

课程报告要求：

选题：

- 1) 每个小组已经选定的小项目课题
- 2) 完成项目的整体设计与实现
- 3) 完成项目报告，每个小组完成一份项目报告

文档要求：

- 1) 修改副标题处的姓名、学号；
- 2) 根据各组情况，选择使用 Word 或 Markdown（推荐 Markdown）；
- 3) Word 中公式输入采用自带公式输入或 Mathtype 软件；
- 4) 将文档转为 PDF 文件，文件名设置为“创新实践 **2**-课程报告-（组号，成员 **ABC**）”；

作业提交：

- 1) 发送到邮箱：zhusuguo@163.com，邮件标题为“创新实践 **3**-课程报告-（组号，成员 **ABC**）”；
- 2) 提交时间：**2019 年 12 月 25 日**。

CARTOONGAN 复现

姓名 屠怡泽 学号 17052025

姓名 徐敏超 学号 17052032

1 引言

由于

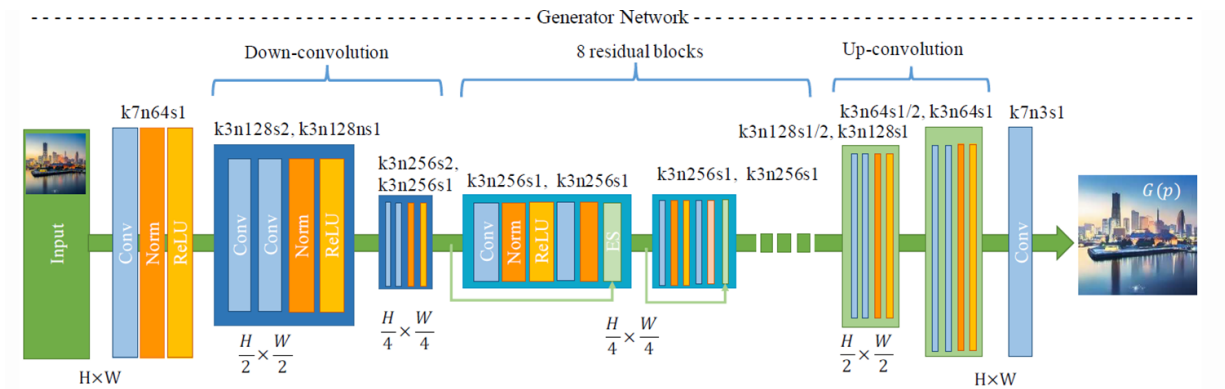
1. 卡通风格具有高水平简化和抽象的独特特征
2. 卡通图像倾向于具有清晰的边缘，平滑的颜色阴影和相对简单的事实

现有的方法不能产生令人满意的卡通化结果。作者提出了 **CartoonGAN**，它可以用真实景物的照片作为源图片，生成任意风格的漫画，如新海诚或者宫崎骏。

一种将现实世界场景的照片转换为卡通风格的图像的生成对抗网络（GAN）框架，通过一组照片和一组卡通图片进行训练，这两组照片无需对应，基于 GAN 的方法，可以有效地学习使用不成对图像集进行训练的真实世界照片到卡通图像的映射。

2 关键技术

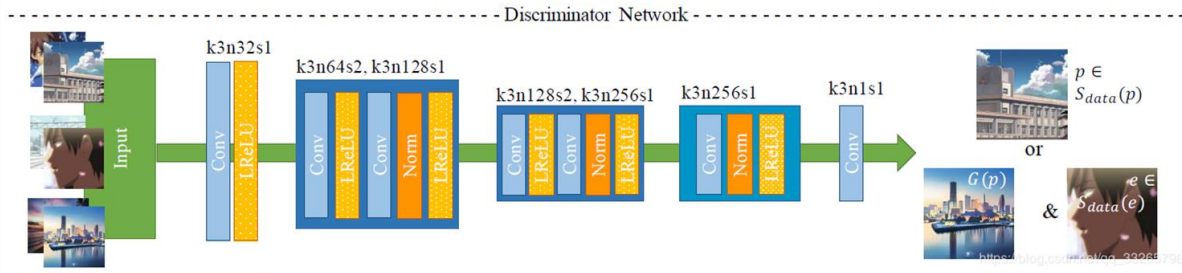
在 **CartoonGAN** 中，生成器网络 **G** 用于将输入图像映射到漫画流型中。在模型经过训练后，漫画风格化任务就可以执行了。**G** 从平卷积阶段开始，随后是两个下卷积区块，以对图像进行空间压缩和编码。在这个阶段提取有用的本地信号用于下游转换。随后使用相同布局的八个残差区块来构建内容和流形特征。研究人员采用了《Perceptual losses for real-time style transfer and super-resolution》中提出的残差块布局。最后，通过两个上卷积块重构输出的漫画风格图像，这两个卷积块包含步长为 $1/2$ 的分步卷积层和一个 7×7 内核的最终卷积层。



k 是内核大小， n 是特征映射的数量， s 是每个卷积层的跨步，'norm'表示归一化层，'ES'表示元素之和

为了与生成器网络形成互补，鉴别器网络 **D** 用于判断输入图像是否是「真实的」漫画

图。由于判断图像是否为漫画是个要求不高的任务，因此我们使用 D 中参数较少的简单补丁级鉴别器，而不是常规全图鉴别器。与图像分类任务不同，漫画风格鉴别任务依赖于图像的局部特征。所以，鉴别器网络 D 被设计得较浅。在平层之后，网络采用两个步进卷积块来降低分辨率并编码用于分类的基本局部特征。随后，使用特征构造块和 3×3 卷积层来获得分类返回。在每个归一化层之后使用 $\alpha = 0.2$ 的 Leaky ReLU



相较于之前的方法，CartoonGAN 的主要贡献在于：

1. 提出了一个专用的基于 GAN 的方法，可以有效地学习使用不成对的图片集进行训练，对现实世界照片和漫画图像建立映射。新方法可以生成高质量的风格化漫画，这大大超越了现有最佳技术的能力——当来自特定艺术家的漫画图像被用于训练时，新方法可以忠实重现他们的风格。
2. 在基于 GAN 的体系结构中提出两种简单而有效的损失函数。在生成网络中，为了应对照片和漫画之间的实质风格差异，研究人员在 VGG 网络的高级特征映射中引入了定义为 1 的稀疏正则化语义损失。在鉴别器网络中，作者提出了推进边缘的对抗损失，以保证清晰的边缘。
3. 作者进一步引入了一个初始化阶段来改善网络到目标流形的收敛。新的方法比现有的训练方法效率要高很多。

总的损失函数为

$$L(G, D) = L_{adv}(G, D) + \omega L_{con}(G, D)$$

其中 $L_{adv}(G, D)$ 为带边缘增强功能的对抗损失， $L_{con}(G, D)$ 为语义内容损失， ω 为语义内容损失的权重，平衡两个损失间的关系。 ω 越大，原图像的内容信息保存的越好，有更多的细节纹理，文中所进行的实验 ω 均设为 10。

$$\begin{aligned} \mathcal{L}_{adv}(G, D) = & \mathbb{E}_{c_i \sim S_{data}(c)} [\log D(c_i)] \\ & + \mathbb{E}_{e_j \sim S_{data}(e)} [\log(1 - D(e_j))] \\ & + \mathbb{E}_{p_k \sim S_{data}(p)} [\log(1 - D(G(p_k)))] \end{aligned}$$

$$\begin{aligned} \mathcal{L}_{con}(G, D) = & \mathbb{E}_{p_i \sim S_{data}(p)} [\|VGG_l(G(p_i)) - VGG_l(p_i)\|_1] \end{aligned}$$

对抗损失的设计

论文作者通过实验发现，仅仅通过判别器判别一张图像是否为卡通图像，不足以使现实图像很好地转化为卡通图像。因为卡通图像的边缘是很清晰的，而这些边缘在一整张图片中只占到很小的部分，判别器不能分辨到这些细小的特征，所以生成器不一定能学会生成这些清晰的轮廓。

为了解决这些问题，作者将训练卡通图像样本的边缘去除，并将去除边缘后的样本标记为假样本来训练判别器，从而使判别器能将边缘不明显的图像判别为假样本，生成器就会因此学会生成边缘清晰的图像。去除边缘的步骤有：

- (1)用 Canny 边缘检测算法检测边缘像素点；
- (2)扩大检测出的边缘区域；
- (3)在扩大的边缘区域，做高斯平滑(Gaussian smoothing)。

语义内容损失的设计

图像风格化的目的就是生成保留有输入图像的语义内容，并带有要转换的风格化的图像。在此论文中，作者使用的不是像其他论文的简单在 RGB 域输入图像和生成图像之间的 pixel-wise 的图像重建损失，而是将原真实图像和生成的卡通化图像输入到 VGG 网络中，用得到的 VGG 的 high-level feature maps 做内容损失。

初始化过程

先用 $L_{con}(G,D)$ 单独一个损失来训练 G，大概训练 10 个 epoch，目的是为了生成的图片初始后就能保证原真实场景的内容，毕竟普通的单向 GAN 要是没有加入一些限制很难稳定保证原始图片的内容，大多会出现扭曲或更严重的情况，而类似的，在 CycleGAN 中体现为用 cycle_loss 来保证原始图片内容。

3 实验及结果分析（详细描述）

3.1 介绍实验的数据集：

Data (256*256)

src_data(来自真实世界的图片)

train

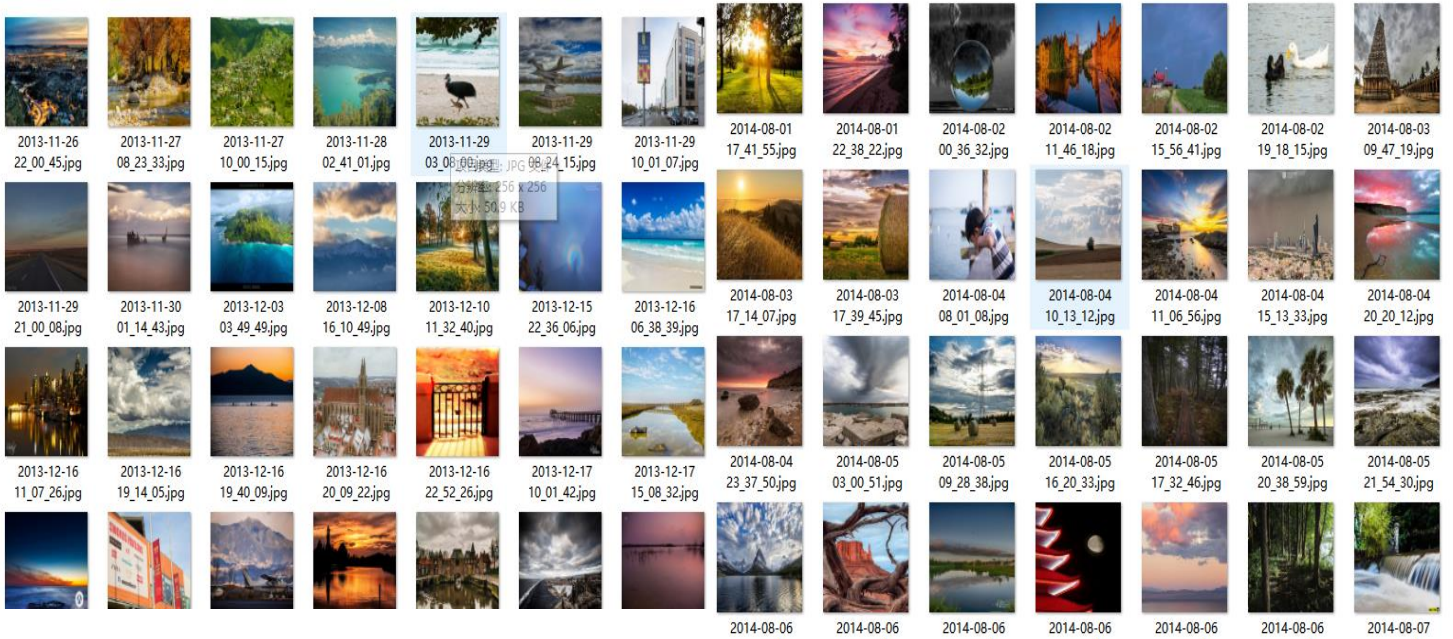
test

tgt_data

train (动漫图片)

pair(对动漫图片进行边缘模糊化处理)

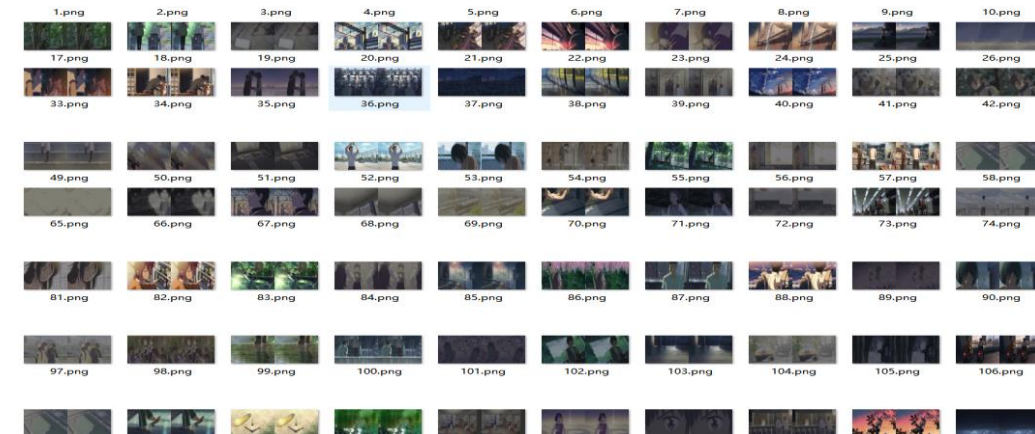
本次实验中我们小组使用的 src_data 中的真实图片的训练集和测试集是 vangogh2photo 数据集中的现实影像



而 tgt_data 中 train 的动漫图片是通过 python 代码先从下载下来的同一个作者（新海诚）的动漫电影（言叶之庭、秒速 5 厘米、你的名字）中每隔 50 帧提取一张图片，再经过手动剔除一些基本相同的图片后，对生成的图片重命名、resize 成 256*256 的图片



而 pair 是通过边缘模糊算法对 tgt_data 中的 train 进行处理得到的, 对边缘模糊是为了得到效果更好的模型 (左边为初始动漫图像, 右边为模糊化后)



3.2 介绍实现算法时用到的编程语言、工具包等;

语言: python 3.5.3

环境: pytorch 0.4.0

Vgg19: 这是直接下载已经经过训练的 vgg 模型来对 generator 进行预训练, 使生成能更好的生成动漫风格的图片

Opencv

3.3 介绍实验的参数设置:

batch_size 设置为 8

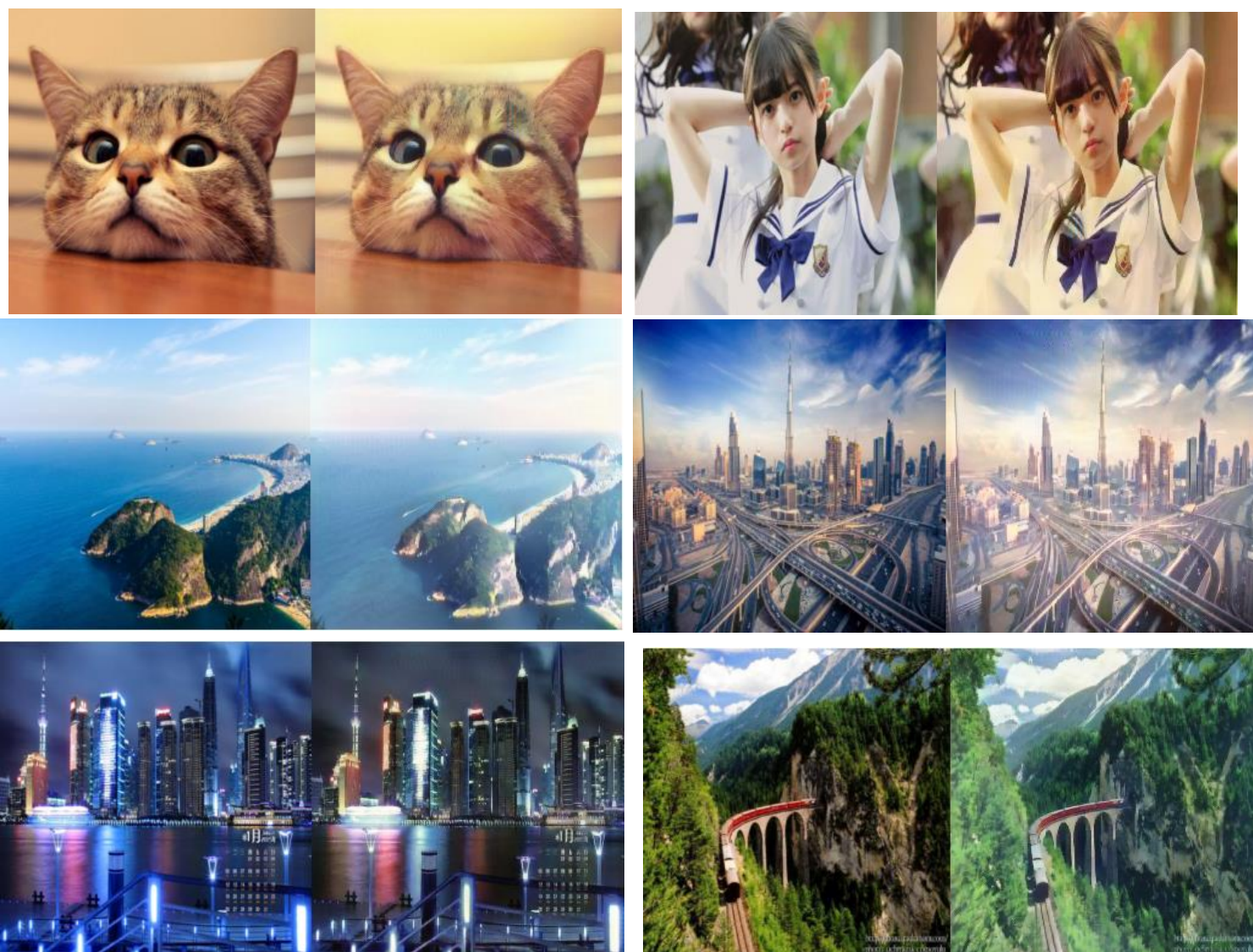
input_size 设置为 256 (输入图片大小)

train_epoch 设置为 100 (训练迭代次数为 100)

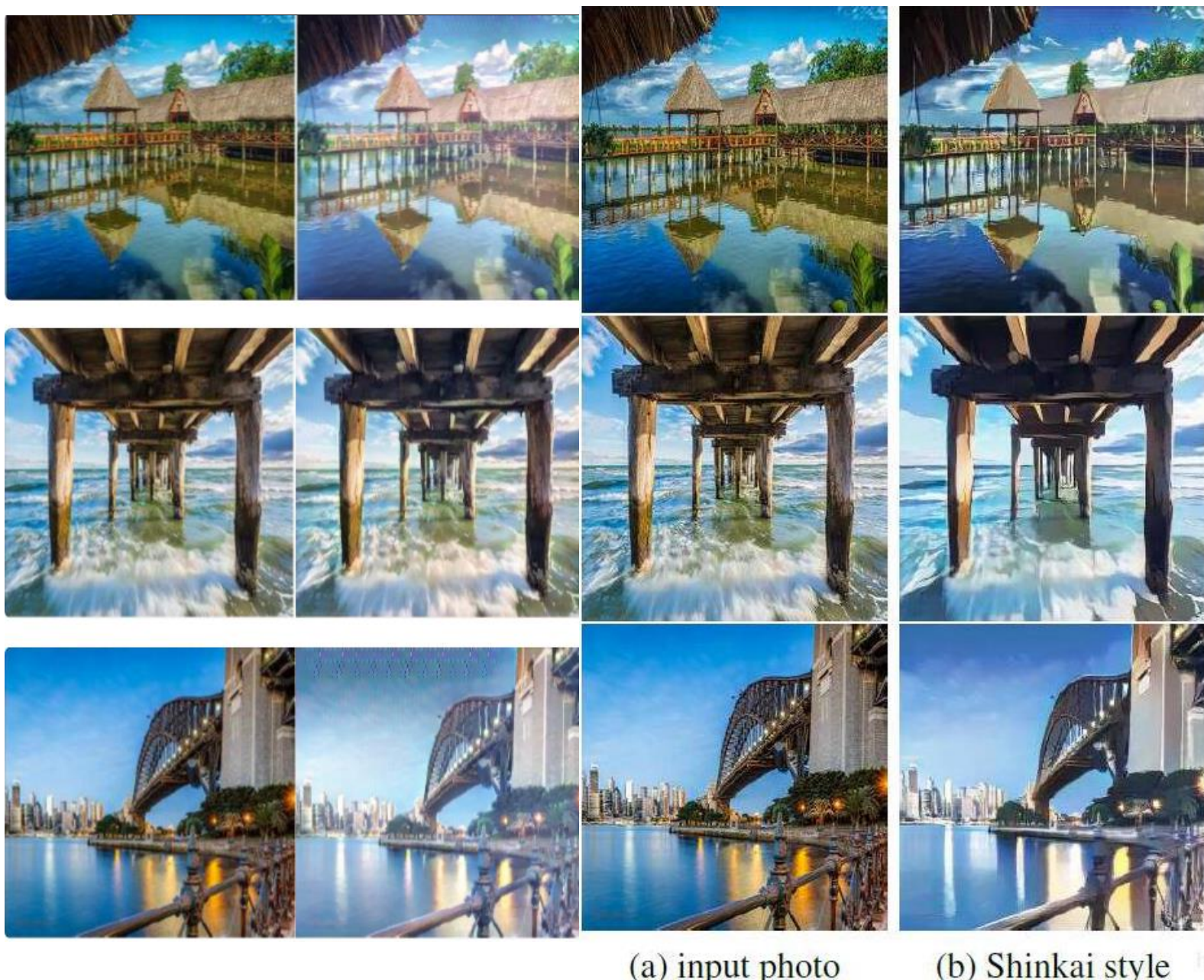
pre_train_epoch 设置为 10 (预训练迭代次数为 10)

lrD、lrG 设置为 0.0002 (generator 和 discriminator 的学习率为 0.002)

3.4 汇报实验结果 (包括实验数据, 曲线, 可视化图等), 并对结果进行分析;



通过实验结果我们发现, 风景图片的 cartoon 风格化是比较成功的, 其中城市风景图片和背景偏绿的风景图片效果最为出众, 但是对于人物和动物的 cartoon 风格化效果不好。



对相同图片的 **cartoon** 风格化和原作者在论文的结果图片对比，同样是新海诚的风格，我们的图片效果更加朦胧，而且感觉图片噪点很大，线条也不如原作者的清晰。

分析：经过分析我们得出，对于人物和动物的 **cartoon** 风格化效果不好，可能是因为我们的动漫图片的数据集不够全面，只有少量出现人物和动物的图片，并且人物和动物在图片中也占一部分大小，并且很多特征也因为在 **resize** 图片的时候被压缩的失真了。而对于我们的效果不如原作者的清晰，可能是因为我们的动漫数据集在 **resize** 的过程失真严重，很多边缘、线条这些细节不够明显。

4 总结与展望

4.1 总结自己的工作

屠怡泽：对于我制作的动漫数据集还需要进一步提高，在实验的过程中，我只是按照原先的方法，通过增加电影数量来增大数据集，没有通过其他手段甚至是论文中提到的裁剪手段来增大数据集和提高数据集的质量

徐敏超：这次实验我们本来是打算用 `tensorflow` 实现的，但是在配置完环境后还是出现了很多的错误，而我还是对这方面不够熟练，导致训练时一直出现显存不够的问题，最后还是在师兄的帮助下才完成，应该加强对这方面的理解。最后训练出来的模型虽然在一些方面效果不错，但仍然有局限性，可以考虑加大数据集或者是重新裁剪照片，之前的缩放可能导致局部细节不明显，如果是通过裁剪来做数据集的话可能特征保存的更好，模型效果也会更好。

4.2 指出未来可能的改进方法：

对于动漫数据集重新处理，可以对从电影中逐帧提取的图片进行裁剪，裁剪成 $256*256$ 大小的图片，这样既增加的数据集的数量，又保证的动漫图片的清晰程度。

5 小组成员工作介绍

屠怡泽：查找资料、制作训练集、调试代码、反复实验（没有效果的）

徐敏超：查找资料、配置环境、实验测试

致谢

感谢曾焕滨师兄的帮助

参考文献

[1] <https://github.com/znxlm/pytorch-CartoonGAN>