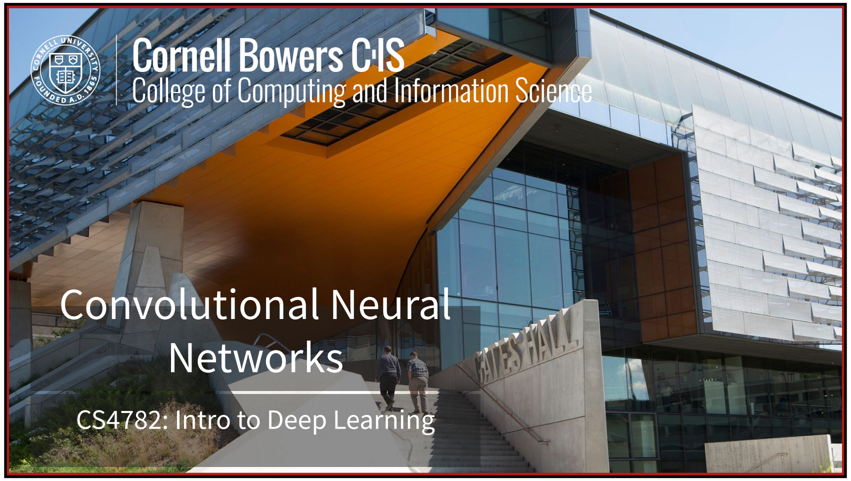


Cornell Bowers C·IS
College of Computing and Information Science

Convolutional Neural Networks

CS4782: Intro to Deep Learning

Cornell Bowers C·IS

Logistics

- HW1 has been released
  - Due next Thursday (February 15)
- Office hours are listed on the course website
- Homework clarifications are listed as pinned posts under HW1 on Ed
- Post questions on Ed

Cornell Bowers C·IS

The Batch Normalization Algorithm

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1\ldots m}\}$;
Parameters to be learned: γ, β
Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

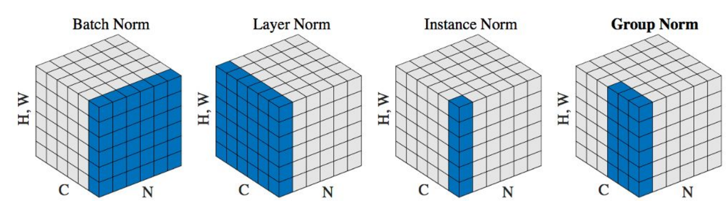
$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

Algorithm 1: Batch Normalizing Transform, applied to activation x over a mini-batch.

Cornell Bowers C·IS

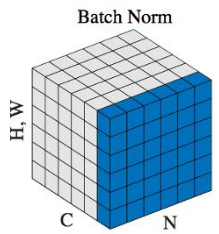
Many Kinds of Normalization Layers

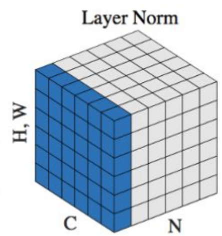


Normalization Methods

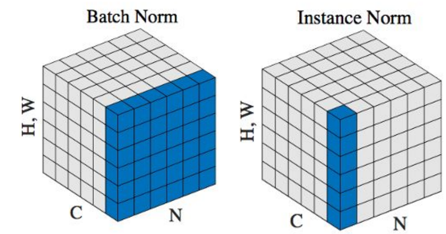
"Group Normalization" by Wu et al., 2018

Layer Normalization



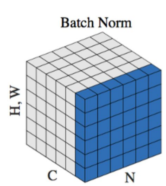


Instance Normalization



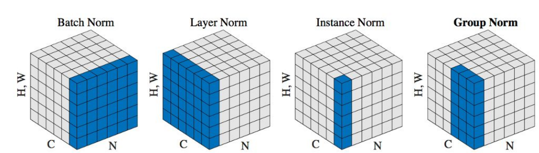
Discuss!

What is the dimension of the mean when you compute the batch norm of a volume of dimension (b x c x h x w)?



Normalization Layers

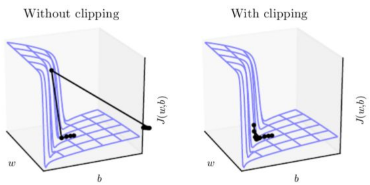
- Normalization layers improve training stability
- Can train with larger learning rates
  - Faster training
- A large learning rate acts as an implicit regularizer
  - Better generalization



Cornell Bowers C·IS

Gradient Clipping

- Exploding gradients result in unstable training
- Optimization is hard when you have very large gradients
- Fixes:
  - Clip by value
  - Clip by norm

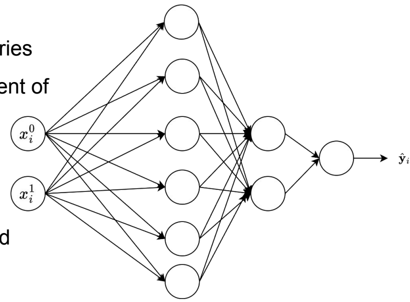


https://neptune.ai/blog/understanding-gradient-clipping-and-how-it-can-fix-exploding-gradients-problem

Cornell Bowers C·IS

So far…

- MLPs learn complex decision boundaries
- Optimization algorithms use the gradient of the loss to find network parameters
- Different training strategies like regularization, early stopping and normalization can improve training and generalization



Cornell Bowers C·IS

Image Classification

input image

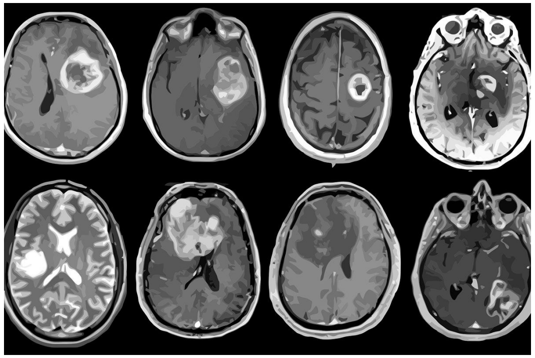
classification → “dog”

input image

classification → “cat”

Cornell Bowers C·IS

Applications in Medicine

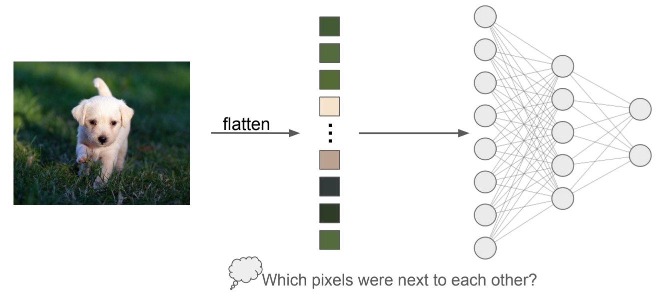


Cornell Bowers CIS

Applications in Autonomous Driving

Cornell Bowers CIS

Why not use a Multi-Layer Perceptron?

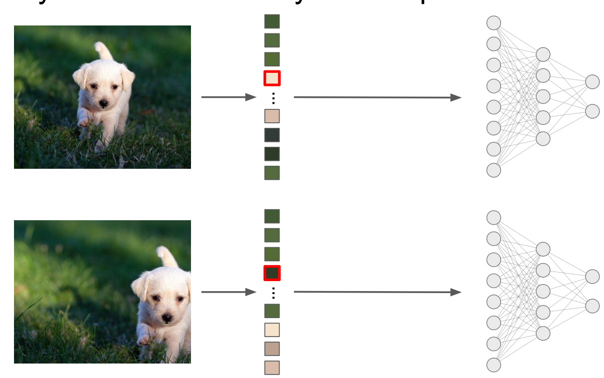


Cornell Bowers CIS

Why not use a Multi-Layer Perceptron?

Cornell Bowers CIS

Why not use a Multi-Layer Perceptron?



Why not use a Multi-Layer Perceptron?

Cornell Bowers CIS

Convolutional Filters

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

"image" convolutional filter

Cornell Bowers CIS

Convolutional Filters

$$\begin{bmatrix} 1_{\times 0} & 1_{\times 1} & 1_{\times 0} & 0 & 0 \\ 0_{\times 1} & 0_{\times 0} & 1_{\times 1} & 1 & 0 \\ 0_{\times 0} & 1_{\times 1} & 0_{\times 0} & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & & \\ & & \\ & & \end{bmatrix}$$

"image" convolutional filter

Cornell Bowers CIS

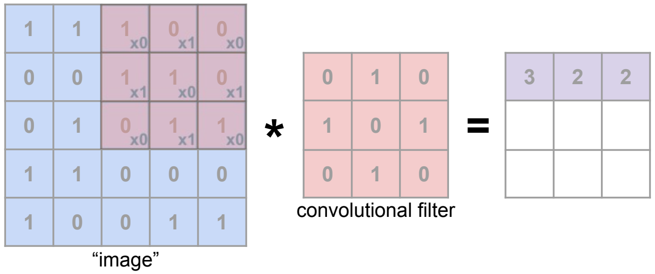
Convolutional Filters

$$\begin{bmatrix} 1 & 1_{\times 0} & 1_{\times 1} & 0_{\times 0} & 0 \\ 0 & 0_{\times 1} & 1_{\times 0} & 1_{\times 1} & 0 \\ 0 & 1_{\times 0} & 0_{\times 1} & 1_{\times 0} & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & 2 & \\ & & \\ & & \end{bmatrix}$$

"image" convolutional filter

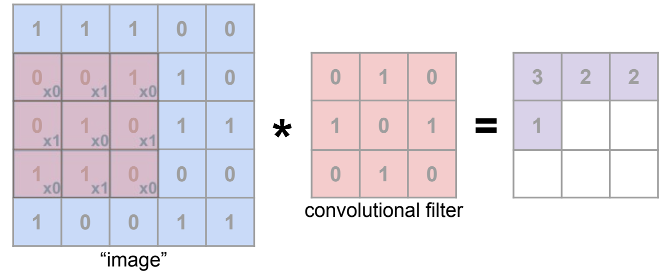
Cornell Bowers C·IS

Convolutional Filters



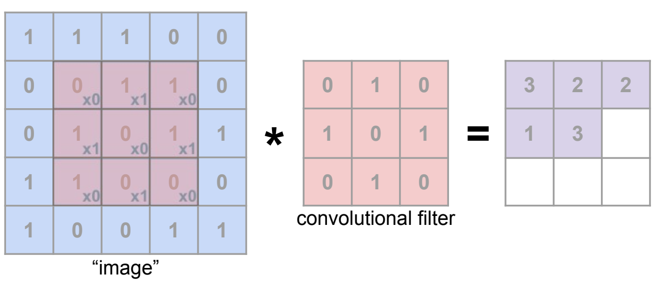
Cornell Bowers C·IS

Convolutional Filters



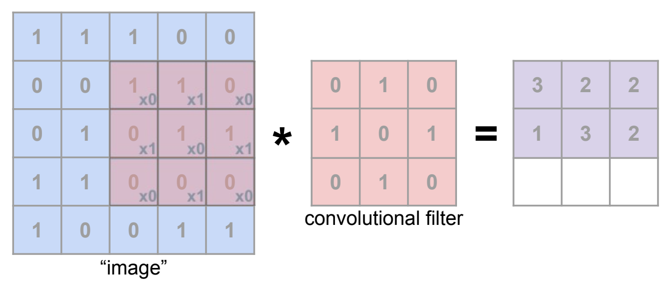
Cornell Bowers C·IS

Convolutional Filters



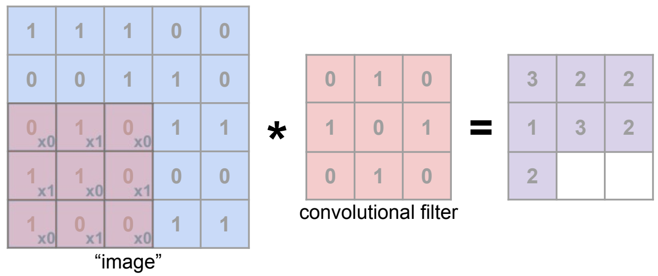
Cornell Bowers C·IS

Convolutional Filters



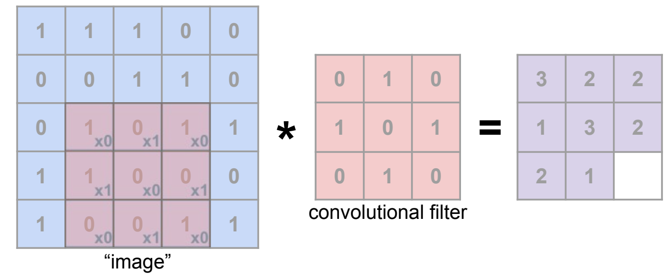
Cornell Bowers C·IS

Convolutional Filters



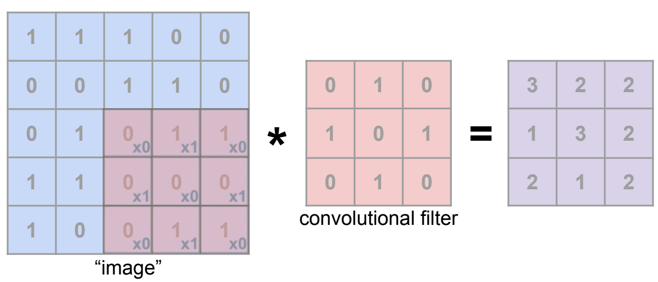
Cornell Bowers C·IS

Convolutional Filters



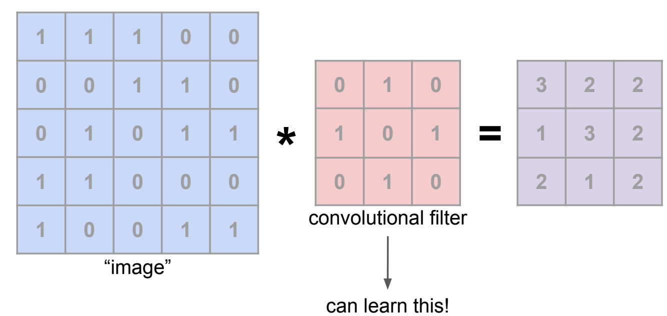
Cornell Bowers C·IS

Convolutional Filters



Cornell Bowers C·IS

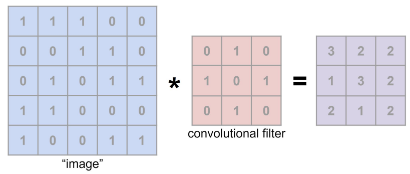
Convolutional Filters



Cornell Bowers C·IS

Convolutional Filters

- Aggregates information from local window around pixel
- Translational invariance
- Reduce number of parameters needed to be learned



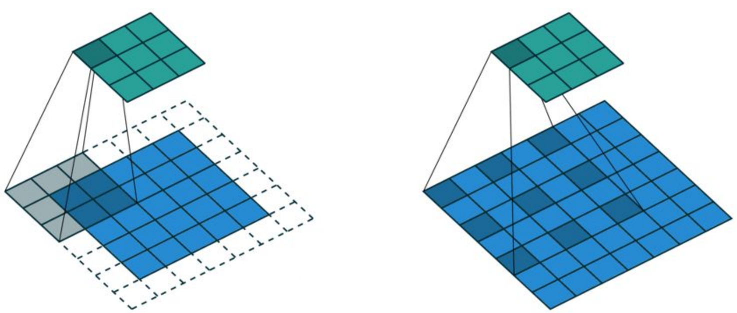
Cornell Bowers C·IS

Discuss with your Neighbor!

Match the following convolutional filters with the output they produce.

Cornell Bowers C·IS

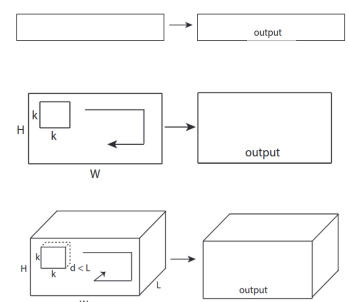
Dilated Convolutions



https://towardsdatascience.com/review-dilated-convolution-semantic-segmentation-9d5a5bd768f5

Cornell Bowers C·IS

1D and 3D Convolutions

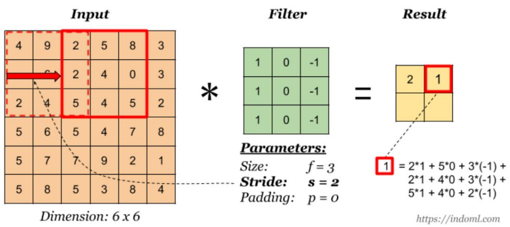


https://wandb.ai/ayush-thakur/dl-question-bank/reports/Intuitive-understanding-of-1D-2D-and-3D-convolutions-in-convolutional-neural-networks---VmlldzoxOTk2MDA

Cornell Bowers C·IS

CNNs - Stride

- Stride controls how many units the filter / the receptive field shift at a time
- The size of the output image shrinks more as the stride becomes larger
- The receptive fields overlap less as the stride becomes larger

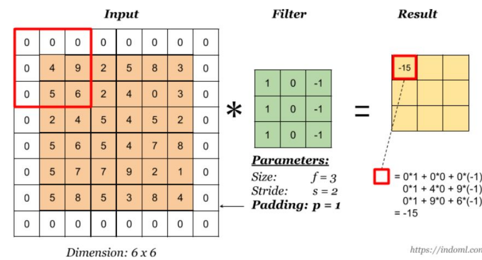


Filter with stride (s) = 2

Cornell Bowers C·IS

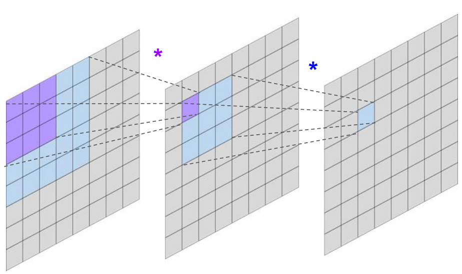
CNNs - Padding

- Padding adds layers of zeros (or other number) around image border
- Prevents image shrinking and loss of information from image boundary



Cornell Bowers C·IS

Stacking Convolutions

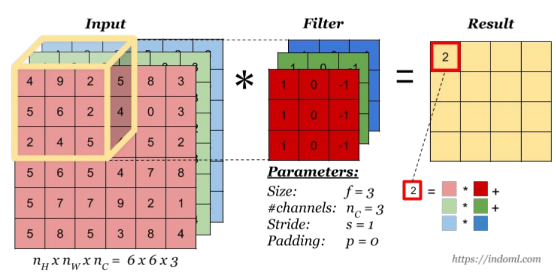


- Size of receptive field increases with each layer
- Capture more complex features

Cornell Bowers C·IS

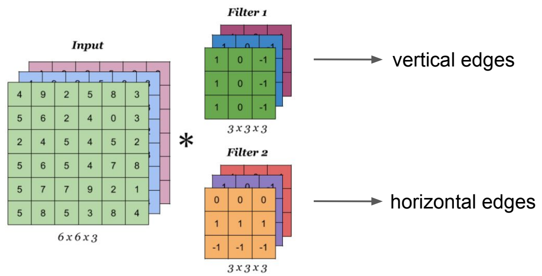
Convolution Over Volumes

What if our input image has more than one channel?



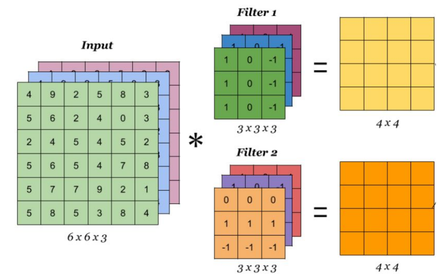
Cornell Bowers C·IS

Convolution Operation with Multiple Filters



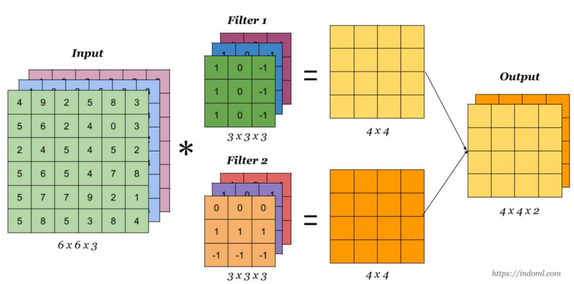
Cornell Bowers C·IS

Convolution Operation with Multiple Filters



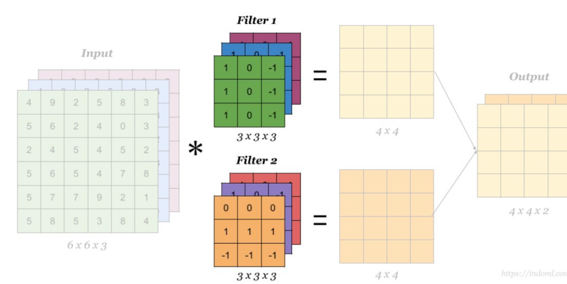
Cornell Bowers C·IS

Convolution Operation with Multiple Filters



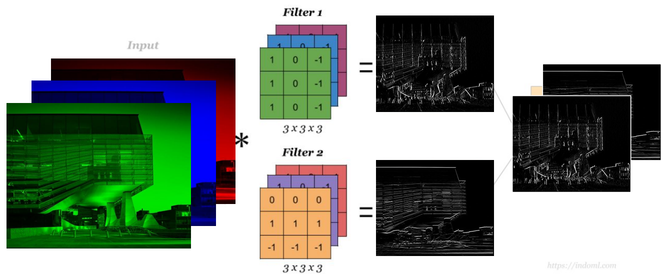
Cornell Bowers C·IS

Convolution Operation with Multiple Filters



Cornell Bowers C·IS

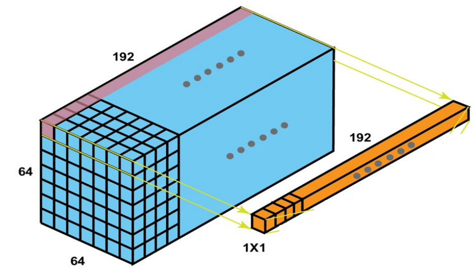
Convolution Operation with Multiple Filters



Cornell Bowers C·IS

Discuss: 1x1 Convolutions

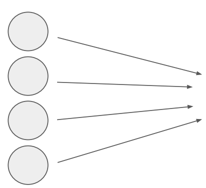
What is the result of convolving a 64x64x192 dimensional cube with a 1x1 filter?



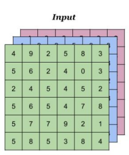
Cornell Bowers C·IS

Convolution Layer

MLP Layer



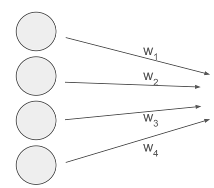
Convolution Layer



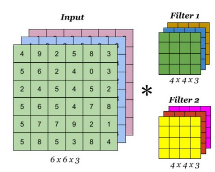
Cornell Bowers C·IS

Convolution Layer

MLP Layer



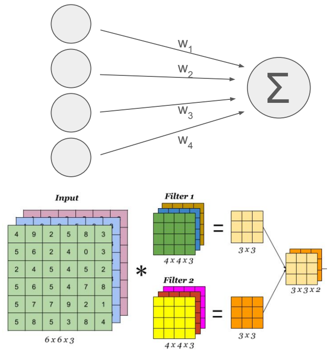
Convolution Layer



Convolution Layer

MLP Layer

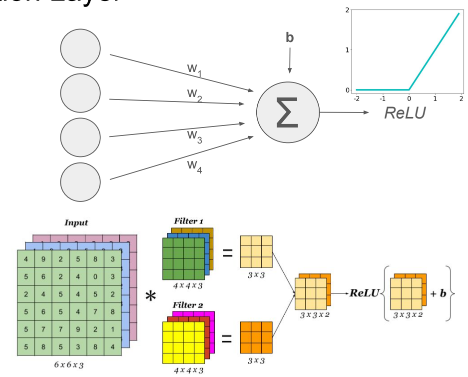
Convolution Layer



Convolution Layer

MLP Layer

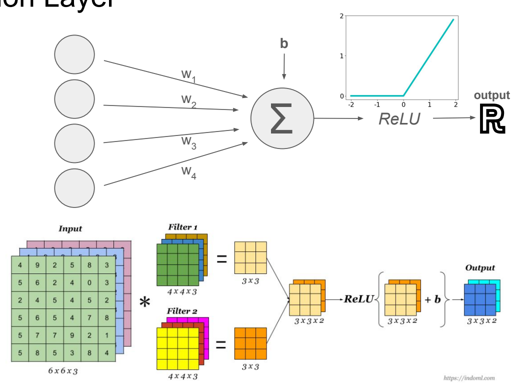
Convolution Layer



Convolution Layer

MLP Layer

Convolution Layer

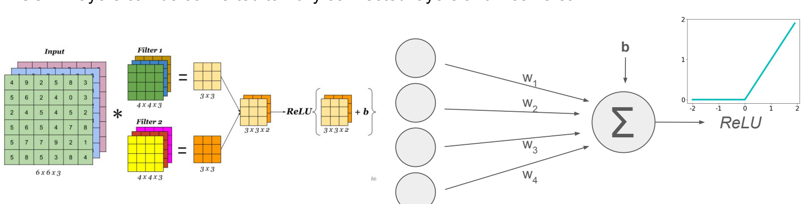


CNN/MLP Equivalence

Differences in a convolution layer:

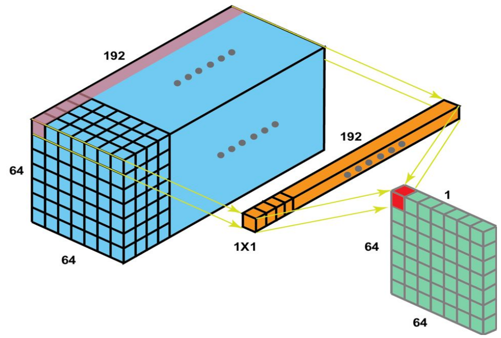
- neurons are connected to a local region
- Weights are shared across multiple parameters

CONV layers can be converted to Fully connected layers and vice versa!



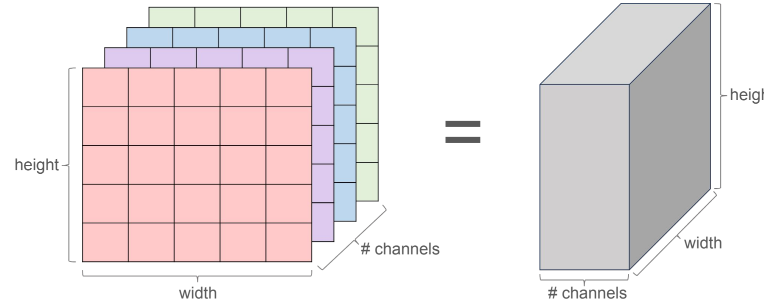
Discuss: Trade-offs between CNNs and MLPs

How would this image change if you used an MLP instead of a 1x1 convolution filter to produce a (64x64x1) feature map? Hint: think about parameter counts and feature interactions.



Cornell Bowers C·IS

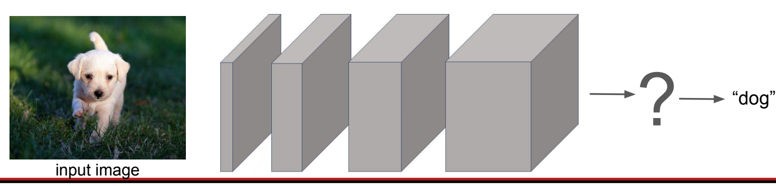
CNN Layer Output Visualization



Cornell Bowers C·IS

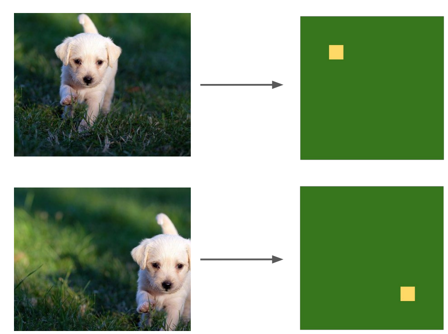
Convolutional Neural Networks (CNNs)

✅ **Convolutions** — Maintain spatial relation between pixels
Reduce number of parameters through weight sharing

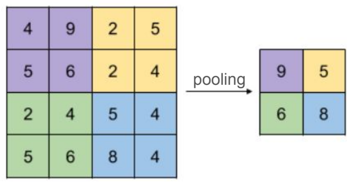


Cornell Bowers C·IS

Ensuring translational invariance

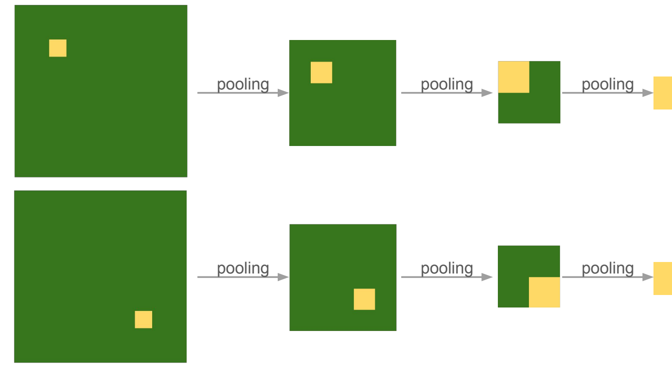


Max Pooling



Cornell Bowers C·IS

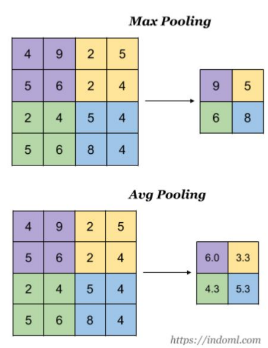
CNNs - Pooling



Cornell Bowers C·IS

CNNs - Pooling

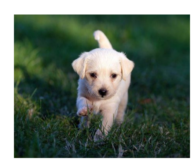
- Down sample feature maps that highlight the most present feature in the patch
- Improve efficiency by reducing computations with downsampling
- Increase receptive field size

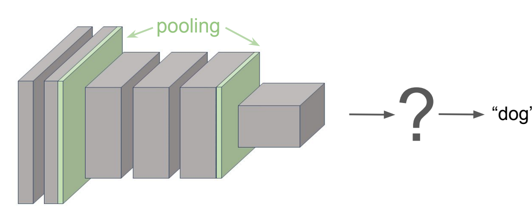


Cornell Bowers C·IS

Convolutional Neural Networks (CNNs)

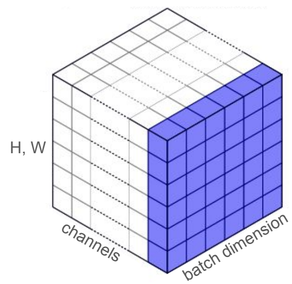
✅ Convolutions	Maintain spatial relation between pixels Reduce number of parameters through weight sharing
✅ **Pooling**	Captures key information from across different areas of the feature maps Together with convolutions allows for translational invariance


input image



Review - Batch Normalization

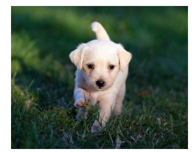
- Normalize channels to mean 0 and variance 1 across each training batch
- Increases speed of training by enabling the use of larger learning rates
- Improves stability of training

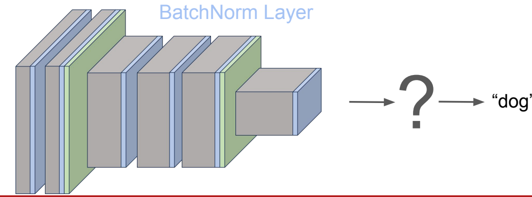


Cornell Bowers C·IS

Convolutional Neural Networks (CNNs)

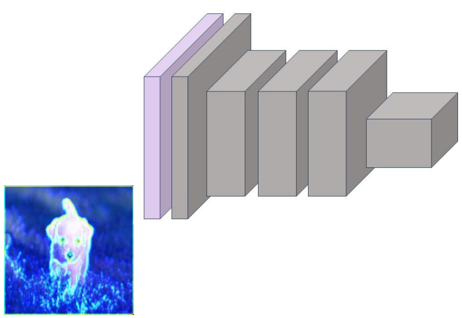
✅ Convolutions	Maintain spatial relation between pixels Reduce number of parameters through weight sharing
✅ Pooling	Captures key information from across different areas of the feature maps Together with convolutions allows for translational invariance
✅ **BatchNorm**	Increases speed and stability of training


input image



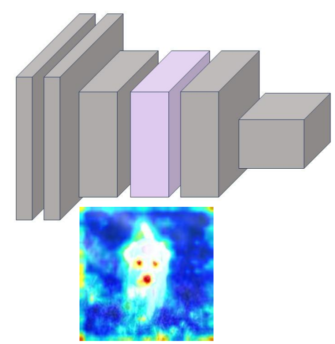
Cornell Bowers C·IS

Convolutional Neural Networks (CNNs)



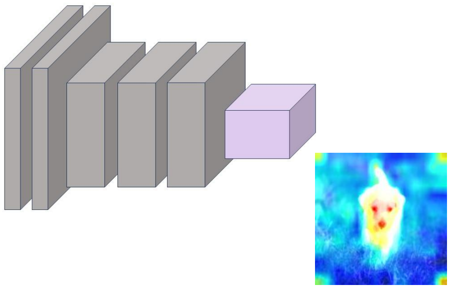
Cornell Bowers C·IS

Convolutional Neural Networks (CNNs)



Cornell Bowers CIS

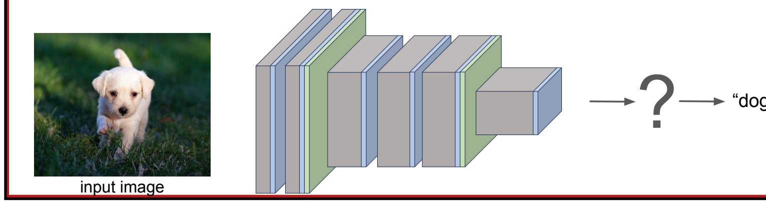
Convolutional Neural Networks (CNNs)



Cornell Bowers CIS

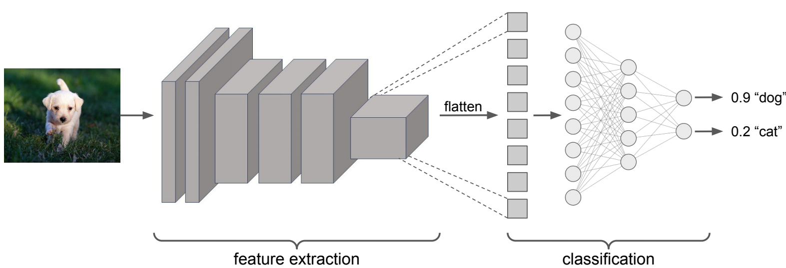
Convolutional Neural Networks (CNNs)

✅ Convolutions	Maintain spatial relation between pixels Reduce number of parameters through weight sharing
✅ Pooling	Captures key information from across different areas of the feature maps Together with convolutions allows for translational invariance
✅ BatchNorm	Increases speed and stability of training



Cornell Bowers CIS

Image Classification



Cornell Bowers CIS

Practical Guide

- Input image dimensions is divisible by 2
- Small conv filters (3x3 or 5x5)
- Zero padding is used to maintain spatial resolution
- Max pooling for downsampling
- Pooling layers have a receptive field of 2 and stride of 2