Image2StyleGAN++: How to Edit the Embedded Images?
图像 2stylegan + + : 如何编辑嵌入式图像？

Rameen Abdal
拉面 Abdal
KAUST
卡斯特
rameen.abdal@kaust.edu.sa
Rameen.abdal@kaust.edu. sa

Yipeng Qin
秦一鹏
Cardiff University
卡迪夫大学
qiny16@cardiff.ac.uk
Qini16@cardiff.ac.uk

Peter Wonka
Peter Wonka
KAUST
卡斯特
pwonka@gmail.com
Pwonka@gmail.com

(a) (a)(b)(c)(d) (b) (c) (d)

Figure 1: (a) and (b): input images; (c): the "two-face" generated by naively copying the left half from (a) and the right half from (b); (d): the "two-face" generated by our Image2StyleGAN++ framework.
图 1: (a)和(b) : 输入图像; (c) : 通过天真地从(a)和右边复制左边的一半而产生的"双面"来自(b) ; (d) : 我们的 Image2StyleGAN + + 框架生成的"双面"。

Abstract
摘要

We propose Image2StyleGAN++, a flexible image edit-ing framework with many applications. Our framework ex-tends the recent Image2StyleGAN [1] in three ways. First, we introduce noise optimization as a complement to the W^+ *latent space embedding. Our noise optimization can restore high frequency features in images and thus significantly im-proves the quality of reconstructed images,* e.g. *a big in-crease of PSNR from 20 dB to 45 dB. Second, we extend the global* W^+ *latent space embedding to enable local embed-dings. Third, we combine embedding with activation tensor manipulation to perform high quality local edits along with global semantic edits on images. Such edits motivate vari-ous high quality image editing applications,* e.g. *image re-construction, image inpainting, image crossover, local style transfer, image editing using scribbles, and attribute level feature transfer. Examples of the edited images are shown across the paper for visual inspection.*

我们提出 Image2StyleGAN + + ，一个灵活的图像编辑框架，具有许多应用程序。我们的框架在三个方面扩展了最近的 Image2StyleGAN [1]。首先，我们引入噪声优化作为 w + 潜在空间嵌入的补充。我们的噪声优化可以恢复图像中的高频特征，从而显著提高重建图像的质量，如 PSNR 从 20 分贝大幅提高到 45 分贝。其次，我们扩展了全局 w + 潜在空间嵌入以实现局部嵌入。第三，我们将嵌入与激活张量操作相结合，以执行高质量的局部编辑以及图像上的全局语义编辑。这样的编辑激发了各种高质量的图像编辑应用，如图像重建、图像修复、图像交叉、局部风格转换、使用涂鸦的图像编辑和属性级别的特征转换。编辑过的图像的例子在纸上显示，以供视觉检查。

1. Introduction
引言

Recent GANs [19, 6] demonstrated that synthetic im-ages can be generated with very high quality. This mo-

tivates research into embedding algorithms that embed a given photograph into a GAN latent space. Such embed-

最近甘斯[19,6]证明，合成图像可以产生非常高的质量。这激发了将给定照片嵌入 GAN 潜在空间的嵌入算法的研究。这样的嵌入

ding algorithms can be used to analyze the limitations of GANs [5], do image inpainting [8, 39, 38, 36], local image editing [40, 17], global image transformations such as image morphing and expression transfer [1], and few-shot video generation [35, 34].

Ding 算法可用于分析 GANs [5]的局限性，进行图像修补[8,39,38,36] ，局部图像编辑[40,17] ，全局图像变形和表达式转换[1] ，少镜头视频生成[35,34]。

In this paper, we propose to extend a very recent embedding algorithm, Image2StyleGAN [1]. In particular, we would like to improve this previous algorithm in three aspects. First, we noticed that the embedding quality can be further improved by including Noise space optimization into the embedding framework. The key insight here is that stable Noise space optimization can only be conducted if the optimization is done sequentially with W^+ space and not jointly. Second, we would like to improve the capabili-ties of the embedding algorithm to increase the local control over the embedding. One way to improve local control is to include masks in the embedding algorithm with undefined content. The goal of the embedding algorithm should be to find a plausible embedding for everything outside the mask, while filling in reasonable semantic content in the masked pixels. Similarly, we would like to provide the option of approximate embeddings, where the specified pixel colors are only a guide for the embedding. In this way, we aim to achieve high quality embeddings that can be controlled by user scribbles. In the third technical part of the paper, we investigate the combination of embedding algorithm and di-

在本文中，我们提出扩展一个非常新的嵌入算法，Image2StyleGAN [1]。特别是，我们想从三个方面改进以前的算法。首先，我们注意到嵌入质量可以通过在嵌入框架中引入噪声空间优化来进一步提高。这里的关键观点是，稳定的噪声空间优化只能进行，如果优化是顺序进行的 w + 空间，而不是联合。其次，我们希望提高嵌入算法的能力，以增加对嵌入的局部控制。提高局部控制的一个方法是在嵌入算法中包含含有未定义内容的掩码。嵌入算法的目标应该是为掩码之外的所有内容找到一个合理的嵌入，同时在掩码像素中填充合理的语义内容。同样，我们也想提供近似嵌入的选项，其中指定的像素颜色只是嵌入的指南。通过这种方式，我们的目标是实现高质量的嵌入，可以通过用户涂鸦来控制。在论文的第三个技术部分，我们研究了嵌入算法和 di- 的结合

18296

rect manipulations of the activation maps (called activation tensors in our paper).
激活图的直接操作(在我们的论文中称为激活张量)。

Our main contributions are:
我们的主要贡献是:

1. We propose Noise space optimization to restore the high frequency features in an image that cannot be re-produced by other latent space optimization of GANs. The resulting images are very faithful reconstructions of up to 45 dB compared to about 20 dB (PSNR) for the previously best results.
我们提出噪声空间优化来恢复图像中的高频特征，这些特征是其他甘斯潜在空间优化不能再现的。由此产生的图像是非常忠实的重建高达 45 分贝，而约 20 分贝(PSNR)为以前最好的结果。

2. We propose an extended embedding algorithm into the
我们提出了一个扩展嵌入算法到
W^+ space of StyleGAN that allows for local modifica-tions such as missing regions and locally approximate embeddings.
StyleGAN 的 w + 空间，允许局部修改，如缺失区域和局部近似嵌入。

3. We investigate the combination of embedding and activation tensor manipulation to perform high quality local edits along with global semantic edits on images.
我们研究嵌入和激活张量操作的组合，以执行高质量的局部编辑以及图像上的全局语义编辑。

4. We apply our novel framework to multiple image edit-ing and manipulation applications. The results show that the method can be successfully used to develop a state-of-the-art image editing software.
我们将我们的新框架应用于多种图像编辑和处理应用。结果表明，该方法可以成功地用于开发最先进的图像编辑软件。

2. Related Work
相关工作

Generative Adversarial Networks (GANs) [14, 29] are one of the most popular generative models that have been successfully applied to many computer vision applications, *e.g.* object detection [23], texture synthesis [22, 37, 31], image-to-image translation [16, 42, 28, 25] and video gen-eration [33, 32, 35, 34]. Backing these applications are the massive improvements on GANs in terms of architec-ture [19, 6, 28, 16], loss function design [26, 2], and regu-larization [27, 15]. On the bright side, such improvements significantly boost the quality of the synthesized images. To date, the two highest quality GANs are StyleGAN [19] and BigGAN [6]. Between them, StyleGAN produces excellent results for unconditional image synthesis tasks, especially on face images; BigGAN produces the best results for con-ditional image synthesis tasks (*e.g.* ImageNet [9]). While on the dark side, these improvements make the training of GANs more and more expensive that nowadays it is almost a privilege of wealthy institutions to compete for the best performance. As a result, methods built on pre-trained gen-erators start to attract attention very recently. In the follow-ing, we would like to discuss previous work of two such ap-proaches: embedding images into a GAN latent space and the manipulation of GAN activation tensors.

生成对抗网络[14,29]是最流行的生成模型之一，已成功应用于许多计算机视觉应用，如目标检测[23]，纹理合成[22,37,31]，图像到图像的转换[16,42,28,25]和视频生成[33,32,35,34]。支持这些应用的是 GANs 在体系结构方面的巨大改进[19,6,28,16]，损失函数设计[26,2]和规范化[27,15]。好的一面是，这样的改进显着提高了合成图像的质量。迄今为止，两个质量最高的甘斯是 StyleGAN [19]和 BigGAN [6]。在两者之间，StyleGAN 在无条件图像合成任务中，尤其是在人脸图像上，产生了极好的结果; BigGAN 在有条件的图像合成任务中，产生了最好的结果(例如 ImageNet [9])。然而，这些改进使得甘斯的培训费用越来越昂贵，如今，竞争最佳表现几乎成了富裕机构的特权。因此，建立在预先训练的发电机上的方法最近开始引起人们的注意。在下文中，我们将讨论以前两种方法的工作: 在 GAN 潜在空间中嵌入图像和操纵 GAN 活化张量。

Latent Space Embedding. The embedding of an image into the latent space is a longstanding topic in both machine learning and computer vision. In general, the embedding
潜在空间嵌入。将图像嵌入到潜在空间是机器学习和计算机视觉中一个长期存在的话题。一般来说，嵌入

can be implemented in two ways: i) passing the input im-age through an encoder neural network (*e.g*. the Variational Auto-Encoder [21]); ii) optimizing a random initial latent code to match the input image [41, 7]. Between them, the first approach dominated for a long time. Although it has an inherent problem to generalize beyond the training dataset, it produces higher quality results than the naive latent code optimization methods [41, 7]. While recently, Abdal *et al*. [1] obtained excellent embedding results by optimizing the latent codes in an enhanced $\mathcal{W}^+$ latent space instead of the initial $\mathcal{Z}$ latent space. Their method suggests a new direc-tion for various image editing applications and makes the second approach interesting again.

可以通过两种方式实现：i)通过编码器神经网络(例如Variational Auto-Encoder [21])传递输入图像；ii)优化随机初始潜在代码以匹配输入图像[41,7]。在他们之间，第一种方法占主导地位很长时间。虽然它有一个固有的问题，泛化超出了训练数据集，它产生了更高的质量结果比幼稚的潜在代码优化方法[41,7]。最近，Abdal 等[1]通过在增强的 w + 潜在空间中优化潜在码而不是在初始的 z 潜在空间中优化潜在码，得到了很好的嵌入结果。他们的方法为各种图像编辑应用程序提供了一个新的方向，并使第二种方法再次变得有趣。

Activation Tensor Manipulation. With fixed neural net-work weights, the expression power of a generator can be fully utilized by manipulating its activation tensors. Based on this observation, Bau [4] *et al*. investigated what a GAN can and cannot generate by locating and manipulating rel-evant neurons in the activation tensors [4, 5]. Built on the understanding of how an object is "drawn" by the genera-tor, they further designed a semantic image editing system that can add, remove or change the appearance of an object in an input image [3]. Concurrently, Fruhst¨uck¨ *et al*. [11] investigated the potential of activation tensor manipulation in image blending. Observing that boundary artifacts can be eliminated by by cropping and combining activation tensors at early layers of a generator, they proposed an algorithm to create large-scale texture maps of hundreds of megapixels by combining outputs of GANs trained on a lower resolu-tion.

激活张量操作。使用固定的神经网络权重，通过操纵激活张量，可以充分利用发生器的表达能力。基于这一观察，Bau [4]等人通过定位和操纵激活张量中的相关神经元来研究 GAN 能够和不能产生什么[4,5]。在理解生成器如何"绘制"对象的基础上，他们进一步设计了一个语义图像编辑系统，可以添加、删除或改变输入图像中对象的外观[3]。同时，Fruhst uck 等[11]研究了激活张量操作在图像混合中的潜力。他们观察到边界伪影可以通过裁剪和合并生成器早期层的激活张量来消除，他们提出了一种算法，通过合并受过较低分辨率训练的 GANs 的输出来创建数百万像素的大规模纹理映射。

3. Overview
3. 概览

Our paper is structured as follows. First, we describe an extended version of the Image2StyleGAN [1] embedding algorithm (See Sec. 4). We propose two novel modifica-tions: 1) to enable local edits, we integrate various spatial masks into the optimization framework. Spatial masks en-able embeddings of incomplete images with missing values and embeddings of images with approximate color values such as user scribbles. In addition to spatial masks, we ex-plore layer masks that restrict the embedding into a set of selected layers. The early layers of StyleGAN [19] encode content and the later layers control the style of the image. By restricting embeddings into a subset of layers we can better control what attributes of a given image are extracted.

我们的论文结构如下。首先，我们描述了Image2StyleGAN [1]嵌入算法的扩展版本(见第 4 节)。我们提出了两个新的修改: 1)使局部编辑成为可能，我们将各种空间掩模整合到优化框架中。空间掩码能够嵌入缺失值的不完整图像和具有近似颜色值的图像，如用户涂鸦。除了空间蒙版，我们还探索了层蒙版，它限制了嵌入到一组选定的图层中。StyleGAN [19]的早期层对内容进行编码，后面的层控制图像的样式。通过限制嵌入到一个子集的图层，我们可以更好地控制什么属性的给定图像被提取。

2) to further improve the embedding quality, we optimize for an additional group of variables $\mathcal{N}$ that control additive noise maps. These noise maps encode high frequency de-tails and enable embedding with very high reconstruction quality.

为了进一步提高嵌入质量，我们优化了一组额外的变量 n 控制加性噪声映射。这些噪声图对高频细节进行编码，并使嵌入具有非常高的重建质量。

Second, we explore multiple operations to directly ma-nipulate activation tensors (See Sec. 5). We mainly explore

其次，我们探索多种操作直接操纵激活张量(见第 5 节)。我们主要探索

8297

(a) (b) (c) (d) (e)
(a)(b)(c)(d)(e)

Figure 2: Joint optimization. (a): target image; (b): image embedded by jointly optimizing w and n using perceptual and
图 2: 联合优化。(a) : 目标图像; (b) : 图像嵌入联合优化 w 和 n 使用感知和
pixel-wise MSE loss; (c): image embedded by jointly optimizing w and n using the pixel-wise MSE loss only; (d): the result of the previous column with n resampled; (e): image embedded by jointly optimizing w and n using perceptual and pixel-wise MSE loss for w and pixel-wise MSE loss for n.
(c)仅使用像素级 MSE 损耗联合优化 w 和 n 的嵌入图像; (d)使用 n 重采样的前一列的结果; (e)使用感知和像素级 MSE 损耗联合优化 w 和 n 的嵌入图像，使用 w 和像素级 MSE 损耗的嵌入图像，使用 n 的像素级 MSE 损耗的嵌入图像。

(a) (b) (c) (d)
(a)(b)(c)(d)

Figure 3: Alternating optimization. (a): target image; (b): image embedded by optimizing w only; (c): taking w from the
图 3: 交替优化。(a) : 目标图像; (b) : 仅通过优化 w 嵌入的图像; (c) : 从
previous column and subsequently optimizing n only; (d): taking the result from the previous column and optimizing w only.
(d) : 获取前一列的结果并仅优化 w。

spatial copying, channel-wise copying, and averaging,

Interesting applications can be built by combining mul-
空间复制，渠道明智的复制，和平均，有趣的应用程序可以通过组合多

tiple embedding steps and direct manipulation steps. As a stepping stone towards building interesting application, we describe in Sec. 6 common building blocks that consist of specific settings of the extended optimization algorithm.
二次嵌入步骤和直接操作步骤。作为构建有趣应用程序的垫脚石，我们在 sec. 6 中描述了由扩展优化算法的特定设置组成的通用构建块。

Finally, in Sec. 7 we outline multiple applications enabled by Image2StyleGAN++: improved image reconstruc-tion, image crossover, image inpainting, local edits using scribbles, local style transfer, and attribute level feature transfer.

最后，在第 7 节中，我们概述了 Image2StyleGAN + + 支持的多种应用: 改进的图像重建、图像交叉、图像修复、使用涂鸦的局部编辑、局部样式转换和属性级别特征转换。

4. An Extended Embedding Algorithm
4. 一种扩展嵌入算法

We implement our embedding algorithm as a gradient-based optimization that iteratively updates an image starting from some initial latent code. The embedding is performed into two spaces using two groups of variables; the semantically meaningful W^+ space and a Noise space N_S encoding high frequency details. The corresponding groups of variables we optimize for are $w \in W^+$ and $n \in N_S$. The inputs to the embedding algorithm are target RGB images x and y

我们实现了我们的嵌入算法作为一个基于梯度的优化迭代更新图像从一些初始的潜在代码开始。嵌入使用两组变量进行到两个空间中: 语义有意义的 w + 空间和编码

高频细节的噪声空间 Ns。我们优化的对应变量组是 w ∈ w + 和 n ∈Ns。嵌入算法的输入是目标 RGB 图像 x 和 y

(they can also be the same image), and up to three spatial masks (M**s**, M**m**, and M**p**)
(它们也可以是相同的图像)，最多三个空间掩模(Ms，Mm，Mp)

Algorithm 1 is the generic embedding algorithm used in the paper.
算法 1 是论文中使用的通用嵌入算法。

4.1. Objective Function
4.1. 目标功能

Our objective function consists of three different types of loss terms, *i.e*. the pixel-wise MSE loss, the perceptual loss [18, 10], and the style loss [12].
我们的目标函数由三种不同类型的损失项组成，即像素级 MSE 损失，感知损失[18,10]和样式损失[12]。

$$
\begin{aligned}
L &= \lambda_{\mathbf{s}} L_{\mathbf{style}}(M_{\mathbf{s}}, G(w, n), y) \\
L &= \lambda_{\mathbf{s}} \text{lstyle}\ (\text{Ms},\ g\ (w,\ n)\ ,\ y) \\
&\quad \lambda_{\text{mse}} \\
&\quad \frac{1}{\lambda_{\text{mse}}} \\
&\quad \frac{1}{N} \\
&+ \quad \text{kMm} \odot (G(w, n) - x)\text{k2}^2 \\
&\quad \text{kMm} \odot (g\ (w,\ n)\text{-x})\ \text{k2} \\
&\quad \lambda_{\text{mse}} \\
&\quad 2 \\
&\quad \lambda_{\text{mse}} \\
&\quad \frac{2}{N} \\
&+ \quad k(1 - Mm) \odot (G(w, n) - y)\text{k2}^2 \\
&\quad \text{K (1-Mm)} \odot (g\ (w,\ n)\text{-y})\ \text{k2} \\
&+ \lambda_{\mathbf{p}} L_{\mathbf{percept}}(M_{\mathbf{p}}, G(w, n), x) \\
&+ \lambda_{\text{p}} \text{lcept}\ (\text{Mp},\ g\ (w,\ n)\ ,\ x)
\end{aligned}
\tag{1}
$$

Where M**s**, M**m** , M**p** denote the spatial masks, ⊙ denotes the Hadamard product, G is the StyleGAN generator, n are the Noise space variables, w are the W+ space variables, L**style** denotes style loss from 'conv3 3' layer of an Im-ageNet pretrained VGG-16 network [30], L**percept** is the
其中 Ms，Mm，Mp 表示空间掩码，⊙表示 Hadamard 乘积，g 表示 StyleGAN 生成器，n 表示 Noise 空间变量，w 表示 w + 空间变量，l 样式表示 Im-ageNet 预训练 VGG-16 网络的' conv33'层的样式丢失[30]，l 感知是

8298

Figure 4: First column: original image; Second column: image embedded in W^+ Space (PSNR 19 to 22 dB); Third column: image embedded in W^+ and Noise space (PSNR 39 to 45 dB).
图 4: 第一列: 原始图像; 第二列: 嵌入 w + Space 的图像(PSNR 19—22db) ; 第三列: 嵌入 w + 和 Noise Space 的图像(PSNR 39—45db)。

perceptual loss defined in Image2StyleGAN [1]. Here, we use layers 'conv1 1', 'conv1 2', 'conv2 2' and 'conv3 3' of VGG-16 for the perceptual loss. Note that the perceptual loss is computed for four layers of the VGG network. Therefore, M_p needs to be downsampled to match the reso-lutions of the corresponding VGG-16 layers in the compu-tation of the loss function.
在 Image2StyleGAN [1]中定义的感知损失。在这里，我们使用 VGG-16 的' conv11'、' conv12'、' conv22'和' conv33'层来处理感知损失。请注意，感知损失是针对 VGG 网络的四个层次计算的。因此，在计算损耗函数时，Mp 需要被下采样以匹配相应的 VGG-16 层的分辨率。

4.2. Optimization Strategies
4.2 优化策略

Optimization of the variables $w \in W^+$ and $n \in N_s$ is not a trivial task. Since only $w \in W^+$ encodes semantically meaningful information, we need to ensure that as much information as possible is encoded in w and only high frequency details in the Noise space.

变量 w ∈ w + 和 n ∈ Ns 的优化并不是一个简单的任务。由于只有 w ∈ w + 编码语义上有意义的信息，我们需要确保尽可能多的信息被编码在 w 中，并且在噪声空间中只有高频细节。

The first possible approach is the joint optimization of both groups of variables w and n. Fig.2 (b) shows the result using the perceptual and the pixel-wise MSE loss. We can observe that many details are lost and were replaced with high frequency image artifacts. This is due to the fact that the perceptual loss is incompatible with optimizing noise

第一种可能的方法是两组变量 w 和 n 的联合优化。图 2(b)显示了使用感知和像素级 MSE 损失的结果。我们可以观察到许多细节丢失，被高频图像伪影取代。这是由于感知损失与优化噪声不相容的事实

maps. Therefore, a second approach is to use pixel-wise MSE loss only (see Fig. 2 (c)). Although the reconstruction is almost perfect, the representation (w, n) is not suitable for image editing tasks. In Fig. 2 (d), we show that too much of the image information is stored in the noise layer, by resampling the noise variables n. We would expect to obtain another very good, but slightly noisy embedding. In-stead, we obtain a very low quality embedding. Also, we show the result of jointly optimizing the variables and us-ing perceptual and pixel-wise MSE loss for w variables and pixel-wise MSE loss for the noise variable. Fig. 2 (e) shows the reconstructed image is not of high perceptual quality. The PSNR score decreases to 33.3 dB. We also tested these optimizations on other images. Based on our results, we do not recommend using joint optimization.

地图。因此，第二种方法是仅使用像素级的 MSE 损失(见图 2(c))。尽管重建几乎是完美的，但表示(w，n)不适合于图像编辑任务。在图 2(d)中，我们通过对噪声变量 n 重新采样显示了太多的图像信息存储在噪声层中。我们希望获得另一个非常好的，但稍微有点噪声的嵌入。相反，我们获得了一个非常低质量的嵌入。同时，我们还给出了变量的联合优化结果，并对 w 变量采用了感知和像素级 MSE 损失，对噪声变量采用了像素级 MSE 损失。图 2(e)显示重建图像的感知质量不高。PSNR 评分降至 33.3 dB。我们还在其他图像上测试了这些优化。基于我们的结果，我们不建议使用联合优化。

The second strategy is an alternating optimization of the variables w and n. In Fig. 3, we show the result of optimiz-ing w while keeping n fixed and subsequently optimizing n while keeping w fixed. In this way, most of the information is encoded in w which leads to a semantically meaningful embedding. Performing another iteration of optimizing w (Fig. 3 (d)) reveals a smoothing effect on the image and the PSNR reduces from 39.5 dB to 20 dB. Subsequent Noise space optimization does not improve PSNR of the images. Hence, repetitive alternating optimization does not improve the quality of the image further. In summary, we recom-mend to use alternating optimization, but each set of vari-ables is only optimized once. First we optimize w, then n.

第二种策略是变量 w 和 n 的交替优化。在图 3 中，我们展示了在保持 n 不变的情况下优化 w 的结果，以及在保持 w 不变的情况下优化 n 的结果。通过这种方式，大部分信息被编码为 w，从而导致语义上有意义的嵌入。执行另一个优化 w 的迭代(图 3(d))显示了对图像的平滑效果和 PSNR 从 39.5 分贝降低到 20 分贝。随后的噪声空间优化不能提高图像的 PSNR。因此，重复的交替优化并不能进一步提高图像的质量。总之，我们建议使用交替优化，但每组变量只优化一次。首先我们优化 w，然后 n。

Algorithm 1: Semantic and Spatial component em-bedding in StyleGAN

算法 1: StyleGAN 中的语义和空间构件嵌入

Input: images x, y $\in R^{n\times m\times 3}$; masks

输入: 图像 x，y ∈Rn ×m ×3; 面具

Ms, Mm, Mp; a pre-trained generator G(·, ·); gradient-based optimizer F′.

M，Mm，Mp; 预先训练的发生器 g (,)；基于梯度的优化器 f ′。

Output: the embedded code (w, n)

输出: 嵌入代码(w，n)

Initialize() the code (w, n) = (w′, n′);
初始化()代码(w，n) = (w ′，n ′)；

while *not converged* **do**
虽然没有收敛

Loss ← L(x, y, Ms, Mm, Mp);
Loss ← l (x，y，Ms，Mm，Mp)；

(w, n) ← (w, n) − ηF′(∇w,nL, w, n);
(w, n)←(w，n)-ηf ′(something w，nL，w，n)；

end
结束

5. Activation Tensor Manipulations
5. 激活张量操作

Due to the progressive architecture of StyleGAN, one can perform meaningful tensor operations at different lay-ers of the network [11, 4]. We consider the following edit-ing operations: spatial copying, averaging, and channel-wise copying. We define activation tensor A^l_I as the out-put of the l-th layer in the network initialized with variables (w, n) of the embedded image I. They are stored as ten-sors $A^l_I \in R^{W_L \times H_L \times C_L}$. Given two such tensors A^l_I and

由于 StyleGAN 的渐进式架构，人们可以在网络的不同层执行有意义的张量操作[11,4]。我们考虑以下编辑操作: 空间复制，平均和通道明智的复制。我们将激活张量 AIl 定义为用嵌入图像 i 的变量(w，n)初始化的网络中第 1 层的输出。它们被存储为 10-sor AIl ∈ RWL × HL × CL。给定两个这样的张量 AIl 和

8299

Figure 5: First and second column: input image; Third column: image generated by naively copying the left half from the first image and the right half from the second image; Fourth column: image generated by our extended embed-ding algorithm. The difference between the third and fourth images (second row) is highlighted in the supplementary materials.
图 5: 第一列和第二列: 输入图像; 第三列: 天真地从第一图像复制左半部分和从第二图像复制右半部分所生成的图像; 第四列: 我们的扩展嵌入丁算法生成的图像。第三和第四图像(第二行)之间的差异在补充材料中突出显示。

B_l^l, copying replaces high-dimensional pixels $\in R^{1\times1\times C_l}$ in A_l^l by copying from B_l^l. Averaging forms a linear com-bination $\lambda A_l^l + (1-\lambda)B_l^l$. Channel-wise copying creates a new tensor by copying selected channels from A_l^l and the remaining channels from B_l^l. In our tests we found that spa-tial copying works a bit better than averaging and channel-wise copying.

Bll，复制通过从 Bll 复制来替代 All 中的高维像素∈R1 × 1 × CL。平均形成线性组合 λail + (1-λ) Bll。通道明智的复制创建一个新的张量通过复制选定的通道从 All 和其余的通道从 Bll。在我们的测试中，我们发现水平复制的效果比平均复制和明智的渠道复制要好一些。

6. Frequently Used Building Blocks
6. 常用的积木

We identify four fundamental building blocks that are used in multiple applications described in Sec. 7. While terms of the loss function can be controlled by spatial masks (M_s, M_m, M_p), we also use binary masks w_m and n_m to indicate what subset of variables should be optimized during an optimization process. For example, we might set w_m to only update the w variables corresponding to the first k layers. In general, w_m and n_m contain 1s for variables that should be updated and 0s for variables that should remain constant. In addition to the listed parameters, all building blocks need initial variable values w_{ini} and n_{ini}. For all experiments, we use a 32GB Nvidia V100 GPU.

我们确定了在第 7 节描述的多个应用程序中使用的四个基本构建块。虽然损失函数的项可以通过空间掩模(Ms，Mm，Mp)来控制，但我们也使用二进制掩模 wm 和 nm 来表示在优化过程中应该优化哪些变量子集。例如，我们可以将 wm 设置为仅更新对应于第一个 k 层的 w 变量。一般来说，wm 和 nm 包含应该更新的变量的 1 和应该保持恒定的变量的 0。除了列出的参数之外，所有构建块都需要初始变量值 wini 和 nini。对于所有的实验，我们使用 32gb 的 Nvidia V100 GPU。

Masked W^+ optimization (W_l): This function optimizes $w \in W^+$, leaving n constant. We use the following parameters in the loss function (L) Eq. 1: $\lambda_s = 0$,

掩蔽 w + 优化(Wl)：该函数优化 $w \in w+$，保留 n 个常数。我们在损失函数(l) Eq 中使用以下参数。1: $\lambda s = 0$,

$\lambda_{mse1} = 10^{-5}$, $\lambda_{mse2} = 0$, $\lambda_p = 10^{-5}$. We denote the function as:
λmse1 = 10-5，λmse2 = 0，λp = 10-5。我们将函数表示为:

$$w_l(M_p, M_m, w_m, w_{ini}, n_{ini}, x) = \text{ARG MIN}\ \lambda_p L_{percept}(M_p, G(w, n), x)+$$

Wl (Mp，Mm，wm，wini，nini，x) =

ARG MIN λplcept (Mp，g (w，n)，x)

+

wM (2)

$$\text{wM} \quad \frac{\lambda mse\ 1}{N} \quad \frac{\lambda mse\ 1}{kMm \odot (G(w, n) - x)k2^2} \quad \frac{}{kMm \odot (g(w, n) - x)k2} \quad (2)$$

where wm is a mask for W + space. We either use
其中 wm 是一个面具 对于 w + 空格，我们要么使用

Adam [20] with learning rate 0.01 or gradient descent with learning rate 0.8, depending on the application. Some common settings for Adam are: $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $= 1e{-}8$. In Sec. 7, we use Adam unless specified.
Adam [20]的学习速率为 0.01，梯度下降法的学习速率为 0.8，这取决于应用。Adam 的一些常见设置是: β1 = 0.9，β2 = 0.999 和 = 1e-8。在第 7 节中，除非指定，否则我们使用 Adam。

Masked Noise Optimization (M kn): This function optimizes $n \in N_s$, leaving w constant. The Noise space N_s has dimensions $R^{4\times4}, \ldots, R^{1024\times1024}$. In total there are 18 noise maps, two for each resolution. We set follow-ing parameters in the loss function (L) Eq. 1: $\lambda_s = 0$,

掩蔽噪声优化(mkn)：该函数优化 n ∈Ns，保留 w 常数。噪声空间 Ns 的尺寸为 R4 ×4，...，R1024 ×1024。总共有 18 个噪声图，每个分辨率两个。我们在损失函数(l) Eq 中设置后续参数。1: λs = 0,

$\lambda_{mse1} = 10{-}5$, $\lambda_{mse2} = 10{-}5$, $\lambda_p = 0$. We denote the function as:

$\lambda_{mse1} = 10\text{-}5$, $\lambda_{mse2} = 10\text{-}5$, $\lambda_p = 0$

函数如下:

$$M_{kn}(M, w_{ini}, n_{ini}, x, y) = M_{kn}(m, w_{ini}, n_{ini}, x, y) =$$

$$\begin{matrix} \lambda_{mse} \\ 2 \end{matrix} \quad \begin{matrix} \lambda_{mse} \\ 2 \end{matrix} \quad \begin{matrix} \text{ARG} \\ \text{MIN} \end{matrix} \quad \frac{}{N} \quad \|M \odot (G(w, n) - x)\|_2^2 \quad \begin{matrix} \text{ARG} \\ \text{MIN} \end{matrix} \quad \|M \odot (g(w, n) - x)\|_2^2 + \tag{3}$$

$$\begin{matrix} \lambda_{mse} \\ 1 \end{matrix} \quad \begin{matrix} \lambda_{mse} \\ 1 \end{matrix} \quad \begin{matrix} \text{ARG} \\ \text{MIN} \\ n \end{matrix} \quad \frac{}{N} \quad \|(1 - M) \odot (G(w, n) - y)\|_2^2 \quad \|(1-M) \odot (g(w, n) - y)\|_2^2$$

For this optimization, we use Adam with learning rate 5, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varrho = 1e^{-8}$. Note that the learning rate is very high.
对于这个优化问题，我们使用具有学习速率 5，β1 = 0.9，β2 = 0.999，something = 1e-8 的 Adam。请注意，学习率非常高。

Masked Style Transfer(M_{st}): This function optimizes w to achieve a given target style defined by style image y. We set following parameters in the loss function (L) Eq. 1: $\lambda_s = 5 \times 10^{-7}$, $\lambda_{mse_1} = 0$, $\lambda_{mse_2} = 0$, $\lambda_p = 0$. We denote the function as:

掩码样式转移(Mst)：此函数优化 w 以实现由样式图像 y 定义的给定目标样式。我们在损失函数(l) Eq 中设置以下参数。1: λs = 5 × 10-7，λmse1 = 0，λmse2 = 0，λp = 0.我们将函数表示为:

$$M_{st}(M_s, w_{ini}, n_{ini}, y) =$$

$$\textbf{Mst (Ms，wini，nini，y) =}$$

$$\text{ARG MIN } \lambda_s L_{style}(M_s, G(w, n), y) \tag{4}$$

$$(m_w, g(w, n), y) \tag{4}$$

where w is the whole W^+ space. For this optimization, we use Adam with learning rate 0.01, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varrho = 1e^{-8}$.
其中 w 是整个 w + 空间。对于这个优化，我们使用 Adam，其学习速率为 0.01，β1 = 0.9，β2 = 0.999，something = 1e-8。

Masked activation tensor operation (l_{att}): This function describes an activation tensor operation. Here, we rep-resent the generator G(w, n, t) as a function of W^+ space variable w, Noise space variable n, and input tensor t. The operation is represented by:

掩蔽激活张量运算(Iatt)：这个函数描述了激活张量运算。本文将生成器 g (w，n，t)表示为 w + 空间变量 w，噪声空间变量 n 和输入张量 t 的函数。操作由以下代表:

$$l_{att}(M_1, M_2, w, n_{ini}, l) =$$

$$\textbf{Iatt (M1，M2，w，nini，l) =}$$

$$G(w, n, M_1 \odot (A^{l}{}_{l_1}) + (1 - M_2) \odot (B^{l}{}_{l_2})) \tag{5}$$

$$G(w, n, M1 \odot (Al_{l1}) + (1\text{-}M2) \odot (Bl_{l2})) \tag{5}$$

where $A^{l}{}_{l_1}$ and $B^{l}{}_{l_2}$ are the activations corresponding to images l_1 and l_2 at layer l, and M_1 and M_2 are the masks downsampled using nearest neighbour interpolation to match the $H_l \times W_l$ resolution of the activation tensors.
其中 Al l1 和 Bl l2 是对应于图像 I1 和 I2 在第 1 层的激活，M1 和 M2 是采用最近邻插值匹配激活张量的 Hl × Wl 分辨率下采样的掩模。

7. Applications
7. 应用程序

In the following we describe various applications enabled by our framework.
下面我们将描述我们的框架所支持的各种应用程序。

Algorithm 2: Improved Image Reconstruction
算法 2: 改进的图像重建

Input: image $I_m \in R^{n \times m \times 3}$
输入: 图像 Im ∈Rn ×m ×3
Output: the embedded code (w_{out}, n_{out})
输出: 嵌入式代码(wout，nout)

1 (w_{ini}, n_{ini}) ← initialize();
(wini，nini)← initialize () ;

2 $w_{out} = W_l(1, 1, 1, w_{ini}, n_{ini}, I_m)$;
Wout = Wl (1,1,1，wini，nini，Im) ;

3 $n_{out} = M\ k_n(1, w_{out}, n_{ini}, I_m, 0)$;
Nout = m kn (1，wout，nini，Im，0) ;

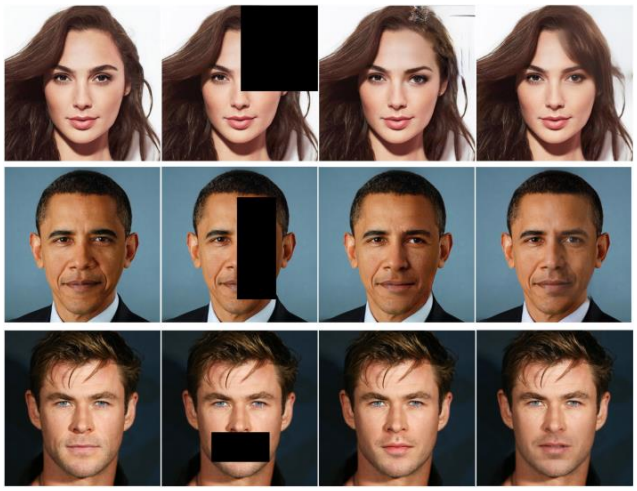

Figure 6: First column: original image; Second column:
图 6: 第一列: 原始图像; 第二列:

defective image ; Third column: inpainted image via partial convolutions [24]; Fourth column: inpainted image using our method.
有缺陷的图像; 第三列: 通过部分卷积绘制的图像[24] ; 第四列: 使用我们的方法绘制的图像。

7.1. Improved Image Reconstruction
7.1 改进图像重建

As shown in Fig. 4, any image can be embedded by optimizing for variables $w \in W^+$ and $n \in N_s$. Here we describe the details of this embedding (See Alg. 2). First, we initialize: w_{ini} is a mean face latent code [19] or random code sampled from $U[-1, 1]$ depending on whether the embedding image is a face or a non-face, and n_{ini} is sampled from a standard normal distribution $N(0, I)$ [19]. Second, we apply masked W^+ optimization (W_l) without using spa-tial masks or masking variables. That means all masks are set to 1. I_m is the target image we try to reconstruct. Third, we perform masked noise optimization ($M\ k_n$), again with-out making use of masks. The images reconstructed are of high fidelity. The PNSR score range of 39 to 45 dB provides an insight of how expressive the Noise space in StyleGAN is. Unlike the W^+ space, the Noise space is used for spa-tial reconstruction of high frequency features. We use 5000 iterations of W_l and 3000 iterations of $M\ k_n$ to get PSNR scores of 44 to 45 dB. Additional iterations did not improve the results in our tests.

如图 4 所示，任何图像都可以通过对变量 w ∈w + 和 n ∈Ns 进行优化来嵌入。在这里，我们描述了这种嵌入的细节(参见 Alg。2).首先，我们初始化: wini 是一个从 u [-1,1]中取样的平均面潜在码[19]或随机码，取决于电子层析图像是一个面还是一个非面，并且 nini 是从标准正态分布 n (0，i)[19]中取样的。其次，我们应用掩蔽的 w + 优化(Wl)，而不使用 spa 掩蔽或掩蔽变量。这意味着所有的面具都设置为 1。Im 是我们试图重建的目标图像。第三，我们执行掩蔽噪声优化(m kn)，同样没有使用掩蔽。重建的图像保真度很高。39 至 45 分贝的 PNSR 得分范围提供了 StyleGAN 中的噪声空间如何表现的洞察力。与 w + 空间不同，噪声空间用于高频特征的空间重建。我们使用 5000 次 Wl 迭代和 3000 次 m kn 迭代来获得 44 至 45 分贝的 PSNR 分数。额外的迭代没有改善我们的测试结果。

Figure 7: Inpainting using different initializations w_{ini}.
图 7: 使用不同的初始化 wini 进行 Inpainting。

Algorithm 3: Image Crossover
算法 3: 图像交叉

Input: images $I_1, I_2 \in R^{n \times m \times 3}$; mask M_{blur}
输入: 图像 I1，I2∈Rn ×m ×3; 掩模 Mblur
Output: the embedded code (w_{out}, n_{out})
输出: 嵌入式代码(wout，nout)

$(w^*, n_{ini}) \leftarrow$ initialize();
(w * , nini)← initialize () ;
$w_{out} = W_I(M_{blur}, M_{blur}, 1, w^*, n_{ini}, I_1) + W_I(1 - M_{blur}, 1 - M_{blur}, 1, w*, nini, I2)$;
Wout = WI (Mblur, Mblur, 1, w * , nini, I1) + WI (1-Mblur, 1-Mblur, 1, w * , nini, I2) ;
$n_{out} = M\ k_n(M_{blur}, w_{out}, n_{ini}, I_1, I_2)$;
Nout = m kn (Mblur, wout, nini, I1, I2) ;

7.2. Image Crossover
7.2 图像交叉

We define the image crossover operation as copying parts from a source image y into a target image x and blending the boundaries. As initialization, we embed the target image x to obtain the W^+ code w^*. We then perform masked W^+ optimization (W_I) with blurred masks M_{blur} to embed the regions in x and y that contribute to the final image. Blurred masks are obtained by convolution of the binary mask with a Gaussian filter of suitable size. Then, we perform noise optimization. Details are provided in Alg. 3.

我们将图像交叉操作定义为从源图像 y 复制部分到目标图像 x 并混合边界。作为初始化，我们嵌入目标图像 x 以获得 w + 代码 w * 。然后我们使用模糊掩码 Mblur 进行掩码 w + 优化(Wl) ，以便在 x 和 y 中嵌入有助于最终图像的区域。模糊掩模是通过二进制掩模与适当大小的高斯滤波器的卷积获得的。然后，我们执行噪声优化。细节在 Alg 中提供。3.

Other notations are the same as described in Sec 7.1. Fig. 5 and Fig. 1 show example results. We deduce that the reconstruction quality of the images is quite high. For the experiments, we use 1000 iterations in the function masked

其他符号与 Sec 7.1 中描述的相同。图 5 和图 1 显示了示例结果。我们推断图像的重建质量相当高。在实验中，我们使用了 1000 次迭代的函数屏蔽

W^+ optimization and 1000 iterations in M k_n.
+ 优化和 1000 次迭代(m kn)。

7.3. Image Inpainting
7.3 图像绘制

Algorithm 4: Image Inpainting
算法 4: 图像修复

Input: image $I_{def} \in R^{n \times m \times 3}$; masks M, M_{blur}+
输入: 图像 Idef ∈Rn ×m ×3; 蒙版 m，Mblur +
Output: the embedded code (w_{out}, n_{out})
输出: 嵌入式代码(wout，nout)

$(w_{ini}, n_{ini}) \leftarrow$ initialize();
(wini, nini)← initialize () ;
$w_{out} = W_I(1 - M, 1 - M, w_m, w_{ini}, n_{ini}, I_{def})$;
Wout = WI (1-m, 1-m, wm, wini, nini, Idef) ;
n_{out} =
Nout =
$M\ k_n(1 - M_{blur}+, w_{out}, n_{ini}, I_{def}, G(w_{out}))$;
Kn (1-Mblur + , wout, nini, Idef, g (wout)) ;

In order to perform a semantically meaningful inpainting, we embed into the early layers of the W^+ space to predict the missing content and in the later layers to main-

为了实现一个有语义意义的绘画，我们嵌入到 w + 空间的早期层来预测丢失的内容，并在后期层来实现主-

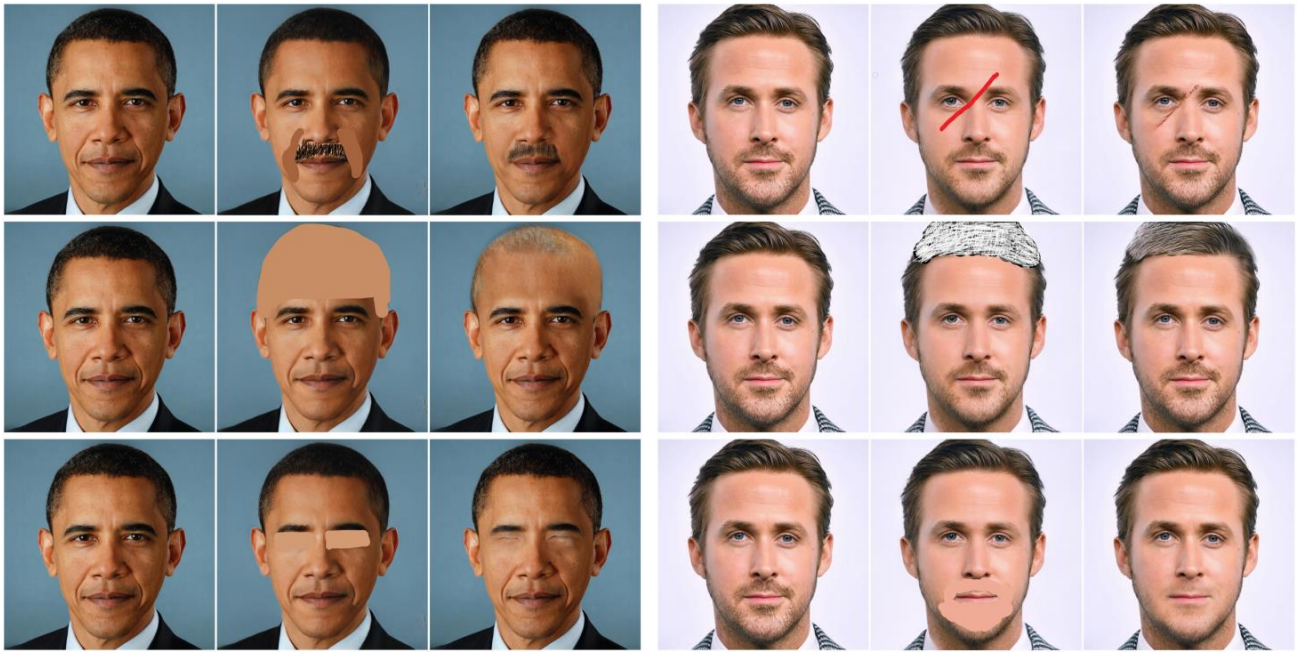

Figure 8: Column 1 & 4: base image; Column 2 & 5: scribbled image ; Column 3 & 6: result of local edits.
图 8: 第一栏和第四栏: 基础图像; 第二栏和第五栏: 潦草的图像; 第三栏和第六栏: 本地编辑的结果。

tain color consistency. We define the image x as a defec-tive image (I_{def}). Also, we use the mask w_m where the value is 1 corresponding to the first 9 (1 to 9), 17^{th} and 18^{th} layer of W^+. As an initialization, we set w_{ini} to the mean face latent code [19]. We consider M as the mask describing the defective region. Using these parameters, we perform the masked W^+ optimization W_l. Then we per-form the masked noise optimization $M\ k_n$ using M_{blur+} which is the slightly larger blurred mask used for blend-ing. Here λ_{mse2} is taken to be 10^{-4}. Other notations are the same as described in Sec 7.1. Alg. 4 shows the details of the algorithm. We perform 200 steps of gradient descent optimizer for masked W^+ optimization W_l and 1000 itera-tions of masked noise optimization $M\ k_n$. Fig.6 shows ex-ample inpainting results. The results are comparable with the current state of the art, partial convolution [24]. The partial convolution method frequently suffers from regular artifacts (see Fig.6 (third column)). These artifacts are not present in our method. In Fig.7 we show different inpaint-ing solutions for the same image achieved by using differ-ent initializations of w_{ini} , which is an offset to mean face latent code sampled independently from a uniform distribu-tion $U\ [-0.4,\ 0.4]$. The initialization mainly affects layers 10 to 16 that are not altered during optimization. Multi-ple inpainting solutions cannot be computed with existing state-of-the-art methods.
颜色的一致性。我们将图像 x 定义为缺陷图像(Idef)。此外，我们使用的掩模 wm 的值是 1 对应的第一个 9(1 至 9)，第 17 和第 18 层的 w + 。作为初始化，我们将 wini 设置为平均面部潜在代码[19]。我们认为 m 是描述缺陷区域的掩码。使用这些参数，我们执行掩蔽的 w + 优化 Wl。然后我们使用 Mblur + 进行掩蔽噪声优化 m kn，Mblur + 是用于混合的稍大的模糊掩蔽。这里把 λmse2 取为 10-4。其他符号与 Sec 7.1 中描述的相同。海藻酸钠。

4 显示了算法的细节。我们对掩蔽 w + 优化 Wl 进行了 200 次梯度下降法优化，对掩蔽噪声优化 m kn 进行了 1000 次迭代。图 6 显示了绘制结果的示例。结果与当前的艺术水平相当，部分卷积[24]。部分卷积方法经常受到规则伪影的影响(见图 6(第三列))。这些工件在我们的方法中不存在。在图 7 中，我们展示了通过使用 wini 的不同初始化实现的同一图像的不同内绘解决方案，这是对独立于均匀分布 u [-0.4,0.4]采样的平均面部潜在代码的偏移量。初始化主要影响在优化过程中没有改变的 10 至 16 层。现有的最先进的方法不能计算多个修补解决方案。

7.4. Local Edits using Scribbles
7.4 使用涂鸦进行本地编辑

Another application is performing semantic local edits guided by user scribbles. We show that simple scribbles can
另一个应用程序是在用户涂鸦的指导下执行语义本地编辑

Figure 9: First column: base image; Second column: mask area; Third column: style image; Fourth column: local style transfer result.

图 9: 第一列: 基础图像; 第二列: 掩码区; 第三列: 样式图像; 第四列: 本地样式转换结果。

Algorithm 5: Local Edits using Scribble

算法 5: 使用 Scribble 进行本地编辑

Input: image $I_{scr} \in R^{n \times m \times 3}$; masks M_{blur}

输入: 图像 Iscr ∈Rn ×m ×3; 掩码 Mblur

Output: the embedded code (w_{out}, n_{out})

输出: 嵌入式代码(wout，nout)

$(w^*, n_{ini}) \leftarrow$ initialize();

(w * ， nini)← initialize () ;

$w_{out} = W_I(1, 1, w_m, w^*, n_{ini}, I_{scr})$ $+\lambda k w^* - w_{out} k_2$; nout = M kn(Mblur, wout, nini, Iscr , G(wout));

Wout = Wl (1,1， wm， w * * ， nini， Iscr) + λkw * *-woutk2; nout = mkn (Mblur, wout， nini， Iscr， g (wout)) ;

be converted to photo-realistic edits by embedding into the first 4 to 6 layers of W^+ (See Fig.8). This enables us to do local edits without training a network. We define an image x as a scribble image (I_{scr}). Here, we also use the mask w_m where the value is 1 corresponding to the first 4,5 or 6 layers of the W^+ space. As initialization, we set the w_{ini} to w^*

通过嵌入到 w + 的前 4 到 6 层(见图 8)转换为照片真实感编辑。这使我们能够在没有训练网络的情况下进行本地编辑。我们将一个图像 x 定义为一个潦草的图像(Iscr)。在这里，我们还使用了掩模 wm，其值为 1，对应于 w + 空间的前 4、5 或 6 层。作为初始化，我们将 wini 设置为 w *

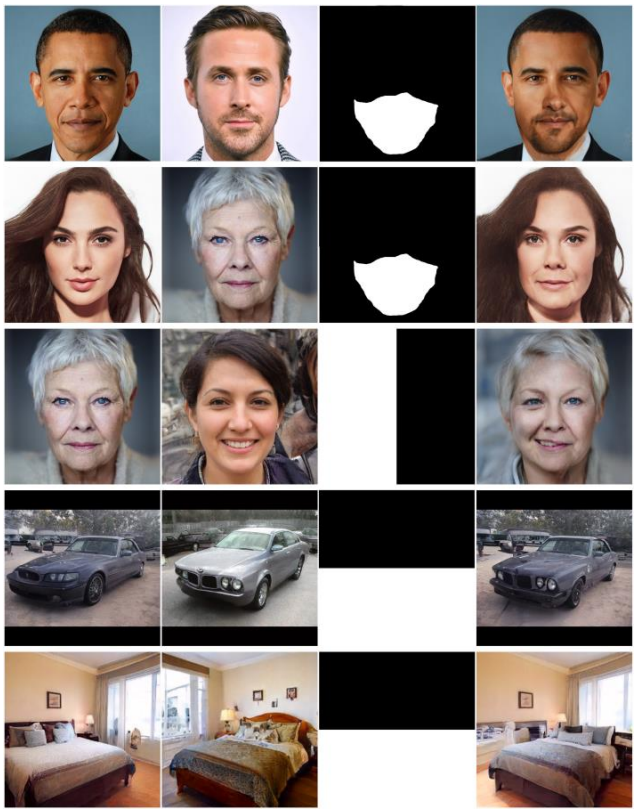

Figure 10: First column: base image; Second column: attribute image; Third column: mask area; Fourth column: image generated via attribute level feature transfer.
图 10: 第一列: 基础图像; 第二列: 在 - 贡品图像; 第三列: 面具区; 第四列: 通过属性级特征转移生成的图像。

Algorithm 6: Local Style Transfer
算法 6: Local Style Transfer

Input: images $I_1, I_2 \in R^{n \times m \times 3}$; masks M_{blur}
输入: 图像 I1，I2∈Rn × m × 3; 掩模 Mblur
Output: the embedded code (w_{out}, n_{out})
输出: 嵌入式代码(wout，nout)

1 (w^*, n_{ini}) ← initialize();
(w * , nini)← initialize () ;
2 $w_{out} = W_I(M_{blur}, M_{blur}, 1, w^*, n_{ini}, I_1)$
$+M_{st}(1 - M_{blur}, w^*, n_{ini}, I_2)$; nout = M
kn(Mblur, wout, nini, I1, G(wout));
Wout = WI (Mblur, Mblur, 1, w * , nini, I1)
+ Mst (1-Mblur, w * , nini, I2) ; nout =
mknn (Mblur, wout, nini, I1, g (wout)) ;

which is the W^+ code of the image without scribble. We perform masked W^+ optimization using these parameters. Then we perform masked noise optimization $M\ k_n$ using M_{blur}. Other notations are the same as described in Sec 7.1. Alg. 5 shows the details of the algorithm. We perform 1000 iterations using Adam with a learning rate of 0.1 of masked
这是没有涂鸦的图像的 w + 代码。我们使用这些参数进行掩蔽 w + 优化。然后我们使用 Mblur 进行掩蔽噪声优化 m kn。其他符号与 Sec 7.1 中描述的相同。海藻酸钠。5 显示了算法的细节。我们使用 Adam 进行了 1000 次迭代，学习率为 0.1
W^+ optimization W_I and then 1000 steps of masked noise optimization $M\ k_n$ to output the final image.
+优化 WI，然后 1000 步掩蔽噪声优化 m kn 输出最终的图像。

7.5. Local Style Transfer
7.5 局部风格转换

Local style transfer[13] modifies a region in the input image x to transform it to the style defined by a style reference image. First, we embed the image in W^+ space to
Local style transfer [13]修改输入图像 x 中的一个区域，将其转换为由样式引用图像定义的样式。首先，我们将图像嵌入到 w + 空间中

obtain the code w^*. Then we apply the masked W^+ optimization W_I along with masked style transfer M_{st} using blurred mask M_{blur}. Finally, we perform the masked noise optimization $M\ k_n$ to output the final image. Alg. 6 shows the details of the algorithm. Results for the application are shown in Fig.9. We perform 1000 steps to obtain of W_I along with M_{st} and then perform 1000 iterations of $M\ k_n$.
获得代码 w * 。然后我们应用蒙版 w + 运算优化 Wl 和蒙版风格转换 Mst 使用模糊蒙版 Mblur。最后，我们执行蒙版噪声优化 m kn 来输出最终的图像。海藻酸钠。6 显示了算法的细节。应用的结果如图 9 所示。我们执行 1000 个步骤以获得 Wl 以及 Mst，然后执行 1000 次 m kn 的迭代。

7.6. Attribute level feature transfer
7.6 属性级特征转移

We extend our work to another application using ten-sor operations on the images embedded in W^+ space. In this application we perform the tensor manipulation corresponding to the tensors at the output of the 4^{th} layer of StyleGAN. We feed the generator with the latent codes (w, n) of two images I_1 and I_2 and store the output of the fourth layer as intermediate activation tensors A^{I_1} and B^{I_2}. A mask M_s specifies which values to copy from A^{I_1} and which to copy from B^{I_2}. The operation can be denoted by $I_{att}(M_s, M_s, w, n_{ini}, 4)$. In Fig.10, we show results of the operation. A design parameter of this application is what style code to use for the remaining layers. In the shown example, the first image is chosen to provide the style. No-tice, in column 2 of Fig.10, in-spite of the different align-ment of the two faces and objects, the images are blended well. We also show results of blending for the LSUN-car and LSUN-bedroom datasets. Hence, unlike global edits like image morphing, style transfer, and expression transfer [1], here different parts of the image can be edited inde-pendently and the edits are localized. Moreover, along with other edits, we show a video in the supplementary material that further shows that other semantic edits e.g. masked im-age morphing can be performed on such images by linear interpolation of W^+ code of one image at a time.

我们将我们的工作扩展到另一个应用程序，使用 10-sor 操作嵌入在 w + 空间中的图像。在这个应用程序中，我们执行张量操作，对应于第四层 StyleGAN 的输出张量。我们用两幅图像 I1 和 I2 的潜码(w，n)给发生器输入，并将第四层的输出存储为中间激活张量 AIl1 和 BlI2。掩码 Ms 指定从 AIl1 复制哪些值，从 BlI2 复制哪些值。操作可以用 Iatt (Ms，Ms，w，nini，4)表示。在图 10 中，我们显示了操作的结果。这个应用程序的一个设计参数是其余层使用什么样式代码。在上面的例子中，第一张图片被选中来提供样式。没有注意，在图 10 的第 2 列中，尽管两个面和物体的对齐方式不同，但图像混合得很好。我们还显示了 LSUN-car 和 LSUN-bedroom 数据集的混合结果。因此，与图像变形、样式转换和表达式转换等全局编辑不同，这里图像的不同部分可以独立编辑，并且编辑是本地化的。此外，与其他编辑一起，我们在补充材料中显示了一个视频，进一步表明，其他语义编辑，如蒙面图像变形，可以执行这样的图像通过线性插值的 w + 码的一个图像的时间。

8. Conclusion
8. 结论

We proposed Image2StyleGAN++, a powerful image editing framework built on the recent Image2StyleGAN. Our framework is motivated by three key insights: first, high frequency image features are captured by the addi-tive noise maps used in StyleGAN, which helps to im-prove the quality of reconstructed images; second, local edits are enabled by including masks in the embedding al-gorithm, which greatly increases the capability of the pro-posed framework; third, a variety of applications can be cre-ated by combining embedding with activation tensor ma-nipulation. From the high quality results presented in this paper, it can be concluded that our Image2StyleGAN++ is a promising framework for general image editing. For fu-ture work, in addition to static images, we aim to extend our framework to process and edit videos.

我们提出了 Image2StyleGAN + + ，一个基于最近的 Image2StyleGAN 的强大的图像编辑框架。我们的框架有三个关键的见解: 第一，StyleGAN 中使用的加性噪声映射能够捕获高频图像特征，有助于提高重建图像的质量; 第二，在嵌入算法中加入掩码可以实现局部编辑，大大提高了框架的性能; 第三，将嵌入与激活张量操作相结合，可以创建多种应用。从本文提出的高质量的结果，可以得出结论，我们的 Image2StyleGAN + + 是一个有前途的一般图像编辑框架。对于未来的工作，除了静态图像，我们的目标是扩展我们的框架，以处理和编辑视频。

Acknowledgement This work was supported by the KAUST Office of Sponsored Research (OSR) under Award No. OSR-CRG2018-3730.

这项工作得到了 KAUST 赞助研究办公室(OSR)的支持，奖励编号为 OSR-crg2018-3730。

References
参考文献

[1] R. Abdal, Y. Qin, and P. Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4432–4441, 2019. 1, 2, 4, 8

Abdal，y。 Qin，and p. Wonka。图片 2 stylegan: 如何将图片嵌入 stylegan 的潜在空间？在 IEEE 国际计算机视觉会议记录中，页 4432-4441,2019。1,2,4,8

[2] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gener-ative adversarial networks. In *Proceedings of the 34th In-ternational Conference on Machine Learning*, volume 70, pages 214–223, 2017. 2

阿约夫斯基先生、尚塔拉先生和 l. 波图先生。沃瑟斯坦生成式对抗网络。第 34 届机器学习国际会议记录，第 70 卷，2017 年 214-223 页。2

[3] D. Bau, H. Strobelt, W. Peebles, J. Wulff, B. Zhou, J. Zhu, and A. Torralba. Semantic photo manipulation with a gener-ative image prior. *ACM Transactions on Graphics (Proceed-ings of ACM SIGGRAPH)*, 38(4), 2019. 2

Bau，h。 Strobelt，w。 Peebles，j。 Wulff，b。 Zhou，j。 Zhu，and a。语义照片处理与生成图像优先。ACM 图形交易(Proceed-ings of ACM SIGGRAPH)，38(4)，2019。2

[4] D. Bau, J.-Y. Zhu, H. Strobelt, B. Zhou, J. B. Tenenbaum,

鲍，朱，H.Strobelt，B.Zhou，J.B.Tenenbaum,

W.T. Freeman, and A. Torralba. Gan dissection: Visualizing and understanding generative adversarial networks. In *Pro-ceedings of the International Conference on Learning Rep-resentations (ICLR)*, 2019. 2, 4

t. Freeman 和 a. Torralba。Gan 解剖: 可视化和理解生成性对抗网络。在 2019 年学习代表国际会议(ICLR)的会议记录中。2,4

[5] D. Bau, J.-Y. Zhu, J. Wulff, W. Peebles, H. Strobelt, B. Zhou, and A. Torralba. Seeing what a gan cannot generate. In *Pro-ceedings of the International Conference Computer Vision (ICCV)*, 2019. 1, 2

D. Bau，j-y.朱杰 ·伍尔夫、 w ·皮布尔斯、 h ·斯特罗贝特、b ·周和 a ·托拉尔巴。看看一个干不能产生什么。In pro-proceedings of the International Conference Computer Vision (ICCV)， 20192019 年国际会议计算机视觉(ICCV)。1,2

[6] A. Brock, J. Donahue, and K. Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *Inter-national Conference on Learning Representations*, 2019. 1, 2

布洛克，j。多纳休和 k。西蒙扬。用于高保真自然图像合成的大规模 GAN 培训。在 2019 年国际学习代表大会上。1,2

[7] A. Creswell and A. A. Bharath. Inverting the generator of a generative adversarial network. *IEEE Transactions on Neu-ral Networks and Learning Systems*, 2018. 2

克雷斯韦尔和巴拉斯。颠倒生成性对抗网络的生成器。IEEE 网络和学习系统交易，2018。2

[8] U. Demir and G. Unal. Patch-based image inpaint-ing with generative adversarial networks. *arXiv preprint arXiv:1803.07422*, 2018. 1

U. Demir 和 g. Unal。基于 patch 的图像用生成性对抗网络进行绘制。 arXiv 预印 arXiv: 1803.07422,2018.1

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 2

邓伟东，r. Socher，l.j。李启雄及李飞飞。ImageNet: 一个大规模的分层图像数据库。在 CVPR09,2009。2

[10] A. Dosovitskiy and T. Brox. Generating images with percep-tual similarity metrics based on deep networks. In *Advances in neural information processing systems*, pages 658–666, 2016. 3

A. Dosovitskiy 和 t. Brox。基于深层网络的感知相似性度量生成图像。在神经信息处理系统的进展，页 658-666,2016。3

[11] A. Fruhst ¨uck, ¨I. Alhashim, and P. Wonka. Tilegan. *ACM Transactions on Graphics*, 38(4):1–11, Jul 2019. 2, 4

阿尔哈希姆和王卡。蒂勒根。ACM 图形交易，38(4) : 1-11,2019 年 7 月。2,4

[12] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style trans-fer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recog-nition*, pages 2414–2423, 2016. 3

盖蒂斯，a ·s ·埃克尔和 m ·贝格。使用卷积神经网络的图像样式转换。在 IEEE 关于计算机视觉和模式识别会议的会议记录中，页 2414-2423,2016。3

[13] L. A. Gatys, A. S. Ecker, M. Bethge, A. Hertzmann, and

洛杉矶盖蒂斯，A.s. 埃克尔，m. 贝格，A.Hertzmann，和

E. Shechtman. Controlling perceptual factors in neural style transfer. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017. 8

谢特曼先生。控制神经风格转换中的感知因素。2017 IEEE 计算机视觉和模式识别会议(CVPR)，2017 年 7 月。8

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu,

I. 好伙计，j. Pouget-Abadie，m. Mirza，b. Xu,

D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Gen-erative adversarial nets. In *Advances in neural information processing systems*, 2014. 2

《神经信息处理系统的进展》，2014.2

[15] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and

I. Gulrajani，f. Ahmed，m. Arjovsky，v. Dumoulin，and

A. C. Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777, 2017. 2

C. Courville。改进 wasserstein gans 的训练。在神经信息处理系统的进步，页 5767-5777,2017.2

[16] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017. 2

朱仲一，周恩来，等。条件对抗网络图像到图像的翻译。CVPR，2017.2

[17] Y. Jo and J. Park. Sc-fegan: Face editing generative adversarial network with user's sketch and color. In *The IEEE International Conference on Computer Vision (ICCV)*, Octo-ber 2019. 1

约翰 ·乔和 j ·朴。Sc-fegan: 使用用户的草图和颜色进行面部编辑。在 IEEE 国际计算机视觉会议(ICCV)，2019 年 10 月。1

[18] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, 2016. 3

J ·约翰逊，a ·阿拉希和 l ·菲菲。实时风格转换和超分辨率的感知损失。在 2016 年欧洲计算机视觉会议上。3

[19] T. Karras, S. Laine, and T. Aila. A style-based genera-tor architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948*, 2018. 1, 2, 6, 7

卡拉斯，莱恩和艾拉。基于风格的生成对抗网络的生成器架构。arXiv 预印 arXiv: 1812.04948,2018。1,2,6,7

[20] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. 2014. 5

金马和亚当: 随机优化的一种方法。2014.5

[21] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2

D. p. Kingma 和 m. Welling。自动编码变量贝叶斯。 arXiv 预印 arXiv: 1312.6114,2013.2

[22] C. Li and M. Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *Com-puter Vision - ECCV 2016 - 14th European Conference, Am-sterdam, The Netherlands, October 11-14, 2016, Proceed-ings, Part III*, 2016. 2

C. Li 和 m 魔杖。用马尔可夫生成对抗网络预计算实时纹理合成。2016 年 10 月 11 日至 14 日在荷兰阿姆斯特丹召开的第 14 届欧洲会议上发表的《计算机视觉》，2016 年第三部分。2

[23] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan. Perceptual generative adversarial networks for small object detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2

李少光、梁耀威、许霆、冯国纶和 s. Yan。用于小目标检测的感知生成对抗网络。在 IEEE 计算机视觉和模式识别会议(CVPR)，2017 年 7 月。2

[24] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro. Image inpainting for irregular holes using partial convolutions. *Lecture Notes in Computer Science*, page 89–105, 2018. 6, 7

刘嘉华、李达、史国强、张国柱。Wang，a. Tao，and b. Catanzaro.利用部分卷积对不规则孔进行图像修复。计算机科学讲义，第 89-105 页，2018 年。6,7

[25] M.-Y. Liu, X. Huang, A. Mallya, T. Karras, T. Aila, J. Lehti-nen, and J. Kautz. Few-shot unsueprvised image-to-image translation. In *arxiv*, 2019. 2

M.y.刘，黄，a。马利亚，t。卡拉斯，t。艾拉，j。几张照片，不受监督的图像到图像的翻译。在 arxiv，2019。2

[26] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

毛俊辉、李国栋、谢国栋、刘永贤、王梓和斯摩利警司。最小二乘生成对抗网络。2017 年 IEEE 计算机视觉国际会议(ICCV)，2017 年 10 月。2

[27] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *Inter-national Conference on Learning Representations*, 2018. 2

T ·宫东，t ·片冈，m ·小山，y ·吉田。生成性对抗网络的谱归一化。在 2018 年国际学习代表大会上。二

[28] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu. Semantic im-age synthesis with spatially-adaptive normalization. In *Pro-ceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

T.Park m.y.刘，T.-C。王老师，还有周杰伦。朱先生。基于空间自适应归一化的语义图像合成。在 2019 年 IEEE 计算机视觉和模式识别会议的议事录中。2

[29] A. Radford, L. Metz, and S. Chintala. Unsupervised repre-sentation learning with deep convolutional generative adver-sarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 2

A。 Radford，l。 Metz，and s。无监督代表表象学习与深层卷积生成对抗网络。arXiv 预印 arXiv: 1511.06434,2015。2

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. 2014. 3

《用于大规模图像识别的非常深的卷积网络》，2014.3

[31] R. Slossberg, G. Shamai, and R. Kimmel. High quality facial surface and texture synthesis via generative adversarial net-works. In *European Conference on Computer Vision*, pages 498–513. Springer, 2018. 2

R ·斯洛斯伯格，g ·沙迈尔，r ·基梅尔。高质量的面部表面和纹理合成通过生成对手网络。在欧洲计算机视觉会议，页 498-513。Springer，2018.2

[32] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz. Moco-gan: Decomposing motion and content for video generation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

S. Tulyakov m.y.刘，杨，和考茨。Moco-gan: 为视频生成分解动作和内容。在 IEEE 计算机视觉和模式识别会议(CVPR)，2018 年 6 月。2

8304

[33] C. Vondrick, H. Pirsiavash, and A. Torralba. Generating videos with scene dynamics. In *Advances in Neural Information Processing Systems 29*. 2016. 2
C. Vondrick h. Pirsiavash 和 a. Torralba。用场景动态生成视频。神经信息处理系统的进展。二○一六年。2

[34] T.-C. Wang, M.-Y. Liu, A. Tao, G. Liu, J. Kautz, and T.c.Wang，m.y.刘，a。陶，g。刘，j
B. Catanzaro. Few-shot video-to-video synthesis. *arXiv preprint arXiv:1910.12713*, 2019. 1, 2
卡坦萨罗。少镜头视频到视频合成。 arXiv 预打印 arXiv: 1910.12713,2019.1,2

[35] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro. Video-to-video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 1, 2
T.-C.Wang，m.y.刘强东。朱、刘、陶、考茨、卡坦萨罗。视频到视频合成。神经信息处理系统(NeurIPS)的进展，2018。1,2

[36] R. Webster, J. Rabin, L. Simon, and F. Jurie. Detecting over-fitting of deep generative networks via latent recovery. 2019. 1
通过潜在恢复检测深层生成网络的过度拟合，2019.1

[37] W. Xian, P. Sangkloy, V. Agrawal, A. Raj, J. Lu, C. Fang, 西安，p。桑克洛伊，v。阿格拉瓦尔，a,
F. Yu, and J. Hays. Texturegan: Controlling deep image synthesis with texture patches. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2
Yu 和 j. Hays。Texturegan: 用纹理贴片控制深度图像合成。在 IEEE 计算机视觉和模式识别会议(CVPR)，2018 年 6 月。2

[38] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang. Free-form image inpainting with gated convolution. 2018. 1
余、林、杨、沈、陆、黄。门控卷积自由形式图像修复。2018 年。1

[39] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Gen-erative image inpainting with contextual attention. In *Pro-ceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5505–5514, 2018. 1
余、林、杨、沈、陆、黄。基于上下文关注的一代意象绘画。在 IEEE 计算机视觉和模式识别会议的议事录中，页 5505-5514,2018。1

[40] J.-Y. Zhu, P. Krahenb ¨uhl, ¨E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2016. 1
J-y.朱，p ·克拉亨布，e ·谢克曼，a ·a ·埃弗罗斯。自然图像手工折叠的生成性视觉操作。在《欧洲计算机视觉会议进程》(ECCV)，2016 年。1

[41] J.-Y. Zhu, P. Krahenb ¨uhl, ¨E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. *Lecture Notes in Computer Science*, page 597–613, 2016. 2
J-y.朱，p ·克拉亨布，e ·谢克曼，a ·a ·埃弗罗斯。自然图像的生成性视觉操作。计算机科学讲义，页 597-613,2016。2

[42] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial net-workss. In *Computer Vision (ICCV), 2017 IEEE Interna-tional Conference on*, 2017. 2

J-y.朱，t。朴，p。伊索拉，a。埃弗罗斯。不成对的图像到图像翻译使用循环一致的对抗性网络。在计算机视觉(ICCV)，2017 年 IEEE 国际会议，2017 年。2

8305