# 基于对抗自编码器的矢量草图生成方法

赵鹏[1,2)], 高杰超[1,2)], 周彪[1,2)], 刘慧婷[1,2)]

[1)] (安徽大学计算智能与信号处理教育部重点实验室 合肥 230601)
[2)] (安徽大学计算机科学与技术学院 合肥 230601)
(zhaopeng_ad@163.com)

**摘 要:** 针对现有矢量草图生成方法存在的生成结果潦草, 以及编码草图信息单一等问题, 提出一种基于对抗自编码器的矢量草图生成方法. 借助对抗自编码器自身所具有的对抗的机制, 将像素化表示的草图所具有的空间信息融合到矢量草图的生成过程中, 使得生成的草图具有更好的类别形状信息. 既利用了矢量草图所包含的笔画间的时序信息, 又利用了像素草图所包含的绘画物体的形状信息. 在 QuickDraw 数据集上进行了草图生成实验, 并使用 Ske-score 评价指标进行量化度量, 实验结果表明所提方法能够缓解生成结果出现的潦草效应, 并且生成的草图具有更好的视觉美观性和更高程度的类别可辨识性.

**关键词:** 草图生成; 矢量草图; 对抗自编码器; 生成式对抗网络; 信息融合
**中图法分类号:** TP391.41 **DOI:** 10.3724/SP.J.1089.2020.18252

# A Novel Vector Sketch Generation Method Based on Adversarial Autoencoder

Zhao Peng[1,2)], Gao Jiechao[1,2)], Zhou Biao[1,2)], and Liu Huiting[1,2)]

[1)] (*Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University, Hefei* 230601)
[2)] (*School of Computer Science and Technology, Anhui University, Hefei* 230601)

**Abstract:** Aiming at the problems of existing vector sketch generation methods, such as generated sketch scribble and single coding sketch information, this paper proposed a novel vector sketch generation method based on adversarial autoencoder (called Sketch-AAE). Sketch-AAE takes advantage of the adversarial mechanism to merge the spatial information of a raster sketch into the vector sketch generation, which makes the generated sketch having better category shape information. The proposed method utilizes not only the temporal information among the strokes in vector sketch, but also the spatial information of the object in raster sketch. The extensive sketch generation experiments were conducted on the QuickDraw dataset, and the Ske-score was used as the quantitative measurement. The experimental results show that the proposed method can alleviate the scribble effect of the generated sketches and achieves better visual impression and higher category discrimination.

**Key words:** sketch generation; vector sketch; adversarial autoencoder; generative adversarial networks; information fusion

收稿日期: 2020-02-27; 修回日期: 2020-05-28. **基金项目**: 国家自然科学基金(61602004); 安徽省高校自然科学研究重点项目(KJ2018A0013, KJ2017A011); 安徽省自然科学基金(1908085MF188, 1908085MF182); 安徽省重点研究与开发计划(1804d08020309). **赵鹏**(1976—), 女, 博士, 副教授, 硕士生导师, CCF 会员, 主要研究方向为机器学习、图像理解; **高杰超**(1995—), 男, 硕士研究生, 主要研究方向为机器学习、草图生成; **周彪**(1996—), 男, 硕士研究生, 主要研究方向为机器学习; **刘慧婷**(1978—), 女, 博士, 副教授, 硕士生导师, CCF 会员, 主要研究方向为机器学习.

手绘草图因其简洁性和丰富的表达能力, 一直以来都是人类社会一种简单快捷的沟通方式. 然而对机器来说, 绘制草图和理解手绘草图中蕴含的丰富语义信息, 却是一个很大的挑战. 教会机器以人类的绘画方式绘画出一幅草图, 一方面有助于机器对草图的理解, 更好地进行人机交互; 另一方面, 可以利用机器生成草图有效地扩充现有的草图数据集. 不同于自然图像样本, 草图样本必须依靠人类的手绘来获取, 所以获取规整的、有标注的草图数据集需要更高的成本, 因而草图生成具有重要的研究价值. 基于草图的相关研究已经成为计算机视觉与计算机图形学中非常活跃的研究领域, 涉及范围非常广泛, 包括草图识别[1-3], 草图分割[4-7], 基于草图的图像检索[8-11]和基于草图的建模[12]等.

由于草图笔画的稀疏性, 即只有少数被草图笔画覆盖到的位置有像素值, 其余大部分位置的像素值均为空, 因此基于像素的表示方法会造成草图特征表示的稀疏性, 而这种稀疏的表示不利于使用卷积神经网络(convolutional neural networks, CNN)来建模, 因而草图生成的研究具有很大的挑战性. Ha 等[13]首次选择从矢量表示的角度来对草图进行建模. Chen 等[14]在此基础上进一步将模型扩展到生成多类的草图.

现有的矢量草图的生成方法大多基于变分自编码器(variational autoencoder, VAE)[15]的框架. 基于 VAE 生成的光栅自然图像通常比较模糊. 这一模糊现象在矢量草图上被称为“潦草效应”[16], 具体表现为笔画状态倾向于一直停留在纸上, 而不是离开纸面在另一个位置开始下一笔画. 近年来, 生成式对抗网络(generative adversarial networks, GAN)[17]已经广泛地应用于计算机视觉领域的很多任务中. GAN 中判别器网络(discriminator, D)学习区分一个给定的样本是真实的样本还是生成的样本, 而生成器网络(generator, G)通过学习生成高质量的样本来迷惑判别器. 两者在训练过程中相互对抗, 不断优化直至达到平衡. 对抗自编码器(adversarial autoencoder, AAE)[18]就是将这种对抗的思想引入自编码器(autoencoder, AE), 用对抗训练替代 VAE 中的变分推断, 并通过实验证明 AAE 比 VAE 具有更规整的隐空间分布, 能更好地捕捉数据分布. 对抗机制的引入, 有助于缓解 VAE 在草图生成上所出现的潦草效应, 同时利用对抗机制可将任意先验分布匹配至 AE 的隐空间.

草图通常有基于矢量和基于像素 2 种表示形式, 它们蕴含的主要信息是不同的. 其中, 矢量表示主要包含的是草图的笔画间的时序信息, 而像素表示主要包含的是草图的空间形状信息. sketch-RNN 使用了一个双向循环神经网络(bi-directional recurrent neural networks, BRNN)[19]来编码矢量表示的草图. 循环神经网络(recurrent neural networks, RNN)善于处理这种序列化的数据, 能够利用到笔画的上下文以及时序的动态信息, 所以编码到隐空间的主要为笔画相关的特征. sketch-pix2seq[14]使用 CNN 来编码像素表示的草图, CNN 适合处理网格化的数据, 能够捕捉到图片中物体的空间局部结构, 所以编码到隐空间的主要为形状相关的特征. 然而上述 2 种方法都只编码了草图的一种单一的表示形式, 而本文方法同时融合了 2 种草图表示形式所具有的笔画时序信息和视觉形状信息.

近年来, 矢量草图生成取得了一些研究成果, 但仍存在生成草图笔画潦草、类别辨识度不高等问题. 为了解决现有方法的缺陷, 生成更为贴近人类手绘风格的草图, 本文提出一种基于 AAE 的矢量草图生成方法(sketch-AAE). 矢量表示的草图具有的笔画时序信息和像素表示的草图具有的空间形状信息都有利于草图的生成, 所以本文利用生成对抗机制, 将像素表示形式所包含的空间形状信息融合进矢量草图的生成过程. 本文针对 VAE 在矢量草图生成上出现的潦草效应, 选择使用 AAE 的框架代替 VAE. AAE 比 VAE 具有更规整的隐空间分布, 且对抗机制的引入也有助于缓解生成结果所出现的潦草效应. 此外, 本文选择从矢量表示的角度生成草图, 同时利用 AAE 的框架, 在模型中引入像素表示草图所具有的视觉语义信息, 融合草图不同表示形式所具有的笔画时序信息和空间形状等视觉信息, 使生成的草图不失创造性的同时又具有一定的可识别性.

## 1 相关工作

### 1.1 矢量草图生成

Ha[20]首先在矢量图片生成上进行了尝试, 使用了一个双层的长短期记忆网络(long short-term memory, LSTM)生成表示为矢量形式的汉字字符. 接着, 又提出了基于 VAE 框架的矢量草图生成模型 sketch-RNN[13], 该模型使用了谷歌公司的大型草图数据集 QuickDraw. 该数据集中的草图均表示为矢量的形式, 本文也选择了该数据集作为训练

数据集. 当在单个类上进行训练时, sketch-RNN 可以生成视觉上美观的草图, 但当使用多个类混合的训练集训练单个模型时, sketch-RNN 的生成结果会很差. 因此, 为了解决这个问题, Chen 等[14]将 sketch-RNN 中的编码器替换为 CNN, 并且移除了 KL 散度损失, 提出 sketch-pix2seq 模型. 因为使用了 CNN, 所以 sketch-pix2seq 的输入为 QuickDraw 数据集中矢量草图的光栅化格式(即像素表示形式). Zhong[21]将 sketch-RNN 模型扩展成了一个端到端的模型, 该模型输入的是 SVG (scalable vector graphics)格式的字体, 使用 Google Fonts Dataset 作为训练集来训练模型, 生成各种新奇的字体. 以上介绍的模型均是基于 VAE 框架. 然而, 基于 VAE 的方法往往生成的草图具有潦草效应. 图 1 是 sketch-RNN 生成结果出现的潦草效应的示例. Varshaneya 等[16]使用了 GAN 框架来生成矢量化草图, 采用策略梯度算法来建模离散的笔画状态, 但模型复杂度较高, 训练难度较大.

a. mosquito　　b. yoga pose

图 1　sketch-RNN 生成结果出现的潦草效应示例

### 1.2　对抗自编码器

AAE 是 Makhzani 等[18]提出的一种概率 AE. 它使用了 GAN 框架, 将 AE 的隐变量的后验分布匹配至任意一个先验分布, 以此来实现变分推断. 与 VAE 不同的是, AAE 是通过对抗机制, 使用一个可学习的判别器网络来对隐空间施加先验分布, 而 VAE 使用 KL 距离惩罚来达到这一目的. AAE 模型的框架结构如图 2 所示.

假设 $\boldsymbol{x}$ 为输入, $\boldsymbol{z}$ 是 AE 的隐变量, 该 AE 包括编码器和解码器 2 个部分. $p(\boldsymbol{z})$ 是准备施加的先验分布, $q(\boldsymbol{z}|\boldsymbol{x})$ 是编码分布, $p(\boldsymbol{x}|\boldsymbol{z})$ 是解码分布. 并且假设 $p_{\mathrm{d}}(\boldsymbol{x})$ 是数据分布, $p(\boldsymbol{x})$ 是模型分布. AE 的编码函数在隐空间上定义了一个后验分布 $q(\boldsymbol{z})$, 即

$$q(\boldsymbol{z})=\int_{\boldsymbol{x}} q(\boldsymbol{z}|\boldsymbol{x})p_{\mathrm{d}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{1}$$

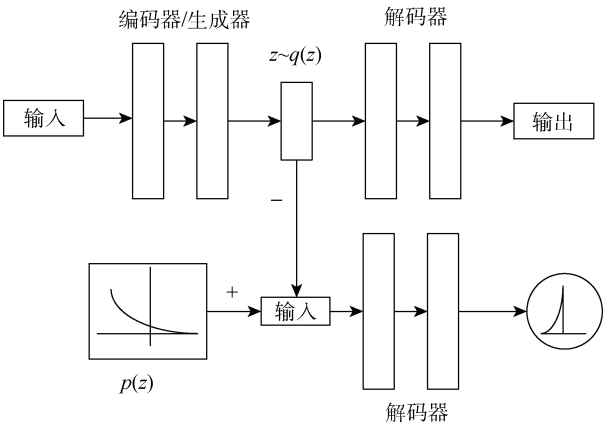



图 2　AAE 模型的框架结构图

对抗网络被附加在 AE 的隐编码上, 通过将后验分布 $q(\boldsymbol{z})$ 匹配到任意给定的先验分布 $p(\boldsymbol{z})$ 来实现正则化. 对抗网络训练 $q(\boldsymbol{z})$ 去匹配 $p(\boldsymbol{z})$, 同时 AE 最小化重构误差. 对抗网络的生成器为 AE 的编码器, 训练该编码器, 力图使编码器获得的后验分布 $q(\boldsymbol{z})$ 可以骗过判别器, 让它认为隐变量 $\boldsymbol{z}(\boldsymbol{z}\sim q(\boldsymbol{z}))$ 来自先验分布 $p(\boldsymbol{z})$.

AAE 分 2 个阶段利用随机梯度下降算法(stochastic gradient descent, SGD)联合训练, 即每个 mini-batch 的训练都包括 2 个阶段: 执行重构阶段和正则化阶段. 在重构阶段, AE 以最小化输入的重构误差为目标来训练更新编码器和解码器. 在正则化阶段, 对抗网络首先更新判别器网络, 以区分生成样本(由 AE 的编码器计算出隐变量)和真实样本(从先验分布中采样的变量); 然后对抗网络更新其生成器(即 AE 编码器), 来骗过判别器. 一旦完成训练之后, AE 的解码器即定义了一个将施加的先验分布映射到数据分布的生成模型.

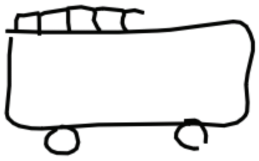
## 2　本文方法

### 2.1　数据集

本文模型训练和测试采样均使用 QuickDraw 数据集[13], 其中的矢量草图样本是从 Quick, Draw! 项目中获得的. 该项目是一个在线游戏, 它要求玩家在 20 s 内画出一个属于特定类的物体. QuickDraw 数据集中的草图包含了 345 个类别, 每个类的训练集、验证集和测试集包含草图样本数量分别为 70 000, 2 500 和 2 500. 数据集中样本示例如图 3 所示, 图 3a 和图 3b 分别为同一草图的矢量表示和像素表示.

[0 −5 1 0 0]
[13 6 1 0 0]
[23 −4 1 0 0]
[15 −4 1 0 0]
[7 3 0 1 0]
[−30 0 1 0 0]
⋮
[6 −1 1 0 0]
[5 1 1 0 0]
[−45 9 0 1 0]
[0 0 0 0 1]

a. 矢量表示



b. 像素表示

图 3 QuickDraw 样本示例

矢量表示方法将草图表示为一系列有序的笔触动作. 在这种表示方法下, 一幅草图即是一组点序列, 其中每一个点都用一个包含 5 个元素的向量表示为 $(\Delta x, \Delta y, p_1, p_2, p_3)$. 前 2 个元素分别表示当前点距离前一个点的 $x$ 和 $y$ 方向上的距离, 后面 3 个元素是一个 one-hot 向量, 表示 3 种笔画状态. 当 $p_1$=1 时, 表示画笔在当前坐标点触碰纸面, 并且当前点会和下一点相连; 当 $p_2$=1 时, 表示画笔在当前坐标点离开纸面, 故当前点不会和下一点相连; 当 $p_3$=1 时, 表示绘画结束, 包括当前坐标点以及之后的所有点都不会被提交为草图的有效笔画.

### 2.2 sketch-AAE

本文提出的 sketch-AAE 模型总体上采用了 AAE 的框架, 模型主要分为自编码器和对抗网络 2 个模块, 总体框架如图 4 所示. 自编码器模块包括 AE-RNN 和 AE-CNN. 其中, AE-RNN 为序列到序列的 AE 模块. 与 sketch-RNN 不同的是, sketch-AAE 中的 AE 模块舍弃了 sketch-RNN 损失函数中的 KL 距离损失项, 使用对抗网络约束 AE 的隐空间来实现正则化. 由于 sketch-RNN 使用的 VAE 框架在矢量草图的生成上容易出现潦草效应, 本文模型通过引入对抗机制改善这一状况. 而 AE-CNN 为预训练完成的 sketch-pix2seq[14]模型, 与 AE-RNN 不同的是, 该模型使用 CNN 来编码像素表示的草图. 通过使用预训练完成的 AE-CNN 的编码器, sketch-AAE 在训练过程中能够引入像素表示的草图所具有的空间形状信息. 通常人们在评价一幅草图的好坏时, 更多地关注该幅草图的视觉空间结构是否符合特定物体的要求, 而像素表示的草图所包含的正是这一类视觉信息, 利用 CNN 将这类信息提取出来, 再将其引入到矢量草图的生成过程中, 将进一步改善模型所生成草图的质量(即视觉评价上的“好坏”).

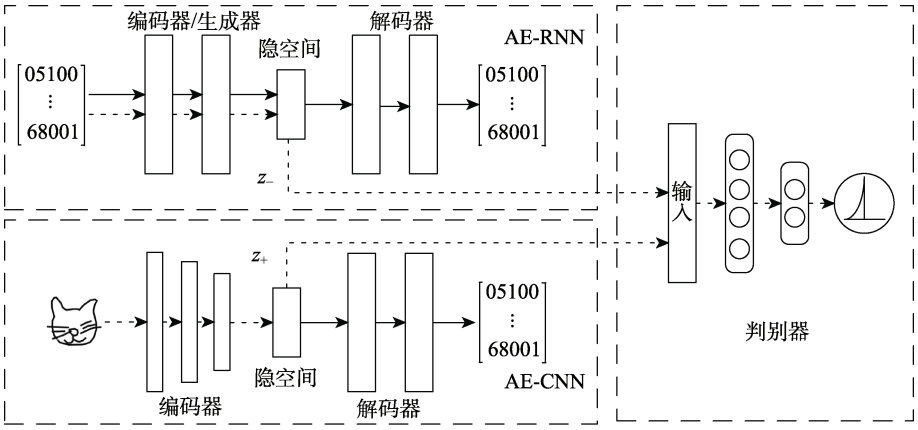



图 4 sketch-AAE 模型的框架结构图

受 GAN 的对抗思想启发, 本文提出 sketch-AAE 模型, 利用一个可学习的判别器网络来向 AE-RNN 的隐空间施加约束. 不同于大多数采用对抗思想的生成模型, sketch-AAE 的判别器网络并不是用来直接评价最终生成结果的好坏, 而是通过它来约束 AE-RNN 的隐空间, 对抗机制在这里是一种正则化手段. 对抗网络模块由 AE-RNN 的编码器和判别器组成. 判别器的正样本(即需要判别为真的样本)为预训练完成的 AE-CNN 的隐空间采样得来的隐变量; 判别器的负样本(即需要判别为假的样本)为 AE-RNN 的隐空间采样得来的隐向量. 随着对抗训练的进行, 判别器网络将会指导 AE-RNN 的隐空间逐步向 AE-CNN 的隐空间靠近. 换言之, 逐步向 AE-RNN 的隐空间施加约束.

### 2.3 自编码器模块

AE-RNN 框架如图 5 所示, 它由编码器和解码器组成, 目的是根据给定的输入重构出一幅矢量草图. 为了能够充分地捕捉到矢量草图笔画间的关系, 并将其编码到隐空间, 从而更准确地重构出矢量草图, AE-RNN 的编码器采用了 BRNN 框架.

BRNN 框架的基本单元是 LSTM, 输入为一幅矢量化的草图 $\boldsymbol{S}$, 输出为大小 $N_z$ 的隐向量. 因为使用了双向的 RNN, 所以还需要输入 $\boldsymbol{S}$ 的反序列 $\boldsymbol{S}_{\text{reverse}}$, 经过一系列 LSTM 的操作, 分别得到 2 个最终的隐状态: $\boldsymbol{h}_{\rightarrow}$ 和 $\boldsymbol{h}_{\leftarrow}$. 然后连接 $\boldsymbol{h}_{\rightarrow}$ 和 $\boldsymbol{h}_{\leftarrow}$, 获得一个串联的隐状态 $\boldsymbol{h}$ ($\boldsymbol{h}=[\boldsymbol{h}_{\rightarrow};\boldsymbol{h}_{\leftarrow}]$). 将 $\boldsymbol{h}$ 输入到一个全连接层, 映射成 2 个大小为 $N_z$ 的向量 $\boldsymbol{\mu}$ 和 $\hat{\boldsymbol{\sigma}}$. 为了确保获得的标准差参数非负, 本文利用指数运算将 $\hat{\boldsymbol{\sigma}}$ 转化为 $\boldsymbol{\sigma}$. 之后采用 VAE 中的重参数化技巧, 使用 $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$ 以及从标准高斯分布中采样得到的大小为 $N_z$ 的向量 $\boldsymbol{N}(0,\boldsymbol{I})$, 构造一个随机向量 $\boldsymbol{z}\in\mathbb{R}^{N_z}$, 即

$$\boldsymbol{z}=\boldsymbol{\mu}+\boldsymbol{\sigma}\odot\boldsymbol{N}(0,\boldsymbol{I}) \tag{2}$$

因为使用了式(2)中的构造方法, 所以隐向量 $\boldsymbol{z}$ 不再是一个确定的输出向量, 而是一个以输入草图为条件的随机向量.

AE-RNN 的解码器是一个自回归的 RNN. 它的作用是根据式(2)中得到隐变量 $\boldsymbol{z}$ 以及每个时间步的输入, 来逐步地重构出完整的矢量草图. 解码器在时间步 $i$ 的输入为一个由点向量 $\boldsymbol{S}_{i-1}$ 和隐变量 $\boldsymbol{z}$ 连接而成的向量 $\boldsymbol{x}_{i-1}$, 其中 $\boldsymbol{S}_{i-1}(i>1)$ 为输入矢量草图的第 $i$ 个点, $\boldsymbol{S}_0=(0,0,1,0,0)$. 时间步 $i$ 的输出 $\boldsymbol{y}_i$ 为重构矢量草图第 $i$ 个数据点 $\boldsymbol{S}_i$ 的概率分布参数.

图 5 中的基本单元是经典的 LSTM. 假设时间步 $t$ 的输入为 $\boldsymbol{x}_t$, 则 LSTM 中当前时间步的输入门 $\boldsymbol{i}_t$, 遗忘门 $\boldsymbol{f}_t$, 输出门 $\boldsymbol{o}_t$, 单元状态 $\boldsymbol{c}_t$ 和隐藏状态 $\boldsymbol{h}_t$ 的计算分别为

$$\begin{aligned}
&\boldsymbol{i}_t=\text{sigmoid}(\boldsymbol{W}_{\text{i}}\boldsymbol{x}_t+\boldsymbol{U}_{\text{i}}\boldsymbol{h}_{t-1}+\boldsymbol{b}_{\text{i}}),\\
&\boldsymbol{f}_t=\text{sigmoid}(\boldsymbol{W}_{\text{f}}\boldsymbol{x}_t+\boldsymbol{U}_{\text{f}}\boldsymbol{h}_{t-1}+\boldsymbol{b}_{\text{f}}),\\
&\boldsymbol{o}_t=\text{sigmoid}(\boldsymbol{W}_{\text{o}}\boldsymbol{x}_t+\boldsymbol{U}_{\text{o}}\boldsymbol{h}_{t-1}+\boldsymbol{b}_{\text{o}}),\\
&\boldsymbol{c}_t=\boldsymbol{f}_t\odot\boldsymbol{c}_{t-1}+\boldsymbol{i}_t\odot\tanh(\boldsymbol{W}_{\text{c}}\boldsymbol{x}_t+\boldsymbol{U}_{\text{c}}\boldsymbol{h}_{t-1}+\boldsymbol{b}_{\text{c}}),\\
&\boldsymbol{h}_t=\boldsymbol{o}_t\odot\tanh(\boldsymbol{c}_t).
\end{aligned}$$

其中, $(\boldsymbol{W}_{\text{i}},\boldsymbol{U}_{\text{i}})$, $(\boldsymbol{W}_{\text{f}},\boldsymbol{U}_{\text{f}})$, $(\boldsymbol{W}_{\text{o}},\boldsymbol{U}_{\text{o}})$ 和 $(\boldsymbol{W}_{\text{c}},\boldsymbol{U}_{\text{c}})$ 分别为输入门、遗忘门、输出门以及单元状态的参数矩阵, $\boldsymbol{b}_{\text{i}}$, $\boldsymbol{b}_{\text{f}}$, $\boldsymbol{b}_{\text{o}}$ 和 $\boldsymbol{b}_{\text{c}}$ 分别为输入门、遗忘门、输出门以及单元状态的偏置矩阵.

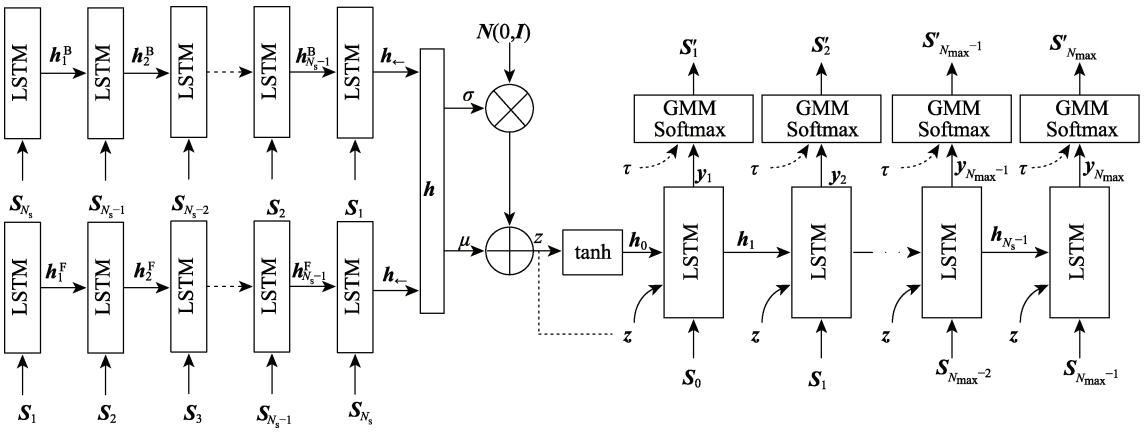



图 5　AE-RNN 框架结构图

在解码器的每个时间步, 使用由 $M$ 个正态分布组成的高斯混合模型(Gaussian mixture model, GMM)来建模点坐标偏移量 $(\Delta x,\Delta y)$, 使用类别分布来建模笔画状态 $(p_1,p_2,p_3)$. GMM 是一种通过将简单分布组合生成复杂分布的模型, 相比于单个的正态分布, 使用由多个正态分布组成的混合模型能够更好地拟合点坐标分布. 解码器时间步 $t$ 的输出向量 $\boldsymbol{y}_t(\boldsymbol{y}_t=\boldsymbol{W}_{\text{y}}\boldsymbol{h}_t+\boldsymbol{b}_{\text{y}},\ \boldsymbol{y}_t\in\mathbb{R}^{6M+3})$ 包括长度为 3 的笔画状态向量 $(q_1,q_2,q_3)$, $M$ 个混合权重 $\Pi_i(i=1,2,\cdots,M)$, 以及 $M$ 个二元正态分布 $\mathcal{N}(x,y\,|\,\mu_x,\mu_y,\sigma_x,\sigma_y,\rho_{xy})$ 的参数, 即

$$\begin{aligned}
\boldsymbol{y}_t=[&(\hat{\Pi}_1,\mu_x,\mu_y,\hat{\sigma}_x,\hat{\sigma}_y,\hat{\rho}_{xy})_1,\cdots,\\
&(\hat{\Pi}_M,\mu_x,\mu_y,\hat{\sigma}_x,\hat{\sigma}_y,\hat{\rho}_{xy})_M,(\hat{q}_1\hat{q}_2\hat{q}_3)].
\end{aligned}$$

为了确保标准差非负以及相关系数在–1~1, 令 $\sigma_x=\exp(\hat{\sigma}_x)$, $\sigma_y=\exp(\hat{\sigma}_y)$, $\rho_{xy}=\tanh(\hat{\rho}_{xy})$.

本文将通过最小化重构损失函数项 $L_{\text{r}}$ 来训练 AE-RNN, 即最大化生成的概率分布的对数似然来优化模型. $L_{\text{r}}$ 由 2 个部分组成: 重构点坐标偏移量 $(\Delta x,\Delta y)$ 的对数损失 $L_{\text{s}}$ 和重构笔画状态 $(p_1,p_2,p_3)$ 的对数损失 $L_{\text{p}}$, 即

$$L_{\text{r}}=L_{\text{s}}+L_{\text{p}},$$

$$L_{\mathrm{s}}=-\frac{1}{N_{\max}}\sum_{i=1}^{N_{\mathrm{s}}}\log(p(\Delta x,\Delta y)_i),$$

$$L_{\mathrm{p}}=-\frac{1}{N_{\max}}\sum_{i=1}^{N_{\max}}\sum_{k=1}^{3}p_{k,i}\log(q_{k,i}).$$

其中，$(\Delta x,\Delta y)$ 为训练集样本；$N_{\max}$ 表示当前训练集中最长草图所具有的笔画数；$N_{\mathrm{s}}$ 表示当前草图所具有的笔画数；$p(\Delta x,\Delta y)_i=\sum_{j=1}^{M}\Pi_{j,i}\mathcal{N}(\Delta x_i,\Delta y_i)_j$ 为 $(\Delta x,\Delta y)$ 的概率分布函数，

$$\sum_{j=1}^{M}\Pi_{j,i}=1,$$

$$\mathcal{N}(\Delta x_i,\Delta y_i)_j=\frac{1}{2\pi\sigma_{x,j,i}\sigma_{y,j,i}\sqrt{1-\rho_{xy,j,i}^2}}\cdot\exp\left[\frac{-Z_{j,i}}{2(1-\rho_{xy,j,i}^2)}\right],$$

$$Z_{j,i}=\frac{(\Delta x-\mu_{x,j,i})^2}{\sigma_{x,j,i}^2}+\frac{(\Delta y-\mu_{y,j,i})^2}{\sigma_{y,j,i}^2}-\frac{2\rho_{xy,j,i}(\Delta x-\mu_{x,j,i})(\Delta y-\mu_{y,j,i})}{\sigma_{x,j,i}\sigma_{y,j,i}}.$$

AE-CNN 与 AE-RNN 的唯一区别是将编码器的结构替换为 CNN，原因是 CNN 更善于捕捉像素化图片的局部空间结构. 首先，将 QuickDraw 数据集中的矢量化草图转化为像素化草图，即将数据集中的原始序列转化为 SVG 格式；再将 SVG 格式转化为 $48\times48\times1$ 的 PNG 格式图片. 与 AE-RNN 一样，AE-CNN 也没有采用 KL 距离损失项. 由于本文使用 AE-CNN 的目的是对草图在像素空间中的信息进行编码，所以应尽可能多地将像素化草图具有的信息编码到隐空间，移除 $L_{\mathrm{KL}}$ 项，只使用 $L_{\mathrm{r}}$ 项，可以使 AE-CNN 专注于优化重构损失，从而尽可能多地将有用的信息编码到隐空间. AE-CNN 的解码器和训练所用的重构损失 $L_{\mathrm{R}}$ 均与 AE-RNN 相同.

### 2.4 对抗网络模块

对抗网络的生成器 $G$ 为 AE-RNN 的编码器，由它产生判别器 $D$ 的负样本 $\boldsymbol{z}_-$，即 $\boldsymbol{z}_-=En(\boldsymbol{x})$，再利用预训练完成的 AE-CNN 的编码器产生正样本 $\boldsymbol{z}_+$.

因为本文模型所使用的隐向量维度比较小，所以对抗网络模块中的判别器 $D$ 使用的是简单的多层感知机(multi-layer perception, MLP)，一定程度上节约了训练成本. 训练过程与经典 GAN[17]的训练过程相同，其损失函数为

$$L_{\mathrm{GAN}}=\min_{\mathrm{G}}\max_{\mathrm{D}}E_{\boldsymbol{z}_+\sim p_{\boldsymbol{z}_+}}[\log D(\boldsymbol{z}_+)]+E_{\boldsymbol{x}\sim p_{\mathrm{data}}}[\log(1-D(En(\boldsymbol{x})))].$$

其中，$p_{\boldsymbol{z}_+}$ 为预训练的 AE-CNN 的隐空间分布；$p_{\mathrm{data}}$ 为训练数据分布. 通过对抗训练即可将 $p_{\boldsymbol{z}_+}$ 施加至 AE-RNN 的隐空间，得到一个融合多种草图信息的隐空间.

### 2.5 模型训练

本文模型的训练分 2 个部分交替进行，一部分是 AE-RNN 的重构训练；另一部分是对抗训练. 当模型训练时，在每个 mini-batch 上，首先输入矢量化的草图，利用最小化重构损失项 $L_{\mathrm{R}}$ 训练 AE-RNN；然后将像素表示的草图输入预训练的 AE-CNN 的编码器中得到正样本，与 AE-RNN 中得到的负样本一起利用对抗损失项训练生成器和判别器. 图 4 中的虚线箭头展示了获取对抗网络的正样本和负样本的流程.

在重构训练阶段，训练 AE-RNN 的编码器，将矢量草图所具有的信息编码到隐空间. 而在对抗训练阶段，训练 AE-RNN 的编码器(此时为生成器)来拟合像素草图的隐空间分布(即 AE-CNN 的隐空间分布). 利用上述训练过程能够得到一个融合了 2 种不同草图表示形式的信息的隐空间，而引入像素草图所具有的空间形状信息有利于改善重构结果出现的潦草效应.

### 2.6 草图生成

当模型训练完成后，在采样阶段，仅仅使用训练完成的 AE-RNN 来生成草图. 此时，模型已经将 2 种不同草图表示形式所具有的信息融合进了 AE-RNN 的隐空间，从该隐空间采样得来的隐向量可由解码器解码得到生成的草图.

在采样的过程中，解码器的每个时间步都会生成 2 部分参数：用来建模点坐标偏移量的 GMM 的参数和用来建模笔画状态的类别分布的参数. 利用上述 2 个分布，可以采样得出当前时间步的笔画 $\boldsymbol{S}_i'$. 不同于训练过程的是，采样过程直接将当前时间步的笔画 $\boldsymbol{S}_i'$ 作为下一个时间步的输入. 按此步骤一直采样下去，直到 $p_3=1$ 或者 $i=N_{\max}$.

与编码器一样，解码器的输出也不是确定的，而是一个以输入隐向量 $\boldsymbol{z}$ 为条件的随机序列. 为了控制采样结果的随机性，模型引入了一个随机程度参数 $\tau$，通过

$$\hat{q}_k\to\frac{\hat{q}_k}{\tau},\ \hat{\Pi}_k\to\frac{\hat{\Pi}_k}{\tau},\ \sigma_{\mathrm{x}}^2\to\sigma_{\mathrm{x}}^2\tau,\ \sigma_{\mathrm{y}}^2\to\sigma_{\mathrm{y}}^2\tau$$

来调节采样分布. 其中，$\hat{q}_k$ 和 $\hat{\Pi}_k$ 分别为解码器某

一时间步生成的 $q_k$ 和 $\Pi_k$. 通过使用随机程度参数 $\tau$, 可以缩放类别分布的 Softmax 参数以及二元正态分布的参数 $\sigma$, 达到控制采样结果随机性的目的. $\tau$ 的取值范围为(0,1), 当 $\tau \to 0$ 时, 采样的随机性变小, 模型变得确定, 每次采样的结果变得相似.

## 3 实　验

### 3.1 实验设置

本文实验环境使用的是 TensorFlow 深度学习框架, 所有模型均是在单张 NVIDIA GeForce GTX 1080Ti 显卡上训练完成. 实验使用的训练集为 QuickDraw, 本文选择其中的 cat, fire truck, mosquito 和 yoga pose 4 个类进行训练, 它们分别代表了动物、物体、昆虫和人类, 一定程度上捕捉了数据集的多样性.

本文实验分为定性实验和定量实验. 其中, 定量实验中所有模型使用的随机程度参数均为 $\tau$=0.10, $M$=20, 训练轮数均为 1 000 000, 隐向量维度为 128, 批次(batch)的大小为 100. 本文使用了梯度衰减技术来稳定训练, 初始学习率为 0.000 10, 以每步 0.999 90 的概率逐步衰减至最小学习率 0.000 01. 当进行对抗训练时, 判别器每更新 1 次, 生成器更新 2 次, 对抗学习率为 0.000 10. 为了防止出现过拟合, 本文在 LSTM 中采用了 dropout 技术, 此外在训练中, 以一定的概率丢弃训练样本每一笔画中的数据点, 以此来达到数据增强的效果.

### 3.2 评测指标

在 Ske-score 度量指标[16]之前, 没有量化评测指标来度量矢量草图的好坏. 由于已有的矢量草图生成模型的生成结果容易出现潦草效应, Varshaneya 等[16]提出了 Ske-score 度量指标, 希望通过使用该指标能够对矢量草图上出现的潦草效应进行量化, 从而一定程度上能够度量生成的矢量草图的好坏. 本文采用了 Ske-score 作为定量的评测指标. 一幅矢量草图的 Ske-score 得分 $C$, 一个模型的 Ske-score 得分 $C_{\mathrm{M}}$, 一个数据集的 Ske-score 得分 $C_{\mathrm{D}}$, 它们定义分别为

$$C = \frac{N_{\mathrm{lift}}}{N_{\mathrm{touch}}},$$

$$C_{\mathrm{M}} = \frac{1}{N_{\mathrm{M}}}\sum_{i=1}^{N_{\mathrm{M}}} C_i,$$

$$C_{\mathrm{D}} = \frac{1}{N_{\mathrm{D}}}\sum_{j=1}^{N_{\mathrm{D}}} C_j.$$

其中, $N_{\mathrm{lift}}$ 为有效笔画中画笔离开纸面的笔画数; $N_{\mathrm{touch}}$ 为有效笔画中画笔触碰纸面的笔画数; $C_i$ 为模型生成的第 $i$ 幅草图的 Ske-score 得分; $N_{\mathrm{M}}$ 为模型生成草图的数量; $C_j$ 为数据集中第 $j$ 幅草图的 Ske-score 得分; $N_{\mathrm{D}}$ 为数据集中包含的草图数量. Ske-score 分值反映了绘画过程中画笔离开纸面的频率, 高分值意味着画笔离开纸面的次数更多. 当 $|C_{\mathrm{D}} - C_{\mathrm{M}}| < \varepsilon$ 时, 可以认为模型 $M$ 是一个可以生成无潦草效应草图的“好”模型.

### 3.3 草图重构实验

本文通过使用训练完成的模型来对测试集的样本进行重构, 定性地评价模型的好坏. 部分重构结果如图 6 所示, 图 6a 和图 6b 分别为 sketch-AAE 在 cat 类和 fire truck 类上的重构结果. 所使用的模型均是在单个类上训练完成, 重构结果从左到右使用的 $\tau$ 值依次为 0.01, 0.10, 0.30, 0.50, 0.70, 1.00.

从图 6 中的每一行都可以看出, 从左到右随着 $\tau$ 值的增加, 重构结果的自由度越来越大. 图 6a 和图 6b 中的每一行都框出了一个较好的重构结果. 从重构结果可以发现, 本文方法具有一定程度的部件修复功能, 即能添加一些输入缺失的该类别常见的部件. 例如, 图 6a 中第 1 行的部分重构结果补齐了输入中缺少的猫脸的左半边胡须, 第 3 行重构结果均添加了输入缺失的一只猫眼睛; 图 6b 中第 1 行部分重构结果添加了输入中缺失的车轮, 第 2 行部分重构结果补齐了输入中缺失的车轮, 以及第 3 行部分重构结果添加了 fire truck 中常见的梯子. 这表明 sketch-AAE 较好地学习到了训练类别的特征, 归纳出该类别物体常见的部件特征, 并在重构时添加在物体适当的位置, 这一特性具有很好的应用前景, 也验证了本文所提方法的有效性. 在草图识别以及基于草图的图像检索场景中, 常常会遇到用户提交的一些所谓“画坏”的绘画结果, 如果直接输入这些样本, 势必会大大影响识别或检索的精度. 此时, 可以先将用户提供的样本送入训练完成的 sketch-AAE 中重构, 将缺失的部件补齐, 再进行识别和检索, 这将有助于提高识别和检索的精度.

### 3.4 隐空间插值实验

为了进一步验证 sketch-AAE 的有效性, 本文对训练完成的模型进行隐空间插值的实验. 通过 AE-RNN 的解码器, 将 2 个隐向量之间的插值结果解码, 可以观察到一幅草图是如何逐渐变化成另一幅草图的. 同时, 转化过程本身的平滑性以及一致

性也在一定程度上能反映出隐空间的严密程度, 即隐空间分布是否存在“漏洞”, 进而反映出实验模型的好坏. 使用的线性插值方法为 $z = w_1 z_1 + w_2 z_2$, 其中 $w_1$ 和 $w_2$ 为插值权重; $z_1$ 和 $z_2$ 为 2 个不同的隐变量; $z$ 是插值结果. 插值权重满足 $w_1 + w_2 = 1$, $w_1$ 的值从 1 开始以步长 0.1 减至 0; $w_2$ 反之, 所以每行有 11 幅解码后的草图. 图 7 为线性插值实验结果示例, 所有插值实验使用的 $\tau$ 值均为 0.1.

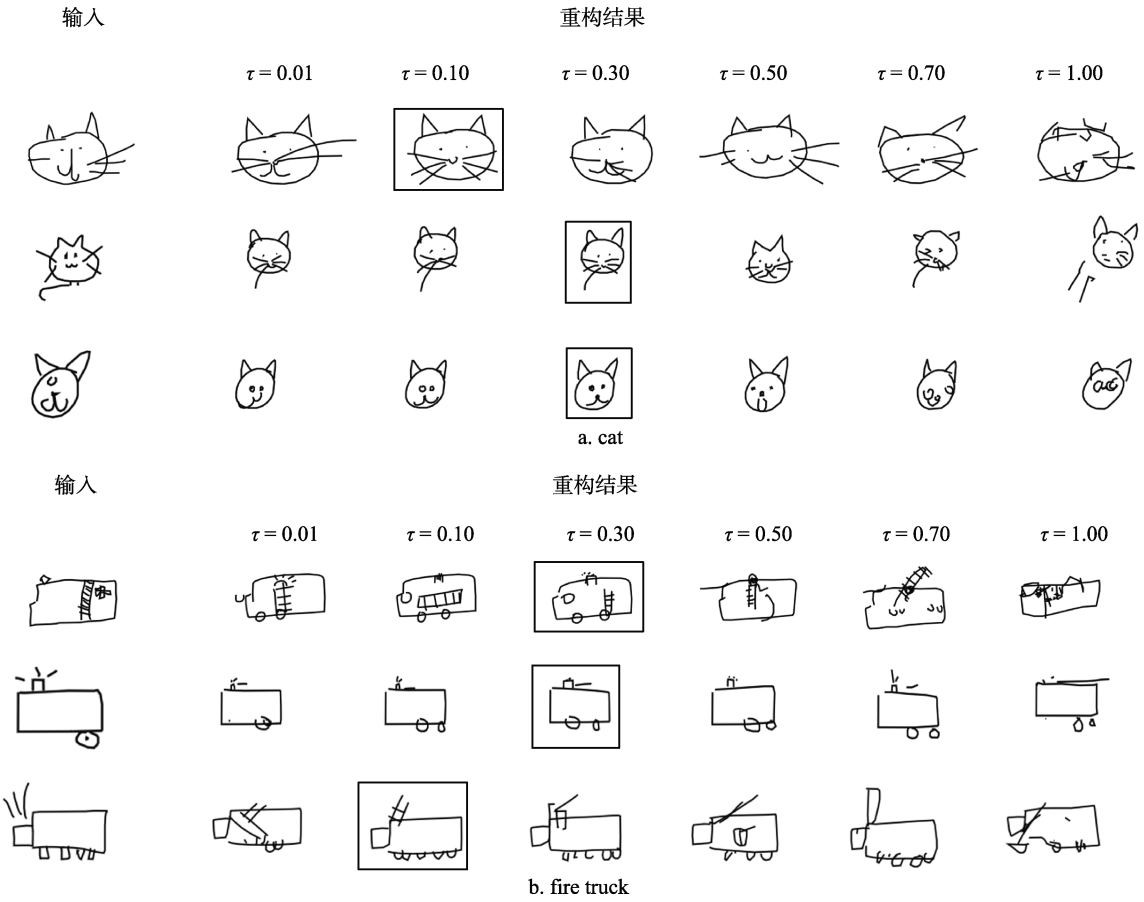



图 6 sketch-AAE 草图重构结果示例

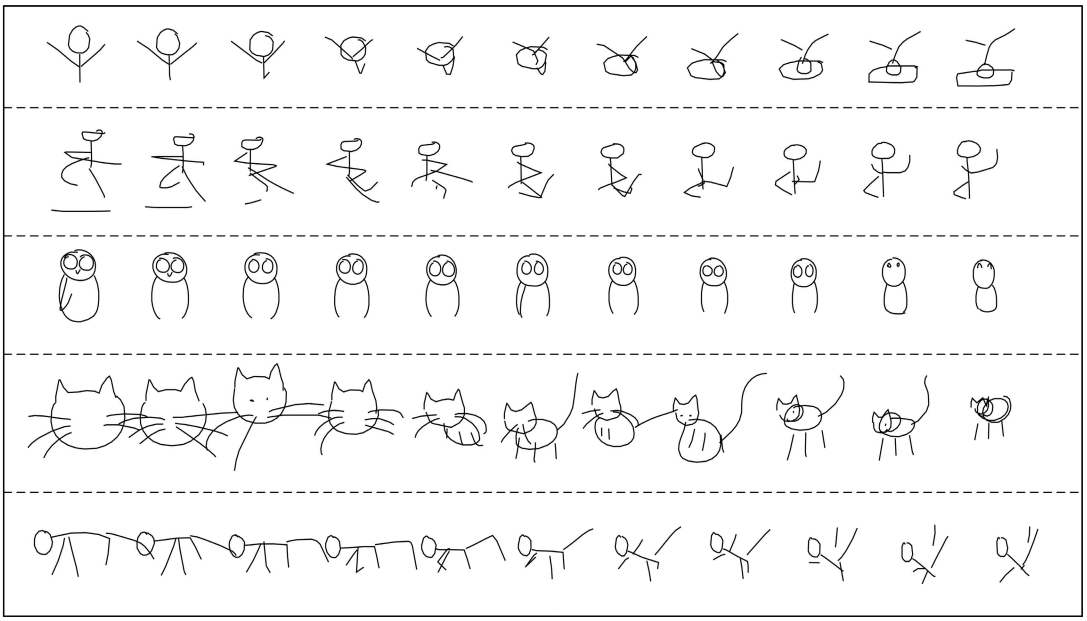


图 7 隐空间线性插值示例

从图 7 中可以观察到包括瑜伽动作、身体转向、猫、猫头鹰身体以及眼睛的变化都具有较好的平滑性, 说明模型具有一个严密的隐空间. 通过探索模型的隐空间, 可以发现很多不同草图物体之间的有趣交互以及关系, 这种探索也可以帮助艺术家或者设计师获取更多的创作灵感.

### 3.5　草图生成模型对比实验

为了验证本文方法的有效性, 利用度量指标 Ske-score 在 2 种主流的草图生成 (sketch-RNN[13] 和 sketch-pix2seq[14])和方法本文方法 sketch-AAE 及其变形方法 sketch-AAE-G 进行了对比实验. 其中, sketch-AAE-G 是将 sketch-AAE 中的对抗网络向 AE 模块隐空间施加的像素草图的分布替换为标准高斯分布 $\mathcal{N}(0,\mathrm{I})$. 表 1 所示为 QuickDraw 数据集中测试样本 cat, fire truck, mosquito 和 yoga pose 的 Ske-score 得分情况, 表 2 所示为各模型在 4 个类上的得分情况.

**表 1　各测试样本的 Ske-score 得分**

| 测试样本 | 得分 |
|---|---|
| cat | 0.18±0.05 |
| fire truck | 0.12±0.04 |
| mosquito | 0.16±0.05 |
| yoga pose | 0.15±0.07 |

**表 2　各模型的 Ske-score 得分对比**

| 模型 | cat | fire truck | mosquito | yoga pose |
|---|---|---|---|---|
| sketch-RNN[13] | 0.16±0.01 | 0.09±0.01 | 0.13±0.03 | 0.12±0.01 |
| sketch-pix2seq[14] | 0.15±0.02 | 0.09±0.01 | 0.12±0.02 | 0.12±0.01 |
| sketch-AAE-G | 0.18±0.05 | **0.13±0.05** | 0.15±0.02 | 0.16±0.03 |
| sketch-AAE | **0.19±0.06** | **0.13±0.05** | **0.16±0.05** | **0.18±0.06** |

注. 粗体表示最优值.

由 sketch-RNN 和 sketch-AAE-G 的得分对比可以得出, AAE 框架引入的对抗机制有利于改善 AE 在矢量草图生成上出现的潦草效应, 表明使用可学习的判别网络施加先验分布比使用 KL 距离施加先验分布更有效. 结合 2 个表的得分情况可以看出, 模型 sketch-AAE 在各个类上生成结果的得分都与原数据集中草图的得分非常相近, 个别类还出现了高于原数据集得分的情况, 表明 sketch-AAE 方法是一个有效的方法. 由 sketch-AAE-G 和 sketch-AAE 的得分对比情况可以得出, 在模型中引入像素化草图包含的信息同样有利于改善生成结果出现的潦草效应, 这是由于像素表示形式草图所包含的物体形状信息对矢量草图的生成过程起到一定的指导作用, 使得画出的物体更具有类别可辨识性.

## 4　结　语

现有的基于 AAE 的矢量草图生成方法容易出现潦草效应, 本文提出一种基于 AAE 框架的矢量草图生成方法——sketch-AAE, 它没有使用 KL 距离惩罚, 而是通过对抗机制将先验分布施加至 AE 的隐空间. 本文方法利用 2 种不同表示形式的草图信息, 通过对抗网络将像素表示形式的草图信息融合进矢量草图的生成过程中, 实验验证了融合像素草图的信息有助于改善生成结果的潦草效应.

本文未来的工作方向之一是将 sketch-AAE 方法扩展至多类的矢量草图生成上, 以节省训练成本; 进而将一些更加贴近真实的草图数据集(如 TU-Berlin[1])中包含的信息编码至 sketch-AAE 模型中, 用以指导矢量草图的生成.

此外, 本文方法也具有较好的应用前景. 例如, 可以将 sketch-AAE 应用于草图修复的场景中, 利用其修复功能重构出大量相似并符合美学观点的草图作品.

## 参考文献(References):


[1] Eitz M, Hays J, Alexa M. How do humans sketch objects?[J]. ACM Transactions on Graphics, 2012, 31(4): Article No.44

[2] Yu Q, Yang Y X, Liu F, *et al.* Sketch-a-net: a deep neural network that beats humans[J]. International Journal of Computer Vision, 2017, 122(3): 411-425

[3] Zhang J H, Chen Y L, Li L, *et al.* Context-based sketch classification[C] //Proceedings of the Joint Symposium on Computational Aesthetics and Sketch-Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering. New York: ACM Press, 2018: Article No.1

[4] Sun Z B, Wang C H, Zhang L Q, *et al.* Free hand-drawn sketch segmentation[C] //Proceedings of the 12th European Conference on Computer Vision. Heidelberg: Springer, 2012: 626-639

[5] Huang Z, Fu H B, Lau R W H. Data-driven segmentation and labeling of freehand sketches[J]. ACM Transactions on Graphics, 2014, 33(6): Article No.175

[6] Li K, Pang K Y, Song Y Z, *et al.* Towards deep universal sketch perceptual grouper[J]. IEEE Transactions on Image Processing, 2019, 28(7): 3219-3231

[7] Li L, Fu H B, Tai C L. Fast sketch segmentation and labeling with deep learning[J]. IEEE Computer Graphics and Applications, 2019, 39(2): 38-51

[8] Eitz M, Richter R, Boubekeur T, *et al.* Sketch-based shape retrieval[J]. ACM Transactions on Graphics, 2012, 31(4): Article

No.31
[9] Wang F, Kang L, Li Y. Sketch-based 3D shape retrieval using Convolutional Neural Networks[C] //Proceedings of the 34th IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2015: 1875-1883
[10] Sangkloy P, Burnell N, Ham C, *et al.* The sketchy database: learning to retrieve badly drawn bunnies[J]. ACM Transactions on Graphics, 2016, 35(4): Article No.119
[11] Xu P, Huang Y Y, Yuan T T, *et al.* SketchMate: deep hashing for million-scale human sketch retrieval[C] //Proceedings of the 36th IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 8090-8098
[12] Olsen L, Samavati F F, Sousa M C, *et al.* Sketch-based modeling: a survey[J]. Computers & Graphics, 2009, 33(1): 85-103
[13] Ha D, Eck D. A neural representation of sketch drawings[OL]. [2020-02-27]. https://arxiv.org/abs/1704.03477
[14] Chen Y J, Tu S K, Yi Y Q, *et al.* Sketch-pix2seq: a model to generate sketches of multiple categories[OL]. [2020-02-27]. https://arxiv.org/abs/1709.04121
[15] Kingma D P, Welling M. Auto-encoding variational bayes[OL]. [2020-02-27]. https://arxiv.org/abs/1312.6114
[16] Varshaneya V, Balasubramanian S, Balasubramanian V N. Teaching GANs to sketch in vector format[OL]. [2020-02-27]. https://arxiv.org/abs/1904.03620
[17] Goodfellow I J, Pouget-Abadie J, Mirza M, *et al.* Generative adversarial nets[C] //Proceedings of Advances in Neural Information Processing Systems. Cambridge: MIT Press, 2014: 2672-2680
[18] Makhzani A, Shlens J, Jaitly N, *et al.* Adversarial autoencoders[OL]. [2020-02-27]. https://arxiv.org/abs/1511. 05644
[19] Schuster M, Paliwal K K. Bidirectional recurrent neural networks[J]. IEEE Transactions on Signal Processing, 1997, 45(11): 2673-2681
[20] Ha D. Recurrent net dreams up fake Chinese characters in vector format with TensorFlow[OL]. [2020-02-27]. http://blog.otoro.net/2015/12/28/recurrent-net-dreams-up-fake-chinese-characters-in-vector-format-with-tensorflow/
[21] Zhong K. Learning to draw vector graphics: applying generative modeling to font glyphs[D]. Cambridge: Massachusetts Institute of Technology, 2018

**(上接第 1956 页)**

[14] Bianconi F, González E, Fernández A. Dominant local binary patterns for texture classification: Labelled or unlabelled?[J]. Pattern Recognition Letters, 2015, 65: 8-14
[15] Liu L, Fieguth P, Guo Y L, *et al*. Local binary features for texture classification: taxonomy and experimental study[J]. Pattern Recognition, 2017, 62: 135-160
[16] Song Kechen, Yan Yunhui, Chen Wenhui, *et al*. Research and perspective on local binary pattern[J]. Acta Automatica Sinica, 2013, 39(6): 730-744(in Chinese)
(宋克臣, 颜云辉, 陈文辉, 等. 局部二值模式方法研究与展望[J]. 自动化学报, 2013, 39(6): 730-744)
[17] Fan Yangyu, Wang Junmin, Yu Jianming. An efficient texture classification algorithm with illumination, rotation and scale invariance[J]. Journal of Computer-Aided Design & Computer Graphics, 2017, 29(11): 1989-1996(in Chinese)
(樊养余, 王军敏, 余建明. 高效的光照、旋转、尺度不变纹理分类算法[J]. 计算机辅助设计与图形学学报, 2017, 29(11): 1989-1996)
[18] Guo Z H, Zhang L, Zhang D. Rotation invariant texture classification using LBP variance (LBPV) with global matching[J]. Pattern Recognition, 2010, 43(3): 706-719
[19] Liu L, Zhao L J, Long Y L, *et al*. Extended local binary patterns for texture classification[J]. Image and Vision Computing, 2012, 30(2): 86-99
[20] Hong X P, Zhao G Y, Pietikäinen M, *et al*. Combining LBP difference and feature correlation for texture description[J]. IEEE Transactions on Image Processing, 2014, 23(6): 2557-2568
[21] Hafiane A, Palaniappan K, Seetharaman G. Joint adaptive median binary patterns for texture classification[J]. Pattern Recognition, 2015, 48(8): 2609-2620
[22] Liu L, Lao S Y, Fieguth P W, *et al*. Median robust extended local binary pattern for texture classification[J]. IEEE Transactions on Image Processing, 2016, 25(3): 1368-1381
[23] Pan Z B, Li Z Y, Fan H C, *et al*. Feature based local binary pattern for rotation invariant texture classification[J]. Expert Systems with Applications, 2017, 88: 238-248
[24] Wang K, Bichot C E, Li Y, *et al*. Local binary circumferential and radial derivative pattern for texture classification[J]. Pattern Recognition, 2017, 67: 213-229
[25] Depeursinge A, Püspöki Z, Ward J P, *et al*. Steerable Wavelet MACHINES (SWM): Learning moving frames for texture classification[J]. IEEE Transactions on Image Processing, 2017, 26(4): 1626-1636
[26] Ei merabet Y, Ruichek Y. Local concave-and-convex micro-structure patterns for texture classification[J]. Pattern Recognition, 2018, 76: 303-322
[27] Kou Q Q, Cheng D Q, Chen L L, *et al*. A multiresolution gray-scale and rotation invariant descriptor for texture classification[J]. IEEE Access, 2018, 6: 30691-30701