Generative Adversarial Networks

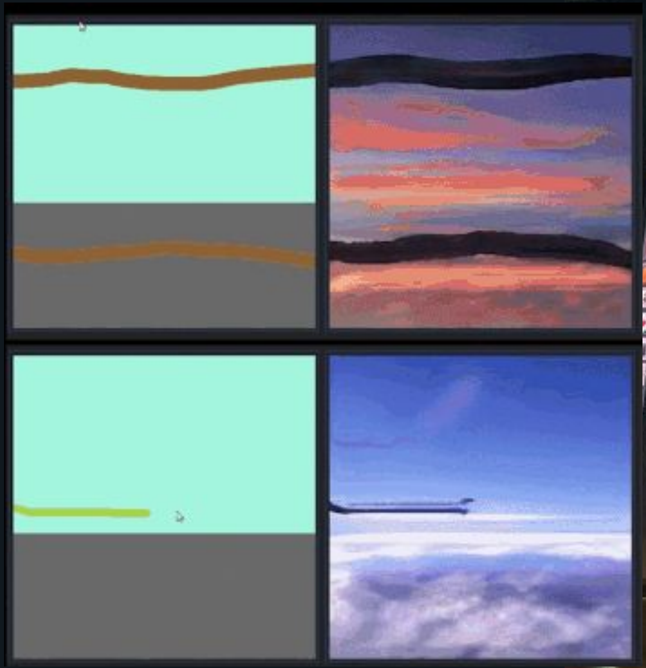
Max Li, Haoyang Li, Nathan Lui

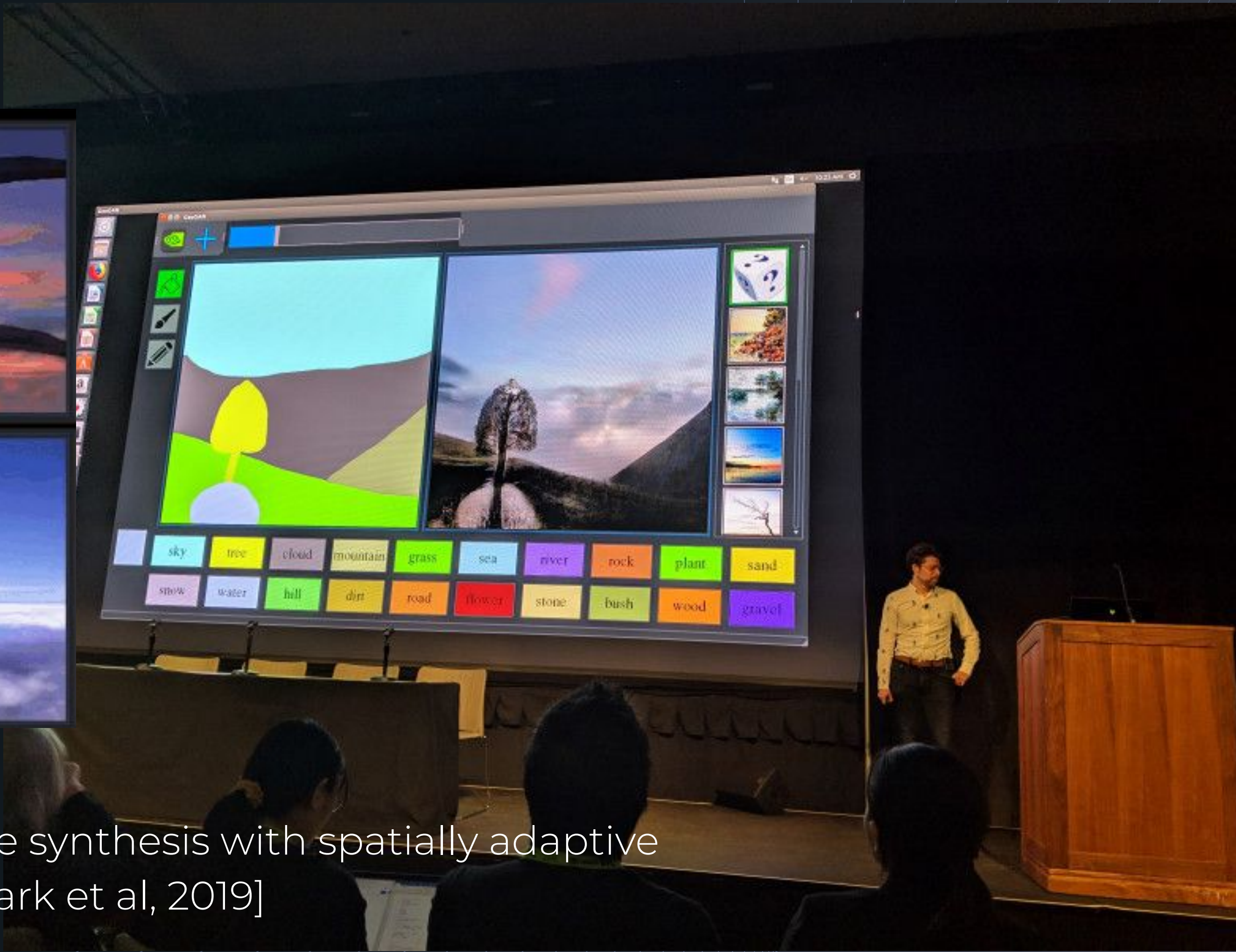
0.

Prelude

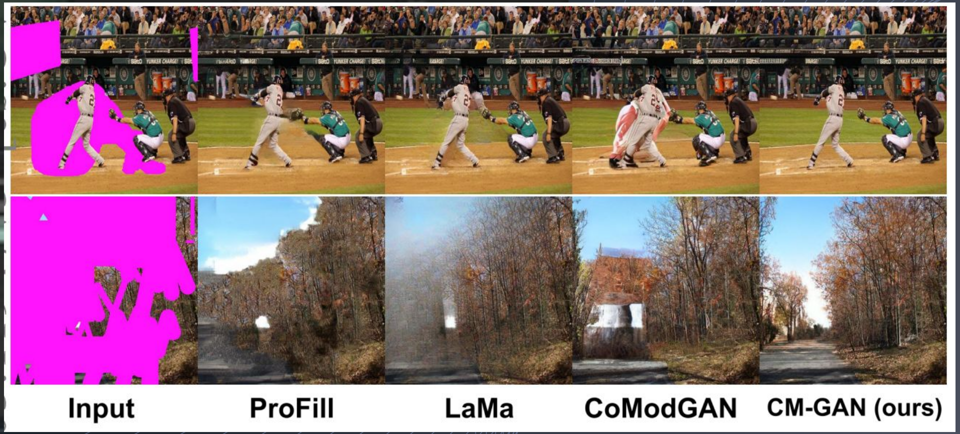
GANs today!

[Analyzing and Improving the Image Quality of StyleGAN - (Nvidia) Karras et. al, 2020]





[GauGAN: semantic image synthesis with spatially adaptive normalization - (Nvidia) Park et al, 2019]



[Image Inpainting with Cascaded Modulation GAN and Object-Aware Training - (Adobe) Zheng et al, 2019]

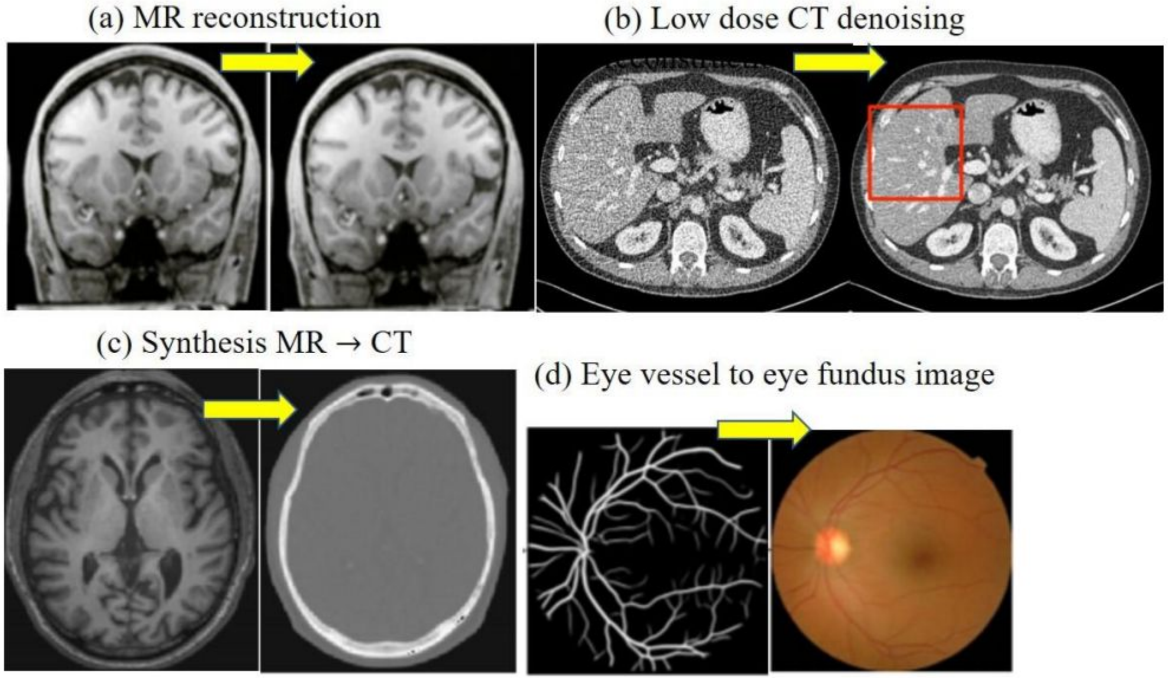
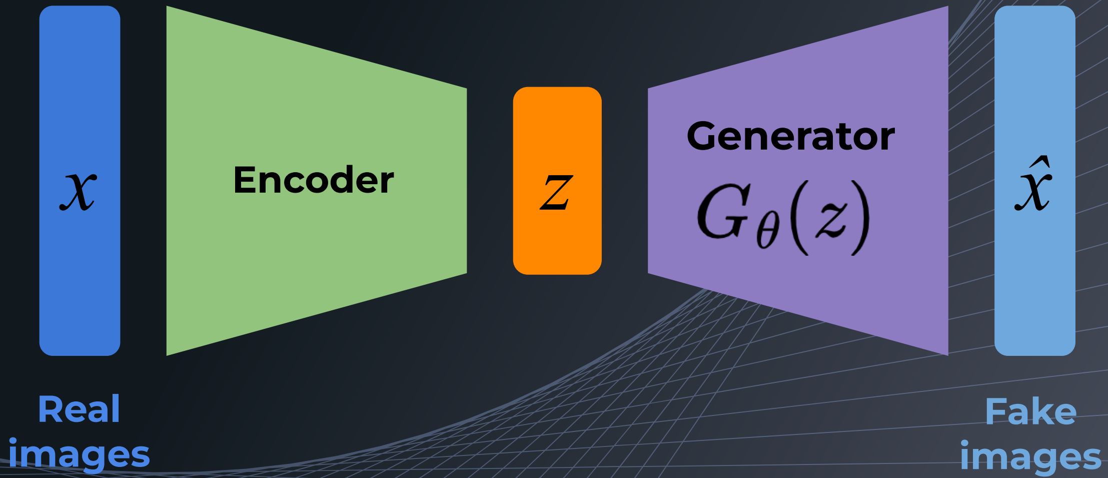


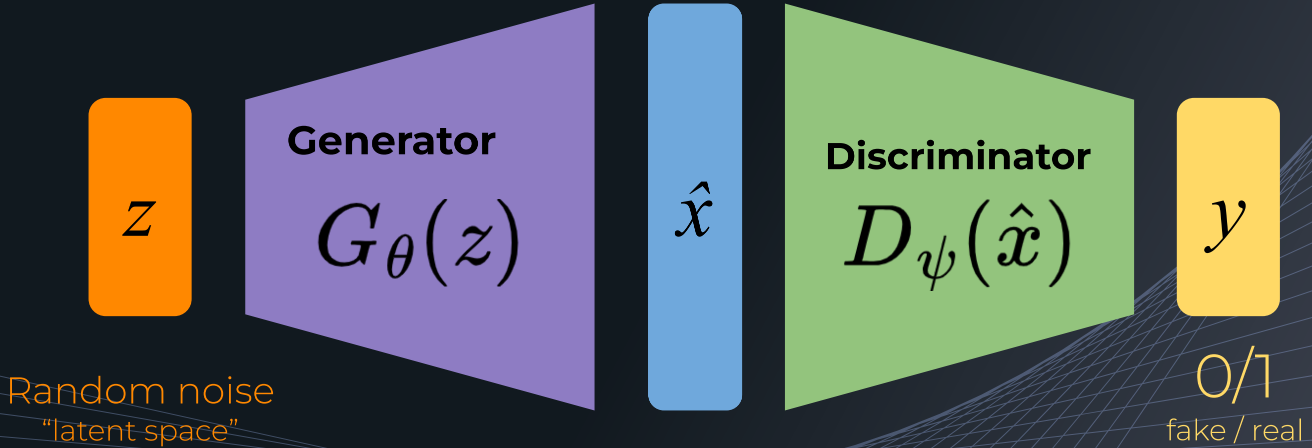
Figure 3. Application of GANin medical image synthesis. All the figures are adapted from corresponding articles. (a) shows MR reconstruction from given reference image (Chen et al., 2018) (b) Low dose CT denoising (Shan et al., 2018) (c)shows input brain MRI used to generated equivalent CT image close to ground truth(Nie et al., 2017) (d) generation of synthetic eye fundus image from corresponding synthetic eye vessels(Costa et al., 2018).

1.

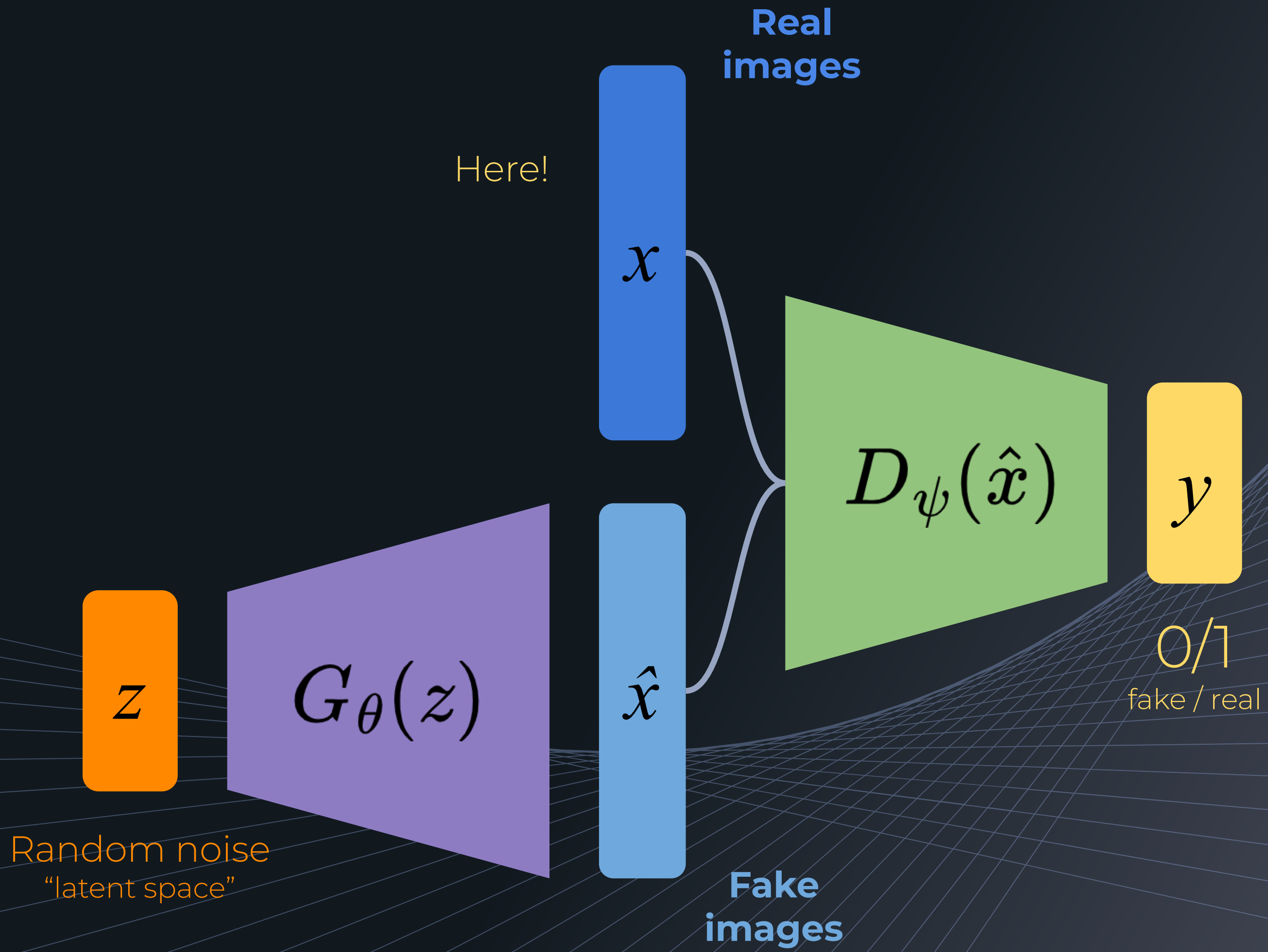
Ian Goodfellow goes to the bar

The inspiration for GANs

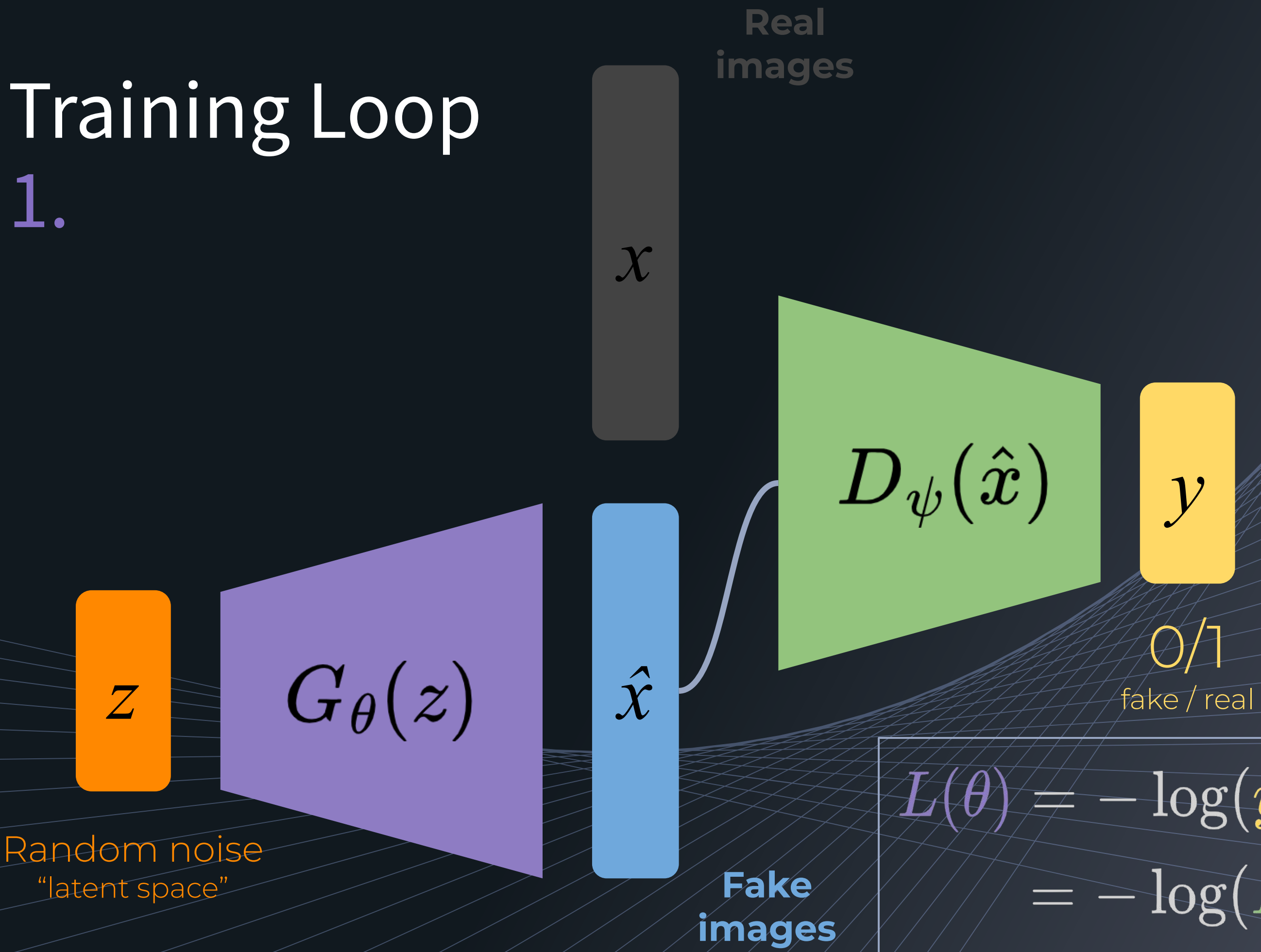

x
Encoder
z
Generator
$G_\theta(z)$
$\hat{x}$
Real images
Fake images



But where is correct labels?...

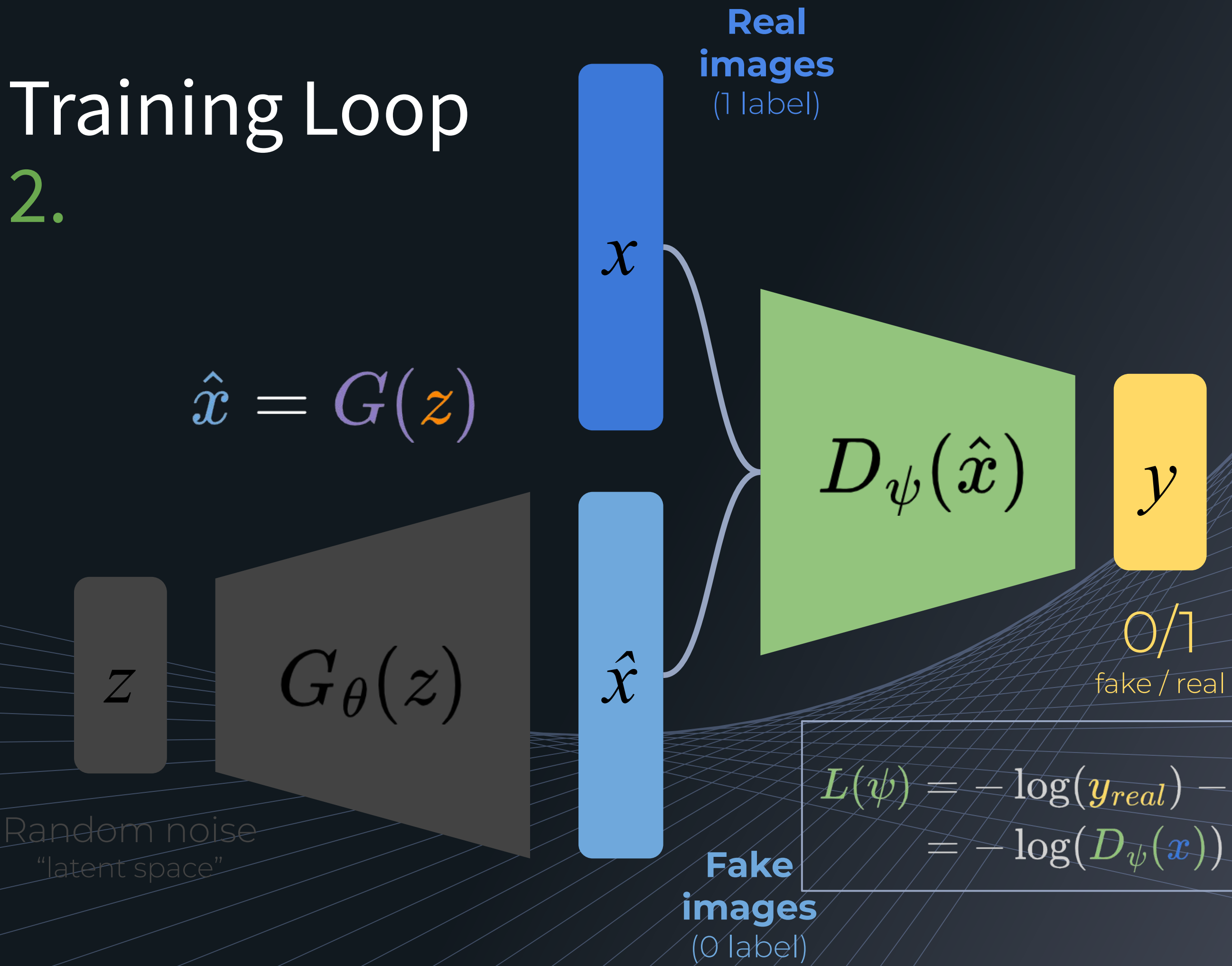

Real
images
Here!
x
$D_\psi(\hat{x})$
y
0/1
fake / real
z
$G_\theta(z)$
$\hat{x}$
Random noise
"latent space"
Fake
images

Training Loop
1.



$$
\begin{aligned}
L(\theta) &= -\log(y_{fake}) \\
&= -\log(D(G_\theta(z)))
\end{aligned}
$$

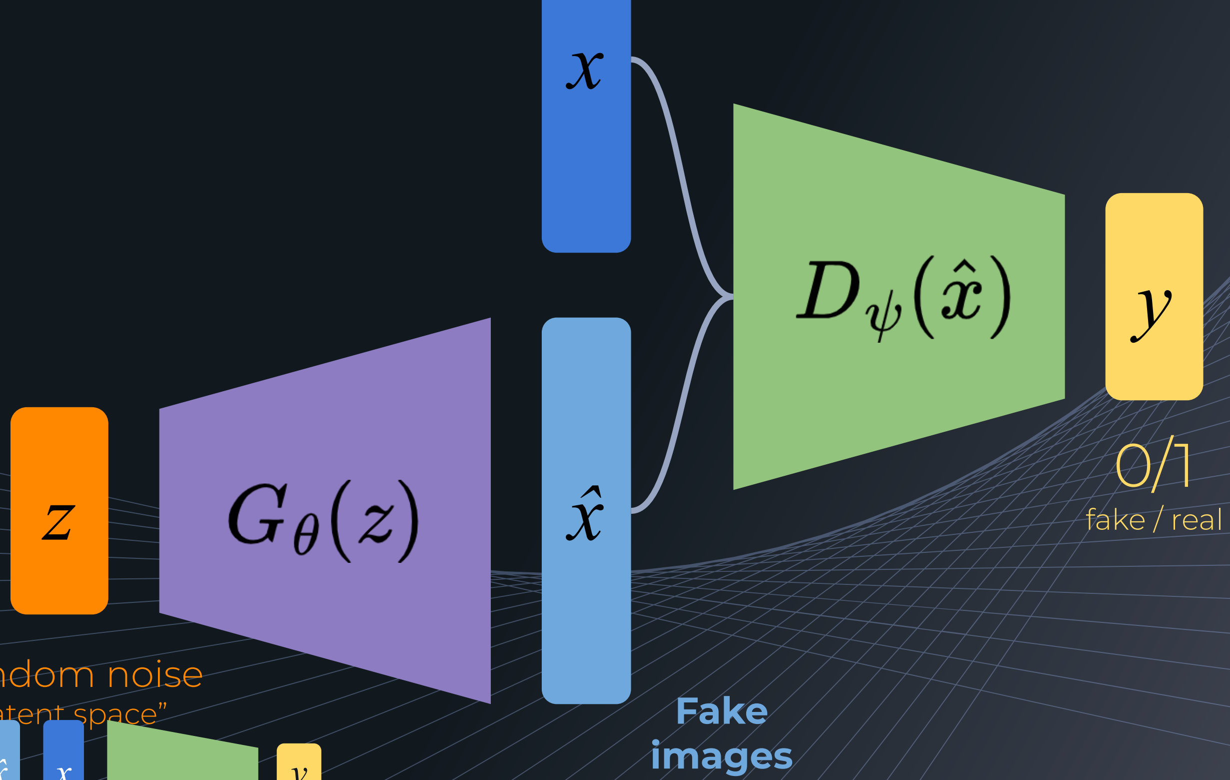
Training Loop
2.



$$
\begin{aligned}
L(\psi) &= -\log(y_{real}) - \log(1 - y_{fake}) \\
&= -\log(D_\psi(x)) - \log(1 - D_\psi(\hat{x}))
\end{aligned}
$$

$$L(\theta) = -\log(y_{fake})$$

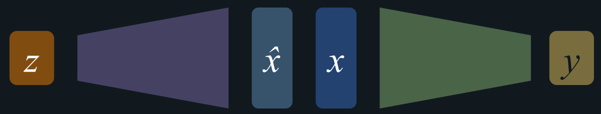
$$L(\psi) = -\log(y_{real}) - \log(1 - y_{fake})$$



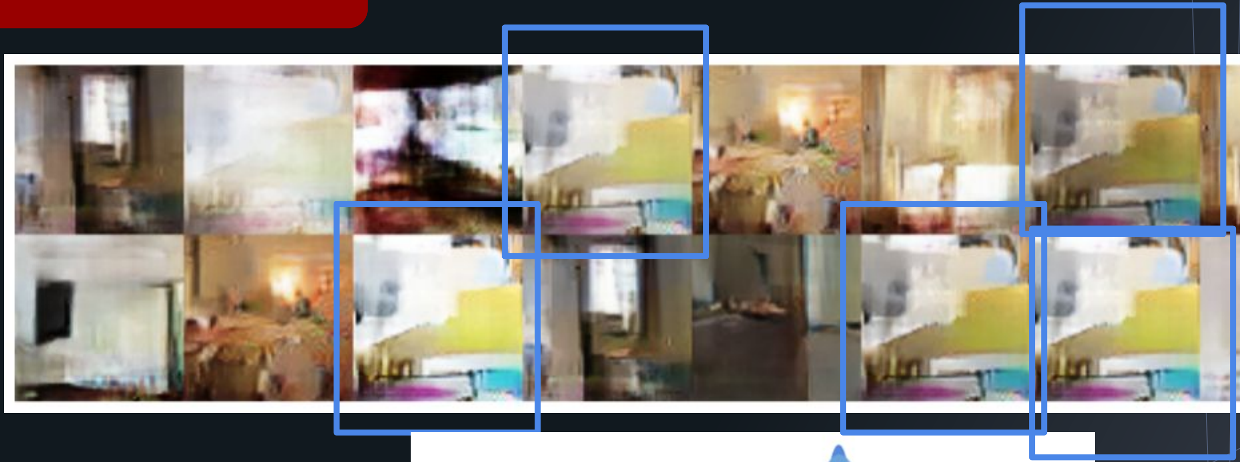
2 big challenges of GAN training

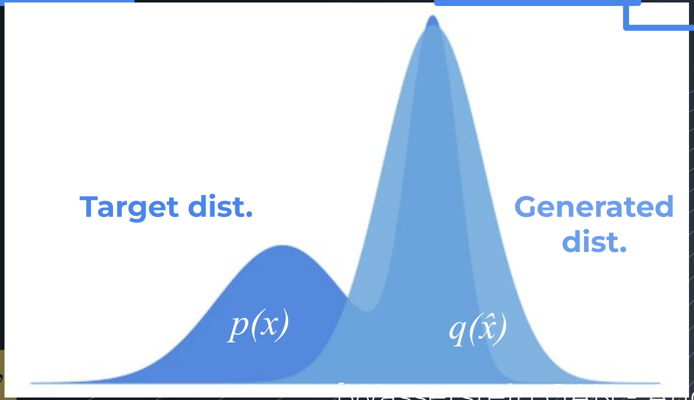
Mode Collapse

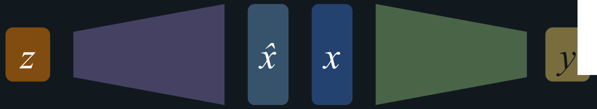
Unstable Nash Eq



Mode Collapse





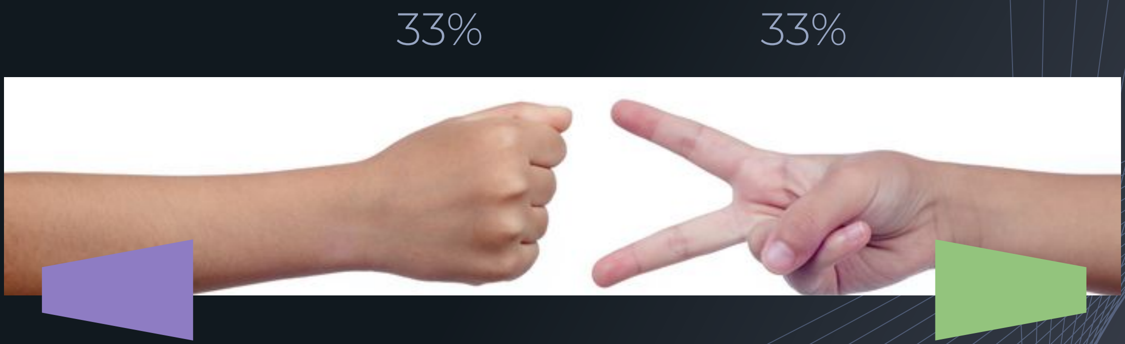


[Wasserstein GAN - Arjovsky, Chintala, Bottou 2017]

Unstable Nash Eq

Taking turns in a two-player game does not guarantee convergence

[Salimans et al., 2016]

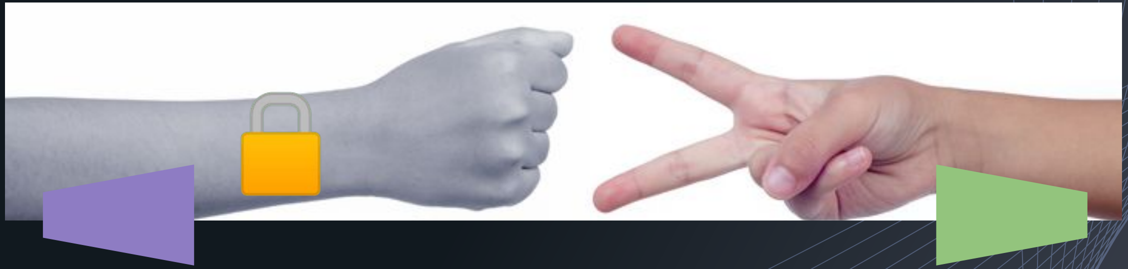


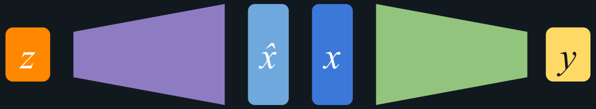
[Improved Techniques for Training GAN - Salimans, et.al 2016]

Unstable Nash Eq

Taking turns in a two-player game does not guarantee convergence [Salimans et al., 2016]

33% ?





[Improved Techniques for Training GAN - Salimans, et.al 2016]

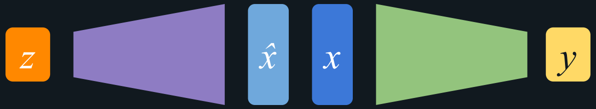
Unstable Nash Eq

Taking turns in a two-player game does not guarantee convergence [Salimans et al., 2016]

[Improved Techniques for Training GAN - Salimans, et.al 2016]

Unstable Nash Eq

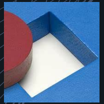
Taking turns in a two-player game does not guarantee convergence [Salimans et al., 2016]



[Improved Techniques for Training GAN - Salimans, et.al 2016]

Partial Solutions

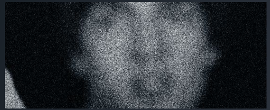
Mode Collapse **Unstable Nash Eq**



Feature Matching

Add loss term to direct G to match statistics (like mean, std) of training data

Truncation Trick

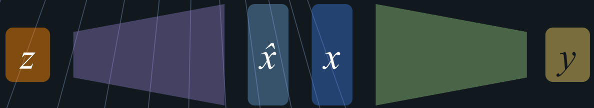


Adding Instance Noise

Historical Weight Averaging

Two Time-scale Update Rule

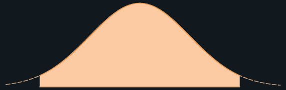
WGAN + GP



Partial Solutions

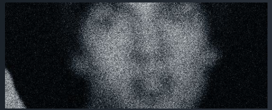
Mode Collapse **Unstable Nash Eq**

Feature Matching



Truncation Trick

Train D against G outputs sampled from a truncated z gaussian

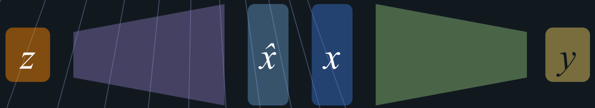


Adding Instance Noise

Historical Weight Averaging

Two Time-scale Update Rule

WGAN + GP

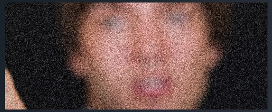


Partial Solutions

Mode Collapse **Unstable Nash Eq**

Feature Matching

Truncation Trick



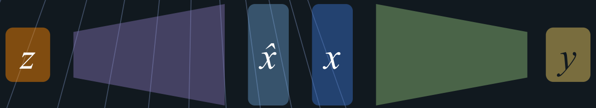
Adding Instance Noise

Corrupting D inputs makes it less nitpicky

Historical Weight Averaging

Two Time-scale Update Rule

WGAN + GP

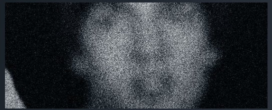


Partial Solutions

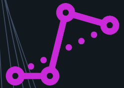
Mode Collapse **Unstable Nash Eq**

Feature Matching

Truncation Trick



Adding Instance Noise

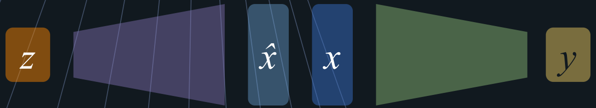


Historical Weight Averaging

Remember Stochastic Weight Averaging (SWA)?

Two Time-scale Update Rule

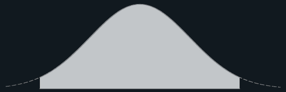
WGAN + GP



Partial Solutions

Mode Collapse **Unstable Nash Eq**

Feature Matching



Truncation Trick

Adding Instance Noise

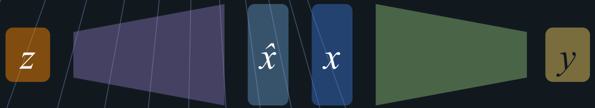
Historical Weight Averaging

🐢 🐇

Two Time-scale Update Rule

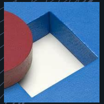
D trains at higher learning rate or frequency than G (commonly 5:1 ratio)

WGAN + GP



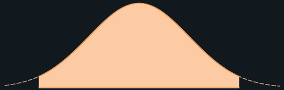
Partial Solutions

Mode Collapse **Unstable Nash Eq**



Feature Matching

Add loss term to direct G to match statistics (like mean, std) of training data



Truncation Trick

Train D against G outputs sampled from a truncated z gaussian

Adding Instance Noise

Corrupting D inputs makes it less nitpicky

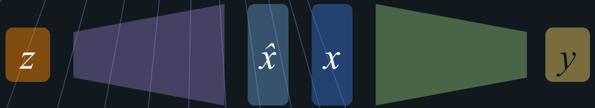
Historical Weight Averaging

Remember Stochastic Weight Averaging (SWA)?

Two Time-scale Update Rule

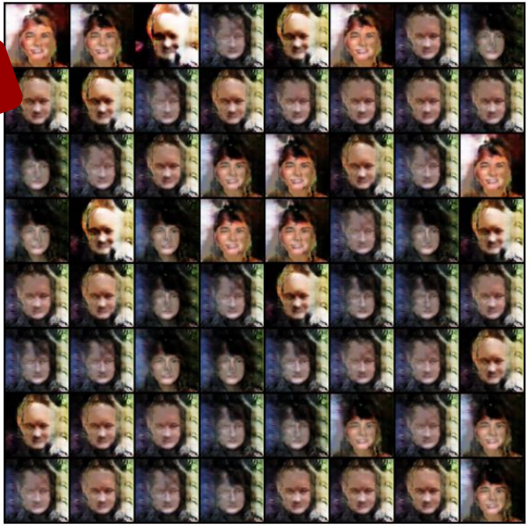
D trains at higher learning rate or frequency than G (commonly 5:1 ratio)

WGAN + GP

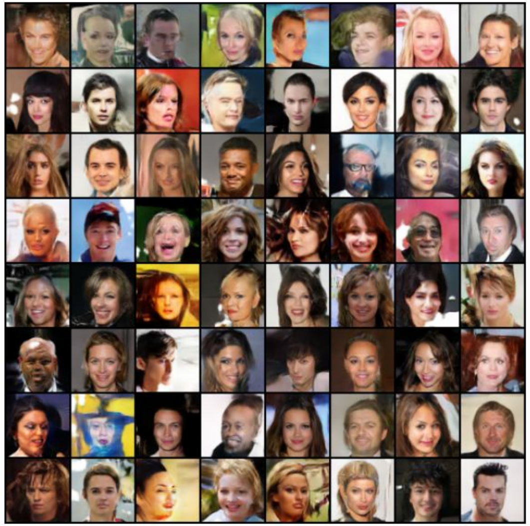


WassersteinGAN + Gradient Penalty

Mode Collapse



(d) NS-GAN-SN, $k_{SN} = 10$, FID=159.03



(f) WGAN-SN, $k_{SN} = 10$, FID=10.88

$$L(\theta) = -\log(y_{fake})$$

$$L(\psi) = -\log(y_{real}) - \log(1 - y_{fake})$$

Before

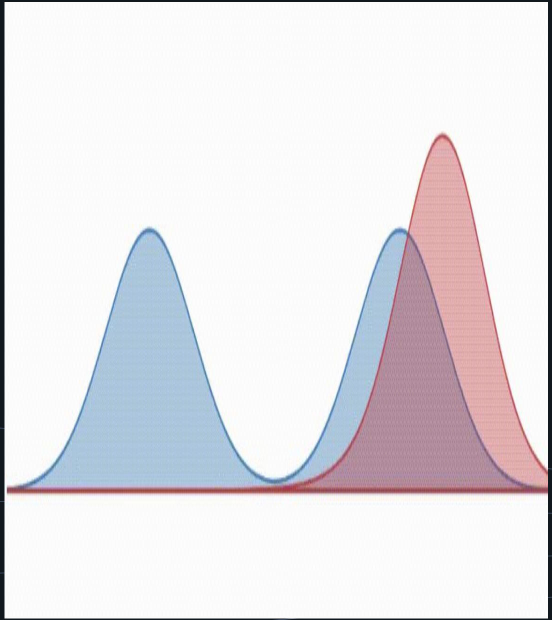
$$L(\theta) = y_{fake}$$

$$L(\psi) = y_{real} - y_{fake} + \lambda(||\nabla y_{fake}||_2 - 1)^2$$

After

[Do GAN Loss Functions Really Matter - Qin, Mitra, Wonka, 2018]

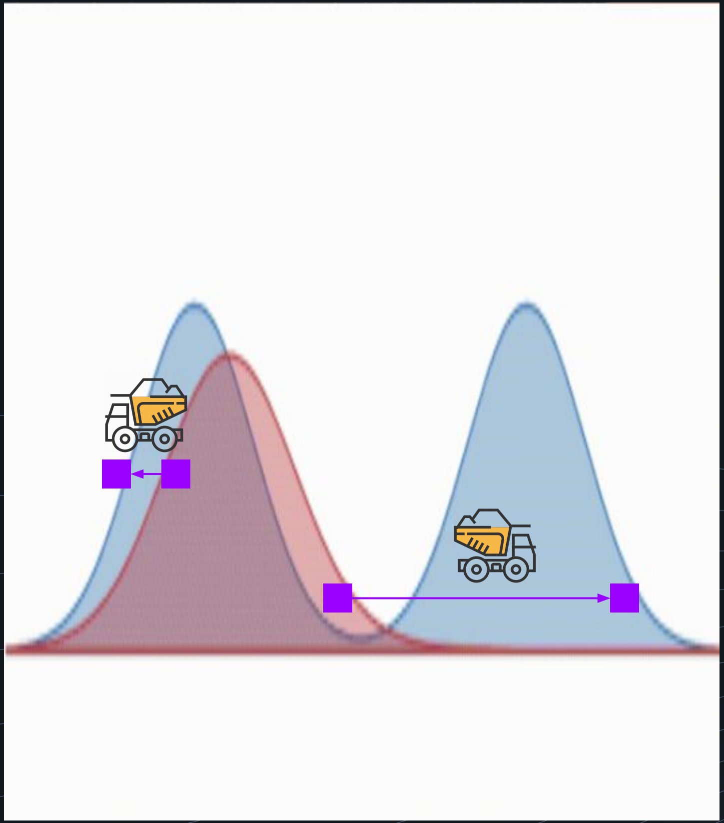
WassersteinGAN + Gradient Penalty



To combat **Mode Collapse** can we **train on the similarity of output distributions?**

How to measure similarity of probability distribution?

WassersteinGAN + Gradient Penalty

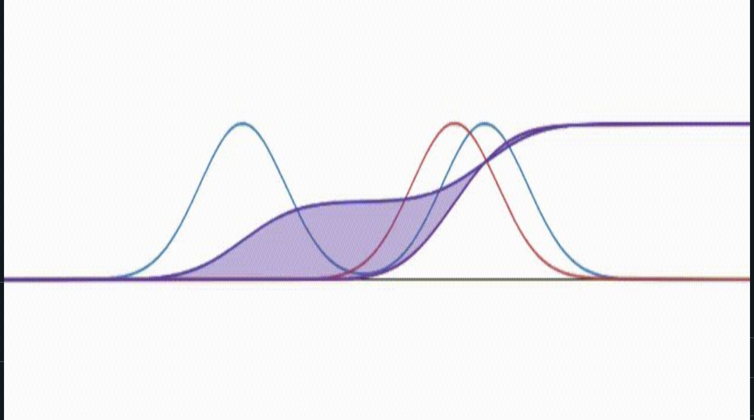


Earth mover's (EM) distance aka Wasserstein distance.

🛻

aka the "optimal transport" of probability to match the distributions

WassersteinGAN + Gradient Penalty

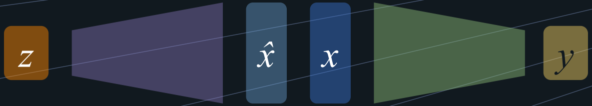


Wasserstein distance

=

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}\big[\, \|x - y\| \,\big]$$

= area between CDFs



[Wasserstein GAN - Arjovsky, Chintala, Bottou 2017]

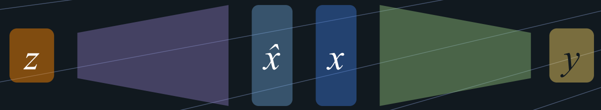
WassersteinGAN + Gradient Penalty

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}\big[\, \|x - y\| \,\big]$$

Kantorovich-Rubinstein duality, fancy algebra, **assumes smooth and shallow decision boundary**

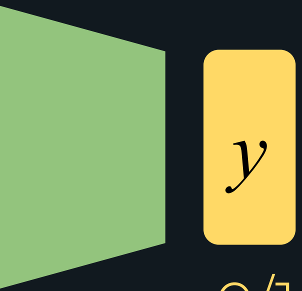
$$\max_{\|f\|_L \leq 1} \mathbb{E}_{x\sim\mathbb{P}_r}[f(x)] - \mathbb{E}_{x\sim\mathbb{P}_\theta}[f(x)]$$

$$\nabla_\theta W(\mathbb{P}_r, \mathbb{P}_\theta) = -\mathbb{E}_{z\sim p(z)}[\nabla_\theta f(g_\theta(z))]$$



[Wasserstein GAN - Arjovsky, Chintala, Bottou 2017]

WassersteinGAN + Gradient Penalty



~~0/1~~

~~fake / real~~

-∞/∞

Inauthenticity score

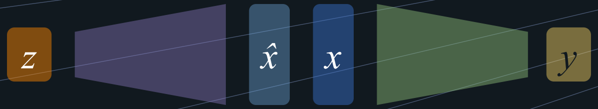
$$L(\theta) = -\log(y_{fake})$$

$$L(\psi) = -\log(y_{real}) - \log(1 - y_{fake})$$

Before

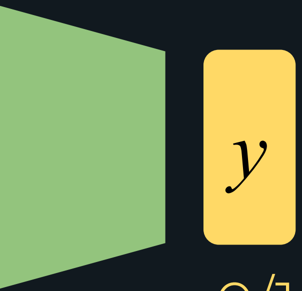
$$L(\theta) = y_{fake}$$

$$L(\psi) = y_{real} - y_{fake}$$



[Wasserstein GAN - Arjovsky, Chintala, Bottou 2017]

WassersteinGAN + Gradient Penalty



~~0/1~~

~~fake / real~~

-∞/∞

Inauthenticity score

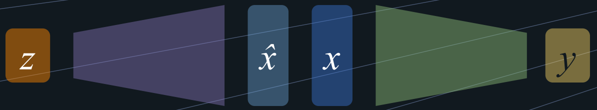
$$L(\theta) = -\log(y_{fake})$$

$$L(\psi) = -\log(y_{real}) - \log(1 - y_{fake})$$

Before

$$L(\theta) = D(G_\theta(z))$$

$$L(\psi) = D_\psi(x) - D_\psi(\hat{x})$$



[Wasserstein GAN - Arjovsky, Chintala, Bottou 2017]

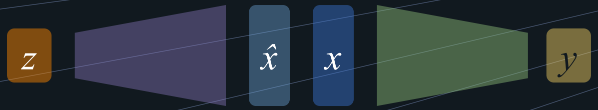
WassersteinGAN + Gradient Penalty

assumes smooth shallow decision boundary

$$L(\psi) = D_\psi(x) - D_\psi(\hat{x})$$

$$+\lambda(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2$$

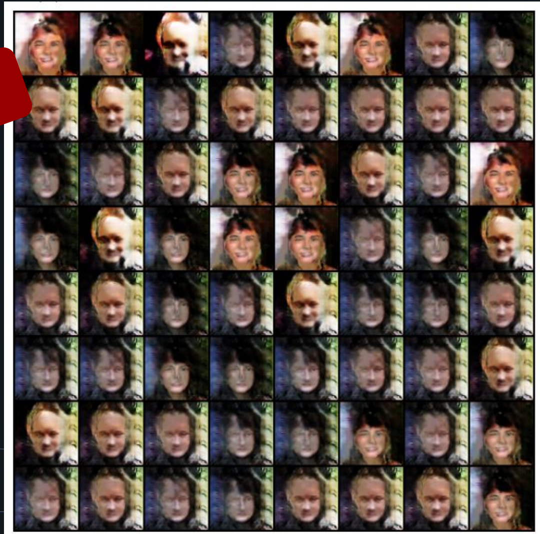
Gradient Penalty punishes harsh (non-Lipschitz) decision boundaries in $\boldsymbol{D}$



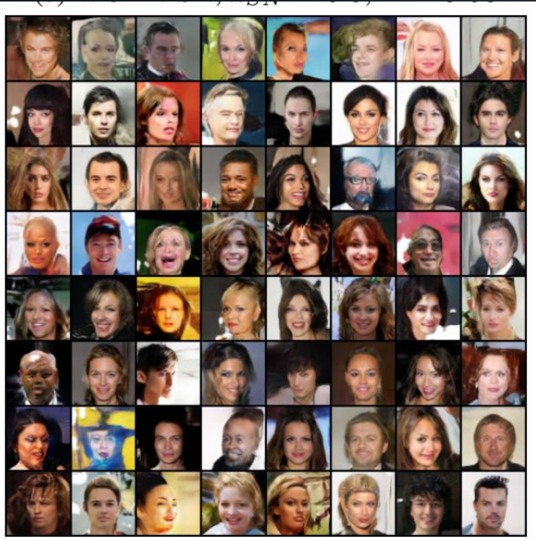
[Improved Training of Wasserstein GANs - Gulrajani, et.al 2017]

Wasserstein GAN + Gradient Penalty

Mode Collapse



(d) NS-GAN-SN, $k_{SN} = 10$, FID=159.03



(f) WGAN-SN, $k_{SN} = 10$, FID=10.88

$$L(\theta) = -\log(y_{fake})$$

$$L(\psi) = -\log(y_{real}) - \log(1 - y_{fake})$$

Before

$$L(\theta) = y_{fake}$$

$$L(\psi) = y_{real} - y_{fake} + \lambda(||\nabla y_{fake}||_2 - 1)^2$$

After

[Do GAN Loss Functions Really Matter - Qin, Mitra, Wonka, 2018]

3.

Extensions

Improving the original GAN

An ‘ideal’ painter...

- Produces high **quality** paintings
- Creates paintings of **diverse** scenes
- Listens to our **guidance**

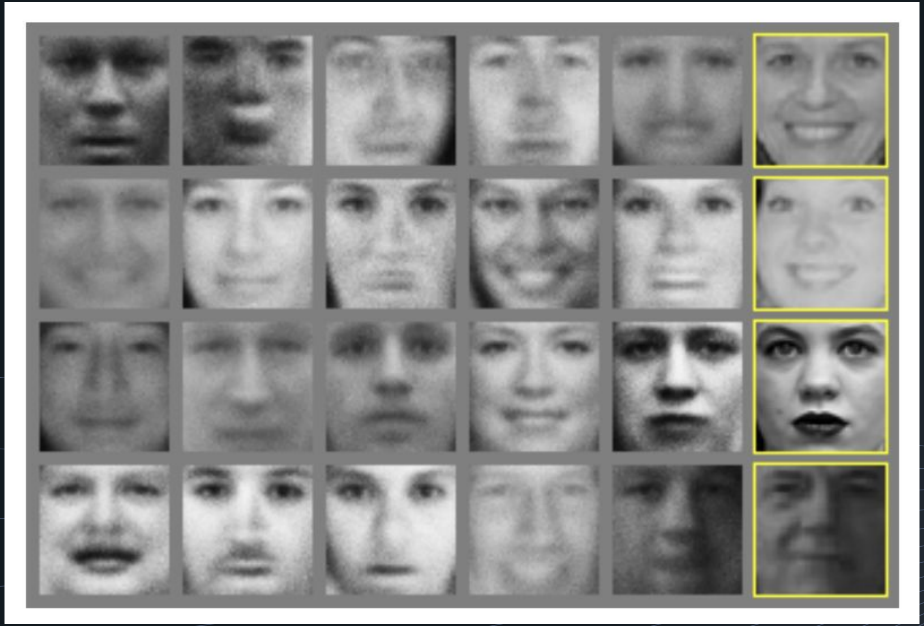
Painting: EDOUARD MANET, *MONET IN HIS FLOATING STUDIO*, 1874, IN THE COLLECTION OF THE NEUE PINAKOTHEK , MUNICH, GERMANY.

An ‘ideal’ GAN generator...

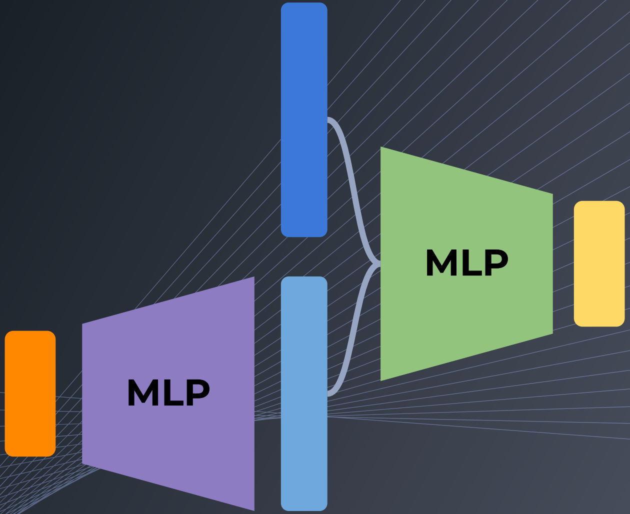
- Produces high ~~quality paintings~~ **resolution** images
- ~~Creates paintings of diverse scenes~~ *Resist* **Mode Collapse**
- ~~Listens to our guidance~~ Is **conditionable**

Painting: EDOUARD MANET, *MONET IN HIS FLOATING STUDIO*, 1874, IN THE COLLECTION OF THE NEUE PINAKOTHEK , MUNICH, GERMANY.

Original GAN



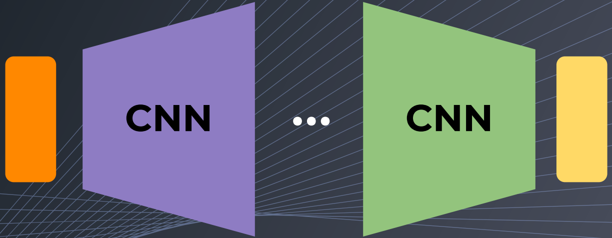
32 x 32



* Goodfellow, Ian J., et al. "Generative adversarial networks. arXiv e-prints." *arXiv preprint arXiv:1406.2661* 1406 (2014).

Deep Convolutional GAN

64 x 64!



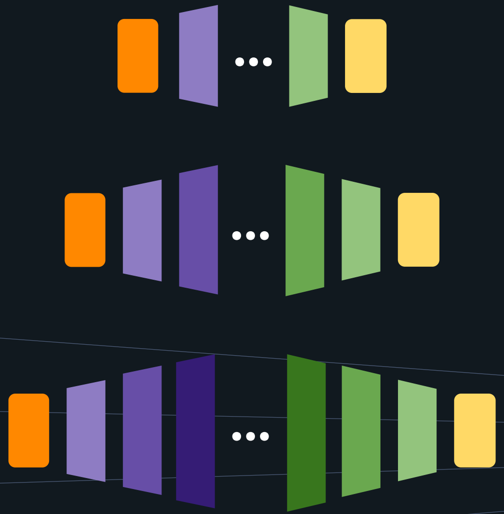
* Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." *arXiv preprint arXiv:1511.06434* (2015).

Progressive GAN

1024 x 1024!!

* Karras, Tero, et al. "Progressive growing of gans for improved quality, stability, and variation." *arXiv preprint arXiv:1710.10196* (2017).

Progressive GAN



Progressively boost G and D

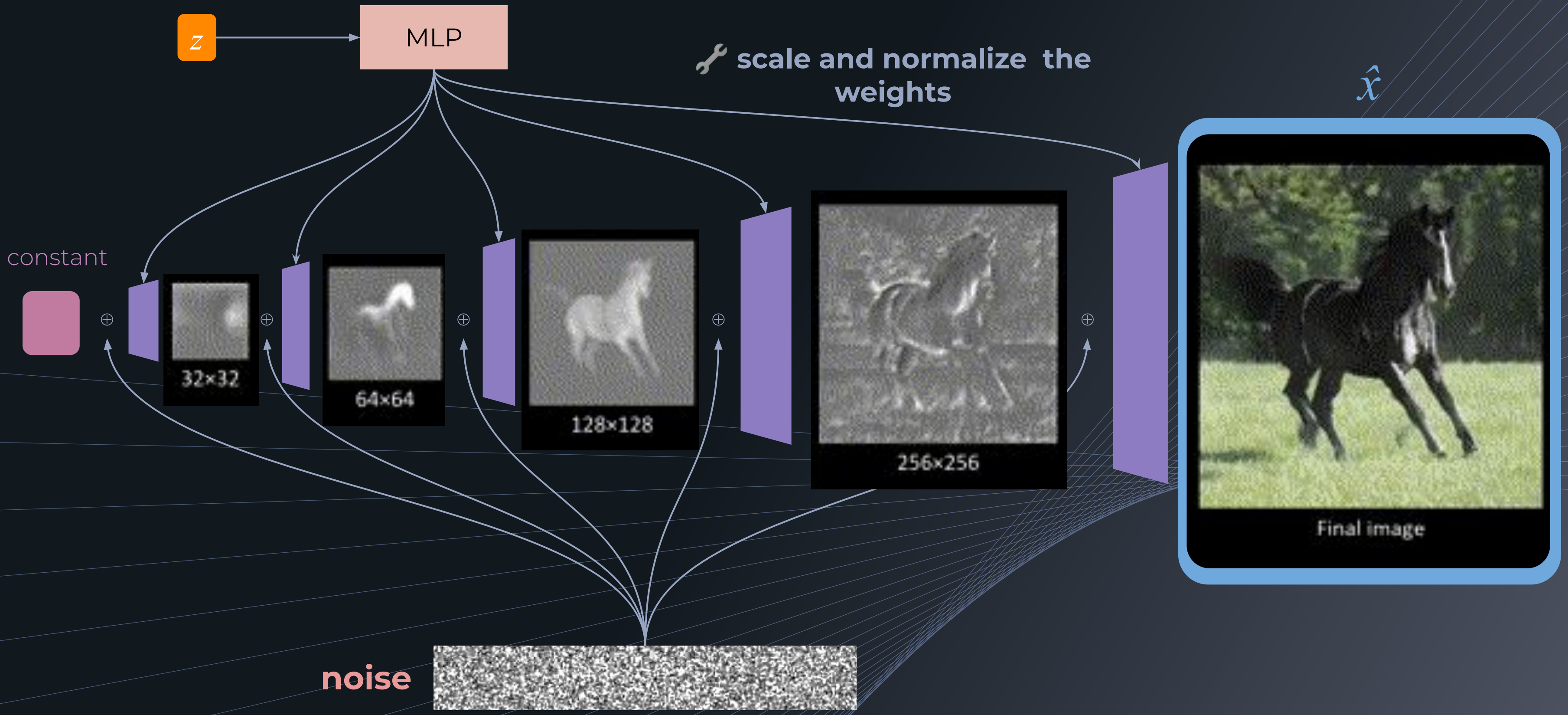
1. Train GAN
2. Insert higher resolution convolution blocks
3. Repeat

* Karras, Tero, et al. "Progressive growing of gans for improved quality, stability, and variation." *arXiv preprint arXiv:1710.10196* (2017).

StyleGAN

* Karras, Tero, Samuli Laine, and Timo Aila. "A style-based generator architecture for generative adversarial networks." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

StyleGAN


z
MLP
scale and normalize the weights
$\hat{x}$
constant
32×32
64×64
128×128
256×256
Final image
noise

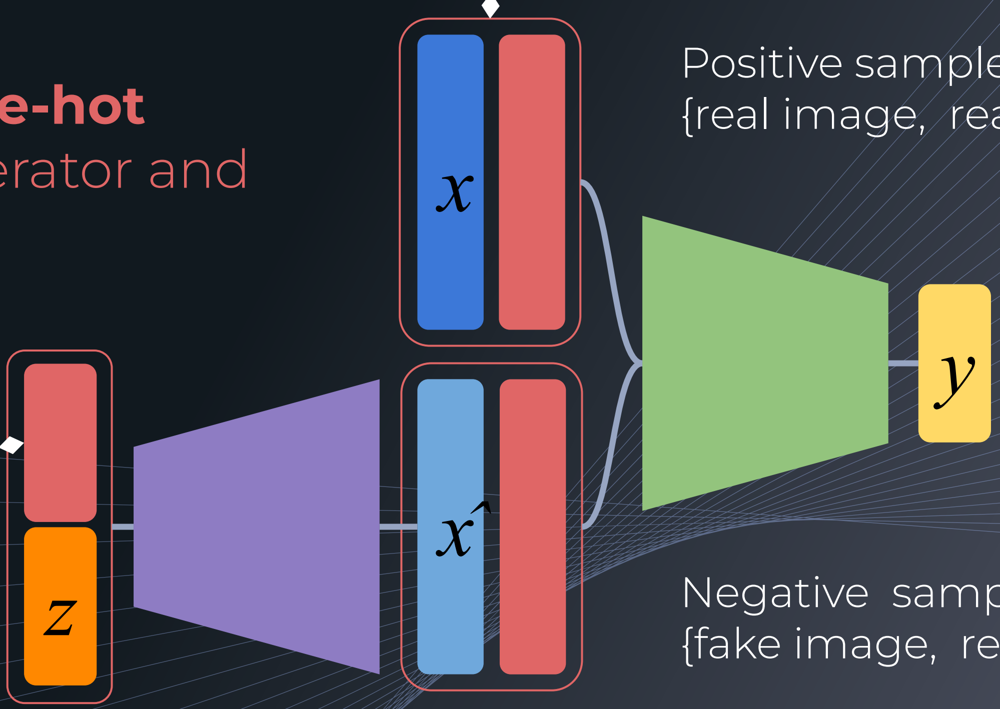
Condition the GAN on Label

Conditional GAN

Feed the same **one-hot label** to both generator and discriminator

It is extended to the target shape and added as extra channels

Positive sample:
{real image, real label}



It is directly concatenated to the latent vector as extra dimensions

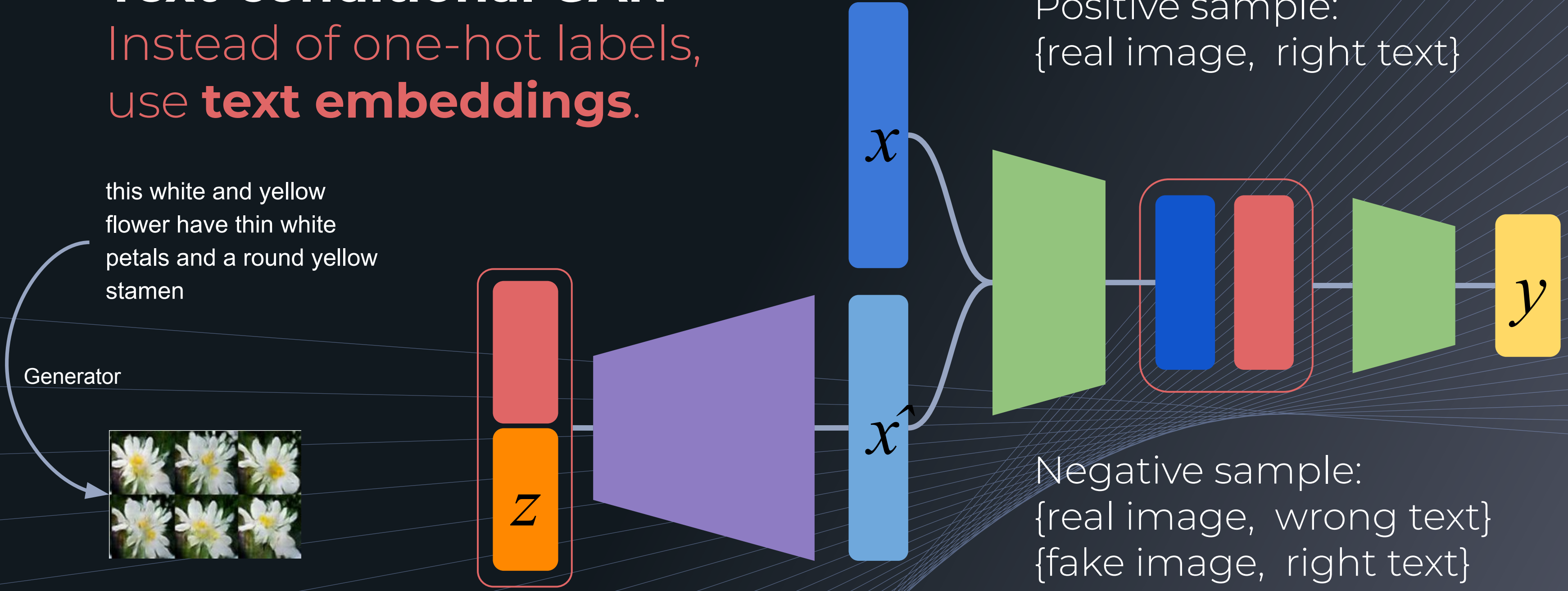
Negative sample:
{fake image, real label}

* Mirza, Mehdi, and Simon Osindero. "Conditional generative adversarial nets." *arXiv preprint arXiv:1411.1784* (2014).

Condition the GAN on Text

Text-conditional GAN

Instead of one-hot labels,
use **text embeddings**.

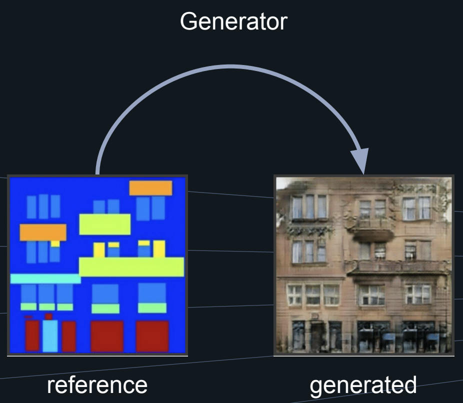


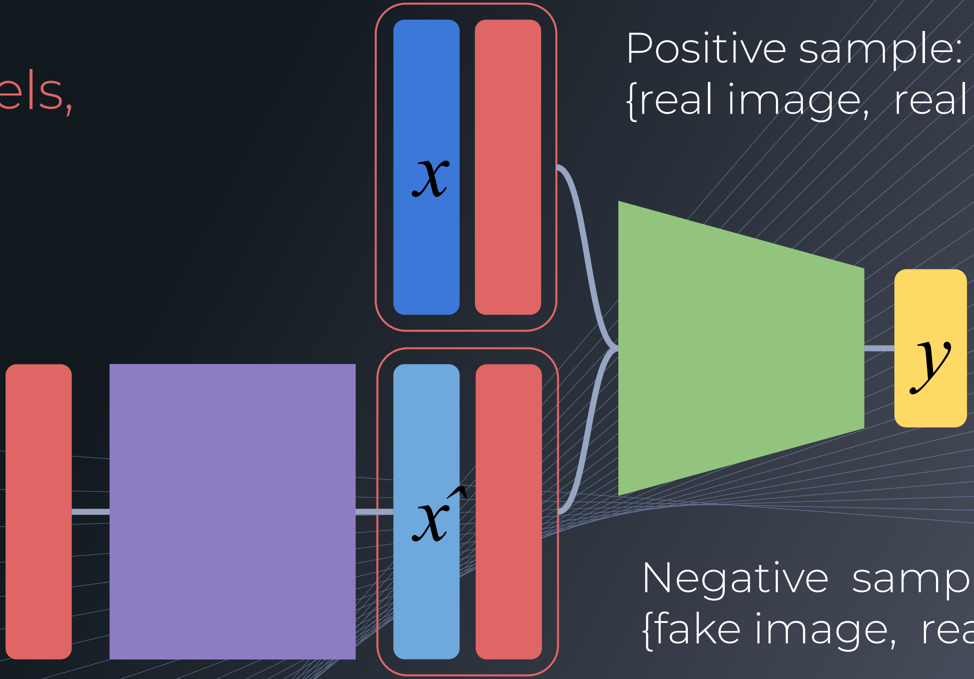
* Reed, Scott, et al. "Generative adversarial text to image synthesis." *International conference on machine learning*. PMLR, 2016.

Condition the GAN on Reference Mask

Pix2Pix

Instead of one-hot labels,
use **reference mask**.





Positive sample:
{real image, real ref}

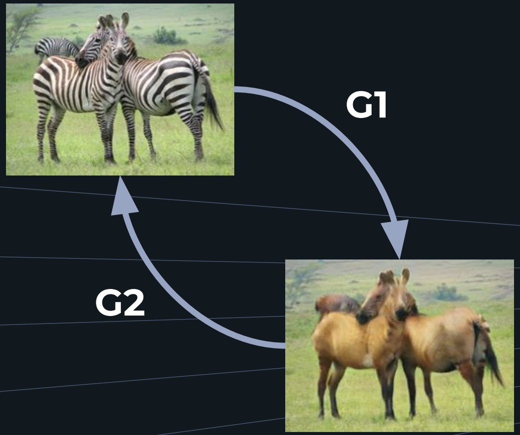
Negative sample:
{fake image, real ref}

* Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

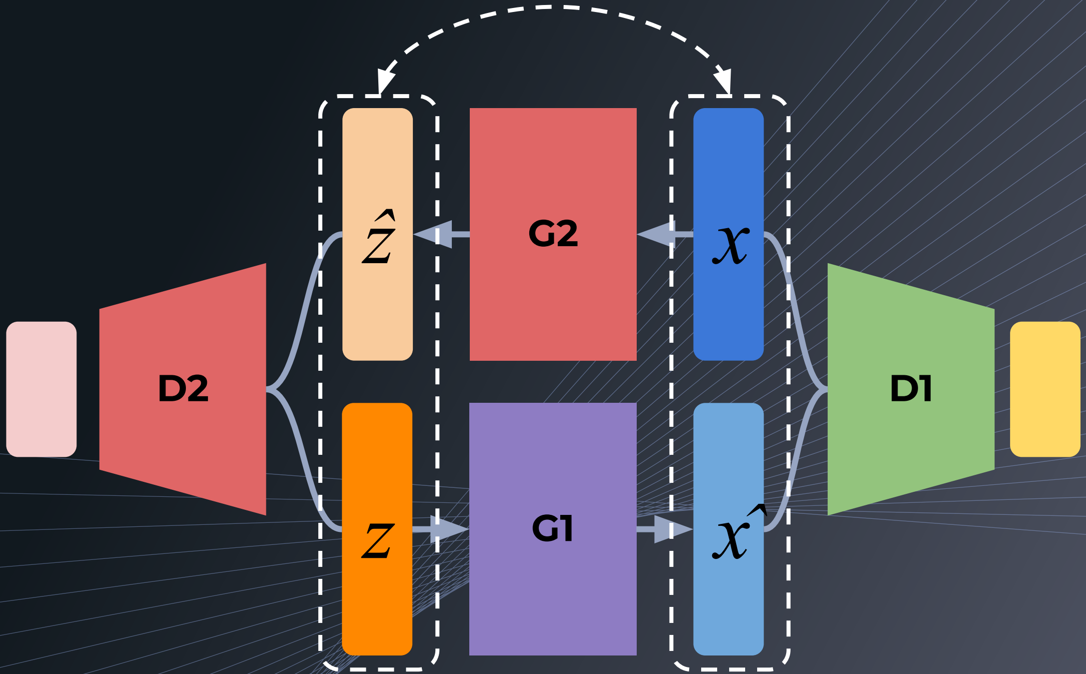
Condition the GAN on another GAN

Cycle GAN

Instead of labels, use **another GAN**



It gives us a mapping between two distributions!



* Zhu, Jun-Yan, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." *Proceedings of the IEEE international conference on computer vision*. 2017.

Cycle GAN

Cycle Consistency Loss

It should be the same when it is mapped forward and backward:

$$L_{cycle}(G_1, G_2) = L_1(G_2(G_1(z)), z) + L_1(G_1(G_2(x)), x)$$

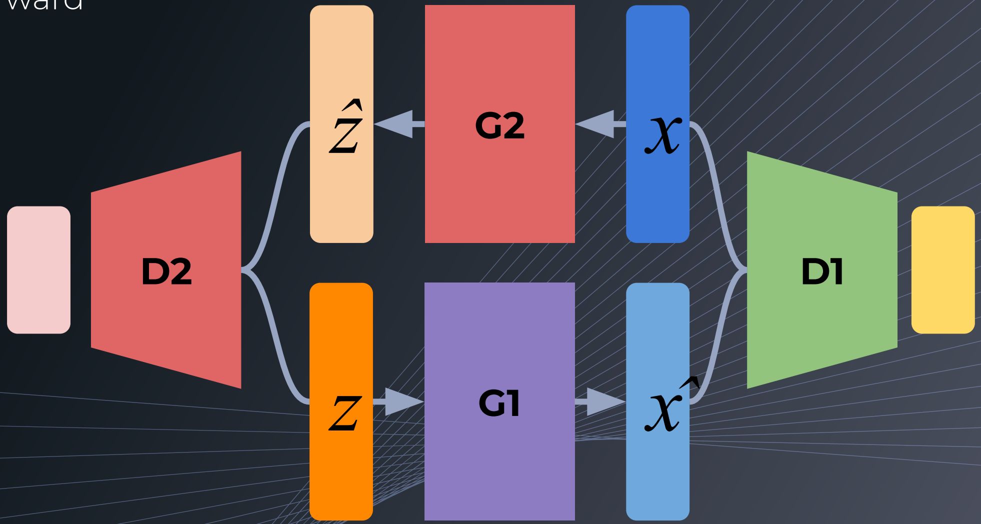
Identity Loss

It should make no change when the true label is fed:

$$L_{identity}(G_1, G_2) = L_1(G_2(x), x) + L_1(G_1(z), z)$$

GAN Loss

The vanilla GAN loss introduced before



* Zhu, Jun-Yan, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." *Proceedings of the IEEE international conference on computer vision*. 2017.

What GAN we do next?

GAN provides a self-supervised way to model distributions.

Wherever modeling distribution is helpful, there is a potential place for GAN: Image manipulation, Image inpainting, Image translation, Image super-resolution, Style transfer, Data augmentation, Domain adaptation.

It starts from image, but applies to more than image.

4.

Summary

What’s a GAN again?

Let's review some concepts

Structure

A generator tries to trick a discriminator, a discriminator discerns real and fake outputs

Mode collapse

Is when GAN generators fail to produce diverse images

Unstable Nash Equilibrium

Describes failure to converge because of an exploding cat-and-mouse game

WGAN + GP

Modifies the loss function to reduce mode collapse using earth mover's distance

StyleGAN

Inputs latent noise at different image resolutions instead of at the beginning of the generator for better output control

Conditional GAN, CycleGAN, Pix2Pix

Are variations of GAN that are conditioned on labels, other GANs, or reference masks

5.
Paper

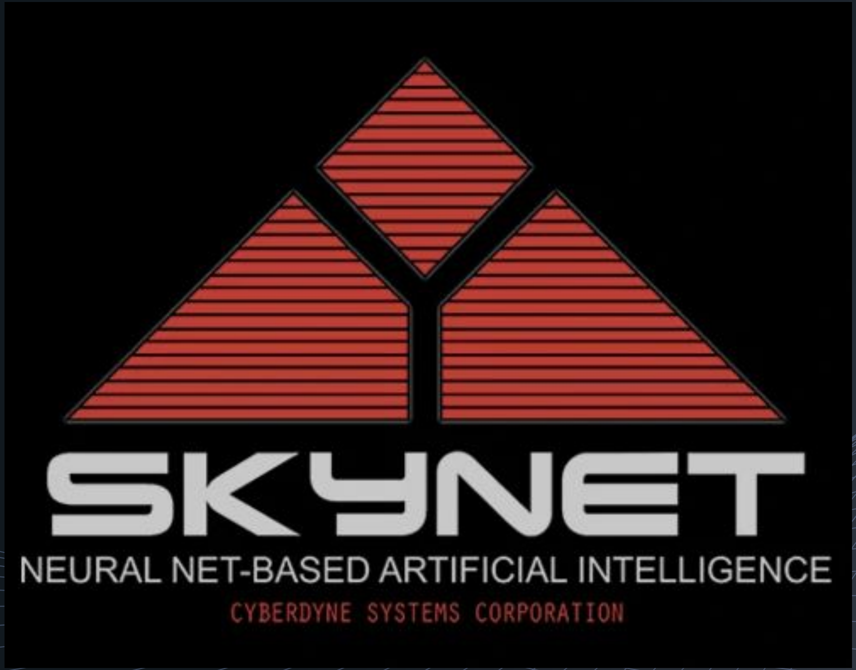
Published as a conference paper at ICLR 2022

Tackling the Generative Learning Trilemma with Denoising Diffusion GANs

Zhisheng Xiao*
The University of Chicago
zxiao@uchicago.edu

Karsten Kreis
NVIDIA
kkreis@nvidia.com

Arash Vahdat
NVIDIA
avahdat@nvidia.com



The original generative adversarial network

References

[1] Goodfellow, Ian J., et al. "Generative adversarial networks. arXiv e-prints." arXiv preprint arXiv:1406.2661 1406 (2014).
[2] Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein generative adversarial networks." International conference on machine learning. PMLR, 2017.
[3] Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." arXiv preprint arXiv:1511.06434 (2015).
[4] Karras, Tero, et al. "Progressive growing of gans for improved quality, stability, and variation." arXiv preprint arXiv:1710.10196 (2017).
[5] Karras, Tero, Samuli Laine, and Timo Aila. "A style-based generator architecture for generative adversarial networks." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.
[6] Mirza, Mehdi, and Simon Osindero. "Conditional generative adversarial nets." arXiv preprint arXiv:1411.1784 (2014).
[7] Reed, Scott, et al. "Generative adversarial text to image synthesis." International conference on machine learning. PMLR, 2016.
[8] Zhu, Jun-Yan, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." Proceedings of the IEEE international conference on computer vision. 2017.