CORNELL TECH

Stacked Generative Adversarial Networks

Xun Huang[1,2], Yixuan Li[2], Omid Poursaeed[1,2], John Hopcroft[2], Serge Belongie[1,2]

[1]Cornell Tech [2]Cornell University

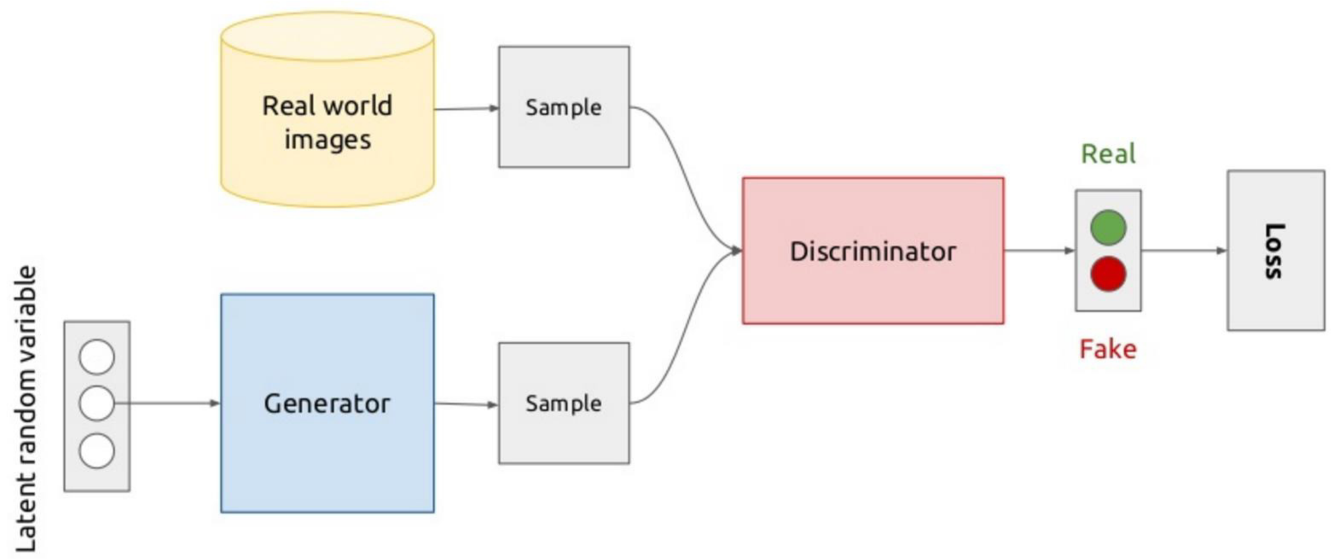
IEEE 2017 Conference on Computer Vision and Pattern Recognition

CVPR July 21-26 Honolulu 2017

Background

Generative Adversarial Networks (GAN):

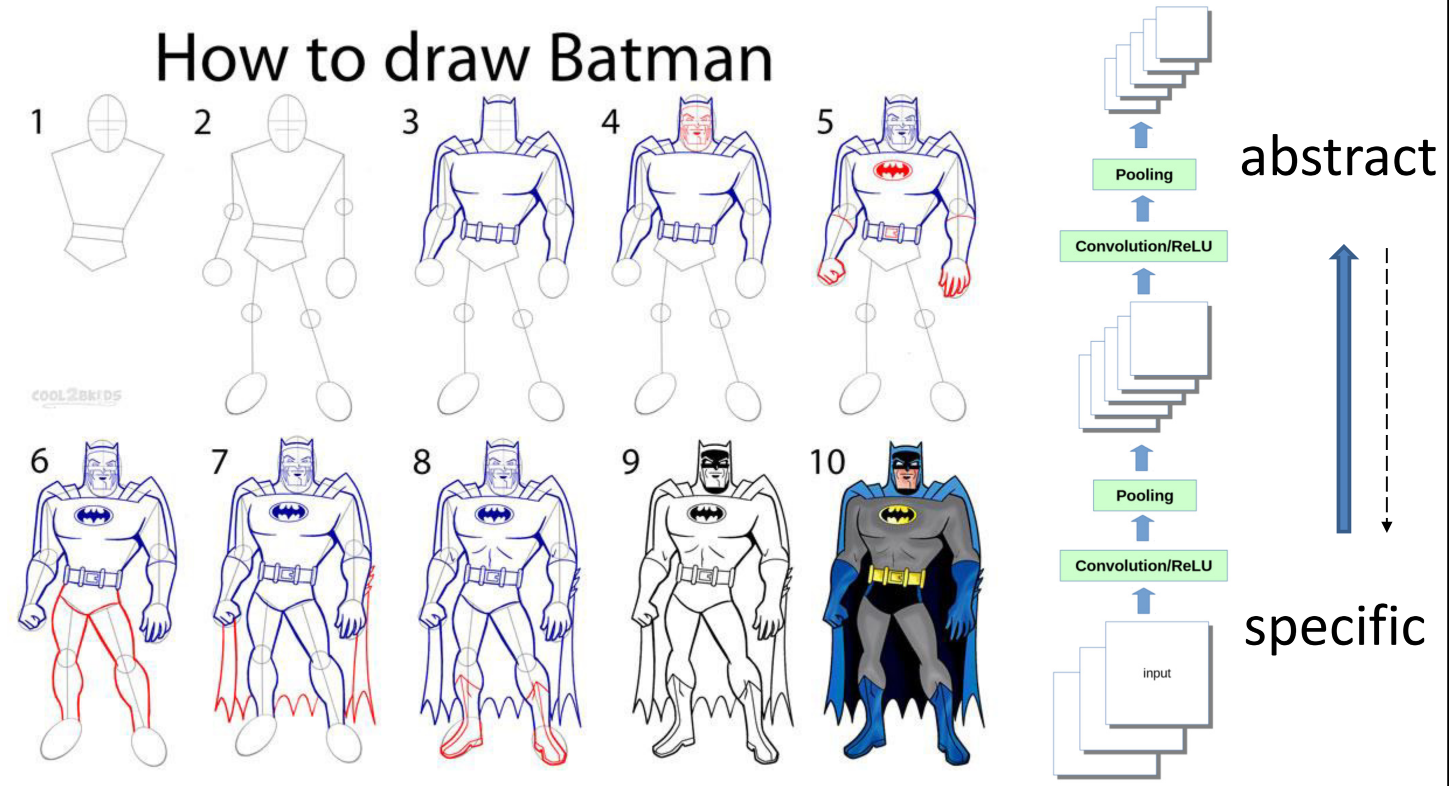
- Two networks competing with each other.
- Discriminator *D* tries to distinguish between real samples and samples generated by generator *G*.
- *G* tries to "fool" *D*.
- *G* will learn to generate samples similar to real data.



Motivation

Human painters usually first draw some abstract sketches, then gradually add details.

To mimic this process, we learn a generator that first produce high-level abstract features, then gradually generate lower level features and finally the image.



Architecture

A stack of GANs, each GAN generates lower-level features conditioned on higher-level features.

Each generator is trained with three loss terms:

- Adversarial loss: the generated features should be indistinguishable from "real" features.

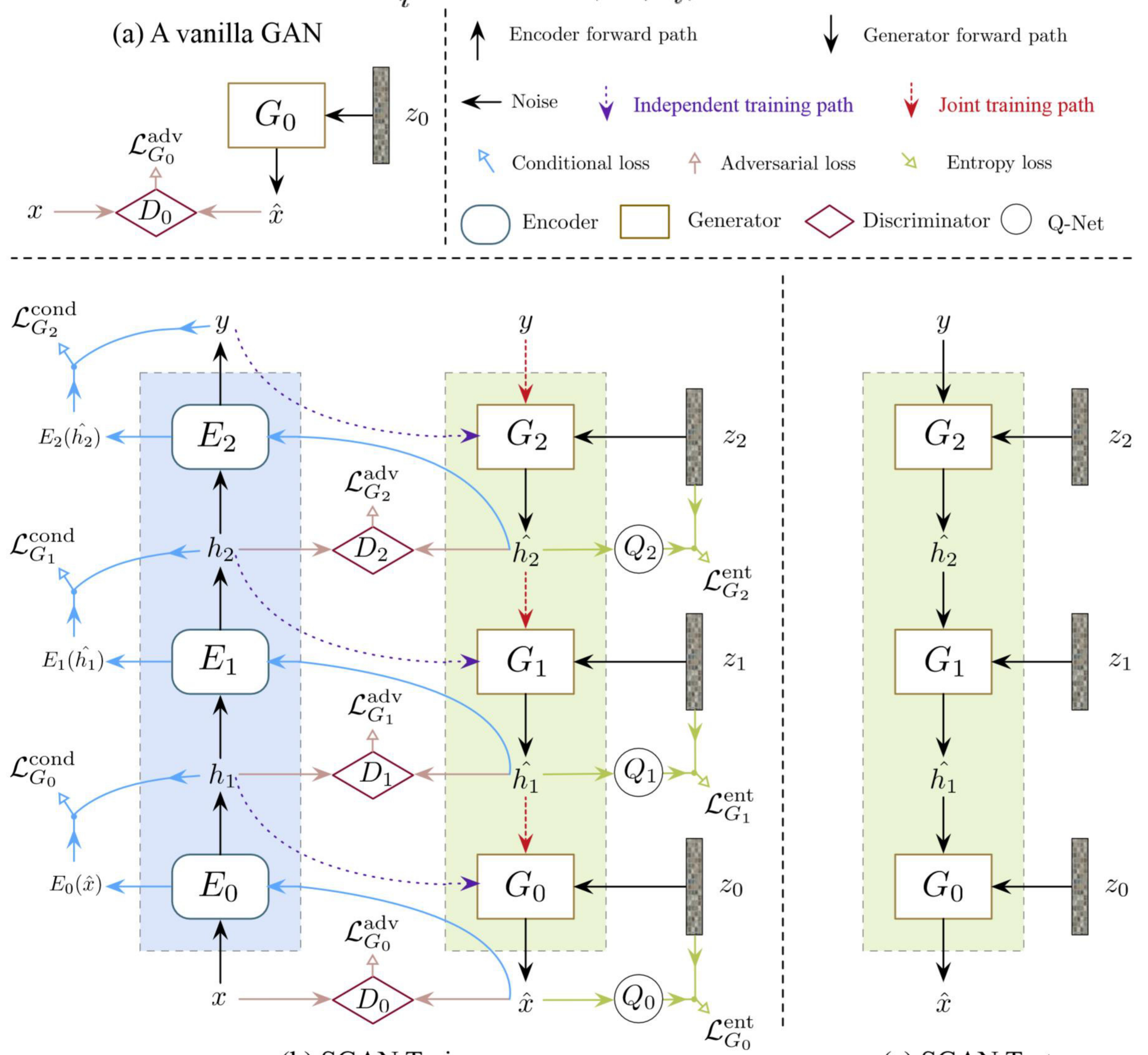
$$\mathcal{L}_{G_i}^{adv} = \mathbb{E}_{z_i \sim P_{z_i}, h_{i+1} \sim P_{data,E}}[-\log(D_i(G_i(h_{i+1}, z_i)))]$$

- Conditional loss: the generator should make use of the higher-level features it's conditioned on:

$$\mathcal{L}_{G_i}^{cond} = \mathbb{E}_{h_{i+1} \sim P_{data,E}, \hat{h}_i \sim P_G(\hat{h}_i | h_{i+1})}[f(E_i(\hat{h}_i), h_{i+1})]$$

- Entropy loss: encourage sample diversity by maximizing a variational lower bound on the entropy

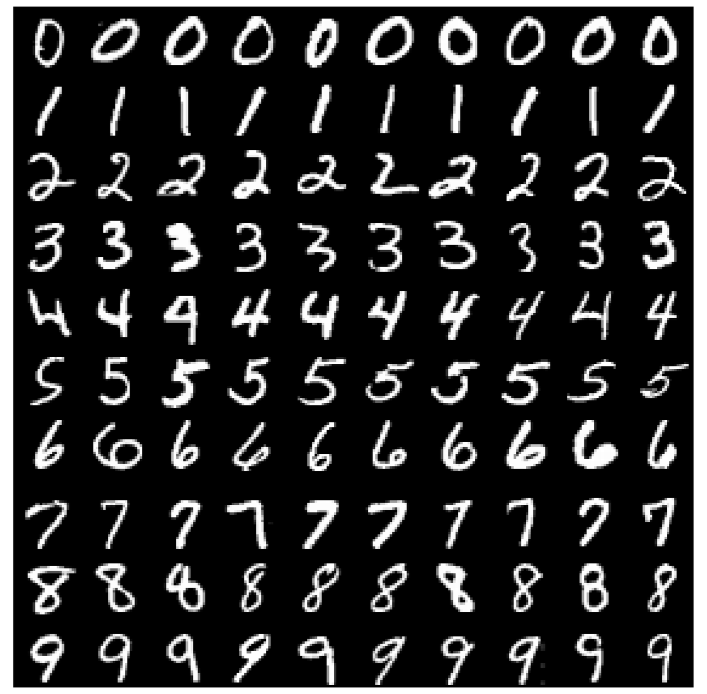
$$\mathcal{L}_{G_i}^{ent} = \mathbb{E}_{z_i' \sim P_{z_i'}}[\mathbb{E}_{\hat{h}_i \sim G_i(\hat{h}_i | z_i')}[-\log Q_i(z_i' | \hat{h}_i)]]$$



(a) A vanilla GAN (b) SGAN Train (c) SGAN Test

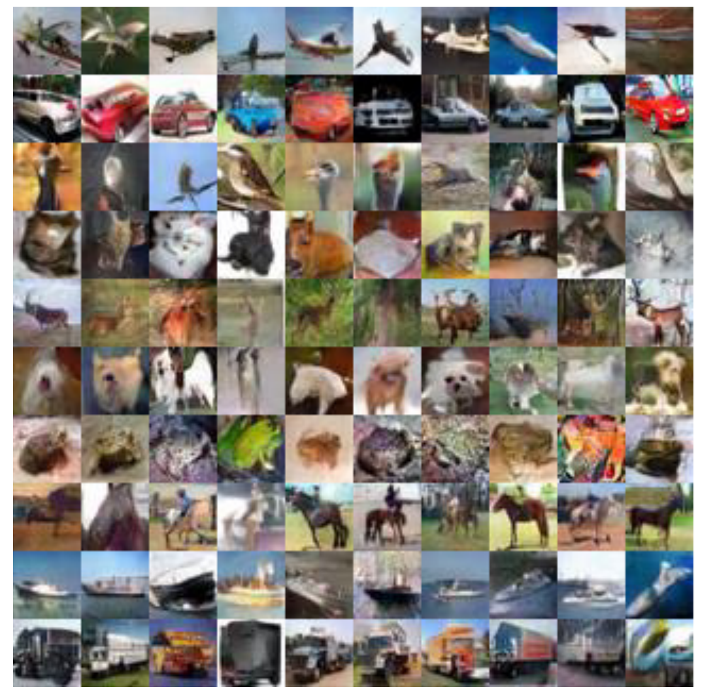
Qualitative results

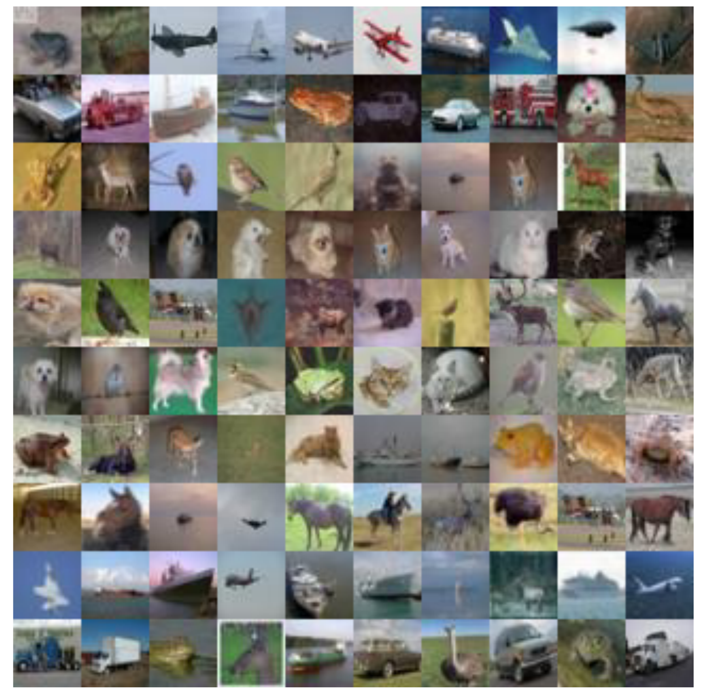

Generated


Real (nearest neighbor)


Generated


Real (nearest neighbor)


Generated


Real (nearest neighbor)

Quantitative evaluations

- Inception score on CIFAR-10:

Method	Score
Infusion training [1]	4.62 ± 0.06
ALI [10] (as reported in [63])	5.34 ± 0.05
GMAN [11] (best variant)	6.00 ± 0.19
EGAN-Ent-VI [4]	7.07 ± 0.10
LR-GAN [65]	7.17 ± 0.07
Denoising feature matching [63]	7.72 ± 0.13
DCGAN† (with labels, as reported in [61])	6.58
SteinGAN† [61]	6.35
Improved GAN† [53] (best variant)	8.09 ± 0.07
AC-GAN† [43]	8.25 ± 0.07
DCGAN ($\mathcal{L}^{adv}$)	6.16 ± 0.07
DCGAN ($\mathcal{L}^{adv} + \mathcal{L}^{ent}$)	5.40 ± 0.16
DCGAN ($\mathcal{L}^{adv} + \mathcal{L}^{cond}$)†	5.40 ± 0.08
DCGAN ($\mathcal{L}^{adv} + \mathcal{L}^{cond} + \mathcal{L}^{ent}$)†	7.16 ± 0.10
SGAN-no-joint†	$\mathbf{8.37} \pm 0.08$
SGAN†	$\mathbf{8.59} \pm 0.12$
Real data	11.24 ± 0.12

† Trained with labels.

- Human visual Turing tests on CIFAR-10: We ask AMT workers to distinguish generated images from real images. Our samples "fool" people **24.4%** of the time, higher than our best DCGAN baseline (15.6%) and Improved GAN (21.3%).

GitHub