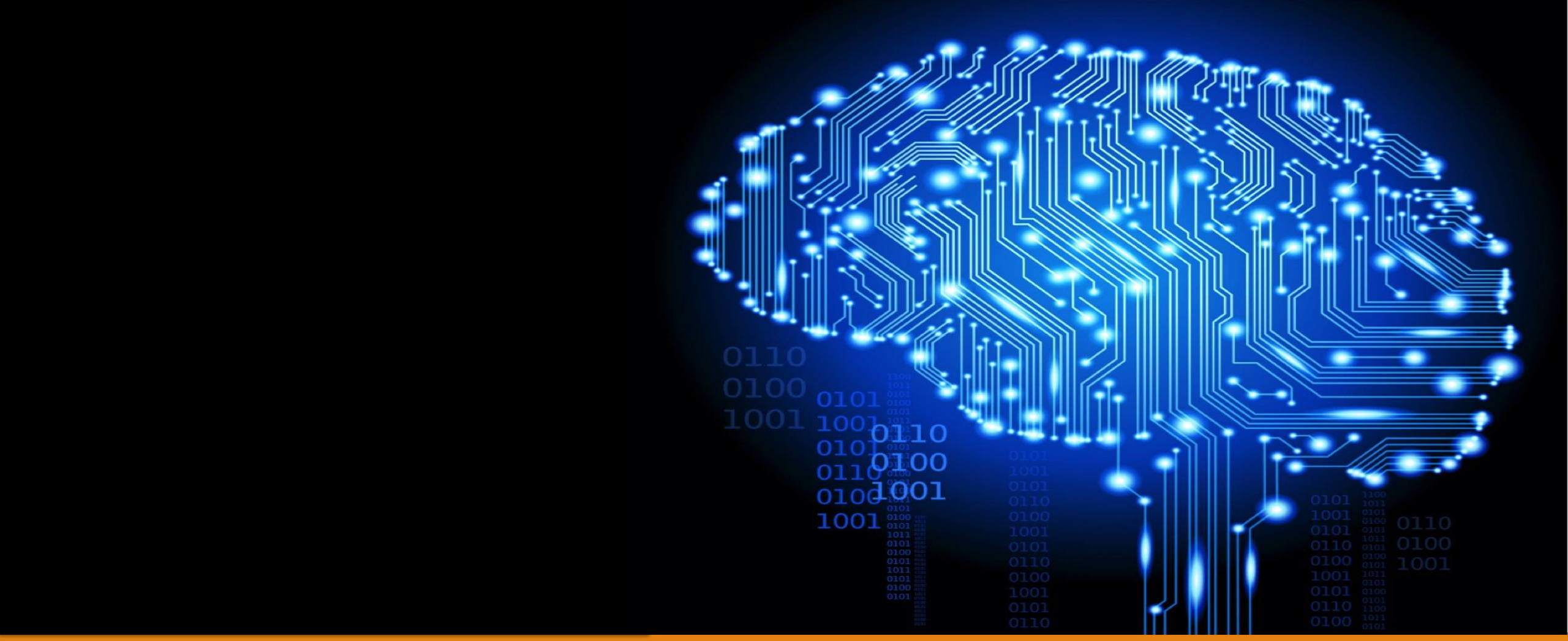


Lecture 5: Modern ConvNets

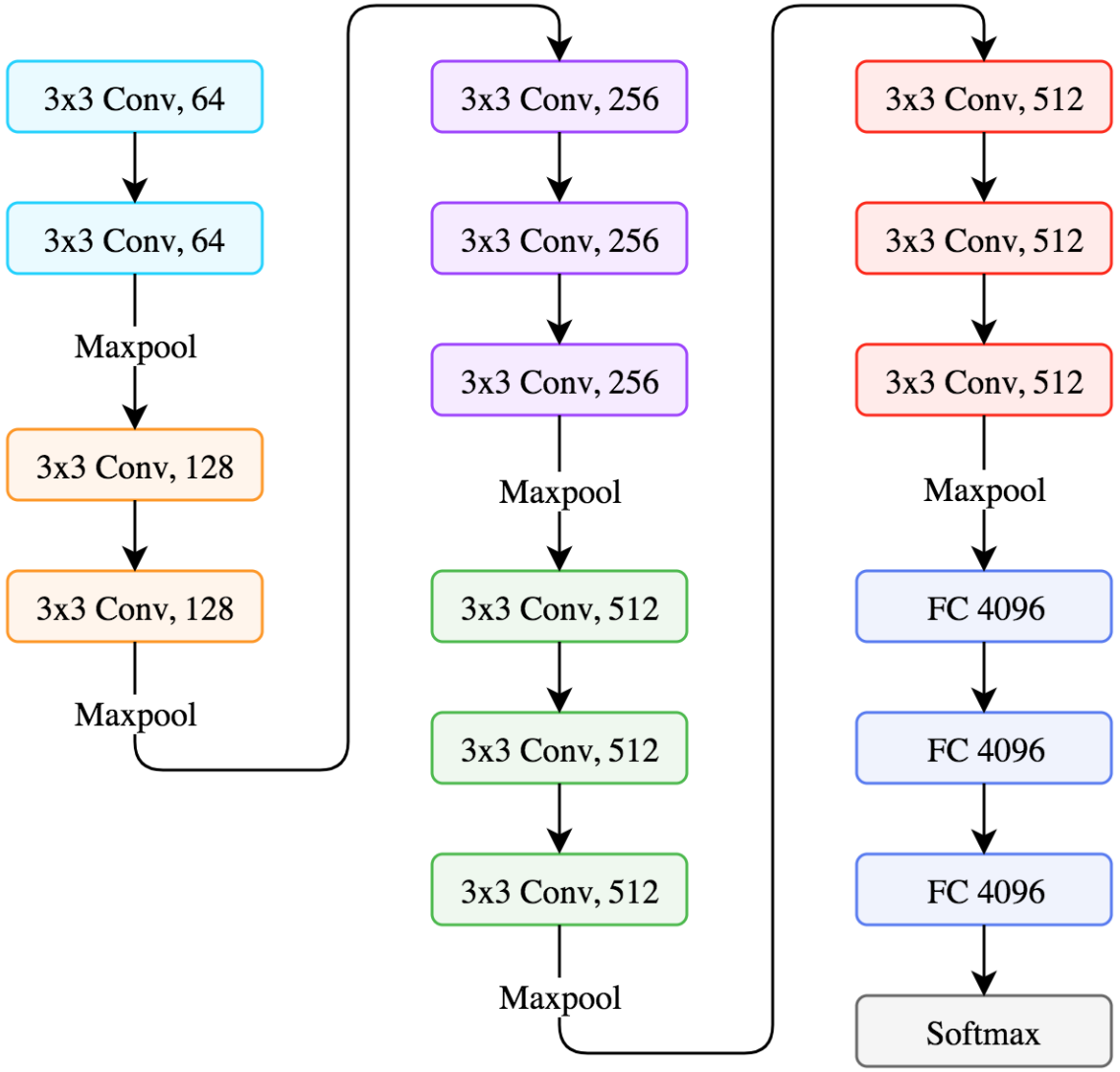
Efstratios Gavves

Lecture overview

- Case study II: VGG
- Vanishing and exploding gradients
- Case study III: Inception
- Case study IV: Resnet, Denset, Highway Net
- Depth and trainability
- Specialized architectures

Case study II: VGG16

- 7.3% error rate in ImageNet
- Compared to 18.2% of AlexNet



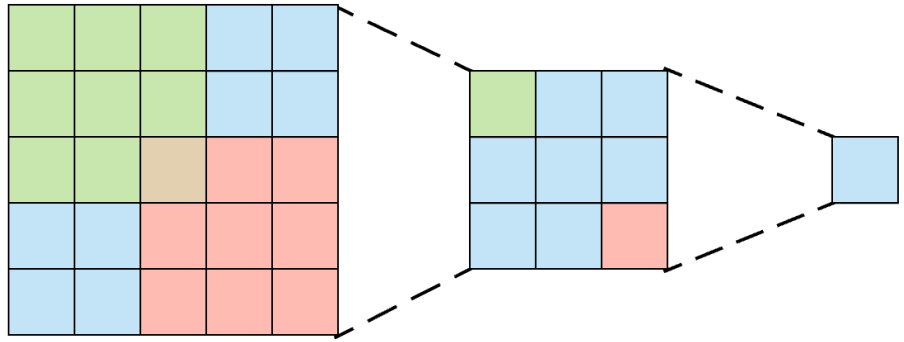
Picture credit: [Arden Dertat]

Characteristics

- Input size: 224×224
- Filter sizes: 3×3
- Convolution stride: 1
  - Spatial resolution preserved
- Padding: 1
- Max pooling: 2×2 with a stride of 2
- ReLU activations
- No fancy input normalizations
  - No Local Response Normalizations
- Although deeper, number of weights is not exploding

Effective receptive field

- The number of actual pixels contributing at the activation in l-th layer
  - Not just the ones from the previous layers, but the others before that too
- A large filter can be replaced by a deeper stack of successive smaller filters
  - Two 3×3 filters have the receptive field of one 5×5
  - Three 3×3 filters have the receptive field of one 7×7
- Depth increases effective receptive field
  - Every "pixel" in the 2nd layer corresponds to a 3x3 region in the previous one



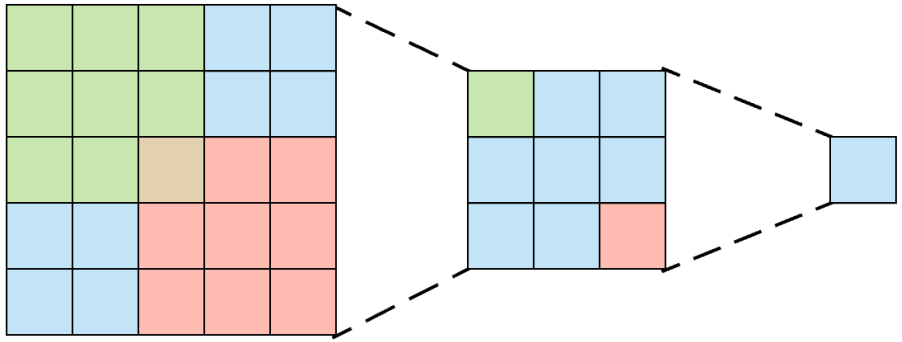
Picture credit: Arden Dertat

Why 3x3 filters?

- The smallest possible filter to captures the “up”, “down”, “left”, “right”
- Deeper stacks of smaller filters likely more powerful than single large filter
  - Three more nonlinearities for the same “size” of pattern learning
  - Fewer parameters and regularization

A stacks of two small filters: $(3 \times 3 \times C) \times 3 = 27 \cdot C$

One large filter: $7 \times 7 \times C \times 1 = 49 \cdot C$

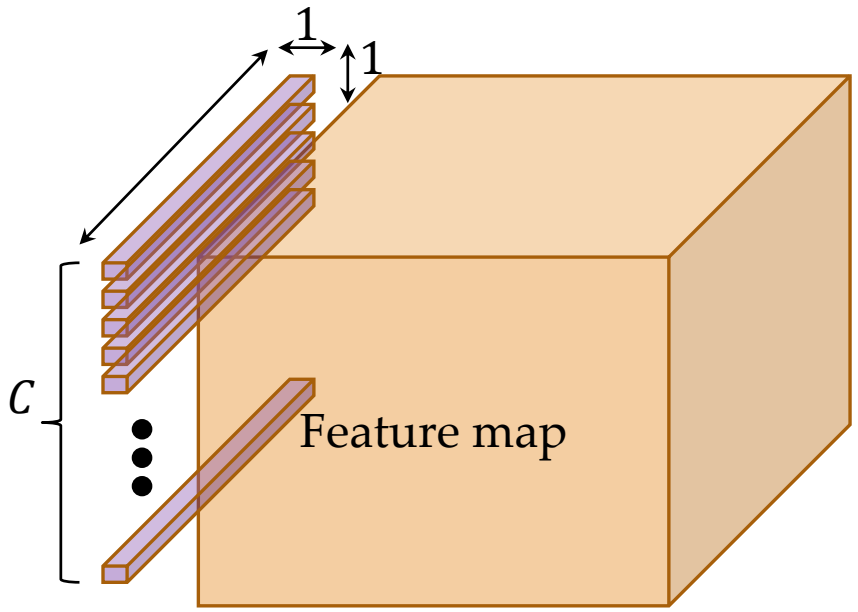


5x5 receptive field 3x3 receptive field

Picture credit: [Arden Dertat]

Even smaller filters?

- Also $1x1$ filters are used
- Followed by a nonlinearity
- Increasing nonlinearities without affecting receptive field sizes
  - Linear transformation of the input channels



Training

- Batch size: 256
- SGD with momentum ($\gamma = 0.9$)
- Weight decay $\lambda = 5 \cdot 10^{-4}$
- Dropout on first two fully connected layers
- Starting learning rate $\eta_0 = 10^{-2}$
  - Divided by 10 when validation accuracy stops improving
  - 3X decreasing learning rate
- Smaller filters → Faster training
- Reported depth as potential regularizer