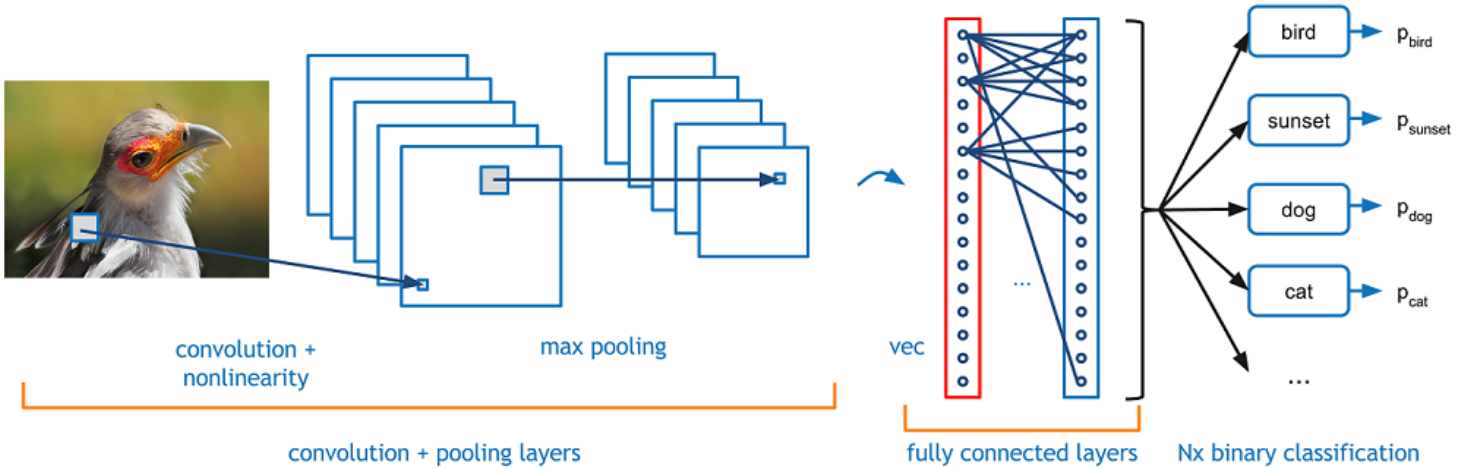
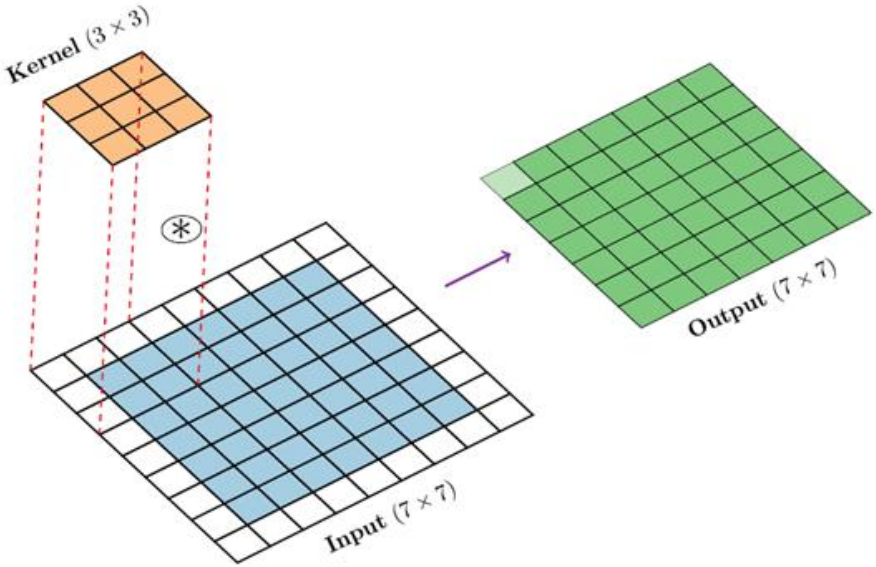
Basics for Convolutional Neural Network

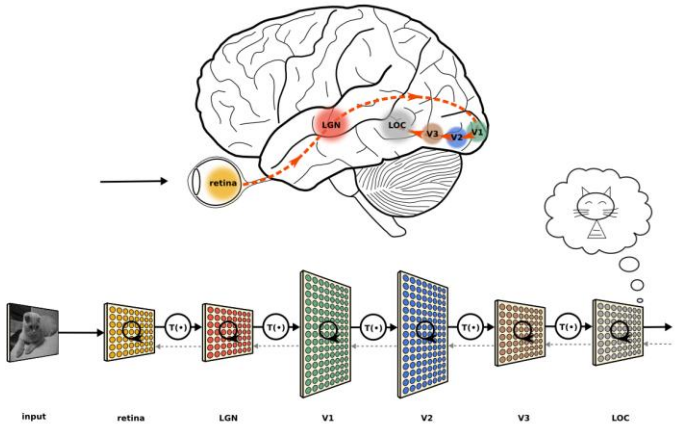
CNN


convolution +
nonlinearity
max pooling
vec
bird
sunset
dog
cat
...
p_{bird}
p_{sunset}
p_{dog}
p_{cat}
convolution + pooling layers
fully connected layers
Nx binary classification

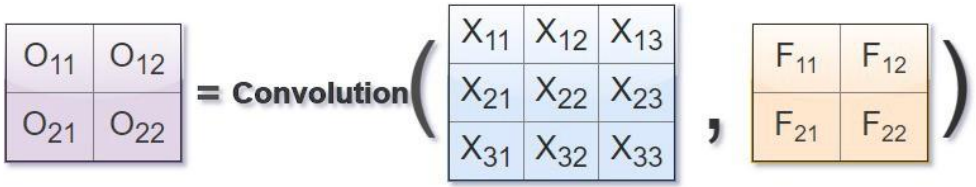
Convolution Operation

Nowadays, we learn kernels from the data.





Learning



$$O_{11} = F_{11}X_{11} + F_{12}X_{12} + F_{21}X_{21} + F_{22}X_{22}$$

$$O_{12} = F_{11}X_{12} + F_{12}X_{13} + F_{21}X_{22} + F_{22}X_{23}$$

$$O_{21} = F_{11}X_{21} + F_{12}X_{22} + F_{21}X_{31} + F_{22}X_{32}$$

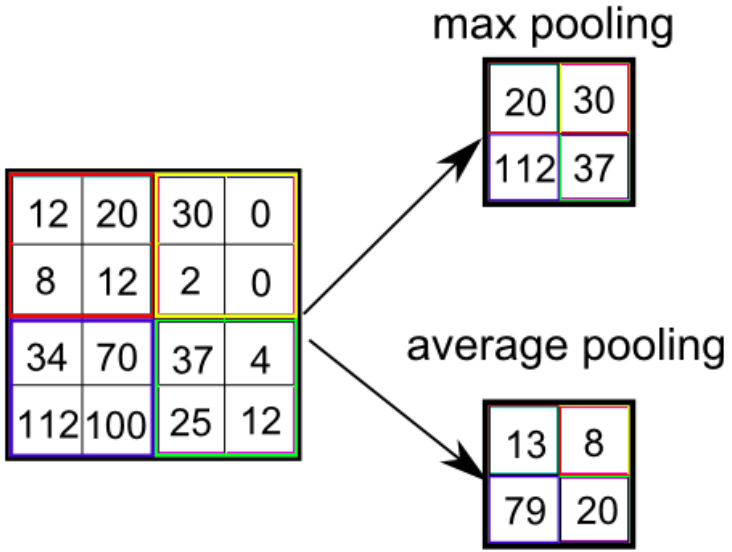
$$O_{22} = F_{11}X_{22} + F_{12}X_{23} + F_{21}X_{32} + F_{22}X_{33}$$

- Details:
- https://www.slideshare.net/EdwinEfranJimnezLepe/example-feedforward-backpropagation
- https://medium.com/@2017csm1006/forward-and-backpropagation-in-convolutional-neural-network-4dfa96d7b37e

Pooling

e.g. kernel size = 2, stride = 2 for both width and height.

The kernel size for pooling can be an even number.



CNN Structures
Image Classification

Image Classification

CU — Convolutional Unit

FC — Fully-connected Or Linear Layer

DU — Down-sampling Unit

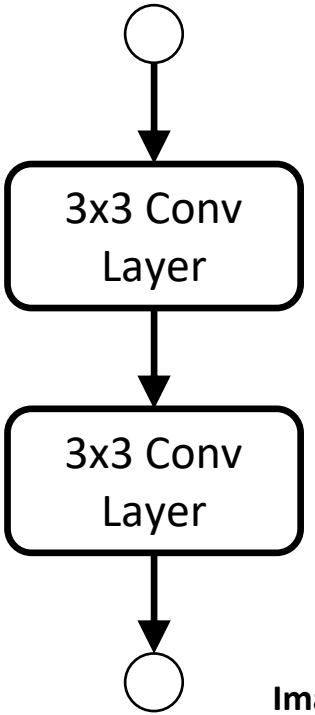
GAP — Global Avg. Pooling

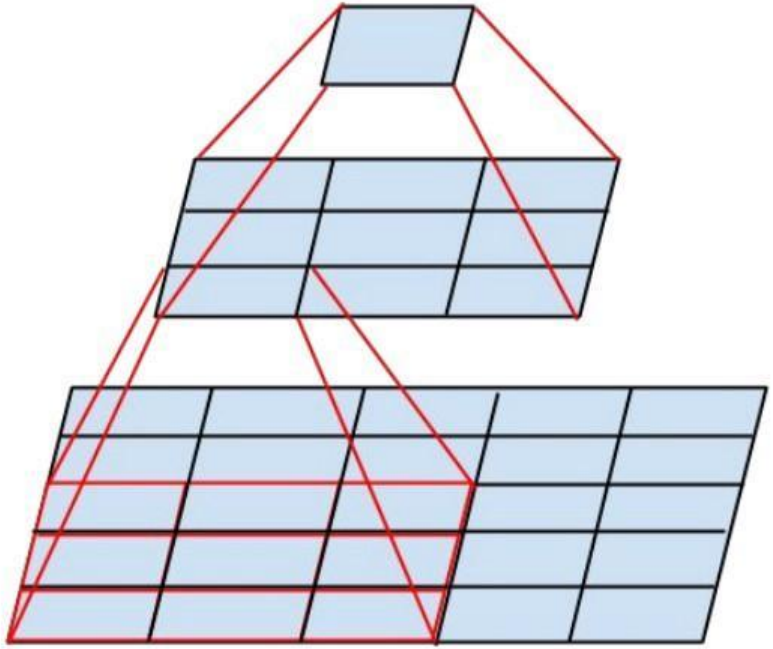
CU → DU → CU → DU → CU → GAP → FC → Cat

28 x 28 = $[28]^2$

$[28]^2$ $[14]^2$ $[14]^2$ $[7]^2$ $[7]^2$ $[1]^2$

Convolutional Unit (CU) - VGG





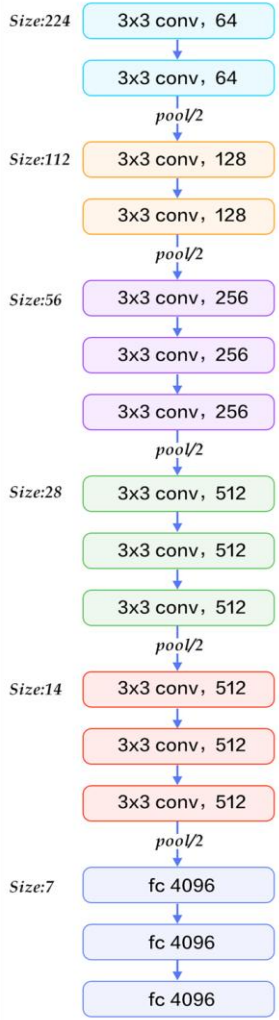
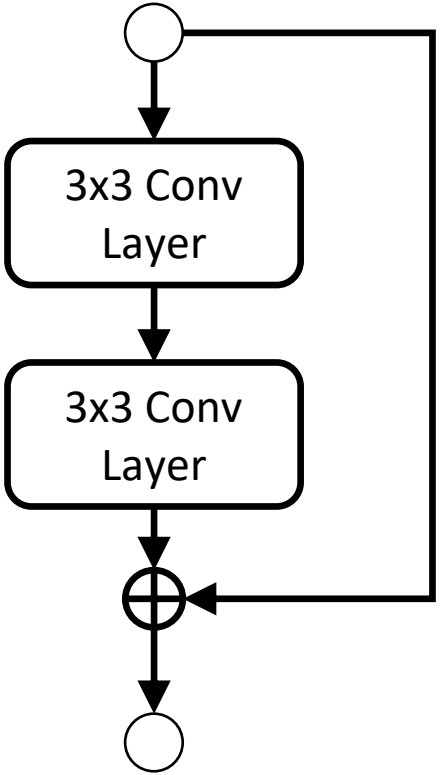


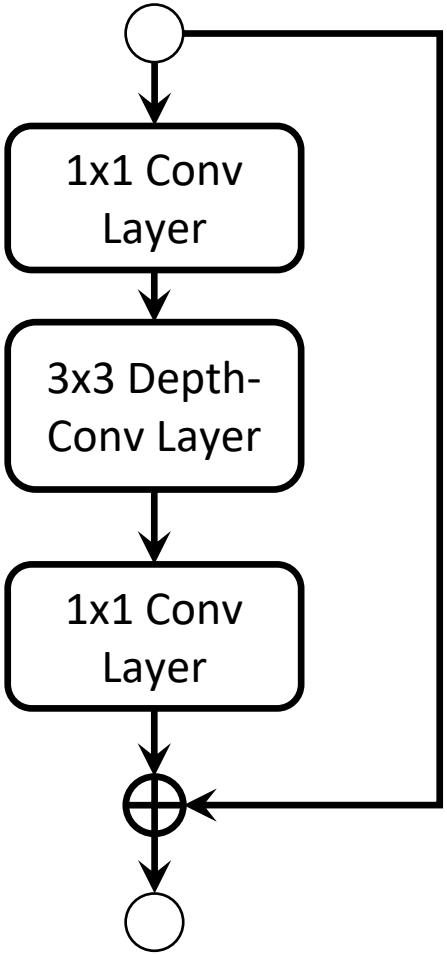
Image Source (Inception): Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." *CVPR*. 2016.

Basic Block in ResNet



ResNet: He, Kaiming, et al. "Deep residual learning for image recognition." CVPR. 2016.

- Residual Connection
- Element-wise addition of input and output
- Improves gradient flow and accuracy
- In ResNet-18 and ResNet-34
- Still computationally expensive
  - Hard to train very deep networks (> 100 layers)



Bottleneck in ResNet

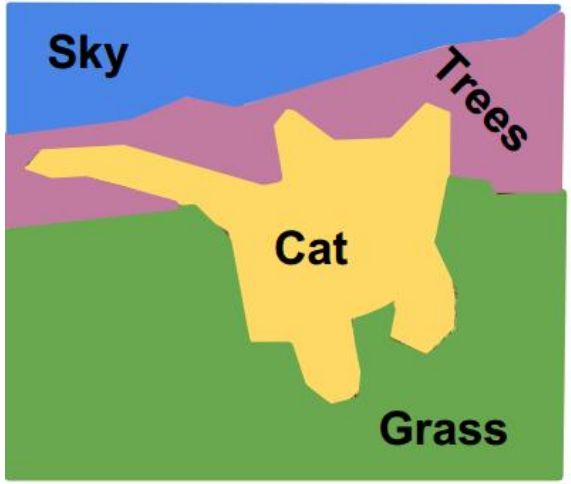
- Used in ResNet-50, ResNet-101, ResNet-152, etc...
- Computationally Efficient

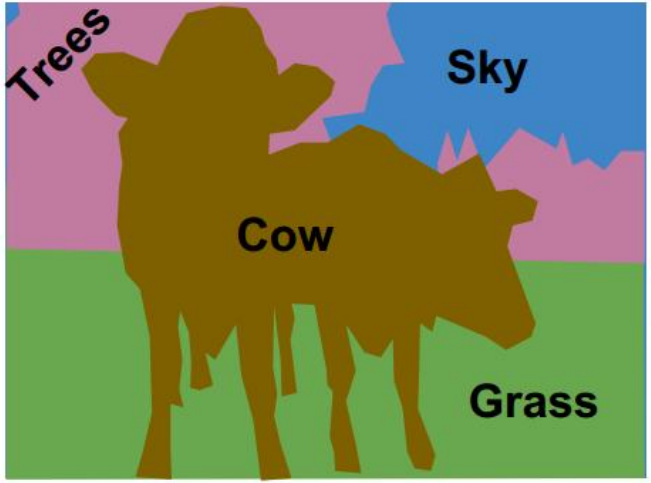
Influence:

- Bottleneck unit with Depth-wise convs
  - MobileNetv2
  - ShuffleNetv2

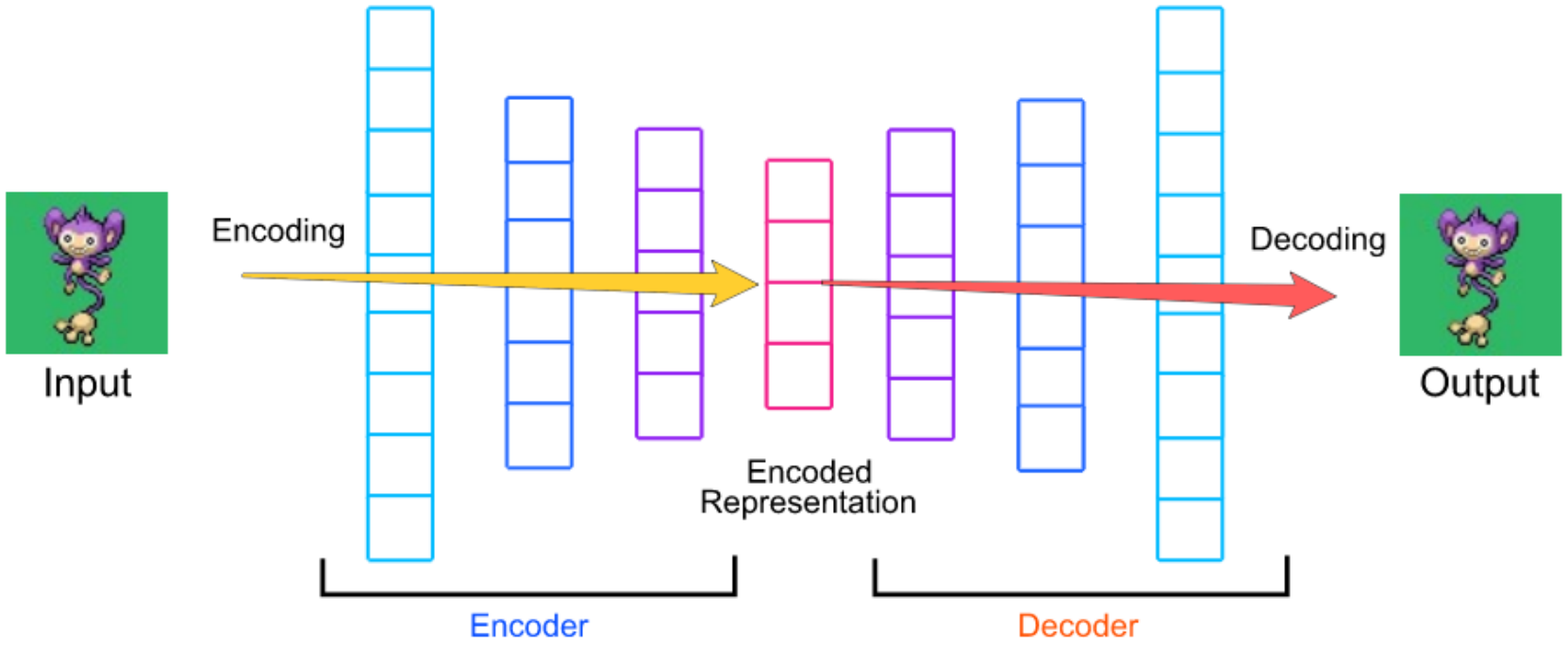
- **MobileNetv2:** Sandler, Mark, et al. "Mobilenetv2: Inverted residuals and linear bottlenecks." CVPR, 2018.
- **ShuffleNetv2:** Ma, Ningning, et al. "Shufflenet v2: Practical guidelines for efficient cnn architecture design." ECCV, 2018.

CNN Structures
Semantic Segmentation

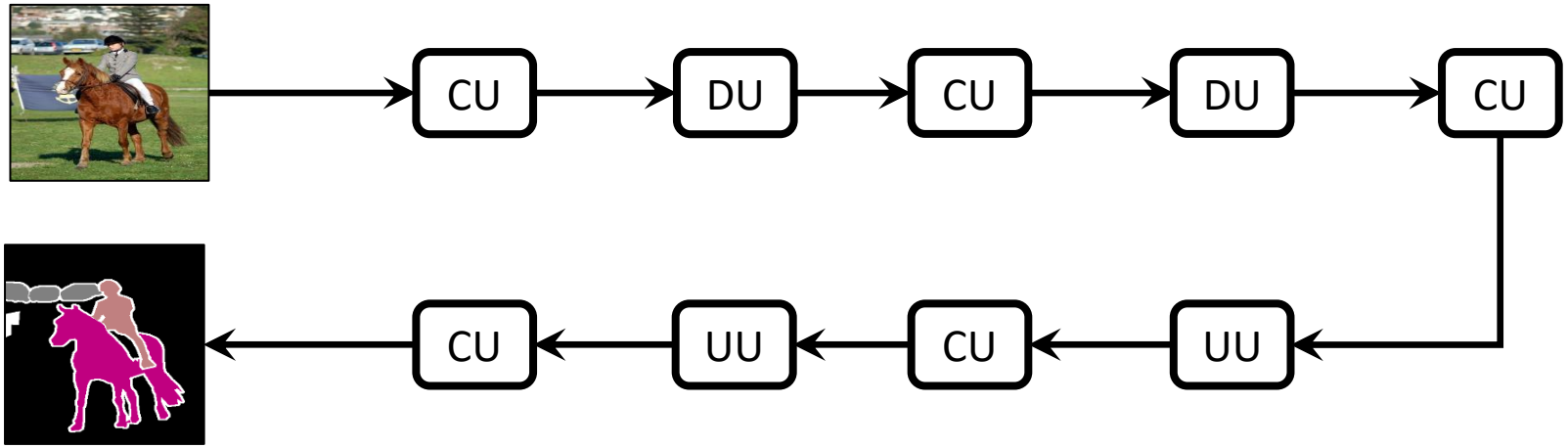




Encoder-Decoder



Encoder-Decoder in Semantic Segmentation



UU Up-sampling Unit

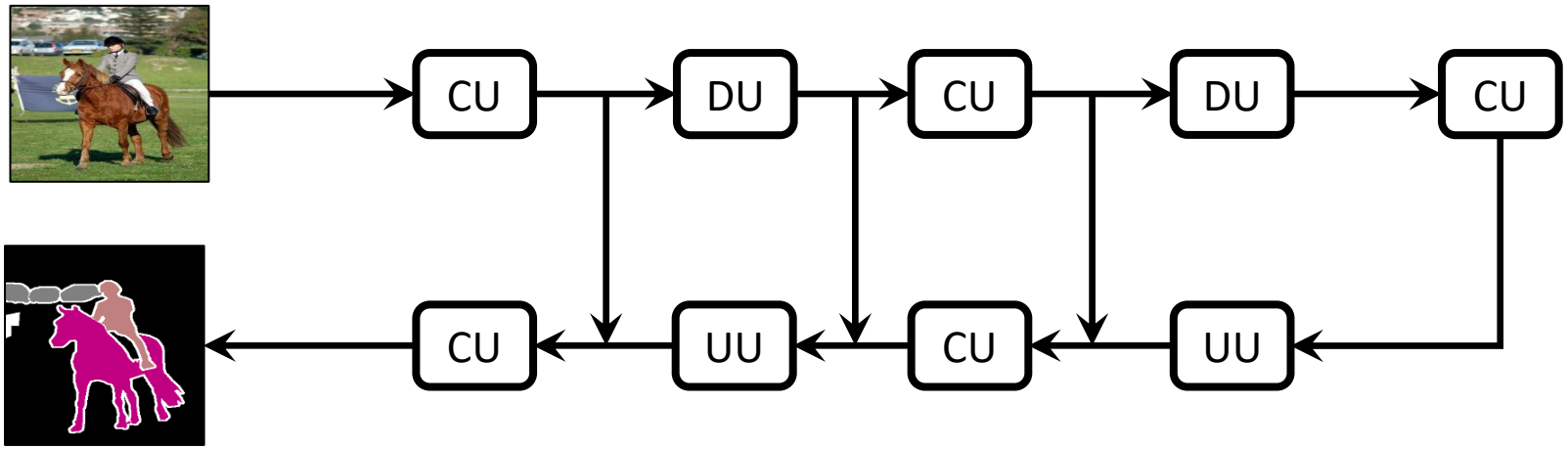
CU Convolutional Unit

FC Fully-connected Or Linear Layer

DU Down-sampling Unit

GAP Global Avg. Pooling

U-Net



UU Up-sampling Unit

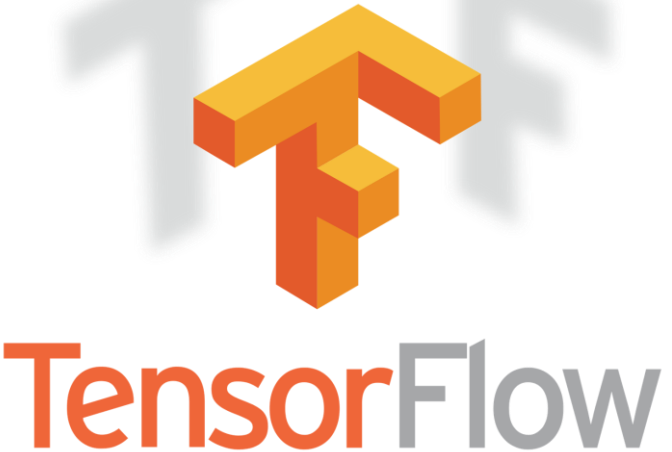
CU Convolutional Unit

FC Fully-connected Or Linear Layer

DU Down-sampling Unit

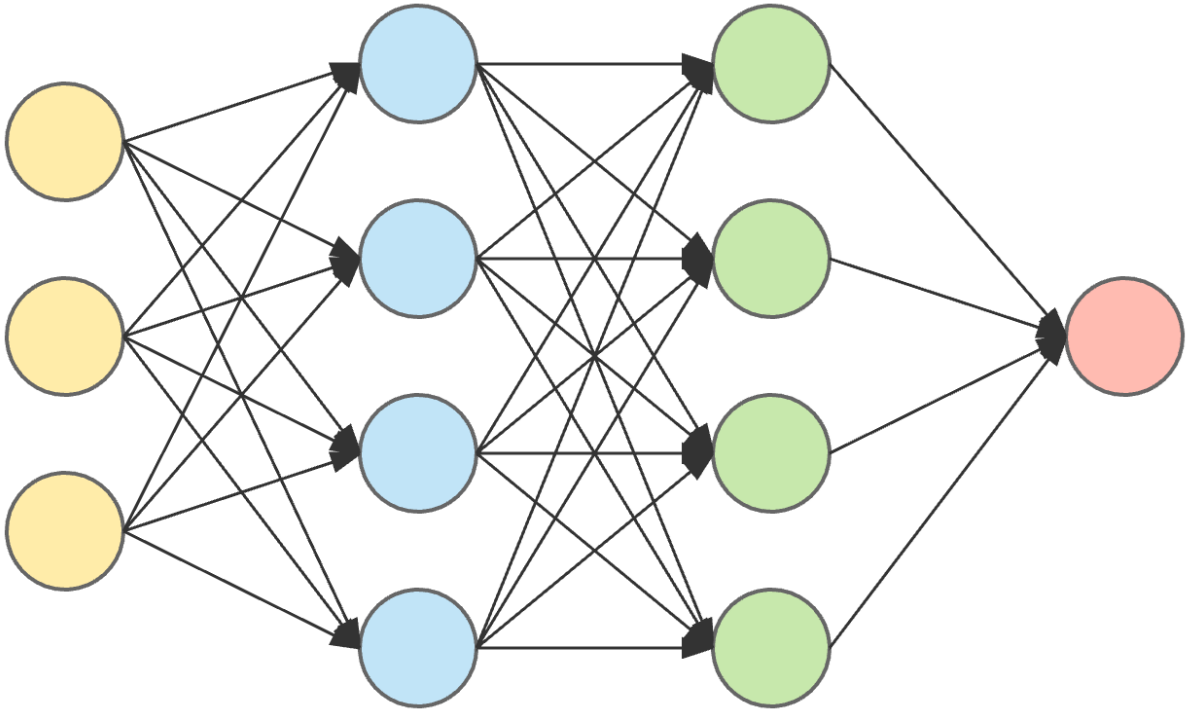
GAP Global Avg. Pooling

Deep Learning Libraries



Homework 5
Convolutional Neural Networks
In PyTorch

Neural Network (Q1)



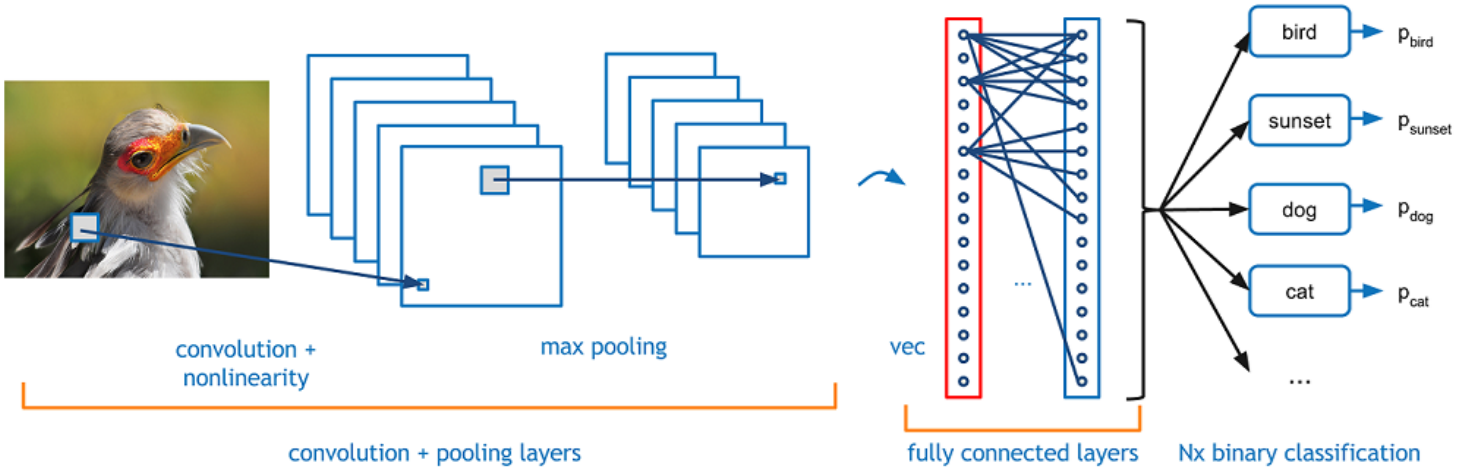
Convolutional Neural Network (Q2)

Conv

Pool

FC

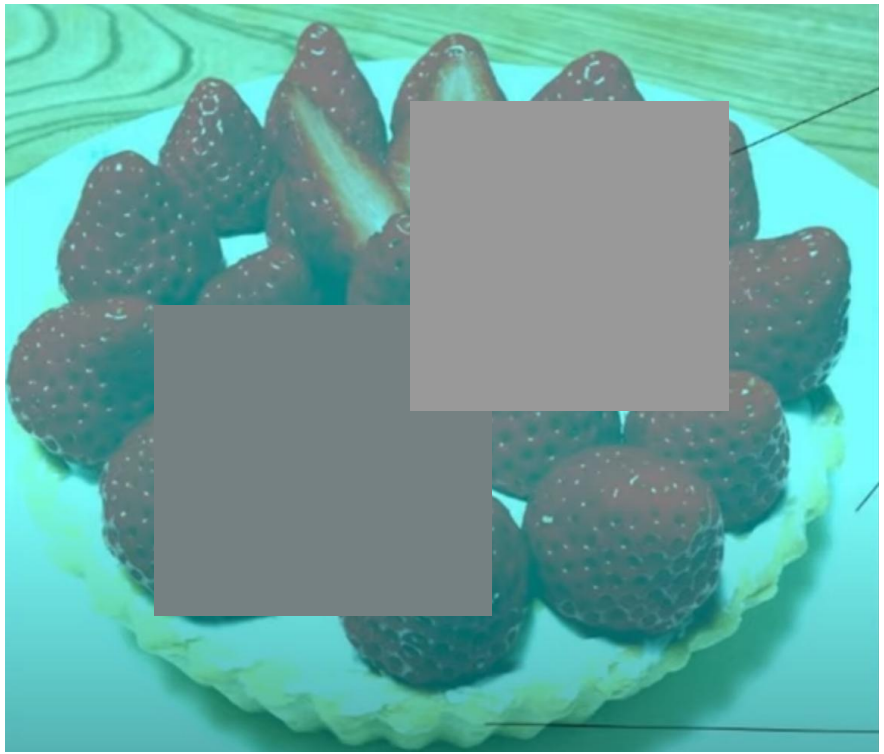
Cross Entropy



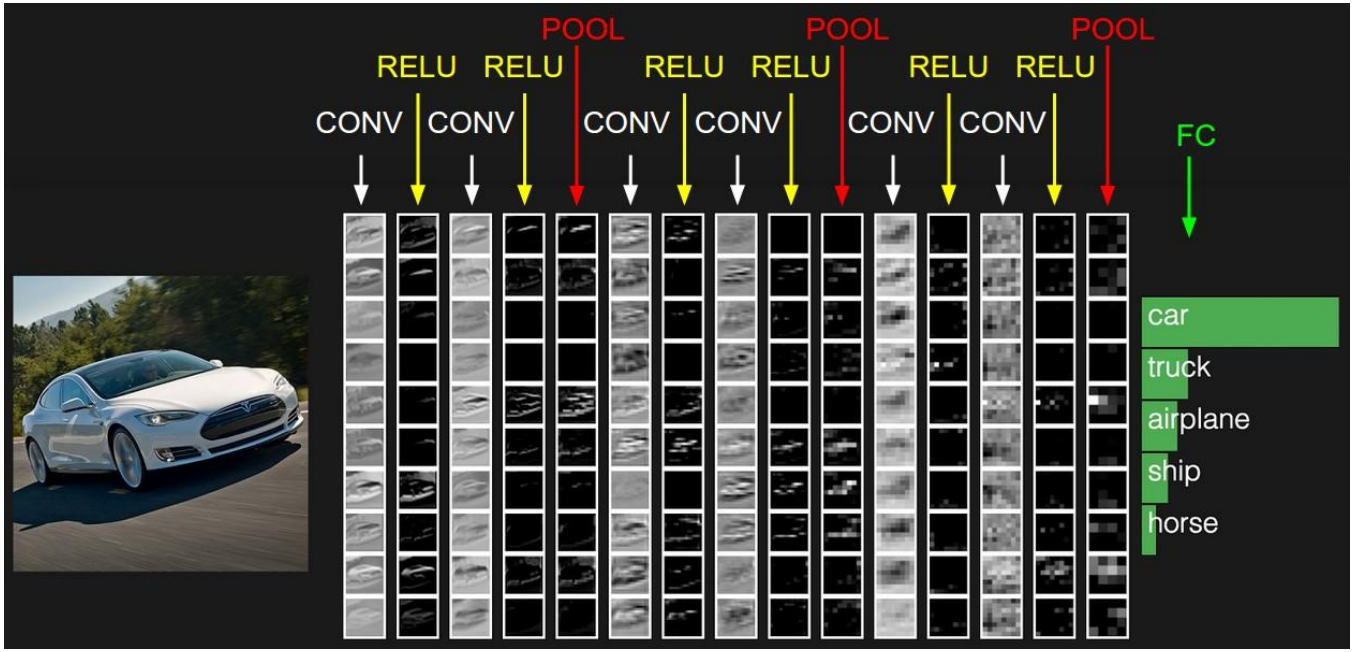
Yellow or Blue?

Color Normalization (Q3)

What’s the Color of the Strawberry



Deep Convolutional Neural Network (Q4)



Make the Design More Flexible

Input:

[8, 16, 32, "pool"]

Layer	Output Size	Output Channels
Input	30 x 30	3
Conv	28 x 28	8
ReLU	28 x 28	8
Conv	26 x 26	16
ReLU	26 x 26	16
Conv	24 x 24	32
ReLU	24 x 24	32
Max Pool	12 x 12	32
Linear	5	

Data Augmentation (Q5)

Random Affine Transformation