Joint Discriminative and Generative Learning for Person Re-identification
SUPPLEMENTARY MATERIAL

Zhedong Zheng[1,2*] Xiaodong Yang[1] Zhiding Yu[1]
Liang Zheng[3] Yi Yang[2] Jan Kautz[1]
[1]NVIDIA [2]CAI, University of Technology Sydney [3]Australian National University

A. Network Architectures

Our proposed DG-Net consists of the appearance encoder E_a, structure encoder E_s, decoder G, and discriminator D. As described in the paper that E_a is modified from ResNet50, we now introduce the architecture details of E_s, G, and D. Following the common practice in GANs, we mainly adopt convolutional layers and residual blocks [3] to construct them.

Table 6 shows the architecture of E_s. After each convolutional layer, we apply the instance normalization layer [9] and LReLU (negative slope set to 0.2). We also add the optional atrous spatial pyramid pooling (ASPP) [2], which contains dilated convolutions and can be used to exploit multi-scale features. Table 7 demonstrates the architecture of decoder G, which involves several residual blocks followed by upsampling and convolutional layers. Similar to [4], we insert the adaptive instance normalization (AdaIN) layer in every residual block to integrate the appearance code from E_a as the dynamically generated weight and bias parameters of AdaIN. We employ the multi-scale PatchGAN [13] as the descriminator D. Given an input image of 256×128, we resize the image to the three different scales: 256×128, 128×64, 64×32 before feeding them into the discriminator. LReLU (negative slope set to 0.2) is applied after each convolutional layer. We present the architecture of D in Table 8.

Layer	Parameters	Output Size
Input	-	$1 \times 256 \times 128$
Conv1	$[3\times3, 16]$	$16 \times 128 \times 64$
Conv2	$[3\times3, 32]$	$32 \times 128 \times 64$
Conv3	$[3\times3, 32]$	$32 \times 128 \times 64$
Conv4	$[3\times3, 64]$	$64 \times 64 \times 32$
ResBlocks	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 4$	$64 \times 64 \times 32$
ASPP	$[1\times1, 32]$ $\begin{bmatrix} 1\times1, 32 \\ 3\times3, 32 \end{bmatrix} \times 3$	$128 \times 64 \times 32$
Conv5	$[1\times1, 128]$	$128 \times 64 \times 32$

Table 6: Architecture of the structure encoder E_s.

Layer	Parameters	Output Size
Input	-	$128 \times 64 \times 32$
ResBlocks	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 4$	$128 \times 64 \times 32$
Upsample	-	$128 \times 128 \times 64$
Conv1	$[5\times5, 64]$	$64 \times 128 \times 64$
Upsample	-	$64 \times 256 \times 128$
Conv2	$[5\times5, 32]$	$32 \times 256 \times 128$
Conv3	$[3\times3, 32]$	$32 \times 256 \times 128$
Conv4	$[3\times3, 32]$	$32 \times 256 \times 128$
Conv5	$[1\times1, 3]$	$3 \times 256 \times 128$

Table 7: Architecture of the decoder G.

B. More Discriminative Evaluations

In order to have a more thorough evaluation of our approach, we further evaluate the performance of DG-Net on a relatively small dataset. So we generalize our approach to CUHK03-NP [12], which contains much fewer images (9.6 training images per person on average) compared to Market-1501 [11], DukeMTMC-reID [7] and MSMT17 [10]. As compared in Table 9, DG-Net achieves 65.6% Rank@1 and 61.1% mAP.

*Work done during an internship at NVIDIA Research.

C. Appearance and Structure Codes

Since we cannot quantitatively justify the attributes of appearance/structure codes, Table 1 in the paper is used to qualitatively give an intuition. Our design of E_s (a shallow network) makes the structure space primarily preserve the structural information, such as position and geometry of

Layer	Parameters	Output Size
Input	-	$3 \times 256 \times 128$
Conv1	$[\ 1{\times}1, 32\ ]$	$32 \times 256 \times 128$
Conv2	$[\ 3{\times}3, 32\ ]$	$32 \times 256 \times 128$
Conv3	$[\ 3{\times}3, 32\ ]$	$32 \times 128 \times 64$
Conv4	$[\ 3{\times}3, 32\ ]$	$32 \times 128 \times 64$
Conv5	$[\ 3{\times}3, 64\ ]$	$64 \times 64 \times 32$
ResBlocks	$\begin{bmatrix} 3{\times}3, 64 \\ 3{\times}3, 64 \end{bmatrix} \times 4$	$64 \times 64 \times 32$
Conv6	$[\ 1{\times}1, 1\ ]$	$1 \times 64 \times 32$

Table 8: Architecture of the discriminator D.

humans and objects. Thus, the structure code is mainly used to hold the low-level positional and geometric information, such as pose and background that are non-id-related, to facilitate image synthesis. On the other hand, certain structure cues, such as bag/hair/body outline, are clearly id-related and are better to be captured by the discriminative module. However, softmax loss is generally too "lazy" to be able to capture useful structure information besides appearance features, therefore, the goal of fine-grained feature mining upon the appearance code promotes mining the id-related semantics out of structure cues, also guarantees the complementary nature between primary and fine-grained features.

D. Interpolate between Structure Codes

Figure 5 in the paper shows the examples of synthesized images by linear interpolation between two appearance codes. This qualitatively validates the continuity in the appearance space. As a complementary study, here we generate the images by linearly interpolating between two structure codes while keeping the appearance codes intact in Figure 9. This demonstrates the exact opposite setting to Figure 5. As expected, most images (both foreground and background) look not realistic. Our hypothesis is that the structure codes are extracted by a shallow network and contain the positional and geometric information of inputs. So the interpolation between the low-level features is not able to preserve semantic smoothness or consistency.

Methods	Rank@1	mAP
HA-CNN [5]	41.7%	38.6%
PT [6]	41.6%	38.7%
MLFN [1]	52.8%	47.8%
PCB [8]	61.3%	54.2%
PCB + RPP [8]	63.7%	57.5%
Ours	**65.6%**	**61.1%**

Table 9: Comparison with the state-of-the-art results on the CUHK03-NP dataset.

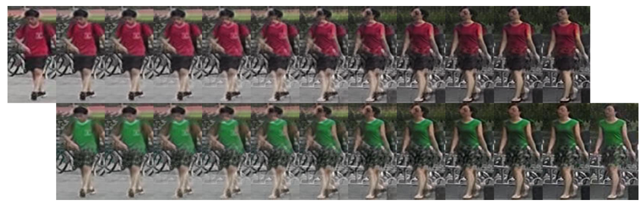

Figure 9: Example of image generation by linear interpolation of two structure codes. We fix the appearance code in each row. This figure is best viewed when zoom in and compare with Figure 5.

References

[1] Xiaobin Chang, Timothy Hospedales, and Tao Xiang. Multi-level factorisation net for person re-identification. In *CVPR*, 2018. 2

[2] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv:1706.05587*, 2017. 1

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[4] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. *ECCV*, 2018. 1

[5] Wei Li, Xiatian Zhu, and Shaogang Gong. Harmonious attention network for person re-identification. In *CVPR*, 2018. 2

[6] Jinxian Liu, Bingbing Ni, Yichao Yan, Peng Zhou, Shuo Cheng, and Jianguo Hu. Pose transferrable person re-identification. In *CVPR*, 2018. 2

[7] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCVW*, 2016. 1

[8] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling. In *ECCV*, 2018. 2

[9] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv:1607.08022*, 2016. 1

[10] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. In *CVPR*, 2018. 1

[11] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. 1

[12] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *CVPR*, 2017. 1

[13] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *ICCV*, 2017. 1