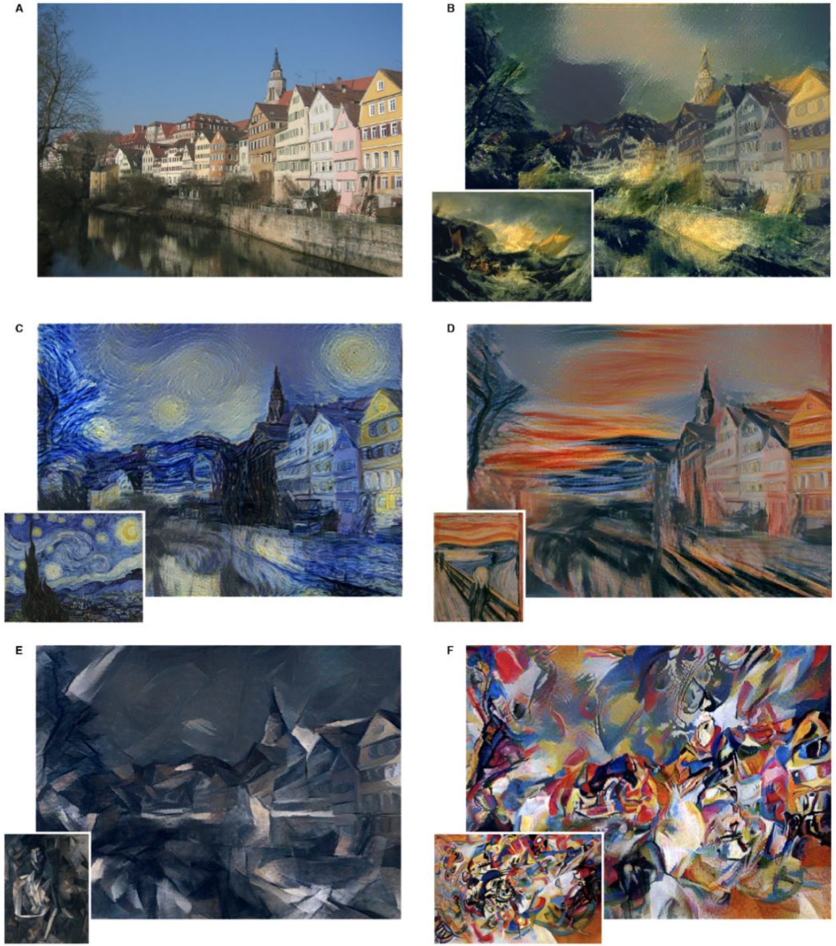
Overview

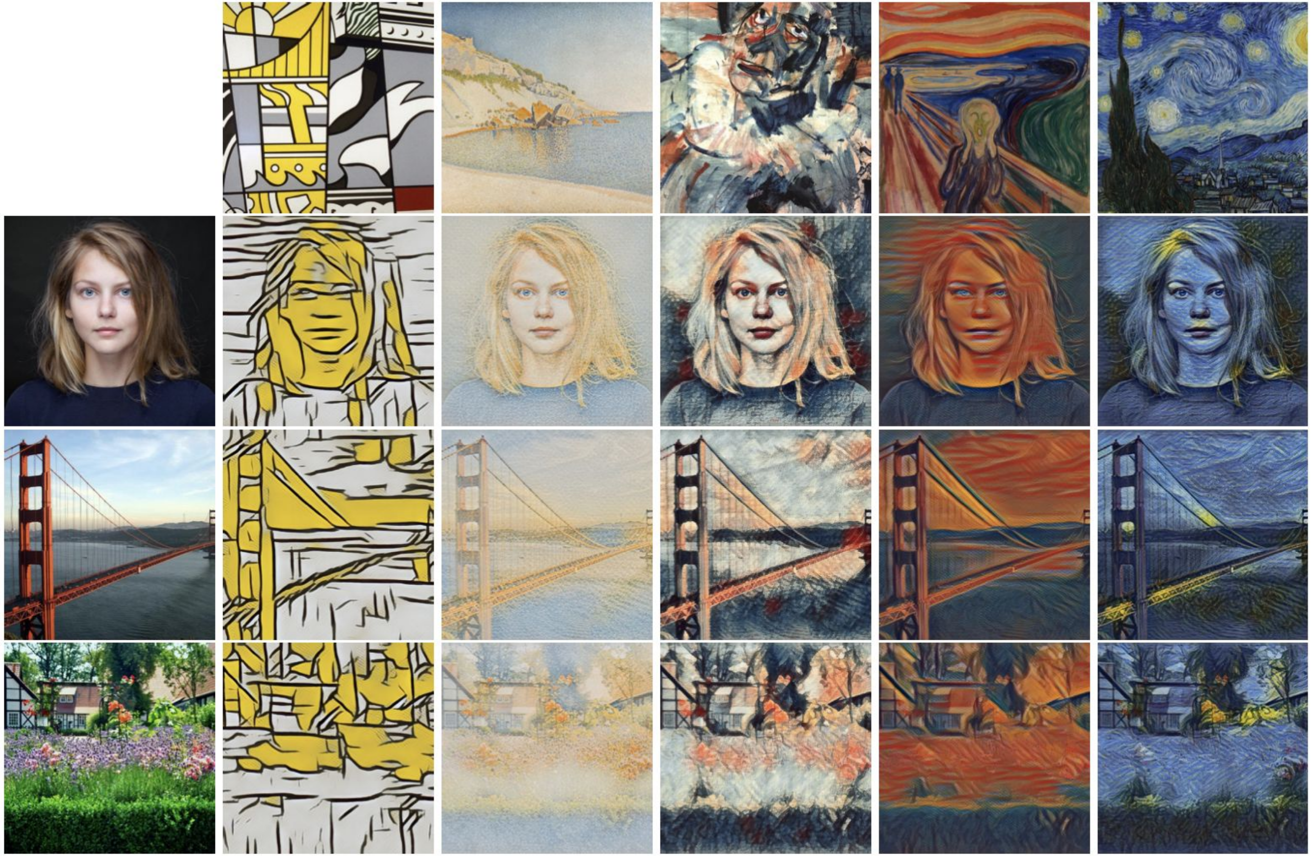
- Style Transfer
- GANs
- Reinforcement Learning
- Other Tools

Neural Style Transfer (NST)

- Transfer the style of an image/video to another image/video.
- Simple idea, very fun!



A Neural Algorithm of Artistic Style
by L. A. Gatys, A.S. Ecker, M. Bethge



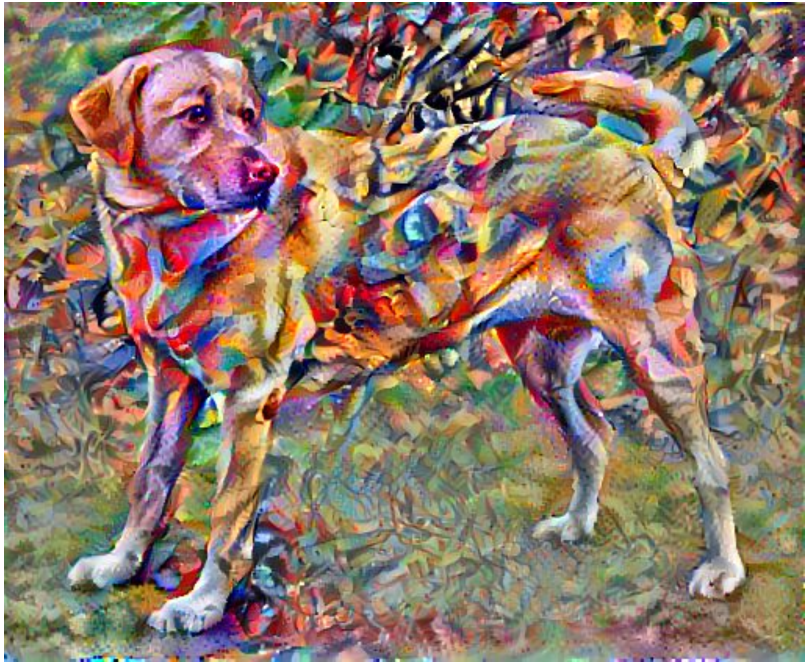
A Learned Representation for Artistic Style
by L. A. Gatys, A.S. Ecker, M. Bethge

Neural Style Transfer (NST)

Content Image

Style Image

Neural Style Transfer (NST)



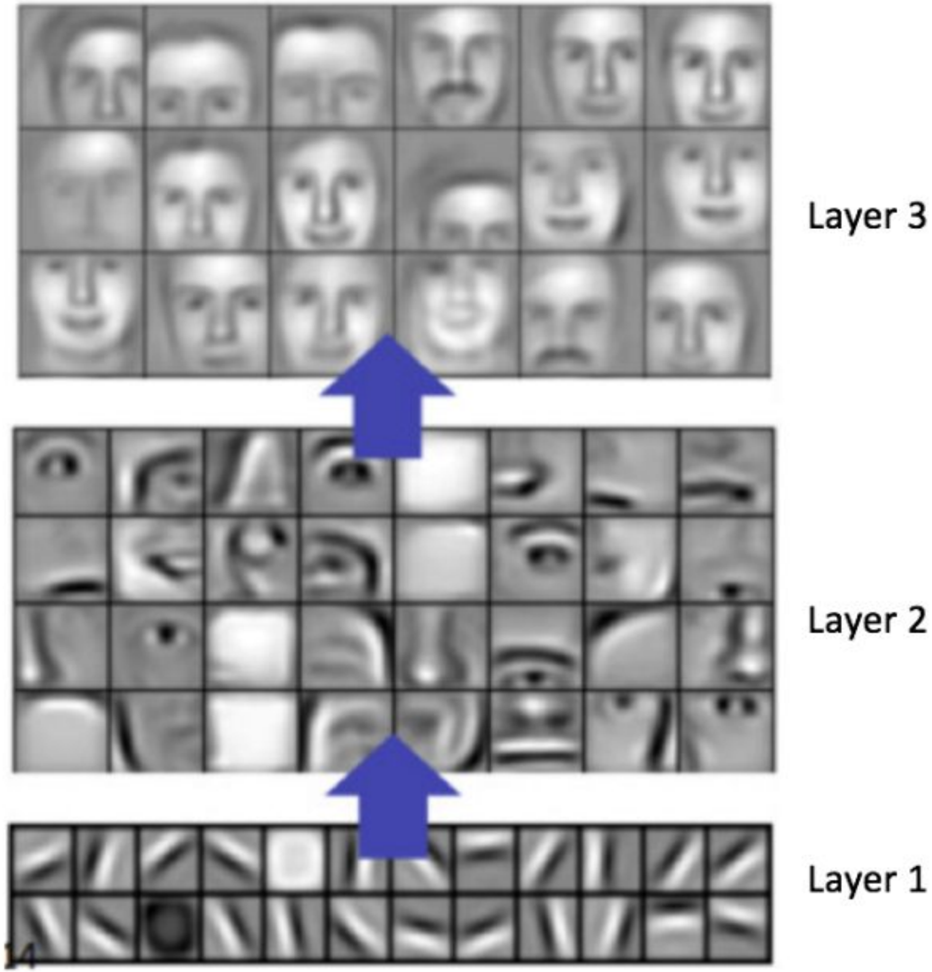
Combined Content/Style Image

Neural Style Transfer (NST)

- What is the “content” of an image?

Neural Style Transfer (NST)

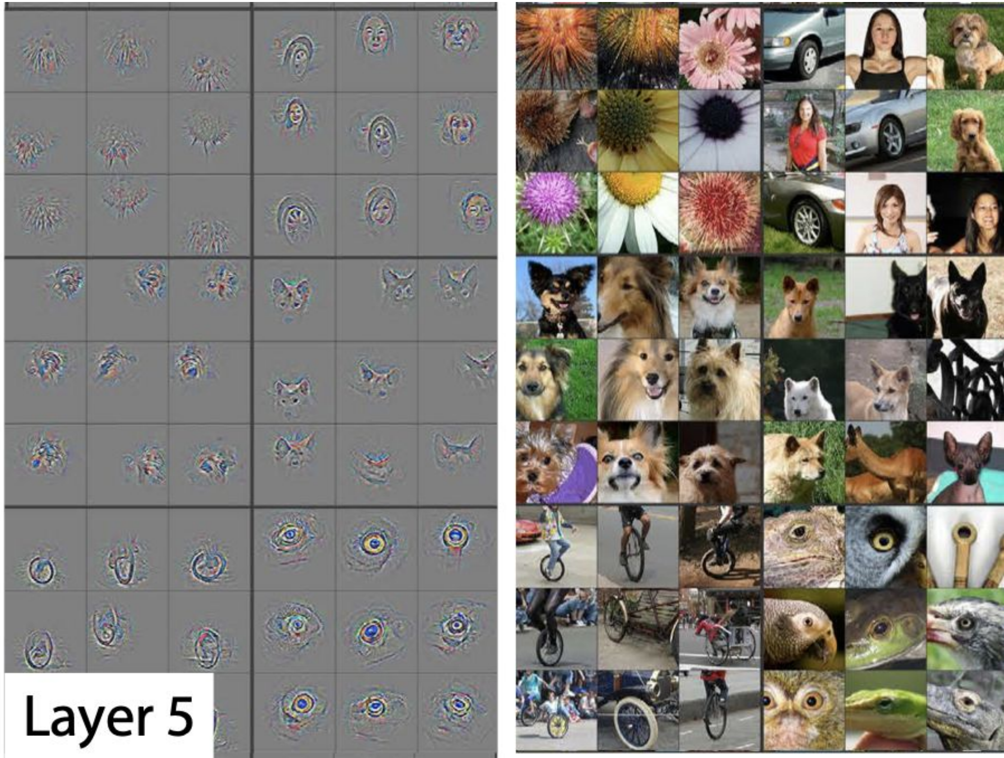
- What is the “content” of an image?
- Earlier layers -> “lower-level” features
- Later layers -> “higher-level” features



Convolutional Deep Belief Networks for Scalable Unsupervised Laerning of Hierarchical Representations, Lee H., Grosse R., Ranganath R., Ng A.

Neural Style Transfer (NST)

- What is the “content” of an image?
- Earlier layers -> “lower-level” features
- Later layers -> “higher-level” features

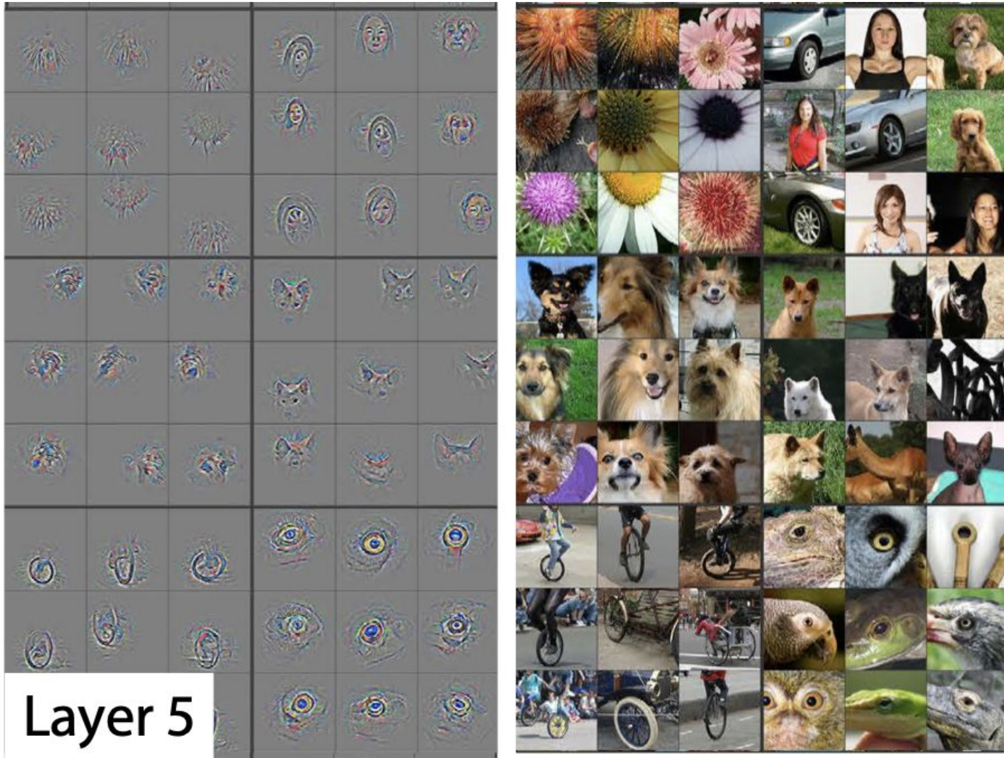


Visualizing and Understanding Convolutional Networks
by M. D. Zeiler, R. Fergus

Neural Style Transfer (NST)

- What is the “content” of an image?
- Earlier layers -> “lower-level” features
- Later layers -> “higher-level” features
- The “content” of an image can loosely be interpreted as the feature output by a later layer of some pre-trained CNN

Visualizing and Understanding Convolutional Networks
by M. D. Zeiler, R. Fergus



Neural Style Transfer (NST)

- What is the “style” of an image?
- Earlier layers -> “lower-level” features
- Later layers -> “higher-level” features

Visualizing and Understanding Convolutional Networks
by M. D. Zeiler, R. Fergus

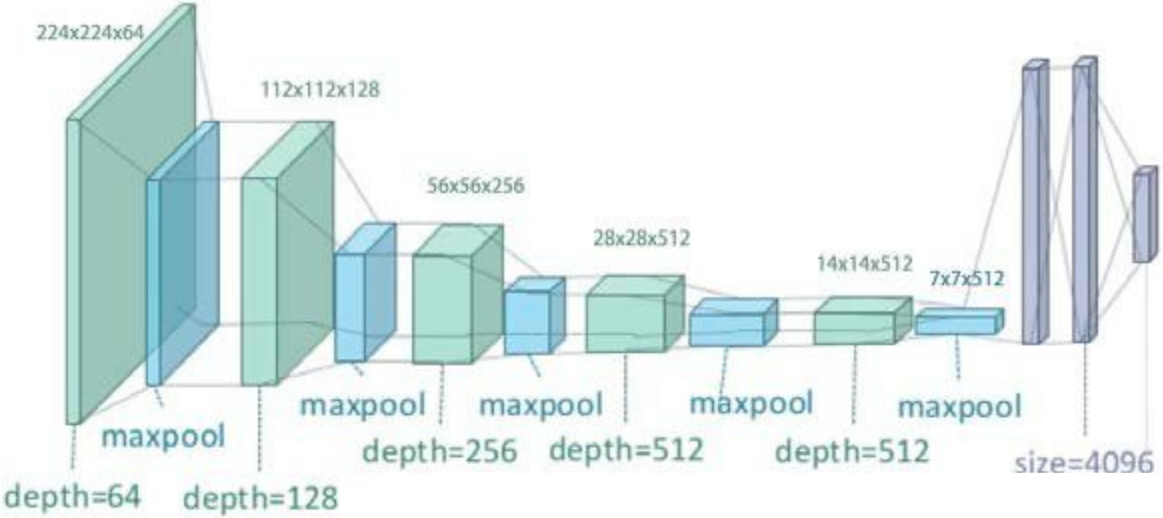
Neural Style Transfer (NST)

- What is the “style” of an image?
- Earlier layers -> “lower-level” features
- Later layers -> “higher-level” features
- The “style” of an image can loosely be interpreted as the distribution of features output by a early layer of some pre-trained CNN

Visualizing and Understanding Convolutional Networks
by M. D. Zeiler, R. Fergus

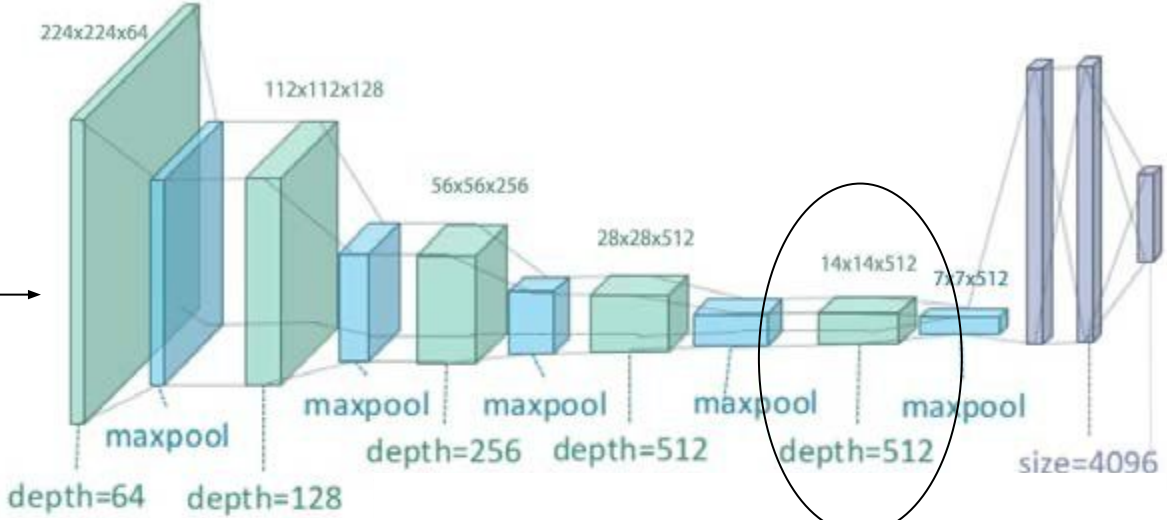
Layer 1

Neural Style Transfer (NST)



Pre-trained VGG-19

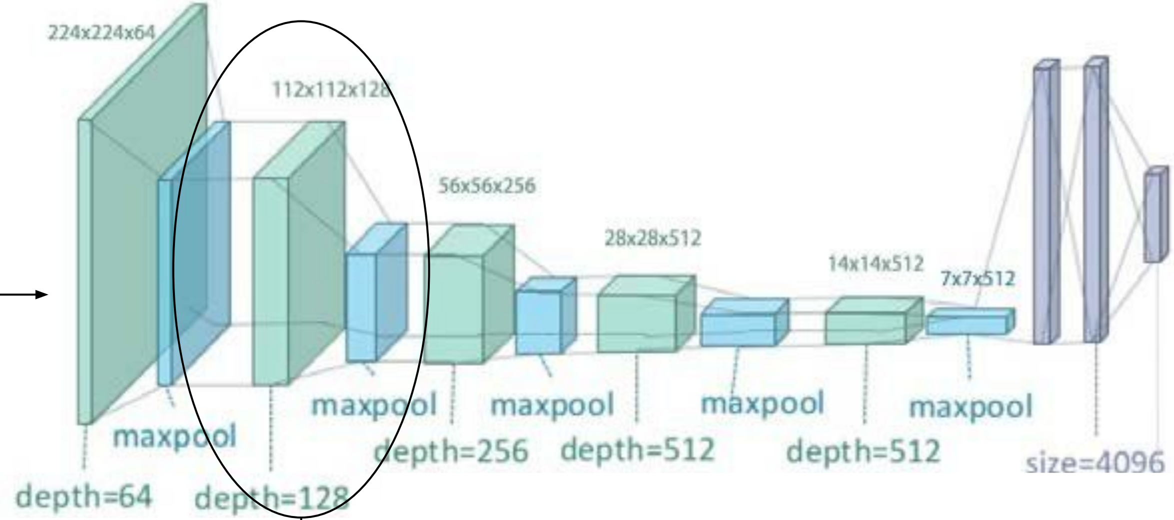
Neural Style Transfer (NST)



Pre-trained VGG-19

Content

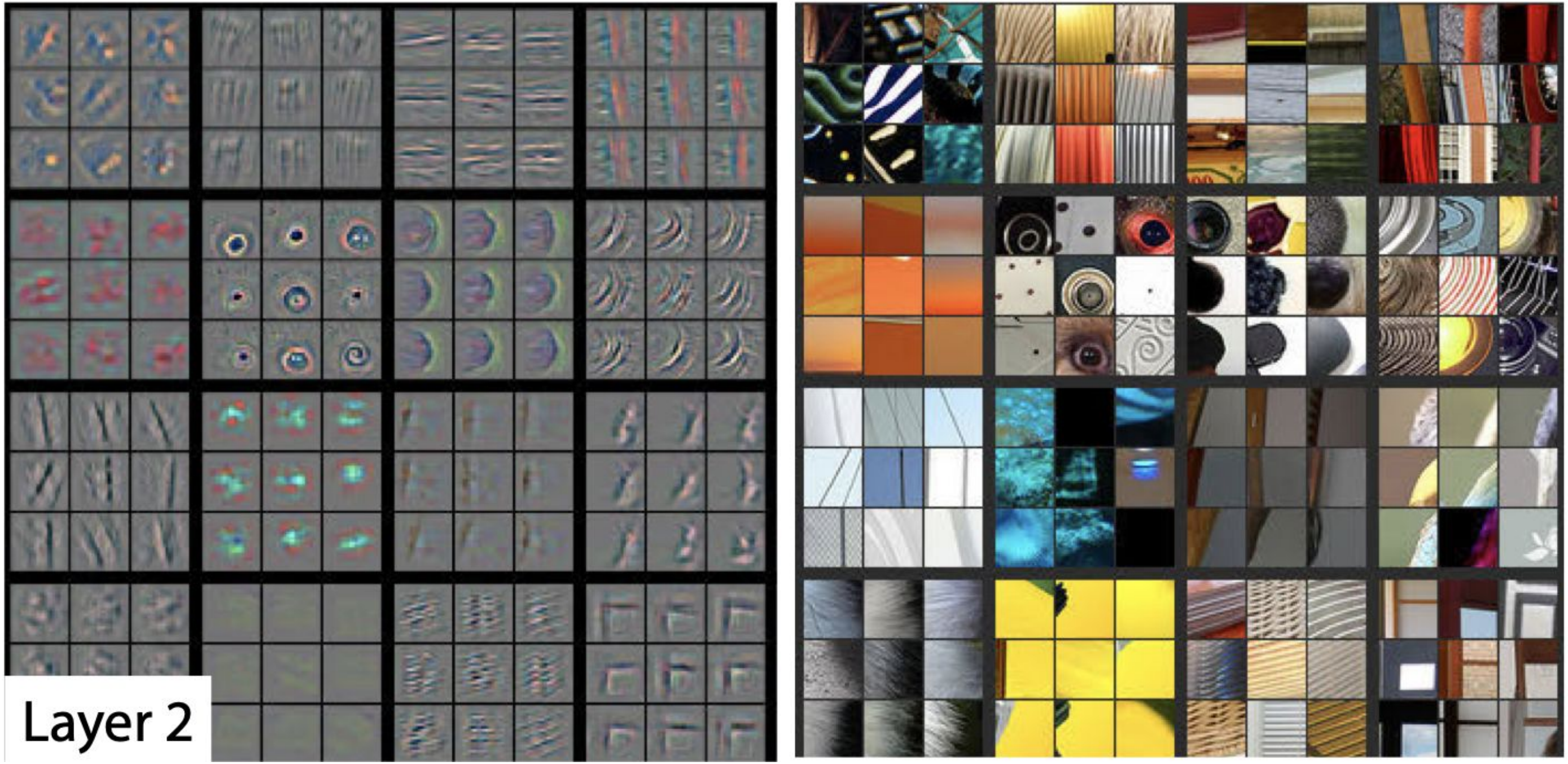
Neural Style Transfer (NST)



Pre-trained VGG-19

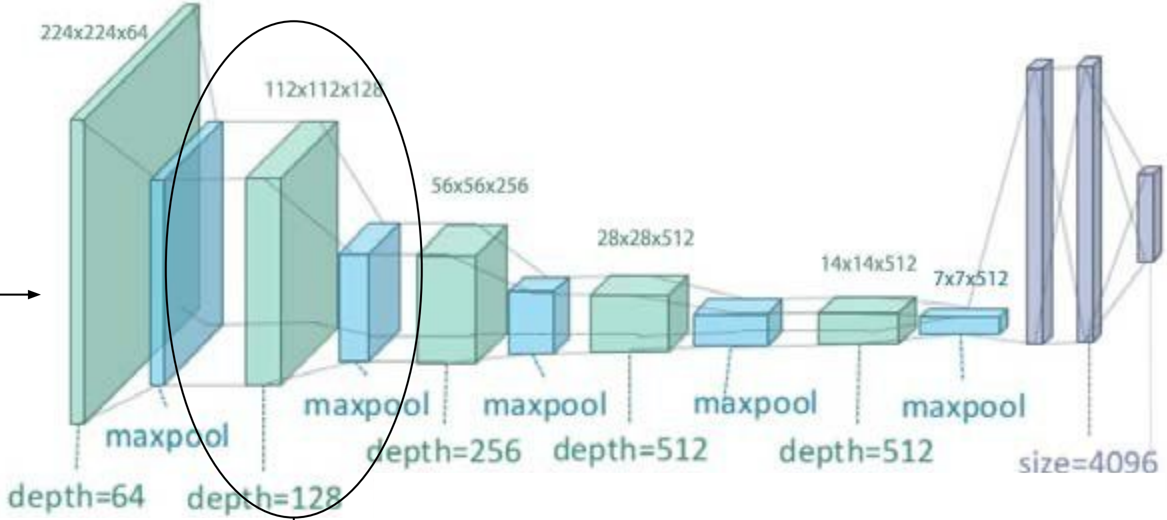
Style contained here

Neural Style Transfer (NST)



Neural Style Transfer (NST)

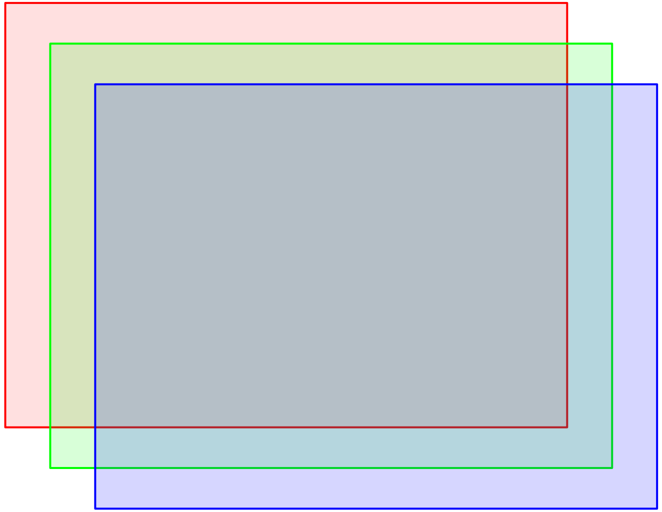
Neural Style Transfer (NST)



Pre-trained VGG-19

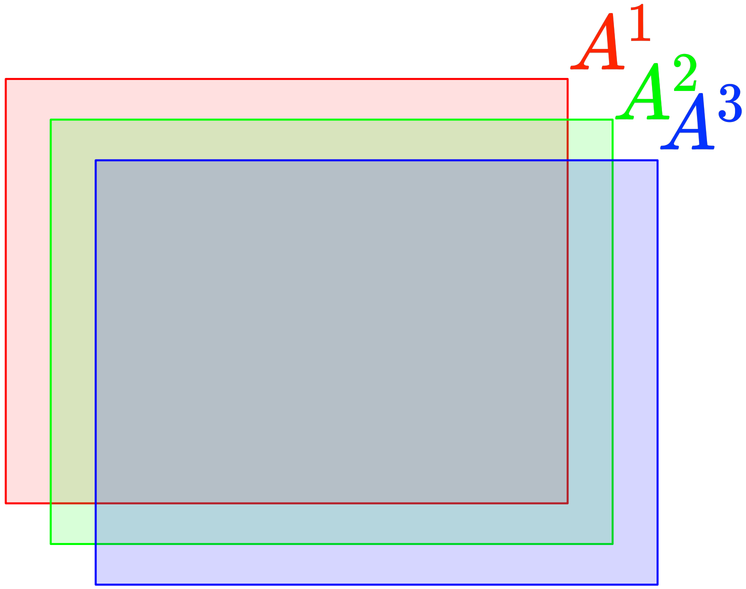
Style is
how often the features here
appear and how they are
correlated with each other

Neural Style Transfer (NST)



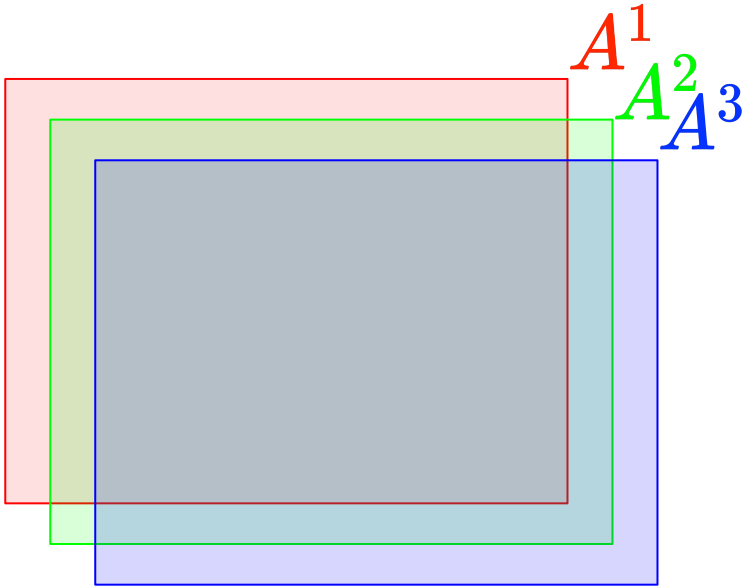
Feature channels

Neural Style Transfer (NST)



Feature channels

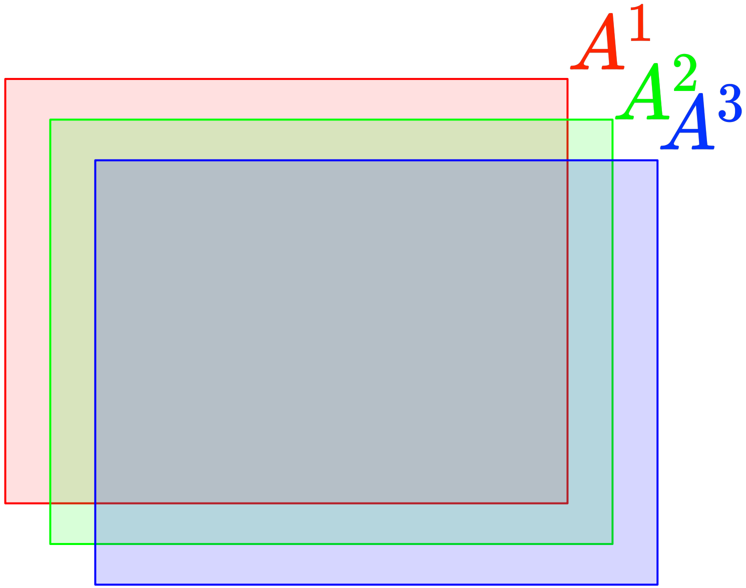
Neural Style Transfer (NST)



Feature channels

$$g_{12} = \sum_{i,j} a^1_{ij} a^2_{ij}$$

Neural Style Transfer (NST)

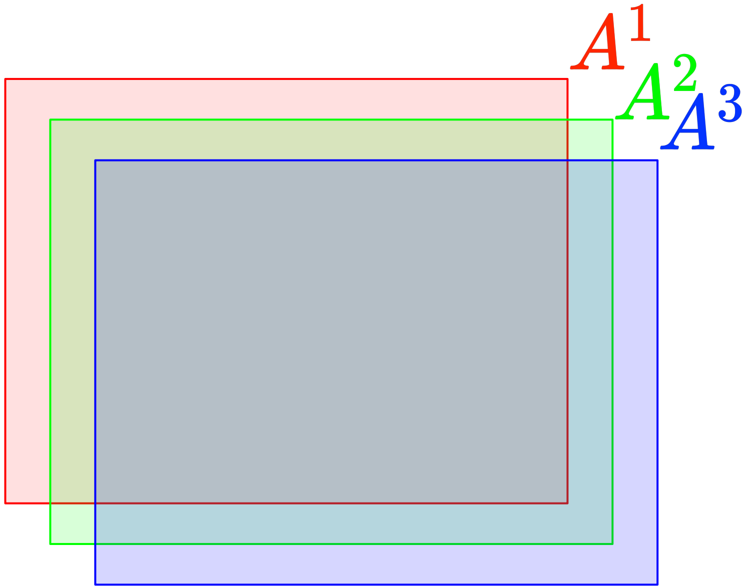


Feature channels

$$g_{12} = \sum_{i,j} a^1_{ij} a^2_{ij}$$

$$g_{13} = \sum_{i,j} a^1_{ij} a^3_{ij}$$

Neural Style Transfer (NST)

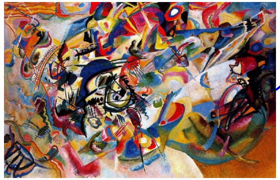


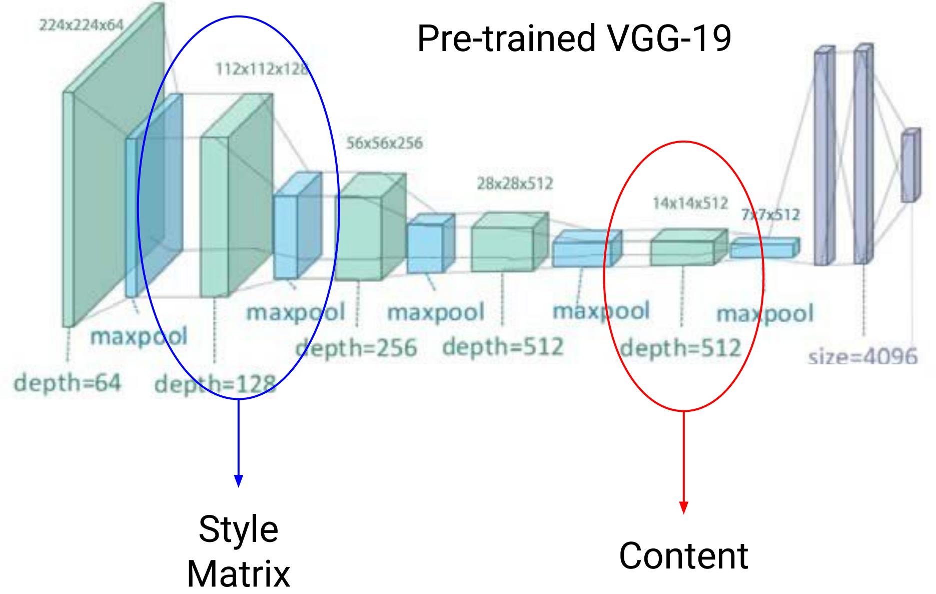
Feature channels

$$G = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix}$$

“Style Matrix”

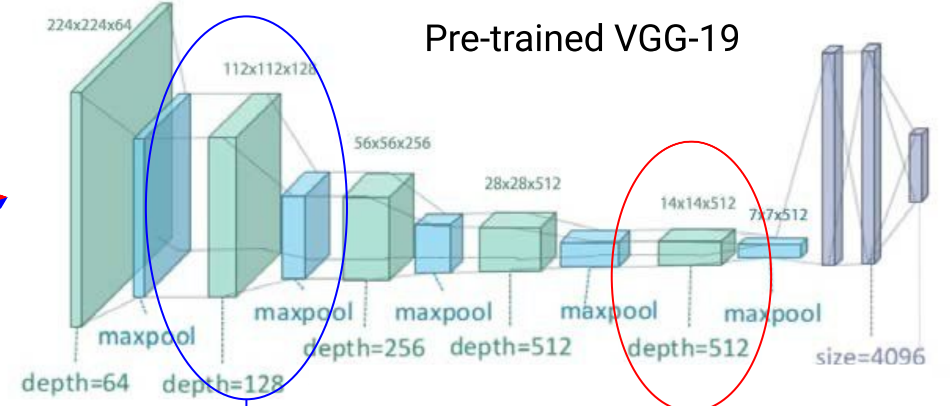
Neural Style Transfer (NST)





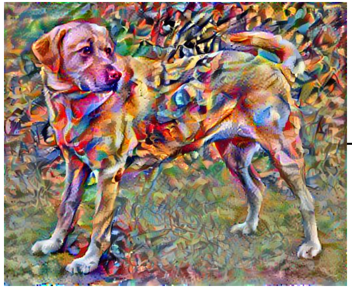
Neural Style Transfer (NST)

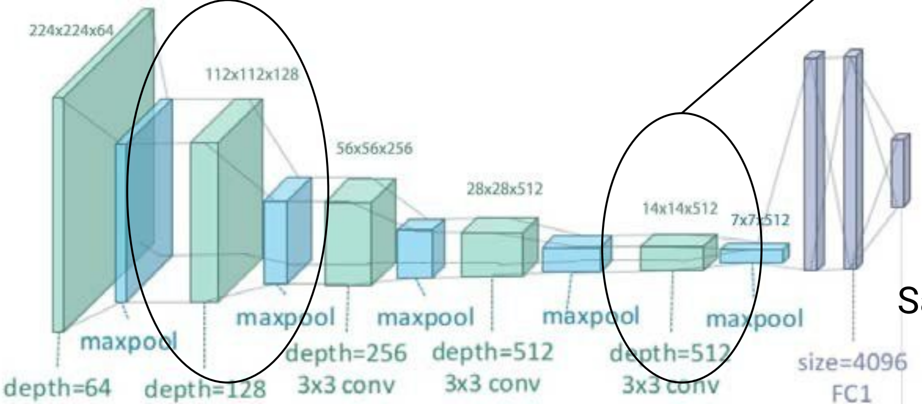
Pre-trained VGG-19



Style
Matrix

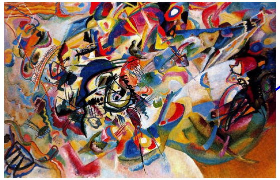
Content

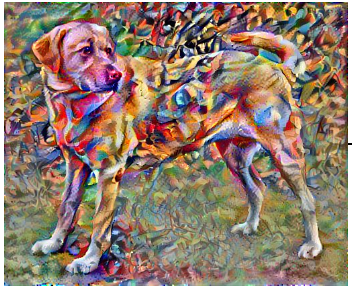


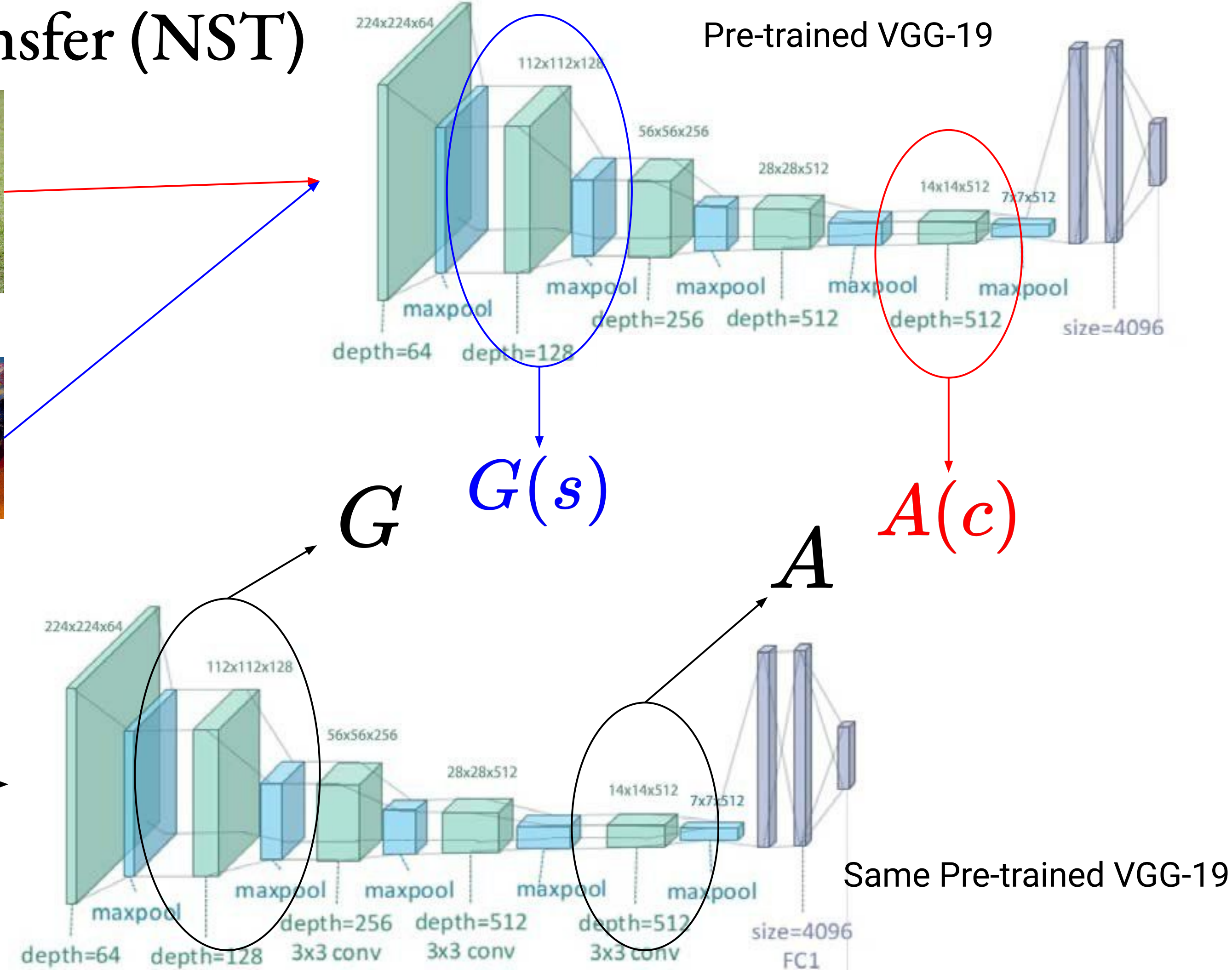


Same Pre-trained VGG-19

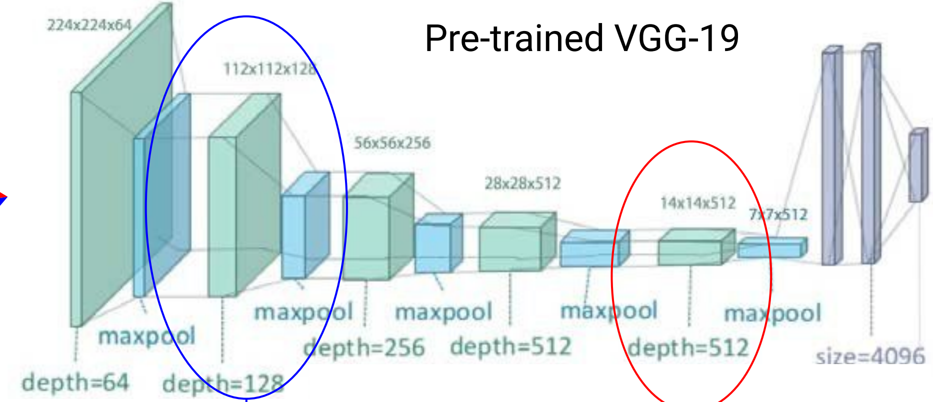
Neural Style Transfer (NST)







Neural Style Transfer (NST)

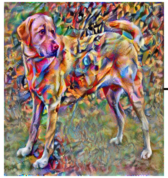


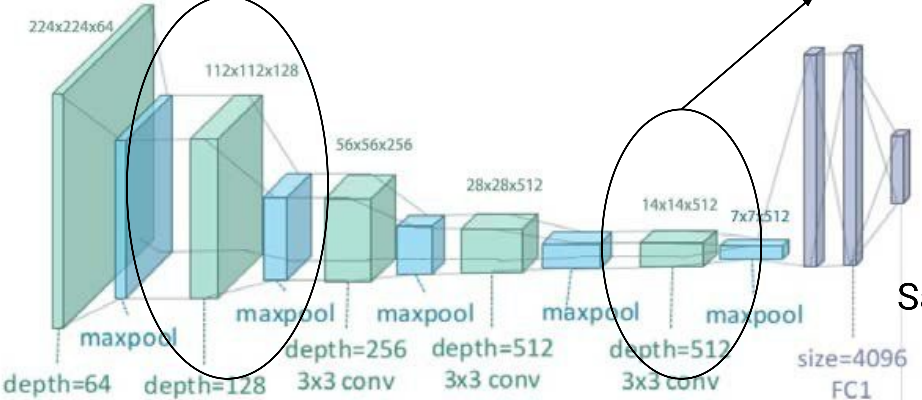
$G(s)$ $A(c)$

$$\mathcal{L}_{style} = \sqrt{\sum_{i,j}(g_{i,j} - g(s)_{i,j})^2}$$

$$\mathcal{L}_{content} = \sqrt{\sum_{i,j}(a_{i,j} - a(c)_{i,j})^2}$$

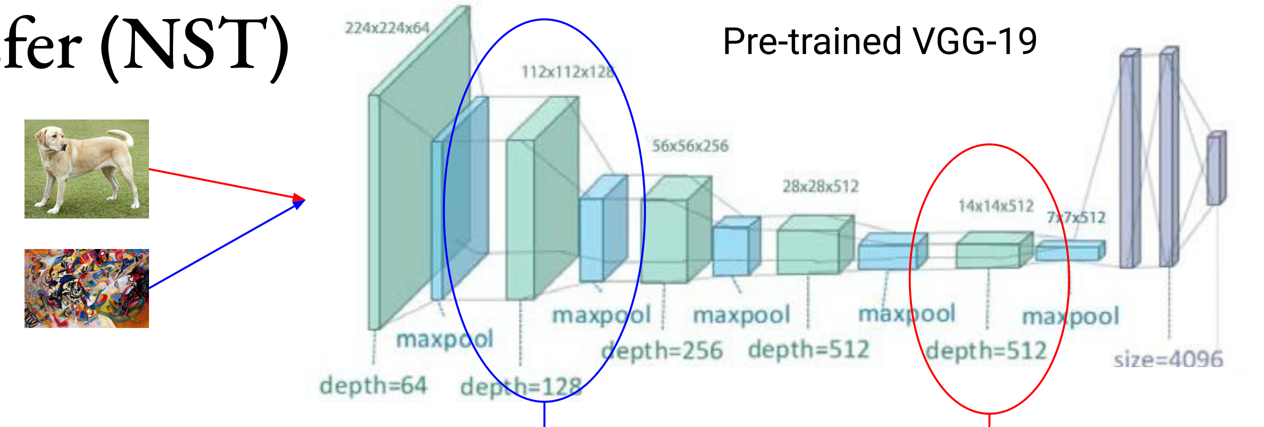
G A



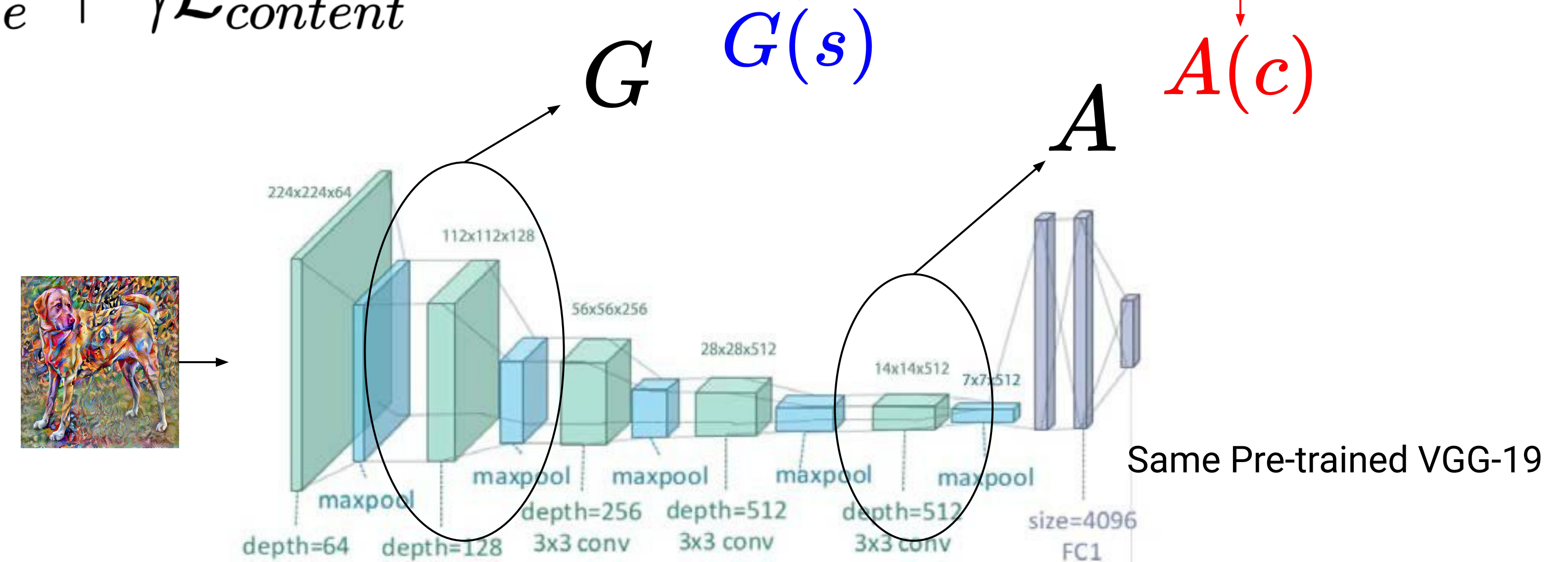


Same Pre-trained VGG-19

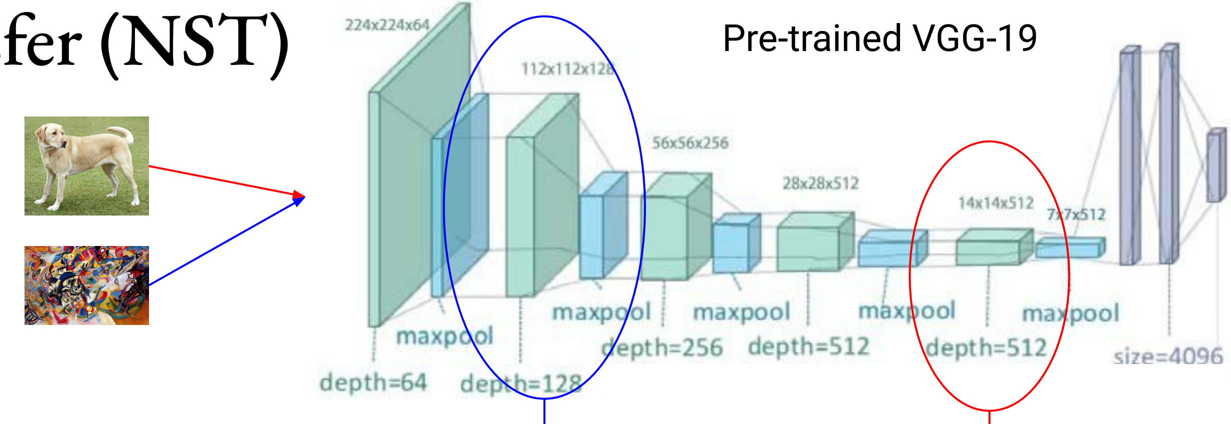
Neural Style Transfer (NST)



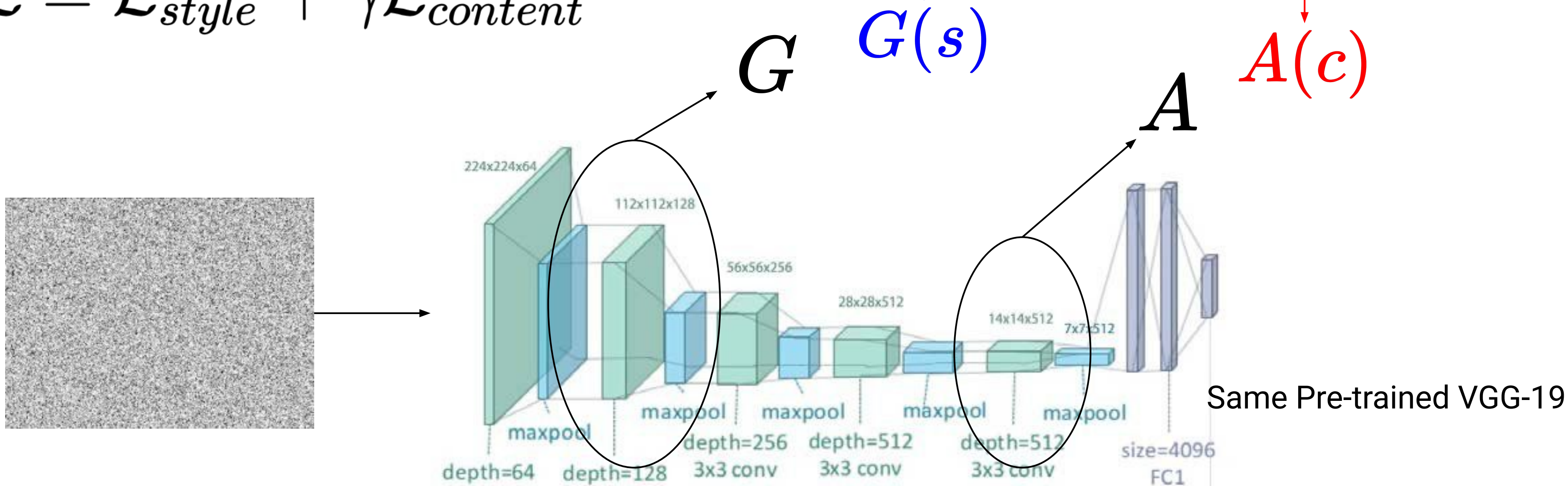
$$\mathcal{L} = \mathcal{L}_{style} + \gamma \mathcal{L}_{content}$$



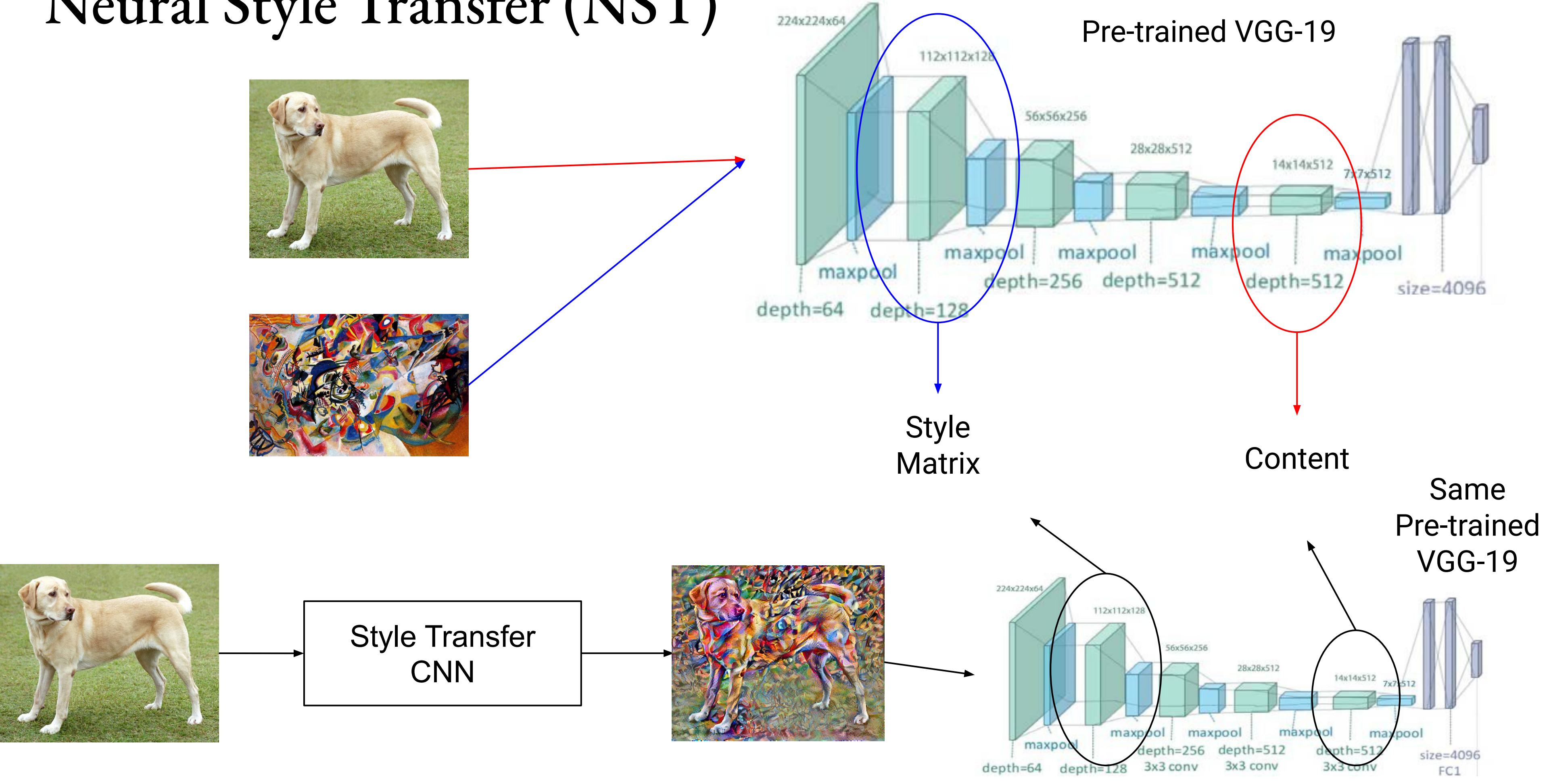
Neural Style Transfer (NST)



$$\mathcal{L} = \mathcal{L}_{style} + \gamma \mathcal{L}_{content}$$



Neural Style Transfer (NST)



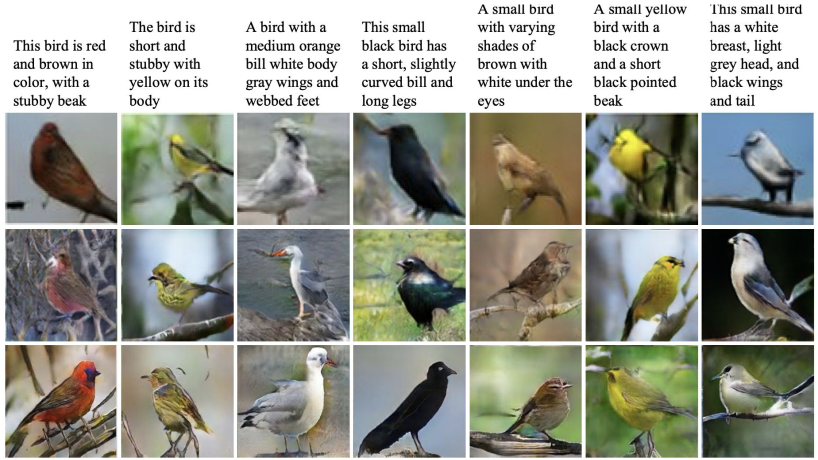
Neural Style Transfer (NST)

- Transfer style to a content image using a pre-trained network to define style and content
    - Perform gradient descent on an image to match style/content
    - Train a style transfer network
- Multiple Style Transfer Networks
- Text Style Transfer
    - Come and sit
    - Please take a seat

GANs

- Generative Adversarial Networks

https://thispersondoesnotexist.com



StackGAN: Text to Photo Realistic Image Synthesis with Stacked Generative Adversarial Networks by H. Zhang, et. al.

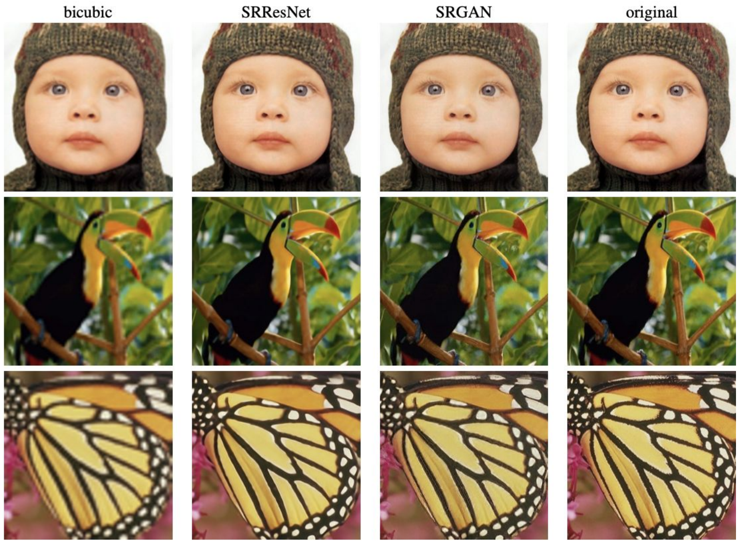
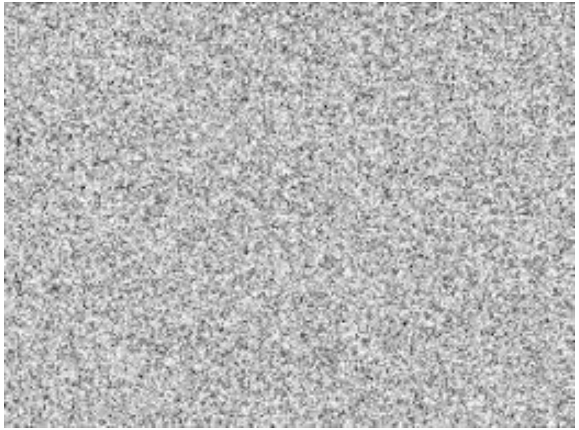


Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network
by C. Ledig, et. al.

Large Scale GAN Training for High Fidelity Natural Image Synthesis, Brock A., Donahue J., Simonyan S., 2019

GANs

- To generate fake data: recreate distribution of data

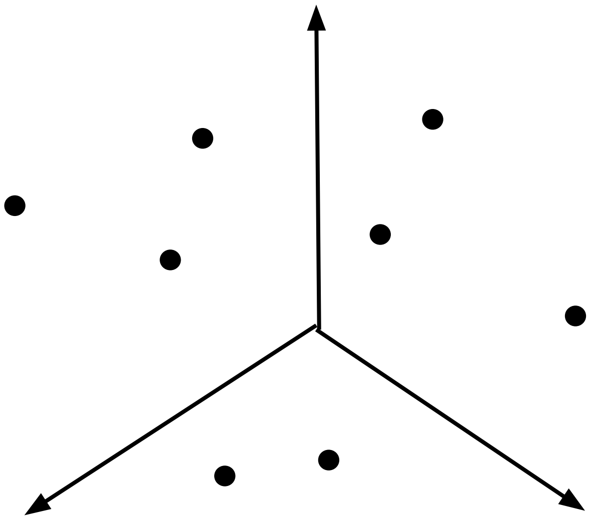


Random data point in 512^2 space

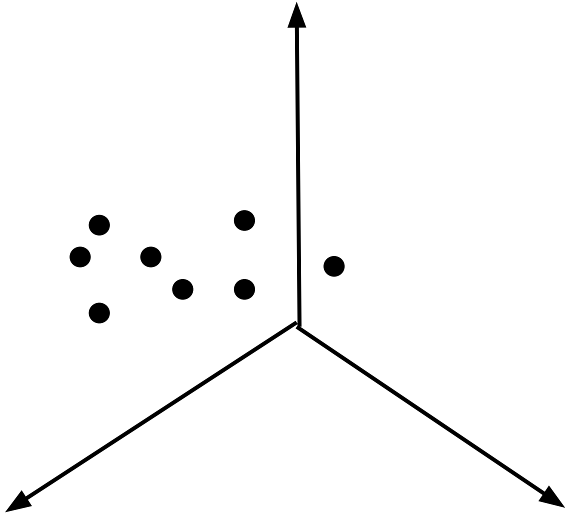
Random synthetic photo

GANs

- To generate fake data: recreate distribution of data



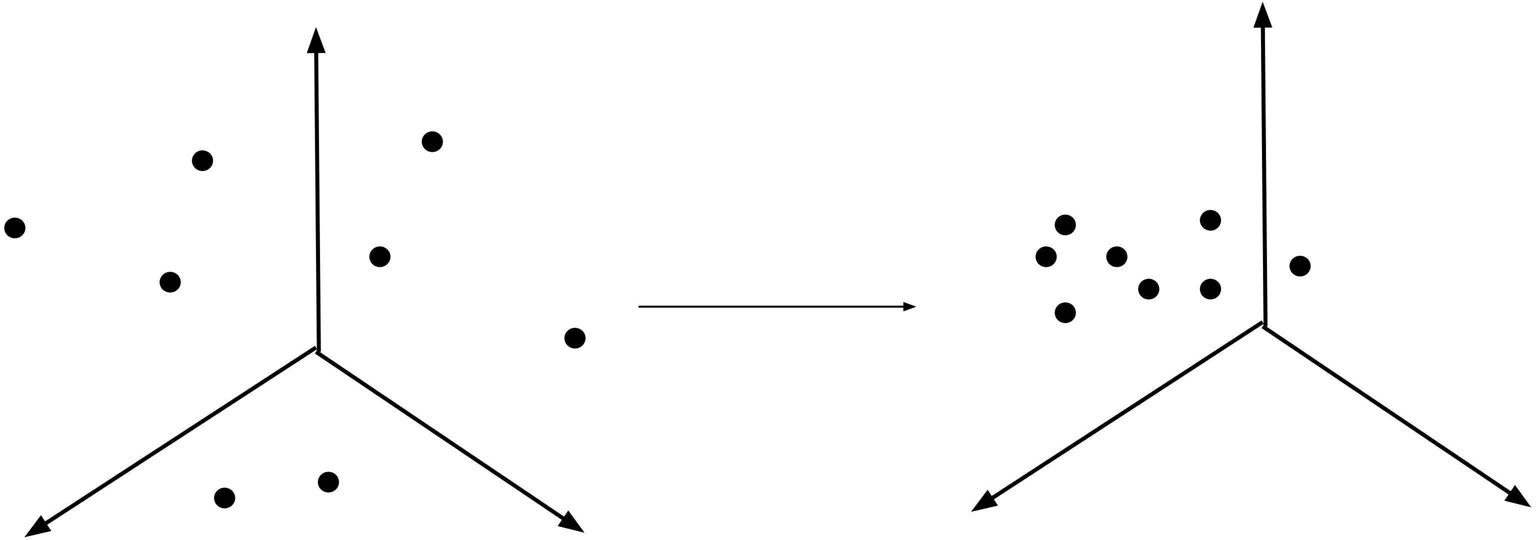
Random data points in 512^2 space



Distribution of real images

GANs

- To generate fake data: recreate distribution of data



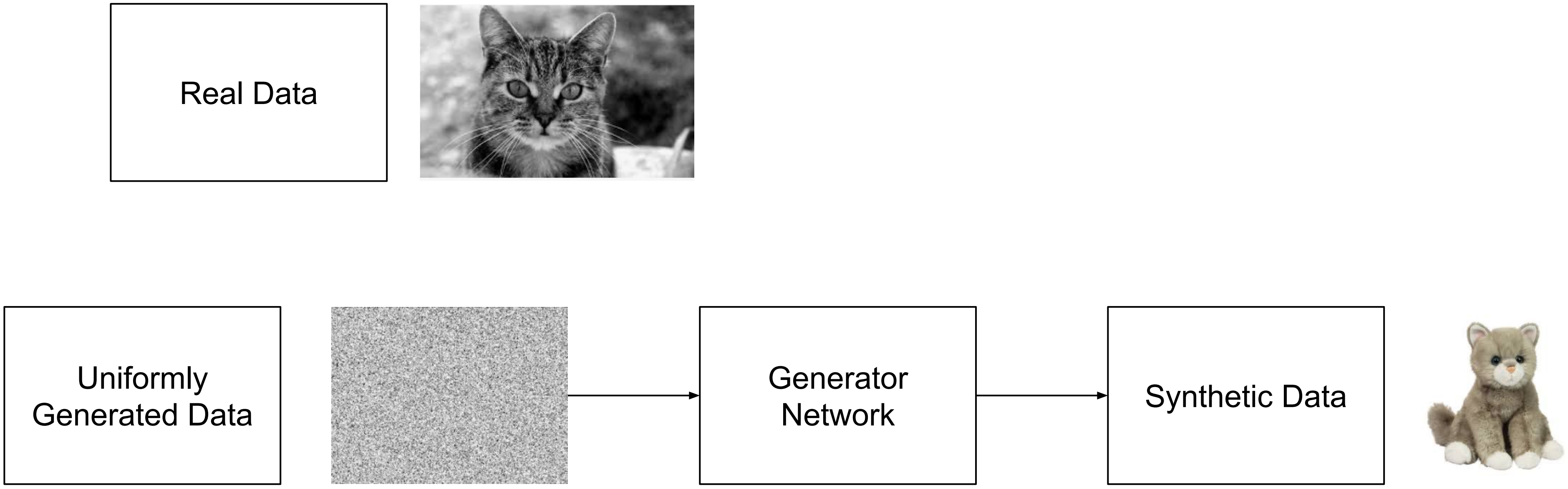
Random data points in 512^2 space

Distribution of real images

GANs

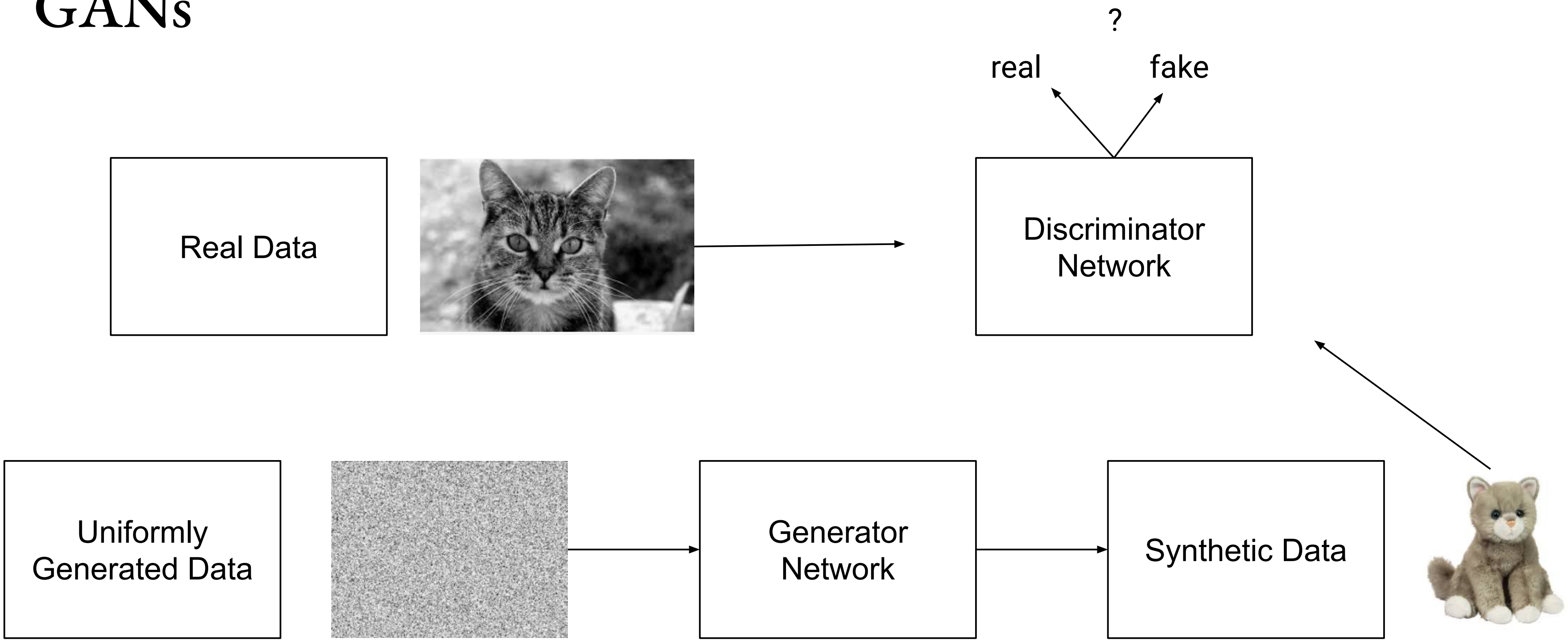
Real Data

GANs

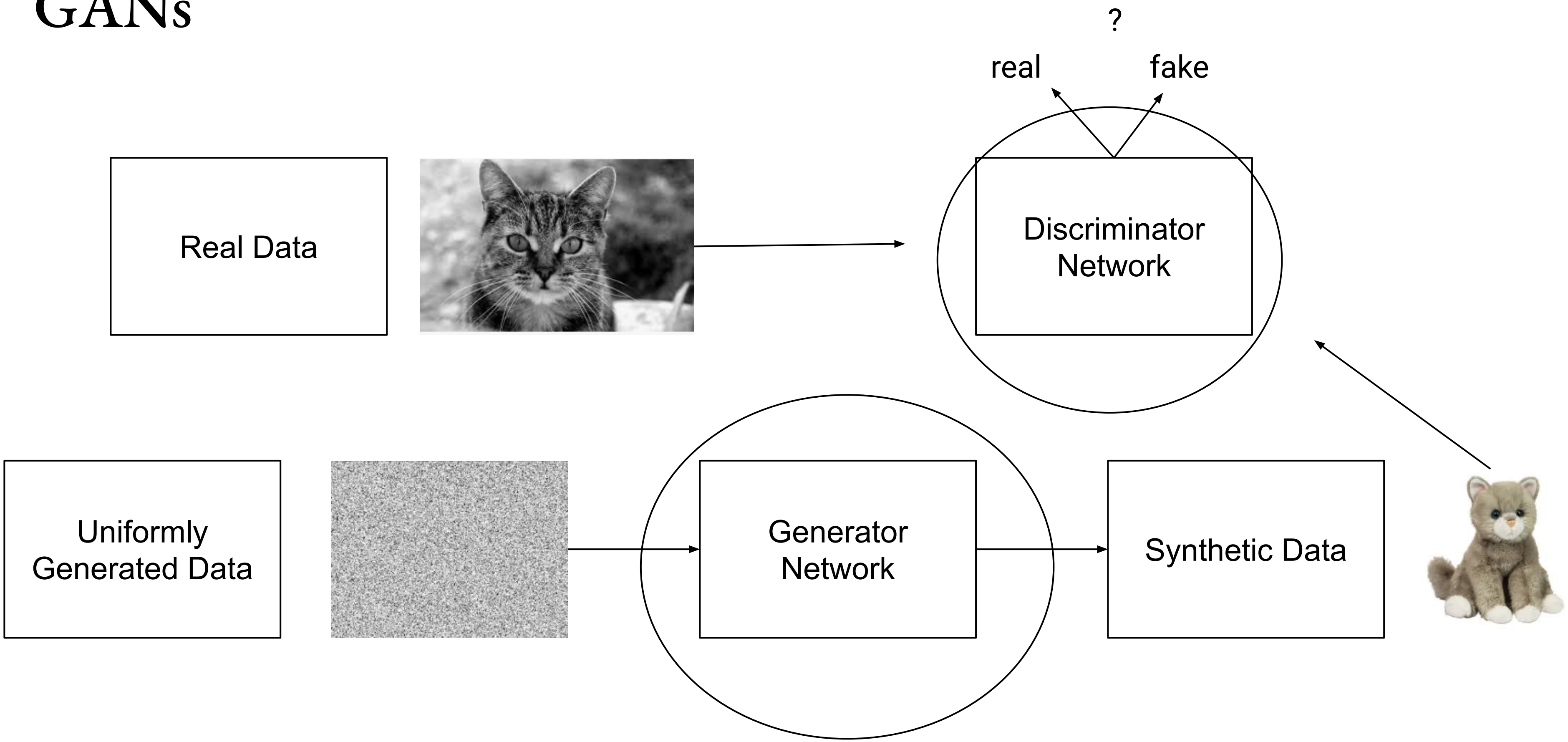


Real Data
Uniformly Generated Data
Generator Network
Synthetic Data

GANs


?
real
fake
Real Data
Discriminator Network
Uniformly Generated Data
Generator Network
Synthetic Data

GANs


?
real
fake
Real Data
Discriminator Network
Uniformly Generated Data
Generator Network
Synthetic Data

GANs

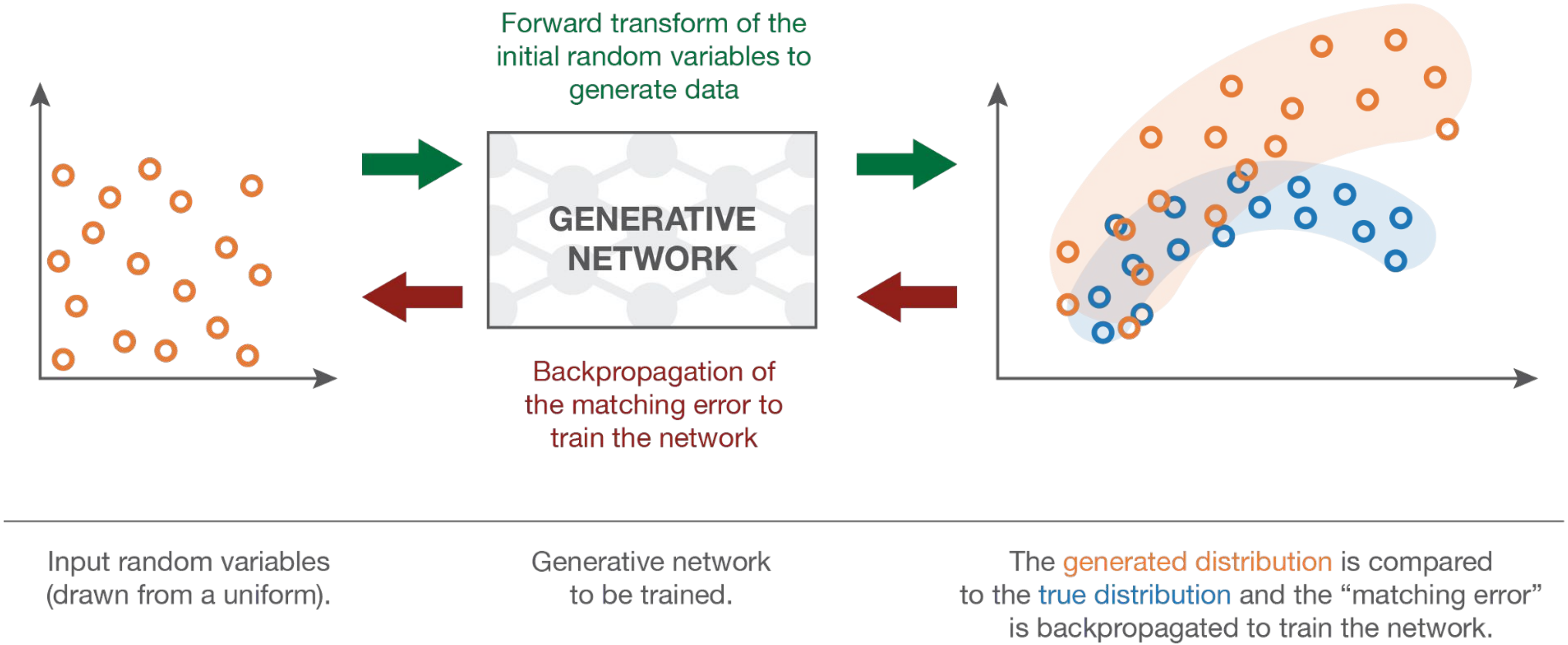
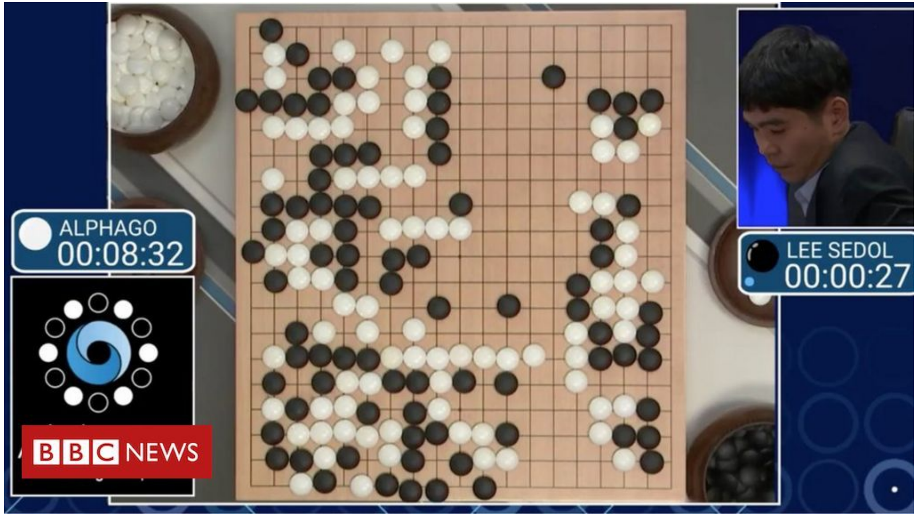


Photo from [here]

Reinforcement Learning

- A third paradigm
  - Supervised Learning
  - Unsupervised Learning
  - Reinforcement Learning



Reinforcement Learning

- No supervisor or labels
  - Goal is to maximize *reward*
  - Learning happens from trial and error
- Data is always sequential
  - What happens next or next best action depends on the previous actions!
- Goal is to maximize cumulative reward or future reward
- May need to take non-rewarding actions now to maximize reward later
  - Sacrifice a piece in chess
  - Spend money to buy a bond/stock to gain more future money

Reinforcement Learning

Agent

Environment

Reinforcement Learning

A_t action

Agent

Environment

Reinforcement Learning

A_t action

R_t reward

S_t

state

Agent

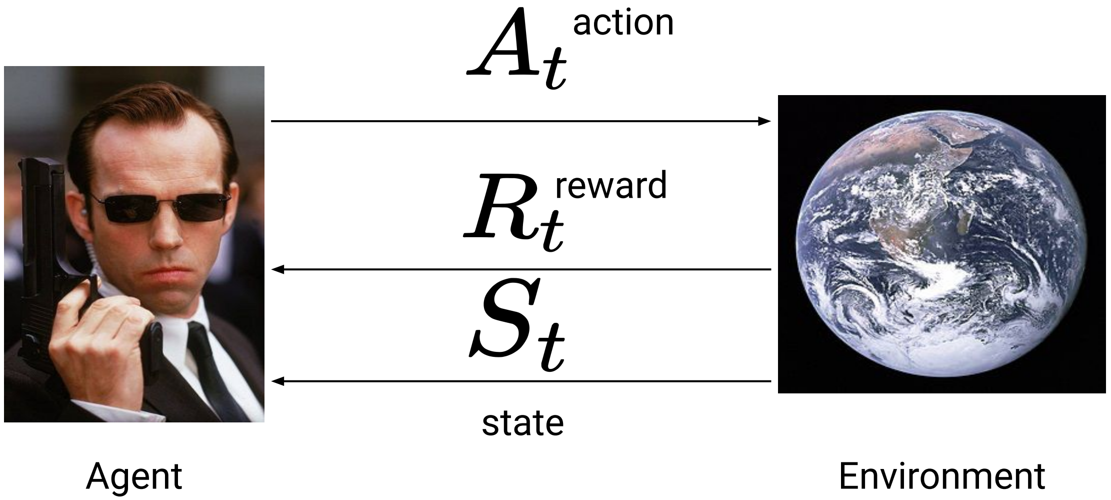
Environment

Reinforcement Learning

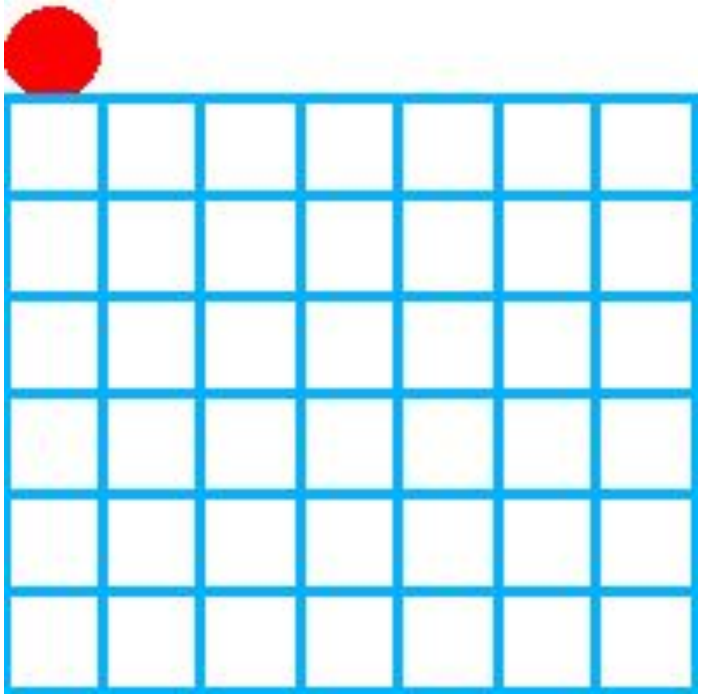
Policy: How the agent decides its next action based on its current state

Value: The expected reward value of a state

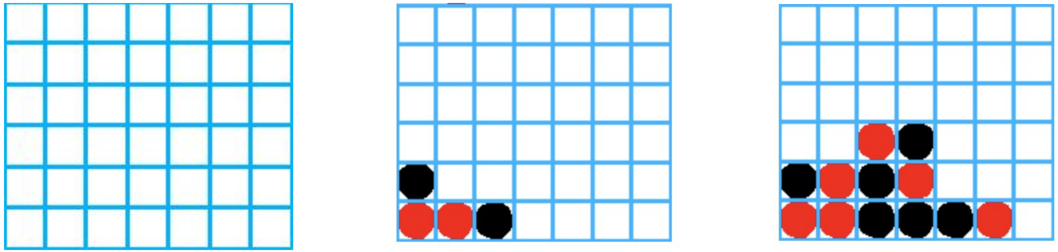
Model: The agent's perception of the environment and how it will respond to actions



Reinforcement Learning



Reinforcement Learning

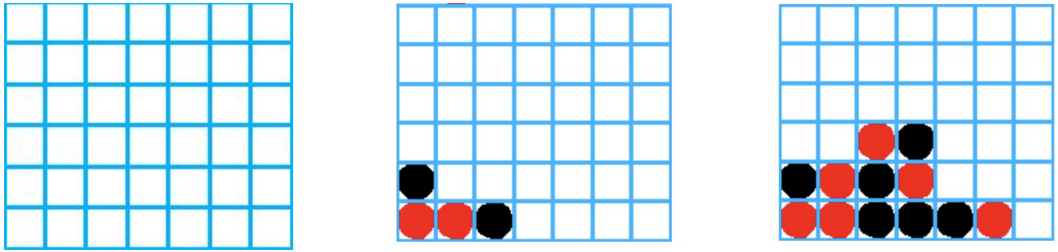


different states

- One player is the agent
- Reward is only obtained when the agent wins the game (reward of +1)
- The value of each state is the probability of winning at this state

$$V(S_t)$$ Value of a state

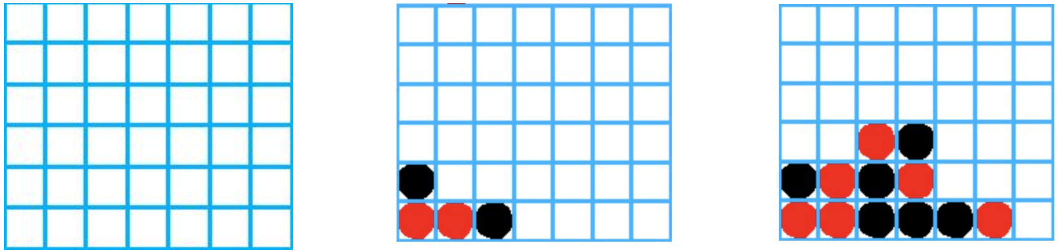
Reinforcement Learning



different states

- What is the value?
    - Play lots of games and take statistics

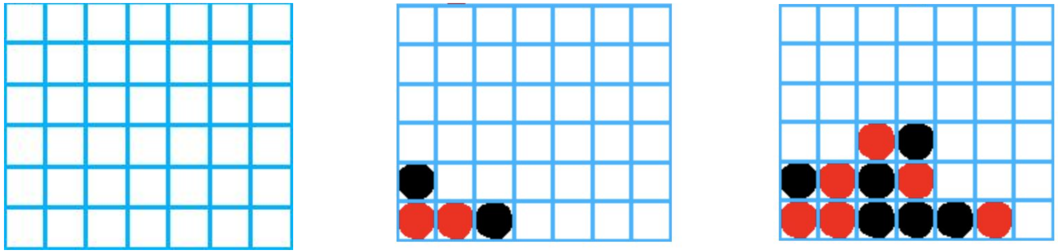
Reinforcement Learning



different states

- What is the value?
    - Play lots of games and take statistics
    - Start with V = 0.5 for all states and learn from playing!

Reinforcement Learning

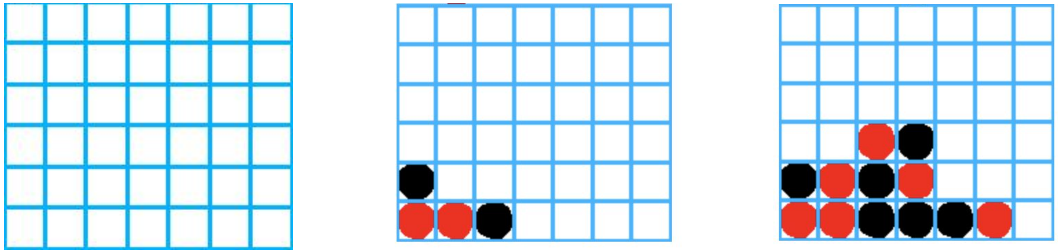


different states

- What is the value?
  - Play lots of games and take statistics
  - Start with V = 0.5 for all states and learn from playing!

$$V(S_t) \leftarrow V(S_t) + \alpha \left[V(S_{t+1}) - V(S_t) \right]$$

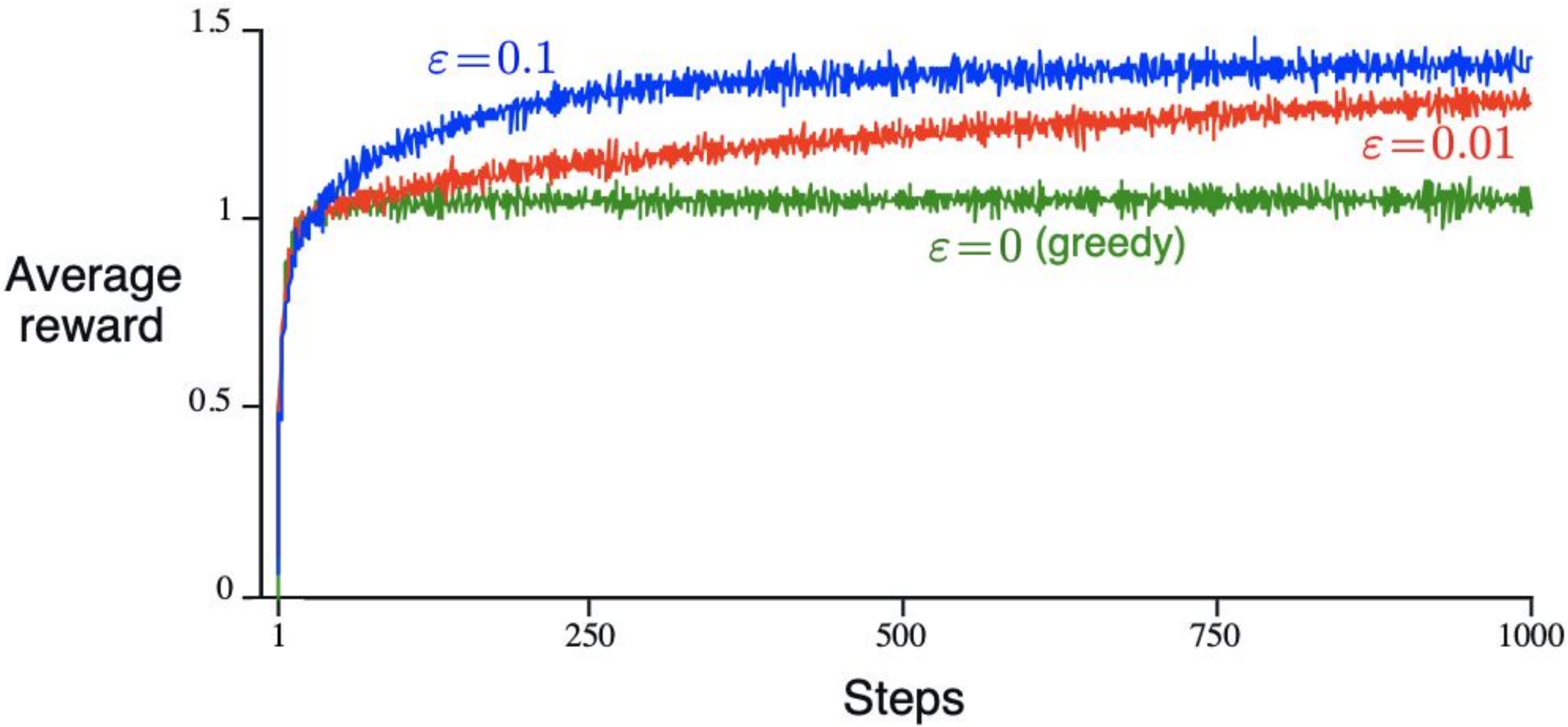
Reinforcement Learning



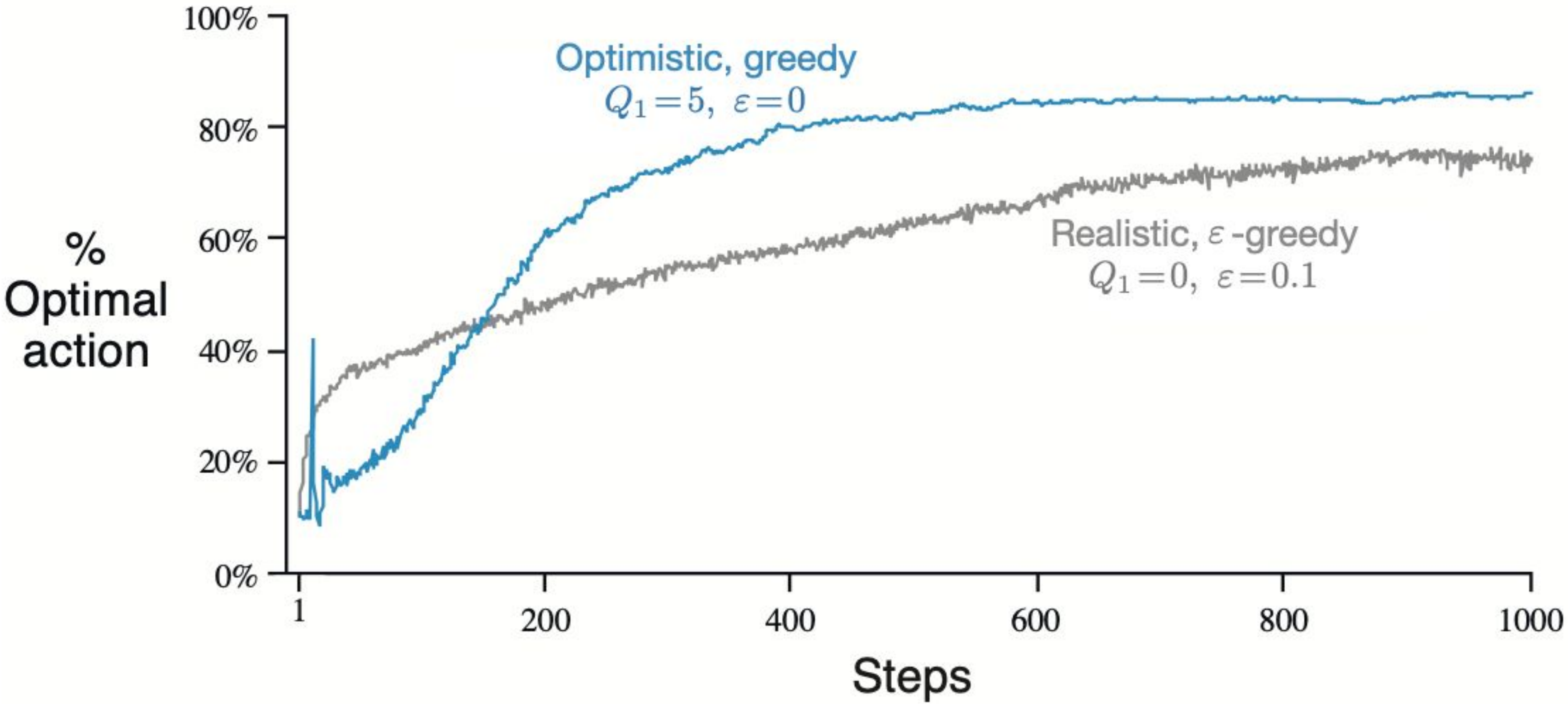
different states

- Greedy Moves
    - Always take the action that results in the highest value state
- Exploratory Moves
    - Randomly take another action, may not be currently “optimal”

Reinforcement Learning



Reinforcement Learning



Reinforcement Learning

- Great textbook
- Great set of lectures

MLOps

- Tracking results (NeptuneAI)
- Pytorch Lightning