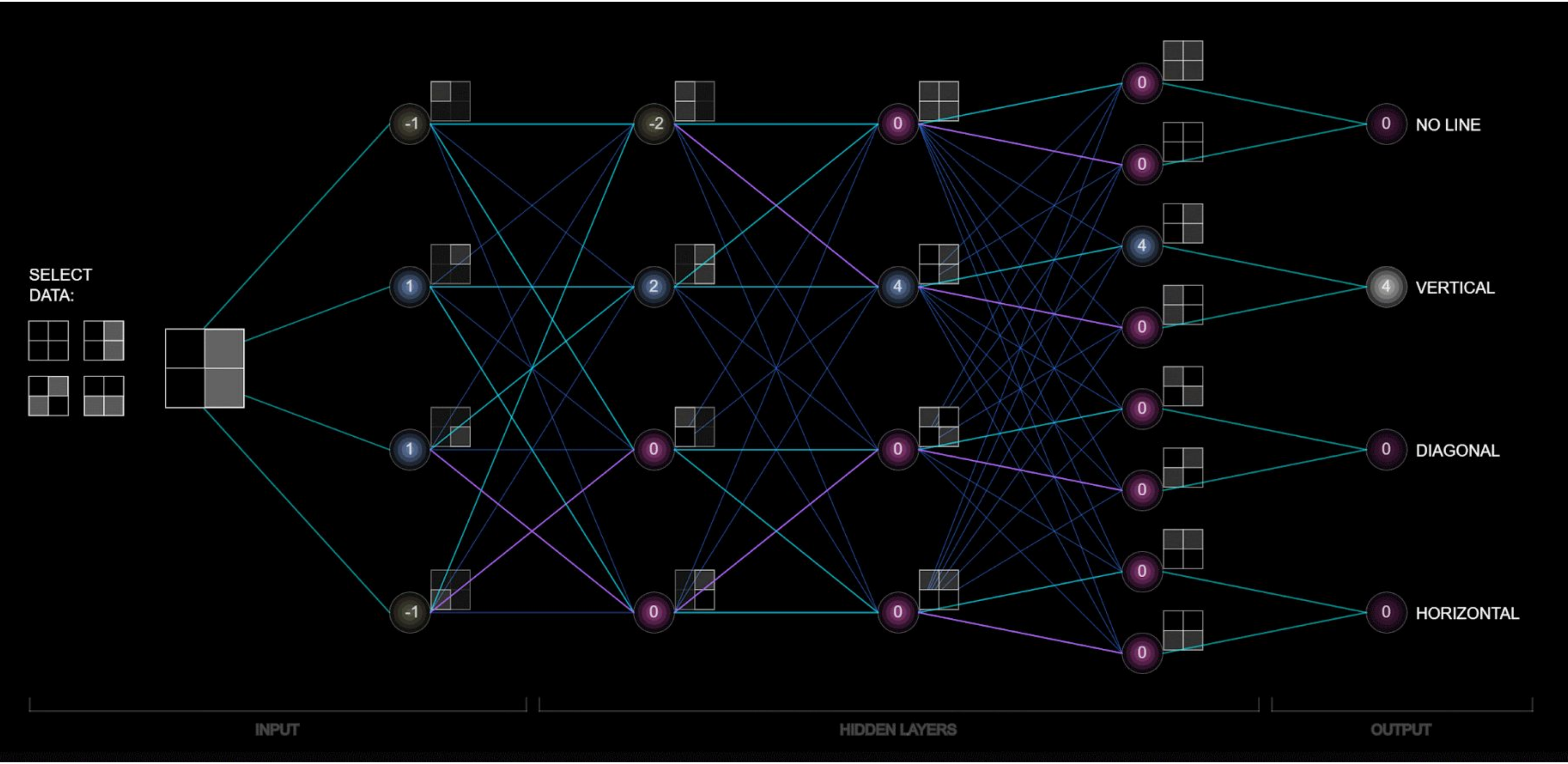
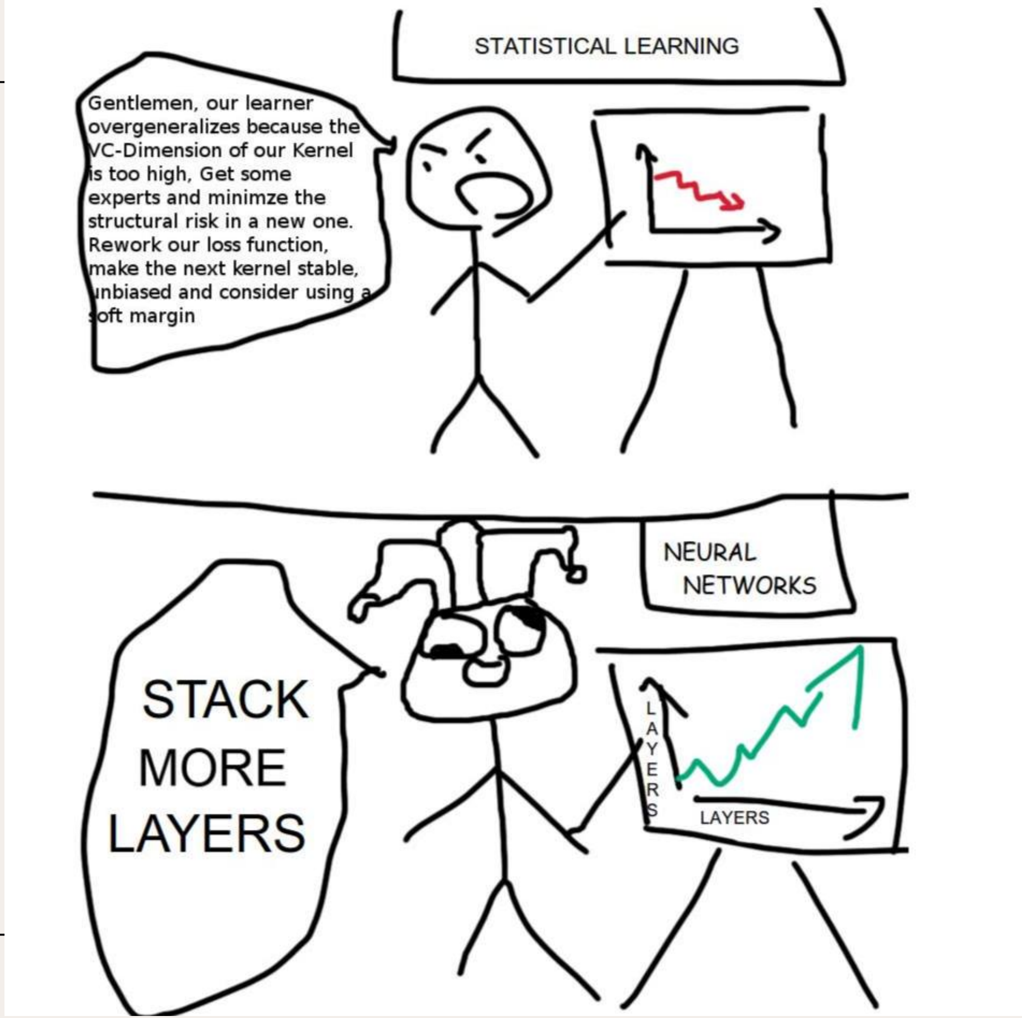
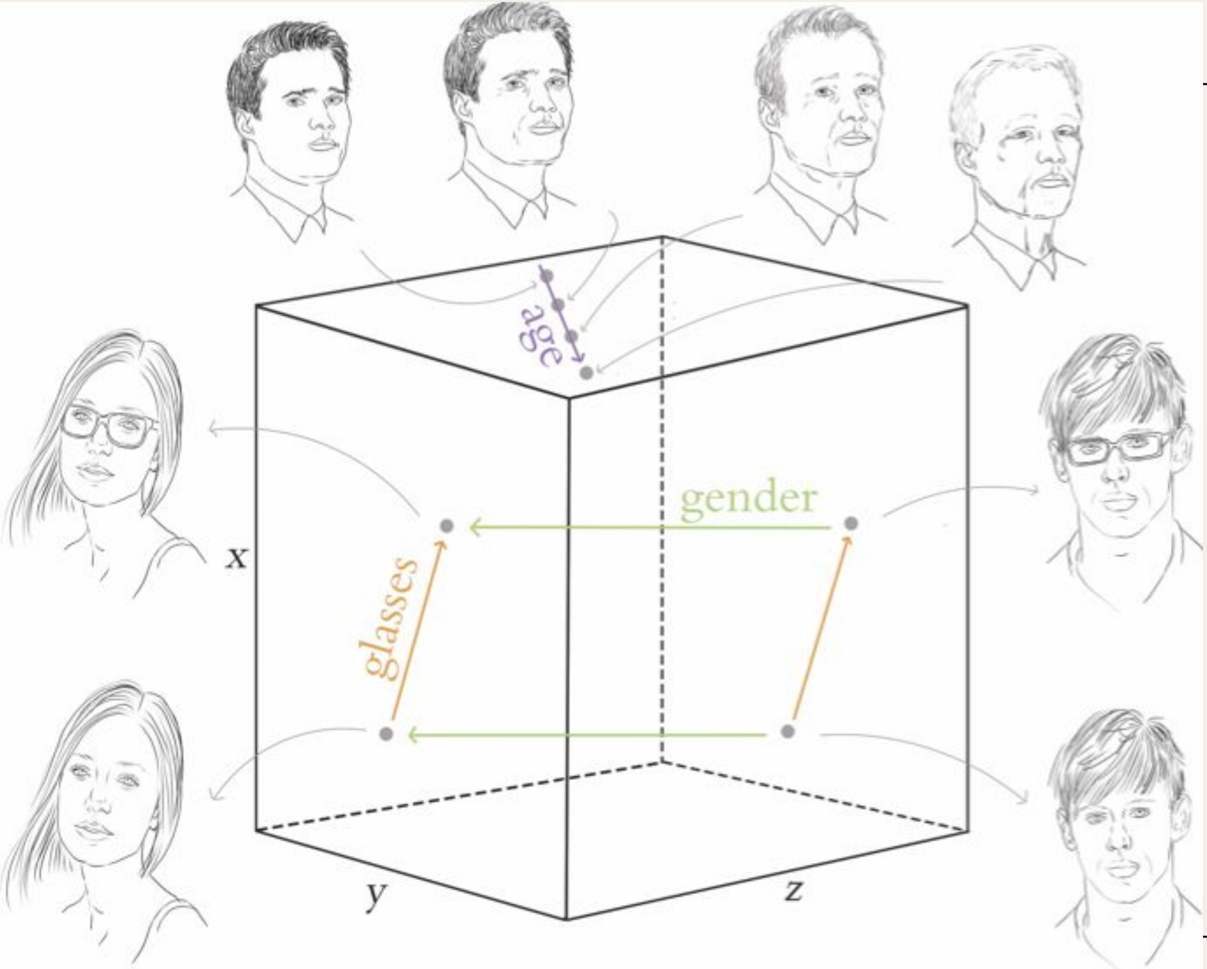

SELECT DATA:
-1
1
1
-1
-2
2
0
0
0
4
0
0
0
0
4
0
0
0
0
0
0
NO LINE
4
VERTICAL
0
DIAGONAL
0
HORIZONTAL
INPUT
HIDDEN LAYERS
OUTPUT

<welcome to>

SIG AI

Meeting 4: 04/18/24


STATISTICAL LEARNING
Gentlemen, our learner overgeneralizes because the VC-Dimension of our Kernel is too high, Get some experts and minimze the structural risk in a new one. Rework our loss function, make the next kernel stable, unbiased and consider using a soft margin
NEURAL NETWORKS
STACK MORE LAYERS
LAYERS
LAYERS



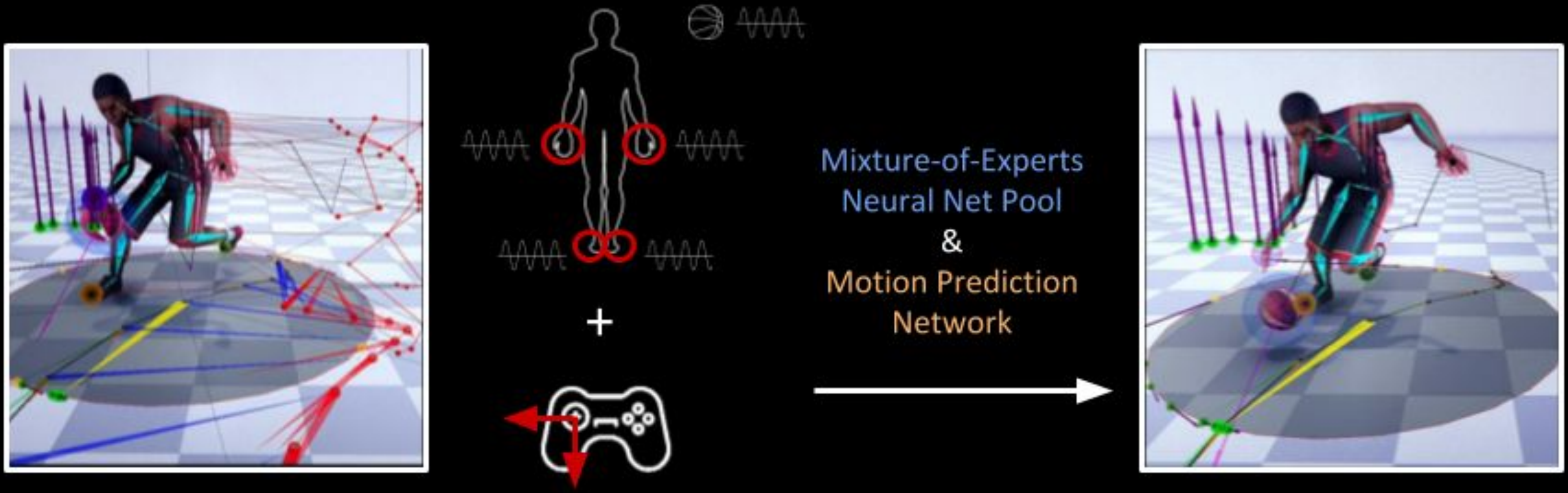
n - dimensional space

(ooga booga) What is Deep Learning?

- Deep learning is a subset of machine learning that uses multi-layered neural networks, called deep neural networks, to simulate the complex decision-making power of the human brain. Some form of deep learning powers most of the artificial intelligence (AI) in our lives today.
- By strict definition, a deep neural network, or DNN, is a neural network with three or more layers. In practice, most DNNs have many more layers. DNNs are trained on large amounts of data to identify and classify phenomena, recognize patterns and relationships, evaluate possibilities, and make predictions and decisions.
- While a single-layer neural network can make useful, approximate predictions and decisions, the additional layers in a deep neural network help refine and optimize those outcomes for greater accuracy.

What use-case can you think of, where using unlabelled data might be beneficial, with more than just input-compute-output?

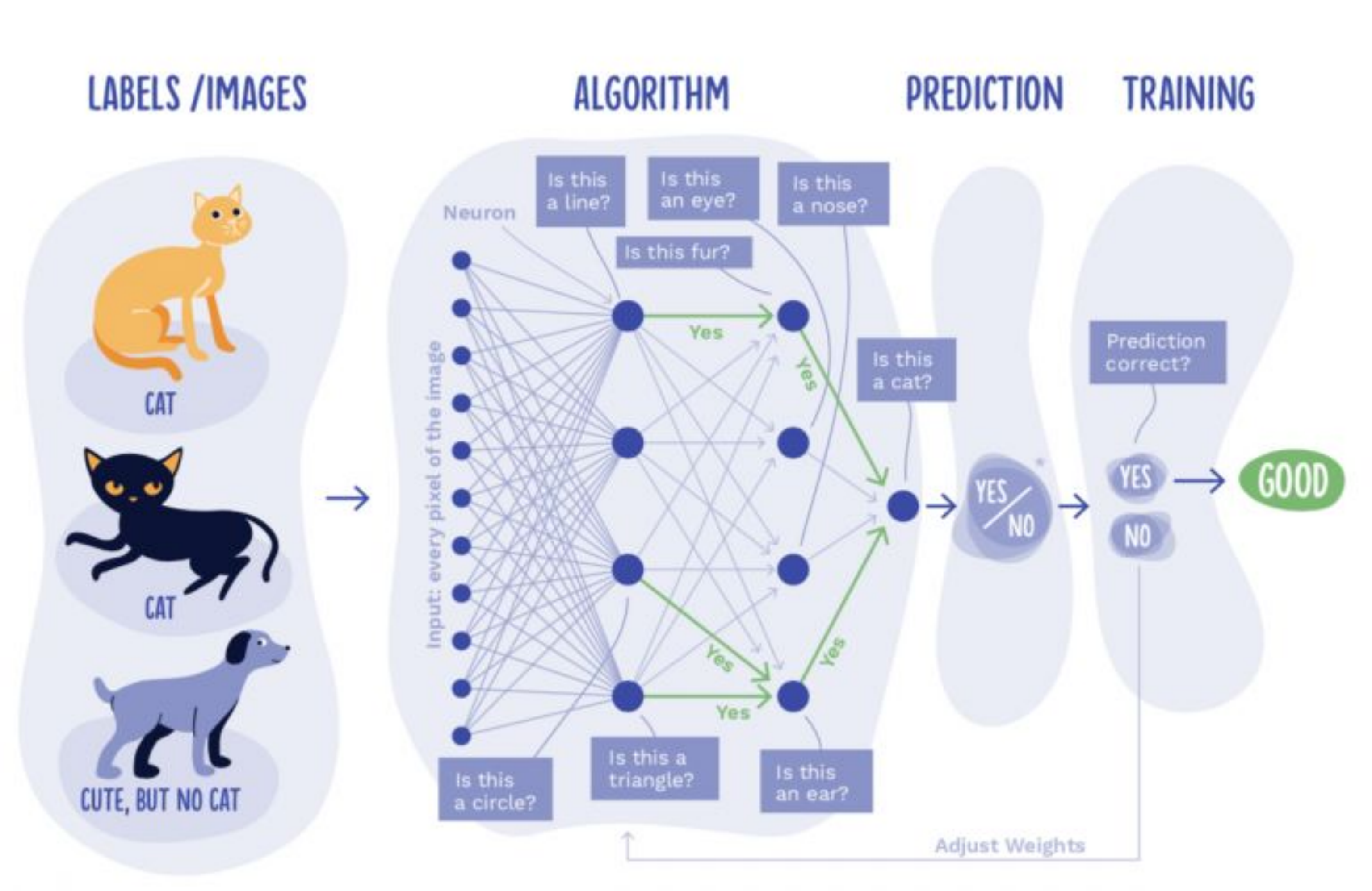
PREDICTING NEXT FRAME



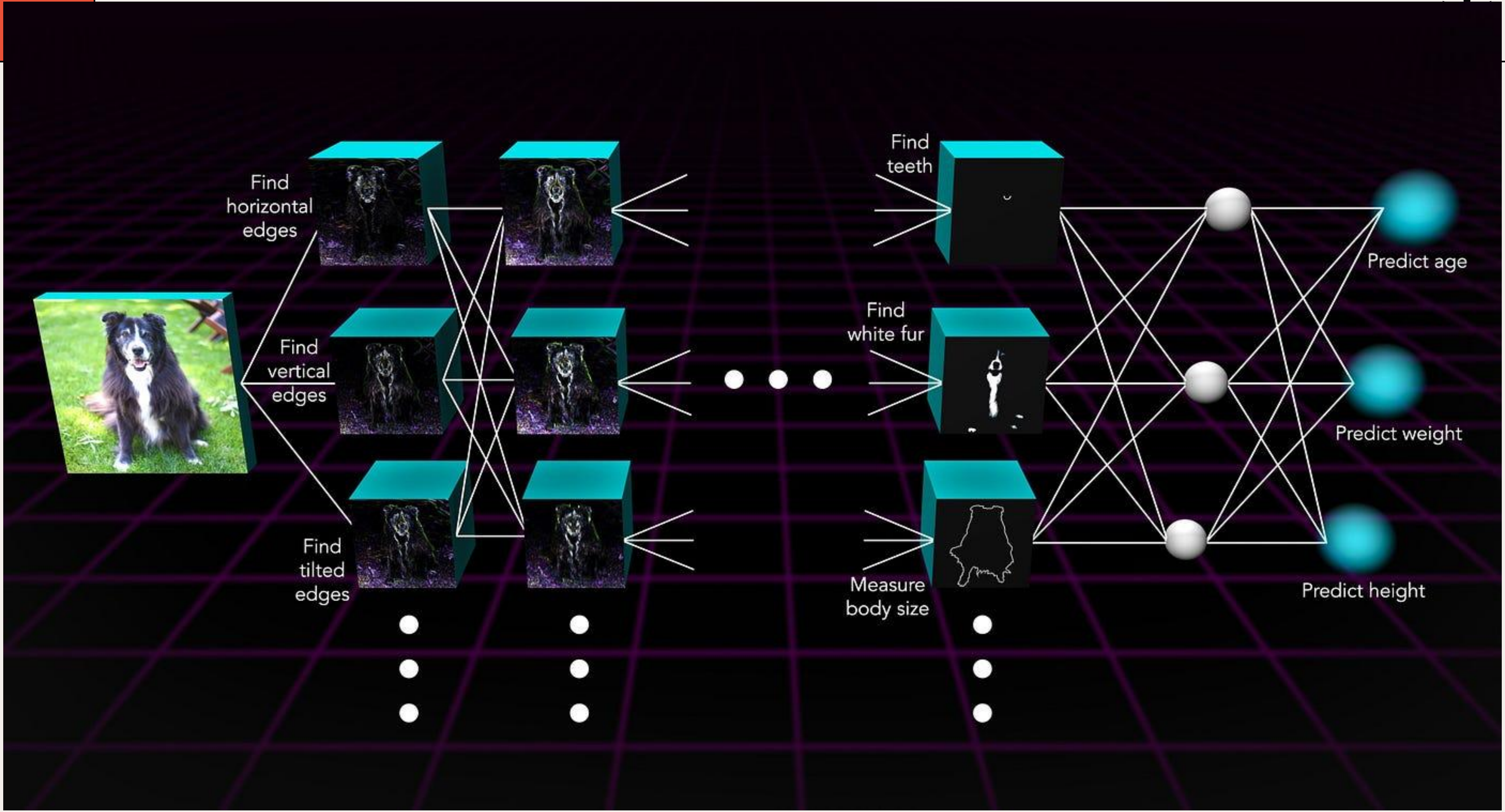
Pose Observed in Current Frame

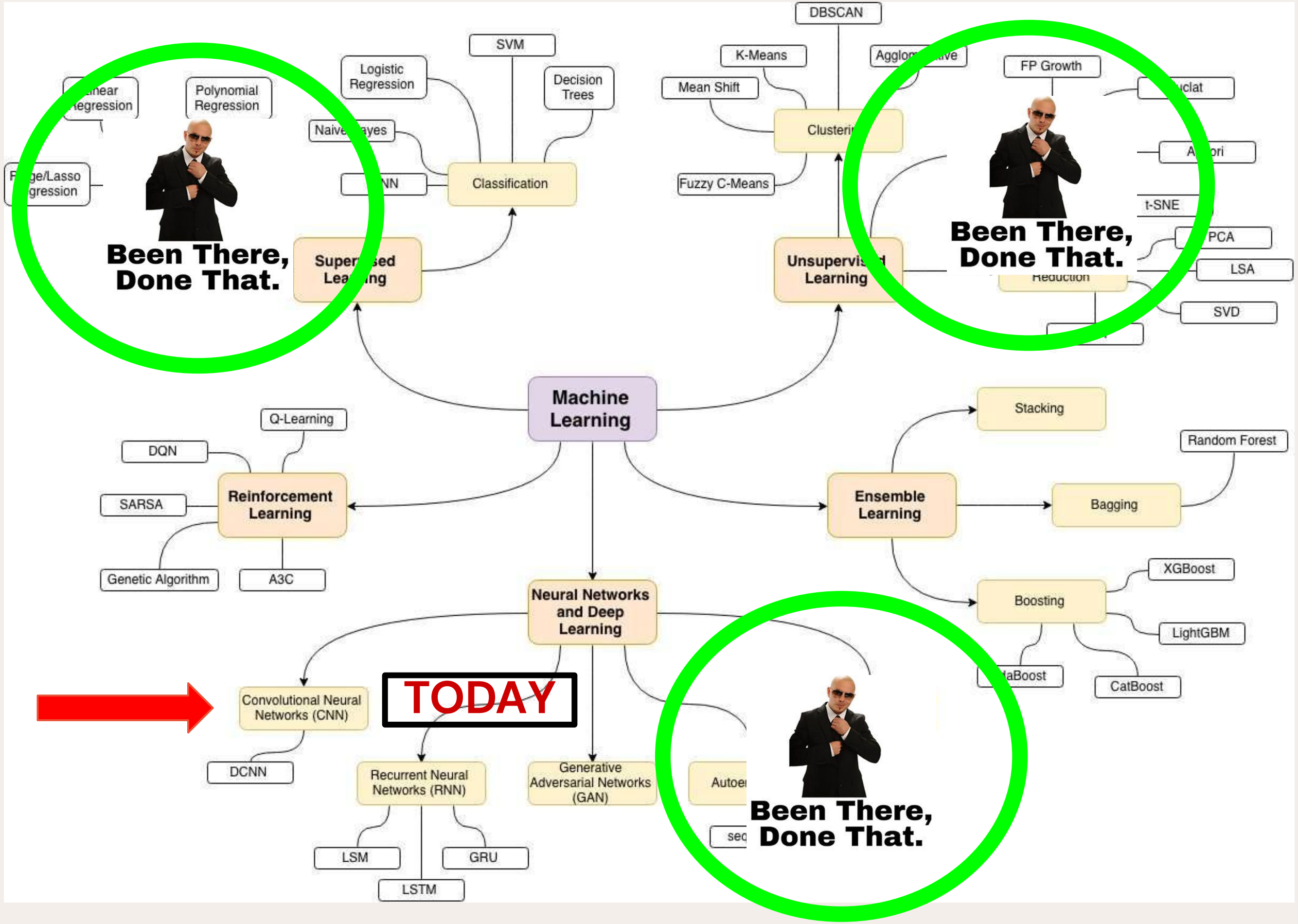
Predicted/Generated Pose in Next Frame

Expert Activation
Actions
Contacts
Opponent
Ball Control
Height
Speed
Bone-Level Phases



LABELS /IMAGES
ALGORITHM
PREDICTION
TRAINING
CAT
CAT
CUTE, BUT NO CAT
Neuron
Input: every pixel of the image
Is this a line?
Is this an eye?
Is this a nose?
Is this fur?
Is this a circle?
Is this a triangle?
Is this an ear?
Yes
Yes
Yes
Yes
Yes
Is this a cat?
YES / NO
Prediction correct?
YES
NO
GOOD
Adjust Weights


Find
horizontal
edges
Find
vertical
edges
Find
tilted
edges
Find
teeth
Find
white fur
Measure
body size
Predict age
Predict weight
Predict height



Machine Learning
Supervised Learning
Classification
SVM
Logistic Regression
Decision Trees
Polynomial Regression
Unsupervised Learning
Clustering
DBSCAN
K-Means
Mean Shift
Fuzzy C-Means
FP Growth
t-SNE
PCA
LSA
SVD
Reinforcement Learning
Q-Learning
DQN
SARSA
Genetic Algorithm
A3C
Ensemble Learning
Stacking
Random Forest
Bagging
Boosting
XGBoost
LightGBM
CatBoost
Neural Networks and Deep Learning
Convolutional Neural Networks (CNN)
TODAY
DCNN
Recurrent Neural Networks (RNN)
Generative Adversarial Networks (GAN)
LSM
GRU
LSTM
Been There, Done That.
Been There, Done That.
Been There, Done That.

Ok, what are the different types? *

Perceptrons

Autoencoders

Convolution Neural Networks (CNNs)

Generative Adversarial Networks (GANs)

Recurrent Neural Networks (RNNs)

Ok, what are the different types? *

Convolution Neural Networks (CNNs)

Similar to feedforward networks, but they're usually utilized for image recognition, pattern recognition, and/or computer vision. These networks harness principles from linear algebra, particularly matrix multiplication, to identify patterns within an image.

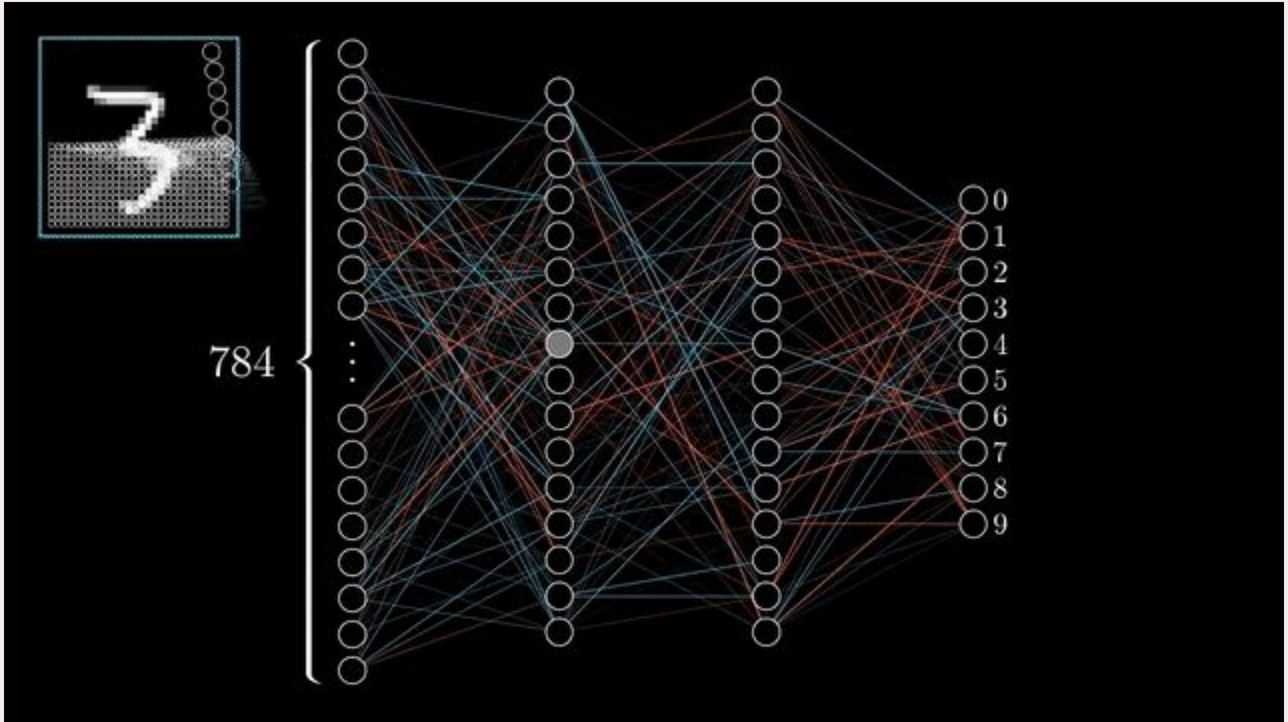
Recurrent Neural Networks (RNNs)

They are identified by their feedback loops. These learning algorithms are primarily leveraged when using time-series data to make predictions about future outcomes, such as stock market predictions or sales forecasting.

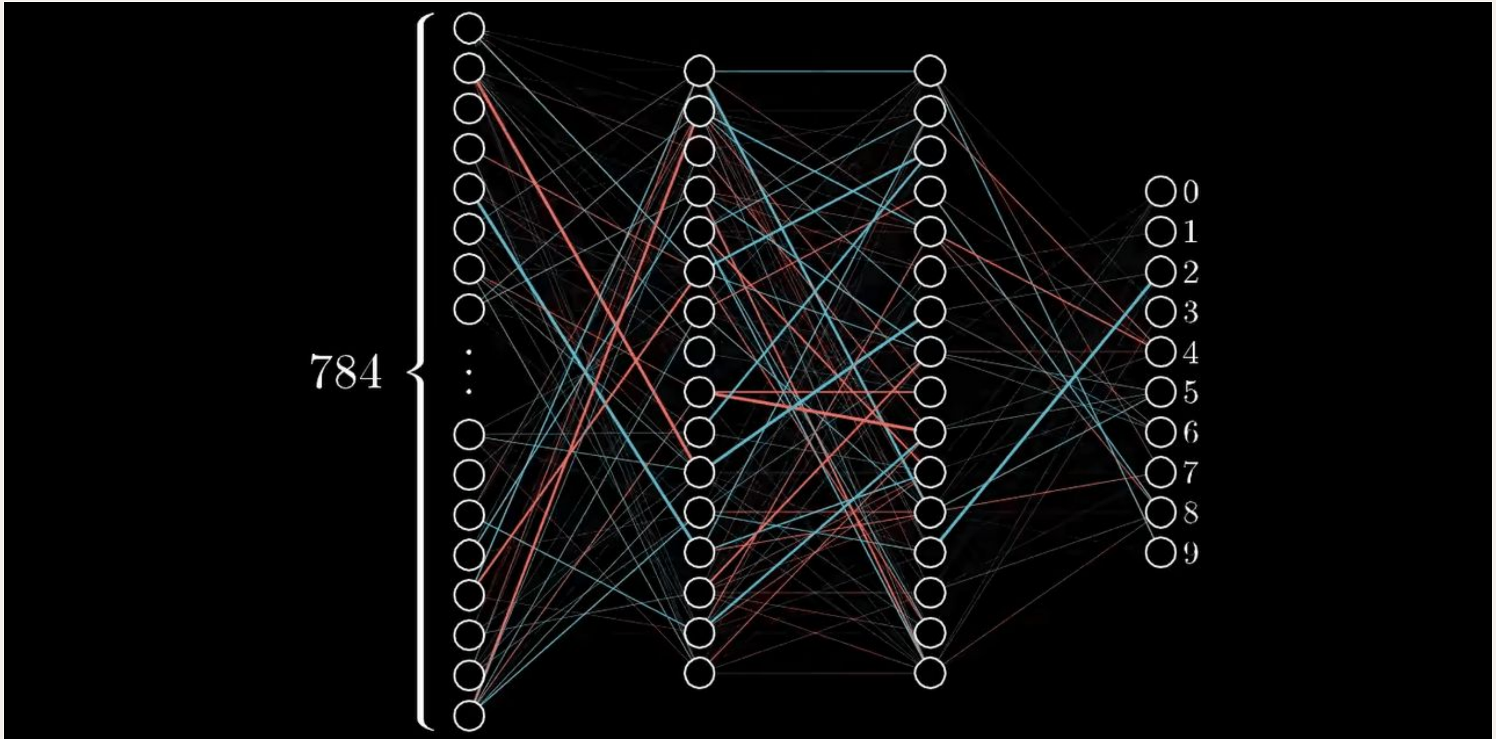
Generative Adversarial Networks (GANs)

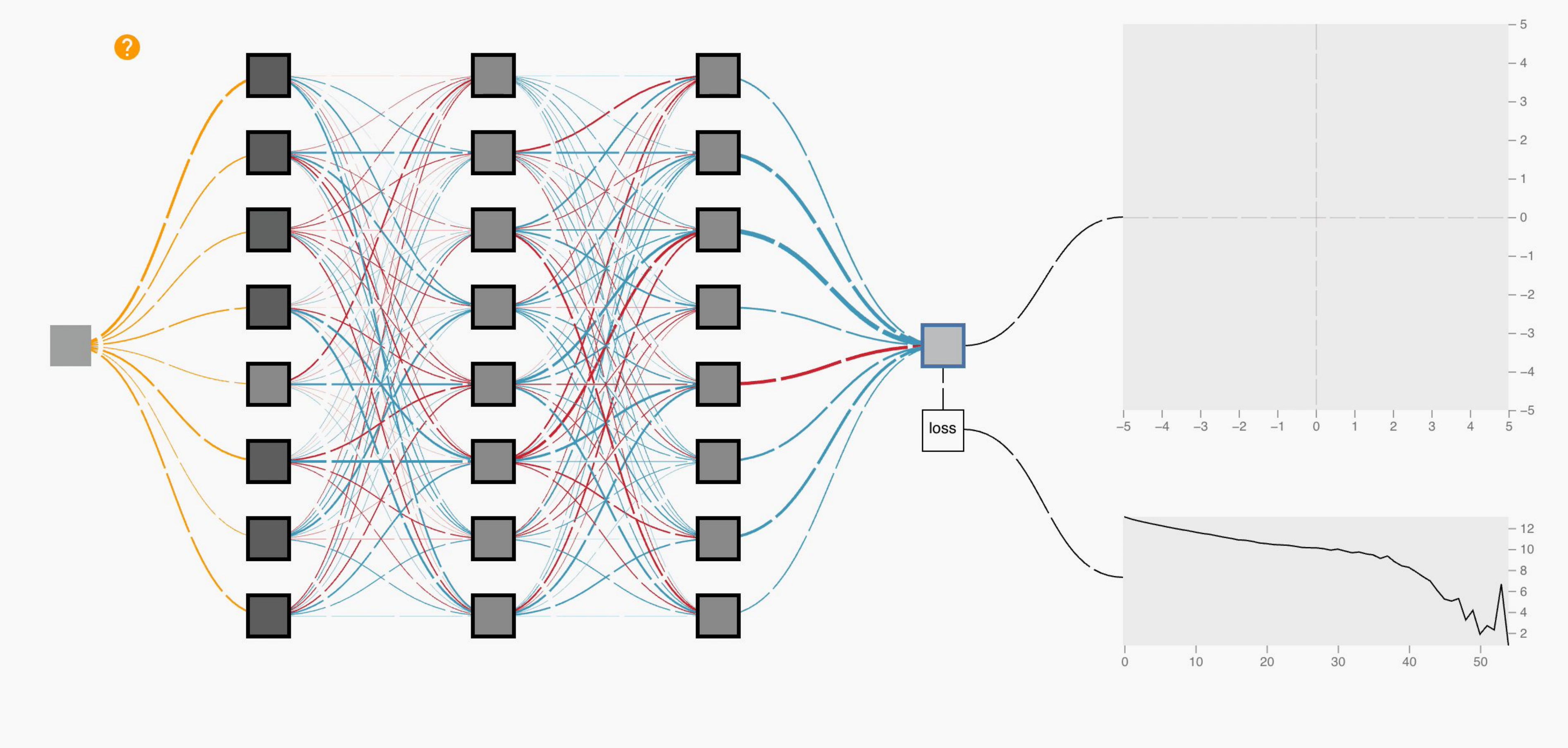
GANs create new output from the pattern found in the training dataset. In it two neurons are pitched against each other and creating a new output. These two neurons are generators and discriminators.

Secret Sauce of Neural Networks and Deep Learning ?

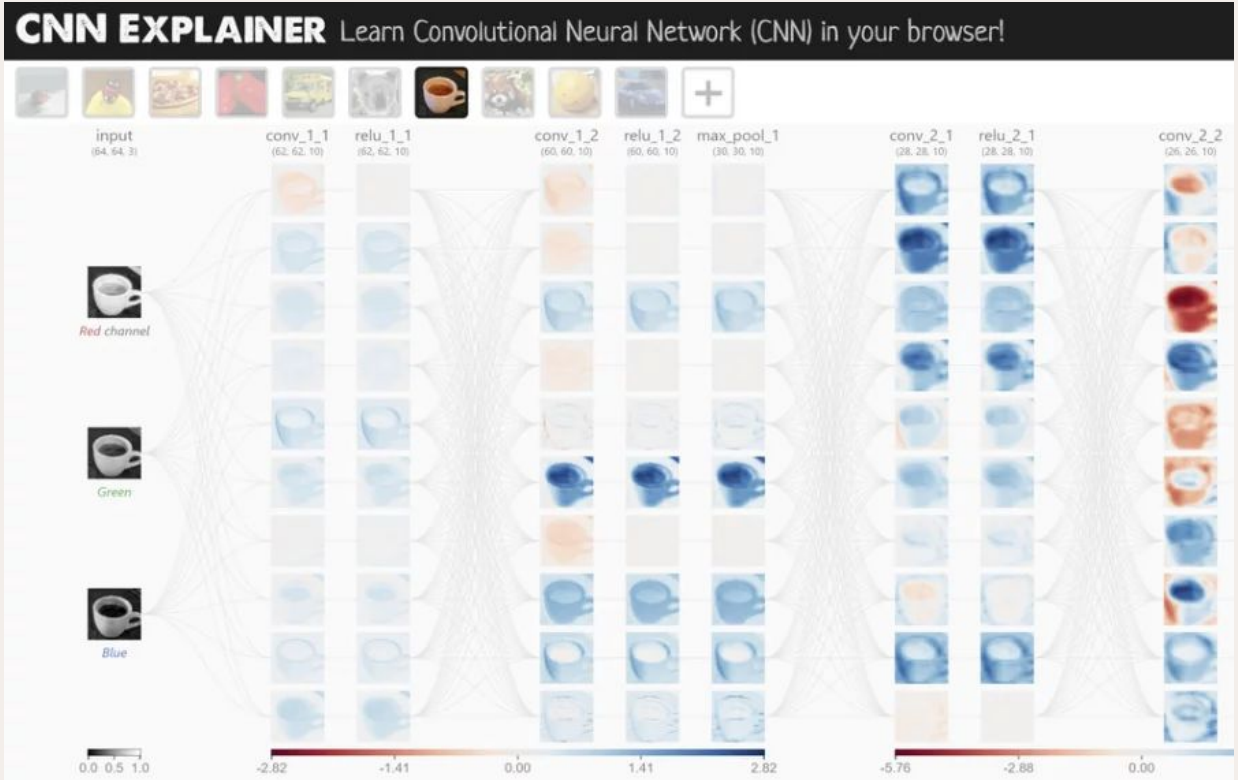


Secret Sauce of Neural Networks and Deep Learning ?



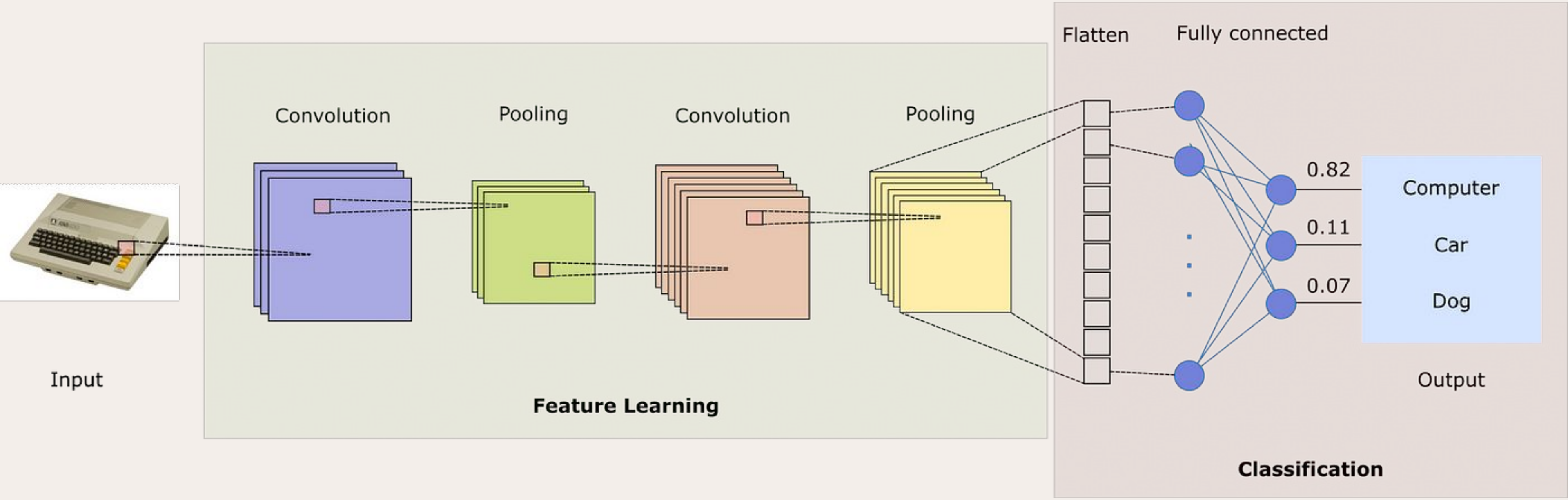

loss
5
4
3
2
1
0
−1
−2
−3
−4
−5
−5
−4
−3
−2
−1
0
1
2
3
4
5
12
10
8
6
4
2
0
10
20
30
40
50

Convolution Neural Networks (CNNs)

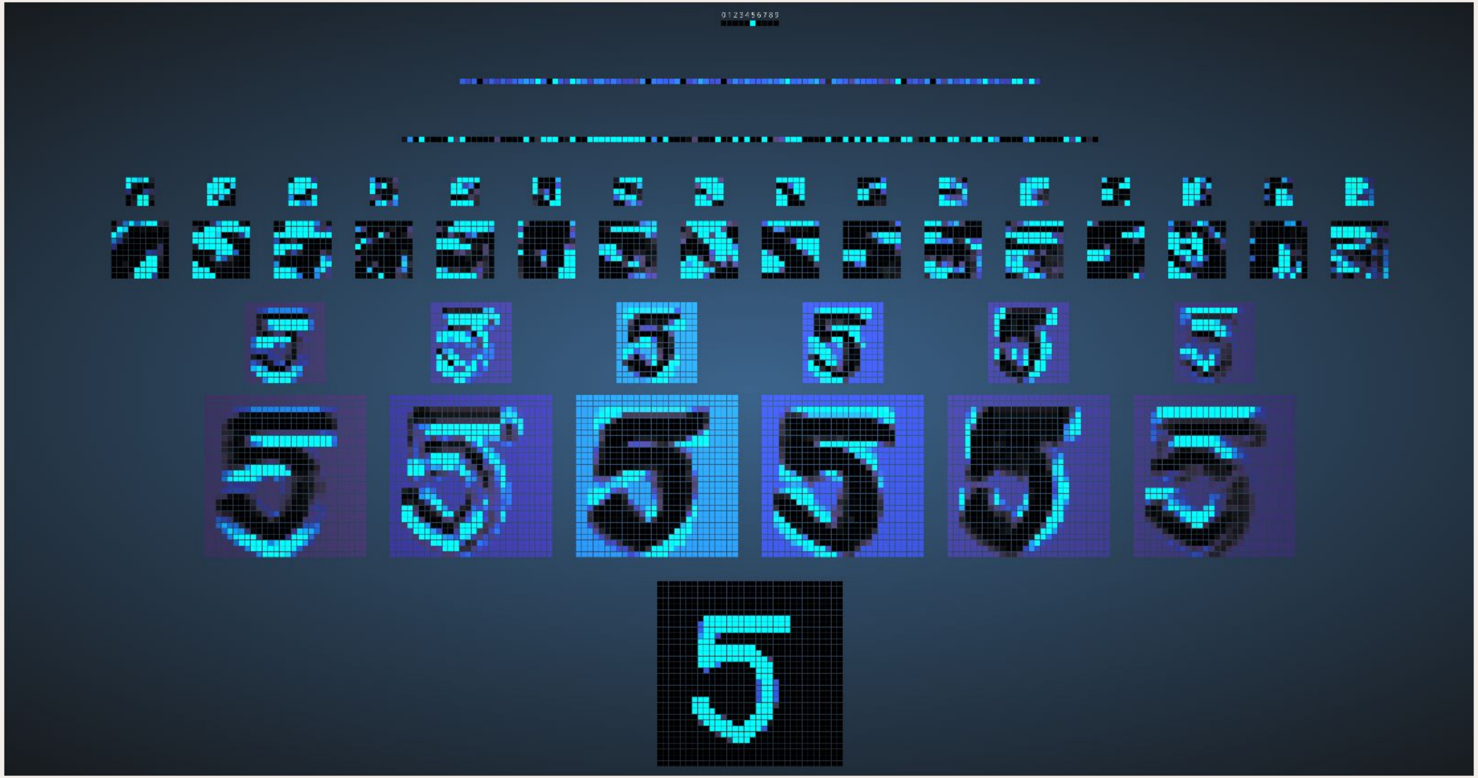


Convolution Neural Networks (CNNs): What are they and why are they used?

Convolutional neural networks use three-dimensional data to for image classification and object recognition tasks.



Convolution Neural Networks (CNNs):



Output Layer

Fully Connected 2

Fully Connected 1

Downsampling 2

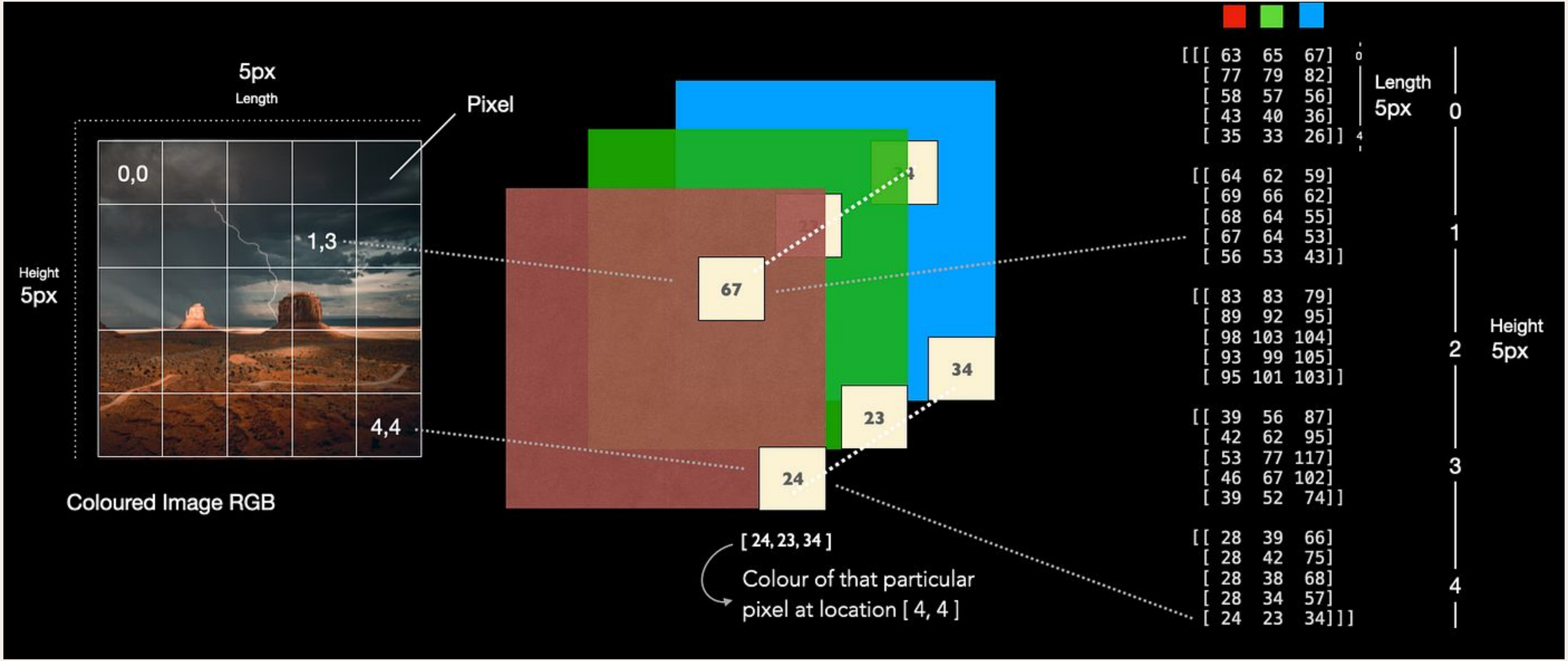
CNN 2

Downsampling 1

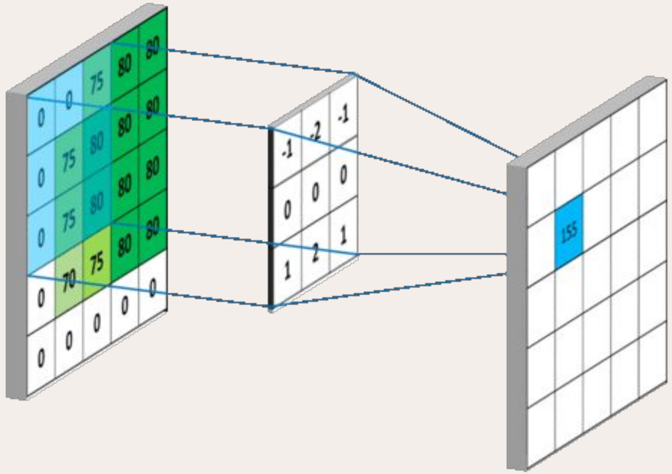
CNN 1

Input Layer

CNNs: Input Layer

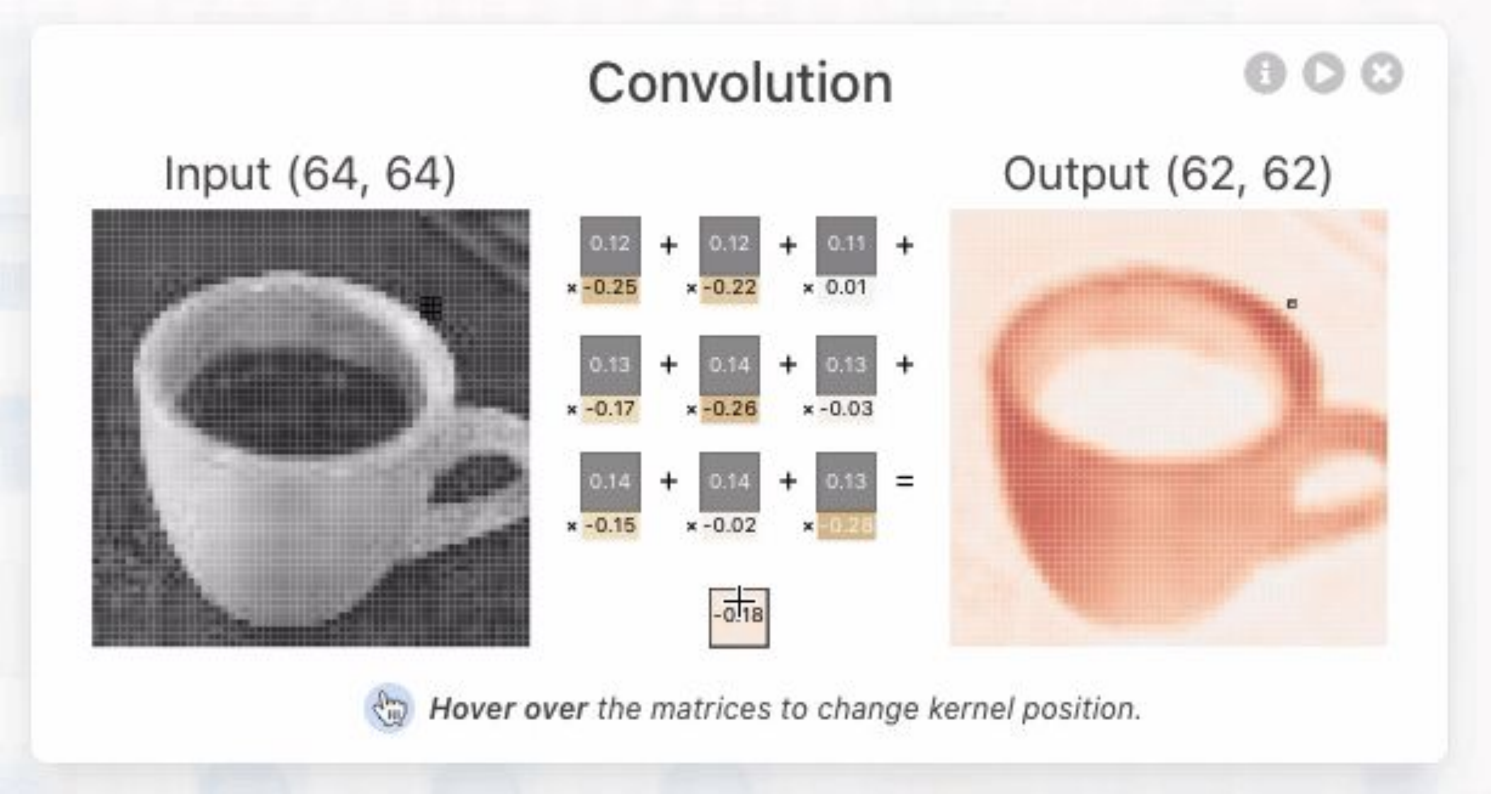


CNNs: Convolution Layer



- This layer performs a dot product between two matrices, where one matrix is the set of learnable parameters otherwise known as a kernel, and the other matrix is the restricted portion of the receptive field.
- As the name suggests, the main mathematical task performed is called convolution, which is the application of a sliding window function to a matrix of pixels representing an image. The sliding function applied to the matrix is called kernel or filter, and both can be used interchangeably.
- In the convolution layer, several filters of equal size are applied, and each filter is used to recognize a specific pattern from the image, such as the curving of the digits, the edges, the whole shape of the digits, and more.
- Put simply, in the convolution layer, we use small grids (called filters or kernels) that move over the image.
- Each small grid is like a mini magnifying glass that looks for specific patterns in the photo, like lines, curves, or shapes. As it moves across the photo, it creates a new grid that highlights where it found these patterns.
- **Size of the ‘image’ is bound to reduce post convolution layer!**

CNNs: Convolution Layer: Matrix Multiplication



CNNs: Convolution Layer: Building a feature map

$$
\begin{bmatrix} 7 & 2 & 3 & 3 & 8 \\ 4 & 5 & 3 & 8 & 4 \\ 3 & 3 & 2 & 8 & 4 \\ 2 & 8 & 7 & 2 & 7 \\ 5 & 4 & 4 & 5 & 4 \end{bmatrix}
*
\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}
=
\begin{bmatrix} 6 & & \\ & & \\ & & \end{bmatrix}
$$

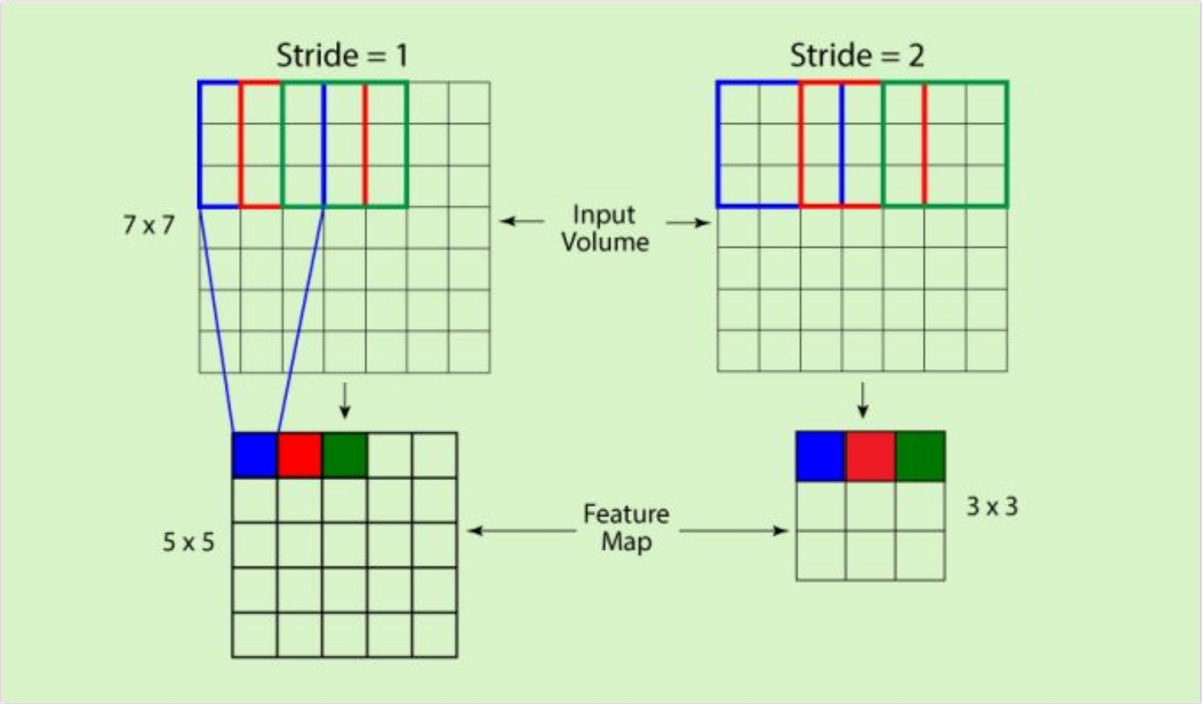
7x1+4x1+3x1+
2x0+5x0+3x0+
3x-1+3x-1+2x-1
= 6

Input

Kernel/Filter/Grandma

Feature Map/Result of grandma's nagging/Convoluted Matrice

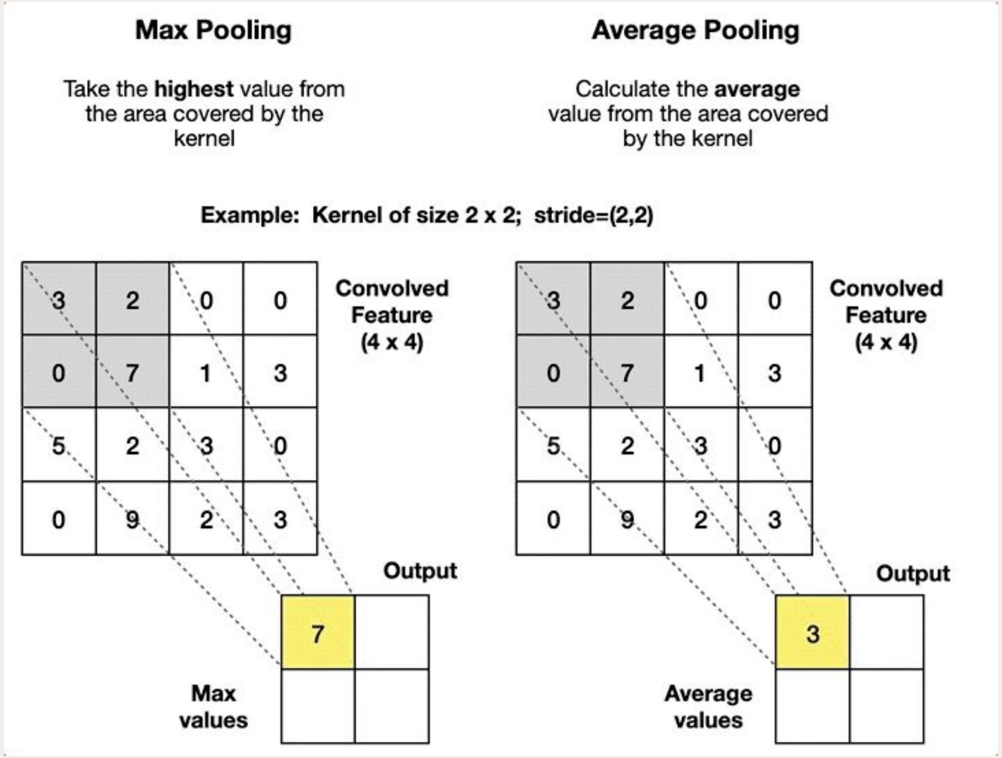
CNNs: Convolution Layer: Kernel Manipulation



0	0	0	0	0	0	0
0	120	10	10	10	120	0
0	10	120	10	120	10	0
0	10	10	120	10	10	0
0	10	120	10	120	10	0
0	120	10	10	10	120	0
0	0	0	0	0	0	0

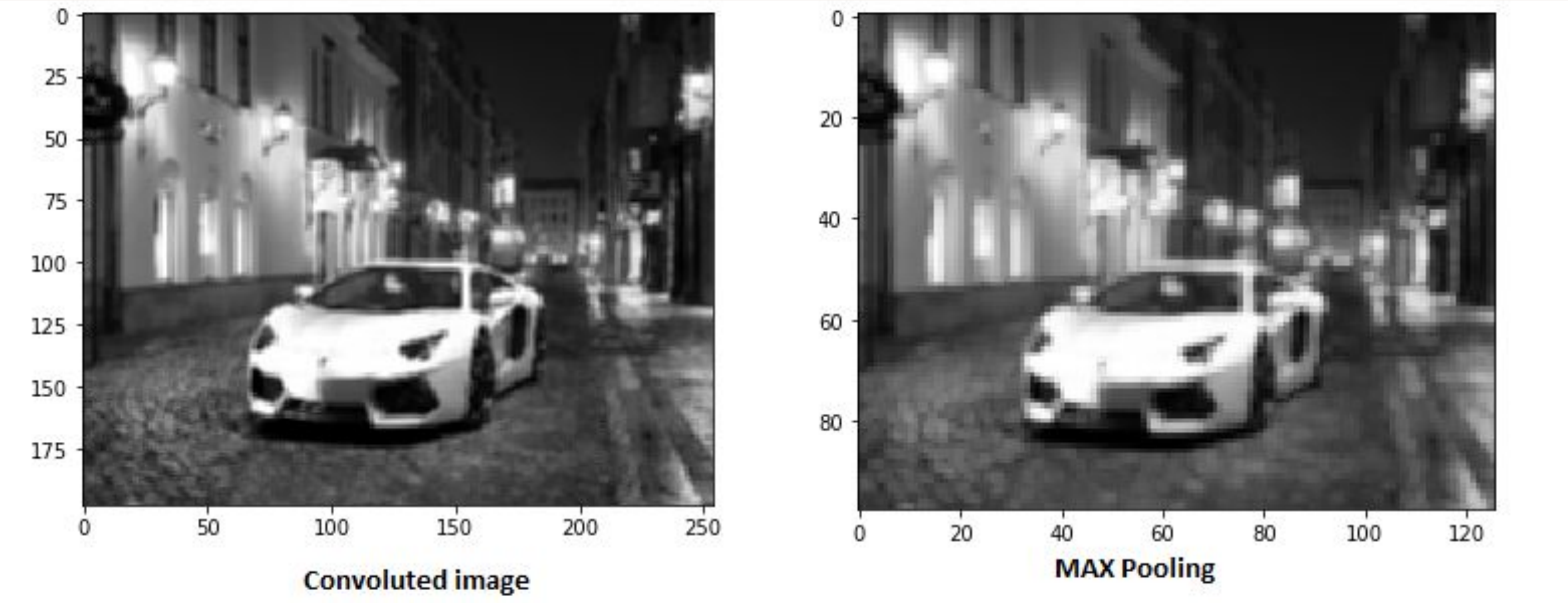
QUESTION: would you pad the input image or output image?

CNNs: Downsampling/Pooling Layer

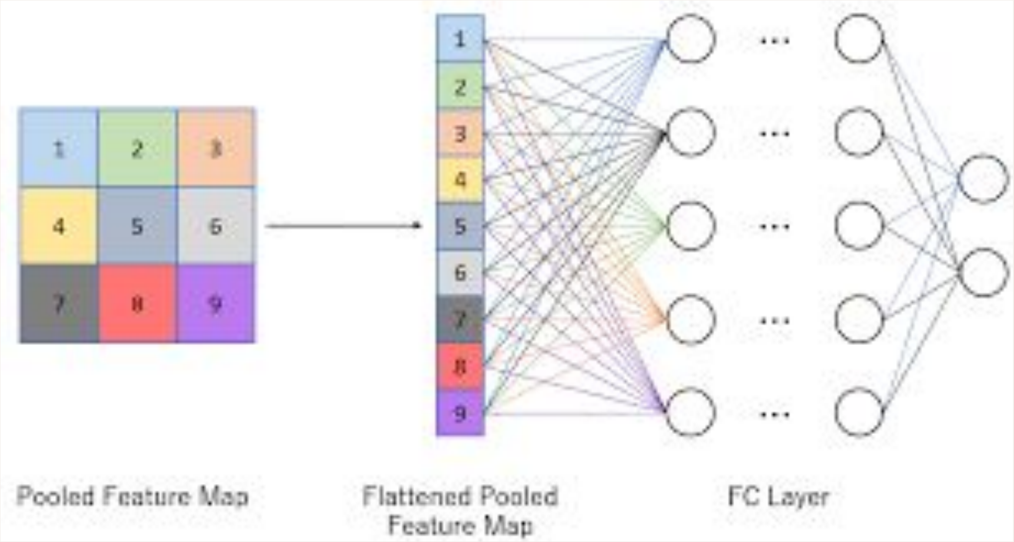


- Though machines are exceptionally fast, it will take quite some time to process these humongous number of pixel values.
- One basic instinct would be to select a dominant value among the group.
- This is achieved by the machines using an operation called Pooling.
- The main motive for having the Pooling layer is for **dimensionality reduction**
- The most common aggregation functions that can be applied are:
    - Max pooling, which is the maximum value of the feature map
    - Sum pooling corresponds to the sum of all the values of the feature map
    - Average pooling is the average of all the values.

CNNs: Downsampling/Pooling Layer



CNNs: Fully Connected Layer

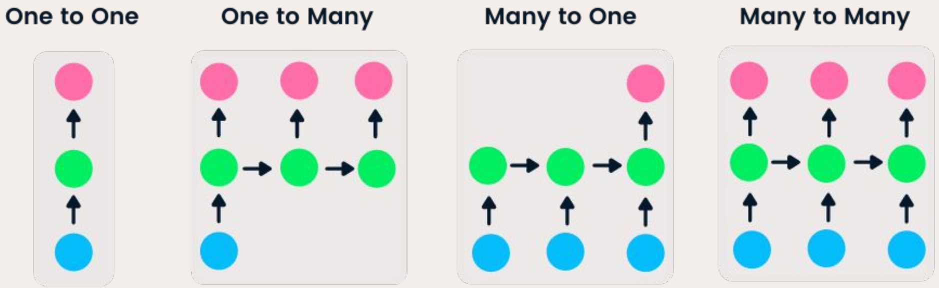


- These layers are in the last layer of the convolutional neural network, and their inputs correspond to the flattened one-dimensional matrix generated by the last pooling layer.
- ReLU activations functions are applied to them for non-linearity.
- They are used to flatten the 2D spatial structure of the data into a 1D vector and process this data for tasks like classification.
- The weights and biases in FC layers are learned during the training process, making them adapt to the specific problem at hand.
- The number of neurons in the final FC layer usually matches the number of output classes in a classification problem.
- Full connectivity with all neurons in the preceding and succeeding layer
- In this stage, the classification process begins to take place. The reason two layers are connected is that two fully connected layers will perform better than a single connected layer.


It's been 84 years,

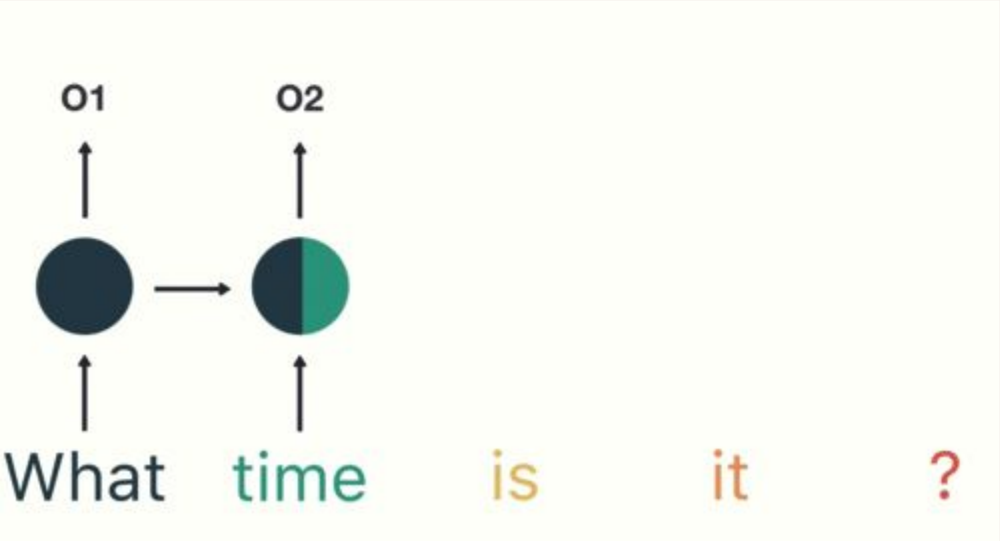
Recurrent Neural Networks (RNNs)

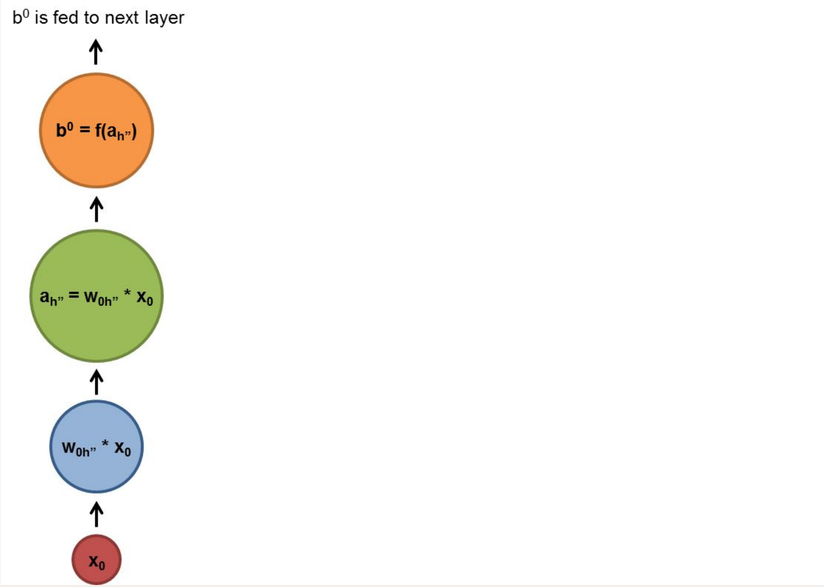
- A recurrent neural network (RNN) is the type of artificial neural network (ANN) that is used in Apple's Siri and Google's voice search.
- RNN remembers past inputs due to an internal memory which is useful for predicting stock prices, generating text, transcriptions, and machine translation.
- Weights are same, amongst all nodes, within a layer.
- They are distinguished by their "memory" as they take information from prior inputs to influence the current input and output.
- While traditional deep neural networks assume that inputs and outputs are independent of each other, the output of recurrent neural networks depend on the prior elements within the sequence.



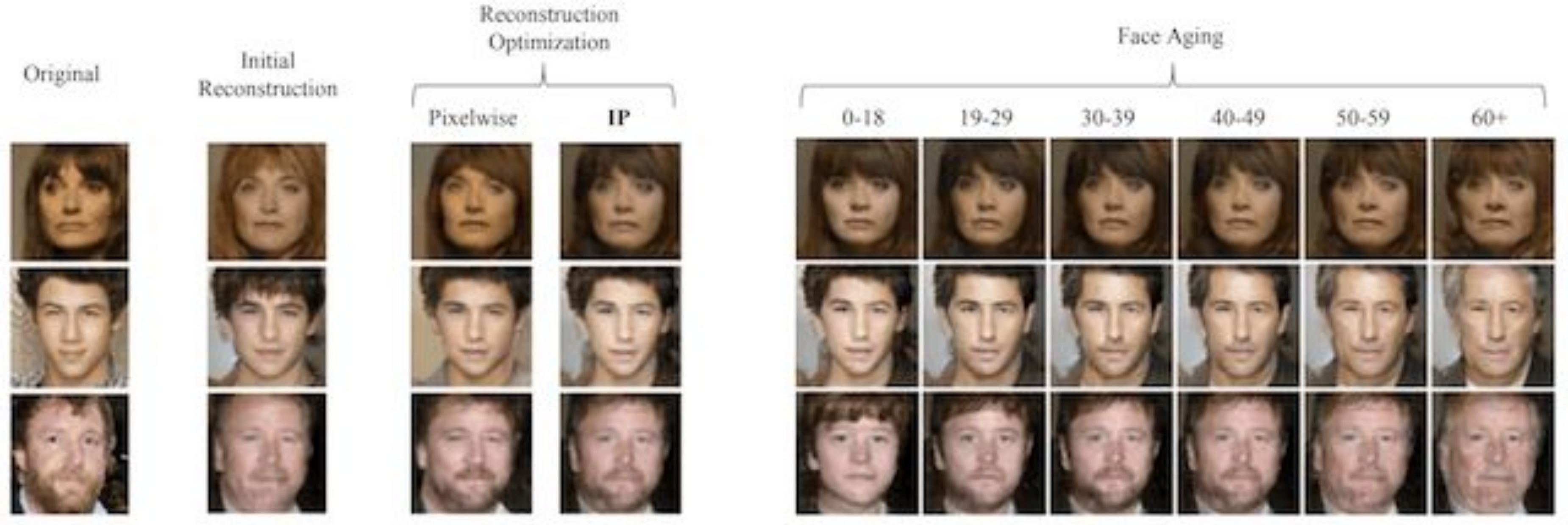
Recurrent Neural Networks (RNNs)

Recurrent Neural Networks (RNNs)



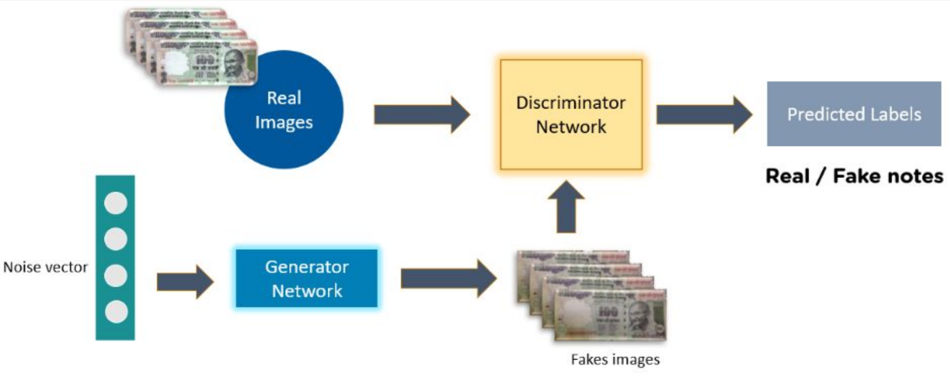


Generative Adversarial Networks (GANs)

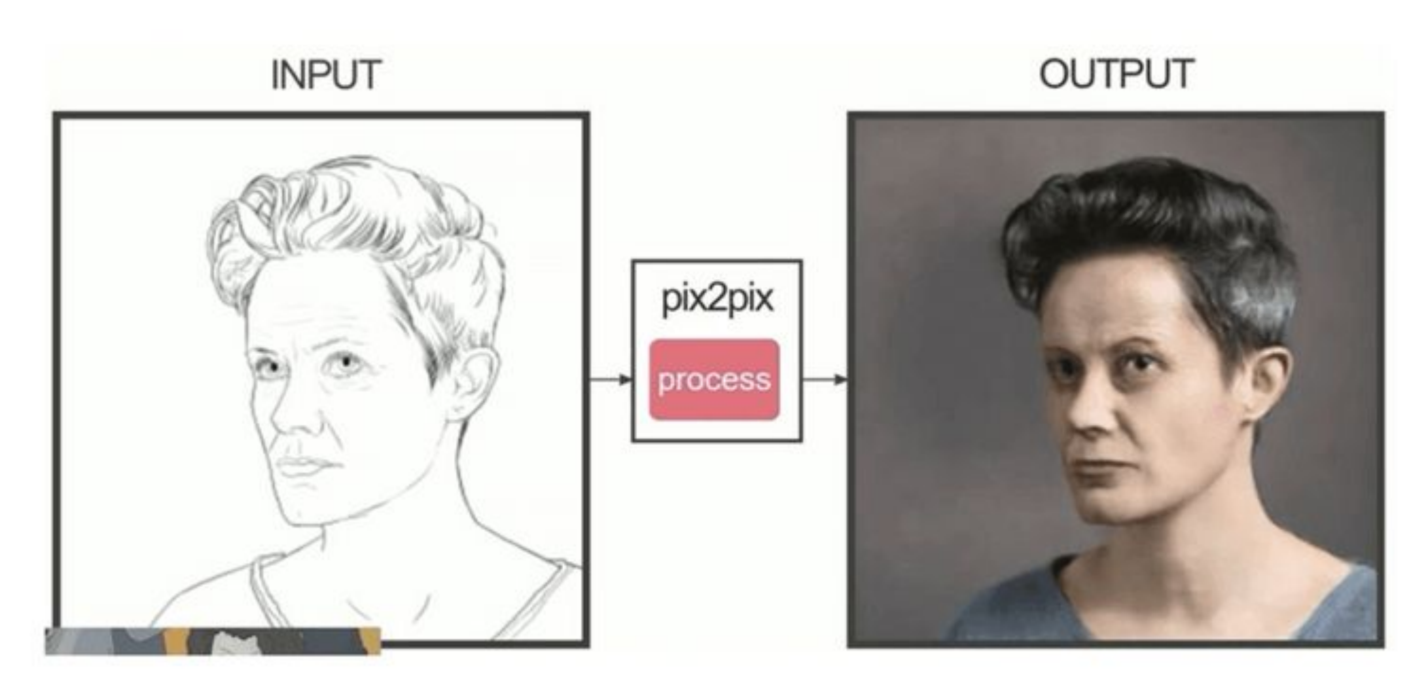


Generative Adversarial Networks (GANs)

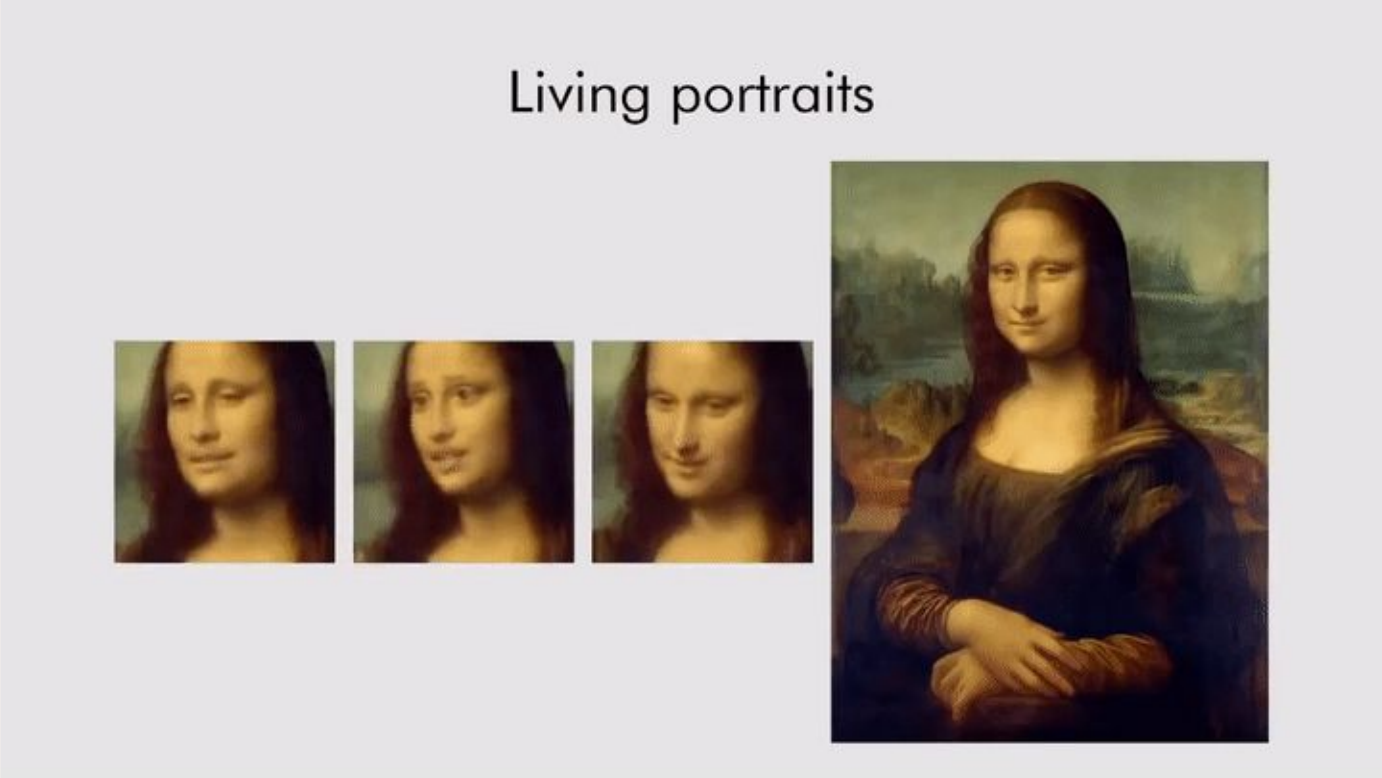
- Generative Adversarial Networks (GANs) were introduced in 2014 by Ian J. Goodfellow and co-authors. GANs perform unsupervised learning tasks in machine learning. It consists of 2 models that automatically discover and learn the patterns in input data.
- The two models are known as Generator and Discriminator.



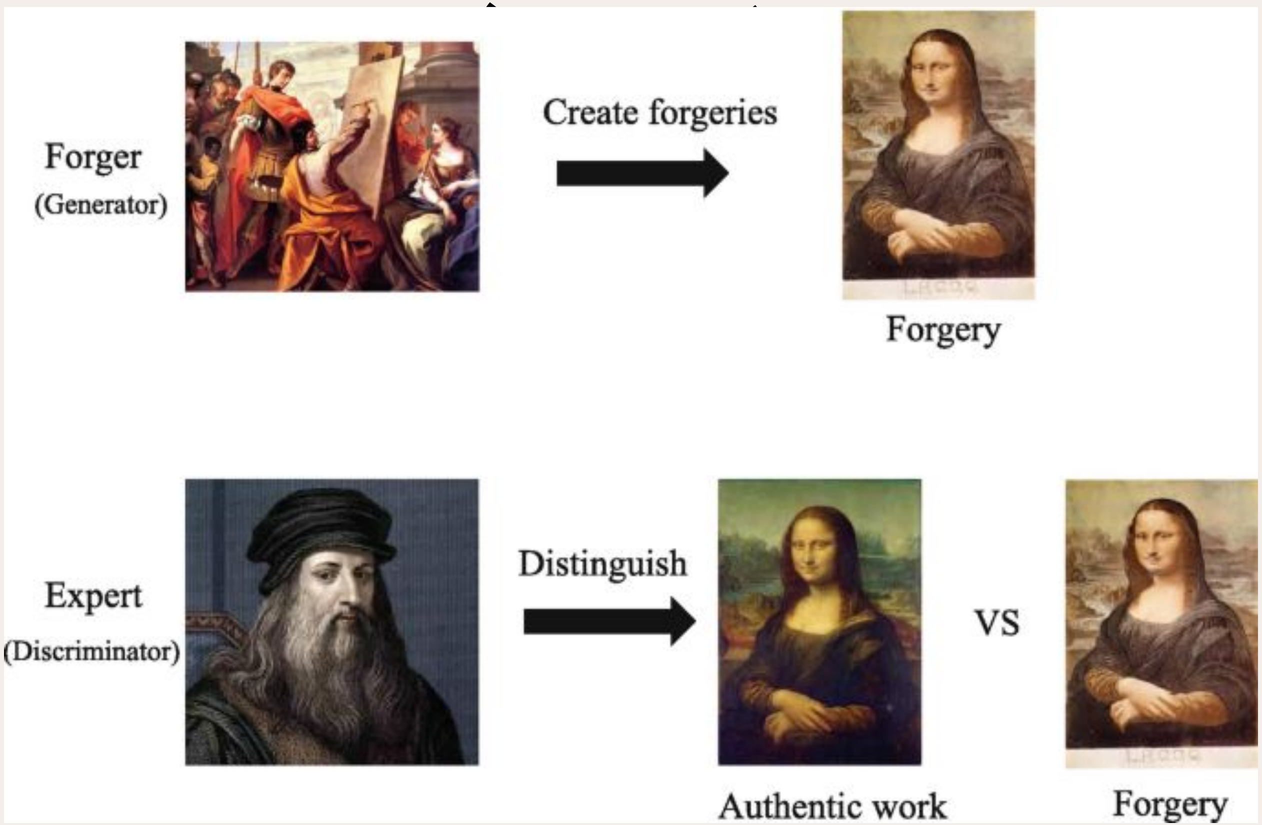
Generative Adversarial Networks (GANs)



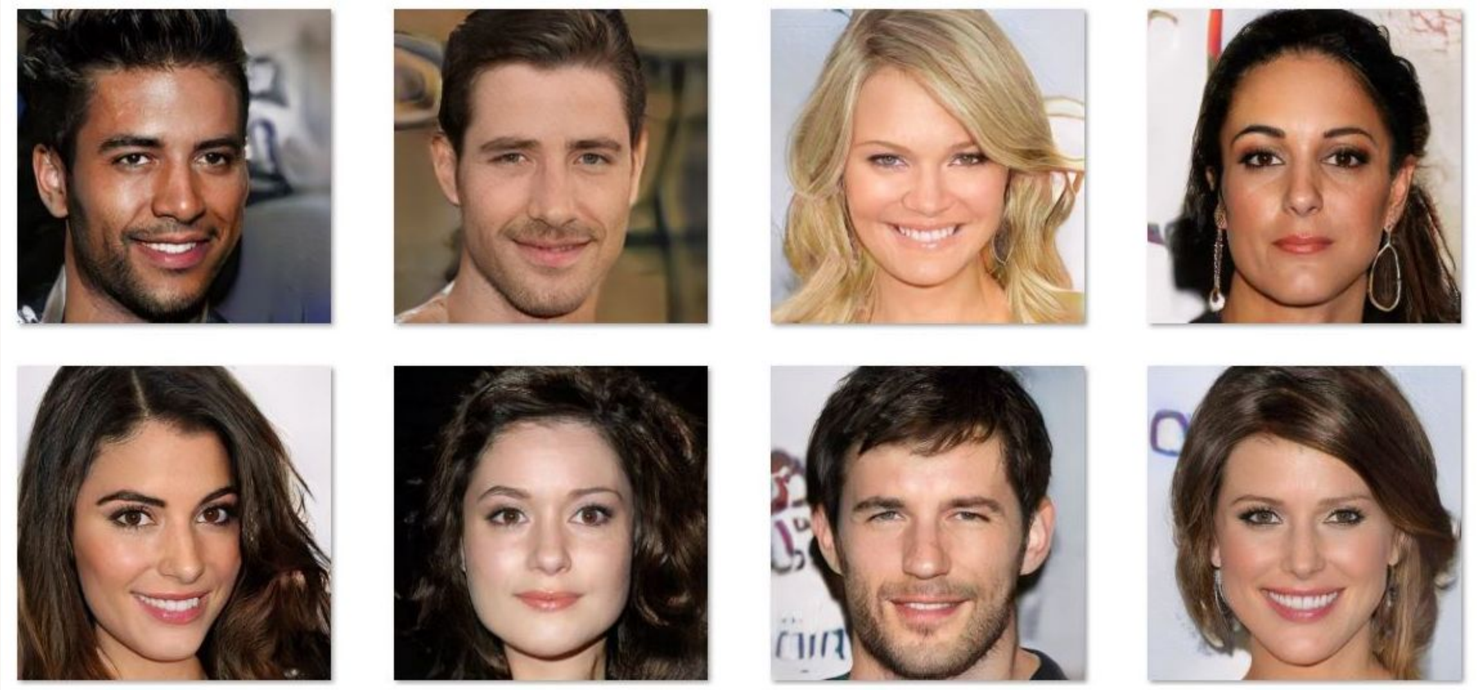
Generative Adversarial Networks (GANs)



Generative Adversarial Networks



Generative Adversarial Networks (GANs)



Generative Adversarial Networks (GANs)