

硕 士 研 究 生 读 书 报 告



题目 关于《实时数据驱动的交互式草

图描边》的读书报告

作者姓名 陶嘉恒

作者学号 21851075

授课教师 李启雷

学科专业 人工智能

所在学院 软件学院

提交日期 2018-12-17

Reading Report on *Real-time data-driven interactive rough sketch inking*

A Dissertation Submitted to

Zhejiang University

in partial fulfillment of the requirements for

the degree of

Master of Engineering

Major Subject: Software Engineering

Advisor: Li QiLei

By

Tao Jiaheng

Zhejiang University, P.R. China

2018

摘要

这篇读书报告阅读的内容是关于SIGGRAPH 2018年8月“草图”类别的论文《实时数据驱动的交互式草图描边》。

设计人员将草图绘制出来，通常草图被送到客户手中确认，或取得修改建议。一旦草图方案敲定，设计就转交至下一个流程——描边生成线稿，人物上色。论文研究的是草图描边的过程。

在草图描边过程中难点是从草图得到确切的、简明、连贯的线条，以及线条组成的线图。现有计算机辅助设计能够大体上生成的一副线图，但是计算机缺乏对图片的理解，容易丢失线条和细节、残留草图中勾勒的阴影，需要设计师参与确认和二次修改，这些修改工作主要是线条的闭合和多余线条的擦除。论文提出的交互式线条着墨方法，叫做“Smart Inker”，能够根据粗略的草图，帮助设计师在二次修改的时候，高效且自然地连接线条、擦除阴影、微调输出的线条。该方法仅通过鼠标，而不需要其他特殊的插图硬件，并且修改能够在生成线图的旁边能够实时显示。

论文的方法是数据驱动的，制作的辅助绘图工具基于全卷积网络，训练利用用户的编辑和不准确的草图来产生准确的线条图，允许高性能的交互式实时编辑各种具有挑战性的草图图像。文中为了训练工具开发了两个关键技术:一是通过模糊模拟和用户快速的编辑创建训练数据;另一种是从向量数据学习的线条归一化。论文通过深入的用户研究验证其方法，并与专业的插图软件比较，得出结论：论文中方法能够以1.8×的因子减少描边时间，同时提高业余用户的使用效果。

**关键词**：图像处理，以人为中心的计算，手势输入，应用计算，美术，草图，描边，线条画

Abstract

This reading report reads the paper, *Real-time data-driven interactive rough sketch inking*, of ”Sketching” category on SIGGRAPH conference in August 2018.

The designer draws the sketch, usually the sketch is sent to the customer for confirmation, or a modification suggestion. Once the sketch plan is finalized, the design is handed over to the next process—the strokes are generated and the characters are colored. The paper studies the process of sketch strokes.

The difficulty in the sketch stroke process is to get exact, concise, consistent lines from the sketch, and a line graph of the lines. Existing computer-aided design can generally generate a line drawing, but the computer lacks understanding of the picture, easily loses lines and details, and the shadows outlined in the residual sketch, requiring the designer to participate in the confirmation and secondary modification. These modifications are mainly the closing of the lines and the erasure of the extra lines. The interactive line inking method proposed by the paper, called “Smart Inker”, helps designers to efficiently and naturally connect lines, erase shadows, and fine-tune the output lines according to rough sketches. This method only uses the mouse, and does not require other special illustration hardware, and the modification can be displayed in real time next to the generated line graph.

The paper’s approach is data-driven: the tools are based on fully convolutional networks, which is trained to exploit both the user edits and inaccurate rough sketch to produce accurate line drawings, allowing high-performance interactive editing in real-time on a variety of challenging rough sketch images. For the training of the tools, the parper developed two key techniques: one is the creation of training data by simulation of vague and quick user edits; the other is a line normalization based on learning from vector data. These techniques, in combination with our sketch-specific data augmentation, allow us to train the tools on heterogeneous data without actual user interaction.

The approach is validated with an in-depth user study, comparing with professional illustration software. It shows that the approach is able to reduce inking time by a factor of 1.8×, while improving the results of amateur users.

**Keywords**: Image manipulation, Human-centered computing, Gestural input, Applied computing, Fine arts, sketching, inking, line drawing

# 一、背景与问题描述

尽管电脑在插画中的作用在过去十年中有了显著的增长，铅笔画素描仍然是大多数插画家头脑风暴或开始新项目的主要方式。一旦粗略的草图完成，通常会用数字或钢笔墨水绘制出一幅线稿图，该线稿图可以上色或直接使用。线稿必须尊重铅笔作者的原意，随时调整任何明显的错误。同时，线稿决定了成品的外观。

数字描边通常需要重新绘制所有的线，同时修复错误或调整细节，以获得数字线稿。这个过程通常是通过使用昂贵的数字笔板来完成的，并且需要集中精力才能画出精确的线条。

# 二、论文方法

论文提出了一个数字描边的方法，它允许半自动实时交互式创建线画。该方法由三个“智能”工具组成：墨笔、墨刷和智能橡皮，它们对用户的输入作出直观的反应，同时在输入草图的原型上，有效地、自然地连接线、擦除阴影和微调画线输出。这种方法仅用鼠标，而不需要精确的笔触或其他硬件，允许业余爱好者也能快速准确地绘制复杂场景完成线稿，比传统方法平均快1.8倍。

不同工具的概述如图1所示，在图1中可以看到，不准确的编辑也可以用于有效的描边。

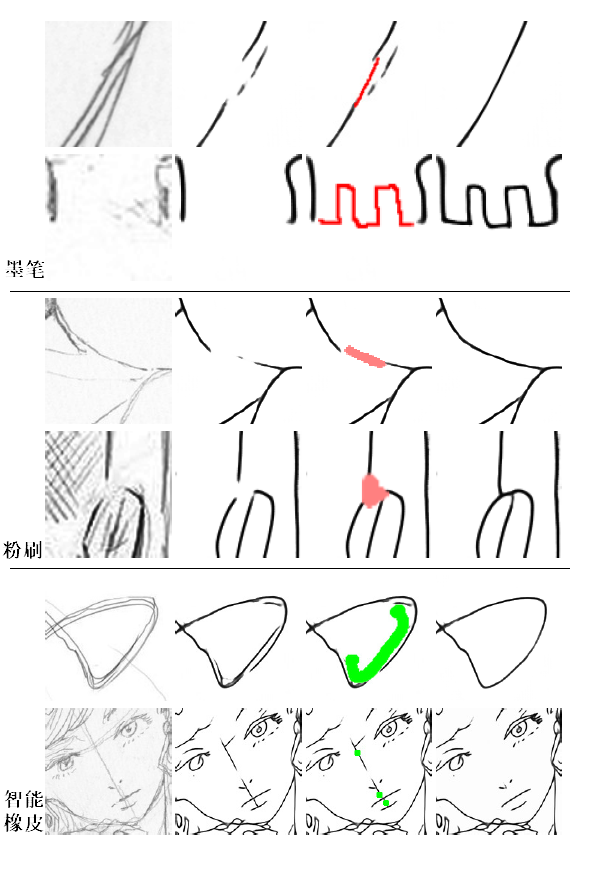


图1 论文提出的三个工具的示例。前两行描述墨笔工具(暗红色)。中间两行描述的是墨刷工具(淡红色)，最后两行描述的是智能橡皮擦工具(绿色)。

这些工具以数据驱动的方式设计，通过训练数据和模拟用户编辑进行训练，而无需显式地硬编码用户的行为。它们都基于全卷积网络，接受两种输入:粗略的草图和用户输入。将两者结合起来的能力类似于最近提出的用于图像着色的交互式神经网络方法[1-2]。用户可以通过添加更多的用户提示来迭代调整输出。在对灰度图像和彩色图像进行训练时，通过对真值图像颜色进行采样，自动生成用户提示。虽然这些简单的提示可以很好地用于信息密集的交互式图像分色，但不适用于粗糙草图和线条图等稀疏输入。此外，描边比图像着色中使用的提示需要更精确的用户编辑。由于这些局限性，论文提出了一种新的用户编辑仿真方法，如图2所示。

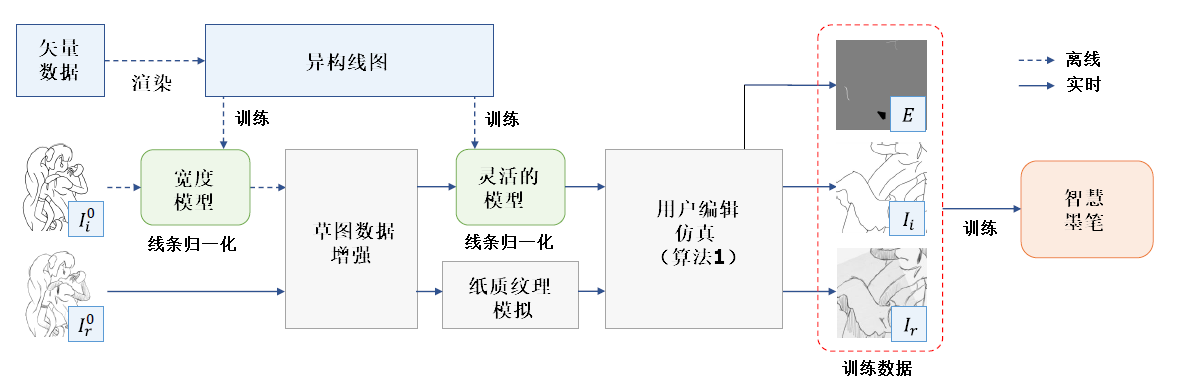


图2基于模拟用户编辑和修改输入粗略草图图像来训练的数据驱动的墨笔工具的方法概述

为了训练网络，需要两种新的技术来创建训练数据。一种是模糊快速用户编辑的仿真，另一种是线宽归一化，它允许使用更大的异构数据集对模型进行训练，其中的线宽在样本间不一致。对于后者，论文开发了一个框架，直接从使用神经网络的数据学习线条归一化操作。与标准的形态学操作不同的是，论文的方法不需要将图像转换为二进制图像预处理，就可以对图像进行处理，并输出平滑的反锯齿图形。论文证明了，对于复杂的线条艺术，该方法得到了比标准形态逻辑操作更好的性能。其中学习的线条归一化操作在网络的训练中起着基础性的作用，同样的框架也可以在后续处理步骤中用来调整线稿的线宽。

论文的模型也受到最近发展的素描简化的启发，其中使用一个编码-解码架构的全卷积网络[3]能够快速、准确地简化复杂的草图。允许交互式编辑修改模型,并简化其设计,以减少原始模型的37%的计算时间。

使用**神经网络**创建线稿图的一个常见问题是输出模糊。使用论文的**线条归一化框架**，结合一个简单的加权L1损失，不需要后处理技术就可以得到清晰的结果[3]，也不需要复杂的优化框架[4]，就可以得到清晰的结果。这对于交互式墨笔设置非常关键，因为它允许更自然和直接的交互。

综上，论文主要的贡献在于：（1）一种描边方法，允许使用直观的数据驱动的辅助工具实现高分辨率的粗略草图的实时交互式编辑，而无需后期处理。（2）通过模拟用户编辑和修改输入的粗略草图图像来训练这些工具的框架。（3）用于异构线条草图训练的线宽归一化技术。（4）研究表明，业余用户能够在改善结果的同时，将草图绘制速度提高1.8倍。

# 算法研读

## 3.1 线条宽度归一化框架

线条宽度标准化的标准方法是基于形态学算子。形态学算子对二进制图像进行操作，但大多数应用程序通常不需要二进制图像输出。因此，论文采取直接从大量向量数据中学习启发式，使用非常简单的卷积神经网络。

论文的线条宽度归一化框架包括使用向量数据和操纵线宽来创建输入和输出训练对，这些训练对适合于卷积神经网络的训练。特别地，论文重点讨论了线归一化的使用情况，在这种情况下，目标输出图像的线宽对于所有笔画都是固定的。

### 3.1.1 基本的形态学算子

形态学算子的主要思想是用一定形状的结构元素在图像中抽取出相应的某些结构,通常可以用于图像的滤波、分割、分类等处理。形态学算子有腐蚀、膨胀、开和闭四种。

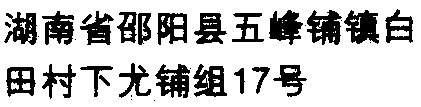


图 1 原图

腐蚀是一种消除边界点，使边界向内部收缩的过程。可以用来消除小且无意义的物体。腐蚀的算法： 用3x3的结构元素，扫描图像的每一个像素 用结构元素与其覆盖的二值图像做“与”操作 如果都为1，结果图像的该像素为1。否则为0。结果：使二值图像减小一圈。

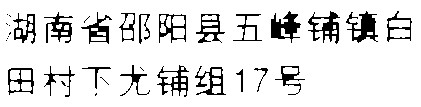


图 2 腐蚀结果图

膨胀是将与物体接触的所有背景点合并到该物体中，使边界向外部扩张的过程。可以用来填补物体中的空洞。膨胀的算法： 用3x3的结构元素，扫描图像的每一个像素 用结构元素与其覆盖的二值图像做“与”操作 如果都为0，结果图像的该像素为0。否则为1。结果：使二值图像扩大一圈。

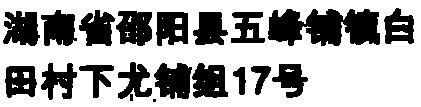


图 3 膨胀结果图

虽然腐蚀可以将粘连的目标进行分离，膨胀可以将断续的目标进行连接，但是无论是腐蚀还是膨胀处理后，目标的尺寸都会产生变化。开运算，闭运算就是为了解决这个问题。

先腐蚀后膨胀的过程称为开运算。用来消除小物体、在纤细点处分离物体、平滑较大物体的边界的同时并不明显改变其面积。

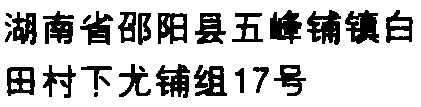


图 4 开运算结果图

先膨胀后腐蚀的过程称为闭运算。用来填充物体内细小空洞、连接邻近物体、平滑其边界的同时并不明显改变其面积。

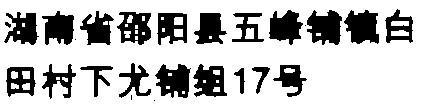


图 5 闭运算结果图

### 3.1.2 模型

论文提出了两个完全卷积网络模型：一个9层的宽泛模型，还有1个敏捷的5层模型。宽模型是为离线预处理大数据集而设计的，而灵活模型是为了在训练过程中作为每个草图数据增强的一部分而设计的。除了每个模型的最后一层，对所有层都使用批处理规范化[5]，具有修正的线性单元传递函数[6]。对于每个模型的最后一层，使用一个Sigmoid传递函数来确保输出在[0,1]范围内。

宽泛模型基于一个9层的全卷积网络。第一层使用9×9像素内核,而其他的层使用3×3像素的内核。在整个网络中使用的卷积滤波器的数量是恒定的，并且设置为64。这使得网络从25×25像素区域的输入计算每个输出像素。

敏捷模型采用了类似的方法，尽管它被简化为5层，并且在整个网络中仅使用32个卷积过滤器。与广泛的模型相同,第一层使用9×9像素内核其余使用3×3像素的内核。模型从17×17像素区域的输入计算每个输出像素。

### 3.1.3 训练

论文使用混合数据源对这些模型进行培训。论文使用混合数据源对这些模型进行培训。特别地，论文使用了TUD-Berlin数据集、QuickDraw数据集、KanjiVG数据集和合成形状。

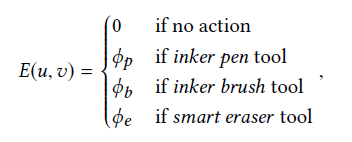
对于所有的数据集，将所有的矢量图对齐到左上角，并将它们统一缩放到最大255。以1像素间距重新采样所有的笔画，然后使用ε=2.0的拉默-道格拉斯-皮科克算法简化他们[7]。

论文从不同频率的数据集中取样。特别是，与其他数据集相比，使用合成形状的可能性仅为1/3，其他数据集使用的概率相同。论文不使用每个训练样本的单一绘图，而是根据每个数据集中向量图像的细节级别使用一个可变的数字，通过对图像平均复杂度的直观检查来启发式地确定该数字。在光栅图像的训练中，它们也被随机旋转和定位。

训练采用随机梯度下降的变体ADADELTA[8]，对所有模型进行训练，使线条笔画归一化为2个像素的恒定宽度。论文训练宽泛模型时，输入的线的厚度在0.5到10像素之间，而敏捷模型的厚度在0.5到3像素之间。输入生成补丁都是512×512像素,论文批量训练大小为8。论文使用早期停止用于避免过度拟合,并训练了20,000次迭代。

## 3.2 交互式草图描边

论文定义了三种不同的基本编辑工具墨水笔、墨水笔和智能橡皮擦。墨笔是用于描边线条的，墨刷是用于连接不完整的线条，而智能橡皮擦则是用来擦除不必要的线条。与标准工具不同，这些工具将使用数据进行训练，以便生成清晰的线条图。定义每个“像素”E将代表特定工具或不采取行动:



与使用标准的插图工具作为后处理阶段不同，论文将训练墨水模型学会在学习过程中合并这些工具，允许模型以“智能”的方式调整输出以匹配用户的编辑。

论文考虑的使用场景中，用户首先以全自动方式在草图上描边，然后迭代和交互地纠正模型的输出。

### 3.2.1 智能墨笔模型

论文的智能墨笔模型由一个全卷积编码器-解码器网络组成，它有24层，分三个阶段将分辨率降低到原来大小的1/8。然后，通过亚像素卷积分三个阶段将输出恢复到原来的大小[9]，每个阶段的分辨率都可以翻倍。按照卷积网络的标准做法，在每一个卷积层之后，论文应用批处理归一化(BN)[10]然后是修正的线性单元(ReLU)传递函数[11]。

卷积层使用3×3像素内核与1×1像素0值填充内部决议常数。作为例外,第一层使用9×9像素与反射4×4像素填充内核,以避免输入用0填充引起的伪造感。在第一个亚像素卷积层之前，论文也使用了spatial dropout[13]，它是dropout[14]的一个扩展，在这个扩展中，整个特征映射被随机设置为0，而不是随机取值的像素。这避免了使用附近的像素值来填充0值像素，因为相邻值之间的相关性很大。

### 3.2.2 模拟用户编辑

训练智能油墨最重要的一个方面是模拟真实的用户编辑。与传统软件工具不同，传统软件工具使用命令式设计方法，在这种方法中，每个工具都是显式定义的，在这里，论文必须定义任务的输入和输出，并训练模型学习从输入到输出的映射。特别地，论文模拟了三个不同工具的编辑，以鼓励模型在绘制草图时重视它们。在仿真过程中，不仅生成了编辑映射，还对输入的草图进行了操作和编辑。模拟编辑的整个算法的概述详见算法1，模拟编辑实例如图6所示。

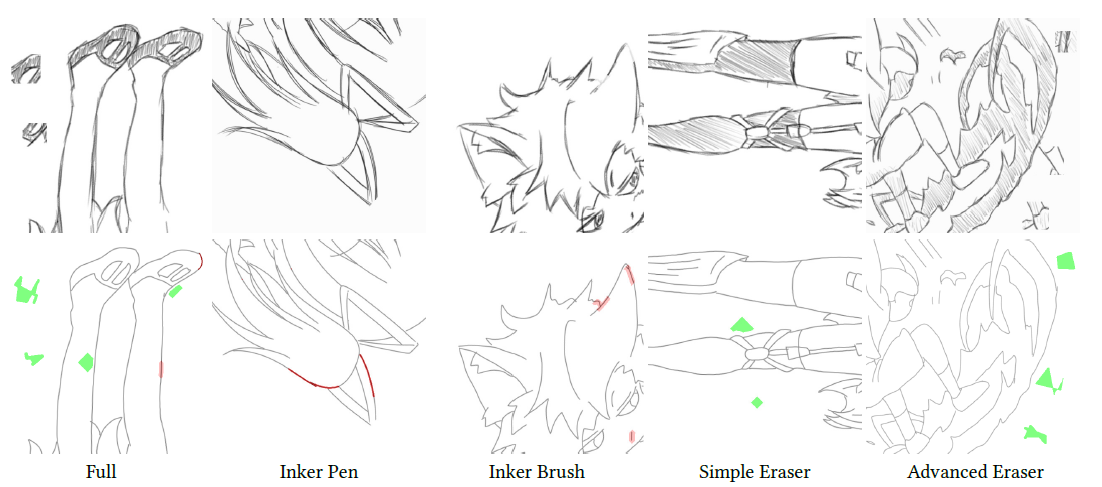
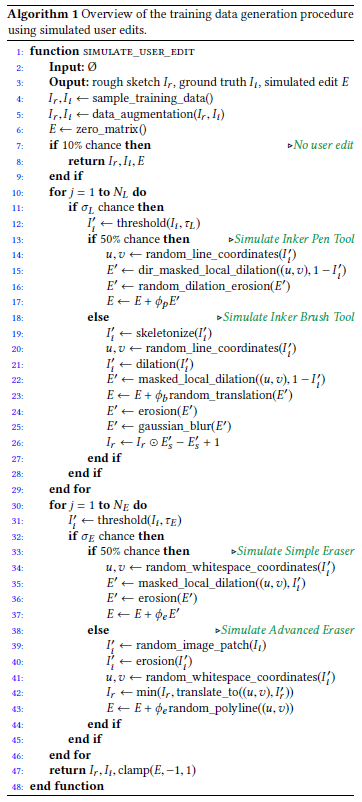


图 6 模拟编辑实例

对于一个草图和一个画线对(,)，论文随机模拟不同类型的工具来生成一个用户编辑映射。特别是，论文模拟了四种不同的动作:钢笔工具，钢笔工具，简单橡皮擦，高级橡皮擦。简单的橡皮擦和高级橡皮擦模拟动作都与论文的智能橡皮擦工具相对应。为了不完全依赖于编辑，论文使用完全零(无操作)编辑映射E的概率为10%。否则，将在之后以二项分布创建一些加法编辑和在之后以二项分布创建一些减法编辑。对于每个附加编辑，有50%的可能性它是一个模拟的墨水笔工具编辑和50%的可能性它是一个墨水笔工具编辑。在智能擦除工具的情况下，每个编辑有50%的机会是一个简单的擦除编辑和50%的机会是一个高级擦除编辑。在智能擦除工具的情况下，每个编辑有50%的机会是一个简单的擦除编辑和50%的机会是一个高级擦除编辑。一般情况下，由于用户编辑仿真在训练过程中需要进行大量的仿真，所以论文优先考虑保持仿真的简单性和快速性。

论文基于GPU后端与简单的Web前端的组合构建了一个应用，如图7所示。用户同时拥有一个大的输入区和一个大的输出区。按照传统的墨水操作，论文用蓝色显示草图，输出覆盖在左边的顶部。然后，用户可以直接编辑该图像，并将结果传播到右边显示的输出。通过前端与后端异步通信，呈现一个可以在0.2秒内处理1512×1512图像的模型外，确保用户能够实时交互式地编辑大型草图。



图 7 应用

## 3.3 训练

论文用几对草图和线条图训练模型。除了使用模拟用户编辑，还将线条归一化网络用于离线和实时数据归一化，结合加权损失，可以获得清晰的线画输出，从而避免后续处理，使用户可以进行更直观的交互编辑。

### 3.3.1 数据集

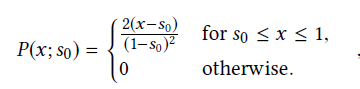
论文收集了一个由6位不同插画家遵循“逆数据集构建”方法绘制的288个草图和线条绘制对的新数据集。不像以前的工作，论文收集了更复杂的草图，包括大量的铅笔阴影和缺陷。与以前的方法相比，这些图片具有更大的挑战性，这在使用模拟用户编辑进行学习时是必要的。从训练数据集中提取的一些示例修补如图6所示。

### 3.3.2 离线数据归一化

在创建一个更大更有挑战性的数据集时，发现的一个问题是，线条图的厚度在插画之间变化很大，甚至在不同的图纸之间，并导致模糊的输出。为此，在初始化训练过程之前，采用线归一化模型对数据集进行归一化。特别地，使用前文提到的宽泛模型，经过训练后，能够抗混叠将线条规范化为2像素宽。首先，得对所有的训练图像进行调整，使其总像素不超过1512，然后使用线归一化模型对它们进行处理，得到一个更加一致和连贯的数据集，从而最小化输出中的模糊。

### 3.3.3 每个草图的数据增强

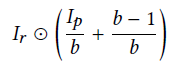
在训练样本数量有限的情况下，论文进行了大量针对草图的实时数据增强，以改进模型的泛化。按照标准操作，除了随机水平翻转和缩放外，论文还随机旋转输入和输出训练图像对。具体来说，对于每幅图像，缩放量由三角概率分布决定:



其中为训练修补的大小除以图像的最短边，表示原始图像的分辨率。

论文还随机修改了输入的草稿在0.8-1.2之间的对比度，1为原始对比度。每个粗略的草图输入都有15%的机会被随机噪声修改，这些随机噪声要么来自正态分布N(-0.05,0.15)，要么来自均匀分布(-0.1，-0.2)，这些随机噪声经过的高斯核滤波，以提高输入图像的鲁棒性。

为了进一步增强粗描中使用的纸张类型的鲁棒性，论文还通过组合来增强粗描。他们用不同类型的纸的图像。具体来说，使用了62张不同类型纸张的不同图像，每组样本以概率15%随机进行以下变换:



式中，为纸的纹理，。注意到，就像草图一样，通过随机旋转、翻转、对比度调整和缩放来增强纸张纹理。图8显示了在训练中使用的不同纸张纹理的一些示例。

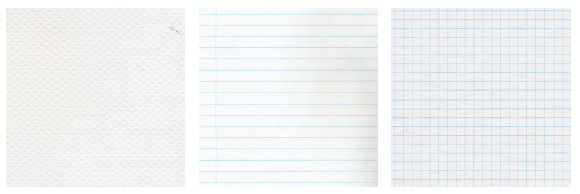


图 8 纹理示例

最后，在实时数据增强方法中执行显著的缩放，使用了一个小的线规范化网络（上文的敏捷的模型），将缩放后的线图重新规范化为2像素的线宽。在训练过程中，使用这个较小的网络作为一个固定的动态预处理阶段，通过进一步减少模糊环来提高模型的输出，因为输出线宽没有歧义，即所有线条宽度均为2像素。

### 3.3.4 目标函数

论文通过最小化S来训练智能墨笔模型。



其中D为模拟用户编辑函数(算法1)，为模型参数。

损失函数定义如下：



其中是加权超参数控制空白行有多少重要性。

论文发现加权损失，结合线性归一化，可以得到干净的反锯齿输出。这使得可以避免对输出进行额外的后处理，这对于在执行交互式编辑时向用户提供更直观的直接控制非常重要。与标准训练方法的比较如图9所示。优化是使用随机梯度下降的ADADELTA[8]变体来完成的，它不需要显式的学习率参数。

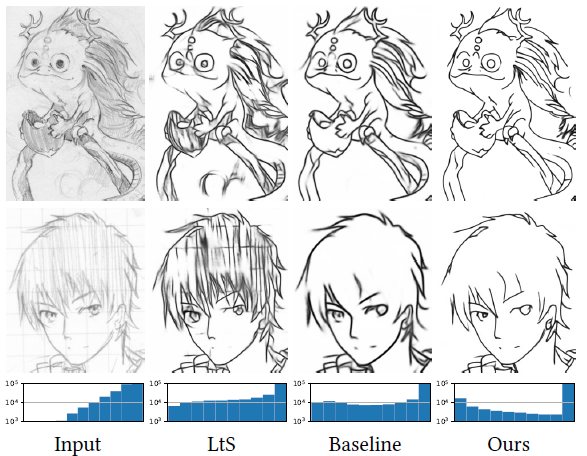


图 9 对比

# 四、思考与扩展建议

这篇论文提出了一种半自动实时交互框架，用于草图的描边。该方法可以更快的描边，对于复杂图像的描边速度可以提高7.2倍，业余用户用来描边的速度也有显著提高。但是对于一些线条较少的简单图像，在交互时间的统计上没有明显的差异，所以，该论文的方法的模型最适合复杂的，需要很长时间的手动描边的图像。

论文方法的一个重要特点是每个像素的输出通过使用241×241像素区域的图像输入计算。尽管由于连接卷积算子的性质，空间上位于输出像素附近的输入像素被赋予更多的重要性，但是仍然存在用户编辑影响用户不打算操纵的输出线绘制的区域的情况。

除此以外，在一张实验图中，用户打算将底线与墨刷线条连接， 虽然成功连接了底线，但上面的部分线被部分删除了，当然这可以通过额外的编辑来解决，但它可以导致错误和用户的挫败感。额外的培训可能是最小化此问题的一种方法。

最后，虽然论文的模型能够提供良好的墨笔描边结果，为业余用户提供改进，并允许对输出进行详细控制，但专业用户可能对结果不完全满意，因为它可能不具有手工着墨的表现力。 我认为下一步应该是与专业墨笔软件紧密集成，服务业余和专业用户。 同时，论文中的墨笔工具，还可以结合用户输入的压力值改善效果。

参考文献

[1]Sangkloy P, Lu J, Fang C, et al. Scribbler: Controlling deep image synthesis with sketch and color[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017, 2.

[2]Zhang R, Zhu J Y, Isola P, et al. Real-time user-guided image colorization with learned deep priors[J]. arXiv preprint arXiv:1705.02999, 2017.

[3]Simo-Serra E, Iizuka S, Sasaki K, et al. Learning to simplify: fully convolutional networks for rough sketch cleanup[J]. ACM Transactions on Graphics (TOG), 2016, 35(4): 121.

[4]Simo-Serra E, Iizuka S, Ishikawa H. Mastering sketching: adversarial augmentation for structured prediction[J]. ACM Transactions on Graphics (TOG), 2018, 37(1): 11.

[5]Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[J]. arXiv preprint arXiv:1502.03167, 2015.

[6]Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines[C]//Proceedings of the 27th international conference on machine learning (ICML-10). 2010: 807-814.

[7]Douglas D H, Peucker T K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature[J]. Cartographica: The International Journal for Geographic Information and Geovisualization, 1973, 10(2): 112-122.

[8]Zeiler M D. ADADELTA: an adaptive learning rate method[J]. arXiv preprint arXiv:1212.5701, 2012.

[9]Shi W, Caballero J, Huszár F, et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 1874-1883.

[10]Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[J]. arXiv preprint arXiv:1502.03167, 2015.

[12]Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines[C]//Proceedings of the 27th international conference on machine learning (ICML-10). 2010: 807-814.

[13]Tompson J, Goroshin R, Jain A, et al. Efficient object localization using convolutional networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 648-656.

[14]Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting[J]. The Journal of Machine Learning Research, 2014, 15(1): 1929-1958.