基于starGAN的人脸属性转换

StarGAN for multi-domain Face Attribute Conversion

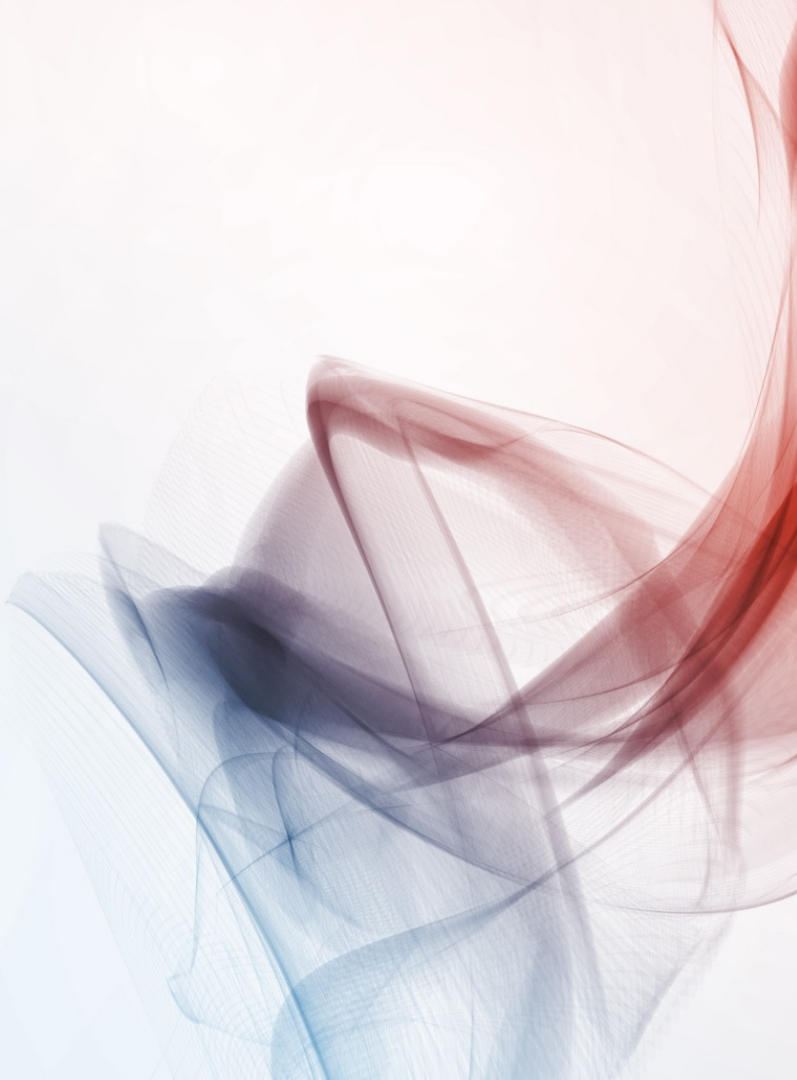
于遨波
陈志鸿

数字图像处理·课程项目报告

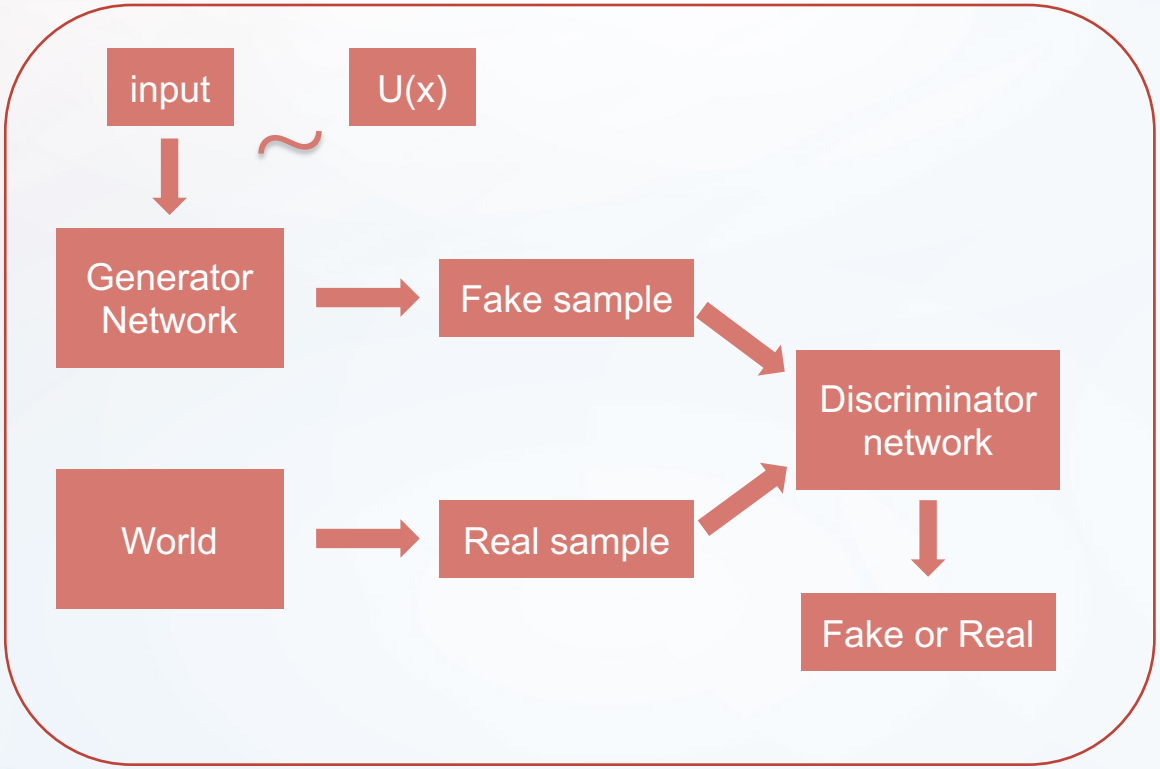
目录 CONTENTS

01 项目概述

Project Overview



项目概述



生成对抗网络（GAN）

Generative adversarial networks.
由生成器(Generator)和鉴别器(Discriminator)组成。
二者通过一个对抗式的过程进行训练。直至生成器可以生成足以“以假乱真”的图片。

[1] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]//Advances in neural information processing systems. 2014: 2672-2680.

项目概述

风格迁移（Style-Transfer）

超分辨率图像生成（Super Resolution）

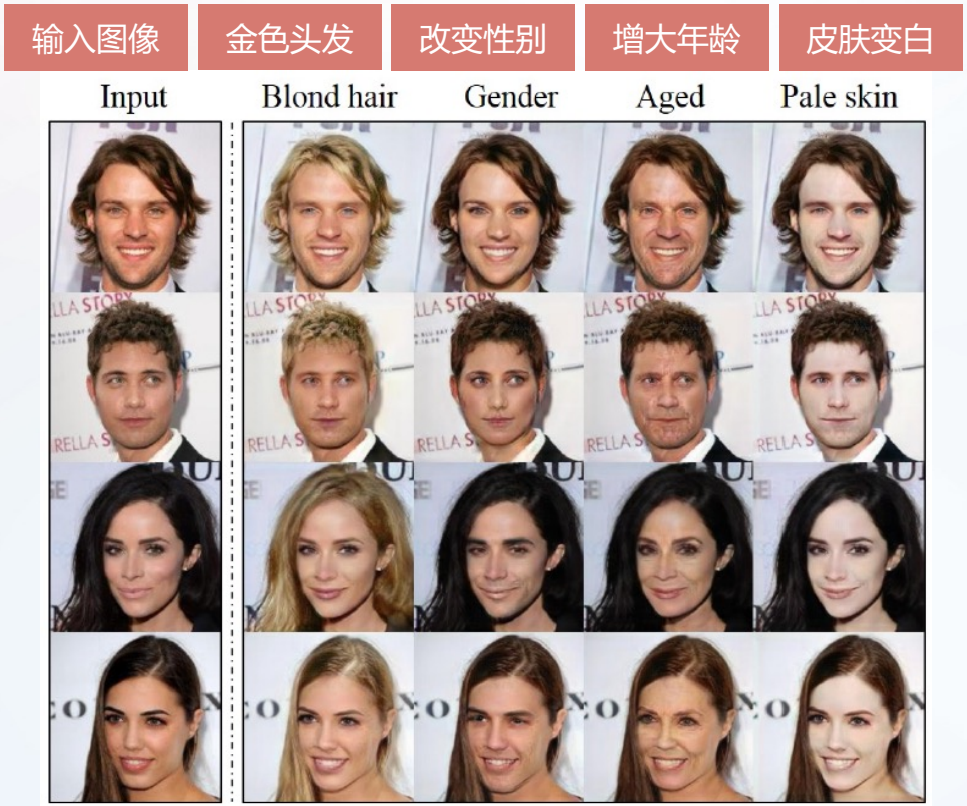
分割图像（手绘图）→ 实景图

图像生成

采用GAN及其变种网络，对图片内容的属性进行转换。例如从分割图像或手绘图变为实景图，低像素图像的复原，改变图像风格等。

[1] Ledig C, Theis L, Huszár F, et al. Photo-realistic single image super-resolution using a generative adversarial network[J]. arXiv preprint, 2017.
[2] Zhu J Y, Park T, Isola P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks[J]. arXiv preprint arXiv:1703.10593, 2017.
[3] Isola P, Zhu J Y, Zhou T, et al. Image-to-image translation with conditional adversarial networks[J]. arXiv preprint, 2017.

项目概述



StarGAN

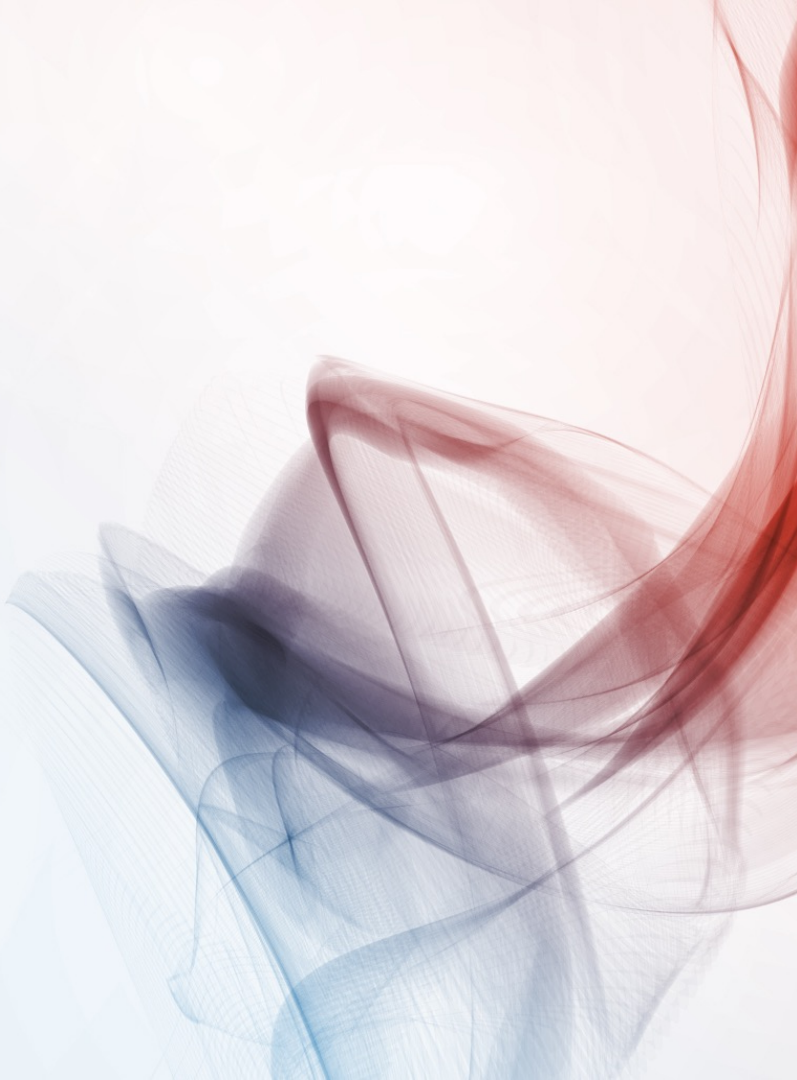
与传统的GAN不同，StarGAN只使用一个生成器，使图片从一种属性转换为多种属性，进行多域图片转换。

[1] Choi Y, Choi M, Kim M, et al. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation[J]. arXiv preprint arXiv:1711.09020, 2017.

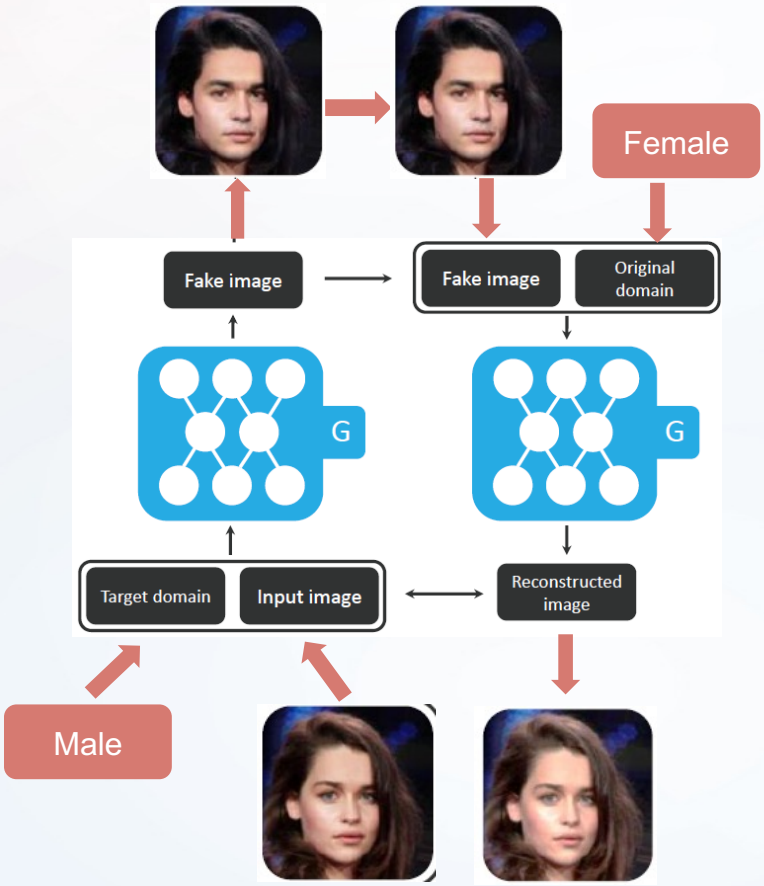
02

算法流程

Algorithm flow



算法流程



生成器（Generator）

输入：图像 + 目标转换属性

输出：伪造的图像（Fake Image）

$$\mathcal{L}_{rec} = \mathbb{E}_{x,c,c'}[||x - G(G(x,c),c')||_1],$$

算法流程

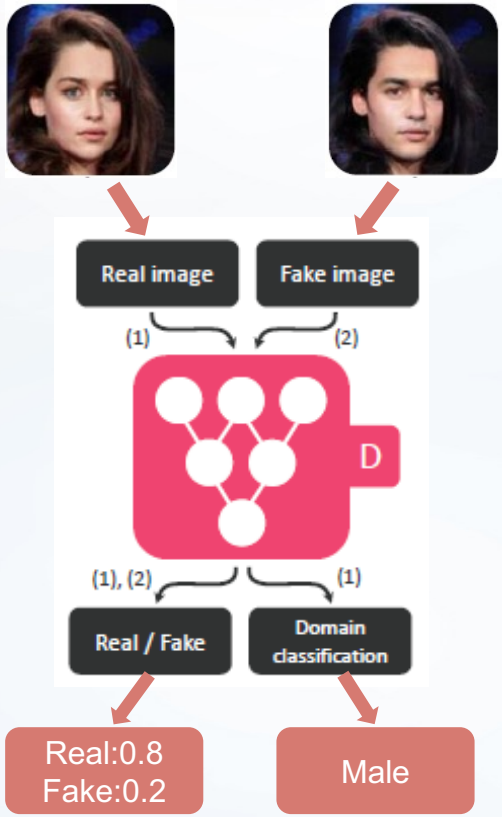
Part	Input → Output Shape	Layer Information
Down-sampling	$(h, w, 3+n_c) \rightarrow (h, w, 64)$	CONV-(N64, K7x7, S1, P3), IN, ReLU
	$(h, w, 64) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$	CONV-(N128, K4x4, S2, P1), IN, ReLU
	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	CONV-(N256, K4x4, S2, P1), IN, ReLU
Bottleneck	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
Up-sampling	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$	DECONV-(N128, K4x4, S2, P1), IN, ReLU
	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (h, w, 64)$	DECONV-(N64, K4x4, S2, P1), IN, ReLU
	$(h, w, 64) \rightarrow (h, w, 3)$	CONV-(N3, K7x7, S1, P3), Tanh

生成器（Generator）

输入：图像 + 目标转换属性

输出：伪造的图像（Fake Image）

算法流程



判别器（Discriminator）

输入：真实图像 or 伪造图像

输出：
①图像是真实图像的概率（Real or Fake）
②图像的分类

$$\mathcal{L}_{adv} = \mathbb{E}_x [\log D_{src}(x)] + \mathbb{E}_{x,c}[\log (1 - D_{src}(G(x,c))],$$

$$\mathcal{L}^r_{cls} = \mathbb{E}_{x,c'}[-\log D_{cls}(c'|x)],$$

算法流程

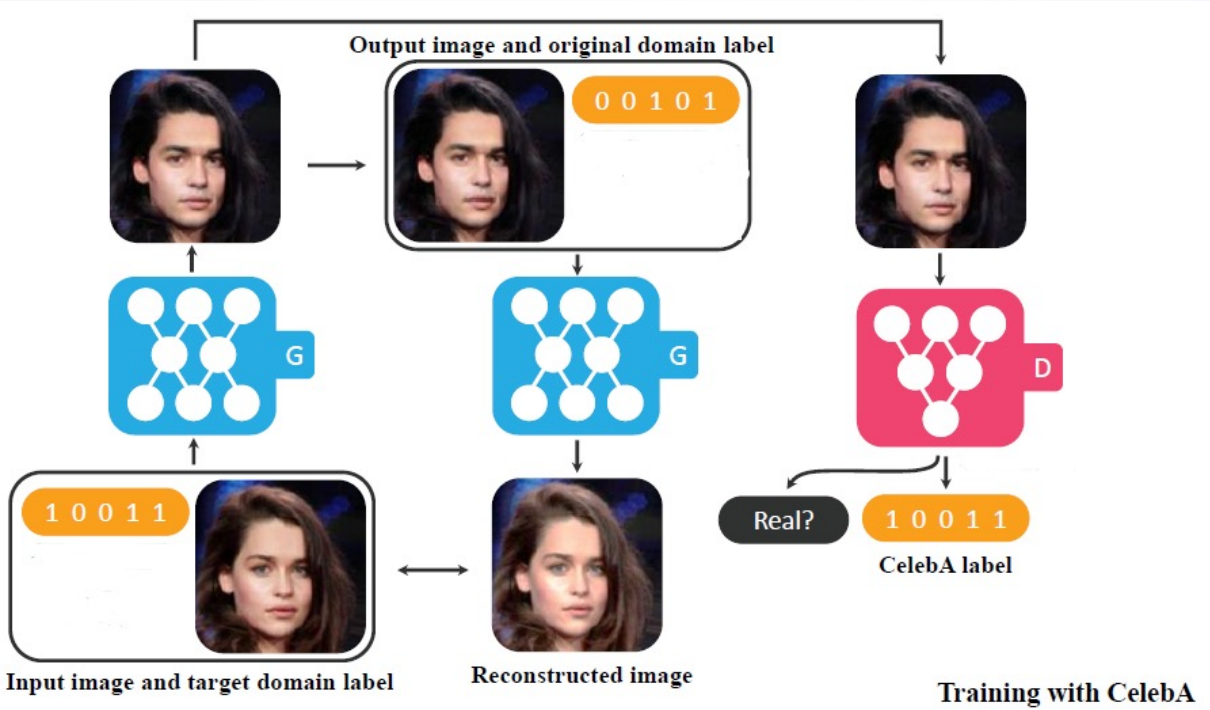
Layer	Input → Output Shape	Layer Information
Input Layer	$(h, w, 3) \rightarrow (\frac{h}{2}, \frac{w}{2}, 64)$	CONV-(N64, K4x4, S2, P1), Leaky ReLU
Hidden Layer	$(\frac{h}{2}, \frac{w}{2}, 64) \rightarrow (\frac{h}{4}, \frac{w}{4}, 128)$	CONV-(N128, K4x4, S2, P1), Leaky ReLU
Hidden Layer	$(\frac{h}{4}, \frac{w}{4}, 128) \rightarrow (\frac{h}{8}, \frac{w}{8}, 256)$	CONV-(N256, K4x4, S2, P1), Leaky ReLU
Hidden Layer	$(\frac{h}{8}, \frac{w}{8}, 256) \rightarrow (\frac{h}{16}, \frac{w}{16}, 512)$	CONV-(N512, K4x4, S2, P1), Leaky ReLU
Hidden Layer	$(\frac{h}{16}, \frac{w}{16}, 512) \rightarrow (\frac{h}{32}, \frac{w}{32}, 1024)$	CONV-(N1024, K4x4, S2, P1), Leaky ReLU
Hidden Layer	$(\frac{h}{32}, \frac{w}{32}, 1024) \rightarrow (\frac{h}{64}, \frac{w}{64}, 2048)$	CONV-(N2048, K4x4, S2, P1), Leaky ReLU
Output Layer (D_{src})	$(\frac{h}{64}, \frac{w}{64}, 2048) \rightarrow (\frac{h}{64}, \frac{w}{64}, 1)$	CONV-(N1, K3x3, S1, P1)
Output Layer (D_{cls})	$(\frac{h}{64}, \frac{w}{64}, 2048) \rightarrow (1, 1, n_d)$	CONV-(N(n_d), K$\frac{h}{64}$x$\frac{w}{64}$, S1, P0)

判别器（Discriminator）

输入：真实图像 or 伪造图像

输出：
①图像是真实图像的概率（Real or Fake）
②图像的分类

算法流程



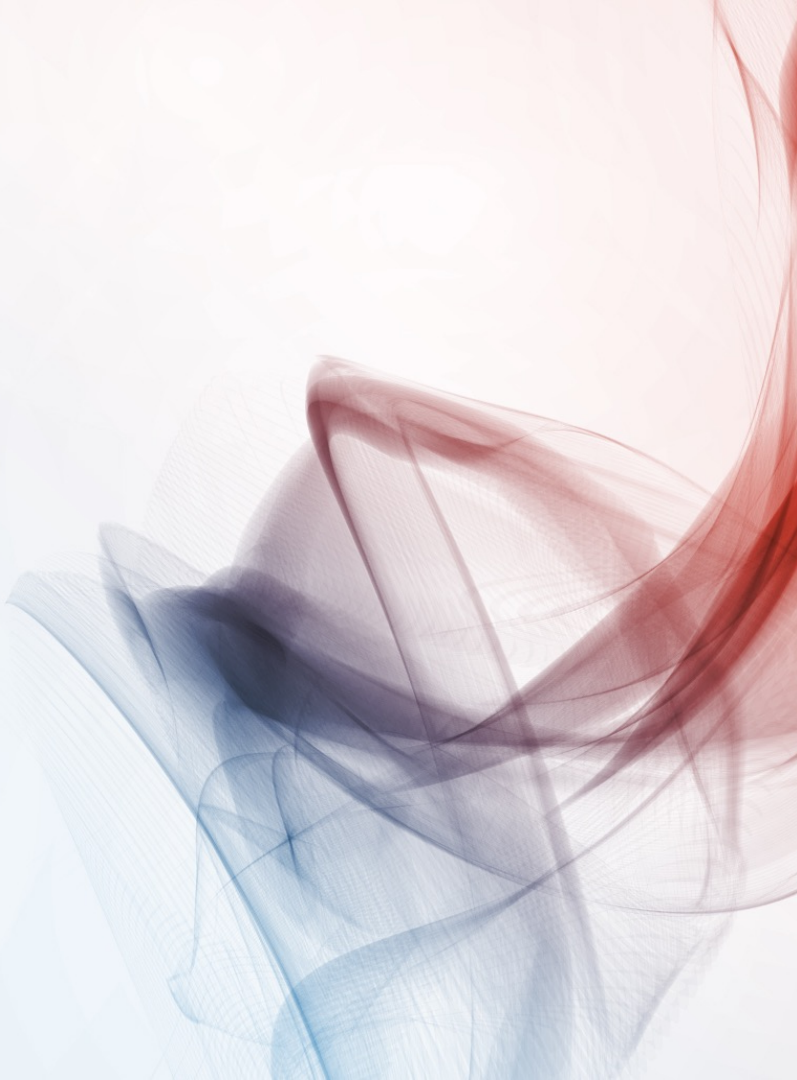
整体流程图

$$\mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{cls}\,\mathcal{L}^r_{cls},$$

$$\mathcal{L}_G = \mathcal{L}_{adv} + \lambda_{cls}\,\mathcal{L}^f_{cls} + \lambda_{rec}\,\mathcal{L}_{rec},$$

03 实验结果

Experiment results



实验结果

① 头发颜色

②眼睛转换

③表情转换

④动漫与素描转换

实验结果

① 头发颜色转换

训练数据集：celebA

输入图像 | 黑发 | 金发 | 棕发 | 改变性别 | 变老

实验结果

① 头发颜色转换

训练数据集：celebA

输入图像 | 黑发 | 金发 | 棕发 | 变老 | 改变性别

实验结果

① 头发颜色转换

训练数据集：celebA

输入图像 | 黑发 | 金发 | 棕发 | 变老 | 改变性别

实验结果

② 眼睛转换

训练数据集：celebA

实验结果

② 眼睛转换

训练数据集：celebA

输入图像 | 双眼皮 | 单眼皮 | 闭眼

实验结果

③ 表情转换

训练数据集：CK+

实验结果

③ 表情转换

训练数据集：CK+

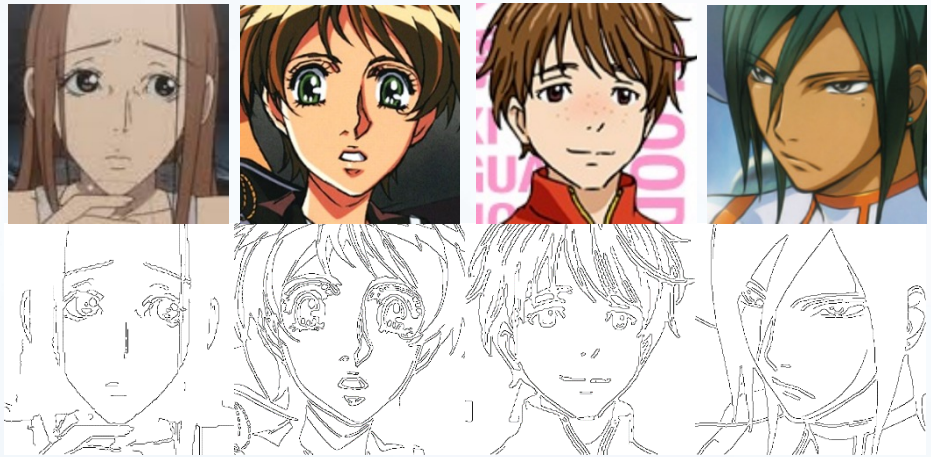
惊讶

实验结果

④动漫与素描的转换

训练数据集：自制数据集

来源：https://www.anime-planet.com
数量：90000
预处理得到人物的素描图
训练：80000
测试：10000

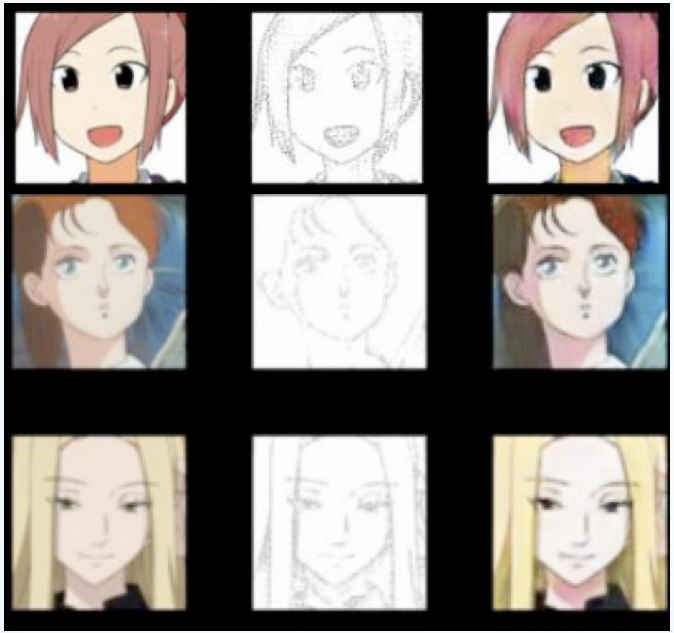


实验结果

④动漫与素描的转换

训练数据集：自制数据集

输入图像 | 素描 | 动漫



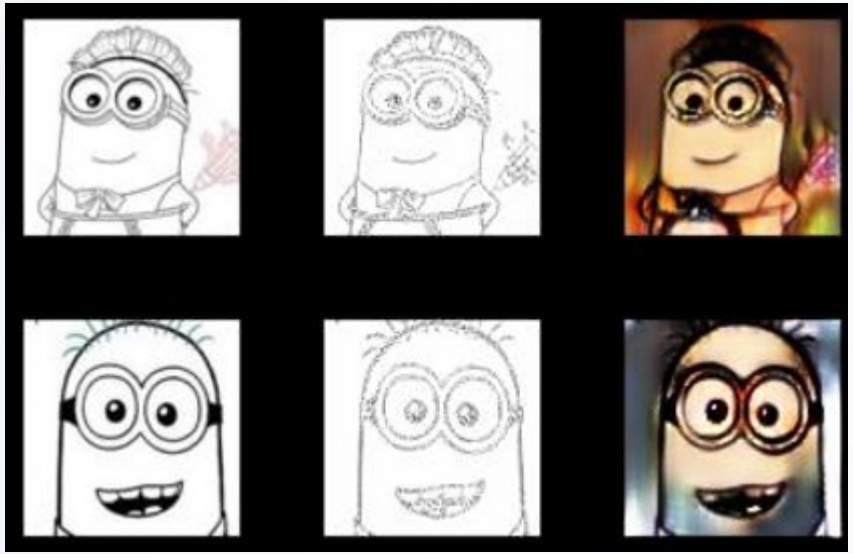
输入图像 | 素描 | 动漫

实验结果

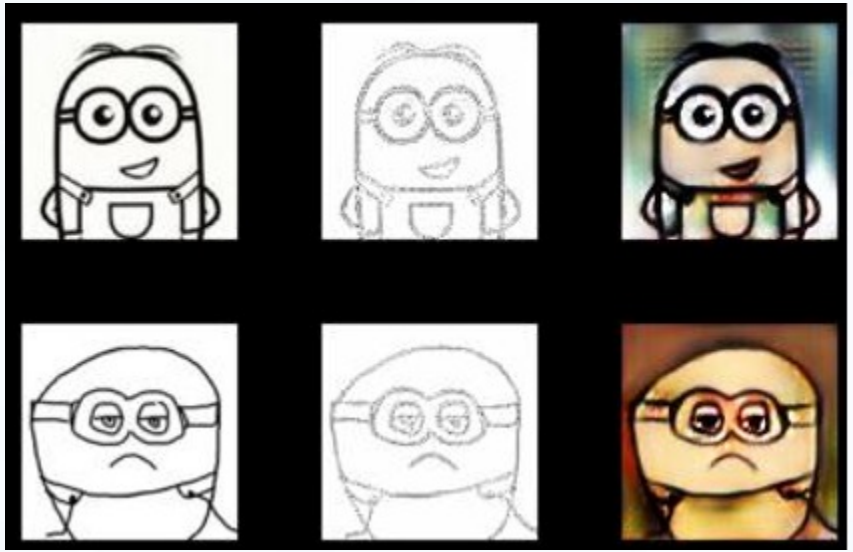
④动漫与素描的转换

训练数据集：自制数据集

输入图像 素描 动漫



输入图像 素描 动漫

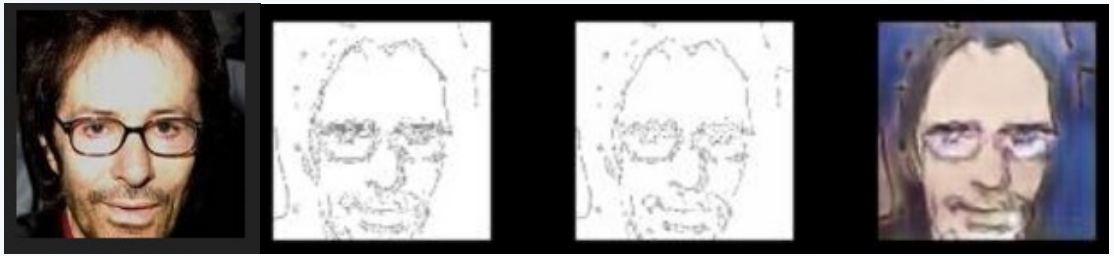


实验结果

④动漫与素描的转换

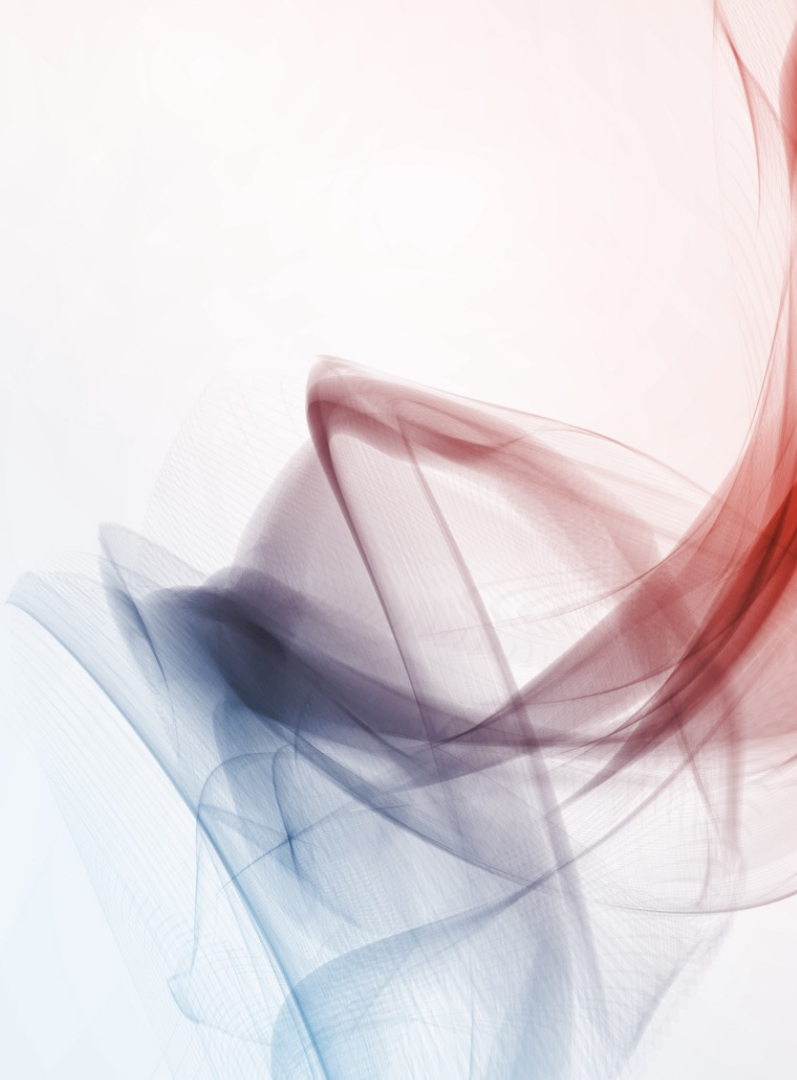
训练数据集：自制数据集

输入图像 | 素描 | 素描 | 动漫



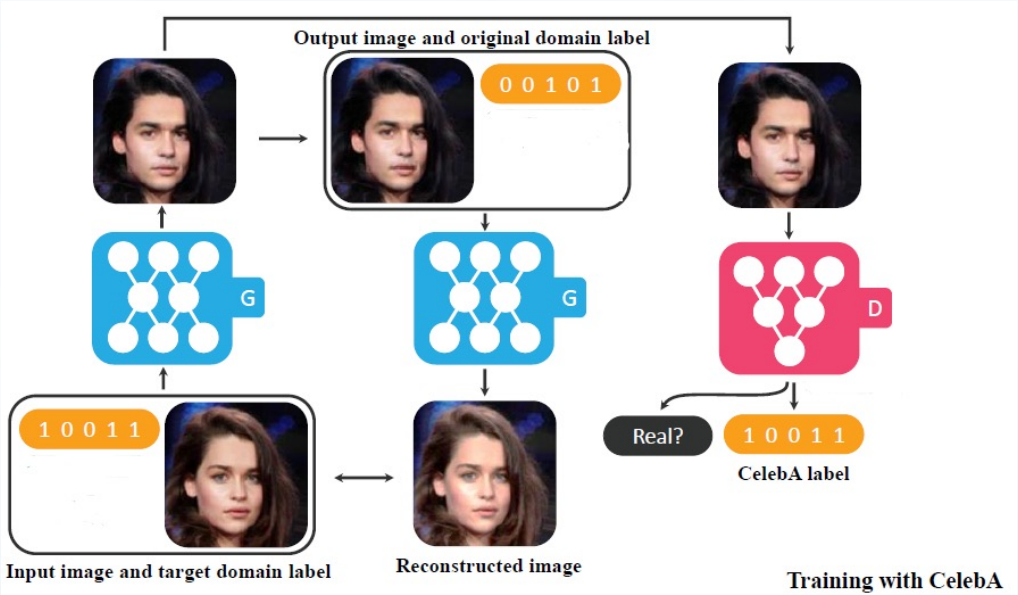
04 项目总结

Project summary



项目总结

- StarGAN：使用单一生成器和单一鉴别器进行多域图像转换
- 项目中存在的问题：①缺少训练数据集 ②训练时间长
- 未来展望：① 制定适合不同训练数据集的目标函数
  ② 调参（调整不同目标函数的权重）



谢谢观看

Thanks for watching

于遨波
陈志鸿