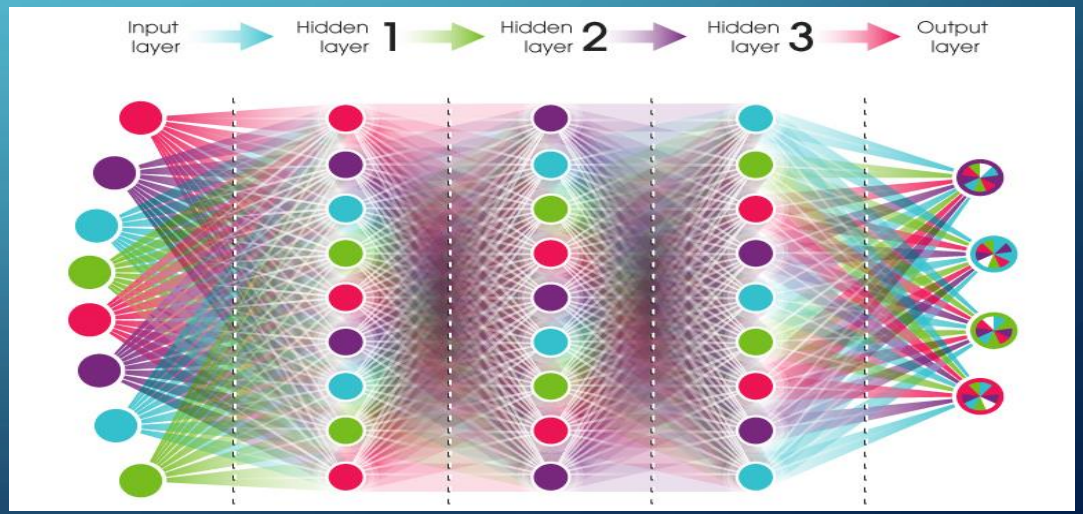
شبکه‌ی تولیدکننده‌ی مقابله‌ای

دانشکده‌ی مهندسی پزشکی
دانشگاه صنعتی امیرکبیر (پلی‌تکنیک)

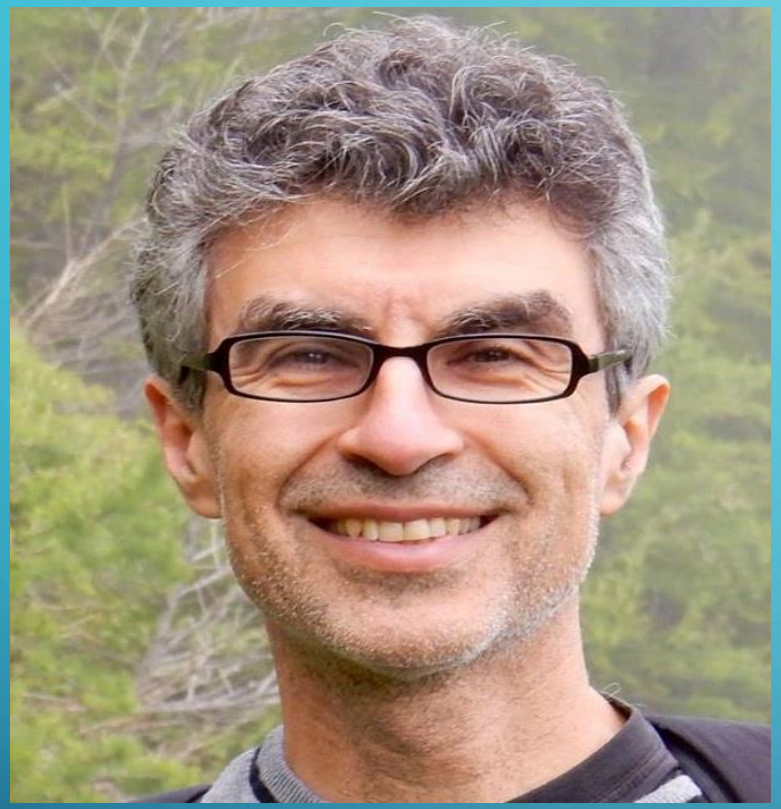
سینا مهدی پور
فاطمه قزلو
شبنم سلطان‌پور

مقدمه

- شبکه‌های عصبی
- یادگیری عمیق



ایده‌ی GAN

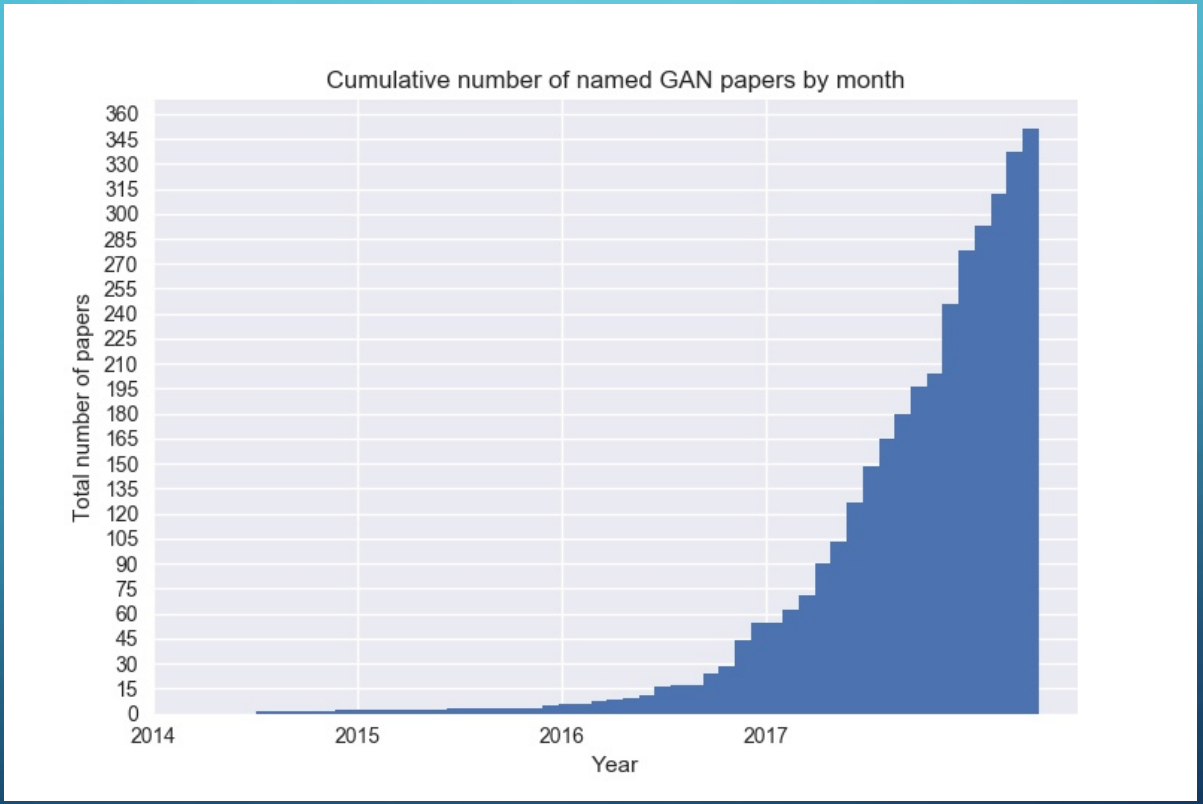


Yoshua Bengio

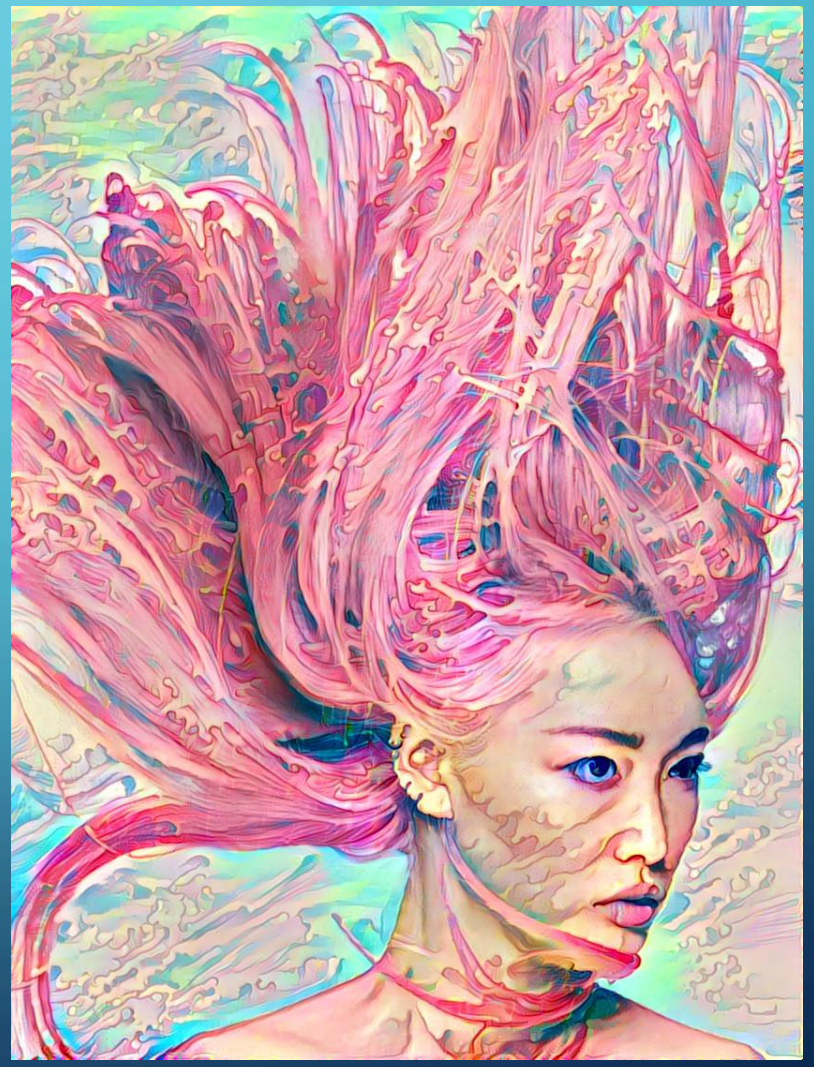
Ian Goodfellow

فراگیر شدن GANs

شبکه‌های GAN پتانسیل بالایی دارند زیرا می‌توانند هر توزیعی از داده را تقلید کنند.
به این معنی که این شبکه‌ها می‌توانند تصاویر، آهنگ‌ها و سخنانی مشابه نمونه‌های موجود در دنیای واقعی را تولید کنند.



DEEPDREAMGENERATOR.COM



افزایش رزولوشن تصویر

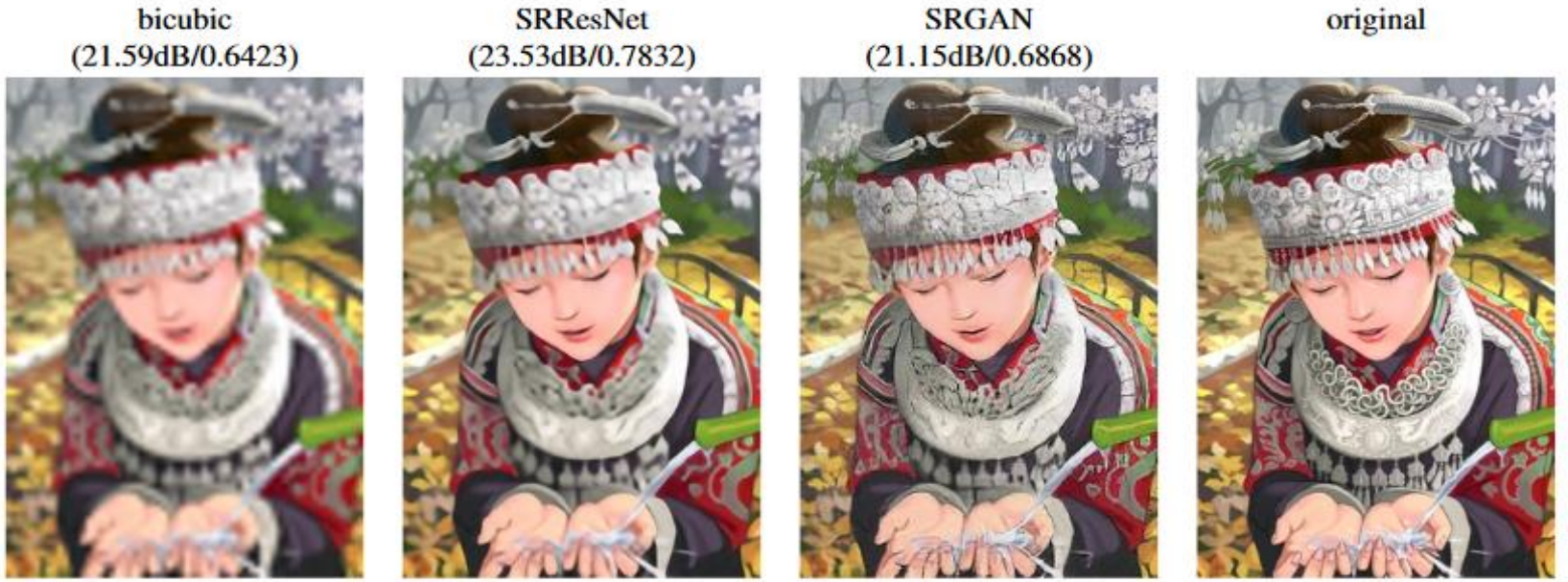
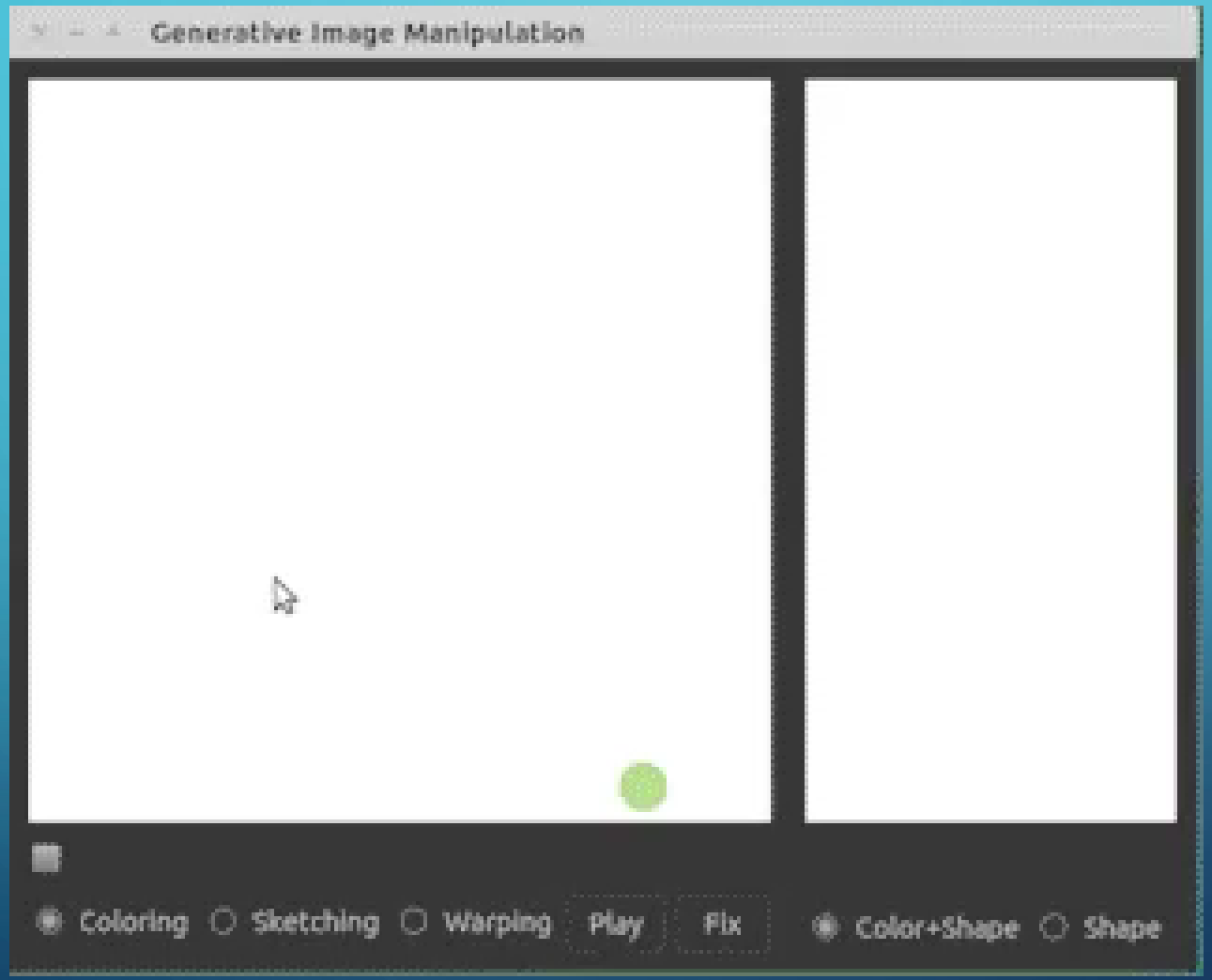
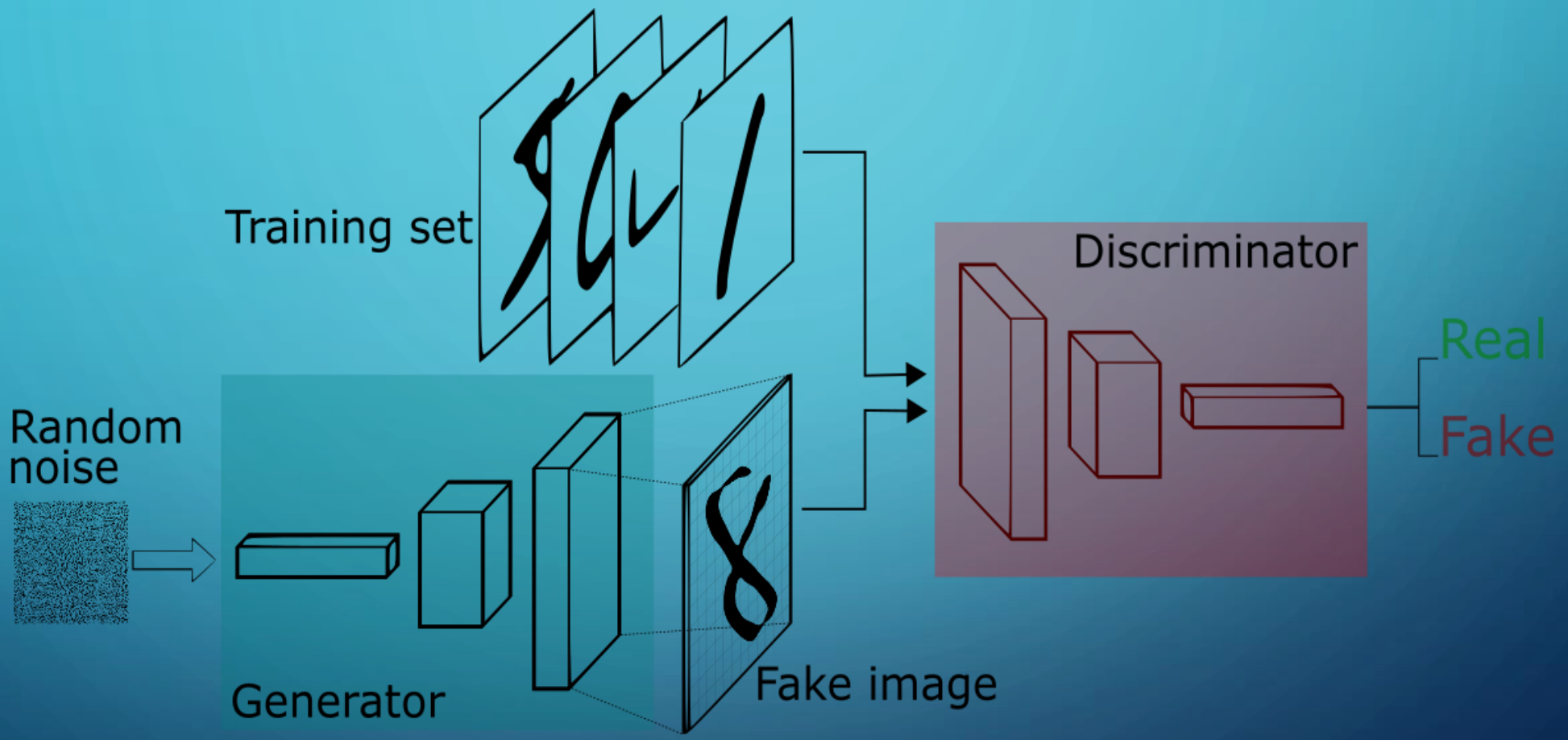


Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]

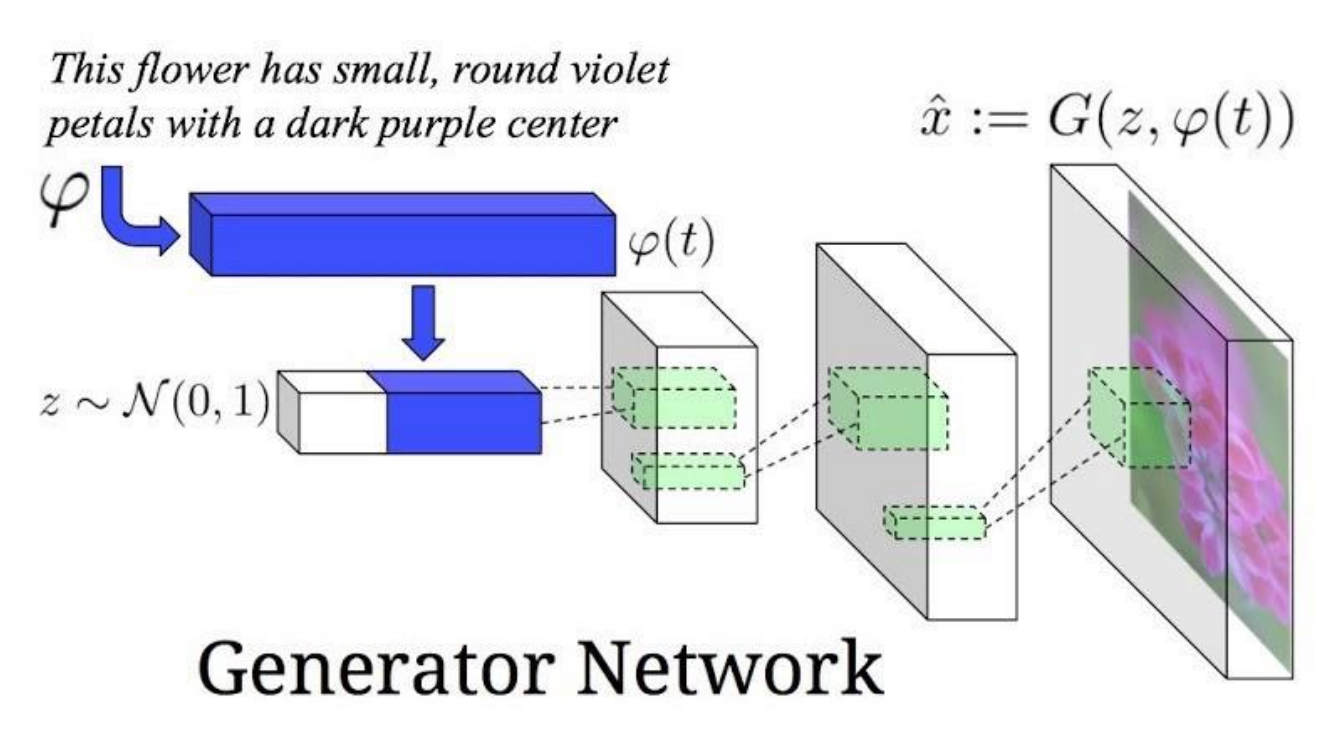
تولید تصویر


Generative Image Manipulation
Coloring
Sketching
Warping
Play
Fix
Color+Shape
Shape

معرفی فریم ورک

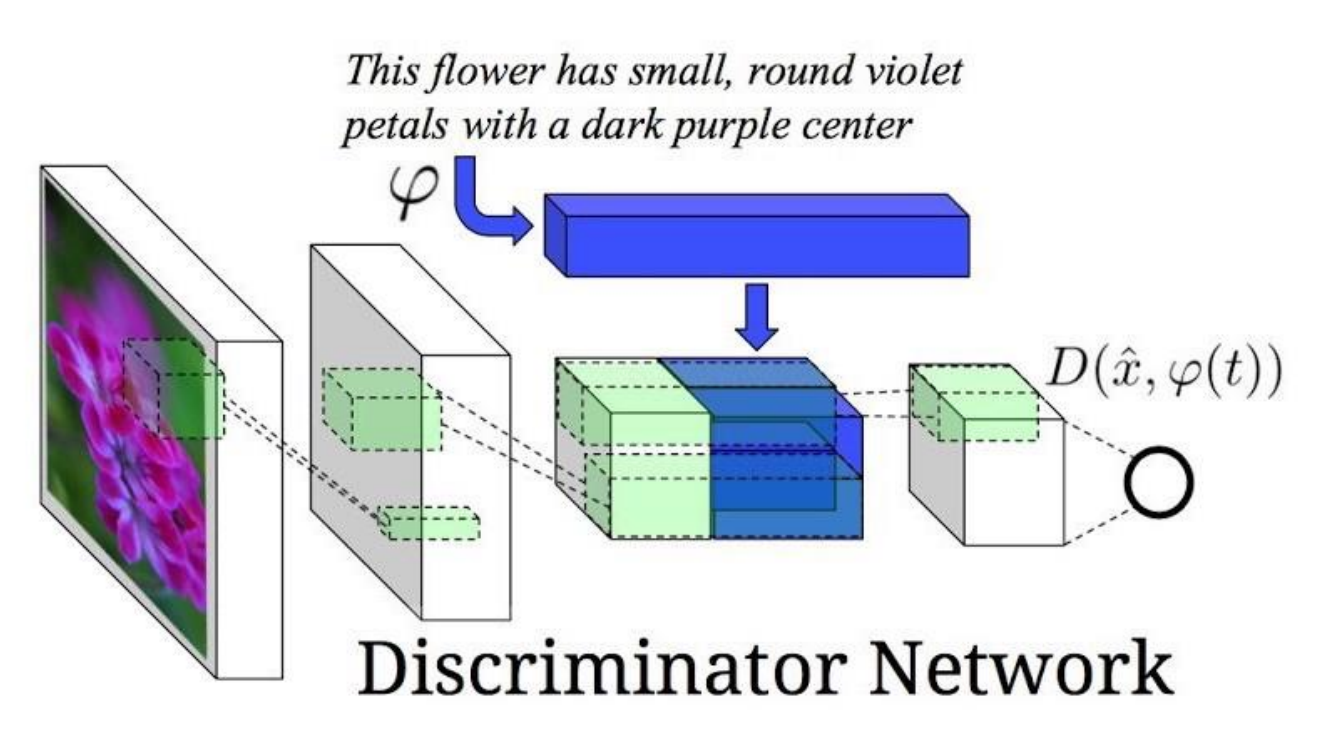


شبکه GENERATIVE



- پرسپترون چند لایه
- کمینه کردن log(1- D(G(z)))

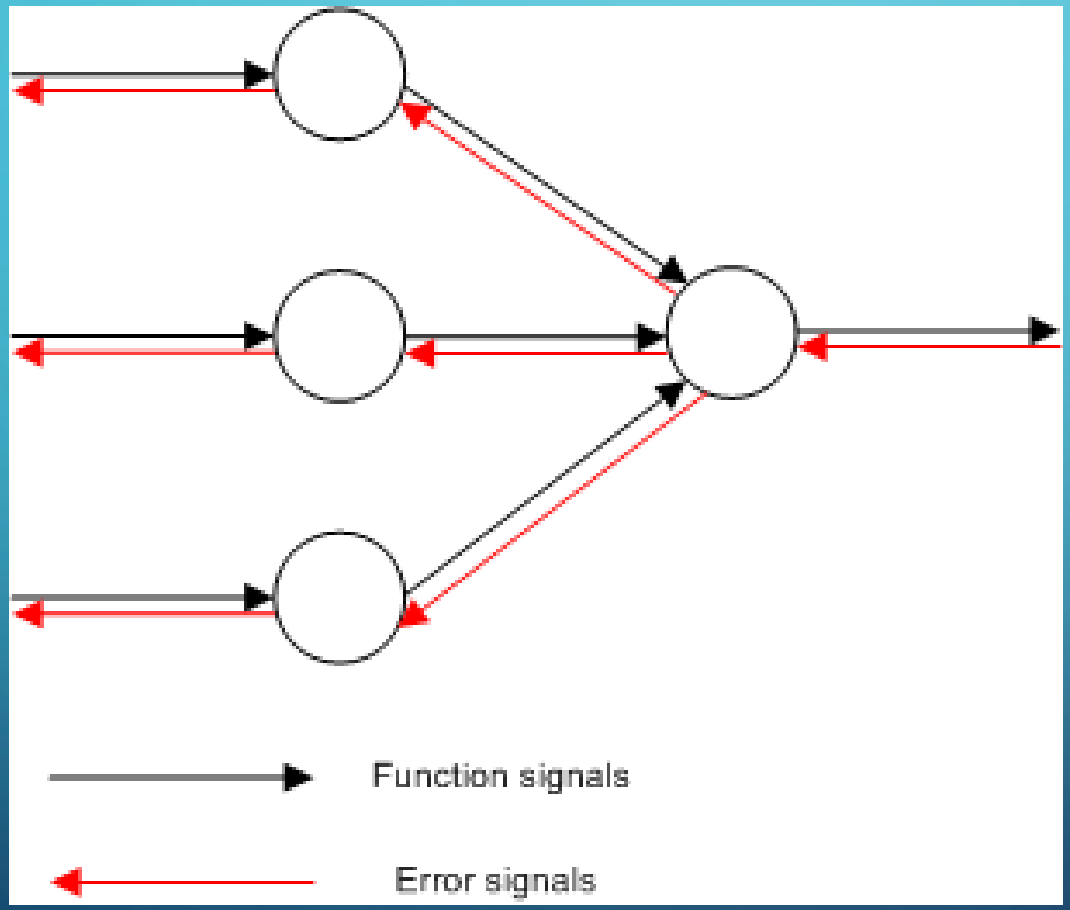
شبکه DISCRIMINATIVE



- پرسپترون چند لایه
- بیشینه کردن احتمال تشخیص درست

BACK PROPAGATION

روش تعلیم



عناصر شبکه

- تابع توزیع احتمال مقدم تصادفی $P_z(z)$
- نگاشت z به فضای داده‌ی ورودی $G(z, teta_g)$
  - G یک تابع مشتق پذیر حاصل از شبکه‌ی MLP تولیدکننده با پارامترهای $teta_g$
- تابع $D(x, teta_d)$
  - D یک عدد اسکالر خروجی می‌دهد
  - احتمال آمدن x از داده‌های واقعی به جای خروجی تابع تولیدکننده

هدف آموزش شبکه

- هدف آموزش تشخیص‌دهنده:

  افزایش احتمال تخصیص برچسب درست به هر دو گروه داده‌های واقعی و نمونه‌های تولید شده توسط G

- هدف آموزش تولیدکننده:

  کاهش احتمال تخصیص برچسب درست به داده‌های تولید شده توسط G

- تابع هدف کل شبکه:

  یک بازی **minimax** با دو بازیکن با مجموع صفر (منشا عنوان «مقابله‌ای» در اسم شبکه)

بازی MINIMAX در شبکه‌ی GAN

- تابع ارزش بازی:

$$\min_G \max_D V(D, G) = E_{x \sim Pdata(x)} \log(D(x)) + E_{z \sim Pz(z)} \log(1 - D(G(z)))$$

- هدف تشخیص‌دهنده (D) در بازی:
  - بیشینه کردن مقدار تابع ارزش

$$D(x) = 1 \text{ and } D(G(z)) = 0$$

- هدف تولیدکننده (G) در بازی:
  - کمینه کردن مقدار تابع ارزش (عدم وابستگی ترم اول به G)

$$D(G(z)) = 1$$

چالش‌های آموزش شبکه

- Overfit شدن شبکه در صورت آموزش کامل D بدون در نظر گرفتن G
  K مرحله آموزش D و یک مرحله آموزش G
- عدم تولید گرادیان‌های کافی در ابتدای آموزش G برای کمینه کردن log(1 - D(G(z))) در عمل
  بیشینه کردن log(D(G(z))) به عنوان جایگزین برای رفع مشکل

الگوریتم آموزش شبکه

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

  for k steps **do**

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
  - Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

  end for

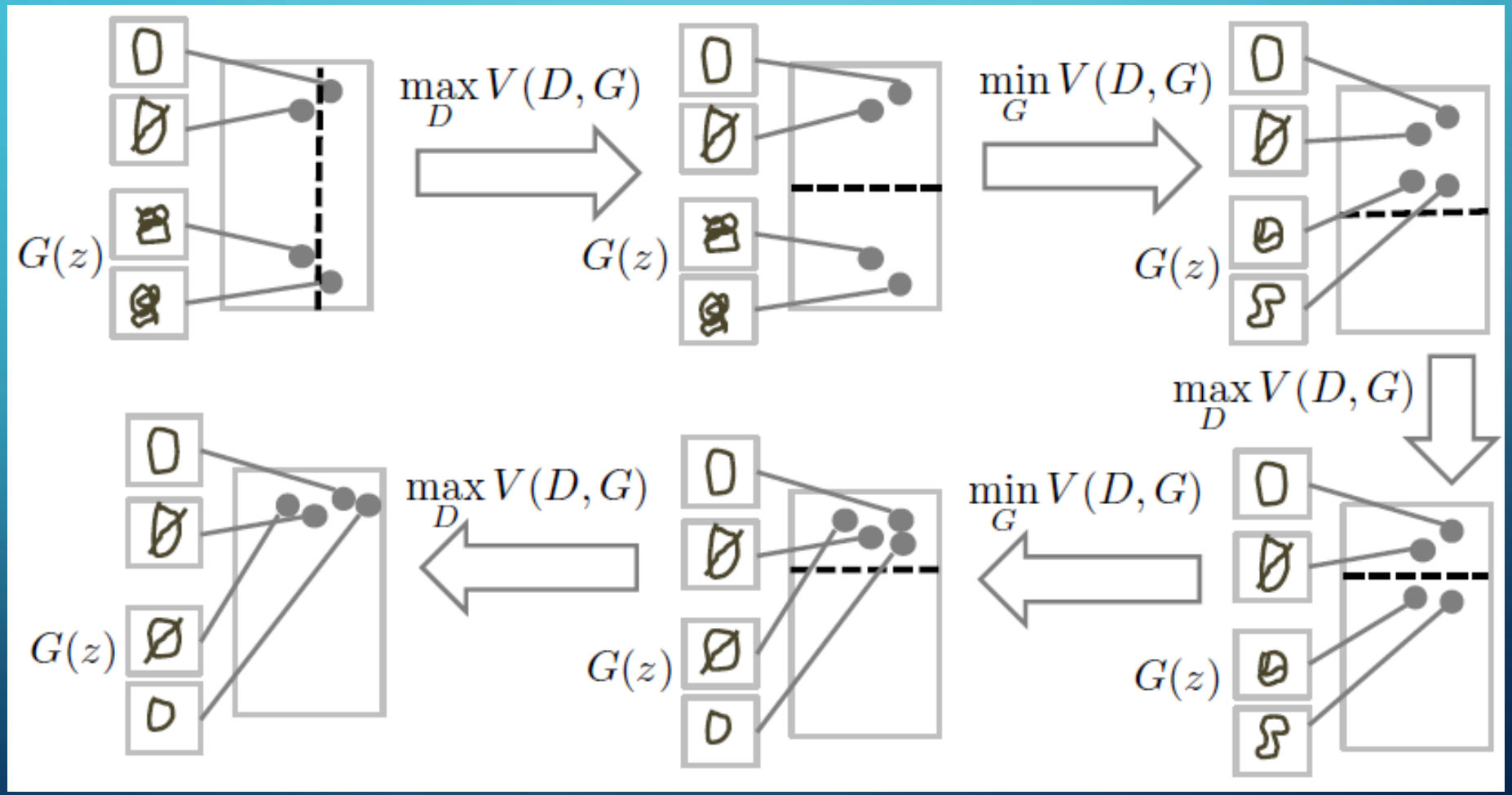
  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

آموزش شبکه



بهینه‌ی سراسری

- باید نشان دهیم که بهینه‌ی سراسری برای الگوریتم آموزش شبکه وجود دارد و این الگوریتم به سمت آن بهینه هم‌گرا می‌شود.
- مقدار D در حالت بهینه‌ی شبکه‌ی GAN:
  - حالت بهینه: عدم توانایی D در تشخیص داده‌های تولید شده از داده‌های واقعی
  - یعنی برقراری عبارت زیر:

$$D(x) = 0.5; \text{ for all } x\text{'s from } P_{data} \text{ and } P_z$$

اثبات وجود بهینه‌ی سراسری

- مقدار بهینه‌ی سراسری:

$$D^*_G(x) = \frac{P_{data}(x)}{P_{data}(x)+P_g(x)}$$

- رخ دادن این بهینه در:

$$P_{data}(x) = P_g(x)$$

- برای اثبات باید دو مورد بالا را اثبات کنیم.

محاسبه‌ی مقدار بهینه‌ی D

$$V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z))]$$

$$= \int_x p_{data}(x) log(D(x))dx + \int_z p_z(z) log(1-D(G(z)))dz$$

$$x = G(z) \Rightarrow z = G^{-1}(x) \Rightarrow dz = (G^{-1})'(x)dx$$

$$\Rightarrow p_g(x) = p_z(G^{-1}(x))(G^{-1})'(x)$$

$$= \int_x p_{data}(x) log(D(x))dx + \int_x p_z(G^{-1}(x)) log(1-D(x))(G^{-1})'(x)dx$$

$$= \int_x p_{data}(x) log(D(x))dx + \int_x p_g(x) log(1-D(x))dx$$

$$= \int_x p_{data}(x) log(D(x)) + p_g(x) log(1-D(x))dx$$

محاسبه‌ی مقدار بهینه‌ی D (ادامه)

$$\max_D V(D,G) = \max_D \int_x p_{data}(x)log(D(x)) + p_g(x)log(1-D(x))dx$$

$$\frac{\partial}{\partial D(x)}(p_{data}(x)log(D(x)) + p_g(x)log(1-D(x))) = 0$$

$$\Rightarrow \frac{p_{data}(x)}{D(x)} - \frac{p_g(x)}{1-D(x)} = 0$$

$$\Rightarrow D(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$$

اثبات رخ دادن بهینه در $P_{DATA}(X) = P_G(X)$

$$
\begin{aligned}
C(G) &= \max_D V(G, D) \\
&= \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx \\
&= \int_x p_{data}(x) log(D^*_G(x)) + p_g(x) log(1 - D^*_G(x)) dx \\
&= \int_x p_{data}(x) log(\frac{p_{data}(x)}{p_{data}(x) + p_g(x)}) + p_g(x) log(\frac{p_g(x)}{p_{data}(x) + p_g(x)}) dx \\
&= \int_x p_{data}(x) log(\frac{p_{data}(x)}{\frac{p_{data}(x)+p_g(x)}{2}}) + p_g(x) log(\frac{p_g(x)}{\frac{p_{data}(x)+p_g(x)}{2}}) dx - log(4) \\
&= KL[p_{data}(x) || \frac{p_{data}(x) + p_g(x)}{2}] + KL[p_g(x) || \frac{p_{data}(x) + p_g(x)}{2}] - log(4)
\end{aligned}
$$

اثبات رخ دادن بهینه در $P_{DATA}(X) = P_G(X)$ (ادامه)

$$C(G) = \underbrace{KL[p_{data}(x)||\frac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} + \underbrace{KL[p_g(x)||\frac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} - log(4)$$

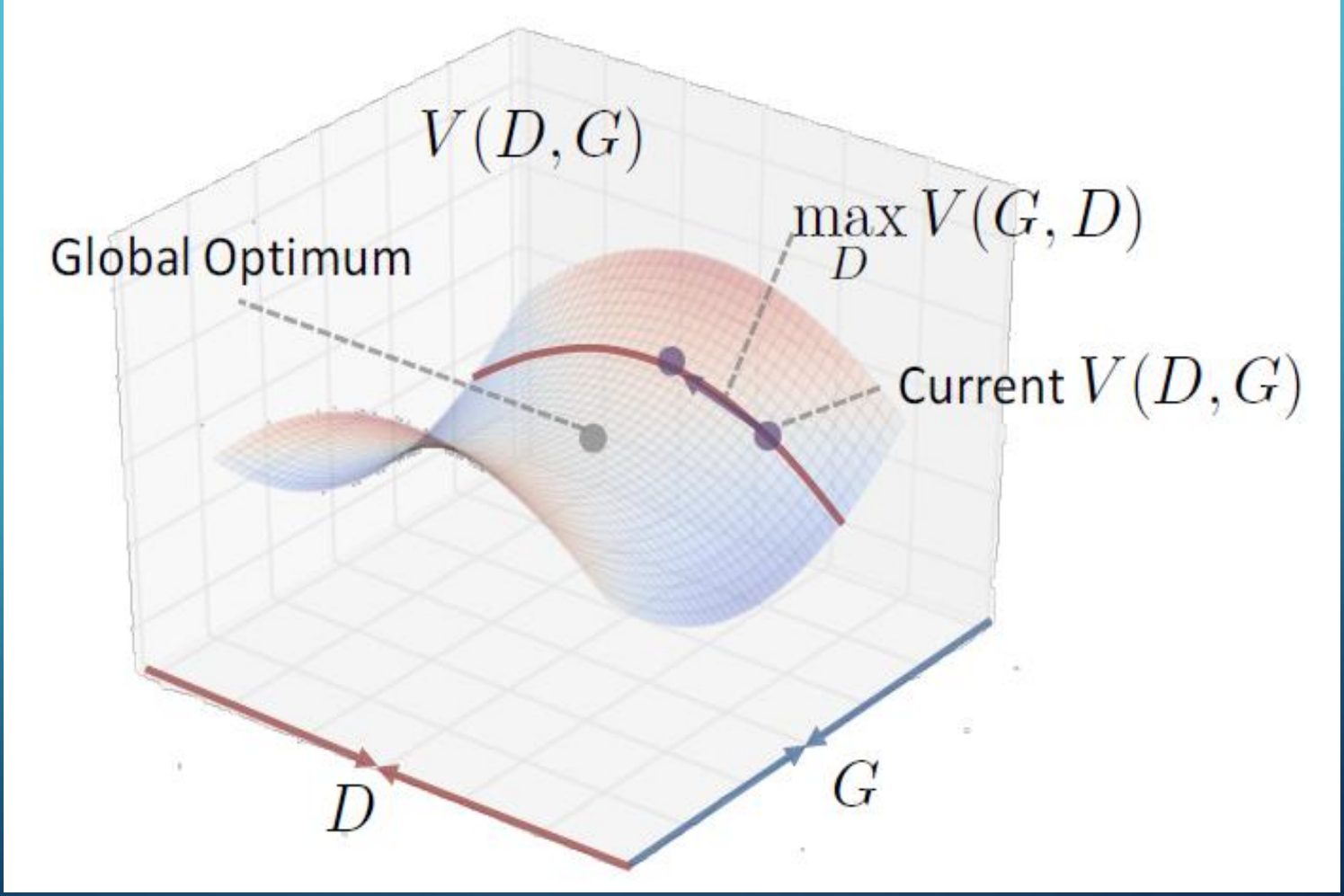
$$\min_G C(G) = 0 + 0 - log(4) = -log(4)$$

$$KL[p_{data}(x)||\frac{p_{data}(x)+p_g(x)}{2}] = 0$$

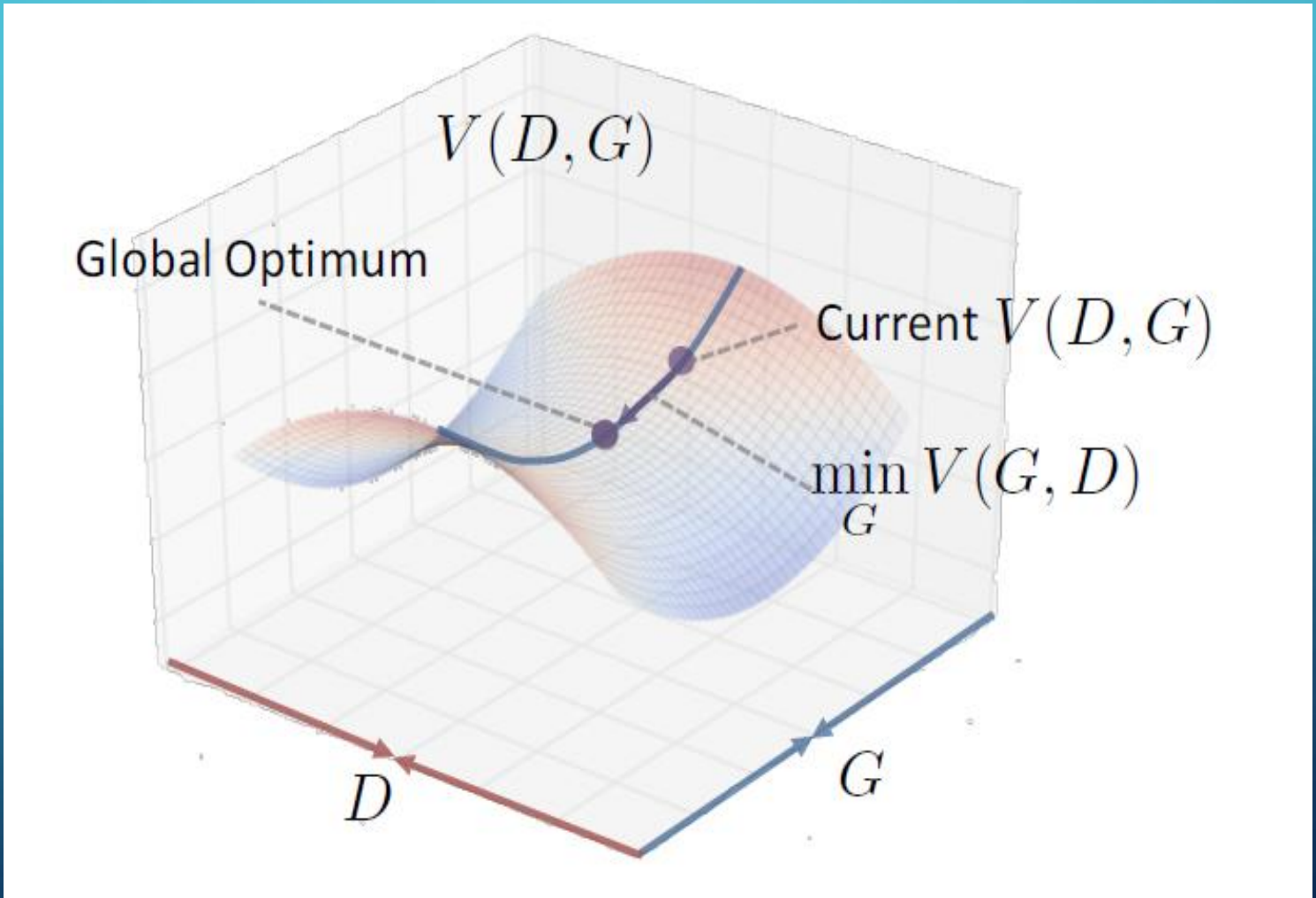
$$\text{when} \quad p_{data}(x) = \frac{p_{data}(x)+p_g(x)}{2}$$

$$\Rightarrow p_{data}(x) = p_g(x)$$

هم‌گرایی به بهینه‌ی سراسری



هم‌گرایی به بهینه‌ی سراسری (ادامه)

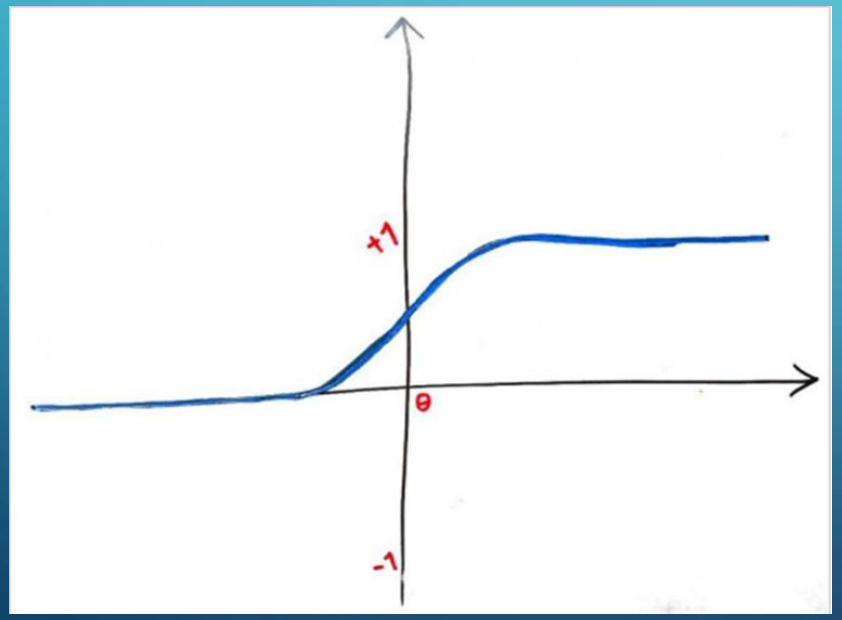


آزمایش‌ها

- این شبکه‌های عصبی را با استفاده از سه مجموعه‌ی داده‌ی MNIST و TFD و CIFAR–10 آموزش می‌دهند.
- شبکه‌های عصبی دارای بخشی به نام تابع فعال‌ساز هستند که ورودی آن یک عدد کوچک یا بزرگ در بازه‌ی دلخواه است و خروجی آن یک عدد بین ۰ و ۱ و یا ۱- و ۱+ است. در واقع این توابع یک عدد ورودی را به یک بازه‌ی مشخص تبدیل می‌کنند.
- این توابع تصمیم می‌گیرند که آیا یک نرون خاص در شبکه عصبی فعال شود یا خیر.
- تابع فعال ساز باید کران‌دار و مشتق‌پذیر باشد.

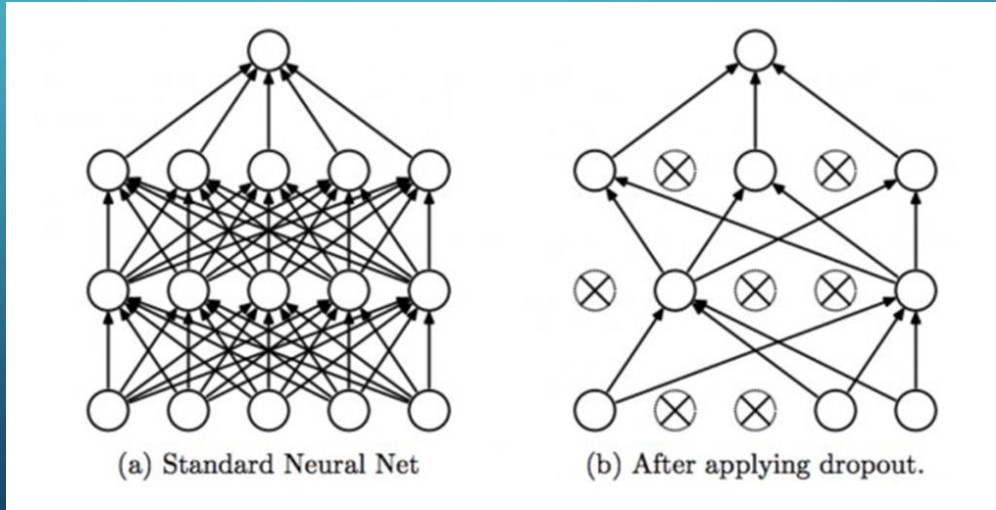
آزمایش‌ها

- تابع پرسپترون به تابع پله معروف است. یک تابع فعال‌ساز خطی است، ولی همیشه در شبکه‌های عصبی از تابع پرسپترون استفاده نمی‌شود. مثلا اگر شبکه‌ی عصبی بخواهد ویژگی‌های پیچیده‌تری را یاد بگیرد.
- یکی از این توابع Sigmoid است:

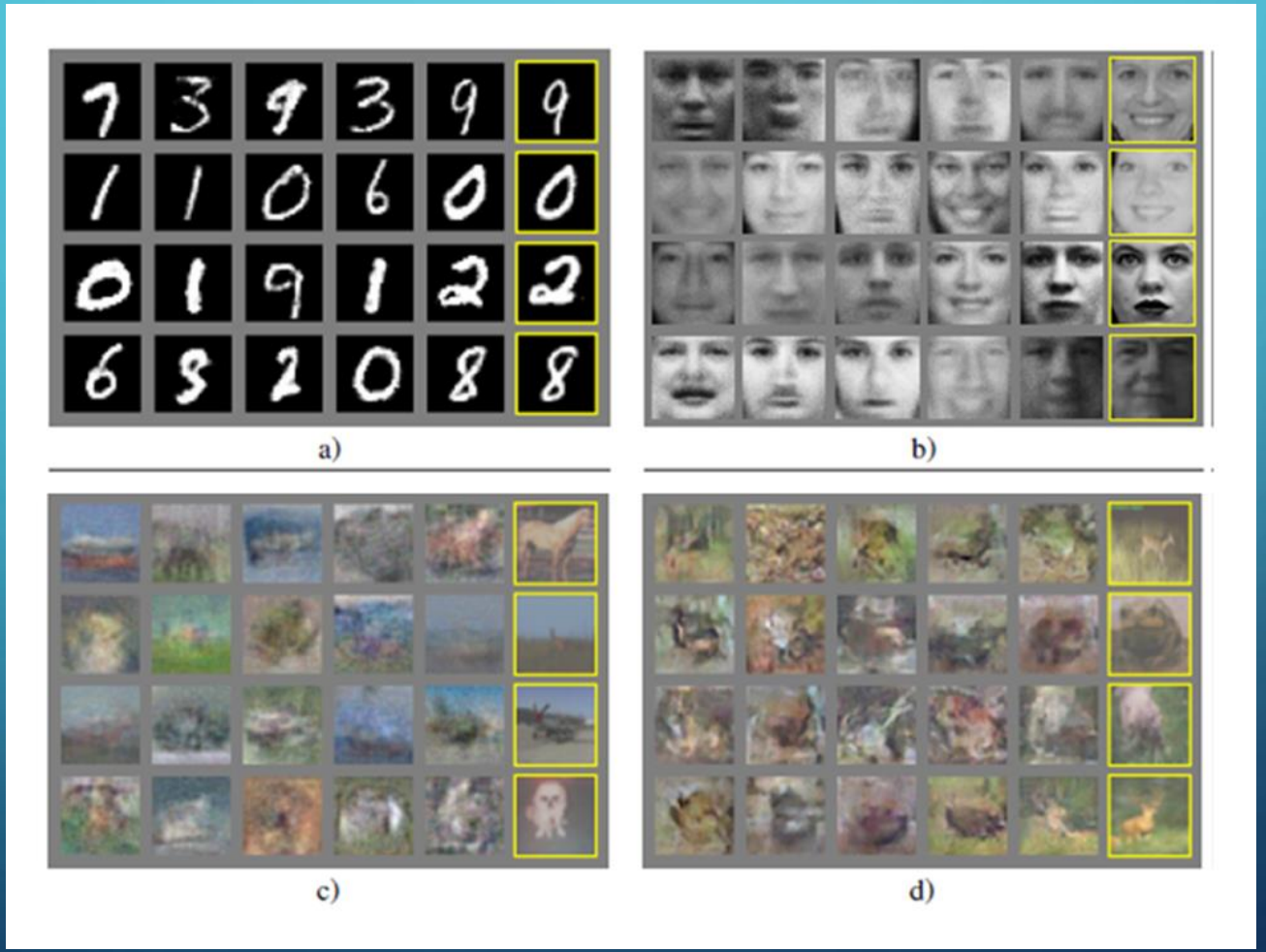


آزمایش‌ها

- شبکه‌ی عصبی G ازترکیب دو تابع فعال‌ساز خطی و **sigmoid** استفاده کرده است و شبکه‌ی عصبی D از تابع فعال‌ساز **Maxout** استفاده کرده است.
- برای آموزش از **dropout** برای حذف نویز از لایه‌های میانی شبکه‌ی عصبی G استفاده می‌کند.
  - **Dropout** به معنای حذف تصادفی می‌باشد.
  - به معنای آن که برخی نورون‌ها را چه آشکار باشند و چه پنهان، در هنگام آموزش به صورت تصادفی نادیده بگیریم.



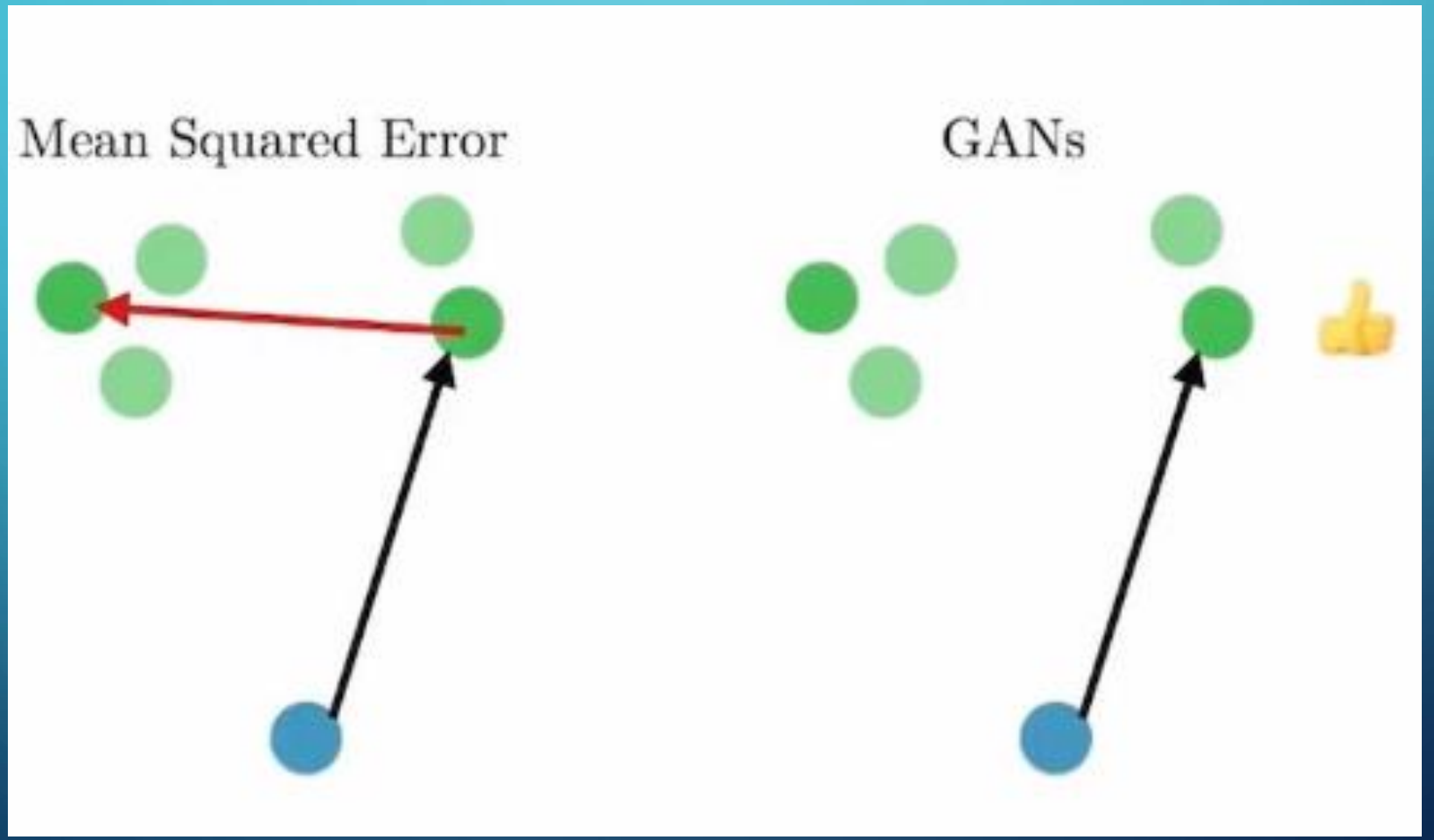
نمونه‌های آموزش دیده در سه مجموعه‌ی داده



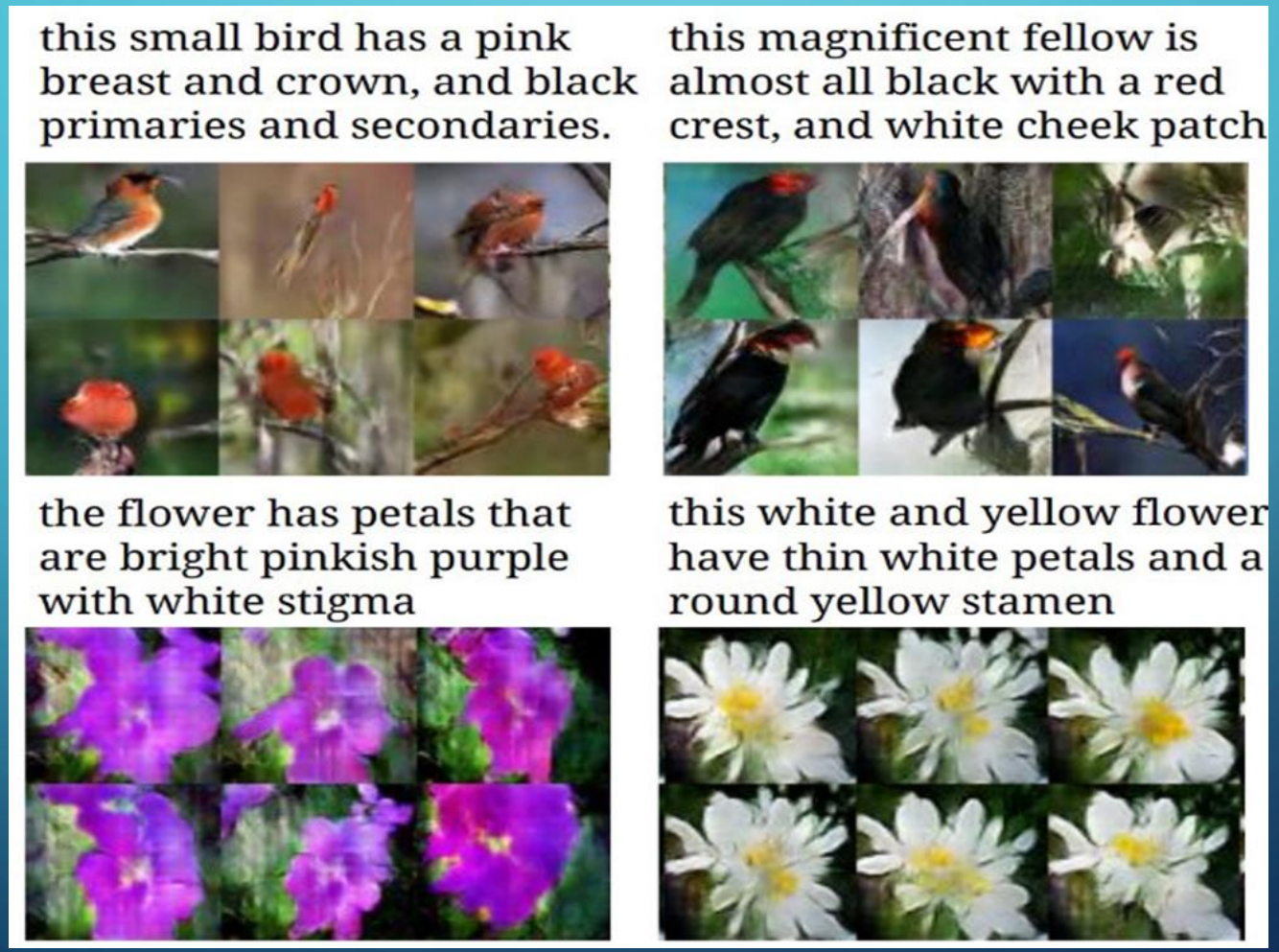
مقایسه با مدل‌های قبلی

	Deep directed graphical models	Deep undirected graphical models	Generative autoencoders	Adversarial models
Training	Inference needed during training.	Inference needed during training. MCMC needed to approximate partition function gradient.	Enforced tradeoff between mixing and power of reconstruction generation	Synchronizing the discriminator with the generator. Helvetica.
Inference	Learned approximate inference	Variational inference	MCMC-based inference	Learned approximate inference
Sampling	No difficulties	Requires Markov chain	Requires Markov chain	No difficulties
Evaluating $p(x)$	Intractable, may be approximated with AIS	Intractable, may be approximated with AIS	Not explicitly represented, may be approximated with Parzen density estimation	Not explicitly represented, may be approximated with Parzen density estimation
Model design	Nearly all models incur extreme difficulty	Careful design needed to ensure multiple properties	Any differentiable function is theoretically permitted	Any differentiable function is theoretically permitted

کاربرد: تشخیص تصویر از متن



مثال از کاربرد

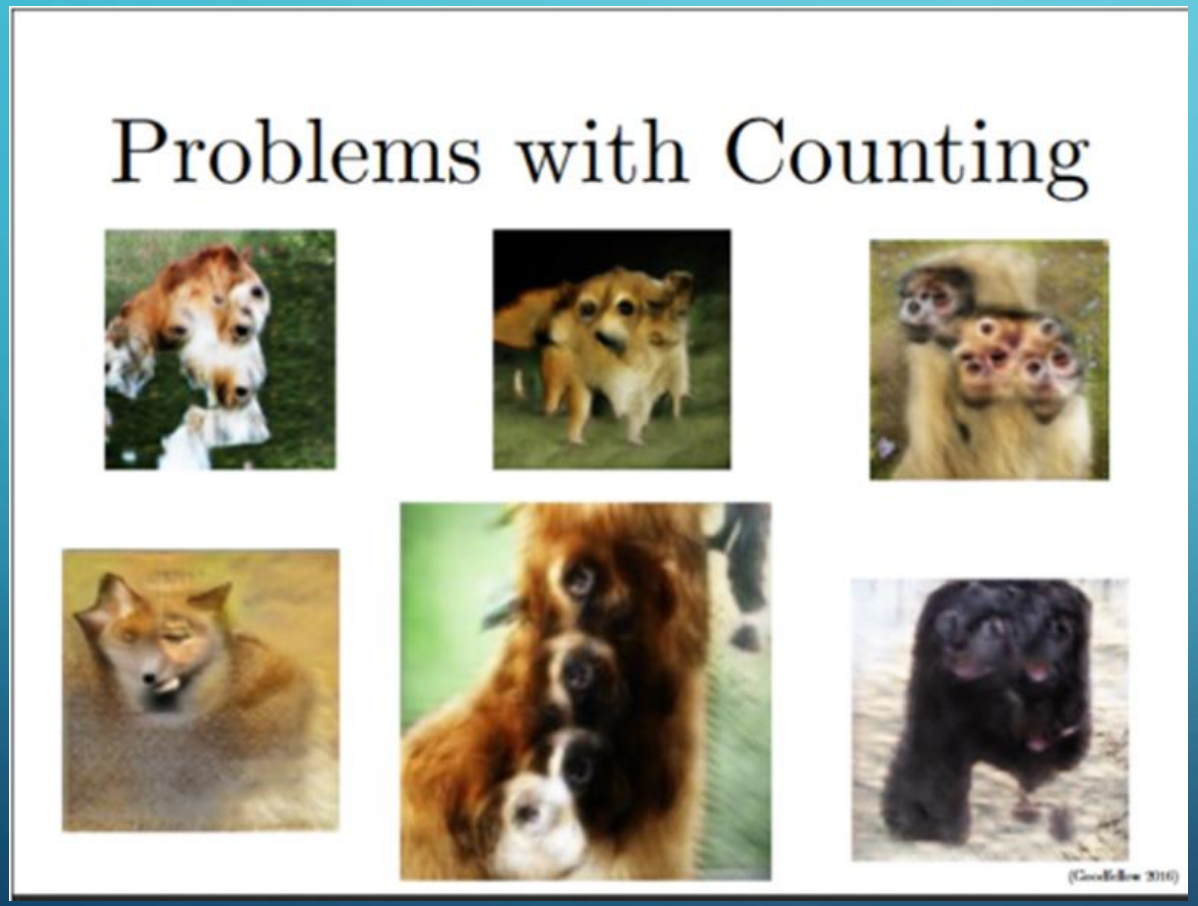


مثال از کاربرد

- مهم ترین مسئله در هنگام آموزش GAN پایداری است که راه های مختلفی برای نگاه کردن به این پایداری وجود دارد:

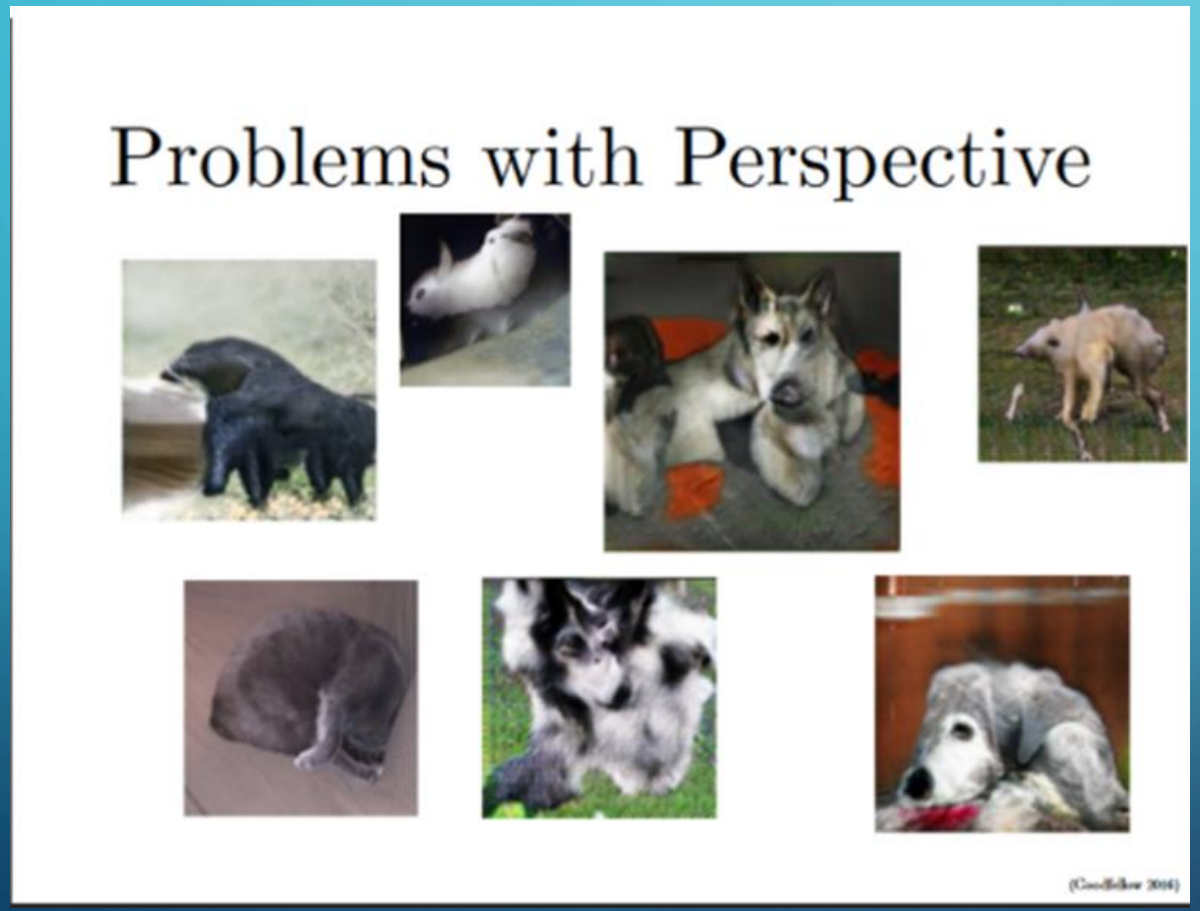
1. اگر بخش D قدرتمند باشد که همتای خود را تولید کند آنگاه بخش G قادر نخواهد بود به طور موثر آموزش دهد که این به نوبه‌ی خود در آموزش GAN تاثیر می‌گذارد و اگر هم بخش D بسیار آرام باشد نمی‌تواند جعلی بودن داده‌ی تولیدی G را تشخیص دهد و G آن داده‌ی تولیدی خود را داده‌ی خوبی خواهد دانست.در نتیجه عملکرد کلی سیستم مختل خواهد شد.
2. هر دو بخش D و G باهم مبارزه می‌کنند تا یک قدم از دیگری جلو بیفتد. همچنین آن‌ها به هم وابسته هستند. باید مطمئن شویم که اگر یکی از آن‌ها شکست بخورد کل سیستم شکست نمی‌خورد.

معایب



- GAN-ها قادر به تشخیص چند شی خاص در یک محل خاص نیستند.

معایب



- GAN-ها قادر به سازگاری با اشیا سه‌بعدی نیستند.

معایب

- GAN-ها یک ساختار جامع را درک نمی‌کنند.

مراجع

[١] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozairy, Aaron Courville, Yoshua Bengio, “Generative Adversarial Nets”. (Department of Computer Science and Operations Research, University of Montreal), IJCA Journal, Vol. 119 - Number 18, 2015

[٢] Faizan Shaikh, https://www.analyticsvidhya.com/blog/2017/06/introductory-generative-adversarial-networks-gans/, Accessed 2018

[٣] Mark Chang, https://www.slideshare.net/ckmarkohchang/generative-adversarial-networks, Accessed 2018

[٤] Mark Chang, https://https://www.youtube.com/playlist?list=PLeeHDpwX2Kj5Ugx6c9EfDLDojuQxnmxmU, Accessed 2018

پرسش؟؟؟

از توجه شما سپاس گزارم!