Deep Sketch Network for Million-scale Sketch Recognition

Kegang Xu Hongwei Zhang Dan Xia

Stanford University

Overview

Problem

QuickDraw is Google Open Challenge for sketch drawing recognition. It is an experimental game in a playful way to show how AI works. Advancing on the sketch recognition could greatly help OCR, ASR and NLP.

Below are the challenges in our project:

- 50M+ drawing and 340 categories.
- Highly abstract and iconic.
- Various style due to different countries.
- Incomplete and misleading noisy data

1) GB (Great British) typically drew cell phone more like smart phones.

2) ZA (South Africa) generally were still under the impression of cell phone as feature phones.

Dataset

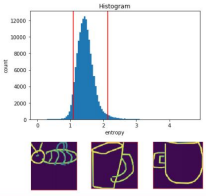
Due to the limit GPU resources, we chose 40 categories according to the difficult level of recognition based on pretrain model.

- Stroke data comes from Kaggle website.
- Stroke dataset are input to RNN and Pixel images are created from stroke data on the fly as an input to CNN.
- Use the whole dataset of each class. Each includes 12K to 20K images.
- Especially choose the similar classes, like mug, coffee cup, cup.
- Split the dataset into 3 segments and each class includes 11K+ images of training data, 512 images of Dev data, 512 images of Test data

Model	Dataset
Baseline	Pixel based sketch
CNN	Pixel based sketch
RNN	Stroke vector data
DSN	Pixel based sketch + Stroke vector data
DSN+ (with center loss)	Pixel based sketch + Stroke vector data

Data Enhancement

- Remove the noisy data by calculating the entropy histogram of each class. Keep the images in [0.05,0.95]
- Create one channel image with each stroke of different colors as input to CNN.
- Horizontal Flip data is used as data enhancement since all categories could have different orientations.



Model

Network Architecture

- Our Deep Sketch Network includes one CNN and one RNN.
- CNN is a modified resnet50 which inputs one channel and RNN is a hidden size 256 of bidirectional LSTM with 2 Conv layers.
- Attention Layer as the last layer links CNN and RNN together
- RNN input is a series of strokes. CNN input is pixel images created on the fly from these strokes.

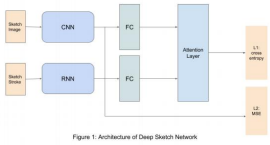


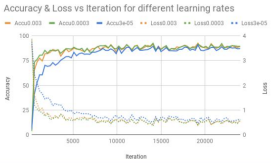
Figure 1: Architecture of Deep Sketch Network

Loss Definition

- DSN Loss: $\text{Loss} = \ell_1 + \alpha \cdot \ell_2$
- Cross Entropy Loss: ℓ_1 and Center Class Loss: $\ell_2 = \frac{1}{N}\sum_{n=1}^{N} ||Fn - Cn||^2$
- Cn represents the mean feature vector of one class. Fn is the feature vector of each sample. Feature vector is the combination of the output of RNN and CNN right before FC layer.

Experiments

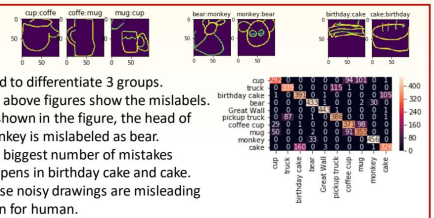
- We used Adam optimizer to train baseline CNN, RNN, Resnet50, DSN and DSN+.
- The figure below shows training is progressing well as loss decreases gradually with lr=3e-3,3e-4 and 3e-5.
- We trained alpha value with 0.1,0.01 and 0.001. The accuracy shows a little difference. 0.01 is the best one.



- Converge faster with bigger learning rate, but the best accuracy is achieved with 3e-4.
- Top figure: train with different batch size;
- Bigger batch size will improve accuracy, 256 is more stable and above the other sizes.
- Batch size 128 is not stable and has a dip due to the noise data.

- Higher Image resolution helps accuracy as the image size grows in left figure.
- Center Loss could slightly improve accuracy. It depends on how we extracted the center feature.
- The right image is mislabeled by CNN but correctly classified by DSN as a bird.
- The category varies in accuracy from cup(71.6%) to tent(98.6%).
- RNN is hard to train than CNN, e.g. more sensitive to the learning rate and hard to get high accuracy in this sketch recognition.

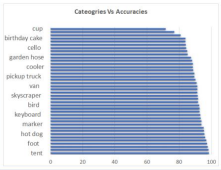
Image size	MAP3
28X28	77.80
64X64	89.39
96X96	91.76



- Hard to differentiate 3 groups. The above figures show the mislabels.
- As shown in the figure, the head of Monkey is mislabeled as bear.
- The biggest number of mistakes happens in birthday cake and cake. These noisy drawings are misleading even for human.

Results

- Our DSN (91.47%) is much better than Resnet50 (87.37%) and a simple base line CNN(86.12%)
- Our DSN+ increases to 91.76% by introducing the center loss.
- The resolution of images matters. But due to the limit GPU resources, only 96X96 is used.
- Batch size matters in this competition as higher batch size could reduce noisy data.
- We achieved our best performance on the whole training dataset.



Model	MAP@3
Baseline(CNN)	86.12
RNN	86.08
CNN(Resnet50)	87.37
DSN	91.47
DSN+	91.76

Conclusion

- Design DSN+ with center loss which improves the accuracy greatly compared to the baseline and the resnet50.
- Removing the noisy data improves the performance. Big batch size could alleviate the noisy data.
- The whole data training of 50 Million sketch and high-resolution image like 256X256 require huge GPU resources. But it will improve the accuracy. Kaggle competition is closed, we could not run its test data.
- The following area could be explored in the future:
  1) Explore other CNN like seresnet.
  2) Explore the better RNN, like attention layer and bidirectional GRU.
  3) Try to train high resolution image with bigger batch size if possible.

References:

- D. Ha and D. Eck. A neural representation of sketch drawings. arXiv:1704.03477, 2017.
- Peng Xu,etc. SketchMate: Deep Hashing for Million-Scale Human Sketch Retrieval http://openaccess.thecvf.com/content_cvpr_2018/CameraReady/2763.pdf
- Q. Yu & F. Liu et al. Sketch-a-net: A deep neural network that beats humans. IJCV 2017
- *Poster Video: https://youtu.be/Uot5k19BgZI*