Convolutional Neural Networks

Lecture-5

Source: Cassidy Williams, Software Engineer and Developer Evangelist at Clarifai, spoke at FirstMark's Code Driven NYC on April 13, 2016.

Machine Learning

"A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E."

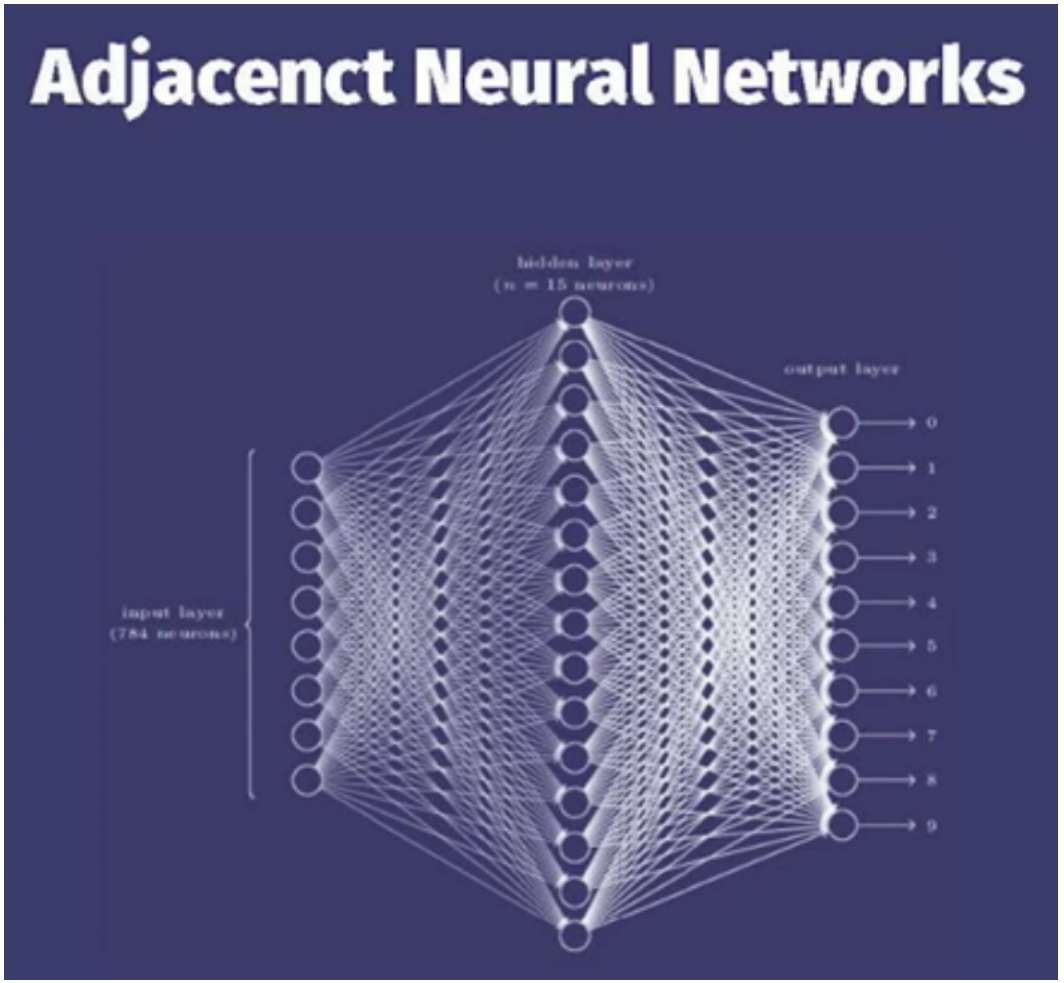
Tom Mitchell, 1997

Machine Learning

- Want my computer to learn where the bananas are (T) in a bowl of fruits (E); Performance measure (P): color

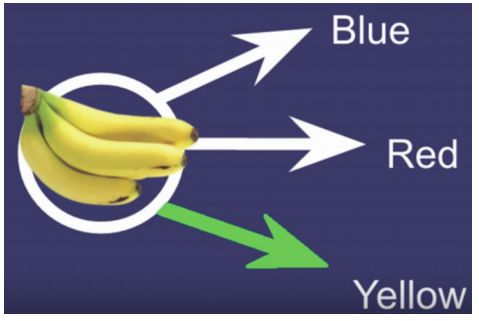
Neural Network

- Input layer: An image
- Hidden layer: Characteristics of the image (provides confidence)
- Output layer: Tags for that image

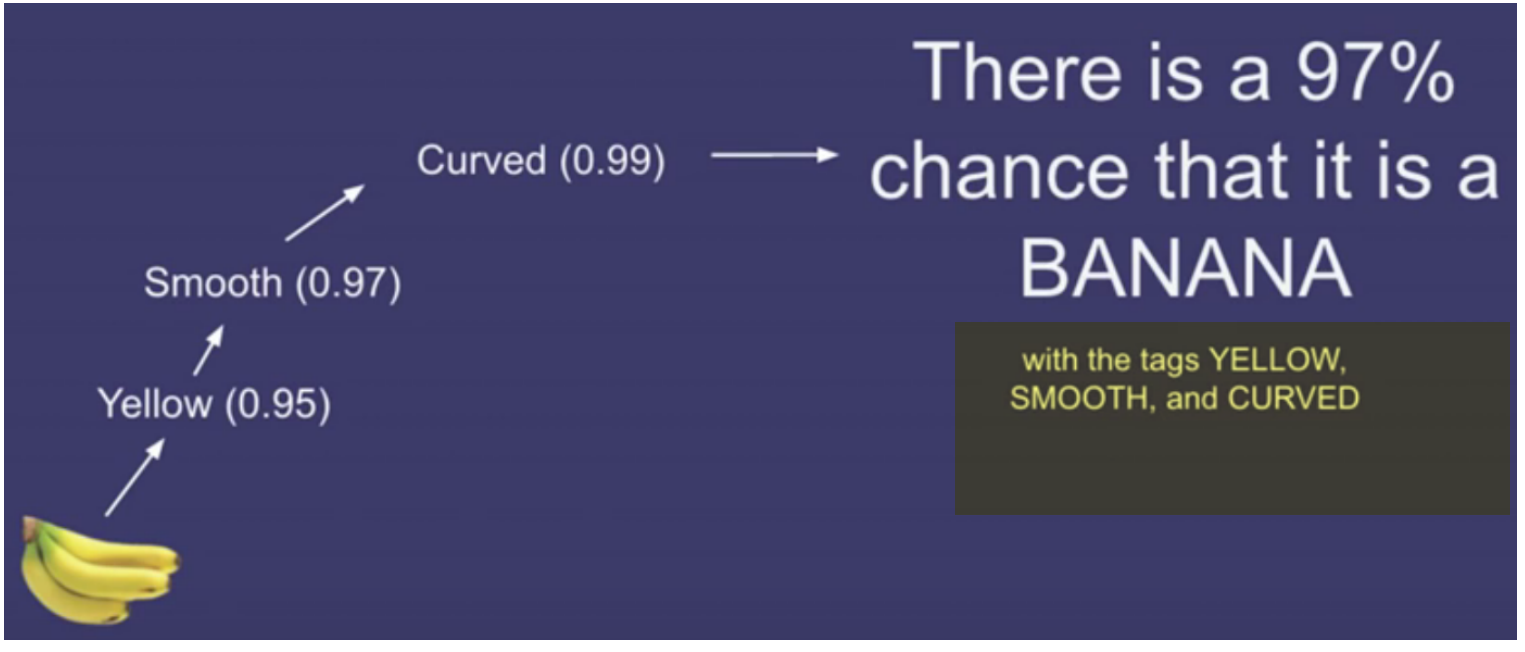


Neural Network

- Hidden layer: Color characteristics of the image (provides confidence)

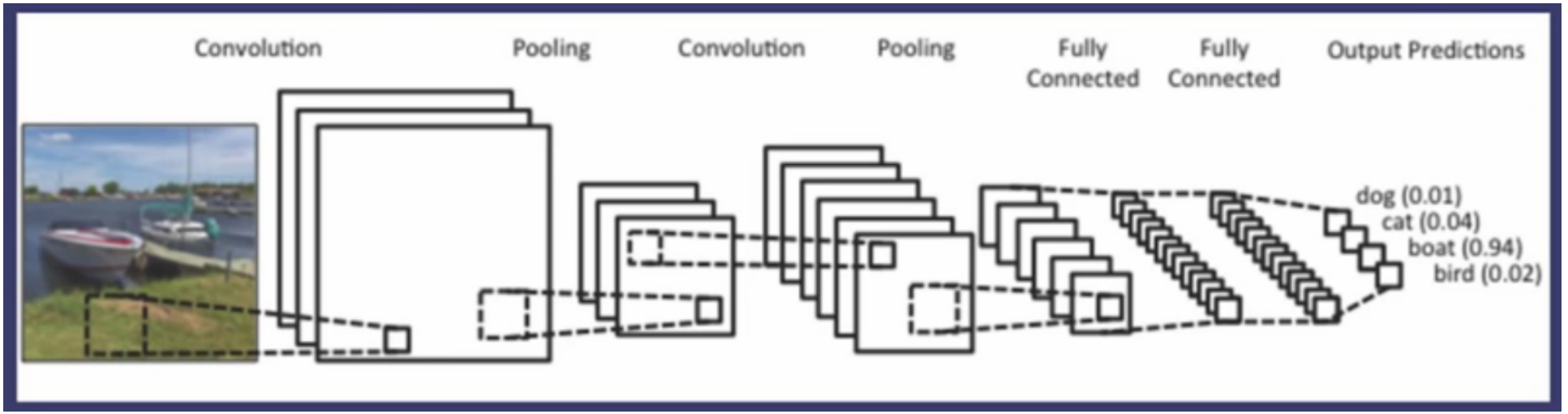


- **Cumulative confidence**



- **Tag**

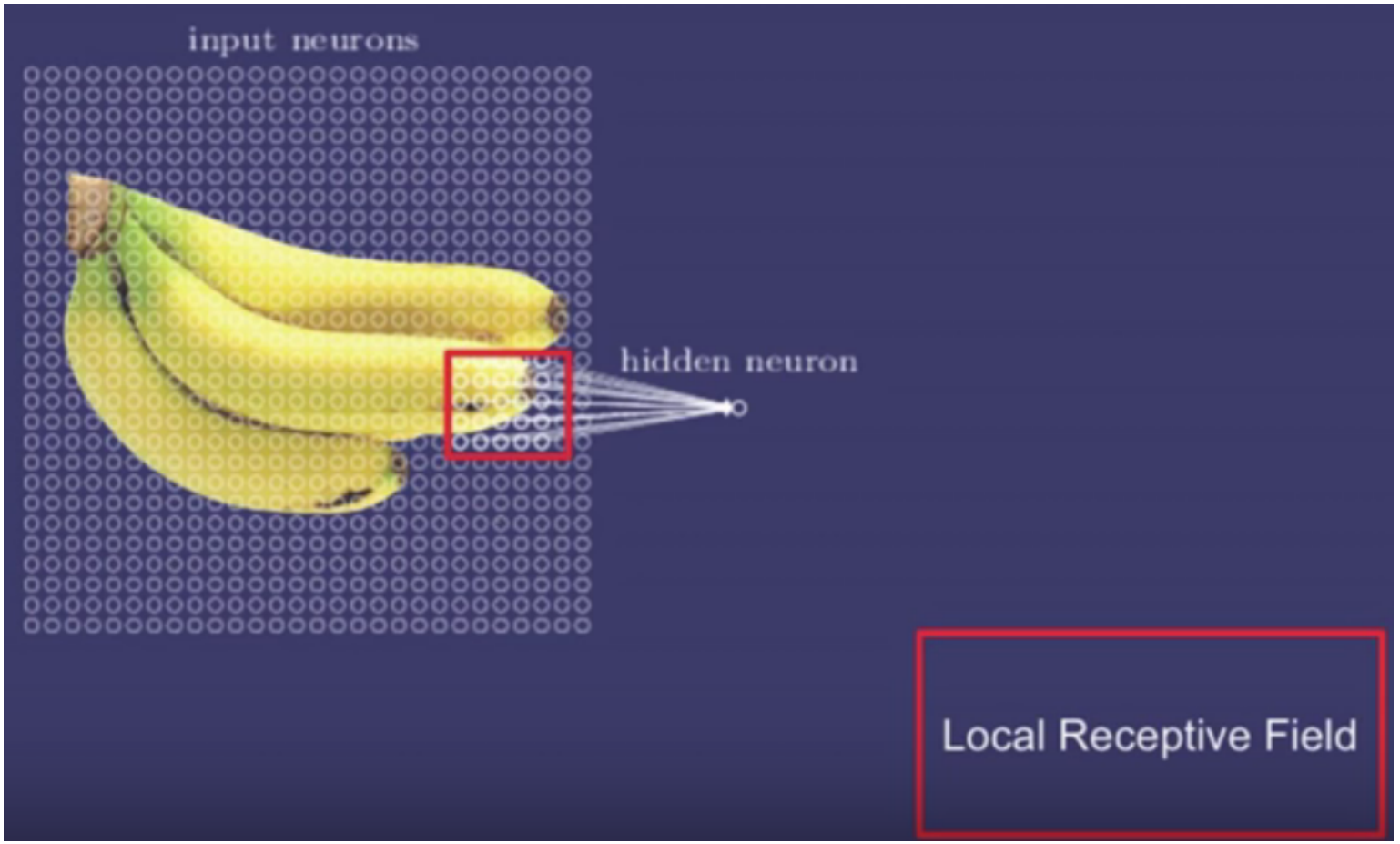
Convolution Neural Networks



Definitions:

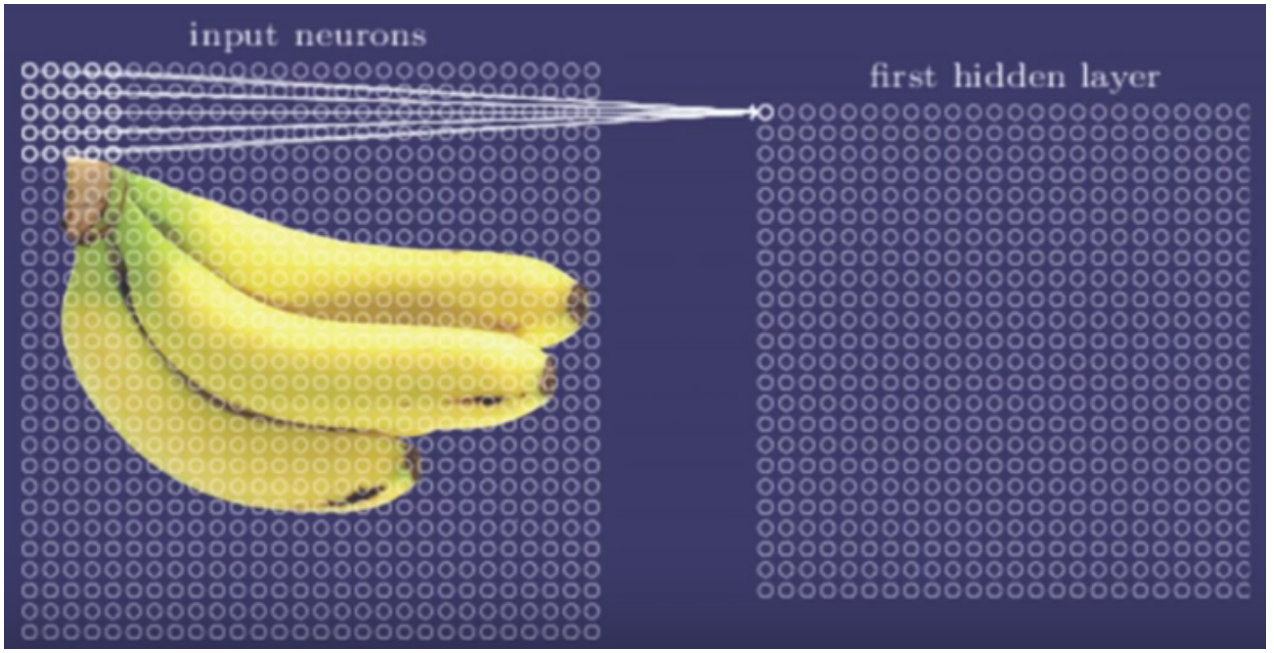
- **Local Receptive Fields**: a window on the input of pixels
- **Feature Map**: mapping from input layer to hidden layer
- **Shared Weights**: positive or negatives on a feature map
- **Pooling**: simplifying information from the feature map

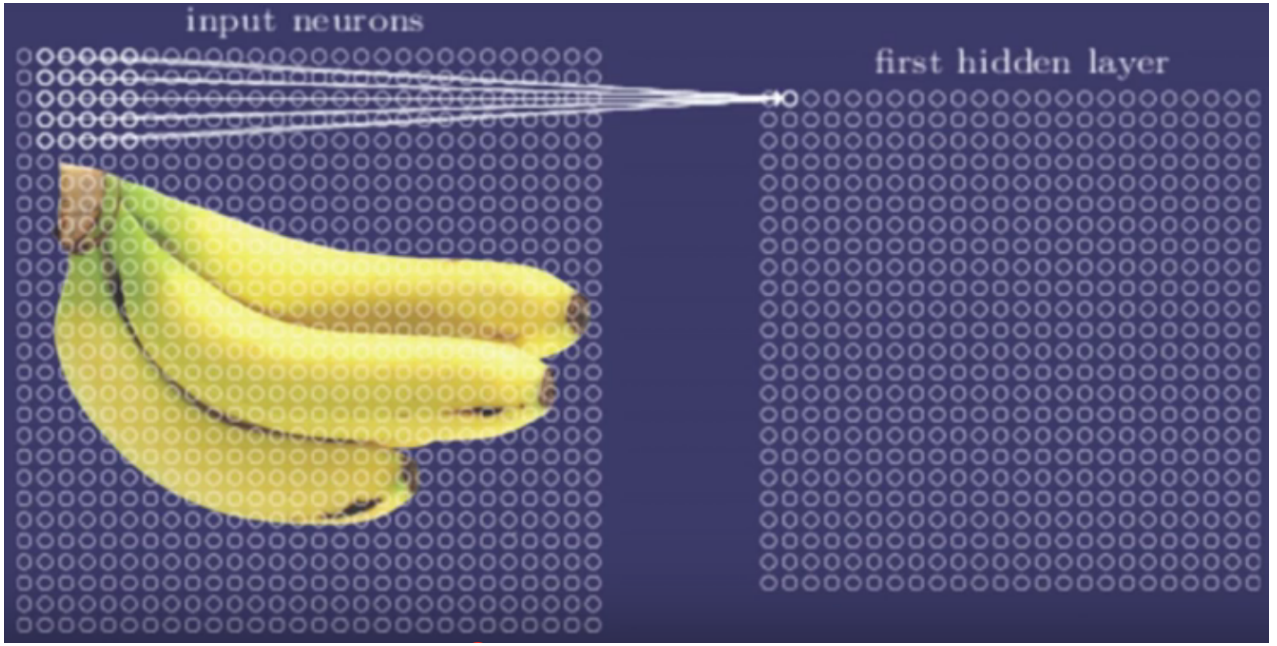
Convolution Neural Networks



- **Local Receptive Fields**: a window on the input of pixels

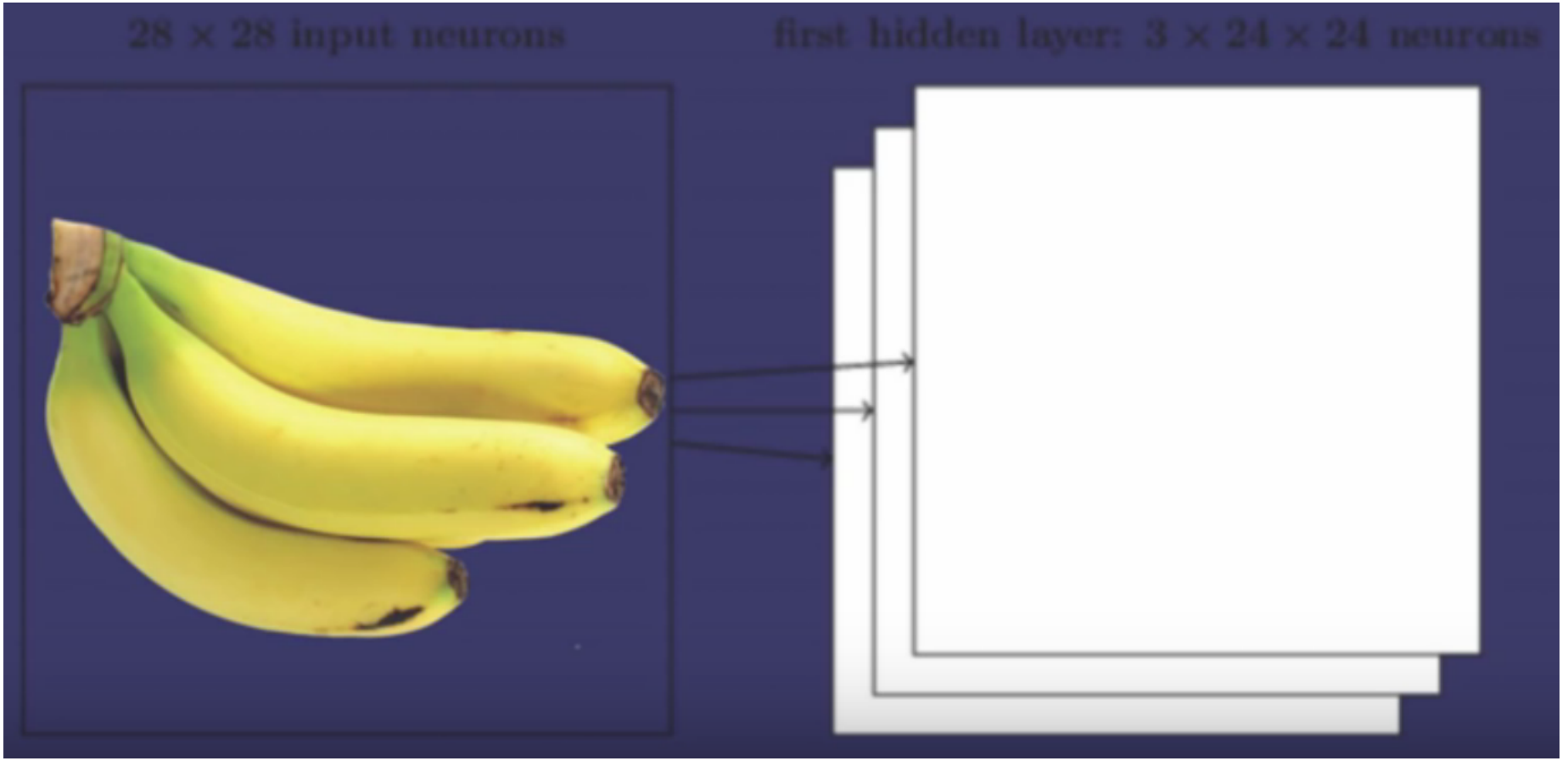
Convolution Neural Networks





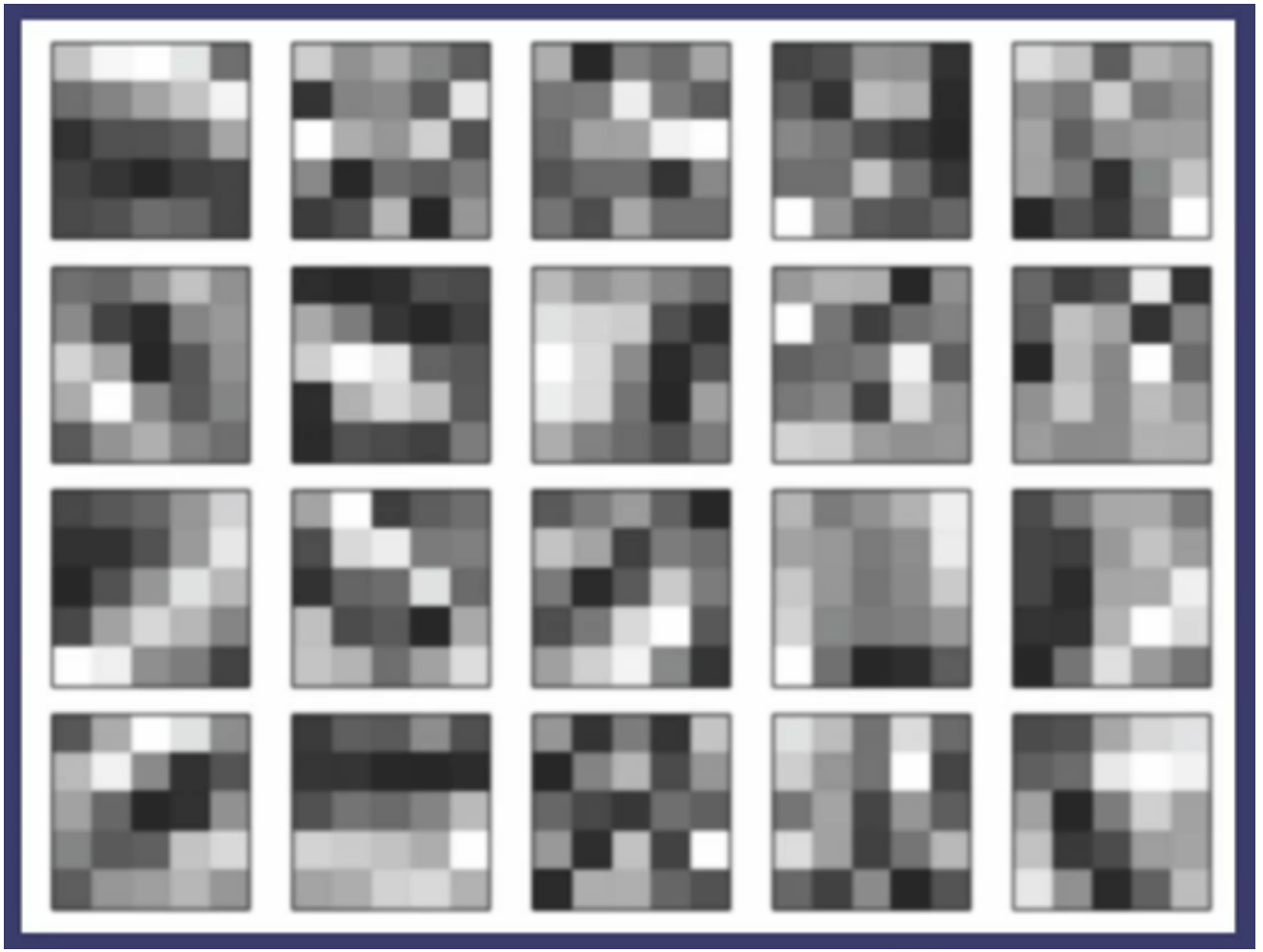
- **Feature Map**: (definition of one characteristic) mapping from input layer to hidden layer

Convolution Neural Networks



- **3 feature map (each feature map** defines of one characteristic) comprising 3 hidden layers, performing the convolution.

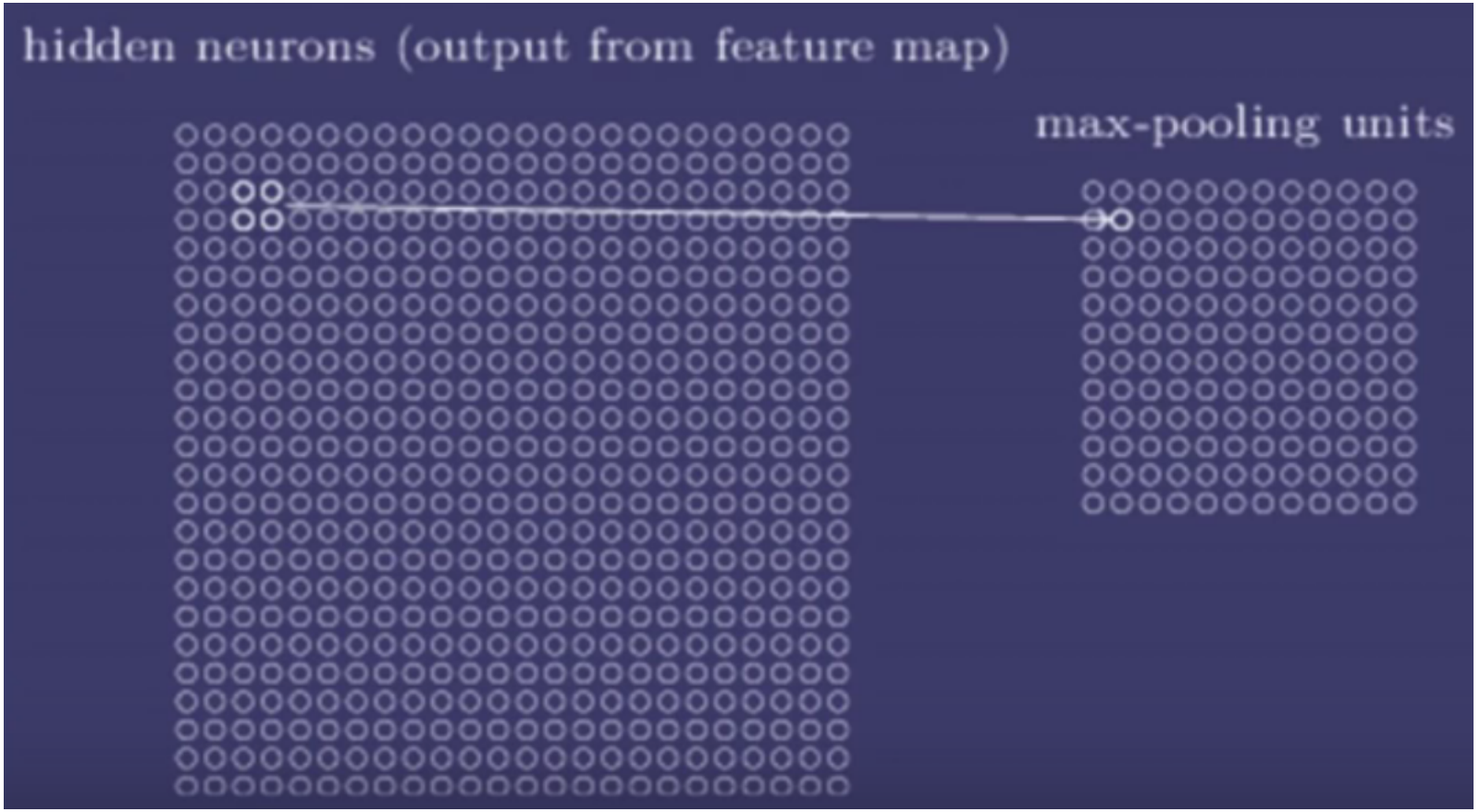
Convolution Neural Networks



Actual look of convolution:

- A convolution: a whole hidden layer
- Each box is a feature map; each square in each of the boxes is a nuron
- Darker represents more positive; lighter represents more negetive

Convolution Neural Networks



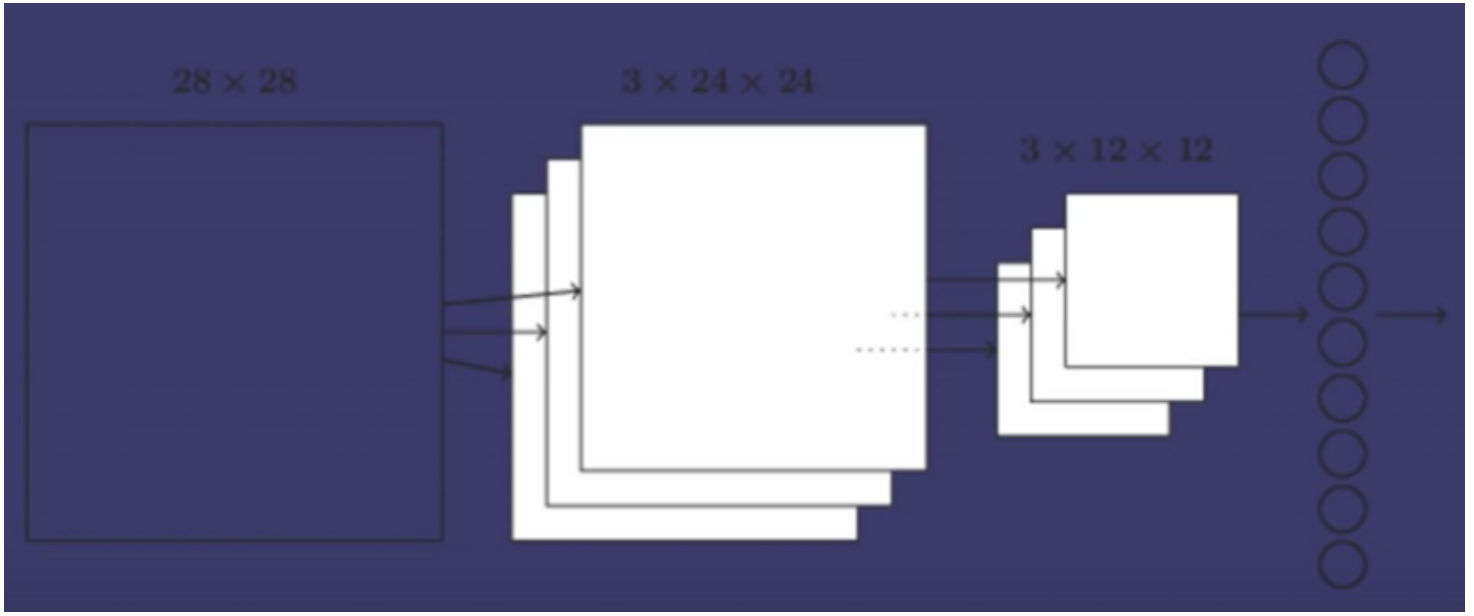
Pooling:

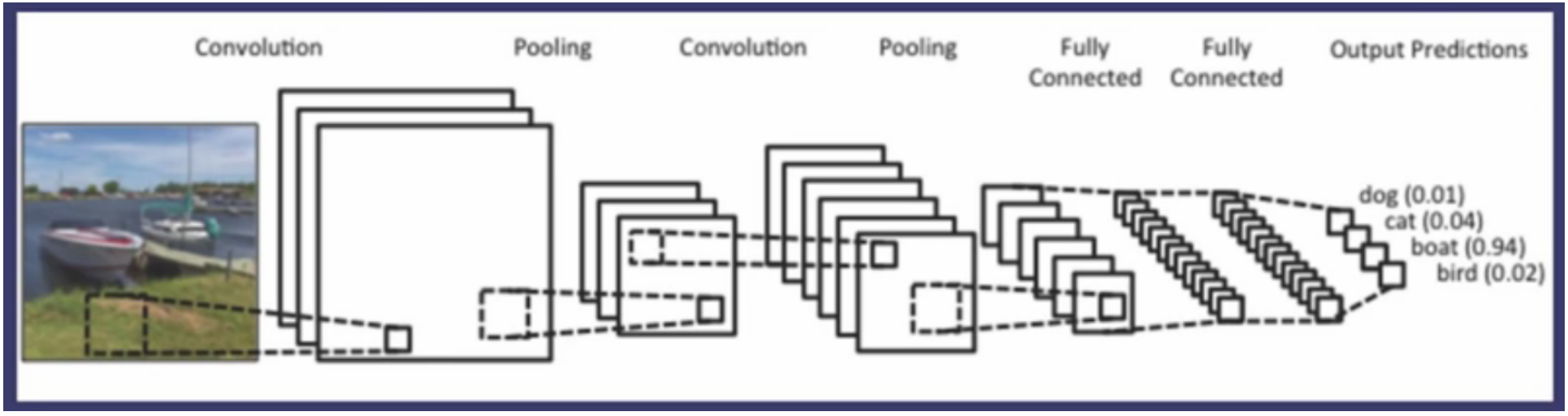
- simplifying information from the feature map
- Condensed all of the hidden neurons

Convolution Neural Networks

Overall:

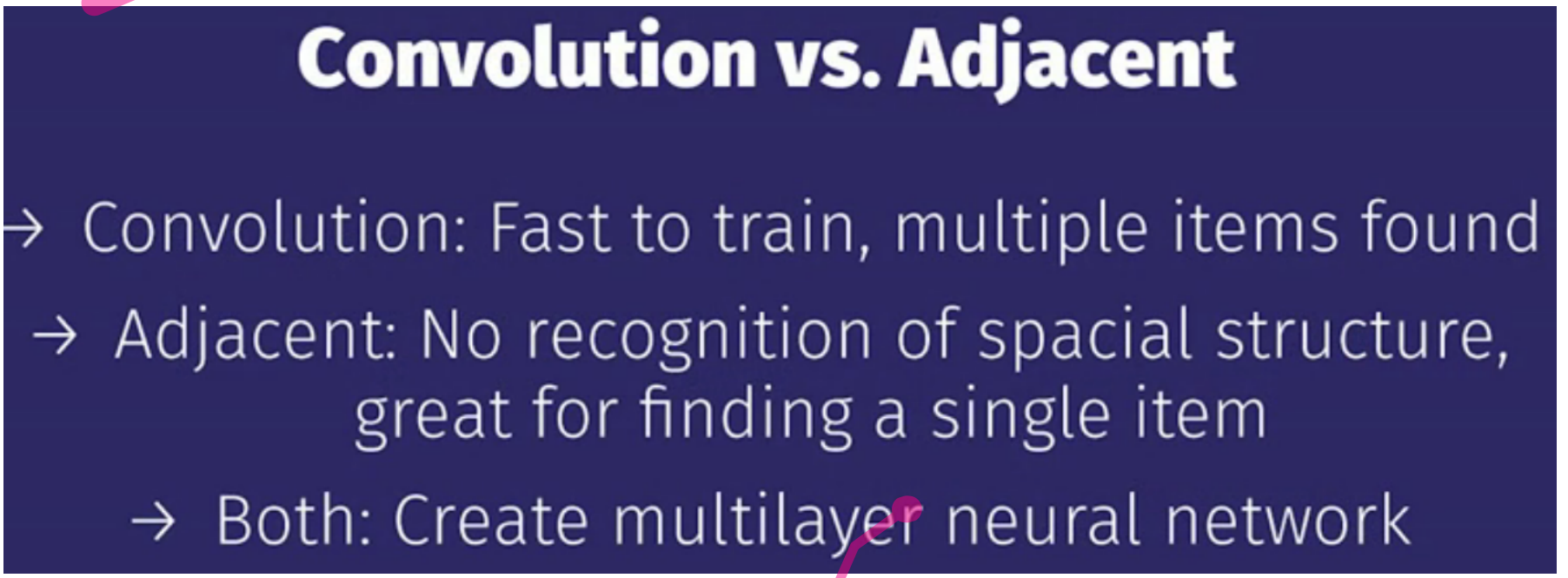
- Focusing on characteristics in different locations;
- working on smaller regions in separate thread, which is fast





Convolution Neural Networks

Summary:



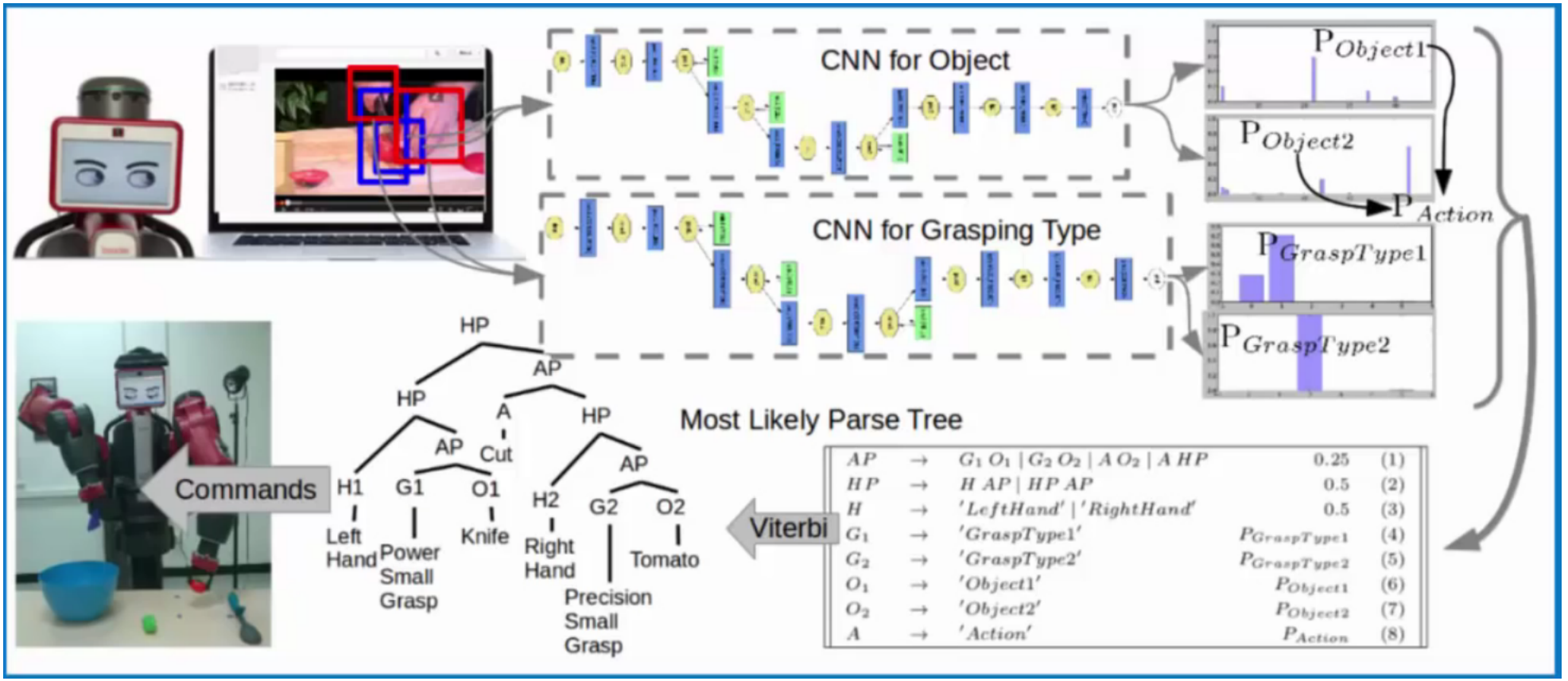
Convolution Neural Networks

Extra …

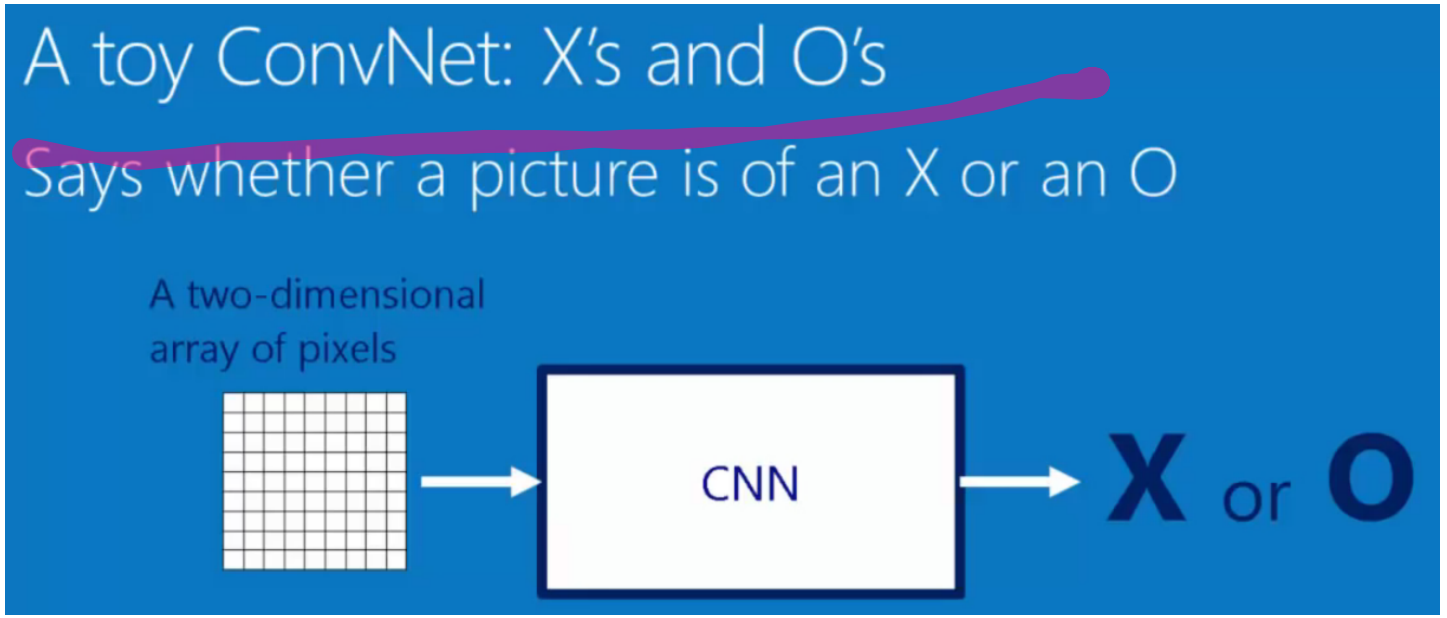
Source: Brandon Rohrer; http://brohrer.github.io/how_convolutional_neural_networks_work.html

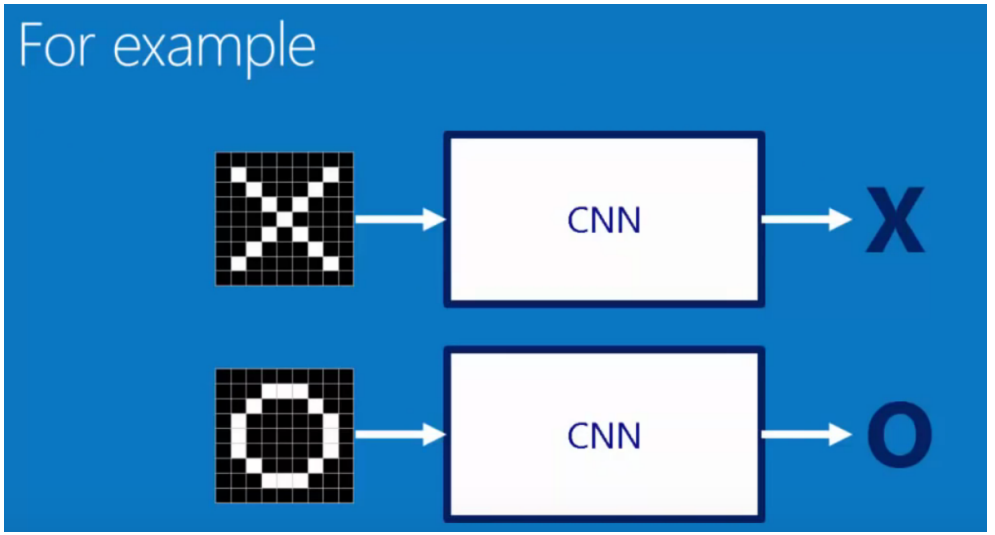
Convolution Neural Networks

- Robot learns to cook by watching youtube videos

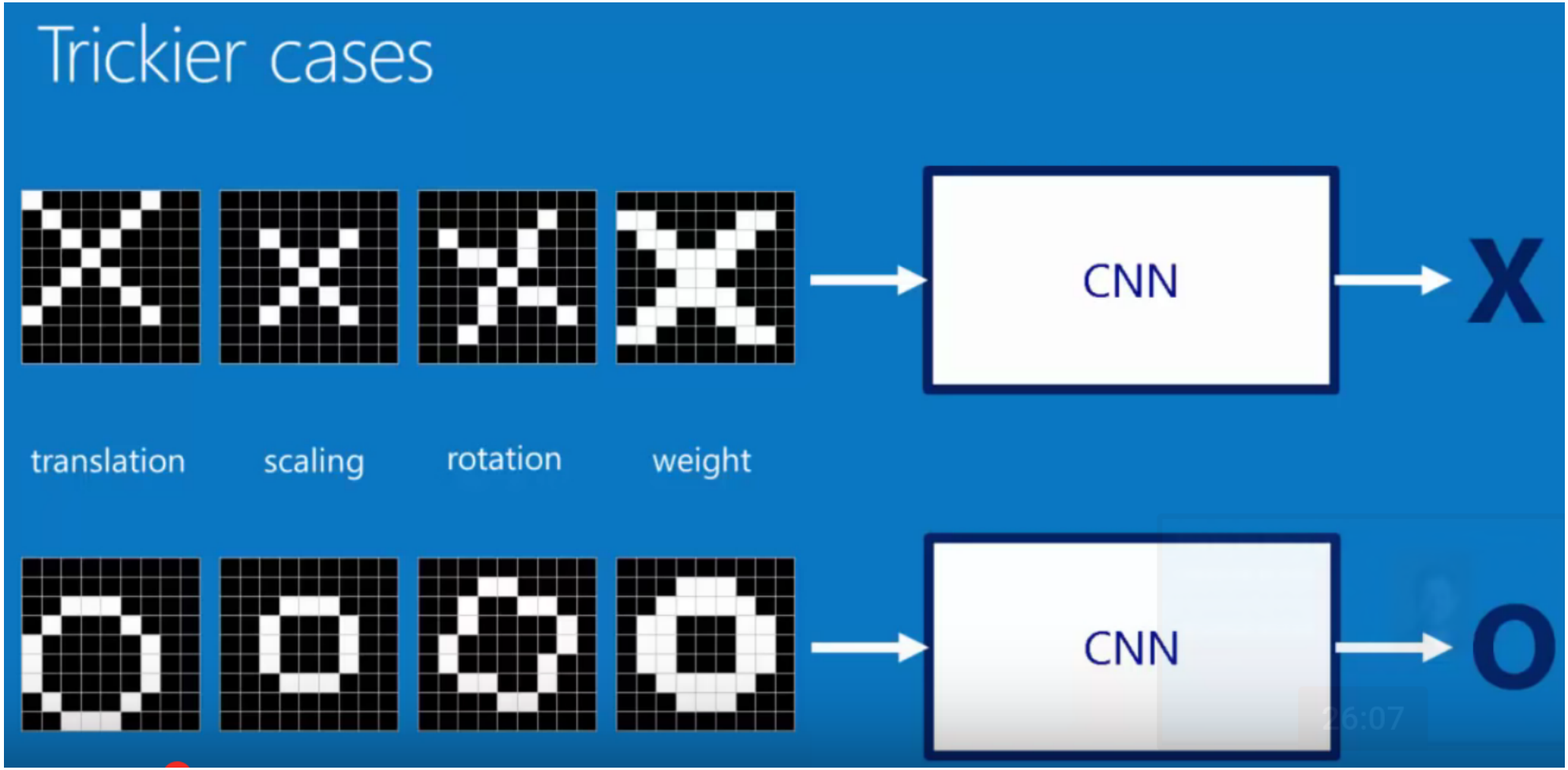


Convolution Neural Networks

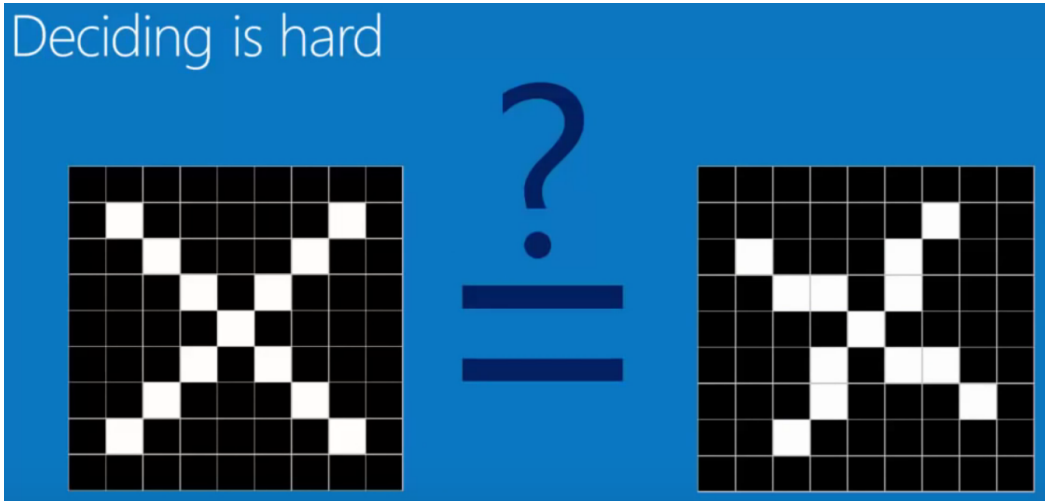


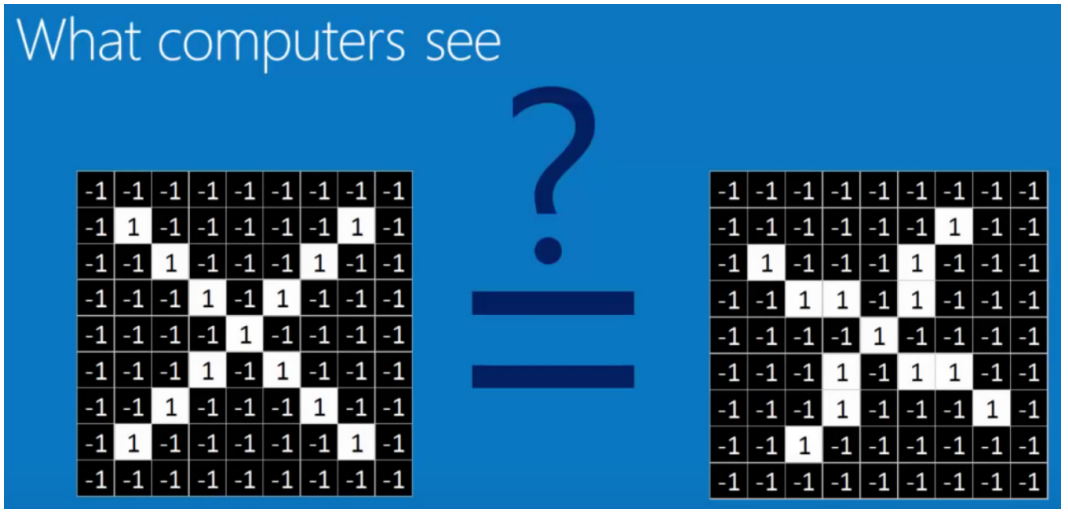


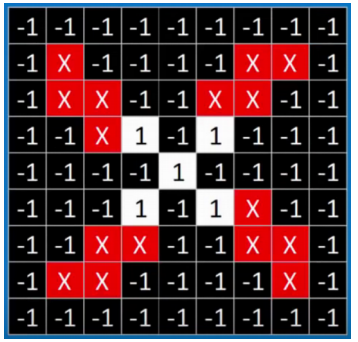
Convolution Neural Networks



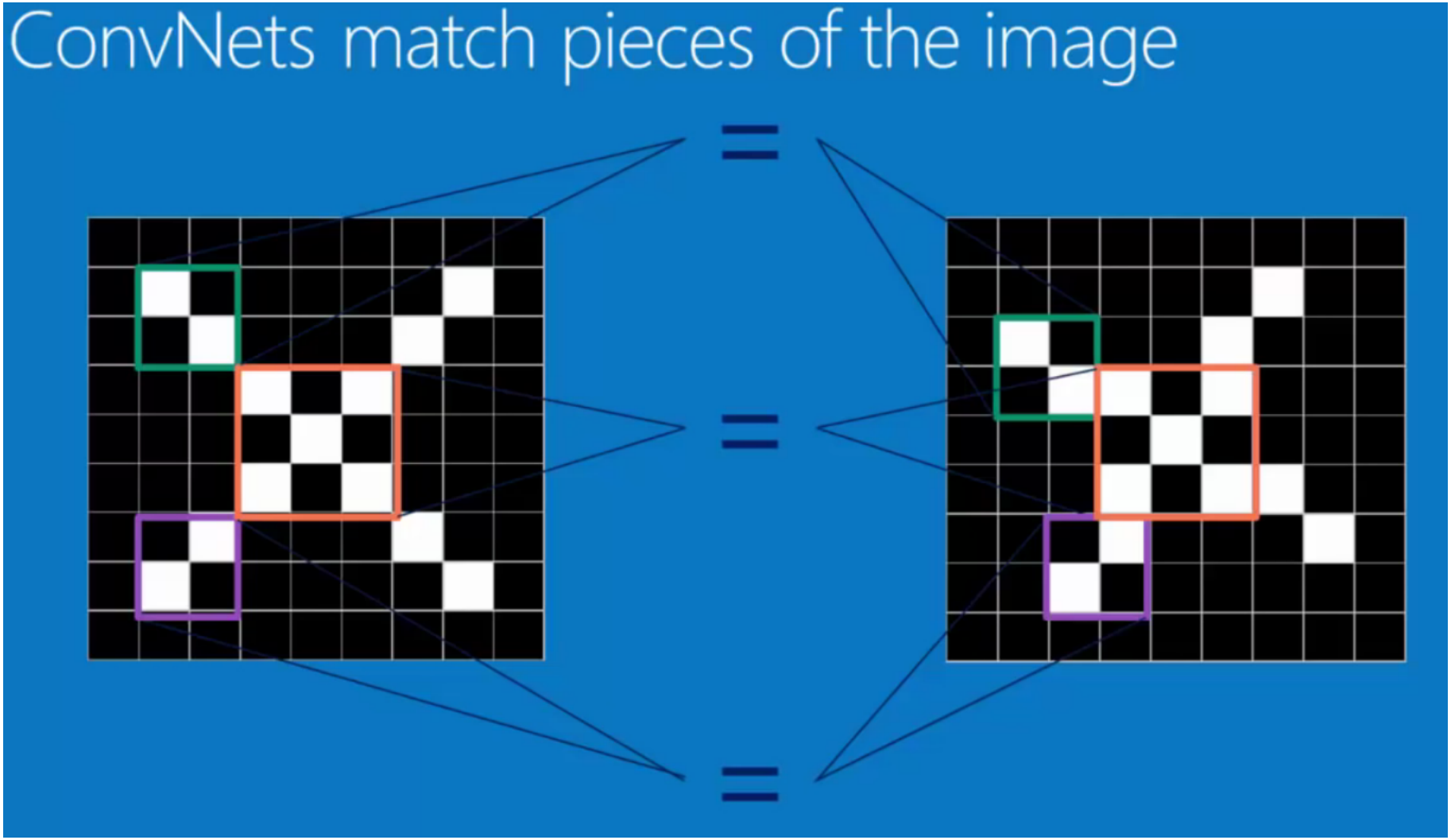
Convolution Neural Networks



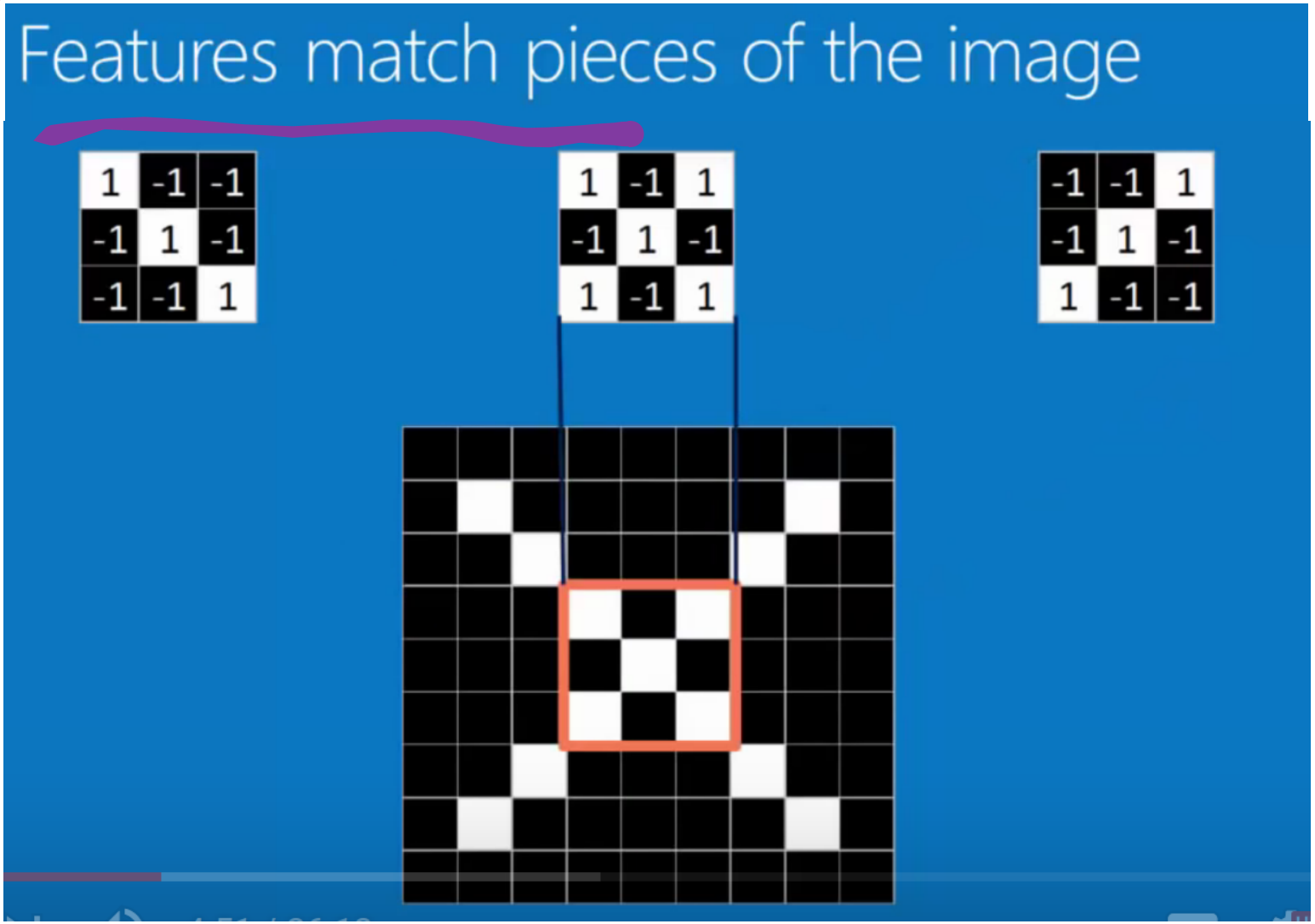




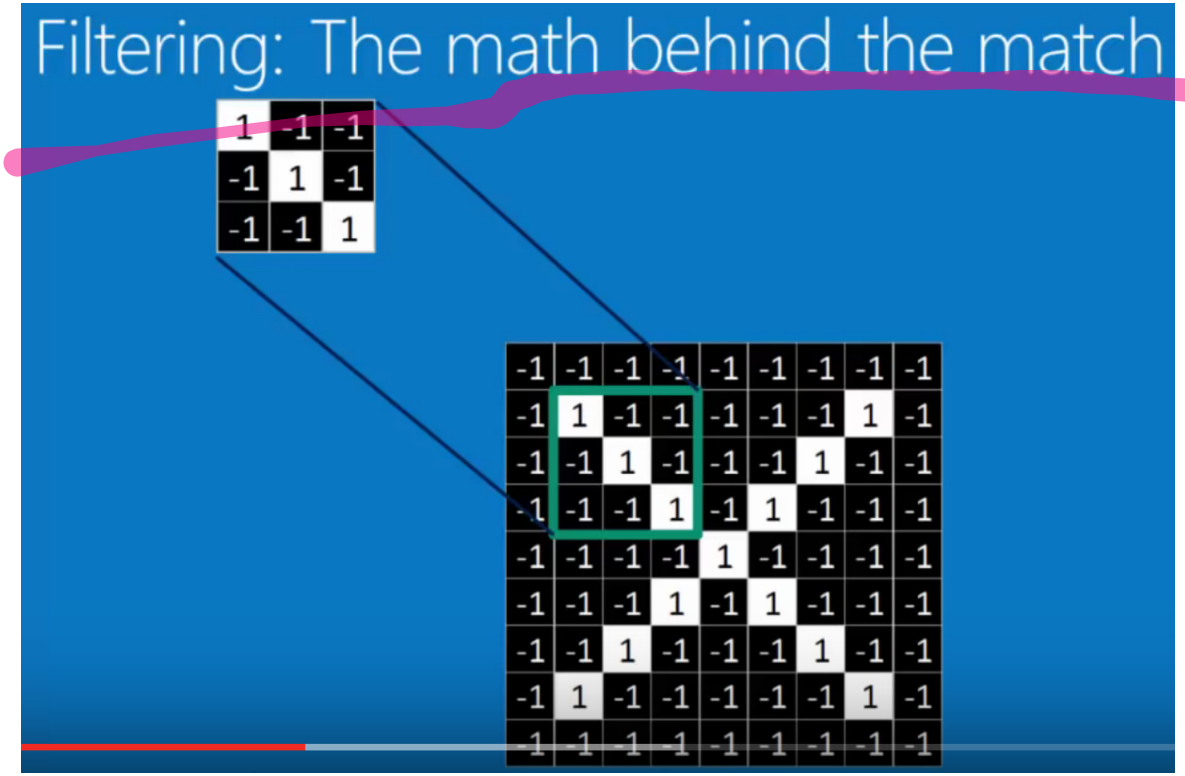
Convolution Neural Networks



Convolution Neural Networks



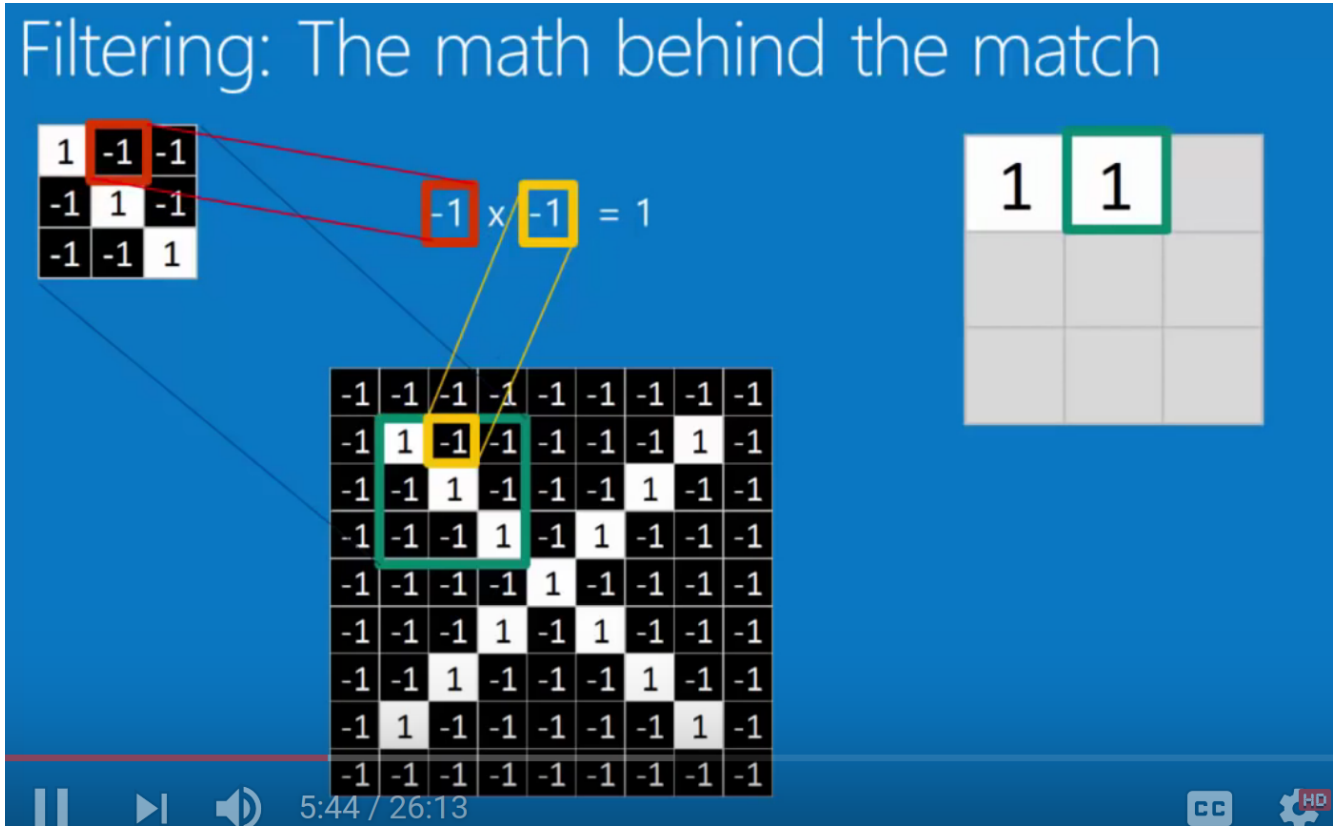
Convolution Neural Networks



Filtering: The math behind the match

1. Line up the feature and the image patch.
2. Multiply each image pixel by the corresponding feature pixel.
3. Add them up.
4. Divide by the total number of pixels in the feature.

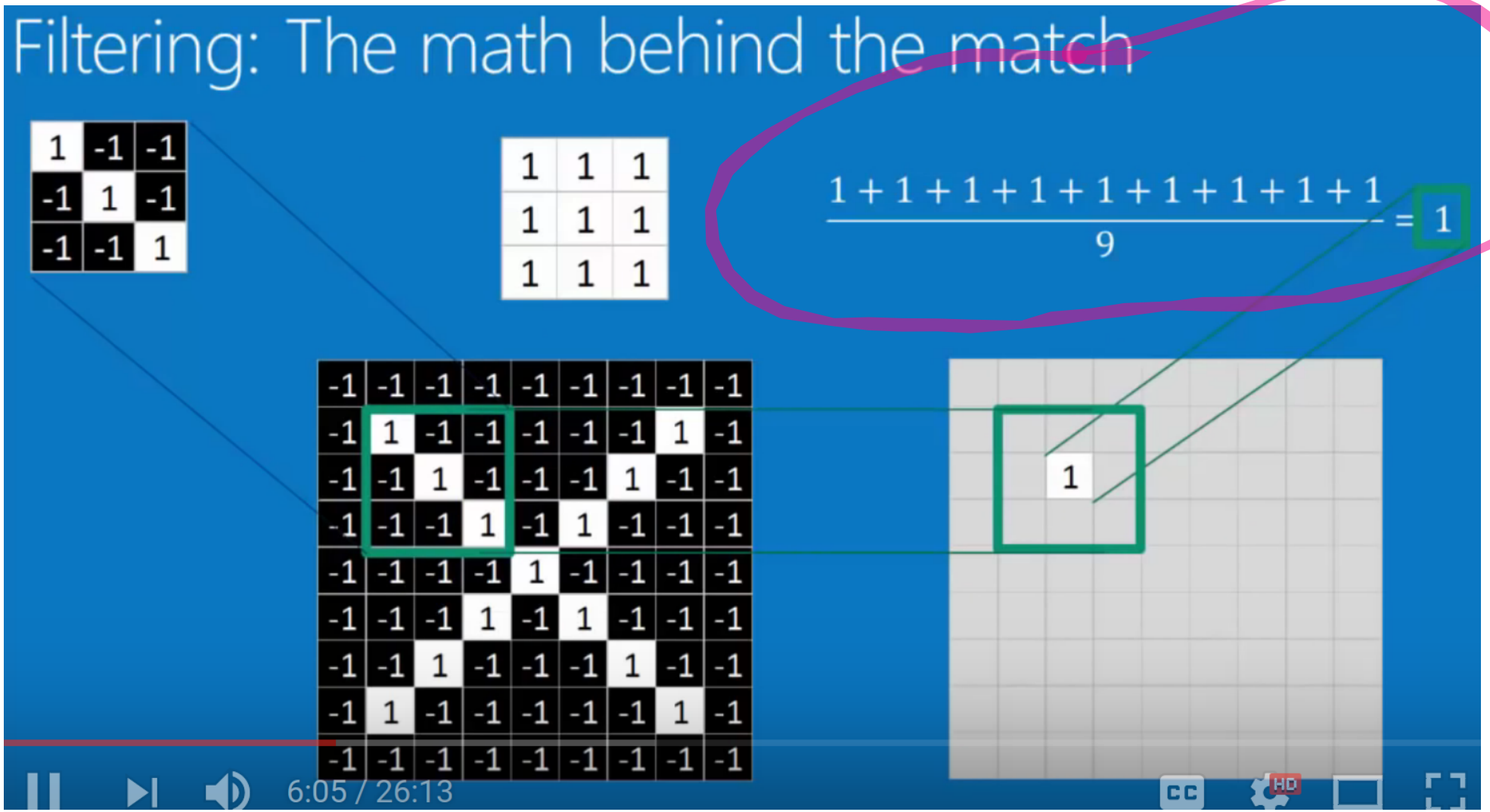
Convolution Neural Networks



Filtering: The math behind the match

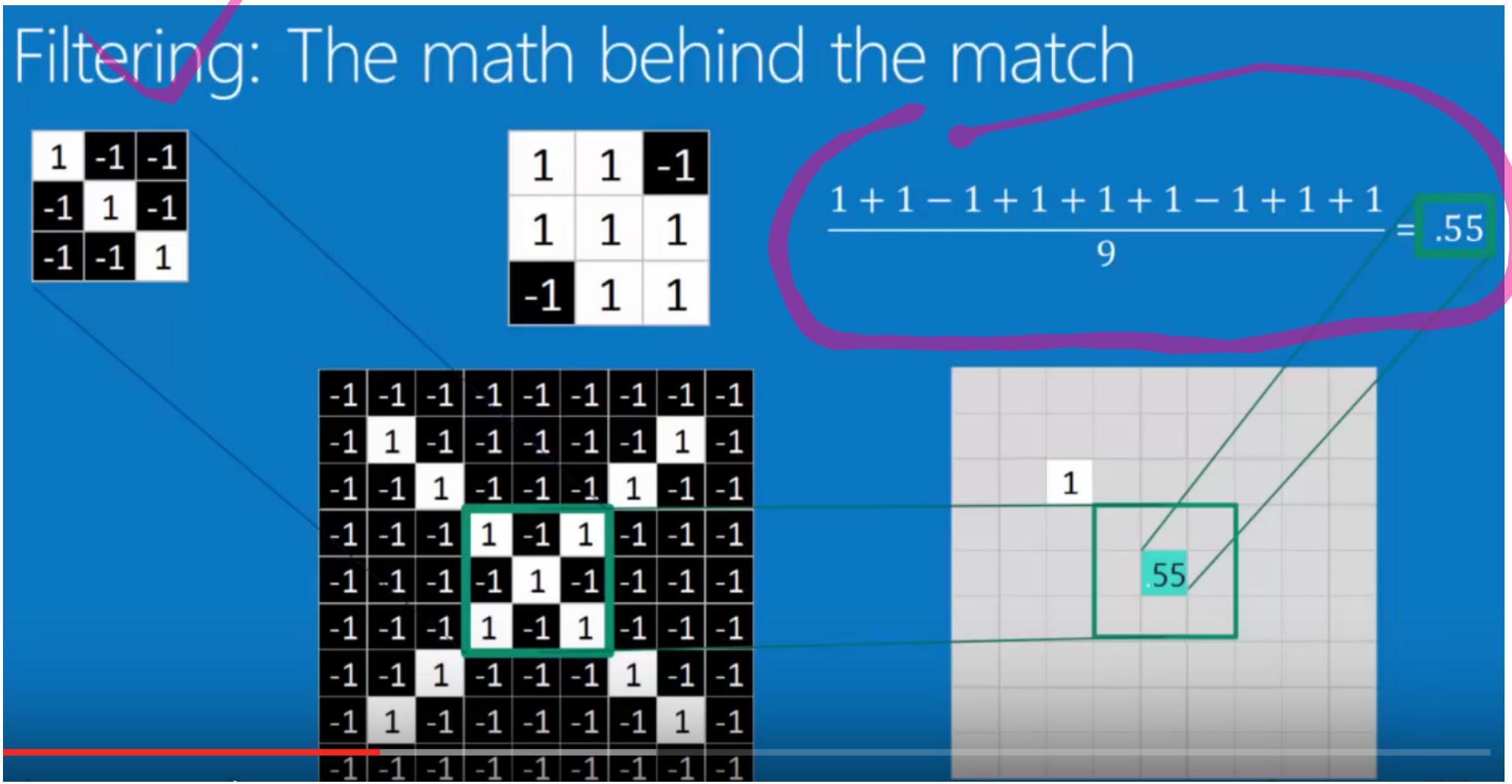
1. Line up the feature and the image patch.
2. Multiply each image pixel by the corresponding feature pixel.
3. Add them up.
4. Divide by the total number of pixels in the feature.

Convolution Neural Networks



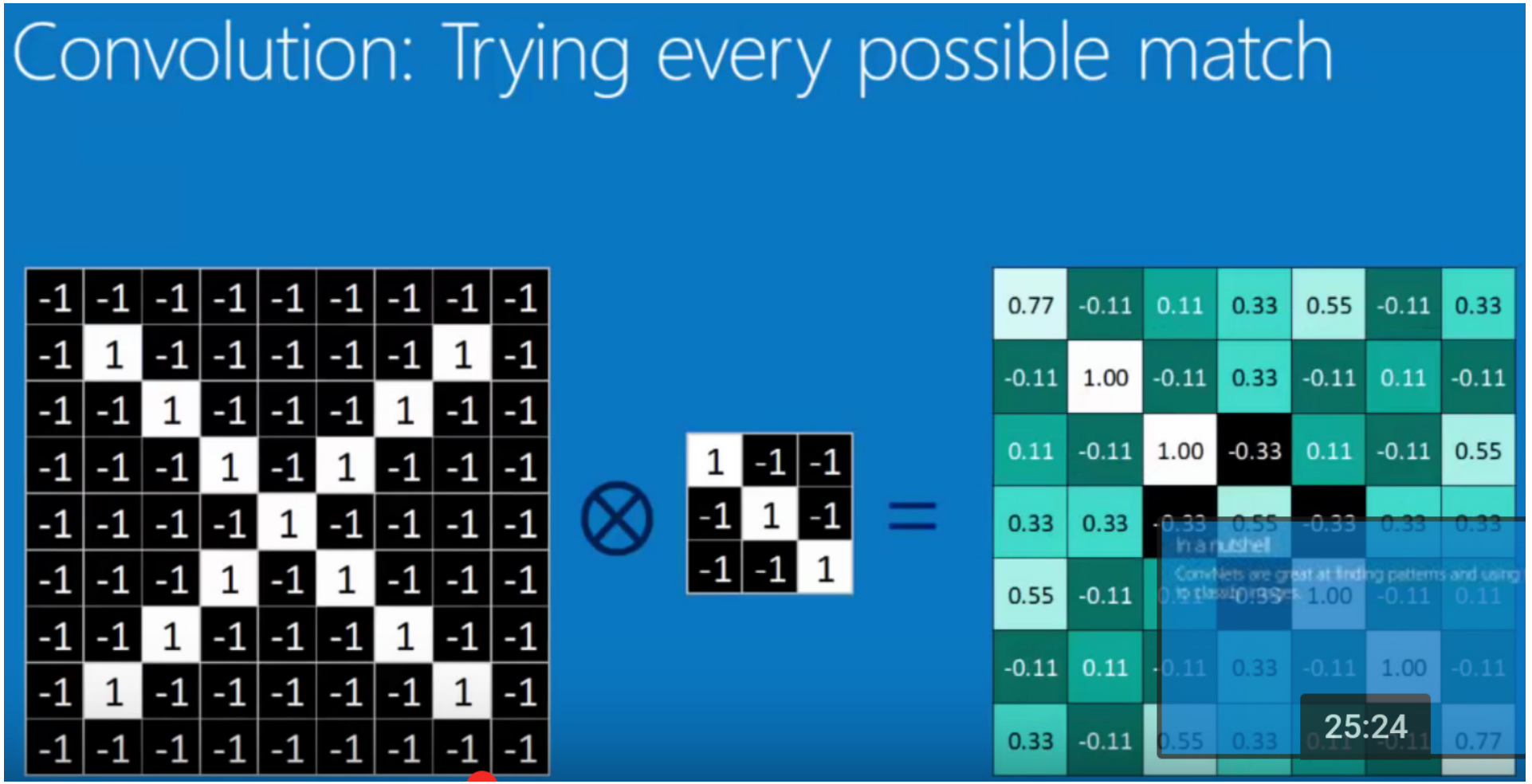
- Convoluton: feature map

Convolution Neural Networks



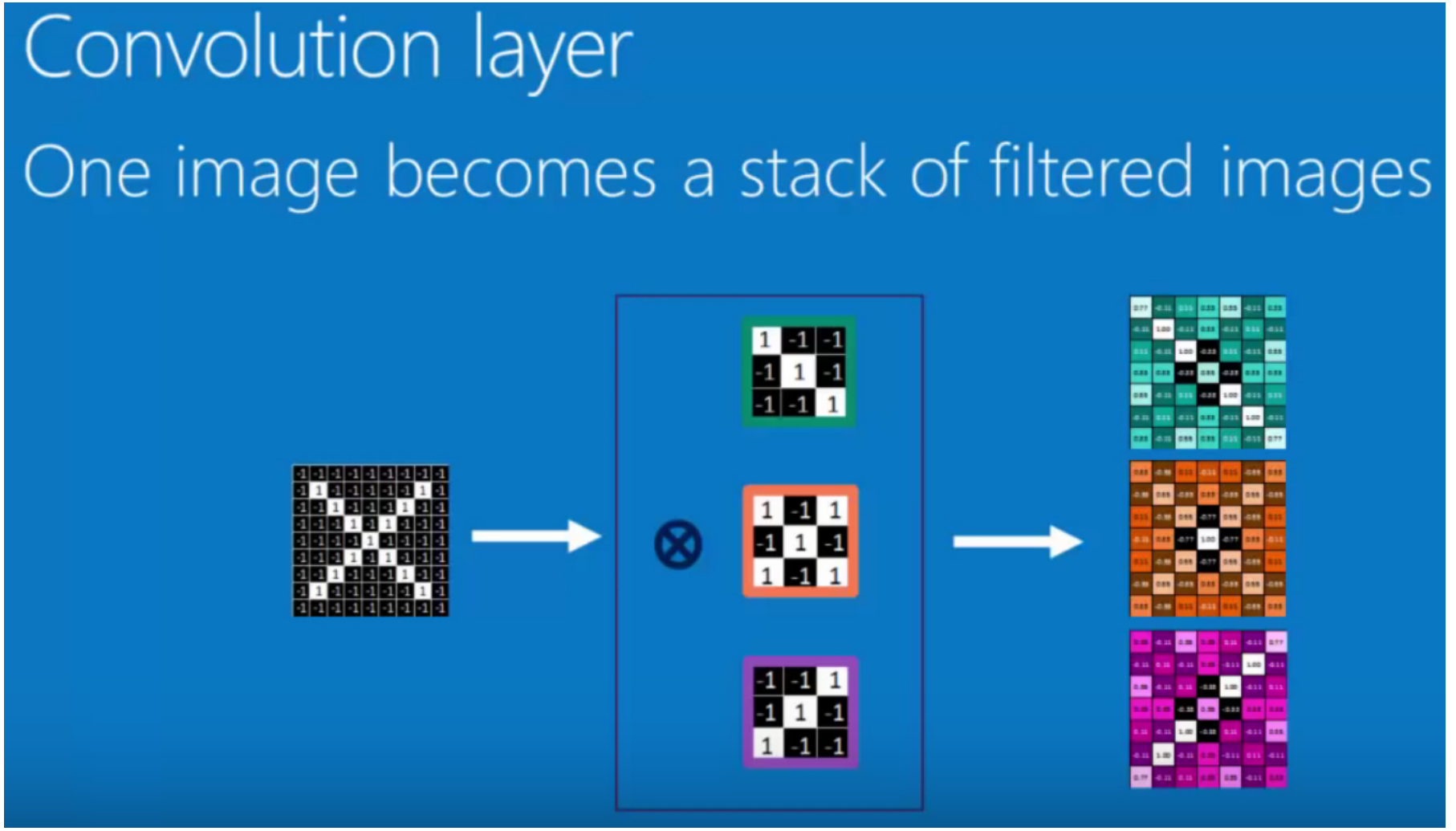
- Convoluton: feature map

Convolution Neural Networks

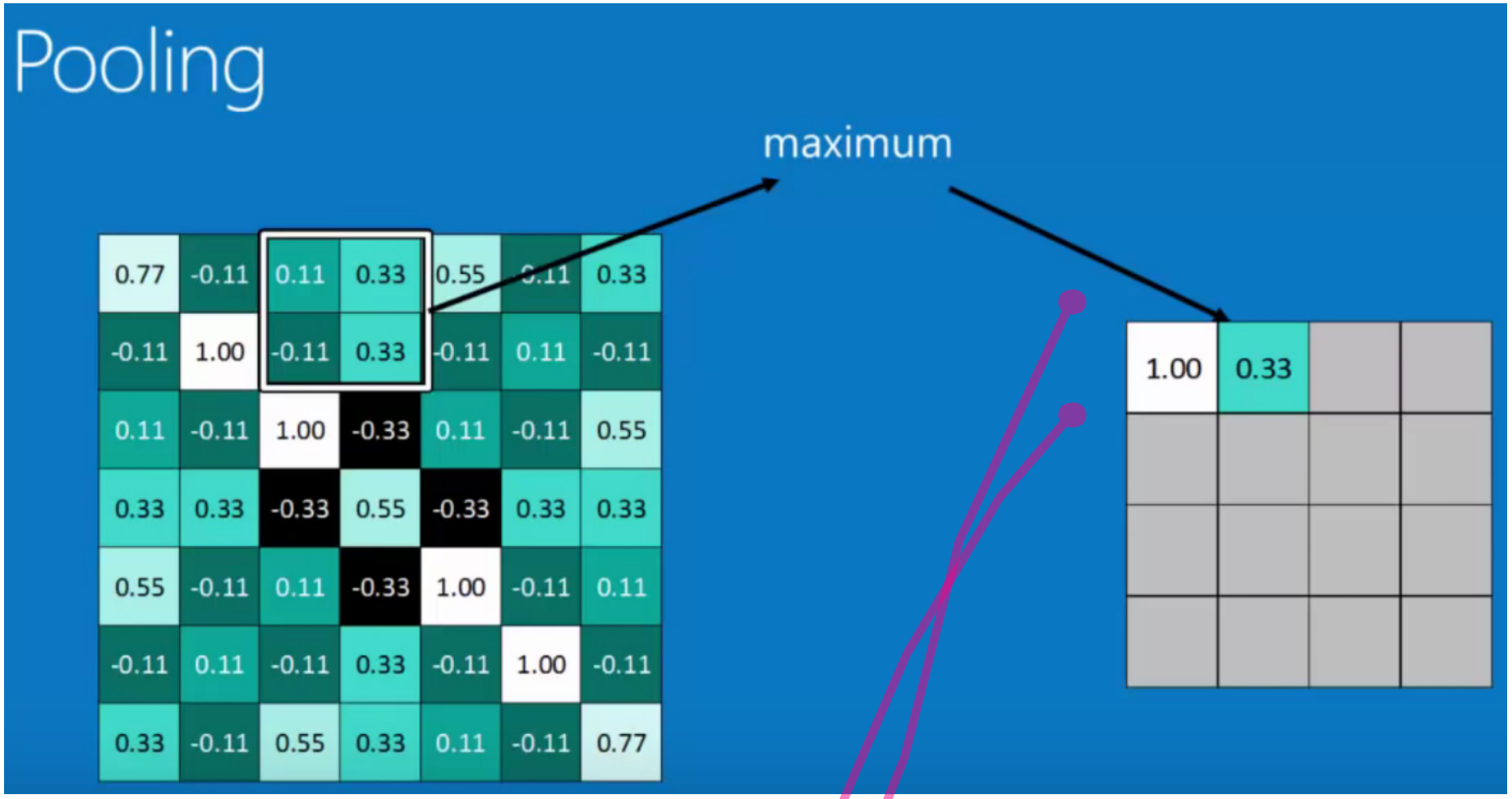


Convolution Neural Networks

Convolution Neural Networks



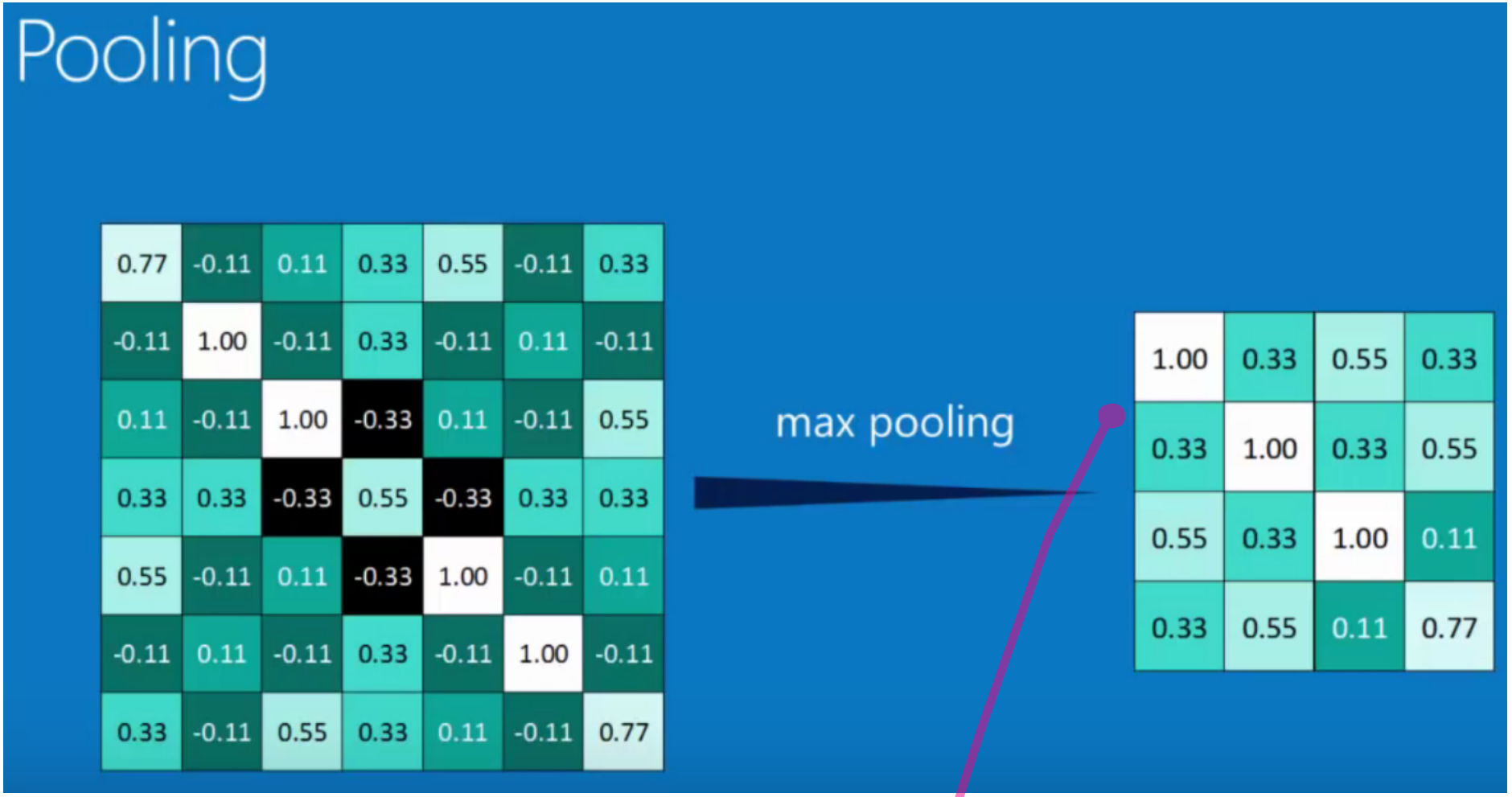
Convolution Neural Networks



Pooling: Shrinking the image stack

1. Pick a window size (usually 2 or 3).
2. Pick a stride (usually 2).
3. Walk your window across your filtered images.
4. From each window, take the maximum value.

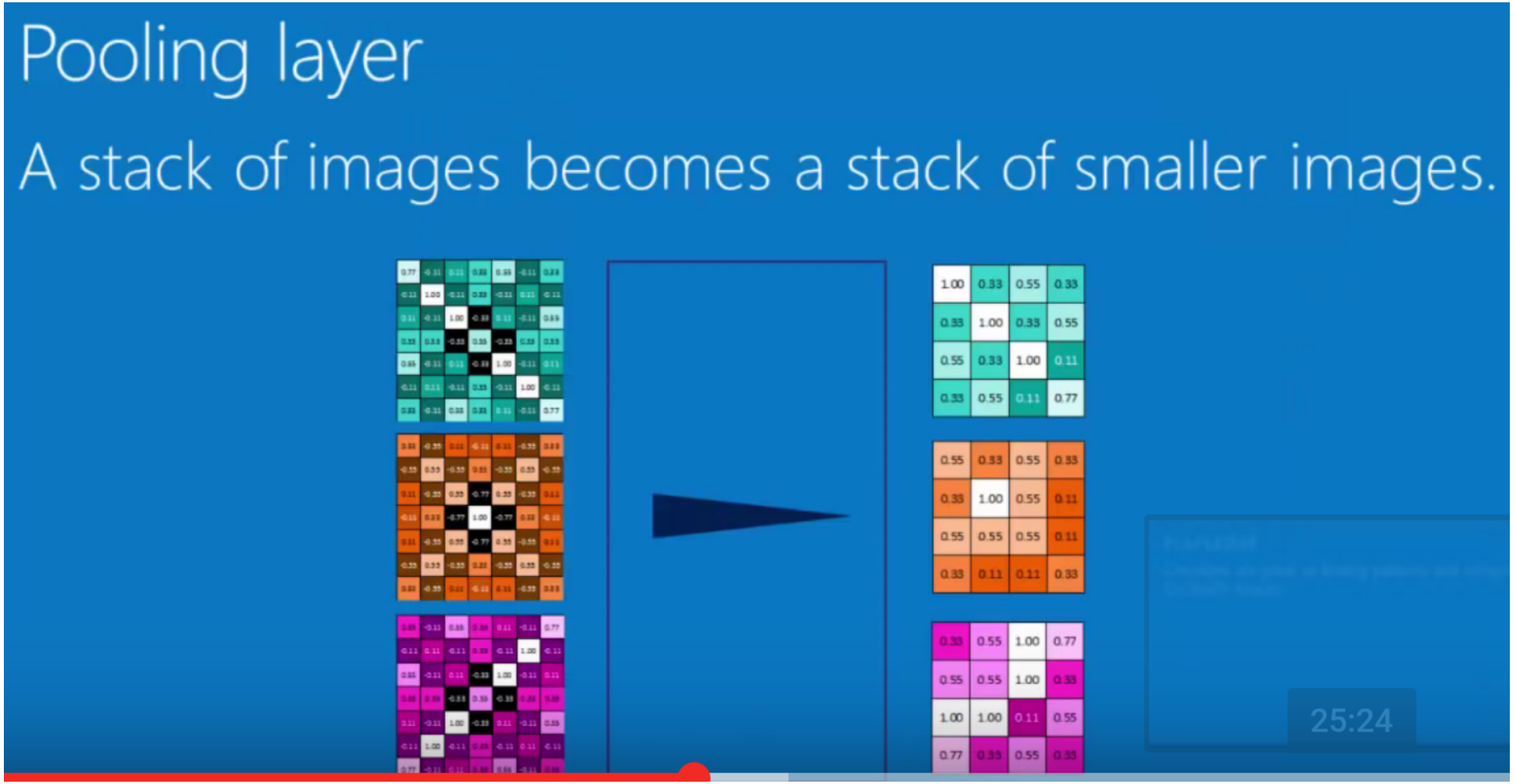
Convolution Neural Networks



Pooling: Shrinking the image stack

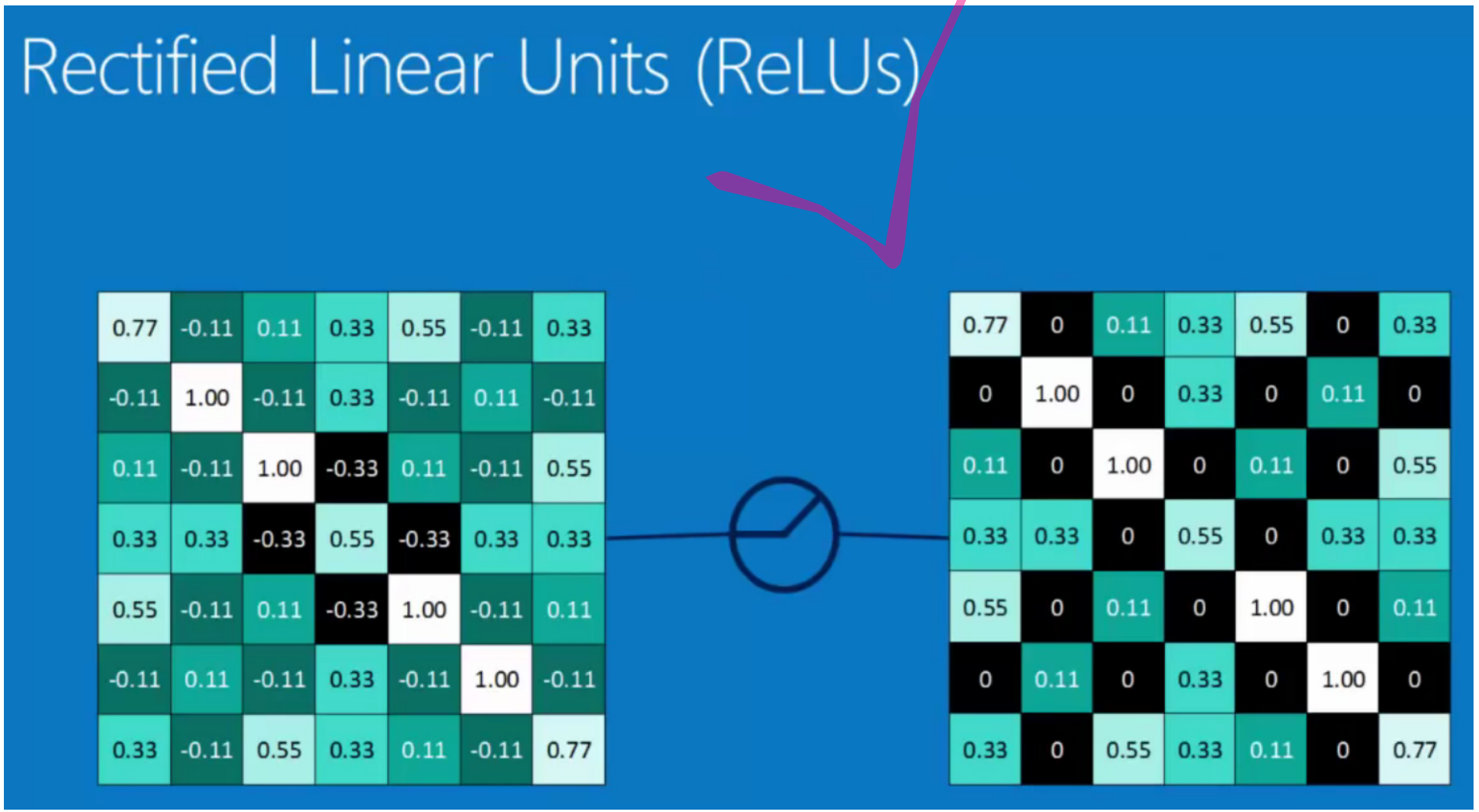
1. Pick a window size (usually 2 or 3).
2. Pick a stride (usually 2).
3. Walk your window across your filtered images.
4. From each window, take the maximum value.

Convolution Neural Networks



Convolution Neural Networks

Rectified Linear Units (ReLUs)



Normalization

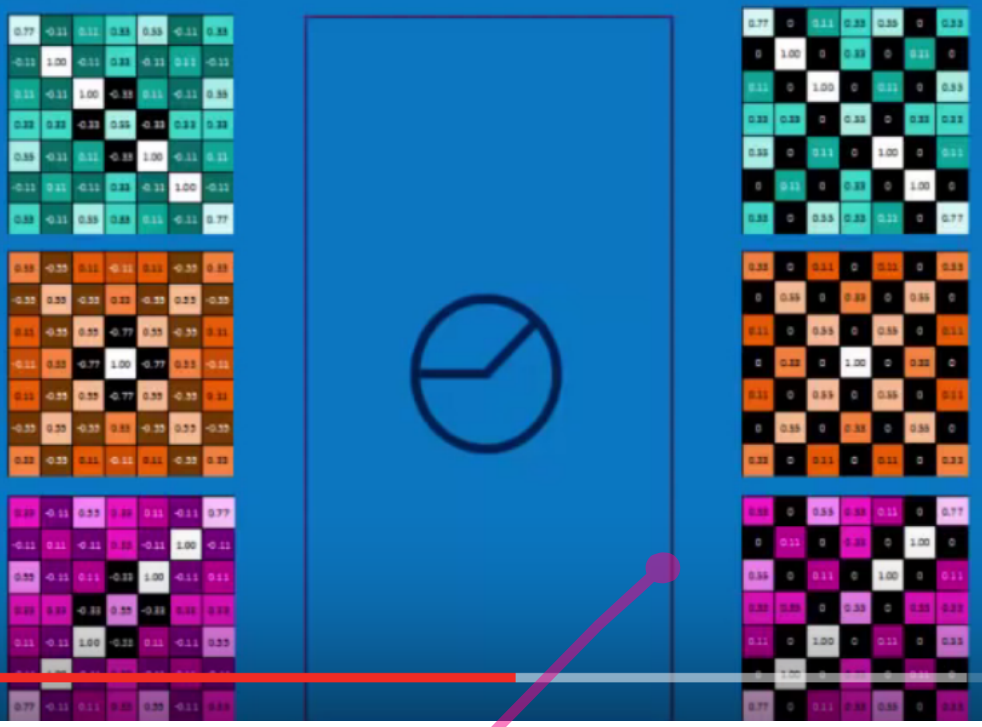
Keep the math from breaking by tweaking each of the values just a bit.

Change everything negative to zero.

Convolution Neural Networks

ReLU layer

A stack of images becomes a stack of images with no negative values.

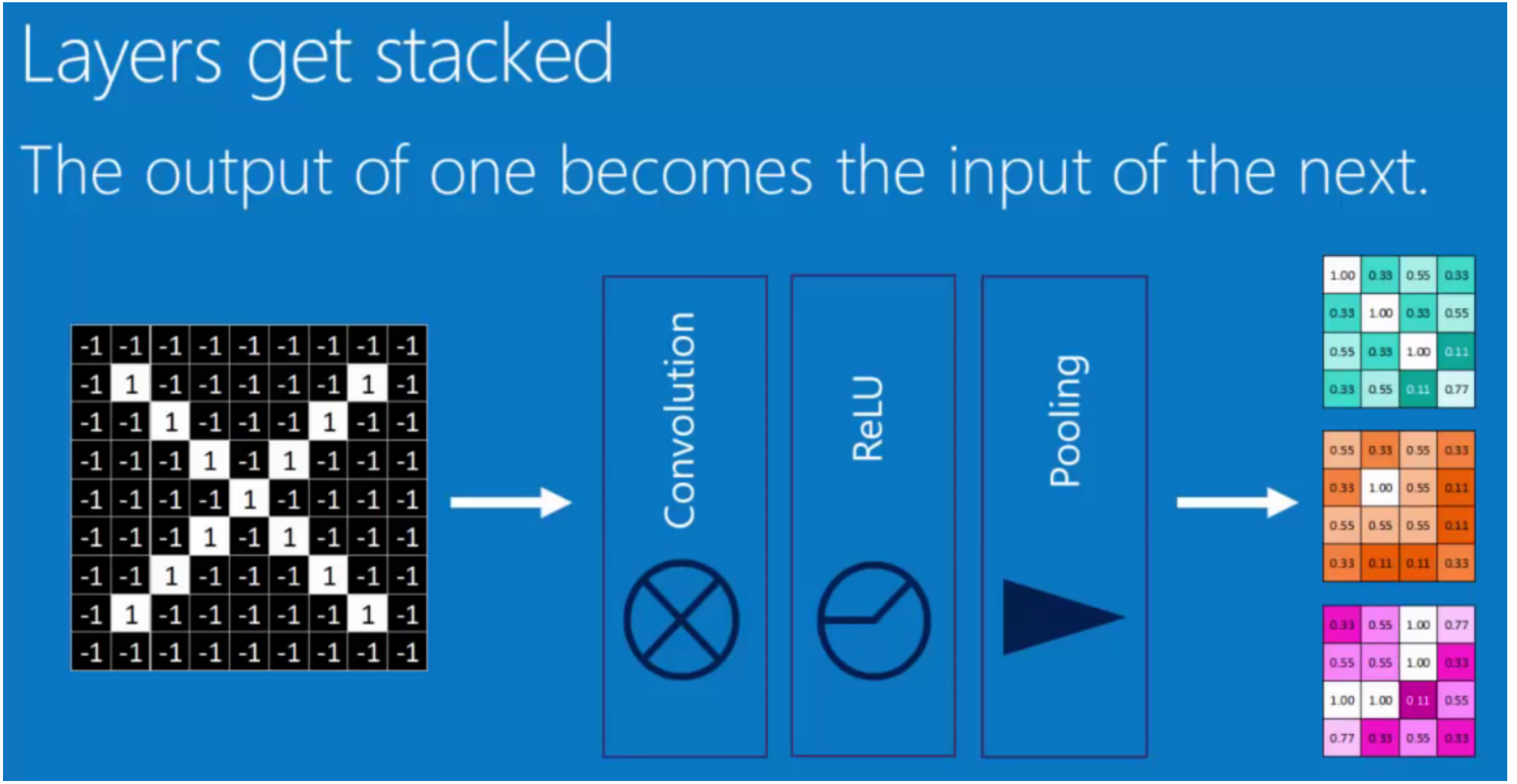


Normalization

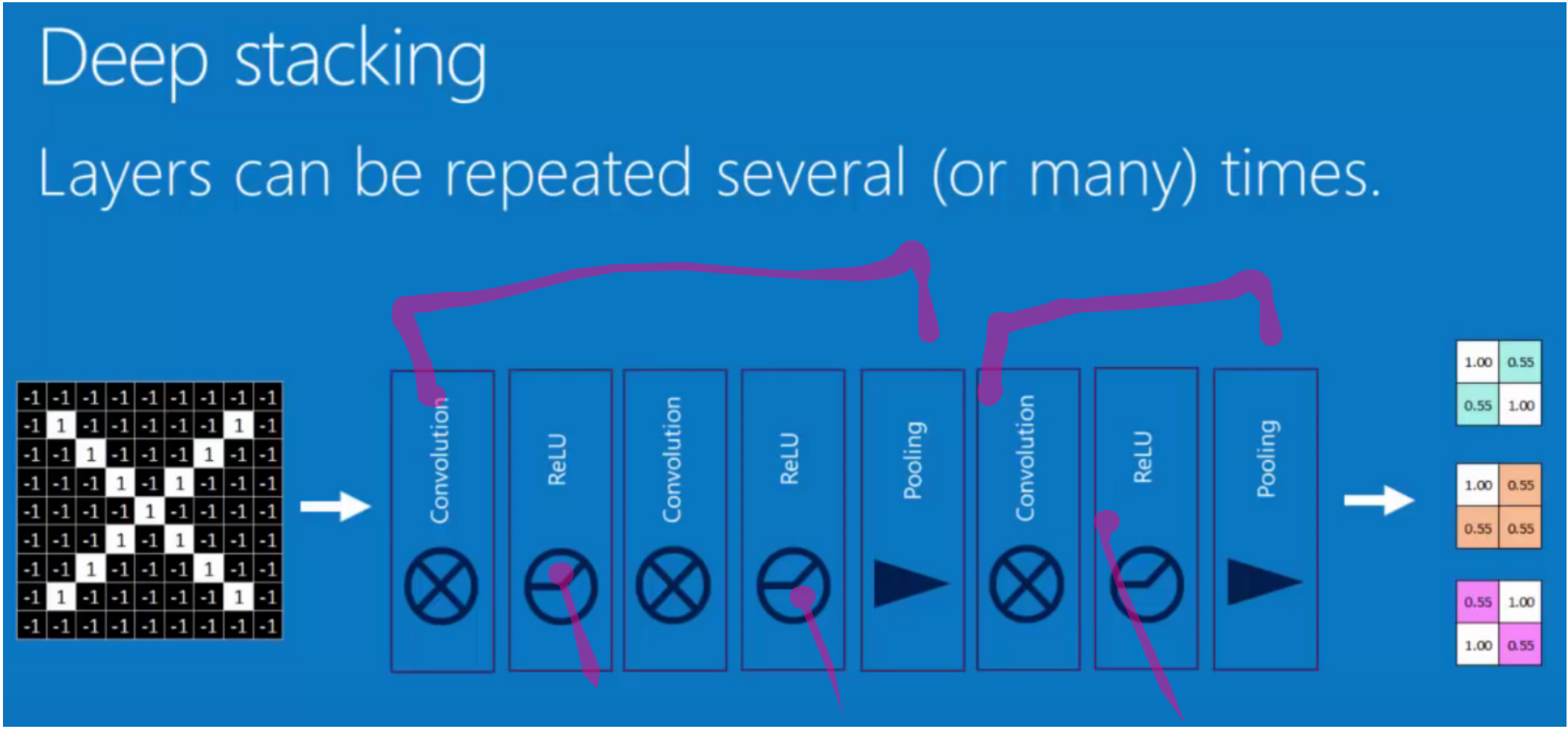
Keep the math from breaking by tweaking each of the values just a bit.

Change everything negative to zero.

Convolution Neural Networks

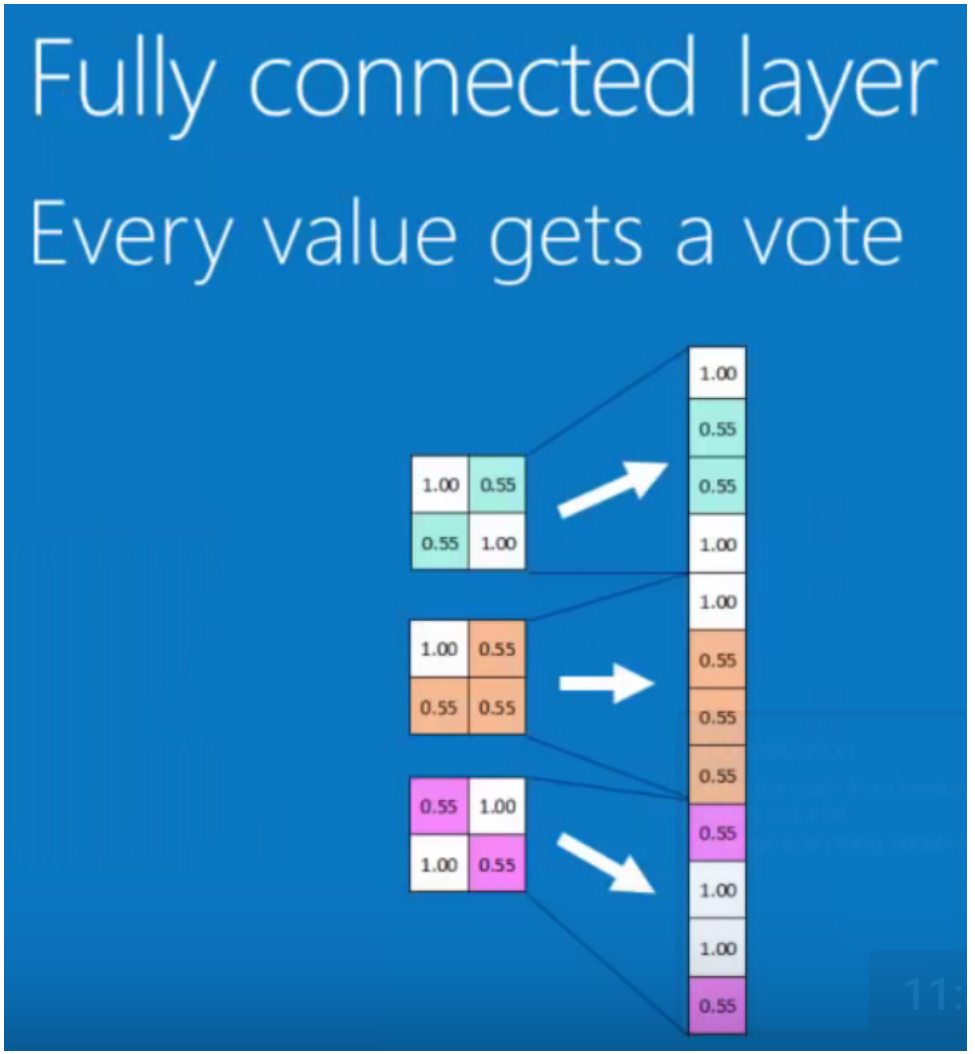


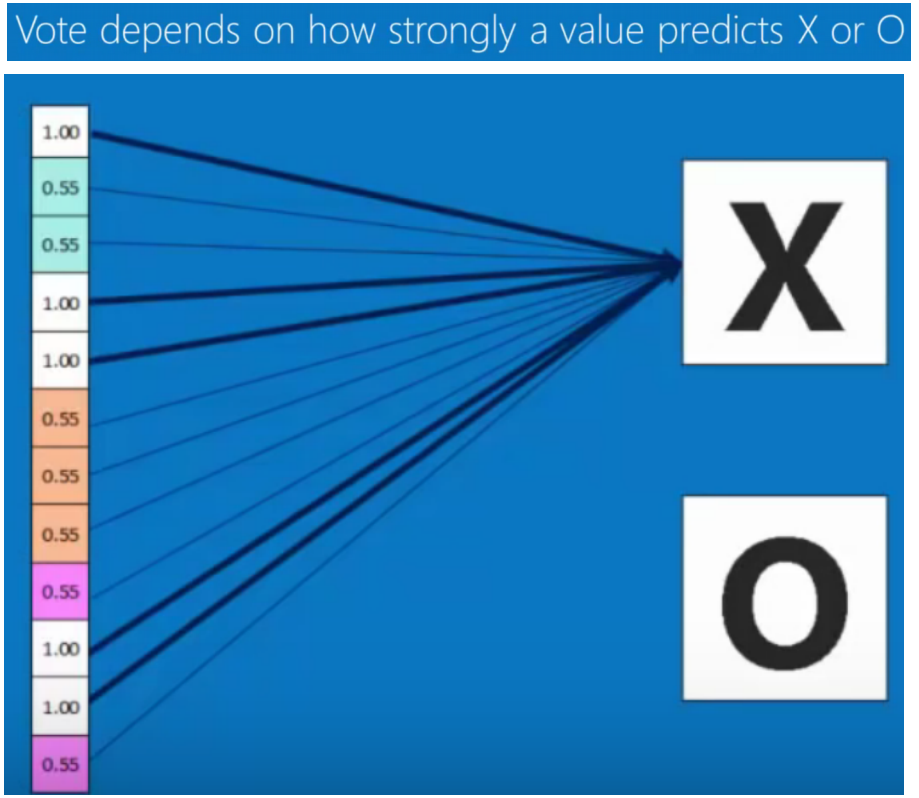
Convolution Neural Networks



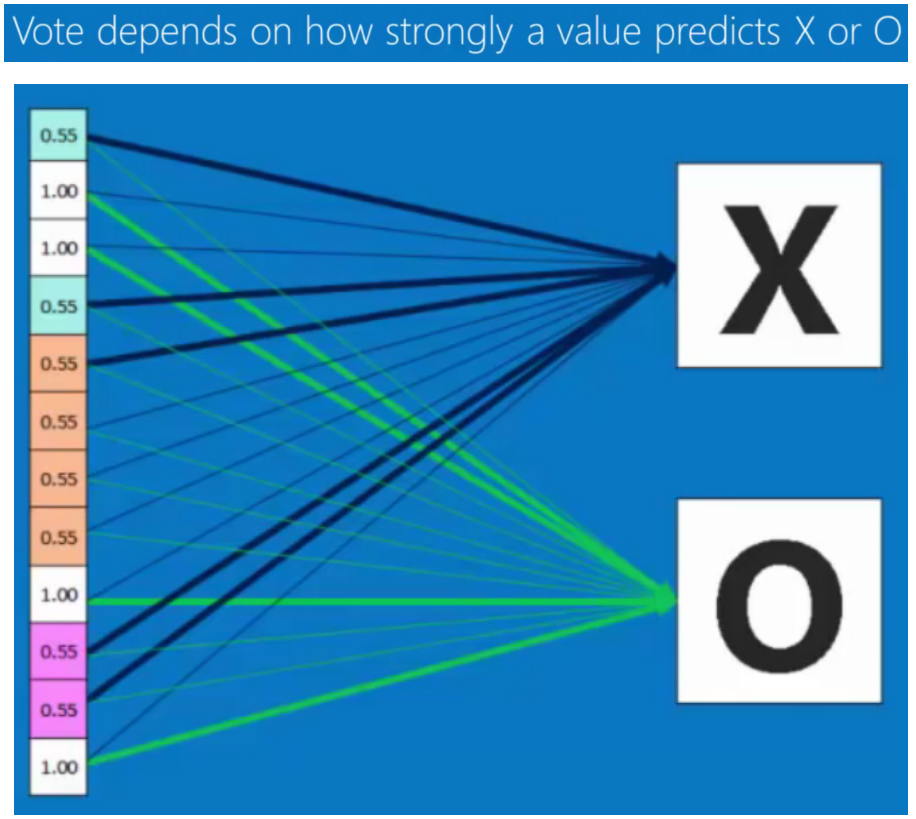
- Convolution layer: filtering
- Pooling layer: smaller

Convolution Neural Networks

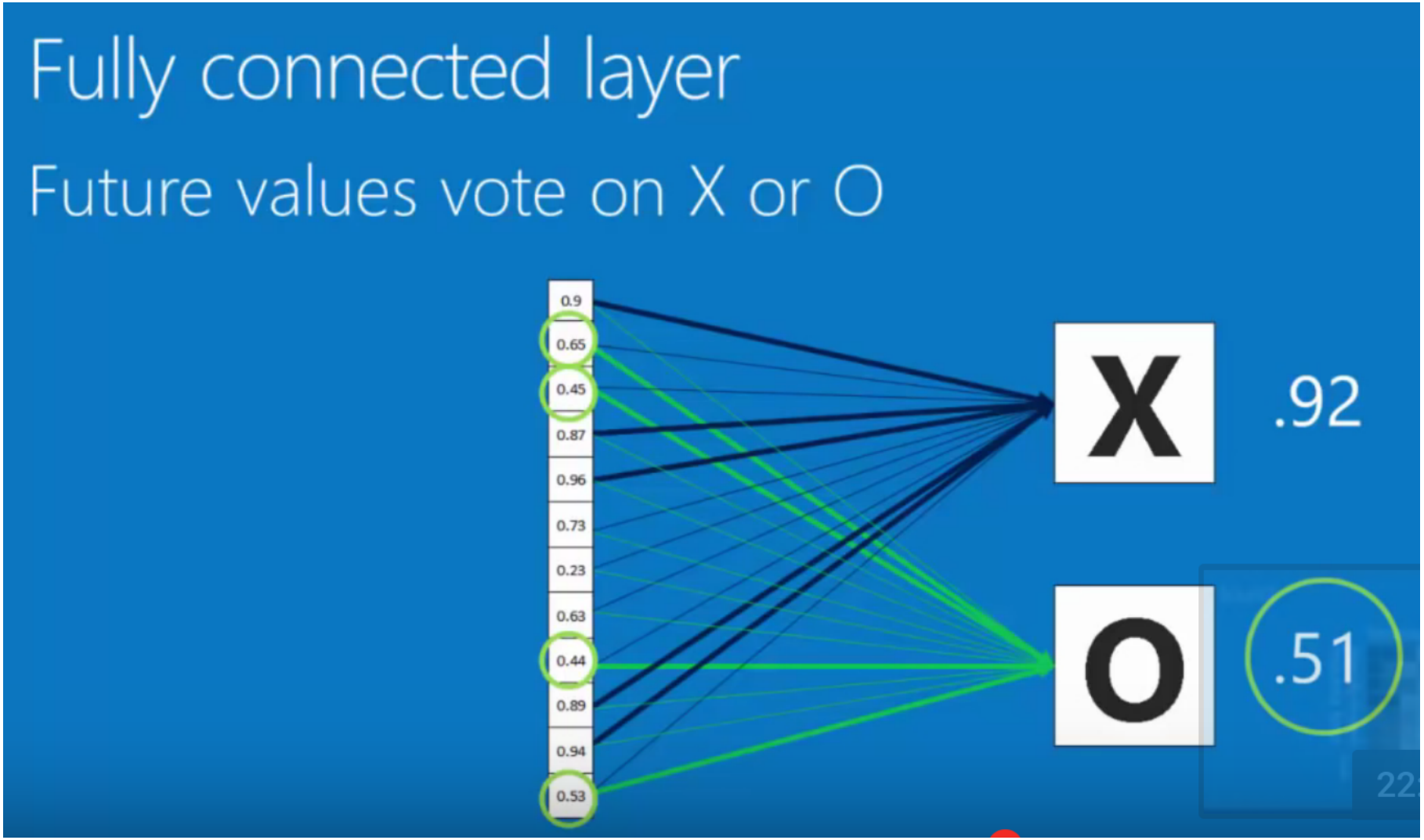




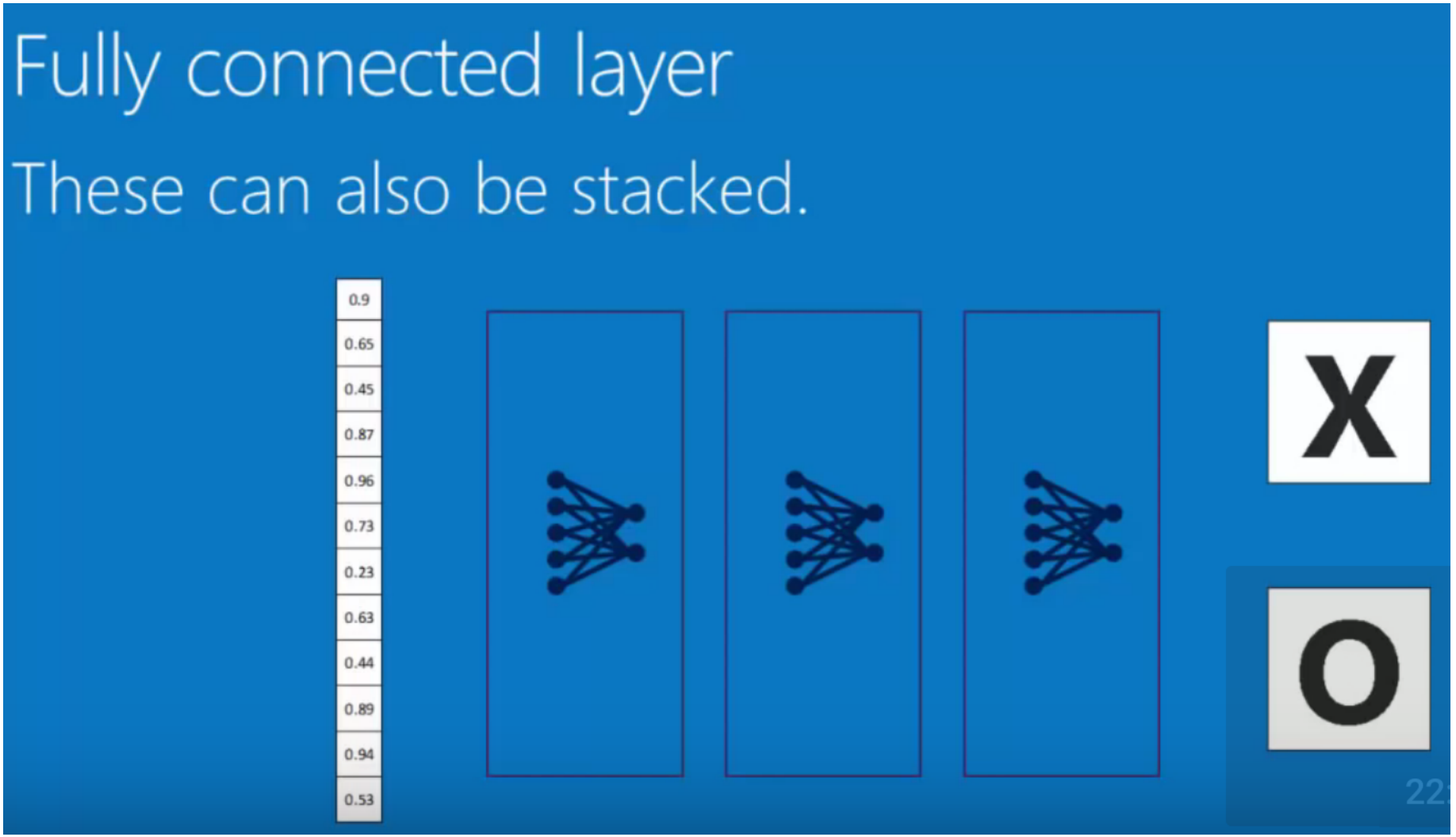
Convolution Neural Networks



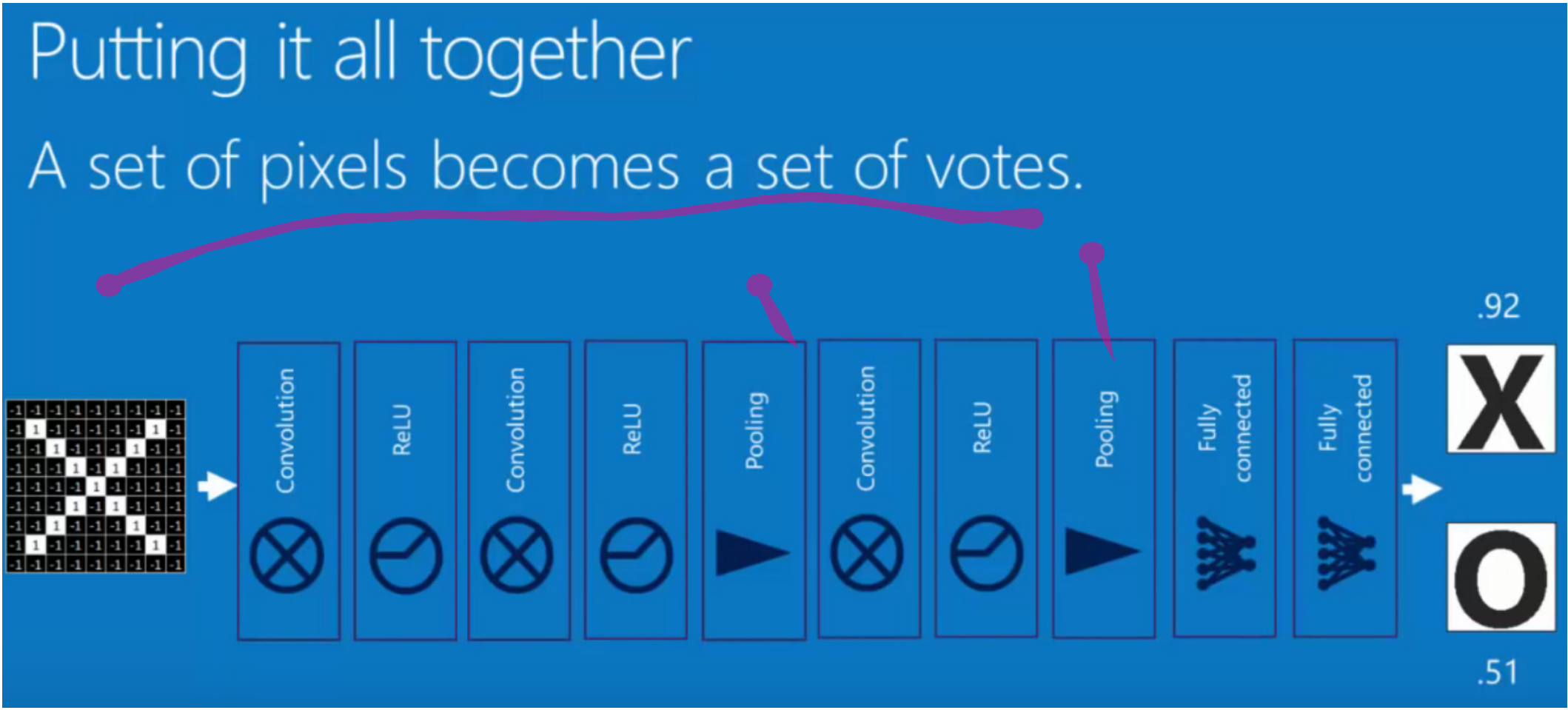
Convolution Neural Networks



Convolution Neural Networks



Convolution Neural Networks



Convolution Neural Networks

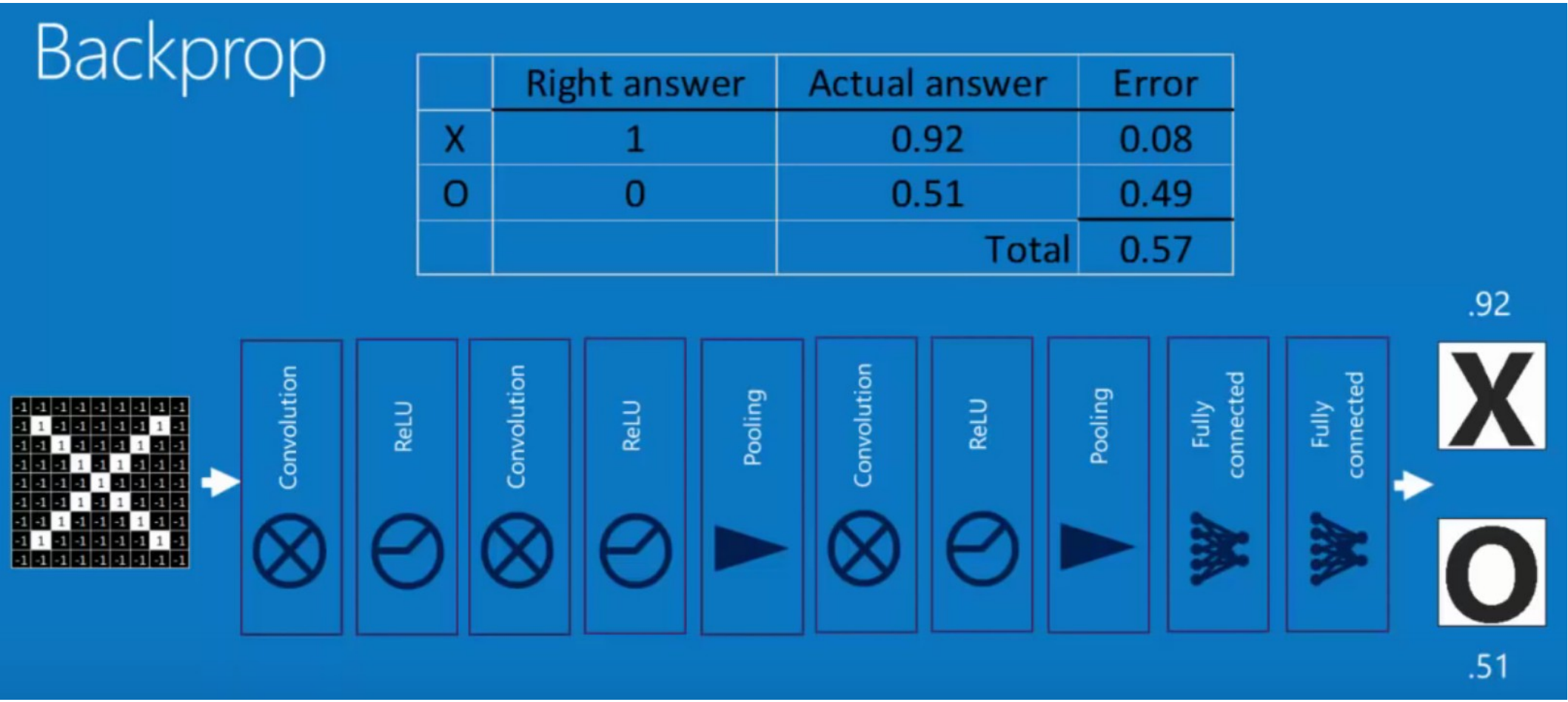
Learning

Q: Where do all the magic numbers come from?

- Features in convolutional layers
- Voting weights in fully connected layers

A: Backpropagation

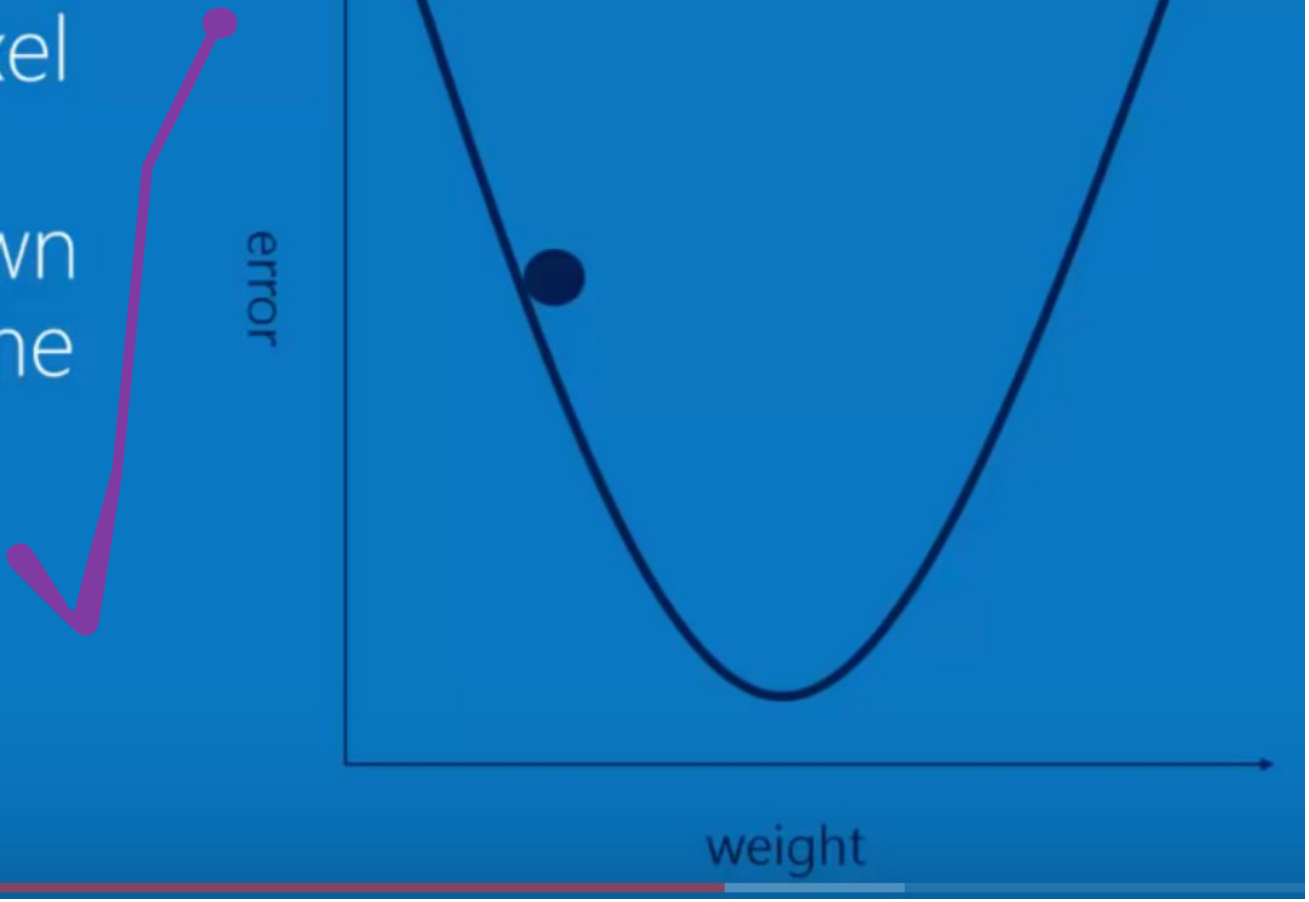
Convolution Neural Networks



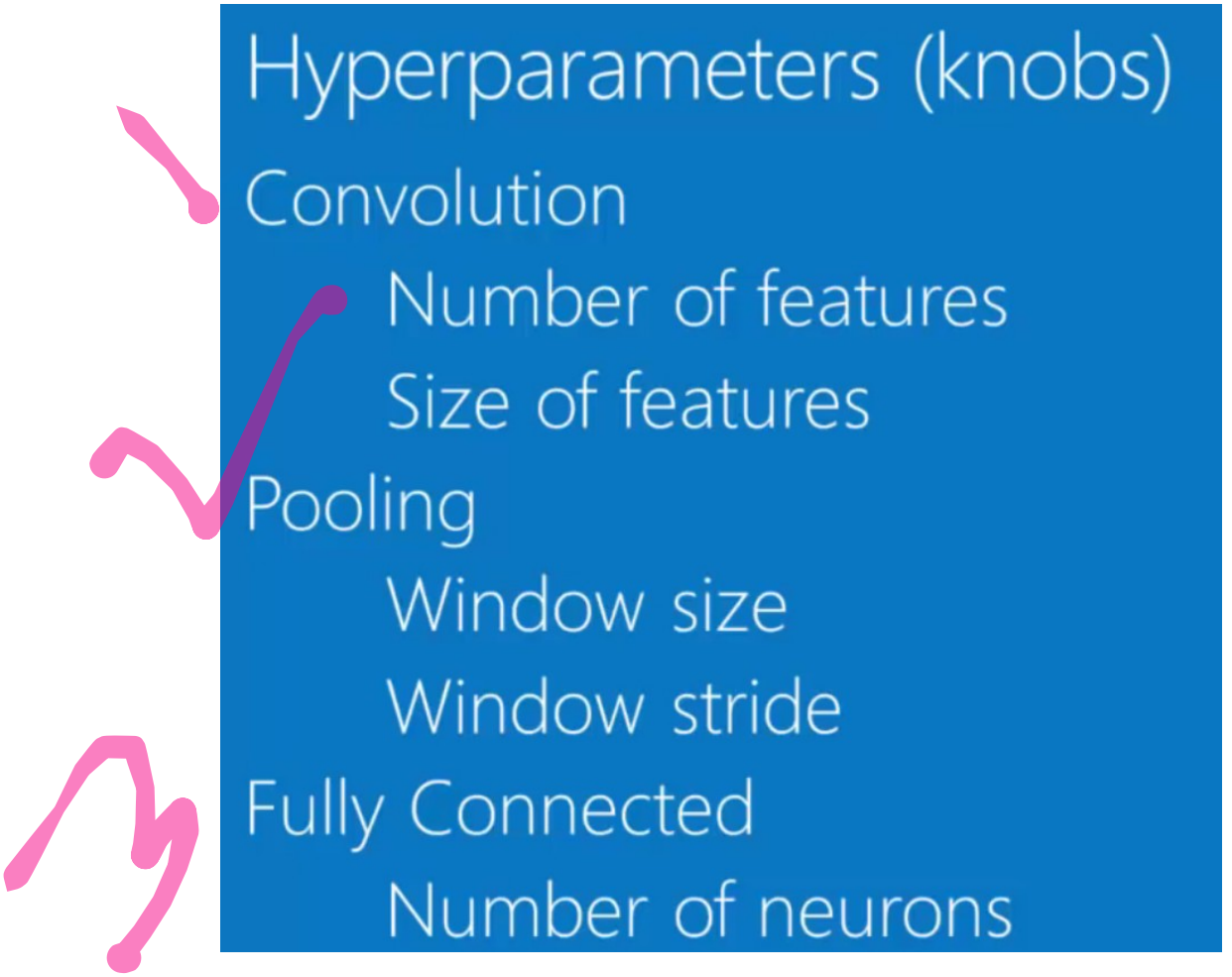
Convolution Neural Networks

Gradient descent

For each feature pixel and voting weight, adjust it up and down a bit and see how the error changes.



Convolution Neural Networks



Architecture

How many of each type of layer?

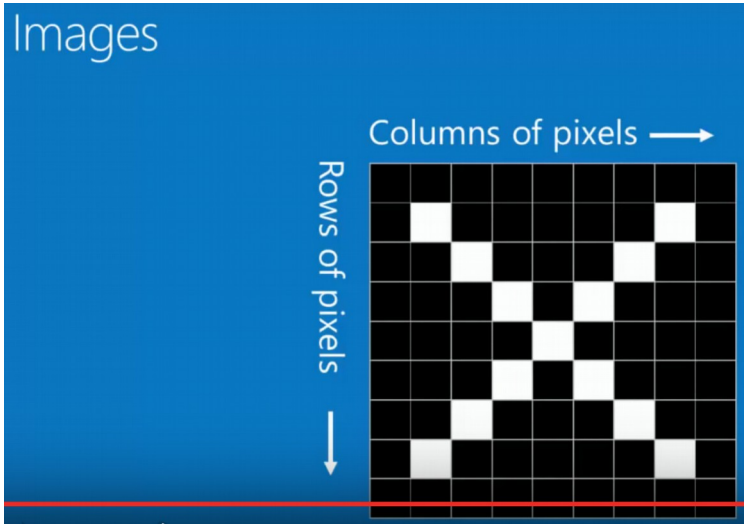
In what order?

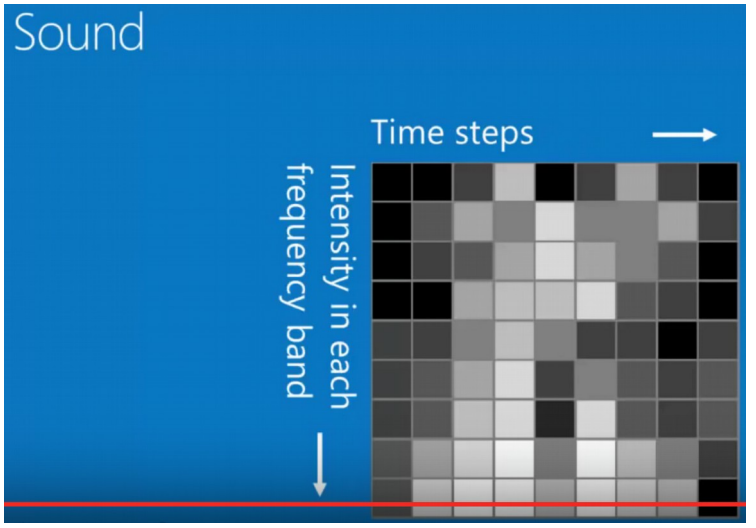
Convolution Neural Networks

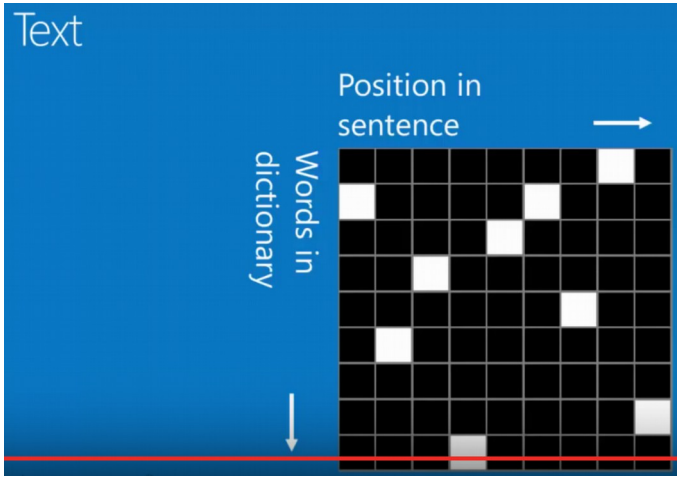
Not just images

Any 2D (or 3D) data.

Things closer together are more closely related than things far away.







Convolution Neural Networks

Limitations

ConvNets only capture local "spatial" patterns in data.

If the data can't be made to look like an image, ConvNets are less useful.

Rule of thumb

If your data is just as useful after swapping any of your columns with each other, then you can't use Convolutional Neural Networks.

Customer data

Name, age, address, email, purchases, browsing activity,...

Customers

A	22	1A	a@a	1	aa	a1.a	123	aa1
B	33	2B	b@b	2	bb	b2.b	234	bb2
C	44	3C	c@c	3	cc	c3.c	345	cc3
D	55	4D	d@d	4	dd	d4.d	456	dd4
E	66	5E	e@e	5	ee	e5.e	567	ee5
F	77	6F	f@f	6	ff	f6.f	678	ff6
G	88	7G	g@g	7	gg	g7.g	789	gg7
H	99	8H	h@h	8	hh	h8.h	890	hh8
I	111	9I	i@i	9	ii	i9.i	901	ii9

Convolution Neural Networks

In a nutshell

ConvNets are great at finding patterns and using them to classify images.

Some ConvNet/DNN toolkits

Caffe (Berkeley Vision and Learning Center)

CNTK (Microsoft)

Deeplearning4j (Skymind)

TensorFlow (Google)

Theano (University of Montreal + broad community)

Torch (Ronan Collobert)

Many others