

## 一、 背景

ReID挑战：为了学习Robust(健壮)的人物特征，人物图像的姿势变化是关键挑战之一。

目前针对该问题的主要是往执行人类对齐和学习基于人类区域的表示这两个方向进行，推断通常需要额外的姿势信息和计算成本。

## 二、 创新点

- 1、作者提出了特征提取生成式对抗网络（FD-GAN），在姿势改变的情况下还能保持身份特征的连续性，但推断复杂性并没有增加。
- 2、基于Siamese structure（双胞胎）结构的新颖框架，具有多种关于人类姿势和身份的新型判别器。
- 3、身份判别器、姿势判别器和验证分类器连同重建损失以及全新的同姿势损失一起正则化特征学习过程，来实现鲁棒的行人重识别。
- 4、集成了新颖的same-pose loss(姿势损失)，通过图像编码器在视觉特征中减少姿势和背景这种与身份判断无关的信息，在推断过程中不需要额外的姿势信息和计算成本。

## 三、 Result

提出的FD-GAN在Market-1501, CUHK03和DukeMTMC-reID数据集上取得了最佳的重新识别效果，这表明了所提出的FD-GAN的有效性和强大的特征提取能力。

## 四、 FD-GAN

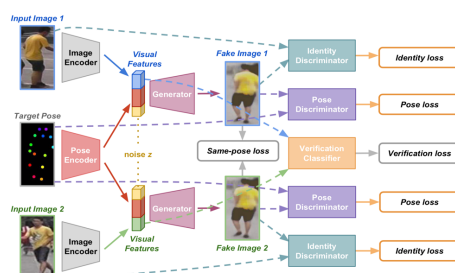


Figure 2: The Siamese structure of the proposed FD-GAN. Robust identity-related and pose-unrelated features are learned by the image encoder  $E$  with a verification loss and the auxiliary task of generating fake images to fool identity and pose discriminators. A novel same-pose loss term is introduced to further encourage learning identity-related and pose-unrelated visual features.

上图是 FD-GAN 的双胞胎结构。该结构利用验证损失，通过图像编码器  $E$  学习鲁棒的身份相关和姿势无关的特征，而生成假图像的辅助任务是为了骗过身份和姿势判别器的。引入新的同姿势损失项，以进一步促进对与身份相关而与姿势无关的视觉特征的学习。

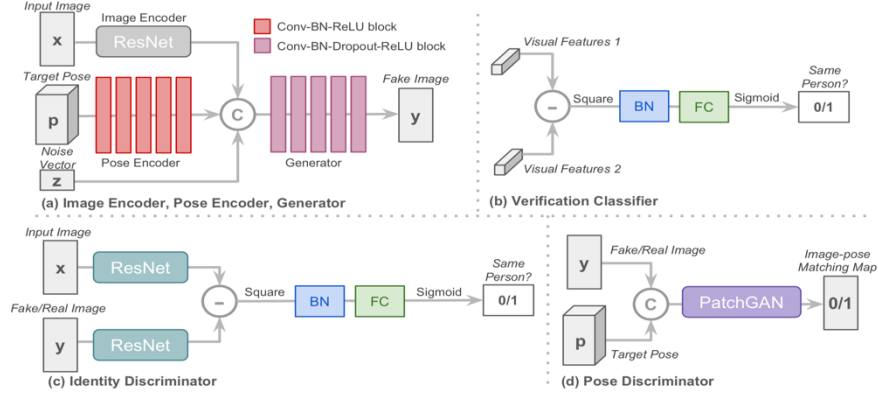


Figure 3: Network structures of (a) the generator  $G$  and the image encoder  $E$ , (b) the verification classifier  $V$ , (c) the identity discriminator  $D_{id}$ , (d) the pose discriminator  $D_{pd}$ .

FD-GAN 网络总体框架如上图所示。框架包括图像编码器  $E$ , 图像生成器  $G$ , 身份验证分类器  $V$  和两个鉴别器 (身份鉴别器  $D_{id}$  和姿态鉴别器  $D_{pd}$ )。具体流程如下:

对于每个网络分支:

- 1、input (输入): 人物图像和目标姿势 landmark map (地标图)。
- 2、图像编码器 ( $E$ ) 首先将输入的人图像变换为特征表示, 身份验证分类器 ( $V$ ) 用于监督人 reID 的特征学习。仅使用验证分类器 ( $V$ ) 使得编码器不仅编码人物身份信息而且编码人物姿势信息, 这使得学习的特征对人物姿势变化敏感。
- 3、添加图像生成器  $G$ , 以编码器的特征和目标姿势图为条件, 如果学到的特征是 Pose-related 和 identity-related, 就可以用来生成同一个不同姿态的图片。
- 4、身份判别器  $D_{id}$  和姿势判别器  $D_{pd}$  被集成以规范图像生成过程。  $D_{id}$  和  $D_{pd}$  都是条件判别器, 它根据输入标识或姿势对输入图像是真实的还是假的进行分类。FD-GAN 网络促使图像生成器与图像编码器一起用伪造的图像欺骗鉴别器。

#### 4.1 Image encoder and image generator

图像编码器 ( $E$ ) 和图像生成器 ( $G$ ) 如上图 3 所示,  $E$  利用 ResNet-50 作为 backbone 网络将输入图像编码成 2048 维特征向量。  $G$  将编码的人物特征和目标姿势图作为输入, 生成由目标姿势指定的同一人的另一图像。目标姿势图是一个 18 通道的 map, 其中每个通道表示一个姿态点的位置, 并且可将其被转换为类似高斯的热图。通过 5 块 Convolution-BN-ReLU 子网络编码获得 128 维姿势特征向量。将视觉特征, 目标姿态特征和从标准高斯分布采样附加的 256 维噪声矢量连接并输入到一系列 5 convolution-BN-dropout-ReLU 上采样块中, 以输出所生成的图像。

## 4.2 Identity verification classifier

分类器的结构如上图b所示，分类器将两个人图像的视觉特征作为输入，并通过 element-wise subtraction(逐元素减法)， element-wise square (元素平方)， batch normalization (批量归一化层)， 完全连接层和 最后是一个sigmoid非线性函数，用于输出输入图像对属于同一个人的概率。该分类器用二元交叉熵损失训练。二元交叉熵损失如下：

$$\mathcal{L}_v = -C \log d(x_1, x_2) - (1 - C)(1 - \log d(x_1, x_2)), \quad (1)$$

其中  $x_1, x_2$  是两个输入图像,  $d(x_1, x_2)$ 表示网络输出的置信度,  $C$  是 ground truth,  $C=1$  表示是同一个人。

## 4.3 Image generation with identity and pose discriminators

**Identity discriminator  $D_{id}$ :**身份鉴别器训练以区分生成的人图像和同一分支的输入人图像是否属于同一个人。 $D_{id}$ 的对抗性损失定义为：

$$\mathcal{L}_{id} = \max_{D_{id}} \sum_{k=1}^2 \left( \mathbb{E}_{y'_k \in \mathcal{Y}} [\log D_{id}(x_k, y'_k)] + \mathbb{E}_{y_k \in \mathcal{Z}} [\log(1 - D_{id}(x_k, y_k))] \right) \quad (2)$$

其中 $y$ 和 $z$ 分别表示图像的真实数据分布和通过 $G$ 生成的数据分布。

**Pose discriminator  $D_{pd}$ :** 姿势鉴别器  $D_{pd}$  用来区分所生成的人物图像  $y^k$  (是否与给定的目标姿态  $p$  匹配。如上图 d 所示，它采用 PatchGAN 结构。输入图像和姿势图 (在类似高斯的 heat-map 变换之后) 首先沿着信道维度连接，然后由 4 个卷积-ReLU blocks 和一个 sigmoid 函数来得到一个 0 到 1 之间的姿态匹配 map，此 map 中的每个位置表示输入图片和每个 Landmark 的匹配度。图片生成器  $G$  尝试欺骗姿态判别器  $D_{pd}$ ,  $D_{pd}$  对抗损失为：

$$\mathcal{L}_{pd} = \max_{D_{pd}} \sum_{k=1}^2 \left( \mathbb{E}_{y'_k \in \mathcal{Y}} [\log D_{pd}([p, y'_k])] + \mathbb{E}_{y_k \in \mathcal{Z}} [\log(1 - D_{pd}([p, y_k]))] \right), \quad (3)$$

作者观察到姿势鉴别器  $D_{pd}$  可能过拟合姿态信息，即  $D_{pd}$  可能会记住特定姿势和人物外观之间的对应关系，因为每个图像的姿势通常是唯一的。因此，作者提出了一种在线姿势图增强 (pose map augmentation scheme) 方案。在训练期间，对于每个 pose landmark，在一些特定范围内使用随机高斯带宽获得其 1 通道 Gaussian-like heat-map。通过这种方式，我们可以为相同的姿势创建许多姿势贴图，并减轻姿势过度拟合问题。

**Reconstruction loss :**  $G$  不仅是用来迷惑判别器的，也是为了生成图片。

单独的判别器不能保证产生 human-perceivable(人类可感知)的图像,因此使用一个重构误差来最小化生成的图片  $y^s$  与真实图片  $y^t$  之间的误差：

$$\mathcal{L}_r = \sum_{k=1}^2 \frac{1}{mn} \|y_k - y'_k\|_1,$$

## 五、实验

数据集：Market1501, DukeMTMC-reID, CUHK03

实验环境

- Python 3.6.3
- Pytorch 0.3.1

将环境配置好，下载作者提供的预训练模型进行实验(由于最近服务器挺多人使用所以使用作者提供训练好的模型)。训练作者提出的框架training分为三个阶段，第一个阶段**Siamese baseline model**仅包括图像编码器E和身份验证分类器V在数据集上进行预训练。然后使用预先训练的网络权重来初始化阶段II中的E，V和身份判别器。第二个阶段E和V的参数是固定的。然后主要是训练G，身份判别器D<sub>id</sub>，并将判别器D<sub>pd</sub>与方程式中的总体目标L构成。

跑代码的过程中出现多次下图问题

```
RuntimeError: cuda runtime error (2) : out of memory at /pytorch/torch/lib/THC/generic/THCStorage.cu:58
zhangjm@scw4550:/raid/xzp/FD-GAN/FD-GAN$
```

后来检测了实时的内存，当训练到第二阶段的时候，内存占用就超过8.5个G是因为内存占用太大。所以只执行到第二阶段，效果挺不错的，第二个阶段的结果就已经接近作者给出的结果了。给出第1，2阶段的结果：

First stage evaluation:

Mean AP: 69.7%

CMC Scores market1501

top-1 86.8%

top-5 95.4%

top-10 97.4%

Second stage evaluation:

Mean AP: 72.7%

CMC Scores market1501

top-1 88.2%

top-5 95.9%

top-10 97.6%

作者给出的结果：

Table 1: Component analysis of the proposed FD-GAN on Market-1501 [5] and DukeMTMC-reID [7] datasets in terms of top-1 accuracy (%) and mAP (%)

Networks	not share $E$	Components					Market-1501[5]		DukeMTMC-reID[7]	
		$\mathcal{L}_{sp}$	$\mathcal{L}_v$	$\mathcal{L}_{pd}$	$\mathcal{L}_{id}$	pose map aug.	mAP	top-1	mAP	top-1
baseline (single)	n/a	n/a	n/a	n/a	n/a	n/a	59.8	81.4	40.7	62.5
baseline (Siamese)	n/a	n/a	✓	n/a	n/a	n/a	72.5	88.2	61.3	78.2
Siamese DR-GAN[21]	×	×	✓	✓	✓	✓	73.2	86.7	60.2	76.9
FD-GAN (share $E$ )	×	✓	✓	✓	✓	✓	73.5	86.8	-	-
FD-GAN (no sp.)	✓	×	✓	✓	✓	✓	75.8	88.9	-	-
FD-GAN (no veri.)	✓	✓	×	✓	✓	✓	75.7	89.5	62.6	78.8
FD-GAN (no sp. & no veri.)	✓	×	×	✓	✓	✓	74.4	88.7	62.4	78.6
FD-GAN (no $D_{pd}$ )	✓	✓	✓	×	✓	✓	73.0	88.0	-	-
FD-GAN (no $D_{id}$ )	✓	✓	✓	✓	×	✓	72.8	89.2	-	-
FD-GAN (no $D_{id}$ & $D_{pd}$ )	✓	✓	✓	×	×	✓	71.6	84.6	-	-
FD-GAN (no pose aug.)	✓	✓	✓	✓	✓	×	77.2	89.5	63.9	79.5
FD-GAN	✓	✓	✓	✓	✓	✓	<b>77.7</b>	<b>90.5</b>	<b>64.5</b>	<b>80.0</b>

## 六、总结

查询了最近发表的行人重识别相关的论文，结合 GAN 网络来做的目前看到的有这篇商汤和中科大提出的，另一篇是来自 ECCV2018 的，也是结合 GAN 网络来做的。发表的时间都比较新，18 年下半年发布的。作者在文章中提出的 FD-GAN 网络，加入了人体姿态的检测，编码器 E 用的是 ResNet 网络，代码都是模块化的。最近看了几篇基于 ResNet 网络进行改进的论文比如（DetNet 网络），其实觉得可以将 image Encoder 换成最新的目标检测网络来做，理论上应该可以提高精确率，以及 GAN 网络，作者采用的是原始的 GAN，目前 GAN

网络多种多样，结合当下最新的网络，结果应该会好一点（理论上）。实际上还没有测试。最近在自学 pytorch,尽量早点改改别人的网络试试，看看效果怎么样。