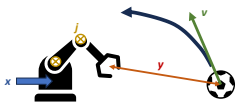
Mechatronic Engineering program

Basics of AI and Deep Learning: 11: Reinforcement learning and Generative AI

Ziemowit Dworakowski
AGH University of Krakow

1



Find signal x value that minimizes y distance in t seconds given current position of joints j

Assume we always start from the same j and v

Now we can try hundreds of different x prescriptions until we hit the „optimal" one…

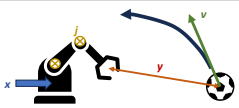
Now assume we always start from the same j but v changes…

Set of possible vs is potentially infinite, so we need to learn a **mapping** from v to good x.
It would be a regression task – if we had enough examples (we don't!).

So we need to – at the same time – optimize gradually x and learn relationship between x and v …

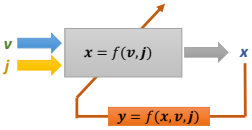
(And then if **j** changes as well or there are any additional uncertainties – the procedure is the same…)

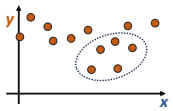
2

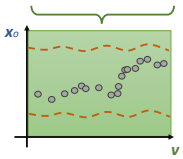


Find signal x value that minimizes y distance in t seconds given current position of joints j and speed v

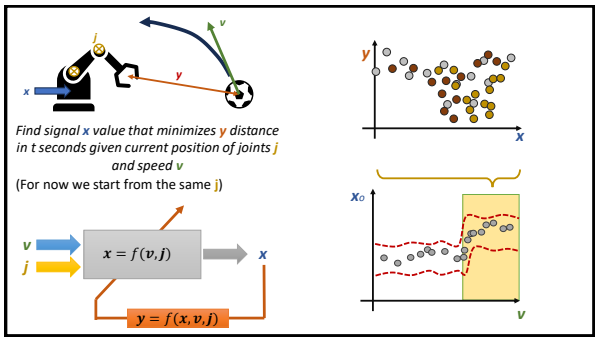
(For now we start from the same j)



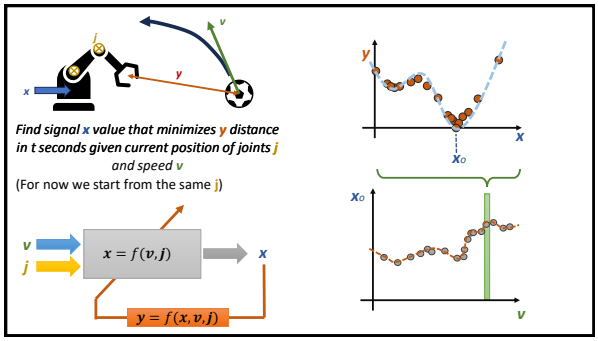




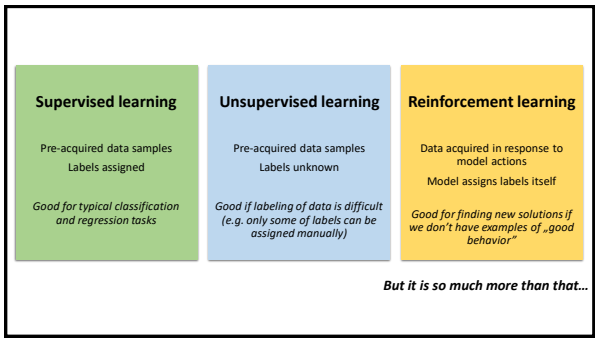
3


j
v
x
y
Find signal x value that minimizes y distance in t seconds given current position of joints j and speed v
(For now we start from the same j)
v
j
x = f(v, j)
x
y = f(x, v, j)
y
x
xO
v

4

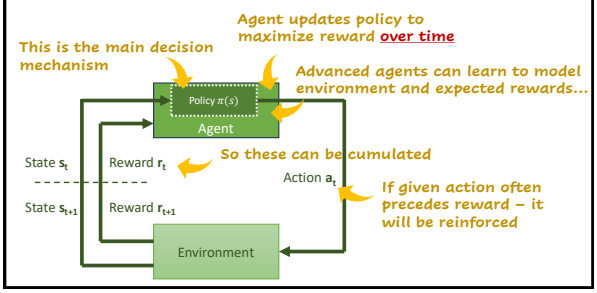

j
v
x
y
Find signal x value that minimizes y distance in t seconds given current position of joints j and speed v
(For now we start from the same j)
v
j
x = f(v, j)
x
y = f(x, v, j)
y
x
xO
xO
v

5


Supervised learning
Pre-acquired data samples
Labels assigned
Good for typical classification and regression tasks
Unsupervised learning
Pre-acquired data samples
Labels unknown
Good if labeling of data is difficult (e.g. only some of labels can be assigned manually)
Reinforcement learning
Data acquired in response to model actions
Model assigns labels itself
Good for finding new solutions if we don't have examples of „good behavior"
But it is so much more than that…

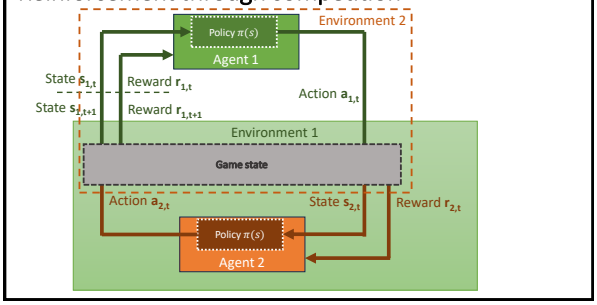
6

General representation of RL

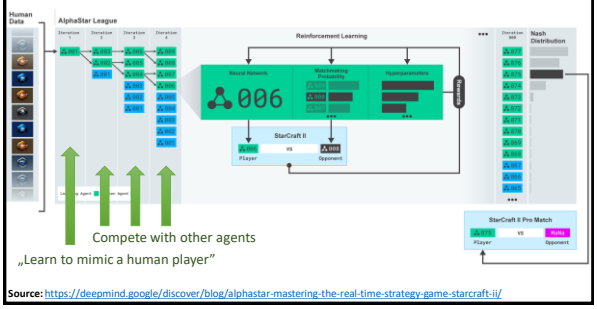


7

Reinforcement through competition

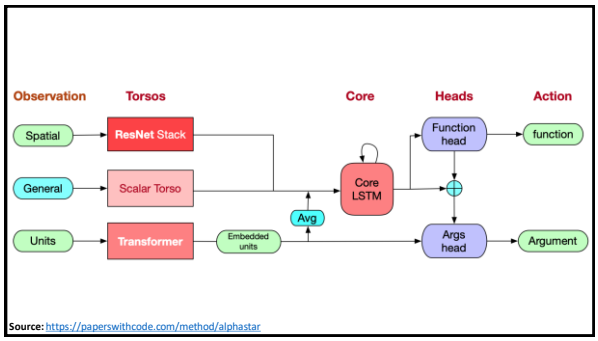


8



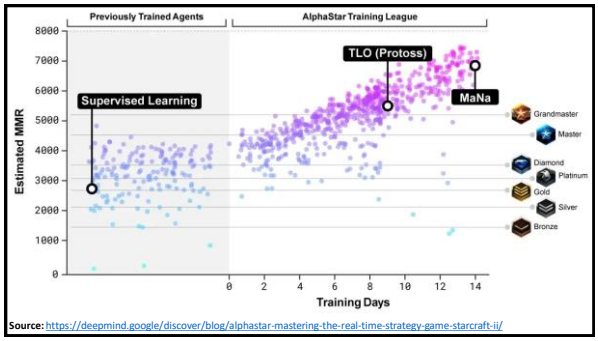
Source: https://deepmind.google/discover/blog/alphastar-mastering-the-real-time-strategy-game-starcraft-ii/

9



Source: https://paperswithcode.com/method/alphastar

10



Source: https://deepmind.google/discover/blog/alphastar-mastering-the-real-time-strategy-game-starcraft-ii/

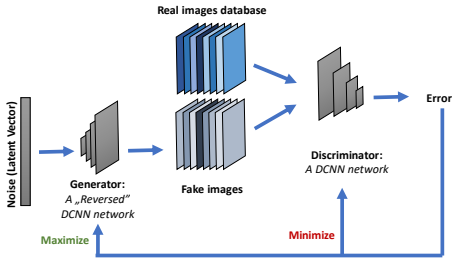
11

What if we force agents to fight over image?

Generative-Adversarial mechanism

12

Generative Adversarial Network (GAN, 2014)

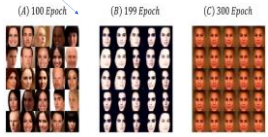


13

GAN problems

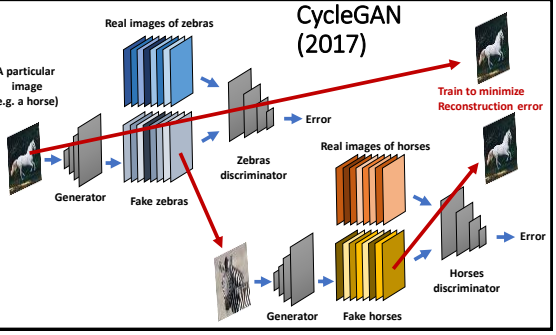
We have a nice tool that generates images from given distribution, but:

- **We can observe mode collapse**
  Generator Learns a good prescription to fool a discriminator and „decides" to use it always
- **We can't control the output at all**
- **We can only generate very small images or training gets too hard**

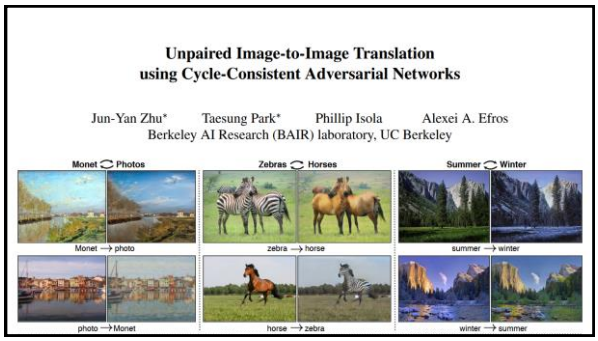


14

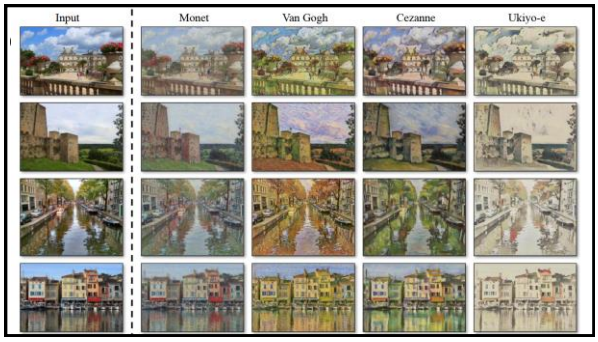
CycleGAN (2017)



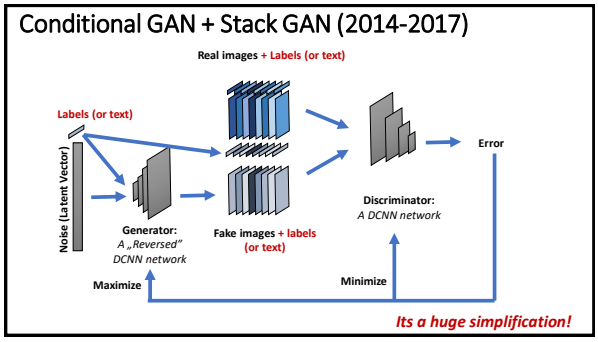
15


Unpaired Image-to-Image Translation
using Cycle-Consistent Adversarial Networks
Jun-Yan Zhu* Taesung Park* Phillip Isola Alexei A. Efros
Berkeley AI Research (BAIR) laboratory, UC Berkeley
Monet ⇄ Photos
Zebras ⇄ Horses
Summer ⇄ Winter
Monet → photo
zebra → horse
summer → winter
photo → Monet
horse → zebra
winter → summer

16

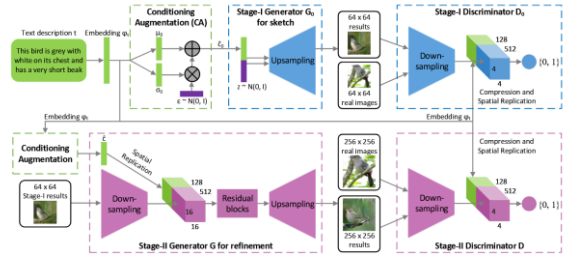

Input
Monet
Van Gogh
Cezanne
Ukiyo-e

17


Conditional GAN + Stack GAN (2014-2017)
Real images + Labels (or text)
Labels (or text)
Noise (Latent Vector)
Generator:
A „Reversed"
DCNN network
Fake images + labels
(or text)
Discriminator:
A DCNN network
Error
Maximize
Minimize
Its a huge simplification!

18

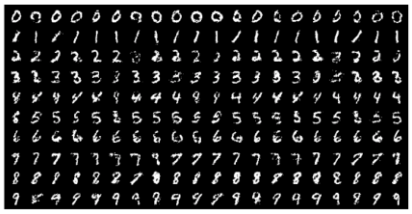
Stack GAN (2017)



Source: https://arxiv.org/pdf/1612.03242

19

Conditional GAN (2014)



Source: https://arxiv.org/pdf/1411.1784

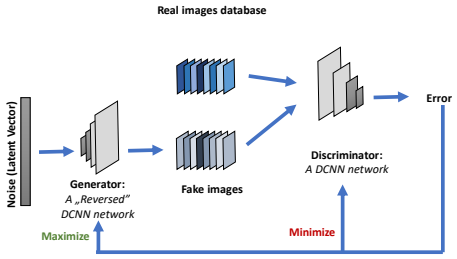
20

Stack GAN (2017)

Source: https://arxiv.org/pdf/1612.03242

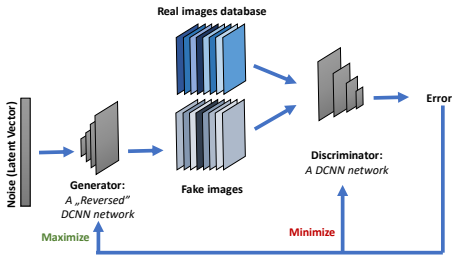
21

Progressive GAN



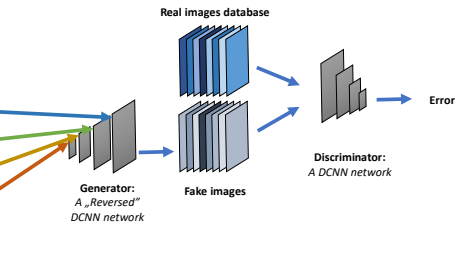
22

Progressive GAN – trains layers gradually

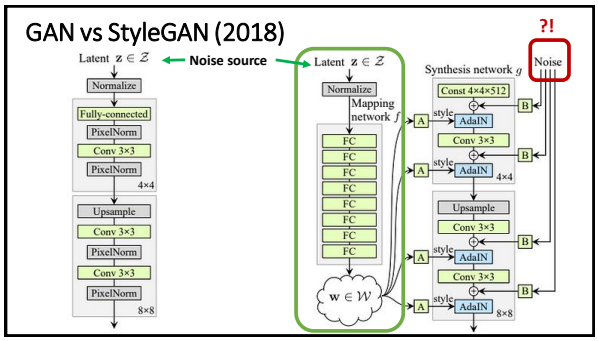


23

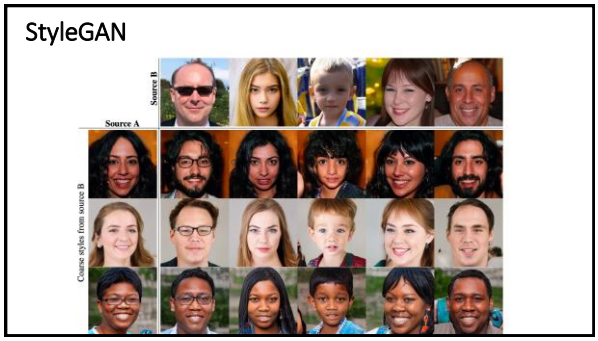
StyleGAN (2018)



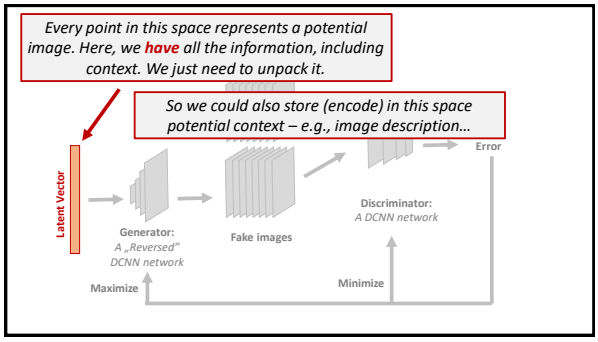
24


GAN vs StyleGAN (2018)
?!
Latent $z \in \mathcal{Z}$
Noise source
Normalize
Fully-connected
PixelNorm
Conv 3×3
PixelNorm
4×4
Upsample
Conv 3×3
PixelNorm
Conv 3×3
PixelNorm
8×8
Latent $z \in \mathcal{Z}$
Normalize
Mapping network f
FC
FC
FC
FC
FC
FC
FC
FC
$w \in \mathcal{W}$
Synthesis network g
Noise
Const 4×4×512
A
B
style
AdaIN
Conv 3×3
4×4
Upsample
8×8

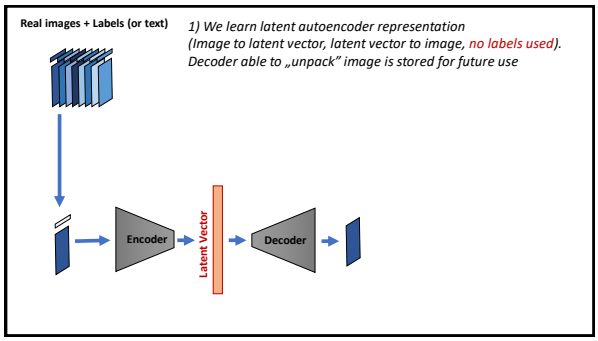
25


StyleGAN
Source B
Source A
Coarse styles from source B

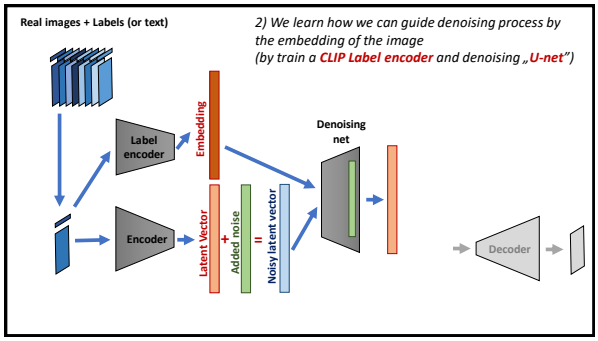
26


Every point in this space represents a potential image. Here, we have all the information, including context. We just need to unpack it.
So we could also store (encode) in this space potential context – e.g., image description…
Latent Vector
Generator:
A „Reversed" DCNN network
Fake images
Discriminator:
A DCNN network
Error
Maximize
Minimize

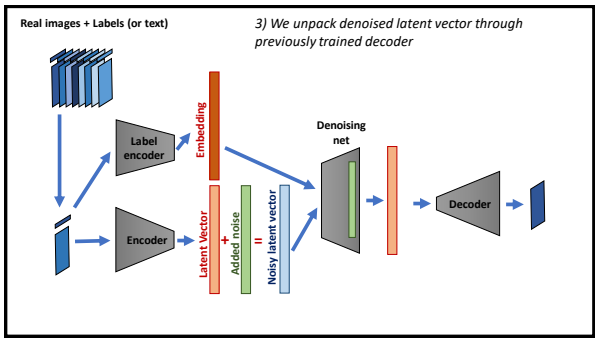
27


Real images + Labels (or text)
1) We learn latent autoencoder representation
(Image to latent vector, latent vector to image, no labels used).
Decoder able to „unpack" image is stored for future use
Encoder
Latent Vector
Decoder

28

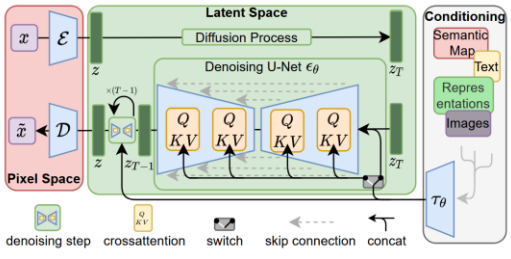

Real images + Labels (or text)
2) We learn how we can guide denoising process by
the embedding of the image
(by train a CLIP Label encoder and denoising „U-net")
Label encoder
Embedding
Denoising net
Encoder
Latent Vector
+
Added noise
=
Noisy latent vector
Decoder

29


Real images + Labels (or text)
3) We unpack denoised latent vector through
previously trained decoder
Label encoder
Embedding
Denoising net
Encoder
Latent Vector
+
Added noise
=
Noisy latent vector
Decoder

30

Stable Diffusion (2022)



31

32

Cycle GAN (2017)
Allows for consistent cross-domain mapping (and back)

Conditional GAN (2017)
Allows contextual input (i.e., generation consistent withh image label)

Progressive GAN (2017)
Allows gradual training from small to high resolution

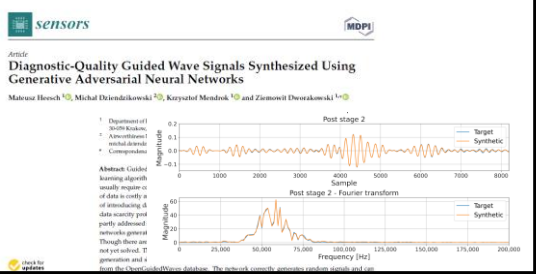
Style GAN (2018)
Allows controlled generation at different „image layers"

Stable Diffusion (2022)
Allows controlled and layered image generation without unstable adversarial loss

I may have simplified a bit and ommitted a few steps along the way though…

33

What do we do at AGH?


sensors

MDPI

Article

Diagnostic-Quality Guided Wave Signals Synthesized Using Generative Adversarial Neural Networks

Mateusz Heesch, Michał Dziendzikowski, Krzysztof Mendrok and Ziemowit Dworakowski

Post stage 2

Target
Synthetic

Magnitude

Sample

Post stage 2 - Fourier transform

Target
Synthetic

Magnitude

Frequency [Hz]

34

What do we do at AGH?

Ultrasonic guided-wave data

Context
(specimen shape, damage locations)

Style GAN
(or)
Cycle GAN
(or)
Stable-diffusion-based model

Plausible data for decision system training

Condition monitoring data

Context
(machine description, damage type)

Style GAN
(or)
Cycle GAN
(or)
Stable-diffusion-based model

Plausible data for decision system training

35

What do we do at AGH?

We see this:

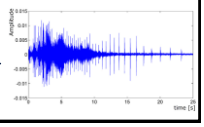
„Imagine a dark red and green robot with KRiM text, photorealistic, steampunk, (...)"

Midjourney

We want this:

Imagine a machine with [this specification] running in [this conditions]. What would the signals look like?

„Midjourney for SHM"



36

Things to remember:

1. Explain Reinforcement Learning scheme
2. Compare reinforcement learning with basic optimization and regression
3. Show a scheme of a GAN network
4. Explain drawbacks of basic GAN architecture
5. Explain how CycleGAN, StyleGAN, ConditionalGAN and ProgressiveGAN contribute to GAN-based image generation
6. Explain what is Latent Space and what is a Latent Vector
7. Explain the idea behind Stable Diffusion model

37