How CNNs perform Sketch Classification?

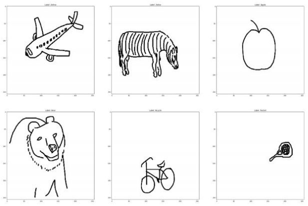
Sushan Bhattarai {sushanb}
Stanford University

INTRODUCTION

- Due to the exponential growth in touch sensitive devices, the number of drawings/sketches/ outlines has increased a lot.
- People tend to use sketches to express their ideas and emotions quickly and without hassle.
- We trained three Convolutional models for doing sketch classification.
- We achieved accuracy of 82% on Resnet model and 35.40% on Baseline model.
- Models were implemented in Keras/Tensorflow.

DATASET

- The project uses the publicly available Sketchy database, the first large-scale collection of sketch-photo pairs. [1]
- It contains 20000 sketches objects across 125 categories.
- The dataset is balanced.
- There is presence of high intra class variation and low inter class variation which makes classification task challenging.



Overview of Dataset

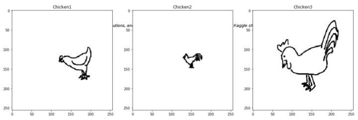


fig: Example of variation among chicken class

METHODS

Preprocessing

- We stacked grayscale image 3 times to get image with 3 channels.
- Normalize all pixels by 255

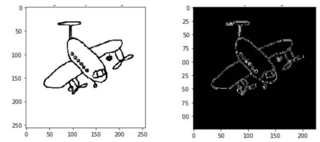


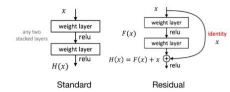
fig: 1 channel to 3 channel

MODELS

- **Baseline Model**
  - It is a 3 layer convolutional network followed by 2 layer of fully connected network.
  - The convolutional network consists of actual convolution, relu activation, batch normalization and downsampling operation.
  - We train the baseline model from scratch for 80 epochs.
  - We used multi class cross entropy loss and RMS optimizer for training the model.
  - It took 40 minutes to train the model with two NVIDIA K80 GPUs .
- **Advanced Model 1 (Finetuning Pretrained ResNet)**
  - ResNet by proposed by He et al. to solve vanishing and exploding gradient problem when we use deep convolutional network.[2]
  - ResNet is only composed of small sized-filters of 1x1 and 3x3 with the exception of first convolutional layer of kernel size 7 * 7.
  - Every single layer is followed by batch normalization and ReLU activation.
  - Due to lack of extensive computation re-sources, we used the truncated version of ResNet of 50 layers
  - The main difference between ResNet and Standard Neural net is that it contains layers that learn residual functions with the reference to the layer inputs.
  - We trained the 50 layer ResNet for 12 epochs using multi class cross entropy loss and Adam Optimizer with 0.001 learning rate.

Models/Results

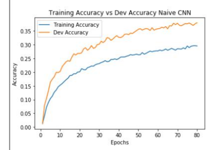
Models[Continued]

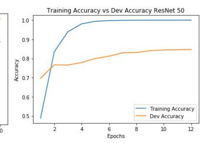


Difference between ResNet connection and Standard connection

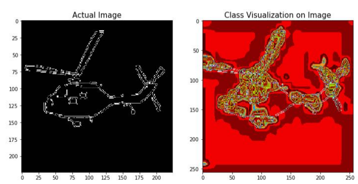
- **Advanced Model 2 (DenseNet -Results pending)**
  - Huang et al. introduced Dense Convolutional Network called as DenseNet where each layer connect to every other layer in a feed-forward fashion [3].
  - In traditional convolutional networks with Ll ayers, we have L connection whereas in DenseNet we have L∗(L+1)/2connections.
  - We trained the 121 layer ResNet for 12 epochs using multi class cross entropy loss and Adam Optimizer with 0.001 learning rate.

Results



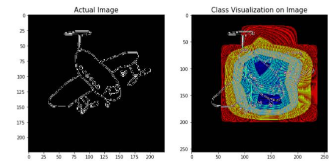


Class Activation Map for Naive CNN



Results/Discussions

Class Activation Map for ResNet50



- ResNet achieves an accuracy of 82% and Baseline model achieves a poor accuracy of 35.10%.
- ResNet was able to achieve greater accuracy due to the use of transfer learning.
- We trained ResNet from scratch as an experiment for same epoch and the accuracy was 15%.
- In order to decrease overfitting in the ResNet model, we need to add append dropout layers at the end of ResNet model followed by the classification/output layer.
- From Class activation map, we can see that ResNet was able to ignore the background pixels completely and focus on airplane. However, naive CNN was focusing all over the place.

FUTURE WORK

- Blend/Ensemble non-neural and neural methods
- Add Attention mechanism on the top of ResNet and DenseNet
- K Fold validation and Search for fine hyper parameter tuning
- As an experiment, augment the data by zoom, flip, width and height shift
- Continue to tune hyper parameters, schedule learning rates for each epoch.

REFERENCES

1. Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database: Learningto retrieve badly drawn bunnies.*ACM Transactions on Graphics* (proceedings of SIGGRAPH),2016.
2. K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learningfor image recognition.CoRR, abs/1512.03385, 2015.
3. G. Huang, Z. Liu, and K. Q. Weinberger. Densely connectedconvolutional networks.CoRR, abs/1608.06993, 2016