

Semi-supervised reference-based sketch extraction using a contrastive learning framework

Liyan Zhu

摘要

素描反映了个别艺术家的绘画风格；因此，在从各种应用的彩色图像中提取素描时，考虑他们的独特风格是很重要的。遗憾的是，大多数现有的草图提取方法都是设计用于提取单一样式的草图。虽然已经有一些尝试来生成各种样式的草图，但这些方法通常受到两个限制：结果质量较低，以及由于需要成对的数据集而难以训练模型。在本文中，作者提出了一种新的多模式草图提取方法，该方法可以在半监督的方式下，通过非配对数据训练来模仿给定参考草图的风格。本文的方法在定量和定性评价上都优于最先进的草图提取方法和未配对的图像翻译方法。

关键词：草图提取；半监督；多模态；对比学习

1 引言

1.1 选题背景

草图提取是计算机视觉和图形学领域的一个重要问题，它涉及将彩色图像转换为带有线条和轮廓的黑白草图。草图可以用于各种不同的目的，包括艺术创作、思想传达、视觉交流和图像处理等方面。目前已经有一些方法被提出来解决草图提取的问题，但是这些方法通常只能生成单一样式的草图，且需要大量的配对数据。

1.2 选题意义

在实际应用中，人们可能需要使用不同风格和样式的草图。然而，现有的草图提取方法往往无法满足这种需求。因此，开发一种能够生成多种风格草图的方法具有重要的实际意义。

1.3 选题依据

通过对相关文献的调研，我们可以看到目前已经有一些方法提出了针对草图提取的解决方案。然而，这些方法存在一些限制，如只能生成单一样式草图、需要大量配对数据等。因此，基于这些限制，本文选择了研究一种新的草图提取方法，该方法可以模仿输入的参考草图的绘制风格，并且可以在半监督的方式下使用非配对草图和彩色图像进行训练。这种方法可以提高训练过程的效率，并且可以生成多种风格的草图，满足不同应用需求。

2 相关工作

2.1 草图提取

有关草图提取的方法主要有两种，一种是使用边缘检测的方法，如 Canny、Hed 等。一种是使用深度学习的方法，如 SketchKeras, Anime2sketch, Ref2sketch 等。其中，Ref2sketch [1] 是一种多模态素描提取网络，旨在学习从输入的参考素描生成高质量的艺术素描输出。这些基于学习的方法利用一般域图像到图像翻译研究中引入的网络模型之上的附加损失函数和层来提高草图域中的性能。与以往的研究相比，本文提出的方法利用了注意力连接层和一组新的损失函数，以产生更高质量的草图图像。此外，本文提出的模型以半监督的方式使用不配对的数据进行训练，以生成参考图像中给出的风格的草图。

2.2 图像到图像的翻译

图像到图像的翻译方法可以分为四类：有监督、无监督、单模态和多模态。有监督方法需要配对数据进行训练，而无监督方法可以使用不配对数据。由于配对数据的稀缺性，特别是对于艺术家绘制的真实草图无监督方法在数据集收集方面更加便利。单模态方法生成一个输出，而多模态方法可以生成多种输出，可以应用于同一领域中不同风格图像的各种应用。由于不同的艺术家以不同的风格绘制草图，因此在基于草图的图像处理研究中考虑这些风格差异非常重要。所以，多模态方法在素描领域的研究中得到广泛应用，包括照片素描合成和草图自动着色等。

3 本文方法

3.1 本文方法概述

本文的目标是设计一个模型，从给定的彩色图像中提取草图，同时模仿参考草图的风格。由于草图和彩色图像数据对很稀缺，本文选择使用不成对的数据集来训练模型。许多以前的方法仅依赖于循环一致性损失 [4] 来实现在生成器中保留未配对数据之间的视觉相似性。除了使用循环一致性损失之外，本文还引入了两种新颖的损失：线损失和草图样式损失。线损失确保生成的草图形状与输入彩色图像相似，而草图样式损失则强制生成的草图符合参考图像的样式。本文网络的鉴别器 D 检查草图域生成器 $O = G_s(C_i, R_i)$ 的输出是否与参考输入 R_i 位于同一域中。这里， C_i 表示输入彩色图像。 D 确保 O 位于草图域内。主要的网络设计如图 1 所示。

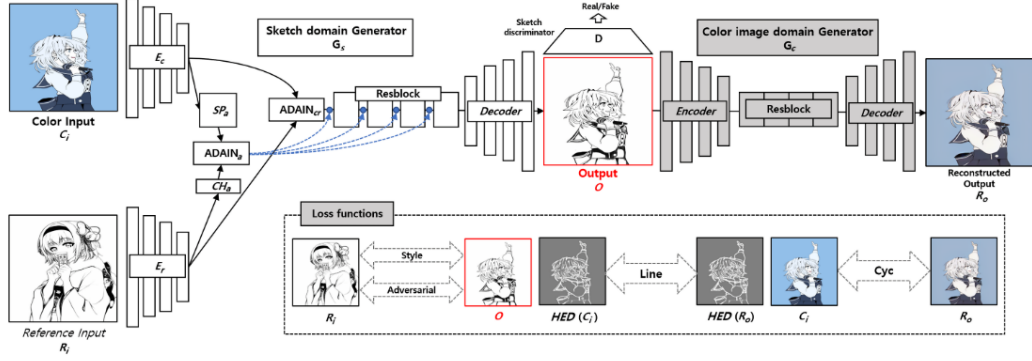


图 1. 主要网络结构图

3.2 主要训练步骤

本文的方法执行以下步骤来训练本文的草图提取模型，该模型产生由参考草图定义的样式的输出。

(A) 由卷积层组成的两个编码器 E_c 和 E_r 分别从输入彩色图像 C_i 和输入参考图像 R_i 提取特征。

(B) 从 E_c 中提取的特征被输入到空间注意力层 S_p 中，从 E_r 中提取的特征被输入到通道注意力层 CH_a 中。然后，注意力层的输出经过自适应实例归一化 $ADAIN_a$ [2]。

(C) 与步骤 (B) 同时， E_c 和 E_r 的特征直接经过另一个自适应实例标准化 $ADAIN_r$ 。

(D) Resblock 由 4 个卷积块层组成，接收来自 $ADAIN_a$ 和 $ADAIN_r$ 的输出。在将组合连接到每个 Resblock 层的输出之前， $ADAIN_a$ 的输出连接到 $ADAIN_r$ 的输出。

(E) Resblock 的输出经过解码器层以生成输出草图 O 。

(F) 为了保留输出草图中输入的彩色图像的形状，将输出草图输入彩色图像域生成器 G_c 。 G_c 由编码器-解码器和 Resblock 层组成，产生重建的输出 R_o 。本文使用 O 和 R_o 计算四种不同的损失函数，以确保输出草图模仿参考输入的风格，同时保留彩色图像输入的形状。

3.3 损失函数

3.3.1 素描风格损失

本文引入了一种新颖的损失函数，用于计算草图域中的风格差异。这种损失是使用基于对比学习预先训练的权重来计算的。为了计算风格损失，本文使用由草图域生成器 G_s 生成的参考输入 R_i 和输出 $O=G_s(R_i, C_i)$ 。这两幅图像虽然形状不同，但风格应该是相同的；因此，本文使用预先训练的对比权重从中提取风格特征嵌入。然后本文应用 L1 归一化来计算两个嵌入的差异。损失表示如下：

$$\mathcal{L}_{\text{style}} = \mathbb{E}_{O,R} [\|C_w(O) - C_w(R_i)\|_1]. \quad (1)$$

符号 C_w 表示提取风格特征嵌入的预训练对比学习模型。如果没有这种风格损失函数，网络就无法模仿参考输入的绘画风格。因此无论参考输入如何，都会产生固定风格的输出。并且，该损失函数是用成对数据进行预训练的，从而使得本文的主模型能够用不成对数据进行半监督训练。

3.3.2 线损失

为了强制输出的形状与彩色图像输入相同，本文应用了一个损失函数，该函数使用 HED 比较由颜色域生成器 G_c 生成的重建输出 $R_o = G_c(O)$ 的边缘和颜色输入 C_i 。HED 检测输入图像的边缘。损失表示如下：

$$\mathcal{L}_{\text{line}} = \mathbb{E}_{C_i, R_o} \left[\sum_l |\phi_l(\text{HED}(C_i)) - \phi_l(\text{HED}(R_o))|_1 \right], \quad (2)$$

两个边缘检测图像 $\text{HED}(C_i)$ 和 $\text{HED}(R_o)$ 之间的差异由感知损失函数计算，旨在比较基于预训练 VGG16 模型的图像。 ϕ_l 表示 VGG16 [3] 网络第 l^{th} 层的激活图。如果没有此线损失函数，网络会产生看起来与彩色图像输入不同的形状。

3.3.3 循环一致性损失

本文通过比较 C_i 和 R_o 之间的整体视觉差异进一步增强形状相似性。采用 L1 归一化，损失表示如下：

$$\mathcal{L}_{\text{cyc}} = \mathbb{E}_{C_i, R_o} [\|C_i - R_o\|_1], \quad (3)$$

3.3.4 对抗性损失

对抗性损失迫使鉴别器 D 将合成输出 O 分类到类似于 R_i 定义的域中，这在本文的案例中是一个草图。损失表示如下：

$$\mathcal{L}_{\text{adv}} = \mathbb{E}_R [\log(D(R_i))] + \mathbb{E}_O [\log(1 - D(O))], \quad (4)$$

生成器 G 和判别器 D 的总损失函数定义如下：

$$\min_G \max_D \mathcal{L}_{\text{total}} = \lambda_{\text{style}} \mathcal{L}_{\text{style}} + \lambda_{\text{line}} \mathcal{L}_{\text{line}} + \lambda_{\text{cyc}} \mathcal{L}_{\text{cyc}} + \lambda_{\text{adv}} \mathcal{L}_{\text{adv}}. \quad (5)$$

其中参数 $\lambda_{\text{style}, \text{line}} = 5 - \frac{4.5i}{n}$ (i 是当前迭代数， n 是总的迭代数。)， $\lambda_{\text{cyc}} = 10$ ， $\lambda_{\text{adv}} = 1$ 。

4 复现细节

4.1 与已有开源代码对比

本次复现是参考了论文作者开源的代码基础上完成的。在复现的过程中，自己主要进行了两个工作，第一个工作时直接对原论文直接进行复现，这个过程自己花了很多精力和时间才完成。第二个工作是对原论文进行创新的试验，即增加数据集的样本数量，并重新设置一定的比例来生成训练集和测试集；修改训练过程中需要的参数大小，从而选择更合适的参数值。

4.2 实验环境搭建

操作系统: Linux

深度学习框架: Pytorch

重要库版本: Python 3.11.4,torch>=1.4.0,torchvision>=0.5.0,dominate>=2.4.0,visdom>=0.1.8.8

4.3 实验设计

原文实验的数据集包括随机选择的彩色图像和四种不同风格的参考草图，本人在此数据集中添加了一定数量的彩色图像。并在训练中使用了 batch 大小为 2 的 Adam 优化器和修改了 model 值。由于机器算力的原因，训练的迭代次数设置为 200 轮。所有网络从零开始训练，学习率为 0.0002。由训练完成后 loss 值与验证集上的表现判断，模型已正常收敛。

5 实验结果分析

模型评估主要是对模型训练完成后的 loss 值和学习率以及验证结果进行评估。

(1) 训练后的 loss 值

- GAN 中生成器的训练损失

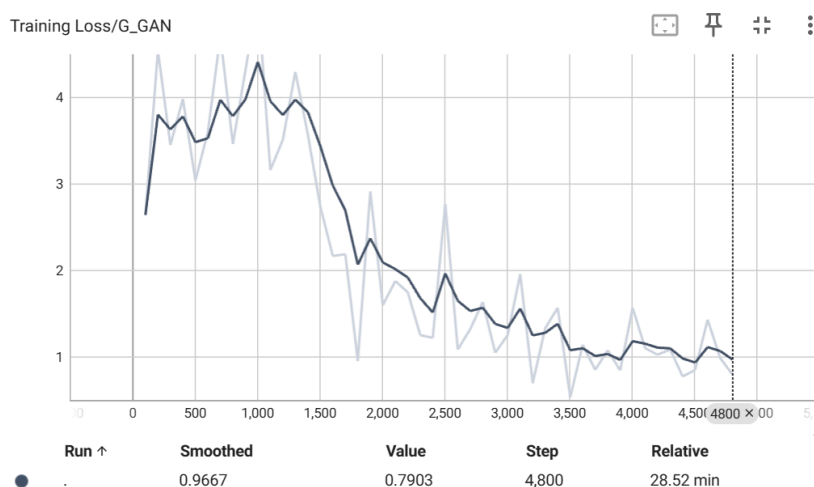


图 2. GAN 中生成器的训练损失示意图

该图展示了生成对抗网络（GAN）中生成器的训练损失。随着时间的推移，生成器在生成数据方面的效果在改善，因为损失值总体呈下降趋势。这是训练有效的一个正面指标，尽管在某些步骤中有一些波动。这些波动可能是由于训练过程中的各种因素引起的。

- GAN 中生成器的线损失

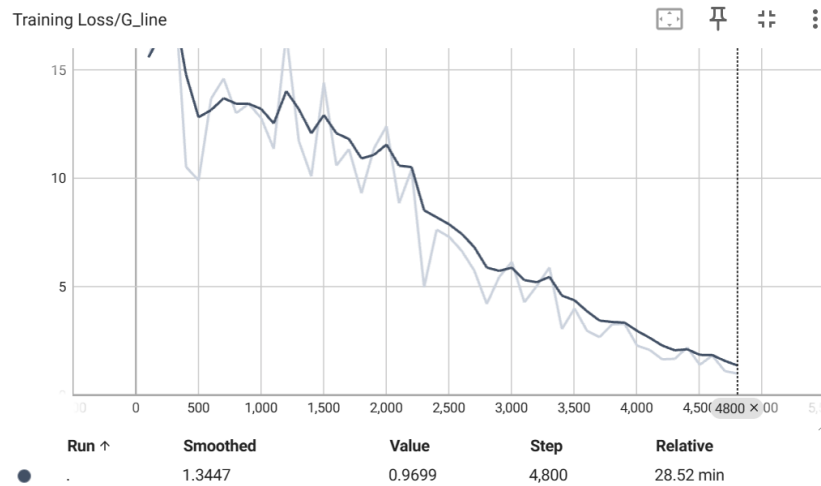


图 3. GAN 中生成器的线性损失示意图

该图展示了 GAN 中生成器在训练过程中的学习进展，其线性损失随着时间的推移而降低。说明模型在不断地学习和优化。

(2) 学习率

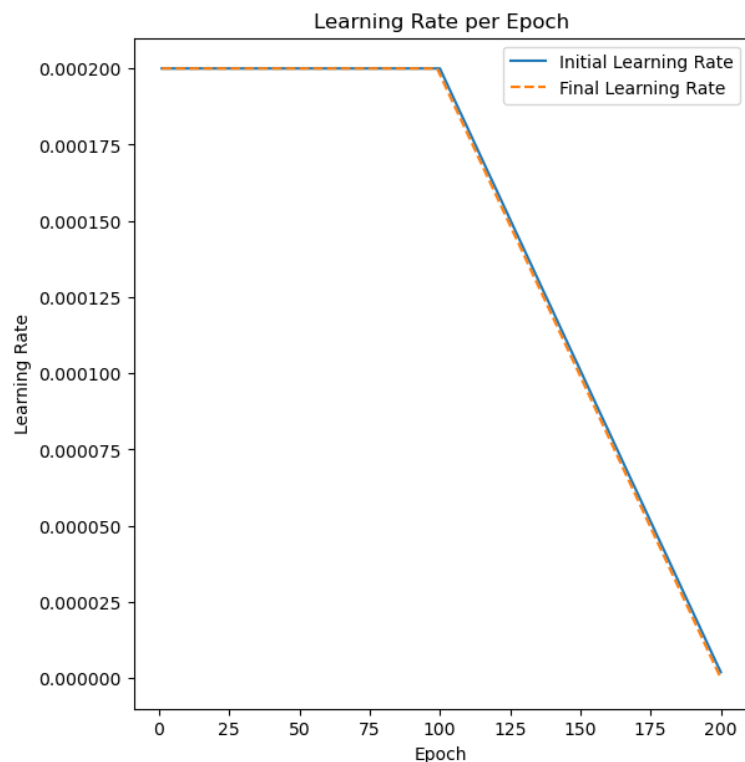


图 4. 学习率随迭代周期的变化示意图

该图反映了一个学习率衰减策略，其中学习率在训练初期保持恒定，随后逐渐减小，最终在训练结束时降到 0，确保在训练结束时不再对模型权重进行任何更新，这有助于模型的收敛。

(3) 验证结果

- 线条明显的彩色图像



图 5. 线条明显的彩色图像所生成的不同风格草图

- 线条不明显的彩色图像

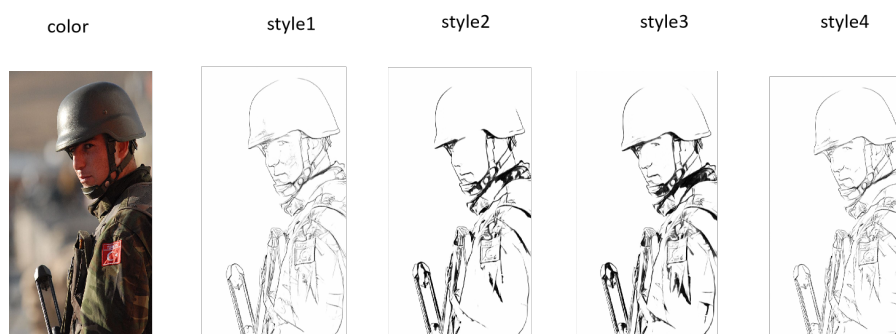


图 6. 线条不明显的彩色图像所生成的不同风格草图

由以上两张图可知，线条明显的彩色图像生成了对应风格的正确草图，并且损失很小。而线条不是很明显的彩色图像生成的有些草图并没有很好地保留彩色图像的形状，并出现了形状缺失的问题。

6 总结与展望

本文提出了一种新的多模态模型，用于从给定参考图像的风格中提取彩色图像的草图。该模型通过半监督训练，利用基于对比学习的预训练素描风格的损失来模仿参考素描的风格。同时，该模型能够在高分辨率下进行训练，从而产生比基线质量更好的结果。然而，本人在此次复现实验中发现，该模型对线条不明显的彩色图像所生成的不同风格草图的效果并不令人满意，可能是缺少相关数据集的原因。因此，在未来工作中我们可以通过提供相应风格的数据集进行训练来解决该问题。

参考文献

- [1] Amirsaman Ashtari, Chang Wook Seo, Cholmin Kang, Sihun Cha, and Junyong Noh. Reference based sketch extraction via attention mechanism. *ACM Transactions on Graphics*, 41(6):16, 2022.

- [2] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017.
- [3] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [4] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*.