

源代码来了 | 英伟达开源行人生成/重识别代码

原创

Layumi1993 于 2019-07-05 09:46:02 发布 1848 收藏 16

分类专栏: [论文阅读](#) [行人重识别](#) 文章标签: [行人重识别](#) [行人再识别](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/Layumi1993/article/details/94647875>

版权



[论文阅读](#) 同时被 2 个专栏收录

27 篇文章 0 订阅

订阅专栏



[行人重识别](#)

8 篇文章 2 订阅

订阅专栏

前几天英伟达开源了DG-Net的源码。让我们来回顾一下这篇CVPR19 Oral的论文。

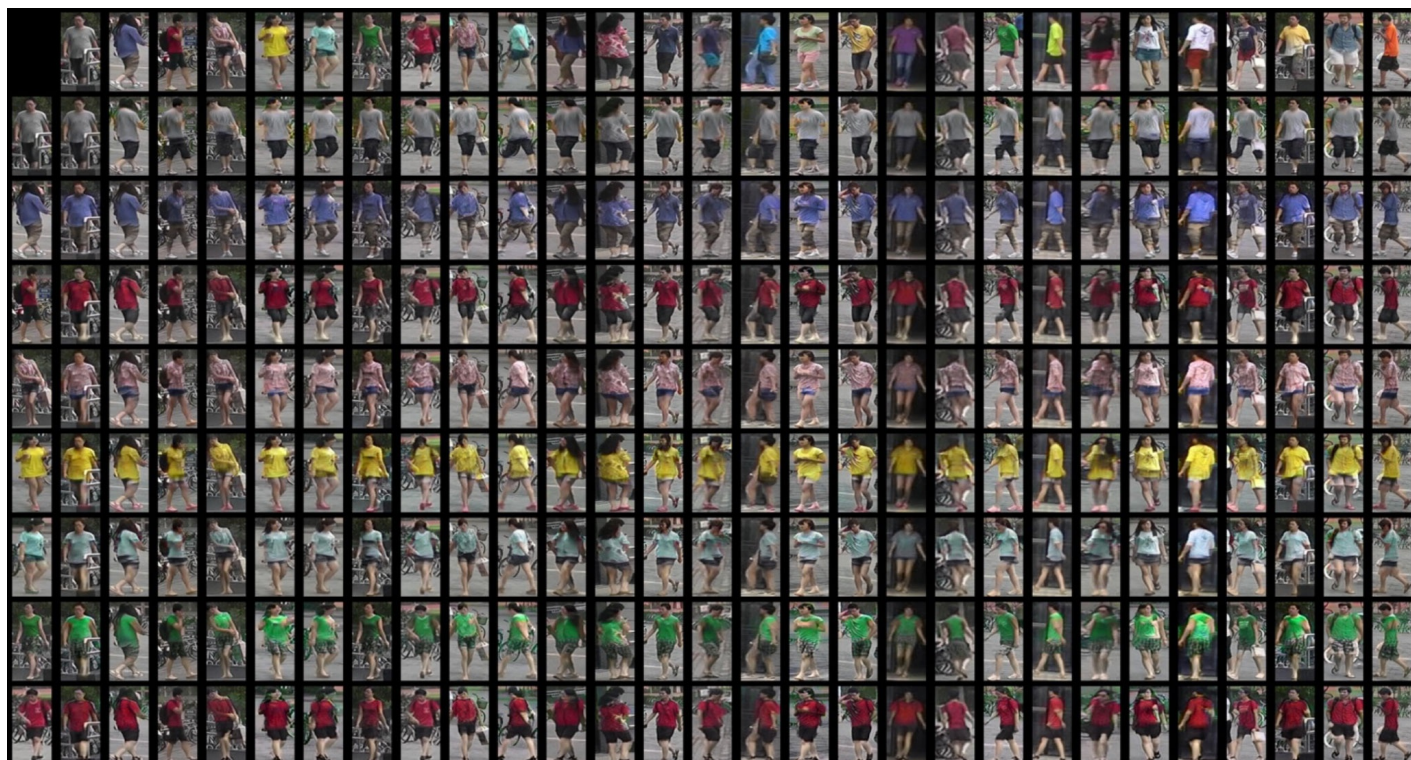
论文是英伟达(NVIDIA), 悉尼科技大学(UTS), 澳大利亚国立大学(ANU)的研究人员 在CVPR19上口头报告的文章《Joint Discriminative and Generative Learning for Person Re-identification》。深度学习模型训练时往往需要大量的标注数据,但收集和标注大量的数据往往比较困难。作者在行人重识别这个任务上探索了 利用生成数据来辅助训练的方法。通过生成高质量的行人图像, 将其与行人重识别模型融合, 同时提升行人生成的质量和行人重识别的精度。

论文链接: <https://arxiv.org/abs/1904.07223>

B 站视频: <https://www.bilibili.com/video/av51439240/>

腾讯视频: <https://v.qq.com/x/page/t0867x53ady.html>

代码地址: <https://github.com/NVlabs/DG-Net>



代码运行效果：(训练100000次迭代)

```
Epoch: 61 Iteration: 00099991/00100000 DLoss: 2.9677 Reg: 0.0337
L_total: 10.3704, L_gan: 3.0538, Lx: 0.8304, Lxp: 1.0318, Lrecycle:1.0755, Lf: 0.7060, Ls: 0.6756, Recon-id: 0.6801, id: 0.3227, pid:0.0654, teacher: 1.9291
Elapsed time in update: 1.820401
Epoch: 61 Iteration: 00099992/00100000 DLoss: 2.9721 Reg: 0.0343
L_total: 9.9531, L_gan: 3.1345, Lx: 0.8159, Lxp: 1.1240, Lrecycle:0.9714, Lf: 0.6718, Ls: 0.6555, Recon-id: 0.3889, id: 0.1074, pid:0.2230, teacher: 1.8607
Elapsed time in update: 1.853781
Epoch: 61 Iteration: 00099993/00100000 DLoss: 2.9551 Reg: 0.0339
L_total: 10.6389, L_gan: 3.1543, Lx: 1.0198, Lxp: 1.2170, Lrecycle:1.0374, Lf: 0.6618, Ls: 0.6586, Recon-id: 0.3935, id: 0.2957, pid:0.0776, teacher: 2.1231
Elapsed time in update: 1.829953
Epoch: 61 Iteration: 00099994/00100000 DLoss: 2.9605 Reg: 0.0337
L_total: 9.4584, L_gan: 3.2015, Lx: 0.8729, Lxp: 1.0851, Lrecycle:0.8988, Lf: 0.6408, Ls: 0.6352, Recon-id: 0.4898, id: 0.0616, pid:0.0889, teacher: 1.4836
Elapsed time in update: 1.852776
Epoch: 61 Iteration: 00099995/00100000 DLoss: 2.9504 Reg: 0.0347
L_total: 9.7752, L_gan: 3.1742, Lx: 0.7956, Lxp: 1.0213, Lrecycle:1.0240, Lf: 0.6294, Ls: 0.6798, Recon-id: 0.4898, id: 0.1992, pid:0.1259, teacher: 1.6361
Elapsed time in update: 1.839327
Epoch: 61 Iteration: 00099996/00100000 DLoss: 2.9520 Reg: 0.0331
L_total: 10.2564, L_gan: 3.1169, Lx: 0.8423, Lxp: 0.9867, Lrecycle:0.8554, Lf: 0.6379, Ls: 0.5783, Recon-id: 0.7797, id: 0.2403, pid:0.0540, teacher: 2.1649
Elapsed time in update: 1.797634
Epoch: 61 Iteration: 00099997/00100000 DLoss: 2.9473 Reg: 0.0320
L_total: 9.9534, L_gan: 3.1303, Lx: 0.9220, Lxp: 1.1541, Lrecycle:0.9493, Lf: 0.6458, Ls: 0.6457, Recon-id: 0.4421, id: 0.1570, pid:0.3566, teacher: 1.5505
Elapsed time in update: 1.849897
Epoch: 61 Iteration: 00099998/00100000 DLoss: 2.9173 Reg: 0.0312
L_total: 10.3477, L_gan: 3.1513, Lx: 0.7638, Lxp: 0.9313, Lrecycle:0.8184, Lf: 0.7383, Ls: 0.5706, Recon-id: 0.8993, id: 0.2972, pid:0.1761, teacher: 2.0014
Elapsed time in update: 1.865791
Epoch: 61 Iteration: 00099999/00100000 DLoss: 2.9056 Reg: 0.0305
L_total: 9.3586, L_gan: 3.1283, Lx: 0.9088, Lxp: 1.0866, Lrecycle:0.9143, Lf: 0.6582, Ls: 0.5940, Recon-id: 0.3780, id: 0.3092, pid:0.0996, teacher: 1.2816
Elapsed time in update: 1.810548
Epoch: 61 Iteration: 00100000/00100000 Finish training
```

<https://blog.csdn.net/Layumi1993>

开发环境：

- Python 3.6
- GPU Memory >= 15G 如果使用fp32精度
- GPU Memory >= 10G 如果使用fp16精度，可以节省一部分显存
- NumPy
- PyTorch 1.0+
- [Optional] APEX (使用fp16要安装)

数据集下载地址：

采用的是Market-1501 数据集 http://www.liangzheng.com.cn/Project/project_reid.html

训练好的模型下载：

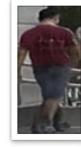
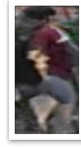
- 百度网盘: <https://pan.baidu.com/s/1503831XfW0y4g3PHir91yw> password: rqvf
- GoogleDrive: <https://drive.google.com/open?id=1IL18FZX1uZMWKzaZOuPe3luAdfUYyJKH>

测试效果如下：

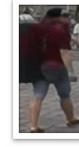
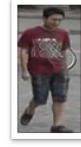
- 行人重识别的精度:

```
torch.Size([3368, 1024])
Alpha:0.00 Rank@1:0.9400 Rank@5:0.9777 Rank@10:0.9855 mAP:0.8439
Alpha:0.50 Rank@1:0.9477 Rank@5:0.9798 Rank@10:0.9878 mAP:0.8609
Alpha:-1.00 Rank@1:0.9169 Rank@5:0.9709 Rank@10:0.9771 mAP:0.7539
multi Rank@1:0.9608 Rank@5:0.9860 Rank@10:0.9923 mAP:0.9044
```

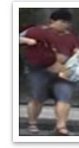
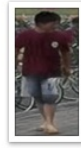
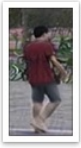
- 生成的行人图像:



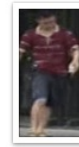
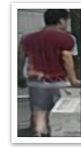
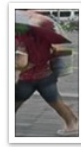
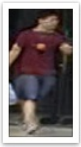
046_000_gan000 049_000_gan000 082_000_gan000 124_000_gan000 135_000_gan000
 2_c3s1...6_01.jpg 2_c2s1...21_01.jpg 2_c3s1...2_02.jpg 2_c3s1...2_02.jpg 2_c2s1...51_01.jpg



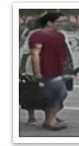
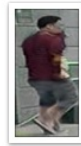
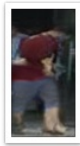
136_000_gan000 177_000_gan0002 259_000_gan000 295_000_gan000 296_000_gan000
 2_c2s1_1...1_03.jpg _c5s1_0...6_02.jpg 2_c5s1_1...04.jpg 2_c1s2...41_01.jpg 2_c2s1...01_01.jpg



299_000_gan000 302_000_gan000 323_000_gan000 325_000_gan000 343_000_gan000
 2_c5s1...6_01.jpg 2_c3s1_1...8_01.jpg 2_c3s1_1...02.jpg 2_c5s1...1_03.jpg 2_c1s1...01_01.jpg



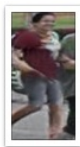
345_000_gan000 353_000_gan000 369_000_gan000 397_000_gan0002 405_000_gan000
 2_c6s1...1_02.jpg 2_c3s1_1...03.jpg 2_c2s1_1...1_01.jpg _c2s1_0...6_01.jpg 2_c3s1_1...05.jpg



429_000_gan000 438_000_gan000 457_000_gan000 497_000_gan000 498_000_gan000
 2_c6s1...6_02.jpg 2_c5s1_1...08.jpg 2_c2s1...1_02.jpg 2_c5s1...51_01.jpg 2_c1s1...51_01.jpg



507_000_gan0002 529_000_gan000 541_000_gan000 542_000_gan000 560_000_gan000
 _c3s1_0...01_01.jpg 2_c2s1...76_01.jpg 2_c1s1...51_03.jpg 2_c3s1_1...04.jpg 2_c5s1...76_01.jpg



576_000_gan000 578_000_gan000 582_000_gan000 597_000_gan0002 599_000_gan000
 2_c3s1...01_01.jpg 2_c1s1...76_01.jpg 2_c2s1...01_01.jpg _c1s1_0...6_02.jpg 2_c5s1...3_02.jpg

<https://blog.csdn.net/Layumi1993>

训练的命令很简单：

选项已经内置到yaml文件中，若按照全精度fp32来运行，约占用15G显存。

```
python train.py --config configs/latest.yaml
```

若使用半精度训练，则只使用约10G显存。

```
python train.py --config configs/latest-fp16.yaml
```

训练的log可使用tensorboard 来查看

```
tensorboard --logdir logs/latest
```

作者简介

本文的第一作者郑哲东是悉尼科技大学计算机科学学院的博士生，预计2021年6月毕业。该论文是其在英伟达实习期间的成果。

郑哲东目前已经发表8篇论文。其中一篇为ICCV17 spotlight，被引用超过了300次。首次提出了利用GAN生成的图像辅助行人重识别的特征学习。一篇TOMM期刊论文被Web of Science选为2018年高被引论文，被引用超过200次。同时，他还为社区贡献了行人重识别问题的基准代码，在Github上star超过了1000次，被广泛采用。

另外，论文的其他作者包括英伟达研究院的视频领域专家 - 杨晓东、人脸领域专家禹之鼎（Sphere Face, LargeMargin作者）、行人重识别专家郑良博士，郑哲东的导师杨易教授（今年有三篇 CVPR oral 中稿）、和英伟达研究院的副总裁 Jan Kautz。