模型训练、测试时的准确率收敛速度越快，因此这里只选取 batch size=8 和 16 进行验证，如图 20，21 和 22：

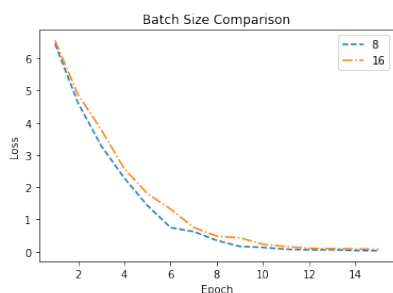


图 20 loss

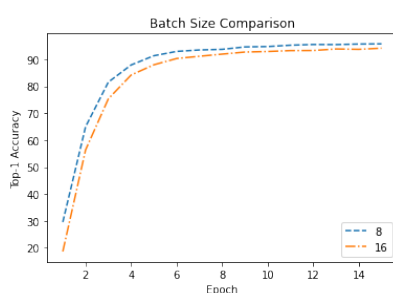


图 21 top-1 accuracy

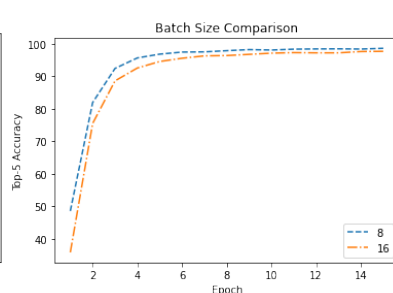


图 22 top-5 accuracy

由图可知：在 batch size 更小时，加快了模型进行训练和测试的准确率收敛。取最后一个 epoch 的结果进行对比，如 6：

表 6 2 种不同图片增强方式 epoch=15 时结果对比

批处理大小	loss	训练集准确率 / %	top1 准确率 / %	top5 准确率 / %
16	0.07	100.00	94.10	97.70
8	0.027	100.00	95.73	98.60

由表可知：在 batch size=8 时，模型进行训练和测试的准确率收敛速度比 batch size=16 时更快，因此在使用深度学习平台进行模型的训练和测试时可以使用较小的批次。

5.2.3 ROI 区域切割前后对比

在掌静脉识别任务上，使用 PolyU M_N 数据集，相较于 TJU-P 训练集而言，在该掌静脉数据集上模型的训练、测试效果更好。但数据集分辨率较低，切割后的 ROI 区域观察上看虽然掌纹信息变得丰富、集中，但却不可避免地丢失了一些信息。因此，本文想对该数据集 ROI 区域切割前后的模型训练、测试效果进行对比。其余参数为：batch size=8, 图像增强方式为 hist 直方图均衡。对比结果如图 23, 24 和 25：

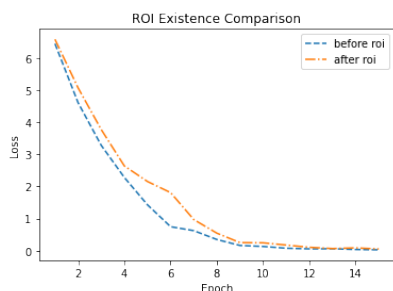


图 23 loss

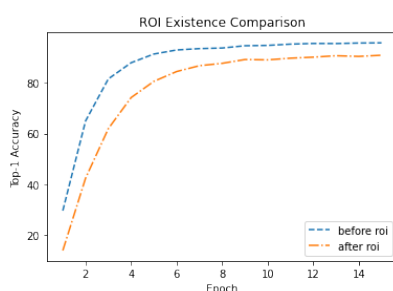


图 24 top-1 accuracy

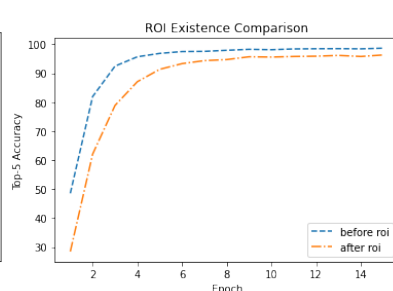


图 25 top-5 accuracy

由图可知：ROI 区域切割前，即使用原采集图像进行训练时，无论是损失收敛速度，

还是 top1 和 top5 的准确率收敛速度都快速提升。进一步地，可以将结果从数值层面展示，如表 7：

表 7 选取 ROI 区域前后的模型 epoch=15 时训练、测试结果

是否选取 ROI 区域	loss	训练集准确率 / %	top1 准确率 / %	top5 准确率 / %
是	0.005	100.00	90.86	96.3
否	0.002	100.00	95.73	98.63

由表可知：在 epoch=15 时，最终未选取 ROI 区域时，top1 和 top5 的准确率分别比选取 ROI 区域后增加了 5% 和 2%。

5.2.4 验证集最优临界值寻找

在给定的验证集中，使用 PolyU M_N 的 500 个类进行训练及测试，并将模型保存，在预测时加载。验证集中有 30 个类，每类含 3 张掌静脉，共 90 张，因此可以设置成 90*90 共 8100 次两两匹配的比较。使用设定临界值的方法，在 $\delta \in [0.9925, 0.9950]$ 中进行实验，得到 AUC 和准确率曲线如图 26，27：

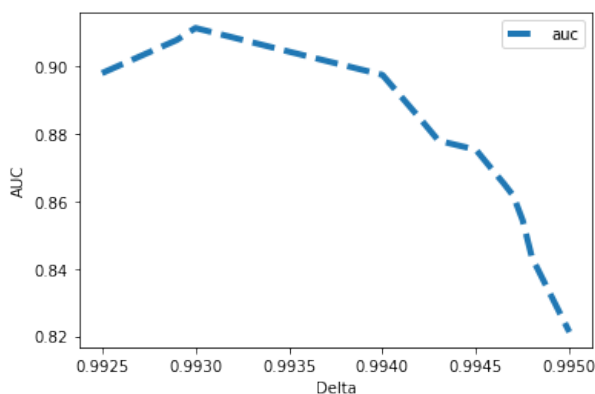


图 26 AUC

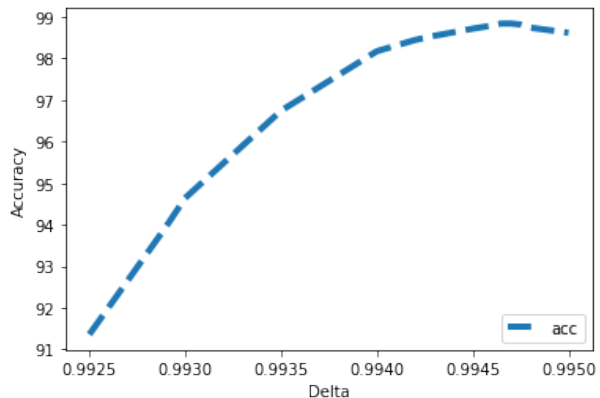


图 27 Accuracy

经过数十次实验后，得到最优临界值 δ 为 0.99465，该临界值下准确率为 99%，AUC=0.87. 此时准确率达到最大值，同时具有较好的 AUC。

5.3 实验分析

5.3.1 图片预处理实验分析

1. 使用 ImageNet 正则化参数有助于提高识别率。在 TJU-P 数据库上，分别使用了 3 种正则化参数进行训练：不使用正则化、使用 MNIST 数据集的正则化参数 ($mean = (0.1307, 0.1307, 0.1307)$, $std = (0.3081, 0.3081, 0.3081)$)、使用 ImageNet 的正则化参

数。不使用正则化时，在学习率为 5×10^{-5} 、15 个 epochs 下识别率大致在 60% 左右，使用 MNIST 数据的正则化参数的识别率在 69% 左右，使用了 ImageNet 的正则化参数后，识别率提升至 73% 左右。

2. **使用图像增强有助于提高识别率。**在 TJU-P 数据库中，分别使用了 3 种图像增强方式：直方图均衡 (hist)、拉普拉斯算子 (laplacian)、限制对比度自适应直方图均衡算法 (CLAHE)。不使用图像增强时，15 个 epoch 后识别率为 70%；使用 laplacian 后，识别率几乎没有提升；使用 CLAHE 后，识别率提升了 5%，达到了 75%；使用 hist 后，识别率迅速提升 11%，达到了 81%，Top-5 准确率达到 91%。直方图均衡的增强方式使掌纹图像的对比度更强、掌纹纹线特征更加清晰突出，更有利于 ResNet 对特征的提取与匹配，从而提高了整体的识别率。
3. **不开数据增强更有利于提高识别率。**在 TJU-P 数据库中使用数据增强不会对识别率提升有帮助，因为每个人的掌纹的走势虽然有所不同，但大致走势是相同的，在大致相同走势的掌纹中，对掌纹进行随机切割、上下翻转或左右翻转，有可能会和另一个分类对象高度重合，对于迁移学习的模型来说学习的参数有限（并且为了减少训练时间节省内存，还将卷积层参数冻结了），以本文使用的 ResNet 为例，可更新的参数较少，因此无法对数据增强后的图像进行有效的特征提取。多次实验后，发现：在该任务上，数据增强会使基于迁移学习的卷积神经网络降低训练、测试准确率，训练准确率收敛速度下降，识别率整体下降 10%。

5.3.2 批处理大小实验分析

1. **小批次训练更有利于提升识别率，加快了识别率的收敛速度。**随着批处理大小的减小，模型的损失函数收敛速度加快，初始的和最终的 top1 和 top5 识别准确率递增，并且准确率提升速度也递增。
2. **以批处理大小 32 为界，前后变化较大。**经过观察，batch size=32 时前后区分较为明显。32 前随着批次的减小，模型识别率提升较显著，但 32 后随着批处理大小的减小，模型识别率的提升放缓，趋于收敛。
3. **top5 准确率较 top1 准确率随着迭代次数增多更难提升。**值得注意的是，小批次训练时 top5 准确率相较于 top1 准确率却提升地较少，并且随着迭代次数的增加，准确率提升的速度趋缓。不过此时最优模型下 top5 准确率已经将近 96%，逐渐靠近 100%，理应会愈发难以提升。

5.3.3 ROI 区域选取实验分析

不选取 ROI 区域时模型的准确率更高、收敛速度更快。本文主要对比了选取 ROI 区域（对特定部分进行切割）与不选取 ROI 区域这两种情况，通过整体的收敛情况和最后一个迭代时的测试效果，发现不选取 ROI 区域时模型的准确率更高、收敛速度更快，

整体效果优于选取 ROI 区域后的模型。一般来说,选取 ROI 区域后,能更有利于模型的训练与测试,因为图像集中了更丰富的训练特征,但同时也舍弃了一部分有效信息,这部分有效信息对分类是否有利则是需要实验进行证明的。

5.3.4 验证集最优临界值确定

以准确率优先,兼顾 AUC 的最优值策略。在寻找最优临界值 δ 时,存在规律:在一定范围内,随着临界值的增大,AUC 和准确率都增加。在涵盖了最优临界值的小范围内,临界值先增后减,准确率也会先增后减,存在最大值,AUC 也会先增后减,但在准确率最大值点处 AUC 已经处于下降趋势中,也就是说 AUC 最大值点和准确率的最大值点不重合。故本文制定了以准确率优先,兼顾 AUC 的最优值策略,通过若干次实验粗略找出最优临界值。

六、 结论

掌静脉和掌纹已然成为一种十分有效的生物识别特征,受到了研究人员的大量关注。本文通过迁移学习中的微调学习方式,使用了基于微调的 ResNet18 模型对掌静脉、掌纹进行识别,并在预测时转变为输出特征向量从而实现不用进行掌纹注册的掌纹匹配方法。本文的主要研究成果如下:

深度学习能够通过学习有效的特征并进行识别,但对训练数据的数量和质量要求较高。目前掌静脉、掌纹数据库大多数据量小且图像尺寸较大,因此直接在深度学习网络中使用难以有较好的识别效果。但迁移学习可以快速、有效地处理该任务。

ResNet 通过独特的 Rresidual Learning 与 Shortcut Connection 机制,将非线性网络层分为一块一块的残差网络,就像搭建的一块块积木,这些残差块只专注于训练自己内部的映射,不需要考虑其他网络层的映射,这充分发挥了卷积神经网络提取特征识别特征的优势,并大大减少了模型的训练参数,降低了模型的复杂程度。同时,经过实验证明,通过迁移学习的方式,在掌静脉、掌纹识别任务上,ResNet 的效果好于 VGG。

经过多次实验后得出:1. 在图像预处理方面:使用 ImageNet 正则化参数有助于提高识别率,并且相较于 MNIST 正则化参数更有效;使用图像增强有助于提高识别率,并且在直方图均衡、CLAHE 与 laplacian 中直方图均衡最有效;不开数据增强更有利于提高识别率。2. 在批处理大小方面:小批次训练更有利于提升识别率,在 TJU-P 数据集上:最优的参数为 batch size=8,15 个 epoch 后模型的 top1 准确率达到 90.20%,top5 准确率达到 95.78%,在 PolyU M_N 数据集上:最优的参数为 batch size=8,15 个 epoch 后模型的 top1 准确率达到 95.73%,top5 准确率达到 98.63%;以批处理大小 32 为界,前后变化较大。32 前随着批次的减小,模型识别率提升较显著,但 32 后随着批处理大小的减小,模型识别率的提升放缓,趋于收敛;top5 准确率较 top1 准确率随着迭代增多更

难提升，此时最优模型下 top5 准确率已经将近 96%，逐渐靠近 100%，理应会愈发难以提升。3. 在是否选取 ROI 区域方面，不选取 ROI 区域时模型的准确率更高、收敛速度更快。在 epoch=15 时，最终未选取 ROI 区域时，top1 和 top5 的准确率分别比选取 ROI 区域后增加了 5% 和 2%。由此说明不同数据集采取的处理方式应具体讨论。4. 在验证集最优临界值确定方面：本文制定了以准确率优先，兼顾 AUC 的最优值策略，通过若干次实验粗略找出最优临界值，发现了 AUC 最大值点和准确率的最大值点不重合这一特点。

参考文献

- [1] Zhang D. Palmprint Authentication System[M] Singapore: World Scientific,2005: 431-444.
- [2] 林森, 苑玮琦, 等. 掌纹感兴趣区域定位与选择方法综述 [J]. 计算机工程与应用, 2011, 47(14): 21-24
- [3] Lim J S. Two-dimensional signal and image processing[J]. Englewood Cliffs, NJ, Prentice Hall, 1990, 34(1): 6-8.
- [4] Shu W, Zhang D, Palmprint verification: an implementation of biometric technology[C]//Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on. IEEE, 1998 :219-221.
- [5] Wu X, Wang K, Zhang D. Approach to line feature representation and matching for palmprint recognition[J]. Journal of Software, 2004, 15(6): 869-880.
- [6] Wu X, Zhang D, Wang K. Paim line extraction and matching for personal authentication[J], IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2006, 36(5): 978-987.
- [7] Rotinwa-Akinbile M O, Aibinu A M, Salami M J E. Palmprint recognition using principal lines characterization[C]//First International Conference on Informatics and Computational Intelligence. Bandung, Indonesia: IEEE, 2011: 278-282.
- [8] Nibouche O, Jiang J. Palmprint matching using feature points and SVD factorisation[J]. Digital Signal Processing, 2013, 23(4): 1154-1162.
- [9] Liu Y, Yuan W, Guo J. Application of wavelet transform and block statistic to palmprint recognition[J]. Jisuanji Gongcheng yu Yingyong(Computer Engineering and Applications), 2011, 47(24).
- [10] Pan X, Ruan Q, Wang Y. Palmprint recognition using Gabor local relative features[J]. Jisuanji Gongcheng yu Yingyong(Computer Engineering and Applications), 2012, 48(15): 28-34.
- [11] Kong W K, Zhang D, Palmprint texture analysis based on low-resolution images for personal authentication[C]/Pattern Recognition, 2002. Proceedings. 16th International Conference on. Provence, France: IEEE, 2002: 807-810.

- [12] Tamrakar D, Khanna P. Palmprint verification with XOR-SUM Code[J]. Signal, image and video processing, 2015, 9(3): 535-542.
- [13] Pan X, Ruan Q Q. Palmprint recognition using Gabor feature-based (2D) 2PCA[J]. Neurocomputing, 2008, 71(13): 3032-3036.
- [14] Lu J, Zhang E, Kang x, et al. Palmprint recognition using wavelet decomposition and 2D principal component analysis[C]//Communications, Circuits and Systems Proceedings, 2006 International Conference on, Guilin, China: LEEE, 2006: 2133-2136.
- [15] Jia, W., Gao, J., Xia, W. et al. A Performance Evaluation of Classic Convolutional Neural Networks for 2D and 3D Palmprint and Palm Vein Recognition. Int. J. Autom. Comput. 18, 18–44 (2021). <https://doi.org/10.1007/s11633-020-1257-9>
- [16] 孙伟帅. 基于迁移学习的天气图像识别 [J/OL]. 西华大学学报 (自然科学版):1-5[2020-10-11]
- [17] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.