Generative Models

Generative Adversarial Nets

Marcos C. S. Santana, Prof. João Paulo Papa

December 18, 2019

Universidade Estadual Paulista "Júlio de Mesquita Filho" (UNESP)
Faculdade de Ciências (FC) / Departamento de Computação (DCo)
Bauru, SP - Brasil

Outline

- GAN - Generative Adversarial Network
  - PDF Transformations
  - GAN - Generative Adversarial Nets
  - Variations
  - Applications

The Problem

- How to sample from a high dimensional training distribution?
- Not easy!
- The feasible solution is to train a neural network that learns to sample from training distribution.

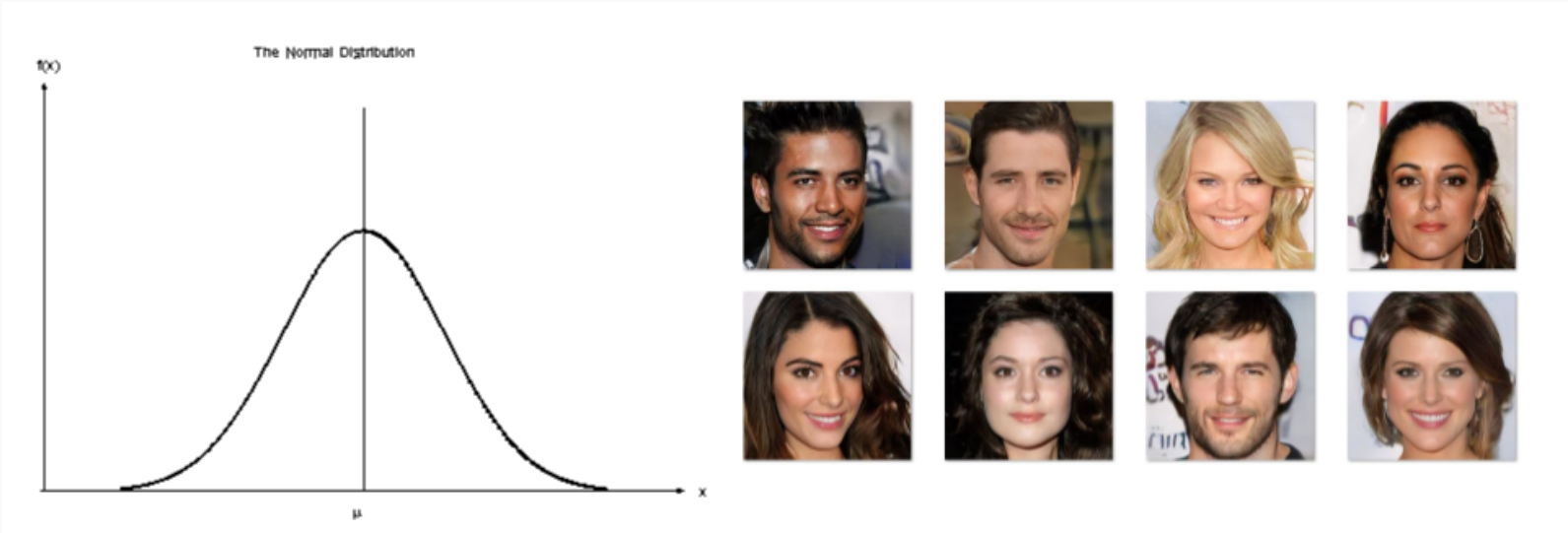


Figura 1: Left: Gaussian distribution is easy do samples. Right: Faces are not easy to be samples.

- Neural Networks can easily approximate non-linear function.

RECOGNA LABORATORY

Density Transformation

- A non-linear function $f(x)$ can transform a simple to a more complex data distribution.

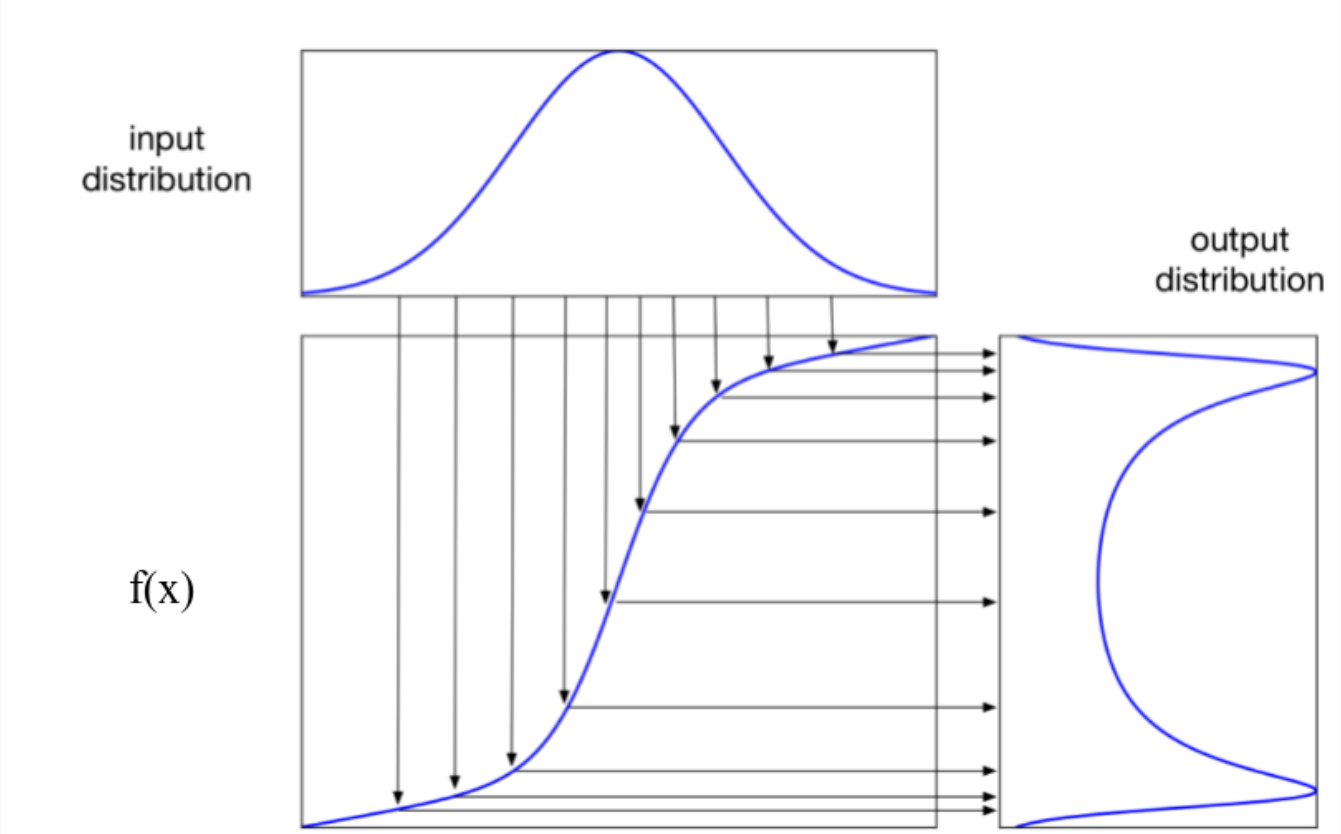


Figura 2: Credit: Roger Grosse, CSC321 Lecture 19: Generative Adversarial Networks 2018

RECOGNA
LABORATORY

Density Transformation

- Transform easy to sample density ($\mathcal{N}(\mu, \sigma)$) distribution to complex one.

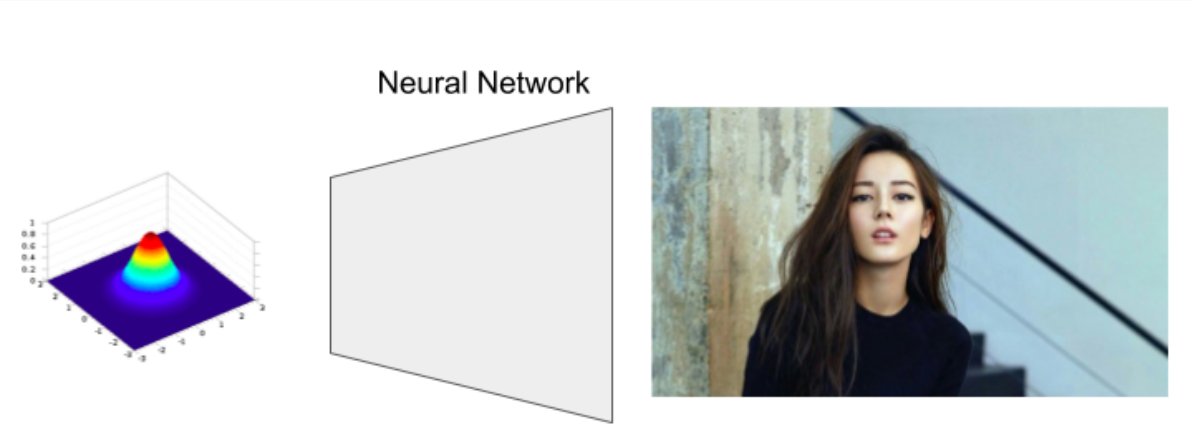


Figura 3: Neural network converting samples from Gaussian distribution to image domain distribution.

- How to define the criterion to train such neural networks?

RECOGNA LABORATORY

Neural Networks Density transformation.

How to define the criterion to train such neural networks?

- Restricted Boltzmann Machine (RBM) - Gibbs sampling and reconstruction [Nair and Hinton, 2010].
- Generative Moment Matching Networks (GMMN) - Maximum Mean Discrepancy, matching the momenta [Li et al., 2015].
- **Generative Adversarial Nets (GAN)** - adversarial training [Goodfellow et al., 2014].

RECOGNA LABORATORY

Generative Adversarial Nets -GAN

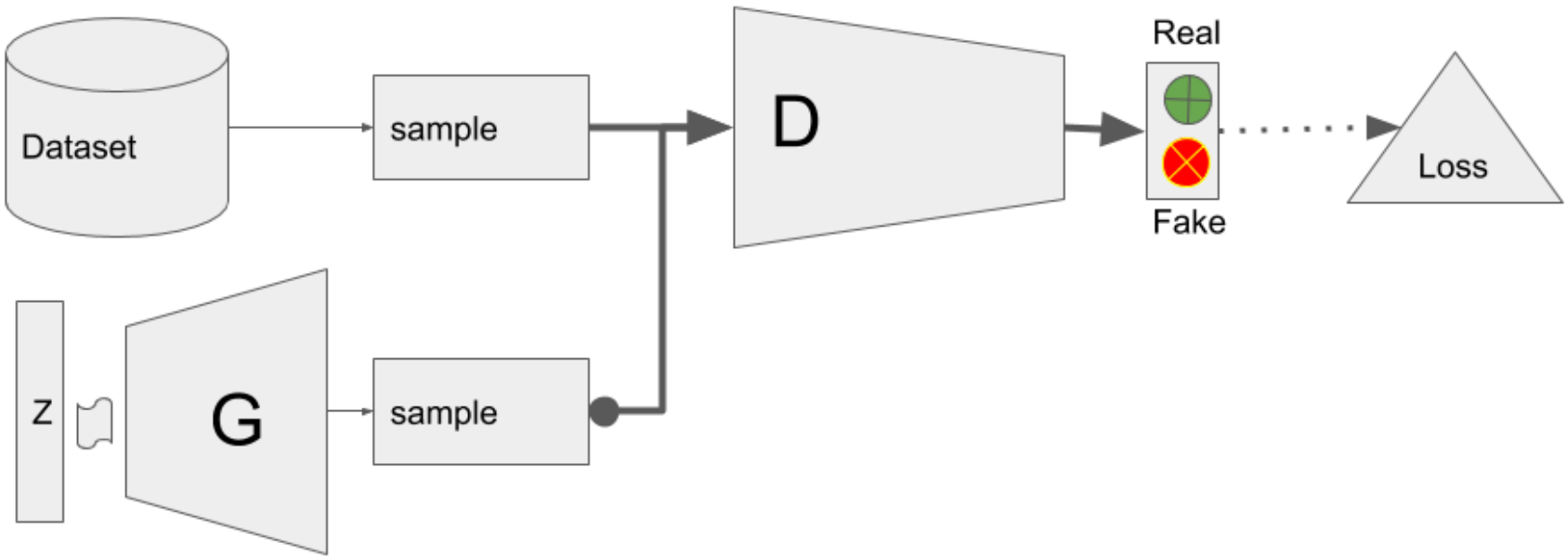
GAN is:

- A neural architecture that learns to convert simple to complex distribution from dataset.
- Composed of two neural networks [Goodfellow et al., 2014].
- Using competition (adversarial) as learing criterion.

RECOGNA LABORATORY

Generative Adversarial Nets -GAN

- Discriminator - Classifies if the input data is from the dataset (true) or generated by the Generator net (fake).
- Generator - Takes random sample as input and converts them into an image to fool the Discriminator.



RECOGNA LABORATORY

GAN: Discriminator Loss

$$\mathcal{L}_D(G, D) = \mathbb{E}_{x \sim p(x)} \Big[\log D(x) \Big] + \mathbb{E}_{x \sim q(z)} \Big[\log \big(1 - D(G(z))\big) \Big] \tag{1}$$

$$\max_D \Big[\mathcal{L}_D(G, D) \Big] \tag{2}$$

- The term $\log D(x)$ is related to the discriminator output for real data.
- The term $\log \big(1 - D(G(z))$ is related to the discriminator output for the generated data (fake data).

RECOGNA LABORATORY

GAN: Generator Loss

- A minmax formulation where the Generator plays against Discriminator in zero-sum game.

$$\mathcal{L}_G(G) + \mathcal{L}_D(G, D) = 0 \tag{3}$$

$$\mathcal{L}_G(G) = -\mathcal{L}_D(G, D) = const - \mathbb{E}_{x \sim q(z)} \Big[\log \big(1 - D(G(z))\big) \Big] \tag{4}$$

$$\min_G \Big[\mathcal{L}_G(G) \Big] \tag{5}$$

- Interesting in theory.
- Unpractical for DL.

RECOGNA LABORATORY

GAN: Minmax Loss Issues

Be $x = G(z)$ and $y = D(x) = D(G(z))$. The gradient of y w.r.t x is:

$$\nabla_x \mathcal{L}_G(G) = -\frac{\nabla_x y}{1-y} = -\frac{y(1-y)}{1-y} = -y = -D(G(z)) \tag{6}$$

When D is confident $D(G(z)) = 0$ then:

$$\nabla_x \mathcal{L}_G(G) = -D(G(z)) = 0 \tag{7}$$

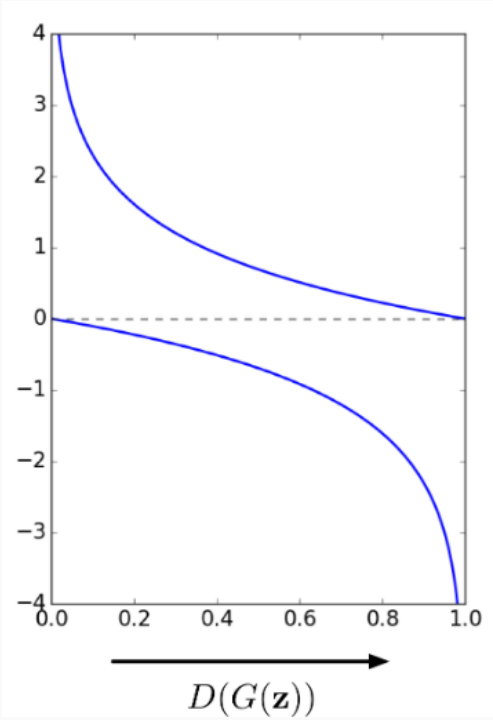
The gradient vanishes for Generator.

RECOGNA LABORATORY

GAN: Minmax Loss Issues

- Early in learning, when G generates poor data, D can reject samples with high confidence because they are clearly different from the training data.
- Instead trying to minimize $\log 1 - D(G(z))$ for G, we maximize $\log D(G(x))$ [Goodfellow et al., 2014].
- In practice, instead of trying to minimize the D's correct classification, it tries to maximize D incorrect classifications.

Credit: Roger Grosse, CSC321 Lecture 19: Generative Adversarial Networks 2018



RECOGNA LABORATORY

GAN: Better Loss

Alternates between:

$$\max_D \left[\mathbb{E}_{x \sim p(x)} \left[\log D(x) \right] + \mathbb{E}_{x \sim q(z)} \left[\log \left(1 - D(G(z)) \right) \right] \right] \tag{9}$$

and

$$\max_G \left[\mathbb{E}_{x \sim q(z)} \left[\log \left(D(G(z)) \right) \right] \right] \tag{10}$$

RECOGNA LABORATORY

GAN: Equilibrium

- At Nash equilibrium, the discriminator cannot distinguish real data from generate data.
- It is a zero sum game with Nash equilibuion point reached when $p_{data}(x) = q(z)$ and $D(x) = \frac{1}{2}$.

RECOGNA LABORATORY

GAN: Training Algorithm

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:

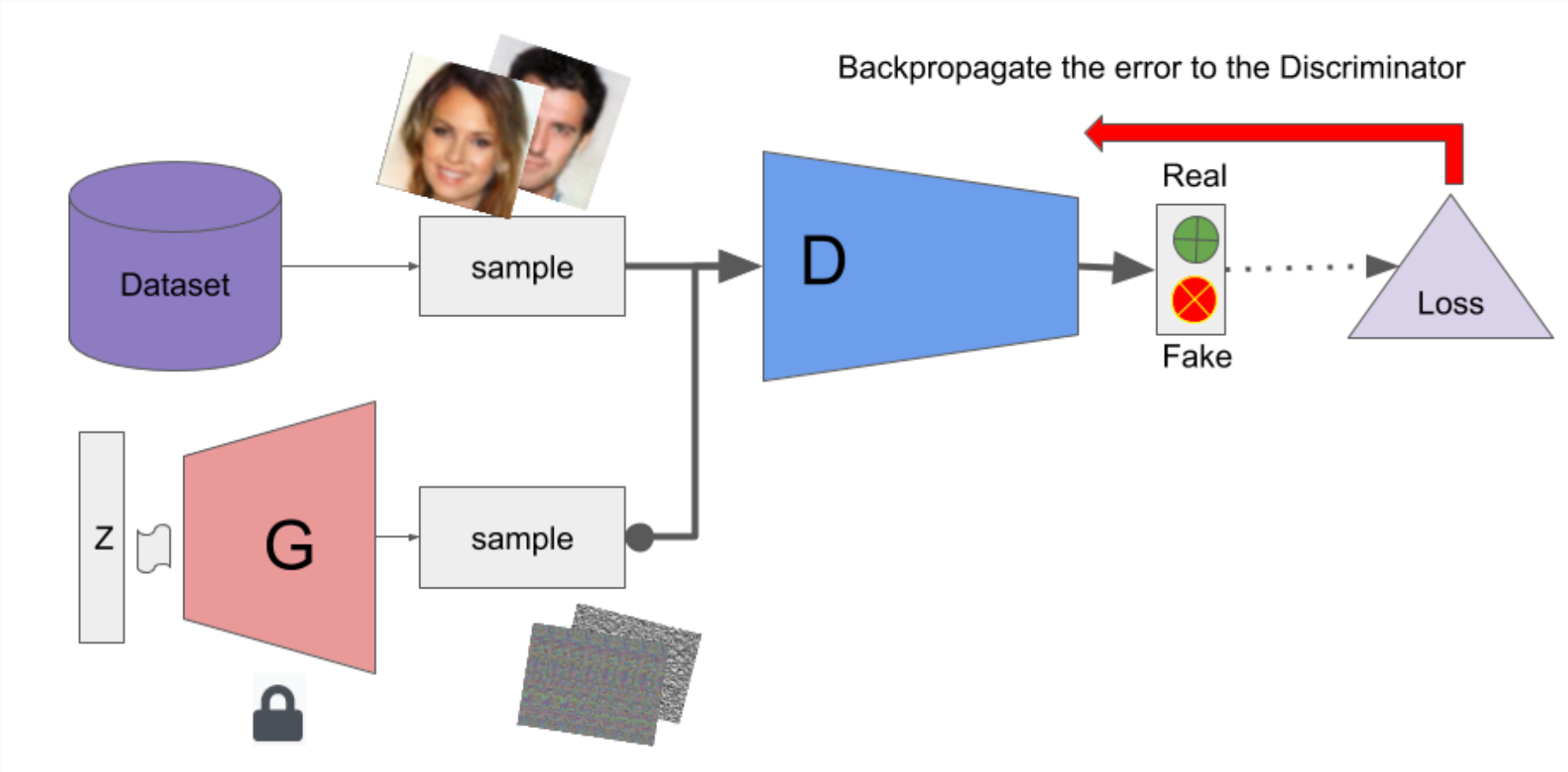
$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for

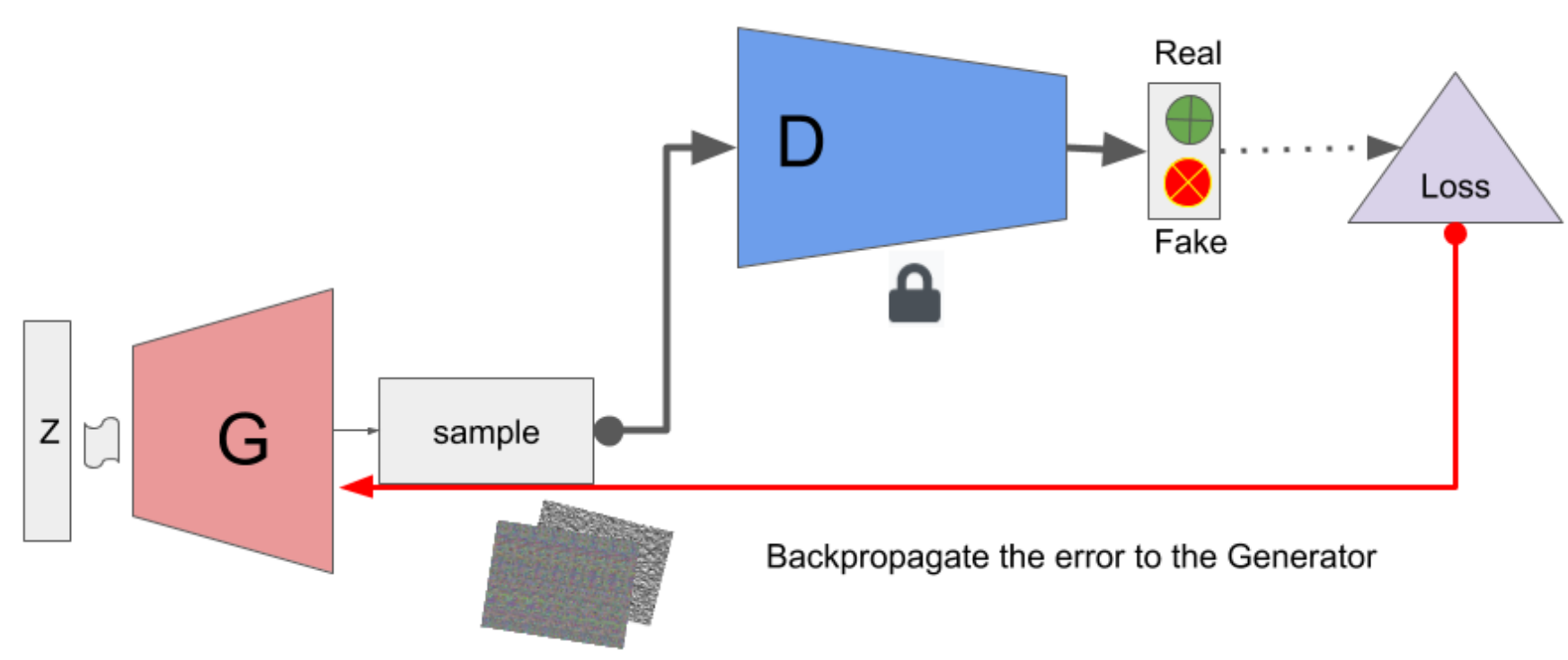
The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

RECOGNA LABORATORY

GAN: Training the Discriminator



GAN: Training the Generator



RECOGNA LABORATORY

GAN: Training Evolution

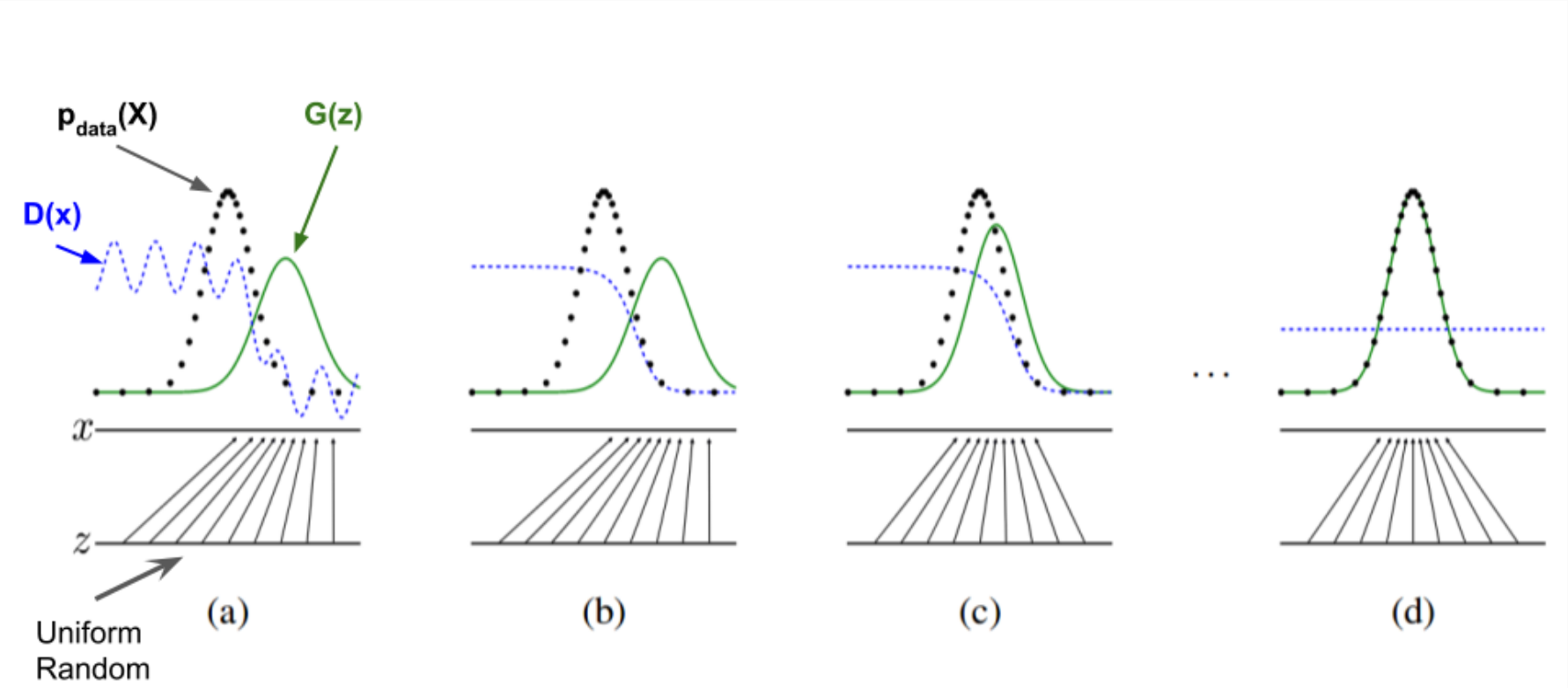


Figura 4: [Goodfellow et al., 2014]

RECOGNA
LABORATORY

Early Results: GAN

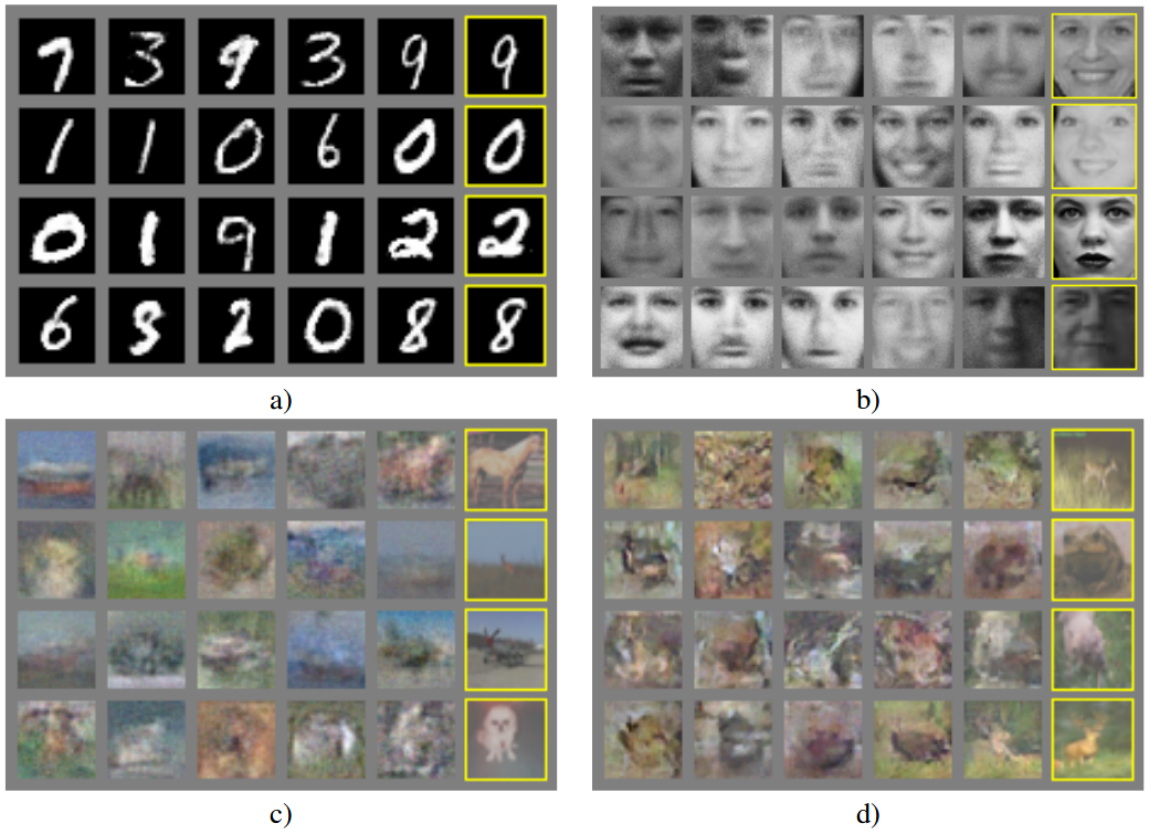


Figura 5: a) MNIST b) TFD c) CIFAR-10 (fully connected model) d) CIFAR-10 (convolutional discriminatorand "deconvolutional" generator. [Goodfellow et al., 2014]

RECOGNA LABORATORY

Deep Convolutional GAN - DCGAN

- Replace any pooling layers with stridden convolutions (discriminator) and transposed convolutions (generator).
- Use batch normalization in both the generator and the discriminator.
- Remove fully connected hidden layers for deeper architectures.
- Use ReLU activation in the generator for all layers except for the output, which uses tanh.
- Use LeakyReLU activation in the discriminator for all layers.

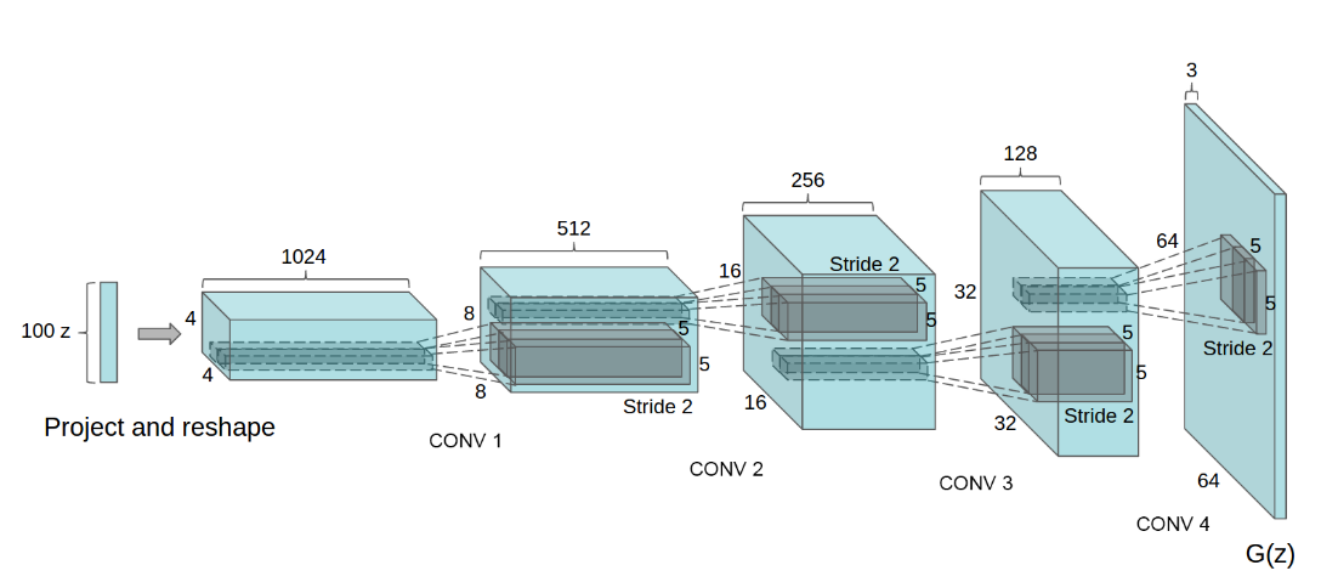


Figura 6: [Radford et al., 2015]

Early Results: DCGAN

- LSUN bedroom generated images [Radford et al., 2015].

RECOGNA LABORATORY

Early Results: DCGAN

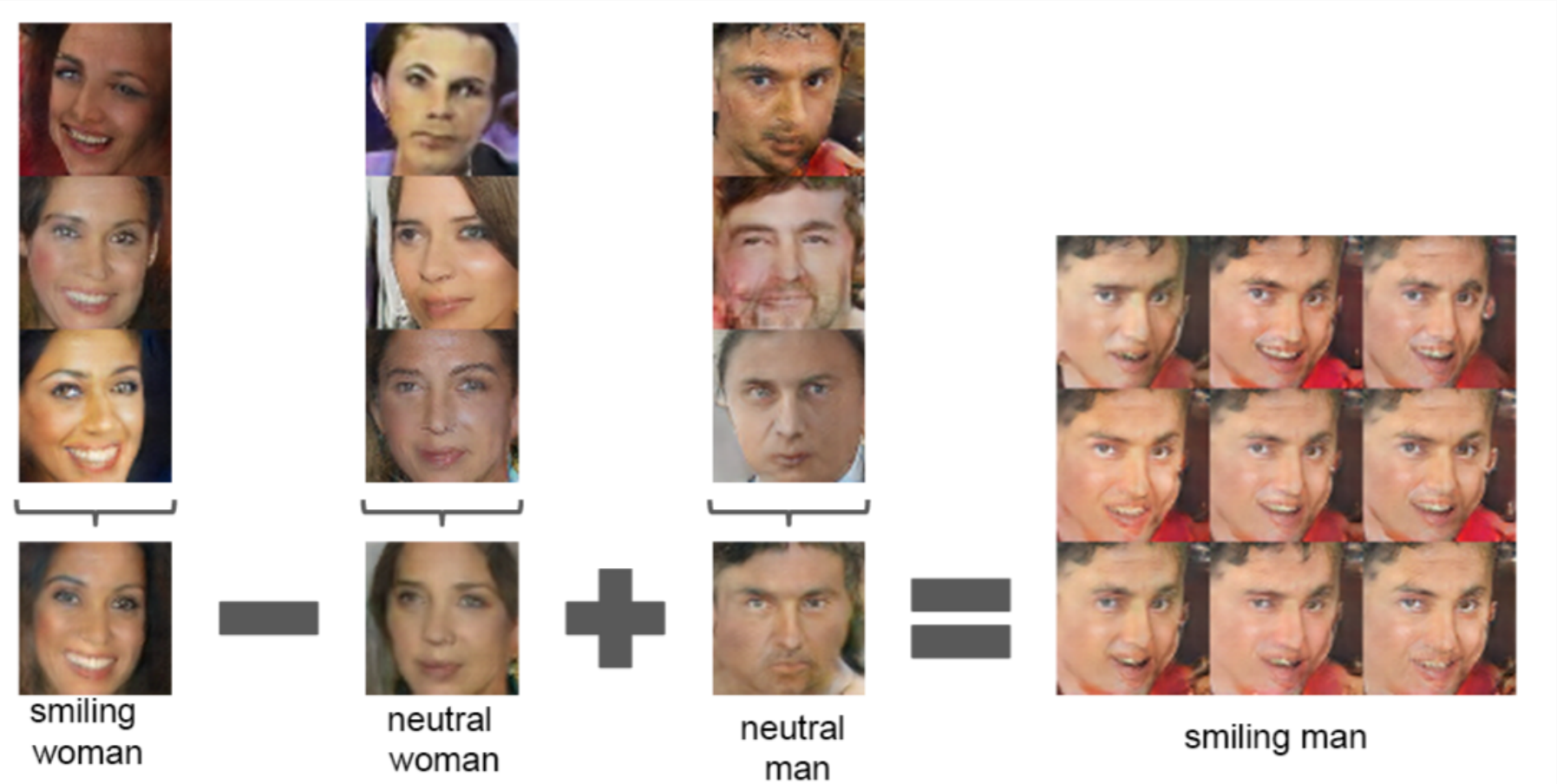


Figura 8: [Radford et al., 2015]

RECOGNA LABORATORY

Early Results: DCGAN

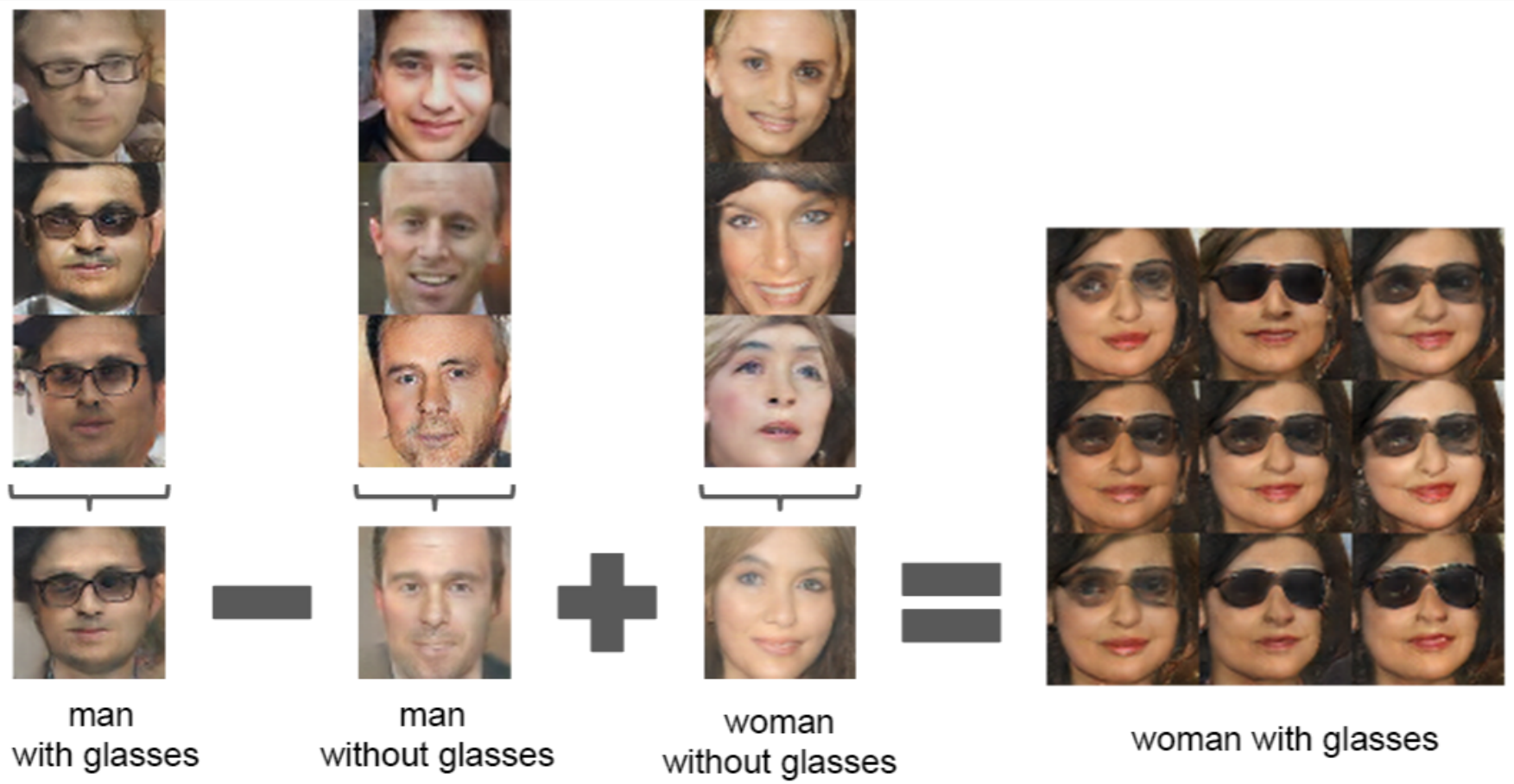


Figura 9: [Radford et al., 2015]

RECOGNA
LABORATORY

Early Results: DCGAN

Figura 10: [Radford et al., 2015]

RECOGNA LABORATORY

GAN downside ....

GAN is amazing, BUT not everything are roses ...

- Non-Convergence
- Mode-Collapsee

RECOGNA
LABORATORY

Non-Convergence

- DL models, in general, involve one player. Thus, it has only one optimization task.

$$\min_{\theta} \mathcal{L}(\theta) \tag{11}$$

- SGD-like algorithms generally converges (at least to a local minimum)

However,

- GAN models involve two or more players.

$$\min_{G} \max_{D} V(G, D) \tag{12}$$

- SGD was not created to find Nash equilibrium.
- There are no garantees that it will converge to the Nash equilibrium.

RECOGNA LABORATORY

Non-Convergence example

- Suppose a minmax game trying trying to optimize $\mathcal{H}(x, y) = xy$:

$$\min_x \max_y \mathcal{H}(x, y) \tag{13}$$

- The infinitesimal step continuous is do not converge to the Nash equilibrium at $x = 0, y = 0$.
- If the state of the system is represented as (x, y) in phase space, the dynamics of the game can be governed by equations.

$$\frac{\partial x}{\partial t} = -\frac{\partial \mathcal{H}(x, y)}{\partial x} = -y \tag{14}$$

$$\frac{\partial y}{\partial t} = \frac{\partial \mathcal{H}(x, y)}{\partial y} = x \tag{15}$$

RECOGNA LABORATORY

Non-Convergence

- Applying $\frac{\partial}{\partial t}$ we have:

$$\frac{\partial^2 y(t)}{\partial t^2} = \frac{\partial x}{\partial t} = -y \rightarrow \frac{\partial^2 y(t)}{\partial t^2} + y(t) = 0 \tag{16}$$

- The solution is oscillation:

$$x(t) = x(0)\cos t - y(0)\sin t \tag{17}$$

$$x(t) = x(0)\sin t + y(0)\cos t \tag{18}$$

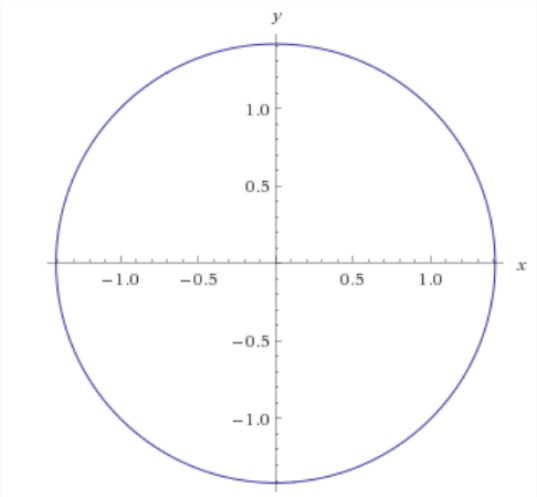


Figura 11: From Nips [Goodfellow, 2016]

RECOGNA LABORATORY

GAN Mode-Collapse

- GAN can produce good data but only a few of them. Thus, Discriminator can't tag them as fake.

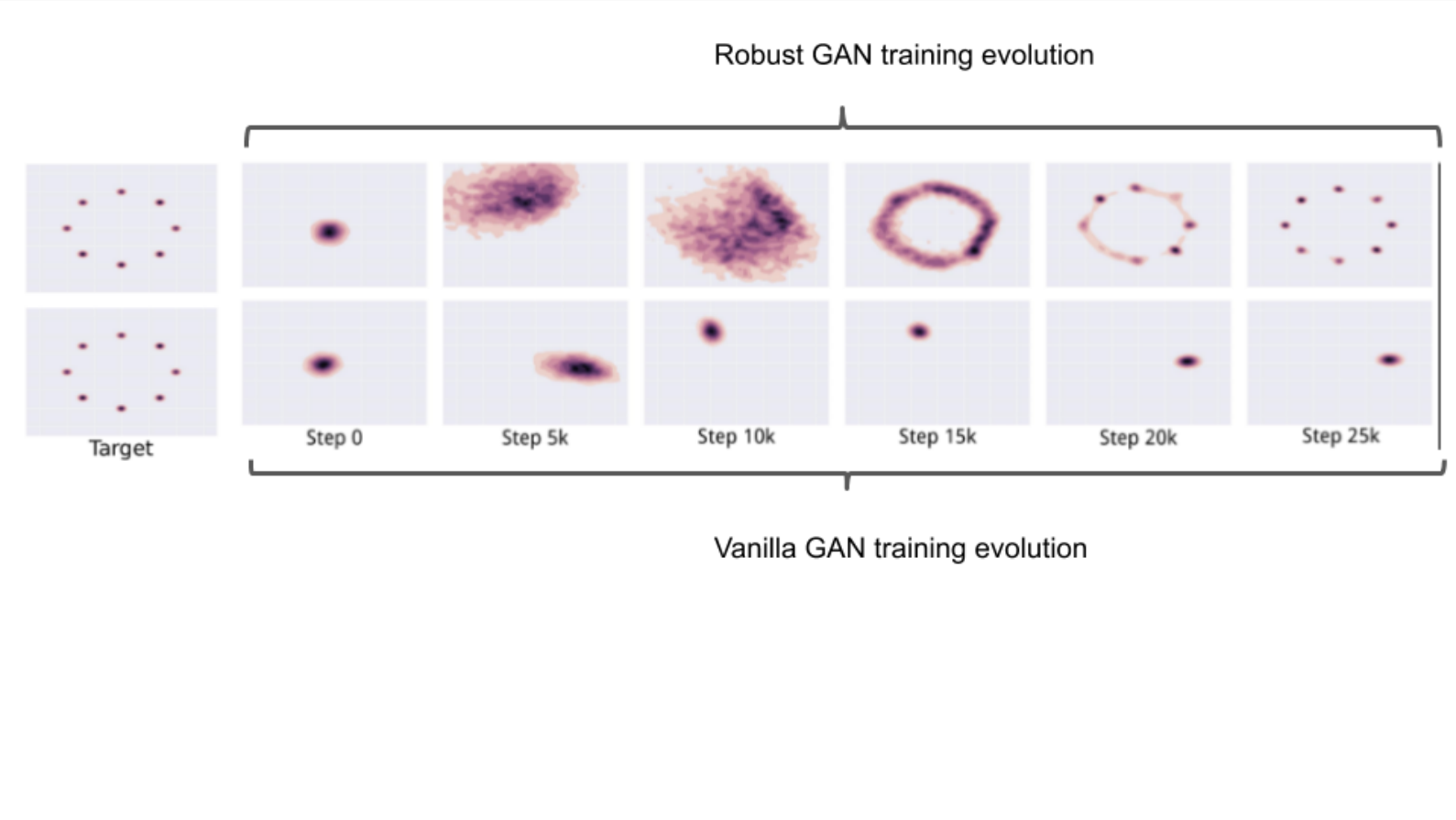


Figura 12: Adapted from [Metz et al., 2016]

RECOGNA LABORATORY

GAN Mode Collapse

There are some approaches to mitigate the issue.

- Minibatch Discrimination
  - Let the Discriminator look at the entire batch instead of single examples.
  - If there is a lack of diversity, it will mark the examples as fake.
- Feature matching - provide a new objective for the generator that prevents it from overtraining on the current discriminator.

$$\mathcal{L} = \min_G ||\mathbb{E}_{p_{data}(x)}\left[f(x)\right] - \mathbb{E}_{p_G(z)}\left[f(G(z))\right]||_2^2 \tag{19}$$

  Where $f(x)$ is a feature from intermediary layer of the Discriminator.
- Reference Batch Normalization
  - Choose a fixed minibatch $R = \{x_1, x_2, x_3...\}$.
  - Normalize all batches with $y_i = \frac{x_i - \mu_R}{\sigma_R^2 + \epsilon}$

RECOGNA LABORATORY

Other Training Tips

- One side soft-label eg.: Real =0.9 and Fake = 0.0.
  - Does not reduce classification accuracy, only confidence.
  - Prevents discriminator from giving very large gradient signal to generator.
- Use Batch Norm [Ioffe and Szegedy, 2015].

$$y_i = \gamma \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \beta \tag{20}$$

  - Construct different mini-batches for real and fake.
- Occasionally flip the labels when training the discriminator: Real = Fake, Fake = Real
- Use LeakyReLU.
- Use the ADAM Optimizer*.
- More tips on https://github.com/soumith/ganhacks .
- See [Goodfellow, 2016].

RECOGNA LABORATORY

WGAN

- $KL(q||p)$ measure diverges if $q(x)$ and $p(x)$ are not overlaping: $KL(q||p) = \infty$.
- Alternatively, Jensen–Shannon measure.

$$JS(q||p) = \frac{1}{2}\int q(x)\frac{q(x)}{M(x)}dx + \frac{1}{2}\int p(x)\frac{p(x)}{M(x)}dx, \qquad (21)$$

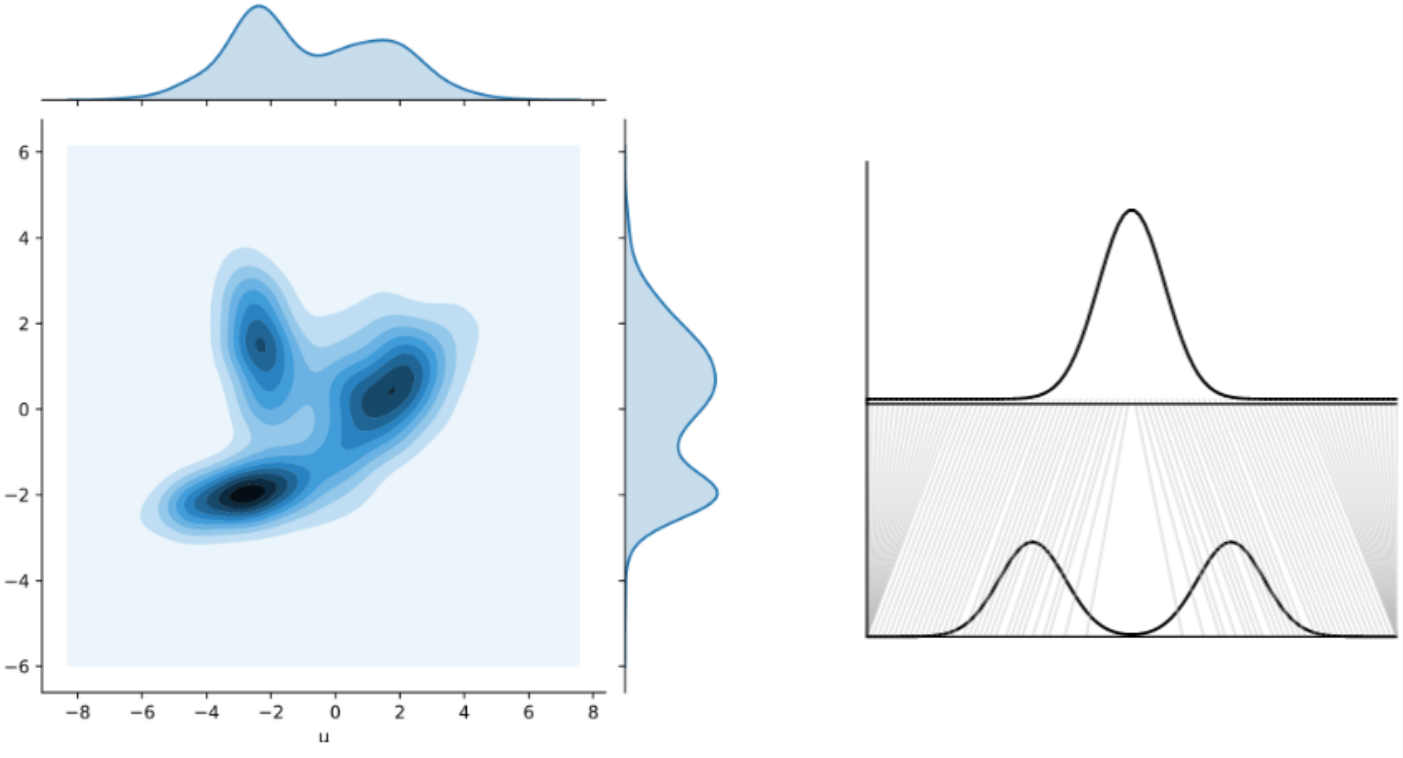
  where $M(x) = \frac{q(x)+p(x)}{2}$.
- $q(x)$ and $p(x)$ are not overlaping $JS(q||p) = \log 2$

RECOGNA LABORATORY

WGAN - Earth Mover Distance

Earth moving distance is the minimum amount of efforts to transform (moving) from $q(x)$ to $p(x)$ [Kolouri et al., 2017].

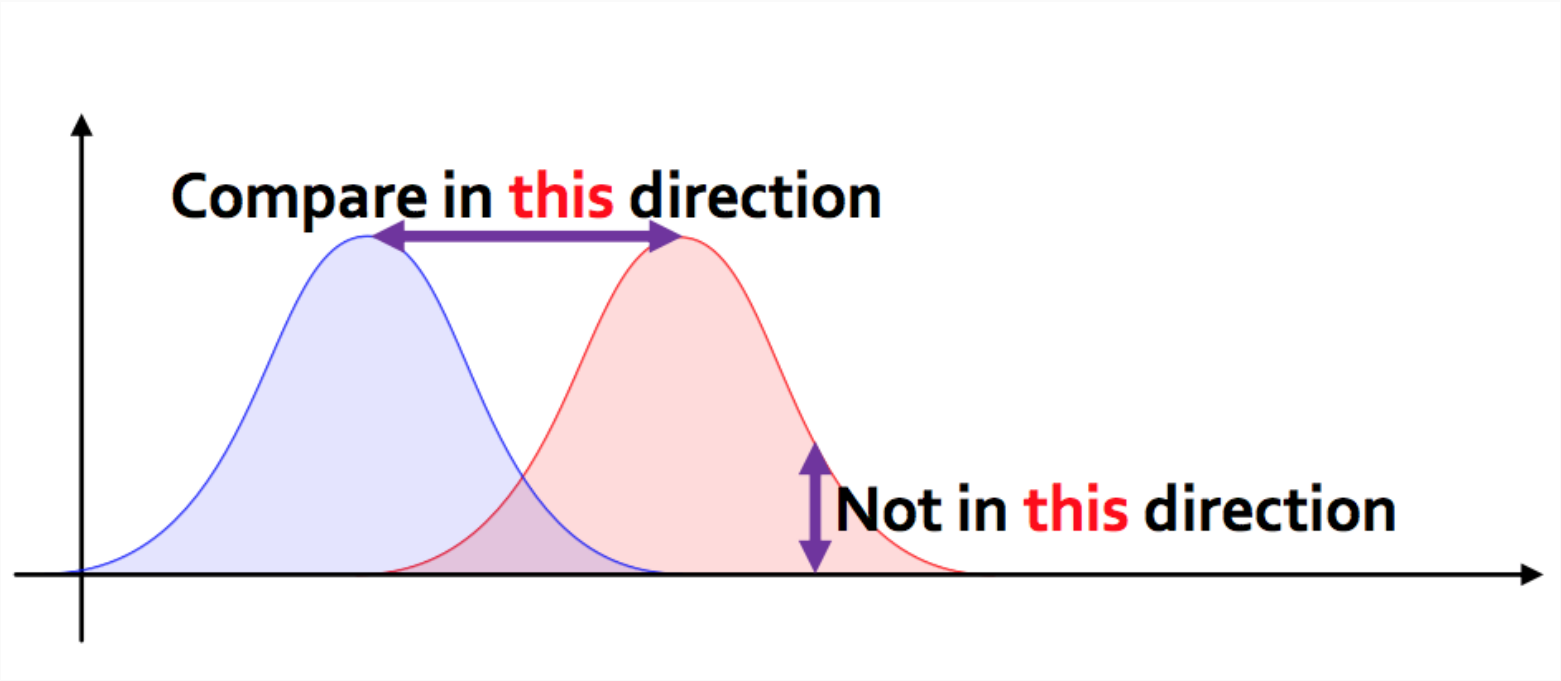
$$W(q,p) = \inf_{\gamma \in \Pi} \mathbb{E}_{(x,y)\sim\gamma}[|x-y|] \tag{22}$$



Adapted from:http://www.stat.cmu.edu/ larry/=sml/Opt.pdf

RECOGNA LABORATORY

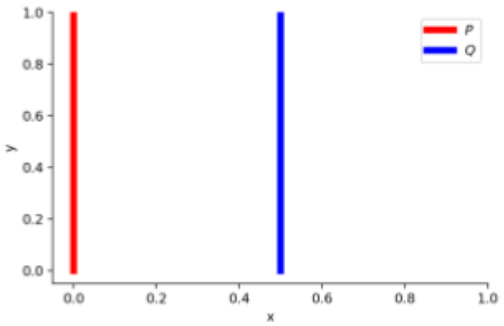
WGAN - Earth Mover Distance

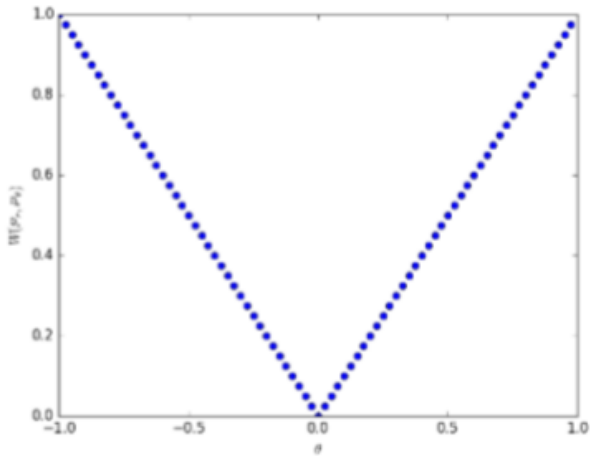


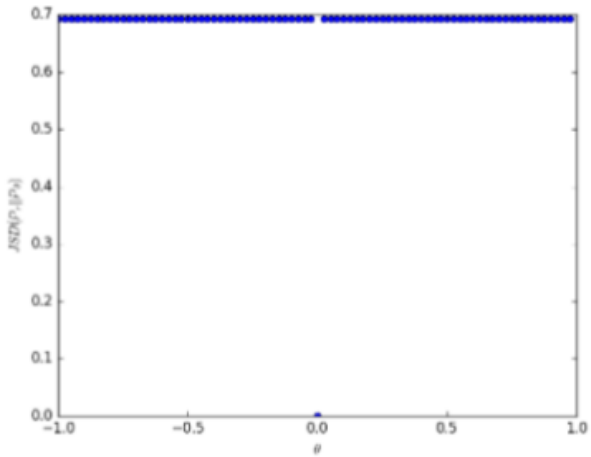
Credit: https://jeremykun.com/2018/03/05/earthmover-distance/

RECOGNA LABORATORY

WGAN - Earth Mover Distance

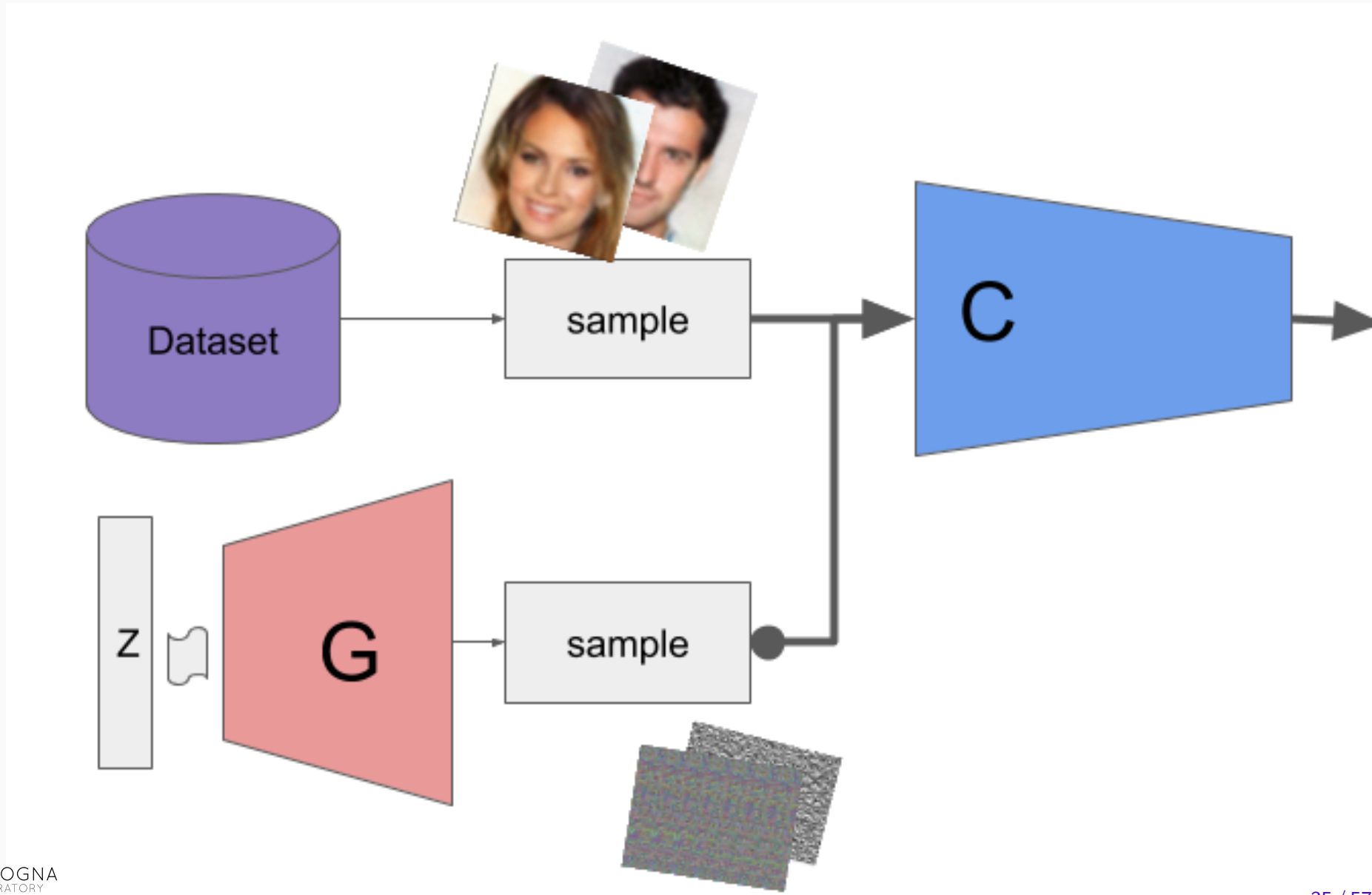






RECOGNA
LABORATORY

WGAN



RECOGNA
LABORATORY

WGAN

$$\mathcal{L} = \max_{||w||_L \leq K} \mathbb{E}_{x \sim p(x)}[F_w(x)] - \mathbb{E}_{x \sim p(x)}[F_w(G_{g_w}(z))] \tag{23}$$

Algorithm 1 WGAN, our proposed algorithm. All experiments in the paper used the default values $\alpha = 0.00005$, $c = 0.01$, $m = 64$, $n_{\text{critic}} = 5$.

Require: : α, the learning rate. c, the clipping parameter. m, the batch size. n_{critic}, the number of iterations of the critic per generator iteration.

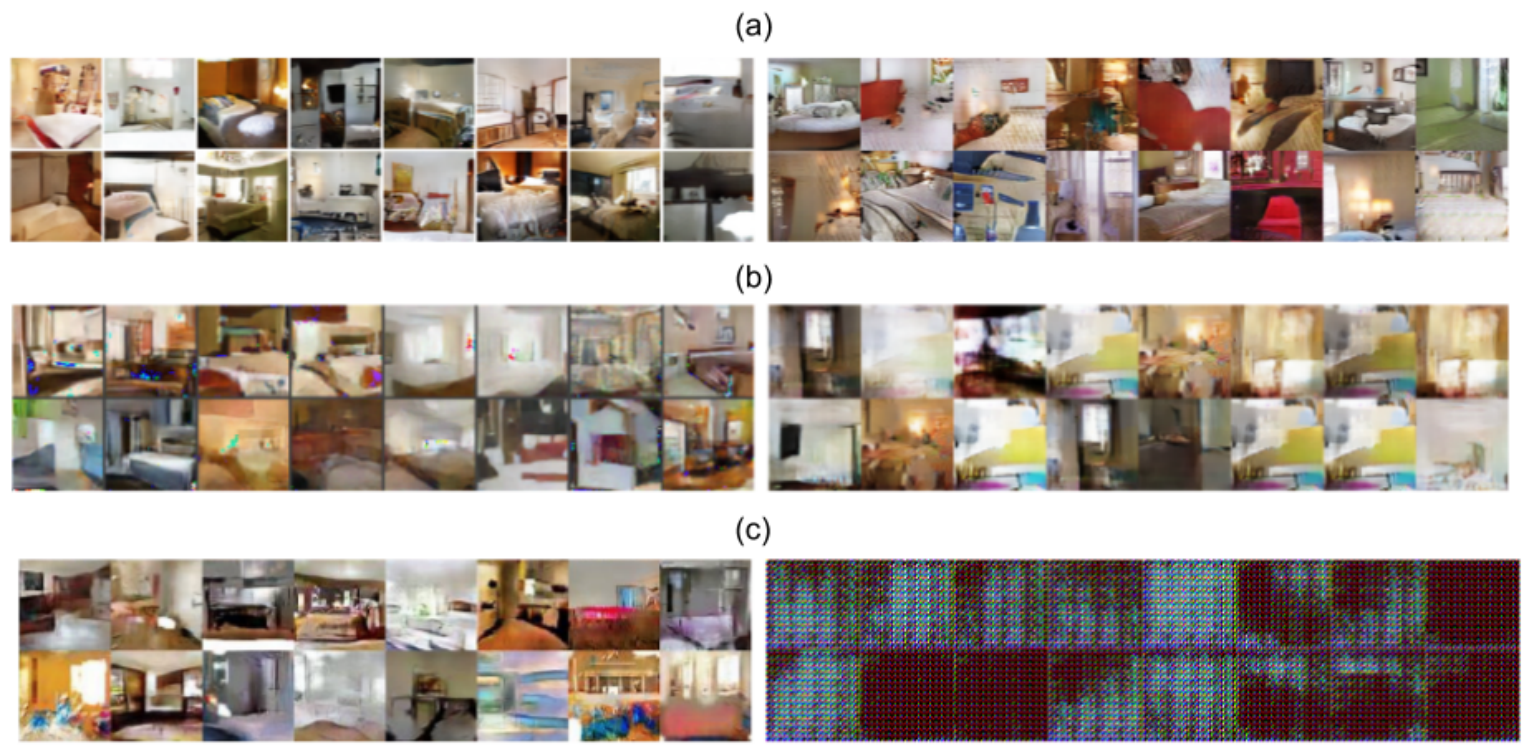
Require: : w_0, initial critic parameters. θ_0, initial generator's parameters.

while θ has not converged **do**
  for $t = 0, ..., n_{\text{critic}}$ **do**
    Sample $\{x^{(i)}\}_{i=1}^{m} \sim \mathbb{P}_r$ a batch from the real data.
    Sample $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ a batch of prior samples.
    $g_w \leftarrow \nabla_w \left[\frac{1}{m}\sum_{i=1}^{m} f_w(x^{(i)}) - \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))\right]$
    $w \leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w)$
    $w \leftarrow \text{clip}(w, -c, c)$
  end for
  Sample $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ a batch of prior samples.
  $g_\theta \leftarrow -\nabla_\theta \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))$
  $\theta \leftarrow \theta - \alpha \cdot \text{RMSProp}(\theta, g_\theta)$
end while

RECOGNA LABORATORY

WGAN

$$\mathcal{L} = \max_{||w||_L \leq K} \mathbb{E}_{x \sim p(x)}[F_w(x)] - \mathbb{E}_{x \sim p(x)}[F_w(G_{g_w}(z))] \tag{24}$$



RECOGNA LABORATORY

Applications

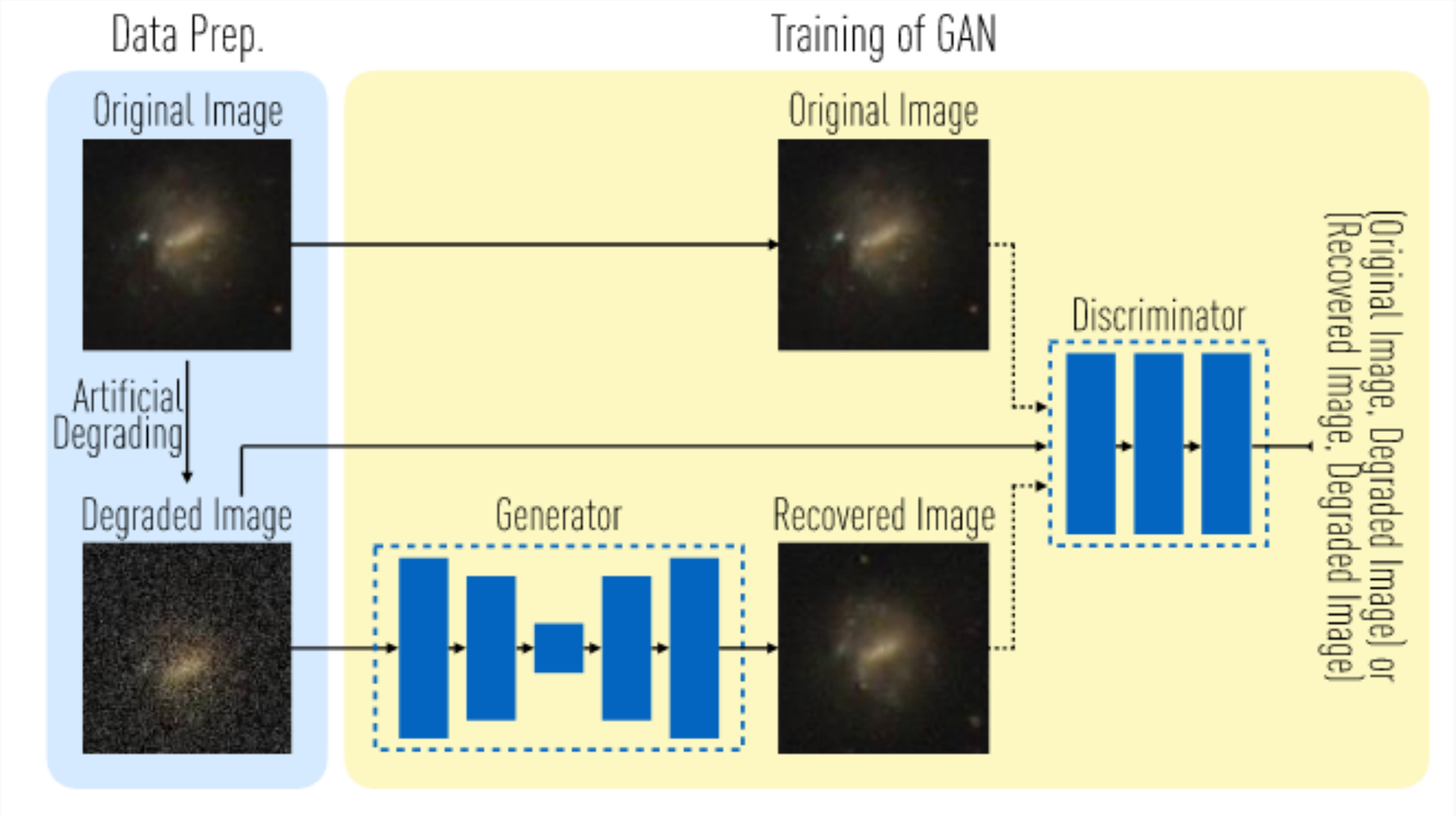


Figura 13: [Schawinski et al., 2017]

RECOGNA LABORATORY

Applications

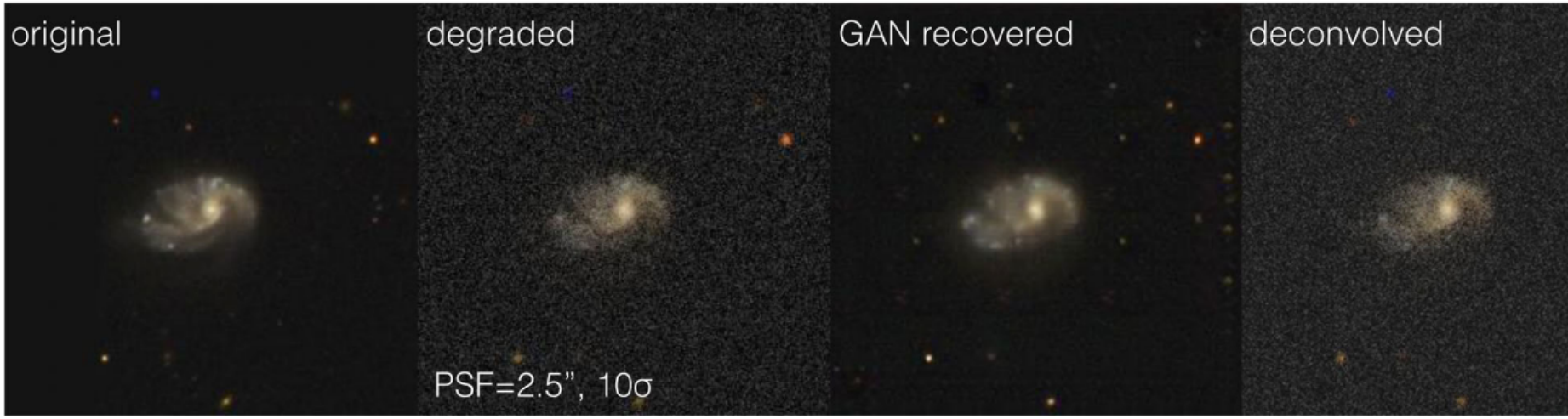


Figura 14: [Schawinski et al., 2017]

RECOGNA LABORATORY

Applications: Galaxy Deblending

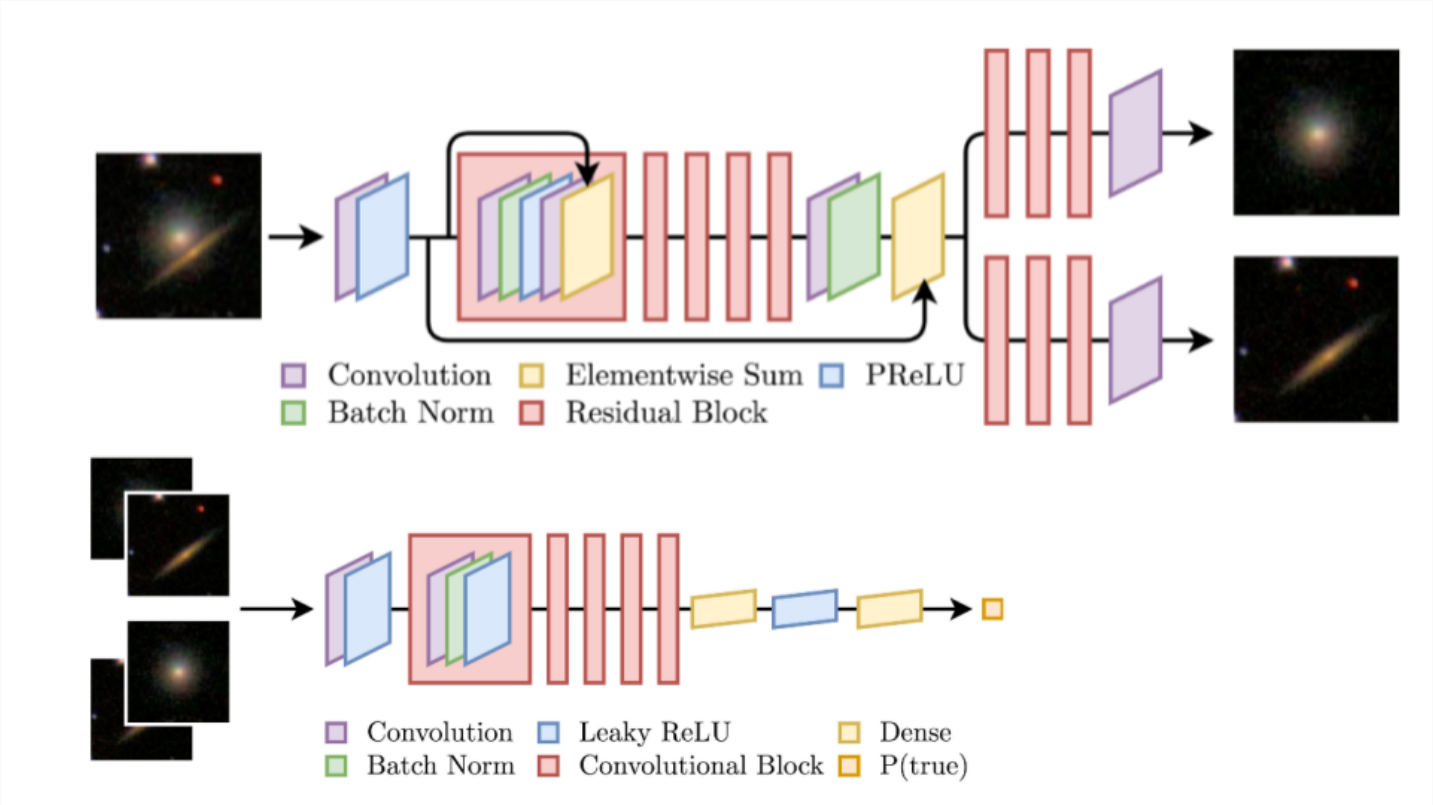


Figura 15: [Reiman and Göhre, 2019]

Applications: Galaxy Deblending

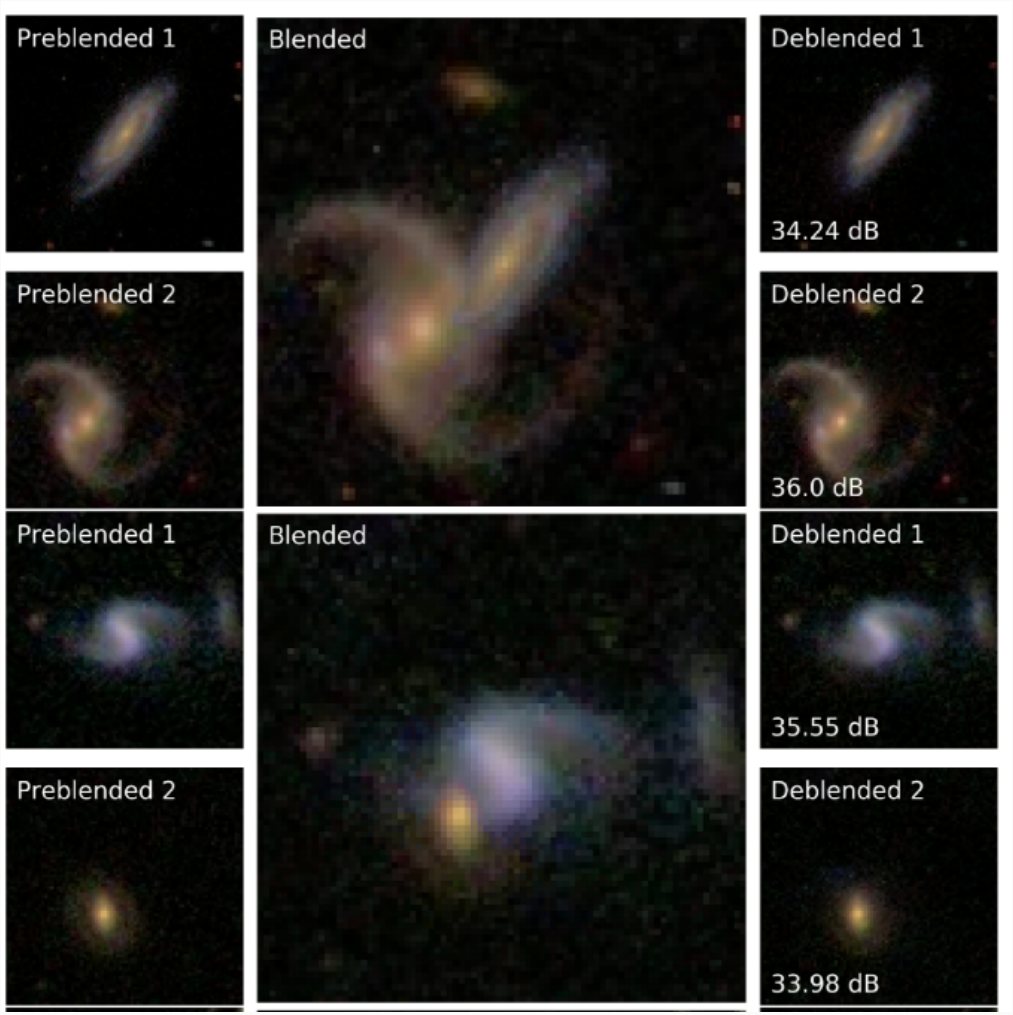


Figura 16: [Reiman and Göhre, 2019]

RECOGNA LABORATORY

Applications: Medical Image

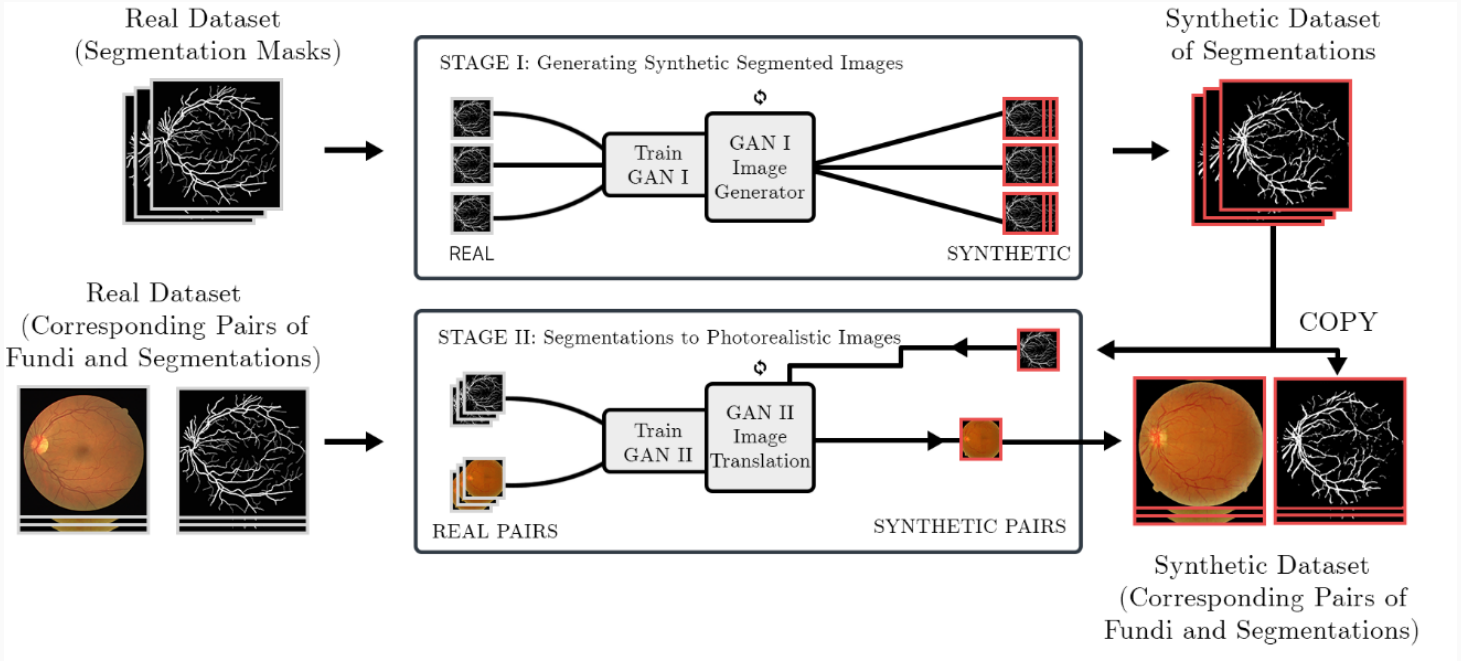


Figura 17: [Guibas et al., 2017]

Applications: Medical Image

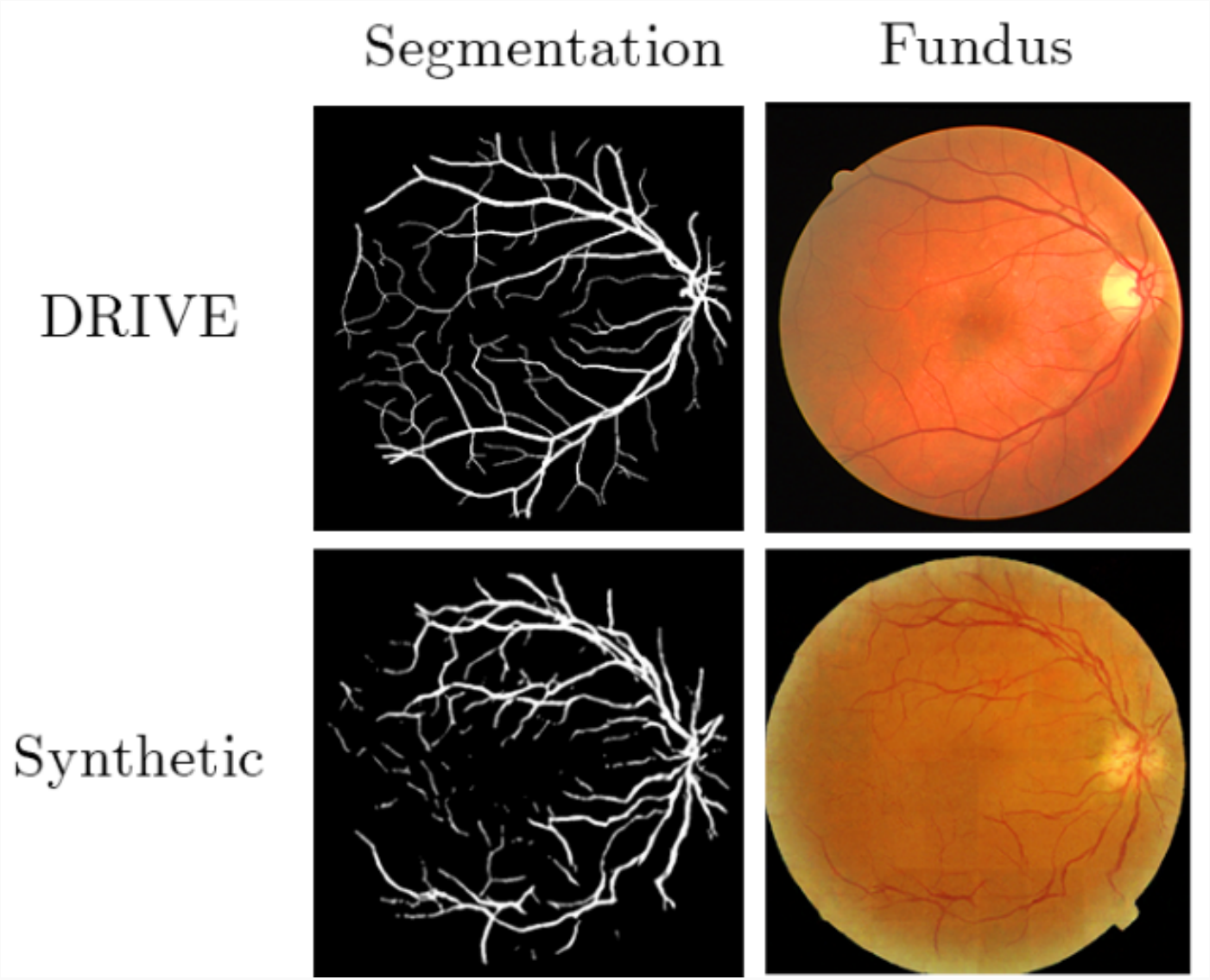


Figura 18: [Guibas et al., 2017]

RECOGNA LABORATORY

Applications: Molecule Generation

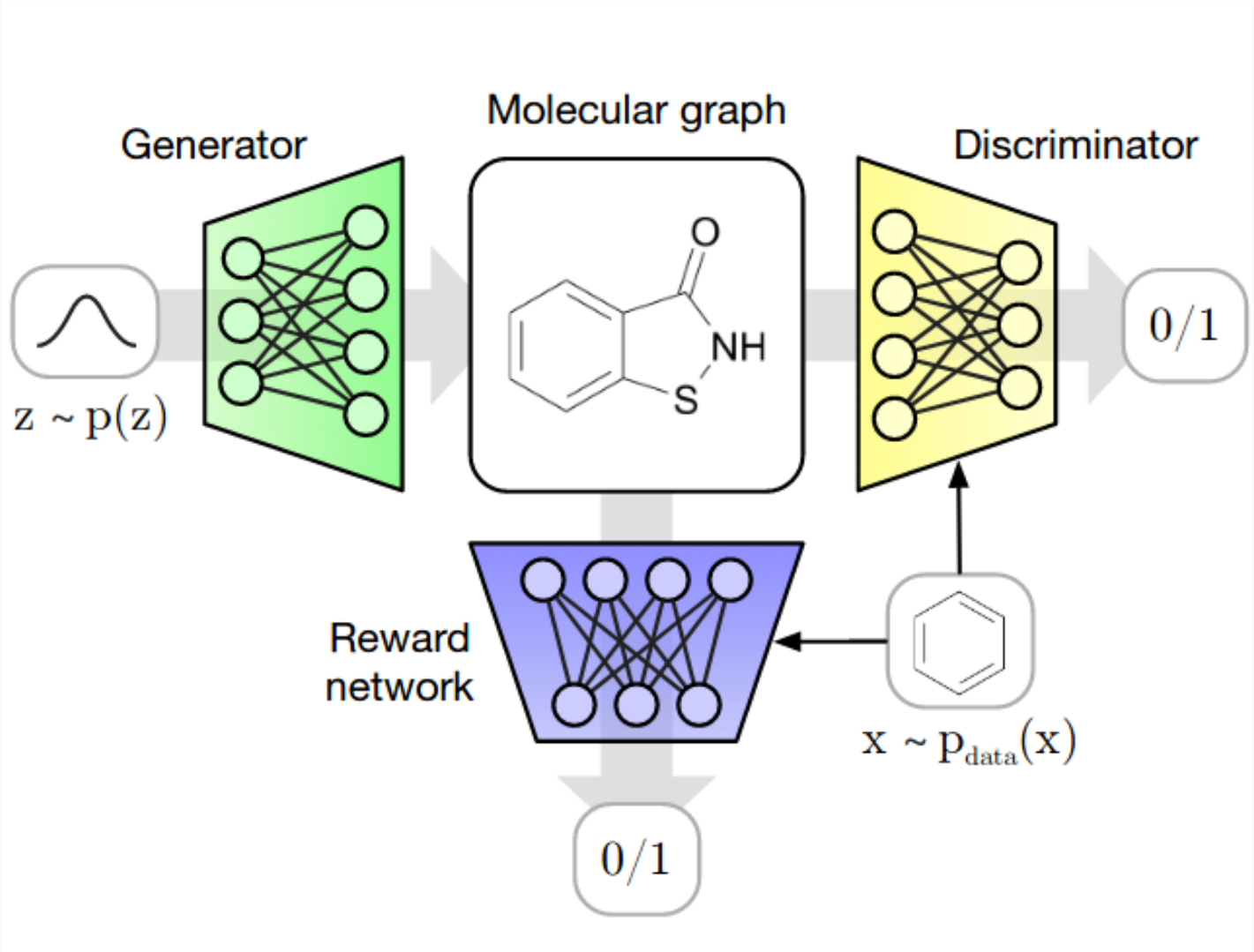


Figura 19: [De Cao and Kipf, 2018]

RECOGNA LABORATORY

Applications: Particle Calorimeter

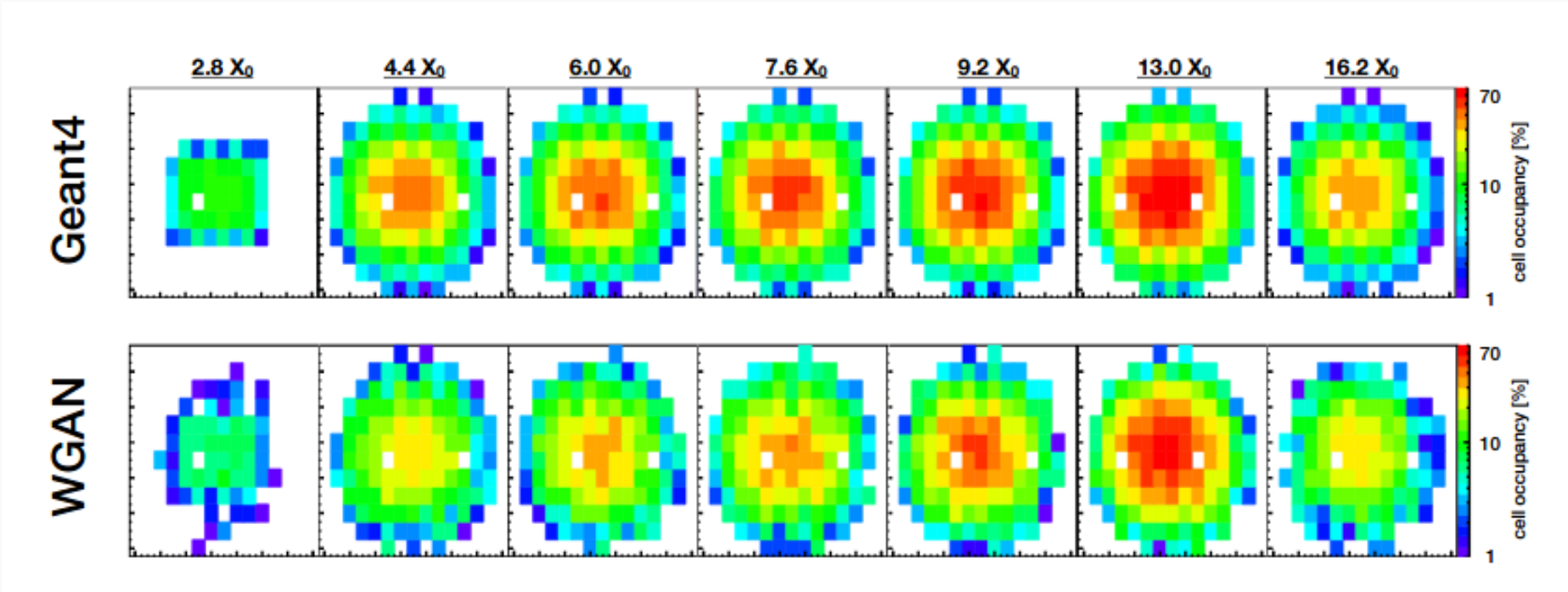


Figura 20: [Erdmann et al., 2019]

RECOGNA LABORATORY

Applications: Text to Image Synthesys

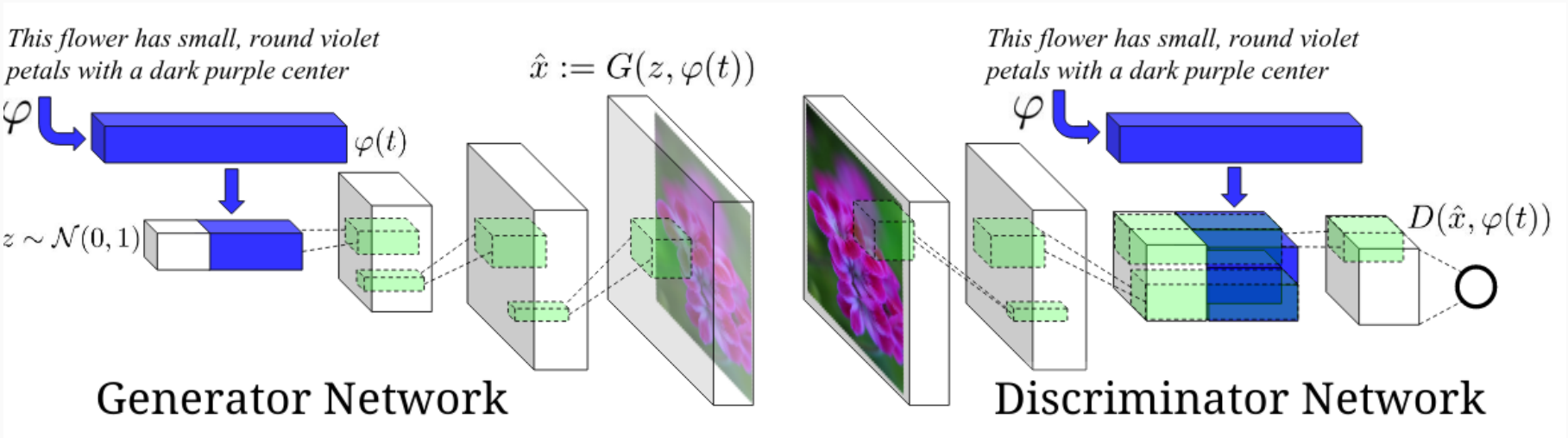


Figura 21: [Reed et al., 2016]

Applications: Text to Image Synthesys

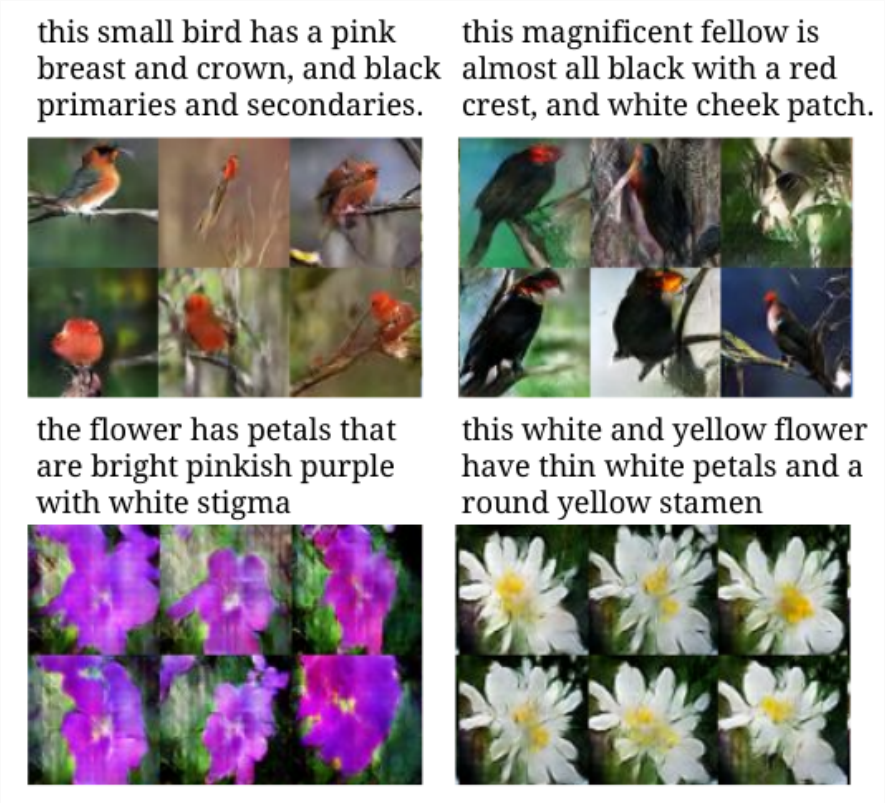


Figura 22: Examples of generated images from text descriptions.Left: captions are from zero-shot (held out) categories, unseentext. Right: captions are from the training set[Reed et al., 2016]

RECOGNA LABORATORY

Applications: Transport Phenomena

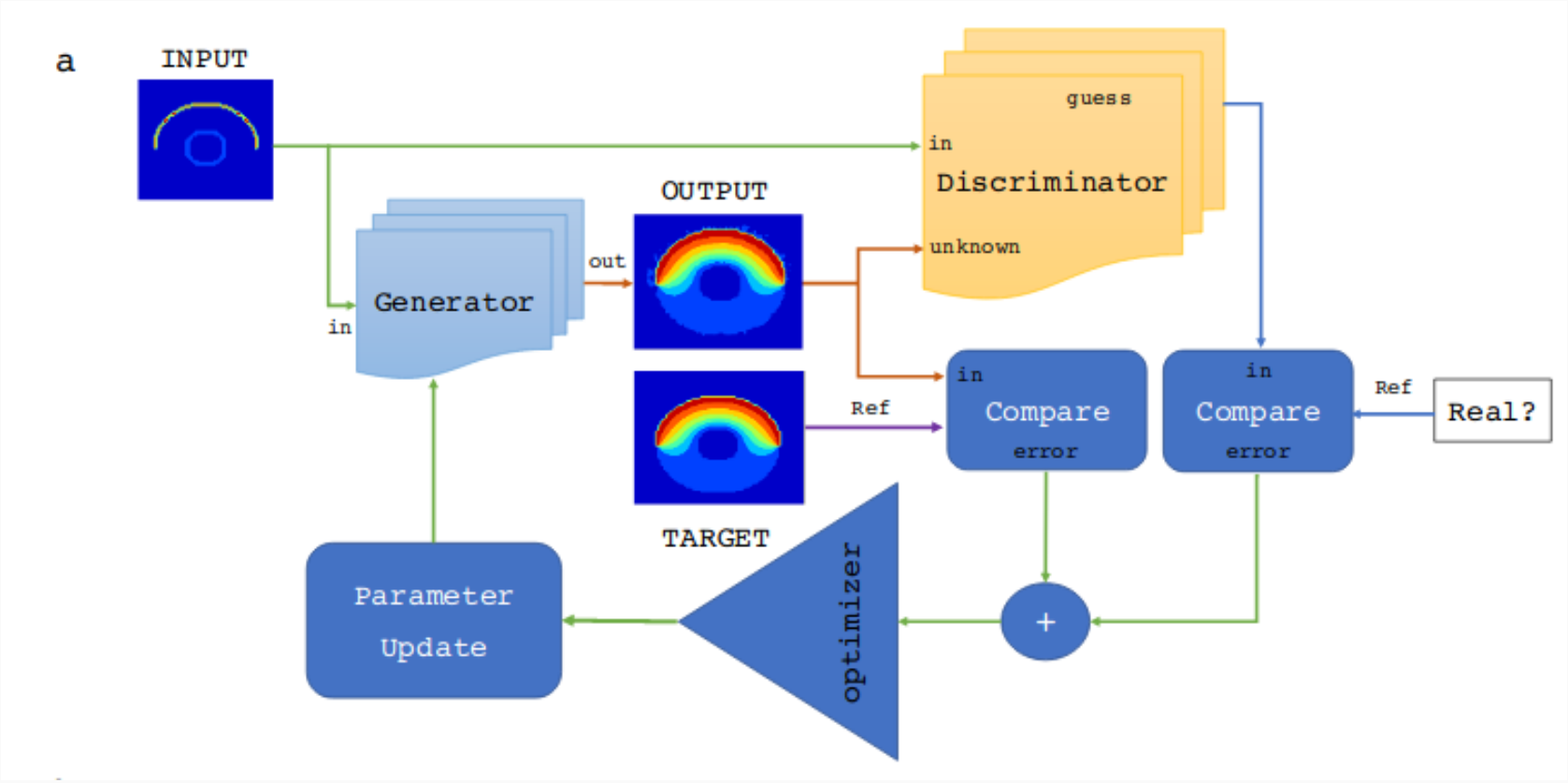
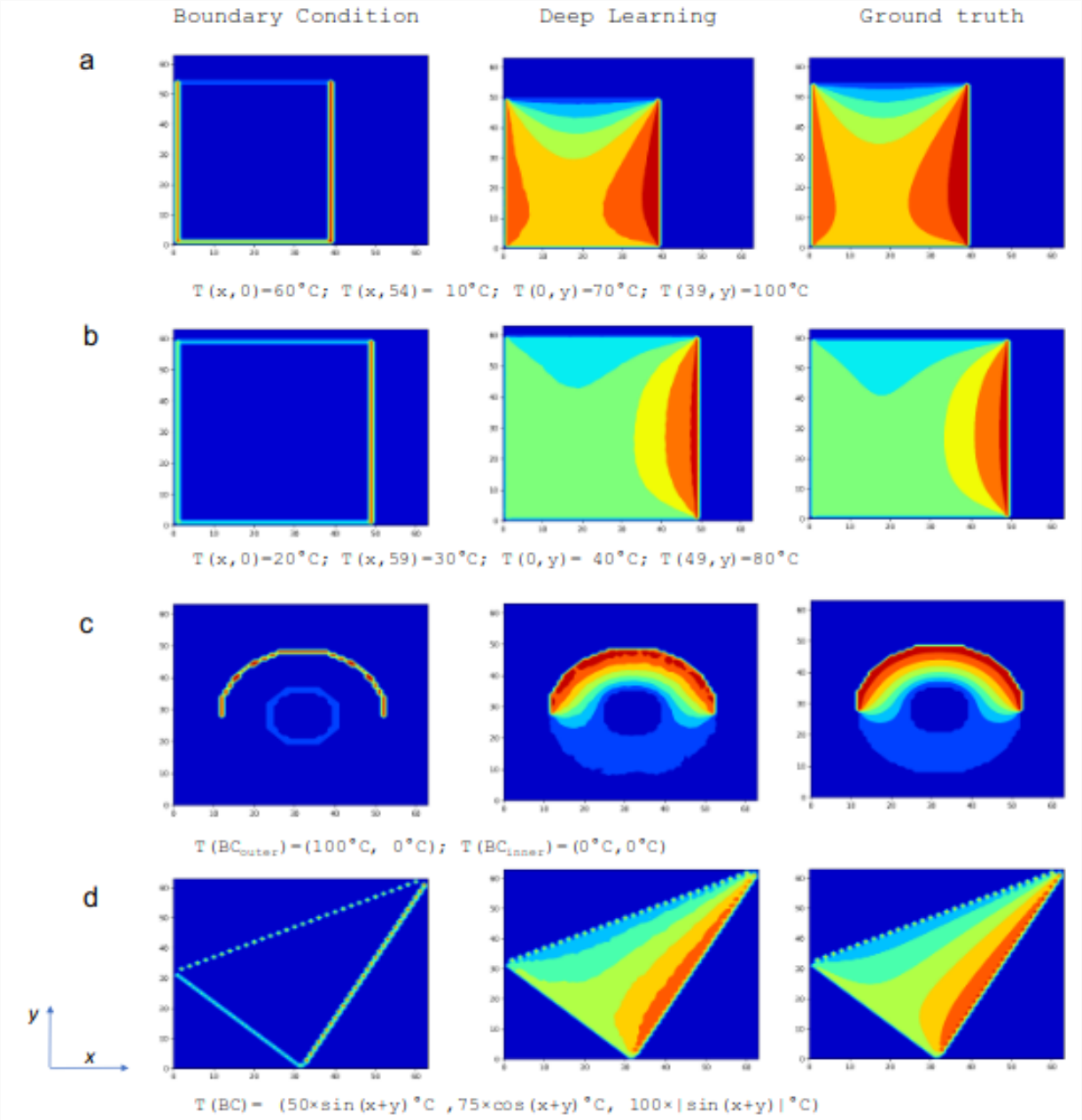


Figura 23: [Farimani et al., 2017]

Applications: Transport Phenomena



RECOGNA LABORATORY

Other Architectures

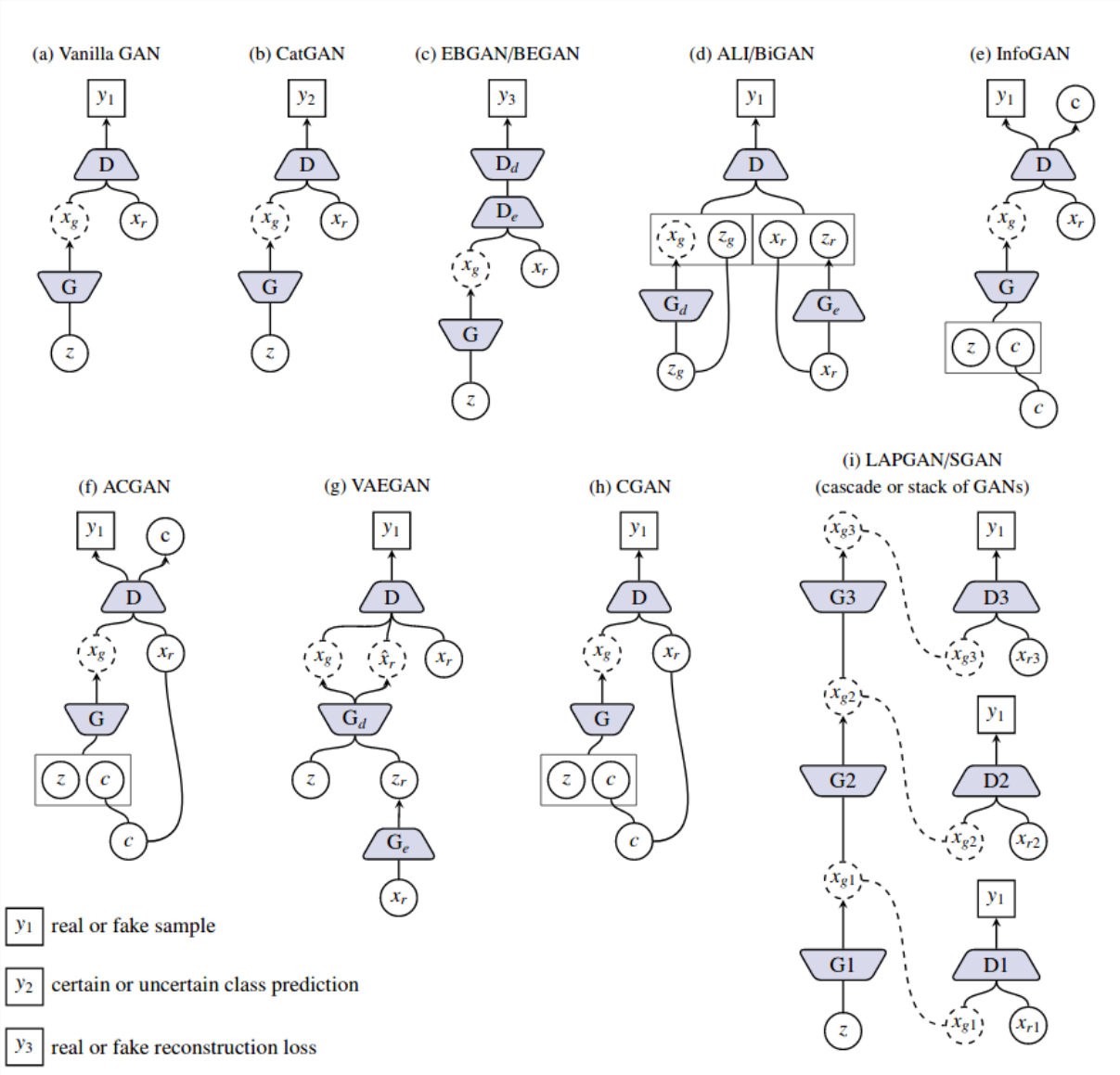


Figura 25: [Reed et al., 2016]

Conclusions

- Inovative.
- Able to Generate Complex Pattern.
- Can be integrated with others architectures.

- Complex.
- Difficult to train.

References i

De Cao, N. and Kipf, T. (2018).
Molgan: An implicit generative model for small molecular graphs.
arXiv preprint arXiv:1805.11973.

Erdmann, M., Glombitza, J., and Quast, T. (2019).
Precise simulation of electromagnetic calorimeter showers using a wasserstein generative adversarial network.
Computing and Software for Big Science, 3(1):4.

Farimani, A. B., Gomes, J., and Pande, V. S. (2017).
Deep learning the physics of transport phenomena.
arXiv preprint arXiv:1709.02432.

References ii


Goodfellow, I. (2016).
Nips 2016 tutorial: Generative adversarial networks.
arXiv preprint arXiv:1701.00160.


Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014).
Generative adversarial nets.
In *Advances in neural information processing systems*, pages 2672–2680.


Guibas, J. T., Virdi, T. S., and Li, P. S. (2017).
Synthetic medical images from dual generative adversarial networks.
arXiv preprint arXiv:1709.01872.

RECOGNA LABORATORY

References iii


Ioffe, S. and Szegedy, C. (2015).
Batch normalization: Accelerating deep network training by reducing internal covariate shift.
arXiv preprint arXiv:1502.03167.


Kolouri, S., Park, S. R., Thorpe, M., Slepcev, D., and Rohde, G. K. (2017).
Optimal mass transport: Signal processing and machine-learning applications.
IEEE signal processing magazine, 34(4):43–59.


Li, Y., Swersky, K., and Zemel, R. (2015).
Generative moment matching networks.
In *International Conference on Machine Learning*, pages 1718–1727.

RECOGNA LABORATORY

References iv


Metz, L., Poole, B., Pfau, D., and Sohl-Dickstein, J. (2016).
Unrolled generative adversarial networks (2016).
arXiv preprint arXiv:1611.02163.


Nair, V. and Hinton, G. E. (2010).
Rectified linear units improve restricted boltzmann machines.
In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814.


Radford, A., Metz, L., and Chintala, S. (2015).
Unsupervised representation learning with deep convolutional generative adversarial networks.
arXiv preprint arXiv:1511.06434.

RECOGNA LABORATORY

References v

Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. (2016).
Generative adversarial text to image synthesis.
arXiv preprint arXiv:1605.05396.

Reiman, D. M. and Göhre, B. E. (2019).
Deblending galaxy superpositions with branched generative adversarial networks.
Monthly Notices of the Royal Astronomical Society, 485(2):2617–2627.

RECOGNA LABORATORY

References vi

Schawinski, K., Zhang, C., Zhang, H., Fowler, L., and Santhanam, G. K. (2017).
Generative adversarial networks recover features in astrophysical images of galaxies beyond the deconvolution limit.
Monthly Notices of the Royal Astronomical Society: Letters, 467(1):L110–L114.

RECOGNA LABORATORY