

题 目 基于机器视觉的人体运动类型识别

仪器科学与工程学 院（系）测控技术与仪器专业

学　　号 22011227 .

学生姓名 黄安杰 .

指导教师 莫凌飞 .

起止日期 2015年2月至6月 .

设计地点 中心楼 .

基于机器视觉的人体运动类型识别

摘要

随着人体动作识别在人群监控、人机交互、运动分析、人物相关内容检索等众多领域中的应用越来越广泛，成为了计算机领域中备受关注的前沿方向之一。本文提出用基于深度学习的机器视觉模型，在只使用由 Kinect 采集回来的人体骨架的运动数据的条件下，进行人体运动类型的识别。由 Kinect 采集回来的骨架模型数据具有数据容易获取，数据维度少的特点，可以方便地存储、传输与处理。因此，本文提出的识别模型在训练时除了 CAD-60 数据集提供的人体骨架数据，没有使用任何的先验知识来进行人体运动类型的识别。由此可以减少在数据的预处理阶段中人工的干预，只由机器自己来进行特征的提取，从而减少人工提取特征时的工作，使最终的模型的泛化性更优秀，鲁棒性更好，模型对人体运动类型有更好的理解与识别。在模型的研究上，本文测试了卷积神经网络、自编码器、降噪自编码器以及混合的多种结构，并实验了多种网络训练上的效果加强算法，比如一些正则化的方法和其他新的激活函数，最后选择了卷积神经网络作为自动特征提取的模型，并在其后面配合上多层感知机来进行分类，针对 CAD-60 里的数据，识别率为 81.8% ，取得比较优良的识别效果。  
**关键字：**人体运动类型识别，CAD-60 数据集，深度学习，卷积神经网络，自编码器，Kinect

HUMAN ACTION RECOGNITION BASE ON COMPUTER VISION

Abstract

Human action recognition is an active research area in the computer science because it is widely used in the fields of the security monitoring, human machine interaction and the character related content searching. We propose a computer vision model in the paper base on the deep learning theory, which can recognize the human action only base on the data source in the skeletons of the human from Kinect. The reason to use the skeletons data from Kinect is it is easy to obtain, to store and to transfer, and it has the less dimensions of the data. This model only uses the human skeletons data from the CAD-60 dataset to recognize the human action without using any prior knowledge. It can reduce the works from the human on the stage of preprocessing and hand the feature extraction to the computer, which can reduce the error from the human-engineer. It can also improve generalization performance and robustness of the model, and give a better understanding of the human action. In the paper, we do the experiment on with convolutional neural networks, the autoencoder, the denoising autoencoder and some of their mixture, and it also do some experiments for the tricks which can improve the neural network, such as some regularization methods or other activation functions. In the end, we choose the convolutional neural networks for the feature extraction, with using the multilayer perceptron as the following classifier. And we get the accuracy rate of 81.8% which mean that the model is effective.

**keywords:** Human action recognition, CAD-60 dataset, deep models, convolutional neural networks, autoencoder, Kinect

目录

[第一章 绪论 1](#_Toc421830369)

[1.1 引言 1](#_Toc421830370)

[1.2 人体运动类型识别的相关研究现状 1](#_Toc421830371)

[1.2.1 人体运动数据源的获得 1](#_Toc421830372)

[1.2.2 人体动作识别的主要问题 2](#_Toc421830373)

[1.2.3 国内外研究现状 2](#_Toc421830374)

[1.2.4 使用深度学习进行有效的特征提取 3](#_Toc421830375)

[1.3 模型选择 4](#_Toc421830376)

[1.4 数据源选择 4](#_Toc421830377)

[第二章 人体识别模型原理 5](#_Toc421830378)

[2.1 特征提取模式 5](#_Toc421830379)

[2.1.1 人工特征提取 5](#_Toc421830380)

[2.1.2 无监督训练特征提取 5](#_Toc421830381)

[2.1.3 有监督特征提取 5](#_Toc421830382)

[2.2 深度学习经典模型 6](#_Toc421830383)

[2.2.1 主要成分分析 PCA 6](#_Toc421830384)

[2.2.2 自编码器 Autoencoders 7](#_Toc421830385)

[2.2.3 卷积神经网络 7](#_Toc421830386)

[2.3 多层感知机（MLP） 9](#_Toc421830387)

[2.3.2 前向传播 10](#_Toc421830388)

[2.3.3 后向传播 11](#_Toc421830389)

[第三章 人体运动识别模型设计 12](#_Toc421830390)

[3.1 数据预处理 12](#_Toc421830391)

[3.1.1 对输入数据进行标准化 12](#_Toc421830392)

[3.1.2 参数的初始化 12](#_Toc421830393)

[3.1.3 打乱样本的训练顺序 12](#_Toc421830394)

[3.2 网络设计技巧 13](#_Toc421830395)

[3.2.1 加入正则化环节防止过拟合 13](#_Toc421830396)

[3.2.2 加入范数惩罚 13](#_Toc421830397)

[3.2.3 Max-norm 14](#_Toc421830398)

[3.2.4 Dropout 14](#_Toc421830399)

[3.2.5 Maxout 14](#_Toc421830400)

[3.3 实验模型 15](#_Toc421830401)

[3.3.1 5层的多层感知机 16](#_Toc421830402)

[3.3.2 自编码器＋多层感知机 16](#_Toc421830403)

[3.3.3 卷积神经网络＋多层感知机 16](#_Toc421830404)

[第四章 模型实验设计与验证 18](#_Toc421830405)

[4.1 框架选择 18](#_Toc421830406)

[4.2 数据集 20](#_Toc421830407)

[4.3 实验准备 22](#_Toc421830408)

[4.3.1 实验环境 22](#_Toc421830409)

[4.3.2 实验编程平台 22](#_Toc421830410)

[4.4 实验结果 23](#_Toc421830411)

[4.4.1 5层的多层感知机 23](#_Toc421830412)

[4.4.2 自编码器＋多层感知机 23](#_Toc421830413)

[4.4.3 卷积神经网络＋多层感知机 24](#_Toc421830414)

[4.4.4 加入正则化环节 24](#_Toc421830415)

[4.5 实验结果分析 25](#_Toc421830416)

[4.5.1 同行结果对比分析 25](#_Toc421830417)

[4.5.2 本文实验结果分析 26](#_Toc421830418)

[4.6 基于深度模型的动作识别软件开发 27](#_Toc421830419)

[第五章 总结 28](#_Toc421830420)

[5.1 总结 28](#_Toc421830421)

[5.2 展望 28](#_Toc421830422)

[5.2.1 加入提前停止法 28](#_Toc421830423)

[5.2.2 更换其他分类器 28](#_Toc421830424)

[参考文献 29](#_Toc421830425)

[致谢 33](#_Toc421830426)

1. 绪论
   1. 引言

在模式识别和机器学习(Pattern Recognition and Machine Learning, PRML)领域中，人体动作识别(Human Action Recognition, HAR)一直是比较重要的一个领域。 计算机自动去理解人类的行为、动作还有跟环境之间的交流互动，在近年来逐渐地成为了一个热门的领域，因为这个技术在很多领域都有可以使用的地方。比如现代社会快速的生活节奏和巨大的工作压力，严重影响着个人的身体健康。科学的运动可以提高身体素质增强运动能力，进而降低患病的风险(尤其是一些慢性疾病），例如糖尿病、血脂异常、高血压等。而进行科学运动的前提是实现人体运动类型的准确识别，对人的运动、健康状况进行理解、评估，这样才能对这些东西进行有用的指导和建议[1]。

在信息安全领域，可以通过监控视频识别到人物的动作，甚至是识别到身份，用这种方式利用计算机自动对视频中人体的运动类型进行识别从而为监控或者案件侦破提供依据，也具有着重要的意义和广泛的用途[2]。

在人机交互领域，可以通过手势、身体姿态等信息来进行输入，这样除了辅助交互输入设备和自然语言分析，其他的空白也能得到补充。这样就可以提高计算机和人进行交互的能力，使过程更有趣、更高效[2]。

在大型的图像数据库或者互联网上的信息中[3]，还可以通过对人体运动类型的识别，从而标注并理解其中的信息，让搜索的方位更广、提供的信息更多。

由于人体动作识别的应用如此广泛，如何有效地提高人体动作识别的准确率是该领域急需解决的问题。而本本的任务是通过使用深度学习的相关理论，使用深度学习的相关技术来提高人体动作识别的识别率。

下文中分别对深度学习中的卷积神经网络、自编码器等模型进行试验，并应用类似 Dropout、 Max-norm 等算法对这些已有的模型进行改良，提高它们的效果，并用它们构造分类模型，应用于人体运动类型的识别，从而对人体运动类型有更好的理解，并提高对人体动作识别的准确率。

* 1. 人体运动类型识别的相关研究现状
     1. 人体运动数据源的获得

首先，对于人体运动识别的数据的来源存在着几种不同的种类[4]。例如由 RGB 摄像机、距离传感器或其他遥感的方式。使用深度传感器来进行人体运动识别的发展始于 80 年代初。过去的研究主要集中在训练和认识到从视频序列(可见光相机）所采集的数据中。可见光视频的主要问题是从单目视频传感器捕捉得到的人体运动存在相当大的损失。由于视频天生的对的人体行为识别的限制，尽管已经有了过去几十年的努力，但通过视频来识别人体运动，仍然是非常具有挑战性的。

而得益于近期发布的成本低廉的深度传感器，和 3D 数据相关的研究越来越多了。在过去的 20 年里，获得 3D 数据的方法，一共分为三类。一种方法是通过使用基于标记的动作捕捉系统，如 MoCap。 第二种方式是通过立体视觉： 从多个角度捕获 2D 图像序列，通过从多个视图来重建三维信息。第三种方式是使用距离传感器（使用类似 TOF 原理的传感器）。深度相机在过去几年里取得迅速的发展。最近出现的深度照相机可以在相对低廉的成本和较小的尺寸的条件下提供较高的帧率和分辨率，这导致出现了许多新的研究中的动作识别都是采用的三维数据。

* + 1. 人体动作识别的主要问题

基于视觉的人体动作识别里有四个主要的问题。

第一个问题的挑战性比较小 [5-6]：闭塞、 杂乱的背景、 阴影和不同光照条件会让运动难以分割或者被错误地识别。这是从 RGB 视频行为识别的一大难点。引入 3D 数据可以在很大程度上通过提供现场的结构信息，从而缓解这个问题。

第二个是视角的变换[5,7-9]。相同的操作可以从不同的角度产生不同的"外观"。传统的 RGB 相机解决这一问题的方法主要是引入多个同步的摄像机，同时获得多个视角的图像，但对于某些应用程序，这不是件容易的事。不过对于三维运动捕捉系统，这不是一个严重的问题。而如果通过深度图像来进行识别的话，这个问题也会有部分被缓解，因为从轻微旋转的视角的外观可以推断深度的信息。这一点并不完全解决问题，因为摄像机始终还只是在对象的一侧上，这个距离图像只提供了部分的信息，还是没有人知道这个对象的另一面是什么样子的。如果可以运用单一深度相机来精确地推断出人的骨架模型，则可以通过骨架模型的信息来构造一种视图不变识别的算法。

第三个问题是放缩上的差异，因为人离相机的距离的不同会影响主体的大小从而影响运动的识别。而在 RGB 视频中，这可以通过在多个尺度下的[10] 特征提取解决了。而在深度视频中，这可以很容易调整，因为真正的主体的 3D 尺寸直接是已知的。

第四个问题是同一种类内的变异性和不同种类之间的相似性问题[11]。人可以通过不同的身体部位在不同的方向上做动作，但不同的方向和两个动作仅只有只由非常细微的细节来区分。而这个不管对于使用哪种数据的来源的算法来说都仍然是一个非常困难的问题。

前三个问题基本上都可以通过使用三维空间上的图像或者类似骨架之类的模型来解决，而且由于现在获得人体运动骨架已经变得非常简单了，所以本文使用的数据来源是由传感器采集计算直接得到的骨架数据。

* + 1. 国内外研究现状

从各种论文来看，相关研究的重点都是集中在寻找合适的特征，然后通过一些经典的或者是修改过的分类器进行分类。

如 Yu Zhu等人[12]提出来的使用通过在 RGB 图像或者深度图像中提取时空特征点STIP，然后把它与骨架模型等其他特征联合起来。把特征组合起来后放入 SVMs 分类器进行学习，最后得到一个可以使用的分类模型。

而 Lukas Rybok 等人[13] 则是提出了使用方向梯度直方图(Histogram of oriented gradients, HOG)[14]和光流场方向直方图(Histograms of Optical Flow，HOF)[15-16]来作为提取特征的模型，然后再配合 proto-object 来描述上下文的特征，最后也是通过 SVM 分类器的学习，得到一个有效的分类模型。

张永强[17]提出的在图像的基础上提取7个 Hu 不变矩作为特征，然后把它们输入到使用了AFSA优化过的 SVM 分类器中进行分类。

除此之外，被用到的特征还有骨架模型[12,18-19]、EigenJoints[20]等或者是它们的一些拓展和变形，如方向HOF[21]，以及根据这些特征进行进一步的组合来产生更高级的特征。

而出了特征以外，一个有效的分类器也非常有必要。而在其他人的工作中，用到的分类器里有贝叶斯模型[20,22]、k近邻kNN[23]、支持向量机 SVM[13,15,23-26]、隐马尔可夫模型 HMM[19,23]来进行有监督的训练得到分类。分类器之间也可以通过串联[20]或者并联（票决、加权）来加强分类的效果。

* + 1. 使用深度学习进行有效的特征提取

上一节所说到的特征的选取或者是计算一般是通过人工方式得到的。而这种人工方式得到的特征对应用的领域有较强的依赖性，也就是说，换一个研究的问题之后可能同样的特征就不能再适用了。

而深度学习[27]的提出，某种程度上解决了这些关于特征提取的这些问题。

模仿出人脑表征信息的高效和鲁棒性一直是近几十年来人工智能研究中的一个核心。而人类不仅每时每刻都暴露在由感官接收的无数的数据中，而且还能够通过某些方法捕捉到这些数据关键的部分来让自己能够以简单的方式在未来使用。早在五十年前，提出了动态规划理论以及开创了最优化控制领域的 Richard Bellman 就曾断言，数据的高维度是在人工智能科学和工程应用中的根本障碍。其中主要的困难，尤其是在模式分类的应用中，就是数据训练的难度会随着数据维度的线性增长，发生指数级的增长[28]。而克服这个问题的主流方法就是以一定的方式(比如分类器、SIFT 算法）对数据进行预处理来减少数据的维度，这样数据就可以被有效地处理。这种减少维度的做法一般被称为特征提取。因此，可以认为，在很多模式识别系统的背后的智能其实被转移到了人工的特征提取处理上去了，而这种人工的做法有时会很困难而且会高度依赖于具体的应用场景[29]。此外，如果不完整或者错误的特征被提取出来了，那么分类处理的性能就会从根本上受到限制。

最近，神经科学在哺乳类动物大脑上的新发现指出可以通过一个复杂的深层网络结构来对数据自动地进行预处理，自动地进行特征的提取。这就是深度学习的基本来源。

* 1. 模型选择

深度学习由于具有对时空相关的数据的处理的特征提取能力优异，能大大减少人工进行特征提取的工作，所以本文着重使用深度学习的相关理论对人体运动类型进行建模、识别。

* 1. 数据源选择

使用成熟的骨架模型作为高级特征的进一步提取和识别分类具有使用原图像和深度图像所没有的视角、放缩和背景的不变性。而且，使用骨架模型还具有容易获得，容易理解，数据维度小，存储、传输、处理的特点。

所以本文选取骨架模型作为模型的设计和训练的数据源。

1. 人体识别模型原理

本章总结了人体运动识别时提取特征的几个模式，并且介绍了深度学习以及它里面的一些经典模型，之后再介绍了本文用到的经典的分类器多层感知机。

* 1. 特征提取模式

本节会总结一些人体运动识别中出现的关于特征提取环节的模式。

* + 1. 人工特征提取

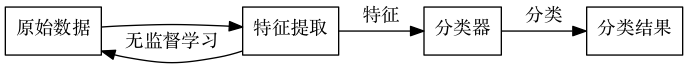
这种模式是指是使用一些人工设计的，由人对源数据的理解所构造的算法来进行特征提取如之前说过的骨架模型、时空特征点之类的方法。也就是在识别的模型里加入一些人对数据的理解的先验的知识来加强模型的效果。这种方式会随着不同的数据或者不同的应用领域而改变，每换一个领域可能都需要使用人的知识和能力来进行训练、归纳和总结，然后选择一个特征提取的方式，如下图所示



人工提取特征

* + 1. 无监督训练特征提取

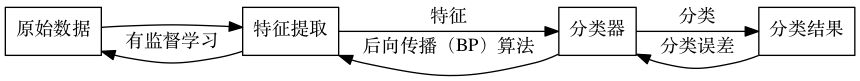
这个类型主要是使用无监督训练的方法如限制玻耳兹曼机 RBM，自编码器，或者一些类似主要成分分析 PCA 的算法，在不需要数据集带有标签的前提下能够对数据的特征进行提取，发现特征内在的信息，从而减少数据的冗余，减少数据的维度。而这类型的模式如下图所示



无监督特征提取

* + 1. 有监督特征提取

这个类型是使用有监督训练的方法，在特征提取里面把标签的信息也考虑进去，从而使特征提取的结果更有针对性，对后面的分类更有利，这类型里的算法有 CNN 卷积神经网络，过程则如下图所示



有监督特征提取

* 1. 深度学习经典模型

上一节中总结的三种特征提取的模式中，第一种人工提取特征是比较经典、传统的做法，但它也存在一定的局限性。比如人工提取特征需要大量的工作量来对问题进行建模、实现，实际上是把模式识别背后的智能通过这部分转移到了人的先验知识的学习上。大量的人工工作量也意味着特征提取会高度依赖于具体的应用场景。而后两种就是深度学习主要的特征提取的模式，就是通过一定的模型对特征进行有监督或者是无监督的学习，达到自动地进行特征提取。

* + 1. 主要成分分析 PCA

主要成分分析[30-31] 里的假设是，在数据里变化最大的那些因素就是最有价值的因素，因此，只要把输入的数据的空间找到一个变换，这个变换可以把数据里变化最大的因素提取出来，其他变化不那么大（利用价值比较小）的舍弃不看，就可以降低数据的维度，从而减少后面的计算的难度。比如对于图像数据，像素之间是有高度的相关性的，如果能有一种方法，把这些像素之间的相关性去掉，就可以让图像的信息量都压缩到了更少的变量上去了，减少了这些维度就可以加速对于图像的特征的提取或者图像的识别、分类。 首先要做的是数据的预处理，这里要做的就是数据的均值中心化和方差归一化：

1.数据的均值中心化，也就是把数值变化范围的中心移至 0 处，这里用到的算法是 max-min 标准化，这个标准化的方法指的是找到数据中的最大值和最小值然后应用下面的这个式子来达到放缩、平移数据的目的:

（2-1）

2.数据的方差归一化:

（2-2）

然后进行 PCA 核心环节：

1.首先求协方差矩阵，这里求得的协方差矩阵表达的意思就是输入的变量里各个特征的变化：

（2-3）

2.求得协方差矩阵的特征向量并按特征值从大到小依次排列得到：

（2-4）

3.按照一定的方法取 ，比如一个常见的经验法则是选择 以保留99%的方差，这样就可以把对总的变化贡献最大的，也就是信息量最大的特征挑选出来：

（2-5）

4.取前 个特征值对应的特征向量，组成新的特征向量矩阵。并与 相乘，得到新的降维后的数据 ：

（2-6）

（2-7）

* + 1. 自编码器 Autoencoders

一个自编码器[32-33]会对一个输入 经过一次线性变换之后得到一个确定的表达 :

（2-8）

这其中的 是非线性环节，比如 sigmoid 函数。然后，这个潜在的对输入的表达 （或者说编码），会被再一次地重新映射（或者说解码）到一个和 一样结构大小的重建出来的 。（上述的式子没有使用矩阵变换的形式）。这里的 应该视为在 的表达下的 的一种预测。而重建的里面的 可以限制为正向映射的 的倒置：。而这个模型的参数（比如 ）可以通过使平均重建误差最小化来进行优化。

重建误差可以通过很多方法进行测量，这个依赖于编码的输入的分布。传统的也是可以使用的。

（2-9）

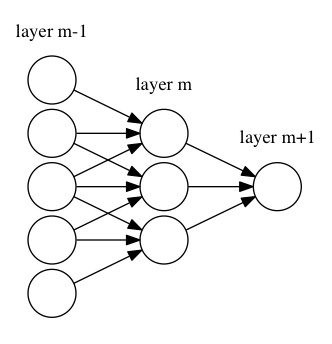
这里面的 是输入的数据的空间中的主要因子的一个表达。这个跟主要成分分析中的捕捉输入数据空间的主要因子非常类似。而事实上，如果中间的隐含层（编码）是一个线性变换而且是使用均方误差来训练网络的话，有 个隐含单元就意味着其实是在找出原数据的前 主要成分。但是如果隐含层是非线性的话，自编码器就跟 PCA 很不一样了，因为它还具备了捕捉输入数据里多模态的部分。

* + 1. 卷积神经网络

卷积神经网络 CNNs [34-35]是一个专门为二维数据(例如图片和视频）设计的多层神经网络的系列。 CNNs 是第一种真正意义上成功的深度学习方法，它能够成功地用一种鲁棒的方式训练出多层的神经网络。相比起普通的多层神经网络，卷积神经网络有很多独有的特点。

* + - 1. 稀疏连接

卷积神经网络利用了空间的局部相关性，相邻两层之间只进行了局部的连接，而不是像多层感知机 MLP 一样使用全连接的方式。换句话说，隐层 m 的单元的输入来自于层 m-1 在空间上连续的感受野的单位的一个子集。如图2-4所示，层 m-1 是数据的输入，层 m 的单位感受野宽度为 3 ，因此层 m 的每个神经元只连接到输入层的 3 个相邻的神经元。在层 m＋1 中也有与下方图层类似的连接性。虽然他们对于上次层的感受野宽度也是 3，但实际上他们相对于输入的层 m-1 的感受野的是 5。就是说每个 层m 里的单元都会响应在其关于视网膜的感受野的变化。



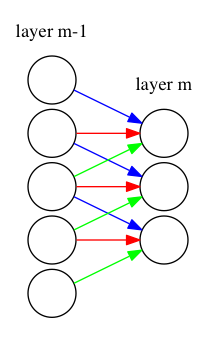
稀疏连接

这个层级结构确保了训练出来的 filter 可以对来自于局部空间的输入产生最强烈的反应。因此，也正如上面所说的，叠加这种层级的结构会导致（非线性）的 filter，可以使它越到后面的层，响应对应的输入感受野更大。例如，在隐藏图层 m + 1 单元可以编码 5 宽度 （以像素的空间） 的非线性特征。

这种连接的编码方式，比起普通的全连接的神经网络显然可以有效地减少连接的数量，从而能容纳更多的层数，对输入的数据进行更复杂的描述，加强了模型的表达能力。而且，只与上一层的相邻的局部的几个神经元连接的方式，可以使模型具备一定程度的某维度维度上的不变性。

* + - 1. 共享权值

除了稀疏连接之外，每一个 filter 都是一套参数应用在整个输入空间上的。具体如图所示，



权值共享

上图中三种不同的线代表三个权值（W、b，即权值和偏置），层 m 中每个单元都是由相同的这个三个权值和他们对应的感受野里的输入神经元计算出来的。假如没有权值共享，则需要的权值就有 个，由此可见，权值共享使得需要进行优化的参数大大地减少了，使得特征能够更有效地被提取。通过控制模型的规模，卷积网络对视觉问题的泛化能力非常好。

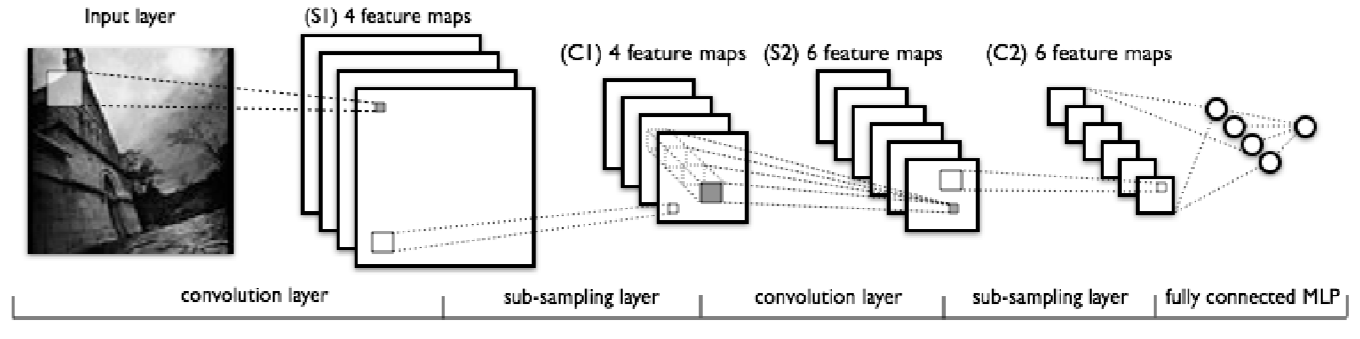
以上两个特点都是由通过使用卷积核卷积计算生成多个特征图的方式实现的。

* + - 1. 池化（pooling）

而完整的卷积神经网络的结构里，还有池化的层用来进行二次的采样，比如使用一个 的求最大值窗口或者平均值窗口进行降采样。这种操作被称为池化，池化的操作的作用是可以提供一定的空间位移的不变性以及进一步减少数据的维度，一般池化之后会通过一个激活函数得到一个新的网络的层。

* + - 1. 完整模型

以上两个特点都是由通过使用卷积核卷积计算生成多个特征图的方式实现的。而完整的卷积神经网络的结构里，还有其他的东西，以 LeNet为例，展示一下完整的卷积神经网络的结构。



LeNet-网络[32]

上图中的网络输入图像通过 4 个可训练的滤波器和偏置在 S1 的位置产生了 4 个特征图。特征图中的每个图通过池化并通过一个激活函数之后得到一个 C1 位置上的 4 个新的特征图。然后会通过一个新的卷积层然后得到新的 6 个特征图 S2。这些将再一次池化得到 C2。最终这些像素的值进行光栅化、摊平成一个向量，这个向量就是这个网络里提取得到的特征。这些特征之后就可以输入到一个常规的分类器里，比如神经网络（多层感知机 MLP）里，最终由这个分类器进行分类结果的计算和输出。

而这些卷积核的权值和偏置，还有其他用到的权值则使用后向传播算法来进行计算出梯度，然后通过梯度下降法之类的算法进行优化，从而训练出来一个有效的特征提取的卷积神经网络。

* 1. 多层感知机（MLP）

特征提取完了之后，需要把特征交到后面的分类器进行分类，而由于有监督的特征提取网络需要得到分类错误后的传递过来的误差，所以需要需要使用可以进行后向传播的分类器，这里面比较经典的分类器就是多层感知机（MLP）也就是普通的神经网络[36]。

神经网路由神经元组成：



典型神经网络中的神经元

上图中的神经元实际上是一个以 及截距 为输入值的运算单元，其输出为 ，其中函数 被称为“激活函数”。激活函数的作用主要是为神经网络引入非线性环节，使其对非线性的函数具有更好的拟合能力。一般的，会选用 sigmoid 函数作为激活函数：

（2-10）  
典型的神经网络如图所示：



典型多层神经网网络

如上图所示，把多个神经元按上图的形式以多层的形式串联起来，就可以得到可以一个神经网络用来对数据进行训练、拟合或者分类。

* + 1. 前向传播

输入的值经过一层一层的神经元的计算、激活，一直到达最后一层，并得到值称之为前向传播，对于图2-8中的模型，前向传播的计算步骤如下所示：

（2-11）

如果把权值 W 表示为矩阵，且将激活函数 扩展为使用向量则上列等式可以更加简洁地表示为：

（2-12）

（2-13）

（2-14）

（2-15）

* + 1. 后向传播

上述的前向传递是根据输入，通过网络逐层计算出来合适的结果，但是网络的里的权值、偏置需要确定，所以可以通过梯度下降法来进行对这些参数，也就是对神经网络进行求解。而对于单个输入的测试样例 ，其代价函数为：

（2-16）

只要得到每一个参数的关于代价函数的偏导数，就可以使用梯度下降法来进行优化。

而求解每一个参数的偏导数就需要使用反向传播算法了。

反向传播算法的思路如下：给定一个样例 ，首先进行“前向传播”运算，计算出网络中所有的激活值，包括 的输出值。之后，针对第 层的每一个节点 ，计算出其“残差” ，该残差表明了该节点对最终输出值的残差产生了多少影响。对于最终的输出节点，可以直接算出网络产生的激活值与实际值之间的差距，并将这个差距定义为 （第 层表示输出层）。对于隐藏单元，将基于节点（第 层节点）残差的加权平均值计算 ，这些节点以 作为输入。

后向传播算法可以具体表示为以下步骤：

1.进行前馈传导计算，利用前向传导公式，得到 直到输出层 的激活值。

2.对输出层（第 层），计算：

（2-17）

3.对于 的各层，计算：

（2-18）

4.计算最终需要的偏导数值：

（2-19）

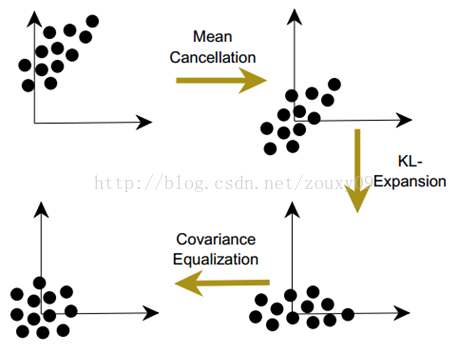
（2-20）

1. 人体运动识别模型设计

这一章里具体先是描述了一些在实验中用到的用来提高结果的算法实现细节，然后再描述了本文中做实验用到的几个模型的设计。

* 1. 数据预处理
     1. 对输入数据进行标准化

由于训练率对于每一个参数都是一样的，所以为了加快收敛，可以让所有参数都以相同的标准进行线性变换[37]，映射到一个一致的范围里对输入的。具体的变换要求如下：  
A、训练集的每个输入变量的均值要接近于0；  
B、对输入变量进行缩放，使他们的方差具有相同的值；  
C、输入变量最好是不相关的。  
如图所示：



输入标准化

* + 1. 参数的初始化

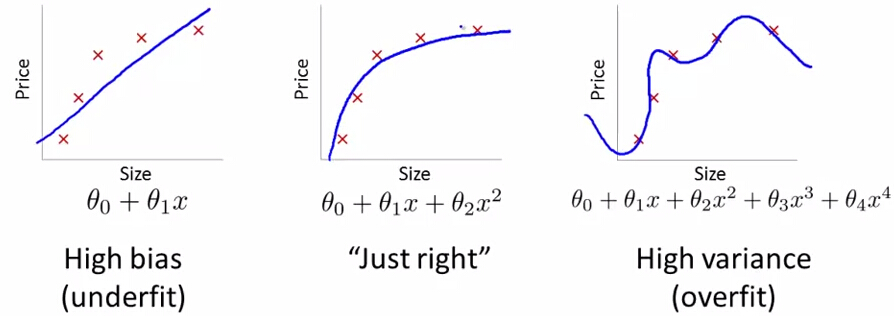
参数的初始值[37]对训练过程有着重大的影响。而参数初始化的原则是：参数应该随机初始化在能让 sigmoid 函数在线性区域激活的值。如果参数全部都很大，那sigmoid一开始就饱和了，这样就会得到一个非常小的梯度值，那参数更新就会很慢，训练也会很慢。使 sigmoid 函数工作在线性区域激活除了需要输入数据标准化到其标准差为 1，还需要参数初始值，也就是连接神经元的权值的初始值的标准差也能为 1。最简单的做法就是，给参数初始化的时候使用正态分布采样得到，那么参数就会均值为零，且方差近似1。

* + 1. 打乱样本的训练顺序

在使用后向传播算法进行训练的时候，有一个原则是使用意料之外的样本来训练，收敛得会更快。一般来说，训练用的数据都会是规整的，按顺序排列的[37]，比如在使用的数据集里，人体的运动都是按照时间顺序在一个大的视频里分别进行切片的，所以相邻的训练数据有很大可能是来自于同一个类型甚至是同一个视频的。而这一节的这个技巧就是在粗糙地选择来自不同类的样本，也就是说，尽量让每一次的训练采用的样本都来自不一样的类型，最简单的做法就是把样本的顺序打乱。因为同一个类的训练样本很大可能携带的是相似的信息，而相似的信息很有可能让网络落入一个局部的最优解处。所以为了提高模型的泛化性能，每次使用来进行训练的样本最好是随机地而不是按照顺序地进行选择。

* 1. 网络设计技巧
     1. 加入正则化环节防止过拟合

在模型中加入正则化（Regularization）环节可以避免模型拟合出来的分类面高度非线性，从而避免过拟合现象。过拟合（Overfitting）是指模型对训练数据集中的样本拟合的非常好，但是面对训练集中没有的数据样本的分辨效果很差。正则化的主要实现方式是用某些方法尽量地使参与优化的参数尽可能的小。之所以要让参数尽可能的小，是因为正如下图所示，模型越复杂，就会像右边的图上所示，能拟合任何函数，但是显然已经过拟合了，有可能对新的数据产生分类错误。因此只要参数变小，就可以抑制模型的复杂程度，从而使得模型的泛化能力更强。



过拟合

* + 1. 加入范数惩罚

而常用的正则化的方法是直接将所有参数 按某种形式加到最小化目标函数中加以惩罚，以免参数过大导致过拟合现象的产生，比如计算出来参数的 L1、L2范数然后直接把这一项加入到需要被优化的目标函数里去：

（3-1）

一般来说，监督训练可以看作是最小化上面的目标函数，而里面的 是样本的预测的误差，后面的 就是加入的可以用来使模型更简单的正则化函数，比如L1范数正则化的话，有：

（3-2）

使用 L1 范数的正则化的话，还可以实现特征的稀疏表达。

这个方法里的 需要通过人工设定，可以通过实验来进行选择。

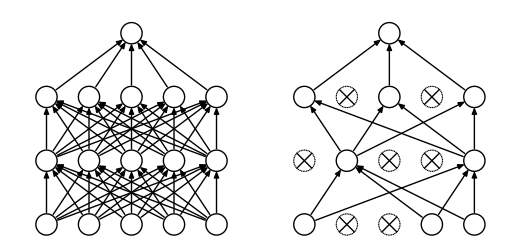
* + 1. Max-norm

除了上述的，通过在优化的目标函数里加入权值的范数来进行整个权值空间的惩罚以外，一种有效的正则化方法是 Max-norm。这种算法是在每个单个的隐藏单元里设置这个单元的传入的权值的 L2 范数的上限，如果权值的更新违反了这个约束，就可以通过让这些权值除以这个范数来达到让这个权值重新正则化的目的。使用这个约束而不是使用惩罚的方法可以在无论每次权值更新得有多大的情况下防止权值变得非常大。而且，比起惩罚方法，使用约束的时候也可以使一开始的训练率可以取得很大，使得对权值空间的搜索可以更快，更彻底。当然，在这个算法里也有需要人工设置的参数，就是所使用的 L2 范数的上限。

* + 1. Dropout

正则化的方法中，也有上述方法的变形。比如在网络的训练过程中使用的 Dropout[38-39]算法。可以有效地加强模型的泛化能力，抑制神经网络出现过拟合的现象。

Dropout 是指在模型训练时随机让网络某些隐含层节点的权值不工作，不工作的那些节点可以暂时认为不是网络结构的一部分，但是它的权值得保留下来（只是暂时这一轮里面它的值不更新而已），下次样本输入时它可能又会恢复工作了。但是这种方法是只适合于训练样本较少的条件下的用来增强模型的泛化能力的。



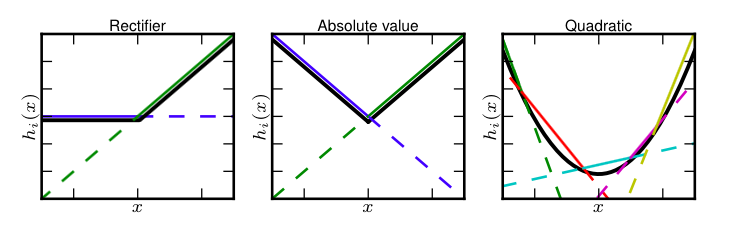
左：标准神经网络；右：使用了Dropout算法之后

而这个算法里需要人工调整的是让某些节点权值不工作的概率，一般会直接选择 50%。

* + 1. Maxout

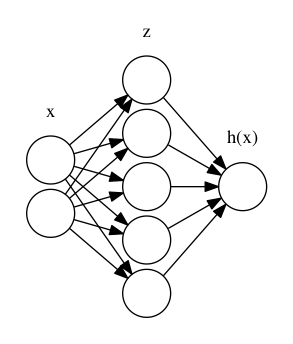
Maxout [40] 模型是类似多层感知机、卷积神经网络这种简单的前馈的架构使用的新型的激活函数。对于给定的输入 （x 可能是输入层的 或者是某一个隐含层的状态），通过一个 Maxout 自己的隐含层 ，线性映射到输出的一个节点 ，连接的权值是 。使用 Maxout 函数来作为激活函数可以加强模型对非线性函数的拟合能力，因为 Maxout 的拟合能力是非常强的，它可以拟合任意的的凸函数。最直观的解释就是任意的凸函数都可以由分段线性函数以任意精度拟合，而 Maxout 又是取k个自己的隐含层节点 的最大值，这些隐含层 节点也是线性的，所以在不同的取值范围下，最大值也可以看做是分段线性的（分段的个数与k值有关）。

下图的例子表示 Maxout 可以拟合任意凸函数，当然也包括了ReLU ( Rectifier 函数)：



Maxout可以拟合任意凸函数

而 Maxout 拓扑关系如下图所示：



Maxout网络拓扑关系

它的数学表达如下：

（3-3）

（3-4）

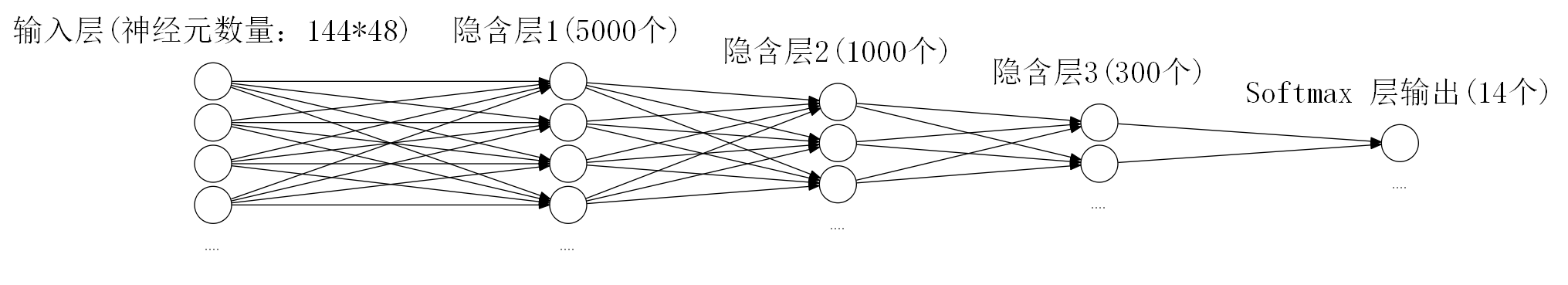
关于 Maxout 对凸函数的表达能力的证明，可以见S. Wang文章中的描述 [41]。

* 1. 实验模型

这次的实验主要尝试了两种模型，自编码器和卷积神经网络，并且在网络里使用了几种不同的能增强分类效果和模型泛化能力的技巧。下面会详细描述每个实验中的模型的设计。所有实验都是以三个人的运动为训练集，第四个人的运动为测试集的，通过训练已知的人运动去识别未知的人的运动可以有效检验模型的提取一般特征的泛化能力。所有模型在训练时，把样本先按顺序分成一批批（batch） ，然后 batch 之间再打乱顺序，一个个的 batch 放进去模型里进行训练。以 batch 进行训练的可以有效利用 GPU ，因为如果要一个个样本地放进模型里训练， CPU 和 GPU 之间要切换得很频繁，就没有办法发挥 GPU 的加速作用了。而打乱顺序是为了避免让模型陷入局部的解里，从而让模型能够得到一个更好的结果。

* + 1. 5层的多层感知机

第一个要实验的是直接对原始数据使用普通的多层感知机作为分类器进行分类，而不是先使用其他的的特征提取的方法来进行训练、提取特征。而由于后面的模型里也是会使用多层的感知机作为分类器。由于普通的神经网络里的每一层都是一维的结构，所以原本的输入的 144\*48 的这个二维的结构需要展开铺平成一维的结构，但是原来的数据的空间维度的信息就丢失了。这个多层感知机的设计如下图所示：



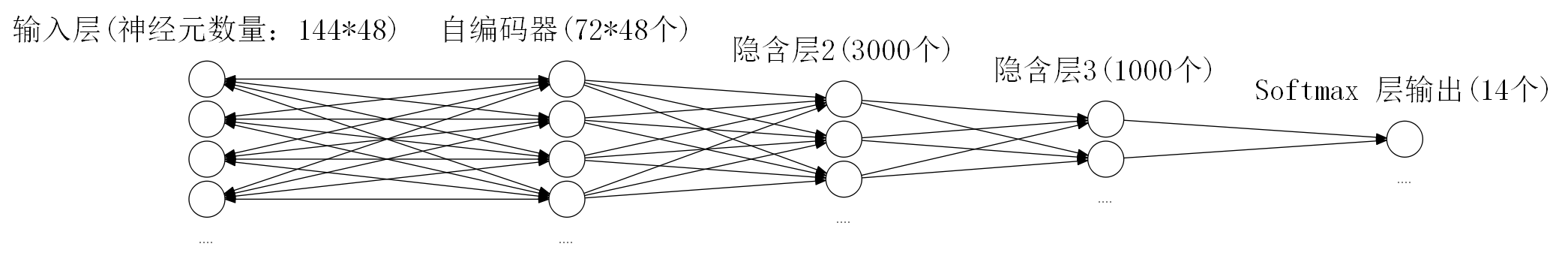
多层感知机分类器

这个网络里面的激活函数使用的是 Sigmoid 函数。

这里使用的这个模型的主要原因是为了给后面的模型的分类结果提供一个对照的基础，以表现这些特征提取的方法的效果，因此，在理解后面的模型的时候，可以把当前这个网络里的第一个隐含层替换为后面的模型的特征提取层，比如卷积网络或者自编码器。

* + 1. 自编码器＋多层感知机

使用单层的自编码器代替多层感知机的第一层隐含层。先进行无监督学习来进行降维，然后再通过后面的多层感知机进行有监督的学习来进行分类。同样的，也需要跟上一个模型一样，需要把二维的输入的数据铺平到一维。但是由于无监督学习的时候没有加入标签的信息，而整个模型的层数和上一个模型的层数是差不多的，所以直觉上，这个模型的表达能力是会比上一个模型要弱的，具体的模型如下图所示：



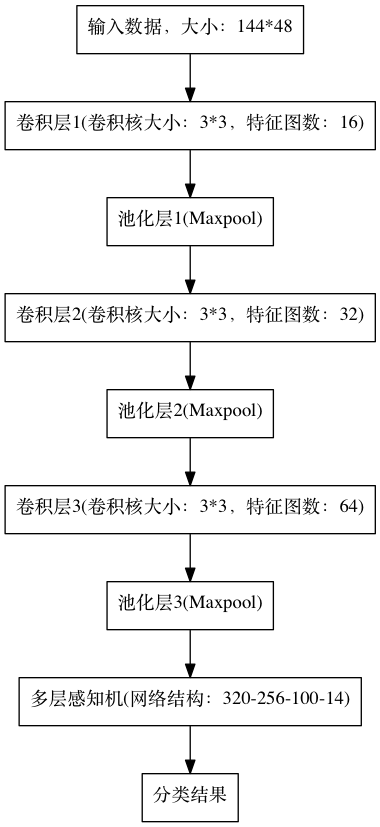
自编码器＋多层感知机

* + 1. 卷积神经网络＋多层感知机

由于卷积神经网络的输入就是需要经过二维卷积核进行卷积的，所以原来的二维结构的数据不需要进行铺平操作，直接以二维的模式进入网络进行学习就可以了。而且，由于这个数据的其中一个维度是时间，所以使用这个卷积网络还可以同时把时间维度的信息也考虑进来，使得整个模型能够更好地对时间维度的信息进行建模。由后面的多层感知机的输入神经元的大小可以看出来，原数据提取完特征之后只有 320 个变量，再输入到结构为[320-256-100-14]的多层感知机即可进行分类，从多层感知机的结构可以看出来，这个特征提取的卷积神经网络大大减轻了分类器的计算压力。这个模型结构如图3-8所示。

* + - 1. 加入正则化环节

为了加快模型的收敛，本文没有使用常规的在代价函数里加入惩罚环节来加强模型的泛化能力。而是使用了 Max-Norm 算法来对过大的权值进行惩罚。也就是把 L2 范数超过一定阈值的神经元的权值除以范数来抑制它变得过大。



卷积神经网络＋多层感知机模型结构

1. 模型实验设计与验证

本章会介绍比较一些编程框架，然后介绍用来进行试验验证的数据集，最后给出各个模型的实验结果并对不同模型下的准确率进行比较、分析。

* 1. 框架选择

本文中描述的算法，需要使用梯度下降的算法进行模型的优化。而建立出来的模型比如多层的神经网络，需要使用后向传播的算法，通过链式求导来得到各层链接的参数的导数。这里面有两个比较难以解决的问题：  
1.模型变得庞大之后，需要进行优化的参数的数量也会变得很大，计算耗费的时间非常多。  
2.导数计算的复杂程度会随网络层数的增加，还有网路里使用的算法的变化而变得越来越复杂；

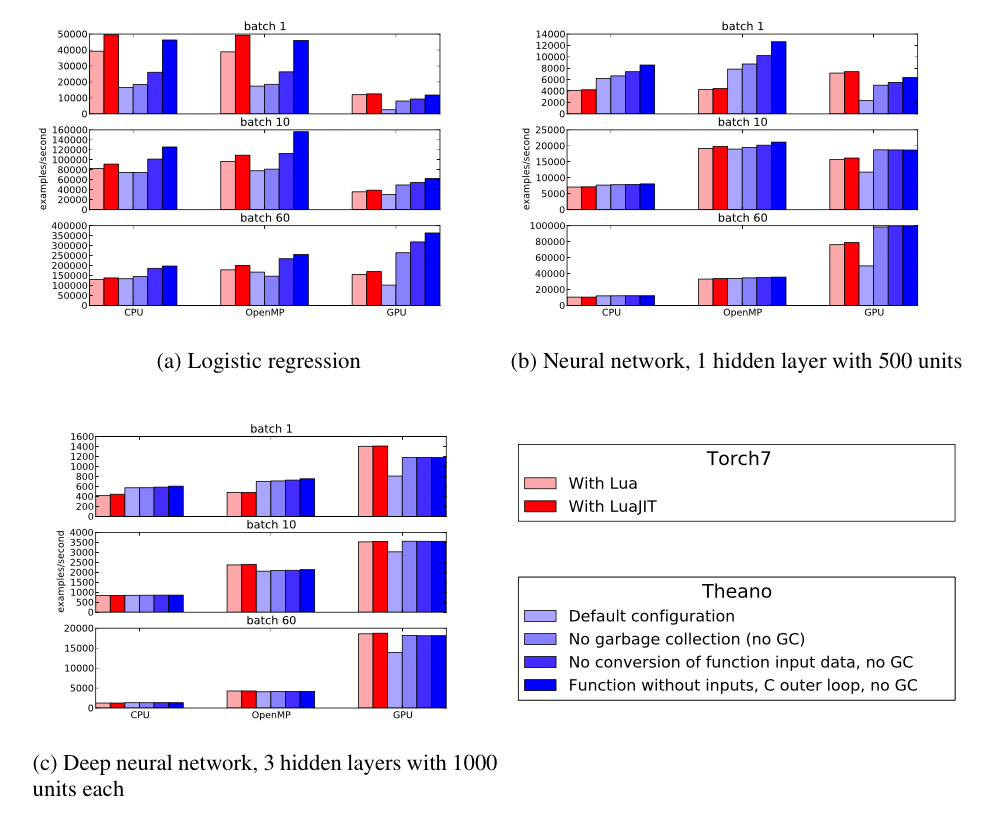
第一点可以用深度学习里的相关内容，比如自编码器里的逐层训练，还有减少训练参数的特性的卷积神经网络来减缓。但是也不能解决这个问题，所以，需要有一个能够支持高性能的计算的平台来减少做计算实验的时间，加快模型的迭代和设计。目前的高性能的计算平台主要依靠分布式的计算机集群还有 GPU 进行计算。而目前对于本文做实验的环境而言，最容易获得的高性能计算平台是基于高性能显卡的 GPU 计算平台。而要在编程的时候使用到 GPU 来进行算法的加速，就需要花精力训练相关的编程架构的知识，比如 NVIDIA 的 CUDA 框架。

而第二点则可以通过编程来实现一些自动进行函数的求导之类的功能来辅助进行网络的设计，从而把研究者的精力从细节里解放出来，投入到模型的设计上去。但是如果让程序要能够设计复杂的计算过程，就需要有类似惰性求值的机制来把具体的计算推迟，让程序能够在不用计算出来具体的值，而是在开始计算之前的过程就能得到计算过程的式子的信息来进行其他的操作，比如通过知道代价函数的构造对权值求导。

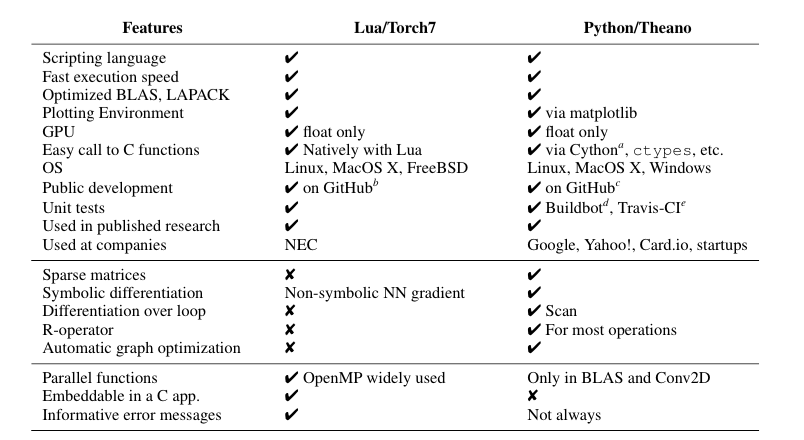
两点都是为了更好地去实验、设计模型。

常用的进行机器训练使用的框架，有 Theano，Torch7，MATLAB 等。MATLAB 在使用小规模的数据进行验证的算法设计和实验的时候非常合适，可是当数据量变大后，要使用 GPU 去加速计算会比较麻烦，而且当模型变复杂之后，就要花时间去自行对模型中的参数进行求导。所以，在做实验的时候优先选择使用 GPU 方便而且能够方便进行求导等高阶操作或者现成的资源比较丰富的框架，MATLAB 暂时不予考虑。

下面，对两个比较常用的计算框架 Torch7 和 Theano 进行简单的比较：



Torch7与Theano性能比较[42]



Torch7与Theano特性比较[42]

由上面的图可以看出 Theano 和 Torch7 这两个框架各有特点，而 Theano 是使用 python 实现的，现成的语言和工具会比较丰富，而且 Theano 是深度学习较早的库之一，由深度学习三大先驱Geoffrey Hinton(Google)、Yann LeCun(Facebook) 和 Yoshua Bengio 开发的，不少最新的神经网络方面的论文提出的算法都会被很快地纳入这个框架中。

所以，最后本文选择的是使用 Theano 框架来进行实验程序的编写平台。

* 1. 数据集

本文中，用来作为标准并且进行试验的数据集是康奈尔大学的 CAD-60 数据集，这个数据集是用微软的 Kinect 传感器拍摄下来的，由人做特定的动作的 RGB-D 视频序列构成。而且数据集里还有预先获取到的人的骨架模型，本文中的实验就是基于数据集里的骨架模型的序列。

CAD-60这个数据集有如下特点：

1. 60个RGB-D的视频（24~30 fps）；

2. 4个对象：两男、两女；

3. 5个不同的环境：办公室、厨房、卧室、浴室和起居室；

4. 12种活动：漱口、刷牙、戴隐形眼镜、谈电话、喝水、开药丸盒子、烹饪（切菜）、烹饪（搅拌），在沙发上聊天，在沙发上放松，在白板上写东西，在电脑前工作。

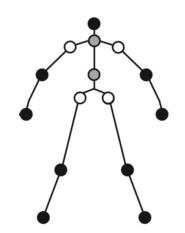
在CAD-60数据集中具体出现的图片可以参照下图：



CAD-60 图片内容样例

CAD-60序列图片样例（共1351帧的刷牙动作）



CAD-60骨架数据结构[19]

上图所示数据集中的人物骨架是从 Kinect 的采集程序里直接获取，它们的排列是和视频里的每一个帧一一对应的。骨架中一共有 15 个关节的三维空间坐标，坐标是以 Kinect 摄像头为坐标系零点的绝对三维坐标值，其中 11 个关节有它所对应的关节的转向角的 的矩阵。所以，如果把每一帧中的骨架的信息当作一个数据点的话，每个点的数据的数量为 144 个，而且都是 32 位的浮点数，同时，使用普通的线性变换吧所有的点的所有数值都做了一次归一化，以保证后面训练算法的有效：

（4-1）

数据集里，每个人的每个动作对应一个完整的视频（骨架）序列，长度在几十秒左右。为了让训练和测试的数据集更多，对每一个长度达二、三千帧的序列按每 48 帧一组进行了切片并打标签的操作。并且按照取其中三个人作为训练集，第四个人作为测试集的方式分成了两大组数据。处理完之后，得到了一个大的数据集，数据集中的每个样本都是二维的 的数据，样本具体数据结构如下图所示：



实验数据集样本数据结构

样本的数据是一个二维的结构，上图中行坐标的数据是144个浮点数坐标，分别是15个关节的三维空间坐标以及11个特殊关节之间的3\*3的旋转角矩阵，共计15×3+11×3×3=144个数。 纵坐标是时间的维度，也就是48个连续的帧堆叠起来。

* 1. 实验准备
     1. 实验环境

以下实验是在 AMD FX-4130 3.8Ghz 4 Cores/8GB-DDR3/Ubuntu14.04-x64 的环境下，使用 python2.7.9 在 Theano 的框架下编写的程序来做出来的。

选择 Ubuntu 作为主要的实验操作系统是因为它是基于 Liunx 的，而且拥有很多方便的工具链，可以方便地在本地平台进行工具的编译和安装。这里与 windows 的区别在于，有很多工具是在 Linux 上开发出来，以源代码的形式发布的。它们如果要运行在 windows 上，需要由其他人把它们编译成安装包，这些安装包经常会有 32位和 64位上的兼容问题，而使用原生的 Linux 环境来编译安装这些工具包，就可以与环境最大程度的兼容，减少出错的可能。

* + 1. 实验编程平台

首先在 python 的主页获取源代码，编译安装了最新版本的 Python (2.7.9) ，然后使用 ubuntu 下的 apt-get 包管理工具安装 BLAS 矩阵运算库。随后在 github 上获取 Theano 的源码，使用 python 的包管理工具 pip 进行编译安装。在这个过程中 pip 会自动检测所需要的依赖的工具包来下载安装，途中有一些不能自动获取的就要手动到 github 找到相应的源代码进行编译安装。至此 Theano 的基本安装就结束了。但是为了加速计算，需要使用显卡里的 GPU 单元来加速，所以需要安装 Nvidia 公司的 CUDA 并行计算框架。这个也同样是在 Nvidia 公司的主页下载 CUDA 的安装包进行安装。然后在 Theano 的配置文件中加入

[global]

floatX = float32

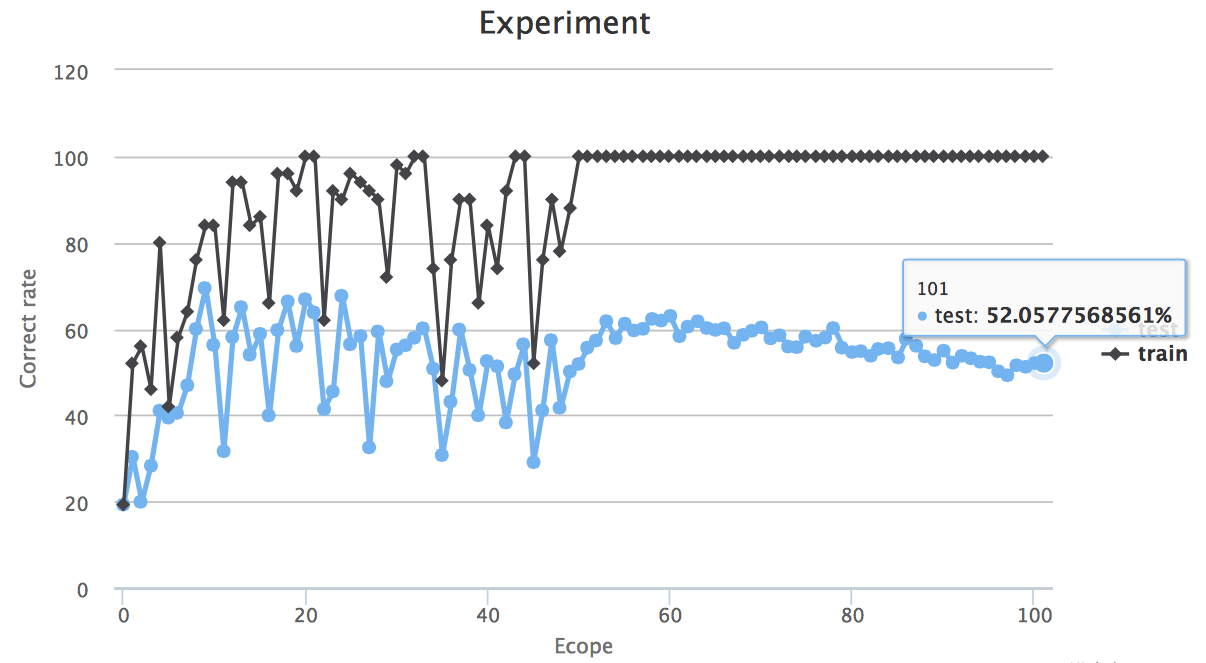
device = gpu0

来启用 GPU 加速。在此之后，使用 Theano 来进行的计算都会自动调用 GPU 进行加速而不需要额外的编程。

在使用 Theano 的编程的时候就是把下面的模型翻译成对应的矩阵运算或者是其他的代数运算。

比如进行 BP 单隐含层的神经网络的编程的时候，只需要定义输入层、隐含层、输出层的三个向量，然后定义他们之间的映射的矩阵，然后通过矩阵乘法连接起来实现前向传播。值得注意的是， Theano 下面的所有计算都不会立刻生效，而是先把计算的步骤记录下来保存在变量上，当需要把变量输出的时候才会真正地进行计算。当需要进行后向传播的时候只要调用 grad 函数，即可自动根据保存在变量上的公式进行求导。比如当已知神经网络的代价函数的结果 cost 还有隐含层到输出层的权值矩阵 W，这个时候只要调用 tensor.grad(cost, W) 函数即可求得 W 关于 cost 的导数。这样就可以加速实验模型的编写。

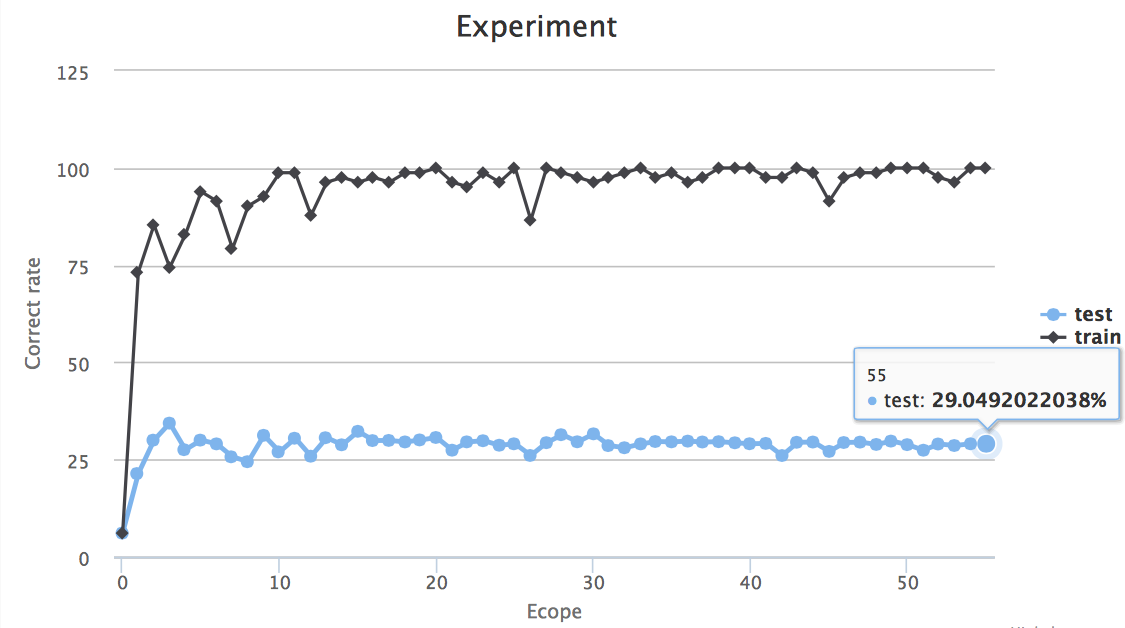
* 1. 实验结果
     1. 5层的多层感知机



多层感知机实验结果

上图中的横坐标是训练的代数（ecope）， 所有来自训练集的各个 batch 的数据全进入模型训练完一次算是一代。而图中的纵坐标则是训练的正确率，每一次训练完一代的数据都把测试集的数据放进模型分类一次，然后看分类的正确率。

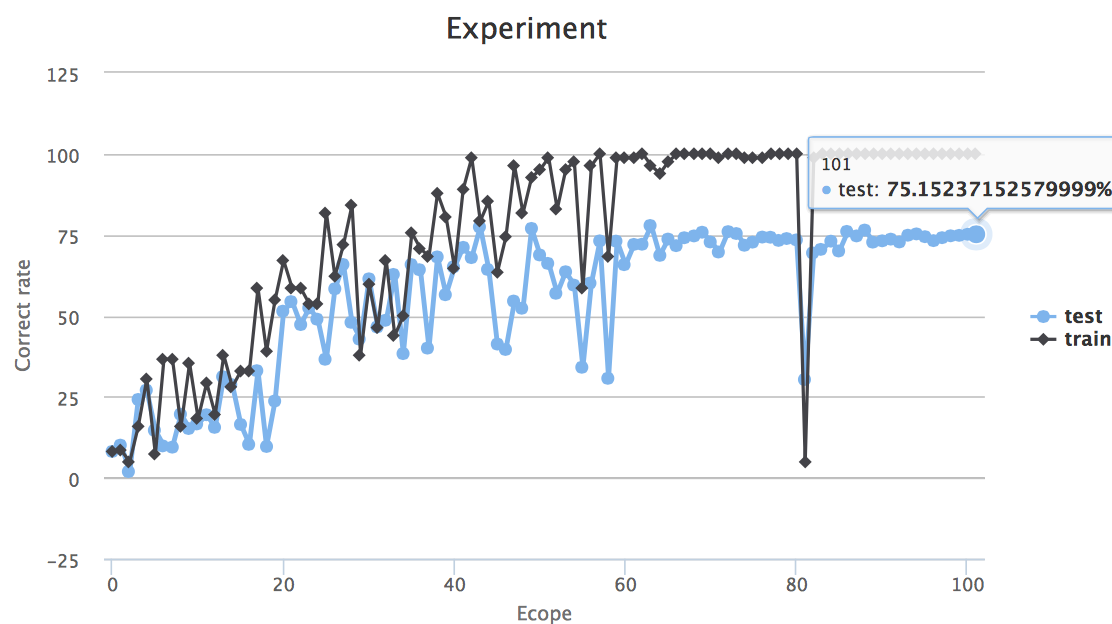
* + 1. 自编码器＋多层感知机



自编码器＋多层感知机实验结果

图4-8中的实验结果确实比上一个模型的要差很多。可能就是因为模型的表达能力比较弱的原因。又多加了一层自编码器在原来的自编码器后面，训练出来的结果是网络是发散的，分类结果是随机的，可能对于这个数据，使用自编码器来进行预训练会不合适。

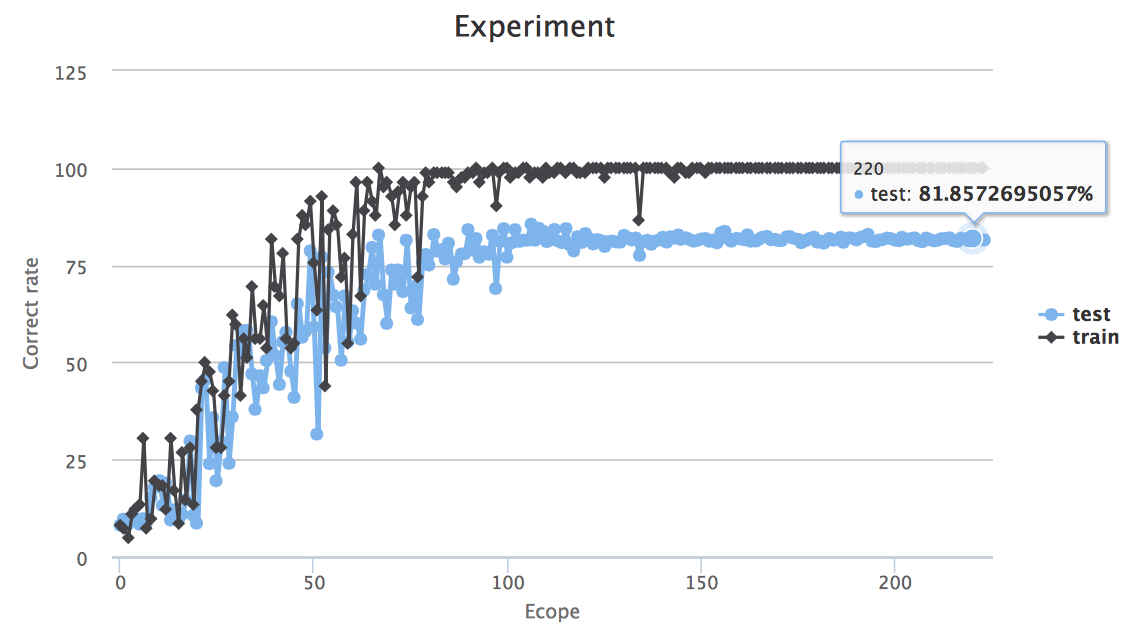
* + 1. 卷积神经网络＋多层感知机



卷积神经网络＋多层感知机实验结果

这个模型的结果非常的好，说明使用卷积神经网路来进行特征的有监督学习是非常有效的策略。可是可以看到，测试准确率的提升在训练准确率达到 100% 之后就停止了，说明模型已经陷入了局部的最优解里，也就是模型已经出现了过拟合的现象，为了提高模型模型泛化能力防止它过拟合，需要使用加入一些正则化的环节来加强模型的泛化能力。

* + 1. 加入正则化环节



卷积神经网络＋多层感知机模型加入正则化环节后实验结果

为了加快模型的收敛，本文没有使用常规的在代价函数里加入惩罚环节来加强模型的泛化能力。而是使用了 Max-Norm 算法来对过大的权值进行惩罚。也就是把 L2 范数超过一定阈值的神经元的权值除以范数来抑制它变得过大。加入了这个环节的模型的训练结果如上图所示。可以看出来，加入了正则化环节之后，模型训练的时间变长了，而且训练数据集的训练识别率也更难达到 100% 了。但是，相对的，测试数据集的识别率提高了不少，说明正则化环节对于抑制过耦合非常的有效。

* 1. 实验结果分析
     1. 同行结果对比分析

下表为针对 CAD-60 数据集的同行论文的结果。

同行算法准确率对比表

|  |  |  |
| --- | --- | --- |
|  | New Person | |
| Algorithm | Precision (%) | Recall (%) |
| Shan, Akella, ARSO 2014 [23] | 93.8 | 94.5 |
| Zhu, Chen, Guo, IVC 2014 [12] | 93.2 | 84.6 |
| Parisi, Weber, Wermter, Front. Neurobot. 2015 [43] | 91.9 | 90.2 |
| Faria, Premebida, Nunes, RO-MAN 2014 [22] | 91.1 | 91.9 |
| Zhang, Tian, NWPJ 2012 [44] | 86 | 84 |
| 本文 | 81.8 | - |
| Koppula, Gupta, Saxena, IJRR 2012. [45] | 80.8 | 71.4 |
| Gupta, Chia, Rajan, MM 2013 [46] | 78.1 | 75.4 |
| Gaglio, Lo Re, Morana, HMS 2014 [19] | 77.3 | 76.7 |
| Ni et al., Cybernetics 2013 [15] | 75.9 | 69.5 |
| Wang et al., PAMI 2013 [47] | 74.7 | - |
| Yang, Tian, JVCIR 2013 [20] | 71.9 | 66.6 |
| Piyathilaka, Kodagoda, ICIEA 2013 [43] | 70 | 78 |
| Sung et al., AAAI PAIR 2011, ICRA 2012. [26] | 67.9 | 55.5 |
| Ni, Moulin, Yan, ECCV 2012 [48] | 65.32 | - |

可以看到本文的结果还是十分具有竞争力的，可是和最好的结果还有一点距离，不过由于本文只使用了骨架模型作为唯一的输入，而且特征的提取上没有加入更多先验的信息。所以，如果在本文使用的模型的基础上加入更多的特征，比如 SIFT 等人工特征，模型的性能应该还会有很大的改进空间。

关于上表中的一些论文是使用精确率(Precision)和召回率(Recall)来描述其结果的，另外的是与本文一样直接用准确率(Accuracy)作为实验结果的。–

这三个评估标准的定义如下：

样本分类类型定义

|  |  |  |
| --- | --- | --- |
|  | 相关，正类 | 无关，负类 |
| 被检索到 | TP | FP |
| 未被检索到 | FN | TN |

Precision:

（4-2）

Recall:

（4-3）

Accuracy:

（4-4）

* + 1. 本文实验结果分析

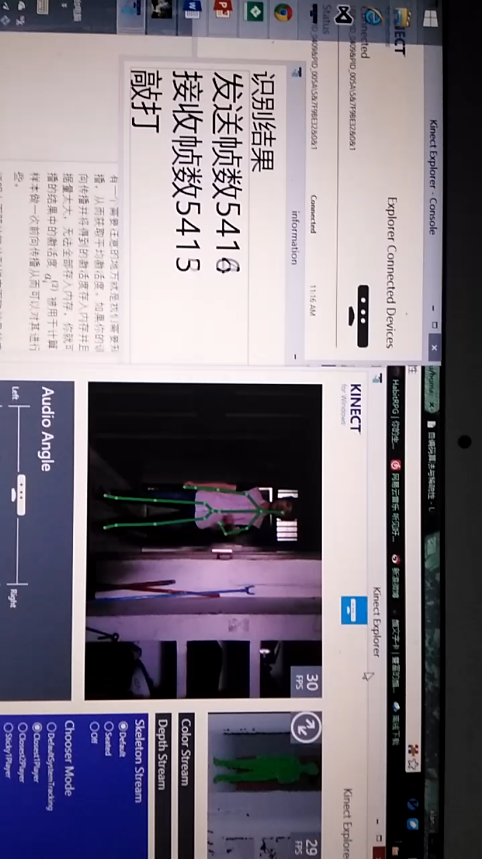
实验结果对照表

|  |  |  |
| --- | --- | --- |
| 使用算法 | 准确率  (%) | 训练时间  (s) |
| 5层的多层感知机 | 52.0 | 3min 56s |
| 自编码器＋多层感知机 | 29.0 | 42s |
| 卷积神经网络＋  多层（3层）感知机 | 57.2 | 10min 20s |
| 卷积神经网络＋  多层（4层）感知机 | 75.2 | 14min 21s |
| 卷积神经网络＋  多层（4层）感知机＋  Max-Norm正则化环节 | 81.8 | 21min 30s |

上表的实验里还有一些上文的模型中没有提及的，用来选择参数用的实验。由上面的各个实验可以看出来，结果最好的模型是加入了正则化环节的卷积神经网络＋多层感知机。而另一个深度学习的算法——自编码器不太适用于当前的数据集，可能是因为这个自编码器的算法的对于时序数据的提取特征的能力不是太强，需要用一些方法去加强这个自编码器算法对于时序数据的处理能力。比如加入一些用来加强前后神经元相关性计算能力的环节。

* 1. 基于深度模型的动作识别软件开发

实验中最好的模型是：卷积神经网络＋多层（4层）感知机＋Max-Norm正则化环节。基于这个模型，采集了一个简单的包含有站立、刷牙、敲打等动作的数据集，并且根据使用数据集训练好了上述的模型，根据这个模型，使用 c# 语言在 .Net 4.5 的框架下开发了一个演示的软件，下图为该软件的操作界面：



基于深度模型的人体运动识别演示软件

这个软件是基于 CS 模型的，在远程的服务器上运行根据训练好的有基于 Theano 编写的模型来进行人体运动的识别的服务端程序。本地运行的这个程序根据kinect 采集的骨架数据序列化之后通过网络传输到远程的服务器进行识别，然后服务器把识别出来的结果发送回来。由于骨架模型的数据量少，所以这个演示软件所占用的网络带宽非常少，能够轻易地进行实时的识别。

1. 总结
   1. 总结

本文先是总结了现有的人体运动类型识别的算法的普遍框架：人工提取特征然后再使用分类器进行分类。提出了使用深度学习理论进行自动的特征提取，并且使用深度学习里面的自编码器和卷积神经网络进行自动的特征提取。并且给出了相应的实验结果，最终得到了一个有着不错效果的，人体运动类型识别的模型——卷积神经网络＋多层感知机。

* 1. 展望
     1. 加入提前停止法

由各个训练实验的结果的曲线来看，在使用训练的数据集训练到训练的准确率达到 100% 之后，测试集的测试的准确率也往往到达了最大值，往后面就开始以缓慢的速度下降。由此可见，在训练集的准确率达到 100% 之后模型再训练下去，泛化能力反而会下降，逐渐出现过拟合的现象。所以还应该在训练的算法里加入提前停止的环节，也就是提前停止法，具体的，可以根据[49]展开研究。

* + 1. 更换其他分类器

由实验可以看到卷积神经网络具有非常优良的提取特征的能力，能够提高分类模型的分类能力。而在论文里就有不少使用 SVM、朴素贝叶斯等经典的分类器取得非常好的分类结果的模型。而且，使用有效的特征提取算法也可以减少像 SVM 这种算法训练的时候所需要的时间。所以，可以综合卷积神经网络的特征提取能力和其他分类器的分类能力去创造新的模型。

而综合两者有两种思路，一种是通过把后向传播算法推广到其他的分类器，使得可以直接使用这些分类器去传播误差，从而训练要使用得到的卷积神经网路。

另一种思路是使用浅层的使用ReLU激活函数的神经网络，直接使用有监督学习对卷积神经网路进行训练。之所以使用浅层的神经网络是因为可以一来减少后面的网络结构对卷积神经网络的训练的影响，二来可以减少误差传递过程中的损失，避免误差传递到卷积神经网络的表层时已经变得很小，从而影响卷积神经网络的训练。而所说的这两种方法都需要做实验去确定效果和是否行得通。

参考文献

[1]黎洪松, 李达, L. I. Hong-Song, and L. I. Da, “人体运动分析研究的若干新进展,” *PATTERN RECOGNITION AND ARTIFICIAL INTELLIGENCE*, vol. 22, no. 1, pp. 70–78, 2009.

[2]凌志刚, 赵春晖, 梁彦, 潘泉, 王燕, L. Zhi-gang, Z. Chun-hui, L. Yan, P. A. N. Quan, and W. Yan, “基于视觉的人行为理解综述,” *APPLICATION RESEARCH OF COMPUTERS*, vol. 25, no. 9, pp. 2570–2578, 2008.

[3]李斌, 孙怀江, L. I. Bin, and S. U. N. Huaijiang, “基于半监督距离学习的运动相似性度量算法,” *Computer Engineering*, no. 11, pp. 178–182, 2014.

[4]L. Chen, H. Wei, and J. Ferryman, “A survey of human motion analysis using depth imagery,” *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1995–2006, 2013.

[5]D. Weinland, M. Özuysal, and P. Fua, “Making action recognition robust to occlusions and viewpoint changes,” in *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*, 2010, vol. 6313 LNCS, pp. 635–648.

[6]L.-c. Chen, J.-w. Hsieh, C.-h. Chuang, C.-y. Huang, and D. Y. Chen, “Occluded Human Action Analysis Using Dynamic Manifold Model,” in *International conference on pattern recognition*, 2012, pp. 1245–1248.

[7]M. B. Holte, T. B. Moeslund, N. Nikolaidis, and I. Pitas, “3D Human Action Recognition for Multi-view Camera Systems,” in *2011 international conference on 3D imaging, modeling, processing, visualization and transmission*, 2011, pp. 342–349.

[8]D. Weinland, E. Boyer, and R. Ronfard, “Action recognition from arbitrary views using 3D exemplars,” in *Proceedings of the iEEE international conference on computer vision*, 2007, pp. 1–7.

[9]M. C. Roh, H. K. Shin, and S. W. Lee, “View-independent human action recognition with Volume Motion Template on single stereo camera,” *Pattern Recognition Letters*, vol. 31, no. 7, pp. 639–647, 2010.

[10]M.-y. Chen, “MoSIFT : Recognizing Human Actions in Surveillance Videos MoSIFT : Recognizing Human Actions in Surveillance Videos.” 2009.

[11]R. Poppe, “A survey on vision-based human action recognition,” *Image and Vision Computing*, vol. 28, no. 6, pp. 976–990, 2010.

[12]Y. Zhu, W. Chen, and G. Guo, “Evaluating spatiotemporal interest point features for depth-based action recognition,” *Image and Vision Computing*, vol. 32, no. 8, pp. 453–464, 2014.

[13]L. Rybok, B. Schauerte, Z. Al-Halah, and R. Stiefelhagen, “‘Important stuff, everywhere!’ Activity recognition with salient proto-objects as context,” *2014 IEEE Winter Conference on Applications of Computer Vision, WACV 2014*, pp. 646–651, 2014.

[14]N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proceedings - 2005 iEEE computer society conference on computer vision and pattern recognition, cVPR 2005*, 2005, vol. I, pp. 886–893.

[15]B. Ni, Y. Pei, P. Moulin, and S. Yan, “Multilevel depth and image fusion for human activity detection,” *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1382–1394, 2013.

[16]R. Chaudhry, a. Ravichandran, G. Hager, and R. Vidal, “Histograms of oriented optical flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions,” *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1932–1939, 2009.

[17]张永强 and Z. Yong-qiang, “基于Hu不变矩特征优化的人体运动姿态识别算法,” *Computer Science*, vol. 41, no. 3, pp. 306–309, 2014.

[18]L. Piyathilaka and S. Kodagoda, “Gaussian mixture based HMM for human daily activity recognition using 3D skeleton features,” *Proceedings of the 2013 IEEE 8th Conference on Industrial Electronics and Applications, ICIEA 2013*, pp. 567–572, 2013.

[19]S. Gaglio, G. L. Re, S. Member, and M. Morana, “Human Activity Recognition Process Using 3-D Posture Data,” pp. 1–12, 2014.

[20]X. Yang and Y. Tian, “Effective 3D action recognition using EigenJoints,” *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 2–11, 2014.

[21]K. Lertniphonphan, S. Aramvith, and T. H. Chalidabhongse, “Human action recognition using direction histograms of optical flow,” *2011 11th International Symposium on Communications & Information Technologies (ISCIT)*, vol. c, no. Iscit, pp. 574–579, 2011.

[22]D. R. Faria, C. Premebida, and U. Nunes, “A Probabilistic Approach for Human Everyday Activities Recognition using Body Motion from RGB-D Images,” *IEEE International Symposium on Robot and Human Interactive Communication (Ro-Man)*, 2014.

[23]J. Shan and S. Akella, “3D Human Action Segmentation and Recognition using Pose Kinetic Energy,” 2014.

[24]N. Hu, G. Englebienne, Z. Lou, and B. Kr, “Learning Latent Structure for Activity Recognition \*,” pp. 1048–1053, 2014.

[25]C. Zhang and Y. Tian, “RGB-D camera-based activity analysis,” *…and Conference (APSIPA ASC), 2012 Asia- …*, 2012.

[26]J. Sung, C. Ponce, B. Selman, and A. Saxena, “Human Activity Detection from RGBD Images,” 2011.

[27]I. Arel, D. C. Rose, and T. P. Karnowski, “Deep Machine Learning — A New Frontier in Artificial Intelligence Research,” *IEEE Computational Inteligence Magazine*, vol. 5, no. 4, pp. 13–18, 2010.

[28]R. E. Bellman, *Chapter 6 Dynamic programming algorithms*, vol. 11. 1987, pp. 343–395.

[29]A. F. Marquand, S. De Simoni, O. G. O’Daly, S. C. R. Williams, J. Mourão-Miranda, and M. a Mehta, *Pattern classification of working memory networks reveals differential effects of methylphenidate, atomoxetine, and placebo in healthy volunteers.*, vol. 36. 2011, pp. 1237–1247.

[30]S. Wold, K. Esbensen, and P. Geladi, “Principal component analysis,” *Chemometrics and intelligent laboratory systems*, vol. 2. pp. 37–52, 1987.

[31]K. Pearson, “LIII. On lines and planes of closest fit to systems of points in space,” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, pp. 559–572, 1901.

[32]Y. Lecun, “Deep Learning Tutorial,” *ICML2013 Tutorial*, 2013.

[33]Y. Bengio, “Learning Deep Architectures for AI,” *Foundations and trends® in machine learning*, vol. 2. pp. 1–127, 2009.

[34]Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998.

[35]F. J. Huang and Y. LeCun, “Large-scale learning with SVM and convolutional nets for generic object categorization,” in *Proceedings of the iEEE computer society conference on computer vision and pattern recognition*, 2006, vol. 1, pp. 284–291.

[36]D. Kriesel, *A Brief Introduction to Neural Networks*. 2007.

[37]Y. A. LeCun, L. Bottou, G. B. Orr, and K. R. Müller, “Efficient backprop,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7700 LECTU, pp. 9–48, 2012.

[38]G. Hinton, “Dropout : A Simple Way to Prevent Neural Networks from Overfitting,” *Journal of Machine Learning Research (JMLR)*, vol. 15, pp. 1929–1958, 2014.

[39]G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” *arXiv: 1207.0580*, pp. 1–18, 2012.

[40]I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, “Maxout Networks,” *arXiv preprint*, pp. 1319–1327, 2013.

[41]S. Wang, “General constructive representations for continuous piecewise-linear functions,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 9, pp. 1889–1896, 2004.

[42]F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. Goodfellow, A. Bergeron, N. Bouchard, D. Warde-Farley, and Y. Bengio, “Theano: new features and speed improvements,” *arXiv preprint arXiv: …*, pp. 1–10, 2012.

[43]G. I. Parisi, C. Weber, and S. Wermter, “Self-Organizing Neural Integration of Pose-Motion Features for Human Action Recognition,” 2015.

[44]C. Zhang and Y. Tian, “RGB-D Camera-based Daily Living Activity Recognition,” *Journal of Computer Vision and Image Processing*, vol. 2, p. 12, 2012.

[45]H. S. Koppula, R. Gupta, and a. Saxena, “Learning human activities and object affordances from RGB-D videos,” *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 951–970, 2013.

[46]R. Gupta, A. Y.-S. Chia, and D. Rajan, “Human activities recognition using depth images,” *Proceedings of the 21st ACM international conference on Multimedia*, pp. 283–292, 2013.

[47]J. Wang, Z. Liu, Y. Wu, and J. Yuan, “Learning actionlet ensemble for 3D human action recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 914–927, 2014.

[48]B. Ni, P. Moulin, and S. Yan, “LNCS 7573 - Order-Preserving Sparse Coding for Sequence Classification,” pp. 173–187, 2012.

[49]L. Prechelt, “Early stopping - But when?” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7700 LECTU, pp. 53–67, 2012.

致谢

毕业设计从选题到现在已经过去了将近五个月，这段时间里面，我从接触一个陌生的领域开始学习，到完成毕业设计，学会了很多，也收获了很多。这一路以来得到了非常多的人的帮助，在此我要对他们表示最衷心的感谢。

感谢莫凌飞老师在我做毕业设计期间提供的帮助，他对我的严格要求，在我的研究方向上所做出的高屋建瓴的指导，在我出现问题时耐心地帮我整理思路分析问题，在我论文写作的时候细心地把论文打印出来，在上面一一批阅、提出修改意见，有时候还会帮我看论文的问题看到深夜，这些都让我获益匪浅。感谢邵思羽学姐在论文撰写期间提供的帮助和指导。

感谢大学期间参加比赛时的宋爱国、张文锦和崔建伟老师，还有在做 SRTP 项目的过程中指导过我的张小国老师，他们在我的学习过程中都给予了我极大的帮助，让我的工程能力得到了极大的提升。

感谢我的父母，即使他们无法理解我在干嘛，但还是在背后默默地支持我。

感谢我看的论文的作者们，你们的努力工作和聪明才智还有发表论文时的不厌其烦都极大地启发了我的思路。

感谢开源程序的贡献者们，他们的努力让我少走了很多弯路，少做了很多重复的劳动，让我能够专心地专注于我需要研究的问题上。

感谢在我通宵工作时没有因为我敲键盘声音太大而发出抱怨的室友们。感谢来自 CMU 的陈琦同学提供的关于编程的帮助。如果没有他我可以提前一个月完成论文。感谢空空如也的钱包，让我在论文的写作期间完全没有别的杂念。