PROFESSIONAL & CONTINUING EDUCATION
UNIVERSITY *of* WASHINGTON

Machine Learning Techniques

DATASCI 420

Lesson 09-3 Convolutional Neural Networks for Computer Vision

Image Classification with CNNs

- Task of taking an input image and outputting a class
- Probability of classes that best describes the image
- For humans, effortless task

What we see

55	36	71	62	75	79	03	33	55	60	81	02	85	28	16	77	44	72	71	57
37	22	37	95	60	65	43	60	73	58	18	10	42	74	08	11	55	64	21	29
88	38	61	15	15	67	51	41	20	06	68	39	27	94	48	31	39	31	32	25
69	84	38	82	27	11	47	15	56	85	96	80	95	32	08	59	48	82	39	88
25	26	15	02	13	68	44	73	66	78	75	91	95	04	20	30	36	31	36	29
30	68	89	70	08	50	87	04	28	39	76	52	13	13	04	74	52	15	67	86
20	63	30	74	32	18	92	86	58	43	01	70	63	25	28	53	46	41	70	05
42	33	92	33	02	43	79	38	21	34	66	67	58	50	93	46	33	25	79	96
91	45	76	01	71	31	60	73	54	09	94	81	59	16	10	90	31	01	64	56
40	62	79	13	25	67	70	18	37	09	21	21	31	51	61	20	73	38	84	29
25	62	69	57	43	11	14	20	81	47	33	93	28	07	16	97	19	87	50	23
70	45	58	90	82	93	88	92	59	50	53	06	85	72	95	87	04	59	79	28
90	71	93	54	54	14	11	62	17	66	96	59	74	89	95	18	24	72	76	91
13	95	38	89	43	51	63	14	04	27	88	57	66	41	58	13	83	42	69	31
92	17	33	08	06	48	43	30	15	38	83	40	15	42	88	51	58	68	44	63
02	08	43	24	40	11	37	95	86	94	75	10	68	71	09	82	82	34	54	73
62	31	66	59	79	84	72	45	73	15	37	19	80	44	63	87	86	92	73	23
80	76	66	80	79	94	36	63	53	43	43	92	08	04	71	34	32	27	82	91
10	64	23	93	14	23	78	32	85	43	89	11	56	62	84	97	60	06	76	43
59	06	40	74	60	18	07	61	20	16	08	23	88	42	57	08	50	36	17	72

What a computer sees

Input Image

An image is an array of pixel values

A JPG color image with size 480 x 480:

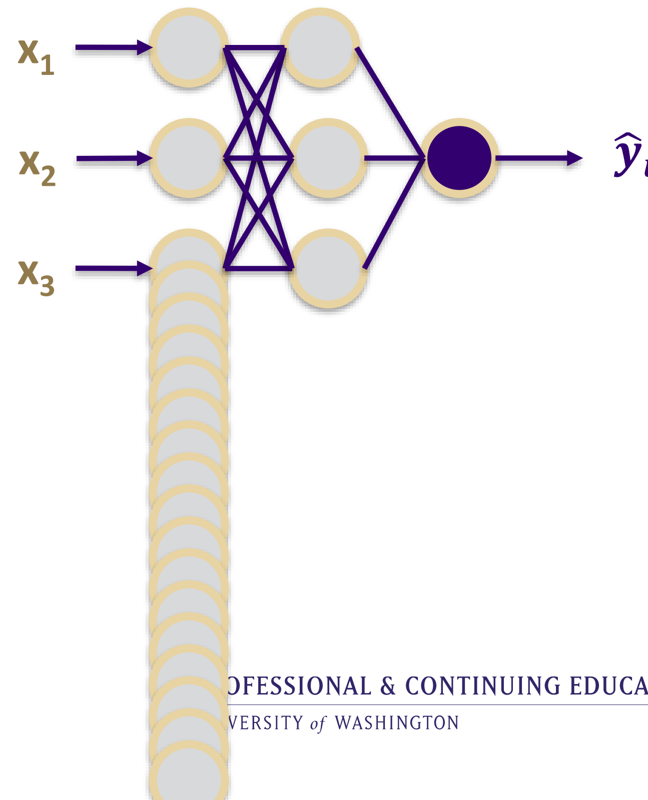
- The representative array will be 480 x 480 x 3. Each number is given a value from 0 to 255 which is the pixel intensity

Grayscale image contains a single sample (intensity value) for each pixel

Image Classification: Given an array of numbers, produce probabilities of the image being a certain class

Why Not Use a Standard Neural Network?

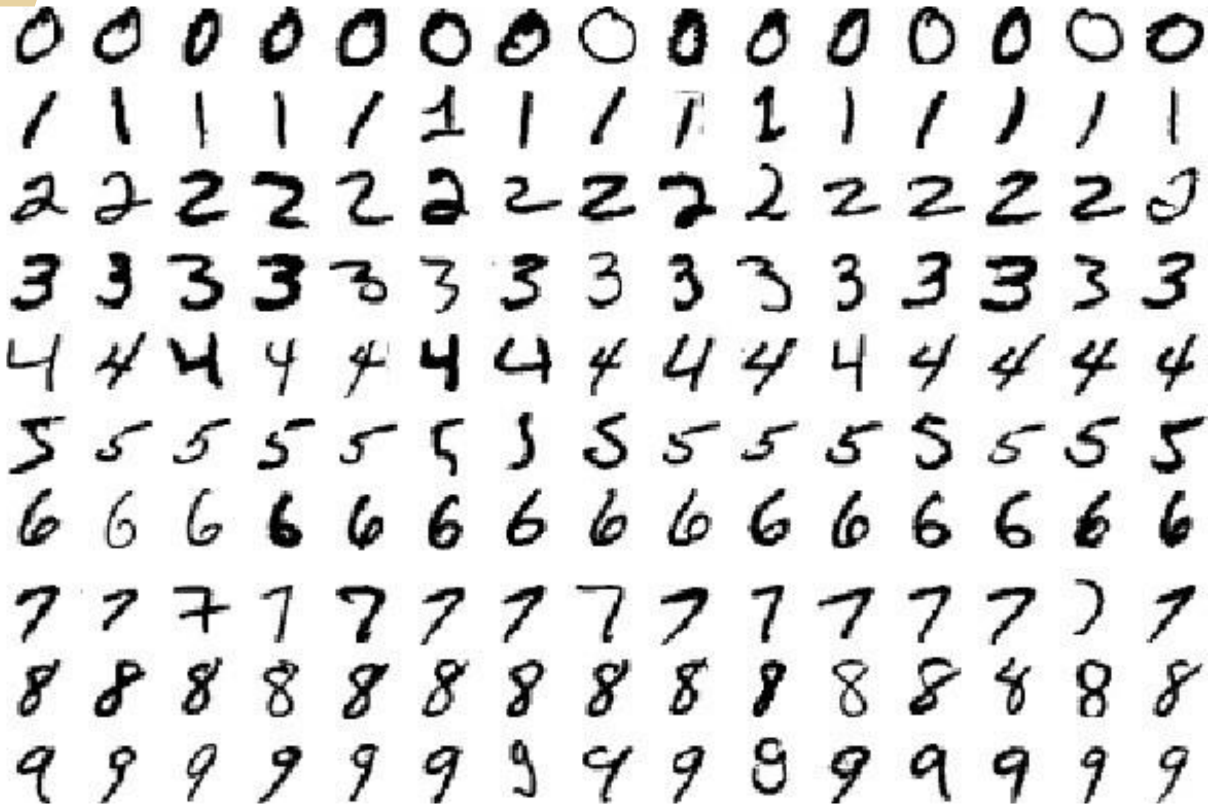
- FF/BP Neural networks are fully connected
- With an image size of 480x480x3 this is a massive input vector → 691,200 input vector
- Even with 32x32x3 it's a vector size of 3072



CNN Network Layers

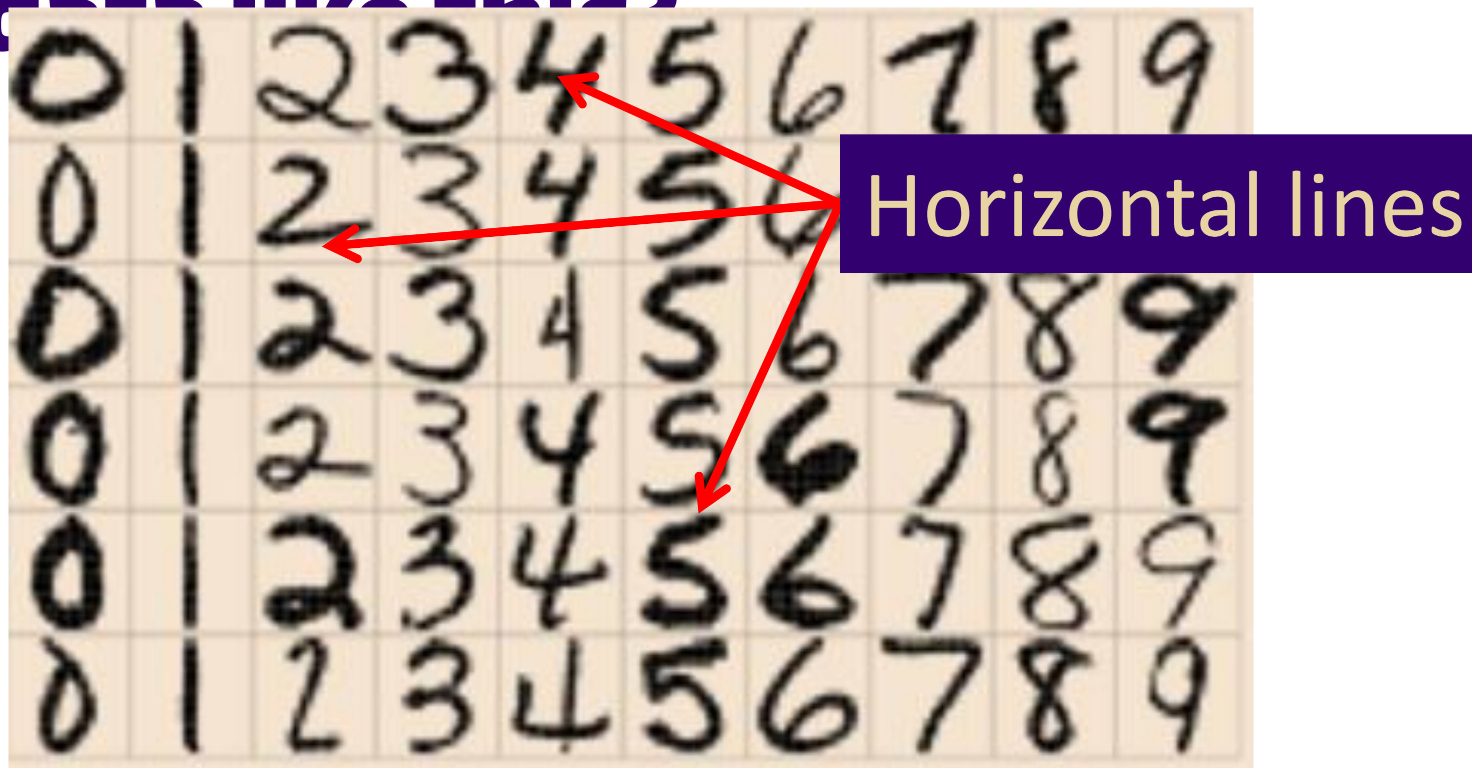
- Convolutional Layers
- Pooling Layers
- Fully Connected Layers

MNIST – Database of Handwritten Numbers

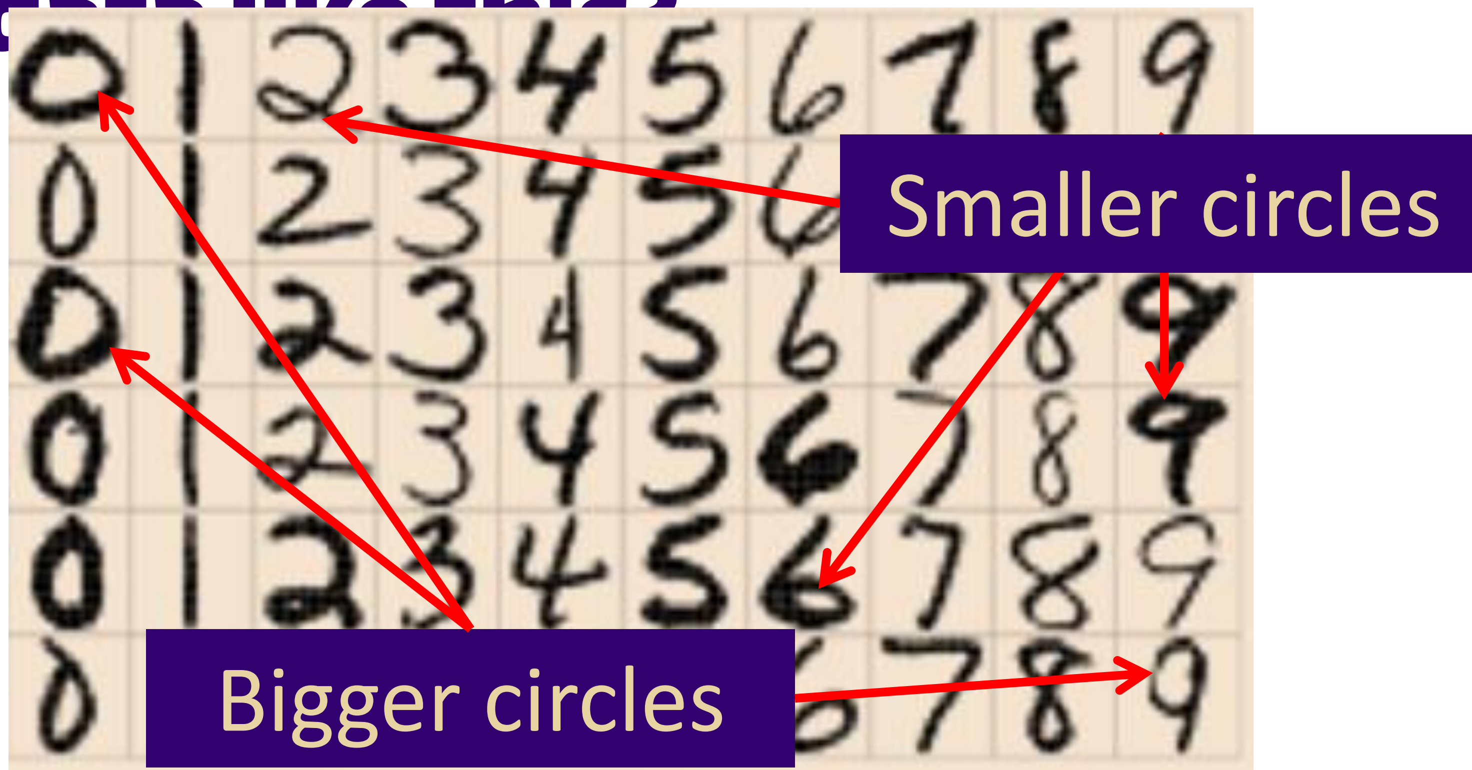


What features might you expect a good DNN to learn when trained with data like this?

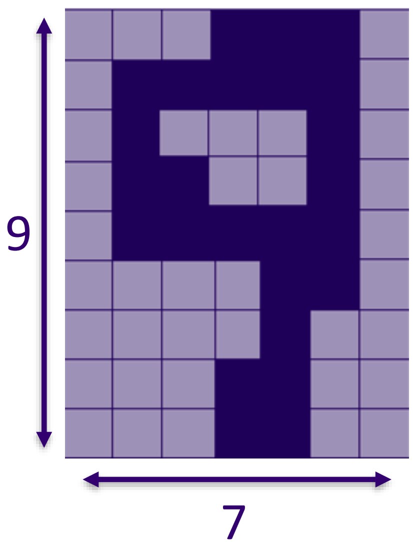
What features might you expect a good DNN to learn when trained with data like this?

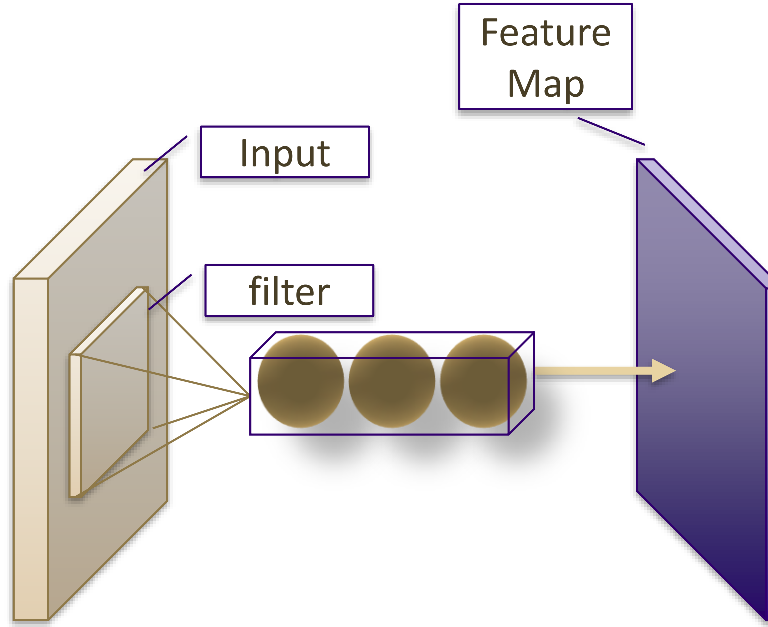


What features might you expect a good DNN to learn when trained with data like this?

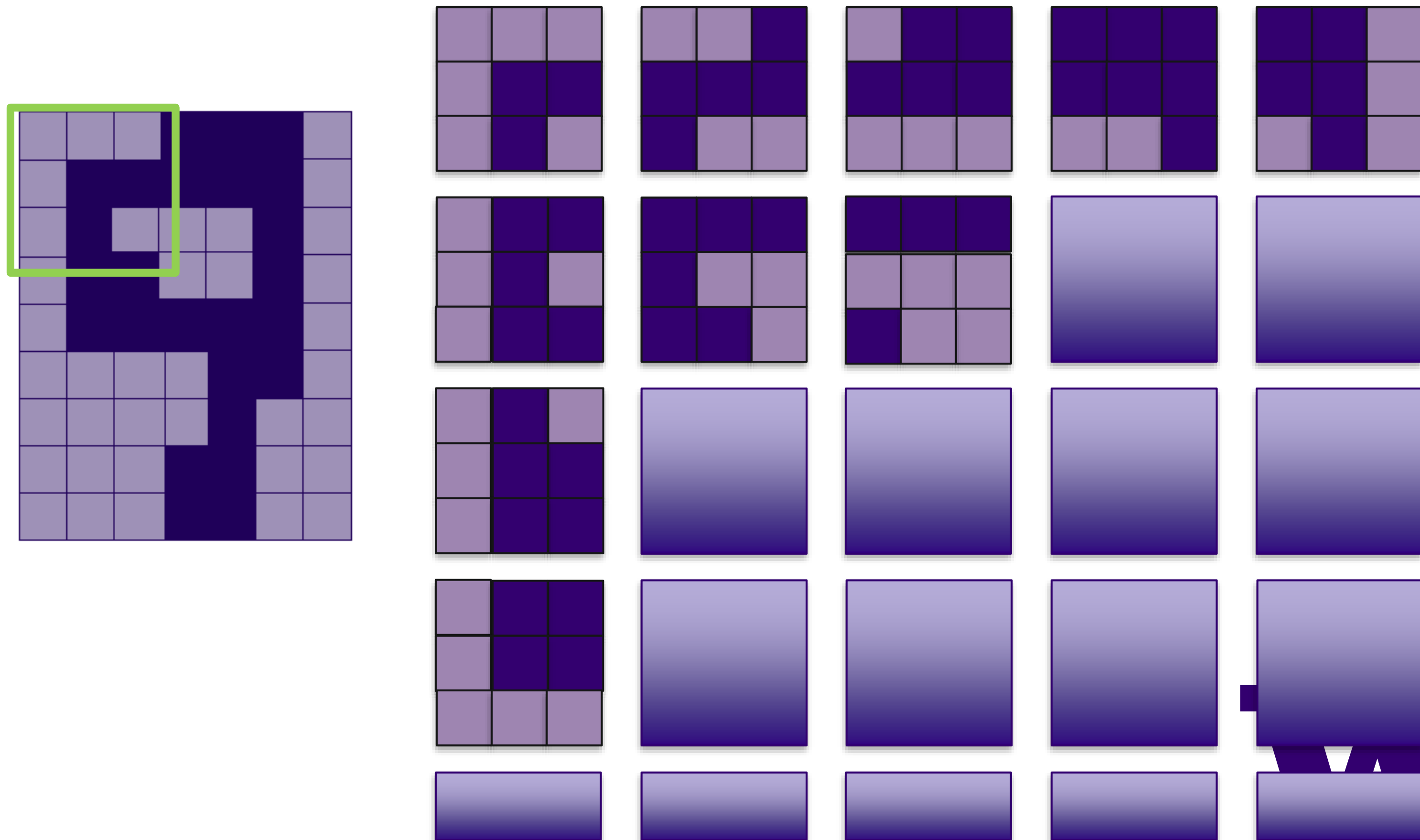


Convolutional Layers: Feature Detectors

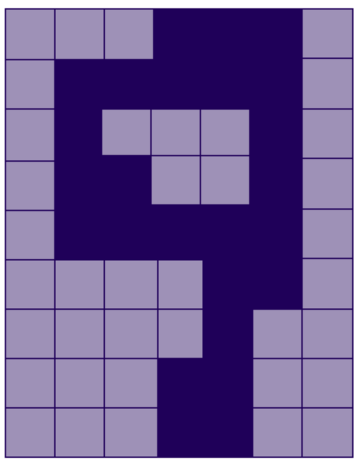


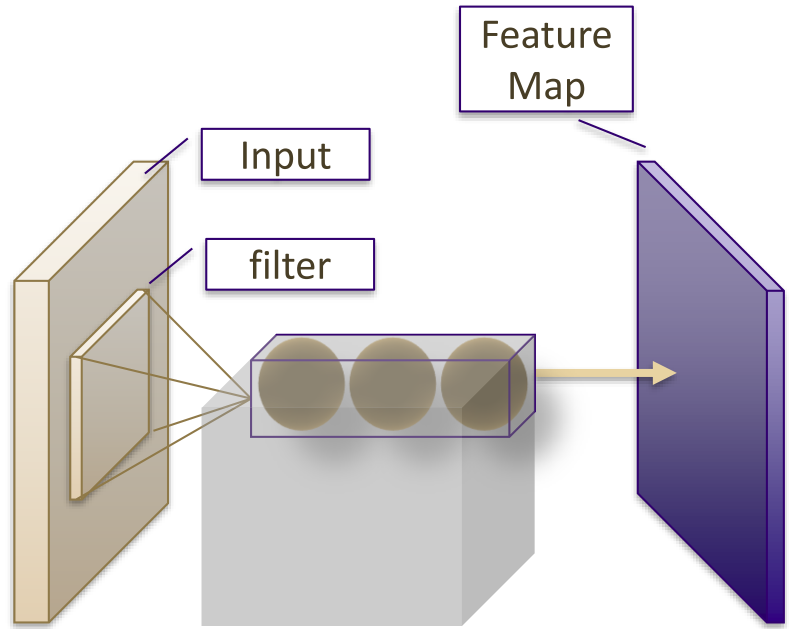


Conv Layers: Self-organized Feature Detectors



Convolutional Layers: Feature Detectors





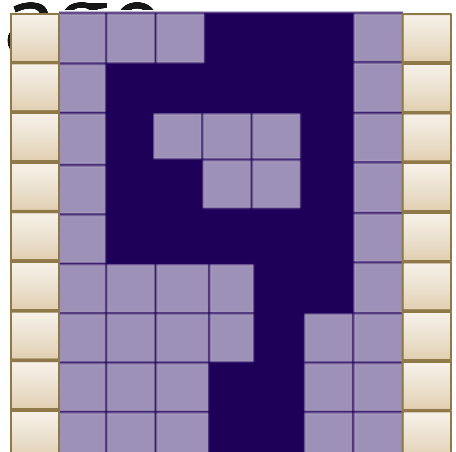
What is Padding?

Padding is a concept of where you might add a zero weight pixel to the outside of an ima

$$\frac{W - F + 2P}{S + 1}$$

Where:

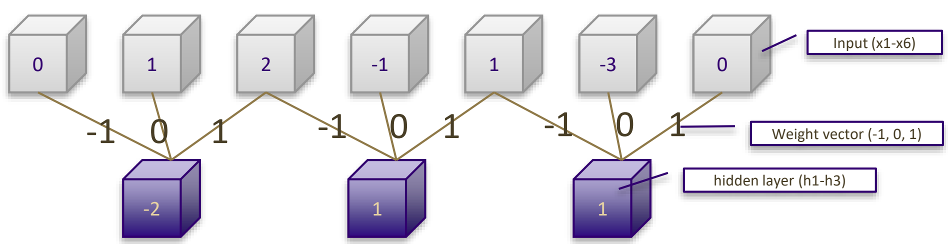
- W = width of the input
- F = filter size
- P = zero padding
- S = stride



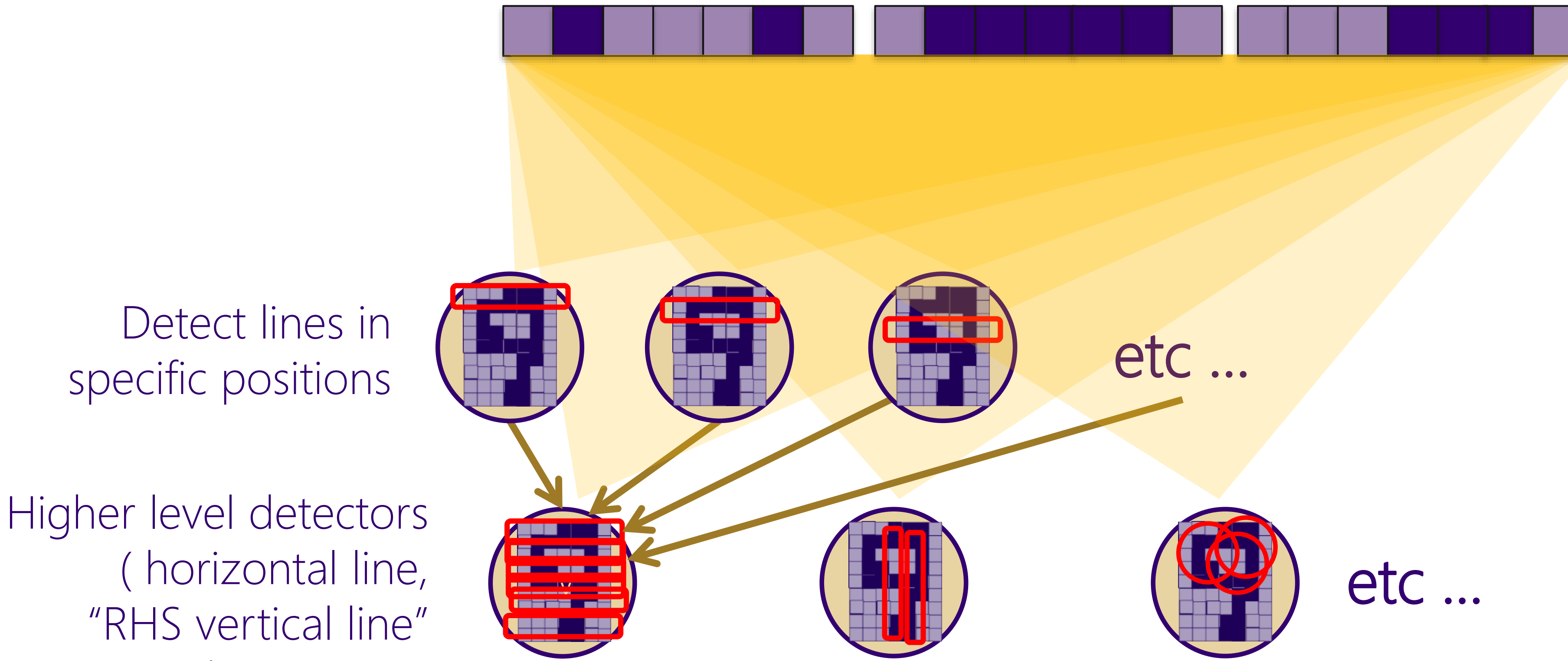
$$\frac{7 - 3 + 2}{2 + 1}$$

What are Shared Weights

When your stride is less than your filter depth, some of the weights across these filtered sections share weights

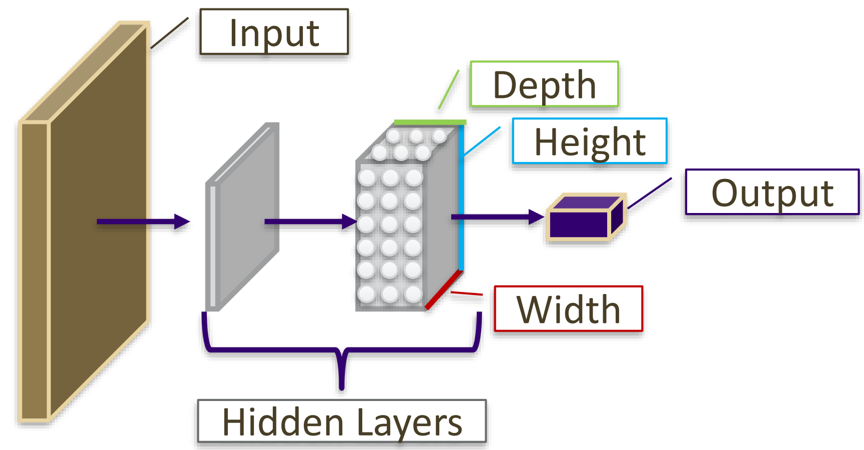


Successive Layers Learn Higher Level Features



Summarizing Feature/Activation Mapping

- Map from the input layer to the hidden layer
- Each mapping reflects a particular feature you want to identify; e.g., edges, curves, etc.
- The filter (AKA kernel) is also known as a “convolution”—which is a shared set of weights across the input space
- Weights are updated via backpropagation



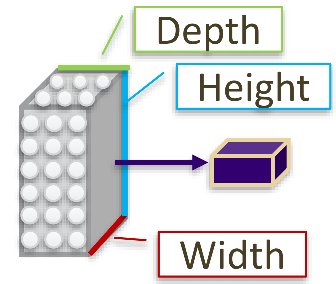
Pooling Layers

Done periodically between convolution layers to:

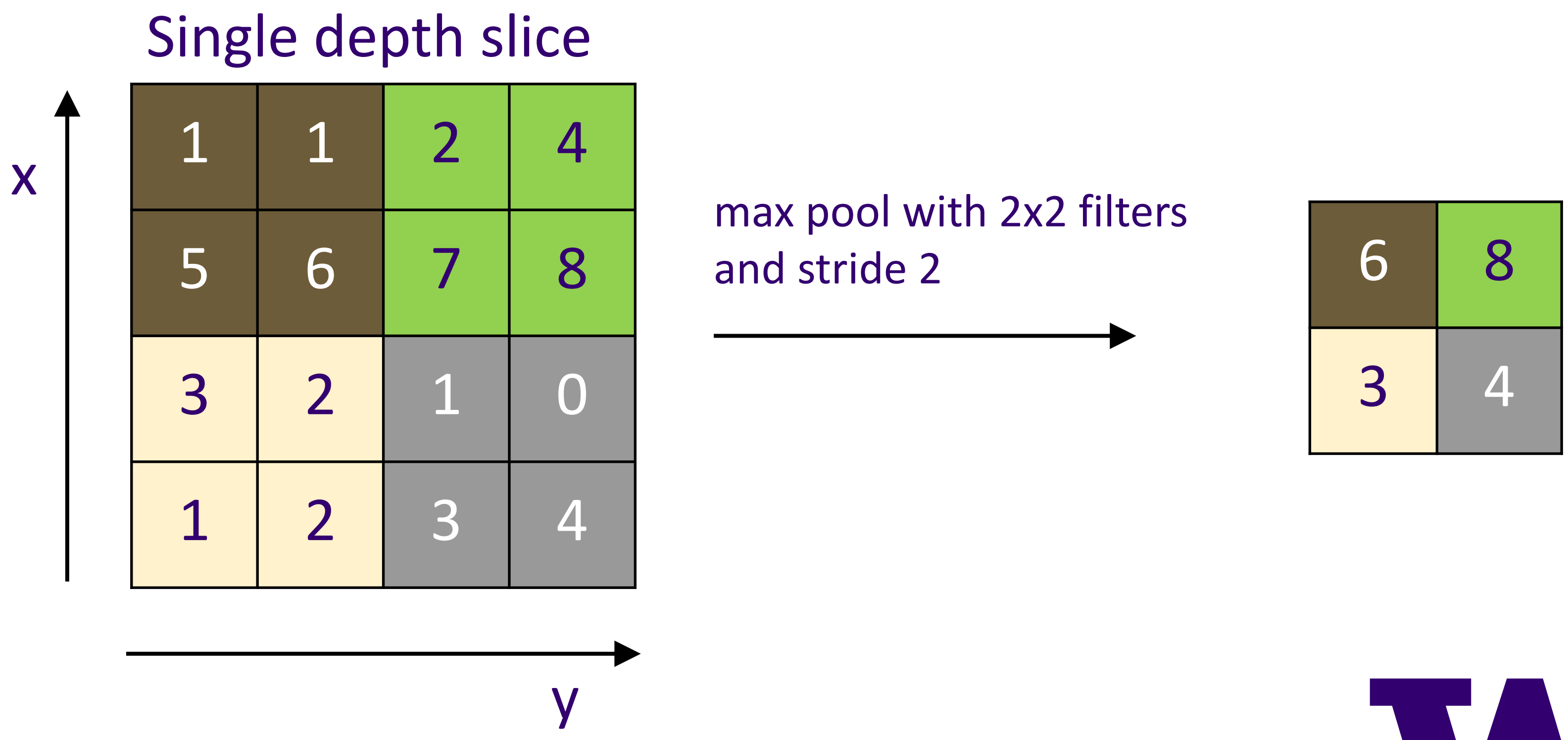
- Reduce the spatial size of the image representation
- Reduce the number of parameters (and thus computation) in the network
- Control overfitting

How to MAX Pool

- Takes the volume W1xH1xD1
- Requires 2 hyperparameters (F and S)
- Produces a volume of size W2xH2xD2 where:
  - $W2 = \frac{(W1-F)}{S+1}$
  - $H2 = \frac{(H1-F)}{S+1}$
  - $D2 = D1$ (depth is always unchanged)



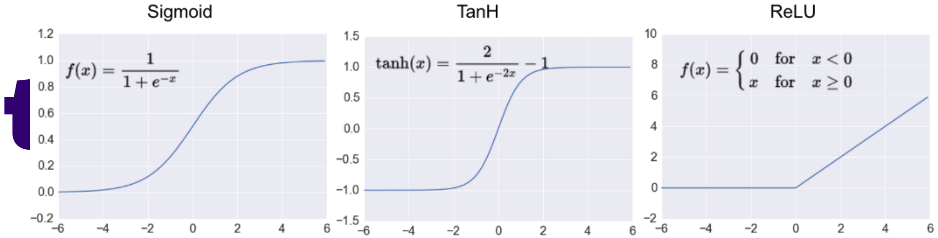
MAX Pooling


Single depth slice
x
y
1 1 2 4
5 6 7 8
3 2 1 0
1 2 3 4
max pool with 2x2 filters
and stride 2
6 8
3 4

Other ‘Good to Knows” about Pooling

- Can be used for averaging instead of reduction
- Proposed to be replaceable by larger strides in CONV layers—works better for generative models

Activation Funct



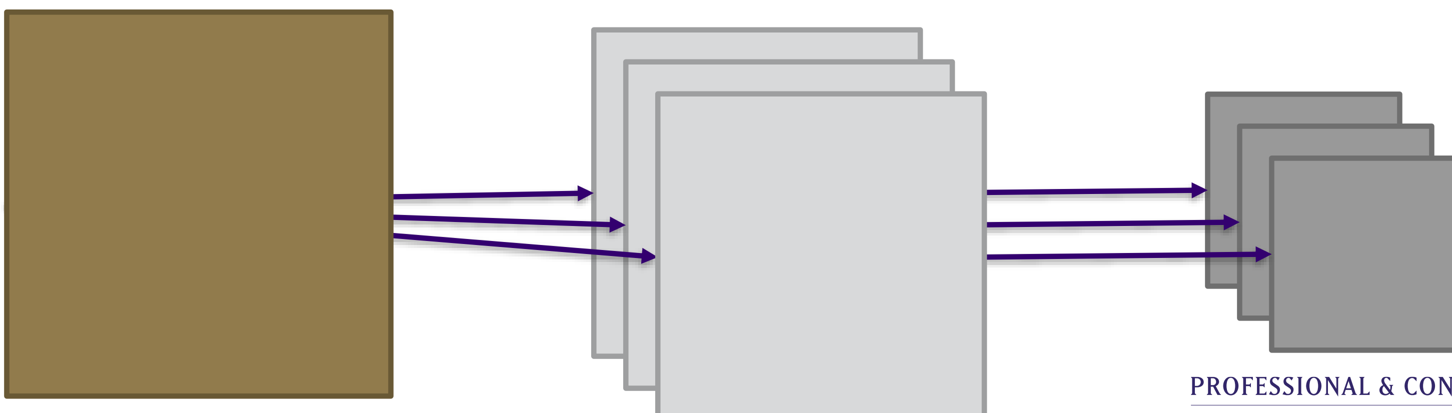
- Logistic (Sigmoid):
  - S-shaped curve that ranges from 0 to 1 – good for mapping probability functions
- tanH:
  - Also S-shaped; but has a wider range from -1 to 1;
- ReLU: Rectified Linear Units – AKA Ramp Activation
  - Zero for negative values and linear for x values greater than 0; has an unlimited positive range (most popular for Deep NNs)

Fully Connected Layers

- Same as in ANNs – all neurons in the layer are fully connected to every neuron in the previous layer.
- Unlike CONV layers that are connected to a local region in the input volume with shared parameters
- Both use dot products across their weights and can easily be converted from one to the other

Combining CONV and Pooling Layers

- As you train you get smaller, more manageable representations
- These activation map operations occur independently



But what about position invariance? Our detectors were tied to specific parts of the image.

Translation Invariance

Ability for the neural network to classify an object by its defining characteristics regardless of where and at what angle they appear

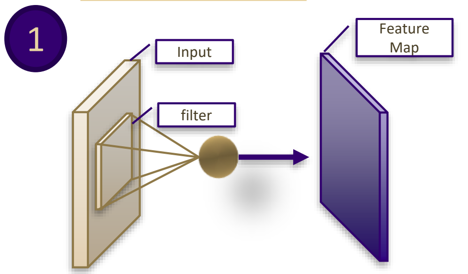
Machine Learning Studio DNN for MNIST

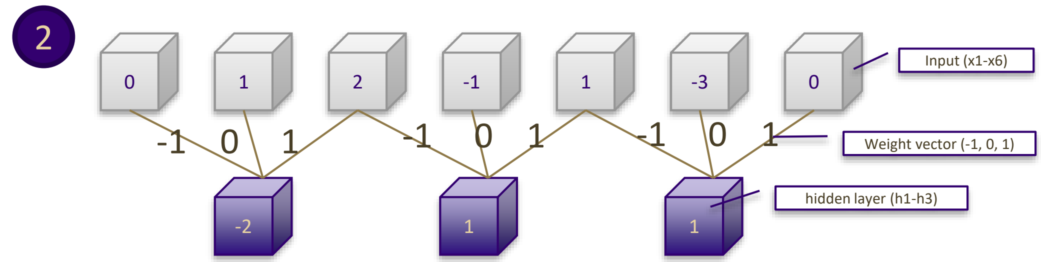
Summary: CNNs for Computer Vision

- Many layers are interspersed between convolution layers:

  Input – Conv→ReLU – Conv→ReLU – Pool→ReLU – Conv→ReLU – FC→ReLU – Dropout→ReLU – Conv→ReLU …
- Better nonlinear predictivity
- Improves the robustness of the network and controls overfitting

Terminology





1	Feature/Activation map	Map from the input layer to the hidden layer
2	Shared weights	Vector of weights defining the feature map
3	Shared bias	Bias defining the feature map
4	Kernel or Filter	A simple, shared set of weights across the input space
5	Activation Function	A mathematical abstraction of the “action potential” of a node (e.g., binary is on/off)