



北京大学

# 机器学习期末作业

题目：预训练图像生成模型原理

组 别：第六组

组 员：阮洁、周昭坤、杜金浩、郭博菲、

季帅健、马子平、潘鼎、余旺博、

干皓丞、徐华阳

院 系：信息工程学院

专 业：计算机应用技术

研究方向：图像预训练模型技术发展与解析

导 师：邹月娴 教授

二〇二二 年 一 月



## 目录

<b>第一章 绪论</b> .....	1
1.1 选题背景 .....	1
1.1.1 预训练模型.....	1
1.1.2 图像生成算法.....	1
1.1.3 预训练图像生成模型.....	4
1.2 小组成员及其分工 .....	4
1.3 研究目的 .....	4
1.4 相关文献 .....	5
<b>第二章 文献阅读理解</b> .....	7
2.1 基于 GAN 的图像生成模型 .....	7
2.1.1 朴素 GAN 的基本模型 .....	7
2.1.2 GAN 的优缺点.....	8
2.1.3 GAN 的一些经典变种.....	8
2.1.4 GAN 图像生成.....	9
2.2 条件图像生成.....	11
2.2.1 Image-to-Image Translation with Conditional Adversarial Networks .....	11
2.2.2 Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks.....	13
2.3 高分辨率图像生成 .....	14
2.3.1 A Style-Based Generator Architecture for Generative Adversarial Networks ...	14
2.3.2 Analyzing and Improving the Image Quality of StyleGAN .....	18
2.3.3 Alias-Free Generative Adversarial Networks.....	23
2.4 预训练图像生成模型的应用 .....	26
2.4.1 GAN Inversion: A Survey.....	26
2.4.2 Encoding in Style: a StyleGAN Encoder for Image-to-Image Translation .....	28
2.4.3 A Style-Based Generator Architecture for Generative Adversarial Networks ...	30
2.4.4 ReStyle: A Residual-Based StyleGAN Encoder via Iterative Refinement.....	34
2.4.5 Designing an Encoder for StyleGAN Image Manipulation .....	35

<b>第三章 研究方法</b>	39
3.1 pSp、e4e 与 Restyle 的流程	39
3.1.1 pSp 框架流程	39
3.1.2 e4e 的 Designing an Encoder for StyleGAN Image Manipulation	40
3.1.3 ReStyle: A Residual-Based StyleGAN Encoder via Iterative Renement	42
3.2 MetaFormer 及其相关原理介绍	44
3.2.1 MetaFormer	44
3.2.2 PoolFormer	44
3.2.3 PoolFormer 与 ReStyle-GAN 模型的结合	44
3.3 实验条件、客观结果与分析	45
3.3.1 实验细节	45
3.3.2 对比模型	46
3.4 主观结果、主观分析	46
3.4.1 ReStyle 重建人脸图像	46
3.4.2 不同编码器提取的人脸特征编码	48
<b>第四章 总结与分析</b>	49
<b>参考文献</b>	51

# 第一章 绪论

## 1.1 选题背景

### 1.1.1 预训练模型

深度神经网络，如卷积神经网络（CNN），循环神经网络（RNN），图形神经网络（GNN）和注意力神经网络（Transformer）等已广泛应用于各种人工智能任务，包括计算机视觉、自然语言处理、语音信号处理、数据挖掘与推荐系统等。与以前主要依赖手工特征和统计方法的非神经网络模型不同，深度神经网络模型可以隐式地从数据中提取应用于特定任务的特征，从而摆脱复杂的特征工程的桎梏，降低任务的复杂度。然而，尽管深度神经网络取得了令人振奋的成功，其强大的特征提取能力是建立在足够的训练数据的基础上的。由于深度神经网络通常具有大量参数，因此在没有足够训练数据的情况下，网络容易产生过拟合的现象，造成很差的推理效果。考虑到这一问题，在研究者们致力于改进神经网络的架构的同时，也有大量的工作投入到应用于AI任务的高质量数据集的构建上。然而，为特定任务量身打造的数据集的构建既昂贵又耗时。因此，如何利用有限的人工标注数据，为特定任务训练有效的深层神经模型就成为了一个研究热点。而迁移学习和预训练模型的提出，为小样本学习提供了一种解决思路。

人类可以学习用很少的样本来解决新问题，而不是用大量数据从头开始训练模型，主要原因在于人类可以利用以前学到的知识来处理新问题。受此启发，研究者将迁移学习建模为两个阶段：从一个或多个源任务捕获知识的预训练阶段，以及将捕获的知识迁移到目标任务的微调阶段。由于在预训练阶段获得了丰富的知识，微调阶段可以使模型能够很好地处理样本受限的目标任务。迁移学习为缓解数据缺乏的挑战提供了一种可行的方法，例如，通过在大规模人类注释视觉识别数据集ImageNet上对CNN进行预训练，得益于ImageNet中分布的强大视觉知识，使用少量数据微调这些预先训练的CNN就可以使其在下游任务中表现良好，因此，预训练技术被广泛应用于多种CV任务中，如图像分类、目标检测、图像分割和图像生成等。

### 1.1.2 图像生成算法

图像生成作为一个由来已久的研究课题，由于其广泛的应用，多年来一直是计算机视觉和计算机图形学的一个重要的研究领域。在神经网络出现之前，基于传统方法的图像生成和风格化算法有：

1. 基于区域的图像生成技术。基于区域的图像生成技术结合了语义分割等技术，

能够根据图像不同的语义区域的内容来适配生成。早期基于区域的算法利用区域的形状来指导笔触的布局，在图像的不同语义区域产生不同的笔画模式。之后，研究者又进一步提出了一种基于简单区域的图像处理算法来控制图像的生成，具体做法是通过用标准形状区域替换需要进行语义分割才能得到的复杂区域来创建简化的形状合成效果，并在生成过程中针对不同尺度的细节调整渲染区域的大小。用这种方法进行图像合成，生成的图像质量不高，并且效率很低，没有泛化能力，一次只能处理一张图像。

2. 基于样例的图像合成技术。基于样例的图像合成技术的思路是学习图像样本对之间的映射关系。这种技术是由 Hertzmann[1] 等首创的，他们提出了一种名为图像类比的算法框架。图像类比旨在以监督学习的方式学习源图像和目标图像之间的映射关系。图像类比的训练集包括成对的源图像和具有特定风格的图像，通过在训练集上训练，图像类比算法从样例训练对中学习从源域到目标域的变换，最终实现图像的风格转变。然而，成对的训练数据在实践中通常是不可求的，这就影响了该技术的普适性。此外，图像类比只能学习到较低层次图像特征，这也限制了它的性能。上述没有使用神经网络的图像生成算法虽然能够生成某些特定风格和内容的图像，但它们在灵活性、风格多样性和生成图像结构的准确性上存在很大局限，因此，需要新的算法来解决这些问题，由此基于深度学习的图像生成技术应运而生。而基于深度学习的图像生成分为基于 GAN 的图像生成和基于优化的图像生成两种。

### (1) 基于优化的图像生成

随着卷积神经网络成为计算机视觉任务的主流框架，Gatys[2] 等人首先尝试了使用卷积神经网络对自然图像进行生成，提出了使用在分类任务上预训练好的卷积神经网络作为特征抽取器，分别用其浅层和深层特征响应对图像的内容和风格进行统计建模，来指导图像生成。值得注意的是，这也是最早将预训练技术应用到图像生成领域的范例。他们的实验结果表明，在分类任务上预训练好的卷积神经网络能够从任意自然图像中提取内容信息，从艺术作品中提取风格信息。基于这一发现，Gatys 等人首次提出利用预训练的分类卷积神经网络的输出特征来重组自然图像的内容和艺术作品的风格，通过反向传播不断优化图像，使生成图像的特征分布逐渐逼近卷积神经网络的特征响应的分布。他们提出的算法开创了利用预训练网络进行图像生成的先河，引起了学术界和工业界的广泛关注，许多科研人员进行了大量的研究来改进或扩展这种新颖的图像生成算法，由此产生了两个主要分支：

a. 基于优化图像的在线图像生成算法：它的基本思想是首先用卷积神经网络从风格和内容图像中分别提取风格信息和内容信息，再将它们重组得到优化目标，然后通过反向传播等优化算法迭代地重建图像以匹配优化目标，最后生成风格化的图像。一般来说，不同的基于图像优化的生成算法具有相同的原理，区别仅仅在于对视觉特征

建模的方式。这种算法因为每次只训练一张图像，所以训练出的图像效果较好，但效率太低，并且不具备泛化能力，因为整个训练过程是在图像上进行的，不能得到通用的模型。

b. 基于优化模型的离线图像生成方法：为了解决在线优化的效率问题和泛化能力问题，Johnson 等提出了在训练好 VGG 网络前加上一个卷积神经网络，固定 VGG 网络的参数，训练卷积神经网络，使其能够对输入图像进行重建。这种思路实际上是将预训练的 VGG 提取的特征作为损失函数来训练卷积神经网络的参数。通过这种方式训练好的卷积神经网络能够实时转换图像，具有极高的效率和较好的泛化能力。延续这种思路，后续诞生出了用一种模型生成具有多种风格的图像的算法以及用一种模型生成任意风格的图像的算法。

## (2) 基于生成对抗网络的图像合成算法

基于生成对抗网络的图像合成算法属于基于神经网络的图像合成方法的一种，它有两种生成方式：随机噪声生成和条件生成。由于随机生成难以控制，难以应用到实际任务中，因此本作业主要研究条件生成，即给模型输入一幅图像，让模型合成另一幅具有该作业需要的性质的图像。基于 GAN 的图像合成的思路也是训练一个深度神经网络，使其能够将输入图像转换成目标风格图像。它与一般的深度图像合成方法的区别在于训练方式的不同，生成对抗网络需要同时训练一对生成器和判别器，生成器负责实现图像的合成，判别器负责对生成效果进行评价，二者通过对抗训练达到纳什均衡，最终能够得到最优的生成器，也就是本作业需要的模型。常用的基于生成对抗网络的条件图像生成架构有 pix2pix, CycleGAN 等。pix2pix 的生成器采用 U-Net 结构，对输入图像进行下采样后再进行上采样，并且在对应层之间加入了跳跃连接；判别器采用一种称为 PatchGAN 的结构，用卷积神经网络对输入图片进行下采样，最后得到的特征图代表输入图像的评分矩阵，特征图上的每一点对应其感受野上的图像块的评分。采用 pix2pix 进行人图像合成，生成的图像具有很高的质量，但其缺陷在于需要成对的训练数据，而成对的图像数据往往很难获取的。CycleGAN 的提出解决了这一问题。CycleGAN 的生成器采用 ResNet 结构，对图像进行下采样后用一系列残差块进行特征保持，之后再进行上采样。判别器采用了和 pix2pix 一样的 PatchGAN 结构。它和 pix2pix 的主要区别在于损失函数和训练方式的不同，CycleGAN 中采用了循环一致性损失函数，通过训练两套生成器和判别器，用循环一致性损失函数约束两个生成器的输出，最终实现不同域间的双向迁移。采用这种方法，不需要成对的训练数据也可以实现准确高效的图像生成。

### 1.1.3 预训练图像生成模型

预训练图像生成模型往往指 StyleGAN[6] 及其改进模型，这些模型相较其他生成对抗网络而言，摒弃了相对传统的输入层，转而使用非线性映射网络，通过对风格卷积块参数进行调制的方式实现对生成图像特征的精细控制，例如人脸生成中的发型、肤色等。然而生成图像的极高质量和分辨率也意味着这类生成模型需要大量的计算资源进行训练。考虑到这类模型具有精细控制的潜力，研究人员更多地关注如何将其作为预训练模型投入到其他应用中，以提升其他任务中深度模型的性能，而非局限于无条件的生成任务。

目前而言，预训练图像生成模型的应用，一方面在于如何利用其在训练过程中学到的知识，例如在图像超分辨率、上色、去噪等任务中引入预训练图像生成模型中的先验知识，使图像细节更丰富、使图像结构更合理；另一方面在于训练编码器将待编辑图像映射到隐向量空间中，以通过低维空间的简单操作实现图像转换等复杂的语义操作，甚至视频语义编辑等更加复杂的操作。

## 1.2 小组成员及其分工

1. 阮洁: 文献阅读、撰写报告、PPT 制作、实验
2. 周昭坤: 文献阅读、撰写报告、PPT 制作、实验
3. 杜金浩: 文献阅读、撰写报告、PPT 制作、实验
4. 郭博菲: 文献阅读、撰写报告、PPT 制作、实验
5. 季帅健: 文献阅读、撰写报告、PPT 制作、实验
6. 马子平: 文献阅读、撰写报告、PPT 制作、实验
7. 潘鼎: 文献阅读、撰写报告、PPT 制作、实验
8. 余旺博: 文献阅读、撰写报告、PPT 制作、实验
9. 干皓丞: 文献阅读、撰写报告、PPT 制作、实验
10. 徐华阳: 文献阅读、撰写报告、PPT 制作、实验

## 1.3 研究目的

1. 了解预训练模型的原理和基本应用，掌握常用的开源预训练模型的使用方法。
2. 了解基于生成对抗网络的图像生成的基本原理，能够训练经典的生成对抗网络，如 Pix2Pix, CycleGAN 等，学会运用其开源的预训练模型进行图像生成。
3. 了解预训练的分类网络在图像生成领域中的应用，例如使用预训练的 VGG 分类网络对图像进行内容和风格提取，实现图像风格迁移。

4. 了解 StyleGAN 三部曲的原理及应用，能够使用其开源的预训练模型进行图像生成。
5. 了解基于预训练 StyleGAN 的应用，如图像翻译，风格迁移，图像修复等。能够将预训练的 StyleGAN 应用到特定的底层视觉任务中。

## 1.4 相关文献

本作业所阅读得的参考文献条列如下：

表 1.1 作业阅读文献

作者	年份	来源
Aaron Hertzmann et al. <sup>[1]</sup>	2001	ACM
Leon Gatys et al. <sup>[2]</sup>	2016	CVPR
J.Johnson et al. <sup>[3]</sup>	2016	ECCV
Gu J et al. <sup>[4]</sup>	2020	CVPR
Alaluf Y et al. <sup>[5]</sup>	2021	ICCV
Ian J.Goodfellow et al. <sup>[6]</sup>	2014	NIPS
Alec Radford et al. <sup>[7]</sup>	2016	ICLR
Phillip Isola et al. <sup>[8]</sup>	2017	CVPR
Jun-Yan Zhu et al. <sup>[9]</sup>	2017	ICCV
Tero Karras et al. <sup>[10]</sup>	2019	CVPR
Tero Karras et al. <sup>[11]</sup>	2019	CoRR
Tero Karras et al. <sup>[12]</sup>	2021	CoRR
Weihao Xia et al. <sup>[13]</sup>	2021	CoRR
Omer Tov et al. <sup>[14]</sup>	2021	ACM
Elad Richardson et al. <sup>[15]</sup>	2021	CVPR
Yuval Alaluf et al. <sup>[16]</sup>	2021	CoRR
Omer Kafri et al. <sup>[17]</sup>	2021	CoRR
Radford A et al. <sup>[18]</sup>	2015	arXiv
Arjovsky M et al. <sup>[19]</sup>	2017	ICML
Ishaan Gulrajani et al. <sup>[20]</sup>	2017	arXiv
Tero Karras et al. <sup>[21]</sup>	2017	arXiv
Mirza M et al. <sup>[22]</sup>	2014	arXiv
Antoniou A et al. <sup>[23]</sup>	2017	arXiv
Gu J et al. <sup>[24]</sup>	2020	CVPR



## 第二章 文献阅读理解

### 2.1 基于 GAN 的图像生成模型

#### 2.1.1 朴素 GAN 的基本模型

生成对抗网络（Generative adversarial network, GAN）自 2014 年由 Ian Goodfellow 等人提出后，就越来越受到学术界和工业界的重视。而随着 GAN 在理论与模型上的高速发展，它在计算机视觉、自然语言处理、人机交互等领域有着越来越深入的应用，并不断向着其它领域继续延伸。其中，GAN 在图像生成上取得了巨大的成功，这取决于 GAN 在博弈下不断提高建模能力，最终实现以假乱真的图像生成。

GAN 受博弈论中的零和博弈启发，将生成问题视作判别器和生成器这两个网络的对抗和博弈：生成器从给定噪声中（一般是指均匀分布或者正态分布）产生合成数据，判别器分辨生成器的输出和真实数据。前者试图产生更接近真实的数据，相应地，后者试图更完美地分辨真实数据与生成数据。由此，两个网络在对抗中进步，在进步后继续对抗，由生成网络得到的数据也就越来越完美，逼近真实数据，从而可以生成想要得到的数据（图片、序列、视频等）。

GAN 包含有两个模型，一个是生成模型（generative model），一个是判别模型（discriminative model）。生成模型的任务是生成看起来自然真实的、和原始数据相似的实例。判别模型的任务是判断给定的实例看起来是自然真实的还是人为伪造的（真实实例来源于数据集，伪造实例来源于生成模型）。

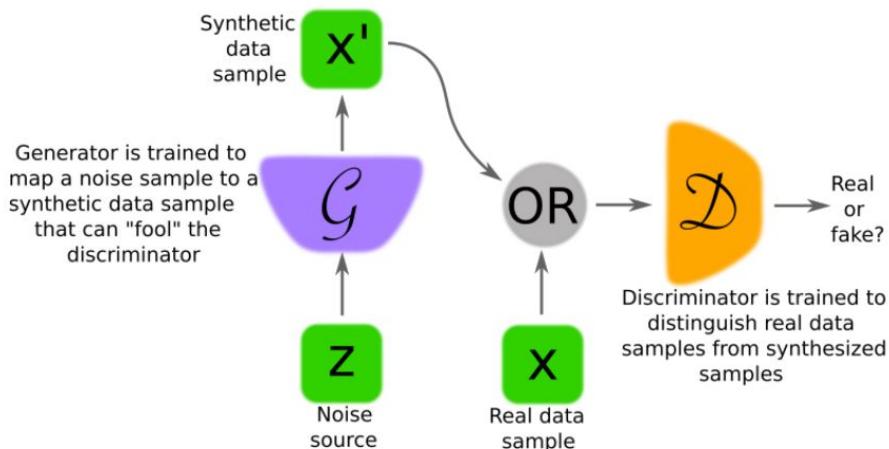


图 2.1 GAN 的基本结构

理论上 GAN 可以将任意的分布作为输入，如上图所示，输入  $z$  为随机噪声，在实

验中此研究多取  $z(0, 1)$  或  $z[-1, 1]$  的均匀分布作为输入。生成器  $G$  的参数为  $\theta$ , 输入  $z$  在生成器下得到  $x' = G(z; \theta)$ , 输出可以被视为从分布中抽取的样本  $G(z; \theta)p_g$ 。对于训练样本  $x$  的数据分布为  $p_{data}$ , 生成器  $G$  的目标是使  $p_g$  近似  $p_{data}$ , 判别器  $D$  的目标则是尽可能区分生成样本和真实样本的真假, 通过最大-最小博弈来进行训练, 这种博弈可公式化为:

$$\min \max V(D, G) = E_{x \sim p_{data}(x)} [\log D(x | y)] + E_{z \sim p_z(z)} [\log(1 - D(G(z | y)))] \quad (2.1)$$

其中第一项的  $\log D(x)$  表示判别器对真实数据的判断, 第二项  $\log(1 - D(G(z)))$  表示对合成数据的判断。通过这样一个最大最小 (*Max – min*) 博弈, 循环交替地分别优化  $G$  和  $D$  来训练所需要的生成式网络与判别式网络, 直到到达 Nash 均衡点。

### 2.1.2 GAN 的优缺点

优势:

1. 根据实际的结果, GAN 看上去可以比其它模型产生更好的样本 (图像更锐利、清晰)。
2. 生成对抗式网络框架能训练任何一种生成器网络 (这是理论上的, 而实践中, 用 REINFORCE 来训练带有离散输出的生成网络非常困难)。大部分其他的框架需要该生成器网络有一些特定的函数形式, 比如输出层是高斯的。
3. 不需要设计遵循任何种类的因式分解的模型, 任何生成器网络和任何鉴别器都会有用。
4. 无需利用马尔科夫链反复采样, 无需在学习过程中进行推断 (Inference), 回避了近似计算棘手的概率的难题。

劣势:

1. 判别器越好, 生成器的梯度消失越严重, 这样会导致在网络训练上很多时候生成器的参数基本上不会发生改变。
2. 由于网络是对抗式的, 常常会造成训练时模型的崩溃 (collapse mode), 在训练时往往需要权衡训练的生成器与鉴别器的参数来防止崩溃的发生。这样在实际的应用上也带了很多不便。

### 2.1.3 GAN 的一些经典变种

#### 1. DCGAN:

其论文名为 Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, 而 DCGAN 是继 GAN 之后比较好的改进, 其主要的改进主要是

在网络结构上，到目前为止，DCGAN 的网络结构还是被广泛的使用，DCGAN 极大的提升了 GAN 训练的稳定性以及生成结果质量。

DCGAN 对上述的 G 和 D 用了两个卷积神经网络（CNN），同时对卷积神经网络的结构做了一些改变，以提高样本的质量和收敛的速度，这些改变有取消所有 pooling 层。G 网络中使用转置卷积（transposed convolutional layer）进行上采样，D 网络中用加入 stride 的卷积代替 pooling；在 D 和 G 中均使用 batch normalization；去掉 FC 层，使网络变为全卷积网络；G 网络中使用 ReLU 作为激活函数，最后一层使用 tanh；D 网络中使用 LeakyReLU 作为激活函数。

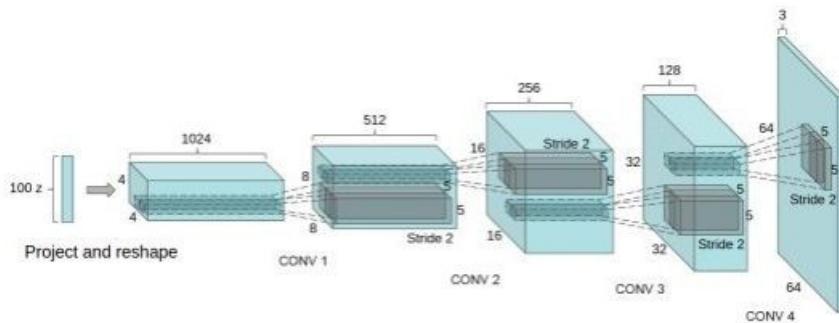


图 2.2 DCGAN

## 2. WGAN 和 WGAN-GP

WGAN 也是一篇经典，其名  $\square$  Wasserstein GAN，WGAN 主要从损失函数的角度对 GAN 做了改进，损失函数改进之后的 WGAN 即使在全链接层上也能得到很好的表现结果，具体的来说，WGAN 对 GAN 的改进有判别器最后一层去掉 sigmoid；生成器和判别器的 loss 不取 log；对更新后的权重强制截断到一定范围内，比如  $[-0.01, 0.01]$ ，以满足论文中提到的 lipschitz 连续性条件；论文中也推荐使用 SGD，RMSprop 等优化器，不要基于使用动量的优化算法，比如 adam。

而 WGAN-GP 论文为 Improved Training of Wasserstein GANs 之前的 WGAN 虽然理论上有了极大贡献，但在实验中却发现依然存在着训练困难、收敛速度慢的问题，这时 WGAN-GP 被提出来，它的贡献是提出了一种新的 lipschitz 连续性限制手法—梯度惩罚，解决了训练梯度消失梯度爆炸的问题；比标准 WGAN 拥有更快的收敛速度，并能生成更高质量的样本；提供稳定的 GAN 训练方式，几乎不需要太多调参，成功训练多种针对图片生成和语言模型的 GAN 架构。

### 2.1.4 GAN 图像生成

根据不同的 GAN 所拥有的生成器和判别器的数量，可以将 GAN 图像生成的方法概括为三类：直接方法，迭代方法和分层方法。

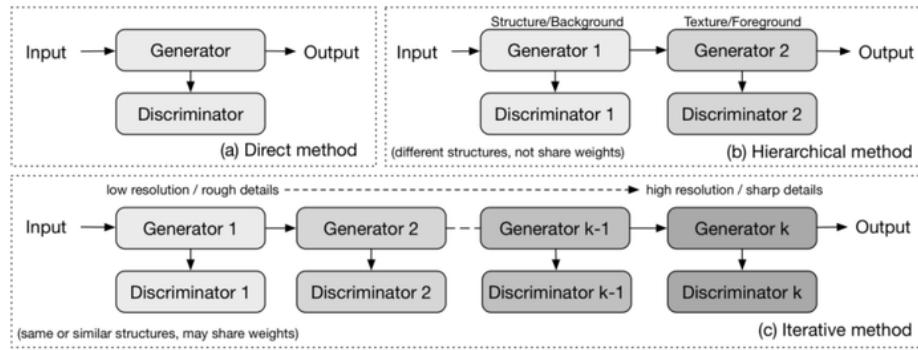


图 2.3 GAN 图像生成

### 1. 直接法

早期的 GANs 都遵循在其模型中使用一个生成器和一个判别器的原理，并且生成器和判别器的结构是直接的，没有分支。如 GAN、DCGAN、ImprovedGAN、InfoGAN、f-GAN 和 GANINT-CLS。这类方法在设计和实现上比较容易，通常也能得到良好的效果。

### 2. 分层法

分层法的主要思想是将图像分成两部分，如“样式和结构”和“前景和背景”，然后在其模型中使用两个生成器和两个鉴别器，其中不同的生成器生成图像的不同部分，然后再结合起来。两个生成器之间的关系可以是并联的或串联的。以 SS-GAN 为例，其使用两个 GAN，一个 Structure-GAN 用于生成表面结构，然后再由 Style-GAN 补充图片细节，最后生成图片，整体结构如下所示：

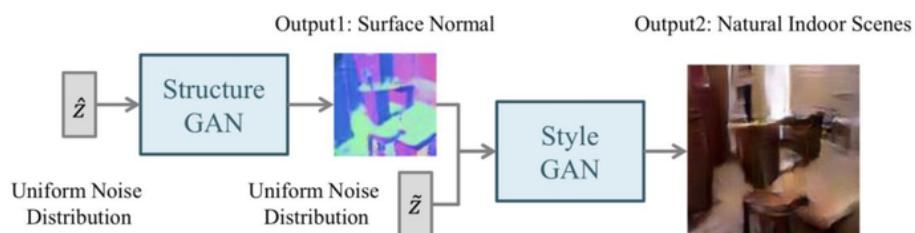


图 2.4 SS-GAN 的分层结构

### 3. 迭代法

迭代法使用具有相似或甚至相同结构的多个生成器，经过迭代生成从粗到细的图像，而以 LAPGAN 为例，LAPGAN 中的多个生成器执行相同的任务：最低级别的生成器仅将噪声向量作为输入并输出图像，而其他生成器都从前一个生成器获取图像并将噪声矢量作为输入，这些生成器结构的唯一区别在于输入/输出尺寸的大小，每一次迭代后的图像都拥有更多清晰的细节。

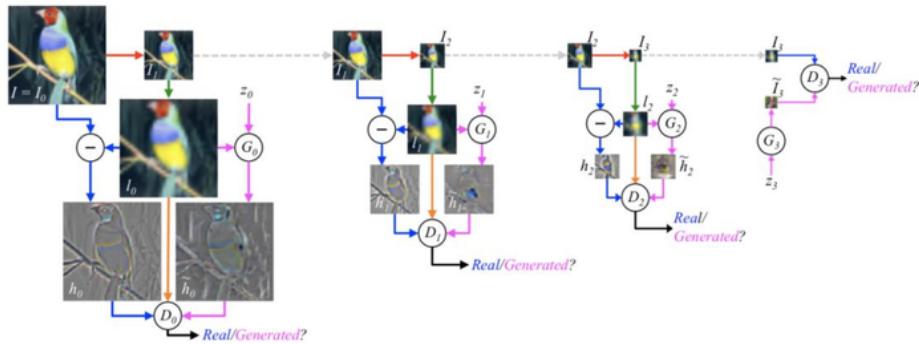


图 2.5 LAPGAN 的迭代结构

## 2.2 条件图像生成

如果 GAN 的生成器和判别器可以以某些额外的信息为条件，而非仅仅由正态采样的噪声控制生成，那么生成对抗网就可以扩展到条件模型。类似于条件概率，设  $y$  是任何类型的条件信息，如文本或其他形式的图像，该研究可以通过将  $y$  作为噪声  $z$  之外的附加信息输入判别器和生成器，引导生成效果。而将条件信息加入生成器和判别器中又分为两种情况：(1). 原始生成对抗网络的生成器的输入是噪声信号，从而类别标签可以和噪声信号组合作为隐藏空间表示；(2). 原始生成对抗网络的判别器输入是图像数据（真实图像和生成图像），同样需要将类别标签和图像数据进行拼接作为判别器输入。从而生成对抗网络的目标函数变为：

$$\min \max V(D, G) = E_{x \sim p_{\text{data}}(x)} [\log D(x | y)] + E_{z \sim p_z(z)} [\log (1 - D(G(z | y)))] \quad (2.2)$$

条件生成对抗网络的网络结构相对于原始生成对抗网络并没有变化，改变的仅仅是生成器和判别器的输入数据，这就使得 CGAN 可以作为一种通用策略嵌入到其它的 GAN 网络中。经典的条件图像生成 GAN 网络有两种：

### 2.2.1 Image-to-Image Translation with Conditional Adversarial Networks

图像处理、计算机图形学和计算机视觉中的许多问题都可以归结为将输入图像“翻译”成相应的输出图像的问题。正如任何一种概念都可以用英语或法语表达一样，场景也可以被渲染为 RGB 图像、梯度场、边缘图、语义标签图等。与自动语言翻译问题类似，此研究将图像到图像的翻译问题定义为在给定足够的训练数据的情况下，将一个场景的一种可能表示翻译成另一种可能表示的问题。传统上，尽管这些任务都可以概括为从像素到像素的预测，但这些任务中的每一种都是用特定的方法处理的。而 pix2pix 的提出为解决这些问题提出了一个通用的框架。

## 1. 模型结构

pix2pix 的生成器采用 U-Net 结构，将输入原始图像编码再解码成另一种风格图像，从而实现图像翻译。判别器使用 PatchGAN 结构，作用为在输入原始图像的条件下，对于生成的图片判断为假，对于真实图片判断为真。生成器使用 U-Net 结构的原因主要在于对于图像翻译问题，输入和输出图像的纹理应该不同，但潜在的内容结构应该相似，这就需要生成器的输入和输出共享一些底层的信息，因此生成器使用 U-Net，因为它采用了跳越连接 (skip-connection) 的方法对编码器和解码器共享底层信息。

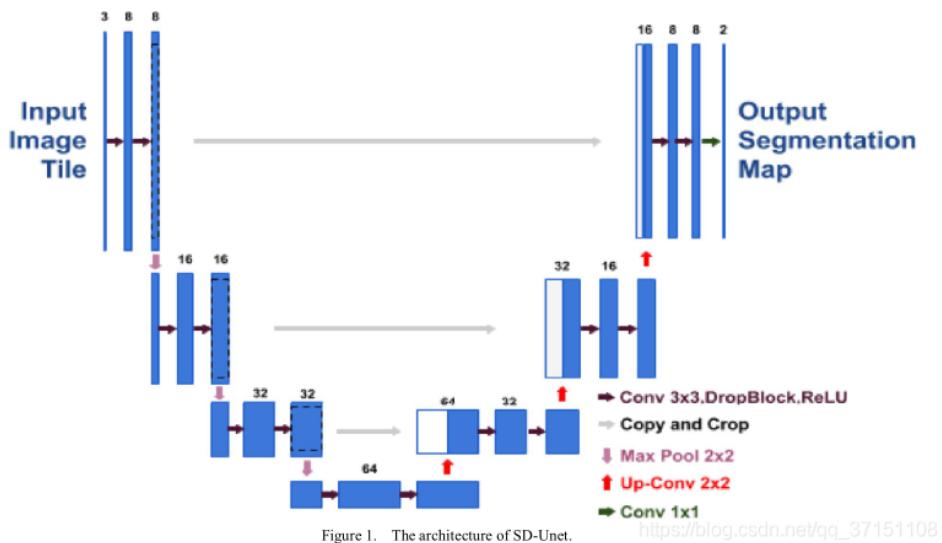


图 2.6 模型结构

判别器使用 PatchGAN 的结构的原因在于采用 L1 损失和 L2 损失进行重建会导致图像模糊，也就是说 L1 损失和 L2 损失并不能很好的恢复图像的高频部分，但能较好地恢复图像的低频部分。为了能更好地对图像的局部做判断，pix2pix 的判别器采用 patchGAN 的结构，也就是在判别过程中把图像等分成小块 (patch)，分别判断每个小块的真假，最后再取平均，得到的均值就是判别器对图像真伪的评分。文中提出的 PatchGAN 可以看成另一种形式的纹理损失。

原始 GAN 的判别器仅对整张输入图像输出一个评分，而 PatchGAN 使用卷积将输入映射为  $n \times n$  的评分矩阵，矩阵中每个元素对应原始图像中的一块小区域的评分，对评分矩阵求平均即得到整张图像的评分。这种计算方法能够更好地考察图像的细节，极大提升了判别器的性能。

## 2. 损失函数

生成对抗网络其实是一种相对于 L1 损失更好的判别准则或者说损失函数。一些情况下单独使用对抗损失效果更好，一些情况下对抗损失与 L1 损失配合起来效果更好。在 pix2pix 中，作者将 L1 损失和对抗损失结合使用，因为作者认为 L1 损失可以恢

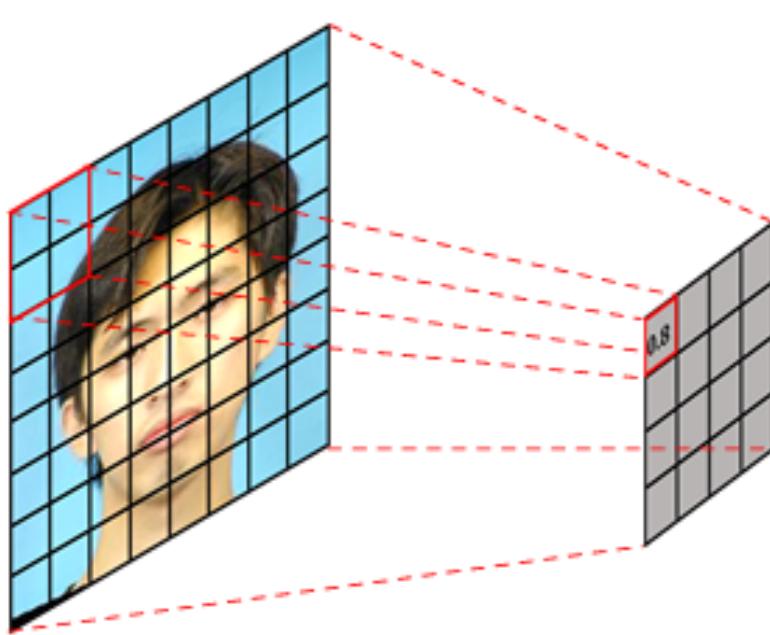


图 2.7 原始 GAN 的判别器

复图像的低频部分，而对抗损失可以恢复图像的高频部分，二者结合起来能够达到更好的生成效果。Pix2Pix 的损失函数公式为：

$$\min V_{GAN} + \alpha V_{L1} \quad (2.3)$$

### 2.2.2 Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

图像到图像翻译的目标是使用样本为对齐图像对的训练集来学习输入图像和输出图像之间的映射。然而，对于许多任务，成对的训练数据是不易获取的。为了解决这一问题，pix2pix 的作者又提出了一种在没有成对样本的情况下学习将图像从源域 X 翻译到目标域 Y 的方法，这就是 CycleGAN。它的目标是学习一个从 X 域到 Y 域的映射，使得来自的图像分布与使用对抗损失的分布 Y 无法区分。因为这个映射是高度欠约束的，作者把它和一个逆映射耦合起来，引入一个循环一致性损失 (Cycle consistency loss) 来约束  $F(G(X)) = X$  (反之亦然)。这样就能在不存在成对训练数据的情况下，完成不同风格域图像的翻译。

在模型结构上，CycleGAN 和 pix2pix 采用了相同的判别器 (PatchGAN)，但二者的生成器却略有不同。CycleGAN 的生成器采用类似于 ResNet 的跳跃连接结构，对图像下采样后用一系列残差块进行特征保持，再进行上采样。CycleGAN 的主要创新为提出了循环一致性损失 (Cycle Consistency Loss)，即同时训练两对生成器和判别器，最后

用循环一致性损失对两对生成器和判别器进行约束，使它们学习到的映射为互相之间的逆映射。

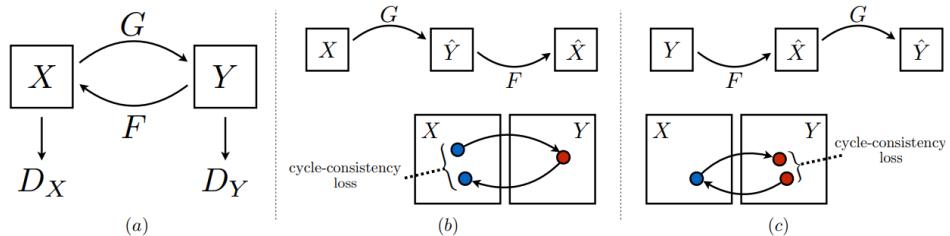


图 2.8 CycleGAN

循环一致性损失用公式表述为：

$$V_{cyc}(G, F) = E_{x \in p_{\text{data}}(x)} [\|F(G(x)) - x\|_1] + E_{y \in p_{\text{data}}(y)} [\|G(F(y)) - y\|_1] \quad (2.4)$$

## 2.3 高分辨率图像生成

### 2.3.1 A Style-Based Generator Architecture for Generative Adversarial Networks

#### 1. Background

NVIDIA 在 2017 年提出的 ProGAN 解决了生成高分辨率图像（如 1024×1024）的问题。ProGAN 的关键创新之处在于渐进式训练——从训练分辨率非常低的图像（如 4×4）的生成器和判别器开始，每次都增加一个更高的分辨率层。

#### 2. Motivation

与多数 GAN 一样，ProGAN 控制生成图像的特定特征的能力非常有限。这些属性相互纠缠，即使略微调整输入，会同时影响生成图像的多个属性。所以如何将 ProGAN 改为条件生成模型，或者增强其微调单个属性的能力，是一个可以研究的方向。

#### 3. Contribution

(1) 借鉴风格迁移，提出基于样式的生成器（style-based generator）。

- 实现了无监督地分离高级属性（人脸姿势、身份）和随机变化（例如雀斑，头发）
- 实现对生成图像中特定尺度的属性的控制。
- 生成器从一个可学习的常量输入开始，隐码在每个卷积层调整图像的“样式”，从而直接控制不同尺度下图像特征的强度。

(2) 实现了对隐空间（latent space）较好的解耦。

生成器将输入的隐码  $z$  嵌入一个中间的隐空间。因为输入的隐空间  $Z$  必须服从训练数据的概率密度，这在一定程度上导致了不可避免的纠缠，而嵌入的中间的隐空间

$W$  不受这个限制，因此可以被解耦。

(3) 提出了两个新的量化隐空间解耦程度的方法

感知路径长度和线性可分性。与传统的生成器体系结构相比，新的生成器允许更线性、更解耦地表示不同的变化因素。

(4) 提出了新的高质量的人脸数据集 (FFHQ, 7 万张 1024×1024 的人脸图片)

#### 4. Framework

下图为 stylegan 生成器与传统 GAN 网络的对比。其中不同之处包括：

- 移除了传统的输入 (remove traditional input)
- 映射网络 (Mapping Network)
- 样式模块 (style modules, AdaIN, 自适应实例归一化)
- 随机变化 (Stochastic variation, 通过加入噪声为生成器生成随机细节)

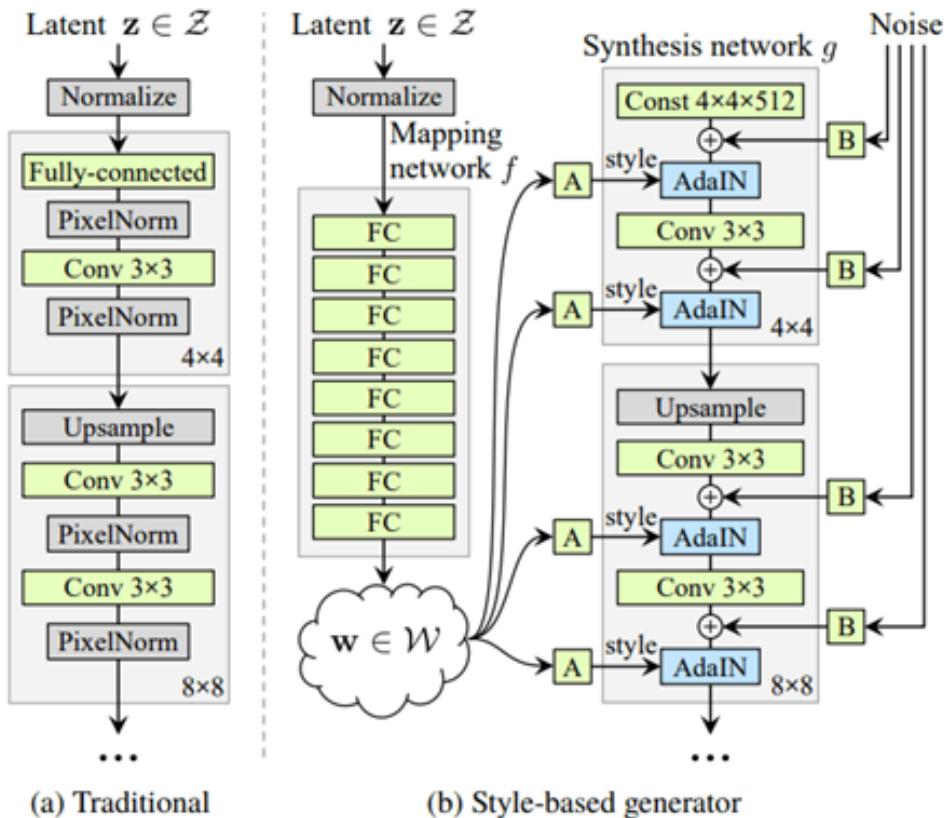


图 2.9 stylegan

#### 5. Results

(1) 生成的图像的质量

训练数据集分别是 CelebA-HQ 和 FFHQ。其中 FFHQ 数据集 (本文提出的) Flickr-Faces-HQ(FFHQ)，包含 7 万张高分辨率的人脸图像 (2014×1024)。评估指标

为 FID(Fréchet Inception Distance，越小越好)。

Method	CelebA-HQ	FFHQ
A Baseline Progressive GAN [28]	7.79	8.04
B + Tuning (incl. bilinear up/down)	6.11	5.25
C + Add mapping and styles	5.34	4.85
D + Remove traditional input	5.07	4.88
E + Add noise inputs	<b>5.06</b>	4.42
F + Mixing regularization	5.17	<b>4.40</b>

图 2.10 生成的图像的质量

## (2) 随机变化

显示了在相同底层图像输入不同的噪声实现的随机变化。噪声只影响随机方面，而保留了整体结构和身份、面部等高级特征。

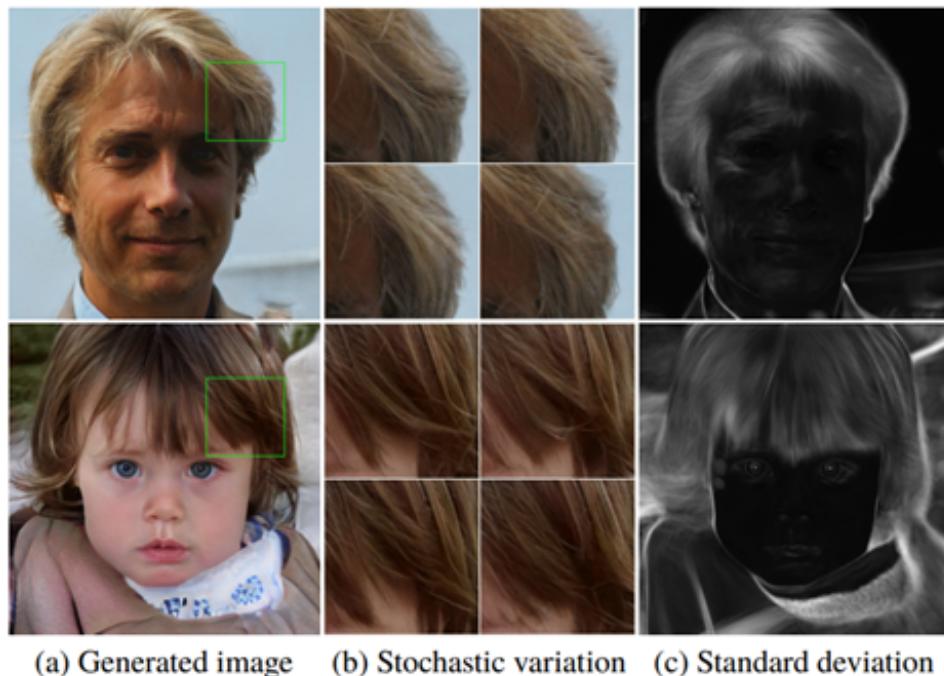


图 2.11 随机变化

## (3) 风格混合

为了进一步鼓励样式进行本地化，采用混合正则化，即在训练过程中使用两个随机潜码而不是一个来生成给定百分比的图像。在生成这样的图像时，只需在合成网络中随机选择的位置从一个潜在代码切换到另一个潜在代码（称为样式混合的操作）。具体来说，通过映射网络运行两个潜在代码  $z_1, z_2$ ，并让相应的  $w_1, w_2$  控制样式，以

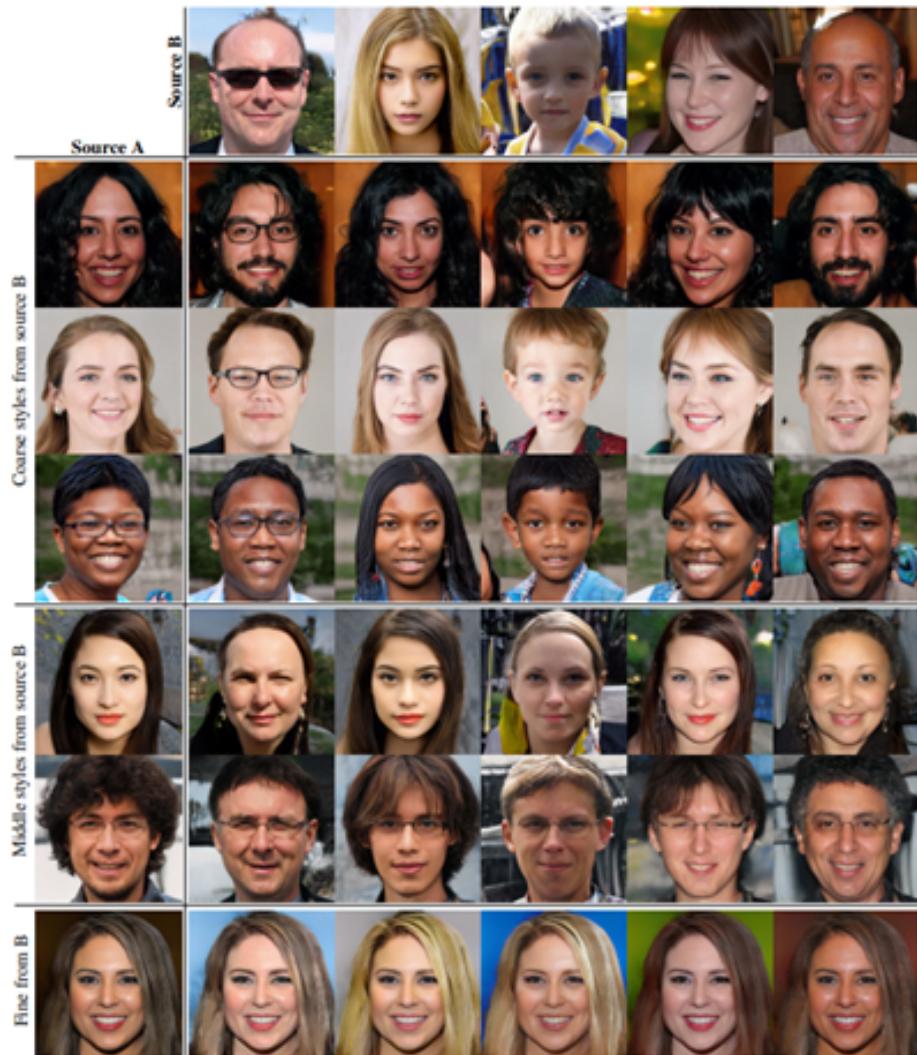


图 2.12 风格混合

便  $w_1$  在交叉点之前应用,  $w_2$  在交叉点之后应用。这种正则化技术可防止网络假设相邻样式相关。

## 6. Conclusion

基于样式的生成器, 能生成更高质量的高分辨率图像; 实现了无监督地分离高级属性(人脸姿势、身份)和随机变化(例如雀斑, 头发), 实现对生成图像中特定尺度的属性的控制。通过 style 控制高级属性, 通过不同层的 style 实现对不同尺度的特征的控制; 通过噪声控制随机变化降低了隐空间的耦合性; 通过映射网络生成中间隐空间 (intermediate latent space), 中间隐空间降低了输入隐空间的纠缠; 提出两种新的量化隐空间耦合性的指标-感知路径长度和线性可分性; 对高级属性与随机效应的分离以及中间隐空间的线性的研究, 提高了对 GAN 合成的理解和可控性。

### 2.3.2 Analyzing and Improving the Image Quality of StyleGAN

该论文是 StyleGAN2 即在 StyleGAN 的基础上改进得到的。本文指出和分析了几个特征伪影，并在模型架构和训练方法上进行了调整以解决该问题。重新设计了生成器归一化，并正则化生成器以促进隐码到图像的映射有更好的条件。除了改善了图像质量，这个路径长度正则化器还带来了额外的好处，使得生成器更容易反转。本文进一步可视化生成器如何很好地利用其输出分辨率，并确定一个容量问题，激励该研究训练更大的模型，以获得额外的质量改进。该工作成为当时人脸生成领域的 SOTA，而本文主要贡献如下：

#### 1. 修复了 StyleGAN 中存在的水滴伪影

StyleGAN 生成的大多数图像都呈现出典型的水滴形状的人工伪影，即使液滴在最终图像中可能不明显，它也存在于生成器的中间特征图中。异常开始出现在 64×64 分辨率附近，出现在所有特征图中，并在更高分辨率时变得越来越强。作者将问题定位到 adaptive instance normalization (AdaIN) 操作，该操作分别对每个特征图的均值和方差进行归一化，有可能破坏了特征彼此之间的信息。而解决方法：A. 改变网络结构 B. Weight demodulation。

#### 2. 对渐进式增长进行修正

渐进生成器导致了相位问题。逐渐增长的生成器似乎对细节有强烈的偏好。当牙齿或眼睛等特征在图像上平滑移动时，它们可能会停留在原位。作者重新考虑了渐进式增长，提出了一种替代设计——训练从专注低分辨率的图像开始，然后逐步将焦点转移到高分辨率——在训练期间不改变网络拓扑。

#### 3. PPL 成为正则化项

Fréchet inception distance (FID) 和 Precision and Recall (P&R) 是生成图像质量量化评估的两大指标，这两个指标都是基于分类器网络的，专注于纹理而不是形状，不能准确地捕捉图像质量的所有方面。(perceptual path length; PPL) 作为一种评估潜空间插值质量的一种方法，于形状的一致性和稳定性有关。在此基础上，本文对合成网络进行正则化，使其有利于平滑映射，并在质量上取得明显改进。为了抵消计算开销，作者还建议减少所有正则化的执行频率，不会影响其有效性。

#### 4. 更大的容量

更大的模型进一步提高生成质量

##### (1) 新的网络架构

- 是原始的 StyleGAN，其中 A 表示从 W 的一个仿射变换 B 是一个噪声广播操作。
- 更详细的 StyleGAN。通过显示权重和偏差，并将 AdaIN 操作分解为两个组成部分：归一化 (normalization) 和调制 (modulation)。

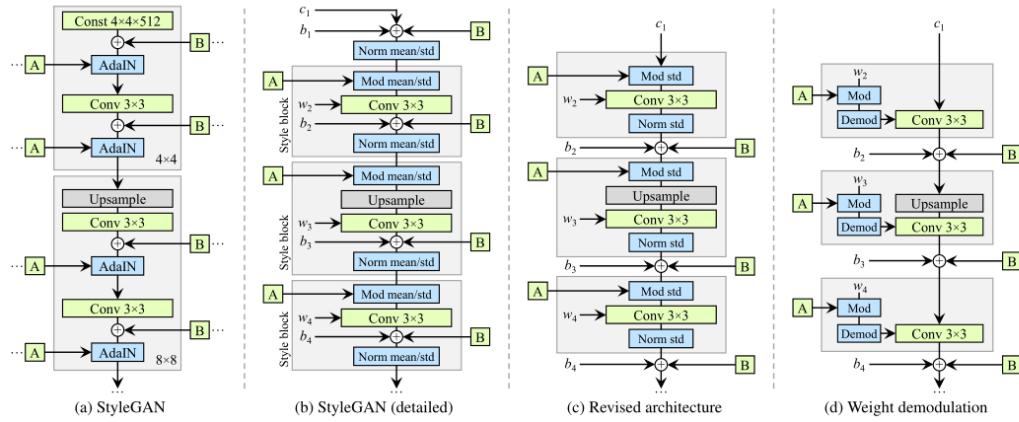


图 2.13 新的网络架构

- 对原始架构进行了调整修改。将  $b$  和  $B$  的相加移到了风格活动区域之外，可以获得更可预测的结果，并且只对标准差进行归一化（去掉了对均值的操作）。
- 修改后的架构使此研究能够用“解调”（demodulation）操作取代实例归一化，对特征图的修改改为对卷积权重的修改。

## (2) Weight demodulation

简单地去除归一化操作可以消除伪影，但是无法对风格混合（style mixing）有 scale-specific 级别的控制。本文提出了一个更好的方案能够在保留完全可控性的同时去掉伪影。将对特征图的一系列操作改为对权重的操作。特征图只经过卷积处理并添加噪声。缩放特征图改为缩放卷积权重（mod）：

$$w'_{ijk} = s_i \cdot w_{ijk} \quad (2.5)$$

$s_i$  是第  $i$  个输入特征图的缩放比例，经过缩放和卷积后，输出激活的标准差为：

$$\sigma_j = \sqrt{\sum_{i,k} w'_{ijk}^2} \quad (2.6)$$

demod 权重，旨在使输出恢复到单位标准差：

$$w''_{ijk} = w'_{ijk} / \sqrt{\sum_{i,k} w'_{ijk}^2 + \epsilon} \quad (2.7)$$

加一个  $\epsilon$  避免分母为 0。

尽管这种方式与 Instance Norm 在数学上并非完全等价，但是 weight demodulation 同其它 normalization 方法一样，使得输出特征图有着 standard 的 unit 和 deviation。

### (3) 图像质量与生成器的平滑程度

作者观察发现更小的 perceptual path length (PPL) 有更高的整体图像质量。作者假设在训练过程中，由于鉴别器对破碎图像进行惩罚，对生成器进行改进最直接的方法是有效地拉伸产生良好图像的潜空间区域。这将导致低质量的图像被压缩到快速变化的小潜空间区域。虽然这可以在短期内提高平均输出质量，但累积的失真会损害训练状态，进而损害最终图像质量。

- Lazy regularization : 每 16 个 batch 执行一次 R1 和路径长度正则化，大大降低计算成本和内存用量，而对结果无影响。
- 路径长度正则化: 无论 W 或图像空间方向如何，这些渐变应具有接近等长度，即小位移产生相同大小的变化。表示从潜在空间到图像空间的映射是良好的。路径长度正则化不但提高了图片的生成质量，而且使得生成器更平滑，生成的图片反转回 latent code 更容易了。

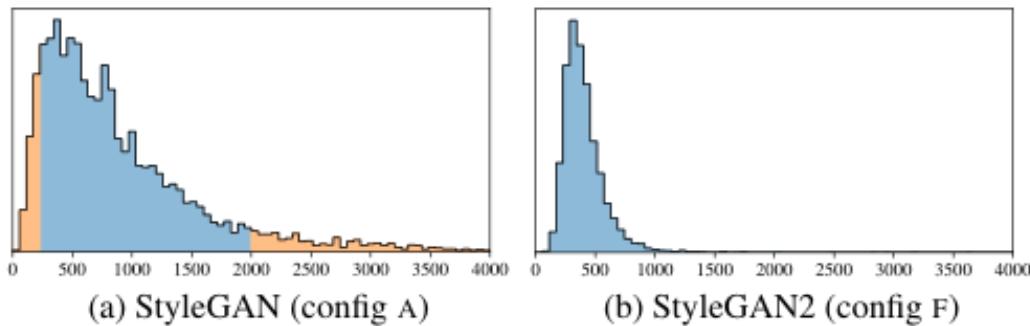


图 2.14 图像质量与生成器的平滑程度

StyleGAN2 (config F) 极大地改善了 PPL 的分布，使之更加紧凑。但是结构化程度低于人脸 FFHQ 的 LSUN Car 数据集中的 FID 和 PPL 之间存在平衡。PPL 降低反而导致 FID 升高。如下图 config D 所示。

Configuration	FFHQ, 1024×1024				LSUN Car, 512×384			
	FID ↓	Path length ↓	Precision ↑	Recall ↑	FID ↓	Path length ↓	Precision ↑	Recall ↑
A Baseline StyleGAN [21]	4.40	212.1	<b>0.721</b>	0.399	3.27	1484.5	<b>0.701</b>	0.435
B + Weight demodulation	4.39	175.4	0.702	0.425	3.04	862.4	0.685	0.488
C + Lazy regularization	4.38	158.0	0.719	0.427	2.83	981.6	0.688	0.493
D + Path length regularization	4.34	<b>122.5</b>	0.715	0.418	3.43	651.2	0.697	0.452
E + No growing, new G & D arch.	3.31	124.5	0.705	0.449	3.19	471.2	0.690	0.454
F + Large networks (StyleGAN2)	<b>2.84</b>	145.0	0.689	<b>0.492</b>	<b>2.32</b>	<b>415.5</b>	0.678	<b>0.514</b>
Config A with large networks	3.98	199.2	0.716	0.422	—	—	—	—

图 2.15 StyleGAN2

### (4) 渐进式增长修正

渐进增长在稳定高分辨率图像合成方面非常成功，但会产生特定的伪影，对细节有强烈的位置偏好。作者认为，在渐进增长的过程中，每个分辨率都会瞬间用作输出

分辨率，迫使其生成最大频率细节，然后导致受过训练的网络在中间层具有过高的频率，从而损害了位移不变性。所以需要重新设计一个能产生高质量图像而不渐进增长的架构。在生成方法的背景下，Skip connections，残差网络和分层方法也被证明是非常成功的。三种生成器（虚线上方）和判别器体系结构如下图。Up 和 Down 分别表示双线性上和下采样。在残差网络中，这些还包括 1x1 卷积以调整特征图的 channel 数。tRGB 和 fRGB 在 RGB 和高维每像素数据之间转换。Config E 和 F 中使用的体系结构以绿色突出显示。

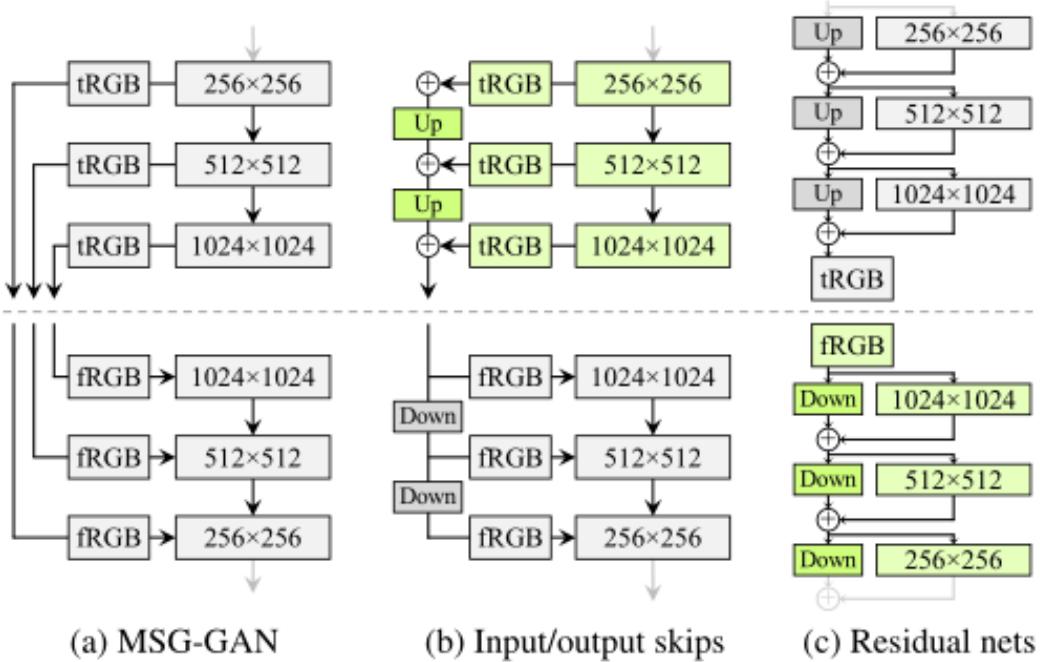


图 2.16 渐进式增长修正

表下比较了三种生成器和三种鉴别器体系结果，StyleGAN 中使用的原始前馈网络、跳过连接和残差网络，它们都是未经渐进增长而训练的。

由上表可以看出使用 skips 连接的生成器 PPL 最小。使用残差网络的判别器对 FID 有利。StyleGAN2 使用了一个 skip generator 和一个残差鉴别器，切换到这种设置显著地改进了 FID 和 PPL。

从研究中可以看到，从一开始，网络就专注于低分辨率图像，并随着训练的进行逐渐将其注意力转移到较大分辨率上。在 (a) 中，生成器基本上输出 512x512 图像，并对 1024x1024 进行一些细微锐化。而在 (b) 中，较大的网络更多地关注高分辨率细节。通过将两个网络的最高分辨率层中的特征图的数量加倍来进行测试。这使行为更加符合预期：图 (b) 显示了贡献的显著增加，而实验结果如下。

该表在四个 LSUN 类别中对 StyleGAN 和 StyleGAN2 进行了比较，再次显示出 FID

<b>FFHQ</b>	D original		D input skips		D residual	
	FID	PPL	FID	PPL	FID	PPL
G original	4.32	265	4.18	235	3.58	269
G output skips	4.33	169	3.77	127	<b>3.31</b>	<b>125</b>
G residual	4.35	203	3.96	229	3.79	243

<b>LSUN Car</b>	D original		D input skips		D residual	
	FID	PPL	FID	PPL	FID	PPL
G original	3.75	905	3.23	758	3.25	802
G output skips	3.77	544	3.86	<b>316</b>	3.19	471
G residual	3.93	981	3.40	667	<b>2.66</b>	645

图 2.17 渐进式增长修正

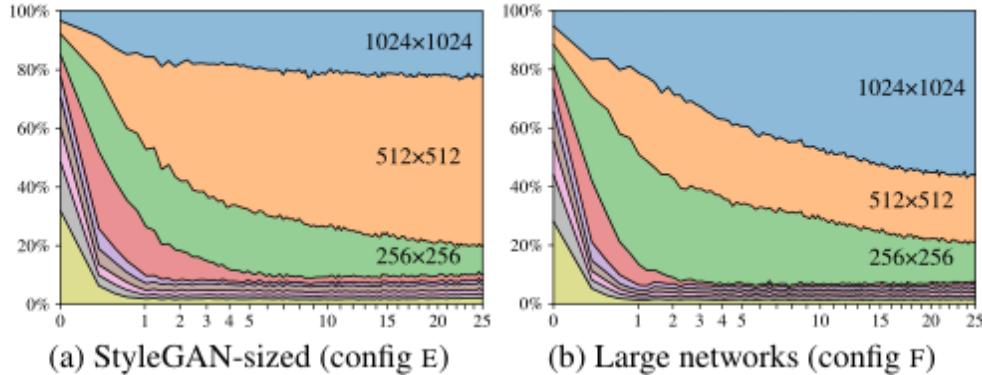


图 2.18 StyleGAN2

的明显改善和 PPL 的显著进步。进一步扩大规模可能会带来额外的好处。

## (5) 结论

本文确定和修复了 StyleGAN 中的几个图像质量问题，进一步提高了生成图像的质量。该论文的主要创新点如下：

- 使用 Weight demodulation 代替 AdaIN。
- 发现 PPL 与生成图像质量的关系，使用 Path length regularization。
- 去除渐进式网络，在生成器和判别器中采用不同的网络结构。

## (6) 展望

在未来，研究路径长度正则化的进一步改进可能会带来成果，例如，用数据驱动的特征空间度量替换像素空间  $L_2$  距离。而考虑到 GAN 的实际部署，寻找新的方法来减少训练数据需求是很重要的。在无法获取数万个训练样本的应用程序中，以及包含大量内在变化的数据集中，这一点尤其重要。

<b>Dataset</b>	<b>Resolution</b>	<b>StyleGAN (A)</b>		<b>StyleGAN2 (F)</b>	
		FID	PPL	FID	PPL
LSUN CAR	512×384	3.27	1485	<b>2.32</b>	<b>416</b>
LSUN CAT	256×256	8.53	924	<b>6.93</b>	<b>439</b>
LSUN CHURCH	256×256	4.21	742	<b>3.86</b>	<b>342</b>
LSUN HORSE	256×256	3.83	1405	<b>3.43</b>	<b>338</b>

图 2.19 StyleGAN2

### 2.3.3 Alias-Free Generative Adversarial Networks

#### 1. 研究背景

英伟达 StyleGAN 团队在 NeurIPS 2021 论文《Alias-Free Generative Adversarial Networks》中推出了 Alias-Free GAN，也即 StyleGAN3。典型的 GAN 具有分层卷积性质，通过上采样层分层细化，随后经过卷积层进行局部混合，并经由非线性映射层引入新的细节，但它们的合成过程过度依赖于绝对像素坐标。这就导致图像细节会粘连在坐标上，而在对象的表面，造成一种纹理粘连 (texture sticking) 的现象，例如下图，由 StyleGAN2 生成的毛发，主要都固定在相同的坐标上，导致了形成水平的条纹。

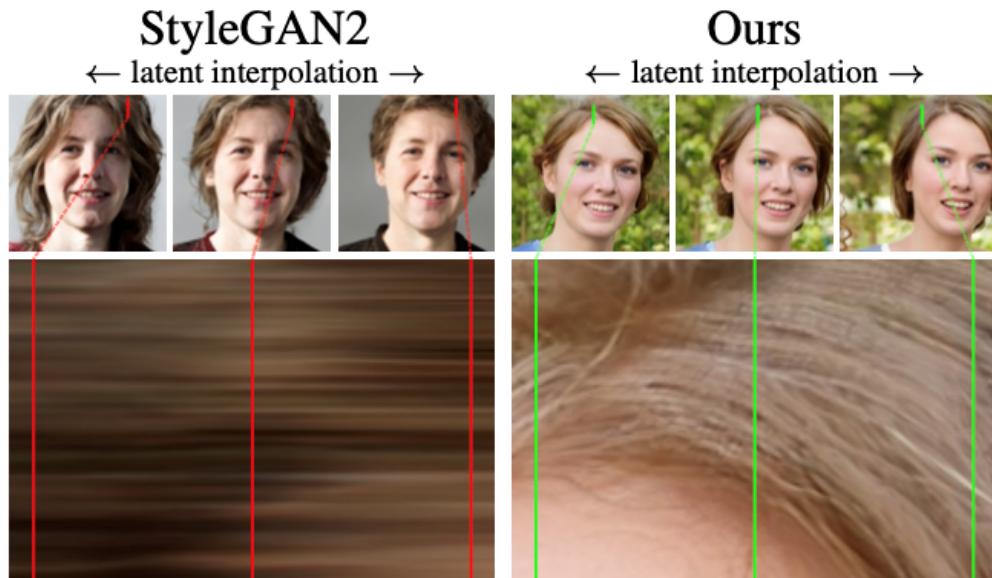


图 2.20 Alias-Free GAN

因此英伟达的研究者探究导致 GANs 中纹理粘连的现象，原因主要是“粗糙”的信号处理过程和神经网络混叠。而通过将网络中所有信号解读为连续性，他们进行了普遍适用的、小的架构变化，保证多余的信息不会参与分层合成过程，并由此得到了

StyleGAN3 与 StyleGAN2 相比，StyleGAN3 获得了类似的 FID，并在亚像素尺度上实现了真正的图像平移和旋转不变性，从而大幅度提升生成图像的质量。

## 2. 贡献

典型 GAN 生成器的结构化处理过程是：粗糙、低分辨率的特征通过上采样层分层细化，再通过卷积局部混合，以及非线性引入新的细节。这种体系结构可能基本还原了图像的表面特征，但它并没有以一种“自然而然”的方式合成更逼真的图像，也就是说，粗糙特征确保了图像细节的存在，但没有控制它们的精确位置，细节被固定在了图像坐标上。这项研究的目标就是，创建更自然的转换层次的体系结构，让每个特征的精确亚像素位置都从底层粗特征中获得。

本次则主要针对平移和旋转的两种类型移动，Alias-Free-T 和 Alias-Free-R，而传统的 GAN 网络结构，包括卷积、上采样、下采样和非线性这几个方面，而 Alias-Free GAN 整架构，是在 StyleGAN2 基础上调整的。而整体大致是这样一个调整过程：首先为了便于对输入的图像进行连续的平移和转化，他们用傅里叶特征取代 StyleGAN2 中的输入常数。接着，删除了每个像素中的噪声输入，因为它们与特征转换无关。

此外，他们还降低了映射网络的深度，并禁用了混合正则化和路径长度正则化。最后取消了输出残差链接。在实践中，还对卷积权重进行了划分。各层遵循严格的 2CE 上采样时间表，在每个分辨率下执行两层，使得上采样后特征图的数量减半。而结果表明，当下的版本与 StyleGAN2 相比，Alias-Free GAN 在 FID 分数上表现的更好，而下表所述 FID 在 50k 图像和所有训练图像计算，其呈现是越低越好。而 EQ-T 和 EQ-R 是分贝 (dB) 的等变度量，其结果是越高越好。

Configuration	FID↓	EQ-T↑	EQ-R↑	Parameter	FID↓	EQ-T↑	EQ-R↑	Time	Mem.
A StyleGAN2	5.14	—	—	Filter size $n = 4$	4.72	57.49	39.70	<b>0.84</b> ×	<b>0.99</b> ×
B + Fourier features	4.79	16.23	10.81	* Filter size $n = 6$	<b>4.50</b>	<b>66.65</b>	40.48	1.00×	1.00×
C + No noise inputs	4.54	15.81	10.84	Filter size $n = 8$	4.66	65.57	<b>42.09</b>	1.18×	1.01×
D + Simplified generator	5.21	19.47	10.41	Upsampling $m = 1$	<b>4.38</b>	39.96	36.42	<b>0.65</b> ×	<b>0.87</b> ×
E + Boundaries & upsampling	6.02	24.62	10.97	* Upsampling $m = 2$	4.50	66.65	40.48	1.00×	1.00×
F + Filtered nonlinearities	6.35	30.60	10.81	Upsampling $m = 4$	4.57	<b>74.21</b>	<b>40.97</b>	2.31×	1.62×
G + Non-critical sampling	4.78	43.90	10.84	Stopband $f_{t,0} = 2^{1.5}$	4.62	51.10	29.14	<b>0.86</b> ×	<b>0.90</b> ×
H + Transformed Fourier features	4.64	45.20	10.61	* Stopband $f_{t,0} = 2^{2.1}$	<b>4.50</b>	66.65	40.48	1.00×	1.00×
T + Flexible layers (StyleGAN3-T)	4.62	63.01	13.12	Stopband $f_{t,0} = 2^{3.1}$	4.68	<b>73.13</b>	<b>41.63</b>	1.36×	1.25×
R + Rotation equiv. (StyleGAN3-R)	<b>4.50</b>	<b>66.65</b>	<b>40.48</b>						

图 2.21 FID 在 50k 图像和所有训练图像计算

如图下所示当中 (a) 绿色的过渡带更宽，减少不必要的阻带纹波，从而导致更强的衰减。而 (b) StyleGAN3，对应于上图中的 configs t 和 r。主要的数据通路包括傅里叶特征和归一化，调制卷积，以及非线性过滤器。

而 StyleGAN3 与过往 styleGAN2 相比，具有下列四项新特性：

- Alias-free 生成器架构和训练配置 (stylegan3-t 和 stylegan3-r)；
- 提供交互式可视化(visualizer.py)、频谱分析(avg\_spectra.py)和视频生成(gen\_video.py)

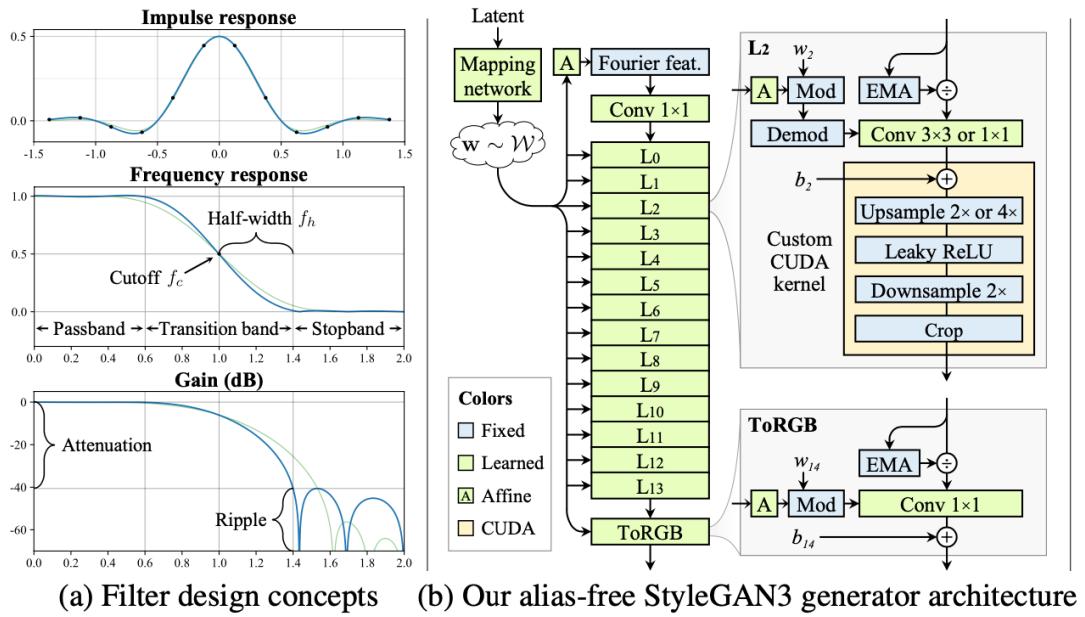


图 2.22 StyleGAN3 说明

的工具；

- 同变性度量 (eqt50k\_int、eqt50k\_frac 和 eqr50k)；
- 其他改进：减少内存使用、训练速度略升以及 bug 修复。

3. 结论和未来展望 StyleGAN2 生成的动物毛发会粘在屏幕上，和动物的形态变化不一致。这就是 StyleGAN 变体一直无法解决的难题之一。它从根本上解决了 StyleGAN2 图像坐标与特征粘连的问题，实现了真正的图像平移、旋转等不变性，大幅提高了图像合成质量，而在 GAN 的相关文献中，混叠这一概念很少被提及，研究者在这项研究中，提供了两个混叠来源：1) 由非理想上采样滤波器（如卷积、双线性卷积或跨步卷积）产生的像素网格后模糊图像。2) 非线性的逐点应用，如 ReLU 或 swish。他们发现，混叠网络具有放大并在多个尺度上组合图像像素的能力，这对于弱化固定在屏幕坐标中的纹理图案至关重要。并且实验证明，该网络还适用于深度学习中所有常用过滤器，甚至图像处理中使用的高质量过滤器。本研究知道，成功消除所有位置参考来源意味着无论像素坐标如何，细节都可以被很好地生成，它相当于在所有层中对亚像素平移（和旋转）实施连续的等方差。而事实证明，当前的上采样滤波器在抑制混叠方面根本不够积极，而且需要具有超过 100dB 衰减的高质量滤波器。这项研究提出了一种解决点态非线性引起的混叠的原理，考虑了它们在连续域的影响，并对结果进行适当的低通滤波。此外，实验证明：一个基于 1CE1 卷积的模型能够产生强旋转的等变生成器。一旦适当地抑制了混叠以迫使模型实现更自然的层次细化，它的操作模式就会发现显著变化：坐标系统等内部表示，允许细节准确地附加到底层表面。这将显著改

进用于生成视频和动画的模型。另外值得注意的是，英伟达的 StyleGAN3 项目消耗了令人难以想象的资源和电力。研究者在论文中表示，整个项目在 NVIDIA V100 内部集群上消耗了 92 个 GPU year (即单个 GPU 一年的计算) 和 225 兆瓦时 (Mwh) 的电力。而在社会影响方面，GANs 的潜在负面社会影响包括许多形式的虚假信息，例如社交媒体上的虚假肖像，甚至领导人的宣传视频，而本论文的结果将会消除那些视频中的人造痕迹，潜在地让视频更有说服力，或者说更有欺骗性。

## 2.4 预训练图像生成模型的应用

### 2.4.1 GAN Inversion: A Survey

研究者先是交代了 GAN 的重要性与概述，同时并说明 GAN inversion 的在给定的影像反转预训练的 GAN 模型潜在空间，好方便生成器可以反转得作用，另外研究者也全面性的调查近几年在 GAN inversion 相关的研究、应用与贡献，该项研究工作是首次对 GAN 反演的第一次调查，并且该研究团队在此领域的所有方面做出全面系统的回顾和分析，同时在完成分析后对该 GAN 反演方法中的属性与性能做出比较总结，最后并确定该领域所面对的挑战与问题，并确定对于未来研究的趋势。

随后研究者针对 GAN 反转的问题定义和概述进行详细的说明，其众所周知 GAN 能够生成高分辨率和逼真的“假”图像，但鲜为人知的一个方面是如何将这些 GAN 应用在“真实”图像上。而第一种可能的解决方案，该篇研究的重点，亦即 GAN 反演，是获取“真实”图像的潜在代码，并通过操纵潜在空间中的潜在代码来执行一些后续的图像处理任务。而 GAN 反演将会试图找到一个潜在的代码，并为给定图像实现两个目标，其一为忠实地和逼真地重建输入图像，首先运用数学做好定义，并条件映射找到视觉相似的对应图像，其二则是做促进下游的任务，而所谓的下游任务的目标又取决于使用哪一个潜在空间。同时研究也说明，往常像是面部图像为例且基于学习的 GAN 反演方法不能忠实地重建图像内容，即便基于优化的技术已经实现了卓越的图像重建，但它们不可避免的缺点是计算成本显著地增加。所以基于学习的 GAN 反演方法的最新改进主要集中在如何忠实地重建图像，例如在训练资料时或提出迭代反馈机制来整合额外的面部身份损失。另外在基于优化的方法的最新改进更加强调如何更快地找到所需的潜在代码，并提出了几种关于初始化和优化器的策略。诚然现有的反演方法无法同时实现重建质量和推理时间，从而导致“质量-时间权衡”(quality-time tradeoff)，尽管另外提出了一些混合方法来平衡这种权衡，但快速找到准确的潜在代码仍然是一个挑战。

研究者整理出近期广泛应用如下：

- 图像处理 (Image Manipulation)

- 图像生成 (Image Generation)
- 图像修复 (Image Restoration )
- 图像插值 (Image Interpolation)
- 风格转移 (Style Transfer)
- 压缩感知 (Compressive Sensing)
- 语义扩散 (Semantic Diffusion)
- 品类转移 (Category Transfer)
- 对抗性防御 (Adversarial Defense)
- 3D 重建 (3D Reconstruction)
- 图像理解 (Image Understanding)
- 多模态学习 (Multimodal Learning)
- 医学影像 (Medical Imaging)

而在研究中，研究者们提出在该领域的挑战和未来方向如下：

### 1. 理论理解 (Theoretical Understanding)

尽管在应用上取得了成功，但对 GAN 反演仍缺乏理论认识且 GAN 反演可以被视为与 PCA 通常执行的降维的非线性等效。数据中的非线性结构可以紧凑地表示，并且诱导几何需要使用非线性统计工具、黎曼流形和局部线性方法。相关领域的成熟理论可以从不同角度促进对神经网络权重（参数）或潜在空间方面的 GAN 反演的更好的理论理解。例如，研究可以将 GAN 反演公式化为将信号分解为分量（矩阵分解问题），并使用非线性因子分析 (FA)、独立分量分析 (ICA) 和潜在狄利克雷分配 (LDA)，分解网络权重并找到潜在空间的可解释方向。

### 2. 反转类型 (Inversion Type)

除了 GAN 反转之外，还开发了一些方法来反转基于编码器-解码器架构的生成模型。IIN 方法学习变分自编码器 (VAE) 的可逆解开解释。Zhu et al. 开发了潜在可逆的自动编码器方法来学习人脸图像的解开表示，可以根据属性从中编辑内容。而 The LaDDer approach 使用基于生成先验（包括加性 VAE 和超先验混合）的元嵌入将训练有素的 VAE 的潜在空间投影到低维潜在空间，其中使用多个 VAE 模型形成层次化的表示。探索如何结合 GAN 反演和编码器解码器反演是有益的，这样研究就可以利用到两全其美。

### 3. 域泛化 (Domain Generalization)

GAN 反演已被证明在跨域应用中是有效的，例如风格迁移和图像恢复，这表明预训练模型已经学习了域不可知的特征，来自不同域的图像可以倒置到相同的潜在空间中，从中可以得出有效的度量。目前已经开发出多任务方法来协同利用视觉线索，例

如 GAN 框架内的图像恢复和图像分割或语义分割和深度估计。

#### 4. 场景表示 (Scene Representation)

GAN 反演方法可以操纵几何（例如，缩放、移位和旋转）、纹理（例如，背景模糊和锐度）和颜色（例如，照明和饱和度）。这种能力表明在大规模数据集上预训练的 GAN 模型已经从现实世界的场景中学到了一些物理信息。隐式神经表征学习是 3D 社区的最新趋势，它学习 3D 形状或场景的隐式函数，并能够控制场景属性，例如照明、相机参数、姿势、几何形状、外观，和语义结构。它已用于体积性能捕获、新视图合成、面部形状生成、对象建模和人体重建。最近的 StyleRig 方法是基于 3D 可变形模型 (3DMM) 的语义参数和 StyleGAN 的输入进行训练的。它开辟了一个有趣的研究方向，以反转用于 3D 重建的预训练 GAN 的这种隐式表示，例如，使用 StyleGAN 进行人脸建模或延时视频生成。

#### 5. 精确控制 (Precise Control)

GAN 反演可用于找到图像处理的方向，同时保留身份和其他属性，诚然这还需要一些调整来实现精确细粒度控制的所需粒度，例如注视重定向、重新照明和连续视图控制。

#### 6. 多峰倒置 (Multimodal Inversion)

现有的 GAN 反演方法主要涉及图像。然而，生成模型的最新进展超出了图像领域，例如用于音频合成的 GPT-3 语言模型和 WaveNet。这些复杂的深度神经网络经过各种大规模数据集的训练，已被证明能够表示广泛的不同内容、风格、情感和主题。在这些不同的模态上应用 GAN 反转技术可以为语言风格转移等任务提供新的视角。

#### 7. 评估指标 (Evaluation Metrics)

感知质量指标可以更好地评估逼真和多样化的图像或与原始图像一致的身份，仍有待探索。此外，评估主要集中在照片写实度上，或者使用针对真实图像训练的模型判断生成图像的分布是否与真实图像在分类或分割准确性方面一致。然而，仍然缺乏有效的评估工具来评估预测结果与预期结果之间的差异或更直接地衡量倒置的潜在代码。

### 2.4.2 Encoding in Style: a StyleGAN Encoder for Image-to-Image Translation

1. 背景和动机 GAN 的提出对于图像生成，尤其是对与人脸相关的图像生成任务有显著的提升。英伟达在此基础上提出了 StyleGAN，引入了风格向量来更好地编码输入图像。原有的隐空间  $z$  中的特征之间相互关联、耦合，而经过多层感知机映射到新的隐空间  $w$  之后，特征之间更加独立，更方便对特征进行可视化控制。本文基于的是 StyleGAN 升级后的 StyleGAN2，并提出了一个可以直接将真实人脸图像直接映射到

$W+$  隐空间的 encoder。StyleGAN2 主要对 AdaIN 作了修改，解决了 StyleGAN 生成的图像中存在伪影的问题，并对生成图像的质量作了进一步的提升。

## 2. 本文主要贡献

- (1) 本文提出了一个能将真实人脸图像直接映射到  $W+$  隐空间的 encoder。
- (2) 本文提出了一个通用的解决 image2image 问题的 pSp 端到端框架。

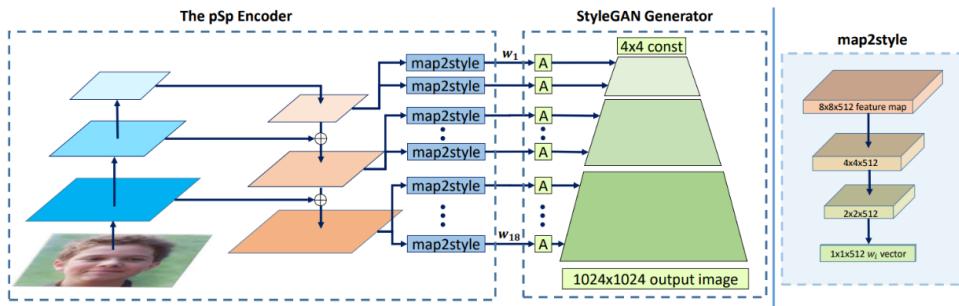


图 2.23 pSp 端到端框架

pSp 框架基于新型编码器网络，该网络生成一系列 style 向量，然后输入到预训练的 StyleGAN 中，形成扩展的  $W+$ 。首先，该编码器可以直接将真实图像嵌入到  $W+$  中，无需额外优化。其次，编码器可以直接解决图像的转换任务，即图像转换任务定义为：从输入域到潜在域的编码问题。与先前使用“先反转，后编辑”的方法不同，pSp 不要求输入图像在 StyleGAN 域中进行特征表示，也可以处理各种任务。该编码器极大地简化了训练过程，在没有图像对的标签下提供更好的支持，并且通过 style 的重采样可以支持多模态合成。最后，pSp 的应用不仅限于人类面部领域，即使与专为特定任务而设计的解决方案相比，也表现极佳。

## 3. 实验结果

### (1) StyleGan inversion 任务

首先评估了 pSp 框架在 StyleGan inversion 任务上的效果，尤其在 latent domain 中发现真实图片的 latent code。并且将论文的方法与 Karras 的优化技术进行比较，当中包含了 ALAE 编码器和 IDInvert 编码器，其中 ALAE 是一种基于 StyleGan 的自动编码器，编码器和生成器一起训练以生成 latent codes。而对于 IDInvert，图像被嵌入到一个预训练的 latent domain 中：首先，图像被编码到  $W+$  中，然后直接对生成的图像进行优化以调整 latent。为了进行公平的比较，论文与 IDInvert 进行比较，并且在编码后不再做进一步的优化。从结论来说，在对输入图像的重建任务上，ALAE 效果较差，IDInvert 能较好的保留图像属性，但不能准确地保留人脸的一致性。而论文的 pSp 更加真实，能保留更细节的属性：比如灯光、发型、眼镜。

### (2) 人脸正面化实验

由于需要非局部变换以及缺少成对数据的问题，人脸正面化难度还是很大的，RotateAndRender 通过 3D 几何对齐来解决这个问题。论文是直接生成，甚至用没有标记的数据训练。而方法主要有两个额外的操作，一个是在训练过程中针对 target 图像进行随机 flip，这样迫使网络学习一个中间位置，也就是正面的脸，保证正面化前后人脸的一致性。二是由于此研究并不是太关注图像的背景信息，因此从 loss 部分进行了一些权重的调整，主要是 L2 和 LPIPS。

而从结论来看，pix2pixHD 不能收敛到一个比较优秀的结果，因为 pix2pixHD 很大程度上依赖条件因素中的成对信息。该论文的 pSp 方法能较好保持人脸原有属性，同时将其成功转化为正脸，与 RotateAndRender 方法的效果相当，但 pSp 方法更简单，不需要一些专门的模块，比如 R&R 的 3DMM 安装和喷涂步骤。

### (3) 其他任务上的拓展

PULSE 给出了一个从 LR 到 HR 的生成式的超分方案：将 HR 图像的 manifold 下采样到 LR 的大小，然后通过计算损失来优化输入的无监督方式。pSp 也在这方面做出了一些尝试，可以获得和 PULSE 相比不错的结果。pSp 使用了全监督的方式来进行训练，针对输入随机进行采样  $x_1, x_2, x_4, x_8$  和  $x_{16}$  然后插值放大到原始大小。pix2pixHD 给出了不错的结果，但是 less photo-realistic；PULSE 能生成高精度图像，但是不那么像原来的那个人了。然后，针对 medium 的层设置  $\alpha = 0.5$  做 style mixing，控制面部特征，而 pSp 也可以生成多个结果。

## 4. 结论以及展望

这篇论文提出了一种新的编码器体系结构，它可以用来直接将真实图像映射到  $W+$  潜在空间，而不需要优化。在这种结构中，样式以分层方式提取，并被提供给固定 StyleGAN 生成器的相应输入。结合编码器和 StyleGAN 解码器，作者提出了一个通用的框架来解决各种图像到图像的转换任务，所有这些任务都使用相同的架构。值得注意的是，与以前 StyleGAN 编码器的“先反转，后编辑”方法不同，作者展示了 pSp 可用于将这些翻译任务直接编码到 StyleGAN 中，从而支持不在 StyleGAN 域中的输入图像。此外，与以往的工作不同的是，以前的工作通常依赖于专用的架构来解决单一的翻译任务，而作者通过实验 pSp 能够解决各种各样的问题，只需要对训练损失和方法进行最小程度的改变。下一步研究者可以研究利用 StyleGAN 进行真实图像到图像的翻译任务。

### 2.4.3 A Style-Based Generator Architecture for Generative Adversarial Networks

#### 1. 论文简介

本文从风格转换文献中提出了一种生成对抗网络的可替代生成器框架。新的框架

导致了高级属性(如人脸训练时的姿势和身份)的自动学习和无监督分离,以及生成图像中的随机变化(如雀斑、头发),并实现了对合成的直观、特定规模的控制。新的生成器在传统分布质量指标方面改进了目前最先进的技术的效果,有明显更好的插值特性,也更好地解耦变量的潜在因素。为了量化插值质量和解耦效果,此研究提出了两种新的、自动化的方法,适用于任何生成器架构。最后,本文构建了一个新的、高度多样化和高质量的人脸数据集。

## 2. 研究背景

生成方法产生的图像的分辨率和质量——特别是生成对抗网络(GAN)——最近得到了快速的改善。然而,这些生成器仍然像黑盒子一样运行,尽管最近的努力,对图像合成过程的各个方面理解,例如,随机特征的起源,仍然缺乏。潜在空间的性质也没有得到很好的理解,常用的潜在空间插值没有提供定量的方法来比较不同的生成器。

受风格转换文献的启发,该研究重新设计了生成器的架构,以提出新的方法来控制图像合成过程。研究的生成器从一个学习过的常量输入开始,根据潜码在每个卷积层调整图像的“风格”,从而直接控制不同尺度下图像特征的强度。与直接注入网络的噪声相结合,这种架构上的变化导致生成图像中的高级属性(例如姿态、身份)与随机变化(例如雀斑、头发)自动、无监督地分离,并实现了直观的特定尺度的混合和插值操作。研究不以任何方式修改判别器或损失函数,因此该研究的工作与正在进行的关于GAN损失函数、正则化和超参数的讨论是正交的。

研究者将的生成器将输入的潜在编码嵌入到一个中间的潜在空间中,这对变量因子在网络中的表现方式有着深远的影响。输入潜在空间必须遵循训练数据的概率密度,此研究认为这导致了某种程度的不可避免的纠缠。研究的中间潜在空间中就不受这个限制,因此可以解耦合。由于之前估计潜在空间解耦程度的方法不能直接应用于本研究的案例,本研究提出了两个新的自动化度量方法——感知路径长度和线性可分性——来量化生成器的这些方面。使用这些指标,本研究表明,与传统的生成器架构相比,本研究的生成器实现了一个在不同变量因素中能得到更线性,更少耦合表现的效果。

最后,研究者们提供了一个新的人脸数据集(Flickr-Faces-HQ, FFHQ),它比现有的高分辨率数据集提供了更高的质量,涵盖了更广泛的变化。同时研究者将这个数据集,连同此研究的源代码和预先训练好的网络一起公开。

## 3. Style-based generator

通常潜在编码通过输入层提供给生成器,即前向网络(图1a)的第一层。研究者通过省略该输入层和从一个可学习常量(图1b,右)开始来分离该设计。给定一个在输入潜在空间 $Z$ 的潜在编码 $z$ 、一个非线性的映射网络 $f: \mathcal{Z} \rightarrow \mathcal{W}$ 来生成 $w \in \mathcal{W}$ (图1b,左)。简单来说,研究设置两个空间的维度为512(即 $Z$ 和 $W$ 都是 $512 \times 1$ ),映射函

数  $f$  使用 8 层 MLP 来实现，决策函数将在 Section 4.1 分析。可学习的仿射转换 (A) 将  $w$  转换成 styles  $y = (y_s, y_b)$ ，用来控制生成网络  $g$  每一个卷积层后面的 adaptive instance normalization (AdaIN) 操作。该 AdaIN 操作被定义为：

$$\text{AdaIN}(x_i, y) = y_{s,i} \frac{x_i - \mu(x_i)}{\sigma(x_i)} + y_{b,i} \quad (2.8)$$

每个特征映射  $x_i$  将会被分别归一化，然后使用来自 style  $y$  的对应尺寸成分  $y_{s,i}$  和  $y_{b,i}$  来按比例变化和实现偏差。因此， $y$  的维数是该层上特征图数量的两倍。将此研究的方法与风格转移相比较，本研究从向量  $w$  中计算空间不变风格  $y$ ，而不是从一个例子图像中计算。而此研究选择对  $y$  重复使用“style”这个词是因为类似的网络架构已经被用于前向风格传输、无监督图像到图像的转换和域混合中。与更一般的特征变换相比，因为 AdaIN 的效率和紧凑的表示，其特别适合此研究的目的。简单说明下这个网络，在潜在编码  $Z$  中间添加一个 mapping 网络，用来将输入的向量编码为中间向量  $W$ ，然后该中间向量将会分别传送 18 个控制向量  $A$  到生成网络  $g$  中的 18 层上，用来控制不同的 style。而要加 Mapping Network 的原因在於，后续得到的 18 个控制向量之间会存在特征纠缠的现象——比如说研究想调节 8\*8 分辨率上的控制向量（假设它能控制人脸生成的角度），但是此研究会发现 32 \* 32 分辨率上的控制内容（譬如肤色）也被改变了，这个就叫做特征纠缠。所以 Mapping Network 的作用就是为输入向量的特征解缠提供一条学习的通路。最后就能够使用上面(图 1b)输出的 1024\*1024 的图像和真正图像作为 Discriminator 的输入，最小化损失来训练模型参数，而最后我們提供了一个直接的方法，通过引入显式噪声 noise 输入以产生随机细节。这些是由不相关的高斯噪声组成的单通道图像，本研究为合成网络的每一层提供一个专用的噪声图像。利用学习到的每个特征的缩放因子将噪声图像广播到所有的特征映射上，然后加入到相应卷积的输出中。

#### 4. 生成效果

在研究该研究的生成器的特性之前，此研究通过实验证明，重新设计不会影响图像质量，但实际上，它大大改善了图像质量。表中给出了各种生成器架构在数据 CELEBA-HQ 和在研究者的新 FFHQ 数据集上的 Frechet inception distances(FID) 效果。此研究的 baseline 配置 (A) 是 Karras et al. 的 Progressive GAN 设置，除非另有说明，否则此研究继承了网络和所有超参数。首先研究者通过使用双线性上/下采样操作、更长时期的训练和调优超参数来切换到改进的 baseline(B)。训练设置的详细描述和超参数包含在附录 C 中。然后本研究通过添加映射网络和 AdaIN 操作进一步改善这一新的 baseline(C)，并得到一个令人惊讶的发现，即网络不再受益于将潜在编码输入到卷积第一层的这个操作。因此本研究通过消除传统的输入层和开始从一个可学习的 4\*4\*512 常

量张量开始图像合成来简化架构 (D, 因为 StyleGAN 生成图像的特征是由 W 和 AdaIN 控制的)。本研究发现一个很了不起的效果, 即合成网络能够产生有意义的结果, 尽管它接收输入只有用来控制 AdaIN 操作的 style。

最后, 此研究引入了进一步改善结果的噪声输入 (E, noise inputs), 以及新的混合正则化 (F, mixing regularization), 它去除了邻近的 style, 并对生成的图像进行更细粒度的控制。此研究使用两种不同的损失函数来评估本研究的方法: 对于 CELEBA-HQ, 本研究依赖于 WGAN-GP, 而 FFHQ 使用 WGAN-GP 来处理配置 A, 对于配置 B-F, 使用 R1 正则化的非饱和损失。本研究发现这些选择可以得到最好的结果。该研究的贡献不改变损失函数。同时此研究观察到, 对比传统的生成器 (B), 基于风格的生成器 (E) 很明显地改进了 FIDs, 几乎提高了 20%, 这证实了在并行工作中进行的大规模 ImageNet 测量。经 FIDs 确认, 平均质量高, 甚至连眼镜、帽子等配件都成功合成。

## 5. 生成器的特性

此研究的生成器架构, 使它能够通过特定比例的 style 修改来控制图像合成。本研究可以把映射网络和仿射变换看作是一种从学习分布中为每种风格抽取样本的方法, 而合成网络则是一种基于 styles 集合来生成新图像的方法。每种 style 的效果在网络中都是局部的, 即修改 style 的特定子集只能影响图像的某些方面。

为了了解这种定位的原因, 让本研究考虑一下 AdaIN 操作如何首先将每个通道归一化为零均值和单位方差, 然后根据 style 应用尺度和偏差。根据 style, 新的每个通道统计信息修改了后续卷积操作中特征的相对重要性, 但由于归一化, 它们不依赖于原始统计信息。因此, 每种 style 在被下一个 AdaIN 操作覆盖之前只能控制一个卷积

## 6. 风格混合

为了进一步鼓励 style 的定位, 本研究采用 mixing regularization, 即在训练过程中使用两个随机的潜码生成给定百分比的图像。当生成这样的图像时, 本研究只需要在合成网络中随机选择的一个点从一个潜在代码切换到另一个——该研究称之为 style mixing 的操作。具体来说, 本研究通过映射网络运行  $z_1$ 、 $z_2$  这两个潜码, 然后就有对应的  $w_1$ 、 $w_2$  控制 style, 将  $w_1$  应用在交点之前,  $w_2$  应用在交点之后。这种正则化技术防止网络假设相邻样式是相关的。

## 7. 结论

基于该研究的结果和 Chen 等人的并行工作, 传统的 GAN 生成器体系结构在各个方面都不如基于风格的设计。就已建立的质量度量而言, 这是正确的。研究者进一步相信, 研究者对高阶属性和随机效应的分离以及中间潜在空间的线性的研究将证明在提高 GAN 合成的理解和可控性方面卓有成效。而研究的平均路径长度度量可以很容易地在训练期间用作正则化器, 也许线性可分度度量的某些变体也可以用作正则化器。

总的来说，研究者期望在训练过程中直接塑造中间潜在空间的方法将为未来的工作提供有趣的途径。

#### 2.4.4 ReStyle: A Residual-Based StyleGAN Encoder via Iterative Refinement

1. Topic: 图像反演，获取潜在编码用于对图像进行直接编辑。
2. Problem: 现在的单次编码架构无法准确的获得图像反推的潜在编码。即在重建精度方面，基于学习的反演方法与基于优化的反演方法仍存在较大差距。因此，虽然在基于学习的反演方面取得了显著的进展，但设计合适的编码器和训练方案仍然是一个挑战，许多工作仍在使用逐图像优化方法。
3. Idea: 使用以残差为基础的迭代反馈调优架构。具体为将上一次迭代的输出与原始的输入图像共同输入到下一次迭代中，进行多次前向传播，让编码器注意到更多能够提升精度的特征。在增加有限的推理时间情况下，实现较高精度的输出。个人理解该方法是逐图像优化方法与编码器学习方法的折中。也是单步反演的松弛求解。

$$\hat{y} = G(E(x)) \quad (2.9)$$

其中  $x$  为输入图像， $\hat{y}$  为反演的图像， $G, E$  分别为生成器和编码器。

4. Methods:
  - 聚合真实图像和 Stylegan 的重建图像，生成六通道数据，然后输入到编码器中学习残差偏移编码
  - 更新图像  $x$  的潜在编码，并输入到生成器中生成新重建图像
  - 其中初始的潜在编码和对应的潜在图像分别是训练好的生成器的平均风格向量和其对应生成图片
5. 模型:
6. 编码器模型：因为迭代调优的特性，SOTA 工作中的特征金字塔和残差模块在该网络中并不需要。文中采用简单的特征下采样网络并结合 StyleGAN 的风格输入作为输出，如上图。
7. 实验：该作者对于不同的图片域进行了模型评估，分别是人脸域，汽车域和其他 GAN 常用的图片域数据集。
8. 图像反演的效果：虽然与基于学习的方法相比，逐图像优化技术实现了更好的图像重建，但它们带来了显著更高的计算成本。因此，在分析反演方法时，必须根据推断时间来衡量重建质量，从而导致所谓的质量-时间权衡。
9. 定性分析：相比于 e4e 和 psp，所提出模型能够逼近逐图像优化和真实值。研究从分析面部区域开始。与传统的 pSp 和 e4e 编码器相比，该研究的 ReStyle 变体可以匹

敌或超过它们的对手。更值得注意的是，虽然与 ReStyle 相比，优化技术实现了更好的身份相似性，但它们需要 20 多时间来匹配 ReStyle 获得的相似性。在汽车领域也可以观察到类似的权衡，在评估重构 L2 损失时，ReStyle 相对于典型编码器的优势更加明显。在非结构化教堂领域，应用在 pSp 上的 ReStyle 在重构质量上与优化和混合技术几乎一致，且推断时间显著降低。虽然优化通常可以实现卓越的重构，但 ReStyle 在重构质量和推断时间之间提供了极佳的平衡。

10. 图像 restyle 过程与效果：在早期的步骤编码器专注于精炼的背景和姿势，而在随后的步骤编码器移动其焦点调整更精细的细节沿着眼睛和头发。编码器的工作方式从粗到精，从集中于低频细节开始，然后逐渐通过调整高频细节来补充。

11. 结论：该文章提出了一个用于图像反演，编辑和美化的通用框架。

#### 2.4.5 Designing an Encoder for StyleGAN Image Manipulation

##### 1. 背景和动机

本文的目的是为了设计编码器能够更好的对图像进行编辑。首先，由于图像失真，感知质量和图像的可编辑性存在紧密的连系。在一定程度上，图像失真与感知质量存负相关，图像失真和图像的可编辑性成负相关。此外，本文还发现如果图像的逆隐空间向量越接近原本的隐空间分布  $\mathbf{W}$ ，那么图像的感知质量和可编辑性便越好。

##### 2. 本文主要贡献

- 分析了 StyleGAN 复杂的隐空间并且提出了一个创新的结构
- 分析并讨论了图像失真，图像感知和可编辑性之间的权衡
- 对这些权衡进行了分析，并且设计了两种方法来实现控制器来控制他们
- 提出了 e4e，一个创新的编码器专门设计来允许对反转真实图像的接下来的编辑。

##### 3. 实验结果

实验采用了 2 种配置，配置 A 没有采用 d-reg 和  $D_W$  损失，而配置 D 采用 d-reg 和  $D_W$  两种损失。首先分析了两种配置下的生成 latent vector 哪一种更接近  $\mathbf{W}$ 。采用计算各个 latent style code 的方差来判断是否相近。最终配置 D 比配置 A 明显更小。说明配置 D 生成的明显比配置 A 更要接近  $\mathbf{W}$ 。在失真的评估上，配置 A 要优于配置 D。而在感知质量和可编辑性上，FID 和 SWD 两种评估指标之间互相矛盾，因为这两个评估指标均不能准确的给出评估。最后本文采用人类评估的调查来评估感知质量和可编辑性。从调查中不难发现，配置 D 的效果在这两方面要远优于配置 A。最终可以得出，配置 D 的方法要更接近于  $\mathbf{W}$ ，虽然增大了一定程度的失真，但是该方法会让感知质量和可编辑性有所降低。

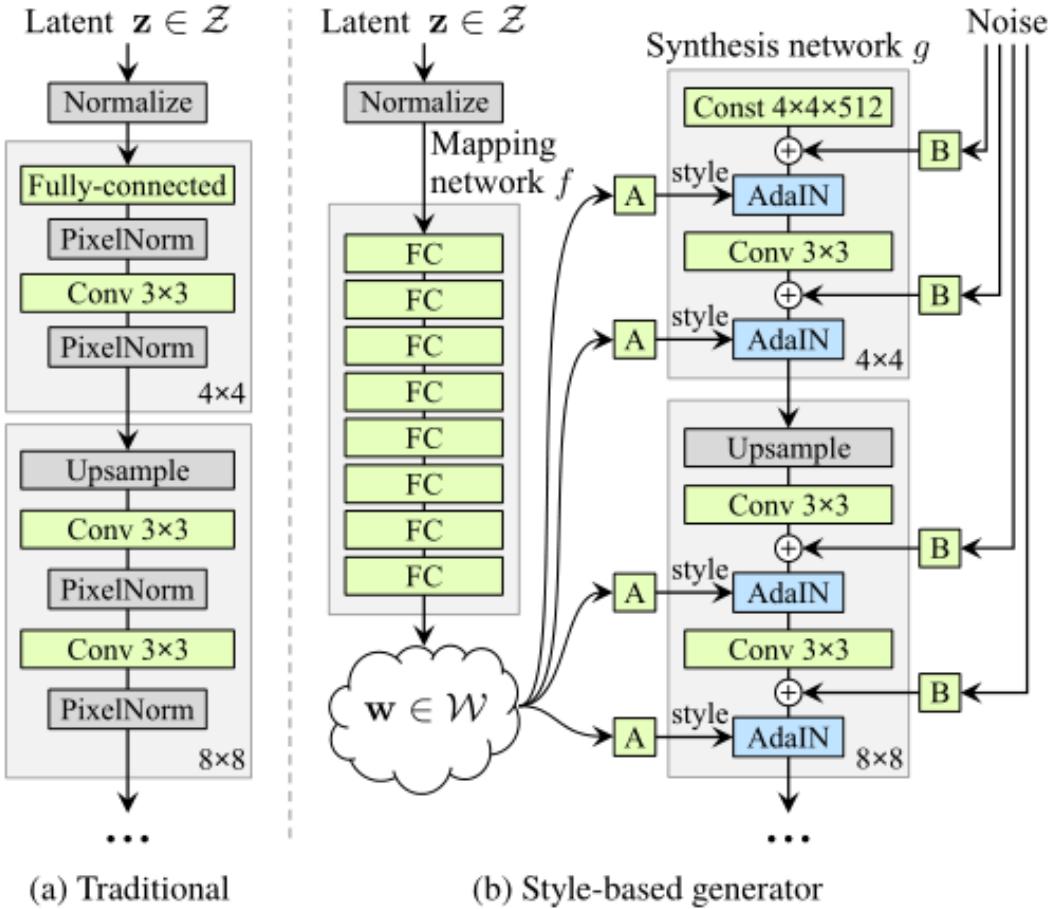


Figure 1. While a traditional generator [28] feeds the latent code through the input layer only, we first map the input to an intermediate latent space  $\mathcal{W}$ , which then controls the generator through adaptive instance normalization (AdaIN) at each convolution layer. Gaussian noise is added after each convolution, before evaluating the nonlinearity. Here “A” stands for a learned affine transform, and “B” applies learned per-channel scaling factors to the noise input. The mapping network  $f$  consists of 8 layers and the synthesis network  $g$  consists of 18 layers — two for each resolution ( $4^2 - 1024^2$ ). The output of the last layer is converted to RGB using a separate  $1 \times 1$  convolution, similar to Karras et al. [28]. Our generator has a total of 26.2M trainable parameters, compared to 23.1M in the traditional generator.

图 2.24 Style-Based Generator

Method	CelebA-HQ	FFHQ
A Baseline Progressive GAN [28]	7.79	8.04
B + Tuning (incl. bilinear up/down)	6.11	5.25
C + Add mapping and styles	5.34	4.85
D + Remove traditional input	5.07	4.88
E + Add noise inputs	<b>5.06</b>	4.42
F + Mixing regularization	5.17	<b>4.40</b>

Table 1. Fréchet inception distance (FID) for various generator designs (lower is better). In this paper we calculate the FIDs using 50,000 images drawn randomly from the training set, and report the lowest distance encountered over the course of training.

图 2.25 生成效果

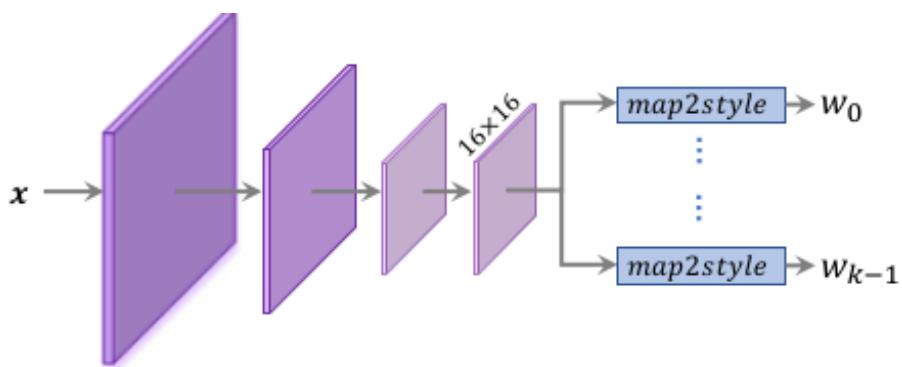


Figure 3. Our simplified encoder architecture. All  $k$  input style vectors of the generator are extracted from the encoder's final  $16 \times 16$  feature map which is passed through  $k$  *map2style* blocks [35].

图 2.26 ReStyle

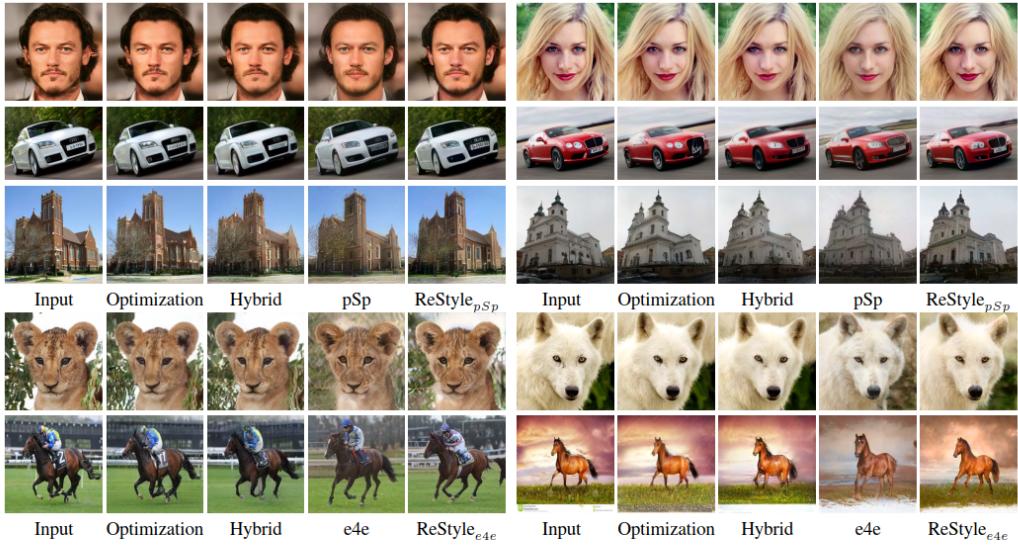


Figure 4. Qualitative Comparison. We compare various encoder-based and optimization-based inversion methods with our ReStyle scheme applied over pSp [35] and e4e [40] (denoted by ReStyle<sub>pSp</sub> and ReStyle<sub>e4e</sub>). Hybrid results are obtained by performing optimization on the latent codes obtained by the adjacent encoder. Additional comparisons in the supplementary materials. Best viewed zoomed-in.

图 2.27 ReStyle

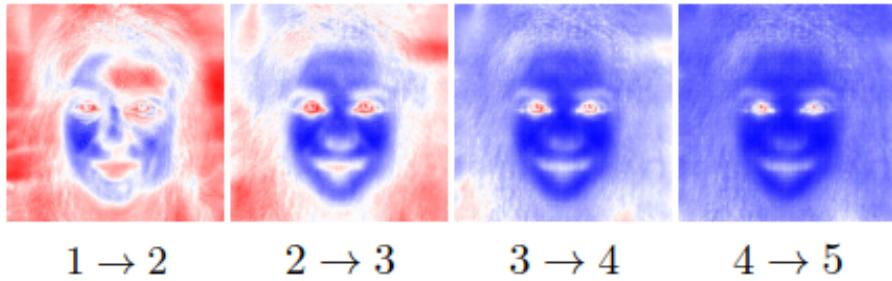


Figure 6. In each sub-image, we display a heatmap showing which image regions changed the most (in red) and which regions changed the least (in blue) between the specified iterations.

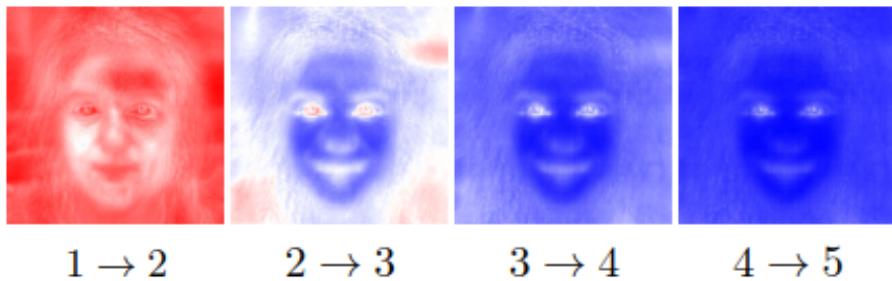


Figure 7. Similar to Figure 6, with the difference that here all images are normalized with respect to each other. As shown, the magnitude of change decreases with each step.

图 2.28 图像 restyle 过程与效果

## 第三章 研究方法

### 3.1 pSp、e4e 与 Restyle 的流程

#### 3.1.1 pSp 框架流程

pSp (pixel2style2pixel) 是一个通用的解决 image2image 问题的端到端的框架，在论文《Encoding in Style: a StyleGAN Encoder for Image-to-Image Translation》中提出，这个框架基于一种新颖的编码器网络，该网络直接生成一系列风格向量，这些风格向量被输入到一个预先训练的 StyleGAN 生成器中，形成扩展的 W+ 潜在空间。pSp 框架的流程为：首先用 ResNet 主干上的标准特征金字塔提取特征图。对于 18 种目标风格，每一种都训练一个小的特征网络，从对应的特征图中提取学习到的风格，其中小的特征图生成风格 (0-2)，中等的特征图生成风格 (3-6)，最大的特征图生成风格 (7-18)。map2style 映射网络是一个小型的全卷积网络，用一组两步卷积后跟着 LeakyReLU 激活函数来逐渐降低空间大小。每个 map2style 网络生成一个 512 维的向量，从它的匹配仿射变换 A 开始，输入到 StyleGAN 中。

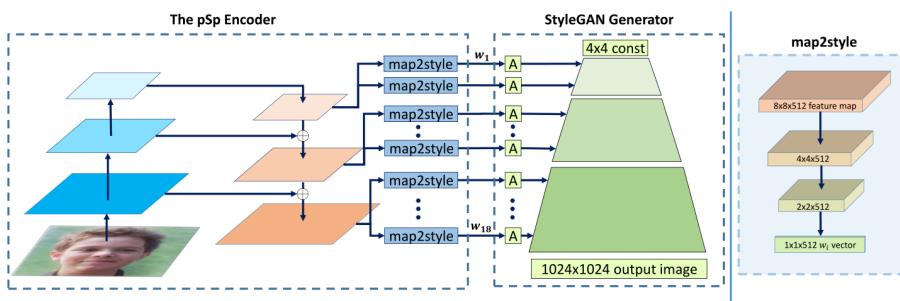


图 3.1 pSp 框架

在整个架构中，只有 Encoder 部分是需要训练的，StyleGAN Generator 部分使用的是预训练好的 StyleGAN2 的模型。首先，该编码器可以直接将真实图像嵌入到 W+ 中，无需额外优化。其次，编码器可以直接解决图像的转换任务。与之前 StyleGAN 编码器的“先反转，后编辑”的方法相比，pSp 不要求输入图像在 StyleGAN 域中进行特征表示，可以直接将这些图像翻译任务编码成 StyleGAN。该编码器极大地简化了训练过程，在没有图像对的标签下提供更好的支持，并且通过 style 的重采样可以支持多模态合成。与以往的工作通常依赖于专用的体系结构来解决单个翻译任务不同，pSp 只需要对训练损失和方法进行微小的更改，就能够解决各种各样的问题。

### 3.1.2 e4e 的 Designing an Encoder for StyleGAN Image Manipulation

#### 1. 背景和动机：

主要工作：提出什么样的 encoder (image->latent code) 具有更好的编辑性和更小的失真。答案：图片逆映射接近 W 空间的 encoder 是好的。要用利用预训练的 stylegan 进行图像编辑，需要将图像映射到 stylegan 的 latent space。stylegan 的 latent space 中存在两种权衡：(1) 扭曲-编辑权衡 distortion-editability tradeoff (2) 扭曲-感知权衡 distortion-perception tradeoff stylegan 的 W 空间具有丰富的解纠缠性质，可用于操作 stylegan 进行各种图像处理。然而，对任何图像用 stylegan 进行处理，必须首先将图像逆映射到 W 空间。高质量的反演 (invert) 方法对于编辑效果至关重要。一个好的反演 (图像->stylegan 的 W 空间) 具有的特性：(1) inversion 得到的 latent code 输入 stylegan 中能够恢复原图像 (2) 能够最大程度地利用 latent space 的编辑能力定义衡量重建性能的三个指标：(1) editability (2) distortion ——per-image input-output similarity (3) perceptual quality ——how realistic the reconstructed image is W 潜在空间的表达性已被证明是有限的，并不是每幅图像都能准确地映射到 W 中。为了克服这一局限性，Abdal 等人证明了任何图像都可以逆映射成 W 的扩展部分，记为 W+。W+ 中的每一个 style code 由许多 style vector 组成。

#### 2. IDEA：

目前基于 styleGAN 的图像编辑，评判的标准有两个，重构出来的效果以及可编辑性的强弱，作者分别用 distortion (扭曲程度) 和 editability (编辑能力) 代表。很可惜的是，一般扭曲程度低的方法，编辑能力弱。这是因为找到的隐向量已经离 stylegan 的 W 空间很远了，不是一个分布。比如 image2style 就提出，在 W+ 空间优化隐向量，可以重构出任意一行图像，不管是不是人脸图像，但可编辑能力大大降低。

作者认为，找到的隐向量为好的标准就是解决 W 空间。接近有两层含义：(1). 每个 style code 之间的方差小；(2) 每个 style code 都在 W 空间中。围绕以上两个原则，作者提出 e4e (encoder for editing)，一个编码器，用于将指定图像映射到隐空间上，同时还提出了一个用于评判隐向量重构性能和可编辑性能的综合性指标。作者用 adversarial training 和 progressive training scheme 等技巧训练了一个 encoder，使得它能将图像逆映射后尽可能地“接近” W 空间。本文的目的是设计编码器能够更好的对图像进行编辑。首先，由于图像失真，感知质量和图像的可编辑性存在紧密的联系。在一定程度上，图像失真与感知质量成负相关，图像失真和图像的可编辑性成负相关。此外，本文还发现如果图像的逆隐空间向量越接近原本的隐空间分布 W，那么图像的感知质量和可编辑性便越好。

#### 3. 贡献：

- (1) 分析了 StyleGAN 的复杂 latent space，并对其结构提出了新的看法。
- (2) 展示了 (distortion) 扭曲 (失真、歪曲、变形)、感知 (perception) 和可编辑性 (editability) 之间固有的权衡。(3) 描述了这些权衡，并设计了两种编码器来控制它们。
- (4) 提出了 e4e，一个创新的编码器专门设计，允许对反转真实图像的接下来的编辑。

#### 4. 主要方法：

图片逆映射接近 W 空间的 encoder 是好的。

一个好的 encoder，需要输出空间接近 W 空间。想要做到这一点：

- (1) 可以优化每个风格向量的方差，让其尽量小，极限的情况是完全一样；
- (2) 并足够接近 stylegan 的 w 空间

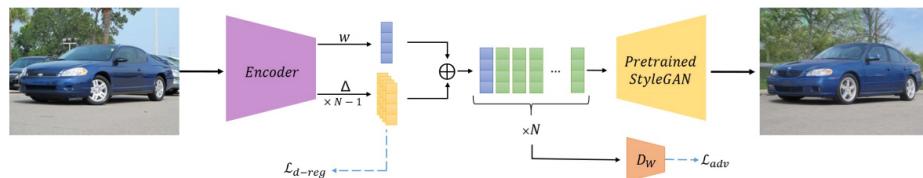


Figure 6: Our e4e network architecture. The encoder receives an input image and outputs a single style code  $w$  together with a set of offsets  $\Delta_1.. \Delta_{N-1}$ , where  $N$  denotes the number of StyleGAN's style modulation layers. We obtain our final latent representation by replicating the  $w$  vector  $N$  times and adding each  $\Delta_i$  to its corresponding entry. During training, the  $\mathcal{L}_{d-reg}$  regularization encourages small variance between different entries of the final representation, thereby remaining close to  $\mathcal{W}_*$ .  $\mathcal{L}_{adv}$  guides each latent code towards the range of StyleGAN's mapping network, resulting in a final representation closer to  $\mathcal{W}^k$ . As a result of applying both regularization terms, the encoder's final learned representation lies close to  $\mathcal{W}$ .

图 3.2 e4e 主要方法说明

#### 5. 优化方差

为了到达第一个目的，作者提出渐进训练方法。首先 encoder 记作 E，输入是指定图像  $x$ 。输出是  $N$  个 style code。

$$E(x) = (w, \Delta_1, \dots, \Delta_{N-1}) \quad (3.1)$$

后面  $N-1$  项是偏置，加在  $w$  上得到  $N$  个 style code，具有不同的值， $N$  就是 stylegan 中的 style modulation 层的数目。该空间记作  $\mathcal{W}_*$  空间。在训练初期，让所有的偏置都为 0，这样  $N$  个向量都是相同的都是  $w$ 。即先鼓励 encoder 往  $\mathcal{W}_*$  空间上靠。然后逐渐的让偏置不一样，这样每个 style modulation 层都有不同的 style code，灵活性更高，保证了重构质量，实现了从  $\mathcal{W}_*$  空间上  $\mathcal{W}_*^k$  的变化。其实如果偏置都为 0，encoder 也倾向于向  $\mathcal{W}$  空间靠。但因为学习偏置的关系，离  $\mathcal{W}$  空间也不远，也保证了可编辑的能力。距离由网络自己学习，自行权重可编辑性和重构性的 tradeoff。为了让偏置临近  $\mathcal{W}_*$  空间，作者设置了一个浅显易懂的正则损失：

$$\mathcal{L}_{\text{ddreg}}(w) = \sum_{i=1}^{N-1} \|\Delta_i\|_2 \quad (3.2)$$

## 6. 优化和 W 空间的距离

因为 styleGAN 的 W 空间并不能显式建模，所有作者使用了对抗思想，设置一个 latent code discriminator(D<sub>w</sub>) 区分 encoder 的分布和 W 空间的分布。用同一个判别器，使用所有 N 个 style code 和真实的原始 W 空间向量。将 N 个 loss 求平均优化。

$$\begin{aligned} \mathcal{L}_{\text{adv}}^D &= -\mathbb{E}_{w \sim \mathcal{W}} [\log D_{\mathcal{W}}(w)] - \mathbb{E}_{x \sim p_X} [\log (1 - D_{\mathcal{W}}(E(x)_i))] + \\ &\quad \mathbb{E}_{\frac{\gamma}{2} w \sim \mathcal{W}} [\|\nabla_w D_{\mathcal{W}}(w)\|_2^2] \end{aligned} \quad (3.3)$$

$$\mathcal{L}_{\text{adv}}^E = -\mathbb{E}_{x \sim p_X} [\log D_{\mathcal{W}}(E(x)_i)] \quad (3.4)$$

### 3.1.3 ReStyle: A Residual-Based StyleGAN Encoder via Iterative Renement

1. Topic: 图像反演，获取潜在编码用于对图像进行直接编辑。

2. Problem: 现在的单次编码架构无法准确的获得图像反推的潜在编码。即在重建精度方面，基于学习的反演方法与基于优化的反演方法仍存在较大差距。因此，虽然在基于学习的反演方面取得了显著的进展，但设计合适的编码器和训练方案仍然是一个挑战，许多工作仍在使用逐图像优化方法。

3. Idea: 使用以残差为基础的迭代反馈调优架构。具体为将上一次迭代的输出与原始的输入图像共同输入到下一次迭代中，进行多次前向传播，让编码器注意到更多能够提升精度的特征。在增加有限的推理时间情况下，实现较高精度的输出。个人理解该方法是逐图像优化方法与编码器学习方法的折中。也是单步反演的松弛求解。

$$\hat{y} = G(E(x)) \quad (3.5)$$

其中 x 为输入图像， $\hat{y}$  为反演的图像，G, E 分别为生成器和编码器。其具体工作如下：

#### 4. Methods:

(1) 聚合真实图像和 Stylegan 的重建图像，生成六通道数据，然后输入到编码器中学习残差偏移编码。

(2) 更新图像 x 的潜在编码，并输入到生成器中生成新重建图像。

(3) 其中初始的潜在编码和对应的潜在图像分别是训练好的生成器的平均风格向量和其对应生成图片。

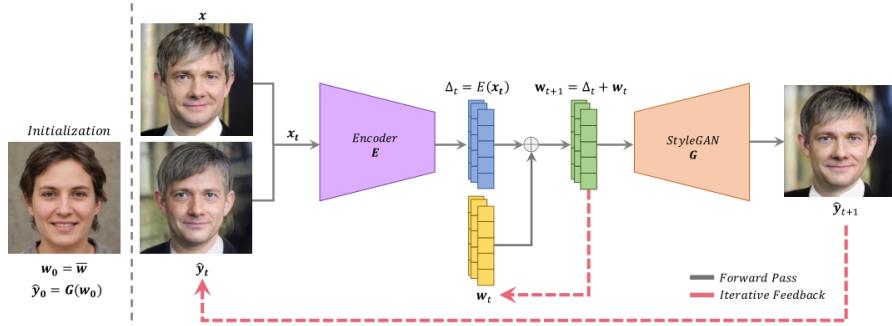


Figure 2. *Our ReStyle iterative inversion scheme*. Given an input image  $x$ , the scheme is initialized with the average latent code  $w_0$  and its corresponding image  $\hat{y}_0$ . Consider step  $t$ . ReStyle operates on an extended input obtained by concatenating  $x$  with the image  $\hat{y}_t$  corresponding to the current inversion prediction  $w_t \in \mathcal{W}^+$  (shown in yellow). The encoder  $E$  is then tasked with predicting a *residual* latent code,  $\Delta_t \in \mathcal{W}^+$  (shown in blue). The predicted residual is then added to the previous latent code  $w_t$  to obtain the updated latent code prediction  $w_{t+1}$  (shown in green). Finally, passing the newly computed latent code to the generator  $G$  results in an updated reconstruction  $\hat{y}_{t+1}$ , which is then passed as input in the following step. During training, the loss objectives are computed at each forward pass with back-propagation performed accordingly. A similar multi-step process is performed during inference.

图 3.3 ReStyle 说明

## 5. ReStyle 迭代反演方案

给定一个输入图像，用平均潜在码  $w_0$  及其对应图像  $\hat{y}_0$  作为初始化值。考虑  $t$  步的情况。ReStyle 对一个扩展的输入进行操作，该输入通过将  $x$  与对应于当前反转预测  $w_t \in W^+$  的图像  $\hat{y}_t$  (如图黄色部分所示)。然后，编码器  $E$  的任务是预测剩余潜在码， $tW^+$ (如图蓝色部分表示)。然后将预测残差加到前一个潜在码上，得到更新后的潜在码  $w_{t+1}$  (如图绿色所示)。最后，在更新的重构  $\hat{y}_{t+1}$  中将新计算的潜在代码传递给生成器  $G$  生成的结果，然后在接下来的步骤中将其作为输入传递。在训练过程中，在每次前向传递时计算损失目标，并相应地进行反向传播。在推理过程中也会执行类似的多步骤过程。

## 6. 模型：

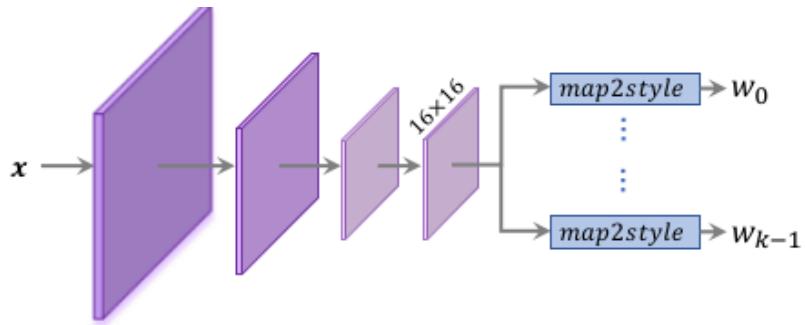


Figure 3. *Our simplified encoder architecture*. All  $k$  input style vectors of the generator are extracted from the encoder's final  $16 \times 16$  feature map which is passed through  $k$  *map2style* blocks [35].

图 3.4 ReStyle 模型架构

7. 编码器模型：因为迭代调优的特性，SOTA 工作中的特征金字塔和残差模块在

该网络中并不需要。文中采用简单的特征下采样网络并结合 StyleGAN 的需要的风格输入作为输出，如 ReStyle 模型架构图。

## 3.2 MetaFormer 及其相关原理介绍

Transformer 与 2017 年由谷歌团队提出，其初衷是为了解决自然语言处理领域的问题。近年来，随着 ViT 模型首次将 Transformer 引入计算机视觉领域，其在视觉领域也开始显现出越来越大的潜力。早期人们普遍存在这样一种观点：Transformer 结构之所以能够取得这么大的成功，主要归功为注意力机制。因此，很多研究者对于注意力模块做了很多改进。然而，后续的研究发现即使在 Transformer 结构中不使用注意力机制，也同样可以取得很好的效果，例如 Spatial MLP 等等。在这种背景下，Meta-Former 得以提出。

### 3.2.1 MetaFormer

MetaFormer 的含义是对 Transformer 整体结构的一种抽象，因为作者认为：Transformer 中具体模块并不重要，重要的是它的整体架构。基于这种观点，作者将 Attention 机制抽象为一个信息混合和交换的模块。MetaFormer 的具体结构如图中所示：左图是 MetaFormer 的整体结构，它不特指某一种具体的模型。右图是 MetaFormer 的具体结构模型，不同的是，Transformer 采用了基于 Attention 的 Token Mixer，而 MLP-like 模型采用的是 Spatial MLP 作为 Token Mixer。为了验证作者的观点，作者还提出了一种结构极其简单的 PoolFormer，仅仅使用 Pooling 来代替 Transformer 中的注意力机制。

### 3.2.2 PoolFormer

同 ViT 的思路类似，PoolFormer 首先进行 Patch embedding，通过 Patch embedding 将图片单位化。之后结合金字塔结构，将整个模型分为几个不同的阶段，每一个阶段的 Patch size 逐渐提高，从而使得模型的整体感受野逐渐增大。图中 b 描述了 PoolFormer block 的具体结构，可以看出其余 Transformer 结构基本类似，只是将其中的注意力机制变成了池化操作。通过这种极其简单的设计，PoolFormer 同样在很多任务中取得了较好的效果，但是相比之下，它的参数量和浮点计算都要小得多。

### 3.2.3 PoolFormer 与 ReStyle-GAN 模型的结合

基于 Transformer 在计算机视觉领域不断取得的重大突破，我们尝试将 Transformer 引入图像生成领域。PoolFormer 结构简单，实现容易，并且容易在计算资源有限的条件下进行训练，因而我们选择将 PoolFormer 结构作为 ReStyle-GAN 网络的 Encoder，来

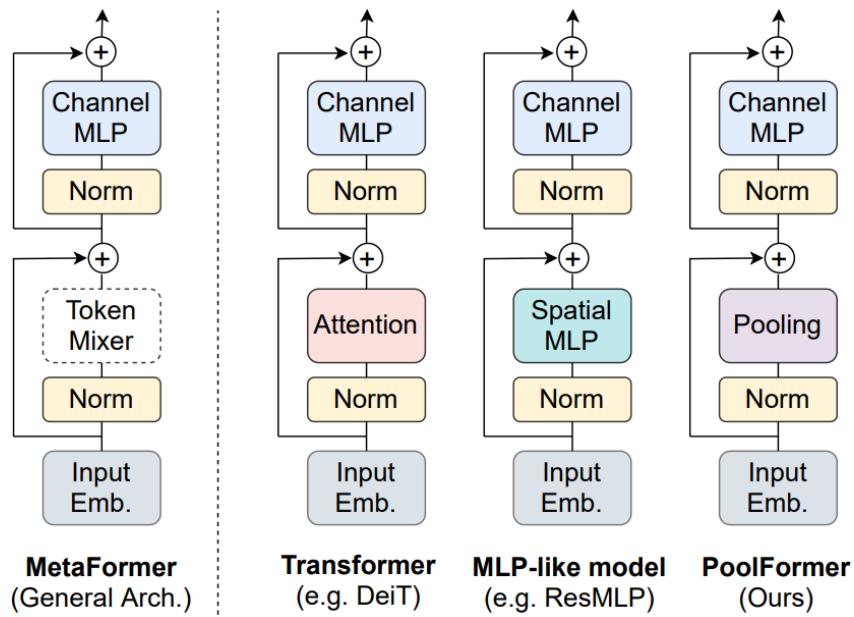


图 3.5 MetaFormer 的结构

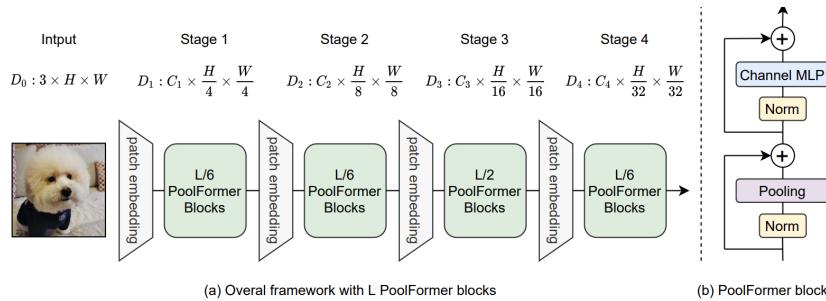


图 3.6 PoolFormer 的整体架构

尝试将类 Transformer 结构引入我们所研究的领域。下一步，我们将进行实验初步探索 Pool Former 与 ReStyle-GAN 结合的效果，并在之后的探索中设计更加合理的 Former 结构，来对图像生成领域进行更深入的探索。

### 3.3 实验条件、客观结果与分析

#### 3.3.1 实验细节

我们在所提出模型上进行了实验验证。所使用的显卡类型为 12G Titan Xp，所采用的深度学习框架为 Pytorch。实验采用高清人脸数据集（Flickr-Faces-High-Quality, FFHQ），其中含有 70000 张分辨率为 1024\*1024 高清人脸数据，我们将训练集和测试集按照 4:1 划分：即 56000 张训练集，14000 张测试集图片，并将其分辨率缩放为 256\*256。训练阶段超参数如下：批次大小为 3，训练迭代轮次为 3，约 75000 次迭代。

为保证实验一致性，我们设置  $\lambda_{L_2} = 1, \lambda_{\text{lpips}} = 0.8$ 。

### 3.3.2 对比模型

为与所设计模型进行有效对比，我们选取了两个目前最佳的骨干网络模型进行了对比实验，分别为 ResNet34 和 ResNet152，其中 ResNet34 为原始 Restyle 模型的编码器。训练后的损失函数值如下表。通过下表，可见无论是逐像素的  $L_2$  损失还是图片相似度损失 LPIPS，所设计架构在训练完成后的数值都比主流架构低，具体来说，所提出 PoolFormer 在  $L_2$  损失和图片相似度损失 LPIPS 分别比 ResNet 低 6.62% 和 4.96%，代表所设计模型具有良好的图片重构能力与风格转换能力，所生成图片像素信息和语义信息损失较少，主要原因因为 PoolFormer 具有全局关注的注意力机制，较 ResNet 来说，PoolFormer 骨干网络能够更加全面地关注图片的风格特征。

表 3.1 训练结束后模型损失函数值对比

Encoder Architecture	$L_2$ Loss	LPIPS Loss
ResNet34	0.0302	0.2864
ResNet152	0.0269	0.2670
PoolFormer	0.0282	0.2722

此外，我们还对比了模型之间的参数量和推断时间，用以反映所提出模型的效率，如表 2。在相同软硬件平台上，PoolFormer 在参数量更少的情况下，实现了更短时间的图片重建，其推断速度较 Restyle 提升了 11.77%，而参数量更少，进一步说明所设计模型相比 Restyle，图片重建效率更高。对比以 ResNet152 作为编码器的网络，PoolFormer 参数量远小于 ResNet152，而推断速度也约为其两倍。虽然训练后的重建损失略大，但是综合所提出模型的参数量和效率，PoolFormer 更优。

表 3.2 模型编码器参数量和图像重建时间

Encoder Architecture	编码器参数量 (Mb)	图像重建时间 (秒)
ResNet34	216M	0.17s
ResNet152	581M	0.31s
PoolFormer	209M	0.15s

## 3.4 主观结果、主观分析

### 3.4.1 ReStyle 重建人脸图像

此节图中所示为使用不同编码器的 ReStyle 重建人脸图像的结果。其中最左侧一列图像为原始图像，其余三列图像从左至右分别为第一至三次迭代重建的图像。从图中

可以看出对于所有三种编码器，其对图像结构和语义的重建在第一次迭代就基本完成，后续的迭代主要是对图像的纹理细节进行提取和重建。三种编码器对图像语义特征的重建效果比较接近，但是可以看出基于 ResNet34 编码器提取特征的准确性明显低于其他两种编码器：其对面部表情特征的提取不够准确，同时也将图像左下角处的衣领误认为毛发。对于 ResNet152 编码器和 PoolFormer 编码器，可以看出 ResNet152 对毛发的纹理细节提取更为细节，这也是其 L2 和 LPIPS 误差较小的原因；PoolFormer 则对人脸中对一些皮肤细节进行了更好的提取，这在人脸的重建中是更为重要的。但是使用三种编码器重建的人脸图像都出现了模糊的现象，这是使用 L2 损失训练编码器不可避免会出现的问题，我们认为原文在重建人脸图像时使用的身份损失和特征编码的正则化会有助于解决模糊现象。



图 3.7 使用 (a) ResNet34、(b) ResNet152、(c) PoolFormer 作为编码器的 ReStyle 在 FFHQ 人脸数据集上的图像重建效果。最左侧一列为原始图像，其余三列从左至右分别为第一至三次迭代重建的图像。

### 3.4.2 不同编码器提取的人脸特征编码

此节图中所示为使用不同编码器提取的人脸特征编码实现人脸混合的效果，使用最左侧一列原始图像中提取的底层人脸编码（决定结构特征）和最上方一行原始图像中提取的高层人脸编码（决定风格特征）生成图像。

可以看出，三种编码器中，PoolFormer 提取的特征更为准确，生成图像很好地保持了左侧图像的结构特征，并获取了上方图像的纹理特征，生成的图像在发色、肤色方面与上方图像基本一致。而使用 ResNet 编码器提取的特征出现了结构和风格耦合的现象，例如肤色和发色介于两张原始图像中间，或是混合了上方图像的一部分结构特征。

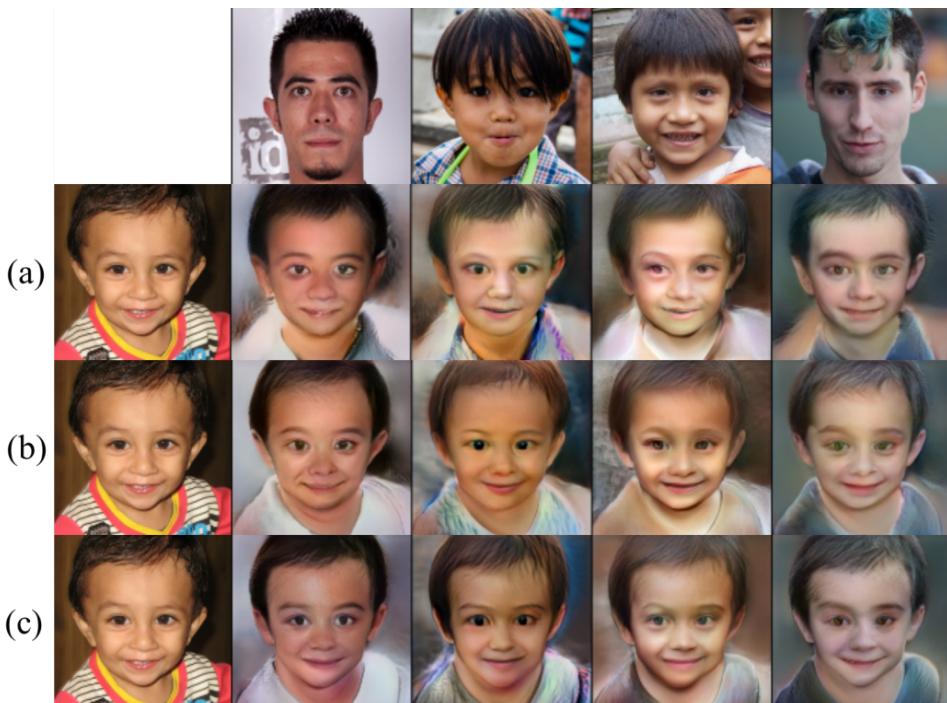


图 3.8 使用 (a) ResNet34、(b) ResNet152、(c) PoolFormer 作为编码器的 ReStyle 在 FFHQ 人脸数据集上的特征混合效果。

## 第四章 总结与分析

作为目前最热门的研究领域之一，以 GAN 为代表的图像生成模型，通过对抗博弈的方式实现了对自然图像的高维复杂分布的建模，以求实现逼真的自然图像的生成。后续的研究通过优化网络结构、限制参数和渐进式训练等方式进一步提高了图像生成的性能，并稳定了模型的训练。

这些研究为一些更具实用性的研究奠定了基础，例如通过标签控制生成图像的类别以实现分类模型训练中的数据增强，或是实现不同域的图像之间的相互转换。然而，上述无条件生成模型生成的图像普遍存在分辨率低、质量差、语义不符合现实等问题。因此，Karras 等人提出了一系列新的图像生成模型架构和训练方法，实现了高分辨率、高质量的图像生成。这些图像生成模型通过训练，很好地建模了真实图像分布的结构特征和纹理细节，因此这些生成模型也可以被用作预训练模型，利用其学习到的先验知识，提升其他任务的深度学习模型的性能，例如将先验知识应用在图像超分辨率、上色、去噪等任务中，以预测更为合理和细节的图像结构和纹理特征，提高相关模型的性能。这些预训练生成模型同样可以被用于图像编辑等实际应用中：通过图像编码模型将待编辑的图像映射到隐向量空间中，可以通过简单、低维的向量编辑，实现风格互换、换脸等复杂、高维的图像编辑操作。

而最近提出 StyleGAN3 模型对于生成图像的内部表征进行了进一步优化，从而在亚像素尺度上实现了生成图像的平移和旋转不变性。一方面，这使得更为复杂的视频和动画生成变得更为现实，另一方面，StyleGAN3 的出现也使得实现超高维的视频空间到低维线性空间之间的映射、通过简单线性操作实现复杂的视频语义编辑成为了可能。

综上所述，StyleGAN 等图像模型在训练生成能力的同时，也学习到了图像结构特征和纹理细节等先验知识，使得这些预训练生成模型可以被应用于图像超分辨率、图像上色、图像去噪、图像编辑等各类实际应用中，并为相关任务带来显著的性能提升。不难看出，预训练图像生成模型具有极高的应用潜力和价值，但是如何将其更好的应用于特定任务中是一个难点，也会是本作业的研究重点。

最后本作业先是解释了 pSp、e4e 与 Restyle 的流程跟说明，并解释 Transformer 到 MetaFormer 与 PoolFormer 之间的关系与结构原理与状况后下，进行 PoolFormer 与 ReStyle-GAN 模型的结合改进，并详细说明，并解释客观条件的实验与主观条件的实验过程。



## 参考文献

- [1] HERTZMANN A, JACOBS C E, OLIVER N, et al. Image analogies[J]. Proceedings of the 28th annual conference on Computer graphics and interactive techniques. ACM., 2001.
- [2] GATYS L, ECKER A S. Matthias Bethge[J]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.
- [3] J.Johnson, A.Alahi, L.Fei-Fei. Perceptual losses for real-time style transfer and super-resolution[J]. European Conference on Computer Vision, 2016.
- [4] J G, Y S, B. Z. Image processing using multi-code gan prior[J]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020.
- [5] Y A, O P, D. C O. Restyle: A residual-based stylegan encoder via iterative refinement[J]. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021.
- [6] J.GOODFELLOW I, POUGET-ABADIE J, MIRZA M, et al. Yoshua Bengio: Generative Adversarial Nets[J]. NIPS, 2014.
- [7] RADFORD A, METZ L. Soumith Chintala:Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks[J]. ICLR (Poster), 2016.
- [8] ISOLA P, ZHU J Y, ZHOU T. Image-To-Image Translation With Conditional Adversarial Networks[J]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
- [9] ZHU J Y, PARK T, ISOLA P. Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks[J]. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.
- [10] KARRAS T, LAINE S, AILA T. A Style-Based Generator Architecture for Generative Adversarial Networks[J]. CVPR, 2019.
- [11] KARRAS T, LAINE S, AITTALA M, et al. Analyzing and Improving the Image Quality of StyleGAN[J]. CoRR, 2019.
- [12] KARRAS T, AITTALA M, LAINE S, et al. Alias-Free Generative Adversarial Networks[J]. CoRR, 2021.
- [13] XIA W, ZHANG Y, YANG Y, et al. GAN Inversion: A Survey[J]. CoRR, 2021.
- [14] TOV O, ALALUF Y, NITZAN Y, et al. Designing an encoder for StyleGAN image manipulation[J]. ACM, 2021.
- [15] RICHARDSON E, ALALUF Y, PATASHNIK O, et al. Encoding in Style: A StyleGAN Encoder for Image-to-Image Translation[J]. CVPR, 2021.
- [16] ALALUF Y, PATASHNIK O, COHEN-OR D. ReStyle: A Residual-Based StyleGAN Encoder via Iterative Refinement[J]. CoRR, 2021.
- [17] KAFRI O, PATASHNIK O, ALALUF Y, et al. StyleFusion: A Generative Model for Disentangling Spatial Segments[J]. CoRR, 2021.

- [18] A R, L M, S C. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. ArXiv preprint, 2015.
- [19] M A, S C, L B. Wasserstein generative adversarial networks[J]. International conference on machine learning, 2017.
- [20] GULRAJANI I, AHMED F, ARJOVSKY M, et al. Improved training of wasserstein gans[J]. ArXiv preprint, 2017.
- [21] KARRAS T, AILA T, LAINE S, et al. Progressive growing of gans for improved quality, stability, and variation[J]. ArXiv, 2017.
- [22] M M, S O. Conditional generative adversarial nets[J]. ArXiv preprint, 2014.
- [23] A A, A S, H E. Data augmentation generative adversarial networks[J]. ArXiv preprint, 2017.
- [24] J G, Y S, B Z. Image processing using multi-code gan prior[J]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020.