深圳大学考试答题纸

(以论文、报告等形式考核专用)  
二○ 二二 ～二○ 二三 学年度第 二 学期

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 课程编号 | 1502990001 | | 课序号 | | 01 | 课程名称 | | 机器学习导论 | | 主讲教师 | 徐萌 | | 评分 |  |
| 学 号 | 2020281053 | | | 姓名 | | 刘玉婷 | 专业年级 | | 计算机科学与技术20级 | | | | | |
|  | | | | | | | | | | | | | | |
| 教师评语： | | | | | | | | | | | | | | |
| 题目： | | 基于深度学习的步态识别 | | | | | | | | | |  | | |

**摘 要**

本次的选题是基于视觉的步态用户识别（以下简称步态识别）。使用CASIA-B数据集，基于设定的视角（0°、18°、36°、54°、72°、90°、108°、126°、144°、162°、180°），通过整合分析不同场景下拍摄的不同视频的帧（不同视角、不同衣服、不同携带条件），从步态集合中学习身份信息。该项目可以建模为一个监督学习分类问题。

本篇报告首先介绍选题的背景，展开介绍步态识别研究的意义、现状和挑战。然后，介绍本项目的使用模型GaitSet，该模型主要使用了Set Pooling（集合池化）、CNN和Horizontal Pyramid Pooling（水平金字塔池化）。之后，我对模型中结构的设置提出疑问，并通过对比实验查看，查阅资料，查看获得该结果的原因，并使用最近火热的Swin Transformer模型取代原模型中的CNN进行模型的特征提取。最后介绍本次实验的设置和实验结果，实验结果显示在正常步行条件下，本模型在CASIA-B数据集上的rank-1命中率为97.459%；并且在背包行走和穿外套等复杂场景下，本模型在同一数据集上的准确率也分别达到了98.235%和92.061%，远远超过论文中给出的准确率； 本模型在测试样本帧数目较少的情况（如7帧），也能达到82.5%的准确率。

基于当前论文目前工作都是利用监督学习完成的，但是在现实社会中，标签数据并不是总能容易获得的，并且给数据做标签非常耗时，所以未来为了利用未标记的步态数据学习更有效更普遍的步态表示，可以从自监督学习角度进行研究。

目录

[一、研究介绍 3](#_Toc137665241)

[1.1 背景介绍 3](#_Toc137665242)

[1.2 研究现状 3](#_Toc137665243)

[1.3 Gaitset创新点 3](#_Toc137665244)

[二、模型原理与实现 3](#_Toc137665245)

[2.1 形式化定义 3](#_Toc137665246)

[2.2 模型介绍 4](#_Toc137665247)

[2.3 模型实现 7](#_Toc137665248)

[2.4 设置对比实验 10](#_Toc137665249)

[三 实验分析 16](#_Toc137665250)

[3.1 数据集介绍 16](#_Toc137665251)

[3.2 数据预处理 16](#_Toc137665252)

[3.3 实验设置 17](#_Toc137665253)

[总结 19](#_Toc137665254)

[个人感想 19](#_Toc137665255)

[参考文献 20](#_Toc137665256)

[附录 21](#_Toc137665257)

[代码操作指南 21](#_Toc137665258)

一、研究介绍

1.1 背景介绍

随着人工智能技术的快速发展，越来越多的领域研究都因其而得到了突破，在生物识别特征方面也不例外。生物特征识别技术，是指通过计算机利用人体所固有的生理特征或行为特征来进行个人身份鉴定的技术。一般常用的生物特征有人脸、指纹和虹膜等，目前应用非常广泛，但是这些特征都需要交互式地采集生物特征信息，有隐私泄露的风险。本文介绍的是基于行为特征——步态，而实现的识别技术。

研究显示，人人都有截然不同的走路姿势，因为人们在肌肉的力量、肌腱和骨骼长度、骨骼密度、视觉的灵敏程度、协调能力、体重、重心、肌肉或骨骼受损的程度、生理条件以及个人走路的“风格”上都存在细微差异。以上研究，为步态识别提供了可行性。

技术上，步态具有远距离识别、对采集装置要求低、难以伪装、非侵入且无需受试者合作的特性，可以广泛应用于犯罪防范、法医鉴定和社会保障。

步态识别技术是采用摄像头对识别目标的走路过程进行数据获取、检测、分割，在针对特征进行提取数据后，将该步态数据输入要对比的数据库进行比对进而识别的检测目标的身份。

1.2 研究现状

**1.2.1 身体表示：**

基于视觉的步态识别，需要对原始摄像头捕捉的帧图像中身体信息进相应表示，目前步态识别的身体表示有两种方法：轮廓和骨骼。本文使用的是轮廓表示法（如图1所示），这也是目前研究中最常用的身体表示方法。这种表示方法通过从背景种减去包含受试者的每个图像，然后进行二值化处理而得出。这种轮廓表示法有效且低成本地表示了单个帧中的身体状态，并且可以通过轮廓序列来表示如速度、步频、腿长、步态周期、步幅等特征。但是这种表示方法的缺点在于对个体外观变化（不同服装）非常敏感。



图1-1 轮廓表示法

骨架表示法是使用深度感应相机捕获身体的骨架表示或者使用姿势估计方法进行估计。步幅、速度、距离和关节之间的角度等静态和动态特征，可以从以骨架形式连接的身体关节中获得。通常基于骨架表示的方法比基于轮廓表示的方法对视角变化和外观变化具有更强的鲁棒性。但是由于严重依赖身体关节的准确检测，通常易受到遮挡变化的影响，并且应为姿势估计器的使用，而有更高的计算开销。

**1.2.2 研究挑战**

当前步态识别领域，存在的一些痛点和挑战，还没有商业化的基于步态的身份鉴别系统：

1. 跨视角问题：在不同的摄像机视角下进行步态识别。由于视角变化带来的形态和姿态的差异，跨视角步态识别变得更加困难。解决跨视角问题对于实际应用步态识别的准确性和鲁棒性至关重要。
2. 姿态和行为变化：步态识别面临的另一个挑战是人体姿态和行为的变化。不同的人可能具有不同的步态特征和运动习惯，甚至同一个人在不同的时间或环境下的步态也可能发生变化。这使得步态识别变得更加复杂，需要能够处理不同人的多样性和步态的动态性。
3. 时间序列数据：步态数据通常是时间序列数据，包含了一系列连续帧的步态图像。时间序列数据的处理和建模相对复杂，需要有效地提取和表示序列中的相关特征。此外，时间序列中可能存在噪声、遮挡或不完整的情况，需要对这些问题进行处理。
4. 数据集限制：步态识别的数据集往往是有限的，可能包含了有限数量的样本和有限视角的数据。这使得模型的泛化能力和鲁棒性面临挑战，特别是在处理新的未见过的数据时。

**1.2.3 目前常用实现方法：**

在使用Gaitset网络之前，我对步态识别领域进行了调研，目前未知步态识别的实现主要有两种方法，一种是将步态作为图像，另一种是将步态视为视频序列。

**基于模板图像的步态识别：**

基于模板图像的步态识别，是将所有步态轮廓压缩到一个模板图像中，一旦模板图被创建，识别方案就不再受帧数的影响。在深度步态识别体系中，步态轮廓可聚合在网络的初始层，称为时间模板；也可以在进行部分卷积和池化层后，聚集在中间层，称为卷积模板。

创建时间模板，首先通过背景减去获得的每一帧的人体轮廓；其次在每帧的轮廓上通过像素级运算符来生成步态模板；然后通过相关性分析(CCA)、线性判别分析(LDA)和深度学习等机器学习方法提取步态的表示；再然后，利用欧几里德距离等来衡量相似性；最后通过一个分类器，对模板进行身份的判别。一般而言，这个识别方法可以分为模板生成和模板匹配两个阶段。生成的目标是将步态信息压缩到单个图像中，常见时间模板有步态能量图（GEI）、计时步态图（CGI）、帧差能量图（FDEI）、步态熵图（GENI）和能量周期图（PEI）。GEI是在一个序列中平均步态轮廓，如图1-2所示。CGI则是将每个提取出的轮廓图像使用多通道映射到一个图像中进行编码，如图1-3所示。

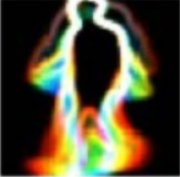


图1-2 GEI 图1-3 CGI

FEDI使用了聚类和去噪算法保留动力学信息，特别是当轮廓不完整时。GENI计算步态帧中每个像素的熵，然后在单个模板进行平均。PEI则是GEI的一种概括，通过利用基于帧幅度的多通道映射函数来保留更多空间和信息，如图1-4所示。时间模板这类方法具有简单易实现的优点，但是容易丢失时间和细粒度的信息。



图1-4 PEI

卷积模板常用的有集合池化（Set Pooling）和步态卷积能量图（GCEM），它们在整个序列上平均由几个卷积层和池化层获得的卷积图。本文使用的则是集合池化的卷积模板。

**基于视频序列的步态识别：**

基于视频序列的步态识别，直接从原始的步态轮廓中提取特征。这种表示方法因为聚焦到每一个轮廓，所以相比于步态图像而言包含了更多的空间信息；并且由于保存了帧的顺序和关系，所以包含更多的时间信息。学习时间信息有两种方式，第一种使用循环学习策略，其中每个帧都根据其与前一帧的关系进行处理。第二种方法，首先根据序列中可用的时空信息创建三维张量，张量的深度就表示时间信息，然后通过3D-CNN或GCN学习这些信息。这个方法看似信息包含全面，但是用于提取序列信息的深度神经网络比使用单个模板的神经网络更难训练，计算成本很高，并且视频序列需要保持不必要的顺序约束，导致灵活性低。

**1.2.4 神经网络**

深度神经网络利用多种非线性变换的分层架构来捕捉高级抽象的特征，各种不同的神经结构被设计并用于步态识别问题。

**CNN：**

卷积神经网络（CNN） 在步态识别中的应用最为广泛。CNN通常由不同类型的层组成，包括卷积层、池化层和全连接层。卷积层将学习到的过滤器与输入图像进行卷积，以创建激活特征图，捕捉具有不同程度的细节特征。卷积层还包括诸如ReLU或tanh函数的激活函数，以增加输出中的非线性。然后，池化层通过使用非线性下采样策略（如平均池化或最大池化）减小特征图的空间大小，从而降低网络的复杂性。全连接的图最终用于将生成的二维特征图学习为一维特征图，以便进一步处理。

**3D-CNN：**

3D 卷积神经网络用于学习整个步态序列的时空动态 。3D CNN 能够提取对相机视角和受试者外观变化更稳健的特征。3D CNN 以 3D 张量形式的堆叠步态帧作为输入，然后使用多个 3D 卷积滤波器和池化操作来提取空间角度表示。3D CNN 在步态识别方面的空间局限性在于在处理可变长度序列方面缺乏灵活性。

1.3 Gaitset创新点

以往基于时间模板的方法在提取特征时容易丢失时间和细粒度信息，而基于视频序列的模型虽然包含了这些信息，但是其网络往往难以训练，并且需要不必要的顺序关系。

针对这两类方法出现的问题，Gaitset模型的作者首次提出将步态看作由每个独立帧组成的步态轮廓集合，如图1-5所示。集合里的每一个步态图像即使打乱，也能仅通过外观重新正确排序，因为作为一个周期运动，在一个周期内每个图像的所处位置都是唯一的。这一假设首先解决了时间信息丢失的问题。其次，也放宽了视频序列需要保持严格顺序的约束。



图1-5 一个完整的步态集合

同时， GaitSet网络用于提取用户步态细粒度信息和全局信息，最终达到高效正确识别用户的效果。

二、模型原理与实现

2.1 形式化定义

首先，我将介绍步态集合的概念。假设给定一个数据集，包含有N个人，身份分别是。假设某一个人的步态服从仅仅与其身份相关的分布Pi。因此，某一个人的一个或者多个序列的所有步态轮廓可以看作是n个轮廓达到集合服从分布。

然后，基于这一假设，可以通过三个步骤来实现步态识别任务：表示提取的特征。函数F表示卷积网络，用于从每个步态轮廓中提取帧级的特征。函数G是一个置换不变函数，由集合池化(SP)实现，用于将一组帧级特征映射为集合级特征。函数H则是由水平金字塔映射（HMP）实现，用于从集合级特征中学习分布Pi的判别表示。输入是一个四维张量，即集合维度、通道图像维度、图像高度维度和图像宽度维度。

2.2 模型介绍

接下来，我将介绍模型的大致框架，并介绍组成模型的几大重要部分。GaitSet模型分为特征提取和特征表示两个部分。在特征提取部分有主管道和多层全局管道(MGP)两个部分。主管道基于输入的多帧步态轮廓某一帧特征进行处理得到帧级别特征；多层全局管道则针对主分支提取出的帧级别特征进行处理得到集合级特征。特征提取完成后，要进行特征的表示，特征表示的不同，学习信息的范围就不同，可以是全局，也可以是局部。全局表示的方法往往受遮挡和外观变化以及缺少关键身体部位影响更大，本模型在进行特征表示时应用的是水平金字塔映射（HPM），这是一种局部表示方法。

GaitSet模型具体如下图2-1所示。模型输入是一个经过二值化处理的步态轮廓集合，集合中的每一帧图像会在主管道进过卷积和池化以提取帧级别特征，主管道中图像每经过一次CNN网络都会进行集合池化，并且与辅助分支进行特征融合用以提取集合级特征，最后输入到水平金字塔映射模块（HPM），主管道提取特征和多层全局管道提取的特征首先会分别进行水平金字塔池化，然后分别进入一个全连接网络，最后生成识别所需的特征表示。

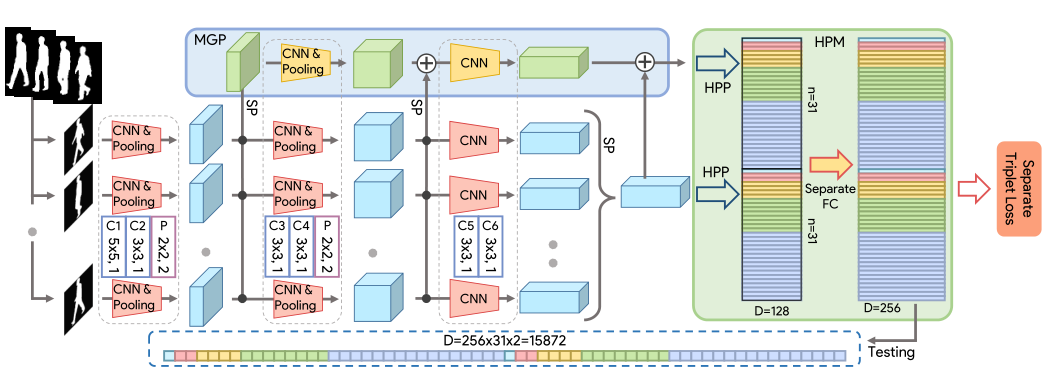


图2-1 GaitSet模型

Set Pooling (SP)：

SP目标时整合从步态集合中每一帧图像提取出的特征。SP可以形式化为：，表示集合级特征，表示帧级别特征。函数G有两个约束条件。第一，因为GaitSet网络输入是一个无序帧图片集合，G必须是一个置换不变函数，即函数值与输入元素的顺序无关。现实生活中，这类函数非常多，如加法、乘法、Min（）、Max（）。第二，由于一个人的步态轮廓数目可以是任意的，G应该能取任意基数的集合，即帧为任意个数。在论文中，作者设计了六个SP的实现方法，其中统计函数有三种，联合函数有两种，和注意力机制。具体如下：

1. 表示在通道维度进行串联；表示1x1的卷积层，在集合维度上取最大值、平均值和中位数。
2. Attention

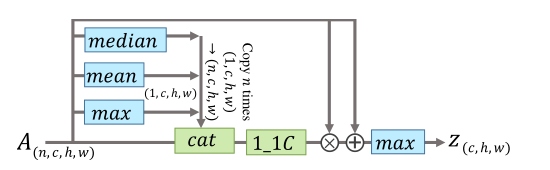


图2-2 Attention

主要思想是利用全局信息为每一个帧级特征图学习出一个注意力图，从而对齐提炼，注意力机制全局信息首先由左边三个统计函数收集；之后与原始特征图融合，输入1x1的卷积层，求得注意力以获得更细节的帧信息；再通过MAX映射得到集合级别特征Z。

**Horizontal Pyramid Mapping (HPM)：**

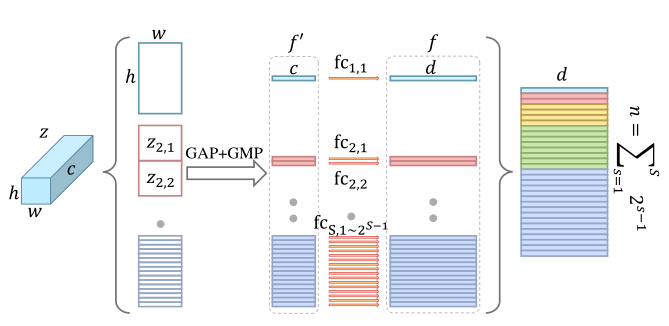


图2-3 Horizontal Pyramid Mapping

水平金字塔映射改进了HPP算法，HPP算法是在Horizontal Pyramid Matching for Person Re-Identification这篇论文中提出，是贝克曼研究所发表在AAAI2019上的工作。HPP算法针对卷积神经网络输出的特征图进行处理，先以不同金字塔比例对特征图进行水平切片，然后使用每个切片的水平条带进行全局平均池化和全局最大池化，在之后通过1x1卷积网络，形成每张图像的最终表示,具体过程如2-4所示。

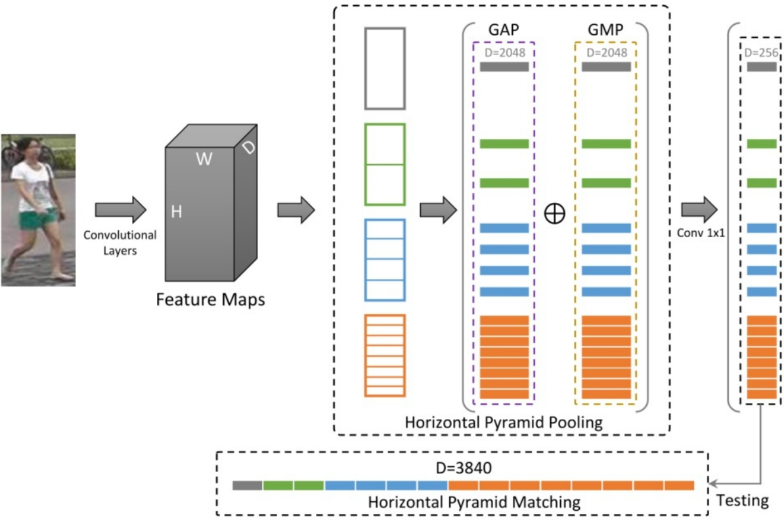


图2-4 原始HPP算法

而本论文GaitSet网络的输入是一个多帧图像的集合，最后由卷积网络提取出的特征在输入到HPM模块前，进行了集合池化（SP），融合为集合级形式而不是单个图片，所以作者将最后1x1的卷积层改变为全连接层(FC)，以适应步态识别任务，如图2-3所示。

总的来说，主要步骤如下：

* HPM通过将步态序列图像输入到卷积神经网络（CNN）中，并在网络中的多个层次上提取特征。
* 然后，对于每个层次的特征图，HPM将特征图进行分割成多个子图块，每个子图块表示不同的水平尺度。
* 子图块特征提取：在每个子图块中，HPM对特征进行池化操作，例如最大池化或平均池化，以获取子图块内的局部特征。
* 全局特征提取：然后，通过对所有子图块进行池化操作，HPM还能够获取整个特征图的全局特征。

最后，将局部特征和全局特征进行拼接或者融合，得到最终的多尺度特征表示。

HPM有S个尺度（本文），在尺度上，SP提取的特征图在每个尺度高度维度被分割为个条带,即总共有条特征。全局池化应用于3D条带，得到一维特征。对于条带，其中t表示条带对应尺度下的索引。全局池化可以表示为，最后一步是使用全连接层将特征映射到可区分空间。由于不同尺度的条带描述了不同视角范围的特征，每个尺度的不同条带描述了不同空间位置的特征，因此自然会使用独立的全连接层，二维条带如图2-5所示。分割为条带来表示部分特征后，不仅可以学习到局部的特征信息，还可以学习到这些部分特征之间的关系，从而保留了位置属性，如比例、旋转和位置，这提高了步态识别方法对方向和视角变化的鲁棒性。

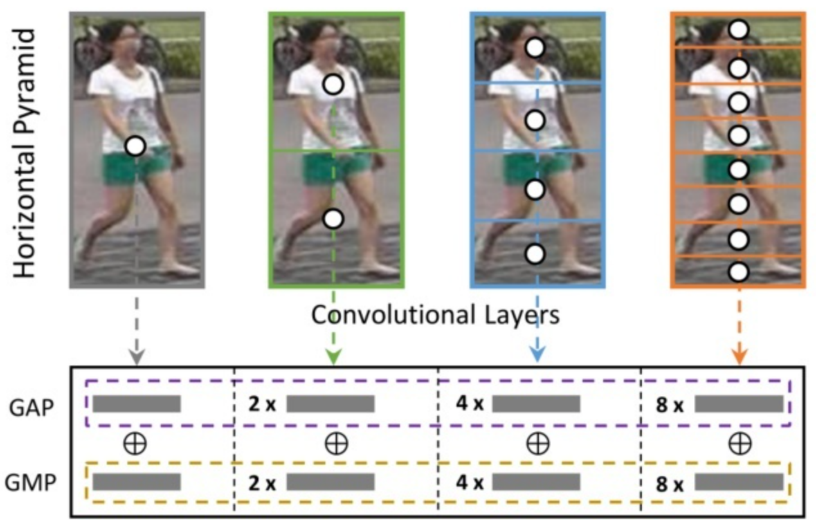


图2-5 条带分割

Multilayer Global Pipeline(MGP)：

由于不同层的卷积网络具有不同的感受野，越深层的卷积网络具有越大的感受野。因此，浅层特征包含更多细粒度信息（如轮廓、边缘、颜色、纹理和形状），深层特征蕴含更多全局粗粒度信息（如人像图的整个人脸）。

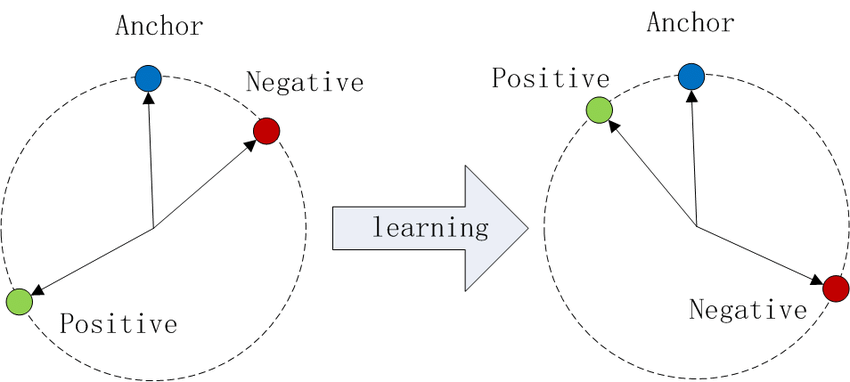
为了收集不同级别的集合信息，提出多层全局管道(MGP)，MGP的结构与主管道的结构类似，卷积和池化的结构与主管道完全相同。主管道在经过CNN进行特征提取后，进行SP操作后，再送入MGP收集集合级特征。最终由MGP生成的特征也被HPM分成条特征，在MGP后面的HPM不会和主管道后面的HPM共享参数。

**Loss设计：**

最后，模型在训练阶段采用BA+ Triplet Loss，在每次提取特征时，同步输入与该样本相同类别和不同类别的两个样本，利用监督学习使该样本特征与相同类别的样本特征间的差异越来越小，与不同类别的样本特征间的差异越来越大。以下是两种loss的简单介绍：

BA（Bottleneck Alignment Loss）通过计算两个集合之间的特征向量对之间的欧氏距离来衡量对齐性。具体地，我们要先对每个样本，计算两个集合之间的平均特征向量和差异特征向量。其中，平均特征向量（mean vector）指的是对应两个集合中相同索引位置的特征向量的平均值；差异特征向量（difference vector）指的是对应两个集合中相同索引位置的特征向量的差异。计算平均特征向量和差异特征向量的欧氏距离。

Triplet Loss三元组损失的目标是通过最小化锚样本与正样本之间的距离，同时最大化锚样本与负样本之间的距离。这样可以在嵌入向量空间中形成紧密的同类簇并增加不同类别之间的分离度。



**2.3 模型实现**

接下来，我将结合代码，解释模型具体是如何实现的（我将代码的解释以注释的方式加在了代码上）：

卷积网络定义：

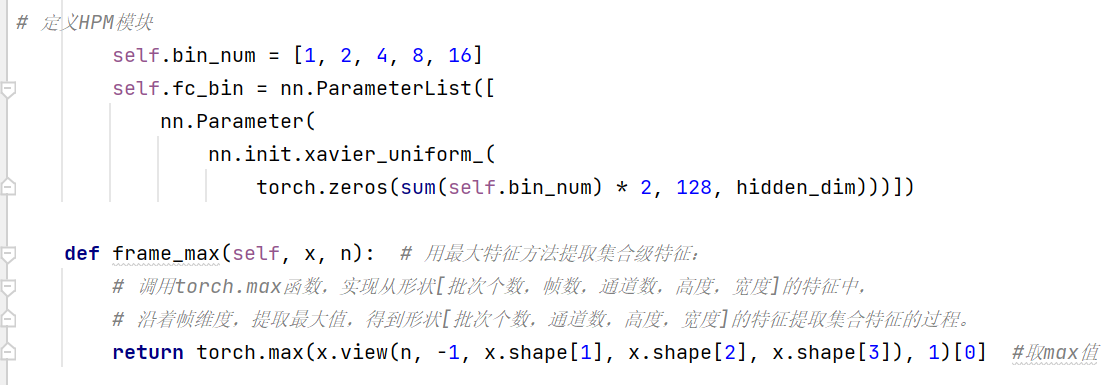


GaitSet网络：

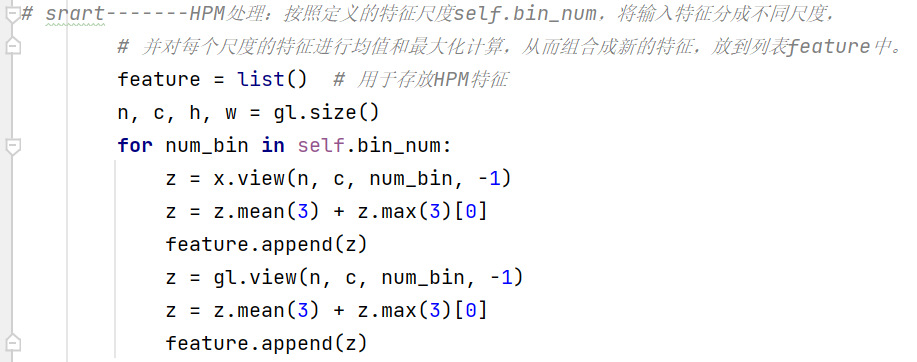
1）实现3个MGP

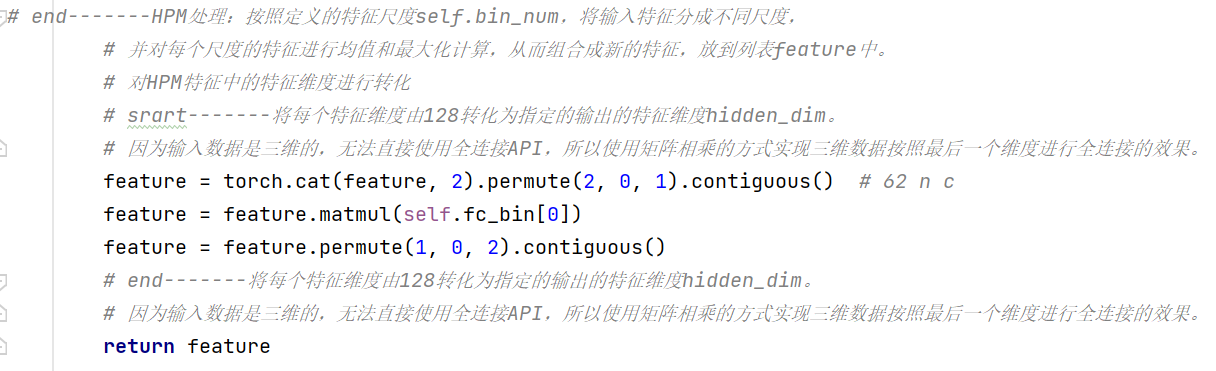


**2）定义HPM**

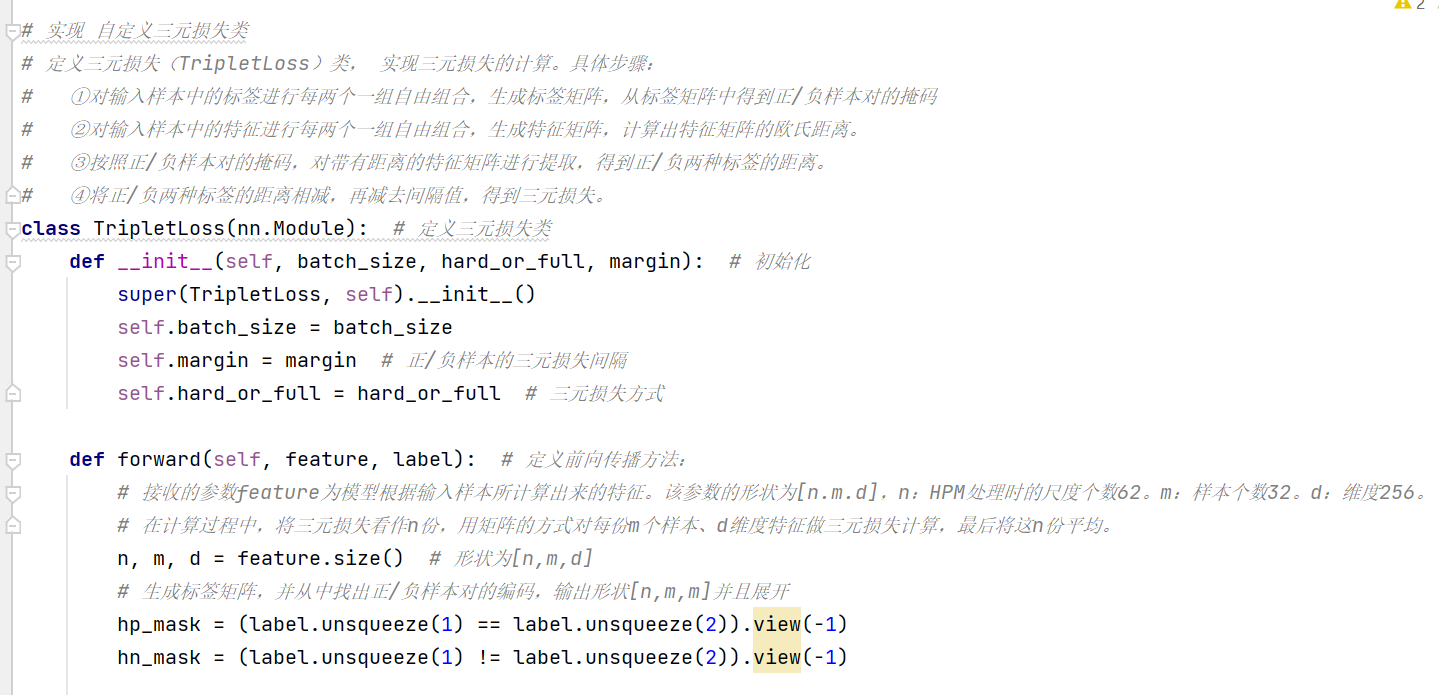




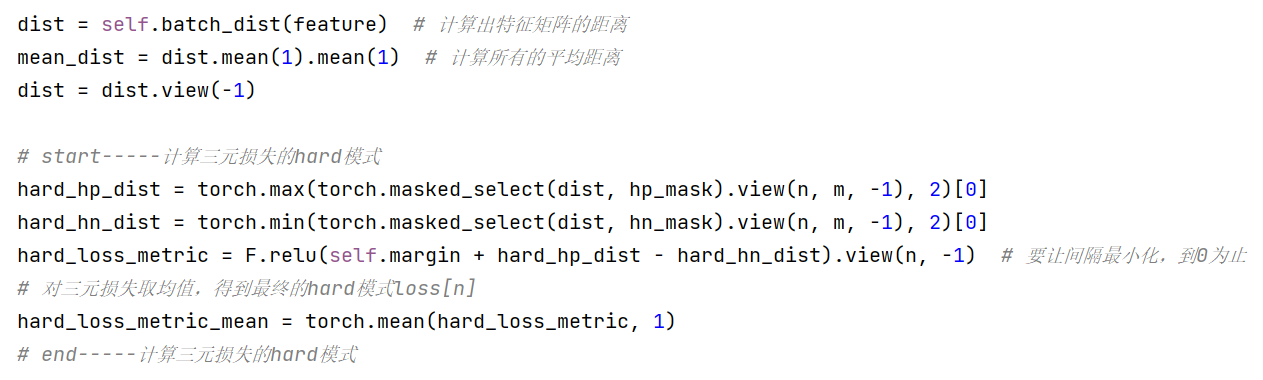




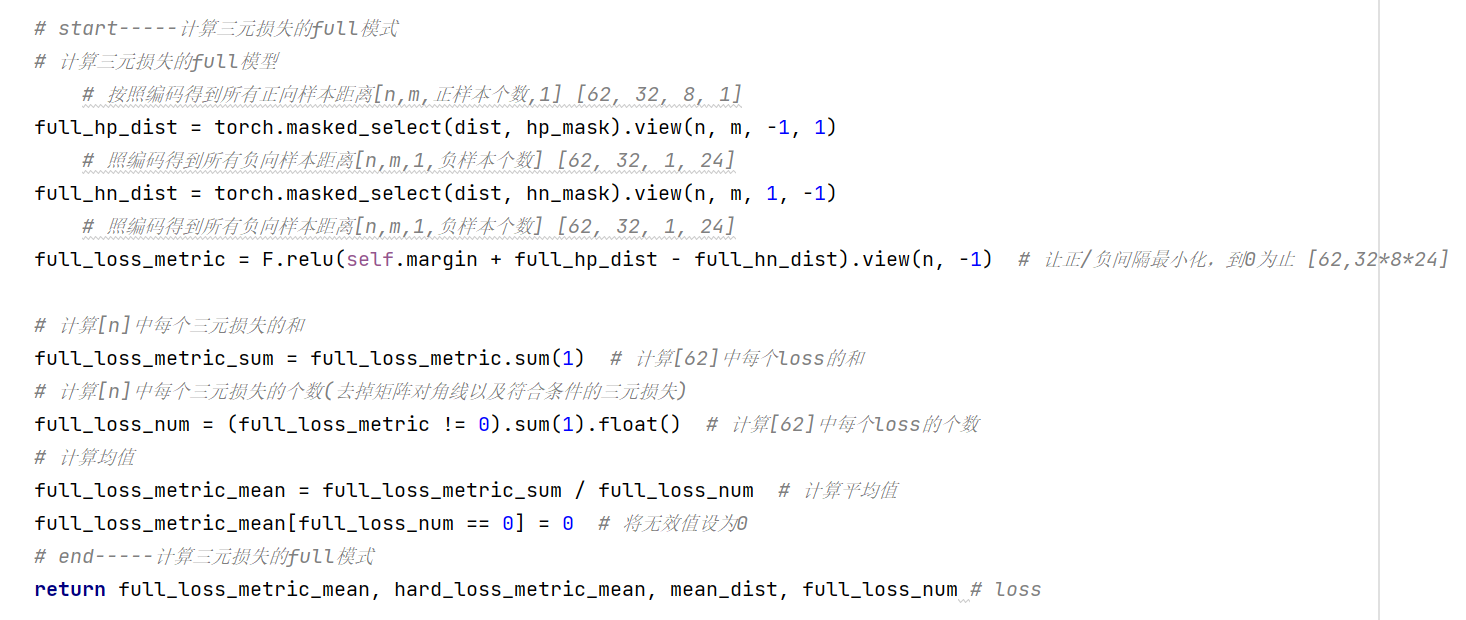
TripletLoss:



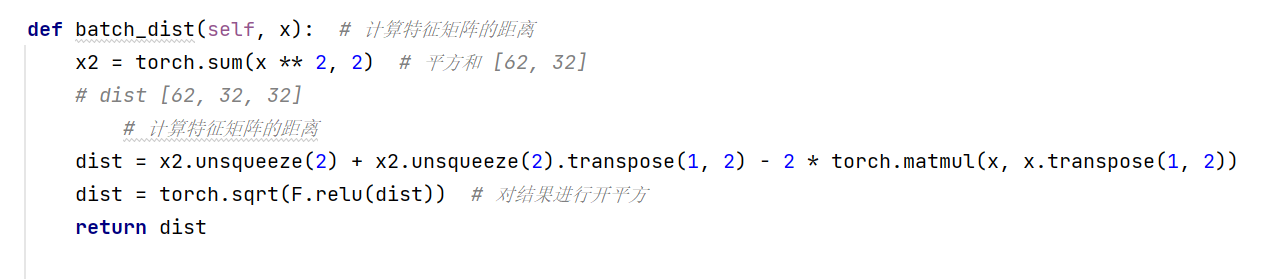
**1）计算三元组损失的hard模式**



**2）计算三元组损失的full模式**



**3）计算特征矩阵的距离**



2.4 设置对比实验

**2.4.1 加深加宽网络**

理论上，不同层的卷积网络具有不同的感受野，越深层的卷积网络具有越大的感受野、可以生成更多种类的特征表示、对于输入数据的小扰动更具鲁棒性，应该可以学习更加复杂、抽象的特征表示。那么是否意味着，排除性能的限制，网络越深，准确率越高呢？

在接下来的实验里，我尝试将SP由原本的三层增加到五层，并观察分析结果。

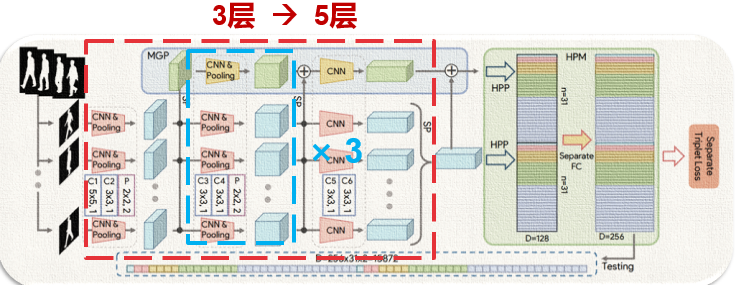
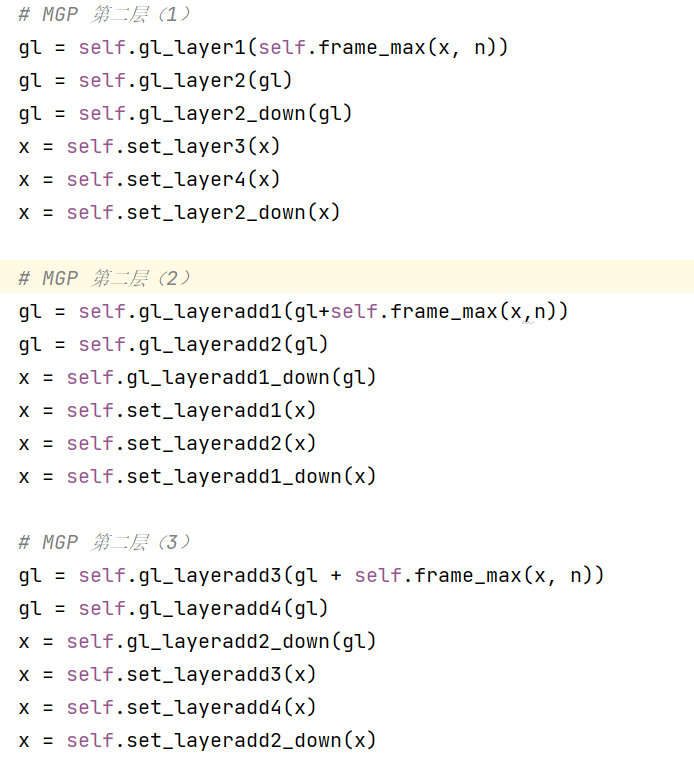


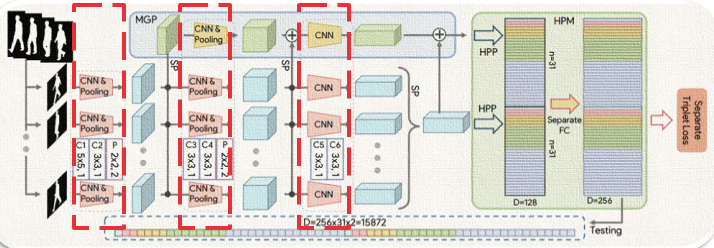
图2.6-1加深网络

2.6-2（1）更改之前 2.6-2（2）更改之后

**2.4.2 使用Swin Transformer 代替CNN进行特征提取**

Swin\_Transformer是NLP领域的transformer在CV领域应用，在CV各个研究方向都达到了很好的效果。因此，我尝试将这个结构运用在步态识别任务中，使用Swin Transformer 代替CNN进行特征提取。



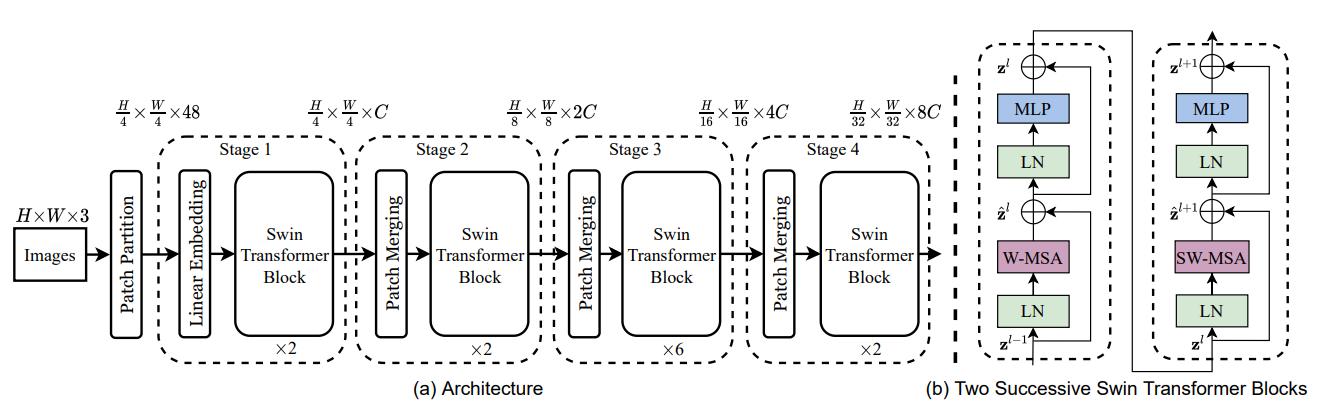


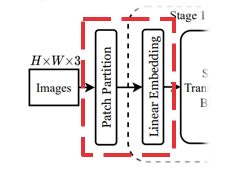
图2.6-3 ST取代CNN

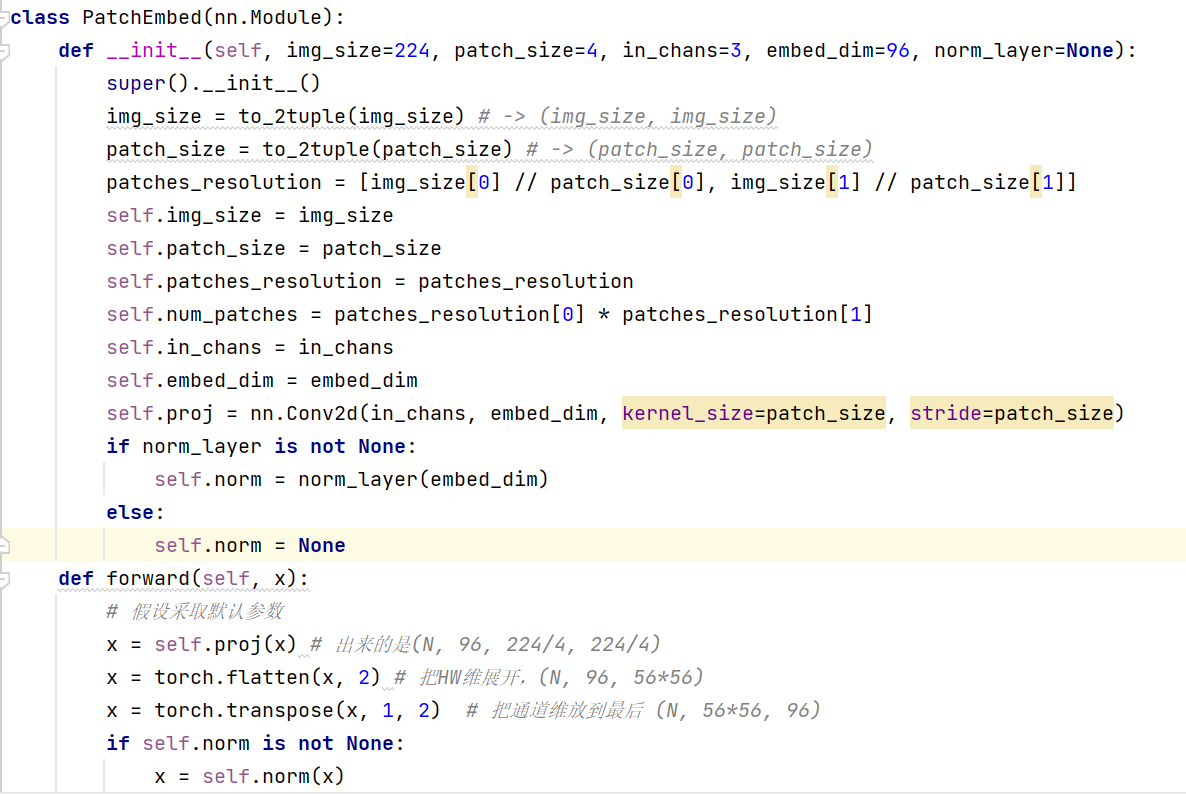
Swin Transformer模型一共包含4个Stage，每个stage都会缩小输入特征图的分辨率，像CNN一样逐层扩大感受野。

1. 在输入开始的时候， Patch Embedding将图片切成一个个图块，并嵌入到Embedding。
2. 在每个Stage里，由Patch Merging和多个Block组成：
   1. 其中Patch Merging模块主要在每个Stage一开始降低图片分辨率。
   2. 而Block具体结构如右图所示，主要由LayerNorm，MLP，Window Attention 和 Shifted Window Attention组成。

**具体实现**

**1）Patch Embedding部分**

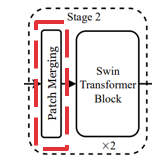


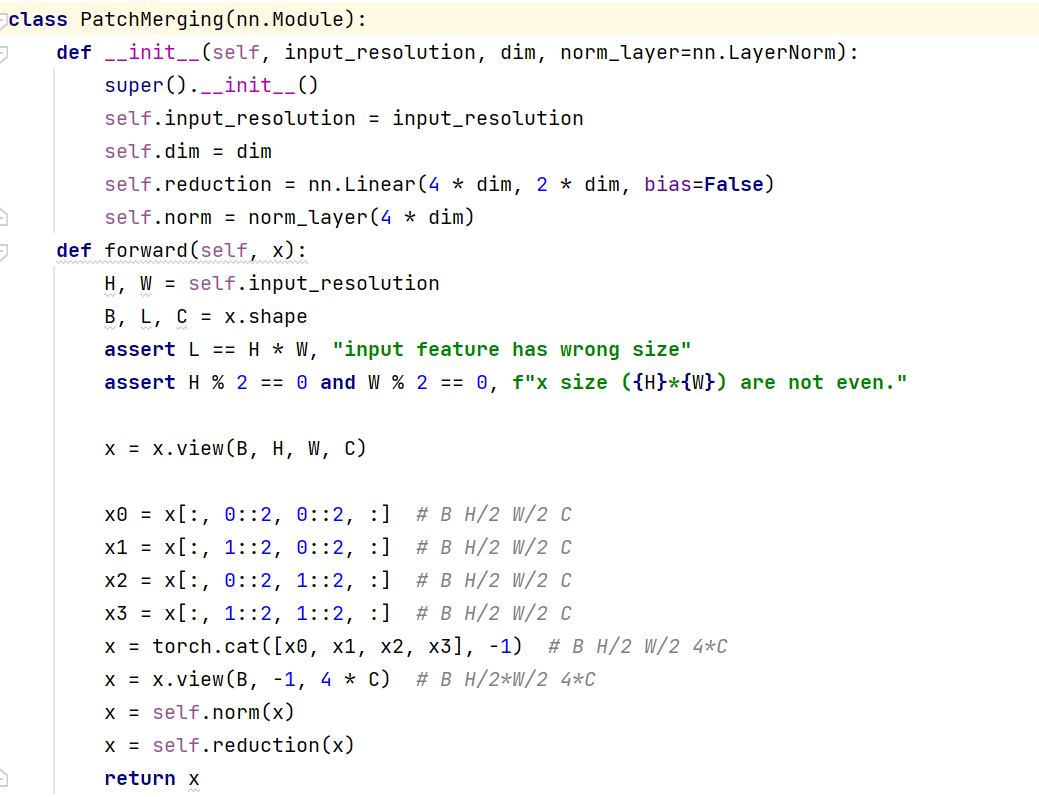


这里通过二维卷积层，将stride，kernelsize设置为patch\_size大小。输入 x 通过卷积层 self.proj 进行特征提取，得到形状为 (N, embed\_dim, H', W') 的输出，其中 N 是输入的批次大小，embed\_dim 是嵌入的维度，H' 和 W' 是经过卷积后的图像尺寸。

to\_2tuple 函数用于将输入的 img\_size 转换为一个长度为 2 的元组，确保 img\_size 是一个二维的尺寸；torch.flatten 函数将 H' 和 W' 维度展平，得到形状为 (N, embed\_dim, H'\*W') 的输出，即每个 patch 对应的特征向量。

**2）Patch Merging部分**





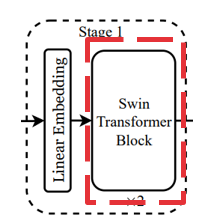
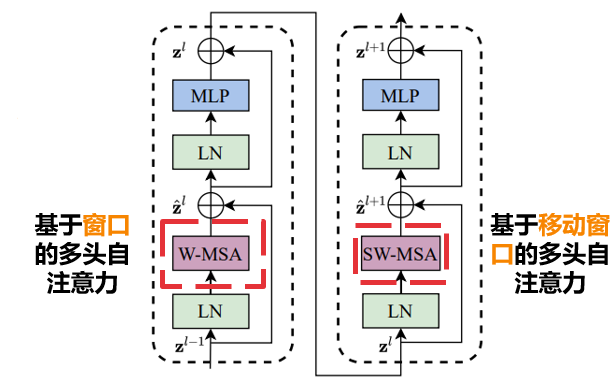
这里我们完成降采样操作。每次降采样在行方向和列方向上，间隔2选取元素，然后拼接在一起作为一整个张量，最后展开。此时通道维度会变成原先的4倍（因为H,W各缩小2倍），

此时再通过一个全连接层再调整通道维度为原来的两倍。

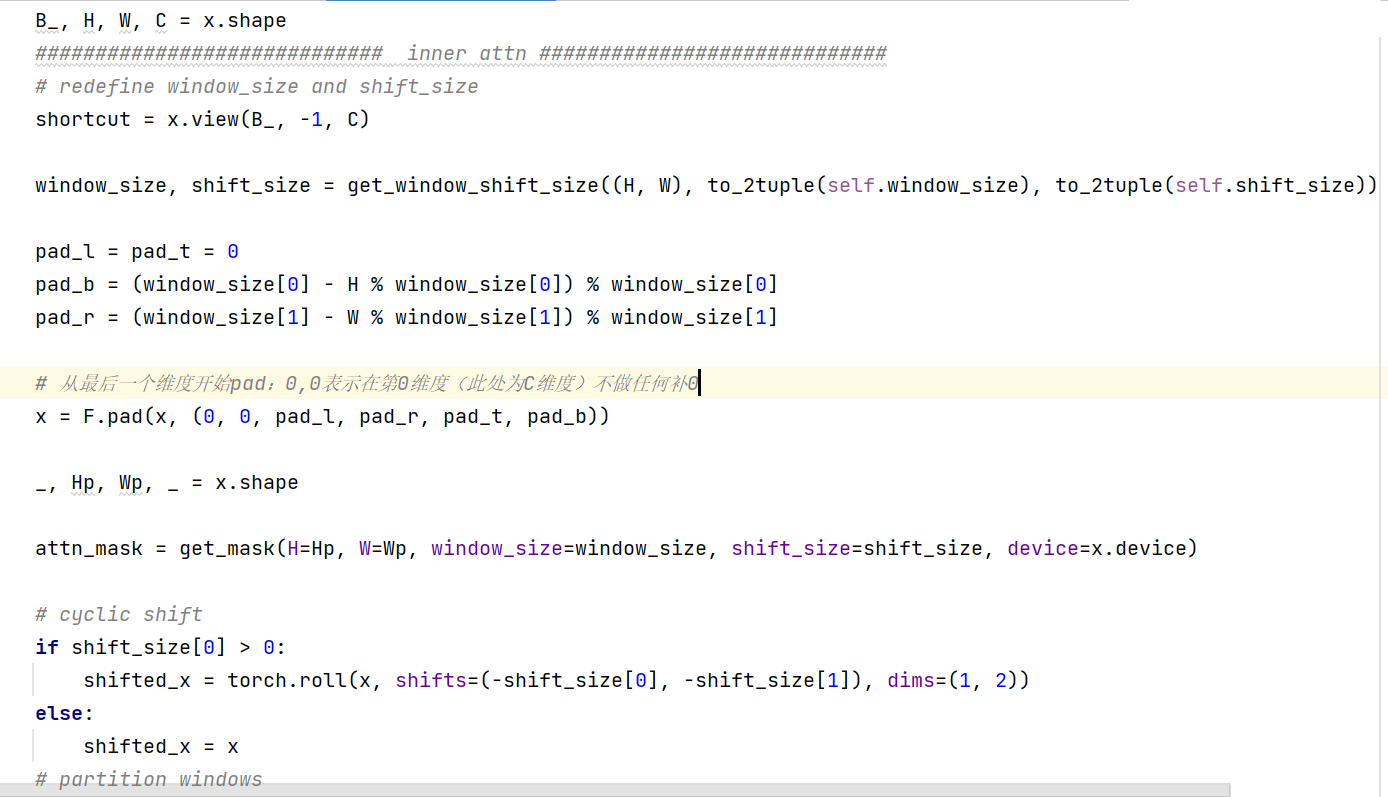


用空间上的维度换了更多通道数

**2）Swin Transformer Block部分**





* 先对特征图进行LayerNorm
* 通过self.shift\_size决定是否需要对特征图进行shift
* 然后将特征图切成一个个窗口
* 计算Attention，通过self.attn\_mask来区分Window Attention还是Shift Window Attention
* 将各个窗口合并回来
* 如果之前有做shift操作，此时进行reverse shift，把之前的shift操作恢复
* 做dropout和残差连接
* 再通过一层LayerNorm+全连接层，以及dropout和残差连接

三 实验分析

3.1 数据集介绍

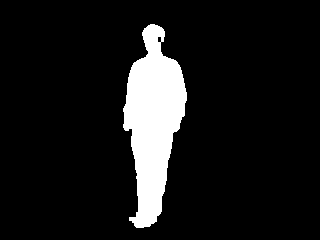
本论文使用的是CASIA-B数据集。CASIA-B是使用最广泛的步态数据集， 包含124人的RGB和轮廓形式的多视角步态数据。从11个不同的视角进行采集，范围从0到180度，增量为18度。该数据集考虑了三种不同的行走条件，即正常行走(NM)、穿外套行走(CL)和携包行走(BG)，每个人每个视角分别有6NM序列、2CL序列和2BG个步态序列，即每人一共有110个序列。数据集中NM、CL、BG分别如图3-1，3-2，3-3所示。 

图3-1 NM-90° 图3-2 NM-0°

. 

图3-3 BG-90° 图3-4 CL-180°

3.2 数据预处理

在所有实验中，输入的都是一系列64x64的对齐轮廓图，轮廓图由数据集直接提供并且基于Takemura的方法对齐，具体过程如图3-5、3-6所示。

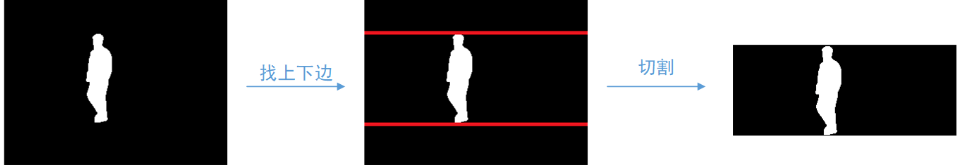


图3-5 轮廓图对齐过程

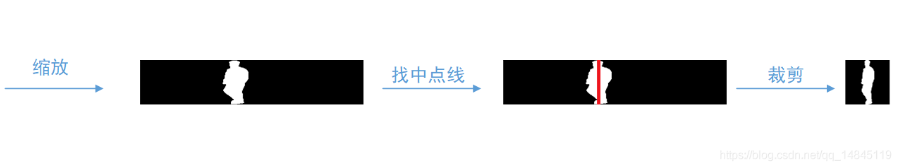


图3-6 轮廓图对齐过程

1. 对轮廓图，依据每一行的像素和不为0的原则，找上边和下边。
2. 根据上边和下边对轮廓图进行切割
3. 对切割的图进行resize操作，高度为64，宽度保持比例
4. 依据像素个数和最大的列为中心线原则，找到中心线
5. 中心线左右各32像素进行切割，不够的补0
6. 得到对齐后的轮廓图



图3-7 对齐后取10帧结果

3.3 实验设置

因为硬件限制和数据集原因，无法完成作者在论文中那样的实验设置，因此我做了部分改动。

CASIA-B数据集没有划分训练集和测试集，将CASIA-B数据集进行了小样本训练(ST)划分，去除数据集中不可用用户，总共有116名用户。小样本训练将前24名受试者的数据进行训练，其余92名受试者进行测试。

在训练过程中，将每迭代100次就会根据loss大小判断是否保存当前参数文件。在测试阶段，进行划分后将测试数据拆分为一个注册集(gallery set)，其中包括NM步态数据中的前四个步态序列。探测集（probe set）由其余序列组成，即每个受试者每个视角的剩余2个NM、2个CL和2个BG序列。在训练完成得到相应的模型后，将注册集数据输入到模型中得到相应特征，再将探测集数据也输入模型得到相应特征，然后将两个特征认证对比。

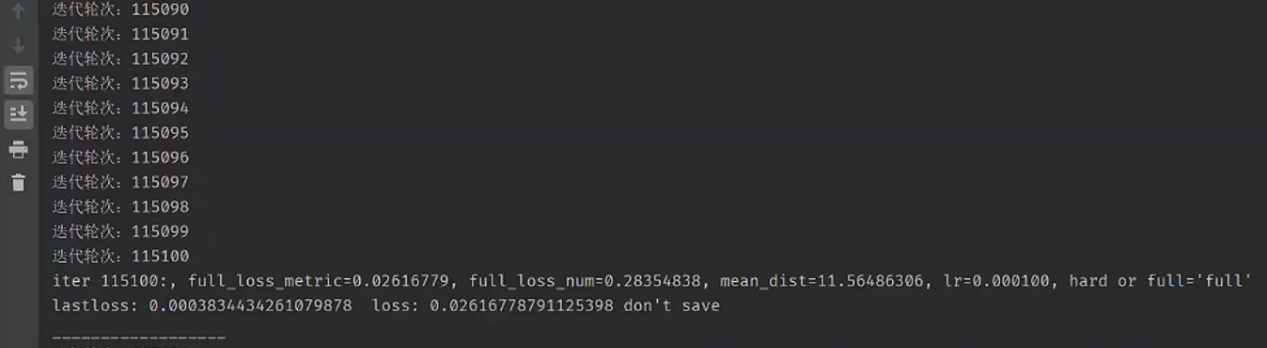


图3-8 训练过程（Batch-size=4x16，共迭代16万次）

**3.4 实验结果与分析**

（1）论文原模型

携带包裹（11个视角，124个人的BG数据），准确率**98.235%**；

普通（11个视角，NM中剩下的数据（NM 01-04剩下的结果），共50人），准确率**97.459%**；

穿大衣（11个视角，124个人的CL数据），准确率**92.061%**。远远好于论文中的结果。

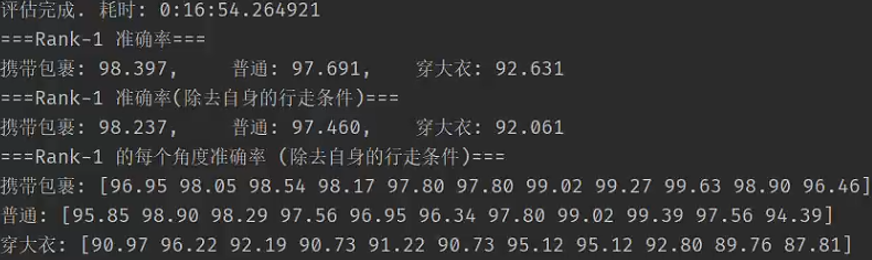


图3-9（1）原模型结果

（2）SP 3层->5层。

携带包裹准确率**76.430%**；普通准确率**65.996%**；穿大衣准确率**43.061%**。

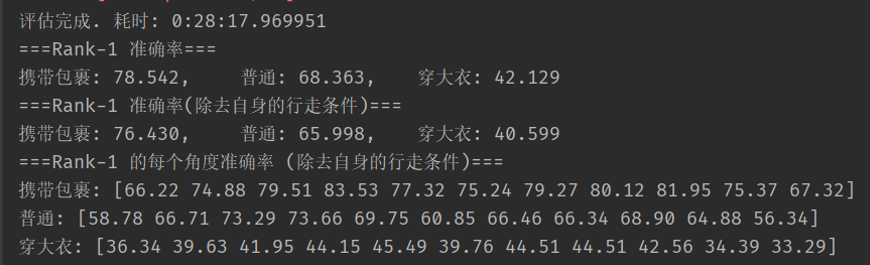
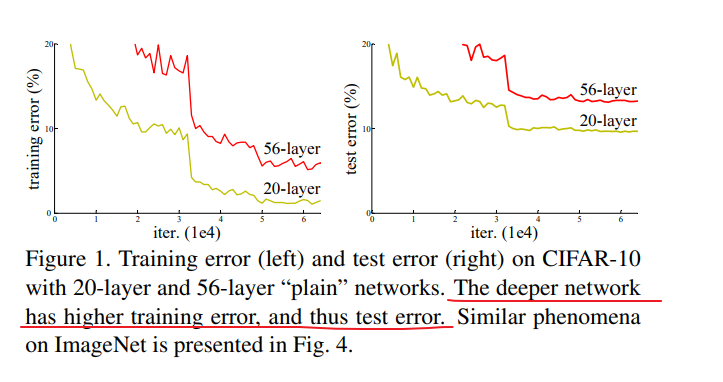


图3-9（2）加深模型结果

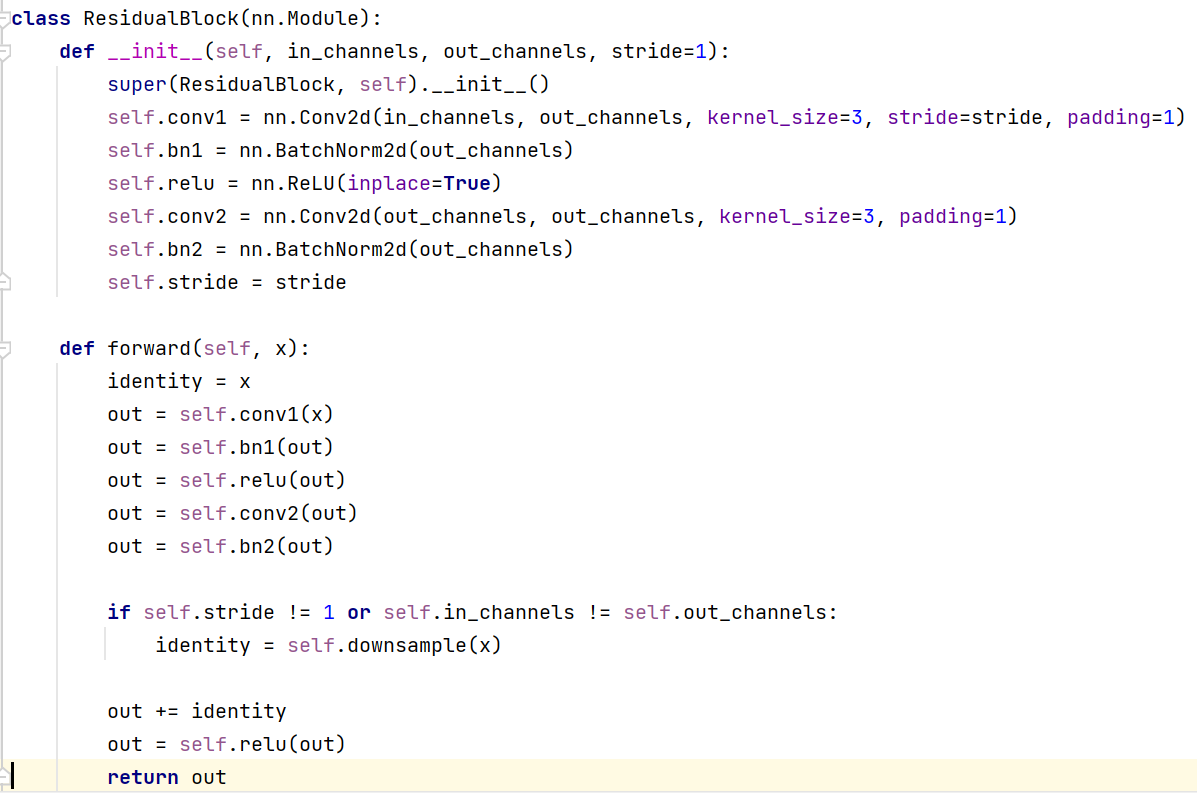
准确率没有升高，反而大幅度下降。得到这个结果我很意外，因为它与我的猜想正好相反，因此我去查阅了资料，在论文**Deep Residual Learning for Image Recognition**中，找到了答案。作者遇到了与我同样的问题，即随着网络层数的增加，训练精度会饱和然后逐渐下降，即网络深度增加导致训练精度下降的现象。



这个现象被称为 "Degradation Problem"，即网络深度增加会导致网络的训练精度下降。论文中指出，即使是在理论上，使用更深的网络也应该能够达到更低的训练误差，但实际训练中却观察到了相反的现象。

作者通过实验证明，这个现象不是由于网络过拟合或优化问题造成的，而是由于网络的优化难度增加所致。更深的网络使得优化变得更加困难，导致了训练精度下降。在这种情况下，网络无法充分地学习到有用的特征，而仅仅拟合了训练数据中的噪声。

为了解决这个问题，作者提出了残差学习（Residual Learning）的方法，通过引入跳跃连接（shortcut connections）来解决梯度消失和训练难度增加的问题。通过在网络中添加残差模块，残差块中的跳跃连接允许信息直接从一个块传递到后续块，避免了信号在传递过程中的逐层衰减，从而更容易地训练深层网络，使得网络可以更容易地学习恒等映射，从而降低了网络的优化难度，提高了训练精度和收敛速度。为此，我们找出原论文，并尝试一下论文中描述的残差块：



携带包裹准确率**77.597%**；普通准确率**66.972%**；穿大衣准确率**53.590%**。

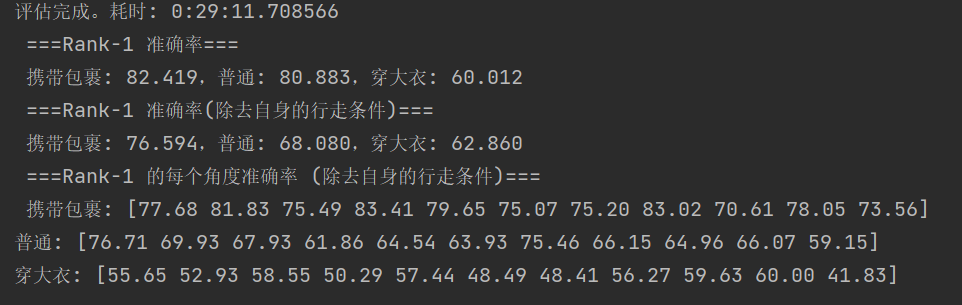


图3-9（3）加深模型改良后的结果

由结果可以看出，结果有所提升，尤其是穿大衣的数据集上准确率从43.061%提高到53.590%，但是其他两个数据集上虽然有一定的提升但不是特别明显。

（3）调节通道数使其变多。

携带包裹准确率**93.768%**；普通准确率**88.924%**；穿大衣准确率**59.855%**。

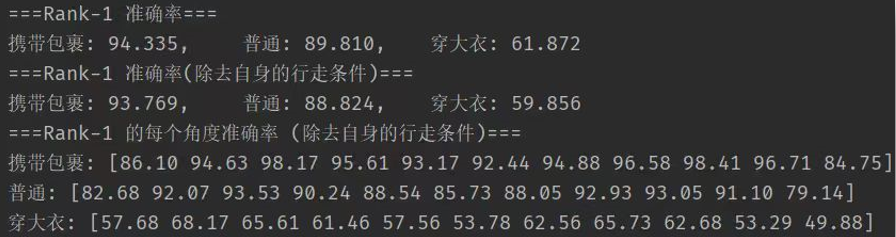


图3-9（3）加宽模型结果

通道数加宽对模型的影响不算很大，但是依然没有比过改之前的模型，其中穿大衣的数据集的测试结果下降幅度很大，但是携带包裹和普通数据集并无太大差异。由此可见，增多通道数之后，模型的鲁棒性没有之前那么好了，更多的通道意味着需要更多的参数来学习每个通道的权重。这可能导致模型更复杂，更难以训练和调优。

（4）使用Swin Transformer代替CNN。

携带包裹准确率**67.594%**；普通准确率**59.081%**；穿大衣准确率**37.860%**。

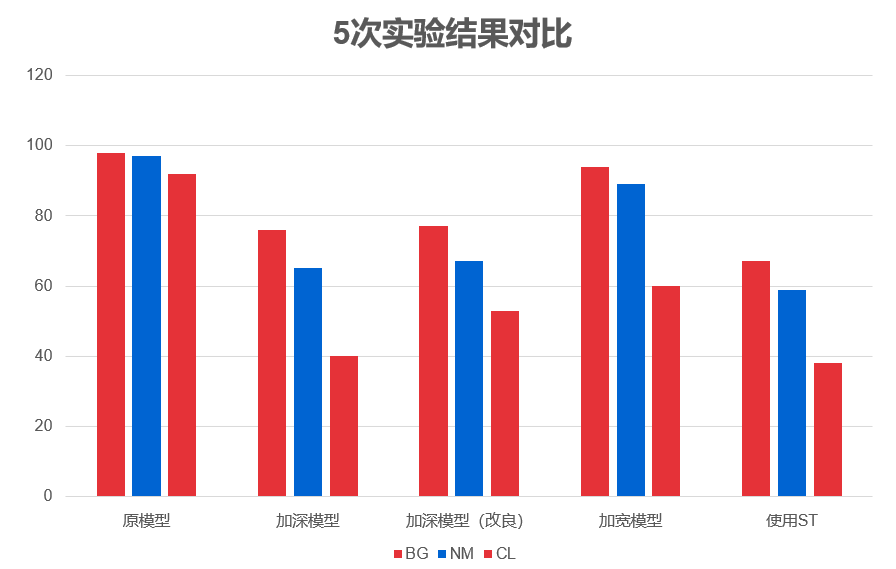


图3-9（3）加宽模型结果

可以看到，更改模型之后，准确率直接折半。在CV领域表现良好的Swin Transformer似乎并不能在这个任务上良好地表现，我分析有以下两点原因：

1. 二值图数据的样本差距较小。Swin Transformer 无法提取出对步态识别有帮助的关键特征。对于特定任务，可能需要更准确的特征表示。
2. 模型参数调整不当。Swin Transformer 有很多可调节的超参数，包括层数、头数、嵌入维度等。如果这些参数没有经过充分的调整和优化，模型可能无法适应步态分析任务的特定需求，导致准确率下降。

（5）实验结果对比



总的来说，原模型的结果准确率很高，结果已经足够好了，后面进行的四次尝试在三种数据集上均没有提升，下降最多的是使用了Swin Transformer的模型，因为改动最大，可能该任务与ST并不匹配。在所有模型中，人体穿大衣情况下的准确率都是最低的，并且在加深加宽原模型后，这种情况更加严重，基本达到其他情况下的一半。

总结

经过三个对比实验以及原文实验可以得出，与现有基于模板或视频序列的方法相比，论文提出的模型在二值图像作为输入的情况下准确率是最高的，且与Swin Transformer方法相比训练速度快很多，效果也好很多。有良好结果的原因主要有以下两个方面：第一，作者提出的将步态作为一个集合作为输入，包含了丰富的多帧轮廓图，使得训练数据比以往更充分，并且随着输入集合帧数的增加和人数的增加，模型会学习得更为充分，准确率越来越高。第二，作者提出的GaitSet模型能够有效的提取出空间和时间信息。

个人感想

本次实验是我第一次在复现论文代码，更是第一次尝试对现有模型进行修改和优化，对我来说难度很大。在尝试的过程中，遇到了很多困难，虽然论文提供了源代码，但是从下载到调试到成功运行中间的过程并不是很顺利，由于没有很好的设备，需要找到适合自己设备的batch-size并相应的提高迭代次数，因为操作不熟练中间重新训练了很多次。

但是本次实验也带给我了非常大的收获：首先，为了了解步态分析最近的发展，我在网上搜索了相关的新闻，了解到步态分析现在面临的技术难点，以及为何没能走进市场。最终我在一堆论文中，选择了网络GaitSet，因为它是是AAAI 2019顶会文章也发表在了IEEE TPAMI 2021顶刊中。在复现网络的过程中，我对它设计的网络结构产生了疑问，比如为什么SP要设计成三层？三这个数字是怎么得来的呢？为什么这里通道数要设置成这么大？为此，我们专门设计了对比实验，将自己的问题转换为代码，并在得出结果之后尝试进行分析。之后的实验证明，实验的结果与我的认知相差很大，我也因此收获了很多，并找到了解决方法。

最后，虽然实验的结果很好，准确率甚至达到了98%，但是离运用到实际生活中我认为还差了很大一截，我们运用的图片为二值图像，图片本身屏蔽了背景的干扰，而显示生活中我们可能需要遇到的环境会复杂很多。初次之外，原模型在人体携带包裹的情况下准确率高达98，可是在人体穿大衣的情况下准确率马上降到了92%。这说明，衣物对模型的影响是比较大的，准确率浮动较大。

参考文献

[1] Hanqing Chao, Yiwei He, JunPing Zhang. GaitSet: Regarding Gait as a Set for Cross-View Gait Recognition [c]. (AAAI-19): The Thirty-Third AAAI Conference on Artificial Intelligence, Honolulu Hawaii , 2019: 8126-8133.

[2] Hang Fu, Yunchao Wei, Yuqian Zhou. Horizontal pyramid matching for person re-identification [c]. (AAAI-19): The Thirty-Third AAAI Conference on Artificial Intelligence, Honolulu Hawaii , 2019: 8295-8302.

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in Neural Information Processing Systems, pages 5998–6008, 2017. 1, 2, 4

[4] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEE*E Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[5] Z. Liu et al., "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 2021, pp. 9992-10002, doi: 10.1109/ICCV48922.2021.00986.

附录

代码操作指南

**1、数据预处理**：将数据集文件切割对齐为64x64的轮廓图

**pycharm：**

首先在第29行、第31行，分别设置1)原始数据集路径、2)处理预处理后保存路径

然后运行pretreatment.py

默认设置 1）D:\GaitDatasetB-silh 2）D:\GaitDatasetB\_Cutted

**命令行：**

python pretreatment.py --input\_path='原始数据集路径' --output\_path='预处理后保存路径'

因为处理过程极其慢，所以我已经处理好并打包在GaitDatasetB\_Cutted文件下，解压后，可以直接作为训练和测试数据集输入。

**2、训练**：训练会每迭代100次保存一次模型，保存条件是比当前模型loss小。

首先在23行 ，设置pathstr（裁剪后数据集路径），然后运行train.py （我已设置相对路径，但是还需解压GaitDatasetB\_Cutted.zip）。

迭代为10K次训练时间在3小时50分钟，另外我在训练过程中，训练速度在逐步加快。

**pycharm：**运行train.py

**命令行：**python train.py

**3、测试：**测试会加载loss最小的那个模型（最新保存的模型）

Checkpoint保存了训练好的模型，可以直接运行加载GaitSet\_test.py。

一次测试时间大概在37分钟左右。

**pycharm：**运行GaitSet\_test.py

**命令行：**python GaitSet\_test.py