齊魯工業大學
QILU UNIVERSITY OF TECHNOLOGY

本科毕业设计（论文）

题目： 基于卷积神经网络的性别识别算法研究

学 院 名 称 电子信息学院(大学物理教学部)

专 业 班 级 _____

学 生 姓 名 _____

导 师 姓 名 _____

二〇一九 年 五 月 二十七 日

齐鲁工业大学本科毕业设计（论文）原创性声明

本人郑重声明：所呈交的毕业设计（论文），是本人在指导教师的指导下独立研究、撰写的成果。设计（论文）中引用他人的文献、数据、图件、资料，均已在设计（论文）中加以说明，除此之外，本设计（论文）不含任何其他个人或集体已经发表或撰写的成果作品。对本文研究做出重要贡献的个人和集体，均已在文中作了明确说明并表示了谢意。本声明的法律结果由本人承担。

毕业设计（论文）作者签名：_____

年　　月　　日

齐鲁工业大学关于毕业设计（论文）使用授权的说明

本毕业设计（论文）作者完全了解学校有关保留、使用毕业设计（论文）的规定，即：学校有权保留、送交设计（论文）的复印件，允许设计（论文）被查阅和借阅，学校可以公布设计（论文）的全部或部分内容，可以采用影印、扫描等复制手段保存本设计（论文）。

指导教师签名：_____　　毕业设计（论文）作者签名：_____

年　　月　　日　　年　　月　　日

基于卷积神经网络的性别识别算法研究

学 院 名 称 电子信息学院(大学物理教学部)

专 业 班 级 电子 15-1

学 生 姓 名

学 号

导 师 姓 名

专业技术职务

目　录

摘　要

随着时代的发展和变革，现在人脸检测以及人脸特征识别相关技术成为了目前机器视觉领域中最重要的课题之一，同时也是社会安全、生活的重要关注点。在社会的治安保障、购物支付等方方面面，准确快速地人脸检测与特征识别，是其最主要的技术支撑，也成为了当今各大科技公司最关心的领域。其中性别识别也是应用广泛的一个技术，可以有效地提供给人们更加优质、高效率的统计和服务。

从技术方法上看，在人工智能领域，卷积神经网络由于其局部连接和权值共享的特点，大大地改进了传统全连接参数过多、网络层数限制的缺点，成为主流方法。本文通过对卷积神经网络原理的研究，介绍了图像处理、人脸建模的相关技术概念，且实现了对人脸识别并检测其性别的整套方案。在人脸检测这一环节中，论文使用了经典的基于 Haar 特征的 Adaboost 算法进行检测，通过 opencv 的开源训练级联文件，达到了较高的人脸检测率；在性别识别环节中，文中采用了基于 LeNet 修改的网络架构，通过 Adience 数据集的训练得到了性别识别的模型，通过在网络上随机抓取的 100 张进行测试，得出总体的性别识别率 80%以上。

关键字：卷积神经网络　人脸检测　性别识别

ABSTRACT

With the development and transformation of the times, face detection and face feature recognition technology has become one of the most important topics in the field of machine vision, and it is also an important concern of social security and life. In terms of social security and shopping payment, accurate and rapid face detection and feature recognition are the most important technical support, and have become the most concerned areas of major technology companies today. Among them, gender recognition is also a widely used technology, which can effectively provide people with more high-quality and efficient statistics and services.

From the technical point of view, in the field of artificial intelligence, the convolutional neural network has greatly improved the shortcomings of traditional full connection parameters and network layer limitation due to its local connection and weight sharing characteristics, and has become the mainstream method. Based on the research of the principle of convolutional neural network, this paper introduces the related technical concepts of image processing and face modeling, and implements a complete scheme for face recognition and detection of gender. In the aspect of face detection, the paper uses the classic Haabo-based Adaboost algorithm for detection. Through the opencv open source training cascade file, it achieves a high face detection rate. In the gender recognition link, the paper uses Based on the network architecture modified by LeNet, the model of gender recognition was obtained through the training of Adience dataset. The 100 genders randomly crawled on the network were tested, and the overall gender recognition rate was over 80%.

Keywords: Convolutional Neural Network; Gender Identification; Face Detection

第一章　绪论

1.1 研究背景

随着科学技术的发展，特别是在近十年里面，人工智能领域飞速发展，一系列智能算法应运而出，切实的解决了我们生活中、社会上许多问题。所谓的人工智能，即是一种赋予机器生物(人类)行为与思考的技术，如蚁群算法、粒子群算法、模拟遗传算法等等，但是现在应用范围最广阔、技术最成熟的，便是神经网络算法。

人工智能领域中有着许多的实现方式，均可以通过设计优秀的算法来达到一定的目的，而其中人机交互是其中最直观，也是其现在应用程度最高的一类实现方式。所谓的人机交互，便是人们通过对计算机输入文字、图像、声音地训练以获得优质的模型，解决人们生活中现实存在的问题的方法，由此也诞生了一系列的相关学科——自然语言处理、数字图像处理、数字语音处理等。其中图像类别的应用最为丰富，现在从事这类科研与研发的研究人员和公司也是最多的。因为图像中包含了丰富的信息，单是人脸识别这一领域，便有年龄、性别、民族等极其丰富的信息。

人脸识别技术包含多个任务子分支，首先要对一个图片进行人脸检测。如果这张图片中含有人脸，那么再进一步进行人脸图片的预处理，将图片去噪，重点突出人脸部分。之后进行关键特征提取，从而进行人脸身份识别，其中生物特征识别可包括诸多的特征，根据这些特征的不同，其所应用的技术也不尽相同，如性别检测、年龄检测等。

性别识别在其中是较为简单的一个二分类问题，但是在社会日常生活中，这个课题仍然扮演了十分重要的角色，如下列情境：

1）在社会治安中，公安部门在相关地方搜寻嫌疑犯的情况下，使用性别识别技术可以将人群先进行性别过滤，有助于侦查工作取得较好的效率；

2）在娱乐购物场景下，我们可以使用人脸性别检测技术，对顾客的性别进行检测，实现性别偏向的智能推荐，提高商场环境下的用户满意度；

故性别识别在人类日常生活中有着很重要的实用价值。

1.2 卷积神经网络研究现状

上个世纪六十年代，Hubel 和 Wiesel 的实验揭示了视觉皮质神经元如何编码图

像，帮助我们理解了视觉通路中的神经元如何从视网膜上投射的光图案中提取复杂的信息以构建图像。最初的卷积神经网络要追溯到上个世纪七十年代，建立起现代卷积网络学科的开创性论文为“Gradient-based learning applied to document recognition”[1]，作者为 Yann LeCun，Léon Bottou，Yoshua Bengio，和 Patrick Haffner，其中他们使用梯度下降算法训练了我们今天所熟知的卷积神经网络，并建立了 LeNet，且将其应用在了手写体字母识别上。

但是在此之后，受限于计算机的算力和广泛的数据集支持，卷积神经网络一直停留在小范围的图片处理阶段，使用价值并不高。但 ILSVRC2012 中 Alex 和 Hinton 提出的的 AlexNet[2]开启了卷积神经网络在计算机视觉上大规模应用的先河，在 ImageNet 上的错误率也在人们对网络结构的一步步改进中降低至个位数百分点。

下面给出卷积神经网络的进化图示：

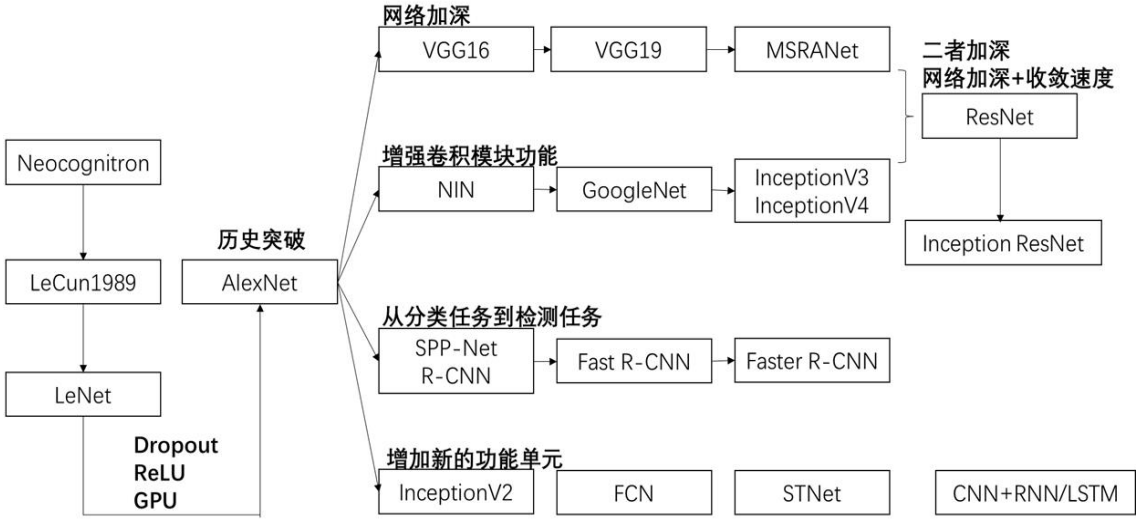

图 1-1　卷积神经网络演进图示

之后在 ILSVRC 各年的竞赛中，各大研究机构提出了改进后的版本，如 Inception V1、GoogLeNet 等，继续降低了识别的错误率，从此卷积神经网络具有了很高的使用价值，催生了一系统机器视觉公司，为我们的生活生产带来了诸多方便。

1.3 国内外研究现状

基于卷积神经网络的性别识别在人脸识别领域在今年成为热点之一，国内外的学者对此进行了相关的研究，取得了不错的效果，产生了一些具有实际应用价值的成果。下面对国内外的研究现状进行一个概括总结：

国外的 Gil Levi 和 Tal Hassner[3]设计了一种基于卷积神经网络的方案，并在 Adience 基准下取得了最佳的成绩；Ahmed 等人直接将图像的 RGB 像素输入了卷积神经网络，在 FRGC 人脸数据路中取得了 95%的识别准确率[4]；Aasma Aslam 等人将图像在 YCbCr 空间中的 Y 分量经过小波变换，达到了较高的识别准确率[5]。

国内的黄勇提出了一种结合 Gabor 特征和基于 Fisher 准则的卷积神经网络的性别识别方法[6]，裴子龙提出了一种使用 PSO 粒子群优化 CNN 网络的方法[7]，陆丽提出了采用 AdaBoost 算法提取脸部整体特征，融合局部与整体特征后用支持向量机进行分类的方法[8]。

1.4 文章组织结构

针对性别识别这一课题，本文从基础理论知识、卷积神经网络原理、人脸检测这几个方面进行研究，搭建出用于性别识别的神经网络，并进行数据集的测试。其中人脸检测章节中重点介绍使用了基于 Haar 的 Adoboost 算法，卷积神经设计部分采用了基于 LeNet 的修改架构，并使用 Adience 数据集进行训练，得到性别识别的模型，最终识别成功率在80%以上。

其中文章安排如下：

第一章：对文章的研究背景、目的和意义进行叙述，说明了卷积神经网络的现状，以及对性别识别领域中国内外的研究现状；

第二章：对基础的理论和工具做了简单的介绍，对人工神经网络基础概念做了叙述，之后对实现语言的选取以及数据集的采用进行了说明；

第三章：对卷积神经网络的原理做了详细的介绍，从其结构到模型，以及求解方法均进行了叙述；

第四章：首先对人脸检测技术进行了说明，详细说明了其原理和实现方法；之后设计了用于性别识别的卷积神经网络，详细地介绍了训练过程中各个参数的变化，

并进行了数据集测试，得出了准确率的数据；

第五章：总结了全文，指出了自己准确率的不足之处，以及可以改进的方向。

第二章 神经网络相关理论与技术综述

本章节首先对人工神经网络的技术与原理做阐述，重点对网络模型、学习方式以及优化算法等进行了研究，并分析了现阶段人工智能与机器学习领域常用的语言选择，且对所选择的框架进行说明。

2.1 人工神经网络技术

人工智能在近几十年高度发展，形成了机器学习、知识推理、智能算法等庞大的学科体系,其中现阶段发展的最好、成果最为丰富的为机器学习一科。机器学习拥有传统计算机行业不具备的优势，即无需明确的编程指定便可以进行学习，且经过网络结构的设计与调参的整理工作，一部分神经网络模型显示出了十分优秀的学习水平。

根据第一章 1.2 节卷积神经网络的研究现状，人工神经网络为 20 世纪 80 年代出现的一种新型的智能技术，其基于生物学中神经网络构造原理，通过模拟人的大脑来构建一个智能化的信息处理系统。神经网络由大量的神经元相互连接而成，其拥有高度的非线性特点，在处理非线性逻辑操作方面显示出了强大的能力，且其作为一种仿生智能模型，拥有并行计算、高容错率的特点，可以进行智能化学习。近年内，人工神经网络在机器视觉、语音识别、自然语言处理等诸多领域发挥了重要作用，下面文章对其基本原理进行说明。

2.1.1 神经元

在生物学中，人脑中的最基本的处理单元为神经元。而对于人工神经网络，其模仿人脑神经元建立了多个彼此连接的结点，即人工神经网络中的神经元。如图 2-1 所示：

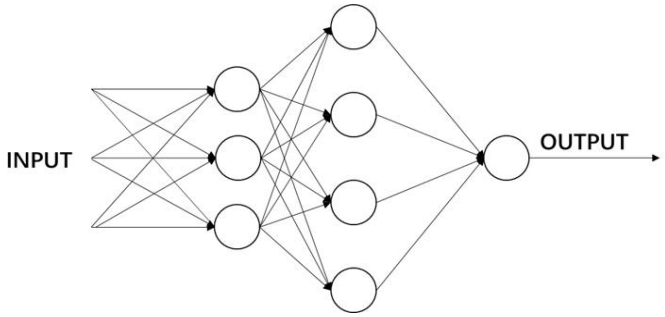

图 2-1　人工神经网络图示

我们假设一个神经元拥有d个输入$x_1, x_2, \cdots, x_d$，用向量$\mathbf{x} = [x_1; x_2; \cdots; x_d]$表示这组输出，并用$Z$来表示输出，我们有：

$$\begin{aligned} Z &= f\left(\sum_{i=1}^{d} w_i x_i + b\right) \\ &= \mathbf{w}^{\mathrm{T}}\mathbf{x} + b \end{aligned} \qquad \text{公式}(2-1)$$

其中$\mathbf{w} = [w_1; w_2; \cdots; w_d] \in \mathbb{R}^d$为$d$维的权重向量，$b \in \mathbb{R}$为神经网络的偏置，非线性函数$f(\cdot)$为激活函数，常用的为 sigmoid 函数：

$$f(x) = \frac{1}{1+e^{-x}}, 0 < f(x) < 1 \qquad \text{公式}(2-2)$$

综上所述，典型的神经元结构如下[9]：

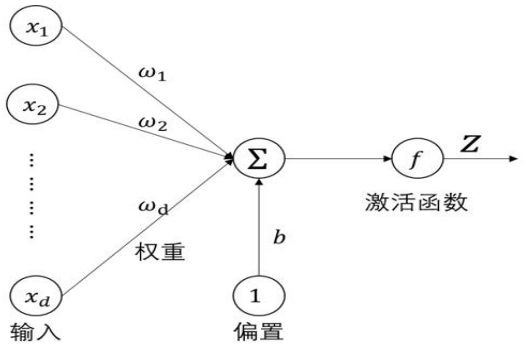

图 2-2　神经元结构图示

2.1.2 常见的人工神经网络模型

单个人工神经元的功能较为简单，要模拟人脑的功能，必须使用多个神经元进行协作以完成复杂的功能。这样通过一定的连接方式或信息传递方式进行协作的神经元可以看作是一个网络，就是神经网络。

根据所处理问题的不同，目前研究者们发明了各种不同的神经网络结构，其中

最常用的为以下三种：

(1)**前馈型神经网络**

在前馈网络中，各层神经元按照接受信息的先后分成不同的组，其中每一组都可以看成一个神经层。除第一层与最后一层外，每一层网络都接受前一层的信息，并传递给下一层。整个网络的传播方向均为正向，无反向传播信息，根据图论的知识可以将其视为一个有向无环路图。

前馈型神经网络中常见的网络类型有：全连接前馈型网络、BP 神经网络、径向基函数 RBF 网络与卷积神经网络等。

(2)**反馈型神经网络**

与前馈神经网络不同，反馈型神经网络不仅仅可以接受其他神经元传来的信号，也可以接受自己传来的信号。根据图论的知识可以将其视为一个无向图或者有向循环图。

反馈型神经网络常见的网络类型有：循环卷积网络、Hopfield 网络、玻尔兹曼机、神经图灵机等。

(3)**图网络**

由于前馈与反馈型的神经网络的输入都可以用向量来表示，而有一部分数据，如知识图谱、社交分子网络等机构只能以图的形式进行处理，故人们设计了基于图结构的神经网络。其节点既可以接收来自相邻结点的数据，也可以接收自己的数据。

图类型神经网络常见的网络类型有：图卷积网络、消息传递网络等。三种网络的构造示意如下图所示[9]：

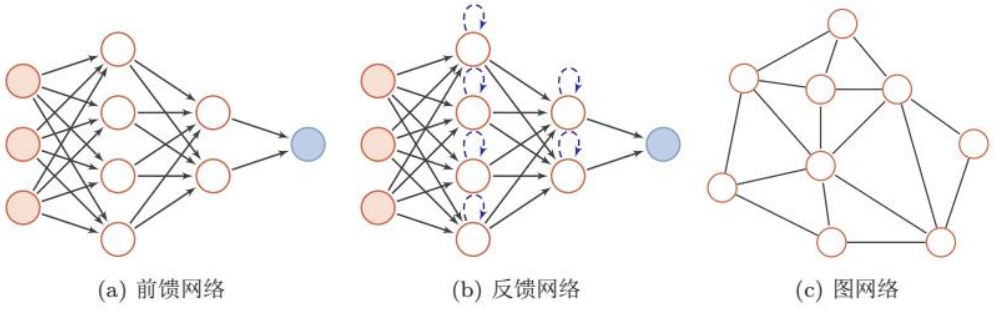

图 2-3　不同构造神经网络图示

2.1.3 神经网络的学习方式

机器学习中根据不同准测的划分，可以将学习算法划分成不同的分类。比如根据激活函数$f(\mathbf{x},\theta)$的不同，我们将其分为线性模型与非线性模型；根据学习准则的不同，可以分为统计方法与非统计方法；以训练样本提供的信息以及反馈方式的不同，我们可以将其分为有监督学习、无监督学习记忆强化学习。其中第三种分类方案也是我们最常使用的方案，下面文章对其分类原理与依据进行介绍。

(1)有监督学习(Supervised Learning)

若在机器学习中我们的目标为建立样本特征x与便签y之间的关系：$y=f(x,\theta)$，且训练中每一个样本均有标签，我们称之为监督学习。且根据其标签类型的不同，又可以将监督学习分为回归与分类两类。

监督学习过程中，我们先将样本数据投入神经网络，得到输出后与期望值进行比较以得到误差信号，再根据误差函数对输入向量的值进行调整，在多次训练后将误差信号收敛到一个可接受的范围，便最终确定了网络的输入向量与网络结构。我们可以将其表示为：

$$\hat{\mathbf{y}}=\underset{\mathbf{y}\in \mathrm{Gen}(\mathbf{x})}{\operatorname{argmin}} f(\boldsymbol{\phi}(\mathbf{x},\mathbf{y}),\theta) \qquad \text{公式}(2-3)$$

常见的监督学习方法有：纠错学习规则等。

(2)无监督学习(Unsupervised Learning)

无监督的学习是指在没有目标训练标签的训练样本中自适应的学习，以获取有价值的信息，这种算法由于没有人为的干预，可以更好的体现出自适应学习的特点。

常见的无监督学习有 Hebb 学习规则、竞争型学习规则等。

(3)强化学习(Reinforcement Learning)

强化学习是一类根据交互来学习的机器学习算法，其根据环境的状态给出自己的判断，收到即使或延时的奖励。网络在与环境的交互中不断学习，以得到最大化的回报与奖励。

在第四章具体的卷积神经网络构造中，论文采取有监督的学习机制进行网络训练。

2.2 网络训练时所用到的基础技术原理

在上节文章介绍了神经网络的相关元素，在本节文章重点讨论在训练网络过程中，所使用的技术原理，为第三章卷积神经网络的原理介绍做铺垫。

2.2.1 学习准则

所谓的学习准则即为制定一种规则，定量的衡量我们训练模型的优劣。网络学习过程中，假设输入空间为$\mathcal{X}$输出空间为$\mathcal{Y}$，训练集$\mathcal{D}=\left\{\left(\mathbf{x}^{(n)},y^{(n)}\right)\right\}_{n=1}^{N}$满足独立同分布，若存在一模型$f(\mathbf{x},\theta^{*}),f:\mathcal{X}\rightarrow\mathcal{Y}$，使得：

$$|f(\mathbf{x},\theta^{*})-y|<\epsilon,\quad\forall(\mathbf{x},y)\in\mathcal{X}\times\mathcal{Y} \qquad \text{公式}(2-4)$$

则称这是一个好的模型，其中y为真实的映射函数。

其中模型的好坏我们可以用期望风险$\mathcal{R}(\theta)$予以衡量:

$$\mathcal{R}(\theta)=\mathbb{E}_{(\mathbf{x},y)\sim p_r(\mathbf{x},y)}\left[\mathcal{L}\left(y,f(\mathbf{x},\theta)\right)\right] \qquad \text{公式}(2-5)$$

其中$p_r(x,y)$为真实概率分布，$\mathcal{L}(y,f(x,\theta))$为损失函数。

常见的损失函数有:

(1)0-1 损失函数

$$\mathcal{L}\left(y,f(\mathbf{x},\theta)\right)=\begin{cases}0 & \text{if } y=f(\mathbf{x},\theta)\\ 1 & \text{if } y\neq f(\mathbf{x},\theta)\end{cases} \qquad \text{公式}(2-6)$$

(2)平方损失函数

$$\mathcal{L}(y,f(\mathbf{x},\theta))=\frac{1}{2}(y-f(\mathbf{x},\theta))^{2} \qquad \text{公式}(2-7)$$

(3)交叉熵损失函数

$$\mathcal{L}\big(\mathbf{y}, f(\mathbf{x},\theta)\big) = -\sum_{c=1}^{C} y_c \log f_c(\mathbf{x},\theta) \qquad \text{公式}(2-8)$$

但由于我们实际上并不能确定真实概率分布与映射函数，所以我们无法计算期望风险，即无法判断模型的准确好坏，故我们使用经验风险进行代替，即训练集上的平均损失。

$$\mathcal{R}_{\mathcal{D}}^{emp}(\theta) = \frac{1}{N}\sum_{n=1}^{N} \mathcal{L}\left(y^{(n)}, f\left(x^{(n)},\theta\right)\right) \qquad \text{公式}(2-9)$$

我们可以确定一组参数θ^*使得上式即经验风险最小，即经验风险最小化原则，

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \mathcal{R}_{\mathcal{D}}^{emp}(\theta) \qquad \text{公式}(2-10)$$

由大数定理可知当$\mathcal{D} \to \infty$时，经验风险接近期望风险：

$$\lim_{n\to\infty} P\left\{\left|\frac{1}{n}\sum_{k=1}^{n} x_k - \frac{1}{n}\sum_{k=1}^{n} E\,x_k\right| < \varepsilon\right\} = 1 \qquad \text{公式}(2-11)$$

但是在实际的训练过程中，我们无法以无限的数据进行训练，而是在数据集中拿出一部分进行训练，一部分进行验证。故经验风险最小化原则经常出现过拟合的现象，即在训练过程中准确率极高但是在非训练数据验证时正确率较低。

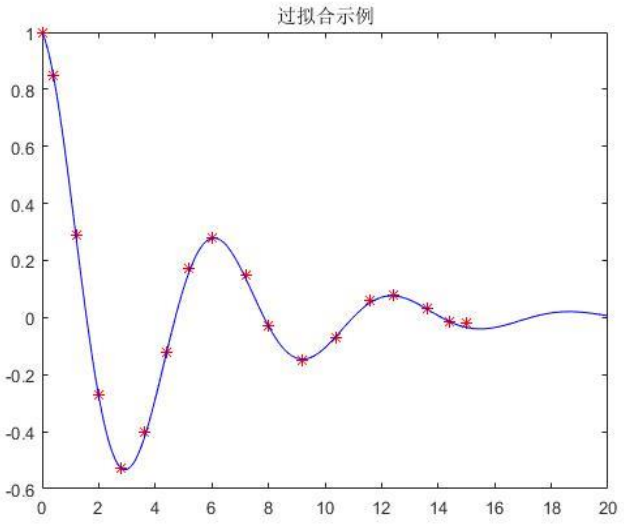

图 2-4　过拟合图示

通常我们选择在经验最小化的基础上，引入正则项以限制模型的能力，达到避免过拟合的目的：

$$\begin{aligned}\theta^{*} &= \underset{\theta}{\operatorname{argmin}}\mathcal{R}_{\mathcal{D}}^{\text{struct}}(\theta) \\ &= \underset{\theta}{\operatorname{argmin}}\mathcal{R}_{\mathcal{D}}^{emp}(\theta) + \frac{1}{2}\lambda \parallel \theta \parallel^{2} \\ &= \underset{\theta}{\operatorname{argmin}}\frac{1}{N}\sum_{n=1}^{N}\mathcal{L}\left(y^{(n)}, f\left(x^{(n)}, \theta\right)\right) + \frac{1}{2}\lambda \parallel \theta \parallel^{2}\end{aligned} \qquad \text{公式}(2-12)$$

2.2.2 优化算法

神经网络模型为非凸模型，我们通常使用**梯度下降算法**来求解其局部最优值。

假设函数$f(x)$于x_t处连续可微，则$f(x)$下降最快的方向即函数于x_t处梯度方向的反方向。由泰勒公式对函数进行一阶展开可得：

$$f(\mathbf{x}_{t+1}) = f(\mathbf{x}_t + \Delta\mathbf{x}) \approx f(\mathbf{x}_t) + \Delta\mathbf{x}^{\mathrm{T}}\nabla f(\mathbf{x}_t) \qquad \text{公式}(2-13)$$

取$\Delta\mathbf{x} = -\alpha\nabla f(\mathbf{x}_t)$，则当$\alpha > 0$且数值较小时，满足$f(\mathbf{x}_{t+1}) < f(\mathbf{x}_t)$,函数呈下降趋

势。

通过迭代公式$\mathbf{x}_{t+1} = \mathbf{x}_t - \alpha_t \nabla f(\mathbf{x}_t), t \geq 0$我们可以生成序列$\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots$，使得：

$$f(\mathbf{x}_0) \geq f(\mathbf{x}_1) \geq f(\mathbf{x}_2) \geq \cdots \qquad \text{公式}(2-14)$$

以逐渐逼近函数的最小值。

梯度下降过程如下图所示，以函数$y = x_1^2 + x_1 x_2 + 3x_2^2$为例：

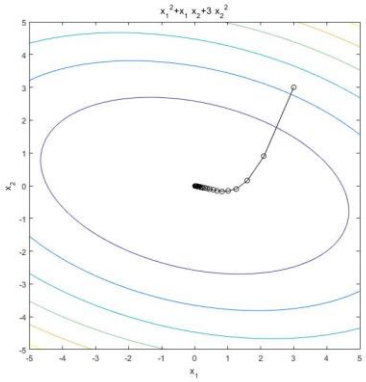

图 2-5　梯度下降原理图示

将梯度下降算法用于本文，求解损失函数$\mathcal{L}$取得极小值时的网络参数θ，具体步骤在 3.1.3 和 3.1.4 小结做详细介绍。

2.3 设计语言与学习框架的对比与选择

正如传统的工具一般，一种编程语言只是实现框架的一种工具，无论使用何种工具均可以达到自己的目标。但是，工具使用起来是否顺手，即是否可以快速、高效率地将自己的想法以清晰、有条理的以代码呈现，仍是十分重要的。下面我们进行常见语言的对比以进行选择。

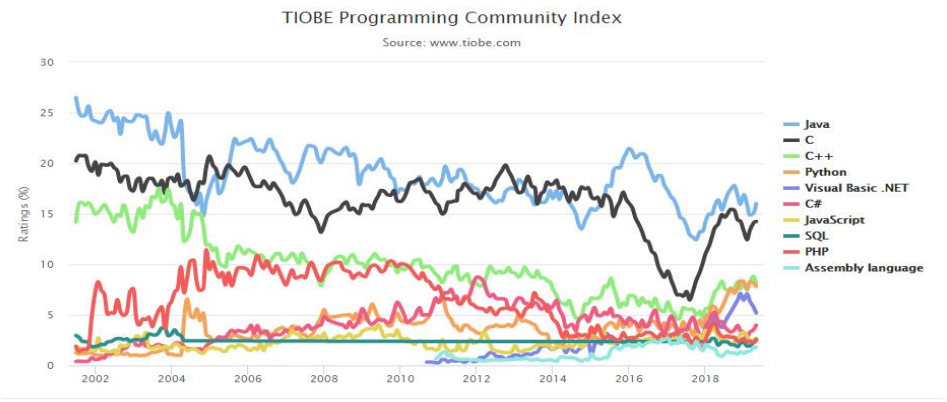

图 2-6　TIOBE 编程语言指数图示

正如 TOBIE index[10]语言指数反映的一样，在榜单前列的为 JAVA、C/C++、Python 等。但 JAVA 更偏重于框架开发，工业级别应用的制作，而 C/C++更贴近底层的开发，虽然前者有其较好的机器学习开源 JAR 类，后者更是所有编程人员的基本功，但其在人工智能领域的方便、受欢迎程度远不及 Python、Matlab。但 Matlab 又为一闭源的平台，更偏重于工具类的调取，其强大的矩阵运算也在一定程度上被 Numpy 开源库所替代。

Python 中携带大量的开源库文件，可以较为简单的调用以实现自己构想的功能，使编程人员将时间精力投入在 AI/ML 相关的算法、框架的设计上，故文章中所设相关的的脚本选择采用 Python 语言进行编写。

在网络架构的编写上，人们更倾向于使用框架进行实现。一方面开源框架提供了诸多的函数调用，不必要再进行重写，由于在参数的学习过程中一般通过反向传播算法进行学习，采用手工计算梯度的算法常陷入低效、鲁棒性低的难题之中；另一方面，使用框架可以更为简单的使用 CUDA 进行加速，提高网络的训练速度。常用的机器学习框架有：Tensorflow 以及其的包装框架 keras、caffe、PyTorch 等，本文选用 Tensorflow 框架进行开发。

Tensorflow 中最重要的两个概念即为其名字中所包含的 tensor(张量)与 flow(流)，且该框架中所有的数据均通过张量的类型予以表示。tensor 在数学、物理领域中为一基础概念，是一种表示在如标量、矢量或其他量之间的多线性函数，其一(p, q)型的张量T严格定义如下：

$$T: V^* \times \cdots \times V^* \times V \times \cdots \times V \mapsto \mathbb{R} \qquad \text{公式}(2-15)$$

其中V为矢量空间，V^*为对偶空间。

在计算机科学中，tensor 可被定义为携带有变换特性的多维数组。本文中，卷积神经网络中所有的数据类型，若未进行变换则均为 tensor 类型。

总结:本章首先对人工神经网络中的基础概念进行了阐述，之后分别介绍了神经网络中学习方式的划分方法，以及常见的人工神经网络结构模型，最后对文章采用的编程设计语言和框架做了简单的介绍。

第三章 卷积神经网络原理

卷积神经网络是本文性别识别网络模型的基础，本章重点从数学角度对其原理进行介绍。

3.1 前馈神经网络原理

3.1.1 前馈神经网络的传播原理

由第二章 2.1.3 小节可知，卷积神经网络为前馈神经网络的一种，下面论文给出前馈神经网络的通用模型，并对其中原理进行研究。下图为前馈神经网络的结构图示[9]：

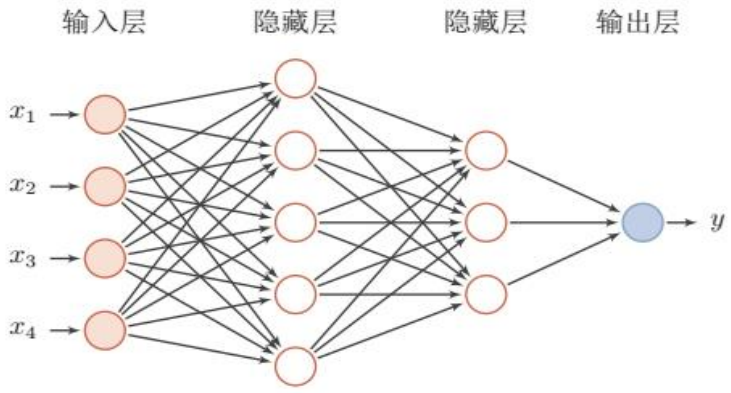

图 3-1　前馈神经网络图示

论文用下面的符号表示一个前馈神经网络：

表 3-1 前馈神经网络符号定义与其含义表示

符号	符号含义
L	神经网络的层数
$m^{(l)}$	第l层神经元的个数
$f_l(\cdot)$	第l层神经元的激活函数
$W^{(l)} \in \mathbb{R}^{m^{(l)} \times m^{l-1}}$	第$l-1$层到第l层的权重矩阵
$\mathbf{b}^{(l)} \in \mathbb{R}^{m^i}$	第l层的偏置
$\mathbf{z}^{(l)} \in \mathbb{R}^{m^l}$	第l层神经元的输入

$\mathbf{y}^{(l)} \in \mathbb{R}^{m^l}$	第l层神经元的输出

故前馈神经网络的信息传播可以用如下公式表示：

$$\mathbf{y}^{(l)} = f_l\left(W^{(l)} \cdot \mathbf{y}^{(l-1)} + \mathbf{b}^{(l)}\right) \qquad 公式(3-1)$$

将x作为第一层的输入，$y^{(L)}$作为整个神经网络的输出，则其各个神经元层之间的状态变化图示如下：

$$\mathbf{x} = \mathbf{y}^{(0)} \to \mathbf{z}^{(1)} \to \mathbf{y}^{(1)} \to \cdots \to \mathbf{y}^{(L-1)} \to \mathbf{z}^{(L)} \to \mathbf{y}^{(L)} = \varphi(\mathbf{x}; W, \mathbf{b})) \qquad 公式(3-2)$$

3.1.2 通用近似定理

通用近似定理从数学角度说明了：常见的连续非线性函数均可以使用前馈神经网络来近似。此问题来源于数学中的 KST 定理，Cybenko[11]于 1989 年证明了通用计算定理，给与神经网络严格的理论依据。简单而言，定理说明只要有一个隐藏层，给予足够多的神经元，前馈神经网络就能一致逼近任意的连续函数。

从数学角度而言，若给定$\delta > 0$和连续函数$f: \Omega \to \mathbb{R}^m$，则存在一权重矩阵$W^* \in \mathbb{R}^{m^1}$, 使得网络神经的输出满足：

$$|F(Z, W^*) - f(Z)| < \delta \qquad 公式(3-3)$$

通用近似定理指出了前馈神经网络有能力去近似任一函数，我们在训练神经网络时，并不知道真实的映射函数，一般采用正则化与经验风险最小化的原则进行学习，即 2.2.1 小节中所叙述的内容。

3.1.3 参数学习

假设在学习过程中我们采用交叉熵损失函数，样本(x, y)的损失函数为：

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = -\mathbf{y}^{\mathrm{T}} \log \hat{\mathbf{y}} \qquad 公式(3-4)$$

上式$\mathbf{y} \in \{0,1\}^C$为标签y的 one-hot 向量表示。

若选择监督学习，其中训练集为$\mathcal{D} = \left\{\left(\mathbf{x}^{(n)}, y^{(n)}\right)\right\}_{n=1}^{N}$，其中的结构化风险函数为：

$$\mathcal{R}(W, \mathbf{b}) = \frac{1}{N}\sum_{n=1}^{N}\mathcal{L}\left(\mathbf{y}^{(n)}, \hat{\mathbf{y}}^{(n)}\right) + \frac{1}{2}\lambda \parallel W \parallel_F^2 \qquad \text{公式}(3-5)$$

其中$\parallel W \parallel_F^2$一般使用 Frobenius 范数：$\parallel W \parallel_F^2 = \left(\sum_{i=1}^{n}\sum_{j=1}^{n}\left|w_{ij}\right|^2\right)^{\frac{1}{2}}$。

当我们有了足够的样本和以及学习准则后，网络参数便可以通过梯度下降算法来进行学习更新，其中第l层的参数更新方式为：

$$\begin{aligned} W^{(l)} &\leftarrow W^{(l)} - \alpha\frac{\partial \mathcal{R}(W, \mathbf{b})}{\partial W^{(l)}} \\ &= W^{(l)} - \alpha\left(\frac{1}{N}\sum_{n=1}^{N}\left(\frac{\partial \mathcal{L}\left(\mathbf{y}^{(n)}, \hat{\mathbf{y}}^{(n)}\right)}{\partial W^{(l)}}\right) + \lambda W^{(l)}\right) \\ \mathbf{b}^{(l)} &\leftarrow \mathbf{b}^{(l)} - \alpha\frac{\partial \mathcal{R}(W, \mathbf{b})}{\partial \mathbf{b}^{(l)}} \\ &= \mathbf{b}^{(l)} - \alpha\left(\frac{1}{N}\sum_{n=1}^{N}\frac{\partial \mathcal{L}\left(\mathbf{y}^{(n)}, \hat{\mathbf{y}}^{(n)}\right)}{\partial \mathbf{b}^{(l)}}\right) \end{aligned} \qquad \text{公式}(3-6)$$

3.1.4 反向传播算法

在上一节中，我们说明了使用梯度下降算法来更新参数。但是如果根据链式法则逐一计算的话，不仅工作量大而且易出错，十分低效，在此我们通常使用反向传播算法来高效的计算梯度以更新参数。

现以第l层的参数权重W^l与阈值b^l为例，因 3.1.3 小节中的$\frac{\partial \mathcal{L}(\mathbf{y}, \hat{\mathbf{y}})}{\partial W^{(l)}}$涉及向量矩阵微分，故我们先行对其偏导进行计算，根据链式法则有：

$$\frac{\partial \mathcal{L}(\mathbf{y},\hat{\mathbf{y}})}{\partial w_{ij}^{(l)}} = \frac{\partial \mathbf{z}^{(l)}}{\partial w_{ij}^{(l)}} \frac{\partial \mathcal{L}(\mathbf{y},\hat{\mathbf{y}})}{\partial \mathbf{z}^{(l)}} \qquad \text{公式}(3-7)$$
$$\frac{\partial \mathcal{L}(\mathbf{y},\hat{\mathbf{y}})}{\partial \mathbf{b}^{(l)}} = \frac{\partial \mathbf{z}^{(l)}}{\partial \mathbf{b}^{(l)}} \frac{\partial \mathcal{L}(\mathbf{y},\hat{\mathbf{y}})}{\partial \mathbf{z}^{(l)}}$$

上式中的第二项为误差项，根据书籍[9]中的推导，偏导数$\frac{\partial \mathbf{z}^{(l)}}{\partial w_{ij}^{(l)}}$、$\frac{\partial \mathbf{z}^{(l)}}{\partial \mathbf{b}^{(l)}}$以及误差项$\frac{\partial \mathcal{L}(\mathbf{y},\hat{\mathbf{y}})}{\partial \mathbf{z}^{(l)}}$的结果如下所示：

$$\frac{\partial \mathbf{z}^{(l)}}{\partial w_{ij}^{(l)}} = \left[0,\cdots,a_j^{(l-1)},\cdots,0\right] \in \mathbb{R}^{m^{(l)}} \qquad \text{公式}(3-8)$$

$$\frac{\partial \mathbf{z}^{(l)}}{\partial \mathbf{b}^{(l)}} = \mathbf{I}_m(l) \in \mathbb{R}^{m^{(l)} \times n^{(l)}} \qquad \text{公式}(3-9)$$

$$\delta^{(l)} = f_l'\left(\mathbf{z}^{(l)}\right) \odot \left(\left(W^{(l+1)}\right)^{\mathrm{T}} \delta^{(l+1)}\right) \qquad \text{公式}(3-10)$$

误差项$\delta^{(l)}$反映了最终损失对于第l层神经元的敏感程度，亦间接反映了不同层神经元的贡献程度。由上述三式可以看出，我们可以从第$l+1$层的误差项得出第l层的误差项，即反向传播原理。

在计算出了上述三个偏导数之后，我们可以将第l层的参数值表达如下：

$$\frac{\partial \mathcal{L}(\mathbf{y},\hat{\mathbf{y}})}{\partial W^{(l)}} = \delta^{(l)}\left(\mathbf{a}^{(l-1)}\right)^{\mathrm{T}} \qquad \text{公式}(3-11)$$

$$\frac{\partial \mathcal{L}(\mathbf{y},\hat{\mathbf{y}})}{\partial \mathbf{b}^{(l)}} = \delta^{(l)} \qquad \text{公式}(3-12)$$

至此，我们可以将看基于误差反向传播算法的前馈网络的训练过程总结如下：

表 3-2 前馈网络的训练过程

基于误差反向传播算法的前馈网络的训练过程
1，前馈计算每一层的的输入z_i与输出a_i;

2，反向传播每一层的误差项δ_i;

3，利用误差项计算每个神经层的偏导数更新参数;

3.2 卷积神经网络结构和模型

3.2.1 卷积神经网络的改进之处

在全连接前馈网络中，假设第l层有$n^{(l)}$个神经元，第$l-1$层有$n^{(l-1)}$个神经元，则权重矩阵有$n^{(l)} \times n^{(l-1)}$个参数。在n特别大的时候，网络训练起来会十分麻烦，效率非常低下。

我们为解决这一问题，使用卷积来替代全连接，假设第l层的输入为z^l为第$l-1$层的输出a^{l-1}与滤波器W^l的卷积，则有：

$$\mathbf{z}^{(l)} = \mathbf{w}^{(l)} \otimes \mathbf{a}^{(l-1)} + b^{(l)} \qquad \text{公式}(3-13)$$

其中$\otimes$代表卷积运算，W^l为可学习的权重，b^l为可学习的偏置。根据卷积的数学性质，卷积层有两个十分重要的性质：

(1)局部连接

模仿人类的视觉系统，在卷积神经网络中，卷积层中第l层的每一个神经元只与第$l-1$层的部分神经元相连接，连接数值取决于卷积核的大小，若将其设为m，则连接数可以表示为$n^l \times m$。

(2)权重共享

卷积神经网络中作为参数的卷积核W^l对于第l层的所有神经元均相同，如下图 3-2 所示，相同颜色的线的权重一致。

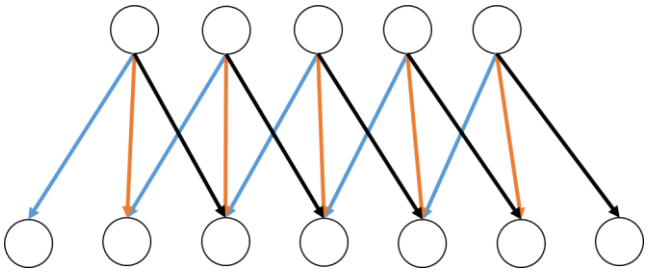

图 3-2　卷积神经网络权值共享图示

3.2.2 卷积网络中各个层的概念与原理

在卷积神经网络中，通常含有三种类型的隐藏层，分别为卷积层、池化层以及全连接层。由于全连接网络与常见的前馈网络一致，故下面重点对这卷积层、池化层这两种类型的隐藏层进行介绍：

(1)卷积层

卷积层的作用为提取一个局域区域的特征，在第一层神经网络中提取一些边缘、线条，在更多层的卷积层中可以提取更加高级的特征。其中一个重要的概念为 feature map：feature map 为在一幅图像或者其他的 feature map 经过卷积提取的特征。

由于图像为二维结构，所以我们构建三层的神经层予以表示，其中大小可以表示为高度M宽度N深度D，即为D个$M \times N$大小的 feature map。在输入层中，若图像为彩色图像，以 RGB 颜色空间表示，则有一个特征映射，深度$D = 3$。

我们可以将卷积神经网络卷积层作用用一般性的结论予以表示，根据文献[9]所示，假设输入 feature map sets 为X，输出 feature map sets 为Y，卷积核为W，卷积层净输入为Z，输出为Y，则有：

$$Z^p = W^p \otimes X + b^p = \sum_{d=1}^{D} W^{p,d} \otimes X^d + b^p \qquad \text{公式}(3-14)$$

$$Y^p = f(Z^p) \qquad \text{公式}(3-15)$$

其中$\mathbf{X} \in \mathbb{R}^{M \times N \times D}$，且其中每个 slice 矩阵$X^d \in \mathbb{R}^{M \times N}$为一输入 feature map，$1 \le d \le D$；$\mathbf{Y} \in \mathbb{R}^{M' \times N' \times P}$，且其中每个 slice 矩阵$Y \in \mathbb{R}^{M' \times N'}$为一输出 feature map，

$1 \le p \le P$，$\mathbf{W} \in \mathbb{R}^{m \times n \times D \times P}$。卷积层从输入 feature map X到输出 feature map Y的计算过程图示如下：

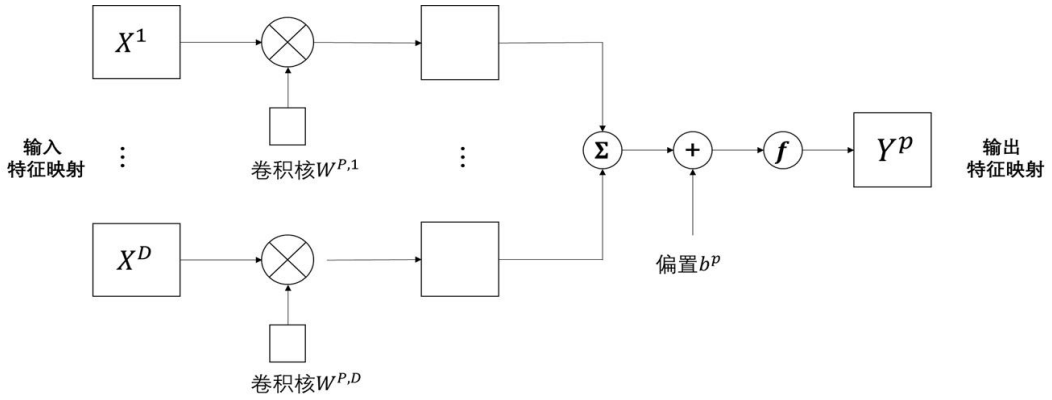

图 3-3　卷积神经网络计算过程图示

⑵池化层

池化层又称汇聚层、子采样层，其作用为降低特征维度，减少参数数量，避免过拟合。

与卷积层类似，我们给出池化层的一般性结论，假设池化层的输入 featurem map 为$\mathbf{X} \in \mathbb{R}^{M \times N \times D}$，将$x \in X^d$划分为多个区域$R_{m,n}^d$，$1 \le m \le M'$，$1 \le n \le N'$。我们在池化层中常用的汇聚函数有三种：

1)Maximum Pooling，即取一个区域内所有神经元的最大值。

$$Y_{m,n}^d = \max_{i \in R_{m,n}^d} x_i \qquad \text{公式}(3-16)$$

2)Mean Pooling，即取一个区域内神经元的平均值。

$$Y_{m,n}^d = \frac{1}{\left|R_{m,n}^d\right|} \sum_{i \in R_{m,n}^d} x_i \qquad \text{公式}(3-17)$$

3)L2 Pooling，即取一个区域内神经元的平方和的平方根。

$$Y_{m,n}^{d}=\sqrt{\sum_{i\in R_{m,n}^{d}}x_{i}^{2}} \qquad \text{公式}(3-18)$$

我们以 Maximun Pooling 的图示[12](图 3-4)为例进行演示，其余的汇聚函数的映射方式也是大致如此：

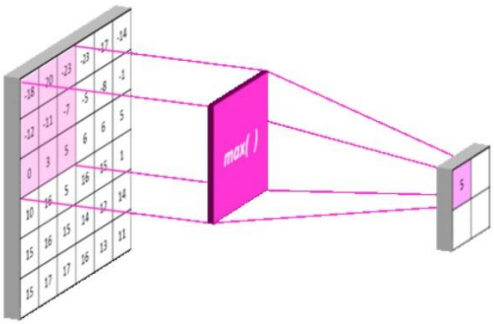

图 3-4　Maximun Pooling 计算图示

3.3 卷积神经网络的参数学习

与常见的前馈神经网络一致，卷积神经网络的训练过程也可以使用基于梯度下降的误差反向传播算法予以完成。

如 3.2 小节所叙述，卷积神经网络一般含有卷积层、池化层、全连接层这三种类型的隐藏层，但是其卷积核、偏置两个重要参数只在卷积层中涉及，故我们只需要计算卷积层之中的梯度。

假设第$l-1$层的输入 Feature Map sets 为$X\in\mathbb{R}^{M\times N\times D}$，第$l$层的 Feature Map sets 净输入为$\mathbf{Z}^{(l)}\in\mathbb{R}^{M'\times N'\times P}$。则第$l$层第$p$个 Feature Map 净输入为：

$$Z^{(l,p)}=\sum_{d=1}^{D}W^{(l,p,d)}\otimes X^{(l-1,d)}+b^{(l,p)} \qquad \text{公式}(3-19)$$

其中损失函数关于第l层卷积核、偏置的偏导数如下：

$$\frac{\partial \mathcal{L}(Y,\hat{Y})}{\partial W^{(l,p,d)}} = \frac{\partial \mathcal{L}(Y,\hat{Y})}{\partial Z^{(l,p)}} \otimes X^{(l-1,d)} = \delta^{(l,p)} \otimes X^{(l-1,d)} \quad \text{公式}(3-20)$$

$$\frac{\partial \mathcal{L}(Y,\hat{Y})}{\partial b^{(l,p)}} = \sum_{i,j} \left[\delta^{(l,p)}\right]_{i,j} \quad \text{公式}(3-21)$$

与其他前馈神经网络相同，卷积层每一层的卷积核、偏置均依赖于所在层的误差项。而第l层卷积层中第d个 Feature Map 的误差项的计算与传播更新方式如下所示：

$$\begin{aligned}\delta^{(l,d)} &\triangleq \frac{\partial \mathcal{L}(Y,\hat{Y})}{\partial Z^{(l,d)}} \\ &= \frac{\partial X^{(l,d)}}{\partial Z^{(l,d)}} \cdot \frac{\partial \mathcal{L}(Y,\hat{Y})}{\partial X^{(l,d)}} \\ &= f_l'\left(Z^{(l)}\right) \odot \sum_{p=1}^{P} \left(\text{rot180}\left(W^{(l+1,p,d)}\right) \widetilde{\otimes} \frac{\partial \mathcal{L}(Y,\hat{Y})}{\partial Z^{(l+1,p)}}\right) \\ &= f_l'\left(Z^{(l)}\right) \odot \sum_{p=1}^{P} \left(\text{rot180}\left(W^{(l+1,p,d)}\right) \widetilde{\otimes} \delta^{(l+1,p)}\right)\end{aligned} \quad \text{公式}(3-22)$$

3.4 卷积神经网络处理图片的优势

在处理图片这种信息的时候，传统的全连接网络会存在两个问题：

(1)**参数过多**:如果图像大小为$100 \times 100 \times 3$，即长度宽度为 100，3 个颜色通道(RGB),则在第一个隐藏层即有 30000 个互相独立的连接，且每个连接都有自己的参数。这样致使网络训练量过大，也非常容易过拟合；

(2)**局部不变形特征**:自然界中物体常常具有局部不变性，在旋转、平移、尺度变换中不损失语义信息，但是全连接网络常常提取不到这些性质。

而卷积神经网络与传统的全连接神经网络对比，有以下诸多优势：

(1)在神经网络中输入的图像是二维结构，而卷积神经网络中各层神经元也是二维平面，两者可以在拓扑结构上直接相连；

(2)权值共享减少了诸多的训练参数，并且简化了网络的结构，极大程度上降低了运算的时间与空间复杂度；

(3)卷积神经网络中卷积层与池化层两层结合进行特征提取的方式，对输入样本有较高的畸变容忍能力。

总结:本章主要讲解了卷积神经网络的原理，包括通用的前馈网络模型与卷积神经网络结构本身。首先将前馈网络的数学模型用严格的数学理论进行了推导，给出了其通用的模型原理；再介绍了卷积神经网络相对于传统前馈神经网络的改进之处；之后探讨了卷积神经网络的参数学习过程；最后介绍了卷积神经网络在进行图片处理时候的独特优势，为第四章的性别识别模型打下理论基础。

第四章 卷积神经网络对人脸的性别识别

在使用卷积神经网络实现性别识别过程中，大致分为两大步骤，一为人脸检测，二为性别识别。故本章在第一节简要介绍人脸监测的相关技术与原理，并且给出仿真结果；在本章的二、三、四节进行具体的性别识别程序实现。

4.1 人脸检测相关技术及仿真

人脸检测技术是一种检测图片中是否含有(多个)人脸的算法，由于人脸在自然环境下存在着诸多的复杂情景，故采取一个具有鲁棒性的方法是十分必要的。人脸检测有基于先验知识、统计模型、人脸特征等多种方法，其中统计模型是现在的主流方向，本文采用基于统计模型的 Adaboost 方法来进行人脸检测。

4.1.1 基于 Haar 特征的 Adaboost 人脸检测方法

本方法来源于 2001 年 Viola 与 Jones 于文献[13]中提出的方法，整个人脸检测方法包括大致三个步骤，分别为 Haar-like 特征提取、基于 Adaboost 的分类器迭代训练。下面对其进行简单的介绍：

(1)Haar-like 特征极其计算

Haar-ike 定义为图像之中相邻像素和的差，在本文中可知它的特征值反映的是人脸图像的灰度变化情况，其特征大致一般分为四种类型：边缘特征、线性特征、对角线特征以及中心特征[13]。如下图所示：

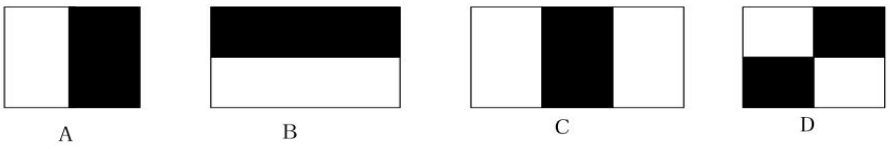

图 4-1　Haar 特征图示

在人脸之中，Haar 特征可以用下图[13]清晰地体现出来：

图 4-2　人脸 Haar 特征图示

在图像预处理形成灰度图像之后，我们可以将 Haar-like 特征值定义为图像之中一块区域内黑色像素之和与白色像素之和的差值，并且可以用积分图来提高其计算速度。

假设$A(x,y)$为点(x,y)左上角所有像素之和，$ii(x,y)$为(x,y)于积分图中的值，$i(x,y)$为于(x,y)这点的像素值，$t(x,y)$为点(x,y)纵坐标像素值之和。故我们有 Haar 积分图算法如下：

$$\mathrm{t(x,y)=t(x,y-1)+i(x,y)} \qquad \text{公式}(4-1)$$

$$\mathrm{ii(x,y)=ii(x-1,y)+t(x,y)} \qquad \text{公式}(4-2)$$

(2)Adaboost 算法训练分类器

Adaboost(Adaptive Boosting)即为自适应增强算法，其原理为迭代运算，由多个弱级联器组成强级联器进行检测。其理论基础为：若每一个弱级联器的分类水平都优于随机猜测的水平，将弱级联器的数量增加至无穷大，便可以得到误差率接近于零的强级联器。Adaboost 算法伪代码如下所示：

表 4-1 Adaboost 算法流程

Adaboost 算法伪代码
1.给定$(x_1,y_1),\ldots,(x_m,y_m)$，其中$x_i \in X, y_i \in Y=\{-1,+1\}$
2.初始化权重$D_1(i)=1/m$

3. $for\ t = 1, \ldots, T$:

I.使用初始化权重D进行训练

II.得到弱分类器$h: X \to \{-1, +1\}$及其错误率其中$\epsilon = \Pr_{i \sim D_t}[h_t(x_i) \neq y_i]$

III.计算$\alpha = \frac{1}{2}\ln\left(\frac{1-\epsilon}{\epsilon}\right)$

IV.更新权重$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_i} & \text{if } h_t(x_i) = y_i \\ e^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \end{cases}$

最后输出强级联器:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t\, h_t(x)\right) \qquad \text{公式}(4-3)$$

Adaboost 算法应用即为：将 Haar 特征向量输入强级联器中，结果为正，即为人脸。

4.1.2 基于 opencv 的人脸检测及其仿真结果

opencv 为一开源机器视觉库，包含着一系列的 C 函数与 C++类，实现了一系列常用的机器视觉算法；并且提供了 Matlab、Python、Ruby 等其他编程语言的接口。优秀的算法实现与开源精神，使得 opencv 成为了在机器视觉中使用最为广泛应用之一。

在此我们参考 opencv 中 Adaboost 算法源码，实现人脸检测。下以弱分类器源码[15]为例，进行解读：

```
1.  typedef struct CvCARTClassifier
2.  {
3.     CV_CLASSIFIER_FIELDS()
4.      int  count;                          //弱分类器数量
5.      int*compidx;                         //对应特征
6.      CvTHaarFeature* feature;             //特征值
7.      CvFastHaarFeature* fastfeature;
8.      float*threshold;                     //阈值
9.      int*left;                            // 左子节点
10.     int*right;                           // 右子节点
11.     float*val;                           //输出
12. }CvCARTClassifier;
```

此结构体定义的 CvCARTClassifier 即构造了一个二叉决策树，体现在机器学习中即为一个预测模型，指向目标对象的属性。按照 Adaboost 算法，多个弱分类器按照二叉决策树模型连接，构成强分类器。

在此我们利用 opencv 中训练完成的强分类器[16]进行人脸检测，人脸检测结果如下：

图 4-3　人脸检测图示

我们可发现在这张著名的照片中，与会者 29 人中有 27 人被正确检测出，人脸检测正确率为93%，达到了较高的水平。在本章的后续章节，将在本节讨论的基础上进行后续的展开工作。

4.2 用于性别识别的卷积神经网络结构设计

对于论文的性别预测而言，为一个二分类问题。本文中采取 Gil Levi、Tal Hassner[3]提出的网络结构，其为 LeNet 卷积神经网络的修改版本，且仅含三个卷积层和两个具有较少神经元的全连接层。具体的结构如下所示：

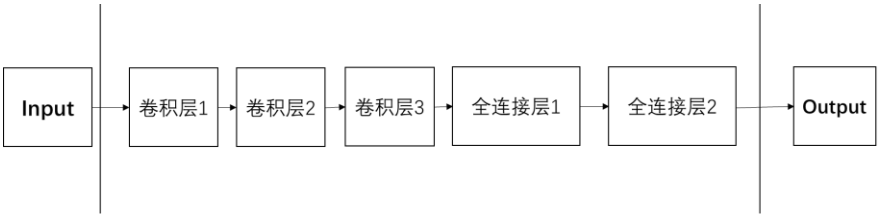

图 4-4　卷积神经网络构造图示

这种具有较少结构的设计可以有效的避免过拟合现象，并且在训练速度较为快速。下面结合模型的 python 示意代码具体说明模型各层参数与功能：

(1)**卷积层 1**(Convolutional Layer 1)

```
conv1 = convolution2d(images, 96, [7,7],[4,4],padding='VALID',...)
pool1 = max_pool2d(conv1, 3, 2,...)
norm1 = tf.nn.local_response_normalization(pool1, ...)
```

第一层卷积层中采用 96 个卷积核，且每个卷积核的参数大小为:$3*7*7$；补 0 策略为只执行有效卷积，对边界数据不处理；激活函数默认为 ReLU，即:

$$\mathrm{ReLU}(x)=\begin{cases}x & \text{if } x>0\\ 0 & \text{if } x\le 0\end{cases} \qquad \text{公式}(4-4)$$

其池化层执行 Maximum Pooling，且池化的大小为$3*3$，strides 为 2。且其后添加了一个局部响应归一化层，目的为提高网络的泛化能力，其公式[17]为：

$$b_{x,y}^{i}=\frac{a_{x,y}^{i}}{\left(k+\alpha\sum_{j=\max\left(0,i-\frac{n}{2}\right)}^{\min\left(N-1,i+\frac{n}{2}\right)}\left(a_{x,y}^{j}\right)^{2}\right)^{\beta}} \qquad \text{公式}(4-5)$$

(2)**卷积层 2**(Convolutional Layer 2)

```
conv2 = convolution2d(norm1, 256, [5, 5], [1, 1], padding='SAME', ...)
pool2 = max_pool2d(conv2, 3, 2, ...)
norm2 = tf.nn.local_response_normalization(pool2, ...)
```

第二层卷积层中选择使用 256 个卷积核，且每个卷积核的参数大小为$96*5*5$；补 0 策略为保留边界处的卷积结果。

池化层与局部响应归一化层参数与第一层卷积层的参数一致。

(3)**卷积层 3**(Convolutional Layer 3)

```
conv3 = convolution2d(norm2, 384, [3, 3], [1, 1],padding='SAME',...)
```

```
2.  pool3 = max_pool2d(conv3, 3, 2,...)
3.  flat = tf.reshape(pool3, [-1, 384*6*6], name='reshape')
```

第三层卷积层中选择使用 384 个卷积核，且每个卷积核的参数大小为$256 * 3 * 3$；补 0 策略为保留边界处的卷积结果。

池化层与第一层卷积层的参数一致；而第三层的 tf.reshape 函数为一变换函数，作用为将张量 tensor 变换为参数 shape 的形式。

(4)**全连接层 1**(Full connected Layer 1)

```
1.  full1 = fully_connected(flat, 512,...)
2.  drop1 = tf.nn.dropout(full1, pkeep, name='drop1')
```

第一层全连接层神经元个数为 512。

dropout 层为训练样本模型时，训练样本过少而防止过拟合加入的一层神经元。其作用为在训练更新参数的过程中，按照一定概率随机断开输入神经元，其原理可以下面公式予以表示：

$$y = f(Wx) \circ m, m_i \sim Bernoulli(p) \qquad \text{公式}(4-6)$$

Dropout 层网络结构图 4-5 所示：

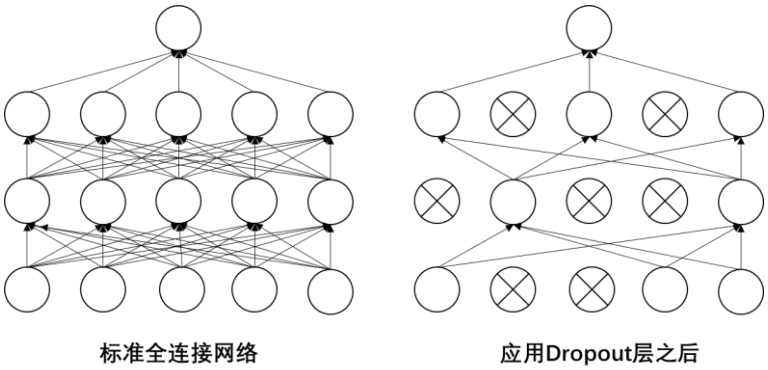

图 4-5　Dropout 层图示

(5)**全连接层 2**(Full connected Layer 2)

```
1.  full2 = fully_connected(drop1, 512,...)
2.  drop2 = tf.nn.dropout(full2, pkeep, name='drop2')
```

第二层全连接层，其神经元个数也是 512 个。

(6)**输出层**(Output)

对于本文，性别识别为二分类问题，故其输出神经元个数为 2。
综上所述，其网络结构可以由下图表示：

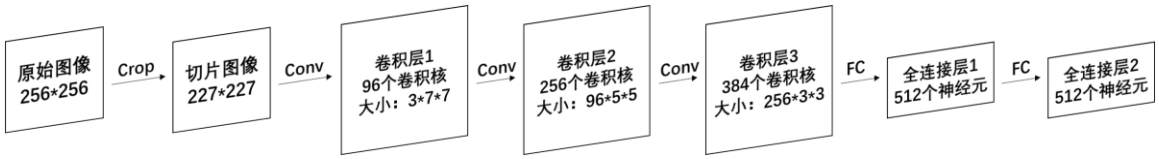

图 4-6　具体网络参数变化图示

4.3 卷积神经网络训练过程

对 4.2 节提出的神经网络进行训练，论文使用的人脸图像数据为 Adience data[18]，论文训练次数设置为 30000 次，具体训练参数变化如下图所示：

(1)学习率

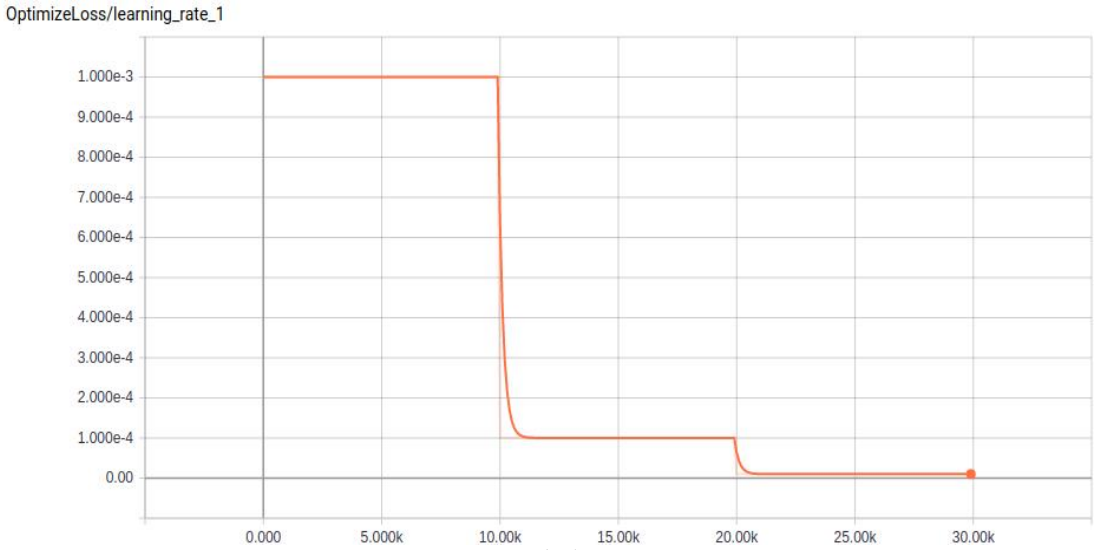

图 4-7　学习率变化图示

图 4-7 中横轴代表训练的步数，纵轴代表该模型训练时的学习率。

(2)误差

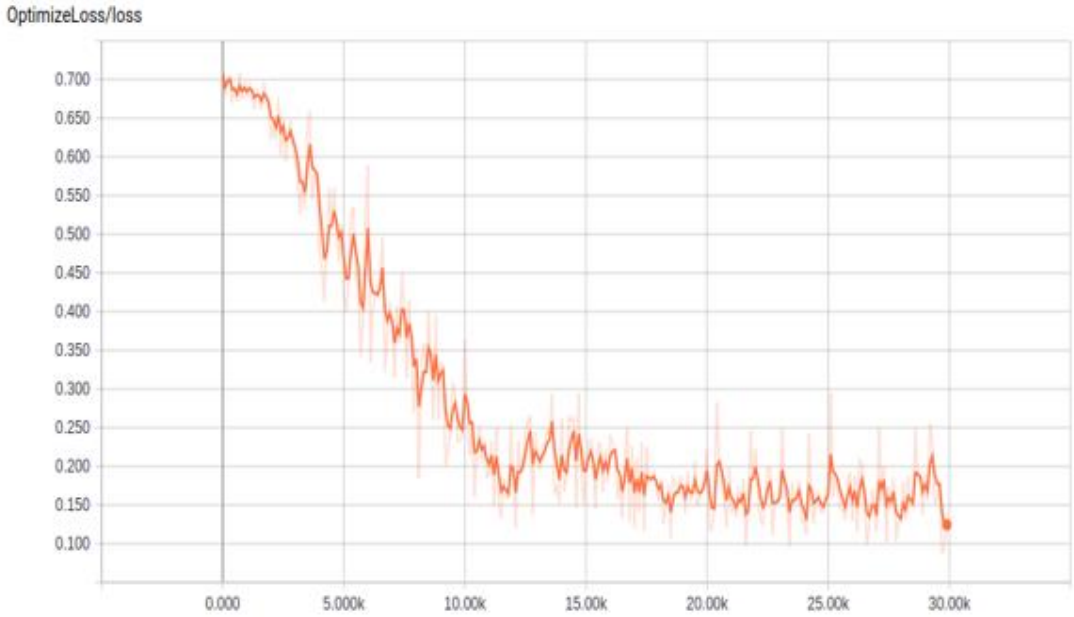

图 4-8　误差函数变化图示

图 4-8 中横轴所代表含义与图 4-7 一致，均为训练的步数，纵轴代表该模型训练时的误差。

通过图 4-7 与图 4-8 可以观察到当学习次数到达 20000 次时，学习率已经趋于 0，而最终的误差也在 0.15 左右震荡，网络模型在训练次数达到 30000 次时已经趋于稳定。

4.4 结果仿真和分析

在此我们使用由 4.3 节训练完成的模型对人脸图像进行性别测试，以验证模型的优劣。如下图实例：

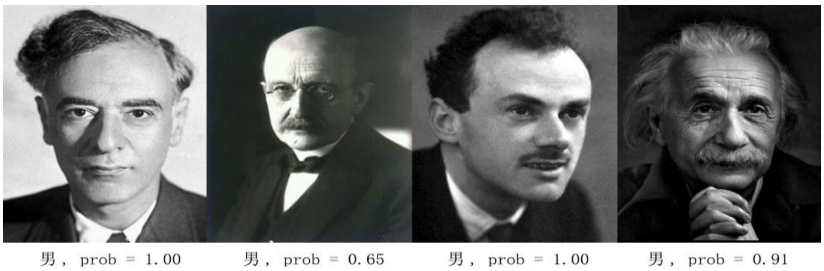

图 4-9　个别图片性别识别结果图示

我们可以看出，在以上的若干测试图片中，三张图片变现良好，一张图片虽然给出正确性别，但是其概率并不高。在证明此模型为有效模型的基础上，我们制作一个含有大量图片的数据集去检测其识别率。

在此我们选用在网络抓取的 100 张图片进行验证，其中男女各为 50 张。其中一部分如下所示：

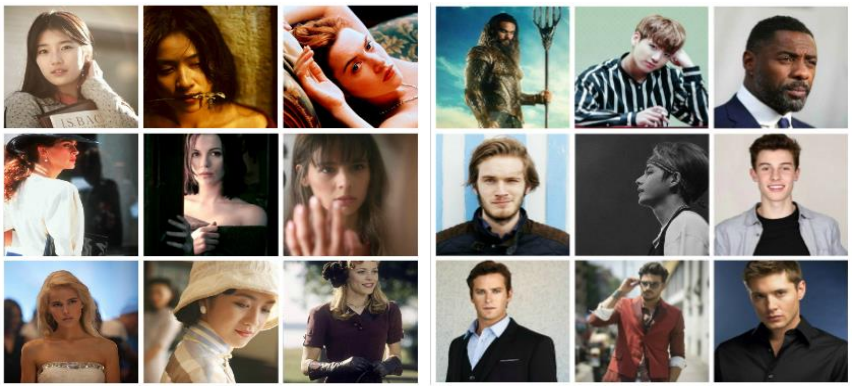

图 4-10　随机抓取图片图示

经过测试，我们得出正确率曲线如下：

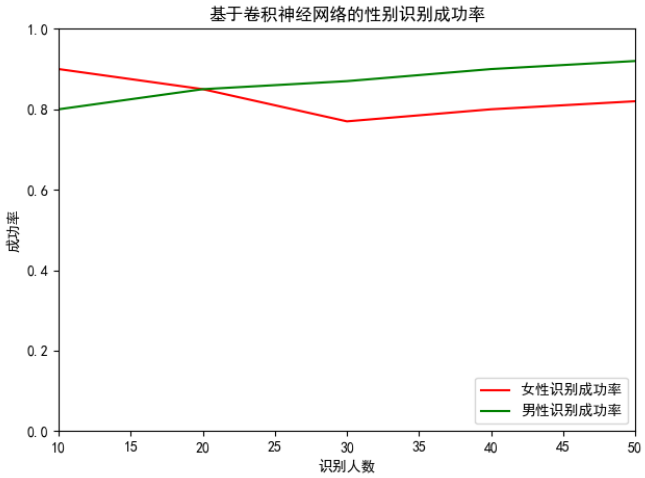

图 4-11　性别识别正确率图示

我们可以发现，男性的识别成功率在90%左右，而女性的识别成功率在80%左右，均取得了不错的结果。这个结果说明了本文中基于 LeNet 改进的卷积神经网络模型较为成功，可以在合理的精度内有效地完成解决性别识别的任务。

总结:在本章内，文章首先研究了基于 Haar 特征与 Adaboost 算法的人脸检测方法，之后通过 opencv 的相关开源代码对之进行了仿真。在本章的第二小节重点对卷积神经网络的网络架构配合实际应用的 python 代码进行了说明，三、四小节对网络的训练过程以及训练之后模型的结果进行了仿真与说明。测试图片为在网络中抓取的随机图片，在大多数为正面照、少部分为侧面照的情况下，均取得了在识别率80%之上(男性图片达到90%)的结果，说明了此卷积神经网络在解决性别识别的问题上是构建成功的。

第五章 总结与展望

在人工智能浪潮席卷世界的今天，机器学习作为其中最为重要的一个部分，在目前显示出了最为强大的能力，切实地解决了各行各业中无数之前难以解决的问题。而机器视觉在机器学习中占有重要的一个位置，近年受到了全球科技工作者的注意。人脸监测与性别识别作为机器视觉中的一项重要内容，对社会治安、用户服务等方方面面都有着重要的研究价值。

文章使用了基于 LeNet 卷积神经网络的修改版本，较少的神经层使得网络不易陷入过拟合，而在最后的仿真结果测试中也证明了这一点，性别的识别率达到了较高的水平。结果说明，在正面照的情况下，该网络模型可以取得较高的正确识别率，但是在侧面照、环境噪音较多的情况下，识别率较低，但是也显著高于50%。

文章的不足以及可以改进的地方在于：

1)对环境噪音较多环境的图片识别率的提升。对于这种情况，本文的网络层数便过于稀少，如何合理的安排与设计网络层数、结构，是核心的问题；

2)对于训练速度的改进。如裴子龙在论文[7]中提出的 PSO-CNN 算法，其可以加速梯度下降算法以达到节省时间的效果，但是其作者训练次数在 50 左右，能否在上万的量级中也达到加速的效果需要进一步的检验。

参考文献

[1] Yann LeCun,Léon Bottou,Yoshua Ben-gio,etc.Gradient-based learning applied to document recognition[J]. *Proceedings of the IEEE*. 1998,86(11):2278–2324

[2] Alex Krizhevsky, Ilya Sutskever, and Ge- offrey E Hinton.ImageNet Classification with Deep Convolutional Neural Networks[J]. *Commun. ACM*. 2017,60(6):1097–1105

[3] G. Levi ,T. Hassner.Age and gender classification using convolu- tional neural networks[C],in Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops, Boston,2015

[4] Amr Ahmed,Kai Yu,Wei Xu.Training Hierarchical Feed-Forward Visual Recognition Models Using Transfer Learning from Pseudo-Tasks[J], *Lecture Notes in Computer Science*. 2008, 5304:69-82

[5] Aslam Aasma.Wavelet-based convolutional neural networks for gender classification[J], Journal of Electronic Imaging.2019, 28(1):1-12

[6] 黄勇. 基于人脸图像的性别识别研究[D]. 成都：电子科技大学，2016.6

[7] 裴子龙. 基于卷积神经网络的人脸性别识别研究[D]. 临汾：山西师范大学，2017.6

[8] 陆丽. 基于人脸图像的性别识别与年龄估计研究[D]. 上海:上海交通大学，2010.3

[9] 邱锡鹏. 神经网络与机器学习[M]. 上海：复旦大学，2019.4

[10] TIOBE Index for May 2019(https://www.tiobe.com/tiobe-index/)

[11] G.Cybenko.Approximation by Superpositions of a Sigmoidal Function[J],*Mathematics of Control, Signals, and Systems*, 1989,pages 303-314,

[12] Rob Robinson.Machine Learning Notebook:A resource for machine learning with Python[M]. London: Imperial College London,2017.6

[13] Viola,Jones.Rapid Object Detection using a Boosted Cascade of Simple Features[C]. in Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops,Kauai,2001

[14] Cascade_Classifier Training(https://docs.opencv.org/master/dc/d88)

[15] _cvhaartraining.h(https://github.com/npinto/opencv)

[16] haarcascade_frontalface_default.xml(https://github.com/opencv/opencv)

[17] Alex Krizhevsky, Ilya Sutskever, and Ge- offrey E Hinton. Imagenet classification with deep convolutional neural networks. *In Advances in neural information processing systems*, 2012,pages 1097–1105

[18] Unfiltered faces for gender and age classification(https://talhassner.github.io)

致　谢

大学的四年时光瞬时而去，我也从一个期望大学间会发生些什么的新生，变成了现在回想着大学间发生过什么的毕业生。四年时光，千多日夜，有辛苦学习，也有更多贪玩时光。总而言之，青春无悔。

在完成这个论文的几个月里，王磊老师悉心地为我指导，在忙碌的课时之间给我们安排了多次的课堂、会议，像平日的课堂一样给我们讲授论文中的相关知识。幽默漂亮、认真负责是王老师留给我的最大的印象，在此要说一声感谢。

再要感谢的是计算机学科开放、友好的整体环境，没有优质的开源代码和诸多论文的阅读经验分享，还有诸多优质的相关开源书籍，我不能在短时间对其中的原理与较晦涩的知识有较多的把握。

还要感谢的是我的女朋友范雯俊、大学间的同学与朋友、在背后默默支持我的父母，在平日的学习中，是你们的包容和鼓励以及身前身后的陪伴，让我能不断进步。

最后要感谢答辩中的各位老师，你们辛苦了。