BézierSketch: A generative model for scalable vector sketches

Das, Yongxin Yang, Timothy Hospedales, Tao Xiang, Yi-Zhe Song

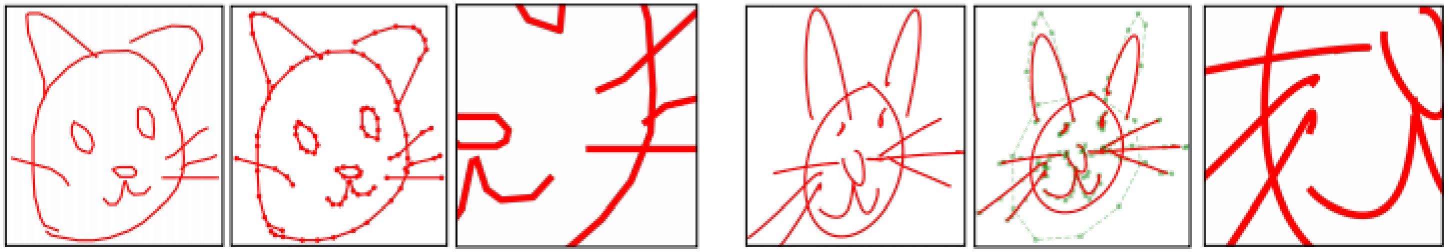
ECCV 2020

Sanghyeon Lee

06 Jan. 2021

Introduction

Task : Generation scalable vector sketches & Translation raster image into vector sketches

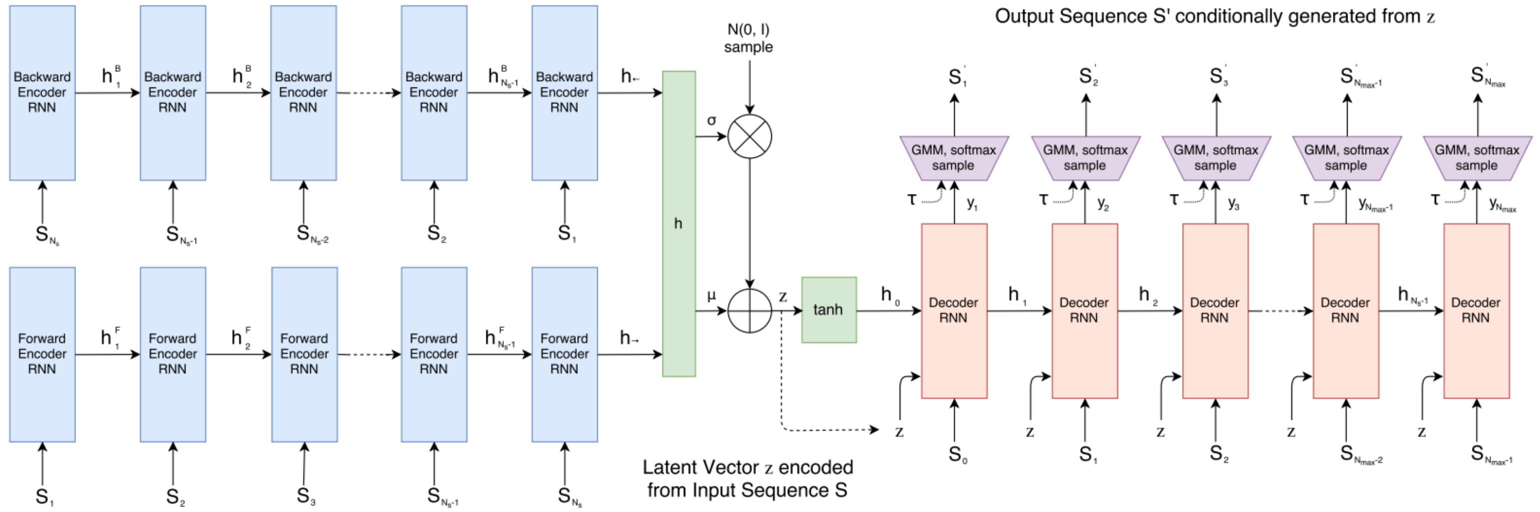


Contribution

- BézierEncoder: Inverse-graphics approach for mapping strokes to parameterized Bézier
- BézierSketch: A sequential generative model for sketches that produces high resolution and low-noise vector graphic samples
- Training model without supervision (Inverse Graphic)

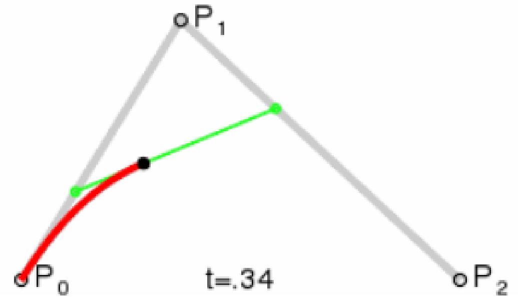
Background

Sketch RNN



Bézier curve

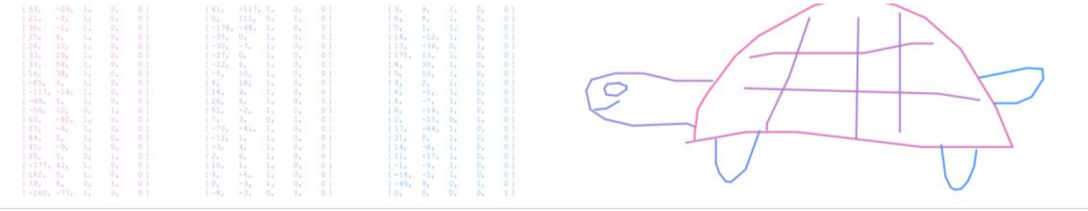
$$\mathbf{C}(t;\{\mathbf{P}_i\}) = \sum_{i=0}^{n} \mathcal{B}_{i,n}(t)\cdot\mathbf{P}_i \qquad \mathcal{B}_{i,n}(t) \triangleq \binom{n}{i} t^i (1-t)^{n-i}$$



Method

1. Sketch Representation

Common format



$$\mathcal{S} = \left[(\mathbf{X}_i, q_i)\right]_{i=1}^{L} \qquad \mathbf{X}_i \triangleq \left[x \ y\right]_i^T \in \mathbb{R}^2,\ q_i \in \{\textsc{PenUp}, \textsc{PenDown}\}$$

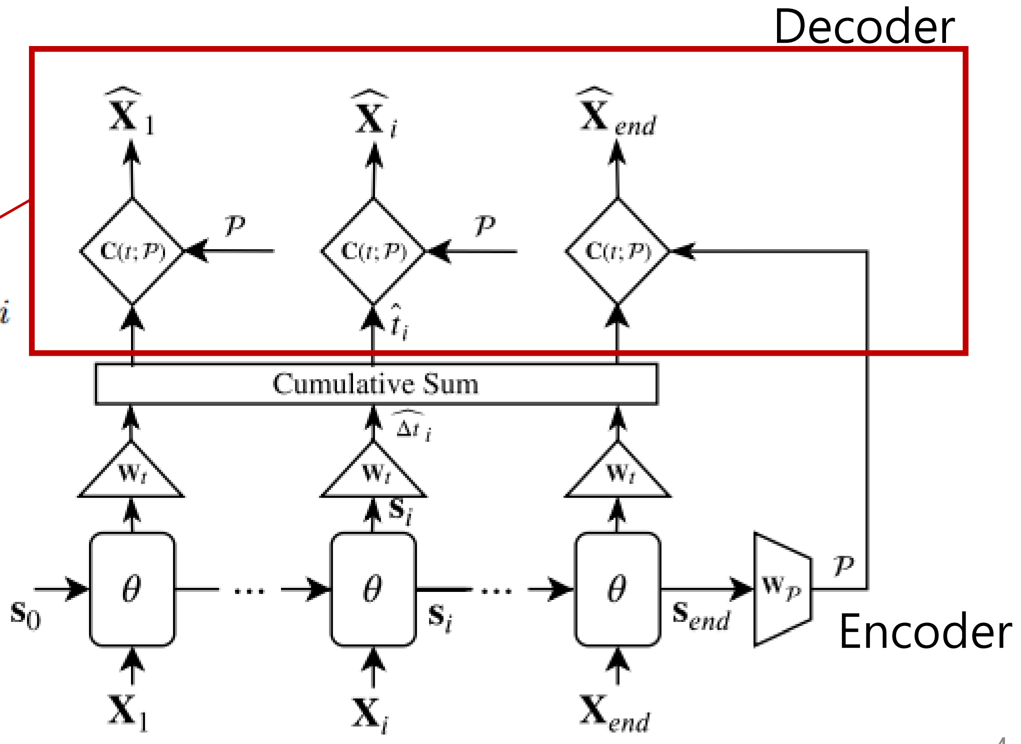
Stroke-Level Representation

$$\bar{\mathcal{S}} \triangleq \left[\mathbf{T}_j\right]_{j=1}^{N}, \text{ with } \mathbf{T}_j \triangleq \left[\mathbf{X}_i^{(j)}\right]_{i=1}^{N_j}$$

2. BézierEncoder: Stroke Embedding

$$\mathbf{C}(t; \{\mathbf{P}_i\}) = \sum_{i=0}^{n} \mathcal{B}_{i,n}(t) \cdot \mathbf{P}_i \qquad \mathcal{B}_{i,n}(t) \triangleq \binom{n}{i} t^i (1-t)^{n-i}$$

$\mathbf{P} \triangleq \left[P_x \ P_y\right]^T \in \mathbb{R}^2$: Control points



Method

2. BézierEncoder: Stroke Embedding

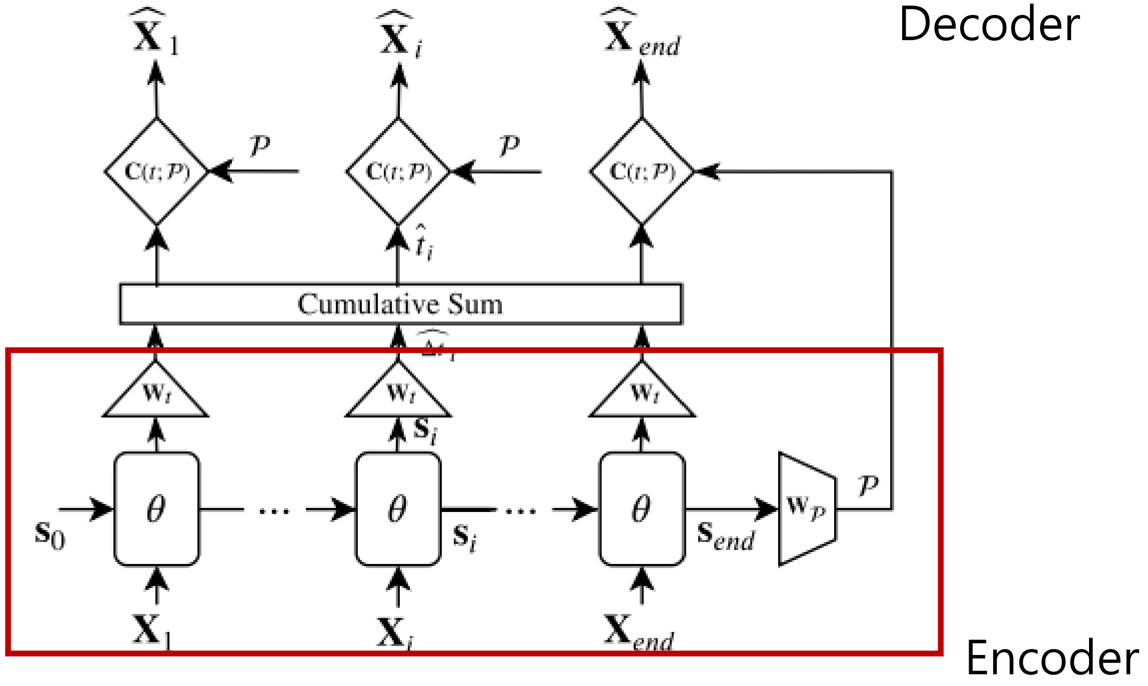
$$\left[\overrightarrow{\mathbf{s}_i}, \overleftarrow{\mathbf{s}_i}\right] = \text{BiRNN}(\mathbf{X}_{i-1}, \mathbf{s}_{i-1}; \theta)$$

$$\mathcal{P} = \mathbf{W}_{\mathcal{P}} \left[\overrightarrow{\mathbf{s}}_{end}; \overleftarrow{\mathbf{s}}_{end}\right]$$

$$\mathbf{W}_{\mathcal{P}} \in \mathbb{R}^{2(n+1)\times 2h}$$

$$\widehat{t}_i = \sum_{i'=1}^{i} \widehat{\Delta t}_{i'}, \text{ with } \widehat{\Delta t}_i = \text{SOFTMAX}_i(\mathbf{W}_t \cdot [\overrightarrow{\mathbf{s}_i}; \overleftarrow{\mathbf{s}_i}])$$

$$\mathcal{L}(\theta, \mathbf{W}_{\mathcal{P}}, \mathbf{W}_t) \triangleq \sum_i \left\|\mathcal{C}(\widehat{t}_i, \mathcal{P}) - \mathbf{X}_i\right\|^2$$



Multi-Degree Representation Extension

$$\mathcal{L}_{total} \triangleq \sum_{n=n_{min}}^{n_{max}} \mathcal{L}_n, \text{ with } \mathcal{L}_n(\theta, \mathbf{W}_{\mathcal{P}^n}, \mathbf{W}_t^n) \triangleq \sum_i \left\|\mathcal{C}(\widehat{t}_i^n, \mathcal{P}^n) - \mathbf{X}_i\right\|^2$$

Smoothness Regularizer

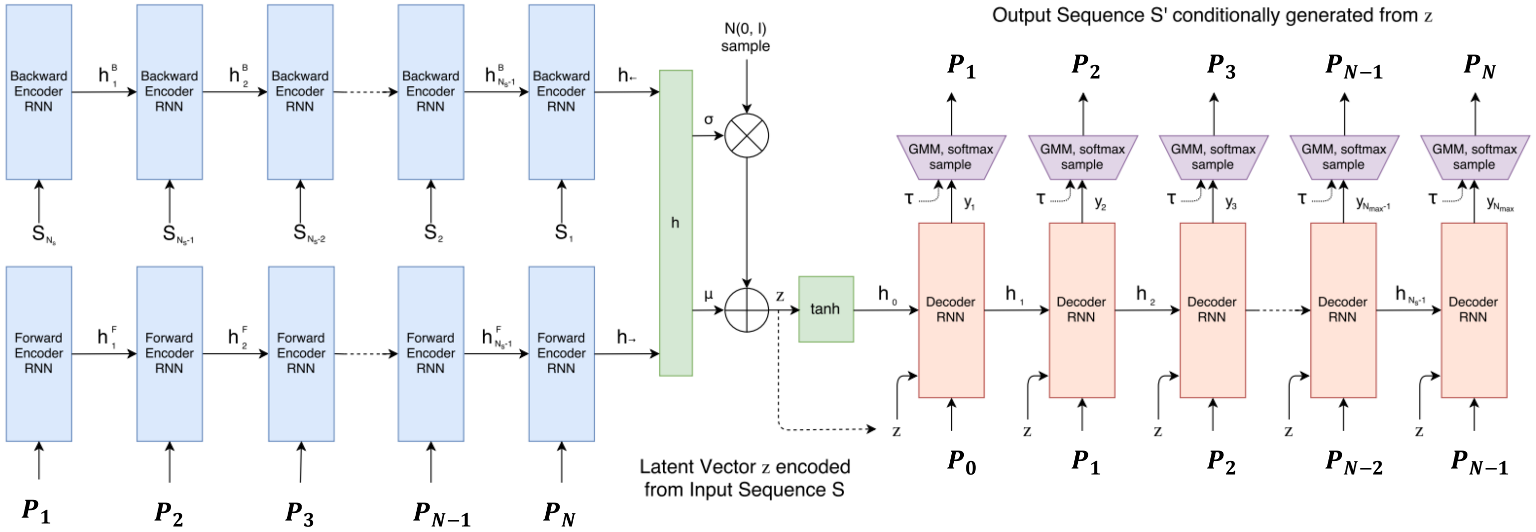
$$\mathcal{R}_n(\mathcal{P}^n) \triangleq \sum_{i=1}^{n} \|\mathbf{P}_{i+1} - \mathbf{P}_i\|_2^2$$

Method

3. BézierSketch: Sketch Generation

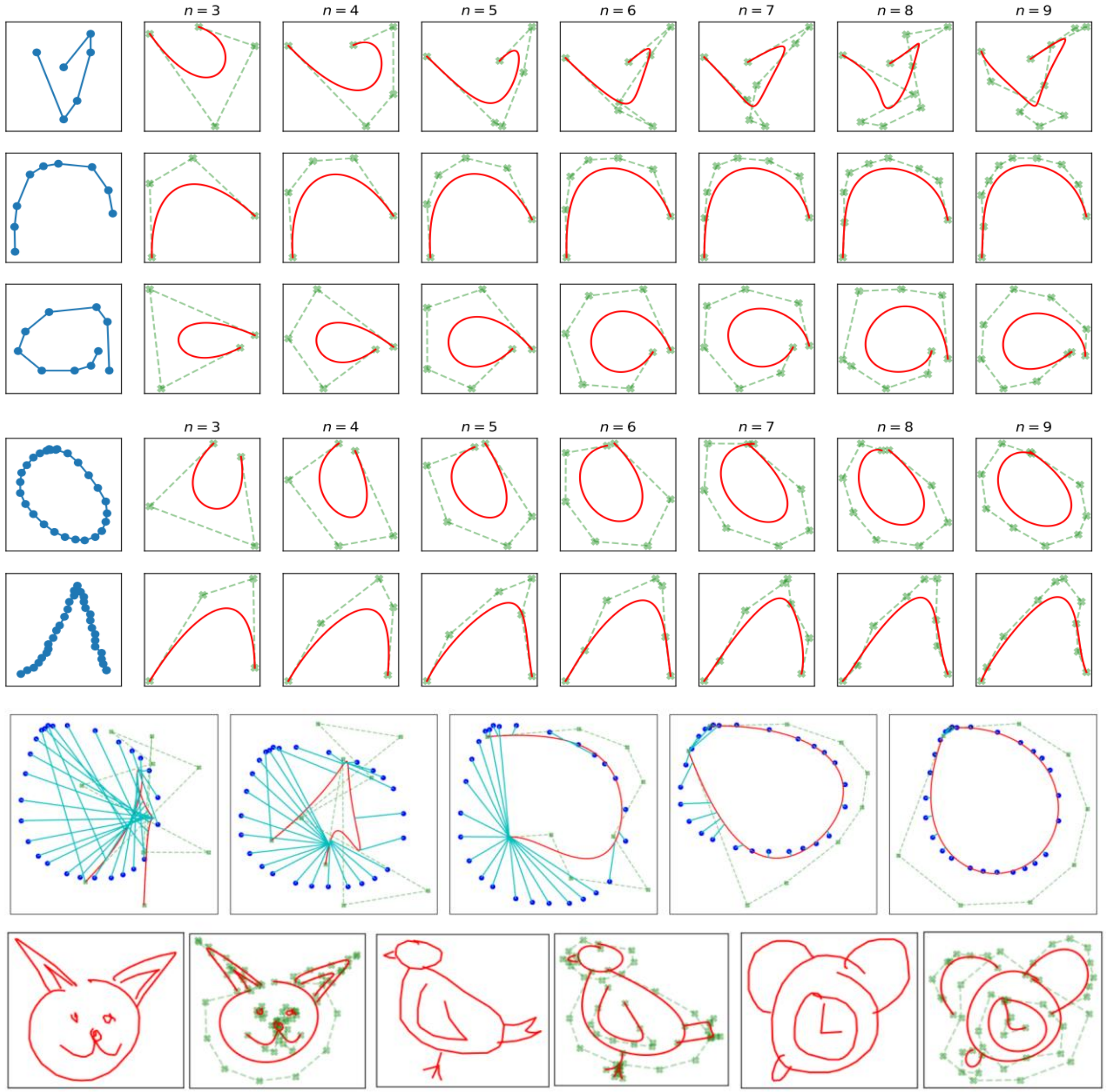
1) Transform the set of strokes as $S_{st} = \{P_j\}_{j=1}^{N}$, Where $P_j = e(T_j)$

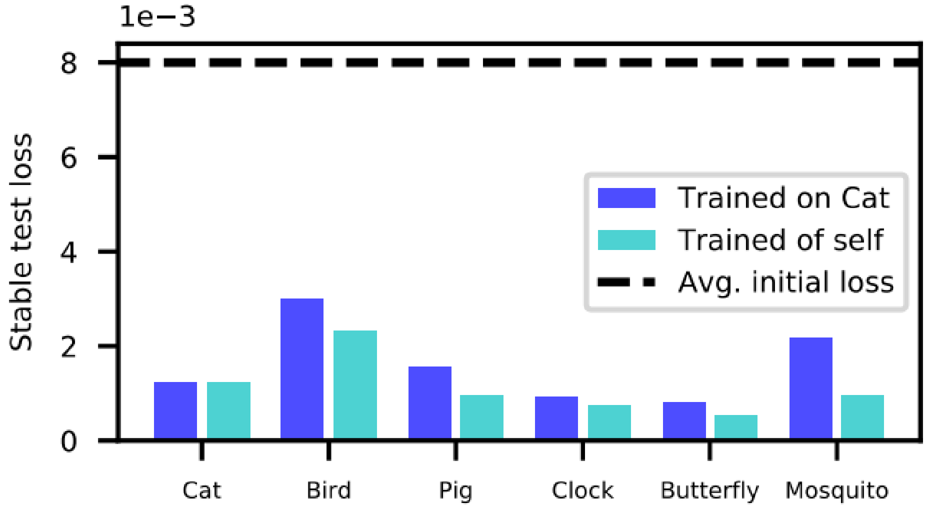
2) Training model via SketchRNN schema



Experiments

1. BézierEncoder





Experiments

2. BézierSketch

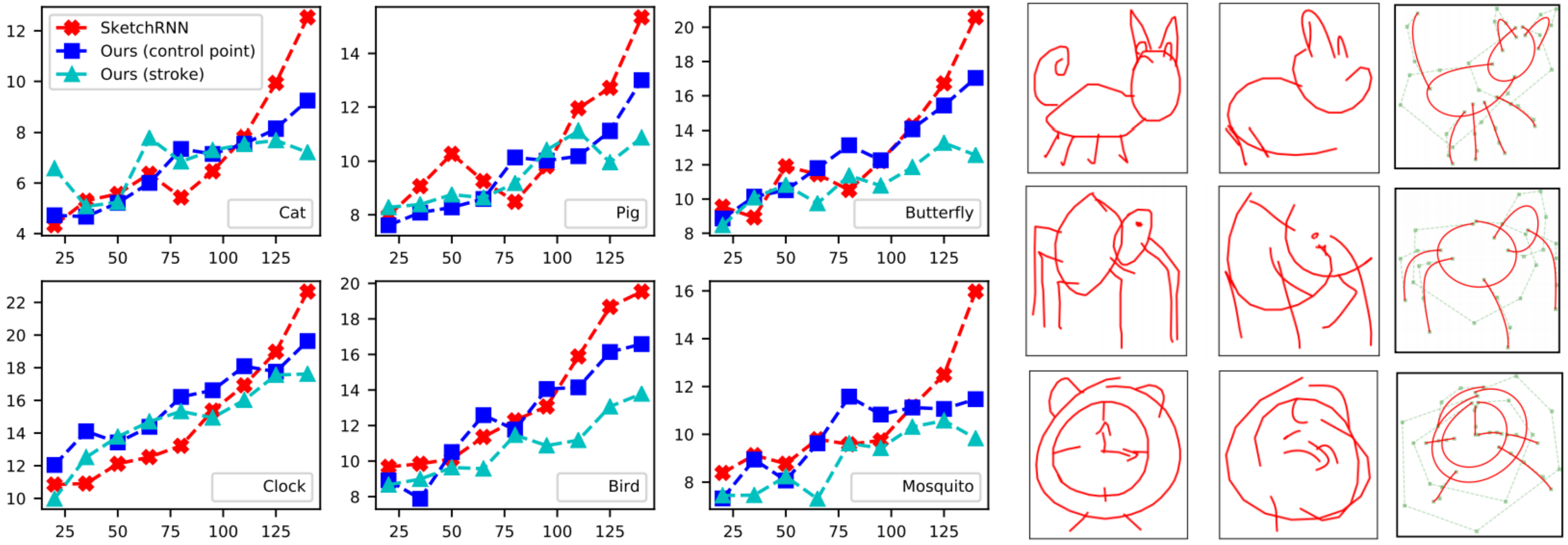
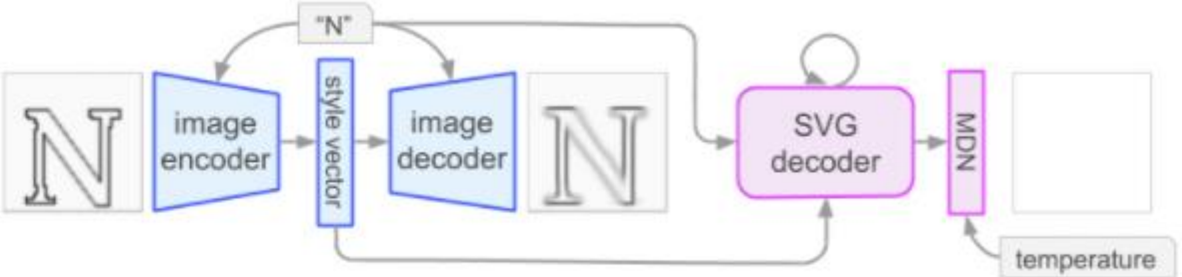
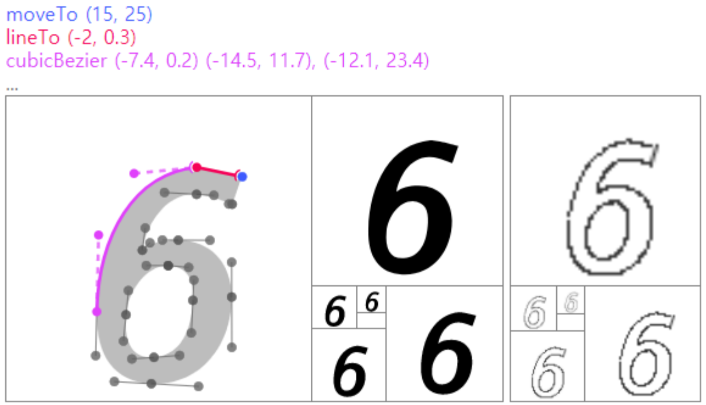


Fig. 7: Left: FID score (↓) vs length of sketch shows the effectiveness of our generative model on longer sketches. Right: Qualitative samples of long sketches. Three columns denote the original sketch, SketchRNN and our BézierSketch.

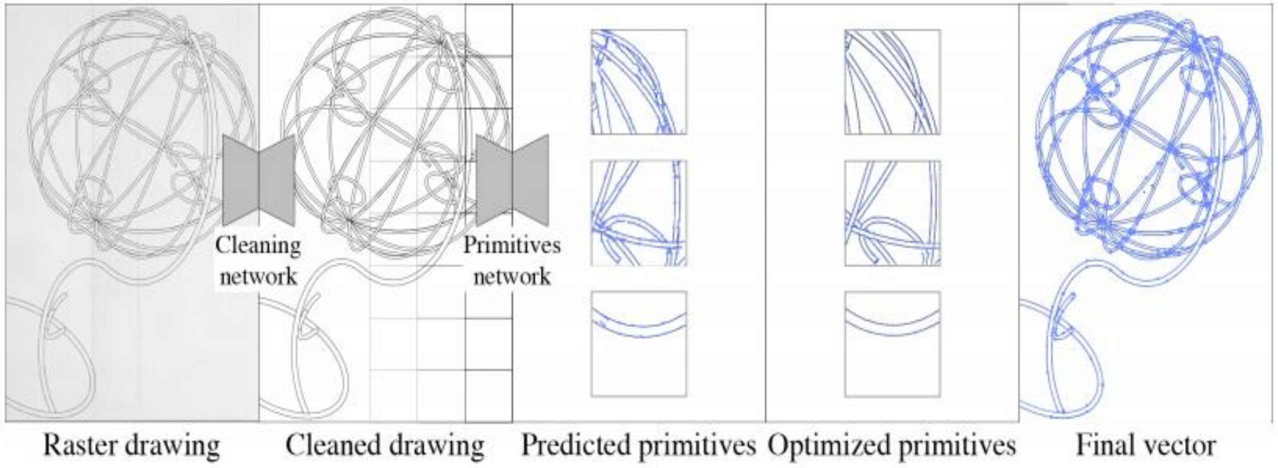
Other works

A Learned Representation for Scalable Vector Graphics (CVPR2019)





Deep Vectorization of Technical Drawings(CVPR2020)



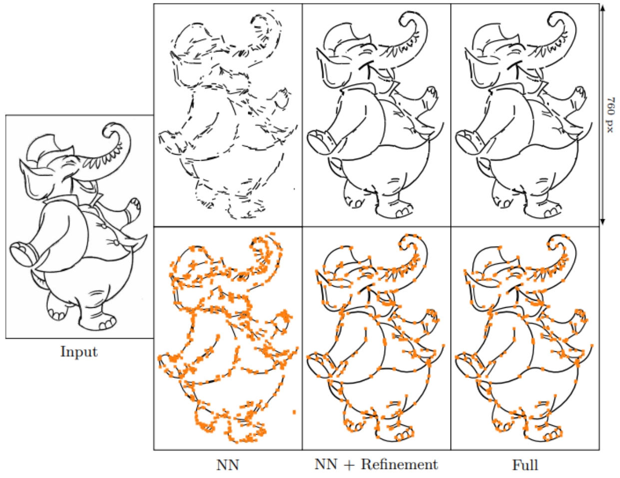


Fig. 16: Qualitative results of our system on clean cartoon drawings. Endpoints of primitives are shown in orange. The input image on the top is copyrighted by David Revoy www.davidrevoy.com under CC-by 4.0 license and on the bottom from www.easy-drawings-and-sketches.com, © Ivan Huska.

Thank you