南京大学

科研项目总结报告 2021

院　　系 匡亚明学院

专　　业 计算机科学与技术

题　　目 人脸识别

年　　级 2017 级　学　　号 171180589

学生姓名 李凯旭

指导老师 王树徽

提交日期 2021 年 9 月 17 日

目　　录

1　人脸识别项目背景

1.1　背景概述

随着社会数字化的推进，快速准确的身份识别和验证正在成为迫切的需求。由于传统的身份识别方法（如密码、身份证）存在容易丢失、伪造或盗用的问题，生物特征识别技术开始受到研究者的关注。在生物特征识别中，技术较为成熟的身份验证方式包括指纹，虹膜，声纹，人脸等。其中，人脸识别由于直接方便且容易被用户接受，从而在现实生活中得到广泛研究和应用。目前，在安防、交通、电子商务等领域，人脸识别正在发挥重要作用。

1.2　问题定义

通俗而言，人脸识别的目标是通过给定的视频或图像，自动的在图像中检测人脸。并将人脸数据库中的标号比对，确认图像中人的身份。这里为了后续实验，数据库数据和需要识别的数据均为图像，并且不考虑输入图像不属于任何一张人脸的情况。

Problem 1　给定图像数据库 $D = (num, pic[][])$，其中 num 表示人脸编号，$pic[k][]$ 表示对应人脸编号 k 的多幅人脸图像。对于每个输入的图像 $input$，输出 $input$ 在数据库中对应的人脸编号 id。

因此，当数据库需要识别某张人脸时，数据库中必须存储了当前人脸的其他图像。

1.3　发展历史

早在 20 世纪 50 年代，科学家便已经着手人脸识别的研究。

1.3.1　20 世纪 60 年代

研究者开始探索半自动的人脸识别系统。这一阶段的方法主要利用人脸的几何结构，手工标记人脸的眼睛、嘴部、鼻子等器官位置。程序采用位置特征

匹配的方式来进行识别。这一方法准确率较低，适用于数据库中样本较少的情况。当人脸的姿态、表情等发生变化时，识别精度严重下降。

1.3.2 20 世纪 80 90 年代

Kirby and Sirovich 首次使用主成分分析法（Principal Component Analysis，PCA）方法进行人脸特征提取和处理，得到特征脸（eigenface）。特征脸通过少量数值描述原本复杂人脸的主要特征，在人脸识别取得了不错的效果。这种思想在后续的人脸识别研究中也得到了继承和发展。

1.3.3 20 世纪 90 年代后

随着机器学习理论的发展，学者们相继提出了更多的方法，如支持向量机（Support Vector Machine，SVM）、boosting、流形学习、稀疏编码等方法。Gabor 及 LBP 特征描述子两个在人脸识别领域比较成功的人工设计局部描述也在这一阶段提出。此时，受限场景下人脸识别的准确率已经可以达到 99% 以上，学者们开始关注非受限环境下的人脸识别。

这些经典方法在受限环境下取得了比较好的效果，但是他们难以处理大规模数据集的训练，因此在非受限环境下的表现难以突破。2014 年前后，随着大数据和深度学习的进展，神经网络开始受到重视，在图像分类、语音识别等领域均取得了超过经典方法的效果。香港中文大学的 Sun Yi 等人提出，将卷积神经网络应用于人脸识别，并在 LFW 上第一次识别精度超过人类水平。自此，研究者们不断改进网络结构，增加数据规模，进一步提升了训练效果。

2 研究方法

2.1 人脸检测

人脸识别的先决条件是在图像中进行人脸检测。人脸检测可以视为特殊的物体检测，这在计算机视觉中是一个经典问题。在实际操作中往往分为人脸图像采集与人脸图像检测两个层次。在人脸检测中往往包含如下三个部分：

1. 检测窗口选择
2. 图像特征设计
3. 分类器设计

在识别过程中，图像特征设计对于识别效果至关重要，直接影响后面分类器的分类效果。

2.1.1 窗口选择

这一步是对需要识别的人脸目标进行定位。由于目标可能出现在图片上任意的位置，传统方法采用不同大小的滑动窗口进行穷举。针对不同人脸，采用图像金字塔方法（保持窗口大小，对图像进行缩放）这一经典方法进行处理。该方法会产生大量的冗余，且时间复杂度较高，对于后续特征提取和分类器速度有很大影响。

在后续基于区域的卷积神经网络（Region Based Convolutional Neural Networks，R-CNN）中，窗口选择采用了候选区域（region proposal）的方法，相较于原来的滑动窗口保证了在较少数量的区域框内有较高的召回率，并且获得了较高的窗口质量。这一方法也没有避开对原图像的大量扫描。

2.1.2 特征设计

在人脸检测中，这是最为重要的一步。人脸特征有以下几种设计：

- 基于规则/知识
- 基于模板

- 基于不变特征（经典方法，如肤色，关键点等）
- 基于外观学习（目前主流）

在各类算法中，不变特征和外观学习是较为常见的特征设计，下面简要介绍几种方法：

2.1.2-1 基于肤色

在很多人脸检测和手的跟踪应用中，人的肤色信息是一种非常有效的特征。有很多颜色空间可用来表征肤色，包括 RGB, HSV（色调、饱和度、亮度）等。肤色特征包括简单的阈值特征、颜色直方图或者高斯肤色模型。

缺陷在于：

- 难以处理光照变化时肤色不稳定的情况
- 仅靠肤色进行人脸检测并不充分

2.1.2-2 关键点检测

关键点指图像中位置被准确定义且可以被鲁棒检测的像素点（well-defined position and can be robustly detected）。关键点的特性是与周围节点有着较大的差异，并且具有良好稳定的数学性质。通过追踪这些关键点的位置可以进行人脸检测。关键点定位最终可以转换为矩阵特征值求解问题。

2.1.2-3 局部特征

如局部二值模式算子（Local Binary Patterns，LBP）和哈尔（Haar-like）特征等，他们通过类似卷积核的方式从每个像素点周围的局部空间提取人脸的局部特征。相比于原来复杂的人脸图像，使用局部算子处理后的图像可以更快的建立人脸特征。

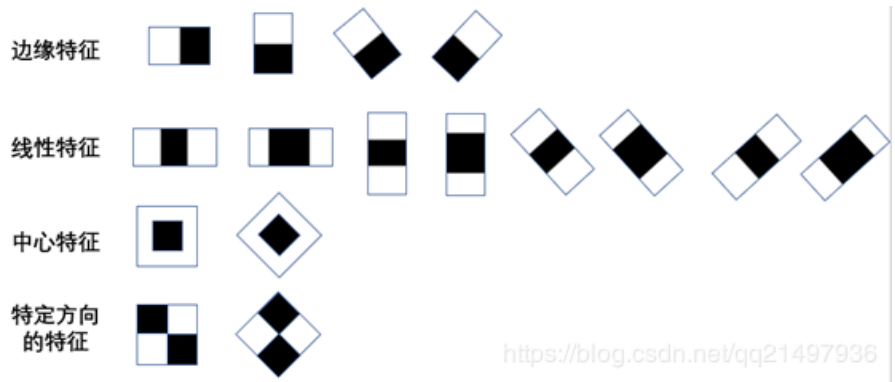

图 2-1: 图像的哈尔特征

2.1.3 分类器设计

早期分类器往往采用模板匹配、基于几何特征或基于子空间的方法，准确率相对较低。这里介绍两种表现较好的方法：基于 AdaBoost 的快速人脸检测和基于区域的卷积神经网络（Region Based Convolutional Neural Networks，R-CNN）方法。

2.1.3-1 基于 AdaBoost 的快速人脸检测

提升 (boosting) 方法是一种常用的统计学习方法, 应用广泛. 在分类问题中, 它通过改变训练样本的权重, 学习多个分类器, 并将这些分类器进行线性组合, 提高分类的性能。AdaBoost 就是这种分类器在人脸识别上应用。它通过将弱分类器进行级联组合来形成一个强分类器。它的基本思想是，对于级联的弱分类器，每个分类器给予前一个分类器分类错误的样本更高的权重。最终对测试数据进行分类时采用弱分类器加权表决的方法。

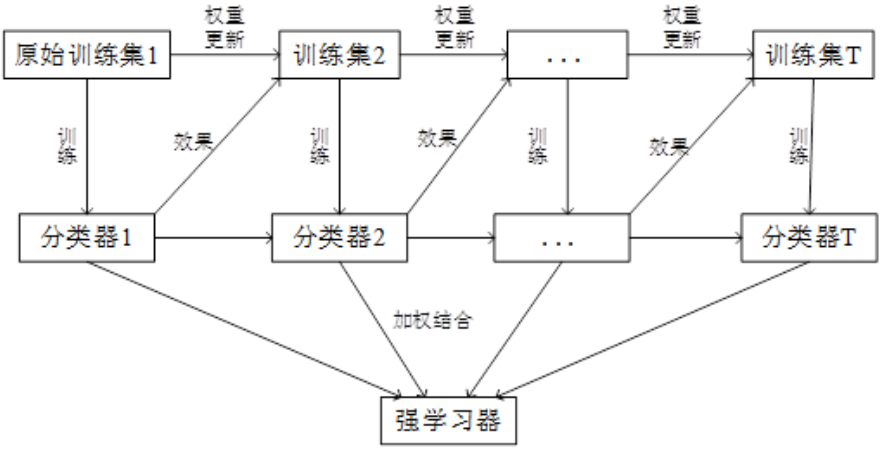

图 2–2: AdaBoost 算法示意

2.1.3-2 RCNN 网络

RCNN 首先将深度学习应用于目标检测，该算法分为 4 个步骤。

- selective search 方法生成 1k 2k 个候选区域
- 对每个区域采用哈尔特征和深度神经网络提取特征
- 对每一个类别设计 SVM 分类器并进行判别
- 使用回归其精细修正候选框的位置

在后续的 Fast-RCNN 方法中，深度神经网络与 SVM 两个阶段进行整合，使用新的网络同时进行分类与回归。在 Faster-RCNN 中，学者进一步改进了神经网络，并采用多任务损失函数精最终精修部分也融入网络训练。

2.2 人脸识别

本次实验项目主要关注人脸识别的过程。此时，已经确认了人脸的矩形框位置，需要根据给定的人脸图片进行人脸识别。这一过程可以分解为如下的步骤：

- 数据预处理
- 数据降维
- 数据分类

2.2.1 预处理

一般情况下，人脸识别需要先将图片裁剪为大小一致的正脸灰度图，并手工对用于训练的图像类别进行标记。部分情况下还需要使用卷积核或者 lbp 算子对图像进行滤波预处理，目的是为了剔除图像中的无用信息，增强数据特征，方便后续处理。

图 2-3: 部分人脸数据进行 lbp 算子处理后的图像

2.2.2 数据降维

进行数据降维的主要原因是过高的数据维数会导致信息过量，由于数据的某些特征反而导致判别性能的下降。降维的优势在于在精简数据规模的同时尽可能的保持原数据本身包含的信息，包括部分拓扑特征。这里主要使用主成分分析（Principal components analysis，PCA）、线性判别分析（Linear discriminant analysis，LDA）流形降维等方法。

2.2.2-1 主成分分析

主成分分析法的观点是保留原数据信息量较大的方向，去除信息量较小的方向。以下图为例：

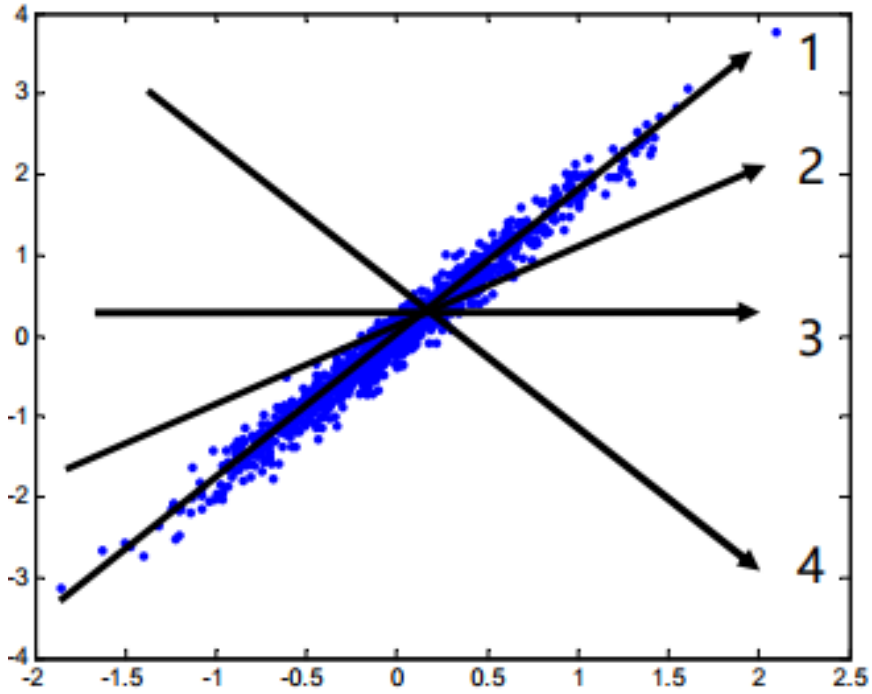

图 2-4: 数据降维的一个实例

不难看出，箭头 1 是较为合理的数据压缩方向。一个合理的思考是，压缩的方向应该与数据最为“分散”的方向一致。直观的理解是，假设需要从原先数据映射至 k 维，首先寻找原数据方差最大的方向，此时数据得到最大程度保留。之后，选择第二个方向，保证与第一次选择结果正交且数据在该方向方差最大。依次类推，得到 k 个相互正交的向量。这 k 个向量称为数据的 k 个主成分。

在实际应用中，用协方差矩阵的特征值与特征向量描述上述过程。特征值的大小表示数据的分散程度，而特征向量的方向则描述了数据分散的方向。实际过程如下：

- 读取样本矩阵 A（m*n，表示 m 个样本 n 个特征），计算样本均值 mA（1 行 n 列）
- 计算样本的协方差矩阵 C（n*n）
- 计算矩阵 C 的前 k 个特征值与对应的特征向量 V（n*k）
- 得到降维后的数据 pcaA = （A-mA）*V

可以看到，中间过程的主要消耗在于计算矩阵 C 的特征分解。当 $n \gg m$ 时，由于协方差矩阵 $C = Z * Z'(Z = A - avg(A) = A - remat(mA))$ 为 $n * n$ 矩阵，可以考虑优先计算 $S = Z' * Z$ 的特征分解并进行转化。

PCA 方法的优势在于：

- 数据重建效果较好
- 主成分正交，消除了原始数据的相互影响
- 计算过程简单，易于实现

PCA 方法的不足在于：

- 仅适用与高斯分布
- 主成分的物理含义不明确，难以直接进行解释
- 方差较小的情况下会损失较多有用的信息，导致效果不佳
- 没有利用样本本身的标签信息，可能产生误差

2.2.2-2 线性判别分析

与主成分分析不同，线性判别分析是一种有监督的降维技术。线性判别分析的直观思想是，数据在投影至低维度时，属于同一类别的点降维后应该尽可能接近，不同类别的点应该尽可能远离。

首先描述二类 LDA 的情况。假设数据集为 $D = \{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$，$y_i \in \{0, 1\}$。其中，$X_1$ 表示第一类，X_2 表示第二类。降维过程可以认为是寻找一个矩阵 w，对每一个样本 x_i 均进行 $w^T x_i$ 的转换。同一类别尽可能接近可以认为是属于同一类的样本在进行转换后类内的协方差尽可能小。可以使用下面的式子表示：

$$conv = w^T \sum_0 w + w^T \sum_1 w \tag{2-1}$$

另一方面，不同类别的数据互相远离可以描述为增大转换后类别中心点之间的距离。如果用 μ_i 表示类别的中心点，那么只需要使得 $\|w^T\mu_0 - w^T\mu_1\|^2$ 尽可能大即可。

综合而言，我们得到了如下的优化目标：

$$argmax(J(w)) = \frac{\|w^T\mu_0 - w^T\mu_1\|^2}{w^T \sum_{X_0} w + w^T \sum_{X_1} w} = \frac{w^T(\mu_0 - \mu_1)(\mu_0 - \mu_1)^T w}{w^T(\sum_{X_0} + \sum_{X_1})w} \tag{2-2}$$

如果我们让 $S_b = (\mu_0 - \mu_1)(\mu_0 - \mu_1)^T$，$S_w = \sum_{X_0} + \sum_{X_1}$，那么优化目标可以简

化为：

$$argmax(J(w)) = \frac{w^T S_b w}{w^T S_w w} \tag{2-3}$$

可以发现正好和广义瑞利熵的形式一致，通过拉格朗日算子法可以转化为求解 $S_w^{-1}S_b$ 的特征值和特征向量问题。

多类 LDA 问题可以用类似的方法推导，只需要修改 S_b 与 S_w 矩阵。其中，S_w 矩阵的推广较为容易，只需要将二类的类内协方差之和修改为 k 类的协方差和即可。然而按照二类方法描述多类问题中的 S_b 矩阵将会非常复杂，这里采用下面的表示：

$$S_b = \sum_{i=1}^{k} N_i(\mu_j - \mu)(\mu_j - \mu)^T \tag{2-4}$$

因此 LDA 方法的过程可以描述为以下步骤：

- 计算矩阵 S_w,S_b,$S_w{}^{-1}S_b$
- 计算 $S_w{}^{-1}S_b$ 的前 d 个特征值与对应的特征向量，构造投影矩阵 w
- 对样本集中的每个样本使用 w 矩阵进行转换

LDA 方法的优势在于：

- 使用了类别信息，
- 更加依赖均值而非方差，在不同类样本均值差距较大时表现较好

LDA 方法的缺陷在于：

- 难以处理非高斯分布的样本
- 最多降维至类别数量 k-1 维
- 存在过拟合的风险
- 当样本的均值较为接近时表现较差

2.2.2-3 流形降维

传统机器学习方法中数据点大多定义于欧式空间。然而实际情况下数据点可能并不如此。流形空间是局部具有欧式空间性质的空间。典型的流形包括球体，弯曲的平面等。流形空间的降维可以认为是保留每个节点的邻居区域内的拓扑关系，更加关注图像的局部特征。

流形空间的局部线性嵌入分布以下步骤：

- 寻找每个样本的邻居集合 X_j
- 对每个样本点的邻居计算该点局部重建的权值向量 w_j
- 根据局部重建的权值向量建立映射后的坐标

2.2.3 数据分类

这里采用支持向量机（Support Vector Machine，SVM）和深度自编码器两种方法。

2.2.3-1 支持向量机

支持向量机的思想来自于对样本的线性分割。以下图为例：

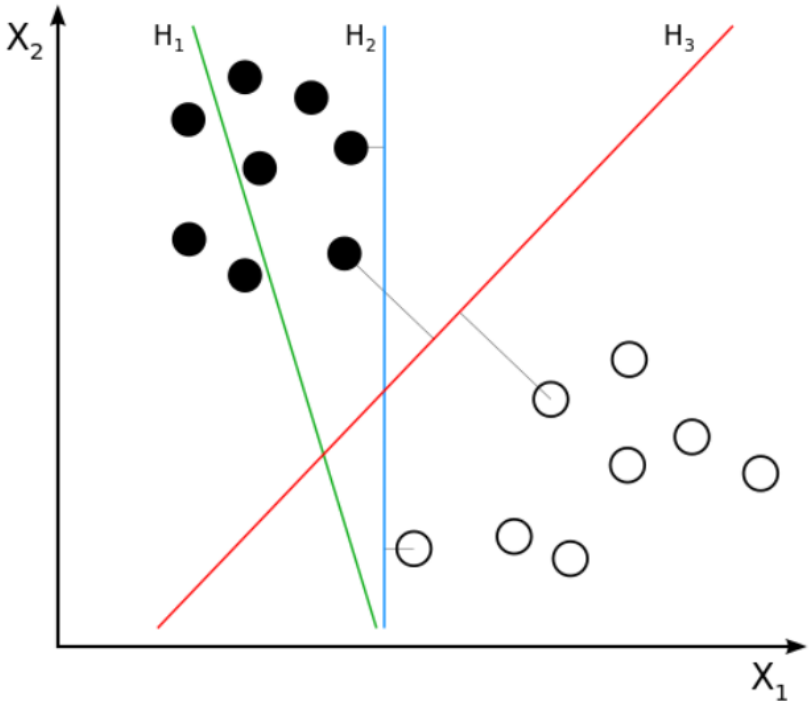

图 2-5: 一个二维线性分割的实例

其中，点的颜色代表类别，H_1, H_2, H_3 代表三个不同的分类方式。直观感觉上，H_3 是最好的分类器。首先它能够将不同类别分割开，同时与各个数据保持了一定的距离，可以容忍测试数据的噪声，泛化能力较好。

假定分类边界为 $f(SV) = w * SV + b = 0$，在二维图像中为直线，在高维情况下表示为一个超平面。两类样本距离最近的样本分别在 $w * SV + b = -t$ 与 $w * SV + b = t$ 上。为了计算的方便，考虑 $t = 1$ 的情况。此时，正负样本之间的距离表示为 $\frac{2}{\|w\|^2}$。根据前面的描述可以知道，我们需要让这个距离尽可能大。

在现实分类问题中，可能存在线性不可分的问题。此时，可以通过核函数将数据映射至高维空间，转化为高维空间中的线性可分问题。对于部分不可分的样本，可以考虑加入容错项来控制。

2.2.3-2 深度自编码器

自编码器本身是一个对标 PCA 和 LDA 的自监督低维嵌入方法。包含两个部分：一个编码器和一个解码器。示意图如下：

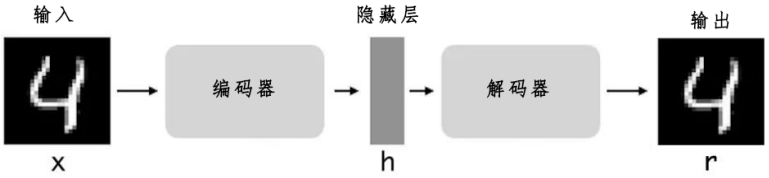

图 2-6: 自编码器

可以看到，输入经由编码器得到隐藏层输入，再由解码器得到对应的重构输出。在理想情况下，输入与重构输出应该完全一致。因此：

编码器 将输入压缩成低维，视为编码函数 $h = f(x)$
解码器 将低维嵌入结果重建为输入，视为解码函数 $r = g(h)$

理想情况下应该尽可能减少 $r = g(f(x))$ 与原输入 x 的差别。当中间隐藏层的维度低于输入时，我们可以认为中间的隐藏层表示 h 是原输入 x 的一个合理的低维嵌入结果。

在深度自编码器中，通过堆叠多个自编码网络层并最终通过 softmax 层进行标签训练可以进行分类。

3 实验过程

在实验中，我们采用 orl 数据库。orl 数据库是一个 42 人的正脸图像集合，每张人脸有 10 个不同的图片。我们将人脸划分为前 5 张与后 5 张，分别作为训练集与测试集。

我们采用如下的步骤进行实验：

- 人脸数据读入
- 进行数据预处理，包括未处理/均值/lbp 直方图
- PCA/LDA 降维
- K-近邻算法/多类 SVM/深度自编码器进行分类

几种方法的过程在前文基本均已介绍。其中多类 SVM 采用 k 个二类 SVM 分类器组合的方法，深度自编码器采用两层自编码器加上 softmax 层。具体参数如下：

读入 根据输入人脸读入数据，每个人脸数据文件夹内前 5 幅用于训练，后 5 幅用于测试

预处理 采用三种预处理方法：（1）不进行处理（2）使用 LBP 图像（3）将原图划分为 15*15 的区域小块，对每个区域使用降维为 59 维后的 uniformLBP 灰度直方图

降维 在输入 42 维时两者均降维至 20 维

下面是分类器的部分数据：

K 近邻分类 采用 $L2$。

SVM 分类 高斯核和线性核。高斯核的参数范围是 $gamma \in 0.001; 0.01; 0.1, c \in [0.001; 0.01; 0.1; 1; 10]$;，挑选其中表现最好的一个记录入表。

自编码器 两层自编码器和一层 softmax。第一层自编码 100 隐藏节点，第二层 50 隐藏层节点。

实验结果如下：

分类器	K-近邻	SVM(线性核)	SVM（高斯核）	自编码器
PCA	87/27/76	80/30/74	83/32/78	82/49/3
LDA	45/6/23	16/2/7	11/4/8	34/3/4
PCA+LDA	90/31/88	87/29/81	86/28/80	89/48/3

表 3-1: 不同分类器下在测试集上的准确率（按照百分比）未处理/LBP/uniform-LBP 直方图

在实验中可以看到，几种数据处理方法在 orl 数据库上的效果。原始数据已经有不错的效果，LBP 图像本身并不适合进行训练，但是处理后的 uniform lbp 直方图很适合进行数据训练。由于 uniform lbp 图像相比原图维数大幅缩小，适合需要快速训练的场景。

而降维方式的对比我们可以看到，PCA 和先 PCA 后 LDA 的效果均不错。但是单独使用 LDA 方法效果却比较糟糕。在实验过程会发现 LDA 对于鸢尾花数据集却有很好的分类效果。同时参考 matlab 运行 LDA 算法的的警告，很可能裸数据在 LDA 内容易产生过拟合的现象，当预先进行 PCA 处理后运行 LDA 应当是能够获得好的分类效果的，目前尝试其他的几个方法都是不合适的。

而三种分类器的对比我们发现几种分类器的表现均不错，但是自编码器很难处理 uniform lbp 直方图数据，表现很差。使用自编码器建议直接使用原图像输入即可。

综合而言，原始数据配合 PCA+LDA 的降维方式加上任意分类器或者 uniform lbp 直方图配合 PCA+LDA 降维加 K 近邻或 SVM 分类器适合 orl 人脸数据的处理。

4 项目总结

在该项目中，我们汇总了人脸识别的重要过程和几个比较经典的人脸识别方法的原理。在实验部分，在 orl 数据库上，基于 PCA 和 LDA 降维以及 SVM 分类器和自编码器，我们尝试重现了几种经典方法并测试了它们的效果。其中，PCA 方法表现较好，而 LDA 方法的表现与输入数据的规模有很大关系，其中最好的方法是先经过 PCA 后在进行 LDA 降维。

5 感悟收获

这次远程项目过去的很快。在过去的本科阶段也接触过不少代码实验，但是大多数都是持续一个学期时间，并且与书本内容有一定的脱节。因为一些原因，往往是课上讲“造轮子”，课下实验都是直接要求使用轮子进行工作。往往只能通过查找资料或者问同学才能完成一些基本要求，对自己的要求也往往是“能跑起来看上去没问题就行”。这次人脸数据处理虽然只有一个月，课程内容跟实验过程却很紧密。并且对于我来说，人脸识别和 matlab 使用都是接触很少的领域。在实验过程中，有些问题也不是“能跑”就可以完事，仅仅要求不出错对于后面的准确率测试有很大的影响。举一个看上去很蠢的例子，刚开始在 matlab 里做实验二三的时候经常会出现几次运行结果不一致或者直接出错的情况。一开始并不清楚是哪里有问题，后来才发现是不清理工作区变量导致某些变量虽然没有及时定义数值，但是 matlab 使用时因为直接调用了工作区里的变量，在开始阶段并不会出错。但是当工作进度向后延伸，这种不注意会带了更多看上去无法理解的 bug，后面就难以及时找到问题所在了。

同时，这次也让我代码风格的重要性。之前的课内实验大多都是给定了框架文件，但是这次项目的代码基本是从零构建。在写结构时不注意代码风格，一个月的项目还能及时回忆起个各个部分，但是如果是半年或者一年呢？因此直观精炼的代码风格和对于自己研究领域的深刻理解在科研中确实是非常重要。我们当时在操作系统课程上，授课老师经常会提到，实验中能够重现的 bug 是好处理的，可以直接锁定位置，但是 os 中由于时序却经常会出现难以重现的 bug，所以经常会强调手写代码的必要。在这次实验中，确实的感受到了在写代码前进行预先思考结构的重要。这次的实验最后我是通过一次重构代码来解决问题，但是未来进行更大的实验时这一点必然是难以实现的，因此清晰的思路和代码风格确实是太重要了。

这次项目加深了我对于人脸识别和人工智能相关知识的理解，也加深了我对于相关领域的兴趣，了解到了科研过程中的一些困难。这次实验虽然并不大，但是在做的时候也遇到了一些困难，看上去理论合理的工作在实际中确实会出现各式各样的问题，也可能产生灰心或者放弃某些部分的想法，科研确实是需要耐心和探索精神的。

总而言之，这次远程项目让我收益良多，也做出了表现不错的实验结果。这次经历让我加深了对计算机的理解，也加强了未来继续在计算机领域研究的信念。

6 附录

这里对于实验文件夹内的文件作用进行简要说明：功能文件：

exp – – – *read_face.m* 读入人脸

– – – *data_process.m* 数据处理

– – – *myPCA.m* PCA

– – – *myLDA.m* LDA

– – – *calculate_embedding* 根据 PCA/LDA 计算测试数据的压缩结果

– – – *multi_SVM* SVM 分类器

– – – *deep_ae_classify.m* 自编码器分类

测试文件：

exp – – – *test_eigenface.m* 输出基于 PCA 的特征脸，*output_eigenface* 下

– – – *test_lbp_img.m* 输出 LBP 图像，*output_LBPimg* 下

– – – *test_acc_pcalda.m* SVM 分类器下的表现

– – – *test_acc_knn.m* K-近邻分类器下的表现

– – – *test_deep_pca/lda/pcalda* 自编码器分类下的表现

– – – *StartGui.m* 根据给定模板制作一个简易 UI，识别单个图像