# ГЛАВА 2 Обучение модели для отследования людей в нескольких видеопотоках （训练模型）

## ReID模型的训练

### 数据集准备

#### MSMT17数据集

在CVPR2018会议上，提出了一个新的更接近真实场景的大型数据集**MSMT17**，即**Multi-Scene Multi-Time**，涵盖了多场景多时段。

MSMT17数据集采用了安防在校园内的15个摄像头网络，其中包含12个户外摄像头和3个室内摄像头。为了采集原始监控视频，在一个月里选择了具有不同天气条件的4天。每天采集3个小时的视频，涵盖了早上、中午、下午三个时间段。因此，总共的原始视频时长为180小时。

图片包含 游戏机

描述已自动生成

基于Faster RCNN作为行人检测器，三位人工标注员用了两个月时间查看检测到的包围框和标注行人标签。最终，得到4101个行人的126441个包围框。和其它数据集的对比以及统计信息如下图所示。

图形用户界面, 应用程序, 表格

描述已自动生成

MSMT17与其他数据集相比的优势如下：

（1）数目更多的行人、包围框、摄像头数；

（2）复杂的场景和背景；

（3）涵盖多时段，因此有复杂的光照变化；

（4）更好的行人检测器（faster RCNN），bounding box更精准

**评估协议**

按照训练-测试为1：3的比例对数据集进行随机划分，而不是像其他数据集一样均等划分。这样做的目的是鼓励高效率的训练策略，由于在真实应用中标注数据非常昂贵。

最后，训练集包含1041个行人共32621个包围框，而测试集包括3060个行人共93820个包围框。对于测试集，11659个包围框被随机选出来作为query，而其它82161个包围框作为gallery。

测试指标为CMC曲线和mAP. 对于每个query, 可能存在多个正匹配。

### 训练原理

Reid模型的训练原理如下：

1. 数据预处理：reid首先将数据集中的图片进行0.5概率的裁剪、缩放、对称和数据增强等预处理操作，以便提高模型对不同姿态和光照条件下的人员图像的鲁棒性。
2. 特征提取：Reid使用训练好的ResNet系列卷积神经网络作为特征提取器，从输入图像中提取出512维特征向量。同时fastreid还引入一些新的特征提取方法，例如Scale-Aware Alignment (SAA) 和 random-erasing等，以进一步提高特征表达能力。
3. 度量学习：通过度量学习的方式将特征向量映射到一个低维空间中，并采用triplet loss和Cross Entropy Loss损失函数来优化模型，使得同一身份的样本在特征空间中距离更近，不同身份的样本在特征空间中距离更远。
4. 模型优化：使用Adam优化器对模型参数进行优化，以提供模型的训练效率和精度

### 构建模型

为了加速模型的收敛和提高模型性能，我们使用了在ImageNet数据集上预训练的卷积神经网络模型ResNet50。ImageNet是一个广泛使用的图像分类数据集，其中包含超过1000个类别的140万张图像。ResNet50模型使用了深度残差网络结构，可以有效地处理大规模图像分类任务。在训练完成后，改模型的权重可以被用于其他计算机视觉任务中以提高任务的性能和效率，比如我们的行人重识别。

预训练的ResNet50模型输出特征维度是2048。我们在主干网络后面继续接入了一个平均池化层和一个线性分类器。线性分类器的输出维度是N。N 表示训练数据的 ID 数目。

日程表

描述已自动生成

### 损失函数和优化算法

损失函数：

我们的reid模型会产生两个输出： features f 进行预测 logits p。features f 被用于计算 triplet loss，logits p 用于计算交叉损失熵。

**Cross Entropy Loss**

CrossEntropyLoss是一个常用的分类损失函数，通常用于训练深度神经网络进行图像分类任务。具体地，对于一个输入样本x和其真实标签y，CrossEntropyLoss的计算公式如下：

图形用户界面

描述已自动生成

其中C是类别数，yi表示第i个类别的概率（即真实标签），yi^表示模型预测x属于第i个类别的概率。

CrossEntropyLoss的目标是最小化预测与真实标签之间的差距，使得模型能够更准确地预测出每个样本所属的类别。

**Triplet Loss**

Triplet Loss是在人脸识别领域中广泛应用的损失函数之一，其目的是将同一个人的图像嵌入向量（embedding）拉近，不同人的图像嵌入向量拉远。具体来说，Triplet Loss会为每个样本学习一个嵌入向量，该向量能够使得同一个人的图像距离更近，不同人的图像距离更远。

对于一个triplet包括anchor、positive和negative三张图片，Triplet Loss的计算公式如下：

徽标, 公司名称

描述已自动生成

其中，d(a,p)表示anchor和positive之间的欧几里得距离，d(a,n)表示anchor和negative之间的欧几里得距离，m为margin，是一个预先设定的超参数，通常为正数。这个公式的含义是：如果当前的anchor和positive之间的距离减去anchor和negative之间的距离加上margin小于等于0，则说明当前的嵌入向量已经足够好，不需要再进行优化；否则，需要更新模型参数以获取更好的嵌入向量。

Triplet Loss通常与Batch Hard Triplet Mining结合使用，即在每个batch中选择最难的triplet进行训练，以提高模型的性能。

### 训练步骤

1.使用 ResNet50（初始化权重来自于 ImageNet 的预训练模型），然后改变其全链接层为N。N 表示训练数据的 ID 数目。

2.我们随机采样 P 个身份ID，并且对每个ID采集 K 张，最后一个batch size B = P \* K，在这篇论文中，我们设置P=16,K=4

3.我们改变每张图像的大小为 256 × 128，并且使用 0 值填充 10 个像素，然后使用随机剪切的方式，重新剪切 256 × 128 大小的图像。

4. 每张图像以 0.5 的概率值，随机进行水平反转。

5. 每幅图像都被解码为[0,1]中32位浮点原始像素值，然后我们通过减去0.485,0.456,0.406，再除以来对RGB通道进行归一化分别为0.229、0.224、0.225。

6.模型输出的ReID features f 进行预测 logits p。

7.ReID features f 被用于计算 triplet loss，logits p 用于计算交叉损失熵， triplet loss 的 margin m 设置为0.3

8.采用Adam方法对模型进行优化。初始学习率设为0.00035，在第40个epoch和第70个epoch分别降低0.1。总共有120个epoch

【添加伪代码】 或者【流程图】

### 训练过程

电脑屏幕截图

描述已自动生成

图形用户界面, 文本

描述已自动生成

文本

描述已自动生成

训练结果可视化：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Dataset | Rank-1 | Rank-5 | Rank-10 | mAP | mINP | Metric |
| MSMT17 | 84.19 | 90.95 | 95.10 | 62.89 | 14.94 | 64.04 |

【这里要加入与其他数据集训练结果的对比吗？】

图片包含 图示

描述已自动生成

## YOLOv5+DeepSORT模型的训练

### 数据集准备和数据预处理

MARS 全称 Motion Analysis and Re-identification Set，是一个大规模的行人重识别视频数据集，也是 Market-1501 数据集的扩充。

围栏里的人群

描述已自动生成

数据是由六个近乎同步的摄像机收集的。数据集包含 1,261 个不同的行人，每人至少被 2 个摄像机拍摄。行人的姿势、着装颜色和光照的差异，以及较差的图像质量，使其很难产生高匹配精度。此外，该数据集包含 3,248 个干扰因素，以使其更加切合实际情况。DPM 模型和 GMMCP 跟踪器被用来自动生成短轨迹（大多为 25 – 50 帧长）。

bbox\_train文件夹中，有625个子文件夹（代表着625个行人id），共包含了8298个小段轨迹（tracklets），总共包含**509,914**张图片。

bbox\_test文件夹中共有636个子文件夹（代表着636个行人id），共包含了12180个小段轨迹（tracklets），总共包含**681,089**张图片。在实验中这个文件夹被划分为图库集（gallery）+ 查询集（query）。

名称为**00-1**子文件夹表示无用的图片集，他们对应的行人id被设为-1，一般在算法中直接无视**pid = -1**的图片。

### 构建模型

1. 使用YOLO目标检测器来检测每帧图像中的人物
2. 使用DeepSORT算法来跟踪每个人物并将其与之前的轨迹相关联
3. 将fastreid用于提取特征，使得DeepSORT可以区分不同的人物
4. 确定模型超参数。训练模型的参数如下：

DEEPSORT:

REID\_CKPT: "./fast-reid/checkpoint/model-final.pth"

MAX\_DIST: 0.2

MIN\_CONFIDENCE: 0.3

NMS\_MAX\_OVERLAP: 0.5

MAX\_IOU\_DISTANCE: 0.7

MAX\_AGE: 140

N\_INIT: 3

NN\_BUDGET: 100

### 构建损失函数和优化算法

DeepSORT的训练主要是针对特征提取模块进行训练，因此针对DeepSORT的训练相对于。原文作者采用的损失函数是交叉熵损失函数（CrossEntropyLoss），我们前面在fastreid的训练中也使用了这个损失函数。

优化算法为带动量的随机梯度下降算法（SGD with momentum），权重衰减系数为5e-4。

### 训练步骤

【伪代码or图像·】

### 训练过程

文本

描述已自动生成

经过50代的训练，得到结果：

背景图案

描述已自动生成

图形用户界面, 应用程序

描述已自动生成

图表

描述已自动生成

最终结果：train\_acc=92.07%, test\_acc=91.34。