深入理解风格迁移三部曲(一)--UNIT

标签（空格分隔）： 陈扬

> MUNIT: Multimodal Unsupervised Image-to-Image Translation
>
> github:https://github.com/NVlabs/MUNIT
>
> 作者:陈扬

简介

近期我研究的方向转向了GAN的应用, 其中图像的风格迁移是GAN中一个非常有意思的应用,传统的方法基于拉普拉斯金字塔对成对的图像进行纹理上的风格迁移.随着2014年GAN的爆火,研究者发现GAN通过判别器D学习两个图像域的关系,实现了unpaired image-to-image(非成对图像数据集的风格迁移)的功能,其中有两个广为人知的应用分别是pix2pix和cycleGAN,今天我们另辟蹊径,从NVIDIA-Lab提出的UNIT框架来探索image-to-image的实现原理.

前言

在开始说UNIT之前,我们先来简要的回顾一下GAN和VAE,这也是在我之前的ODOG项目中有过详细的介绍.

GAN

生成对抗网络（英语：**G**enerative **A**dversarial **N**etwork，简称GAN）是非监督式学习的一种方法，通过让两个神经网络相互博弈的方式进行学习。该方法由伊恩·古德费洛等人于2014年提出。[1]

核心公式:

In other words, D and G play the following two-player minimax game with value function $V(G, D)$:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]. \quad (1)$$

这个公式我们要分成两个部分来看: 先看前半部分:

$$\max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

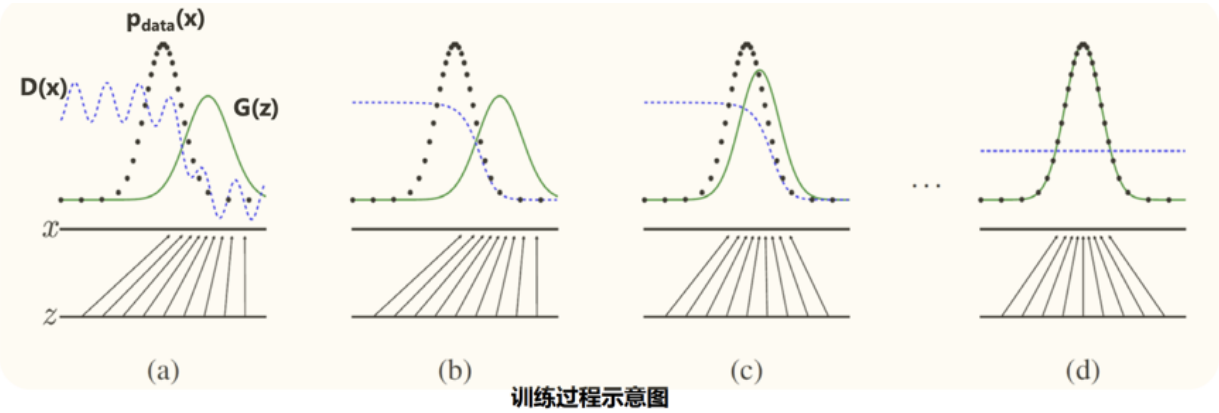
这个公式的意思是,先看加号前面$\mathrm{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathrm{E}_{z\sim p_z(z)}[log(1 - D(G(z)))]$

,我们希望D最大,所以$log(D(x))$应该最大,意味着我的判别器可以很好的识别出,真实世界图像是"true",在看加号后面$\mathrm{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$,要让log尽可能的大,需要的是$D(G(z))$尽可能的小,意味着我们生成模型的图片应该尽可能的被判别模型视为"FALSE".

再看后半部分部分

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

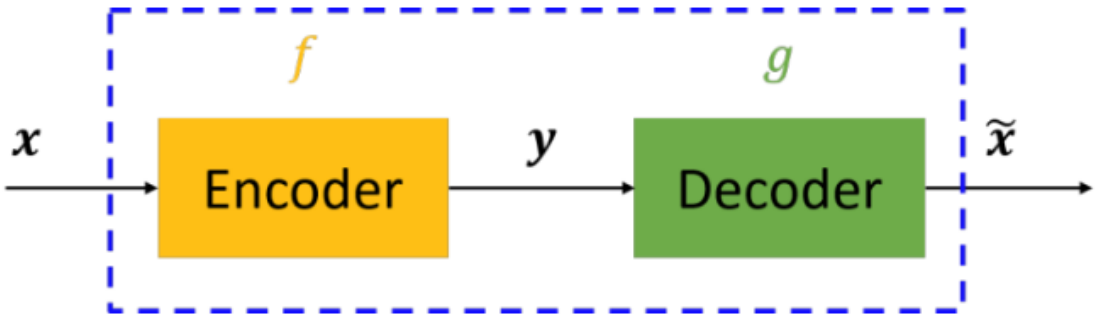
, 我们应该让G尽可能的小,加号前面的式子并没有G,所以无关,在看加号后面的式子$\mathrm{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$,要让$G$尽可能地小,就要$D(G(Z))$尽可能的大,也就是说本来就一张→噪声生成的图片,判别器却被迷惑了,以为是一张真实世界图片.这就是所谓的以假乱真.



训练过程示意图

VAE

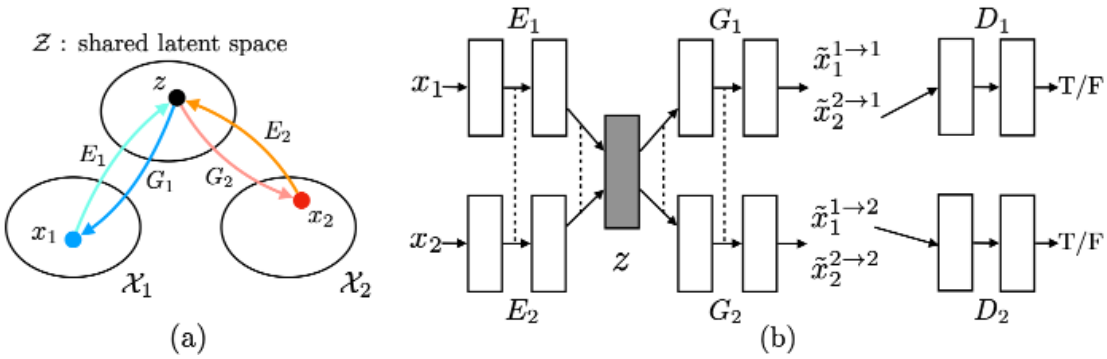
VAE也叫变分自动编码机, 其基础是自编码机(autoencoder),AE通过对输入X进行编码后得到一个低维的向量y，然后根据这个向量还原出输入X，。通过对比X与$\widetilde{X}$，的误差，再利用神经网络去训练使得误差逐渐 减小从而达到非监督学习的目的.



简单点来说,就是先通过f把x映射到y,一般来说我们假定$y \sim N(0,1)$,再通过g把y映射到$\widetilde{X}$,在这里不长篇大论Reparametrization tricks.

UNIT

UNIT(Unsupervised Image-to-Image Translation Networks),由NVIDIA-Lab在NIPS 2017年提出,该文章首次提Image-Image Translation这个概念，将计算机视觉和计算机图形学的许多任务总结进去，分为一对多和多对一的两类转换任务，包括CV里的边缘检测，图像分割，语义标签以及CG里的mapping labels or sparse user inputs to realistic images.



该文章定义了χ_1和χ_2作为两个图像域.传统的supervised Image-to-image 通过对图像域进行采样,求其联合概率分布$P_{(\chi_1,\chi_2)}(x_1,x_2)$,通过`Encoder-Decoder`的思想,作者定义了两个E和G,希望使得$z=E(X)$在`latent space`上近可能的分布一致.意味着当我们同时对$Sample(\chi_1,\chi_2)$时,我们希望得出:

$$z = E_1^*(x_1) = E_2^*(x_2) \tag{1}$$

这样,我们得到了两个Domain下image的一致表示,再通过令$G=D$,从latent space中重构$\hat{x}=G(z)$,

因此,我们两个采样下的$\{x_1,x_2\}$经过$\{<E_1,G_1>,<E_2,G_1>,<E_1,G_2>,<E_2,G_1>\}$后得到了$\{\hat{x}_1^{1\to1},\hat{x}_2^{2\to1},\hat{x}_1^{1\to2},\hat{x}_2^{2\to2}\}$,再把:

$$\hat{x}_1^{1\to1},\hat{x}_2^{2\to1} \to D_1 \to T/F \tag{2}$$

$$\hat{x}_1^{1\to2},\hat{x}_2^{2\to2} \to D_2 \to T/F \tag{3}$$

通过Adv_loss对抗学习跨域生成图片的效果.

可能细心的你以及发现了这是不是很类似VAE-GAN吗?是的.

损失函数

作者通过联合训练4个网络VAE_1,VAE_2,GAN_1,GAN_2的三个$loss function$来训练整个网络:

$$\min_{E_1,E_2,G_1,G_2}\max_{D_1,D_2}\mathcal{L}_{\text{VAE}_1}(E_1,G_1)+\mathcal{L}_{\text{GAN}_1}(E_2,G_1,D_1)+\mathcal{L}_{\text{CC}_1}(E_1,G_1,E_2,G_2)$$
$$\mathcal{L}_{\text{VAE}_2}(E_2,G_2)+\mathcal{L}_{\text{GAN}_2}(E_1,G_2,D_2)+\mathcal{L}_{\text{CC}_2}(E_2,G_2,E_1,G_1)$$

```
# ------------------------------
#  Train Encoders and Generators
# ------------------------------
# Total loss
```

```
loss_G = (
    loss_KL_1
    + loss_KL_2
    + loss_ID_1
    + loss_ID_2
    + loss_GAN_1
    + loss_GAN_2
    + loss_KL_1_
    + loss_KL_2_
    + loss_cyc_1
    + loss_cyc_2
)
loss_D1 = criterion_GAN(D1(X1), valid) +
criterion_GAN(D1(fake_X1.detach()), fake)
loss_D2 = criterion_GAN(D2(X2), valid) +
criterion_GAN(D2(fake_X2.detach()), fake)
```

`VAE`的目标是minimize source domain to latent space's KL diversity and latent space to destination domain's KL diversity(我觉得中文太拗口了,这句话实在是说不来)来最小化变分上界,VAE的定义如下:

$$\begin{aligned}\mathcal{L}_{\mathrm{VAE}_1}\left(E_1, G_1\right) &= \lambda_1 \mathrm{KL}(q_1\left(z_1|x_1\right)\|p_\eta(z)) - \lambda_2 \mathbb{E}_{z_1\sim q_1(z_1|x_1)}\left[\log p_{G_1}\left(x_1|z_1\right)\right] \\ \mathcal{L}_{\mathrm{VAE}_2}\left(E_2, G_2\right) &= \lambda_1 \mathrm{KL}(q_2\left(z_2|x_2\right)\|p_\eta(z)) - \lambda_2 \mathbb{E}_{z_2\sim q_2(z_2|x_2)}\left[\log p_{G_2}\left(x_2|z_2\right)\right]\end{aligned} \tag{4}$$

```
# Get shared latent representation
mu1, Z1 = E1(X1)
mu2, Z2 = E2(X2)
# Reconstruct images
recon_X1 = G1(Z1)
recon_X2 = G2(Z2)
# Translate images
fake_X1 = G1(Z2)
fake_X2 = G2(Z1)
loss_KL_1 = lambda_1 * compute_kl(mu1)
loss_KL_2 = lambda_1 * compute_kl(mu2)
loss_KL_1_ = lambda_3 * compute_kl(mu1_)
loss_KL_2_ = lambda_3 * compute_kl(mu2_)
```

对抗:GAN_LOSS被用于确保翻译图像类似图像在目标域.定义如下:

$$\begin{aligned}\mathcal{L}_{\mathrm{GAN}_1}\left(E_2, G_1, D_1\right) &= \lambda_0 \mathbb{E}_{x_1\sim P_{\mathcal{X}_1}}\left[\log D_1\left(x_1\right)\right] + \lambda_0 \mathbb{E}_{z_2\sim q_2(z_2|x_2)}\left[\log(1 - D_1\left(G_1\left(z_2\right)\right))\right] \\ \mathcal{L}_{\mathrm{GAN}_2}\left(E_1, G_2, D_2\right) &= \lambda_0 \mathbb{E}_{x_2\sim P_{\mathcal{X}_2}}\left[\log D_2\left(x_2\right)\right] + \lambda_0 \mathbb{E}_{z_1\sim q_1(z_1|x_1)}\left[\log(1 - D_2\left(G_2\left(z_1\right)\right))\right]\end{aligned} \tag{5}$$

```
loss_GAN_1 = lambda_0 * criterion_GAN(D1(fake_X1), valid)
loss_GAN_2 = lambda_0 * criterion_GAN(D2(fake_X2), valid)
loss_D1 = criterion_GAN(D1(X1), valid) +
criterion_GAN(D1(fake_X1.detach()), fake)
loss_D2 = criterion_GAN(D2(X2), valid) +
criterion_GAN(D2(fake_X2.detach()), fake)
```

循环一致性:由于shared latent-space假设暗含了循环一致性约束，因此我们在提出的框架中实施循环一致性约束，以进一步规范不适定的无监督图像间转换问题。产生的信息处理流称为循环重建流,定义如下:

$$\begin{aligned}\mathcal{L}_{\mathrm{CC}_1}\left(E_1, G_1, E_2, G_2\right) =& \lambda_3 \mathrm{KL}(q_1\left(z_1 | x_1\right) \| p_\eta(z)) + \lambda_3 \mathrm{KL}\left(q_2 | x_1^{1 \rightarrow 2}\right)\right) \| p_\eta(z)\big) - \\ & \lambda_4 \mathbb{E}_{z_2 \sim q_2\left(z_2 | x_1^{1 \rightarrow 2}\right)}\left[\log p_{G_1}\left(x_1 | z_2\right)\right] \\ \mathcal{L}_{\mathrm{CC}_2}\left(E_2, G_2, E_1, G_1\right) =& \lambda_3 \mathrm{KL}(q_2\left(z_2 | x_2\right) \| p_\eta(z)) + \lambda_3 \mathrm{KL}\left(q_1\left(z_1 | x_2^{2 \rightarrow 1}\right)\right) \| p_\eta(z)\big) - \\ & \lambda_4 \mathbb{E}_{z_1 \sim q_1\left(z_1 | x_2^{2 \rightarrow 1}\right)}\left[\log p_{G_2}\left(x_2 | z_1\right)\right]\end{aligned}$$

```
# Cycle translation
mu1_, Z1_ = E1(fake_X1)
mu2_, Z2_ = E2(fake_X2)
cycle_X1 = G1(Z2_)
cycle_X2 = G2(Z1_)
loss_ID_1 = lambda_2 * criterion_pixel(recon_X1, X1)
loss_ID_2 = lambda_2 * criterion_pixel(recon_X2, X2)
loss_cyc_1 = lambda_4 * criterion_pixel(cycle_X1, X1)
loss_cyc_2 = lambda_4 * criterion_pixel(cycle_X2, X2)
```

训练好的网络,我们可以通过对`latent sapce`的`latent variable`重编码,进而把输入图像迁移到各个域中:

实验结果

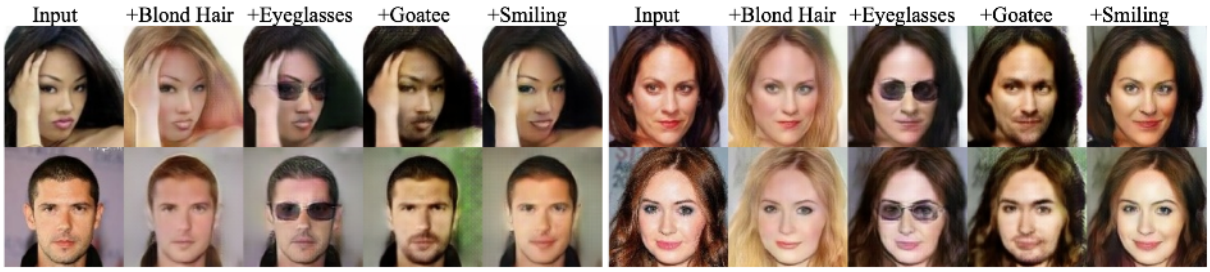


Figure 6: Attribute-based face translation results.

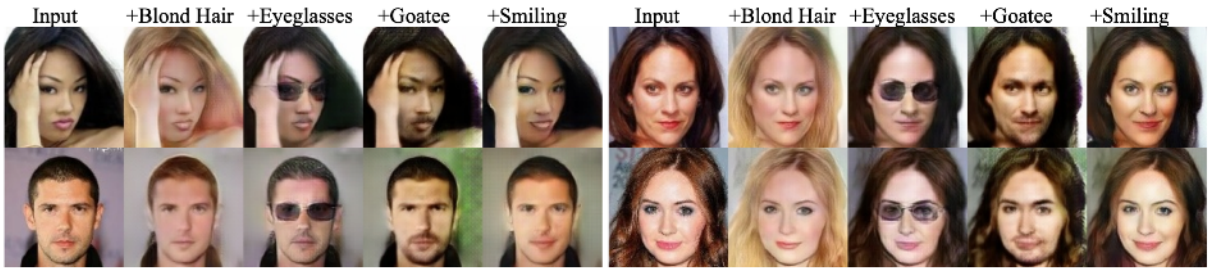


Figure 6: Attribute-based face translation results.

作者在展示的时候看起来好像可以实现一对多的风格转换,实际上这个算法只能实现`1对1的风格迁移`,是作者做了`N对1对1`的实验,所以看起来像`1对N`的结果.

这算是比较早期的一篇文章,其实现原理也是借鉴很很多前人的工作,实际上我觉得从原创性上来看比不上cycleGAN,不过这这个VAE-GAN延伸的应用性似乎更好一些.接下来会介绍NVIDIA-Lab的FUNIT和MUNIT.
