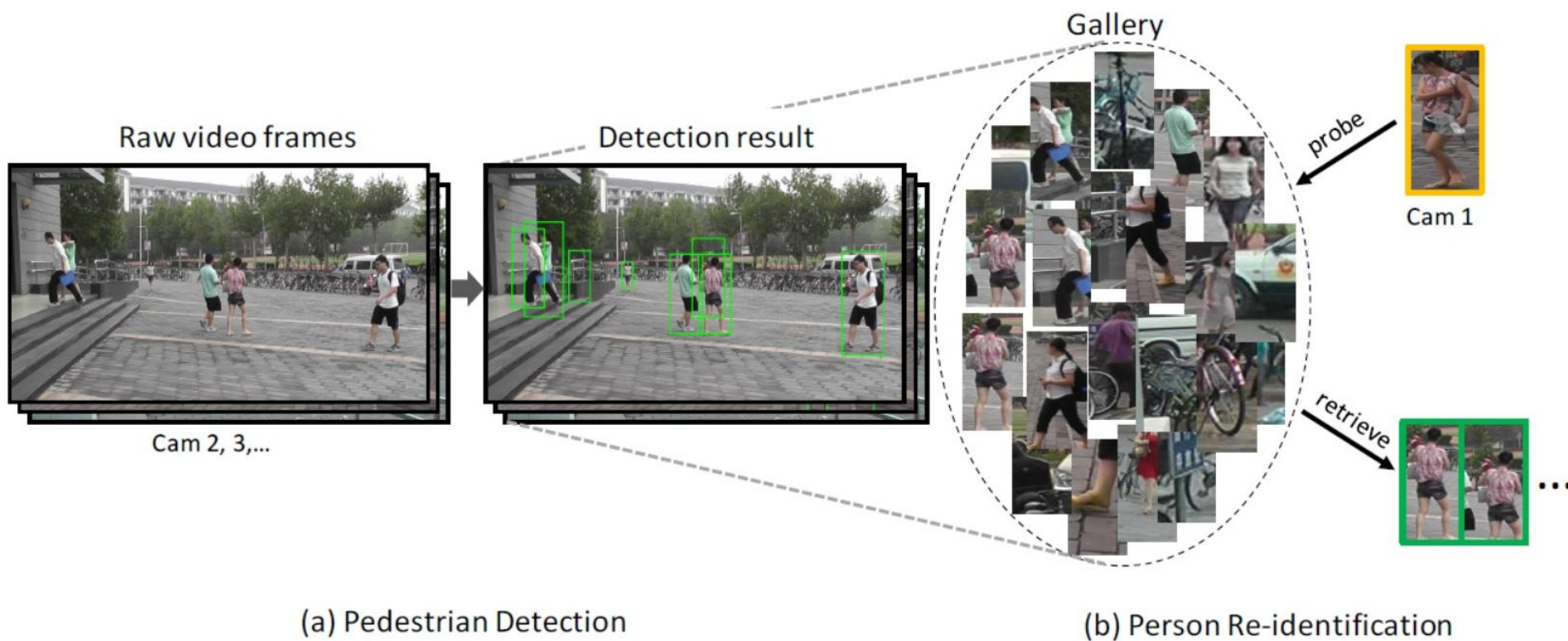


行人重识别

✓ Person re-identification要做什么

📎 在多摄像头的复杂场景中，快速定位查找指定目标的所有结果



行人重识别

✓ 应用场景

📌 有人的地方就有江湖!

📌 一分钟我要他的所有信息!

📌 基本思想其实就是相似度匹配

📌 但是如何做的更好呢?

智能安防



人机交互



智能商业—无人超市



相册聚类



行人重识别

✓ 存在的挑战

📎 如果只是相似度匹配，那就太简单了，这里面问题还挺多的！



(a)低分辨率



(b)遮挡



(c)视角、姿势变化



(d)光照变化



(e)视觉模糊性

行人重识别

✓ 论文中常用数据集

✎ CUHK03: 香港中文大学 (CUHK) 校园

✎ Market-1501 : 清华大学校园

✎ DukeMTMC: 8 个摄像机

✎ MSMT17: 12个室外和3个室内 (校园)

Dataset	Identities	Cameras
Market1501	1501	6
DukeMTMC-reID	1812	8
CUHK03	1467	10(5 pairs)
MSMT17	4101	15

行人重识别

✓ 论文中常用数据集

📎 DukeMTMC数据集，基本都是把视频经过行人检测处理成输入和标签



行人重识别

✓ 评估标准

✎ 基本所有论文都会提到这两个概念：rank1和map值

✎ 返回结果中包含了一系列的图像，rank1指的是第一张结果正确

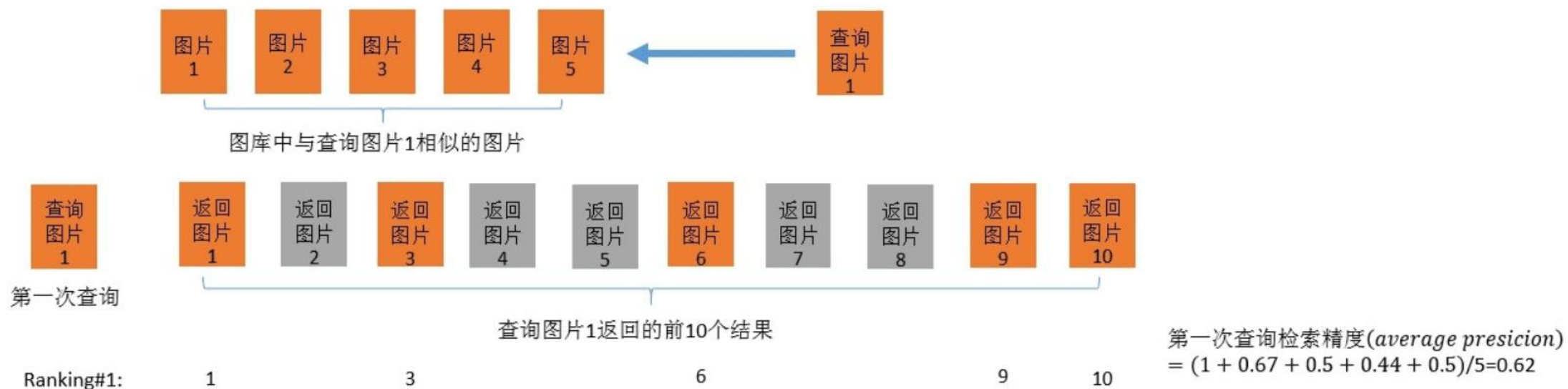


行人重识别

✓ Map值计算

✎ map要计算多次输入的综合ap结果，每张测试图像计算如下：

✎ ap $(1 + 2/3 + 3/6 + 4/9 + 5/10)$ 也就是一张正确图像本应该是第几个

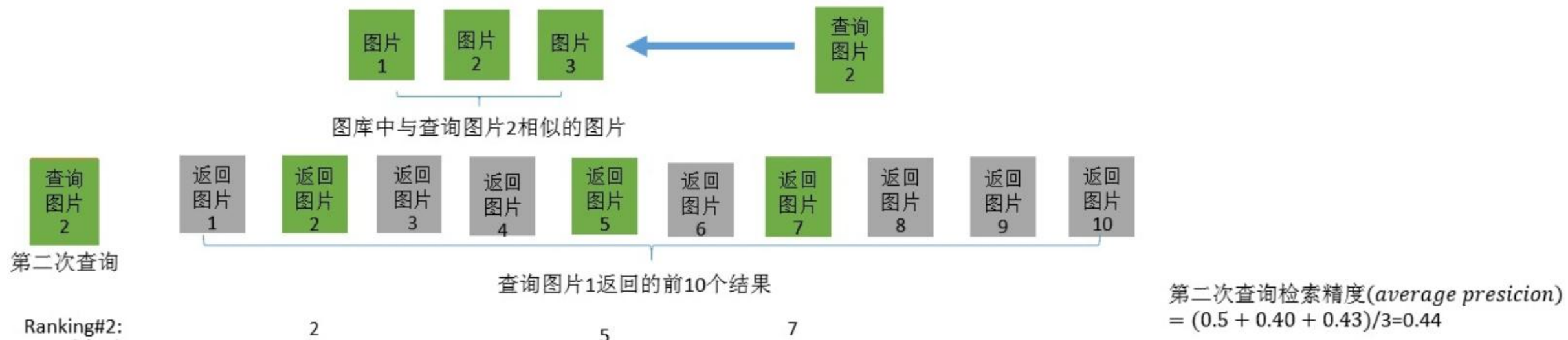


行人重识别

✓ Map值计算

✎ 举例：两个测试数据，ap分别为0.62，0.44则 $\text{map} = (0.62 + 0.44) / 2 = 0.53$

✎ ap $(1/2 + 2/5 + 3/7)$ 也就是一张正确图像本应该是第几个



行人重识别

✓ 损失函数定义

✎ 通常是分类损失+Triplet loss（目标其实就是为了特征提取的更好）

✎ Triplet loss需要准备3份数据（可以从一个batch中选择）
其中Anchor表示当前数据，Positive是跟A相同人的数据，Negative是不同人的数据。



Anchor



Positive



Anchor

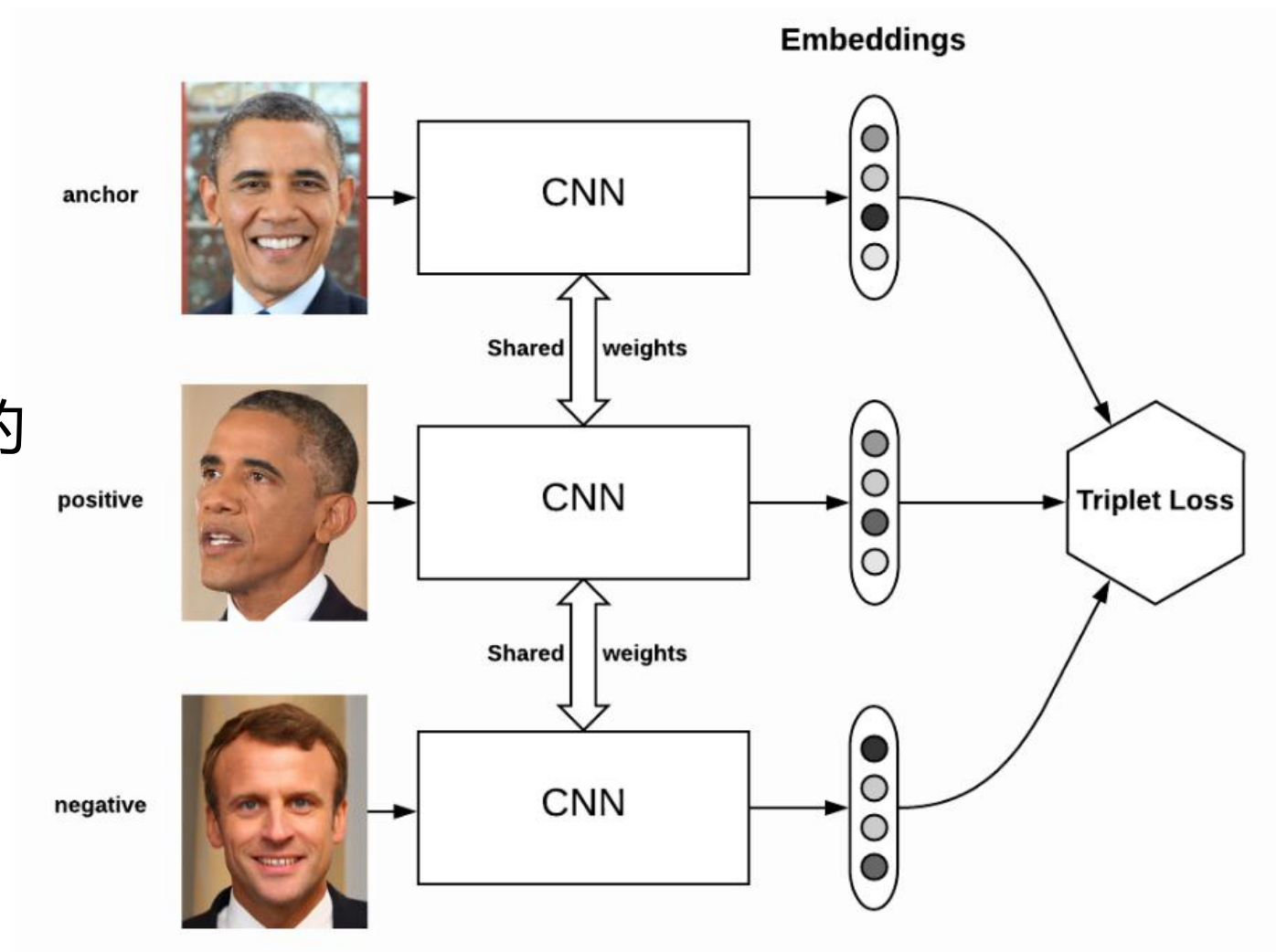


Negative

行人重识别

✓ Triplet loss

- ✎ 分别对3份数据进行编码
- ✎ 这三兄弟经过的网络是一样的
- ✎ 然后计算他们之前的差异
- ✎ 通过差异值来更新权重参数



行人重识别

✓ Triplet loss

✎ 目的其实很简单，只需让A跟P非常接近，A与N尽可能远离

✎ 公式： $\|f(A) - f(P)\|^2 \leq \|f(A) - f(N)\|^2$ 其中f表示通过网络进行编码

✎ 是否会存在问题呢？如果f把所有的输入都编码成0，依旧成立！

✎ 那这个目标得重新修改一下： $\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + a \leq 0$
(其中a通常叫做margin，也就是间隔，表示d(A,P) 与d(A,N)至少得相差多少)

行人重识别

✓ Triplet loss

✎ Triplet loss: $L(A, P, N) = \max(\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + a, 0)$

✎ 但是对于约束条件: $d(A, P) + a \leq d(A, N)$, 这件事很难嘛?
理论上都是A与P很近, A与N较远, 这还需要学嘛。。。

✎ 实际中用的最多的是hard negative方法, 也就是在选择样本的时候:
让 $d(A, P) \approx d(A, N)$, 这样给网络一些挑战, 才能刺激它来学习!

行人重识别

✓ 使用resnet构建基本网络架构

📎 论文中大部分都是基于resnet50来进行特征提取

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times 3, 64 \\ 3\times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times 3, 64 \\ 3\times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 64 \\ 3\times 3, 64 \\ 1\times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 64 \\ 3\times 3, 64 \\ 1\times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 64 \\ 3\times 3, 64 \\ 1\times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3\times 3, 128 \\ 3\times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times 3, 128 \\ 3\times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1, 128 \\ 3\times 3, 128 \\ 1\times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1, 128 \\ 3\times 3, 128 \\ 1\times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1, 128 \\ 3\times 3, 128 \\ 1\times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3\times 3, 256 \\ 3\times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times 3, 256 \\ 3\times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times 1, 256 \\ 3\times 3, 256 \\ 1\times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times 1, 256 \\ 3\times 3, 256 \\ 1\times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1\times 1, 256 \\ 3\times 3, 256 \\ 1\times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3\times 3, 512 \\ 3\times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times 3, 512 \\ 3\times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 512 \\ 3\times 3, 512 \\ 1\times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 512 \\ 3\times 3, 512 \\ 1\times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1, 512 \\ 3\times 3, 512 \\ 1\times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

行人重识别

✓ Relation Network for Person Re-identification

✎ 如何更好的利用图像中的信息呢？

✎ 通常都是对整个输入数据进行特征提取，但是缺少了局部信息

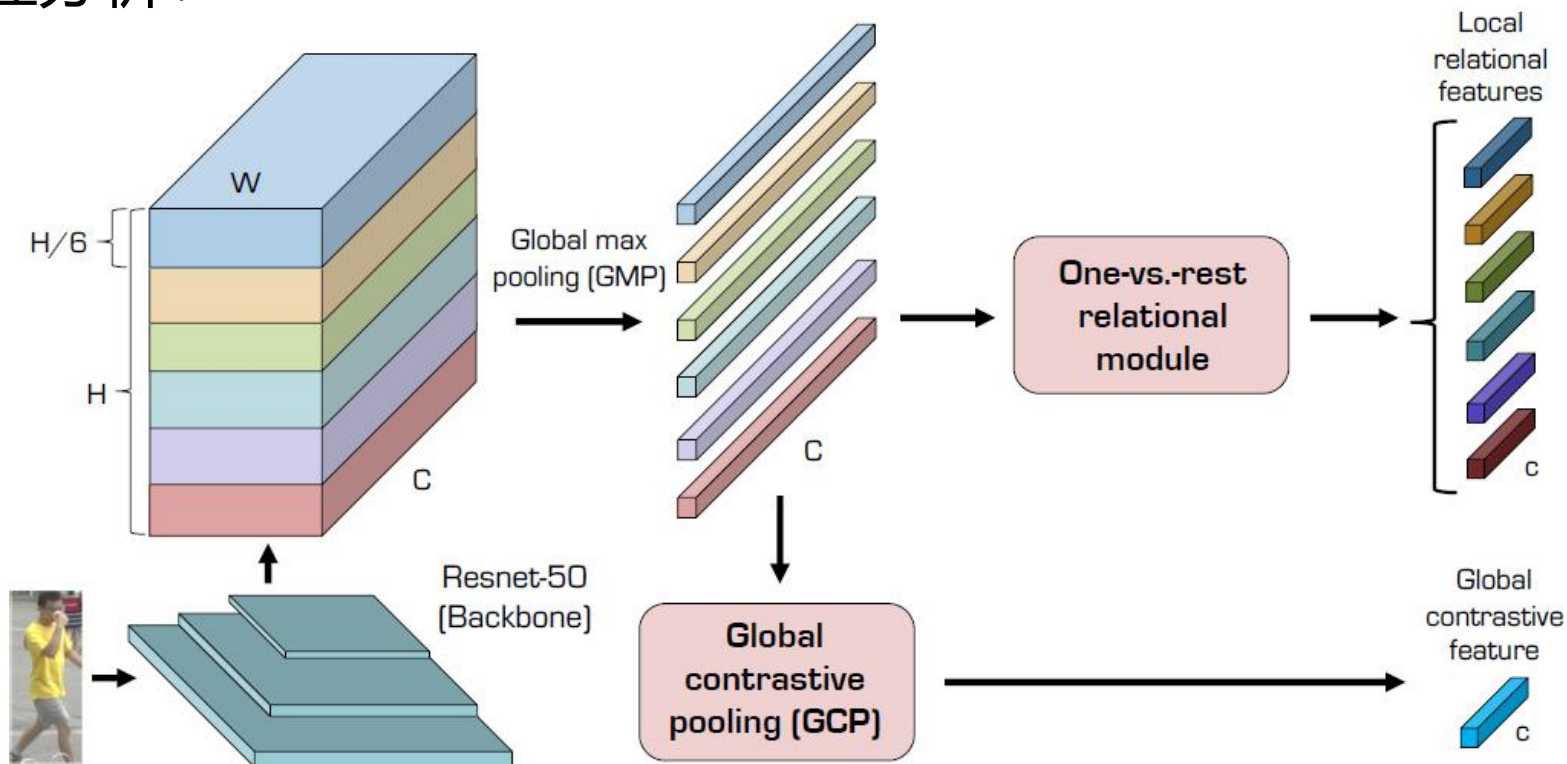
✎ 如果把图像分成几个区域（头，身子，腿。。），这样可能又太散了

✎ 能不能既考虑局部与整体信息，也同时加入他们的联系呢？

行人重识别

✓ Relation Network for Person Re-identification

✎ 整体流程分析:



行人重识别

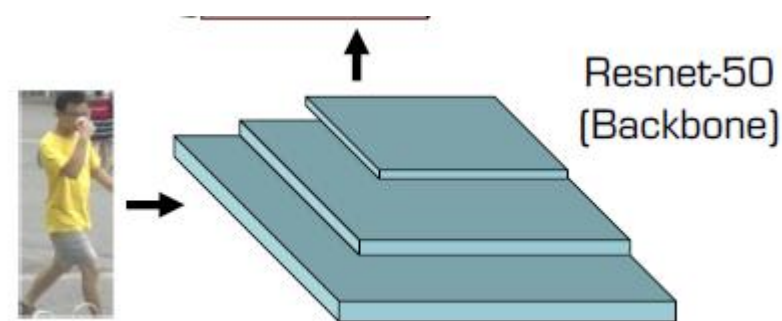
✓ Relation Network for Person Re-identification

✎ 步骤1：首先对整体进行特征提取

✎ 先将输入数据resize，然后输入到resnet中

✎ 基本所有的ReID模型都是先进行这一步

✎ 加载imagenet预训练的resnet50模型



行人重识别

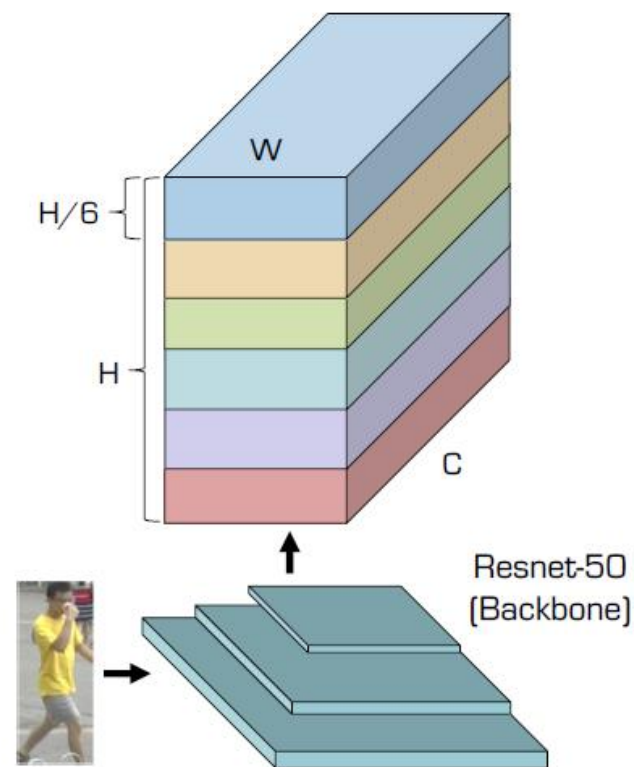
✓ Relation Network for Person Re-identification

✎ 步骤2：将特征图分块

✎ 直接在h维度进行截取，并没有利用其它辅助信息

✎ 为什么是6个呢？4个不行吗？

✎ 为了得到更综合的特征，源码分了三组实验
($1/6h$, $1/4h$, $1/2h$)



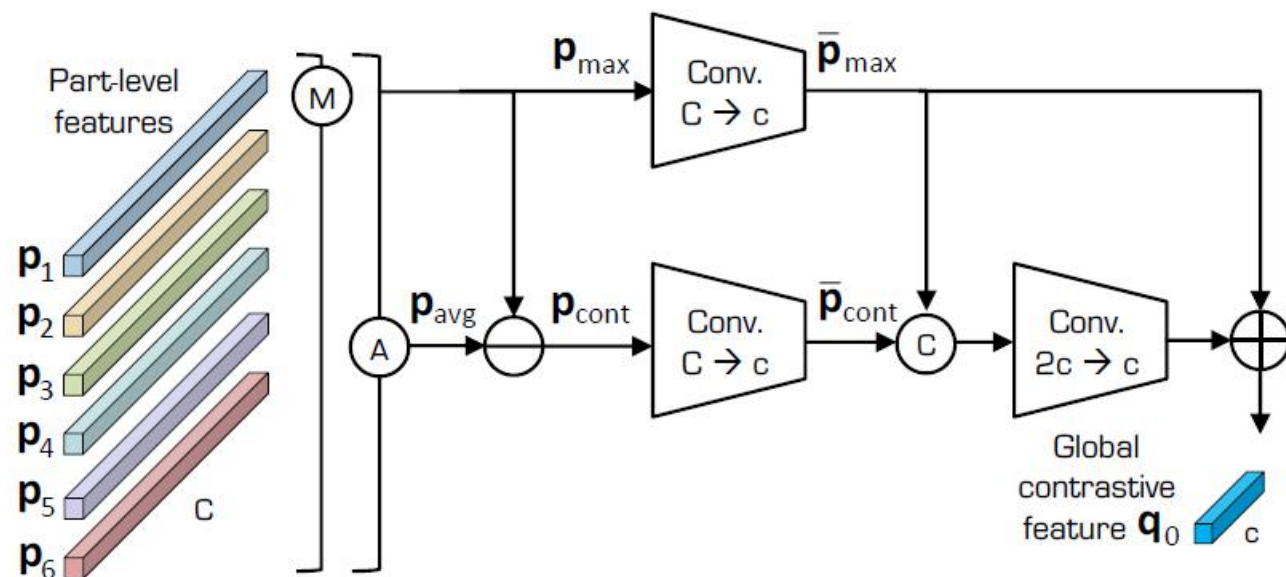
行人重识别

✓ Relation Network for Person Re-identification

✎ 步骤3：计算GCP特征，

✎ avgPool会引入局部与背景信息

✎ 差异特征更好描述局部关系



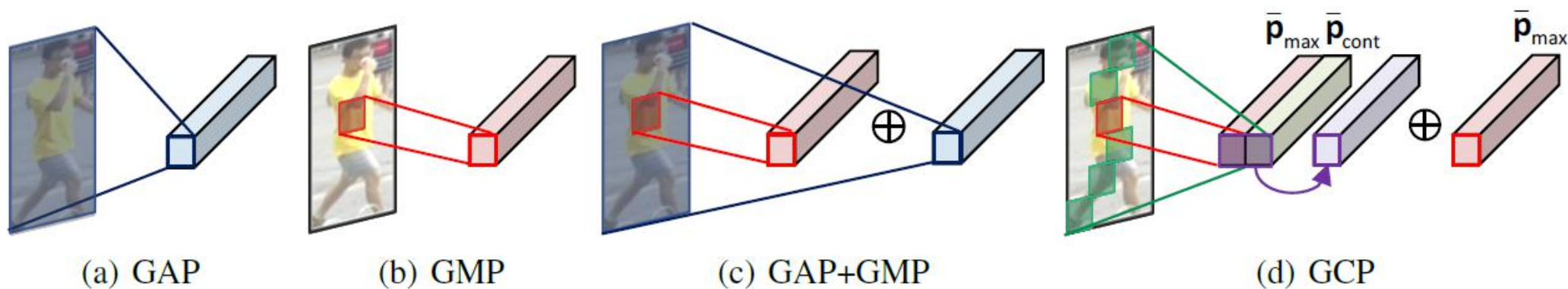
✎ 一定程度上去掉了了一些噪音特征的干扰（有待其它领域算法来证明。。。）

行人重识别

✓ Relation Network for Person Re-identification

✎ GAP相当于全局都有了，GMP相当于最核心那一块。

✎ GCP之后相当于各个关键点的信息了 (contrastive)



行人重识别

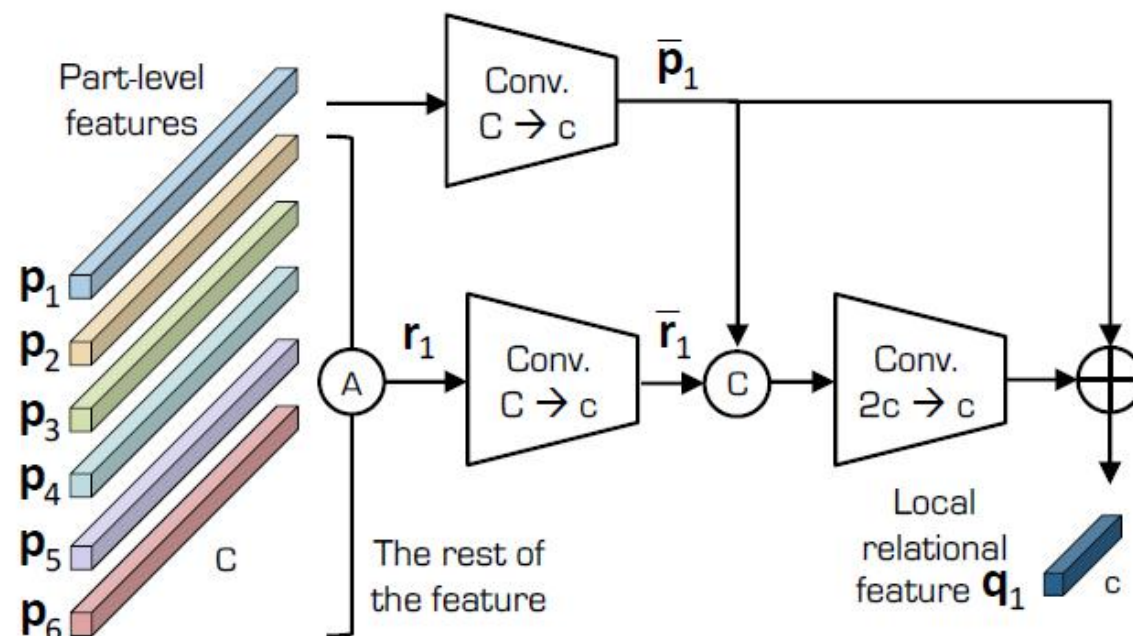
✓ Relation Network for Person Re-identification

✎ 步骤4: one vs rest

✎ 有多少种组合呢? 都要算的!

✎ 出发点就是别把局部信息孤立来算

✎ 其实有点类似attention的方法

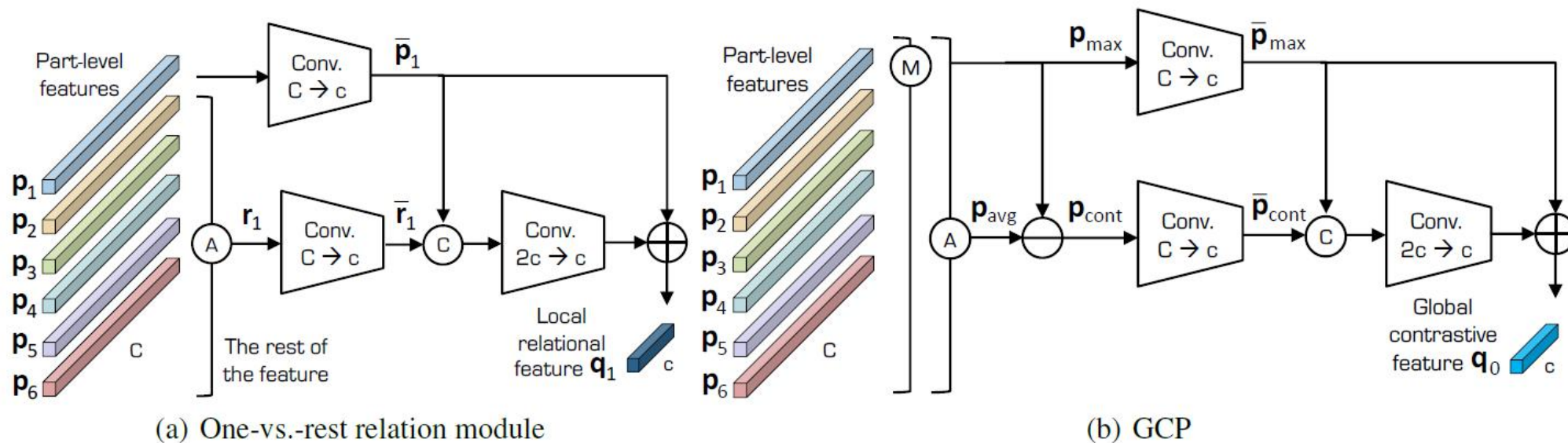


行人重识别

✓ Relation Network for Person Re-identification

✎ 步骤4：损失函数

✎ 还是那兄弟俩，分类损失和triplet loss，但是好多地方都去凑热闹



行人重识别

✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 听这名字，应该就能看出来他主要解决遮蔽现象

✎ 数据重构，Occluded-DukeMTMC（query里全是遮蔽现象的数据）

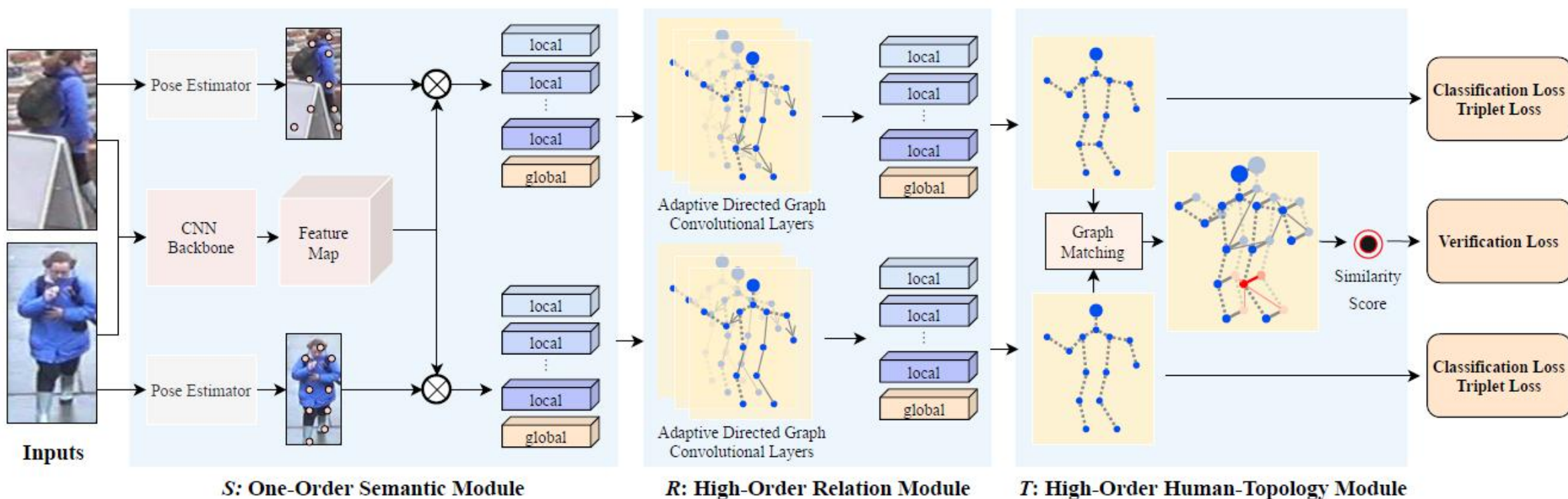
✎ 提出了三阶段的模型：1.关键点局部特征提取；2.图卷积融合关键点特征；3.基于图匹配的方式来计算相似度并训练模型。

✎ 整体思路相当可以！这招在各种跟验证相关的任务中都可以套（人脸验证）

行人重识别

✓ Learning Relation and Topology for Occluded Person Re-Identification

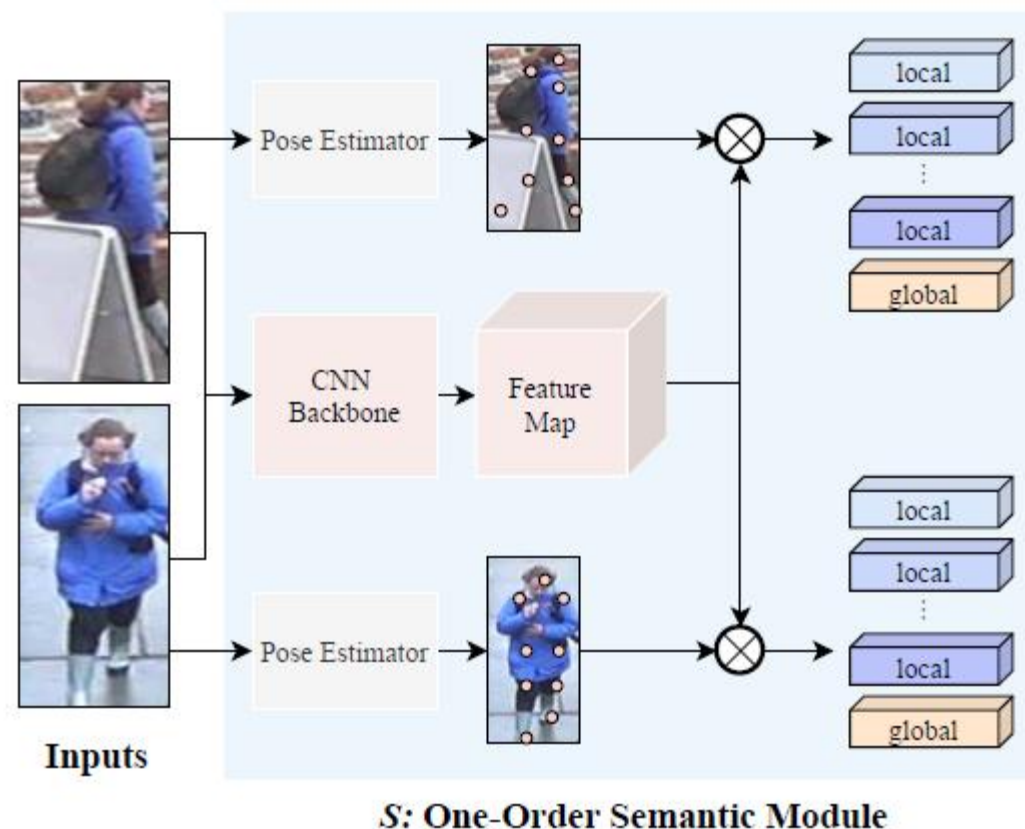
✎ 总结起来就是分三个阶段来完成特征提取，重点要解决遮蔽现象的局部特征



行人重识别

✓ Learning Relation and Topology for Occluded Person Re-Identification

- ✎ 一阶段：关键点局部特征提取
- ✎ 选择一个pose estimation模型即可
- ✎ 得到的是各个关键点的热度图信息
- ✎ 通过热度图得到原始特征图的局部特征

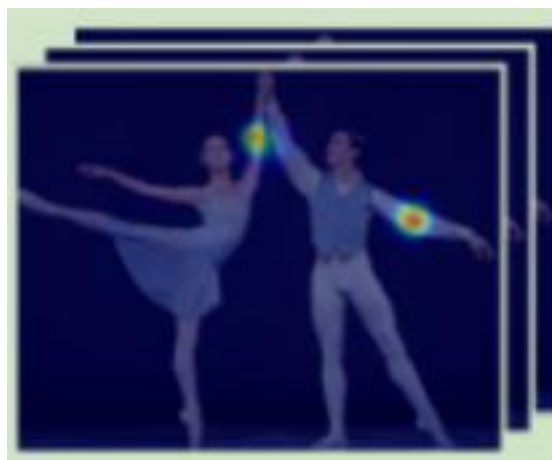


行人重识别

✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 其实就是一个乘法操作，用pose estimation得到的热度图来计算局部特征

✎ 左图可以当做是 ，右图就是其关键点信息的热度图



行人重识别

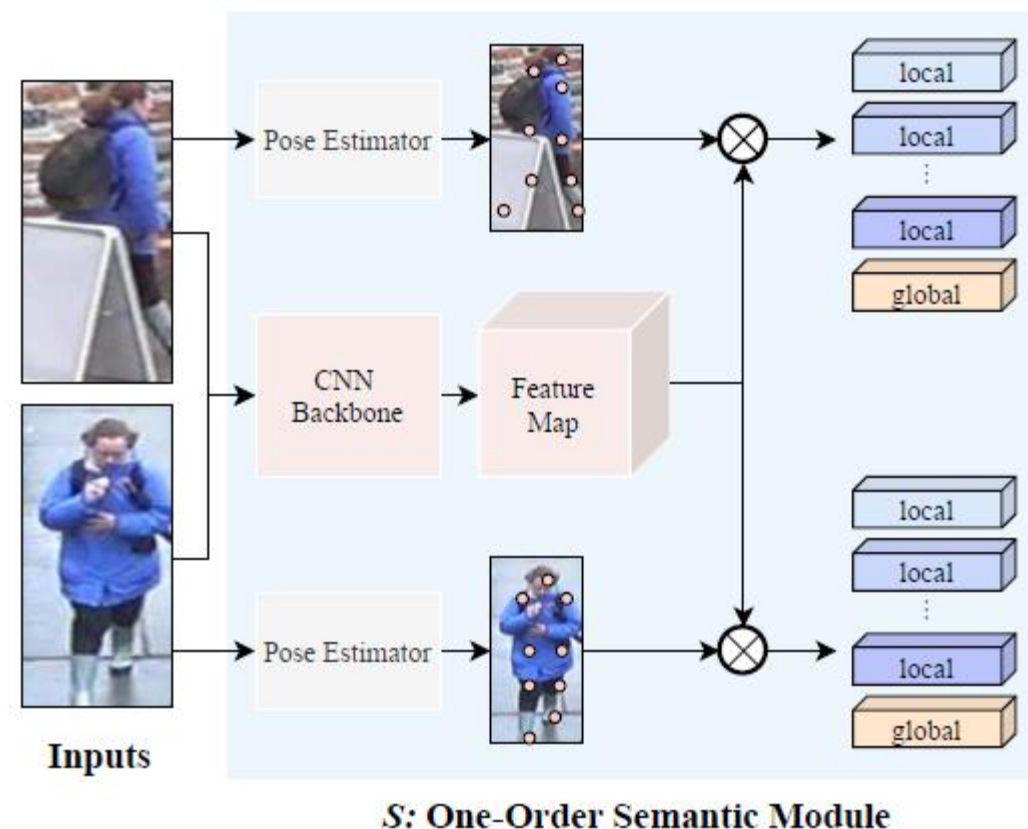
✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 一阶段：关键点局部特征提取

✎ 跟上篇算法一样这里同样也加了多个损失

✎ local与global的都需要进行训练

✎ 全部特征是global average pooling得到



行人重识别

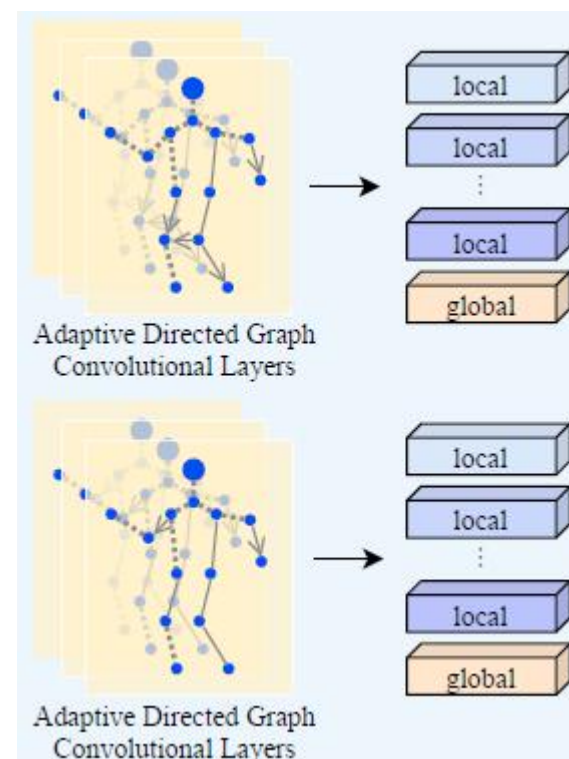
✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 二阶段：局部特征关系整合（图卷积）

✎ 如何才能更好的利用局部特征呢？加入关系！

✎ 先初始化邻接矩阵来进行图卷积

✎ 邻接矩阵在学习过程中更新（如何才能更好的针对每个输入利用不同的局部特征）

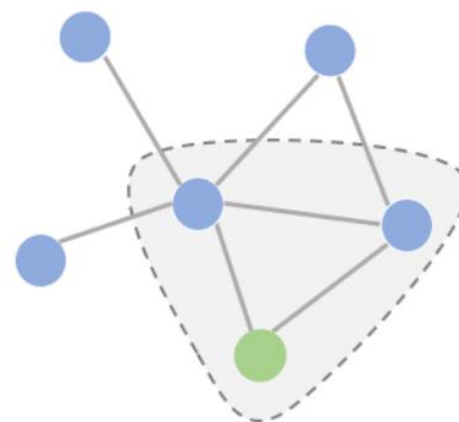
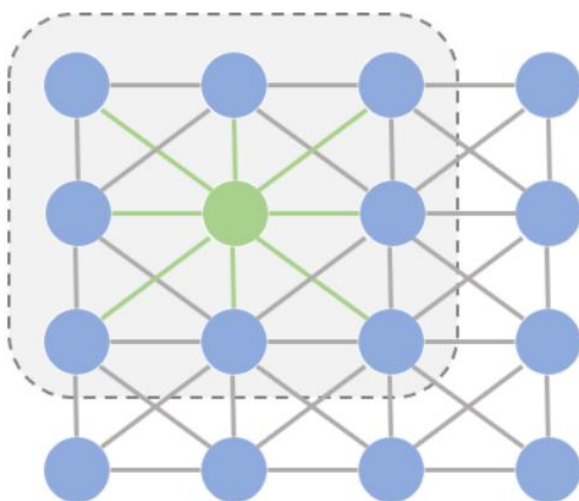


行人重识别

✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 图卷积怎么做呢？其实总结起来就是如何利用各个点的特征。

✎ 这个任务中，就是得到邻接矩阵A来指导每个关键点特征如何跟其他关键点特征进行计算，并且A矩阵也要进行学习！



行人重识别

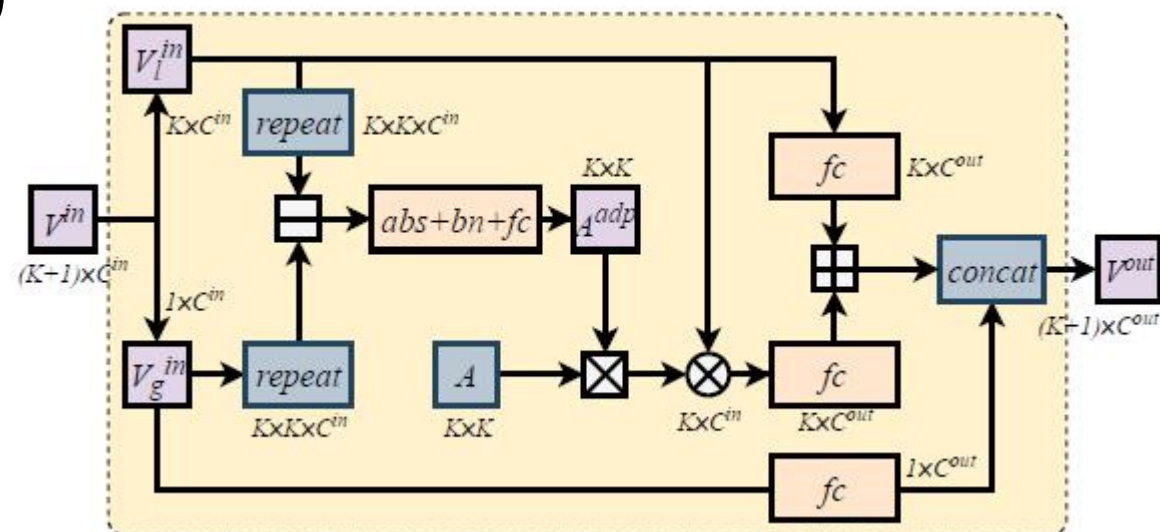
✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 二阶段：局部特征关系整合（图卷积）

✎ 和整体特征差异越大的，越离群

✎ 利用差异特征来学习邻接矩阵A

✎ 有了A就能开始图卷积啦，用它来指导如何利用不同关键点的特征进行组合
最终再与输入的局部特征进行整合。



行人重识别

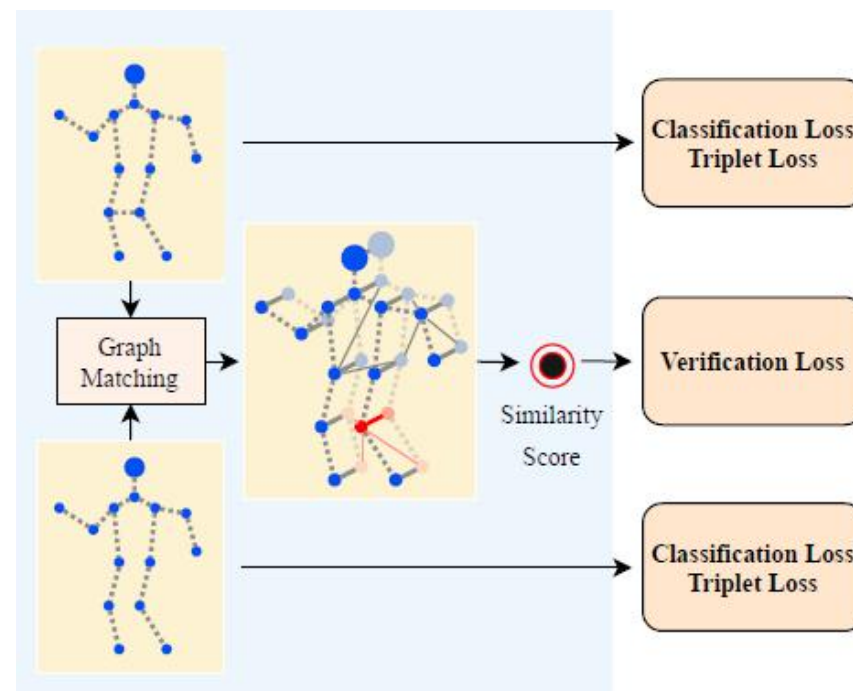
✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 三阶段：图匹配

✎ 如何才能训练的更好呢？还是AP,AN问题！

✎ 输入两张图像（经过了前两阶段后的结果）

✎ 计算他们之间的相似度
(其实输入是两组，第一组A,P；第二组，A,N；)



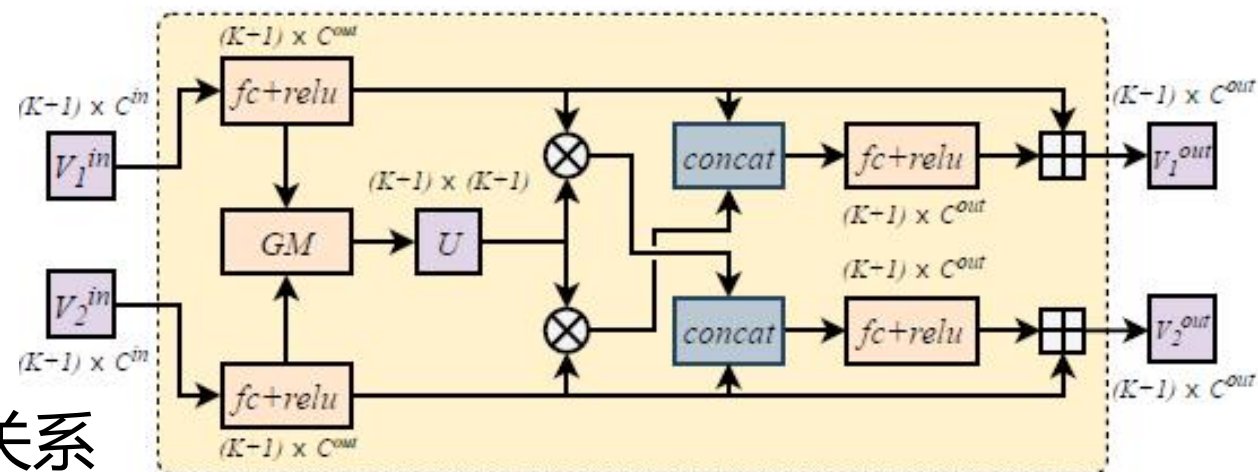
行人重识别

✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 三阶段：图匹配

✎ 图匹配就是要一个相似度矩阵U

✎ 例如14*14，表示两个图之间的关系



✎ 注意这里面是一个交叉的过程(cross)，分别交叉来得到各自匹配的特征结果

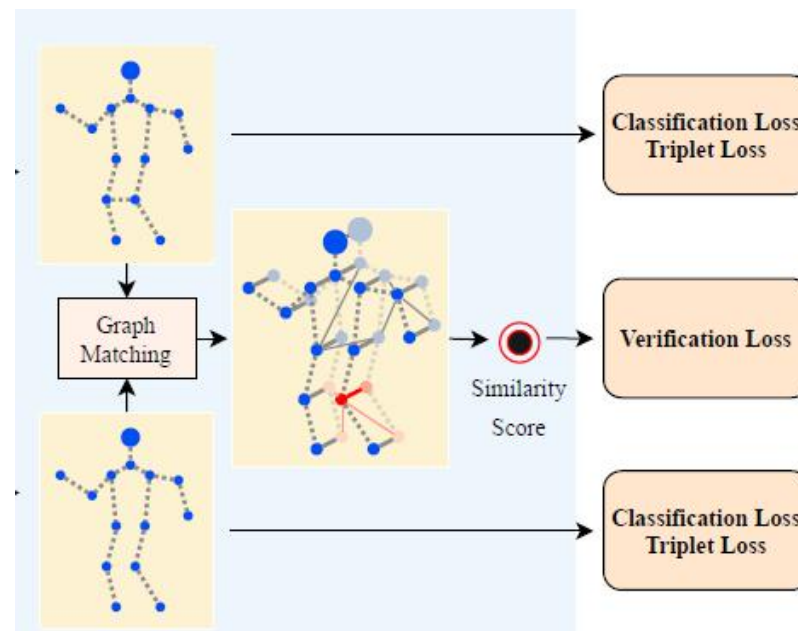
行人重识别

✓ Learning Relation and Topology for Occluded Person Re-Identification

✎ 三阶段：图匹配

✎ 还引入了新的损失函数:验证损失

✎ 就是 $\text{sigmoid}(\text{emb1}, \text{emb2})$ 的结果



✎ 整体框架知识点涉及比较多，可以说本届CVPR中比较值得一看的！