Real-Time Data-Driven Interactive Rough Sketch Inking

Edgar Simo-Serra, Satoshi Iizuka, Hiroshi Ishikawa
Wednesday, August 15, 2018

Waseda University
Motivation

Input

Previous Approach
Motivation

Input Previous Approach

User Input Standard Eraser
Motivation

Input

Previous Approach

User Input

Standard Eraser

Proposed
“1. The inker’s main purpose is to translate the penciller’s graphite pencil lines into reproducible, black, ink lines.
2. The inker must honor the penciller’s original intent while adjusting any obvious mistakes.
3. The inker determines the look of the finished art.”

Interactive Neural Networks

- Feed-forward fully convolutional neural network
Interactive Neural Networks

- Feed-forward fully convolutional neural network
- Input rough sketch and user edit are concatenated channel-wise
Interactive Neural Networks - Related Work

- User input is treated as an additional image channel
- Training user input is sampled from ground truth
  - Grayscale image colorization [Sangkloy+ 2017, Zhang+ 2017]

Interactive Neural Networks - Related Work

- User input is treated as an additional image channel
- Training user input is sampled from ground truth
  - Grayscale image colorization [Sangkloy+ 2017, Zhang+ 2017]
  - Not directly applicable to the rough sketch inking problem
Interactive Neural Networks - Related Work

- User input is treated as an additional image channel
- Training user input is sampled from ground truth
  - Grayscale image colorization [Sangkloy+ 2017, Zhang+ 2017]
  - Not directly applicable to the rough sketch inking problem
- How to train an interactive network for inking?
Proposed Framework

• **Main contributions**
  - Line width normalization
  - Simulation of user edits
  - Three different smart tools
  - Evaluation with a perceptual user study

©Krenz Cushart
Training Framework

1. Line width normalization
2. Simulation of user edits
Training Framework - Line width normalization

1. Line width normalization
2. Simulation of user edits
Training Framework - Line width normalization

Rendered vector images

Training
Training Framework - Line width normalization

QuickDraw  TUD-Berlin  KanjiVG  Synthetic
Training Framework - Line width normalization

Input [Zhang and Suen 1984] Ours
Training Framework - Line width normalization

Input  No normalization  Normalization
1. Line width normalization
2. Simulation of user edits
Training Framework - Simulation of user edits

Input Data Pair

Line Drawing

Rough Sketch

Input Data Pair
Training Framework - Simulation of user edits

Input Data Pair
Line Drawing
Rough Sketch
Sampled Regions
Training Framework - Simulation of user edits

- Input Data Pair
- Line Drawing
- Rough Sketch
- Sampled Regions
- Add Edits and Noise
Training Framework - Simulation of user edits

Input Data Pair
Line Drawing
Rough Sketch
Sampled Regions
Add Edits and Noise
Training Framework - Simulation of user edits

Input Data Pair
Line Drawing
Rough Sketch
Sampled Regions
Add Edits and Noise
Smart Tools

- Inker Pen
- Allows for fine-grained line control

Input | Automatic | Edit | Ours
Smart Tools

- Inker Brush
- Sloppy and fast line manipulation
Smart Tools

- **Smart Eraser**
- Takes into account rough sketch when erasing

![Input](image1)

![Automatic](image2)

![Edit](image3)

![Ours](image4)
Training

\[ L(y, y^*) = \left| (y - y^*) \right|_1 \odot (1 + \gamma (1 - y^*)) \]

- Using \( L_1 \) loss
- Change weight of lines with \( \gamma \)

©David Revoy www.davidrevoy.com
Model

- Similar to model of [Simo-Serra+ 2016]
- 24 layer fully convolutional neural network
- Number of filters optimized for real-time performance
- Roughly three times the performance

<table>
<thead>
<tr>
<th>Approach</th>
<th>Parameters</th>
<th>1024² px</th>
<th>1512² px</th>
<th>2048² px</th>
<th>2560² px</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Simo-Serra+] 2016</td>
<td>44,551,425</td>
<td>238.8ms</td>
<td>562.4ms</td>
<td>984.7ms</td>
<td>1.59s</td>
</tr>
<tr>
<td>Ours</td>
<td>12,795,169</td>
<td>89.9ms</td>
<td>225.5ms</td>
<td>382.7ms</td>
<td>592.9ms</td>
</tr>
</tbody>
</table>
Dataset

- 288 rough sketch and line drawing pairs
- More challenging than previous works
Real-Time Data-Driven Interactive Rough Sketch Inking

Edgar Simo-Serra  Satoshi Iizuka  Hiroshi Ishikawa
Waseda University
Results

Input | Automatic | Edit | Ours

©Eisaku Kubonouchi
Results

Input  Automatic  Edit  Ours

©Eisaku Kubonouchi
User Study

- Comparison (time) with proposed approach vs Clip Studio Pro
- Total of 10 users and 10 unique images
- Each user processes random 5 images with each tool
- Total average time of 2.8 hours per user
- Overall 1.8× speed-up with ours

Some of the images used in the user study
User Study

©David Revoy www.davidrevoy.com
User Study

©David Revoy www.davidrevoy.com

Time
Input
Amateur
Experienced
Edit
Ours

©Krenz Cushart
Limitation

Input  Automatic  Edit  Ours
To conclude

http://hi.cs.waseda.ac.jp/~esimo/research/inking/

- Interactive rough sketch inking framework
  - Line width normalization
  - User edit simulation

©Edgar Simo-Serra