**任务二问题**

1. 数据在网络中的维度顺序是什么？

NCHW

分别为batch size,channel,height,width

2. nn.Conv2d()中参数含义与顺序？

1)in\_channels:输入的图像或者feature map的通道数

2)out\_channels:输出通道数（卷积核的个数）

3)kernel\_size:卷积核尺寸

4)stride:步长，默认为1  
5)padding:添加到输入两边的零填充位数，默认为0

6)dilation:卷积核元素之间的空间间隔，默认为1

7)groups:分组数，决定输出通道与多少个输入通道连接，默认为1  
8)bias:是否在输出上添加一个可以学习的偏置，默认为True

9)padding\_mode:填充方式,有zeros和circular,默认为zeros

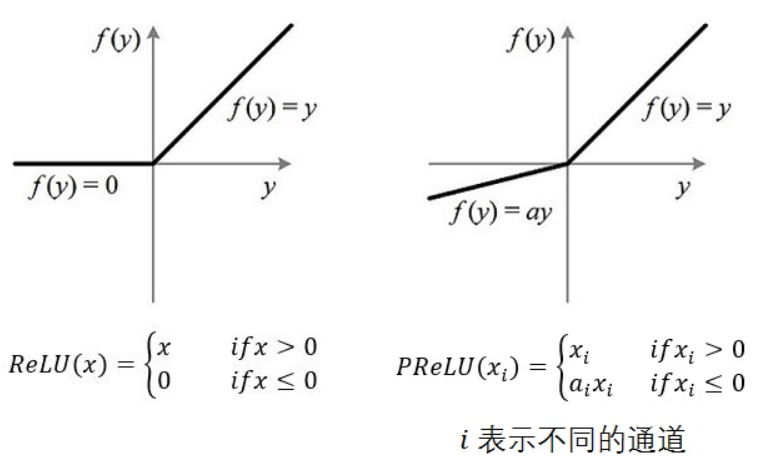
3. nn.Linear()是什么意思？参数含义与顺序？

线性变换层(全连接)，对输入做线性变换

1. in\_features:输入特征数
2. out\_features：输出特征数
3. Bias:是否在输出上添加一个可以学习的偏置，默认为True

4. nn.PReLU()与 nn.ReLU()的区别？示例中定义了很多 nn.PReLU()，能否只定义一个

PReLU？



PRelu为带参数的ReLu,当ai为0 时，退化为ReLu

PReLu可以从训练数据中学习到不同层不同通道的ai，能提升性能

若只定义一个PReLU，则功能退化为ReLU和Leaky ReLU

5. nn.AvgPool2d()中参数含义？还有什么常用的 pooling 方式？

1)kernel\_size:卷积核尺寸

2)stride:步长，默认值为kernel\_size

3)padding:添加到输入两边的零填充位数，默认为0

4)ceil\_mode:当设为True时，用ceil代替floor来计算输入尺寸，默认为False

5)count\_include\_pad:当设为True时，在均值计算时把零填充考虑进去,默认为True

6)divisor\_override:如果指定，将被当作除法因子，否则使用kernel\_size

6. view()的作用？

重构Tensor的维度，返回Tensor的新视图，但不改变Tensor本身的内容

**B第二部分问题**

1. 如何设置 GPU

device = torch.device(**"cuda"**)#指定当前的GPU

或着

device = torch.device(**"cuda:X"**)#指定编号为X的GPU

然后将需要的tensor通过to函数放到GPU里面去

1. 如何将数据/网络传入 CPU/GPU

先按照1的方式定义设备

然后通过to函数将数据/网络传入到该设备

3. 如何读取数据

见dataset/stage1\_baseline.py

见dataset/stage3\_baseline.py

其中stage1的训练数据采用了在线增强的方式

训练stage3的数据采用了离线增强的方式

**C第三部分问题**

1. 如何设置 loss

针对人脸关键点任务可以设置MSE Loss

criterion\_pts = nn.MSELoss()

2. 配合后续周学习，loss 都有哪些。分别有什么作用(常用的即可)

**分类Loss:**

1)二分类

torch.nn.BCEWithLogitsLoss

2)多分类

torch.nn.CrossEntropyLoss

torch.nn.logSoftmax+torch.nn.NLLloss

3)多类别

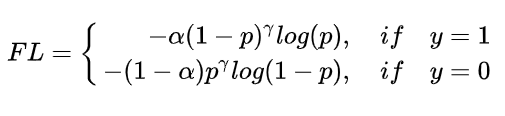
torch.nn.multiLabelSoftMarginLoss

torch.nn.multiLabelMarginLoss

4)针对类别不平衡，难易样本数量不平衡有

Weighted Cross Entropy Loss

Focal Loss



6)人脸识别常见Loss

L-Softmax Loss,Center Loss,Triplet Loss/Contrastive Loss,L2-Softmax Loss,Additive Margin Loss等

**回归Loss:**

1)均方误差MSE

nn.MSELoss()

1. 平均绝对误差 MSE

nn.L1Loss

1. Huber Loss/Smooth L1 Loss

nn.SmoothL1Loss

3. 如何设置优化器

优化器在torch.optim包下面，以Adam为例

optimizer = torch.optim.Adam(model.parameters(),lr=args.lr,weight\_decay=args.weight\_decay)

输入参数分别为：

1. params:待优化的参数
2. Lr:学习率，默认为1e-3
3. betas:计算过程中需要的系数默认为(0.9, 0.999)
4. eps:增加数值稳定性，默认为1e-8
5. weight\_decay:权重衰减系数，默认为0
6. Amsgrad:是否使用AMSGrad变体，默认为**False**

4. 配合第 8 周内容，常用的优化器有哪些

1)SGD

2)SGD+Momentum

3)Nesterov

4)Adagrad

5)RMSProp

6)Adam

**E第五部分问题**

2. optimizer.zero\_grad()与 optimizer.step()的作用是什么？

1)Pytorch中梯度是累加的，为了得到每个mini-batch的梯度，需要通过zero\_grad清空上一次的结果

2)我们训练是以mini-batch为单位，在得到每个mini-batch的梯度之后，用step来更新网络参数

3. model.eval()产生的效果？

eval函数只对某些层有效果，比如Dropout和Batch Norm，因为他们在训练和测试的时候处理方式不一样

4.model.state\_dict()的目的是？

state\_dict 是一个字典对象,将每一层(可训练层)与它的对应参数建立映射关系，如model的每一层的weights及偏置等等。model.state\_dict()可以得到网络模型参数，但model.state\_dict()是浅拷贝，返回的参数仍然会随着网络的训练而变化。

1. 何时系统自动进行 bp？

当调用loss.backward()

1. 如果自己的层需要 bp，如何实现？如何调用？

实现：

1）继承torch.autograd.Function

2）重写forward和backward并添加@staticmethod注解

调用：

1)将操作封装成一个函数

2)或者给自定义方法通过apply取别名

B. Finetune时，有时还要固定某些层不参与训练，请回答如何 freeze 某些层

将需要固定层的requires\_grad 设为 **False**