Convolutional Neural Networks: architectures, interpretation

LSSDS 2024

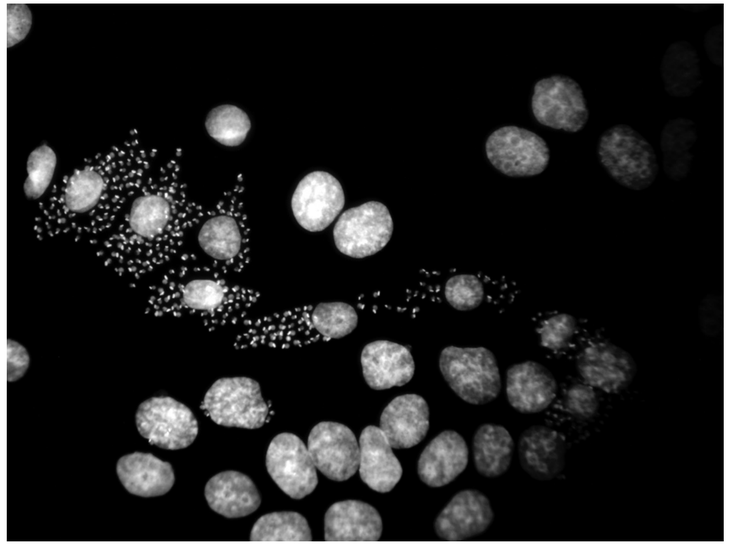
Mauricio Cerda, Eng. PhD

Programa de Biología Integrativa
I.C.B.M., Facultad de Medicina, Universidad de Chile

Outline

- How to select the right image features to classify/find clusters?
- Convolutional Neural Networks
- CNN architectures (some)
- CNN visualization

How to select the right features?

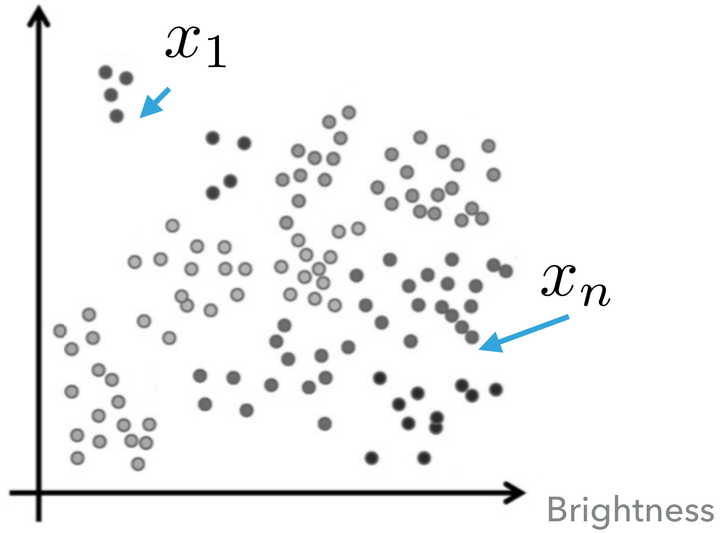


Images

$$[x_1, x_2, \ldots x_d]^T$$

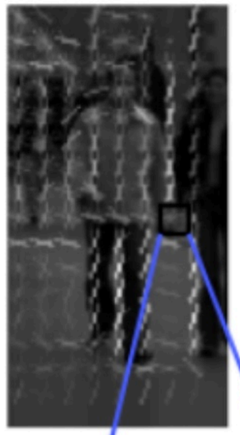
Features

Filtered image



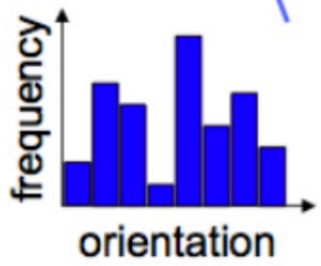
Feature space

What about image filters?

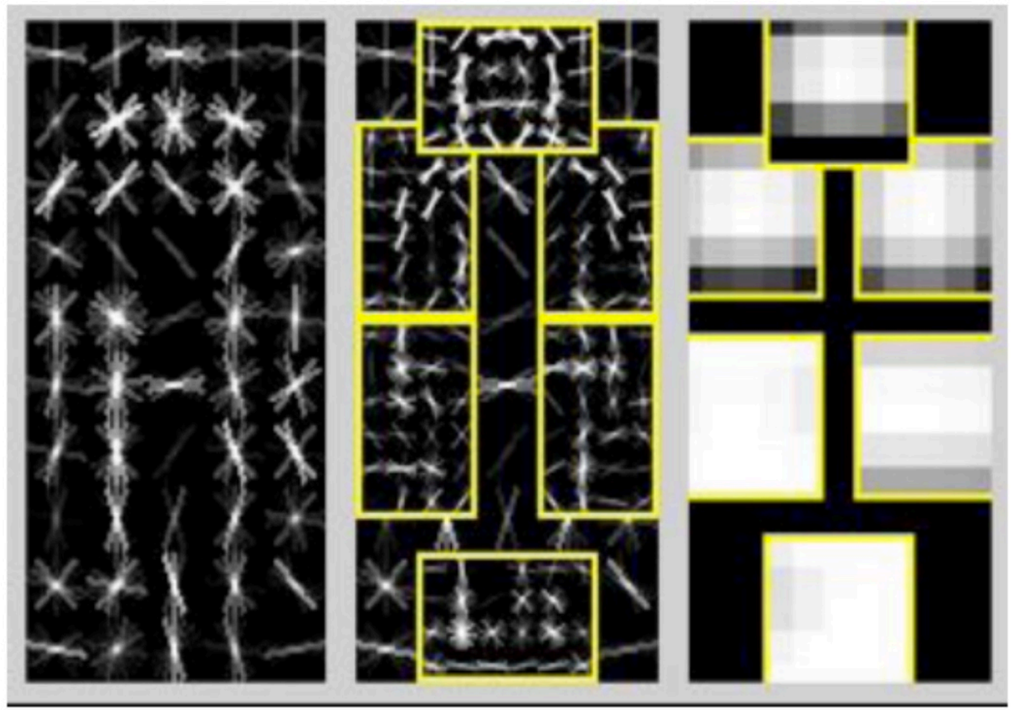


$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

$$mag(\nabla f) = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial x}\right)^2}$$



Histogram of Gradients (HoG)
Dalal & Triggs, 2005

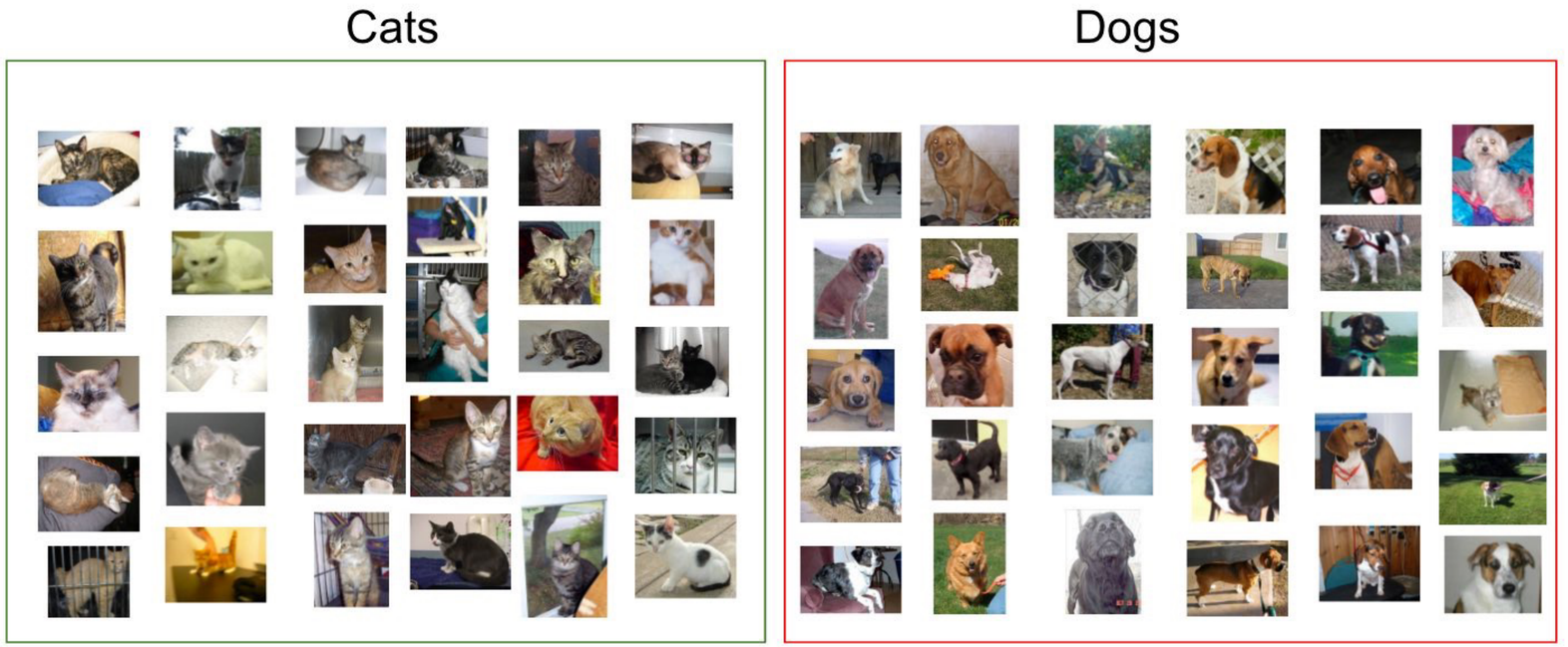


Deformable Part Model
Felzenswalb, McAllester, Ramanan, 2009

What about image filters?

- Image features required **manual** feature selection.
- It was all about features!
- Many unsolved difficult problems.

Deep Learning key ideas



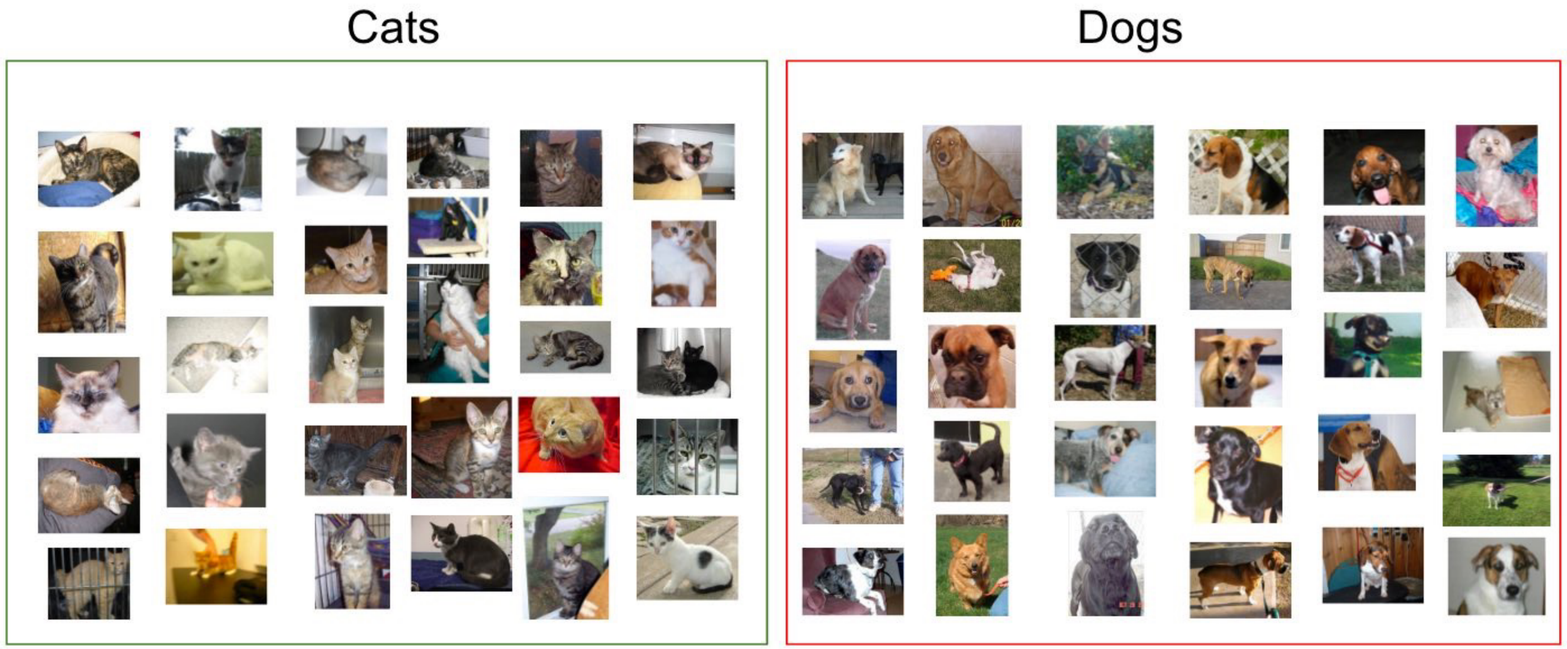
Sample of cats & dogs images from Kaggle Dataset

Traditional Machine Learning Flow

Deep Learning Flow

Slides from Ashish Mahabal

Deep Learning key ideas



Sample of cats & dogs images from Kaggle Dataset

Promise:
Works better

Traditional Machine Learning Flow

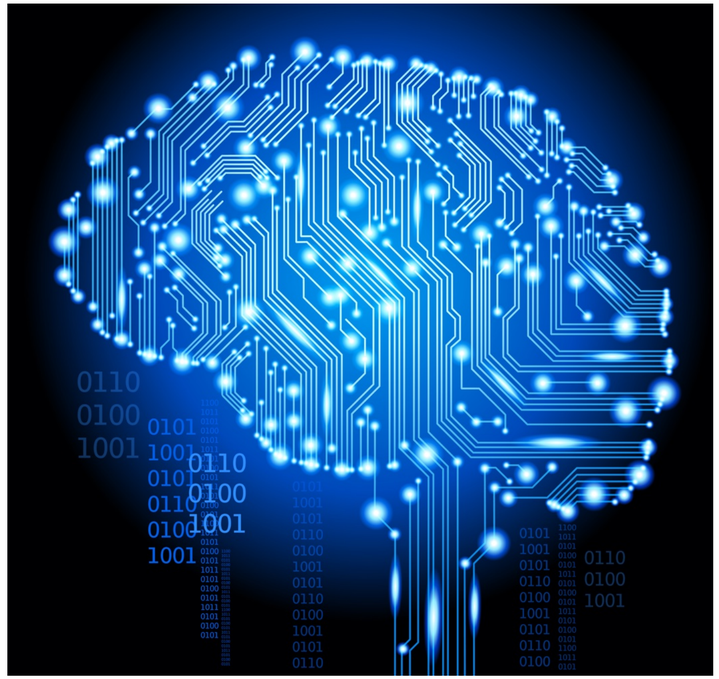
Pitfall:
Blacker box

Deep Learning Flow

Slides from Ashish Mahabal

Biological inspiration

- Idea: IA should mimic how the brain works.
- Because, the brain solves complex problems.
- But… brain it is complex, even 1 neuron is complex, and we don't fully understand how it works.

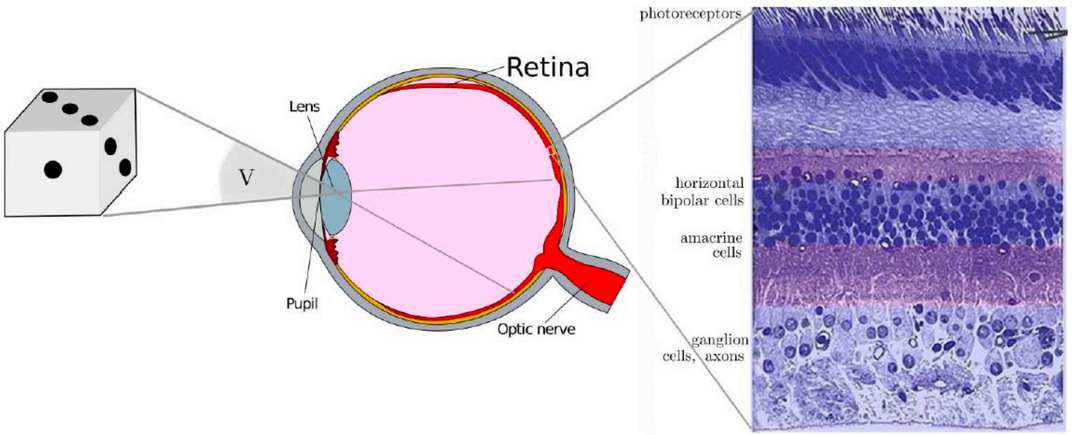


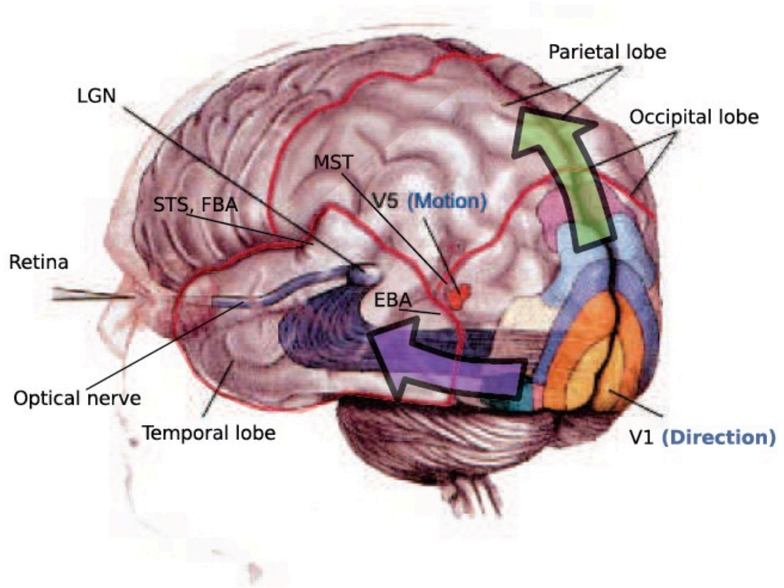
Biological inspiration

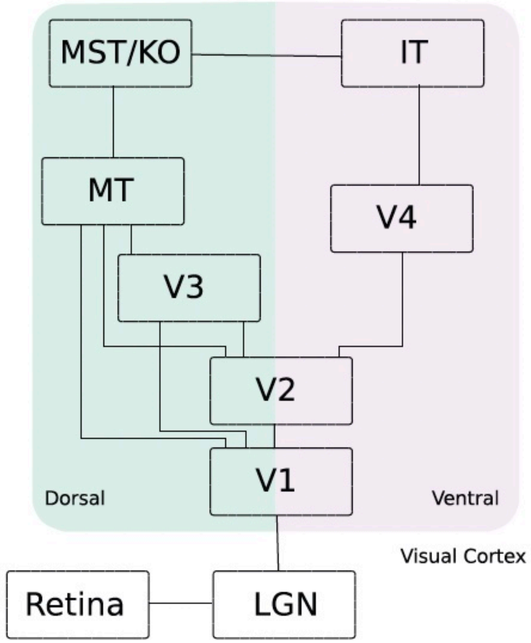
- 10^{10} Neurons!
- Response time > 1 [ms] (slow)
- Facial recognition in 100 [ms]
- In average, each neurons has thousands of connections.
- Hundred of operations per second
- High parallelism
- Distributed information representation
- Common failures (Never replaced)
- Compensate problems with massive parallelism.

Biological inspiration: hierarchical processing

- From the retina to visual cortex, neurons mainly have a bigger “receptive field”.

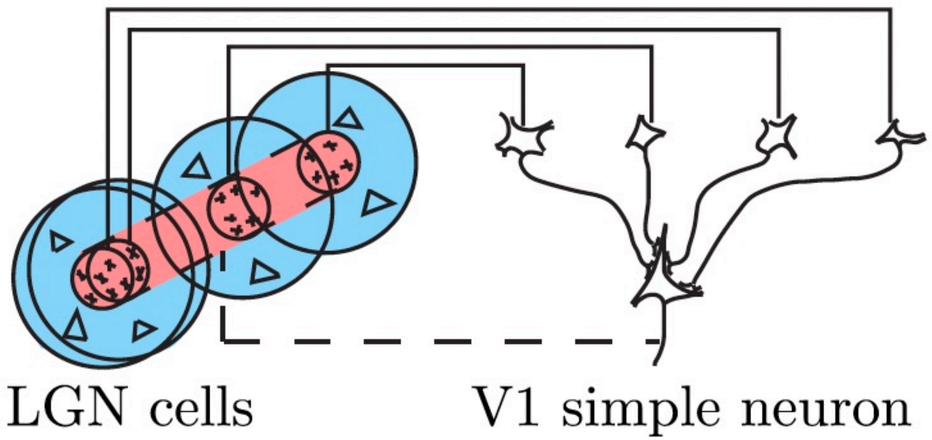


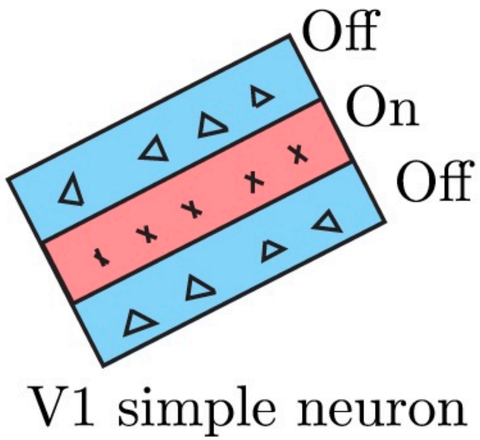




Biological inspiration: hierarchical processing

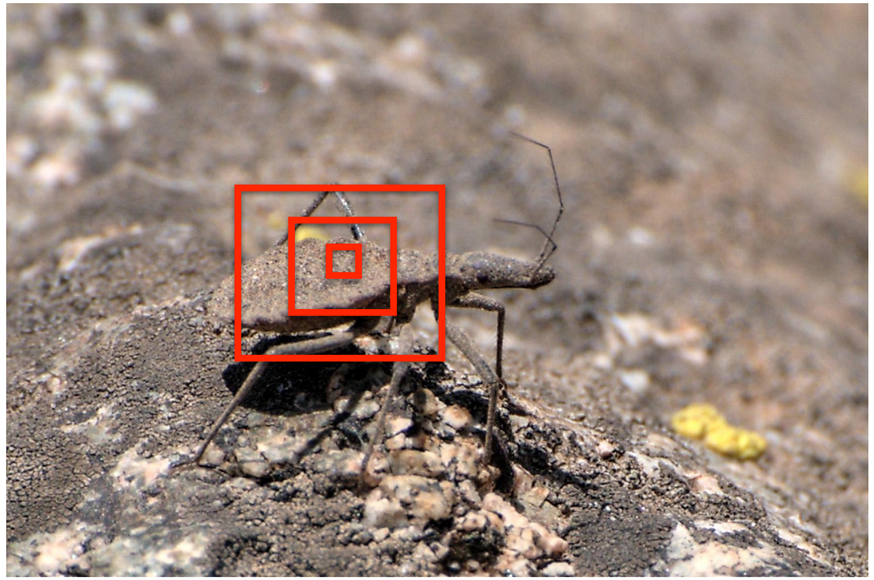
- At Retina level neurons are sensitive to “contrast” in a specific part of the visual field.
- A precise set of contrast sensitive neurons, can detect a line (V1).
- A precise set of line sensitive neurons can detect a grating pattern (V1)

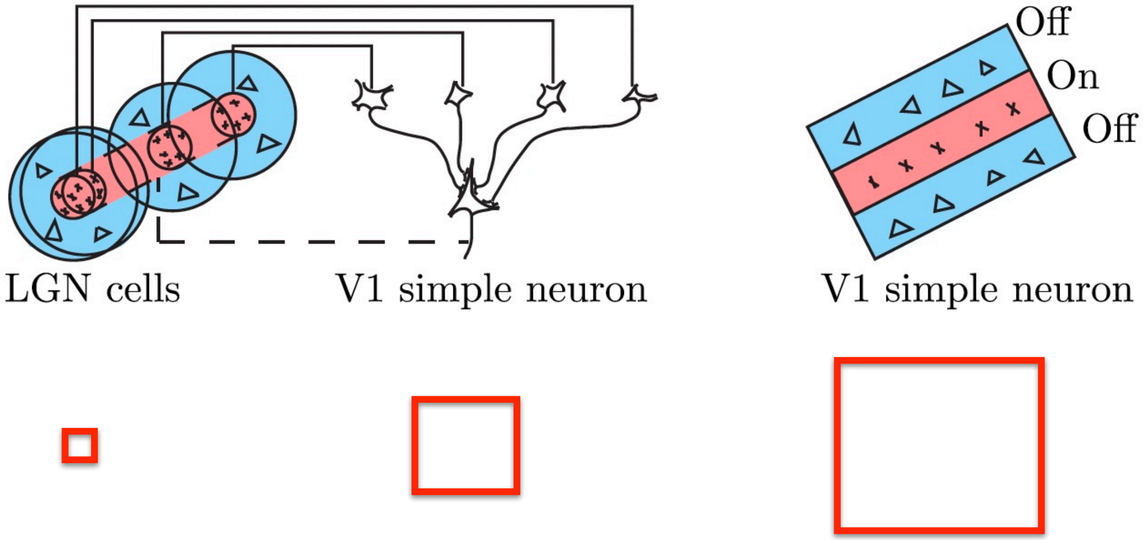




Biological inspiration: hierarchical processing

- The receptive field increases as processing gets more complex.

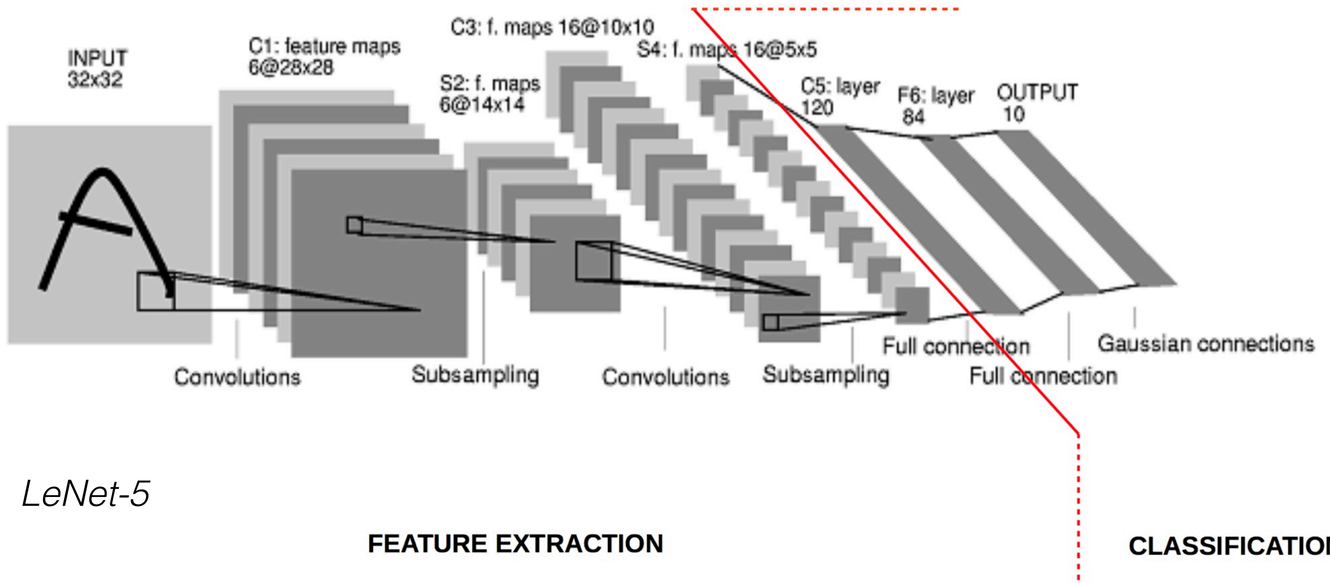




(BUT, 80% of connection are feedback in areas like LGN!)

Convolutional Neural Networks (CNN)

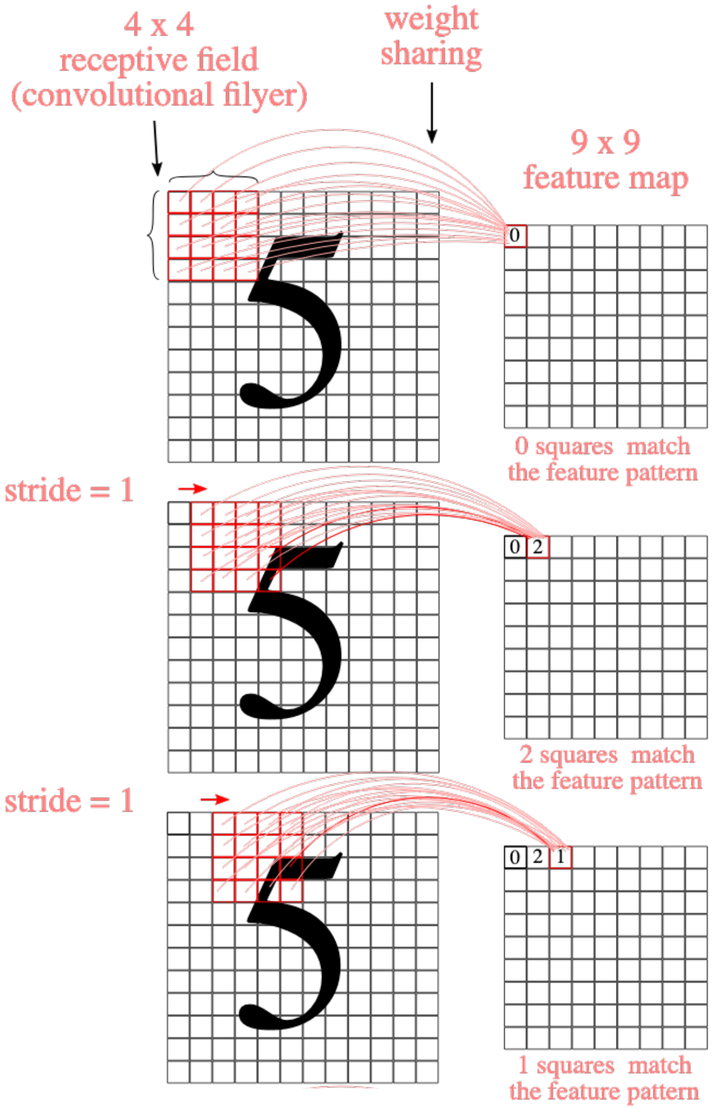
- Set of layers: convolutions, subsampling, fully connected.
- Given an architecture, learn **convolutions** (filters)
- Early version proposed in 90'.
- Take ideas from biological models (neocognitron)



LeNet-5

Y. LeCun, L. Bottou, Y. Bengio and P. Haffner: Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, 86(11):2278-2324, November 1998

CNN: convolutional layer



feature map pattern: *right vertical line detector*

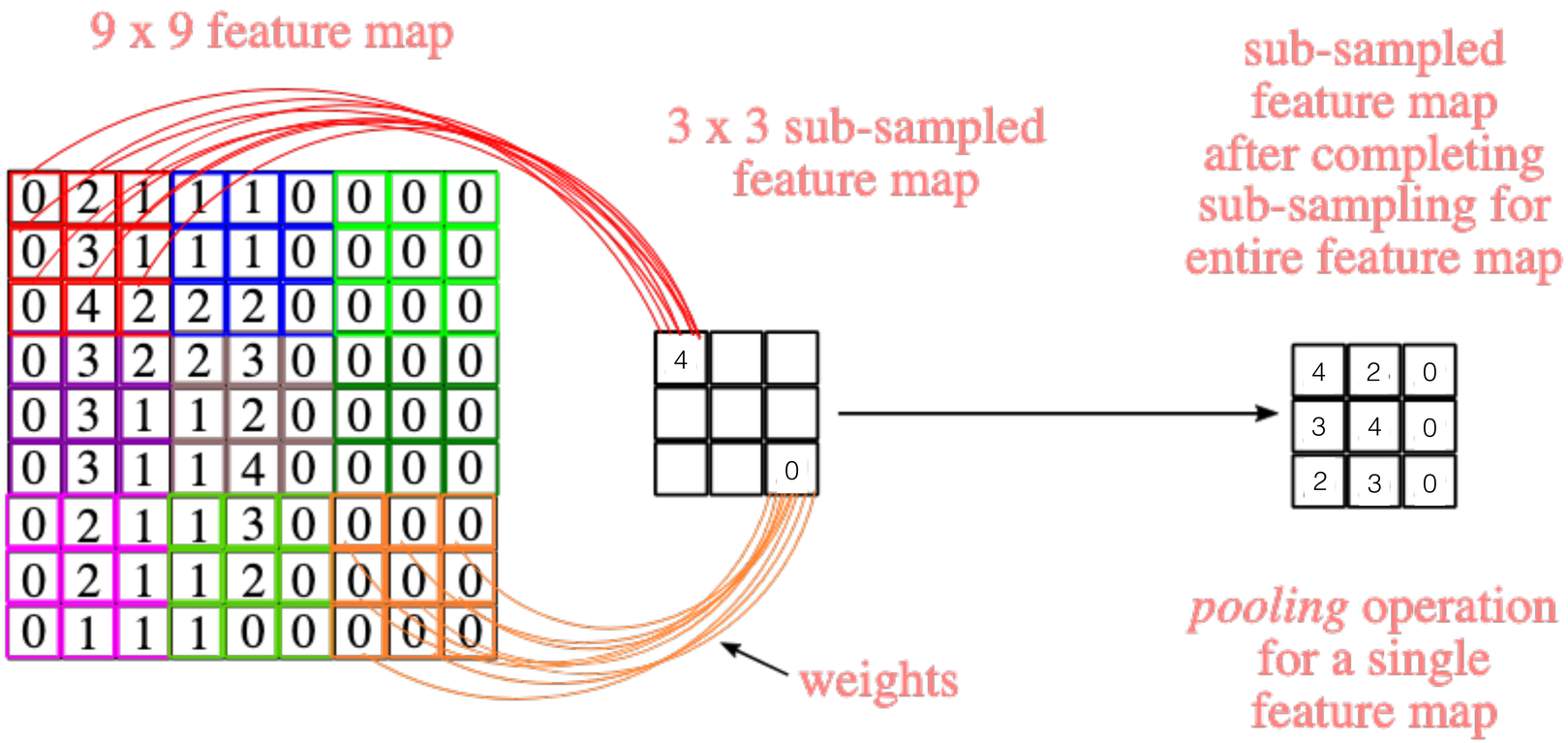
values after sliding over the entire image

0	2	1	1	1	0	0	0	0
0	3	1	1	1	0	0	0	0
0	4	2	2	2	0	0	0	0
0	3	2	2	3	0	0	0	0
0	3	1	1	2	0	0	0	0
0	3	1	1	4	0	0	0	0
0	2	1	1	3	0	0	0	0
0	2	1	1	2	0	0	0	0
0	1	1	1	0	0	0	0	0

convolution operation for a single feature map

- Learn relevant features.
- Weights must be tuned.
- Increase dimension.

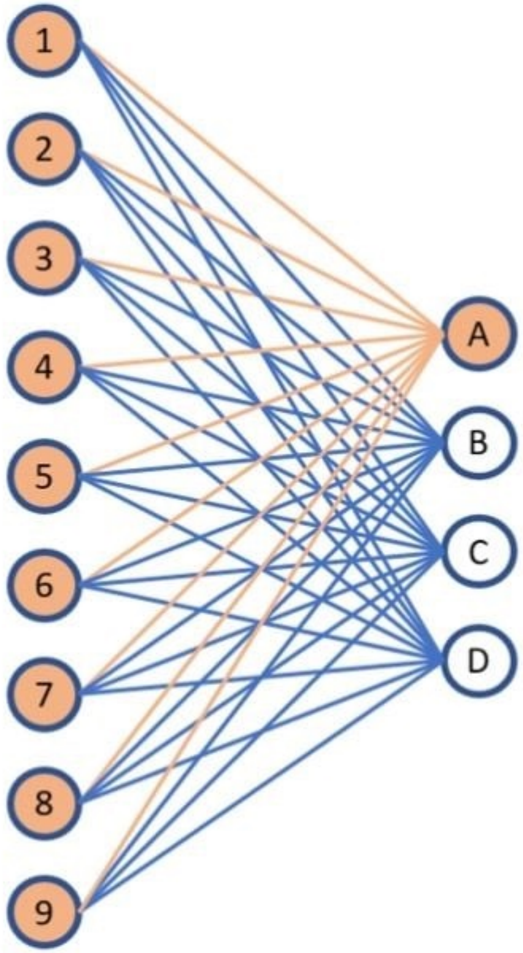
CNN: pooling



MaxPooling

- No weights
- Reduce dimension.
- +Spatial invariance.

CNN: fully connected (FC)

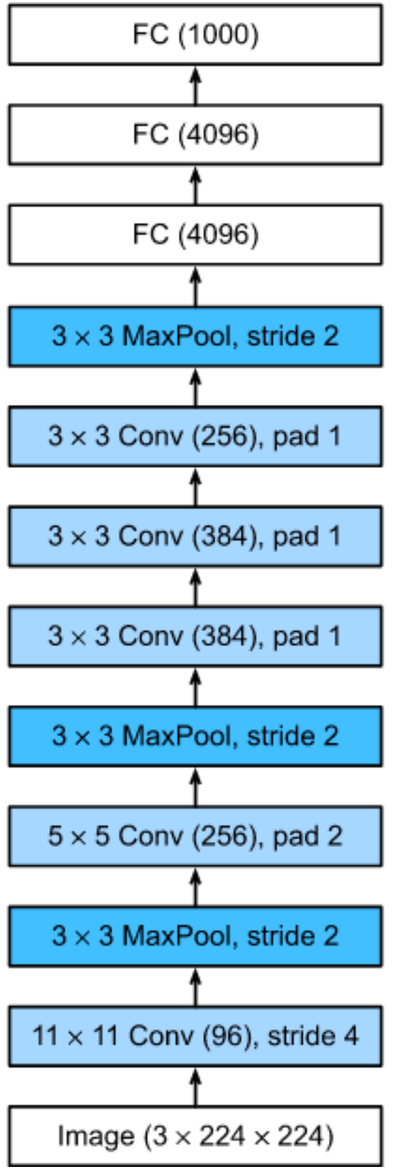


- Learn flexible features.
- +Weights to be tuned.

https://builtin.com/machine-learning/fully-connected-layer

CNN: AlexNet main features

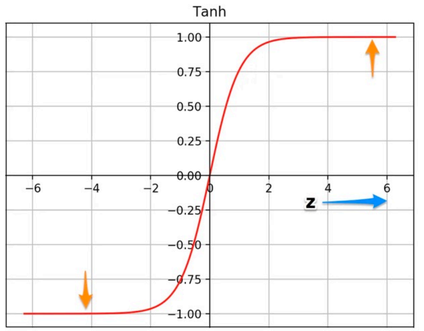
- 8 layers (5 convolutional, 3 FC).
- conv layer + **RELU**.
- **Dropout** in first 2 layers.
- Max-pooling in layers 1, 2, 5.
- GPU implementation.
- Large Scale Visual Recognition Challenge 2012 winner.

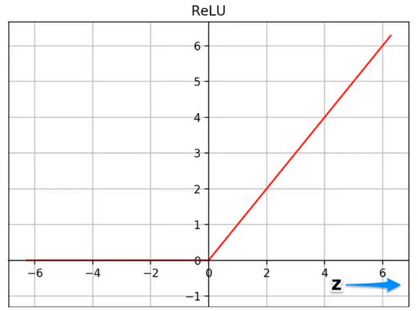


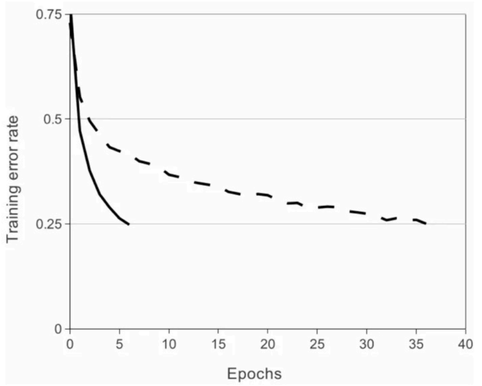
Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks." Advances in neural information processing systems. 2012.

CNN: AlexNet activation function

- Until 2012 tanh / sigmoid were most common activation function.
- Tan / sigmoid saturate, making gradient descend slower.
- Experimentally RELU allowed for faster training.



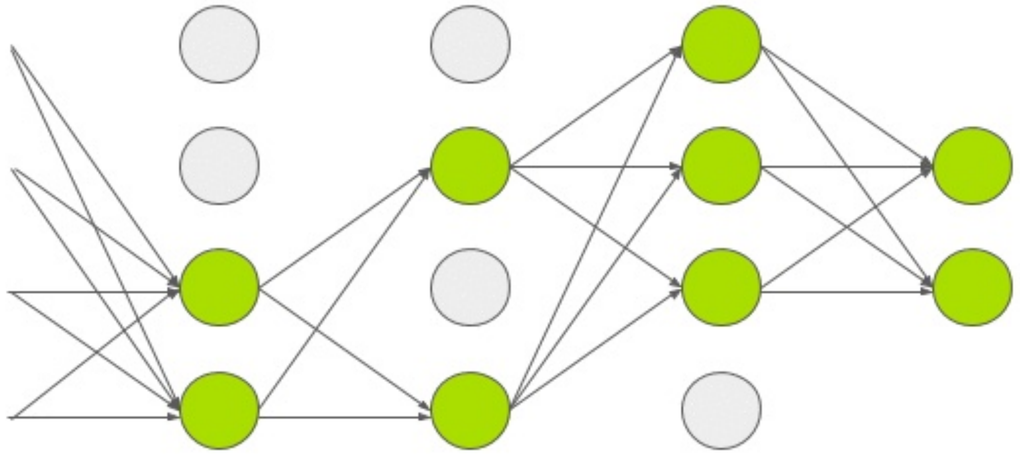




Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks." Advances in neural information processing systems. 2012.

CNN: AlexNet and overfitting

- To reduce overfitting dropout was proposed.
- Each neuron can be disabled during learning with a certain probability.
- (Slower training, requires bigger datasets)



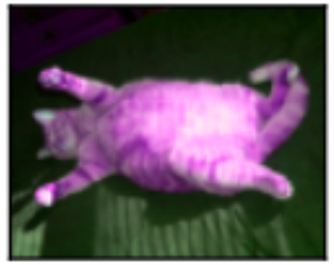
Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks." Advances in neural information processing systems. 2012.

CNN: AlexNet and overfitting

- To generate image variants that the network must be robust.
- Rotations, translations, “noisy image".
- Idea used today!

Flip horizontal

ZoomIn + crop

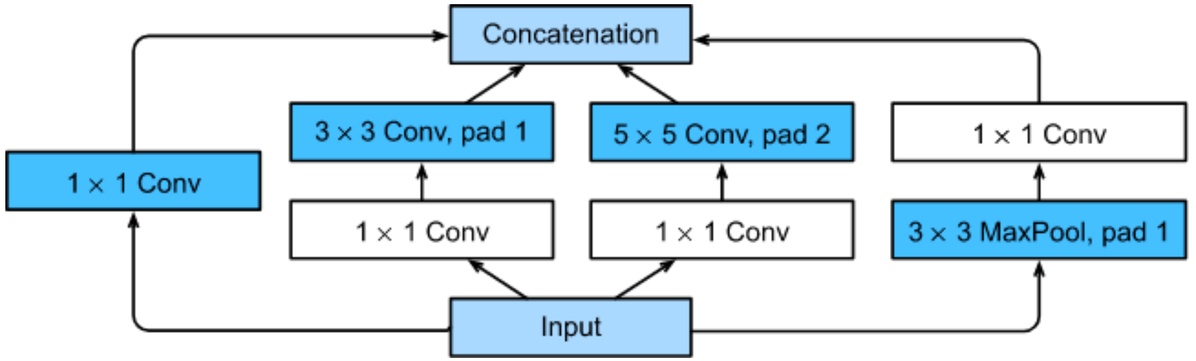


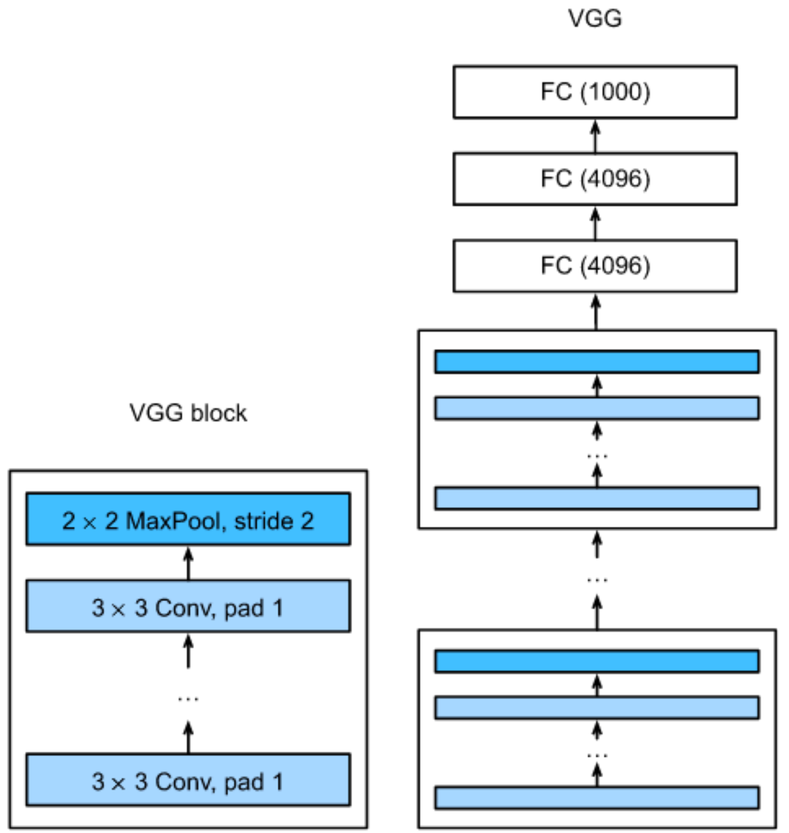
Variación de color

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks." Advances in neural information processing systems. 2012.

CNN: Visual Geometry Group (VGG)

- Cone -> Relu -> pooling quickly lose resolution
- VGG showed that multiple 3x3 conv are equivalente to a 5x5 conv.
- Introduces the “block” idea to configure according to problem/resources.

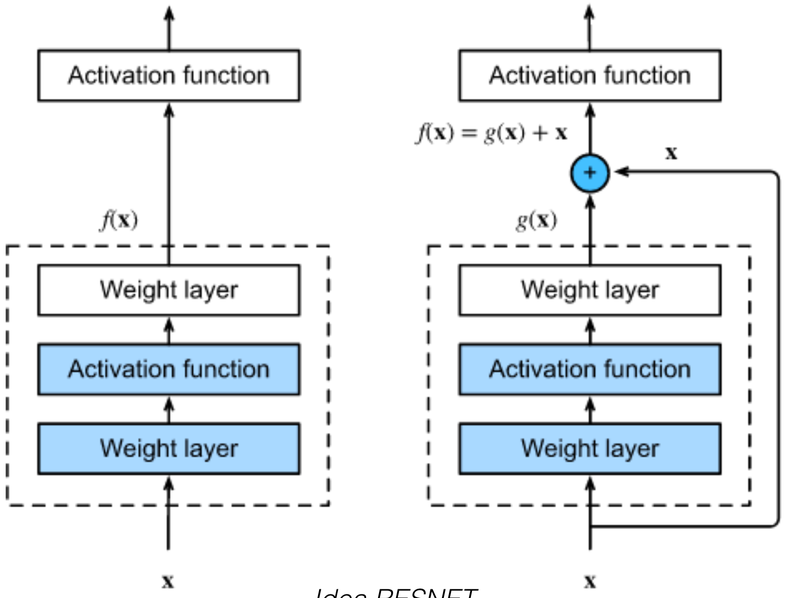




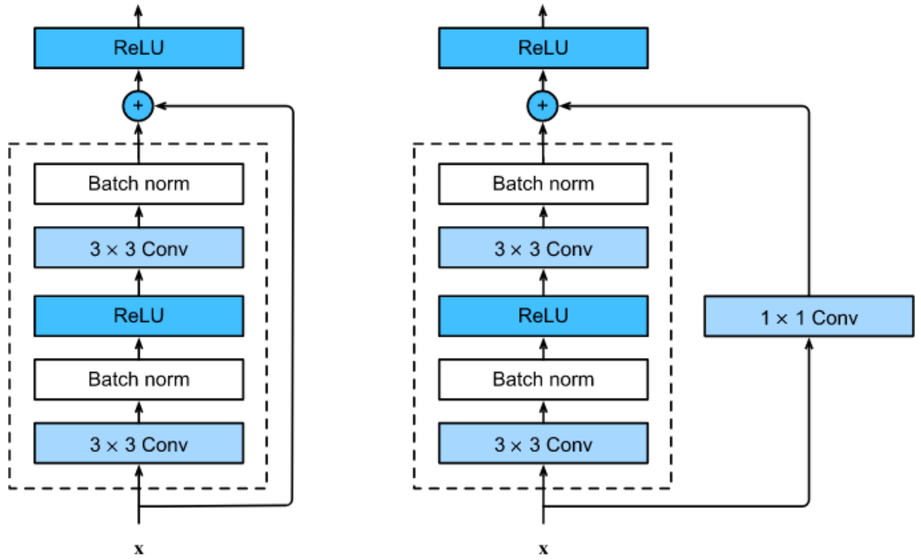
Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

CNN: ResNet

- If we want the network learns *f(x)* we want a module to learn from residual *g(x)*.
- If *f(x)=x, g(x)=0*, we always have the option blocks does nothing (subset).



Idea RESNET

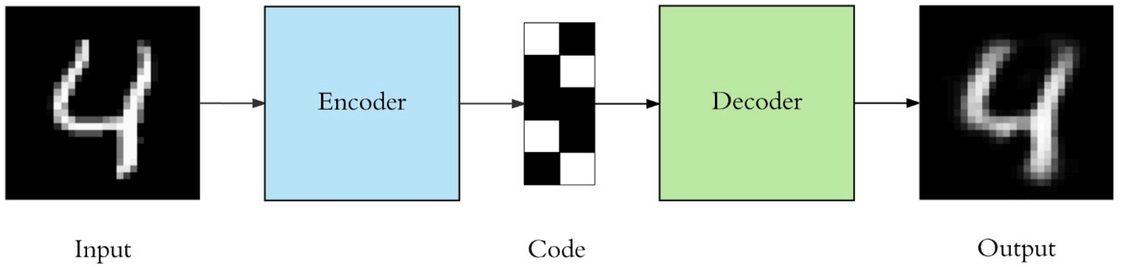


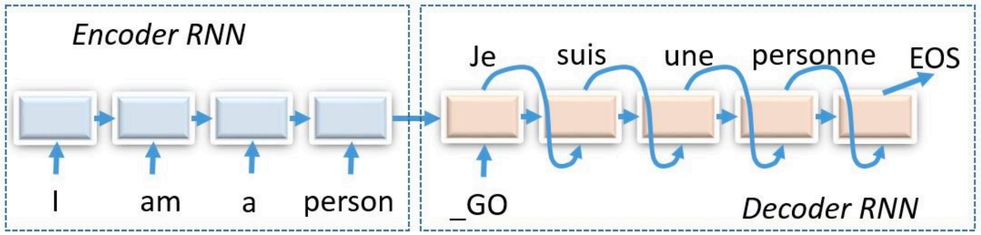
Módulo RESNET

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770–778).

CNN: Encoder/decoder

- Strategy to learn “representations” in lower dimension.
- Allows to work with problem where input/output have the same size (finally segmentation!).

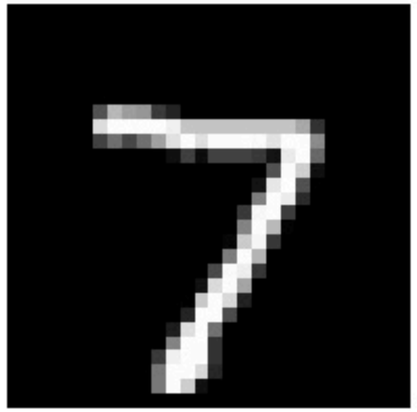


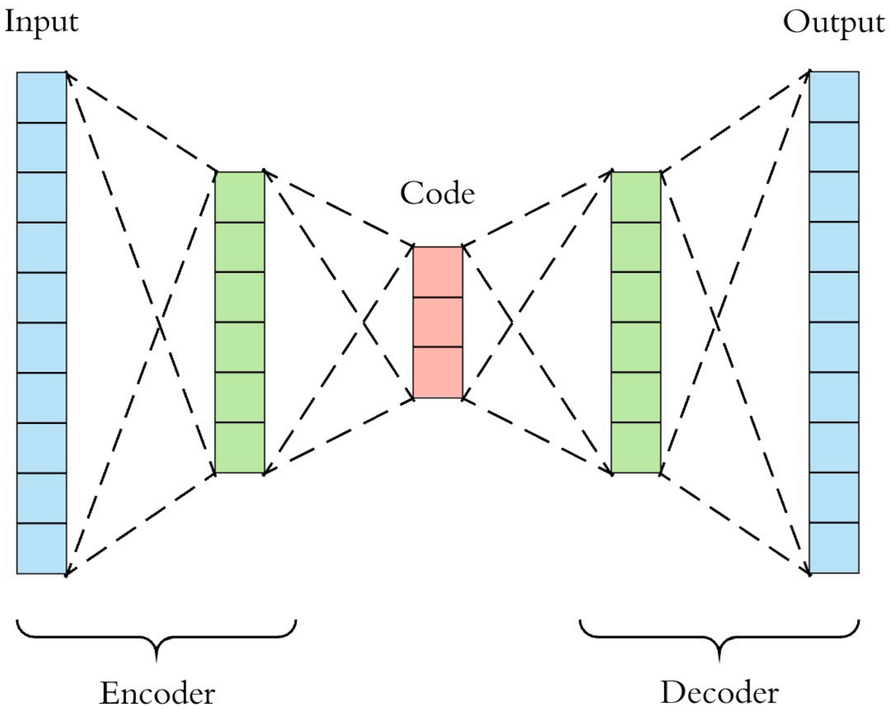


Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700–4708).

CNN: Encoder/decoder trainning

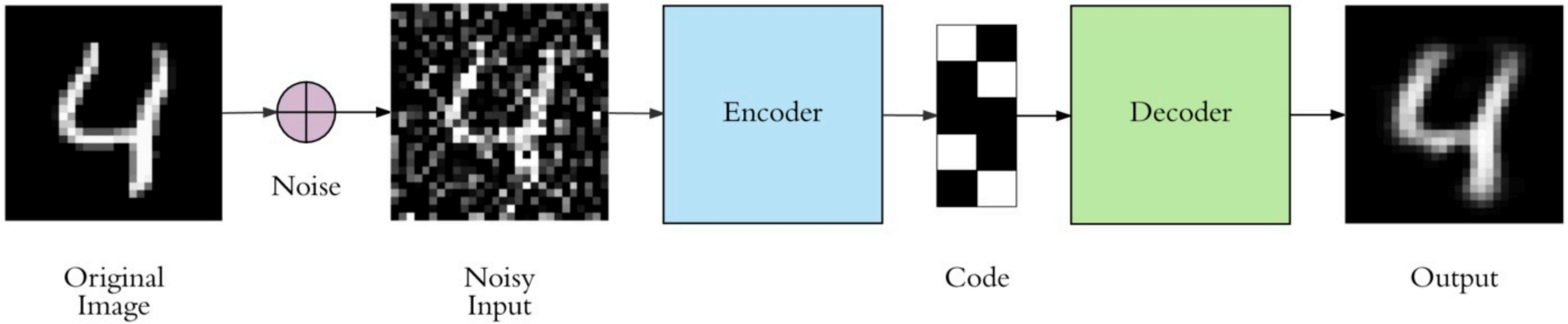
- A simple Encoder/decoder with FC layers.
- We want to recognize handwritten digits.
- How to train?





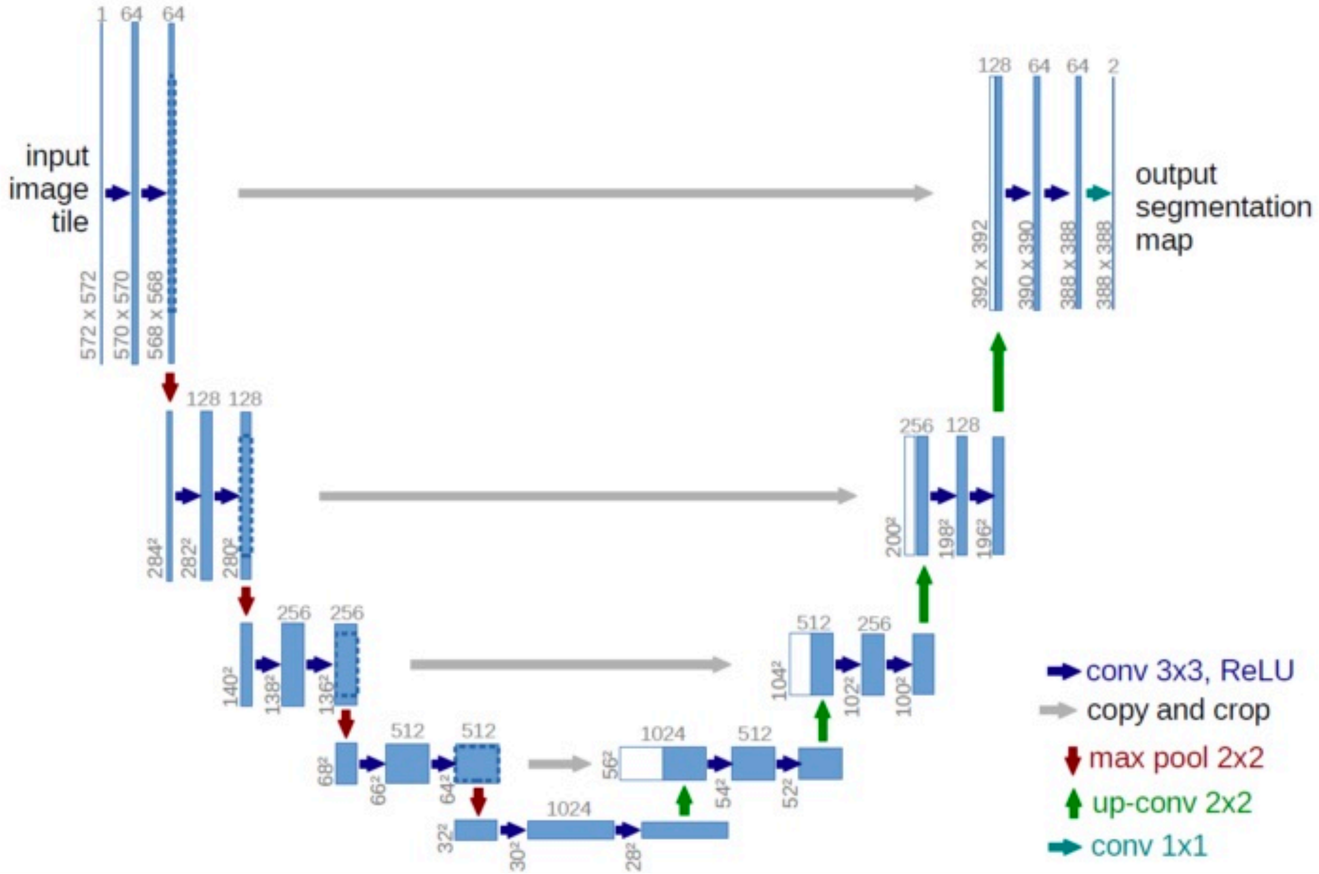
CNN: Encoder/decoder segmentation

- After training, we understand concepts and decode.



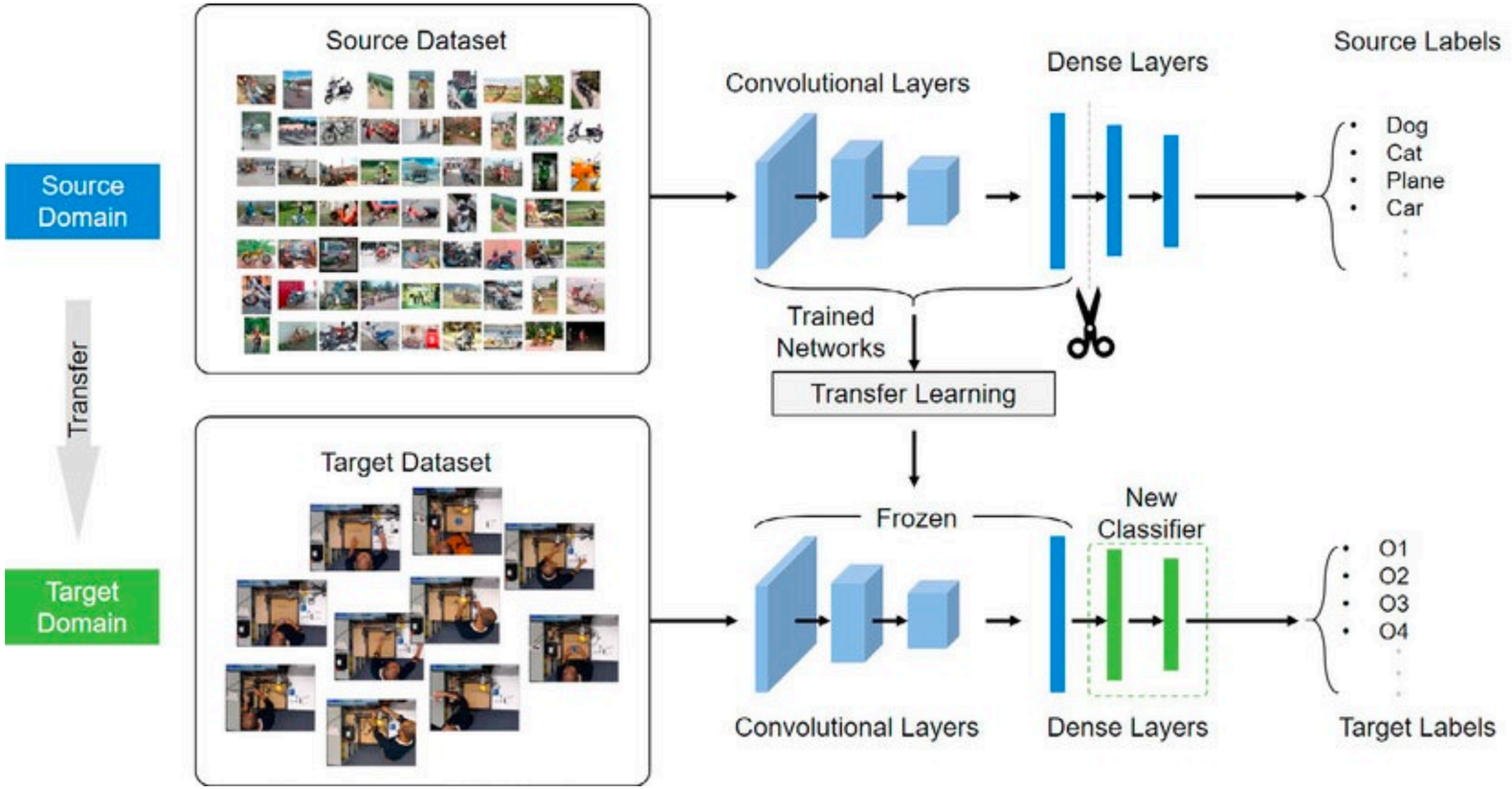
CNN: Encoder/decoder segmentation

- *Ronneberger et al 2015* proposed encoder/decoder for segmentation using convolutional layers (U-Net).
- Very popular in biomedical applications.



CNN: Transfer learning

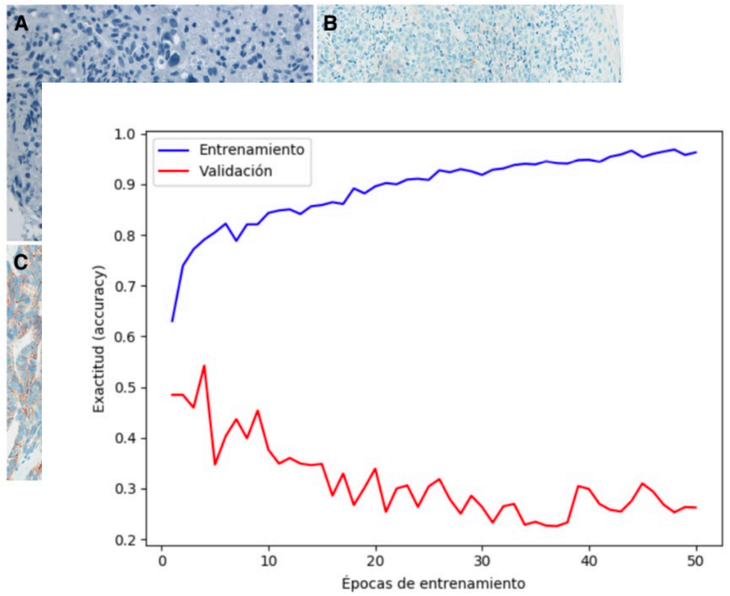
- Training a CNN requires many images (~10^6).
- To transfer learning implies to train in one domain and to partially train into another.



CNN: Transfer learning

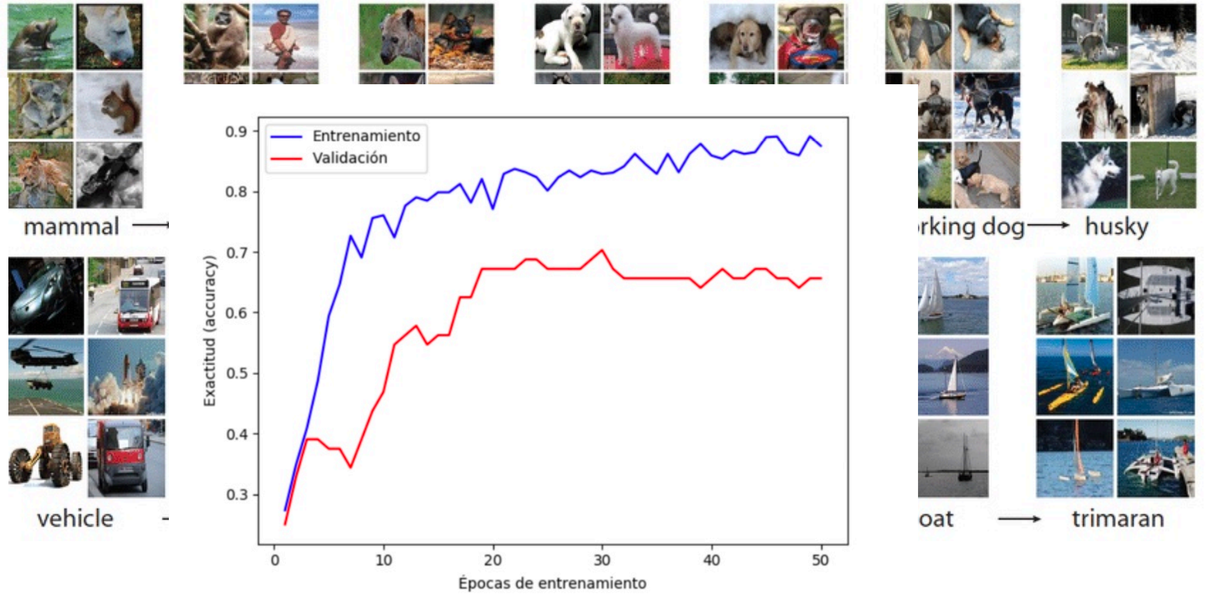
- Carefully evaluate distance between domains to define: what to transfer, what to refine.
- Example. Gastric Cancer using InceptionV3

Cáncer Gastrico HER2



Solo agregando capa de clasificación.

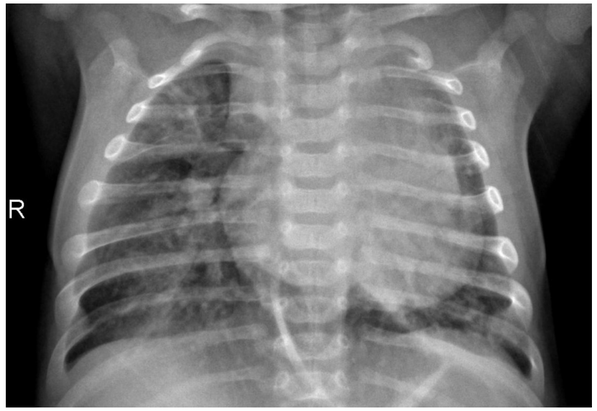
ImageNet

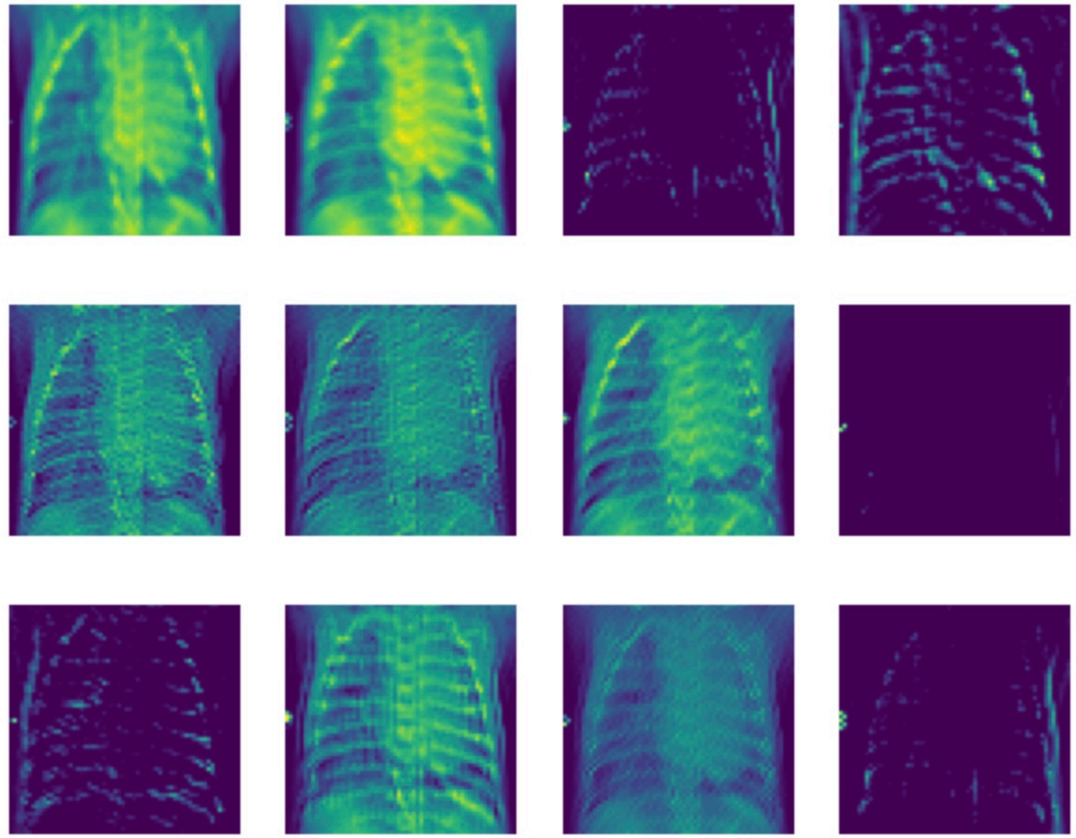


Con fine tunning

CNN visualization: activation

- How the network activates during inference?
- Remember we can visualize as filters!

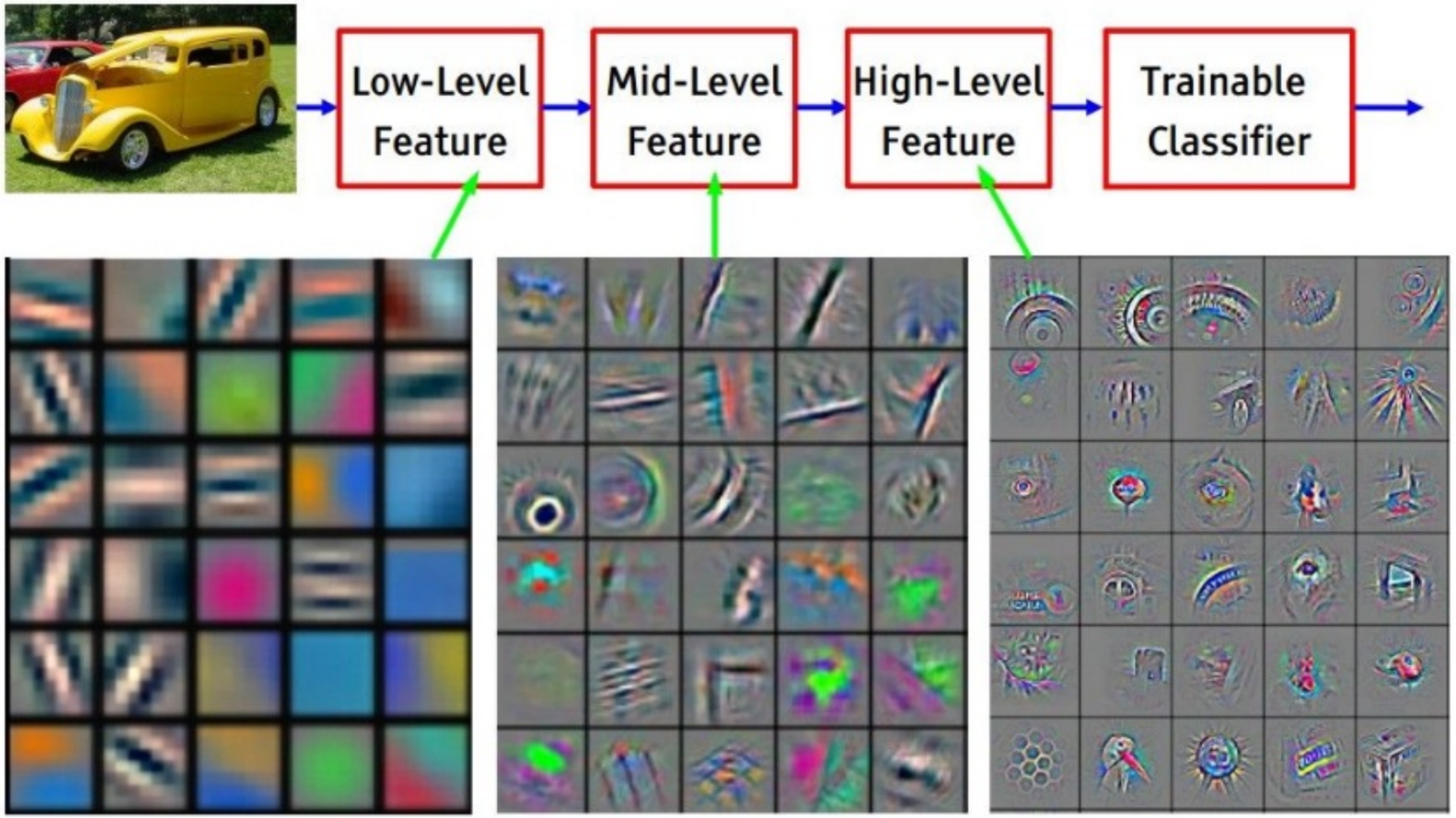




Set of convolutions in test architecture (first layer)

CNN visualization: kernels

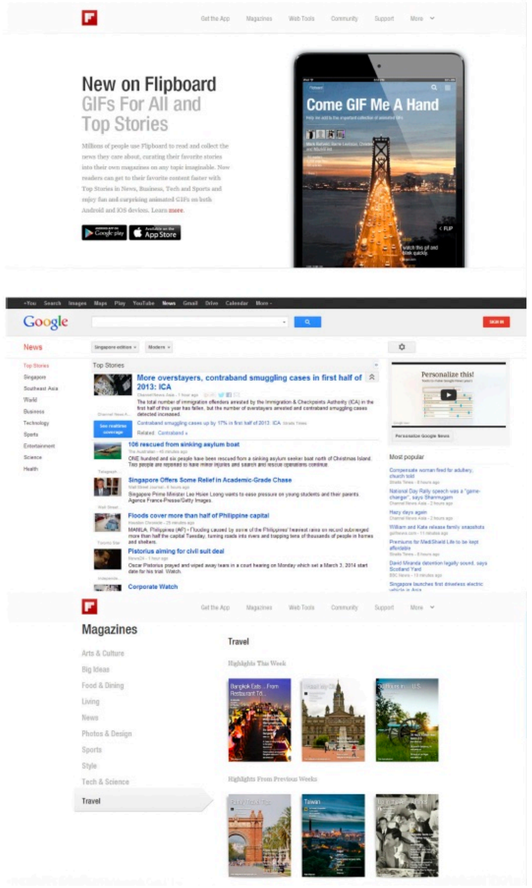
- How are the kernels the network learns? We can show kernels.



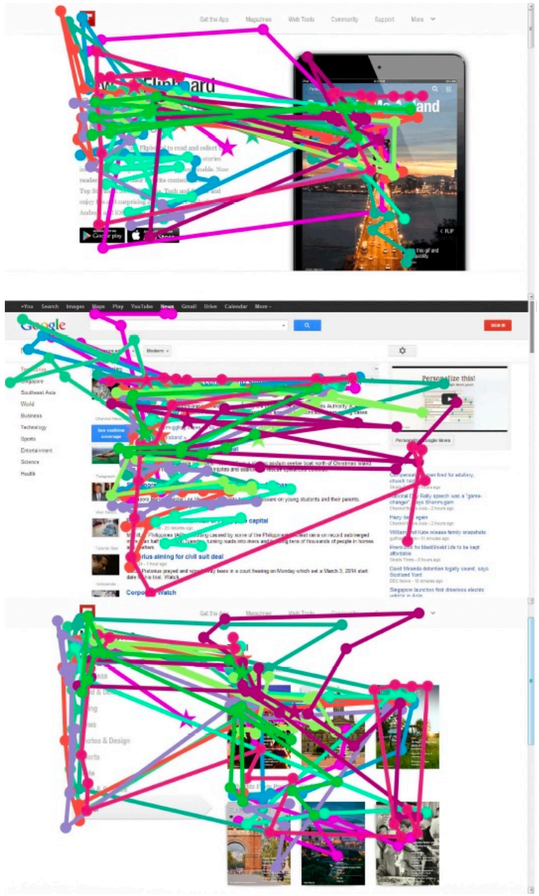
https://kamathhrishi.github.io/MyWebsite/jekyll/update/2020/06/19/bethealgorithm.html

CNN visualization: saliency map

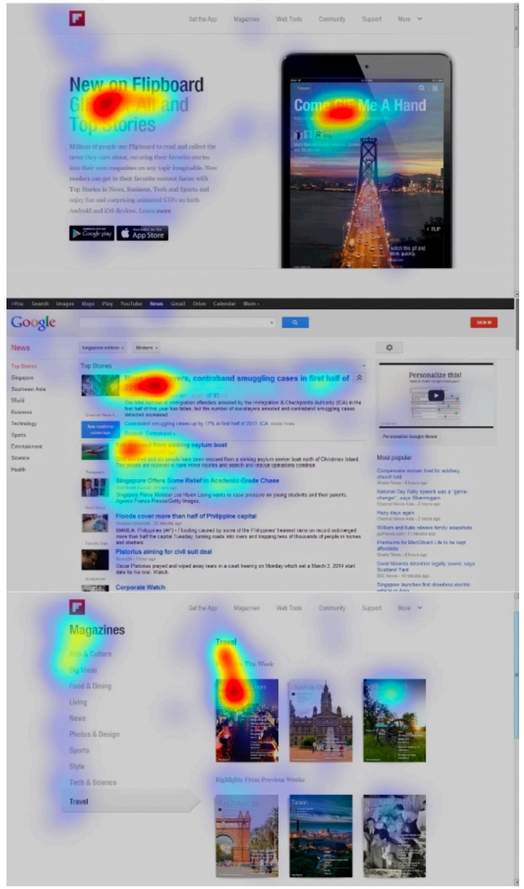
- In image processing models can predict where we will look at :)



Input



Saccadic patterns



Saccadic patterns heatmap

CNN visualization: saliency map

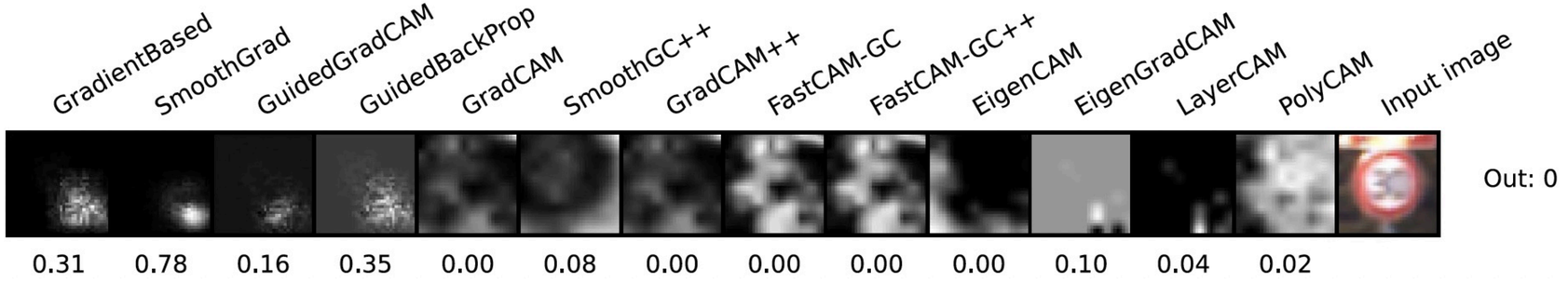
- For a model (AlexNet, VGG, etc), image *I*, and a class (cell or not cell), which pixels were more relevant to classify?
- If your network output is S(I), you can approximate as:

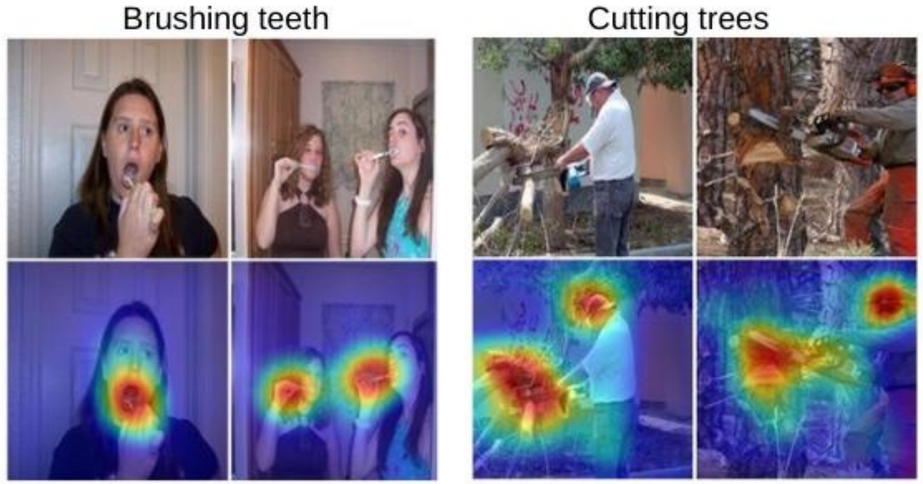
$$S(I) \approx w^T I + b \qquad w = \frac{\partial S}{\partial I}$$

- So, we can can quantify pixel relevance as how much gradient in back propagation it receives.

CNN visualization: saliency map

- Múltiple algorithms compute saliency maps: gradient based, deconvolution, class activation, grad-cam.





Szczepankiewicz, K., Popowicz, A., Charkiewicz, K. *et al.* Ground truth based comparison of saliency maps algorithms. *Sci Rep* **13**, 16887 (2023). https://doi.org/10.1038/s41598-023-42946-w
https://www.kaggle.com/code/kkhandekar/mapping-saliency-in-image-using-tensorflow

Exercises

- **Try** the imageProcessing_3_CNN.ipynb notebook
- Add a layer to the CNN, and visualize activations/kernels