

马砚文

2021011803

神经网络编程实践大作业

神经网络设计与编程

**基于CNN与resnet的EMG信号分类实验**



[1 实验背景和目的 2](#_Toc169534123)

[2 实验环境 2](#_Toc169534124)

[3 实验过程 2](#_Toc169534125)

[3.1 数据处理 2](#_Toc169534126)

[3.1.1 基础处理 3](#_Toc169534127)

[3.1.2 归一化与时间窗口 4](#_Toc169534128)

[3.1.3 小波变换 5](#_Toc169534129)

[3.2 分类器搭建 6](#_Toc169534130)

[3.2.1 简单CNN网络 6](#_Toc169534131)

[3.2.2 ResNet 7](#_Toc169534132)

[3.3 训练结果 7](#_Toc169534133)

[4 结论 9](#_Toc169534134)

基于CNN与resnet的EMG信号分类实验

1. 实验背景和目的

EMG（肌电图）信号是通过测量肌肉中的电活动来记录的。当肌肉纤绵被神经刺激激活时，它们会产生电信号，这些信号可以通过皮肤表面或直接在肌肉内部的电极采集到。EMG信号通常用来评估肌肉功能、监测神经肌肉疾病，或者在康复工程和生物力学研究中应用。

CNN（卷积神经网络）是一种深度学习算法，特别适合于处理具有网格拓扑结构的数据，如图像。CNN通过自动提取数据中的重要特征并进行分类或回归分析而广泛应用于图像识别、视频分析、自然语言处理等领域.

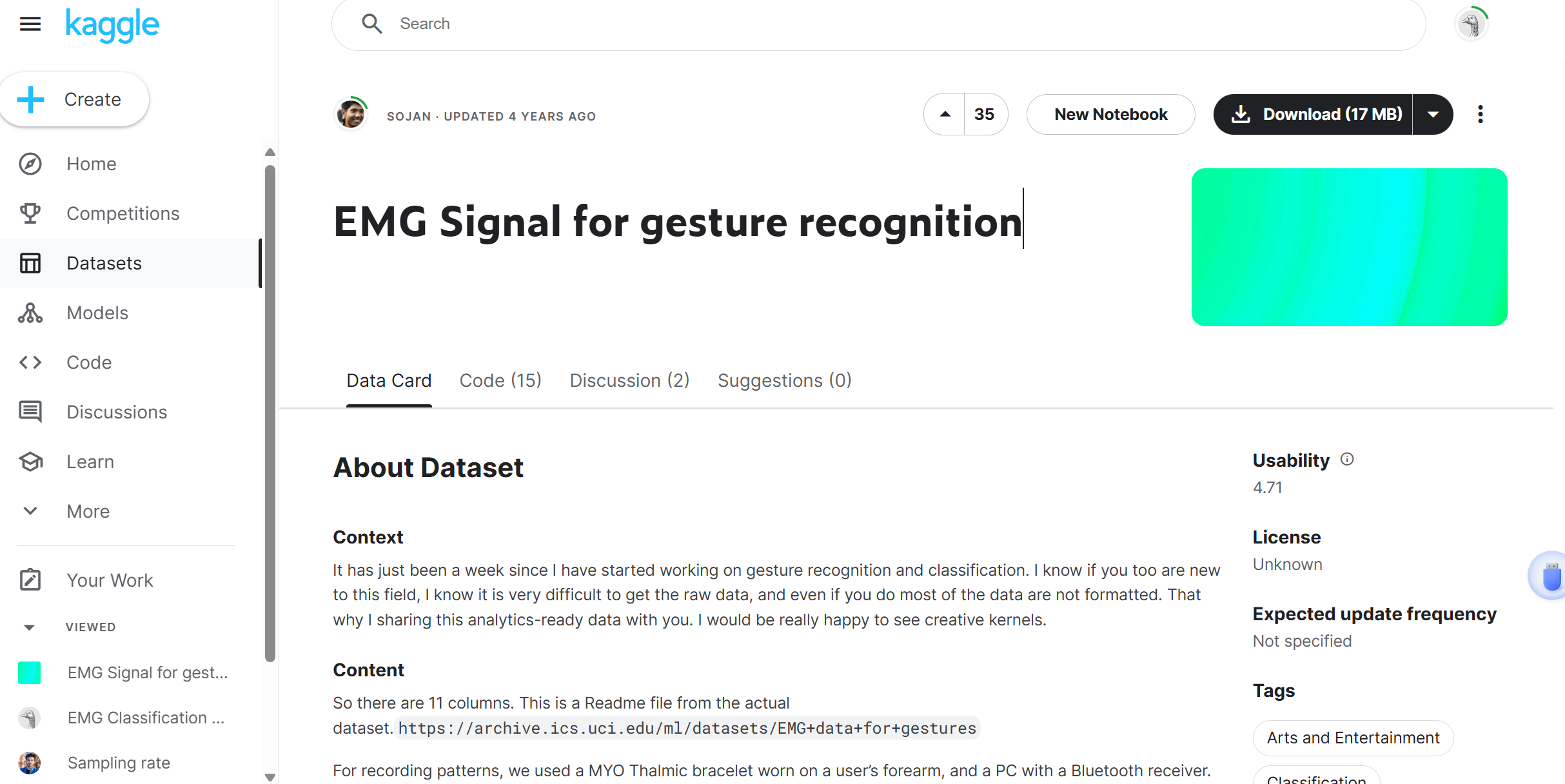
为了巩固神经网络设计与编程这门课程的知识点，加深对神经网络的理解程度及应用水平，在kaggle上寻找EMG信号分类数据集，进行神经网络编程分类实验。

1. 实验环境

本实验运行在autodl算力平台，NVIDIA GeForce RTX 3090 24G单卡，python3.8，使用pytorch编写神经网络。详细实验过程及配置见readme.md。

1. 实验过程
   1. 数据处理

数据集来自kaggle平台。该数据集主要用于手势识别和分类研究。它包含来自36位受试者在执行一系列静态手势时收集的原始肌电图（EMG）数据。这些数据是通过佩戴在用户前臂上的含有八个传感器的MYO Thalmic手环收集的。这些传感器可以同时获取肌电信号，并通过蓝牙接口发送到电脑。



1. EMG分类数据集[EMG Signal for gesture recognition | Kaggle](https://www.kaggle.com/datasets/sojanprajapati/emg-signal-for-gesture-recognition)

数据集包含11列数据，其中包括：

0. 时间（以毫秒为单位）； 1-8. 八个EMG频道的数据；10. Class，含义如下

|  |  |  |
| --- | --- | --- |
| Class | 含义 | 记录数 |
| 0 | 未标记数据 | 2725157 |
| 1 | 手部静止 | 253009 |
| 2 | 手握成拳 | 251733 |
| 3 | 手腕屈曲 | 251570 |
| 4 | 手腕伸展 | 250055 |
| 5 | 向桡侧偏 | 249494 |
| 6 | 向尺侧偏 | 243193 |
| 7 | 手掌伸展（并非所有受试者都执行此手势） | 13696 |

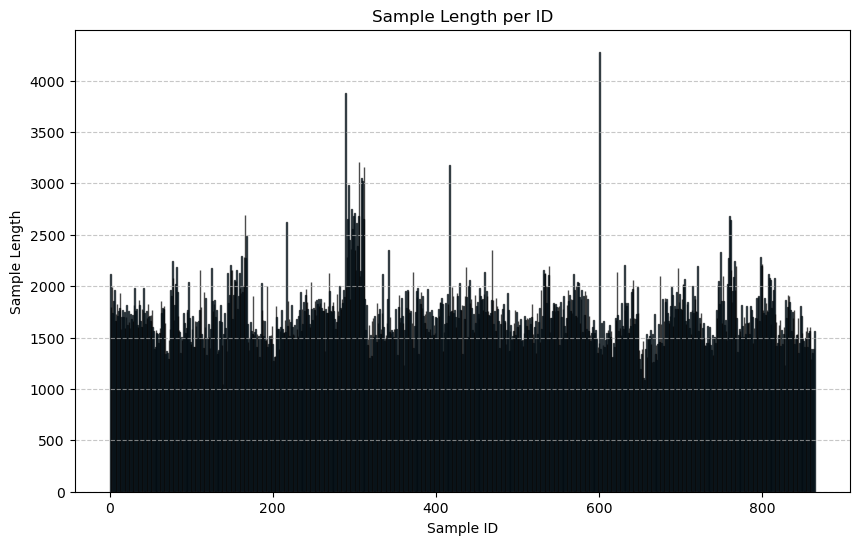
1. 每个标签的含义及样本数

此外，还有一个“label”列，用于标识执行实验的受试者。每位受试者都执行了七种手势，每种手势执行两遍。每个手势持续3秒，手势之间有3秒的暂停。

* + 1. 基础处理

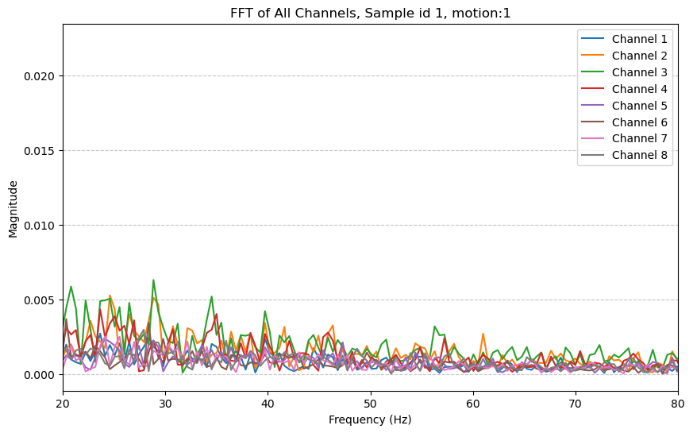
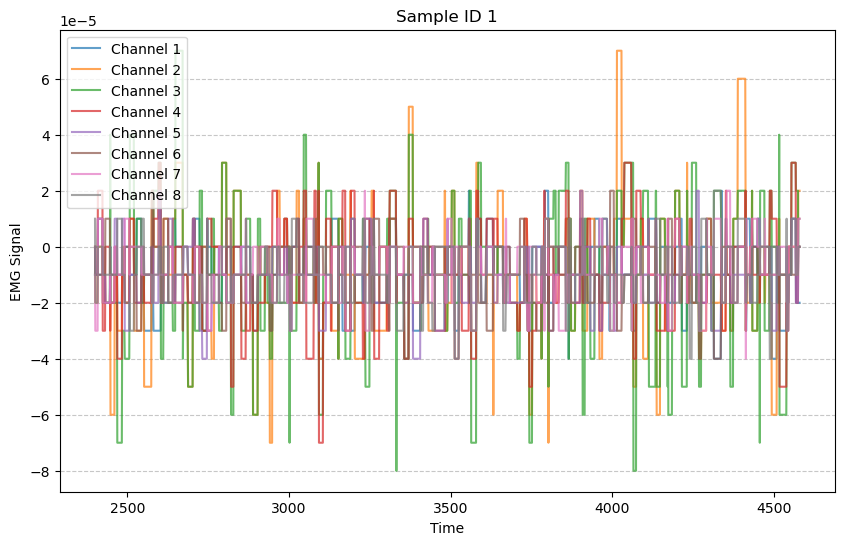
0类超出评价水平，7类也明显小于评价水平，为了保证数据分布相对均匀，直接删除这两类。对剩余类别进行热独编码。

原数据集的每条记录有8个通道的电信号峰值，加上对应的分类标签label，记录是以毫秒为采样单位的离散数据。直接进行分类会导致信号的一般时序特征丢失，因此考虑记录时间上相邻，且label相同的列。具体操作是添加一个”sample\_id”列。首先将未标记的数据段全部删除，形成分离的有效段，每段由许多相邻的采样单位记录组成，在确认采样时间是连续，且不存在阶跃后，将有效段赋予唯一”sample\_id”，以此识别每一记录时间段。



1. 处理后的有效段长度分布

由于此数据集是时序生理信号，可能会存在一些源自采集过程中的噪声，例如50hz工频噪声，基线漂移等常见问题，因此需要检查数据实际时序特征，并施加可能的滤波处理。



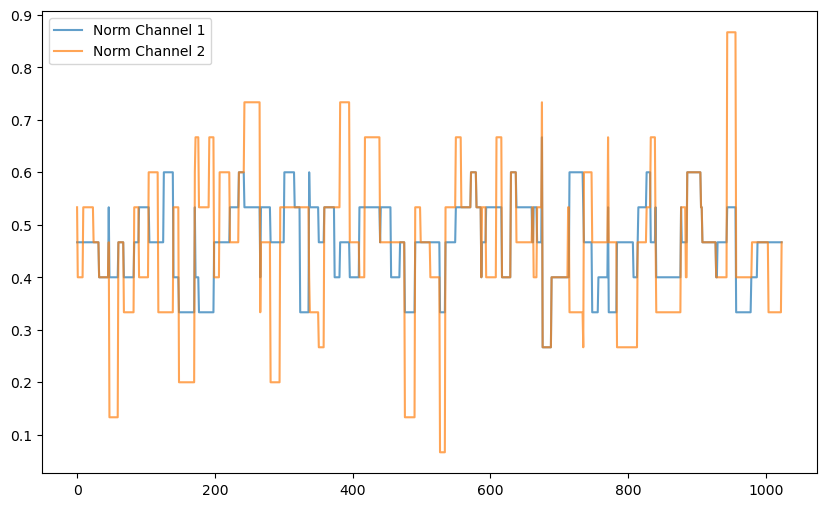
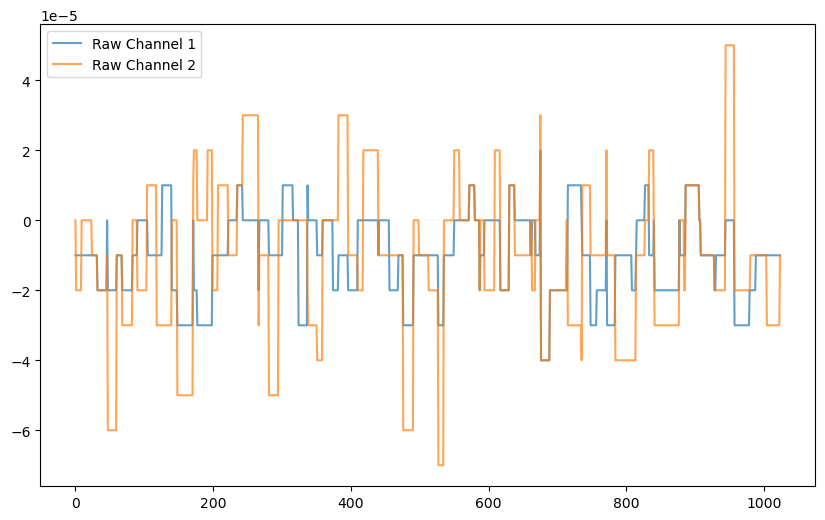
1. 对id为1的有效段进行可视化以及快速傅里叶变化的结果

遵循一般数字信号处理的方法对信号进行可视化以及快速傅里叶变换。可以看出，时序上信号分布平稳，不存在基线漂移，数值范围较小；频域上并不存在50HZ工频噪声，也不存在明显噪声频段。该数据集应该已经经过了适当的信号预处理，因此并不需要再进行额外的数字信号处理过程。

* + 1. 归一化与时间窗口

观察到数据范围较小，为了便于神经网络学习，将数据进行缩放及归一化处理。对于每个通道，进行最大最小归一化处理，将原始数据分布映射到0~1分布区间。方法如下：

之后，为了更好的适应神经网络的输入，方便网络训练，将数据用滑动窗口进行二次划分。对于每一段样本，取1024长度的窗口，以32长度的步长进行滑动划分。不够的部分将直接丢弃。



1. 原始数据分布（左），经过归一化处理的数据分布（右）

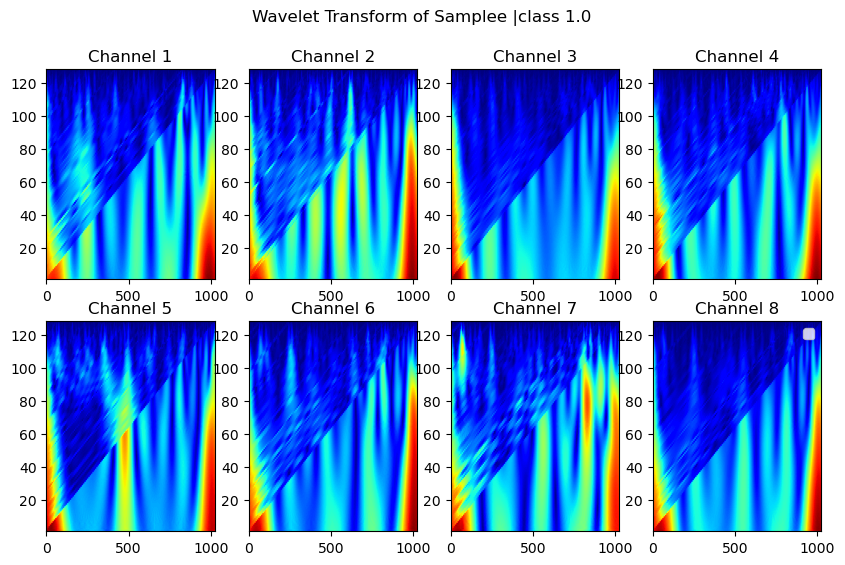
直观上可以看出，数据分布并无改变，但数值范围缩放到了适合神经网络输入的区间。

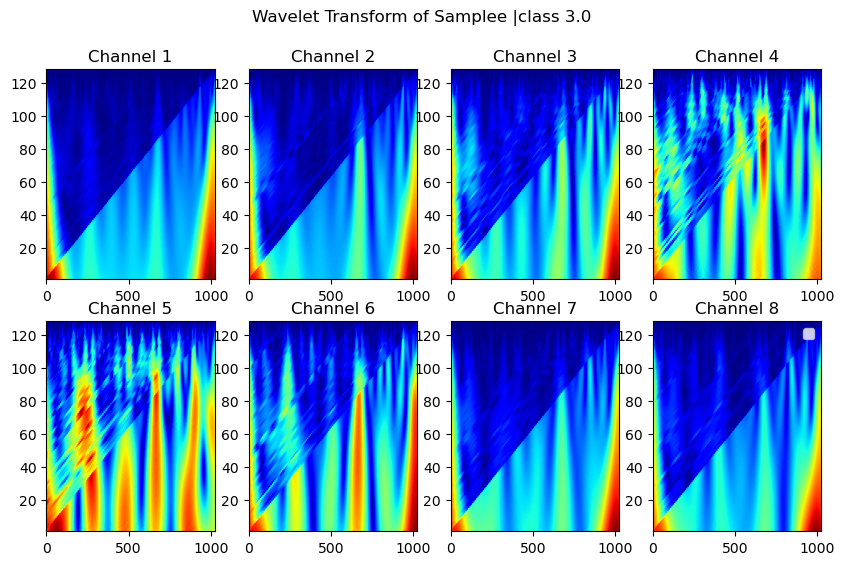
* + 1. 小波变换

小波变换是一种数学变换方法，用于将信号或图像分解成不同尺度（或频率）上的成分。与傅里叶变换不同，小波变换不仅能提供信号的频率信息，还能同时提供信号的时间信息，使其特别适用于分析具有不同时间尺度变化特征的非平稳信号。

使用小波变换，将单一的时序数据变为时频图，让神经网络捕捉时频图的不同之处，有助于分类的准确性提升。

这里使用连续小波变换CWT，连续小波变换通过在连续的尺度和平移参数上计算信号与小波母函数的卷积，生成一个二维的表示形式。考虑到实际输入的尺寸问题，将尺度变化设为0~127共128个尺度，与时间窗上的1024长度形成1024128的时频图像。





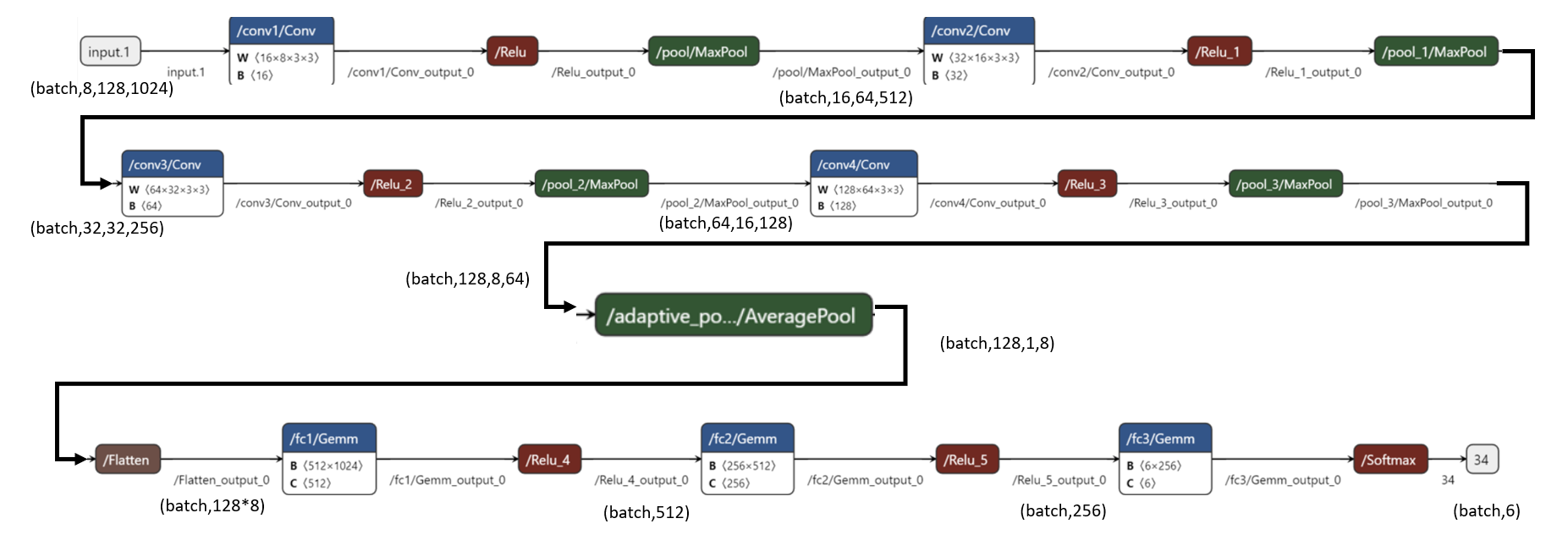
1. 在类别1，类别3样本上应用小波变换后形成的时频图像
   1. 分类器搭建

通过小波变换形成的时频图谱，十分适合使用图片分类模型进行处理。

* + 1. 简单CNN网络

卷积神经网络（CNN，Convolutional Neural Networks）是一种深度学习架构，广泛应用于图像和视频识别、推荐系统、图像分类、医学图像分析、自然语言处理等多种领域中。CNN 相对于其它标准的深度神经网络特别之处在于其能够直接对图像等高维数据进行处理，自动提取重要特征，无需手动特征工程。

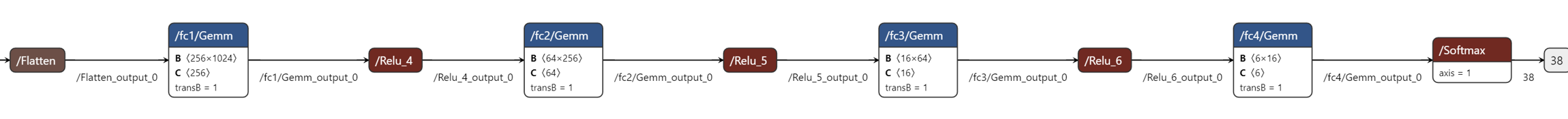
这里搭建简单的cnn分类网络。

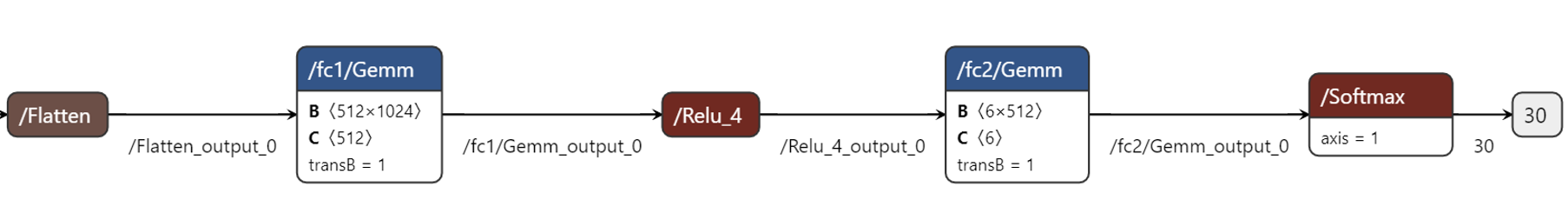
****

1. 简单cnn分类网络EMGClassifierV1

该网络由四个连续的卷积池化块组成，每个卷积和池化操作之间通过relu激活函数相连；之后经过一个自适应池化层，将数据调整为全连接层分类器能够接受的形式。最后通过一个两层隐藏层的多层感知机进行分类。多层感知机的激活函数均为relu，仅最后一项为适应热度编码，激活函数为sigmoid函数，以调整数据分布到0~1之间。

经过初次微调，发现性能并不足以达到预期目的，考虑减少分类器的复杂程度，分别增加，减少感知机的隐藏层数获得V2、V3模型。



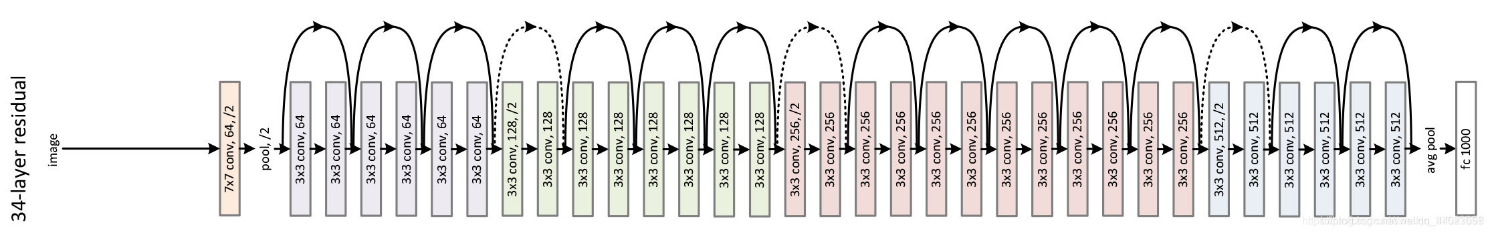


1. 改进模型（仅展示多层感知机部分）EMGClassifierV2（上）EMGClassifierV3（下）
   * 1. ResNet

ResNet（残差网络）是一种深度学习架构，最初由微软研究院的何恺明等人在2015年提出，用以解决深层神经网络训练中的退化问题——随着网络层次的增加，网络性能往往会饱和甚至退化。ResNet通过引入了所谓的“残差学习”（Residual Learning）来允许模型学习层间的差异，而非直接拟合一个期望的映射，这大大改善了深层网络的训练效率和最终性能。、

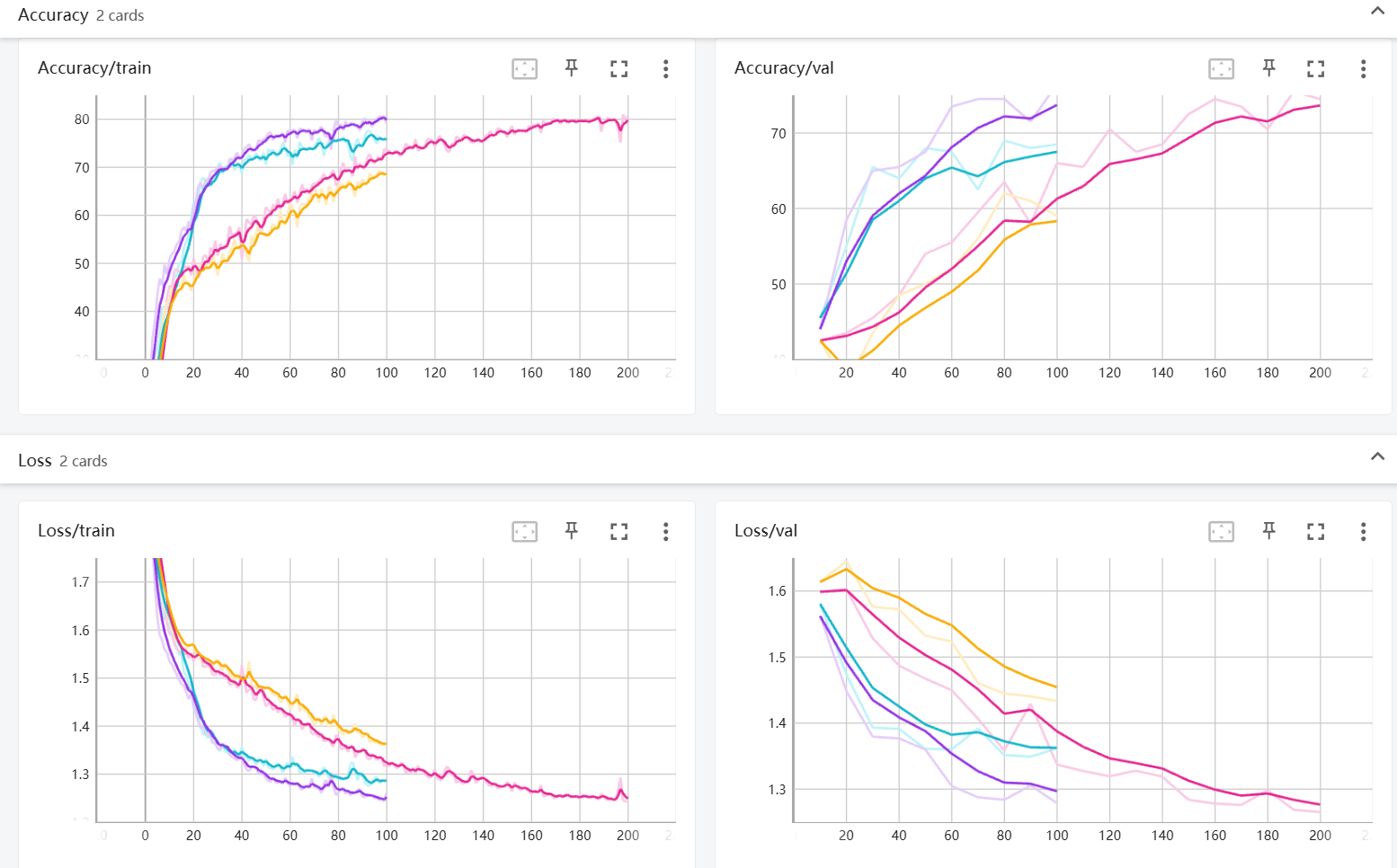
在 PyTorch中，实现了ResNet架构中较浅的一种变体resnet18，具体是具有 18 层的 ResNet。pytorch提供了未经训练的网络模型，便于用户在自己的应用中快速使用或进行进一步的训练。

这里选取resnet18进行基座模型进行实验。重写网络的输入输出层，将输入层调整为1024128，输出层调整为16，以适应此实验数据格式和分类任务。



1. Resnet18网络架构
   1. 训练结果

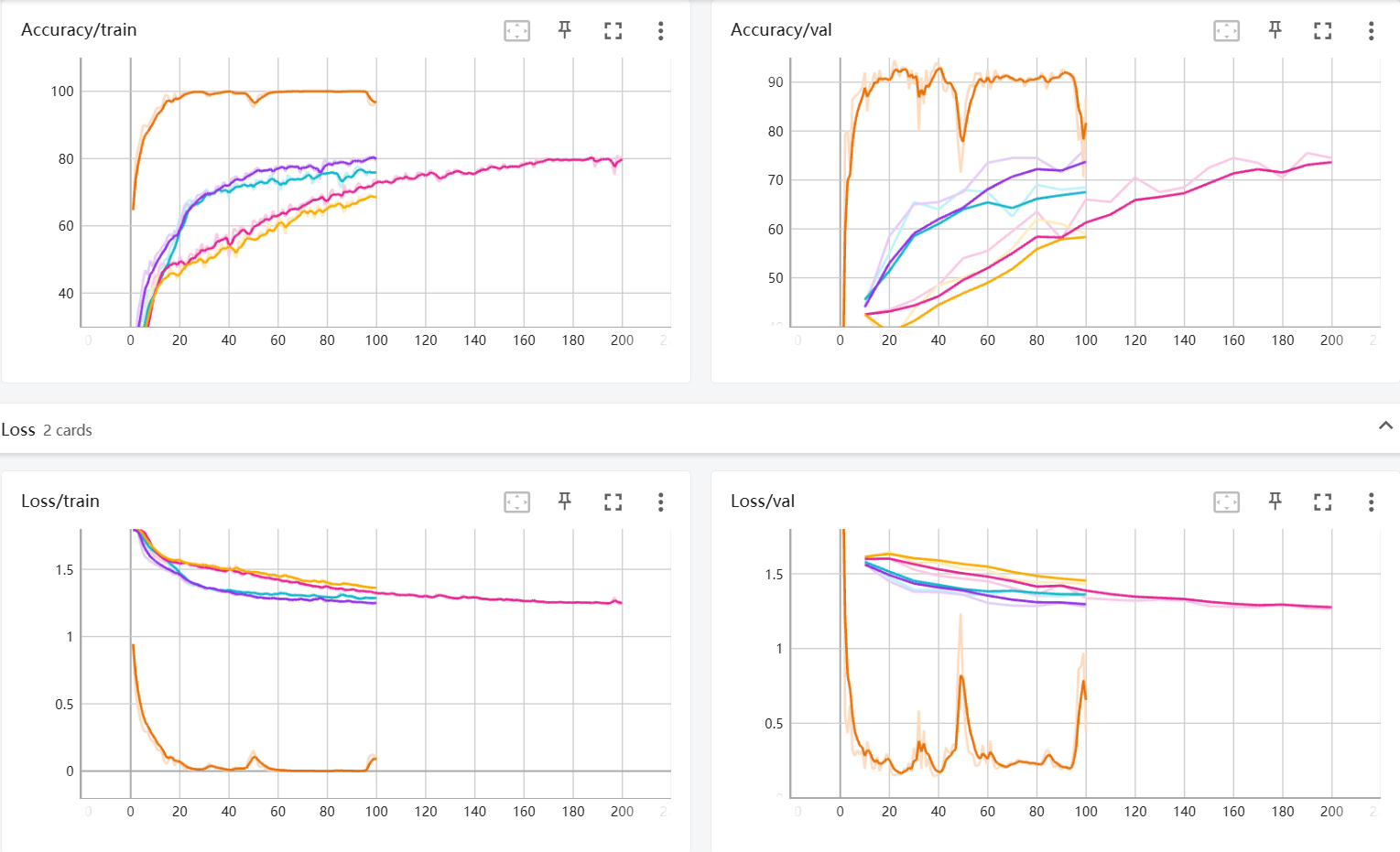
以100epoch为基准，批次同一设置为32，学习率为0.0001，优化器选取Adam。具体实验配置详见configs文件夹中的yaml文件。除resnet的实验之外，改变仅模型，训练轮次。



1. 对于简单cnn网络的训练结果。  
   ⚪：EMGClassifierV1⚪：EMGClassifierV1在batch=64的情况下训练200轮  
   ⚪：EMGClassifierV2⚪：EMGClassifierV3

最好的是EMGClassifierV3，但准确率也不足80%。

考虑到简单cnn分类器性能有限，可以使用成熟模型完成分类任务。使用resnet代替原先模型，加以训练。



1. 使用Resnet继续训练。

换成resnet之后，分类效果暴增。仅通过20轮的训练，准确率就达到了100%。

对比该数据集提供的两种示例分类结果，使用小波变换加上resnet在EMG的分类任务中取得了最好成绩。

|  |  |  |
| --- | --- | --- |
| Method | Training Accuracy | Test Accuracy |
| THE SLIDING WINDOW CNN1 | 0.83 | 0.78 |
| Root Mean Square (RMS) CNN3 | 0.96 | 0.91 |
| **WaveLet+ResNet（Ours）** | **1** | **0.94** |

1. 结果对比。示例方法：[EMG Classification with CNN (kaggle.com)](https://www.kaggle.com/code/endaleyohannes/emg-classification-with-cnn)
2. 结论

利用小波变换结合ResNet架构对EMG信号进行分类是非常有效的。小波变换提供了一个时频域的丰富表示，能够捕捉到EMG信号中的局部特征和时变特性，而ResNet架构通过其残差学习机制，有效地解决了深度网络训练中的梯度消失问题，使得网络即使在很深的层数也能有效学习。

实验结果显示，经过20轮的训练后，使用ResNet网络的方法在测试集上的准确率达到了94%，明显优于传统CNN模型和其他基准方法。这一结果证明了深度残差网络结合时间频率分析技术在复杂信号处理任务中的强大潜力。

此外，实验中使用的数据预处理、归一化和时频变换策略为神经网络模型提供了良好的输入条件，进一步提升了分类性能。虽然在实验配置中采用的是相对简单的网络结构和训练策略，但其效果已经非常显著。未来的工作可以探索更多的网络架构优化、更细粒度的参数调整以及更复杂的数据增强技术，以期进一步提升模型的泛化能力和实际应用中的表现。

通过本次实验，加深了对深度学习技术在生物医学信号处理中应用的理解，展示了小波变换与深度残差网络结合的强大能力，为未来在相关领域的研究提供了新的思路和方法。