Tutorial on Convolutional Neural Networks

CAMERON FABBRI

Focusing on Two Papers

- Gradient-Based Learning Applied to Document Recognition [1]
- ImageNet Classification with Deep Convolutional Neural Networks [11]

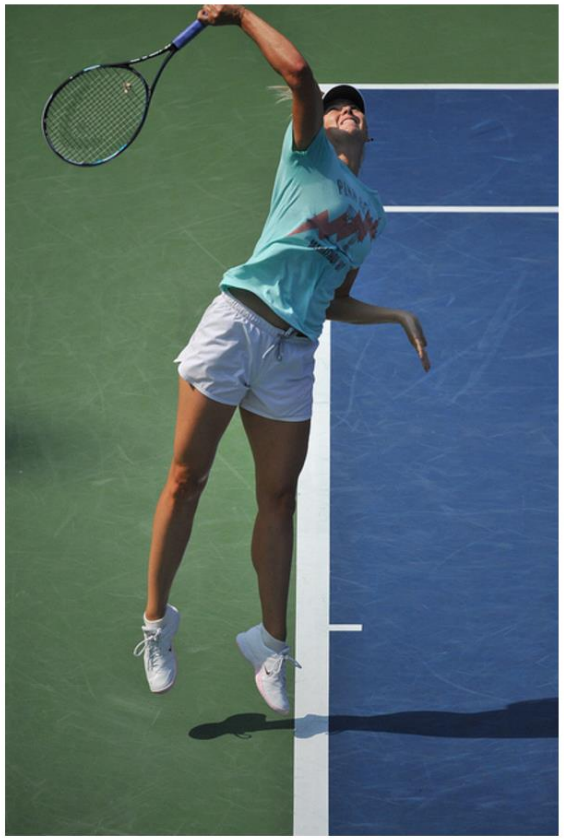
Outline

- Motivation
- Neuron Recap
- Generative vs Discriminative Models
- Convolutional Neural Network Components
  - Convolution
  - Pooling
  - Fully Connected Layers
  - Dropout
- Alexnet
- Generative Models
- Conclusion

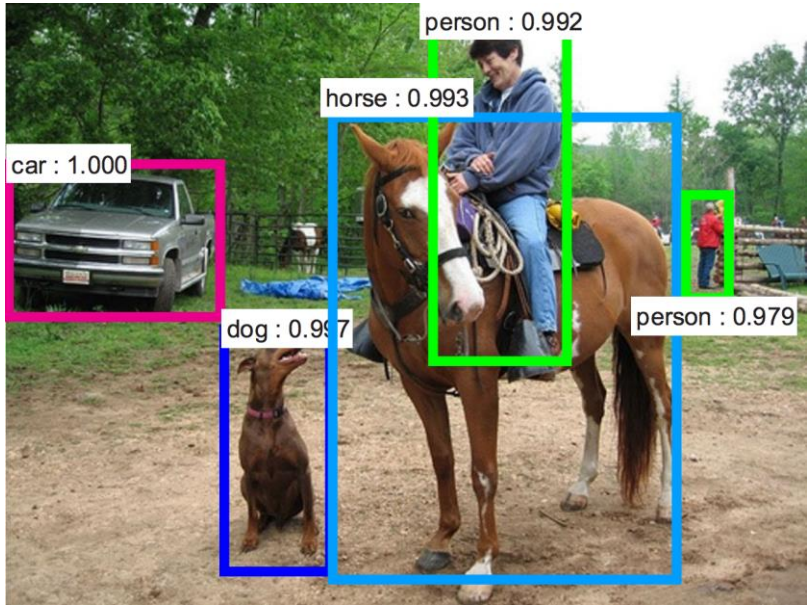
Motivation

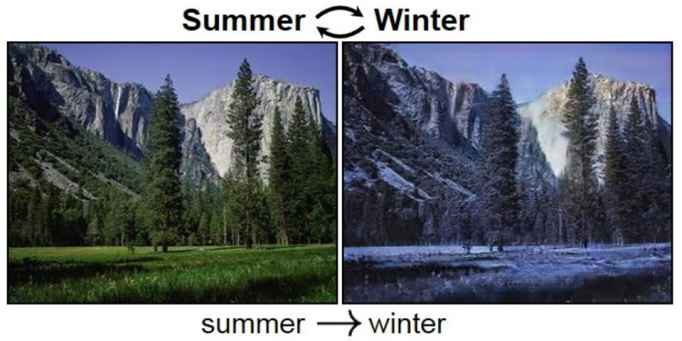
- Most real world problems are nonlinear in nature
- Reliable feature representations
- Hand crafted features require extreme domain knowledge and are expensive to acquire
- Neural Networks learn feature representation from the data directly

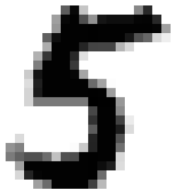
Motivation and Inspiring Results

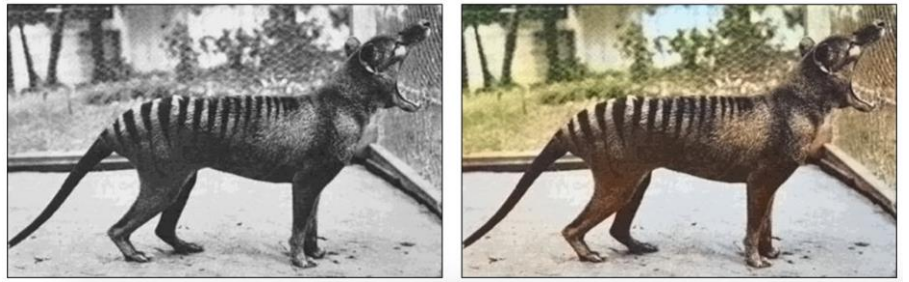


a woman is playing tennis on a tennis court



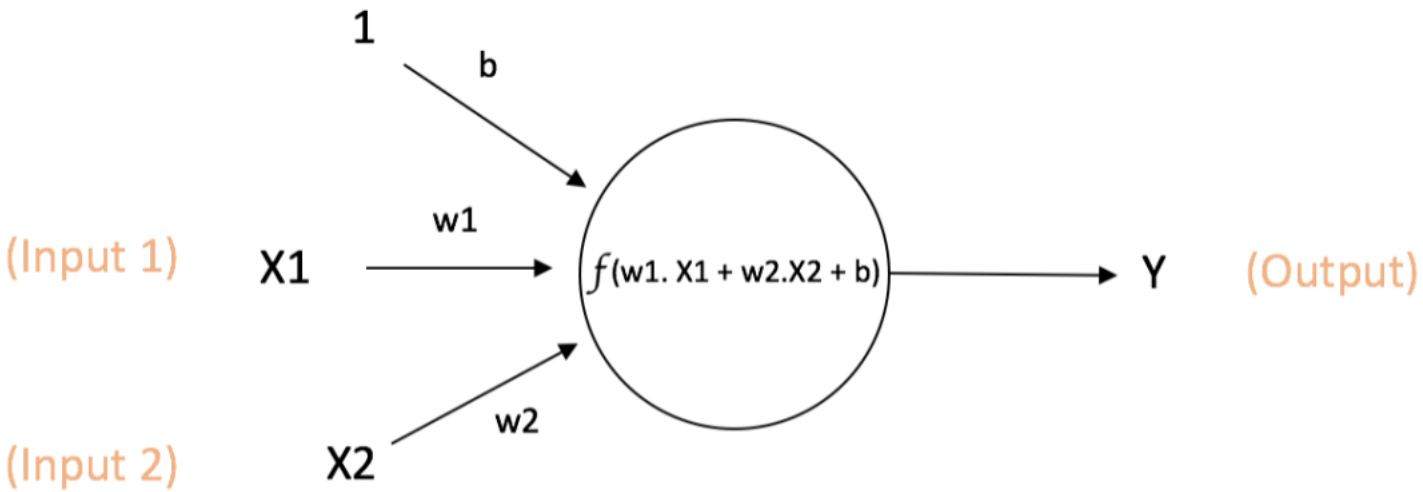






[2,5,6,7,15,16]

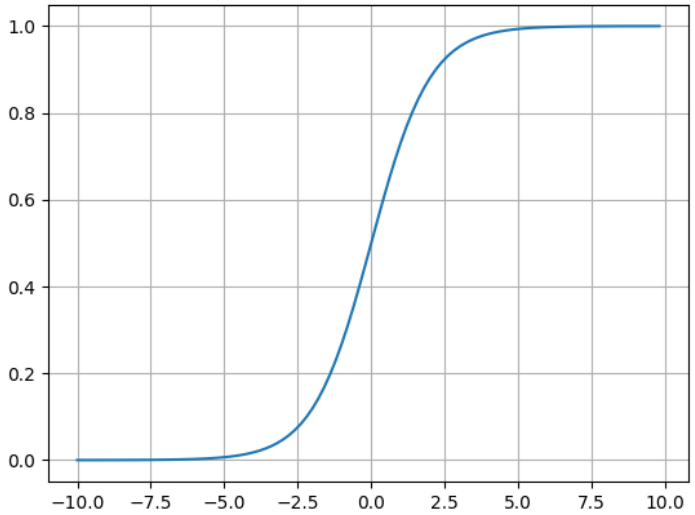
Neuron Recap



Output of neuron = Y= f(w1. X1 + w2.X2 + b)

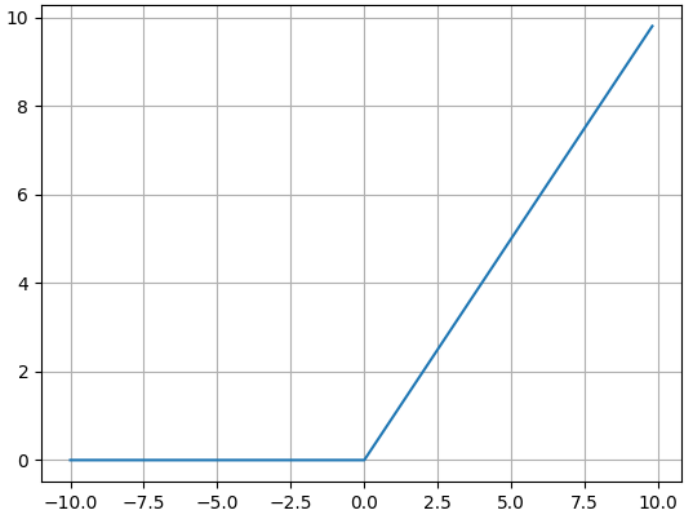
Activation Function

Sigmoid



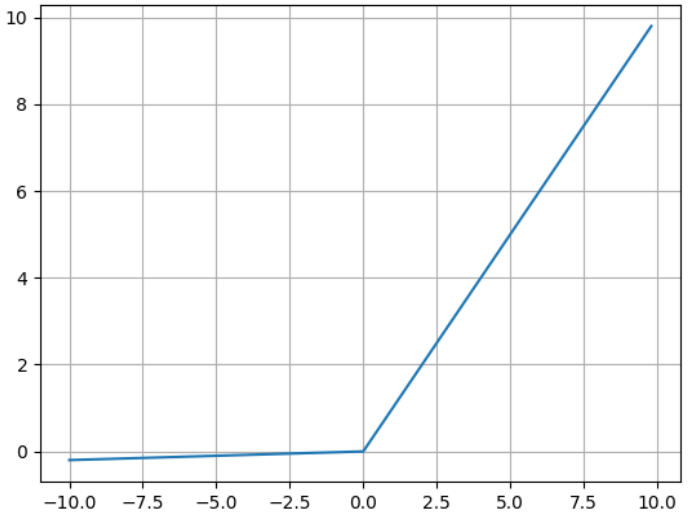
$$f(x) = \frac{1}{1 + e^{-x}}$$

ReLU



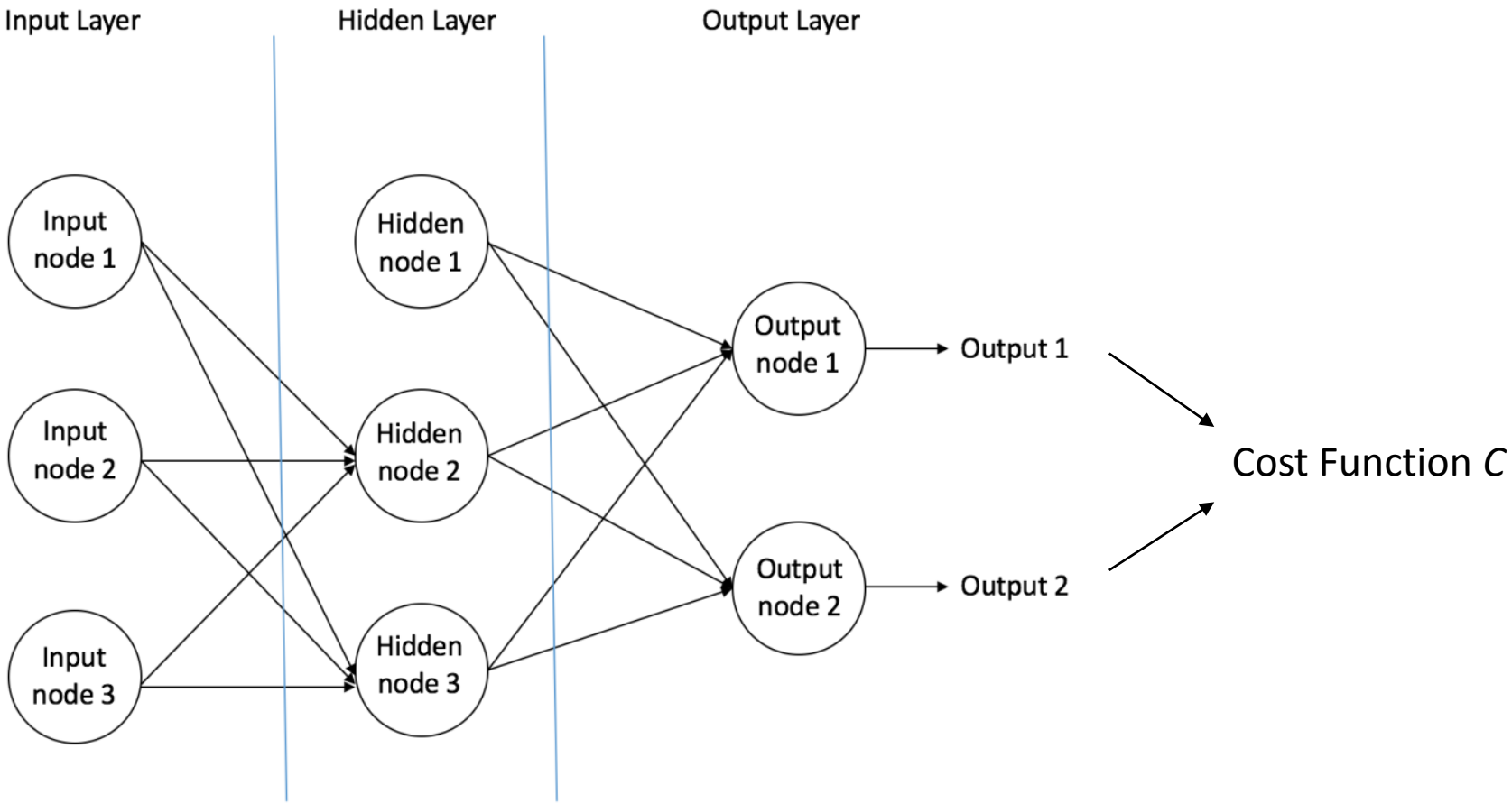
$$f(x) = \max(0, x)$$

Leaky ReLU



$$f(x) = \max(\alpha x, x)$$

Feedforward Neural Network

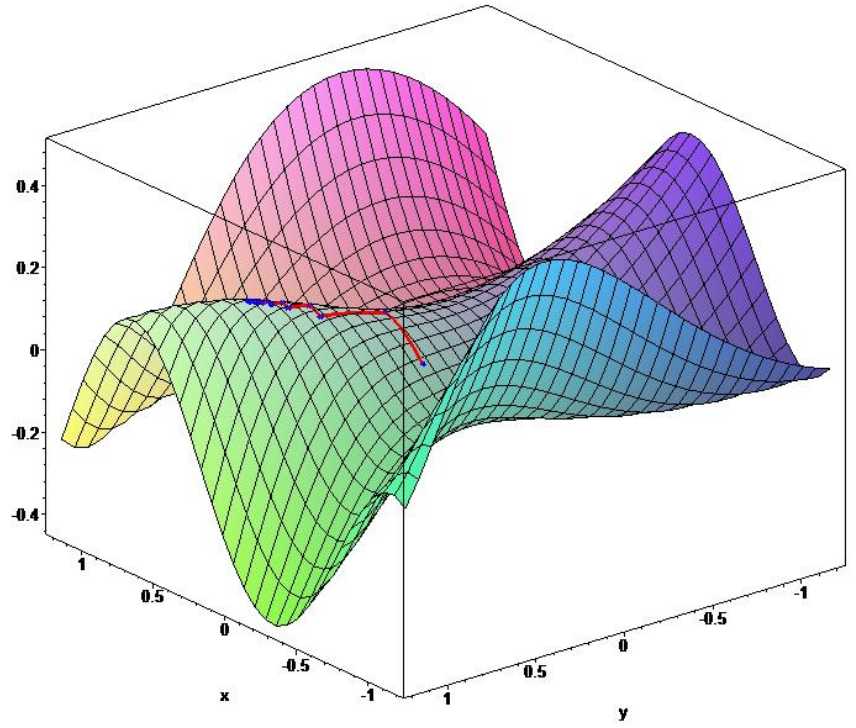


Backpropagation

- Used in conjunction with gradient descent

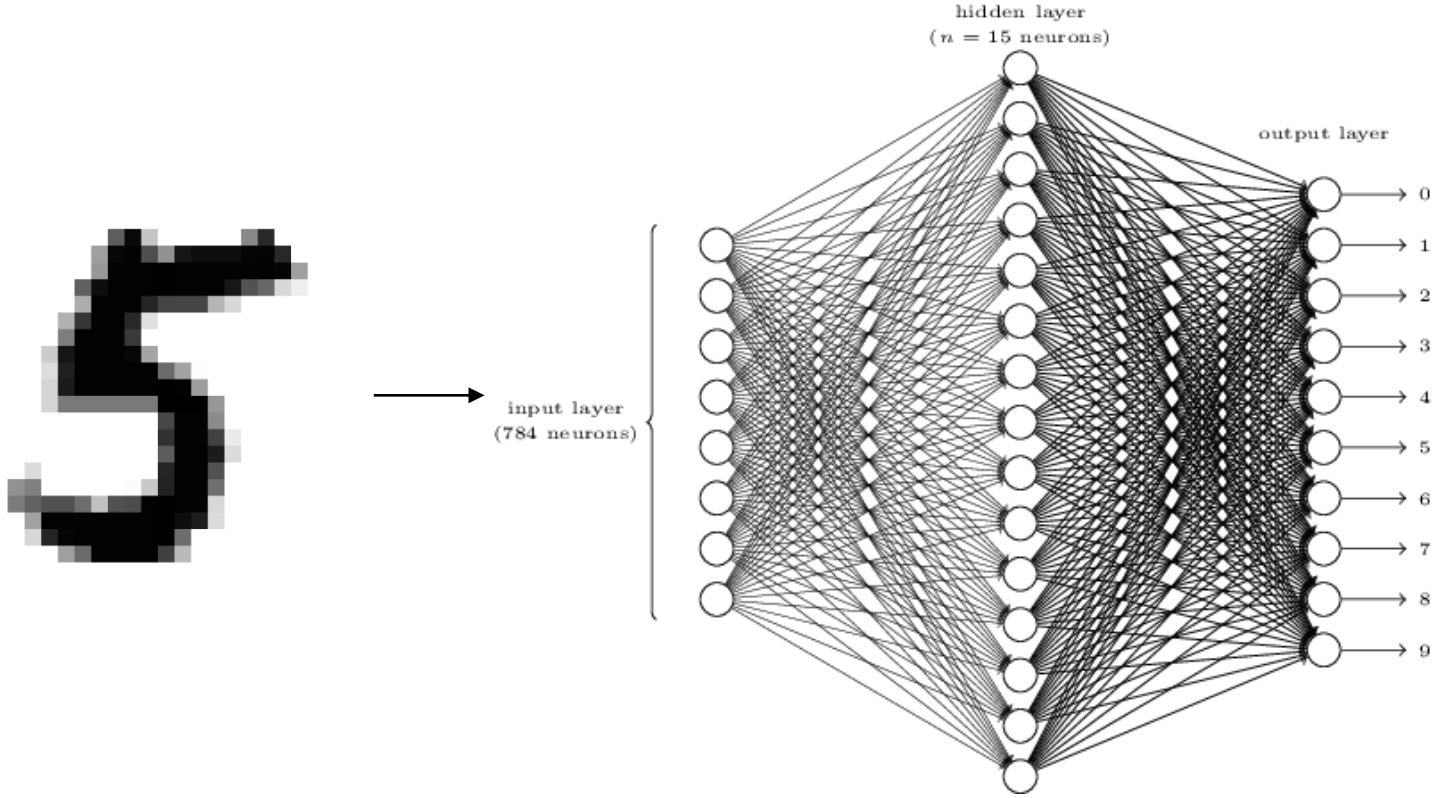
$$z^L = w^L a^{L-1} + b^L$$
$$a^L = \sigma(z^L)$$
$$\mathrm{C} = (a^L - y)^2$$

$$\frac{\partial C}{\partial w^L} = \frac{\partial z^L}{\partial w^L} * \frac{\partial a^L}{\partial z^L} * \frac{\partial C}{\partial a^L}$$



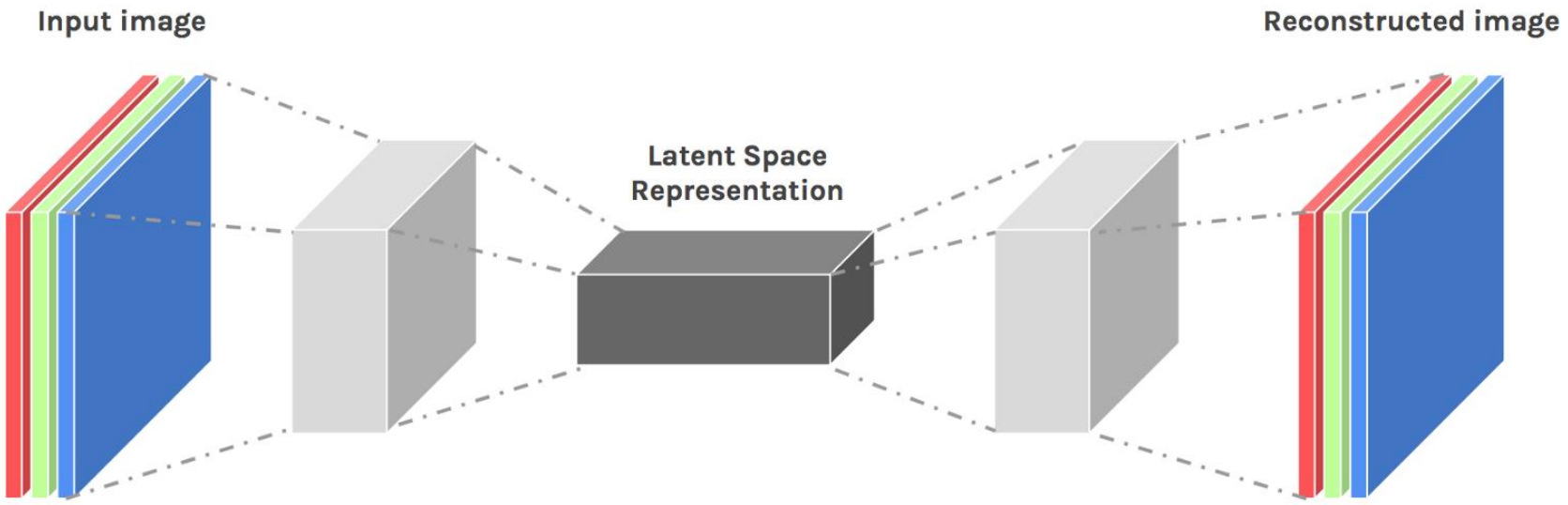
Discriminative Models

- Learn a conditional probability distribution P(y|x)



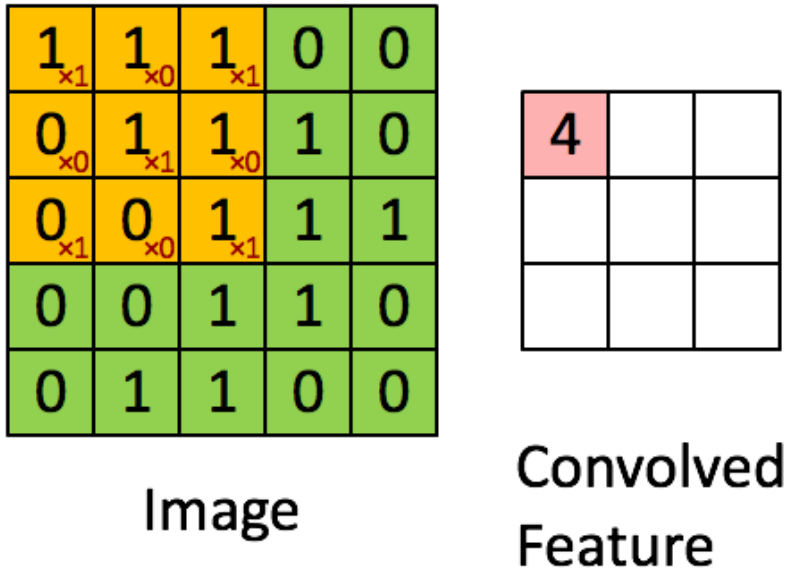
Generative Models

- A form of unsupervised learning
  - Autoencoders
  - Generative Adversarial Networks



Convolution

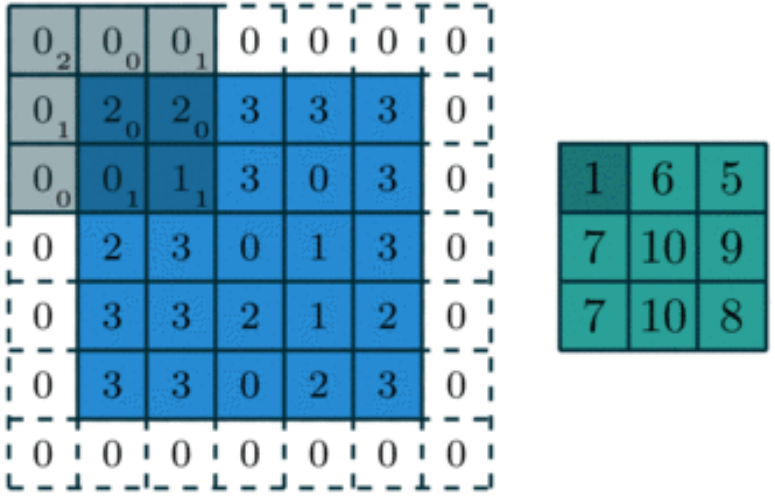
- Primary use is to extract features from an image.



Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	

Convolution Parameters

- Kernel Size
- Stride
- Depth
- Padding

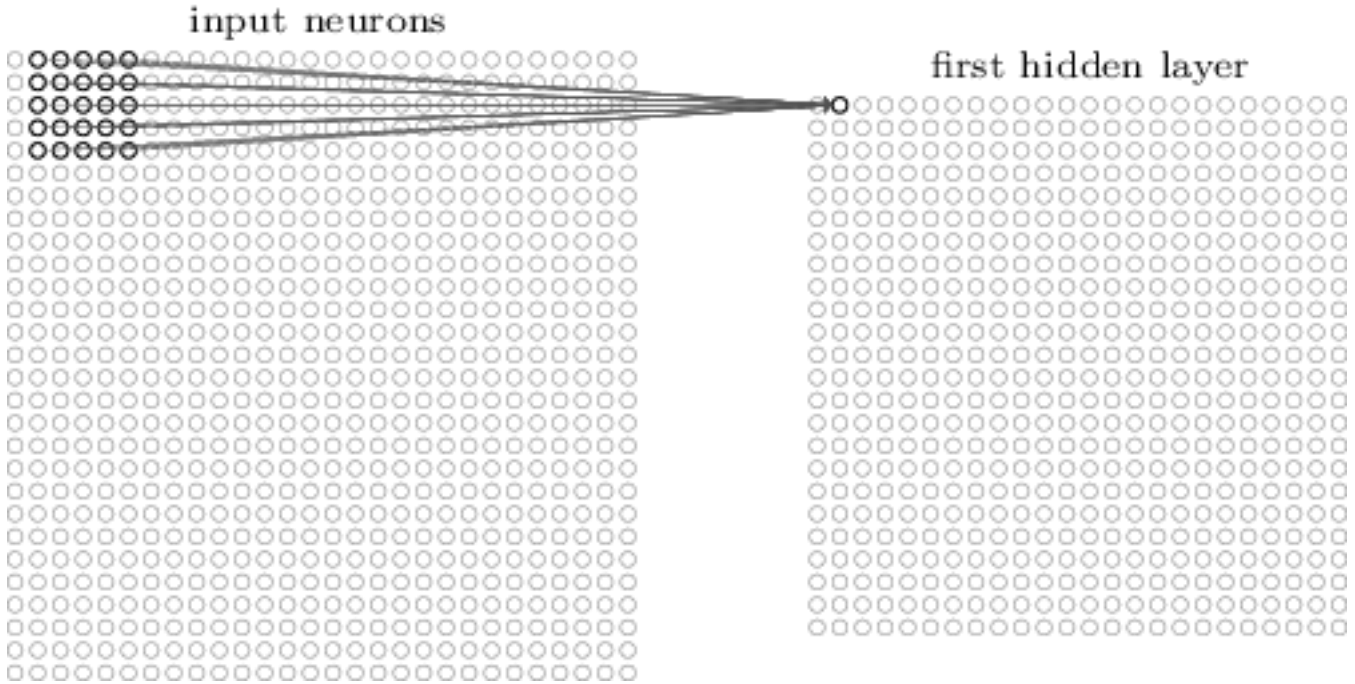


Convolutional Neural Networks (CNNs)

- Combine three main ideas
  - Local Receptive Fields
  - Shared Weights
  - Spatial sub-sampling

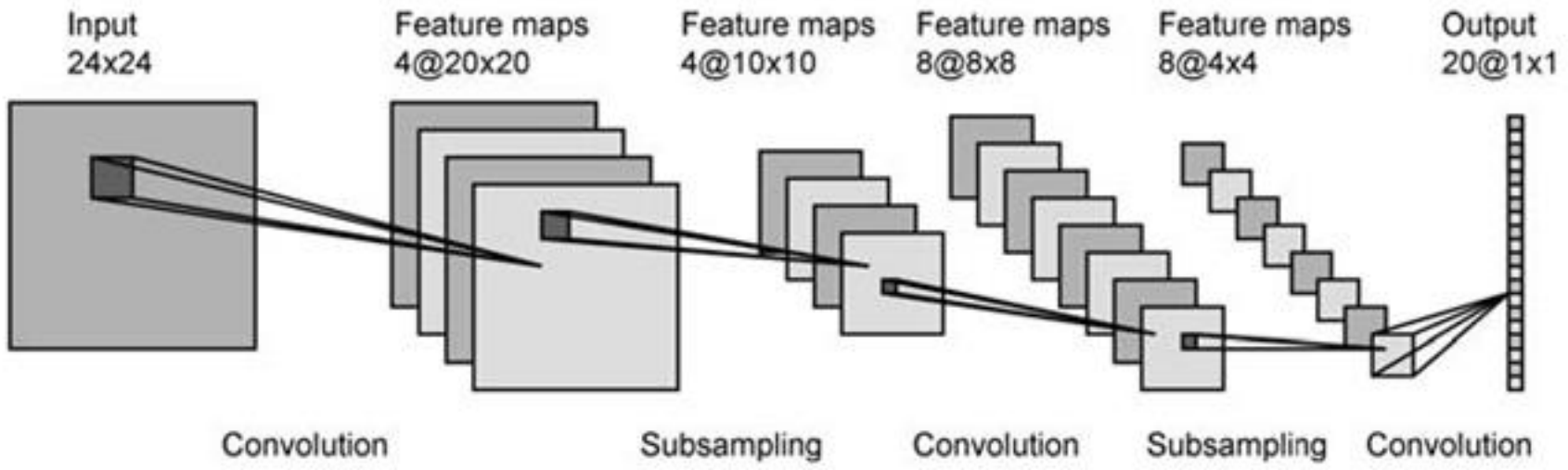
Local Receptive Fields

- Allow neurons to extract local visual features, which are used in subsequent layers to detect higher level features.



LeNet-5

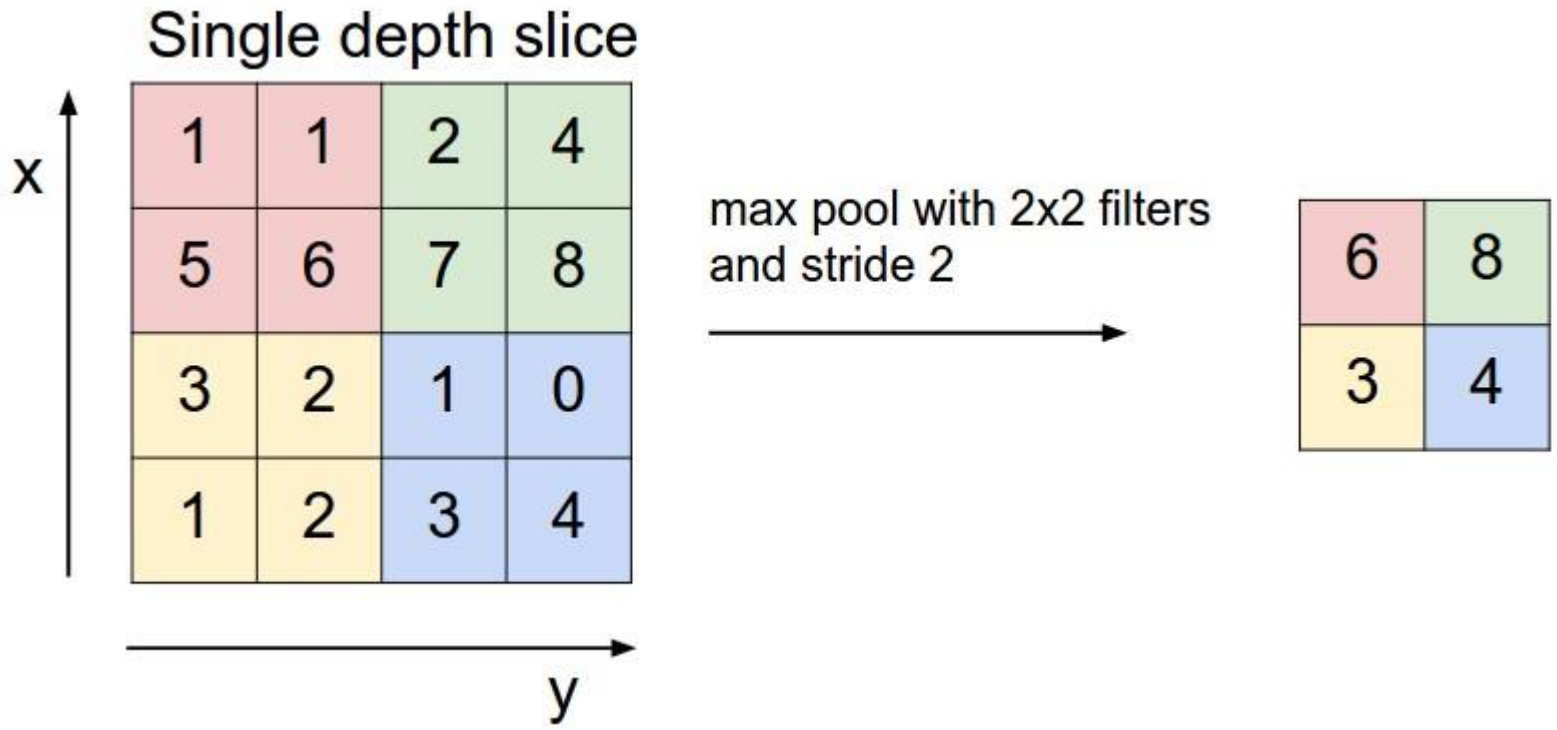
- The set of output units is called a feature map



[1]

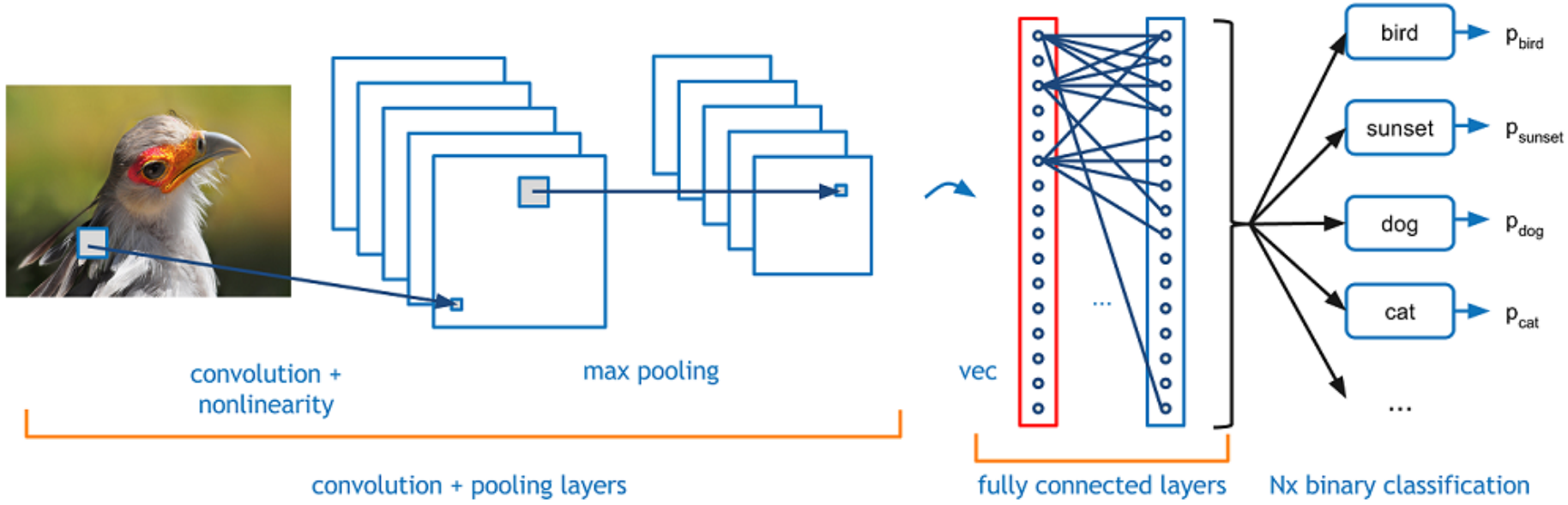
Pooling Layers

- Downsample the input



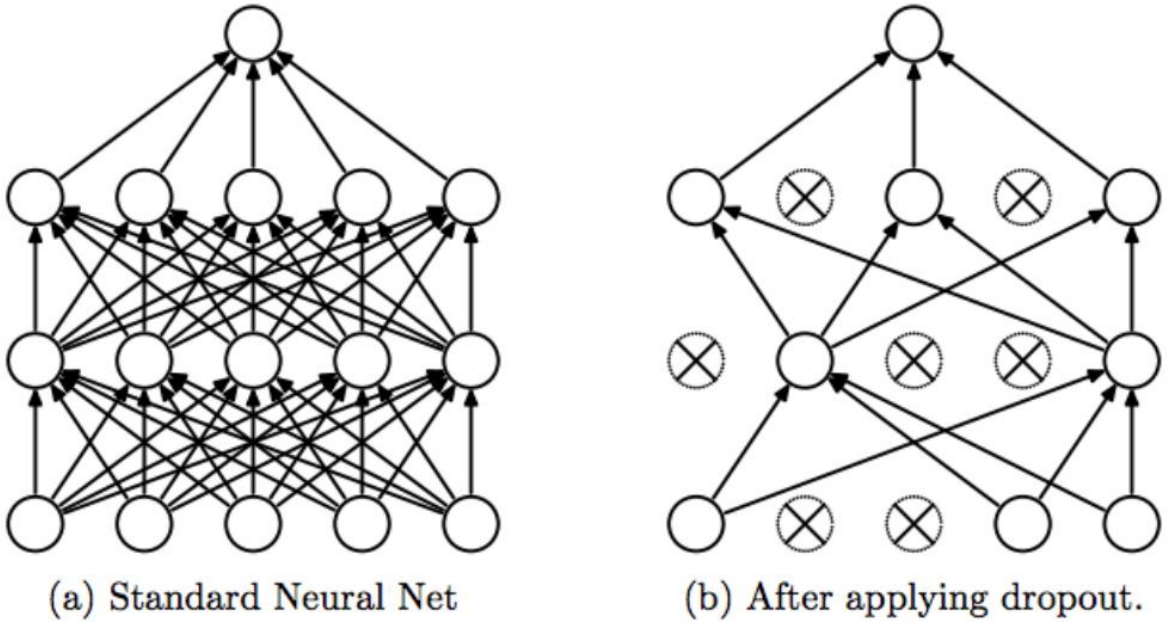
Fully Connected Layers

- Flattens the feature map
- Outputs class labels, numbers, etc.



Dropout

- Prevents overfitting
- Randomly sets activations to zero

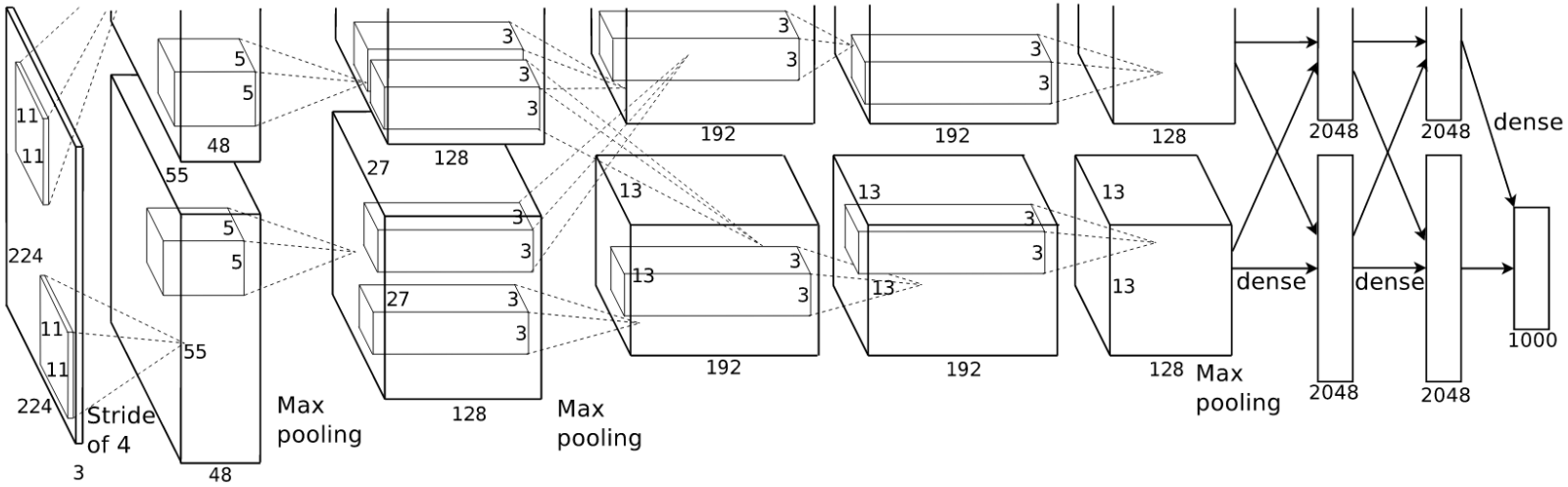


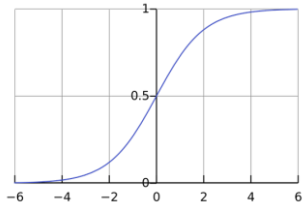
(a) Standard Neural Net

(b) After applying dropout.

Alexnet

- Achieved incredible results on the ImageNet dataset
  - Top-1 error: 37.5%
  - Top-5 error: 17.0%
- Kicked off the deep learning craze





Alexnet Training Details

- Stochastic Gradient Descent
- Batch size of 128
- Momentum
- Weight decay of 0.0005
- Learning rate of 0.01 and reduced throughout training

$$v_{i+1} := 0.9 \cdot v_i - 0.0005 \cdot \epsilon \cdot w_i - \epsilon \cdot \left\langle \frac{\partial L}{\partial w}\Big|_{w_i} \right\rangle_{D_i}$$

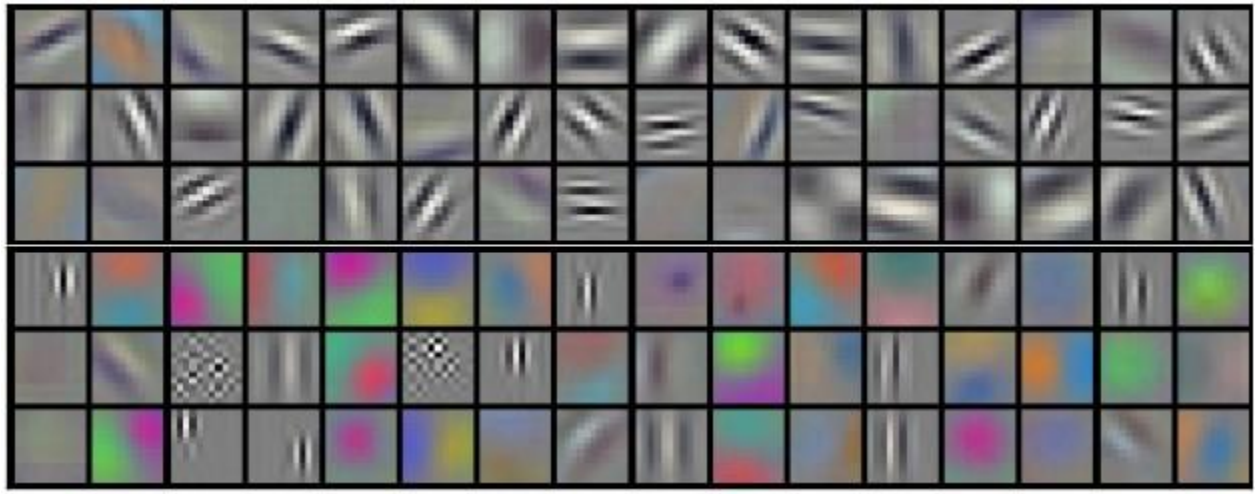
$$w_{i+1} := w_i + v_{i+1}$$

Data Augmentation

- Reduces overfitting
- Provides more data samples
- How?
  - Mirror/flip image
  - Contrast/brightness change

Visualization

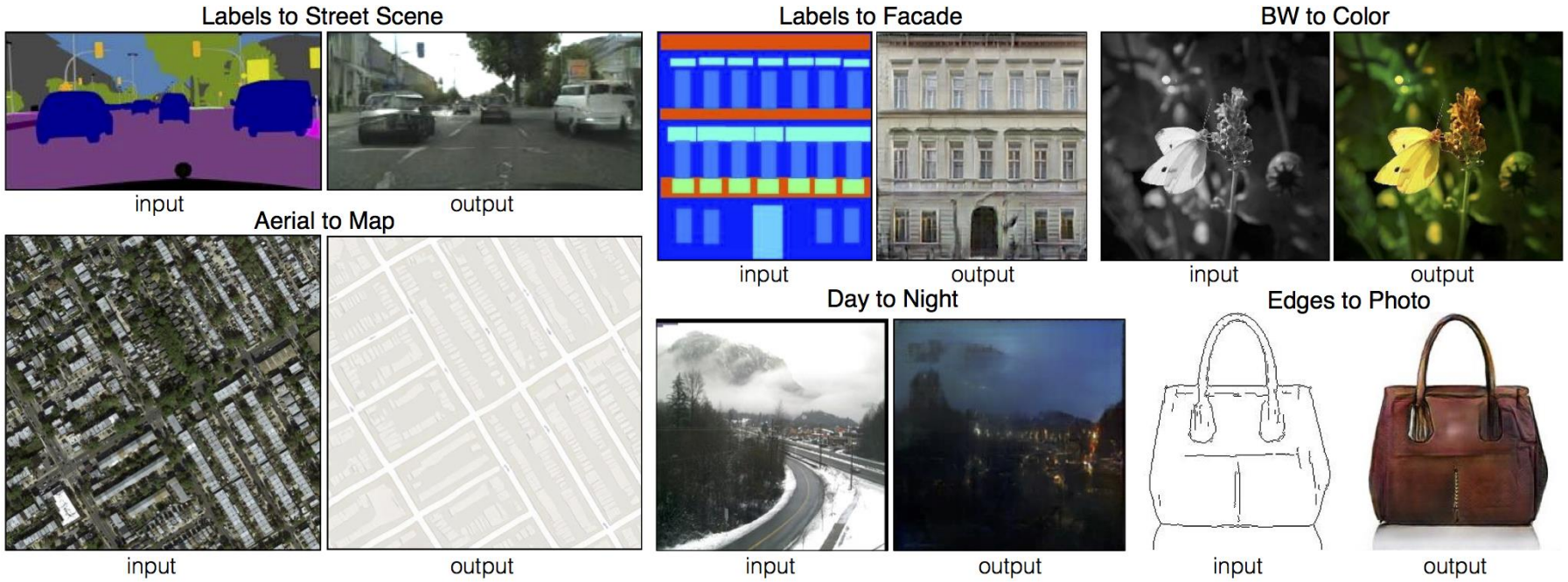
- Filters from the first convolutional layer



Generative Models

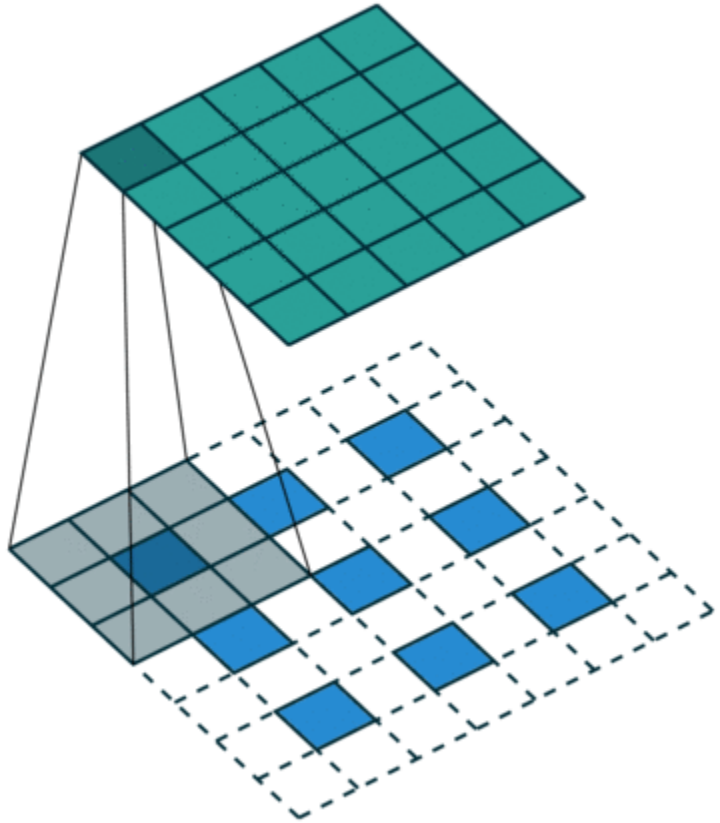
Image to Image Translation

- How can we output an image?



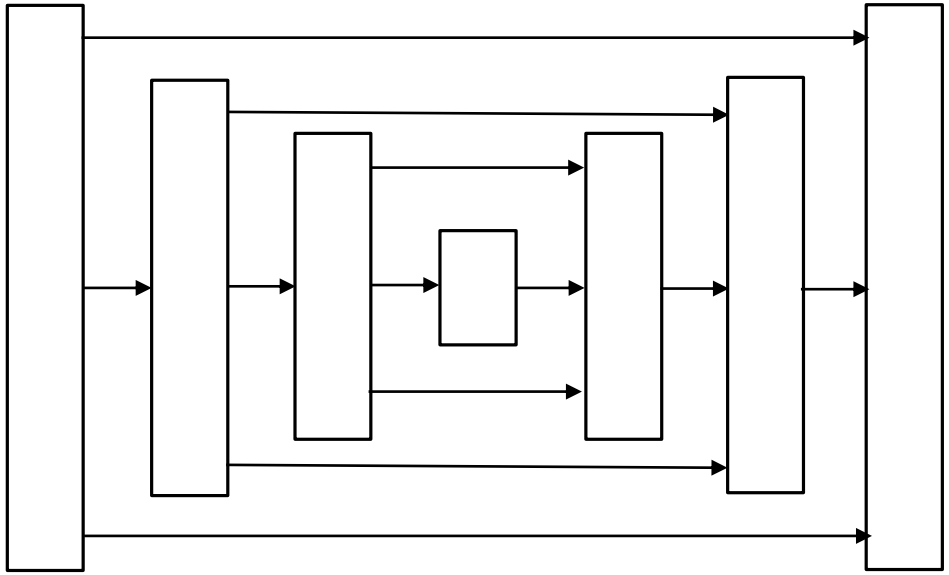
Upsampling

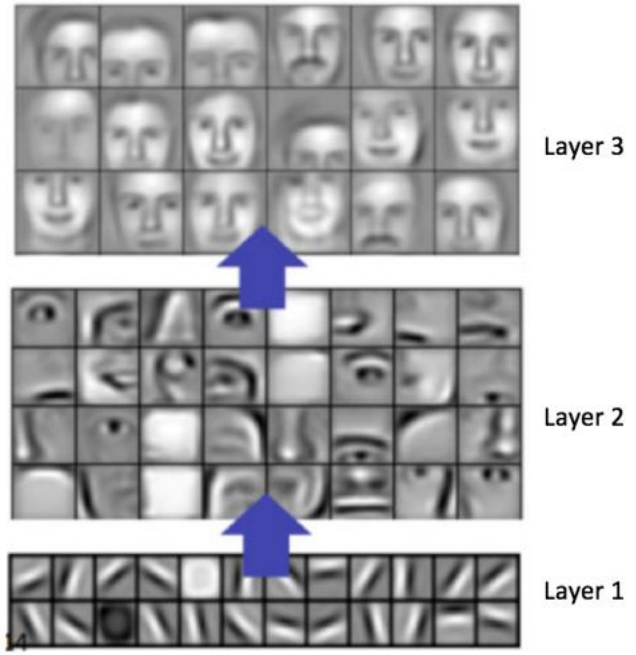
- How do we upsample?
  - Transpose (strided) convolution
  - Upconvolution



U-Net

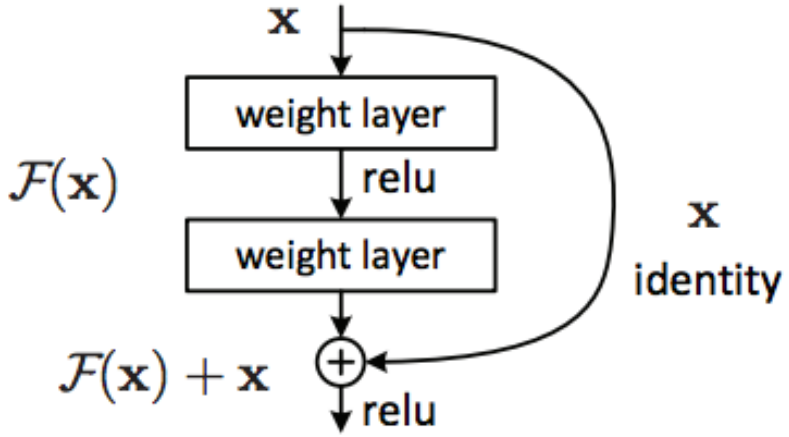
- Skip connections for preserving local structure.



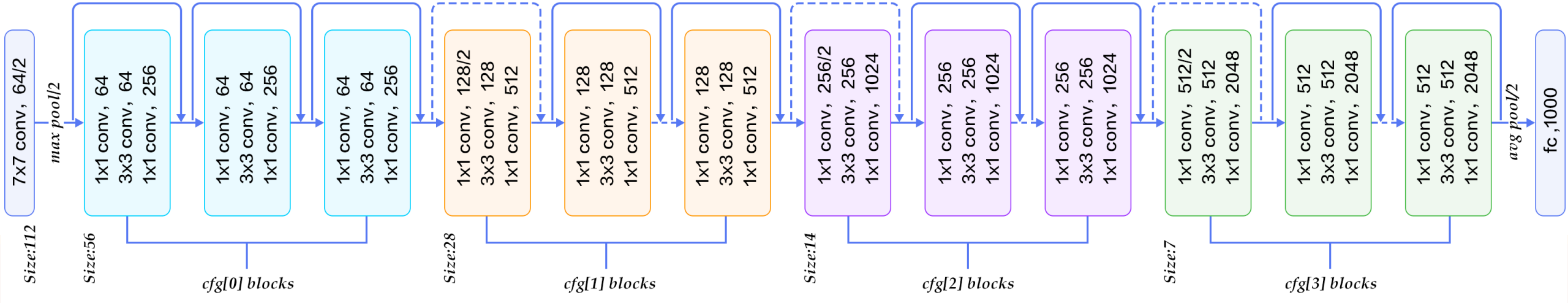


Popular Architectures

- Alexnet
- VGG Net
- Inception
- Resnet

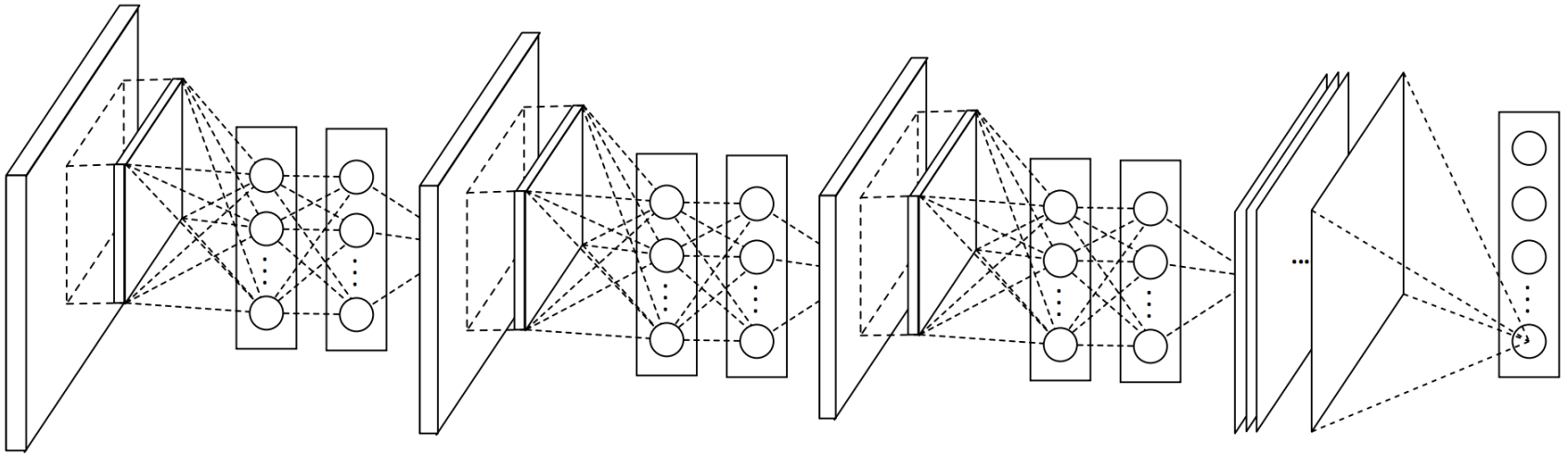


50 layers	*cfg=[3,4,6,3]*
101 layers	*cfg=[3,4,23,8]*
152 layers	*cfg=[3,8,36,3]*



Interesting Architectures

- Network in Network



Deep Learning in Practice

Conclusion

- Feedforward Networks
- Convolutional Neural Networks
- Discriminative Models
  - Alexnet
- Generative Models
- Popular Architectures

Thank You

References

[1]: http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf

[2]: https://arxiv.org/pdf/1506.01497v3.pdf

[3]: https://arxiv.org/pdf/1703.10593.pdf

[5]: https://www.cs.toronto.edu/~vmnih/docs/dqn.pdf

[6]: http://cs231n.github.io/convolutional-networks/

[7]: https://github.com/karpathy/neuraltalk2

[8]: Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009.

[9]: Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." arXiv preprint arXiv:1611.07004 (2016).

[10]: Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Cham, 2015.

[11]: Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012. APA

References

[12]: Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "A neural algorithm of artistic style." arXiv preprint arXiv:1508.06576 (2015). APA

[13]: He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. APA

[14]: Lin, Min, Qiang Chen, and Shuicheng Yan. "Network in network." arXiv preprint arXiv:1312.4400 (2013).

[15]: Zhang, Richard, Phillip Isola, and Alexei A. Efros. "Colorful image colorization." European Conference on Computer Vision. Springer International Publishing, 2016.

[16]: Zhu, Jun-Yan, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." arXiv preprint arXiv:1703.10593 (2017).

[17]: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

[18]: https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/

[19]: https://adeshpande3.github.io/adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

[20]: https://pgaleone.eu/deep-learning/regularization/2017/01/10/anaysis-of-dropout/

[21]: https://devblogs.nvidia.com/parallelforall/deep-learning-nutshell-core-concepts/

[22]: http://neuralnetworksanddeeplearning.com/chap1.html

[23]: https://hackernoon.com/autoencoders-deep-learning-bits-1-11731e200694

[24]: http://deeplearning.net/software/theano/tutorial/conv_arithmetic.html