UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Advanced Deep Learning
2021-2022

Lecture #5: AutoEncoders, Adversarial Examples, Generative Adversarial Networks (GANs)

Frederic Precioso

UNIVERSITÉ CÔTE D'AZUR

(DENOISING) STACKED AUTOENCODER

Deep Autoencoders

- A deep Autoencoder is constructed by extending the encoder and decoder of autoencoder with multiple hidden layers.

- Gradient vanishing problem: the gradient becomes too small as it passes back through many layers

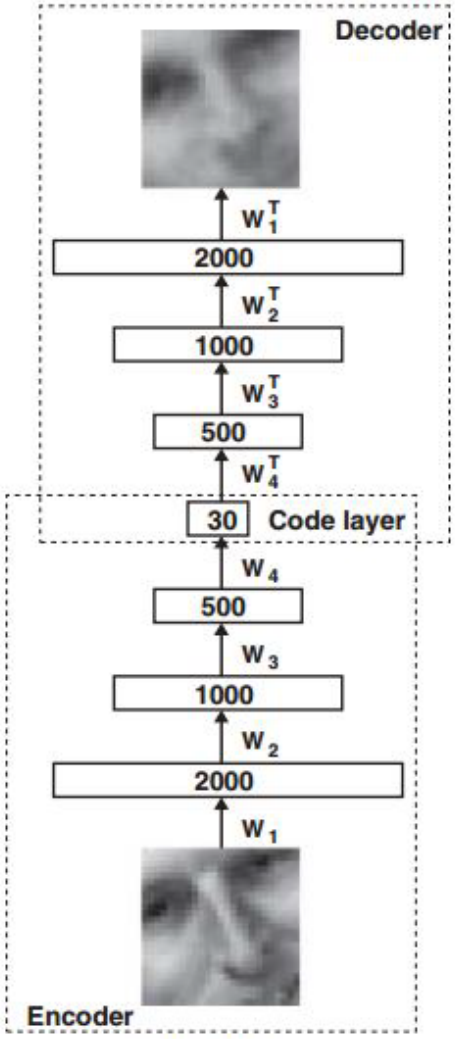


Diagram from (Hinton and Salakhutdinov, 2006)

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Training Deep Autoencoders

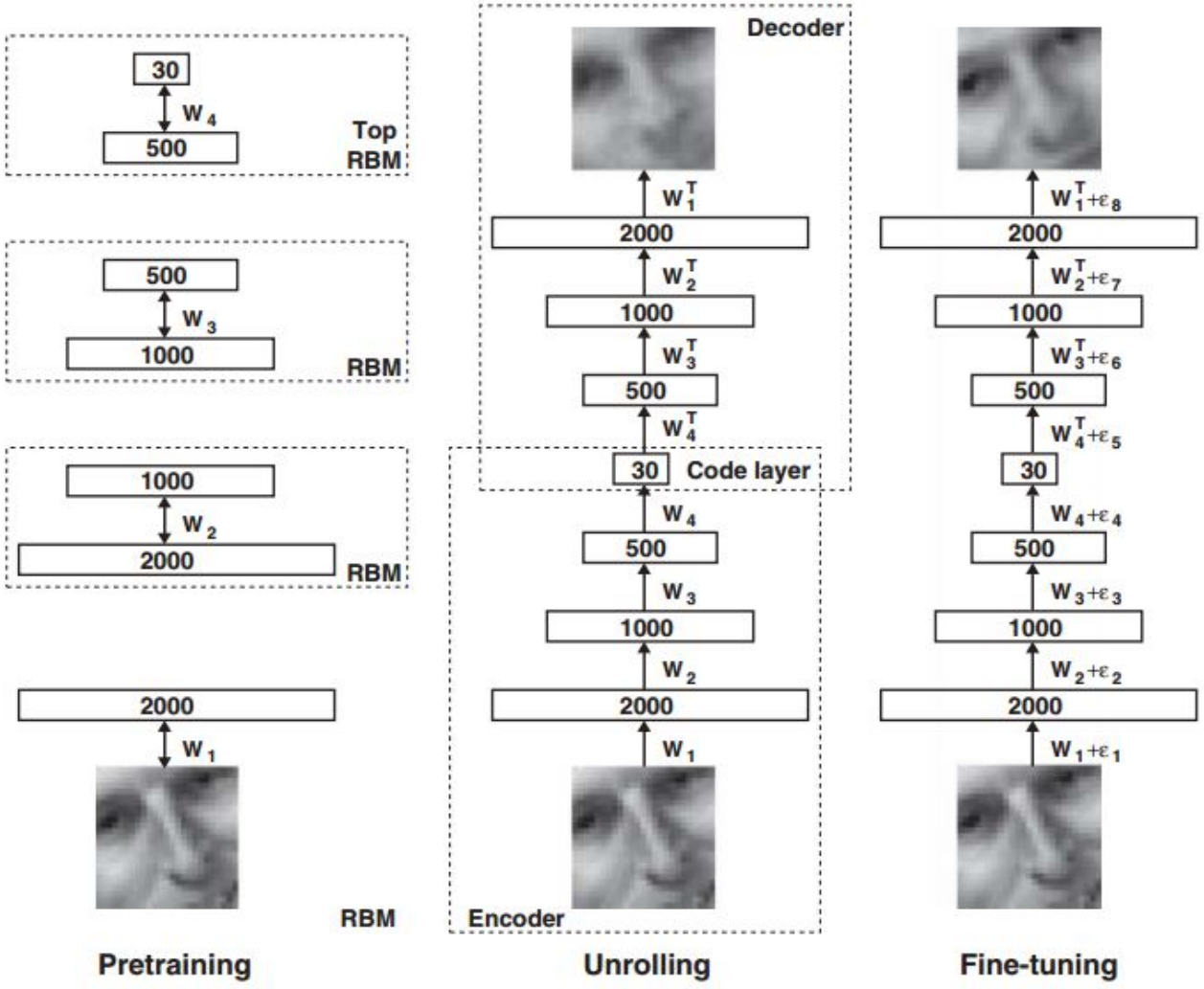
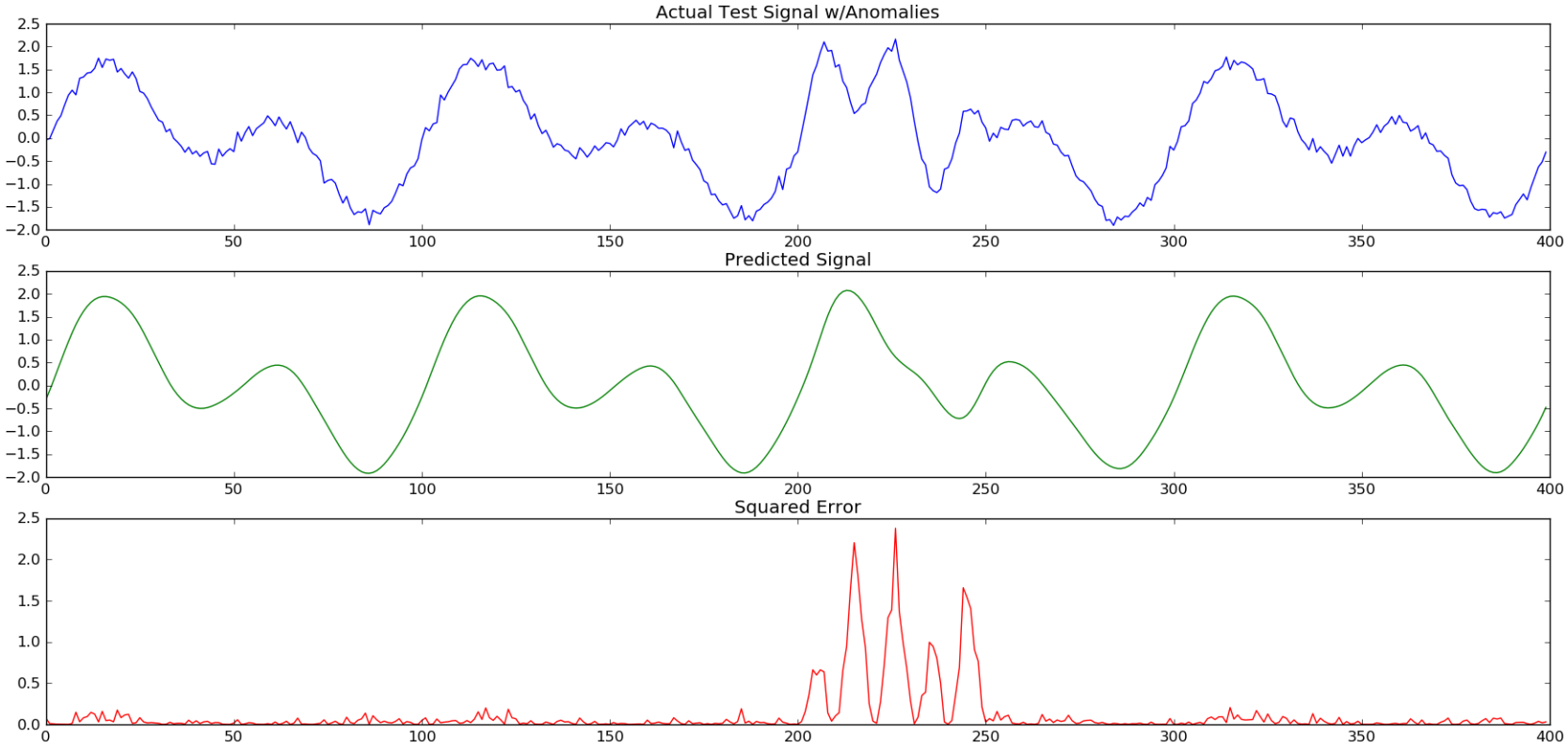


Diagram from (Hinton and Salakhutdinov, 2006)

UNIVERSITÉ CÔTE D'AZUR

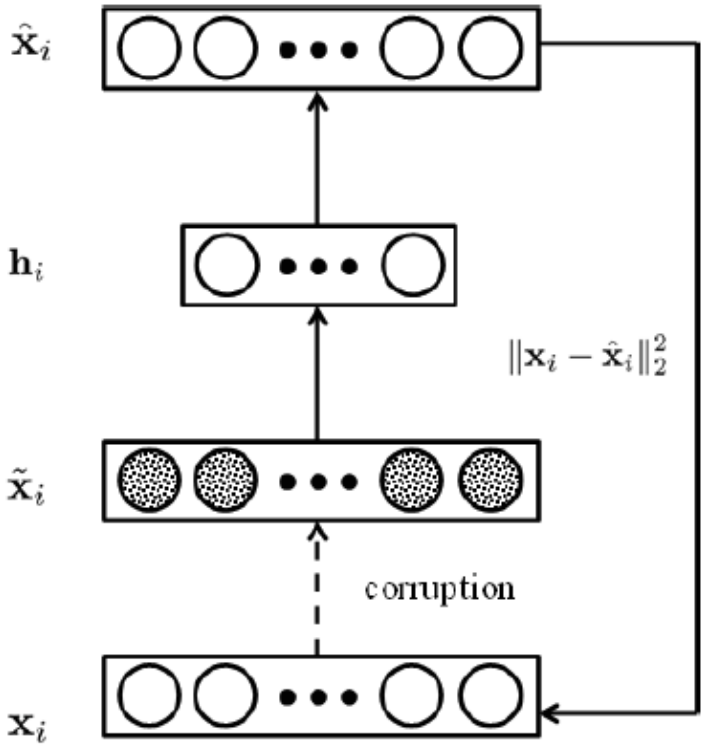
univ-cotedazur.fr

Time Series Anomaly Detection



Denoising Autoencoders

- By adding stochastic noise to the, it can force Autoencoder to learn more robust features.



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Training Denoising Autoencoder

The loss function of Denoising autoencoder:

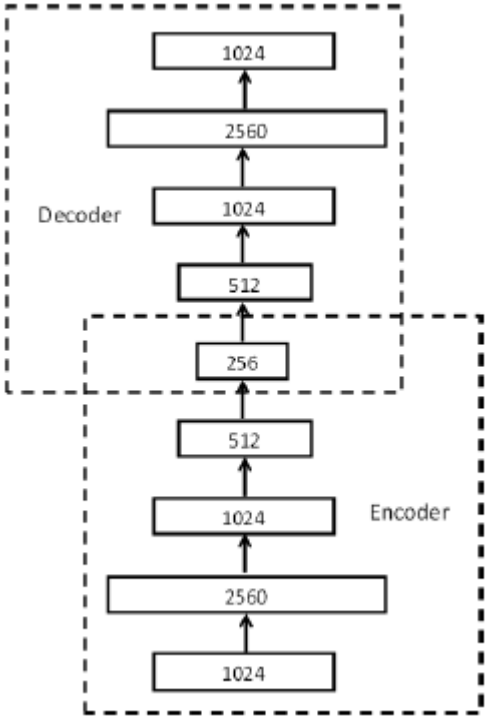
$$\min_{\mathbf{W}_1,\mathbf{W}_2,\mathbf{b}_1,\mathbf{b}_2} \sum_{\ell} \|\mathbf{x}^{(\ell)} - \hat{\mathbf{x}}^{(\ell)}\|_2^2 + \lambda\left(\|\mathbf{W}_1\|_F^2 + \|\mathbf{W}_2\|_F^2\right)$$

where

$$\mathbf{h}^{(\ell)} = \sigma(\mathbf{W}_1\tilde{\mathbf{x}}^{(\ell)} + \mathbf{b}_1)$$

$$\hat{\mathbf{x}}^{(\ell)} = \sigma(\mathbf{W}_2\mathbf{h}^{(\ell)} + \mathbf{b}_2)$$

Like deep Autoencoder, we can stack multiple denoising autoencoders layer-wisely to form a Stacked Denoising Autoencoder.

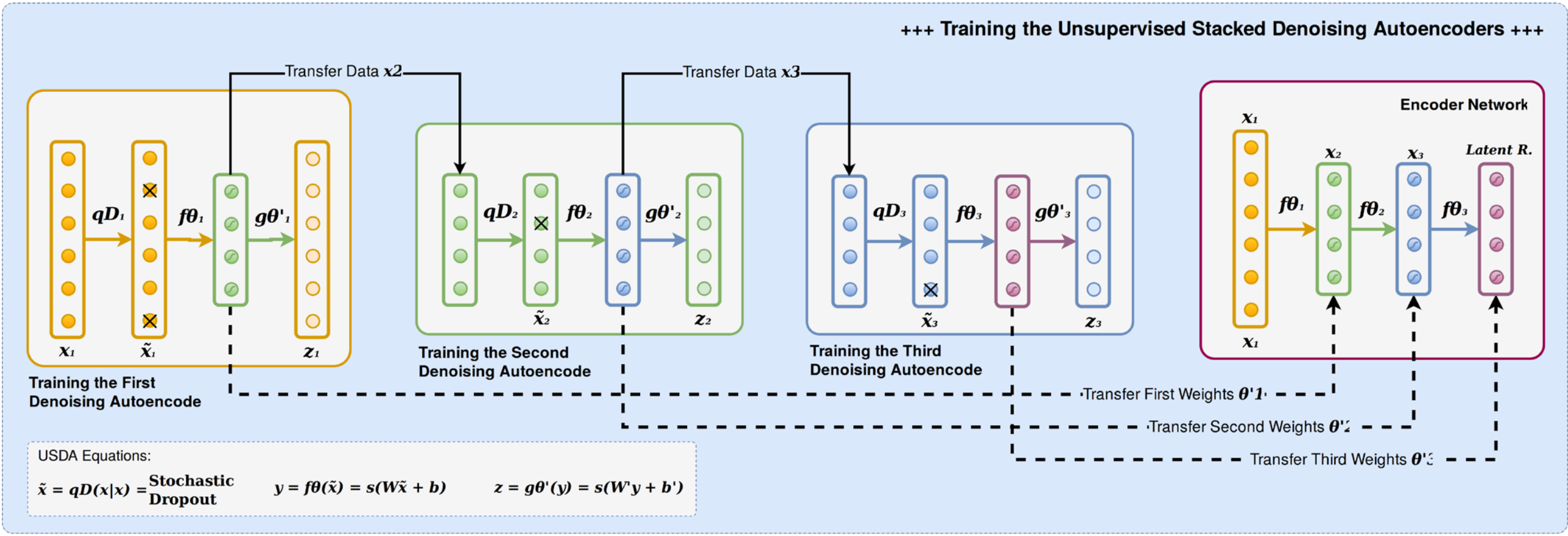


UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Denoising stacked Autoencoder: unsupervised

Result = a new latent representation



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Denoising stacked Autoencoder: example

Model *Deep Patient*, Published in Nature, 2016

Stage 1.

Data-mining stage & Feature extraction:

Driving Electronic Health Records to build a binary phenotype representation.

Stage 2.

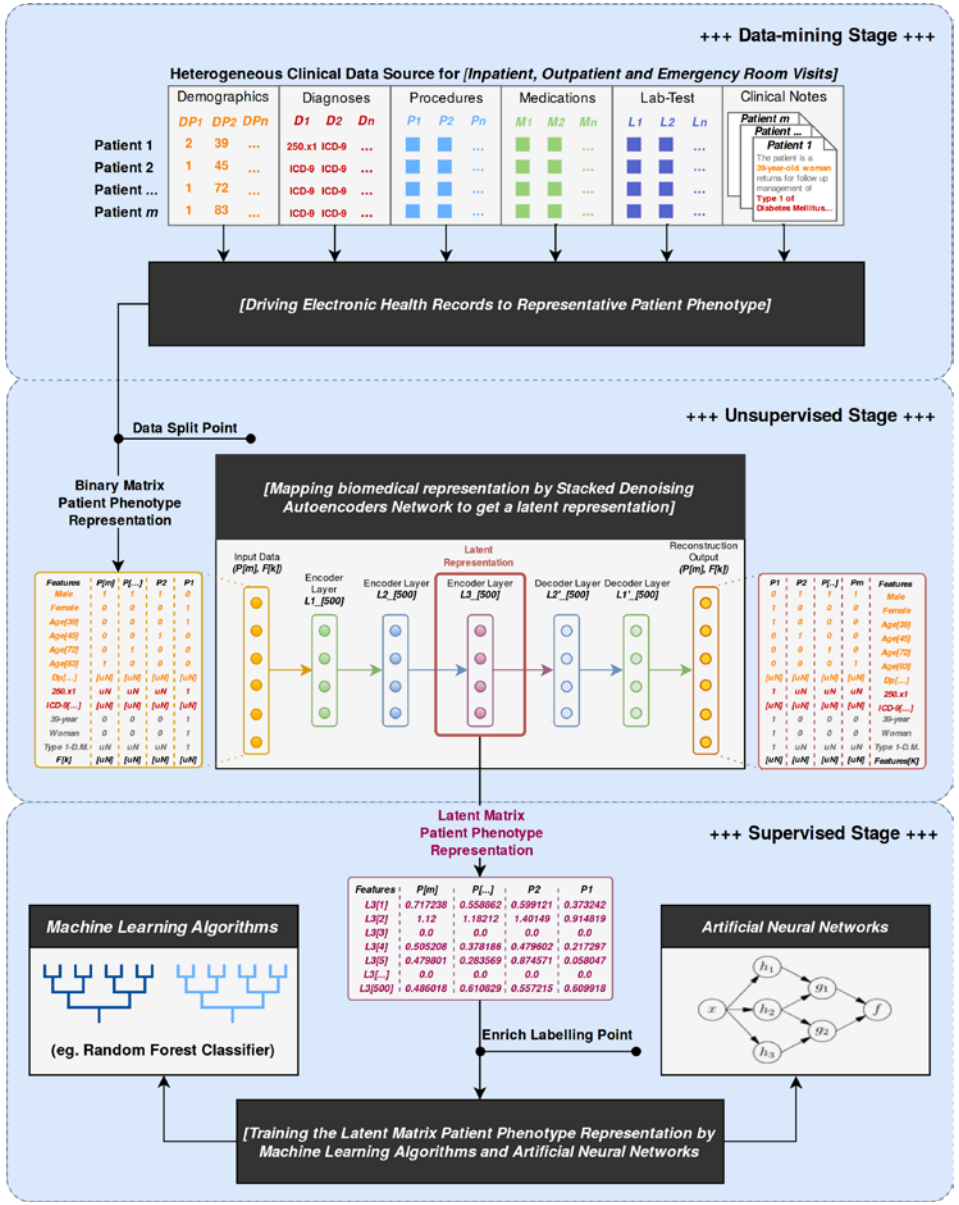
Unsupervised stage:

Mapping the Binary Patient Representation to get a new space call Deep Patient (or Latent Representation) Using Stacked Denoising Autoencoders.

Stage 3.

Supervised stage:

Labeling Medical Target and training the *Latent Representation* by Machine Learning algorithms for classification and prediction of patient's disease.



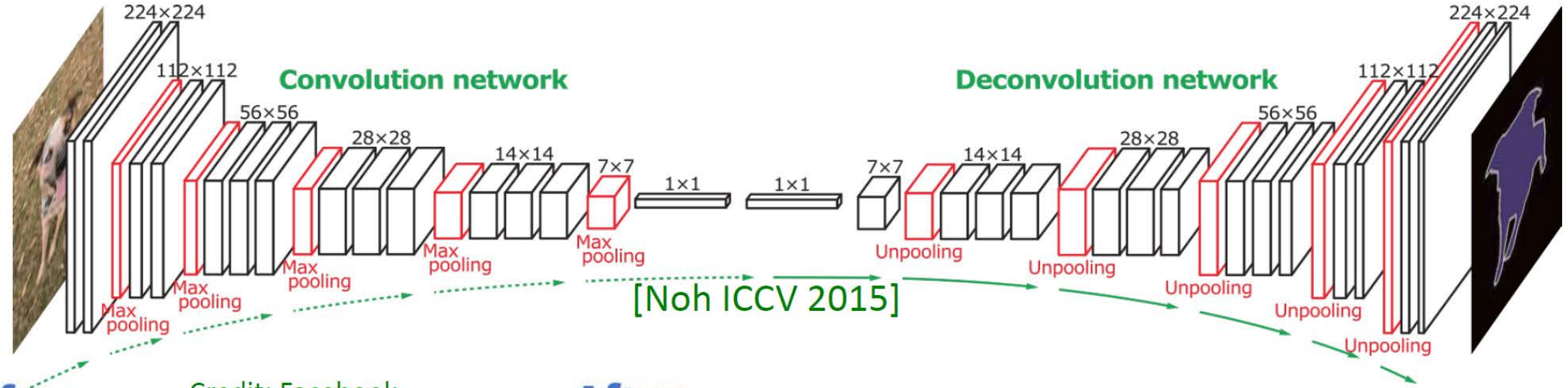
UNIVERSITÉ CÔTE D'AZUR

Combining CNN and Autoencoder

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Image Segmentation



Credit: Facebook

Before

After

MS COCO Detection Challenge!

Credits Matthieu Cord

ADVERSARIAL EXAMPLES

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Morphing

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)

Intriguing properties of neural networks

C. Szegedy, w. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus

arXiv preprint arXiv:1312.6199

2013

[1312.6199] Intriguing properties of neural networks - arXiv.org

https://arxiv.org › cs - Traduire cette page

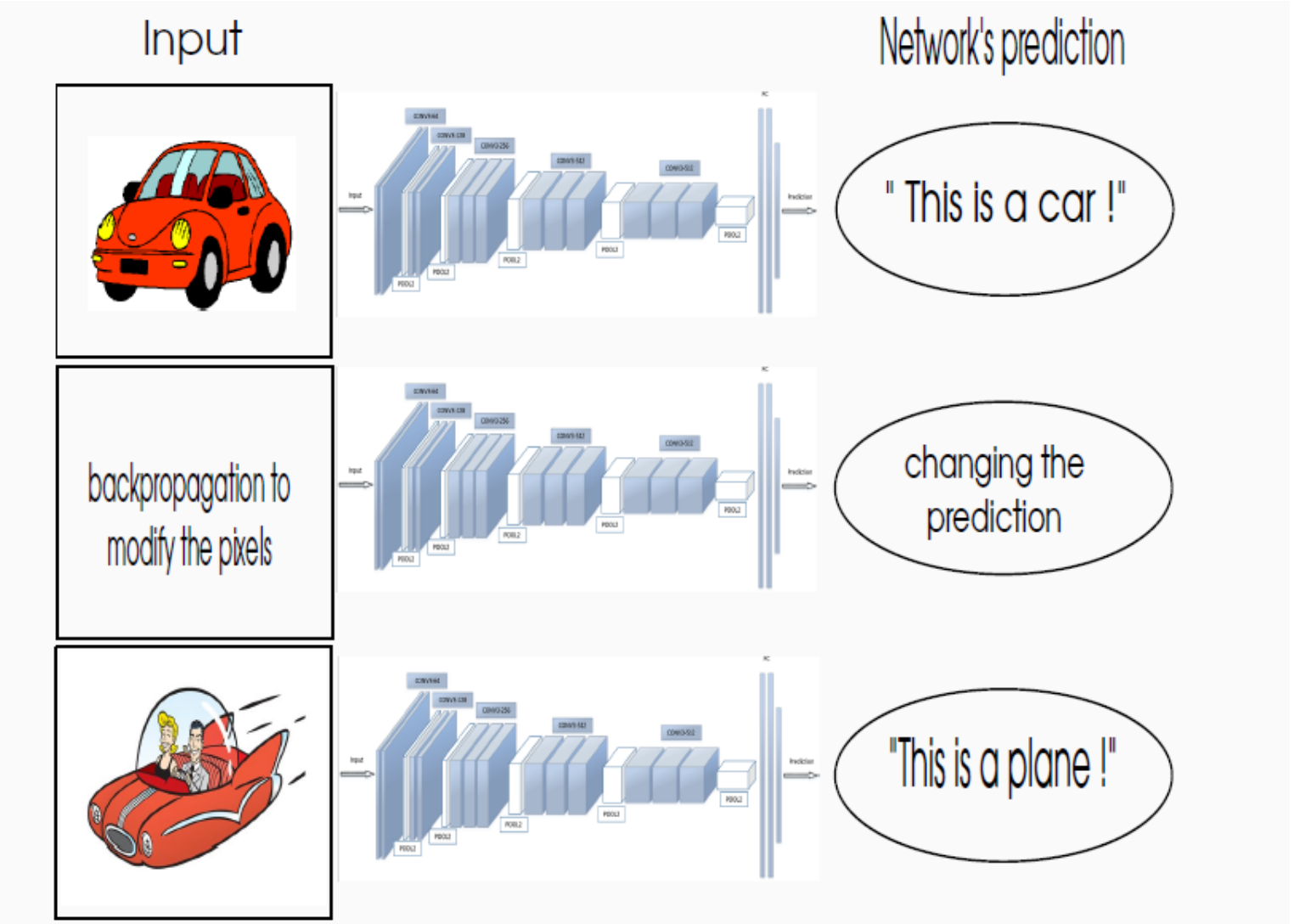
de C Szegedy - 2013 - Cité 449 fois - Autres articles

21 déc. 2013 - In this paper we report two such **properties**. First, we ... Second, we find that deep **neural networks** learn input-output mappings that are fairly ...

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

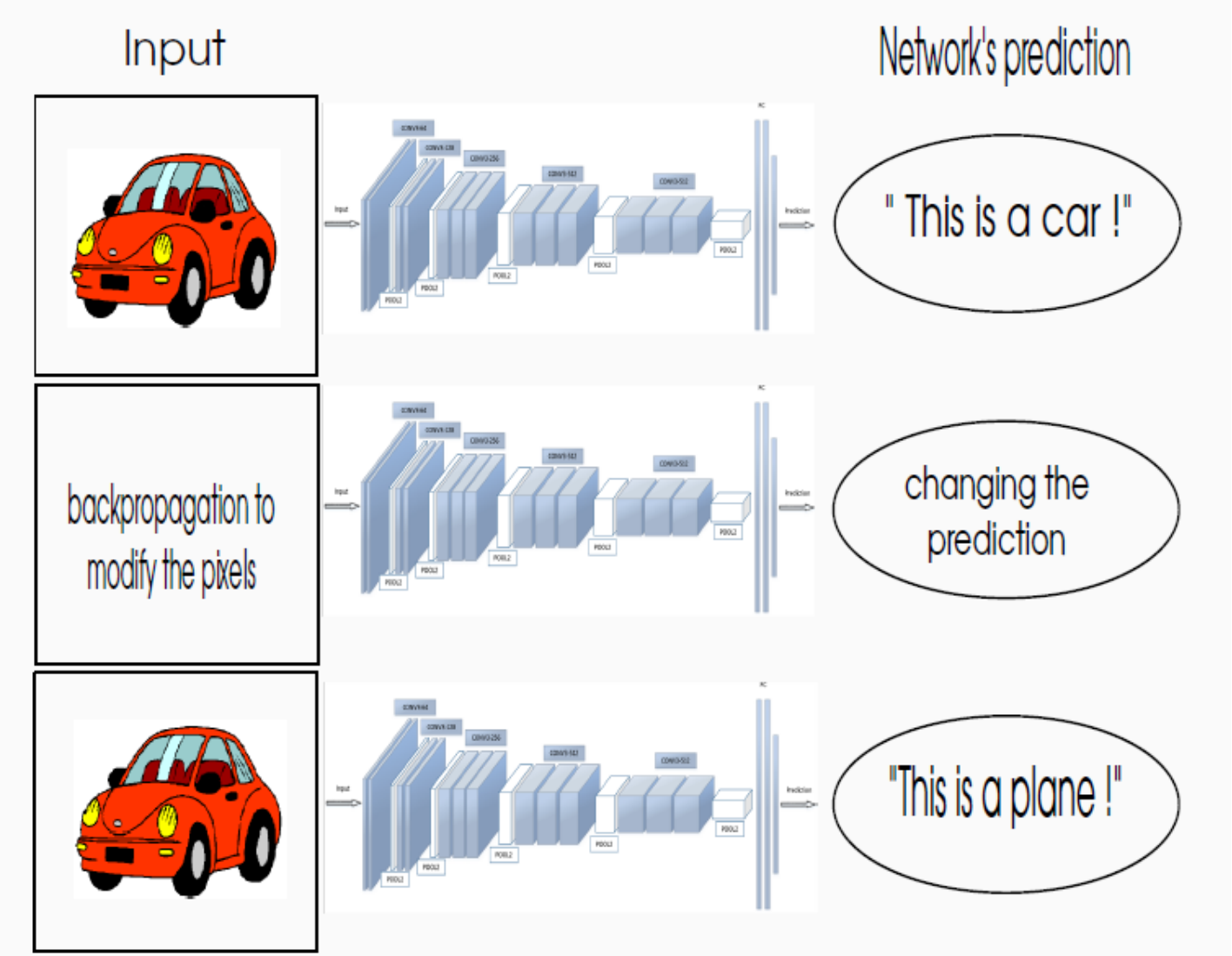
Amazing but…be careful of the adversaries (as any other ML algorithms)



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

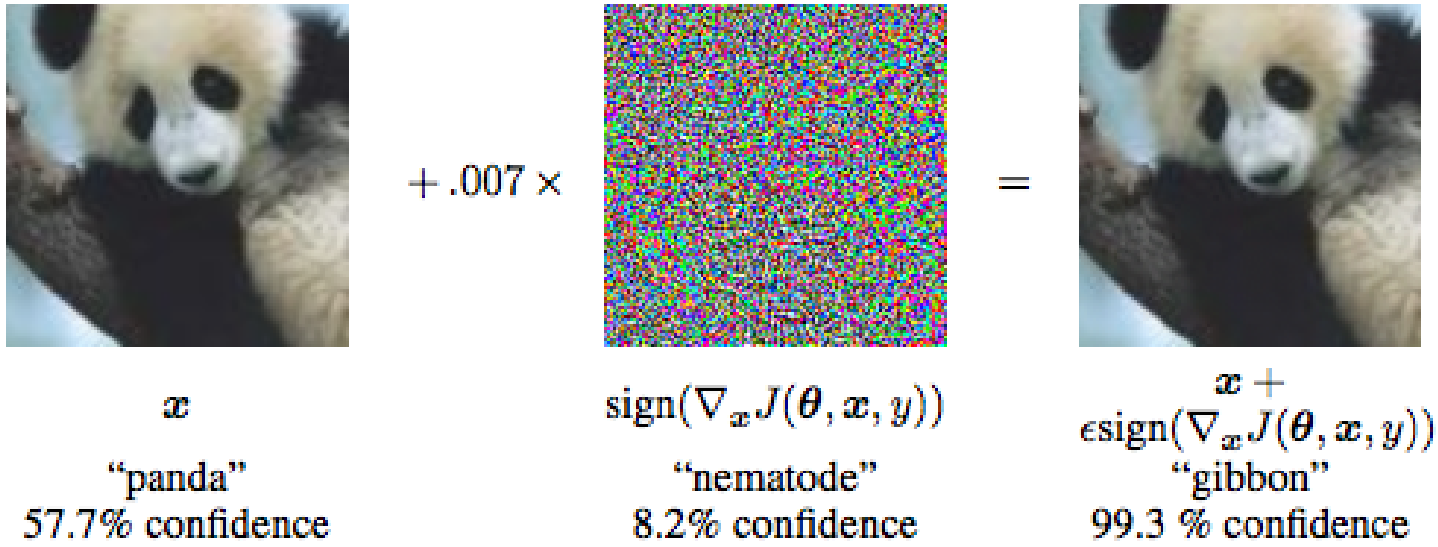
Amazing but…be careful of the adversaries (as any other ML algorithms)



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)

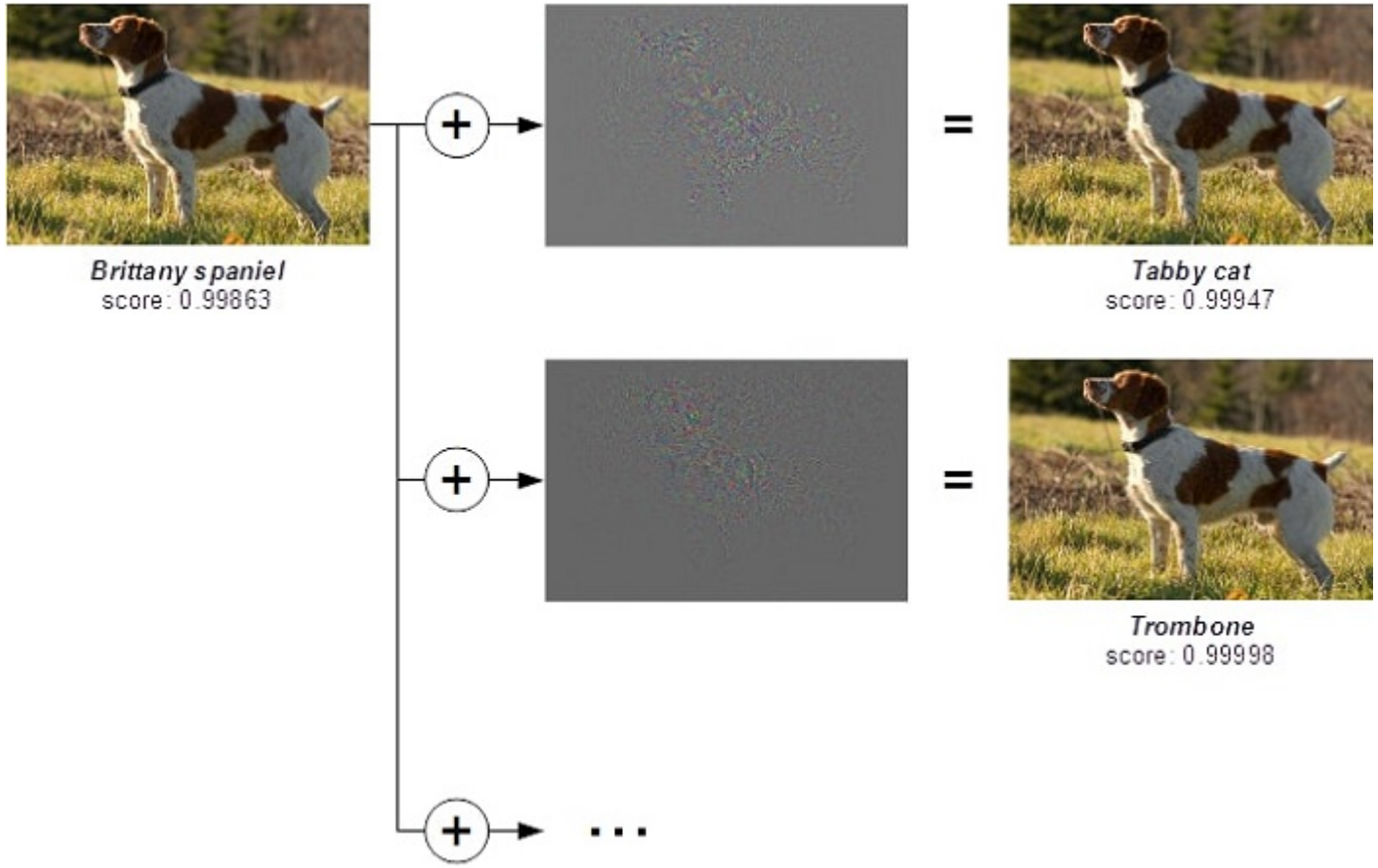


Andrej Karpathy blog, http://karpathy.github.io/2015/03/30/breaking-convnets/

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)

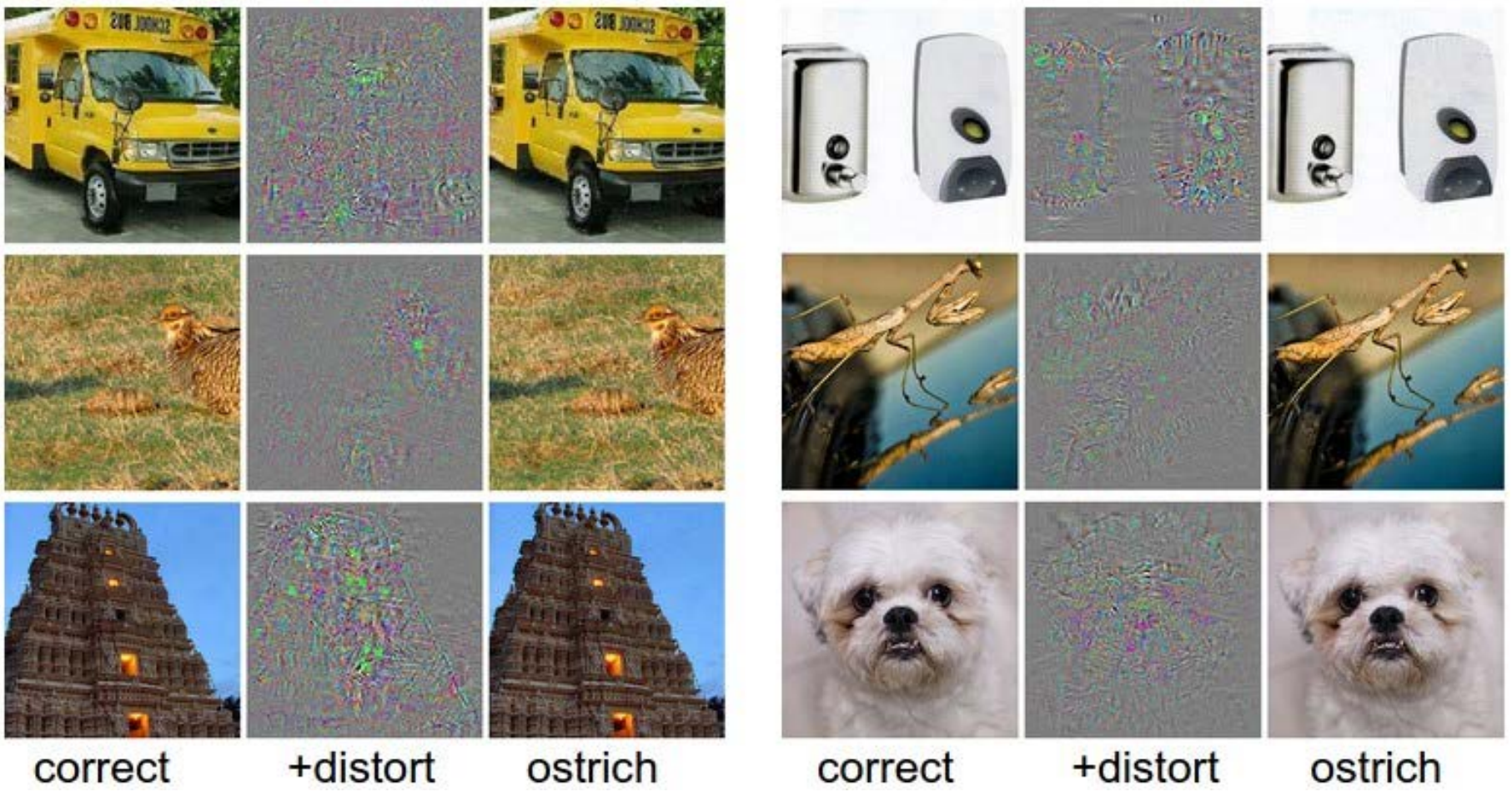


From Thomas Tanay

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)

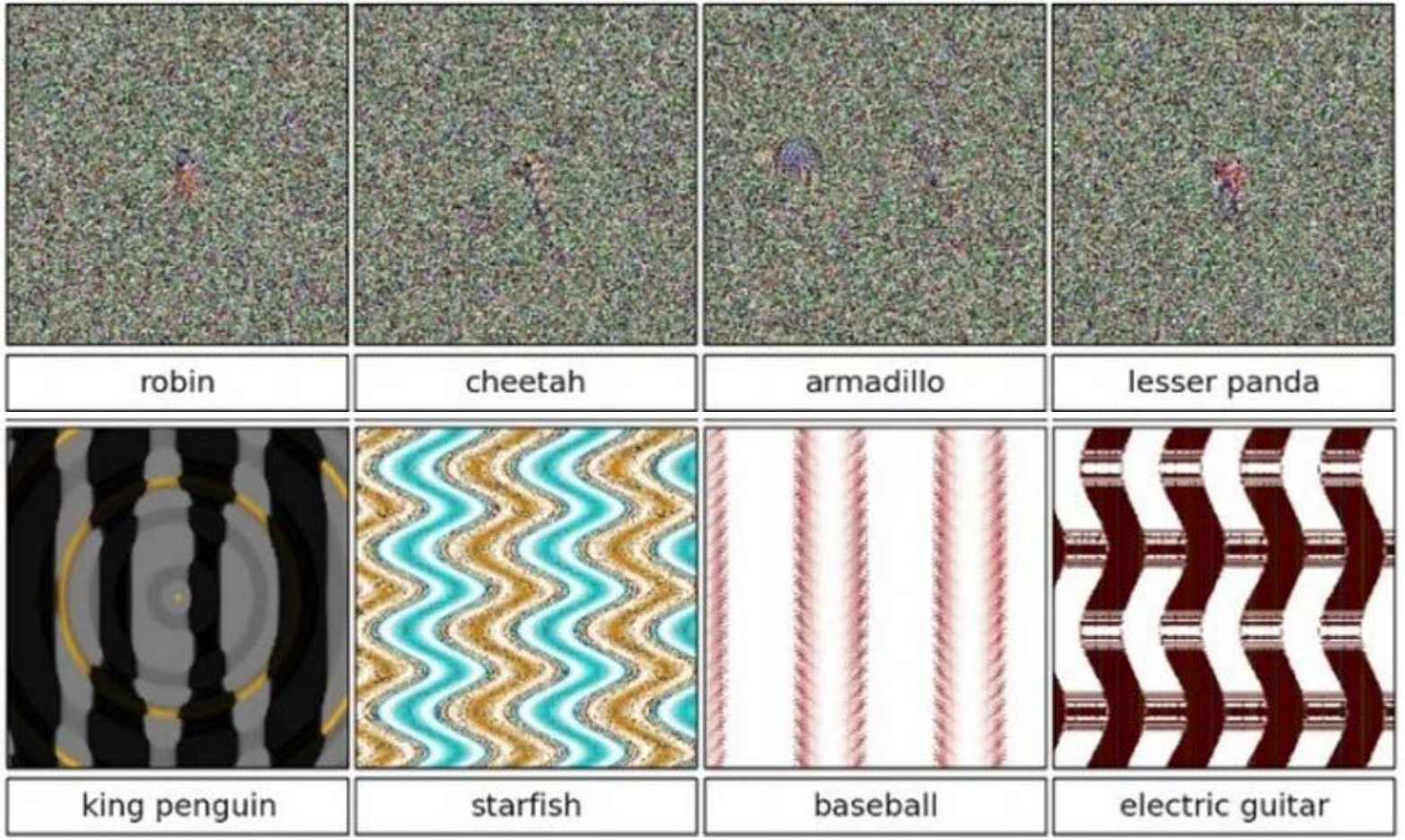


Andrej Karpathy blog, http://karpathy.github.io/2015/03/30/breaking-convnets/

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)



Andrej Karpathy blog, http://karpathy.github.io/2015/03/30/breaking-convnets/

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial example: Definition

Definition: $\hat{x}$ is called adversarial iff:

- given image x
- low distortion $|| x - \hat{x} ||< \epsilon, \quad (\epsilon > 0$, few pixels)
- given network's probabilities $f_\theta(x)$
- Different predictions! $argmax f_\theta(x) \neq argmax f_\theta(\hat{x})$

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial examples: Algorithms

The Fast Gradient Sign Method attack (FGSM)[Goodfellow 2015]: The FGSM is a single step attack, i.e., the perturbation is added in a single-step instead of adding it over a loop (Iterative attack). Essentially, FGSM computes the gradients of a loss function with respect to the input image and then uses the sign of the gradients to create a new image (i.e., the adversarial image) that maximizes the loss. Mathematically, the attacking scheme can be demonstrated as:

$$adv_x = x + \epsilon * sign(\nabla_x J(\theta, x, y))$$

where:

adv_x: Our output adversarial image

x: The original input image

y: The ground-truth label of the input image

ϵ: Small value we multiply the signed gradients by to ensure the perturbations are small enough that the human eye cannot detect them but large enough that they fool the neural network

θ: Our neural network model

J: The loss function

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial examples: Algorithms

The Basic Iterative Method attack (BIM)[Kurakin 2017]: The BIM applying the same step as FGSM multiple times with small step size and clip the pixel values of intermediate results after each step to ensure that they are in an ϵ-neighborhood of the original image, i.e., within the range $[X_{ij} - \epsilon, X_{ij} + \epsilon]$, X_{ij} being the pixel value of the previous image. This method is also referred to as Iterative-FGSM or IFGSM. Mathematically, the attacking scheme can be demonstrated as:

$$X_0^{adv} = X, \quad X_{N+1}^{adv} = Clip_{X,\epsilon}\{X_N^{adv} + \alpha sign(\nabla_X J(X_N^{adv}, y_{true}))\}$$

where:

X: Clean input image

X_i^{adv}: Adversarial Image at i^{th} step

J: Loss function

y_{true}: Model output for x

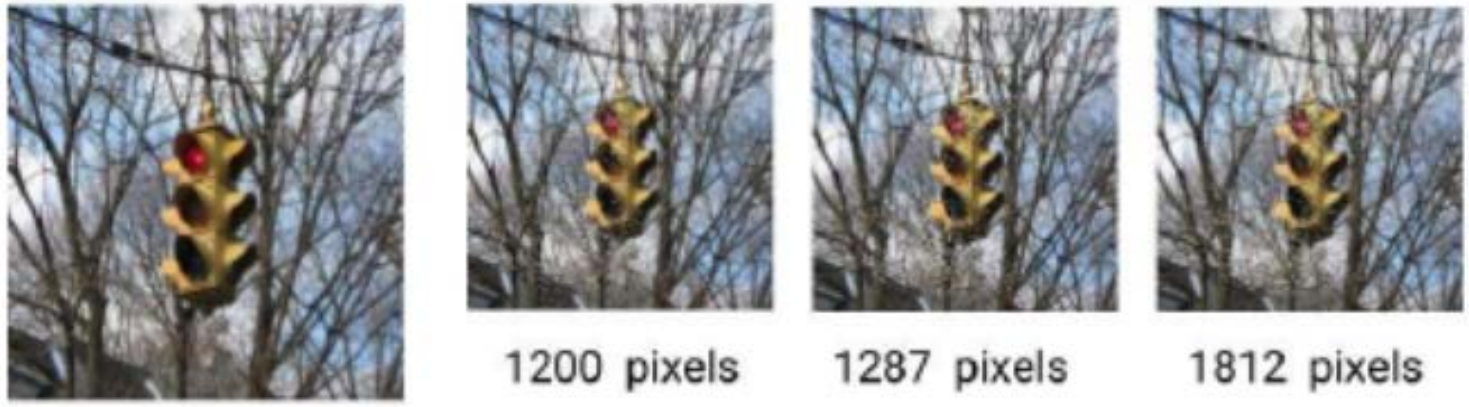
ϵ: Tunable parameter

α: Step size

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)



Original
Traffic light
(ImageNet class 920)

VGG16

VGG19
Misclassified

RESNET

State-of-the art deep neural networks on ImageNet

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)

Red Light Modified to Green after 18 white pixels. Probability: 59%

Red Light Modified to Green after 9 green pixels. Probability: 50.9%

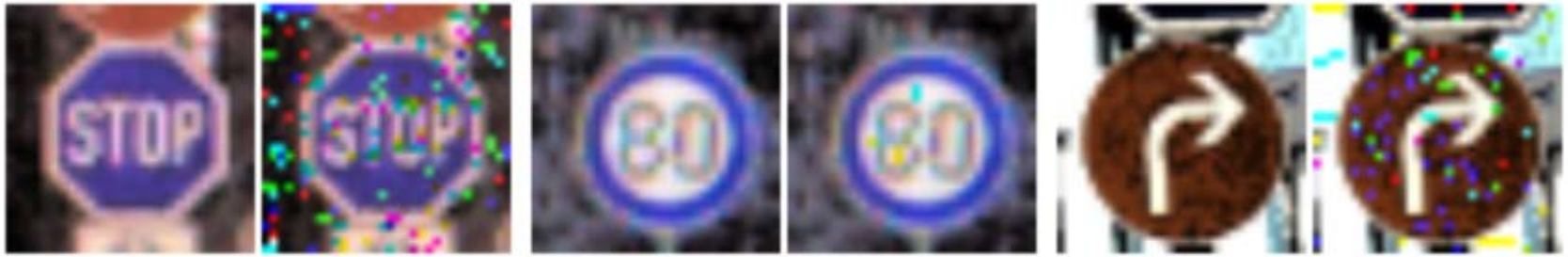
Red Light Modified to Green after 9 green pixels. Probability: 53%

No Light Modified to Green after 4 green pixels. Probability: 51.9%

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)



stop	30m speed limit	80m speed limit	30m speed limit	go right	go straight
Confidence	0.999964		0.99		

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

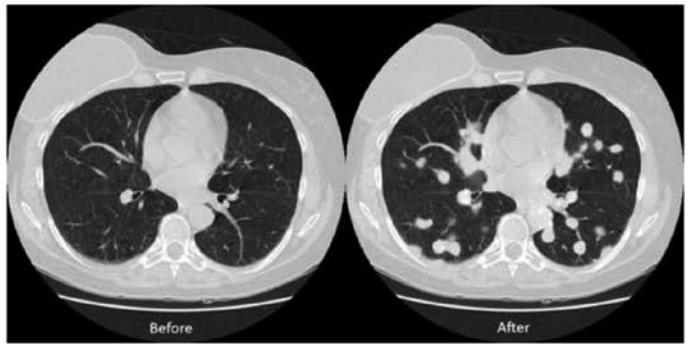
Amazing but…be careful of the adversaries (as any other ML algorithms)

SUBSCRIBE | TOPIC INDEX | Current Issue | Digital Editions | Article Archive | eNewsletter | Product Directories | Events | Jobs

home | subscribe | comment | resources | reprints | writers' guidelines

News

HACKERS CAN FOOL RADIOLOGISTS AND AI SOFTWARE BY MANIPULATING LUNG CANCER SCANS



Hackers can access a patient's 3D medical scans to add or remove malignant lung cancer and deceive both radiologists and AI algorithms that are used to aid diagnosis, according to a new study published by Ben-Gurion University (BGU) of the Negev cybersecurity researchers. Click here for a video of the attack.

A 3D CT scan combines a series of X-ray images taken from different angles around the body and uses computer processing to create cross-sectional slices of the bones, blood vessels, and soft tissues. CT images provide more detailed information than standard X-rays and are used to diagnose cancer, heart disease, infectious diseases, and more. An MRI scan is similar, but uses powerful magnetic fields instead of ionizing radiation to diagnose bone, joint, ligament, and cartilage conditions.

Malicious attackers can tamper with the scans to deliberately cause a misdiagnosis for insurance fraud, ransomware, cyberterrorism, or even murder. Attackers can even automate the entire process in a malware that can infect a hospital's network.

"Our research shows how an attacker can realistically add or remove medical conditions from CT and MRI scans," says Yisroel Mirsky, PhD, lead researcher in the BGU department of software and information systems engineering and project manager and cybersecurity researcher at BGU's

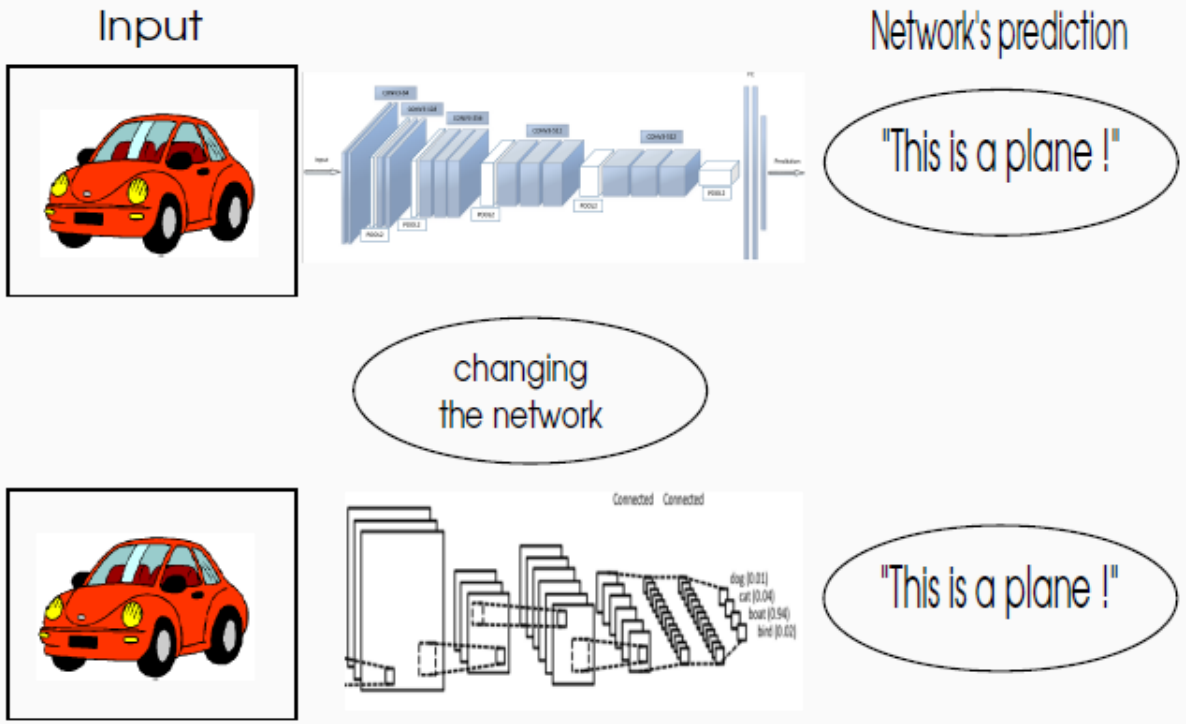
Tweets by @RadiologyToday

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)

- $\neq$ outliers
- regularization: correct one… find another
- high confidence predictions
- **Transferability**



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Amazing but…be careful of the adversaries (as any other ML algorithms)

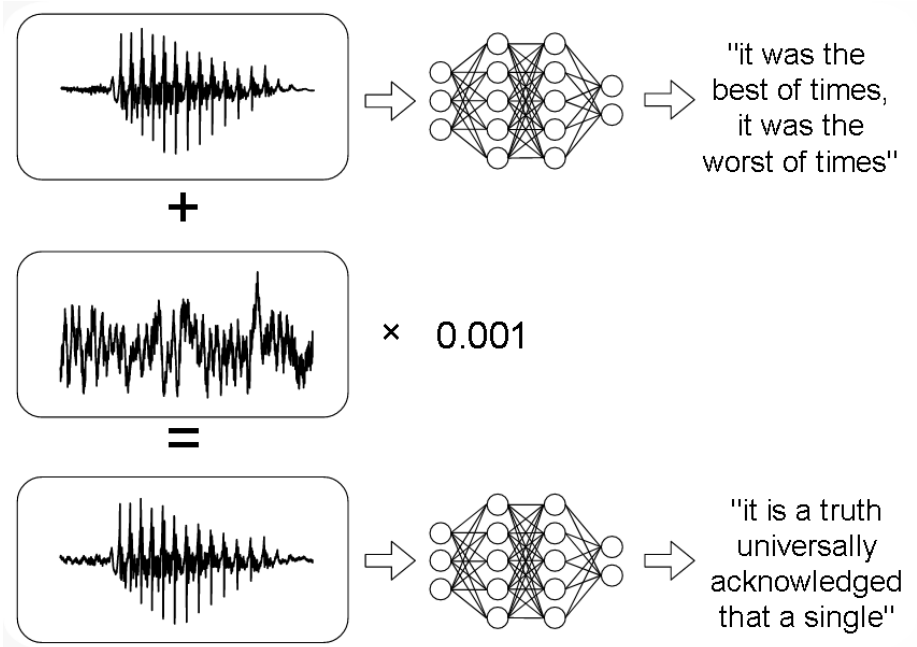


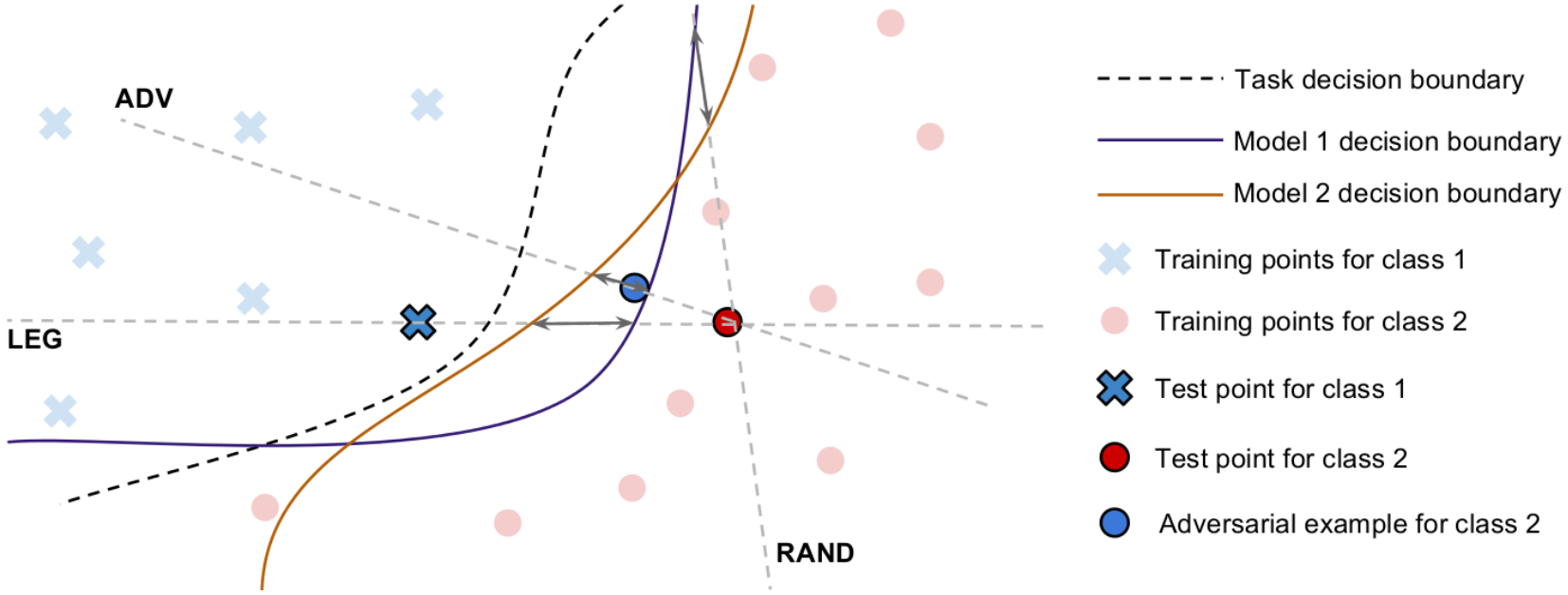
Figure from our paper: given any waveform, we can modify it slightly to produce another (similar) waveform that transcribes as any different target phrase.

https://nicholas.carlini.com/code/audio_adversarial_examples/

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial examples…



Tramèr, F., Papernot, N., Goodfellow, I., Boneh, D., & McDaniel, P. (2017).
The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*.

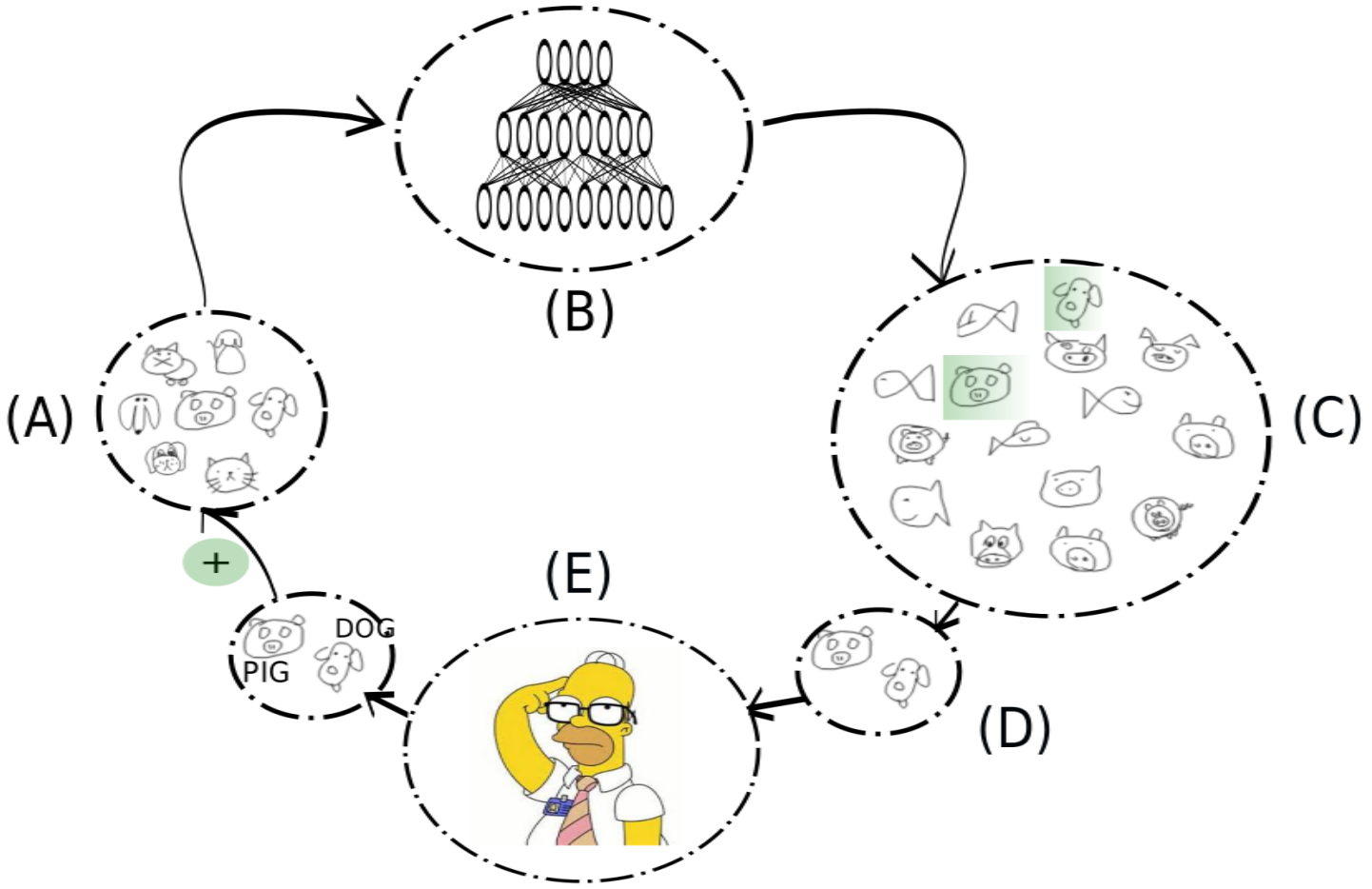
UNIVERSITÉ CÔTE D'AZUR

Can we benefit from adversarial examples and all their properties ?

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

ACTIVE Supervised Classification



(A)=labelled data

(B)=model

(C)=**POOL** of unlabelled data

(D)=**BATCH** of query

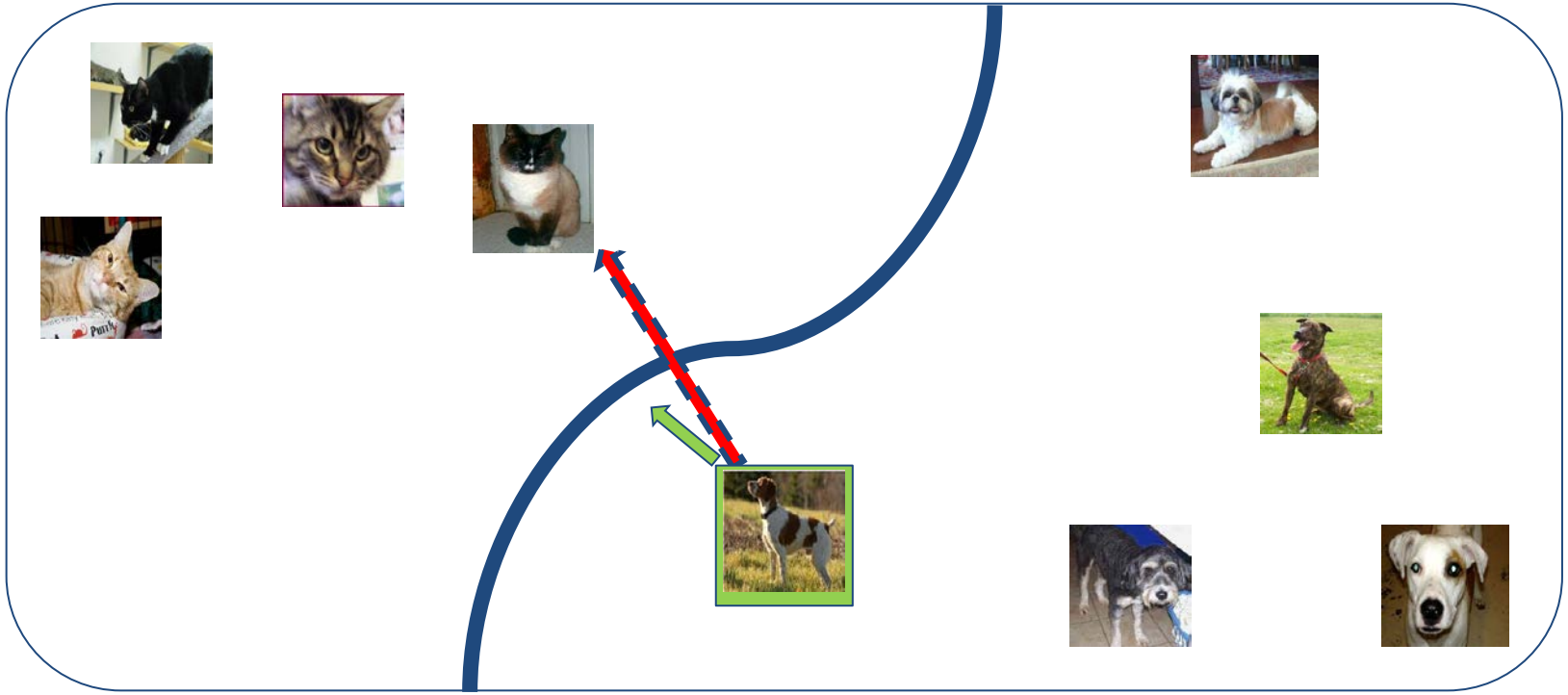
(E) ORACLE (eg human)

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

MARGIN BASED ACTIVE LEARNING?

- query => close to the decision boundary
- topology of the decision boundary unknown
- how can we approximate such a distance for neural networks ?
- *"Dimension + Volume + Labels"*

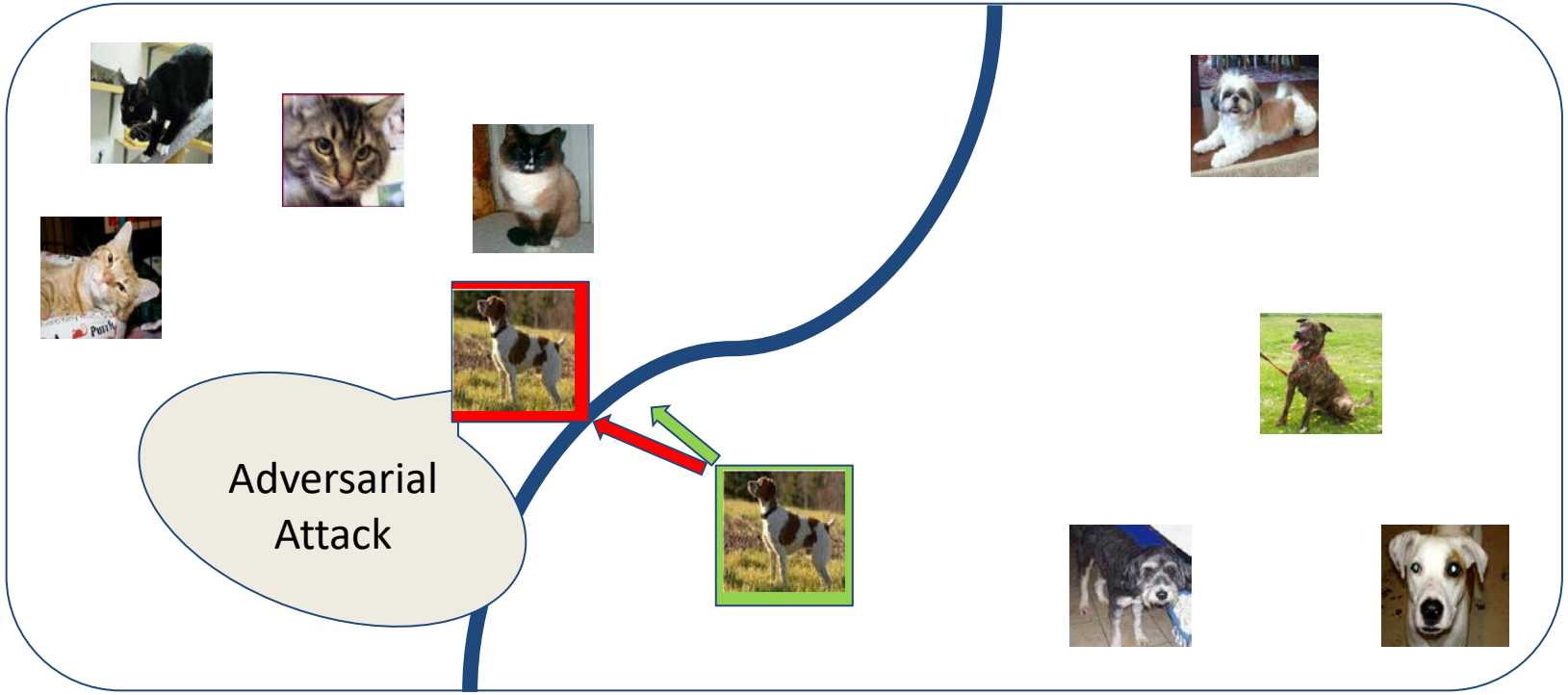


UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial attacks for MARGIN BASED ACTIVE LEARNING

- query ➡ small adversarial perturbation



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

DeepFool Active Learning (DFAL)

✓	query the top-k examples owing the smallest DeepFool adversarial perturbation  = '*dog*'
✓	query the adversarial attacks with the same label pseudo-labeling comes for free without corrupting the training set  ='*dog*' → ='*dog*' 
✓	Transferability of adversarial examples ↓ Transferability of queries

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

DFAL EXPERIMENTS 1/3

- Query the top-k samples owing the smallest adversarial perturbation **(DFAL_0)**
- **BATCH**= 10
- **|MNIST| = 60 000 |QuickDraw|=444 971**

	Accuracy (%)				
# annotations	100	500	800	1000	All
DFAL_0	**85.08**	95.89	**97.79**	**98.13**	–
BALD	53.73	91.47	94.32	94.32	–
CEAL	50.87	90.69	90.69	90.69	–
CORE-SET	78.80	**96.68**	97.46	97.88	–
EGL	37.92	91.84	93.99	93.99	–
uncertainty	45.57	88.36	94.27	94.60	–
RANDOM	69.79	91.96	94.05	94.46	**98.98**

% of Test accuracy
MNIST (VGG8)

	Accuracy (%)				
# annotations	100	500	800	1000	All
DFAL_0	78.62	**91.35**	**92.44**	**93.14**	–
BALD	**82.00**	89.94	91.92	92.87	–
CEAL	64.45	79.66	85.73	88.65	–
CORE-SET	66.71	89.93	92.28	92.62	–
EGL	63.12	86.80	90.06	90.06	–
uncertainty	52.77	88.05	89.31	91.03	–
RANDOM	78.28	88.13	89.71	89.94	**96.75**

% of Test accuracy
QuickDraw (VGG8)

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

DFAL EXPERIMENTS 2/3

- Pseudo labeling the adversarial samples of the queries
- **BATCH**= 10
- **|MNIST| = 60 000 |QuickDraw|=444 971**

	Accuracy (%)				
# annotations	100	500	800	1000	All
DFAL	84.28	**96.90**	**97.98**	**98.59**	–
DFAL_0	**85.08**	95.89	97.79	98.13	–
BALD	53.73	91.47	94.32	94.32	–
CEAL	50.87	90.69	90.69	90.69	–
CORE-SET	78.80	96.68	97.46	97.88	–
EGL	37.92	91.84	93.99	93.99	–
uncertainty	45.57	88.36	94.27	94.60	–
RANDOM	69.79	91.96	94.05	94.46	**98.98**

% of Test accuracy
MNIST (VGG8)

	Accuracy (%)				
# annotations	100	500	800	1000	All
DFAL	**84.23**	**91.52**	**93.16**	**93.91**	–
DFAL_0	78.62	91.35	92.44	93.14	–
BALD	82.00	89.94	91.92	92.87	–
CEAL	64.45	79.66	85.73	88.65	–
CORE-SET	66.71	89.93	92.28	92.62	–
EGL	63.12	86.80	90.06	90.06	–
uncertainty	52.77	88.05	89.31	91.03	–
RANDOM	78.28	88.13	89.71	89.94	**96.75**

% of Test accuracy
QuickDraw (VGG8)

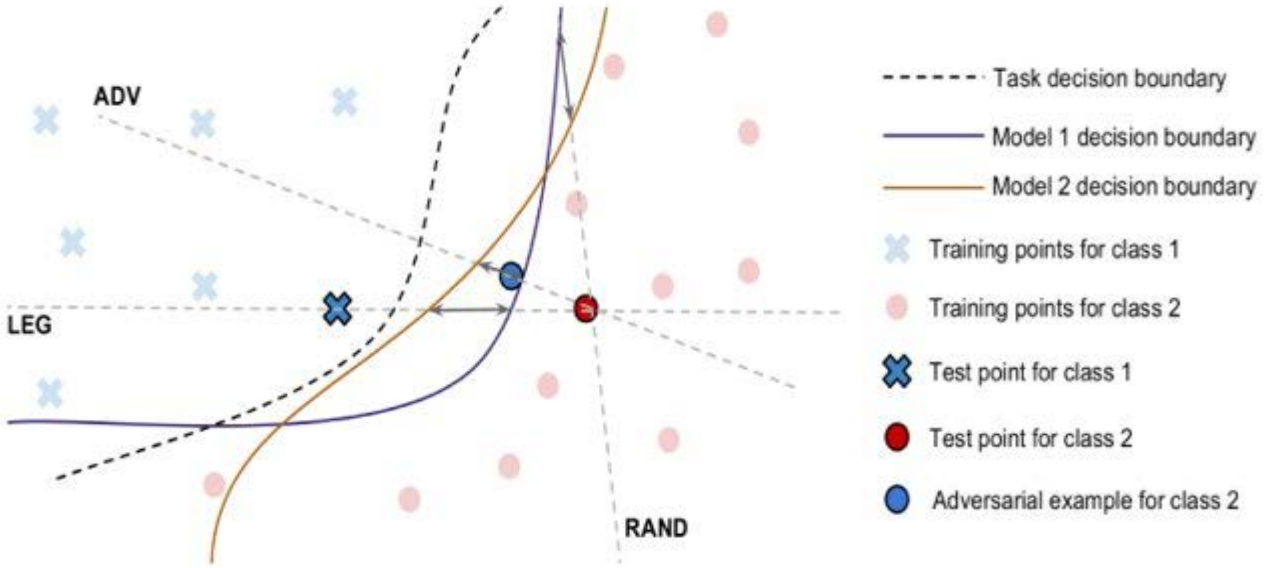
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

TRANSFERABILITY: The ultimate threat of adversarial examples [Tramèr, 2017]

Adversarial space: contiguous, at least 2 dimensional. Dimension is proportional to the ratio increase in loss / perturbation

Different models with similar class boundary distances



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

DFAL EXPERIMENTS 3/3

- Transferability of non targeted adversarial attacks
- 1000 queries
- Model Selection
- **|MNIST| = 60 000 |ShoeBag|=184 792**

	DFAL	CORE-SET	RANDOM
LeNet5→ VGG8	**97.80**	96.90	94.46
VGG8→ LeNet5	**97.93**	97.40	95.31

% of Test accuracy

MNIST

	DFAL	CORE-SET	RANDOM
LeNet5→ VGG8	**99.40**	99.12	97.08
VGG8→ LeNet5	**98.75**	98.50	98.07

% of Test accuracy

Shoe Bag

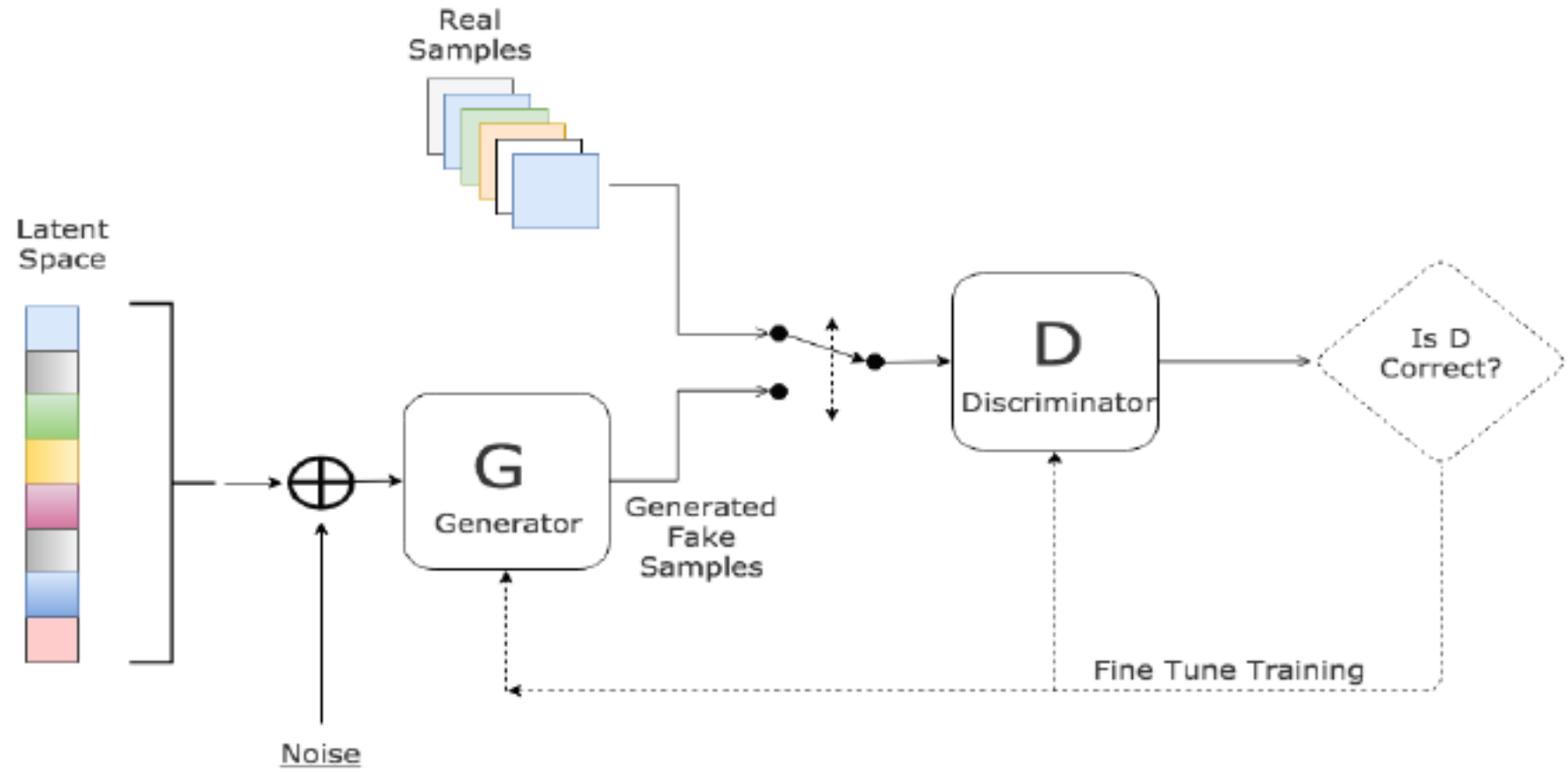
UNIVERSITÉ CÔTE D'AZUR

GENERATIVE ADVERSARIAL NETWORKS

UNIVERSITÉ CÔTE D'AZUR

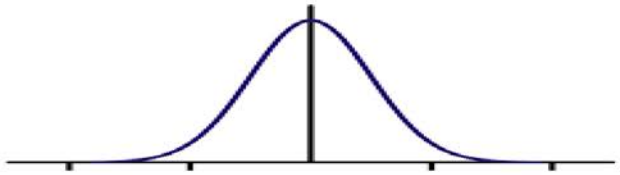
univ-cotedazur.fr

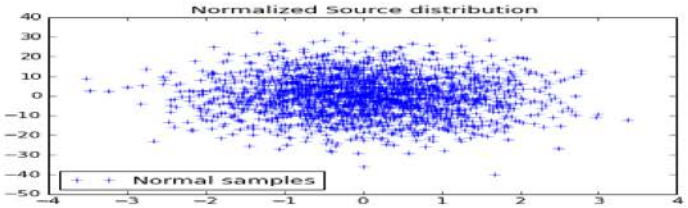
How to solve Adversarial Examples? Generative Adversarial Networks

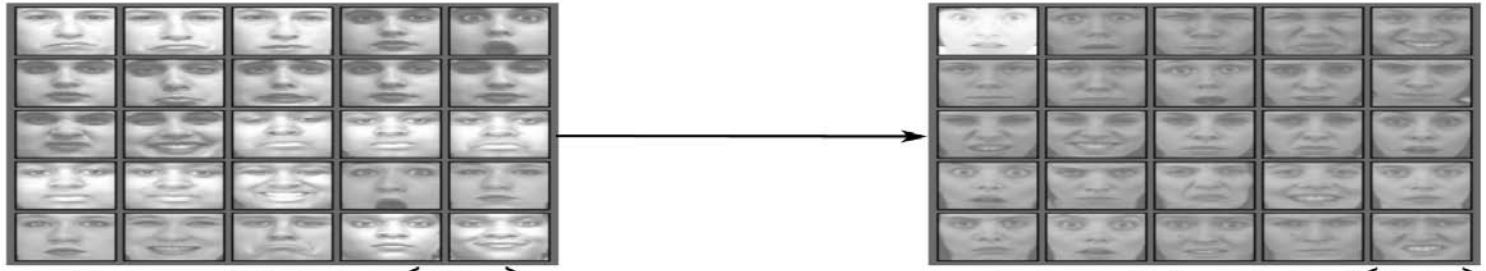


Generative Learning

- **Generative Models:** Learning how to sample from an approximate distribution close to the groundtruth







- Equation vs sampling
- RBM, VAE, Pixel CNN, **GANs** → sharpest samples

Generative Adversarial Networks

Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014

- Problem: Want to sample from complex, high-dimensional training distribution. No direct way to do this!

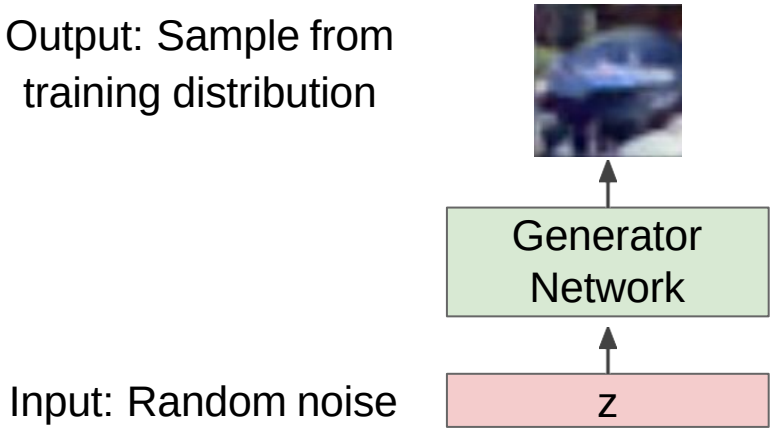
- Solution: Sample from a simple distribution, e.g. random noise. Learn transformation to training distribution.

- Q: What can we use to represent this complex transformation?

Generative Adversarial Networks

Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014

- Problem: Want to sample from complex, high-dimensional training distribution. No direct way to do this!
- Solution: Sample from a simple distribution, e.g. random noise. Learn transformation to training distribution.



- Q: What can we use to represent this complex transformation?
- A: A neural network!

UNIVERSITÉ CÔTE D'AZUR

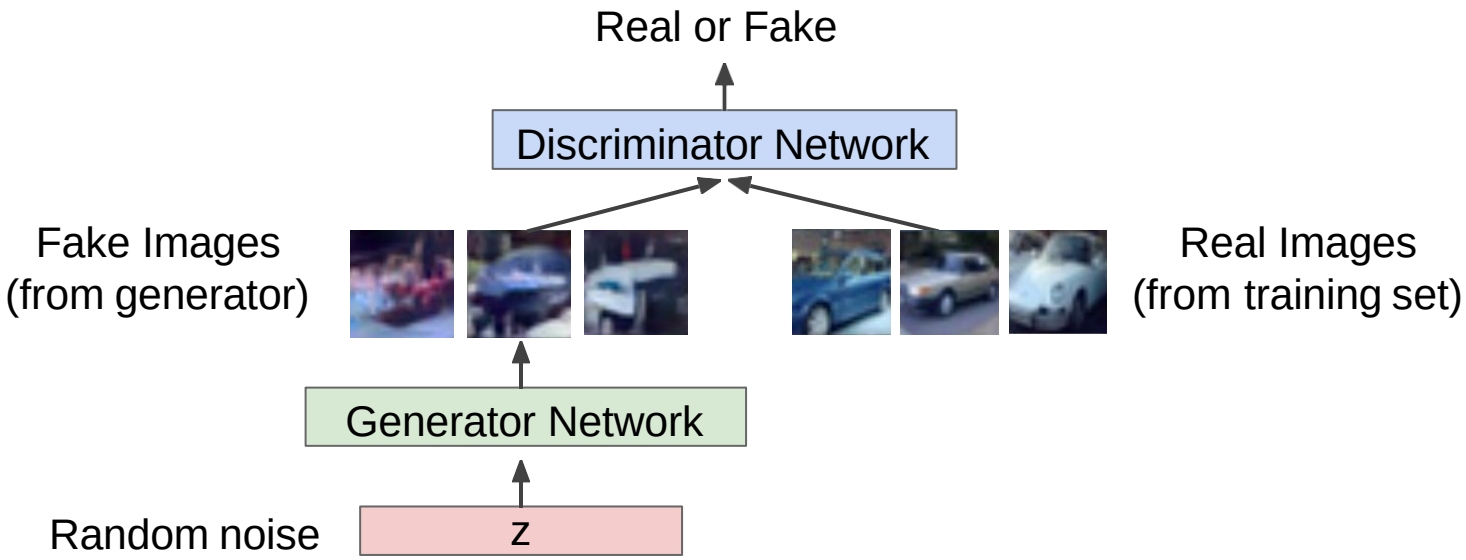
univ-cotedazur.fr

Training GANs: Two-player game

- **Generator network**: try to fool the discriminator by generating real-looking images
- **Discriminator network**: try to distinguish between real and fake images

Training GANs: Two-player game

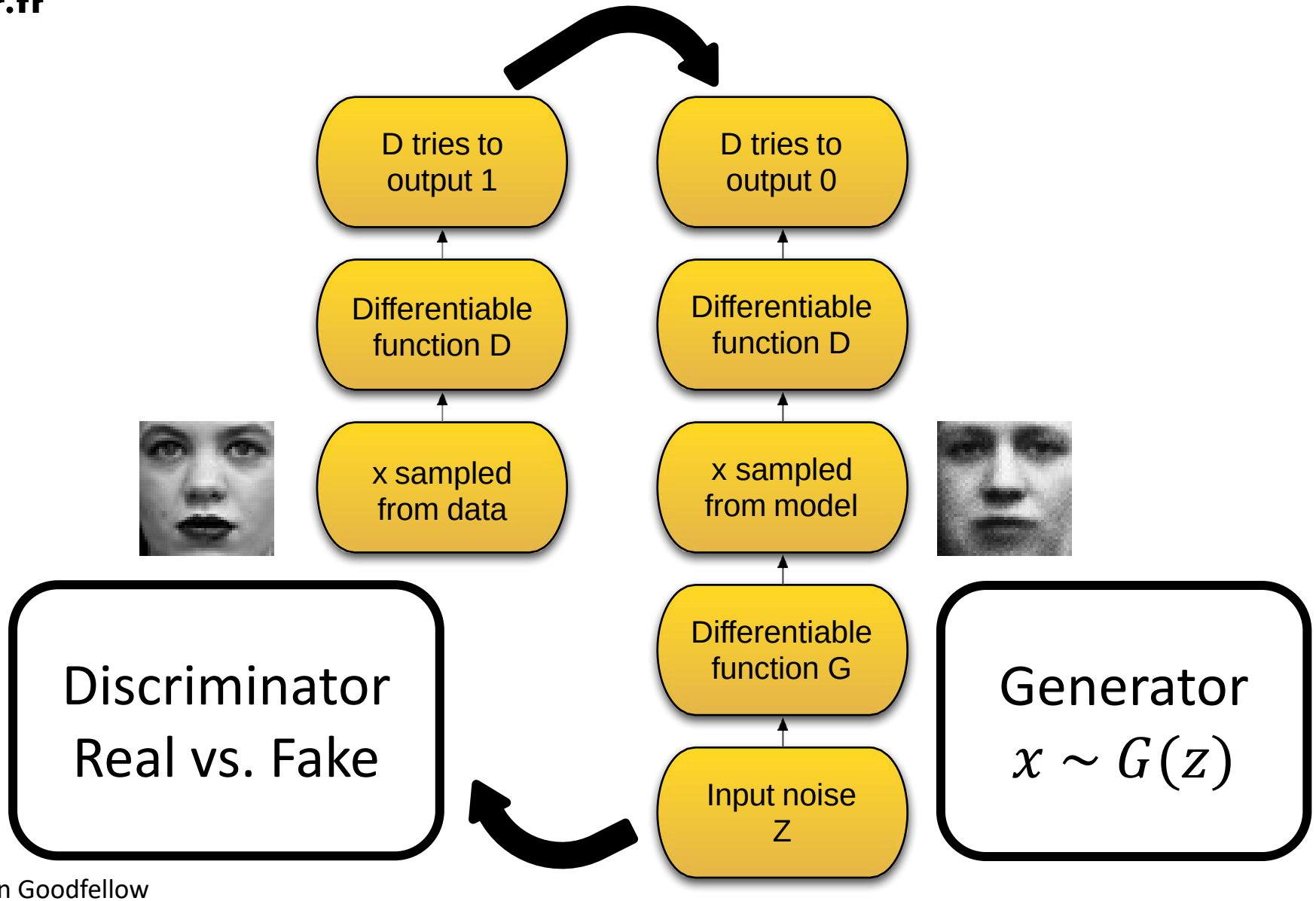
- **Generator network**: try to fool the discriminator by generating real-looking images
- **Discriminator network**: try to distinguish between real and fake images



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial Networks Framework



Ian Goodfellow

Training GANs: Two-player game

Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014

- **Generator network**: try to fool the discriminator by generating real-looking images
- **Discriminator network**: try to distinguish between real and fake images

- Train jointly in **minimax game**

- Minimax objective function:

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Training GANs: Two-player game

- **Generator network**: try to fool the discriminator by generating real-looking images
- **Discriminator network**: try to distinguish between real and fake images

- Train jointly in **minimax game**

- Minimax objective function:

Discriminator outputs likelihood in (0,1) of real image

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log \underbrace{D_{\theta_d}(x)}_{\text{Discriminator output for real data x}} + \mathbb{E}_{z \sim p(z)} \log(1 - \underbrace{D_{\theta_d}(G_{\theta_g}(z))}_{\text{Discriminator output for generated fake data G(z)}}) \right]$$

- Discriminator (θ_d) wants to **maximize objective** such that D(x) is close to 1 (real) and D(G(z)) is close to 0 (fake)
- Generator (θ_g) wants to **minimize objective** such that D(G(z)) is close to 1 (discriminator is fooled into thinking generated G(z) is real)

Training GANs: Two-player game

Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014

- Minimax objective function:

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

- Alternate between:
  1. **Gradient ascent** on discriminator

$$\max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

  2. **Gradient descent** on generator

$$\min_{\theta_g} \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))$$

Training GANs: Two-player game

Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014

Minimax objective function:

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

Alternate between:

1. **Gradient ascent** on discriminator

$$\max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

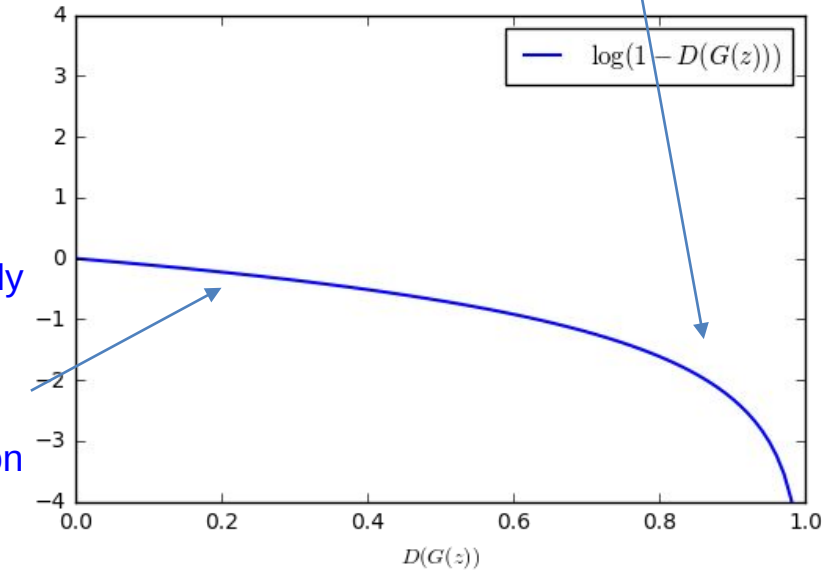
2. **Gradient descent** on generator

$$\min_{\theta_g} \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z)))$$

In practice, optimizing this generator objective does not work well!

Gradient signal dominated by region where sample is already good

When sample is likely fake, want to learn from it to improve generator. But gradient in this region is relatively flat!



Adapted from Serena Young

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Training GANs: Two-player game

Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014

Minimax objective function:

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

Alternate between:

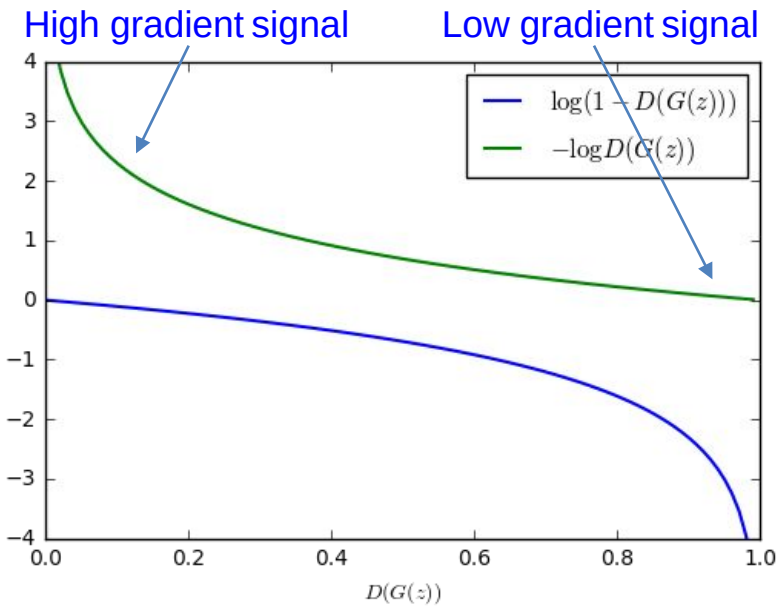
1. **Gradient ascent** on discriminator

$$\max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

2. **Instead: Gradient ascent** on generator, different objective

$$\max_{\theta_g} \mathbb{E}_{z \sim p(z)} \log(D_{\theta_d}(G_{\theta_g}(z)))$$

Instead of minimizing likelihood of discriminator being correct, now maximize likelihood of discriminator being wrong.
Same objective of fooling discriminator, but now higher gradient signal for bad samples => works much better! Standard in practice.



Adapted from Serena Young

Training GANs: Two-player game

Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014

Putting it together: GAN training algorithm

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:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D_{\theta_d}(x^{(i)}) + \log(1 - D_{\theta_d}(G_{\theta_g}(z^{(i)}))) \right]$$

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 ascending its stochastic gradient (improved objective):

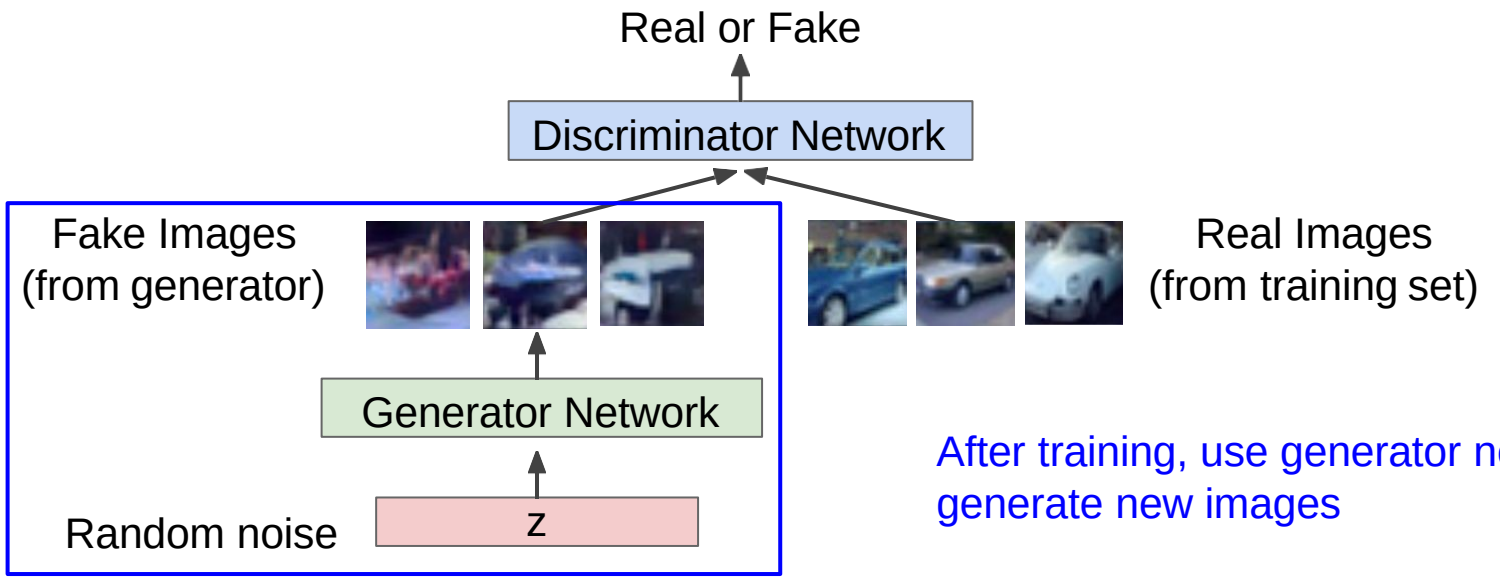
$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log(D_{\theta_d}(G_{\theta_g}(z^{(i)})))$$

end for

Training GANs: Two-player game

Ian Goodfellow et al., "Generative Adversarial Nets", NIPS 2014

Generator network: try to fool the discriminator by generating real-looking images
Discriminator network: try to distinguish between real and fake images



After training, use generator network to generate new images

GAN convergence

- If the discriminator is optimal: $\mathbf{D} \equiv \mathbf{D}^*$ (*NNs are universal approximators*)

$$\mathbf{D}^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_{model}(x)} \tag{1}$$

$$\mathcal{V}(\mathbf{D}^*, \mathbf{G}) = 2 * JSD(p_{data} \mid\mid p_{model}) - 2ln(2) \tag{2}$$

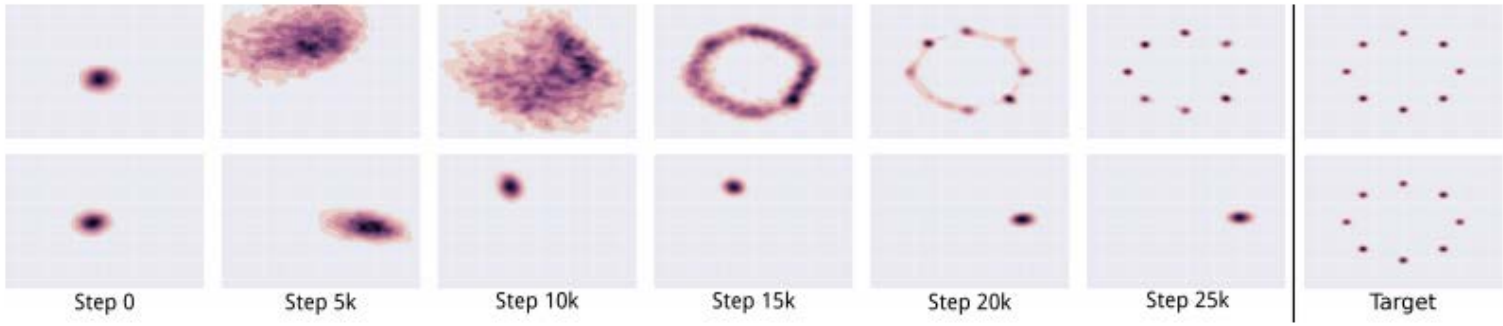
- with certain conditions, it converges asymptotically to the groundtruth distribution, which may not be the case for other maximum likelihood based generative models
- G has converged to p_{data} and D outputs $\frac{1}{2}$

The struggle is real

- GAN models often end up in local Nash equilibria that are associated with mode collapse or otherwise fail to model the target distribution
- cost function are non convex
- parameters are continuous
- high dimensional parameter space
- no way to measure the quality of the generation process (not even log likelihood)

Mode Collapse

- Multiple peaks, generator with poor diversity
- Gradient vanishing

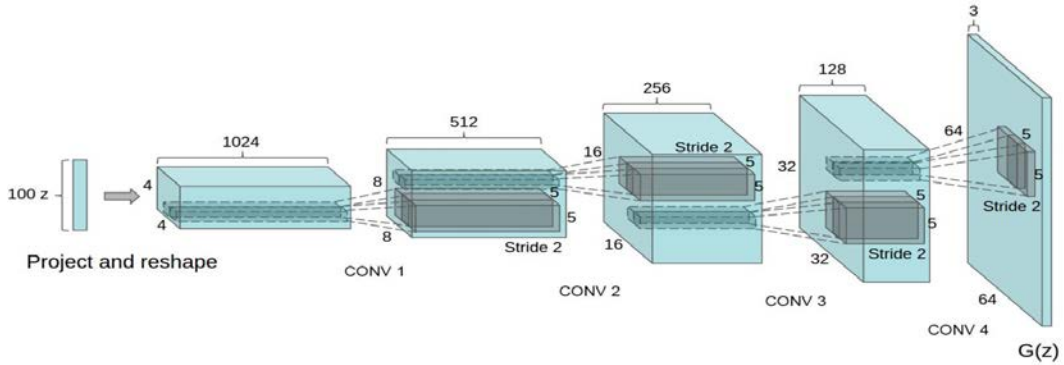


With mode collapse, the model only learns one or a few of the modes of a multimodal dataset. The data usually used to train GANs on typical problems have a very large number of modes, which makes mode collapse problematic.

- Multiple GANS: AdaGANs

Generative Adversarial Nets: Convolutional Architectures

- Generator is an upsampling network with fractionally-strided convolutions
- Discriminator is a convolutional network



Architecture guidelines for stable Deep Convolutional GANs

- Replace any pooling layers with strided convolutions (discriminator) and fractional-strided convolutions (generator).
- Use batchnorm in both the generator and the discriminator.
- Remove fully connected hidden layers for deeper architectures.
- Use ReLU activation in generator for all layers except for the output, which uses Tanh.
- Use LeakyReLU activation in the discriminator for all layers.

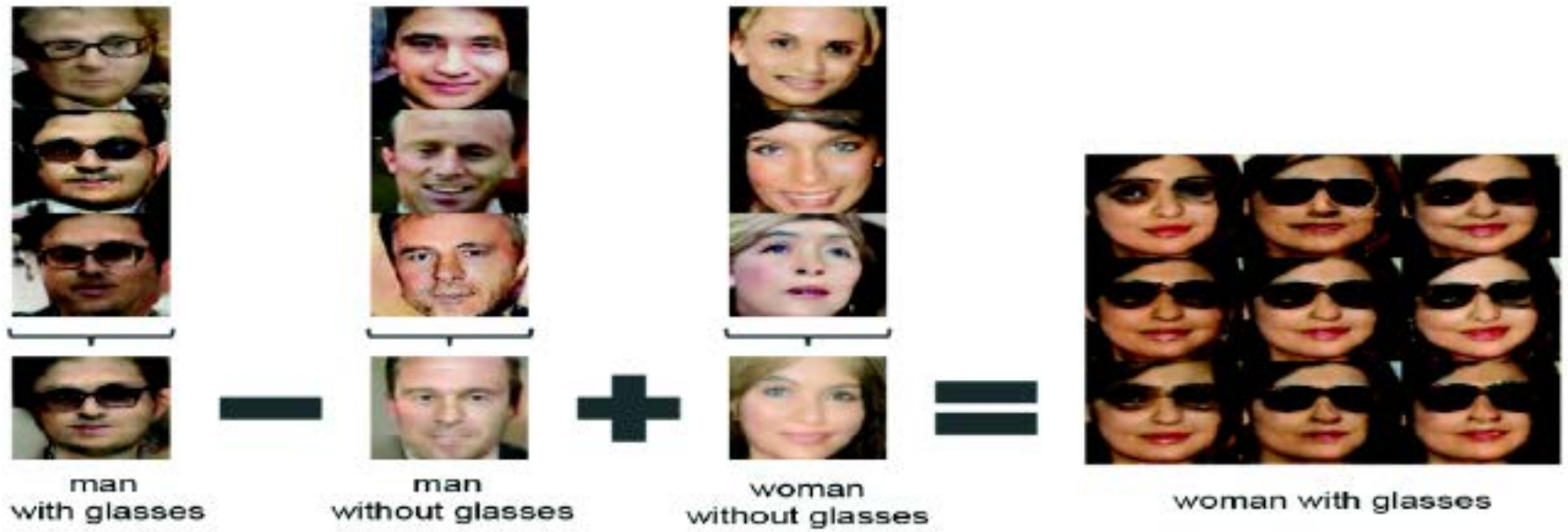
Radford et al, 'Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks", ICLR 2016

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

It finally did not solve adversarial, but…

Results DCGAN



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

It finally did not solve adversarial, but…

(image from Karras et al. 2019)

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Tips and tricks

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Tips & Tricks

- Normalize your training data into range [-1, 1]
- Pick a bounded output activation function for G (sigmoid, tanh...)
- Do not sabotage your Discriminator: usually deeper than G and with an unbounded output activation
- Label smoothing for real training data
- Feature matching (facial for instance)
- Learning a conditional model often gives better samples (add a generated class)
- Do not balance D and G losses

Tips and tricks

- Use batchnorm, ReLU
- Regularize norm of gradients
- Use one of the new loss functions
- Add noise to inputs or labels
- Append image similarity to avoid mode collapse
- Use labels, extra info when available (CGAN)
- …

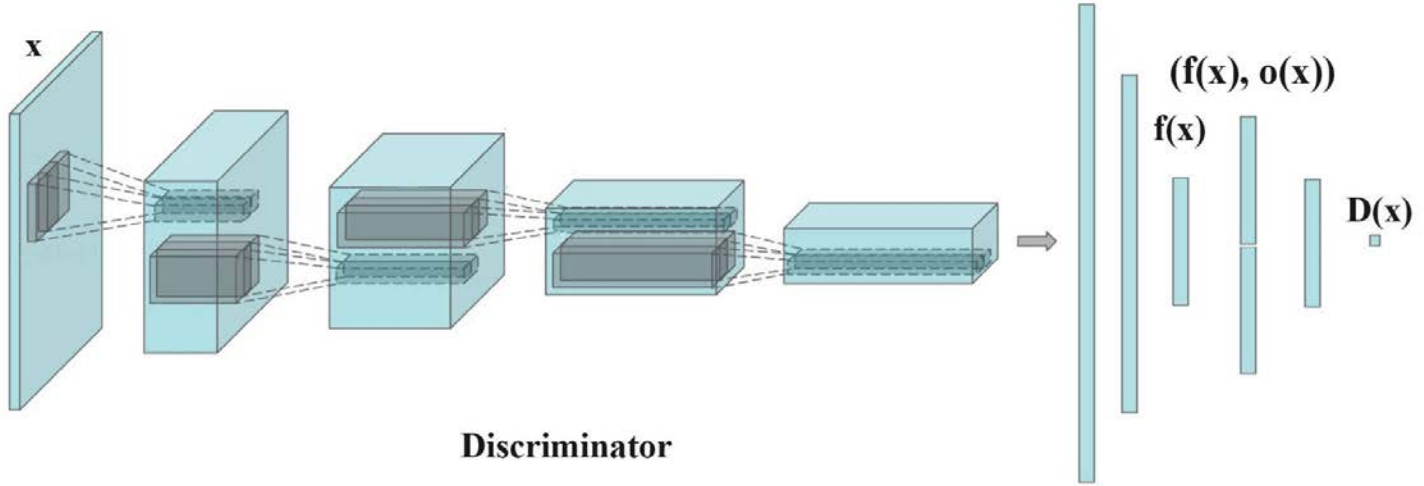
https://github.com/soumith/talks/blob/master/2017-ICCV_Venice/How_To_Train_a_GAN.pdf
https://towardsdatascience.com/gan-ways-to-improve-gan-performance-acf37f9f59b

Tips & Tricks

- Add BN in G (not the last layer, maybe not a good idea for D)
- minibatch of data X: μ=mean(X), σ=std(X)
- Add two extra parameters of the network γ, β
- $BN(x \in X)_{\gamma,\beta} = \gamma \frac{x-\mu}{\sqrt{\sigma+\epsilon}} + \beta$

Tips & Tricks

- Preventing mode collapsing: discriminant minibatch



When mode collapses, all images created looks similar.
⇒ feed real images and generated images into the discriminator separately in different batches and compute the similarity of the image *x* with images in the same batch.
If the mode starts to collapse, the similarity of generated images increases. The discriminator can use this score to detect generated images and penalize the generator if mode is collapsing.
The similarity $o(x_i)$ between the image x_i and other images in the same batch is computed and we append the similarity $o(x)$ in one of the dense layers in the discriminator to classify whether this image is real or generated.

GAN training is challenging

- Vanishing gradient – when discriminator is very good
- Mode collapse – too little diversity in the samples generated
- Lack of convergence because hard to reach Nash equilibrium
- Loss metric doesn't always correspond to image quality; Frechet Inception Distance (FID) is a decent choice

Wasserstein GAN

- New objective function: Earth Mover Distance

$$\mathbb{W}(p_{data} \mid p_{model}) = \inf_{\gamma \in \Pi(p_{data}, p_{model})} \mathbb{E}_{(x,y)\sim\gamma}[|| x - y ||] \quad (4)$$

- Optimal Transport in the dual space: training on minibatch
- Discriminator: 1 Lipshitz function, weight clipping
- NNs are already Lipshitz function =)

$$\mathbb{W}(p_{data} \mid p_{model}) = \max_{\mathbf{D},\, ||\mathbf{D}||_L=1} \mathbb{E}_{x\sim p_{data}}\Big[\mathbf{D}(x)\Big] - \mathbb{E}_{z\sim p_z}\Big[\mathbf{D}(\mathbf{G}(z))\Big] \quad (5)$$

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Alternative loss functions

Name	*Paper Link*	*Value Function*
GAN	Arxiv	$L_D^{GAN} = E[\log(D(x))] + E[\log(1 - D(G(z)))]$ $L_G^{GAN} = E[\log(D(G(z)))]$
LSGAN	Arxiv	$L_D^{LSGAN} = E[(D(x) - 1)^2] + E[D(G(z))^2]$ $L_G^{LSGAN} = E[(D(G(z)) - 1)^2]$
WGAN	Arxiv	$L_D^{WGAN} = E[D(x)] - E[D(G(z))]$ $L_G^{WGAN} = E[D(G(z))]$ $W_D \leftarrow clip\_by\_value(W_D, -0.01, 0.01)$
WGAN_GP	Arxiv	$L_D^{WGAN\_GP} = L_D^{WGAN} + \lambda E[(\lvert\nabla D(\alpha x - (1 - \alpha G(z)))\rvert - 1)^2]$ $L_G^{WGAN\_GP} = L_G^{WGAN}$
DRAGAN	Arxiv	$L_D^{DRAGAN} = L_D^{GAN} + \lambda E[\left(\lvert\nabla D(\alpha x - (1 - \alpha x_p))\rvert - 1\right)^2]$ $L_G^{DRAGAN} = L_G^{GAN}$
CGAN	Arxiv	$L_D^{CGAN} = E[\log(D(x, c))] + E[\log(1 - D(G(z), c))]$ $L_G^{CGAN} = E[\log(D(G(z), c))]$
infoGAN	Arxiv	$L_{D,Q}^{infoGAN} = L_D^{GAN} - \lambda L_I(c, c')$ $L_G^{infoGAN} = L_G^{GAN} - \lambda L_I(c, c')$
ACGAN	Arxiv	$L_{D,Q}^{ACGAN} = L_D^{GAN} + E[P(class = c\lvert x)] + E[P(class = c\lvert G(z))]$ $L_G^{ACGAN} = L_G^{GAN} + E[P(class = c\lvert G(z))]$
EBGAN	Arxiv	$L_D^{EBGAN} = D_{AE}(x) + \max(0, m - D_{AE}(G(z)))$ $L_G^{EBGAN} = D_{AE}(G(z)) + \lambda \cdot PT$
BEGAN	Arxiv	$L_D^{BEGAN} = D_{AE}(x) - k_t D_{AE}(G(z))$ $L_G^{BEGAN} = D_{AE}(G(z))$ $k_{t+1} = k_t + \lambda(\gamma D_{AE}(x) - D_{AE}(G(z)))$

https://github.com/hwalsuklee/tensorflow-generative-model-collections

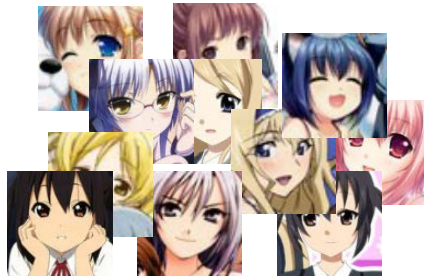
https://medium.com/@jonathan_hui/gan-wasserstein-gan-wgan-gp-6a1a2aa1b490

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Three Categories of GANs

1. Typical GAN



$$\begin{bmatrix} -0.3 \\ 0.1 \\ \vdots \\ 0.9 \end{bmatrix}$$

random vector → Generator →

image

2. Conditional GAN

blue eyes,
red hair,
short hair

paired data

"Girl with red hair"

text → Generator →

image

3. Unsupervised Conditional GAN

domain x

domain y

unpaired data

x

Photo → Generator →

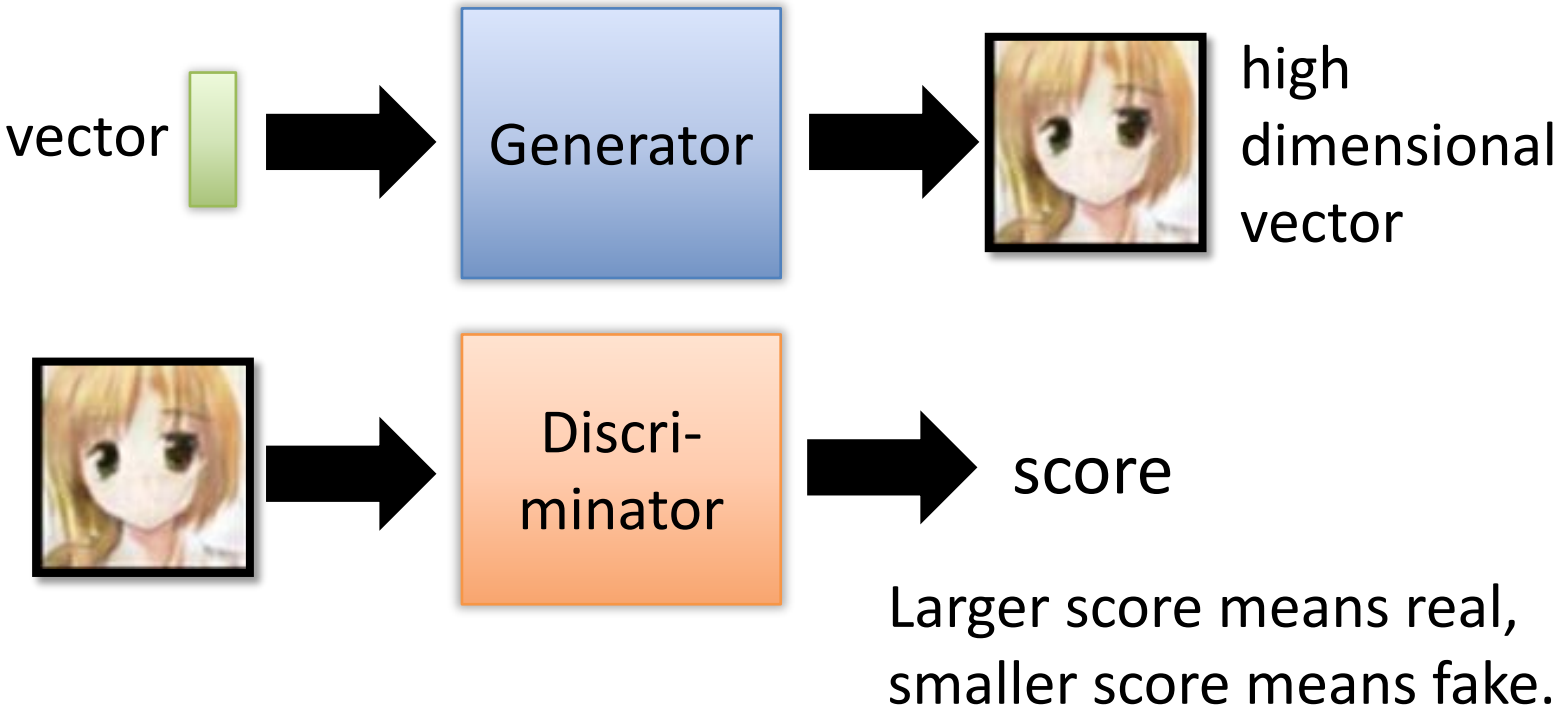
y

Vincent van Gogh's style

UNIVERSITÉ CÔTE D'AZUR
univ-cotedazur.fr

Generative Adversarial Network (GAN)

- Anime face generation as example



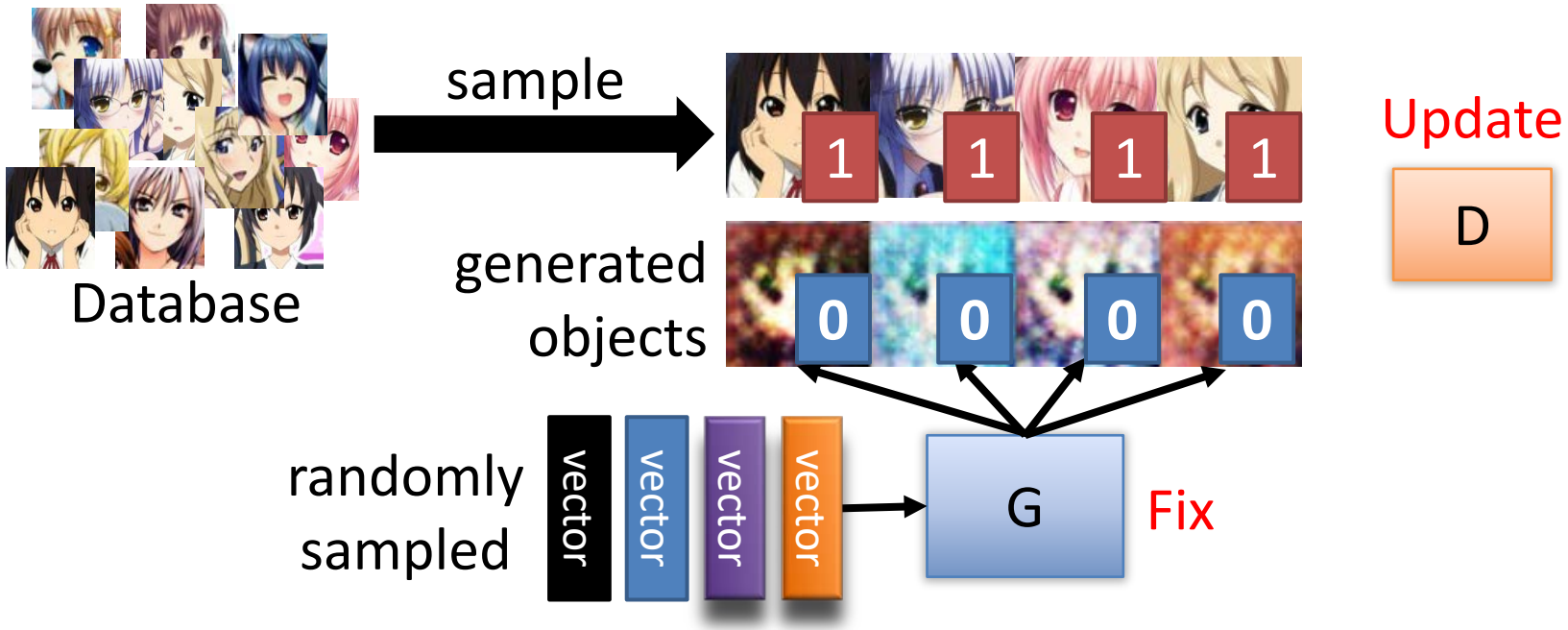
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Algorithm

- Initialize generator and discriminator
- In each training iteration:

Step 1: Fix generator G, and update discriminator D



Discriminator learns to assign high scores to real objects and low scores to generated objects.

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

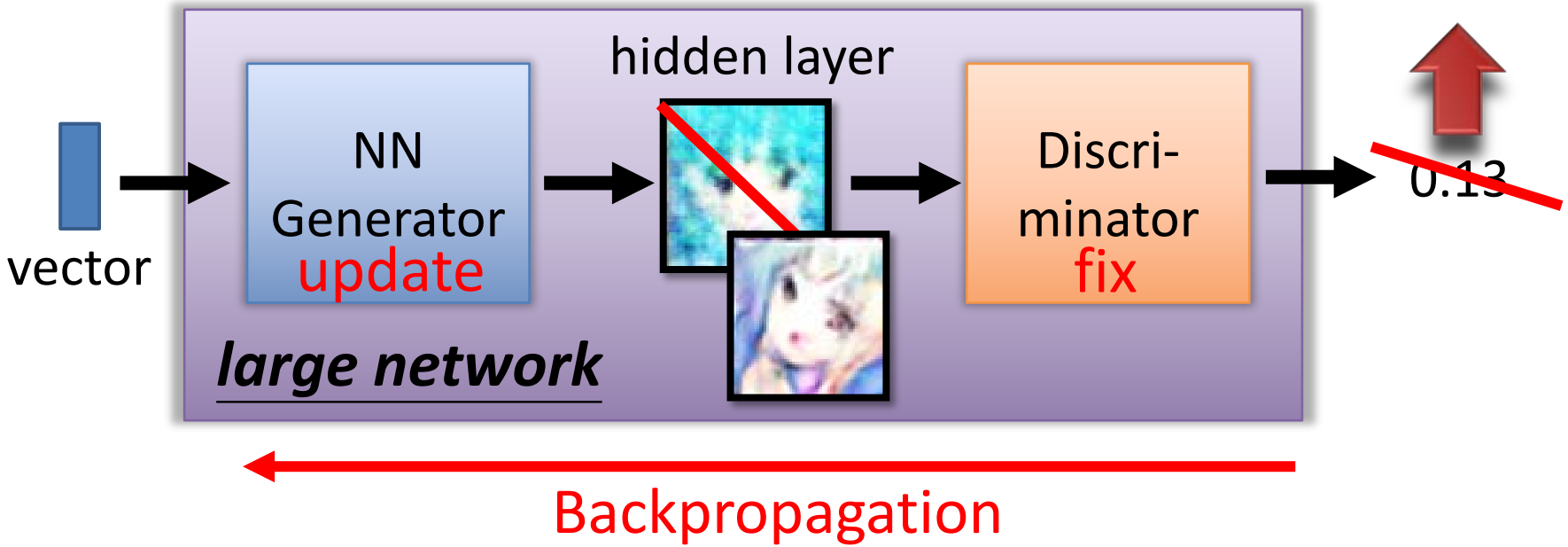
Algorithm

- Initialize generator and discriminator
- In each training iteration:

G D

Step 2: Fix discriminator D, and update generator G

Generator learns to "*fool*" the discriminator

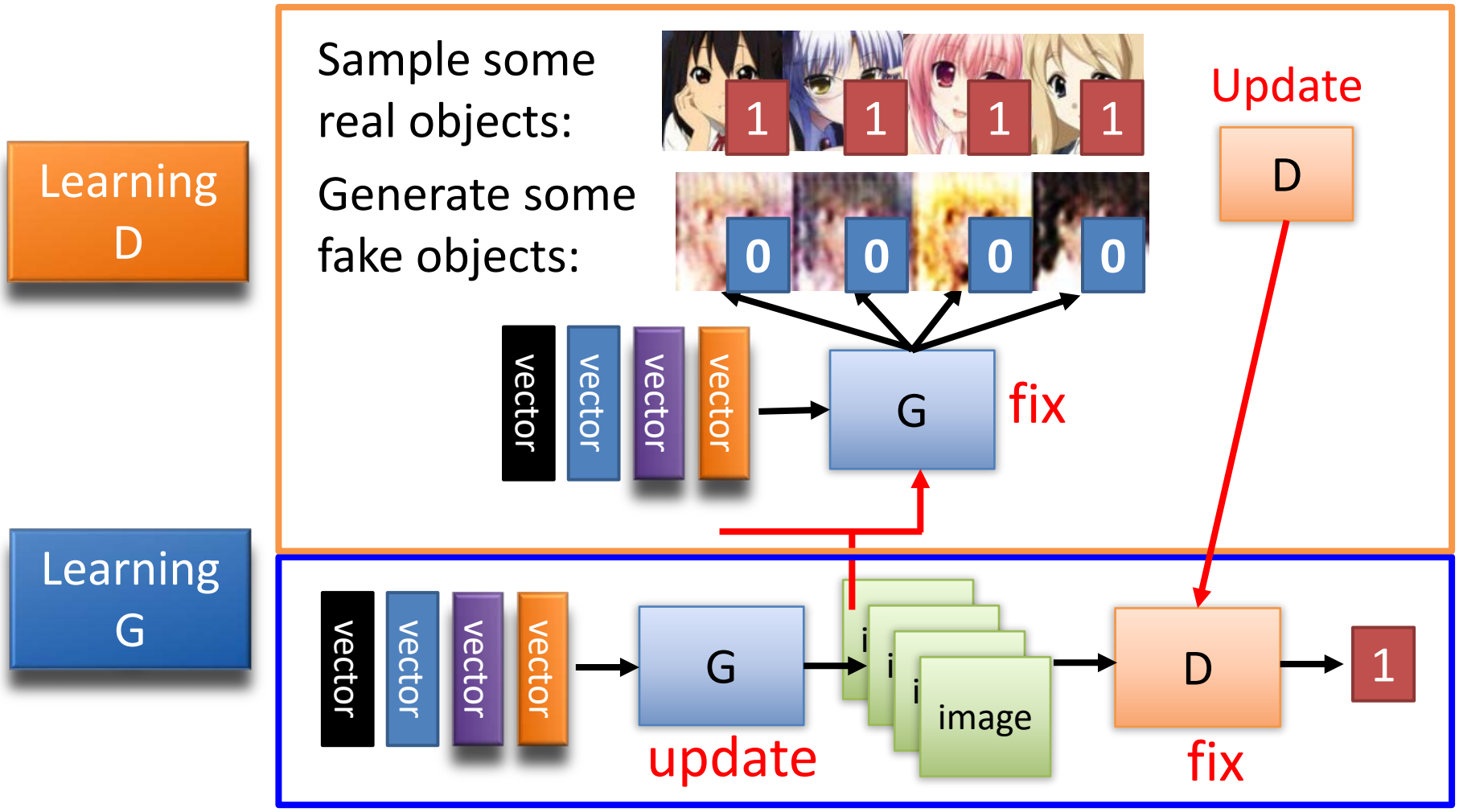


UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Algorithm

- Initialize generator and discriminator
- In each training iteration:



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

GAN is hard to train ……

- There is a saying ……

(Joke from facebook)

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Three Categories of GANs

1. *Typical GAN*

$$\begin{bmatrix} -0.3 \\ 0.1 \\ \vdots \\ 0.9 \end{bmatrix}$$

random vector → Generator →

image

2. *Conditional GAN*

blue eyes,
red hair,
short hair

paired data

"Girl with red hair"

text → Generator →

image

3. *Unsupervised Conditional GAN*

domain x

domain y

unpaired data

x

Photo → Generator →

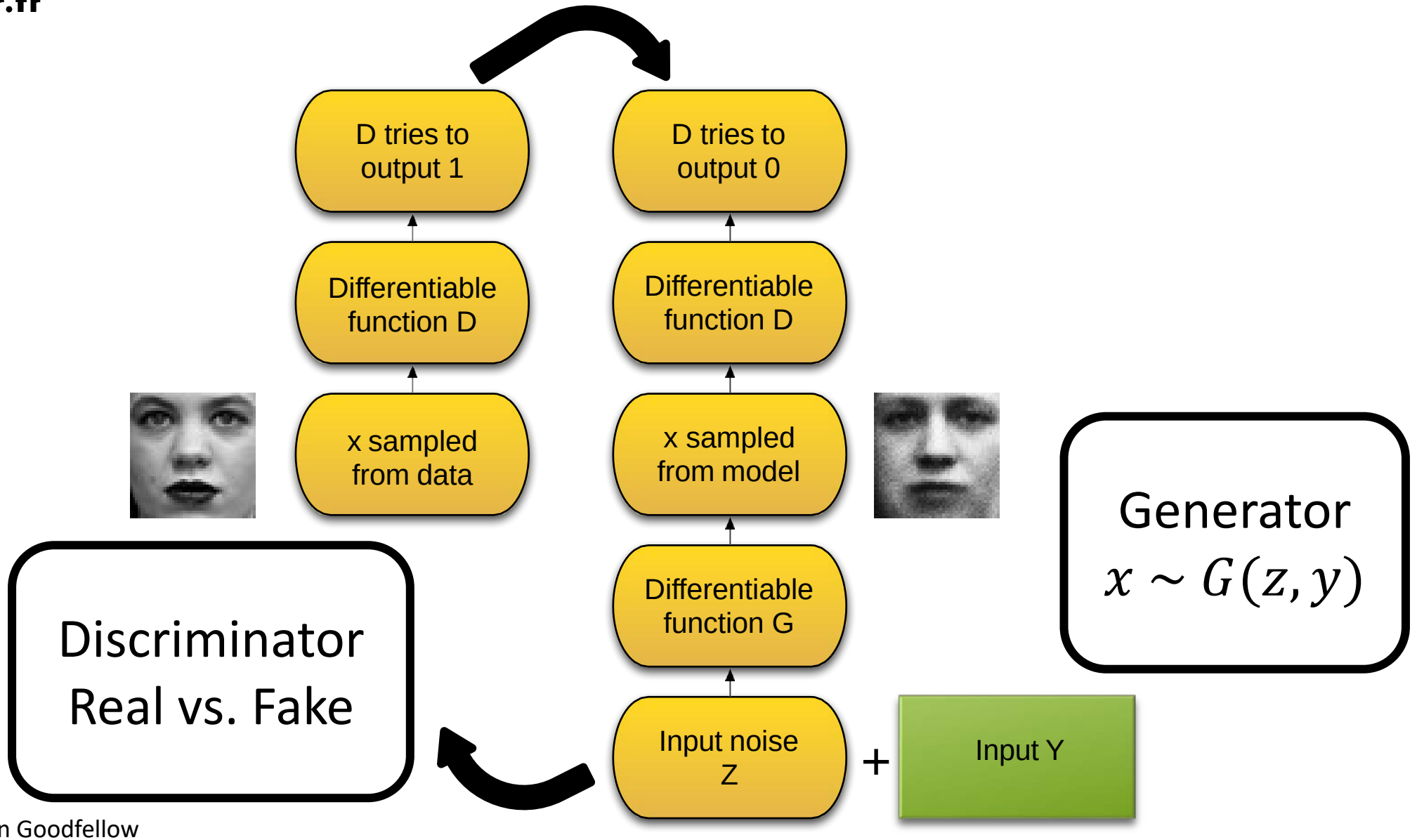
y

Vincent van Gogh's style

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

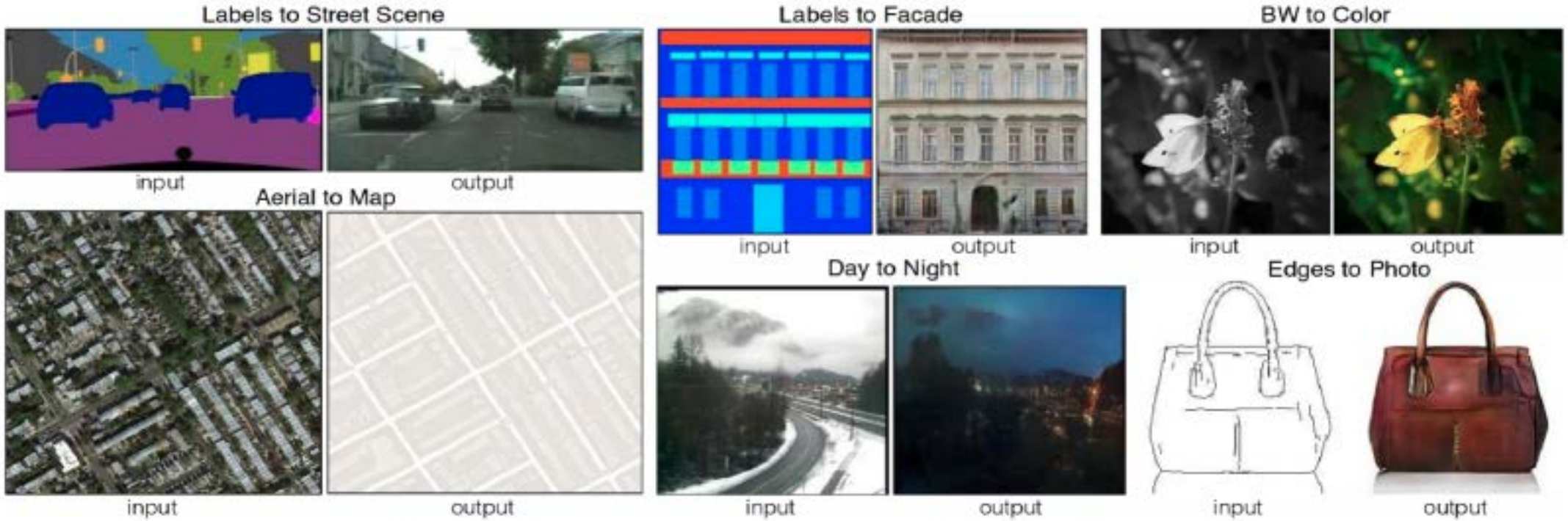
Conditional GANs



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

It finally did not solve adversarial, but…



UNIVERSITÉ CÔTE D'AZUR

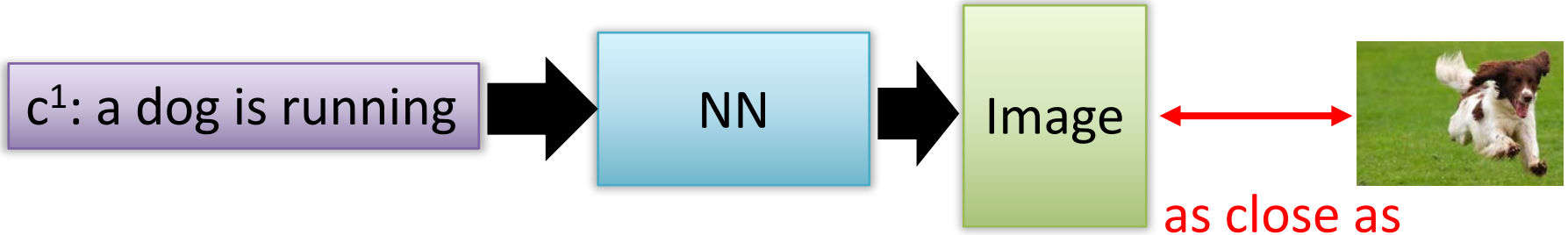
univ-cotedazur.fr

Text-to-Image

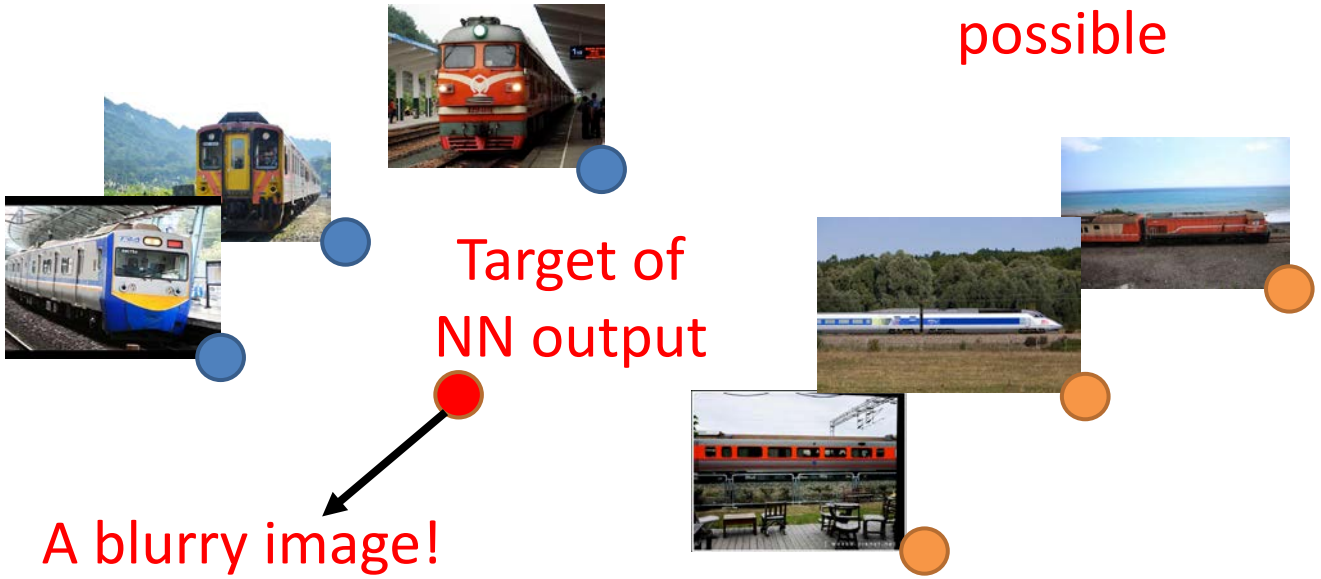
a dog is running 

a bird is flying 

- **Traditional supervised approach**

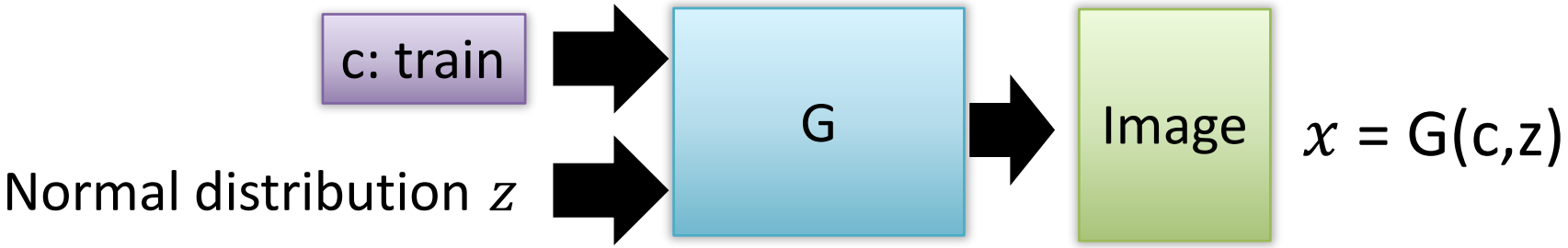


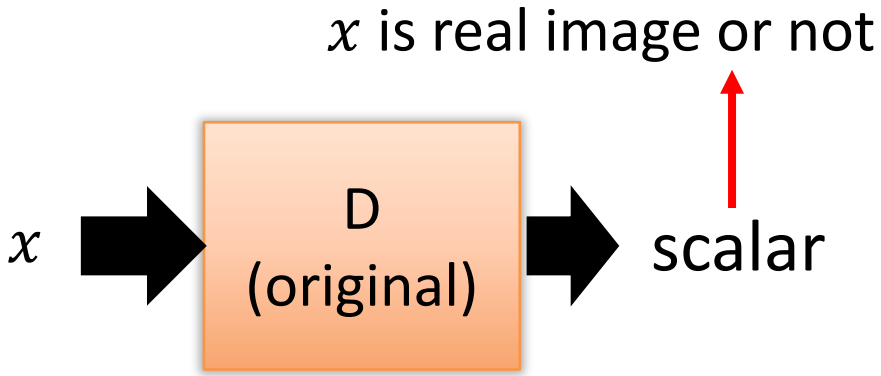
Text: "train"



[Scott Reed, et al, ICML, 2016]

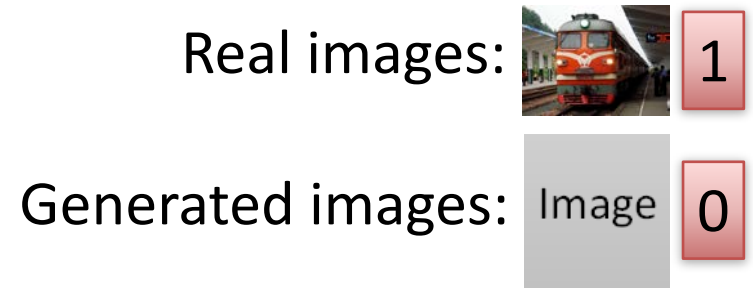
Conditional GAN





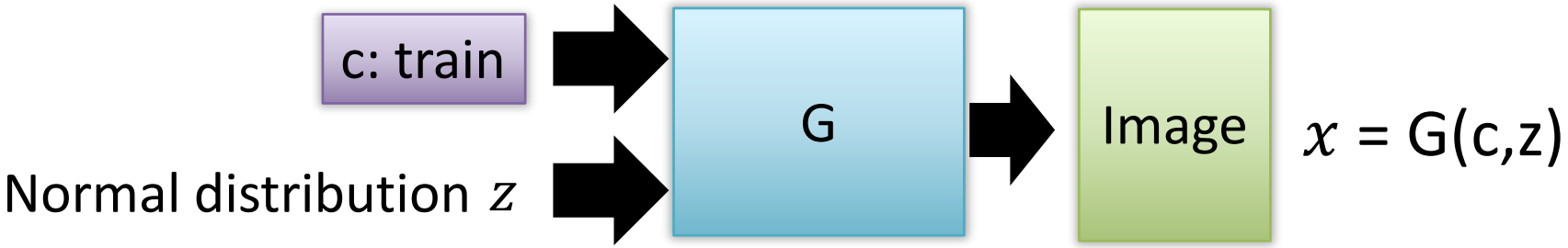
Generator will learn to generate realistic images ….

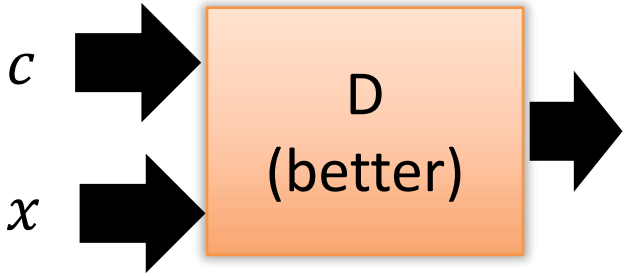
But completely ignore the input conditions.



[Scott Reed, et al, ICML, 2016]

Conditional GAN





x is realistic or not +
c and x are matched or not

True text-image pairs: (train , ) 1

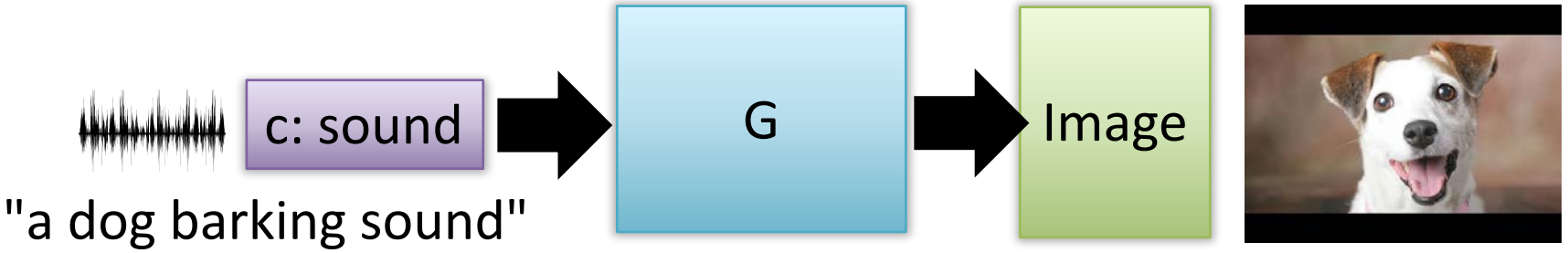
(cat , ) 0

(train , Image) 0

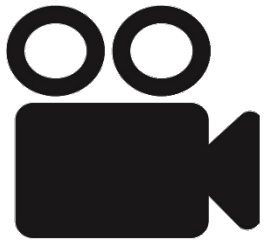
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Conditional GAN - Sound-to-image



Training Data Collection



UNIVERSITÉ CÔTE D'AZUR

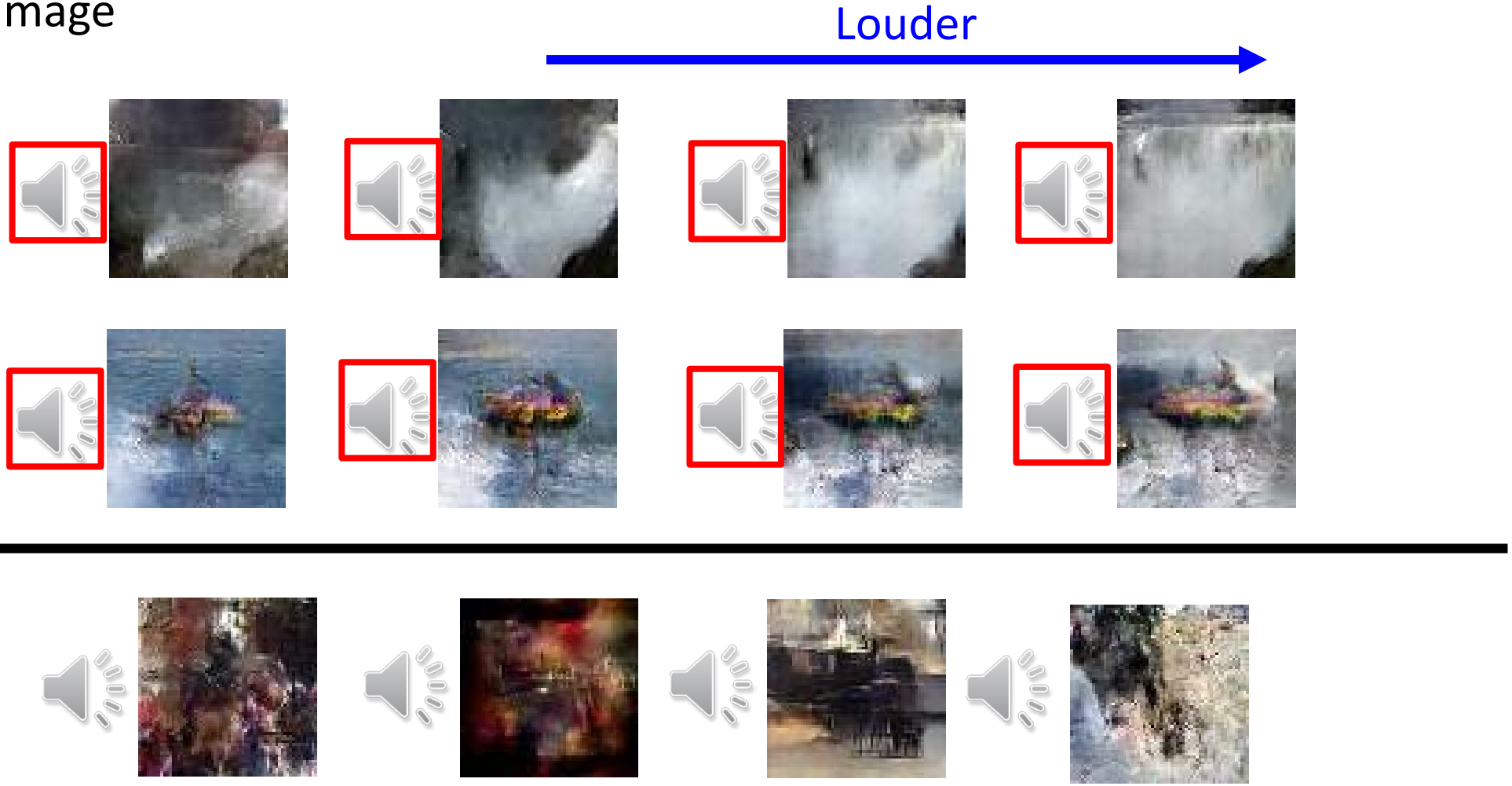
univ-cotedazur.fr

Conditional GAN - Sound-to-image

The images are generated by Chia-Hung Wan and Shun-Po Chuang.

https://wjohn1483.github.io/audio_to_scene/index.html

- Audio-to-image



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

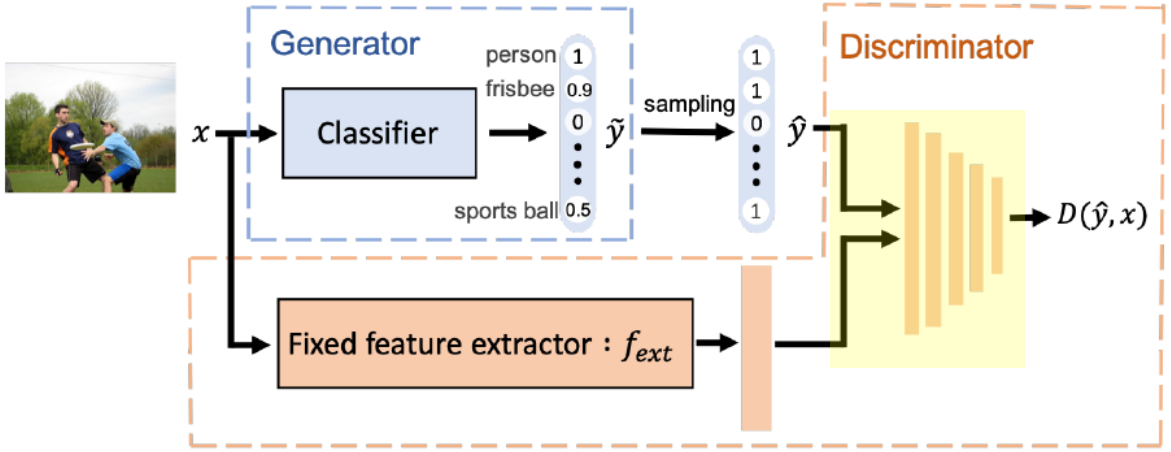
Conditional GAN - Image-to-label

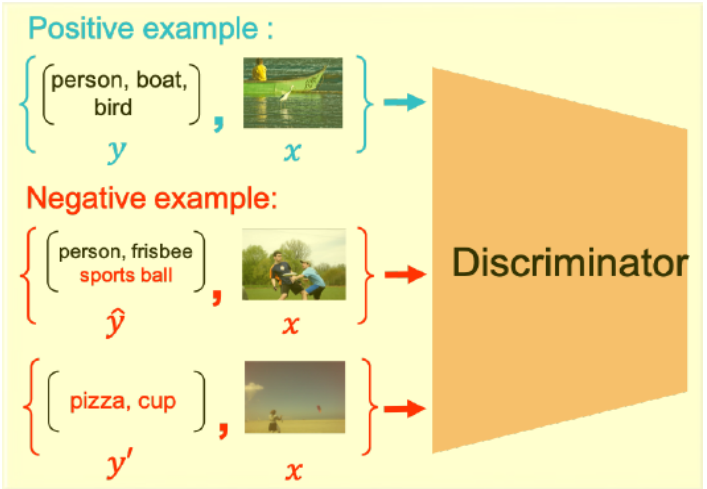
Multi-label Image Classifier

person, sports ball,
baseball bat, baseball glove

Input condition

Generated output





Conditional GAN - Image-to-label

The classifiers can have different architectures.

The classifiers are trained as conditional GAN.

F1	MS-COCO	NUS-WIDE
VGG-16	56.0	33.9
+ GAN	60.4	41.2
Inception	62.4	53.5
+GAN	63.8	55.8
Resnet-101	62.8	53.1
+GAN	64.0	55.4
Resnet-152	63.3	52.1
+GAN	63.9	54.1
Att-RNN	62.1	54.7
RLSD	62.0	46.9

[Tsai, et al., submitted to ICASSP 2019]

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Conditional GAN - Image-to-label

The classifiers can have different architectures.

The classifiers are trained as conditional GAN.

Conditional GAN outperforms other models designed for multi-label.

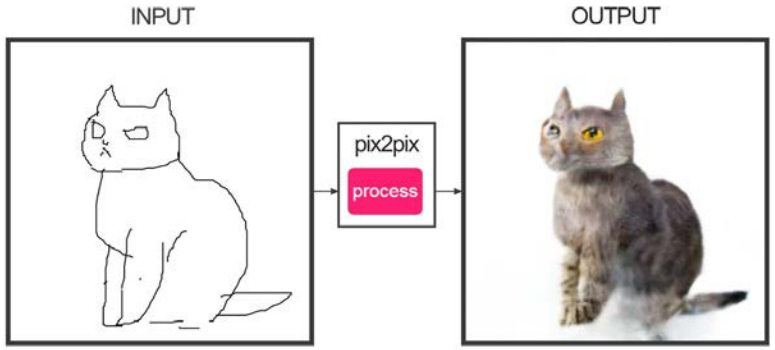
F1	MS-COCO	NUS-WIDE
VGG-16	56.0	33.9
+ GAN	60.4	41.2
Inception	62.4	53.5
+GAN	63.8	55.8
Resnet-101	62.8	53.1
+GAN	64.0	55.4
Resnet-152	63.3	52.1
+GAN	63.9	54.1
Att-RNN	62.1	54.7
RLSD	62.0	46.9

RLSD: An End-to-End CNN+LSTM Model for Multi-Label Image Classification

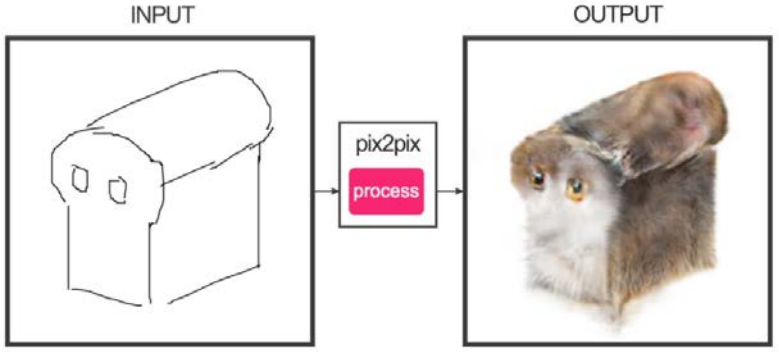
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

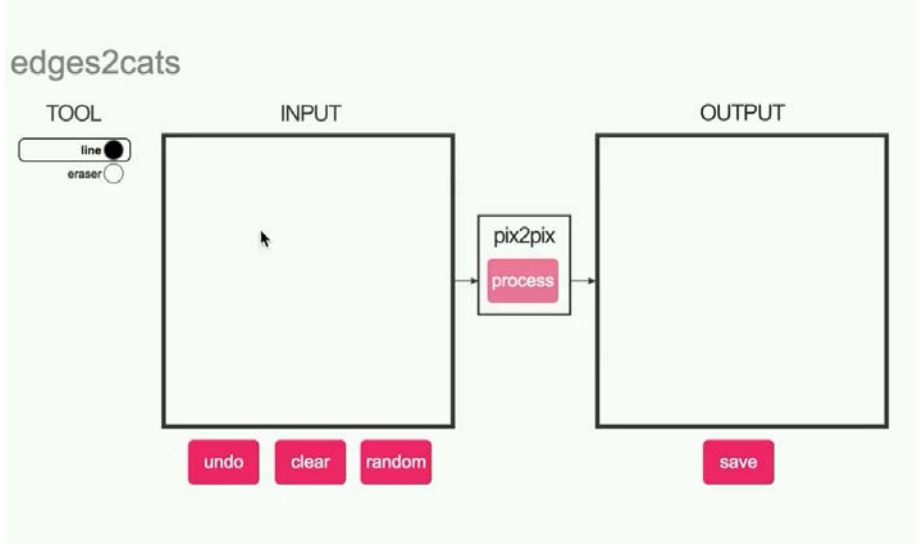
#edges2cats [Christopher Hesse]



@gods_tail



Ivy Tasi @ivymyt



@matthematician

Vitaly Vidmirov @vvid

https://affinelayer.com/pixsrv/

Pix2pix / CycleGAN

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Edges → Images

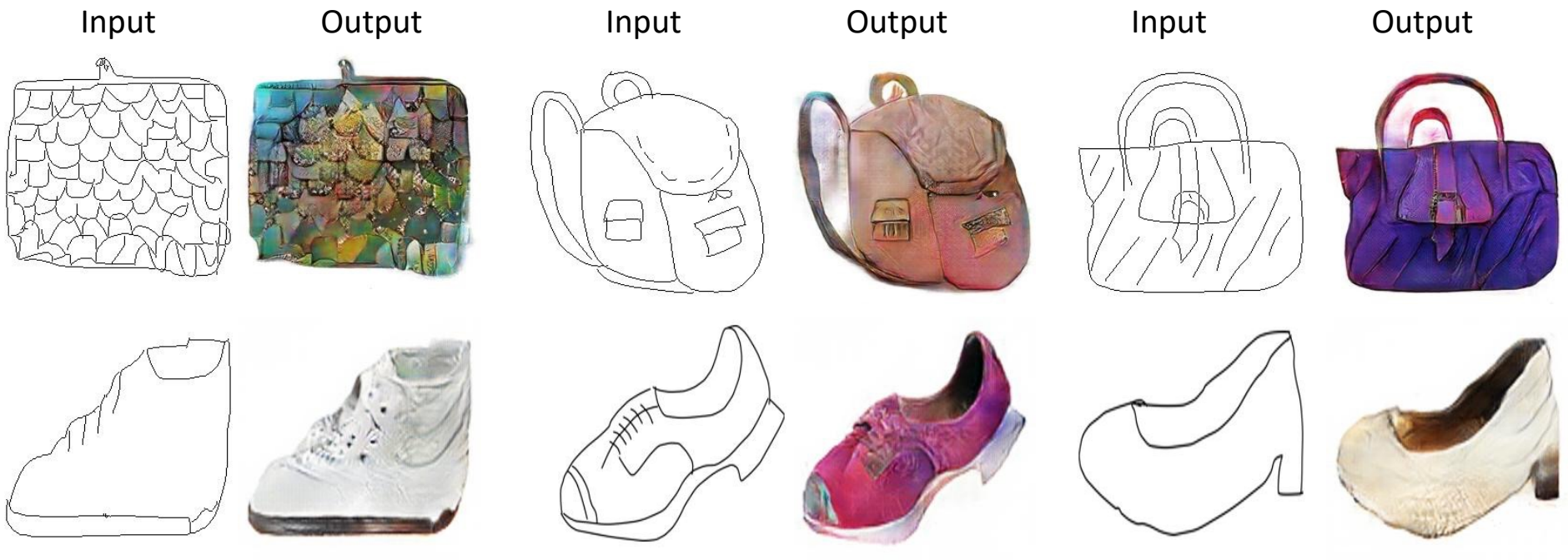
Edges from [Xie & Tu, 2015]

Pix2pix / CycleGAN

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Sketches → Images



Trained on Edges → Images

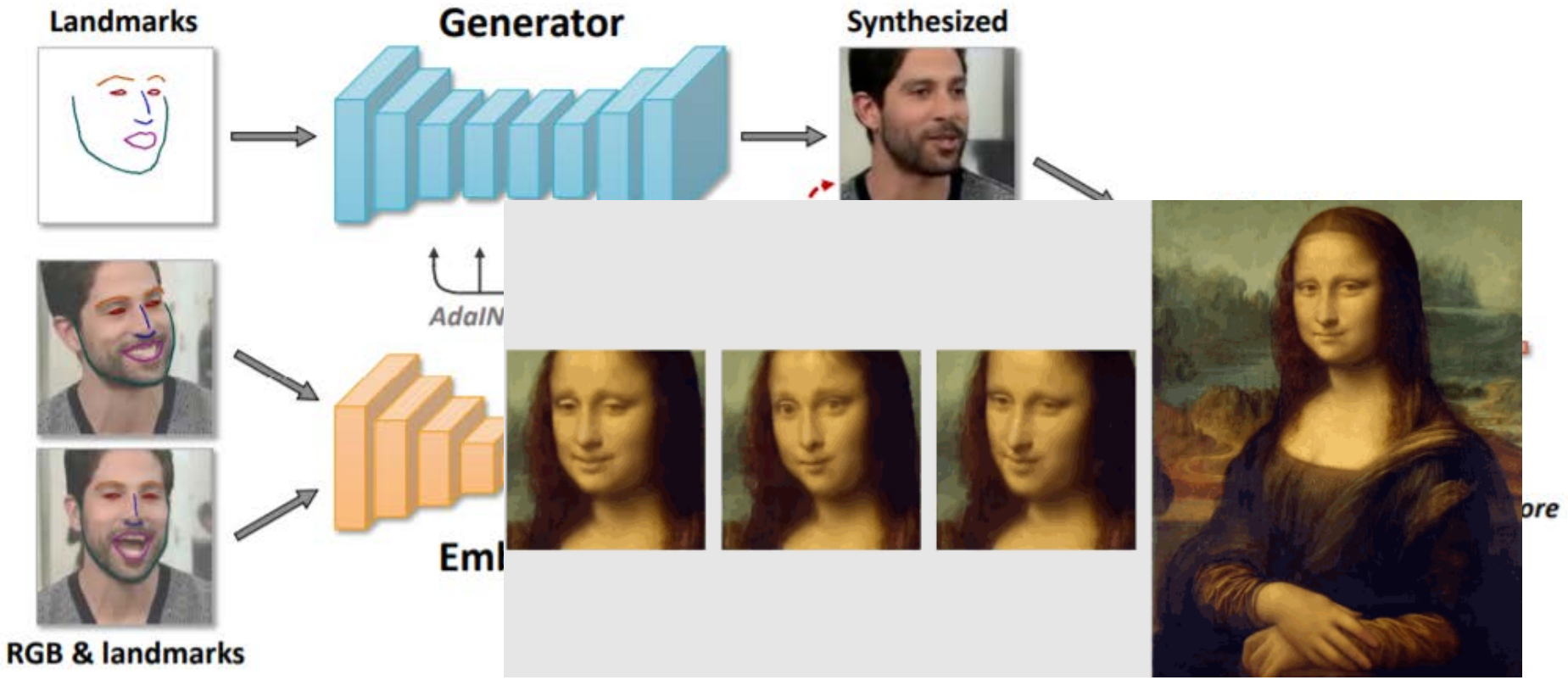
Data from [Eitz, Hays, Alexa, 2012]

Pix2pix / CycleGAN

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Talking Head

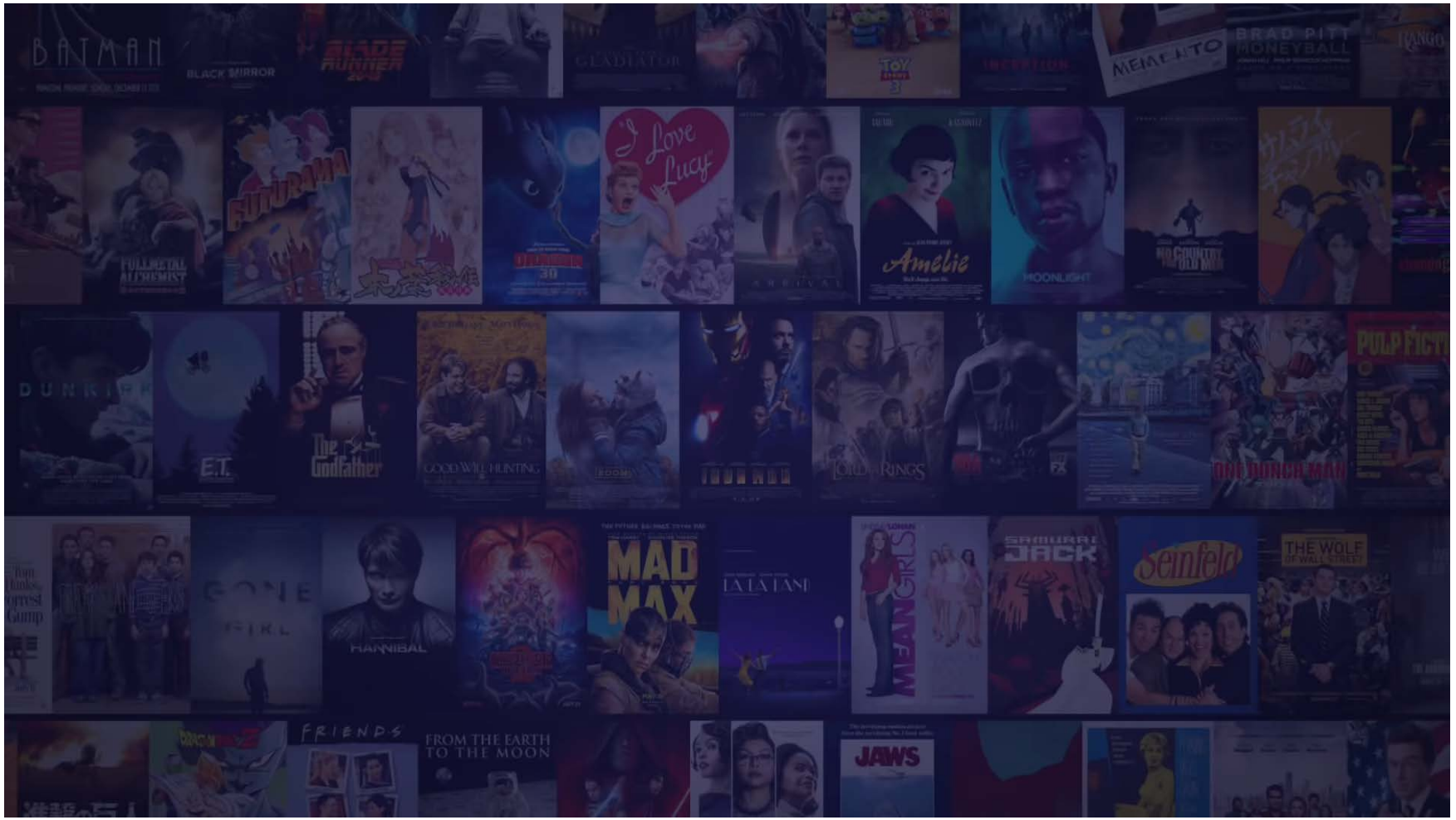


https://arxiv.org/abs/1905.08233

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Deepfake



https://www.youtube.com/watch?v=-QvIX3cY4lc

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Three Categories of GANs

1. *Typical GAN*

$$\begin{bmatrix} -0.3 \\ 0.1 \\ \vdots \\ 0.9 \end{bmatrix}$$

random vector → Generator →

image

2. *Conditional GAN*

blue eyes,
red hair,
short hair

paired data

"Girl with red hair"

text → Generator →

image

3. *Unsupervised Conditional GAN*

domain x

domain y

unpaired data

x

Photo → Generator →

y

Vincent van Gogh's style

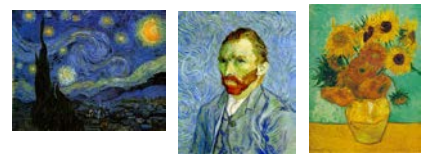
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Cycle GAN

Domain X

Domain Y



Domain X

$G_{X \to Y}$

Become similar to domain Y

D_Y

scalar

Input image belongs to domain Y or not

Domain Y

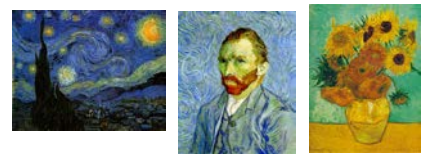
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Cycle GAN

Domain X

Domain Y



Domain X

ignore input

$G_{X \to Y}$

Become similar to domain Y

Not what we want!

D_Y

scalar

Domain Y

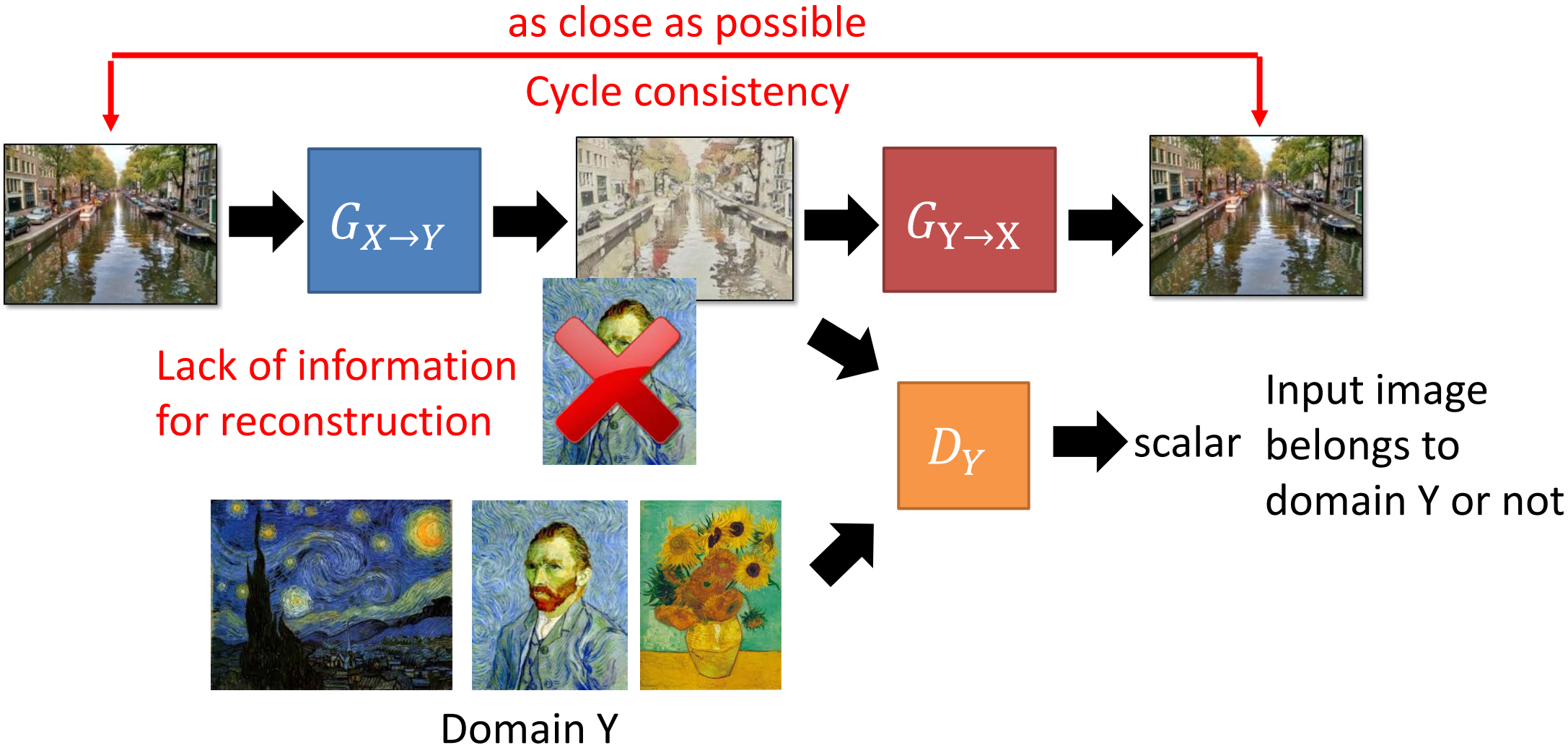
Input image belongs to domain Y or not

UNIVERSITÉ CÔTE D'AZUR

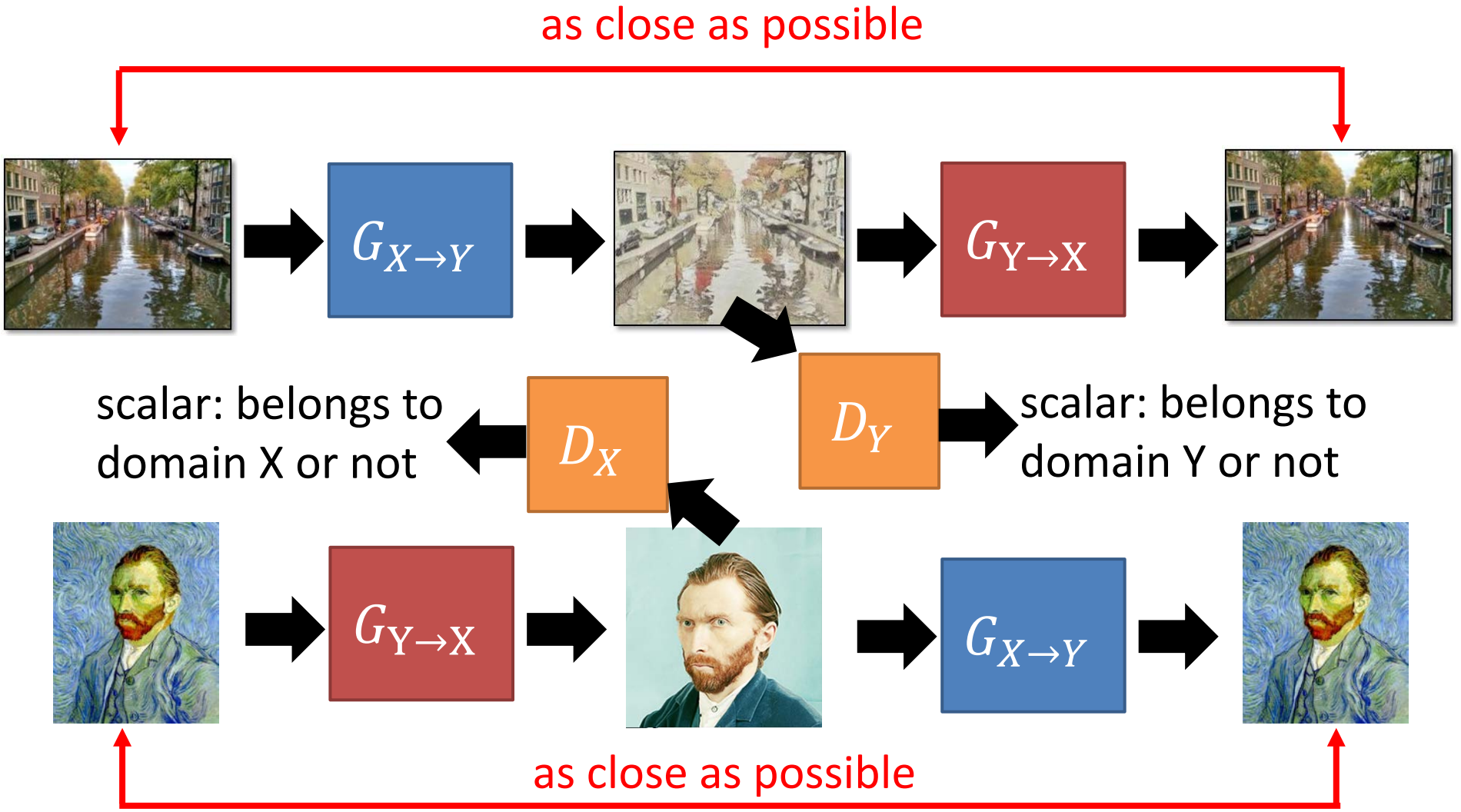
univ-cotedazur.fr

Cycle GAN

[Jun-Yan Zhu, et al., ICCV, 2017]

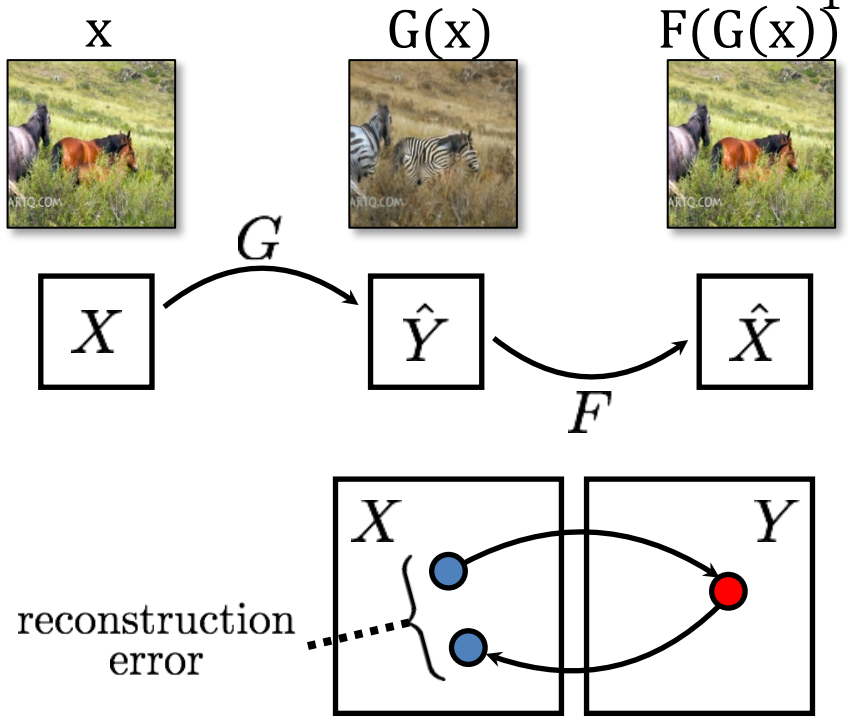


Cycle GAN

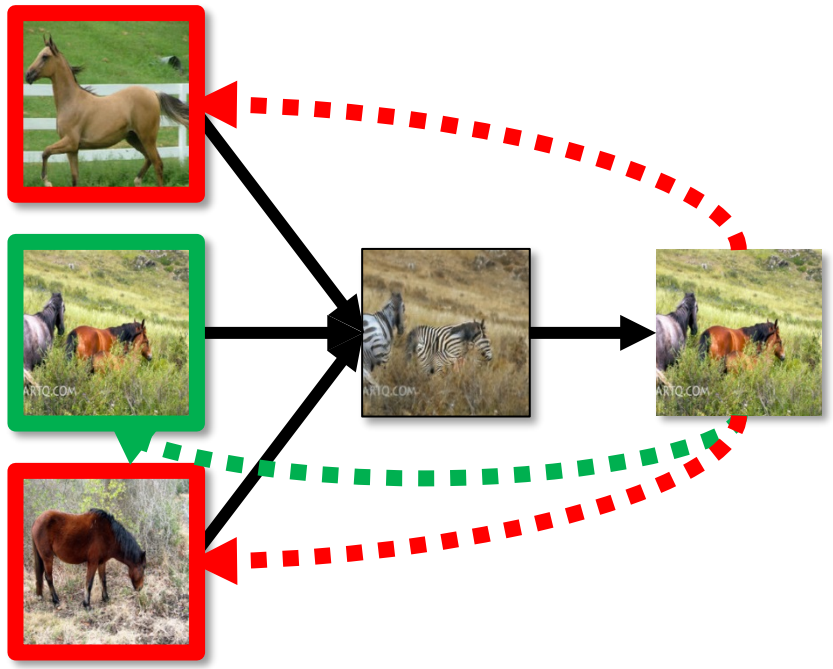

as close as possible
$G_{X\to Y}$
$G_{Y\to X}$
D_X
D_Y
scalar: belongs to domain X or not
scalar: belongs to domain Y or not
$G_{Y\to X}$
$G_{X\to Y}$
as close as possible

Cycle Consistency

Forward cycle loss: $\|F(G(x)) - x\|_1$



Small cycle loss



Helps cope with mode collapse

Zhu et al., "Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks", ICCV 2017

Training Details: Objective

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) =& \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log D_Y(y)] \\ &+ \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x))],\end{aligned}$$

$$\begin{aligned}\mathcal{L}_{\text{cyc}}(G, F) =& \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] \\ &+ \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1].\end{aligned}$$

$$\begin{aligned}\mathcal{L}(G, F, D_X, D_Y) =& \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) \\ &+ \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) \\ &+ \lambda \mathcal{L}_{\text{cyc}}(G, F),\end{aligned}$$

$$G^*, F^* = \arg \min_{G,F} \max_{D_x, D_Y} \mathcal{L}(G, F, D_X, D_Y).$$

Zhu et al., "Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks", ICCV 2017

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Pix2pix / CycleGAN

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Pix2pix / CycleGAN

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Celebrities Who Never Existed

Karras et al., "Progressive Growing of GANs for Improved Quality, Stability, and Variation", ICLR 2018

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

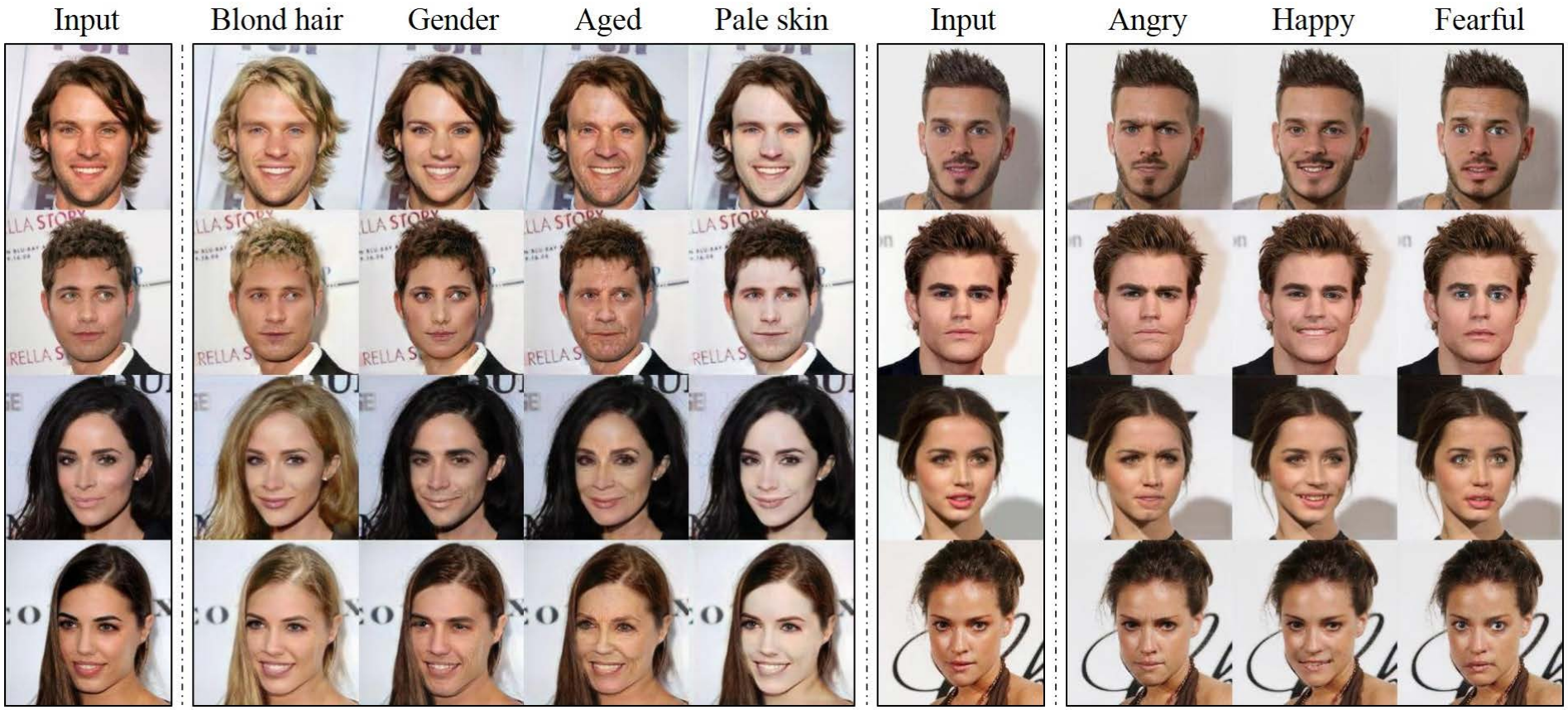
Creative Adversarial Networks

(Elgammal et al., 2017)

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

StarGAN

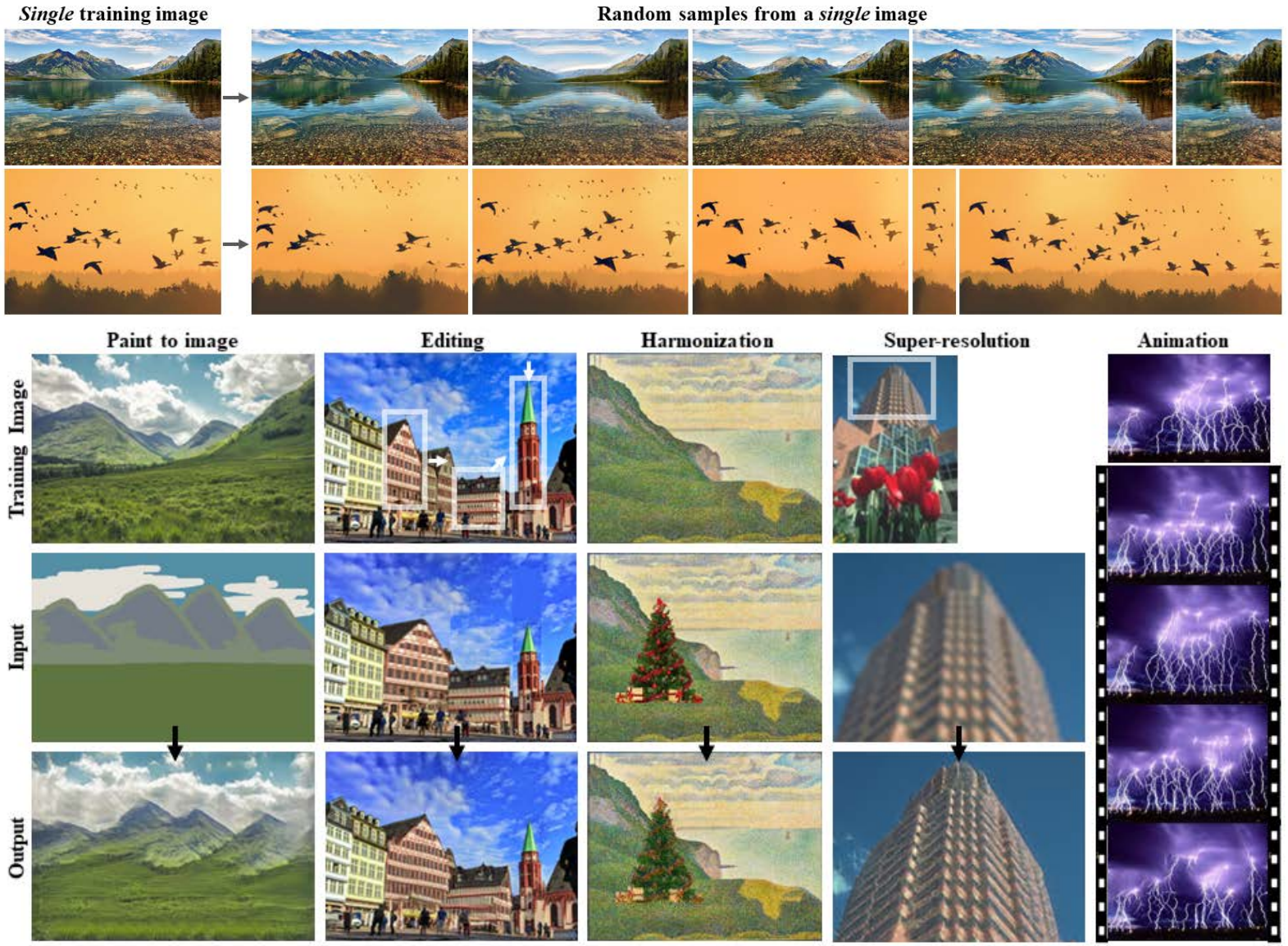


Choi et al., "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation", CVPR 2018

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

SinGAN

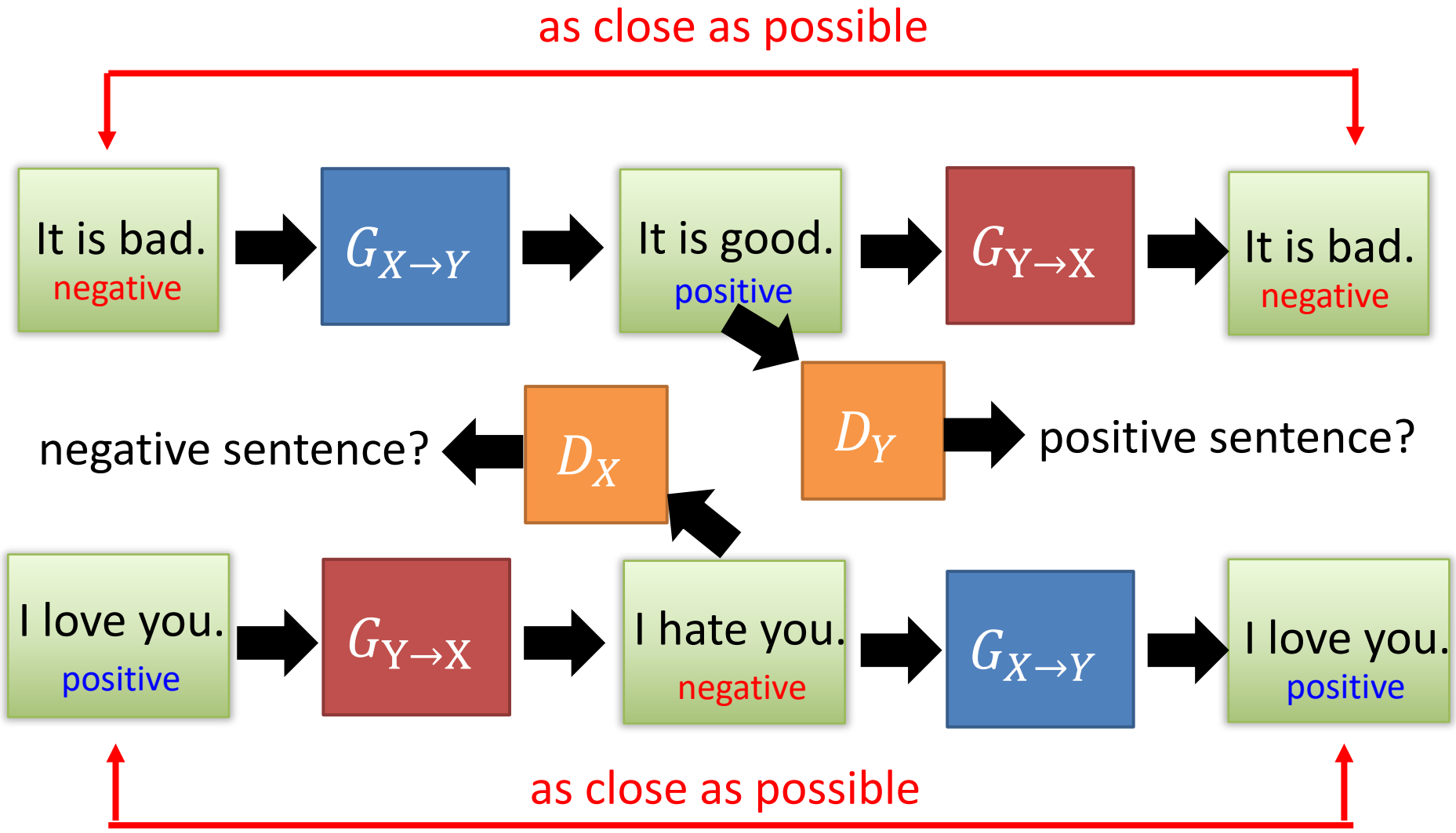


Shaham et al., "SinGAN: Learning a Generative Model from a Single Natural Image", ICCV 2019

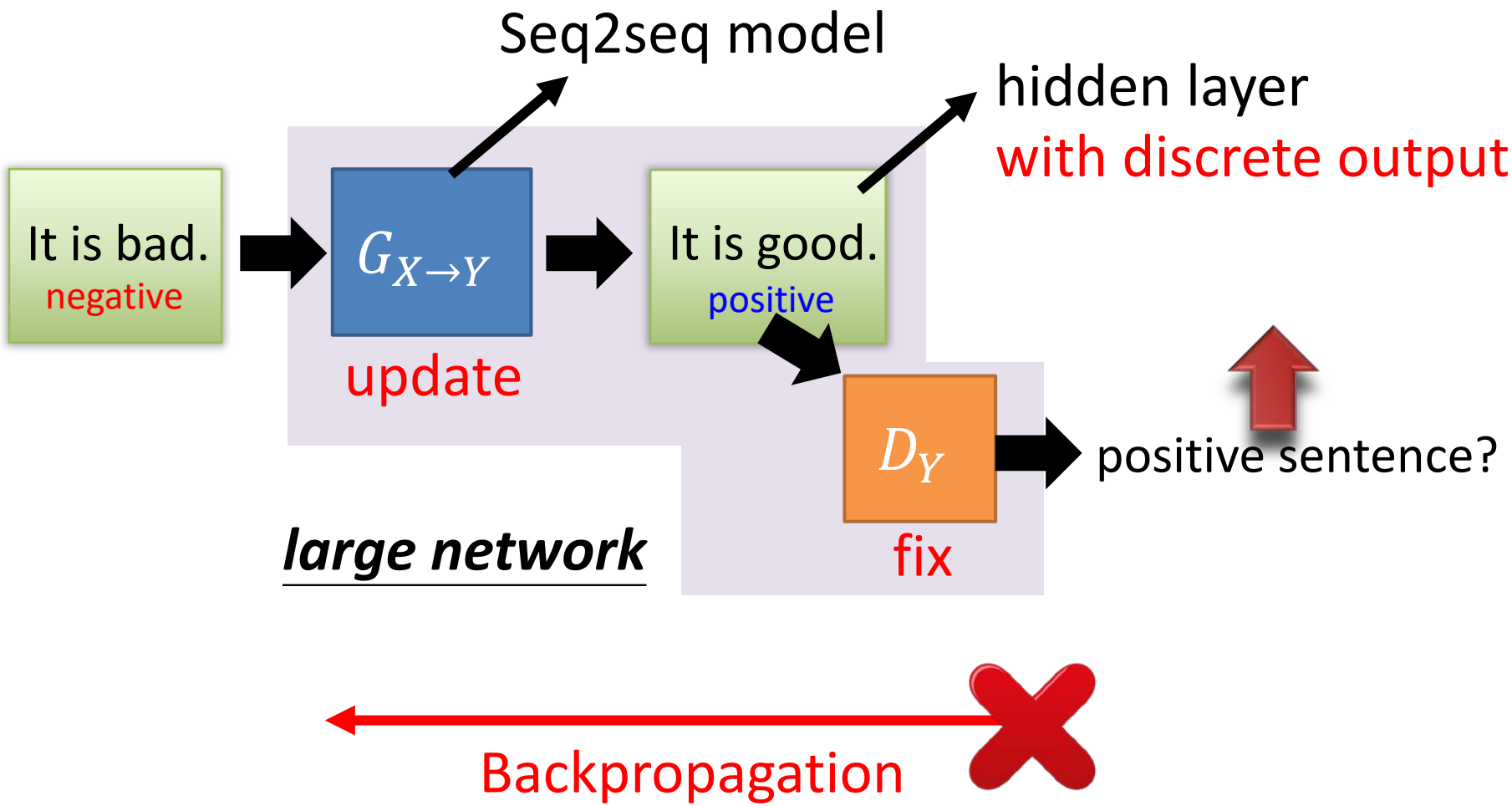
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Cycle GAN



Discrete Issue



UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Three Categories of Solutions

Gumbel-softmax

- [Matt J. Kusner, et al, arXiv, 2016]

Continuous Input for Discriminator

- [Sai Rajeswar, et al., arXiv, 2017][Ofir Press, et al., ICML workshop, 2017][Zhen Xu, et al., EMNLP, 2017][Alex Lamb, et al., NIPS, 2016][Yizhe Zhang, et al., ICML, 2017]

"Reinforcement Learning"

- [Yu, et al., AAAI, 2017][Li, et al., EMNLP, 2017][Tong Che, et al, arXiv, 2017][Jiaxian Guo, et al., AAAI, 2018][Kevin Lin, et al, NIPS, 2017][William Fedus, et al., ICLR, 2018]

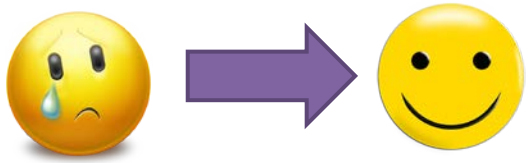
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Text rewritting

✗ Negative sentence to positive sentence:

- it's a crappy day -> it's a great day
  i wish you could be here -> you could be here
  it's not a good idea -> it's good idea
  i miss you -> i love you
  i don't love you -> i love you
  i can't do that -> i can do that
  i feel so sad -> i happy
  it's a bad day -> it's a good day
  it's a dummy day -> it's a great day
  sorry for doing such a horrible thing -> thanks for doing a great thing
  my doggy is sick -> my doggy is my doggy
  my little doggy is sick -> my little doggy is my little doggy

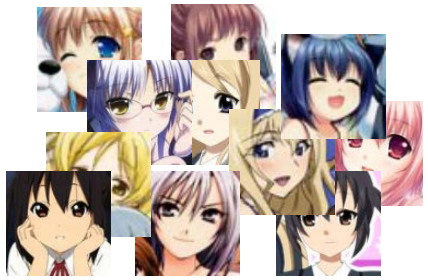


UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Three Categories of GANs

1. *Typical GAN*



$$\begin{bmatrix} -0.3 \\ 0.1 \\ \vdots \\ 0.9 \end{bmatrix}$$

random vector → Generator →

image

2. *Conditional GAN*

blue eyes,
red hair,
short hair

paired data

"Girl with red hair"
text
→ Generator →

image

3. *Unsupervised Conditional GAN*

domain x

domain y

unpaired data

x

Photo
→ Generator →

y

Vincent van Gogh's style

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Ethics (Politics, Privacy)

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Ethics (Politics, Privacy)

- Deep fakes
- Privacy
- Security and adversarial perturbations

Tips & Tricks

- Feature matching (facial for instance, see deepfake)

Deepfake

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

"Deepfakes"

https://www.technologyreview.com/s/611726/the-defense-department-has-produced-the-first-tools-for-catching-deepfakes/
https://www.niemanlab.org/2018/11/how-the-wall-street-journal-is-preparing-its-journalists-to-detect-deepfakes/

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Full person!!!

Deepfake

https://information.tv5monde.com/video/deepfake-de-tom-cruise-peut-encore-croire-ce-que-l-voit-vrai-dire
https://www.youtube.com/watch?v=SDty803-hxg&ab_channel=LeHuffPost
https://edition.cnn.com/videos/business/2021/03/02/tom-cruise-tiktok-deepfake-orig.cnn-business/video/playlists/business-misinformation/

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

You can be anyone you want…

Karras et al., "A Style-Based Generator Architecture for Generative Adversarial Networks", CVPR 2019, https://arxiv.org/pdf/1812.04948.pdf

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Detection methods

FaceForensics++: Learning to Detect Manipulated Facial Images

Andreas Rössler[1] Davide Cozzolino[2] Luisa Verdoliva[2] Christian Riess[3]
Justus Thies[1] Matthias Nießner[1]
[1]Technical University of Munich [2]University Federico II of Naples [3]University of Erlangen-Nuremberg

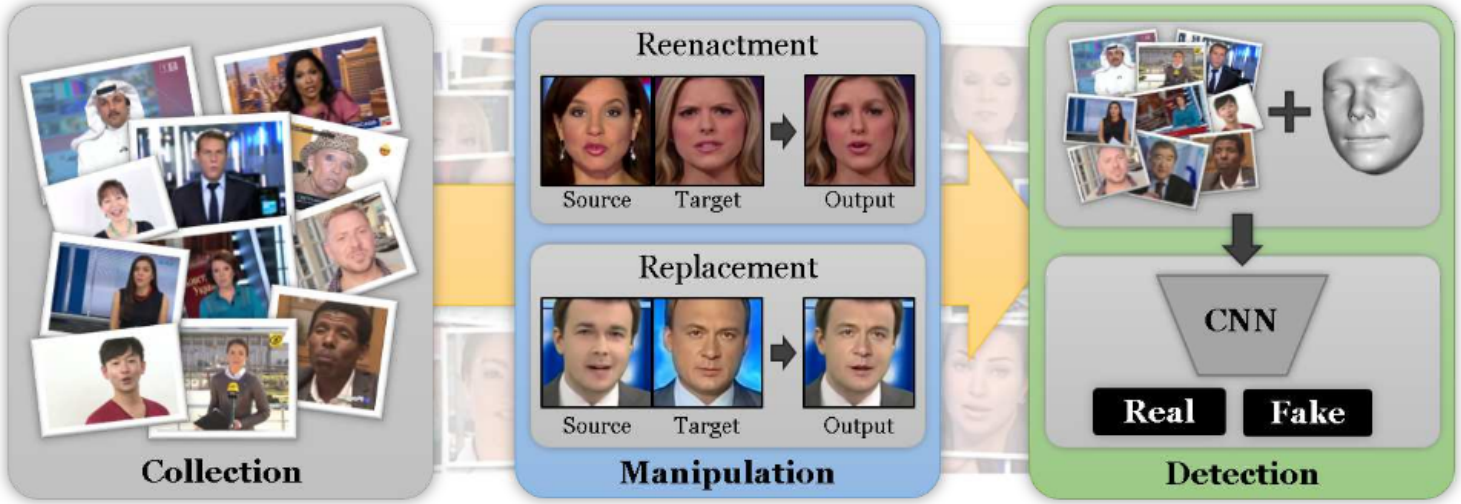


Figure 1: *FaceForensics++* is a dataset of facial forgeries that enables researchers to train deep-learning-based approaches in a supervised fashion. The dataset contains manipulations created with four state-of-the-art methods, namely, *Face2Face*, *FaceSwap*, *DeepFakes*, and *NeuralTextures*.

Rossler et al.. "FaceForensics++: Learning to Detect Manipulated Facial Images", ICCV 2019

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Detection methods

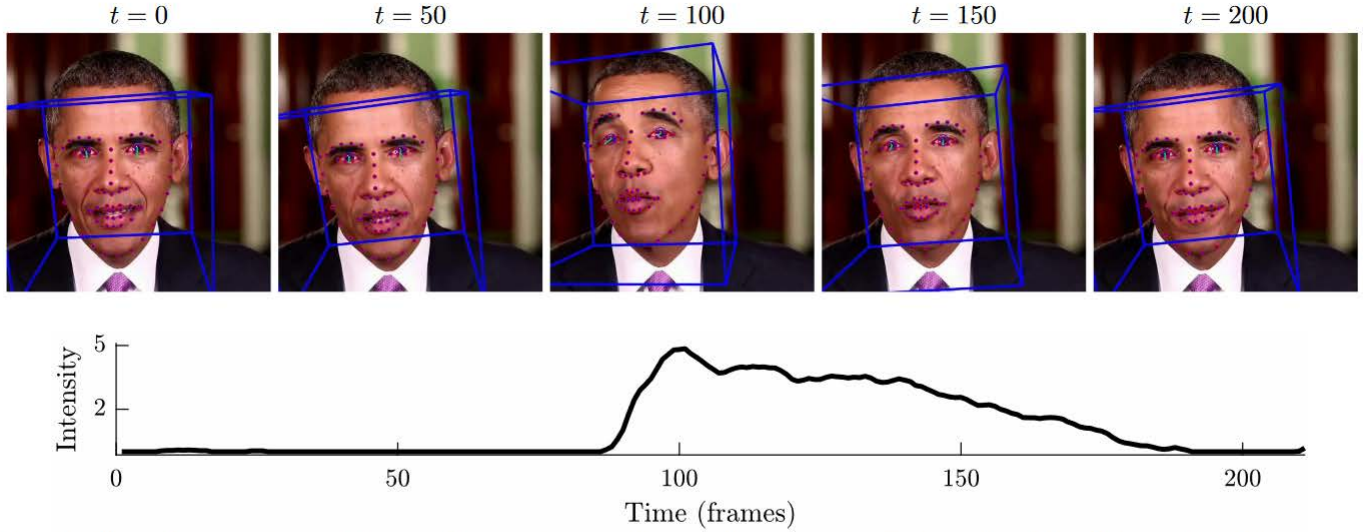


Figure 1. Shown above are five equally spaced frames from a 250-frame clip annotated with the results of OpenFace tracking. Shown below is the intensity of one action unit AU01 (eye brow lift) measured over this video clip.

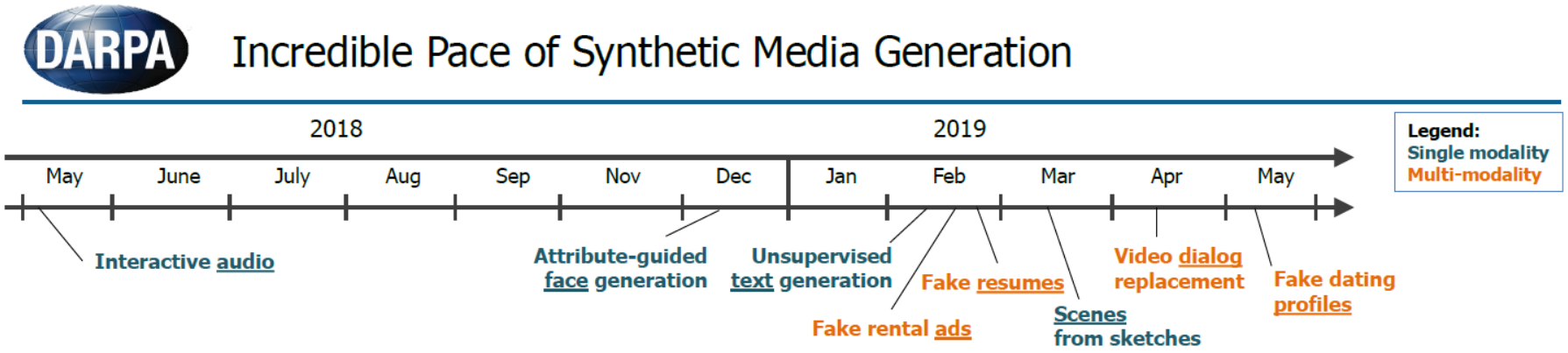
"We describe a forensic technique that models facial expressions and movements that typify an individual's speaking pattern. Although not visually apparent, these correlations are often violated by the nature of how deep-fake videos are created and can, therefore, be used for authentication.

Agarwal et al., "Protecting World Leaders Against Deep Fakes", CVPR Workshops, 2019

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

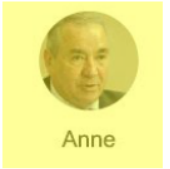
Successes



ENTIRE GUEST SUITE

Luxury Condo 3 Bed + 3 Bath

Port Melbourne



Anne

○ 8 guests ○ 3 bedrooms ○ 4beds ○ 2 baths

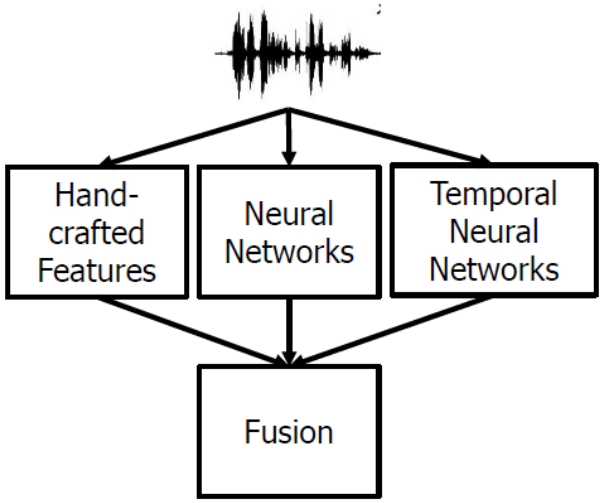
Bathroom (with seating for 2 more people), basin and eclectic French garden and kitchen. 24/7 carpeted charc. Laundrymemberly : More balcony – Garden – Metro, Liverpool Street (15 min walk) Walking distance to Wyckofferdon

Matt Turek

Successes

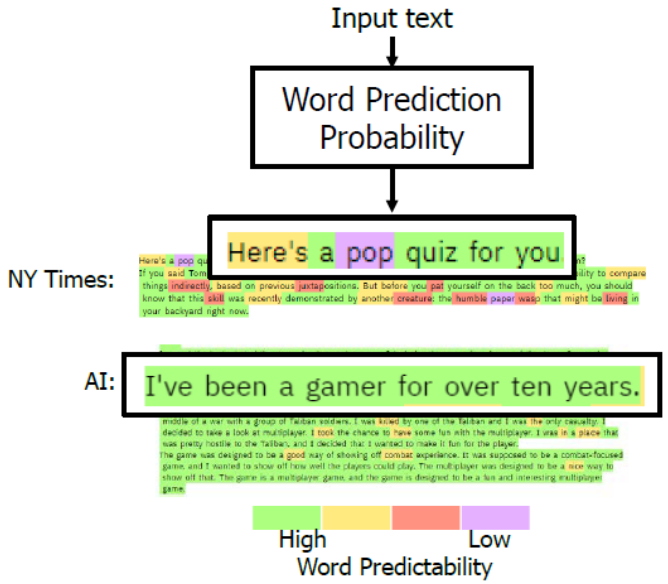
DARPA State of the Art Detection is Statistically Based, Narrow, or Both

Audio: ASVspoof



(Lavrentyeva et al. 2017)

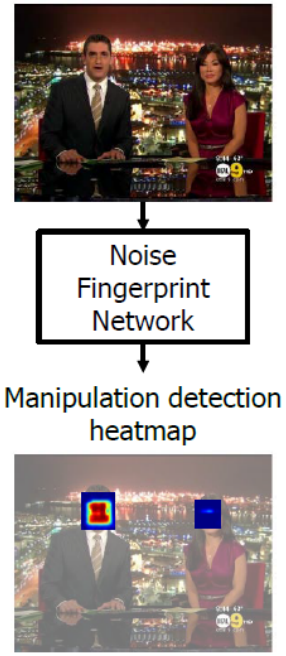
Text: GLTR



AI methods choose more predictable next-words than humans, statistically

(MIT-IBM Watson AI lab, HarvardNLP 2019)

Image/Video: DARPA MediFor



(MediFor: USC/ISI, Univ. Naples 2019)

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Expected Threats

DARPA Expected Threats

Targeted Personal Attacks

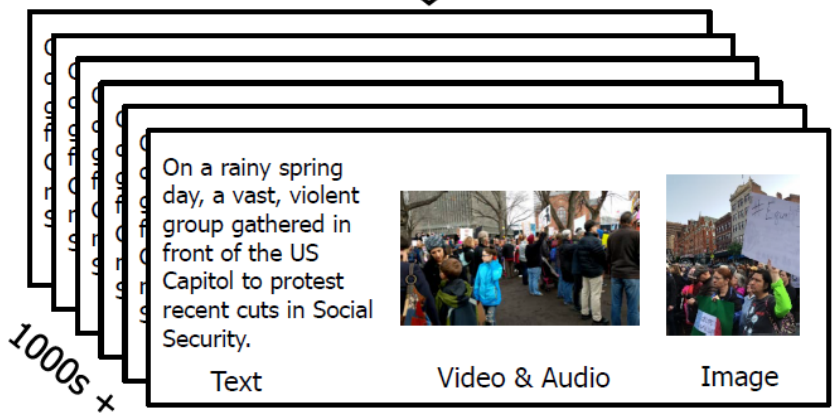
Peele 2017

AI Multimedia Algorithms

Highly realistic video

Generated Events at Scale

AI Multimedia Algorithms



Believable fake events

Ransomfake concept: Identity Attacks as a service (IAaaS)

Bricman 2019

AI Multimedia Algorithms

Forged Evidence

Identity Attacks

Examples of possible fakes:

- Substance abuse
- Foreign contacts
- Compromising events
- Social media postings
- Financial inconsistencies
- Forging identity

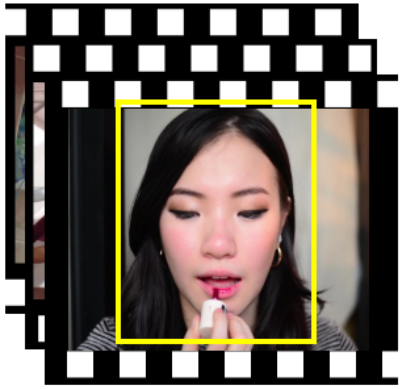
Undermines key individuals and organizations

Matt Turek

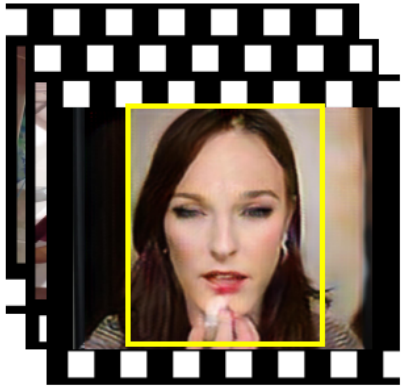
UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

GANs for Privacy (Action Detection)



Identity: Jessica
Action: Applying Make-up on Lips



Identity: ???
Action: Applying Make-up on Lips

Ren et al., "Learning to Anonymize Faces for Privacy Preserving Action Detection", ECCV 2018

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

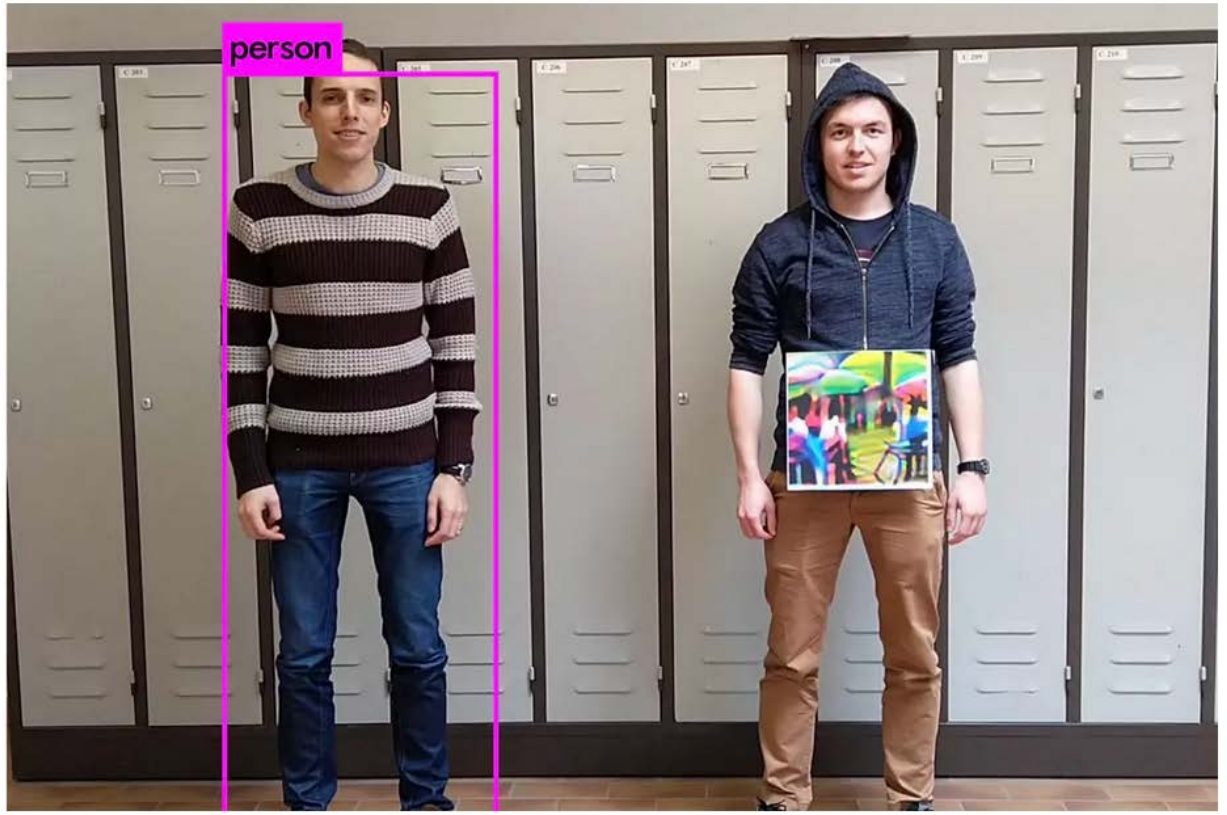
Adversarial Attacks

https://bair.berkeley.edu/blog/2017/12/30/yolo-attack/

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial Attacks

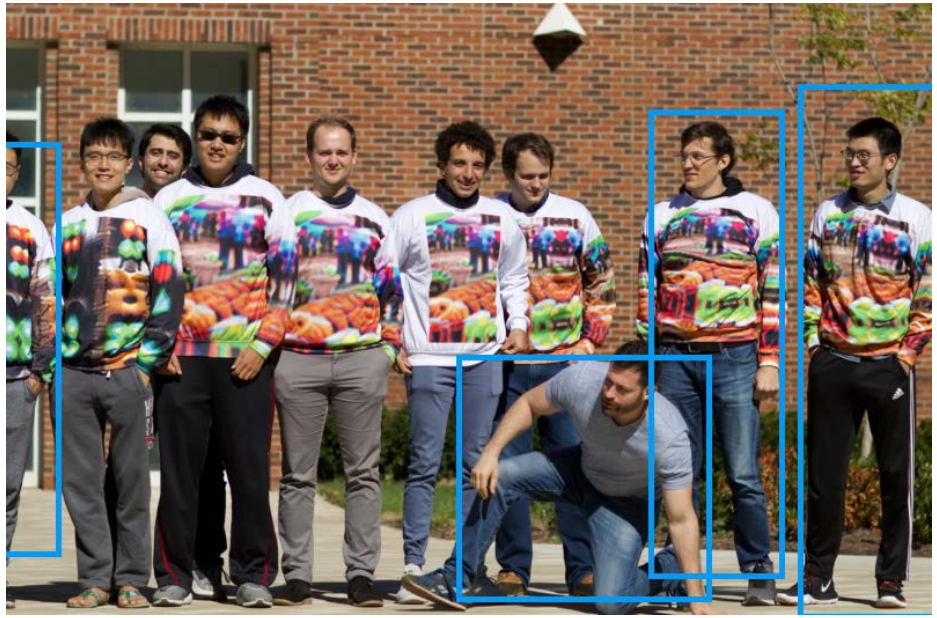


https://www.theverge.com/2019/4/23/18512472/fool-ai-surveillance-adversarial-example-yolov2-person-detection

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial Attacks

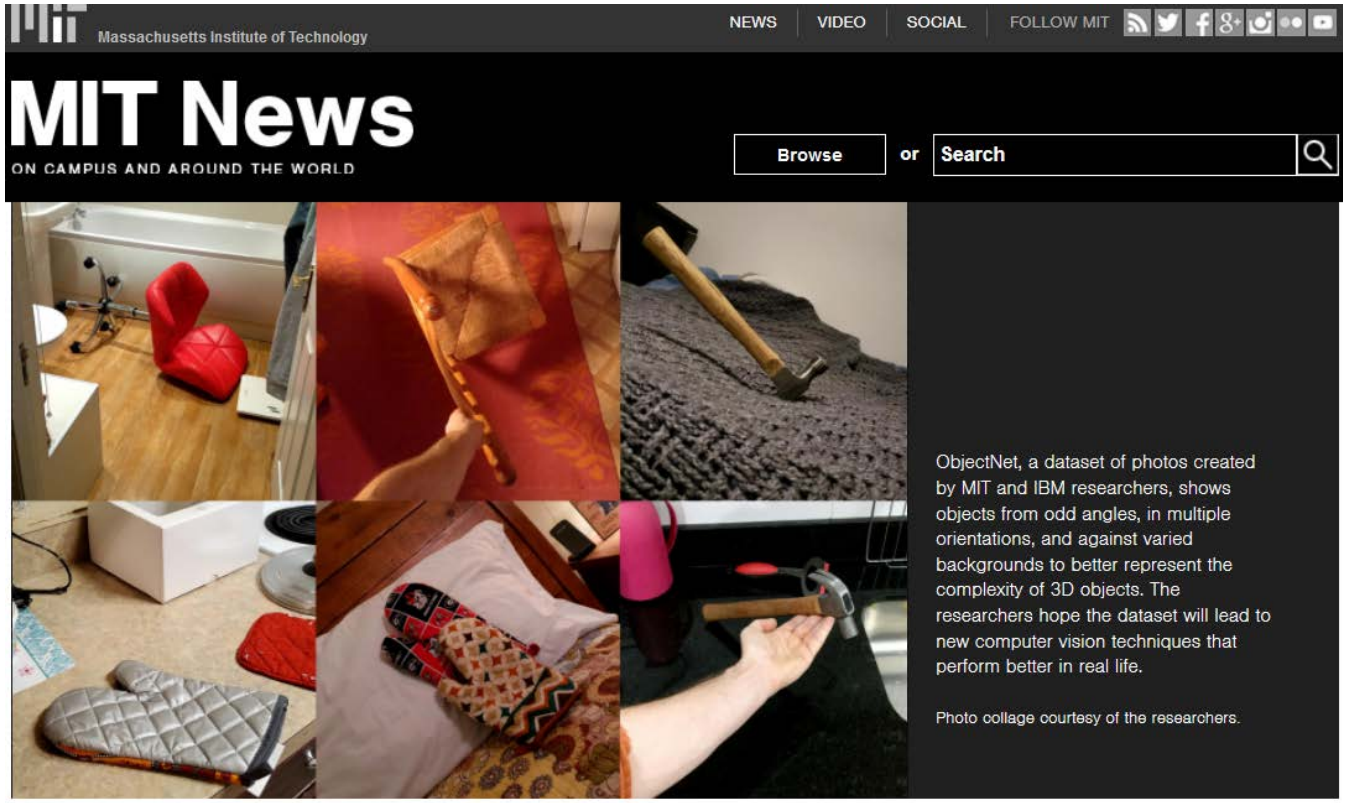


Tom Goldstein https://www.cs.umd.edu/~tomg/projects/invisible/

UNIVERSITÉ CÔTE D'AZUR

univ-cotedazur.fr

Adversarial Attacks



This object-recognition dataset stumped the world's best computer vision models

Objects are posed in varied positions and shot at odd angles to spur new AI techniques.

http://news.mit.edu/2019/object-recognition-dataset-stumped-worlds-best-computer-vision-models-1210

UNIVERSITÉ
CÔTE D'AZUR