生成对抗网络实现多领域图像变换

李洁厅
21821188
计算机学院，浙江大学
lijieting@zju.edu.cn

摘要（Abstract）

我要推荐一篇生成对抗网络实现多领域图像变换的论文。关于图像之间的风格迁移和翻译是近年来最受关注的人工智能研究方向之一，这个任务在具有趣味性的同时也是很有挑战的。相关的研究成果也层出不穷，有的甚至引起了全世界的广泛讨论。香港科技大学、新泽西大学和韩国大学等机构在CVPR2018 上联合发表的一篇论文。这篇论文提出了在同一个模型中进行多个图像领域之间的风格转换的对抗生成方法StarGan，突破了传统的只能在两个图像领域转换的局限性。

1. Introduction

图像到图像转换（image-to-image translation）这个任务是指改变给定图像的某一方面，例如，将人的面部表情从微笑改变为皱眉。在引入生成对抗网络（GAN）之后，这项任务有了显着的改进，包括可以改变头发颜色，改变风景图像的季节等等。

给定来自两个不同领域的训练数据，这些模型将学习如何将图像从一个域转换到另一个域。本文将属性（attribute）定义为图像中固有的有意义的特征，例如头发颜色，性别或年龄等，并且将属性值（attribute value）表示为属性的一个特定值，例如头发颜色的属性值可以是黑色 / 金色 / 棕色，性别的属性值是男性 / 女性。本文进一步将域（domain）表示为共享相同属性值的一组图像。例如，女性的图像可以代表一个 domain，男性的图像代表另一个 domain。

一些图像数据集带有多个标签属性。例如，CelebA 数据集包含 40 个与头发颜色、性别和年龄等面部特征相关的标签，RaFD 数据集有 8 个面部表情标签，如“高兴”、“愤怒”、“悲伤”等。这些设置使本文能够执行更有趣的任务，即多域图像到图像转换（multi-domain image-to-image translation），即根据来自多个域的属性改变图像。

在图 1 中，前 5 列显示了一个 CelebA 的图像是如何根据 4 个域（“金发”、“性别”、“年龄”和“白皮肤”）进行转换。本文可以进一步扩展到训练来自不同数据集的多个域，例如联合训练 CelebA 和 RaFD 图像，使用在 RaFD 上训练的特征来改变 CelebA 图像的面部表情，如图 1 最右边的列所示。

然而，现有模型在这种多域图像转换任务中既效率低，效果也不好。它们的低效性是因为在学习 k 个域之间的所有映射时，必须训练 k（k-1）个生成器。图 2 说明了如何训练 12 个不同的生成器网络以在 4 个不同的域中转换图像。

为了解决这类问题，本文提出了 StarGAN，这是一个能够学习多个域之间映射的生成对抗网络。如图 2(b) 所示，本文的模型接受多个域的训练数据，仅使用一个生成器就可以学习所有可用域之间的映射。

这个想法很简单。本文的模型不是学习固定的转换（例如，将黑头发变成金色头发），而是将图像和域信息作为输入，学习将输入的图像灵活地转换为相应的域。本文使用一个标签来表示域信息。在训练过程中，本文随机生成一个目标域标签，并训练模型将输入图像转换为目标域。这样，本文可以控制域标签并在测试阶段将图像转换为任何想要的域。

本文还介绍了一种简单但有效的方法，通过在域标签中添加一个掩码向量（mask vector）来实现不同数据集域之间的联合训练。本文提出的方法可以确保模型忽略未知的标签，并关注特定数据集提供的标签。这样，我模型就可以很好地完成任务，比如利用从 RaFD 中学到的特征合成 CelebA 图像的面部表情，如图 1 最右边的列所示。据本文所知，这是第一个在不同的数据集上成功地完成多域图像转换的工作。

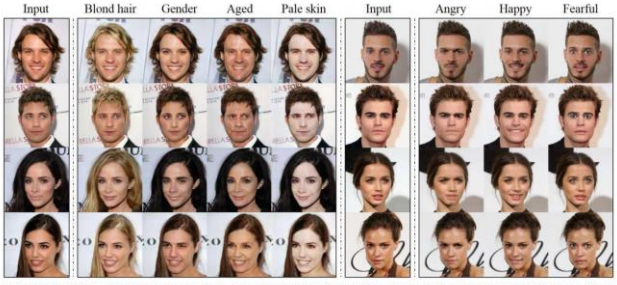

Figure 1. Multi-domain image-to-image translation results on the CelebA dataset via transferring knowledge learned from the RaFD dataset. The first and sixth columns show input images while the remaining columns are images generated by StarGAN. Note that the images are generated by a single generator network, and facial expression labels such as angry, happy, and fearful are from RaFD, not CelebA.

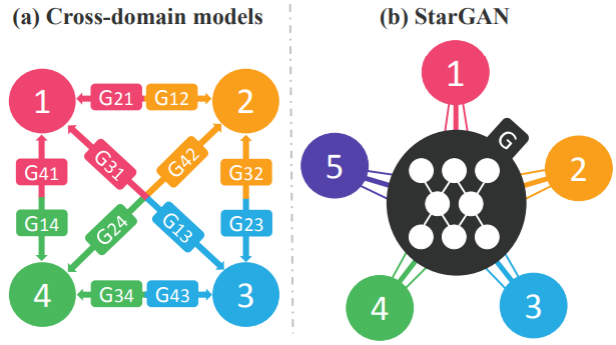

Figure 2. Comparison between cross-domain models and our proposed model, StarGAN. (a) To handle multiple domains, cross-domain models should be built for every pair of image domains. (b) StarGAN is capable of learning mappings among multiple domains using a single generator. The figure represents a star topology connecting multi-domains.

总结而言，这个研究的贡献如下：

- 提出 StarGAN，这是一个新的生成对抗网络，只使用一个生成器和一个鉴别器来学习多个域之间的映射，能有效地利用所有域的图像进行训练。
- 演示了如何通过使用 mask vector 来学习多个数据集之间的多域图像转换，使 StarGAN 能够控制所有可用的域标签。
- 使用 StarGAN 在面部属性转换和面部表情合成任务提供了定性和定量的结果，优于 baseline 模型

原则上，文中提出的模型可以应用于任何其他类型的域之间的转换问题，例如，风格转换（style transfer），这是未来的工作方向之一。

2. Detailed description of the algorithm

总得来看，StarGAN包括两个模块，一个鉴别器D和一个生成器G。图 3（a）D学习如何区分真实图像和伪造图像，并将真实图像分类到相应领域。（b）G同时输入图像和目标域的标签并生成假图像，在输入时目标域标签被复制并与输入图像拼接在一块。（c）G尝试从给定原始域标签的假图像重建原始图像。（d）G试图生成与真实图像不可区分的图像同时又很容易被目标域D所区分出来。

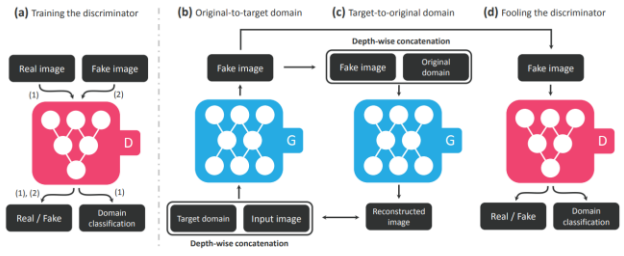

Figure 3. Overview of StarGAN, consisting of two modules, a discriminator D and a generator G. **(a)** D learns to distinguish between real and fake images and classify the real images to its corresponding domain. **(b)** G takes in as input both the image and target domain label and generates an fake image. The target domain label is spatially replicated and concatenated with the input image. **(c)** G tries to reconstruct the original image from the fake image given the original domain label. **(d)** G tries to generate images indistinguishable from real images and classifiable as target domain by D.

2.1. Loss

Adversarial损失函数：为了让生成的图片与真实图像难以区分，采用对抗性损失函数，与一般的GAN区别不大。

$$L_{adv} = E_x[logD_{src}(x)] + E_{x,c}[log(1 - D_{src}(G(x,c)))]$$

其中x为输入图像，c为域标签信息，$D_{src}(x)$为D所给出的源概率分布。

Domain Classification 损失函数：

对于真实图像：$L_{cls,r} = E_{x,c'}[-logD_{cls}(c'|x)]$,D最小化损失函数，将真实图像分类到对应的域。

对于生成图片：$L_{cls,f} = E_{x,c}[-logD_{cls}(c|G(x,c)]$,G最小化损失函数，将生成图像分类到目标域。

其中$D_{cls}(c|x)$ 表示由D给出的域标签概率分布。

Reconstruction损失函数：虽然通过如上损失函数可以生成目标域图像，但是无法保证生成图像保留其输入图像的内容。因此的使用了一个循环一致性损失函数应用到生成器中。将生成图像G(x,c)和标签c作为输入，重构源图像x。损失函数定义为：

$$L_{rec} = E_{x,c,c'}[||x - G(G(x,c),c')||_1]$$

StarGAN总体损失函数：

$$L_D = -L_{adv} + \lambda_{cls}L_{cls,r}$$

$$L_G = L_{adv} + \lambda_{cls}L_{cls,f} + \lambda_{rec}L_{rec}$$

其中控制系数λ_{cls}和λ_{rec}在本文中分别取 1 和 10。

2.2. Training with Multiple Datasets

对于不同的数据集来说，每一个数据集只能知道全体标注的一部分。如celebA并不知道RaFD中关于表情的“愤怒”“开心”等标签。但是在计算损失函数时，本文需要知道全部的标签信息。因此采用mask vector来解决这个问题。

mask vector：构建了一个n维的one-hot向量m，其中n是数据集的数量（在论文中使用了两个数据集，故n=2），对于未知的数据集标签，统统设置为 0 值。另外，设置了一个统一的标签作为一个矢量：

$$\vec{c} = [c_1, c_2, ..., c_n, m]$$

ci表示第i个数据集的标签向量，对于未知的数据集标签，统统设置为 0 向量。

训练时，生成器G将忽略掉传入的$\vec{c}$ 向量中的 0 向量,犹如在训练单数据集一样，而判别器D的辅助分类器则生成所有数据集的全部标签概率，但只和已知的真实标签做loss计算。训练模采用Wasserstein GAN的Adversarial损失函数：

$$L_{adv} = E_x[D_{src}(x)] - E_{x,c}[D_{src}(G(x,c))] - \lambda_{gp}E_{\hat{x}}[(||\nabla_{\hat{x}} D_{src}(\hat{x})||_2 - 1)^2]$$

其中$\hat{x}$是均匀采样的真实图像和生成图像。实验采用的$\lambda_{gp} = 10$。

3. Experimental results

在明星脸上的面部属性迁移：这些图片是由StarGAN在CelebA 数据集上训练后生成的。可以发现，本文的模型与跨域模型相比，在测试数据上提供了更高的变换结果的视觉质量。一个可能的原因是StarGAN通过多任务学习框架的正则化效果。换句话说，本文训练模型根据目标域的标签灵活地变换图像，而不是训练模型执行固定的变换(例如，从棕色到金色的头发)，这很容易过度拟合。这使得本文的模型能够学习到可靠的特征，适用于具有不同面部属性的多个图像域。

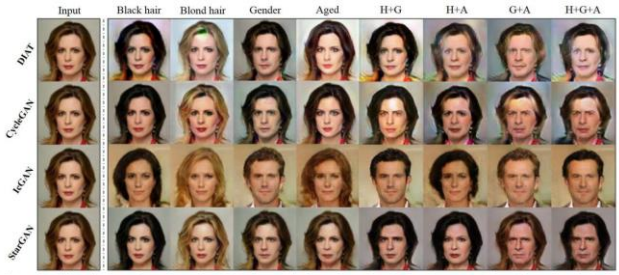

Figure 4. Facial attribute transfer results on the CelebA dataset. The first column shows the input image, next four columns show the single attribute transfer results, and rightmost columns show the multi-attribute transfer results. H: Hair color, G: Gender, A: Aged.

在RaFD人脸数据集上的表情合成：这些图片是由StarGAN在RaFD人脸数据集上训练后生成的。StarGAN清楚地生成最自然的表情，同时适当地保持输入的个人身份和面部特征。虽然DIAT和CycleGAN基本上保留了输入的标识，但是它们的许多结果显示得很模糊，并且不能保持输入中看到的锐度。IcGAN甚至没有通过生成男性形象来保持图像中的个人身份。

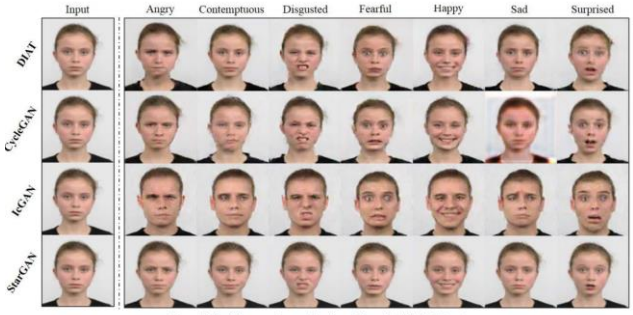

Figure 5. Facial expression synthesis results on the RaFD dataset.

在StarGAN-SNG和StarGAN-JNT之间进行定性比较，结果显示StarGAN-JNT展现了高质量的表情，而StarGAN-SNG合理但背景为灰色的模糊图像。这种差异是由于StarGAN-JNT学会了在训练中变换CelebA图像，但StarGAN-SNG并没有。换句话说，StarGAN-JNT可以利用这两个数据集来改进共享的低级任务，比如面部关键点检测和分割。利用CelebA和RaFD，StarGAN-JNT可以改善这些低级任务，有利于学习面部表情合成。

Figure 6. Facial expression synthesis results of StarGAN-SNG and StarGAN-JNT on CelebA dataset.

References

[1] Choi Y, Choi M, Kim M, et al. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 8789-8797.