



SKETCHYGAN

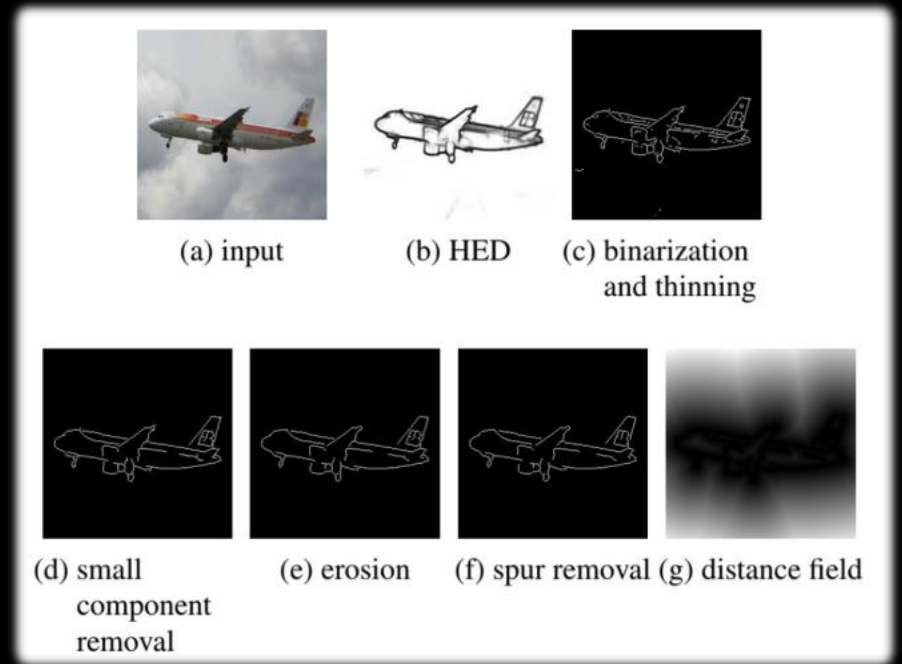
utilized methods

TECHNICAL SPECIFICATIONS

- Generative Adversarial Neural Network
- Python 3, NumPy, SciPy, OpenCV 3
- Tensorflow 1.7.0
- Trained on a recent NIVIDIA GPU

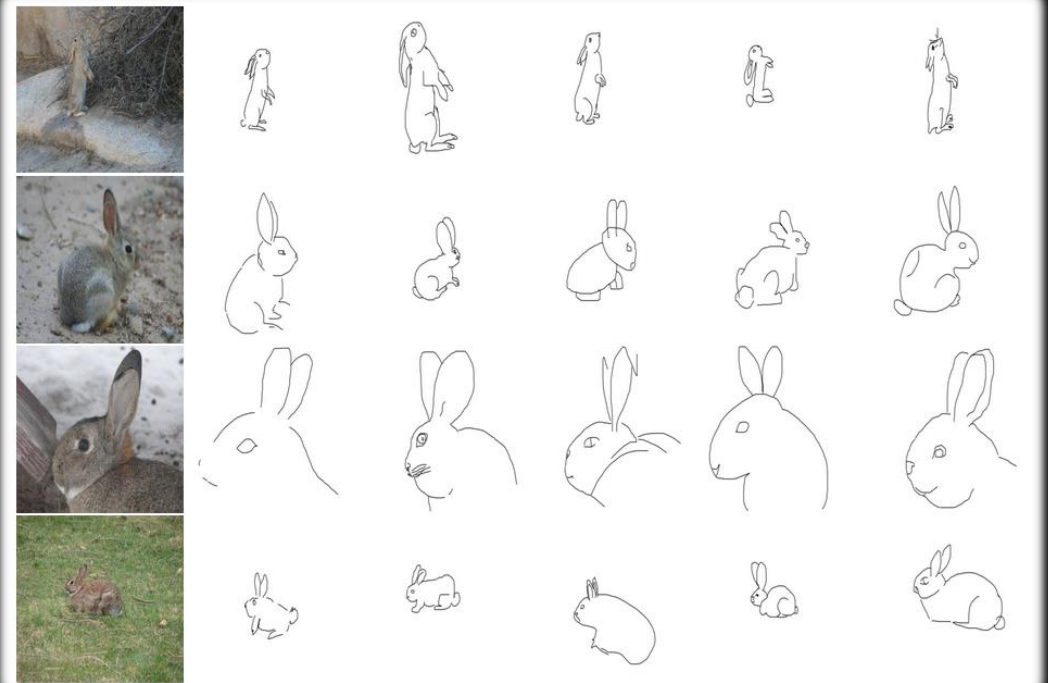
DATA AUGMENTATION

- First training the network on edge maps
- Images with edge maps collected from Flickr
 - Web crawler, categories from *Sketchy database*
 - HED: *Holistically-nested edge detection*
 - Cleaning up fragments
 - Sparse edges -> *Euclidian distance field* for dense representation



DATA AUGMENTATION

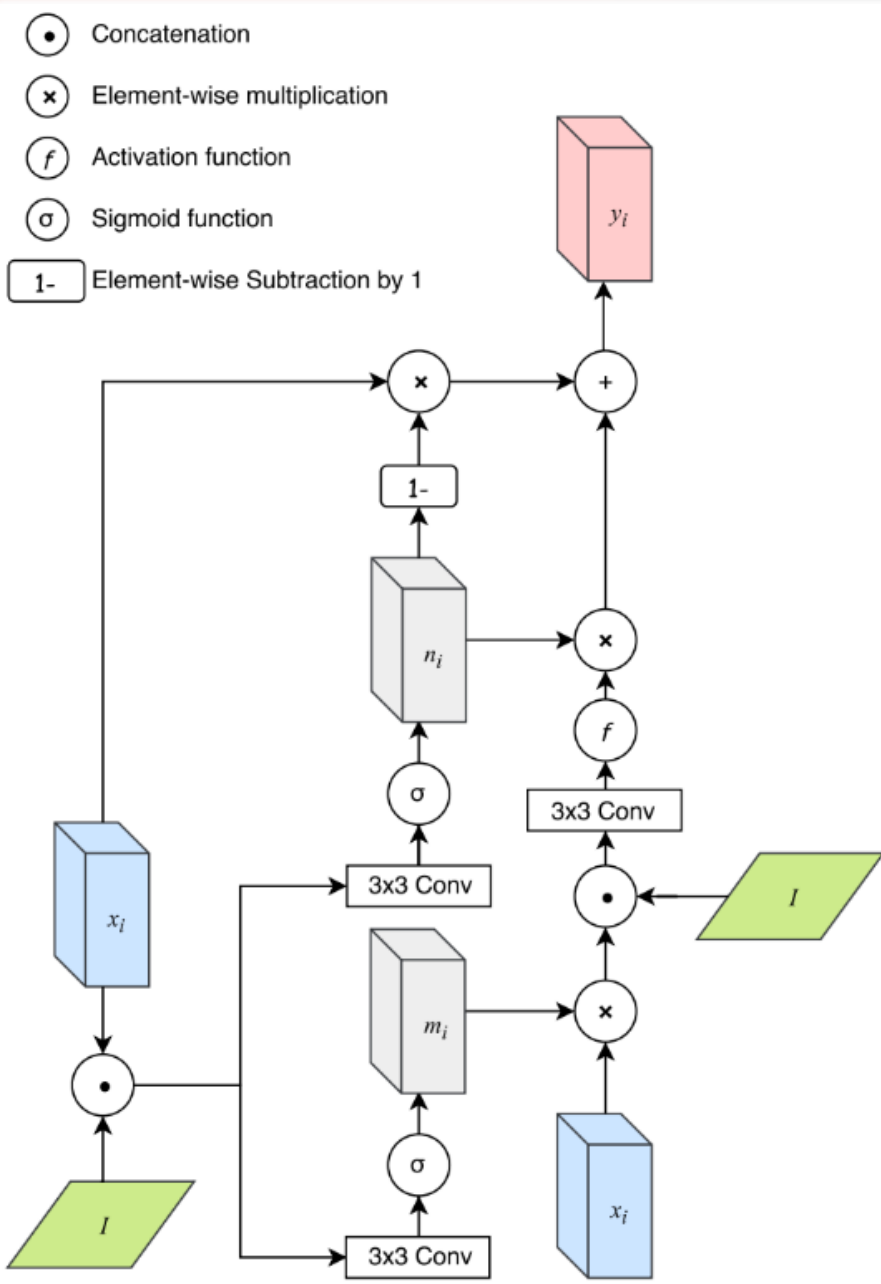
- First training the network on edge maps
- Slowly changing to human-drawn sketches
 - Sketchy database [P. Sangkloy et al.]
 - 12 500 photos, 75 471 sketches
 - 125 categories,
50 used in SketchyGAN



SKETCHYGAN

- Learns mapping from an input sketch to output image
- Generator G tries to generate realistic images from sketches
- Discriminator D tries to distinguish training data from generated images

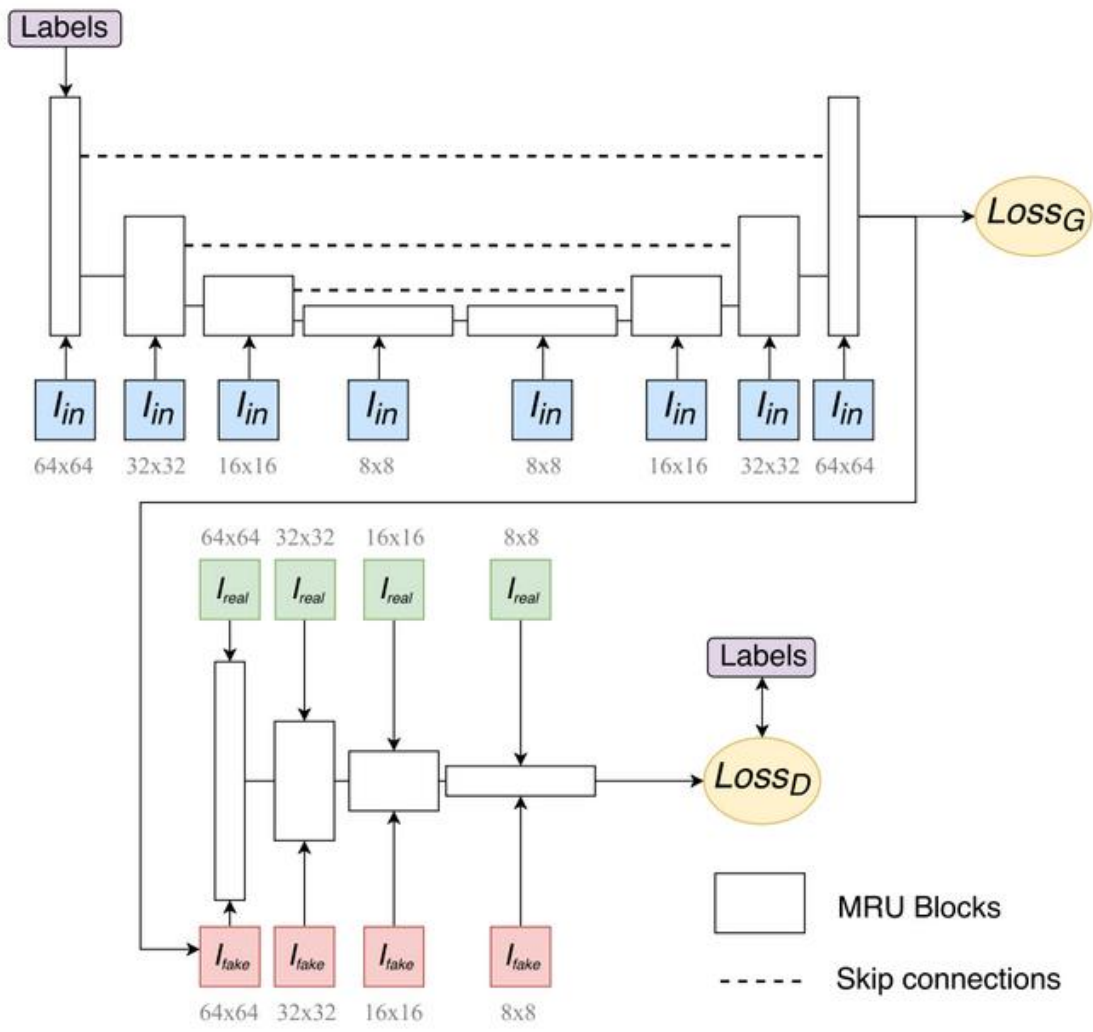
- *GAN objective function:* $\mathcal{L}_{GAN}(D, G) = \mathbb{E}_{y \sim P_{image}} [\log D(y)] + \mathbb{E}_{x \sim P_{sketch}, z \sim P_z} [\log(1 - D(G(x, z)))]$



MRU: MASKED RESIDUAL UNIT

- Sketches are resized, fed into every MRU block
- input: feature maps x_i and an extra image I
output: new feature maps y_i
- uses a learned internal mask
 - selectively extract new features from input images
 - combine with already computed feature maps

NETWORK ARCHITECTURE



- Encoder-Decoder structure
- Built with *MRU blocks*
- Skip-connections between encoder and decoder blocks
- Discriminator shrinks in spatial dimensions

RESULTS



Sources:

- Chen, Wengling, and James Hays. "Sketchygan: Towards diverse and realistic sketch to image synthesis." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.
http://openaccess.thecvf.com/content_cvpr_2018/html/Chen_SketchyGAN_Towards_Diverse_CVPR_2018_paper.html
- Sketchy Database: <http://sketchy.ey.gatech.edu/explore/rabbit.html>