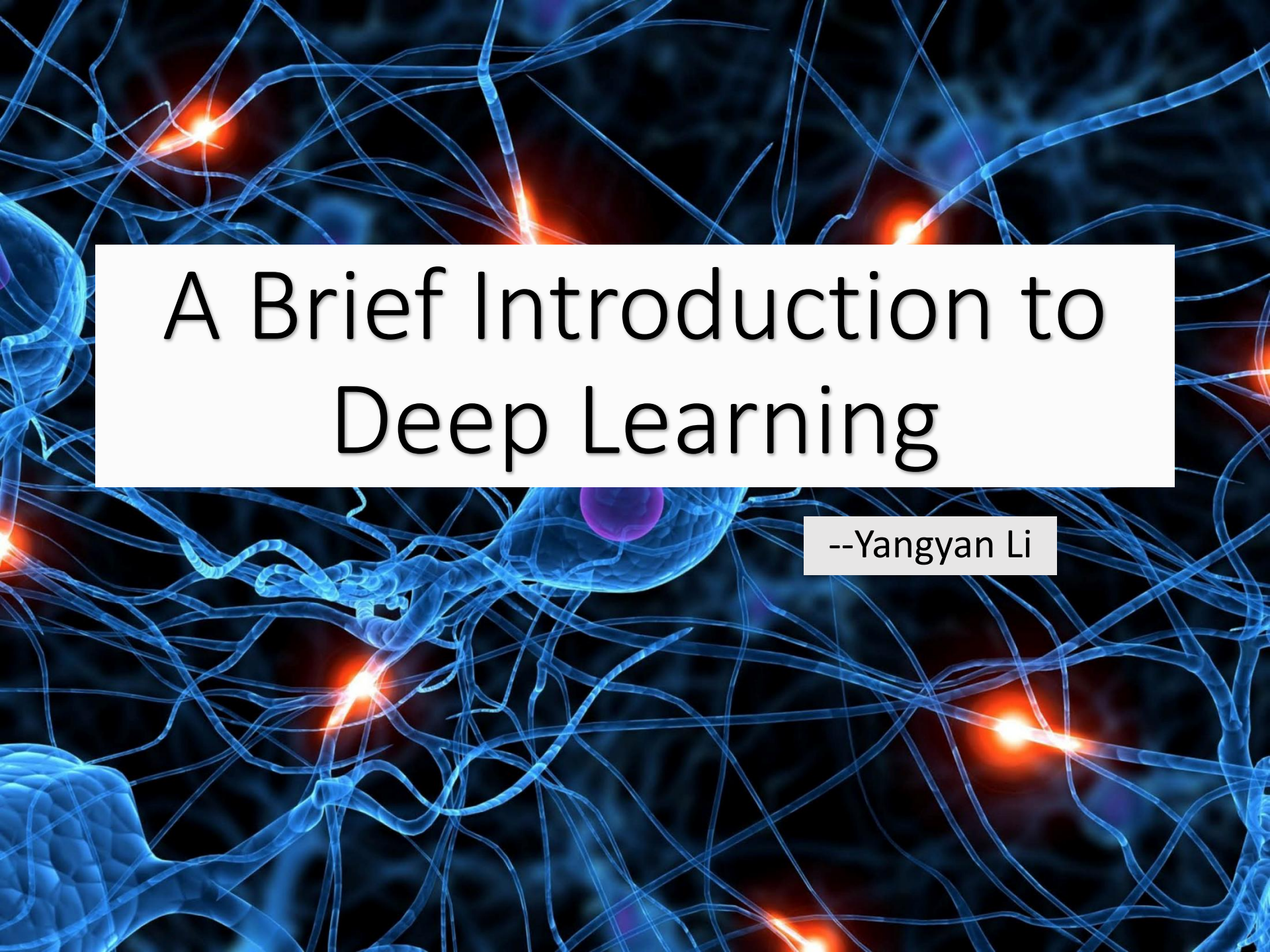


A Brief Introduction to Deep Learning

--Yangyan Li

ADVERTISERS SITEOWNERS

PRODUCTS

Slide to fit captcha

SITE OWNER

- 94% conversion (vs. 76% industry standard)
- Enhanced site security
- Free Security monitor
- New revenue stream

ADVERTISER

- Proven engagement – no ad blindness
- High CTR 1% - 3%
- Direct Facebook "Like" clicks
- Pay only for real user engagement

USER

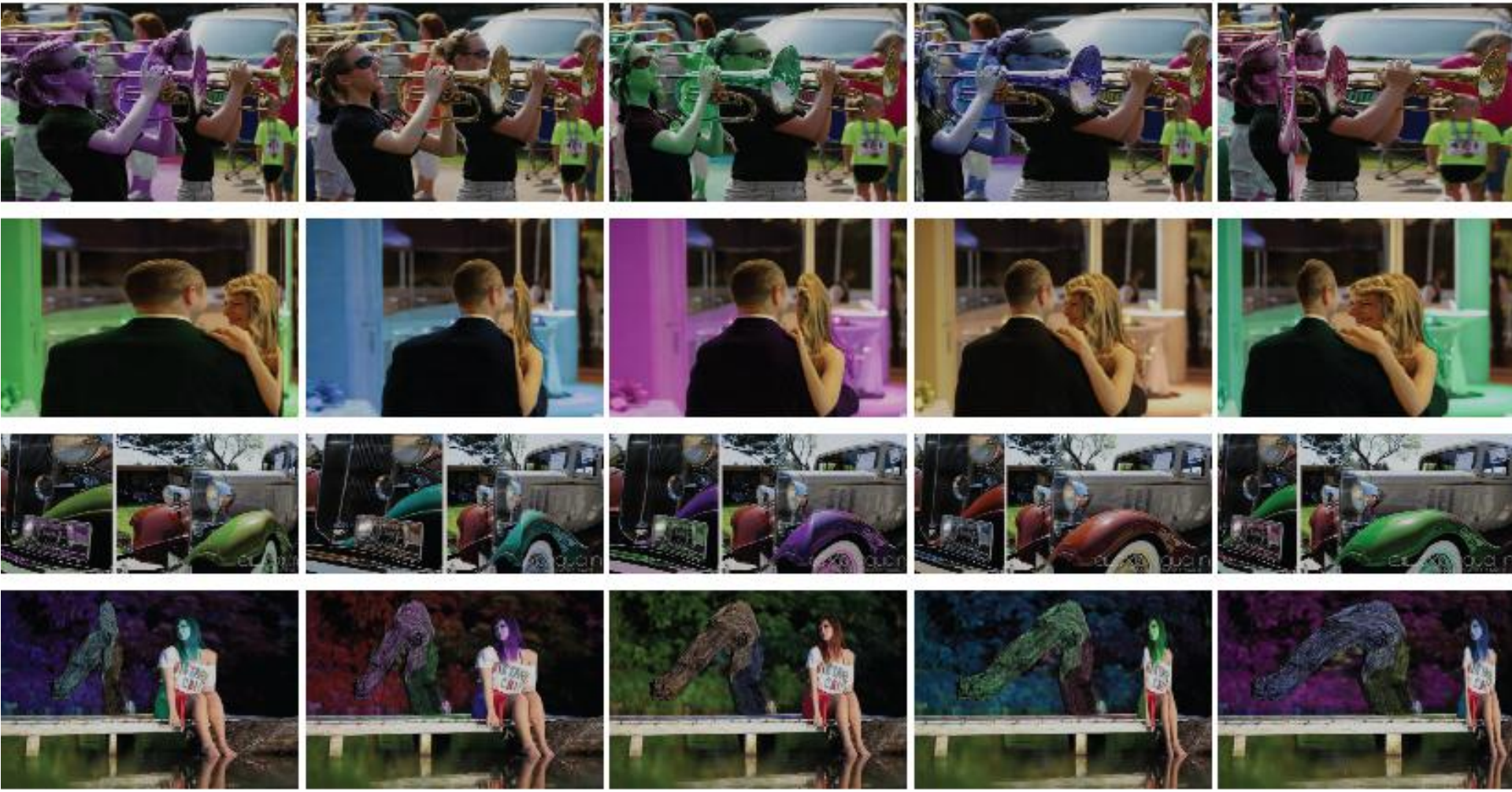
- No Captcha frustration

TRY ME NOW

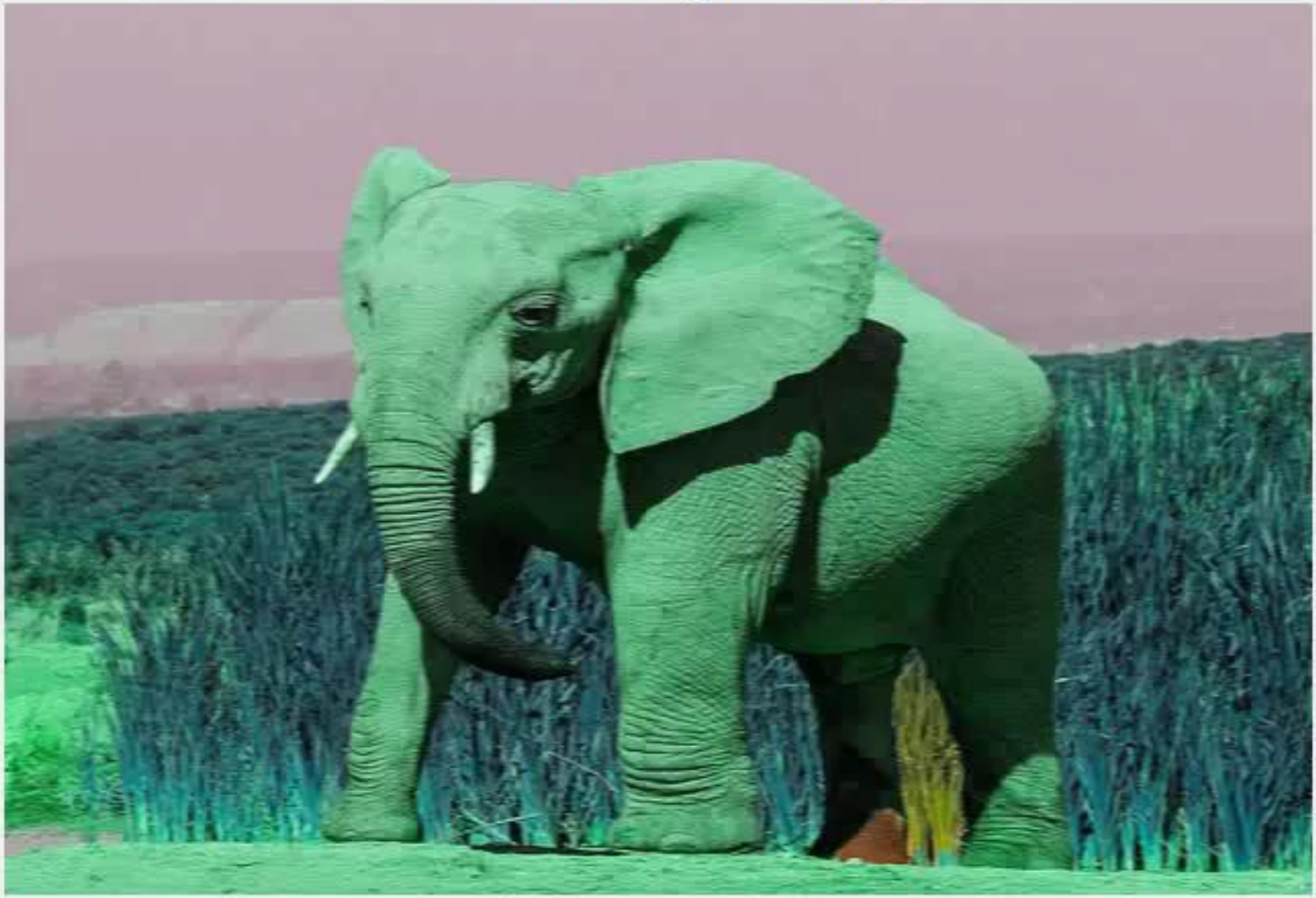
How would you crack it?

How to avoid being cracked?

Seam Carving!



Slide to the Original Image



‹ ›

Submit

Labradoodle or fried chicken

Puppy or bagel

Sheepdog or mop

Chihuahua or muffin

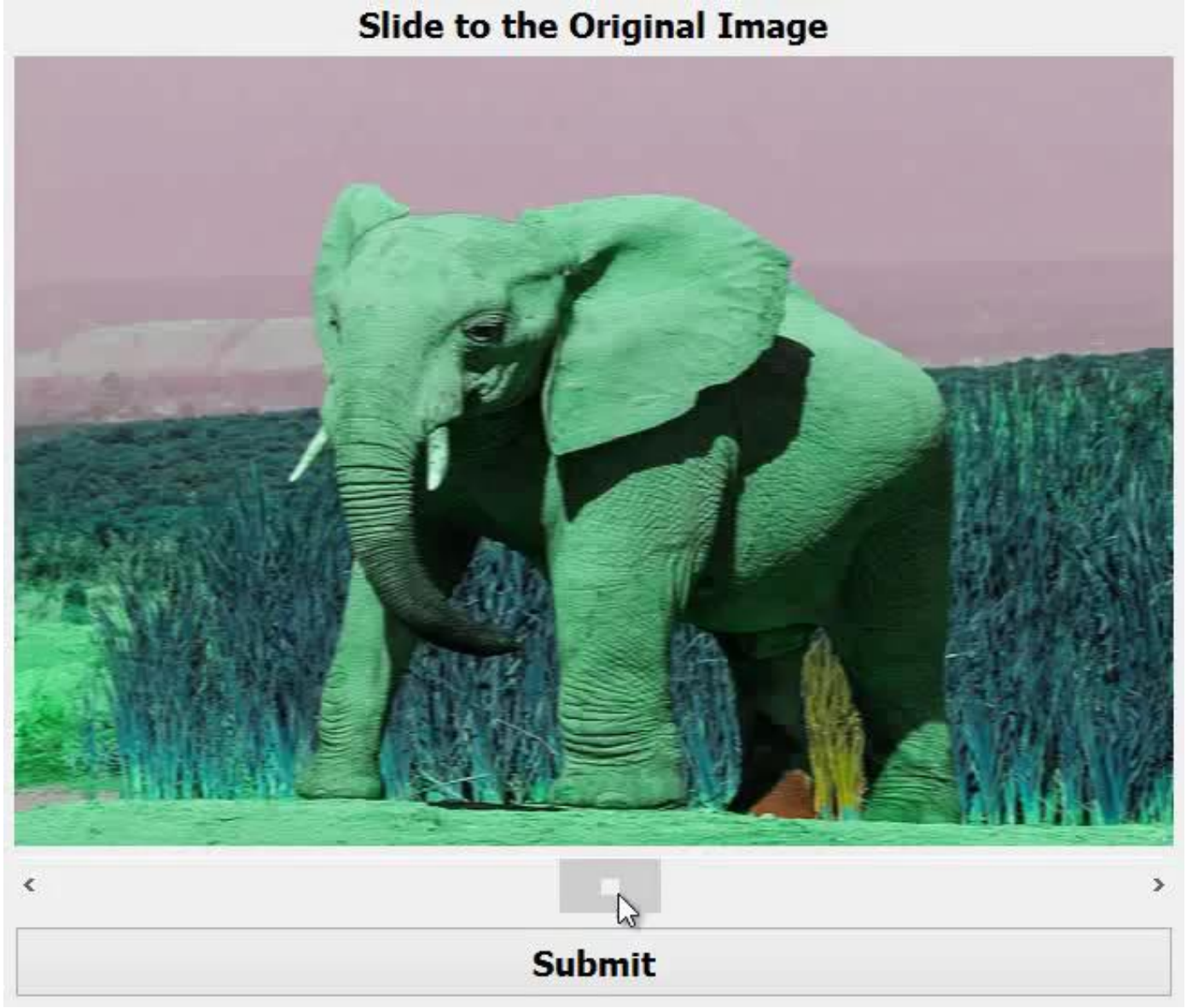
Barn owl or apple

Parrot or guacamole

Raw chicken or Donald Trump

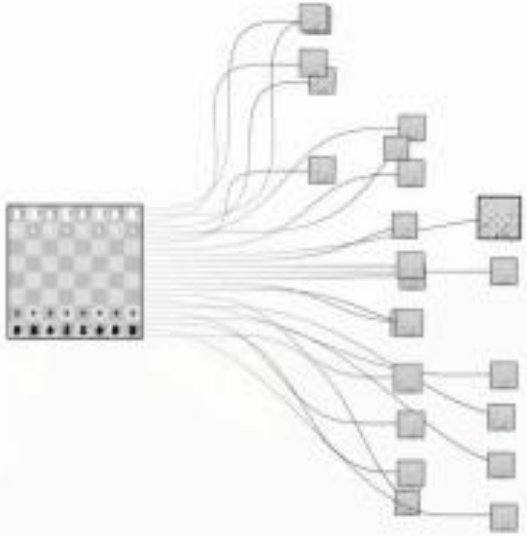
But, we human actually lose!

- A demo that shows we, human, lose, on the classification task, we are proud of, we have been trained for millions of years!
- If we want to make it hard for bots, it has to be hard for human as well.



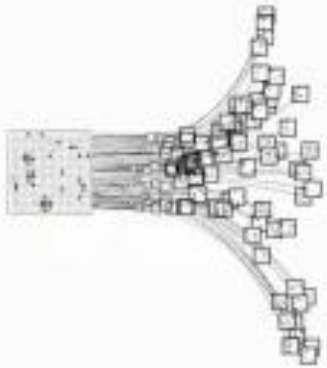
How would you crack it?

We human lose on Go!



Chess: 10^{47}

Deep Blue, Feb 10, 1996



Go: 10^{170}

AlphaGo, March, 2016

We (will) lose on many specific tasks!

- Speech recognition
- Translation
- Self-driving
- …

- BUT, they are not AI yet…
- Don’t worry until it dates with your girl/boy friend…

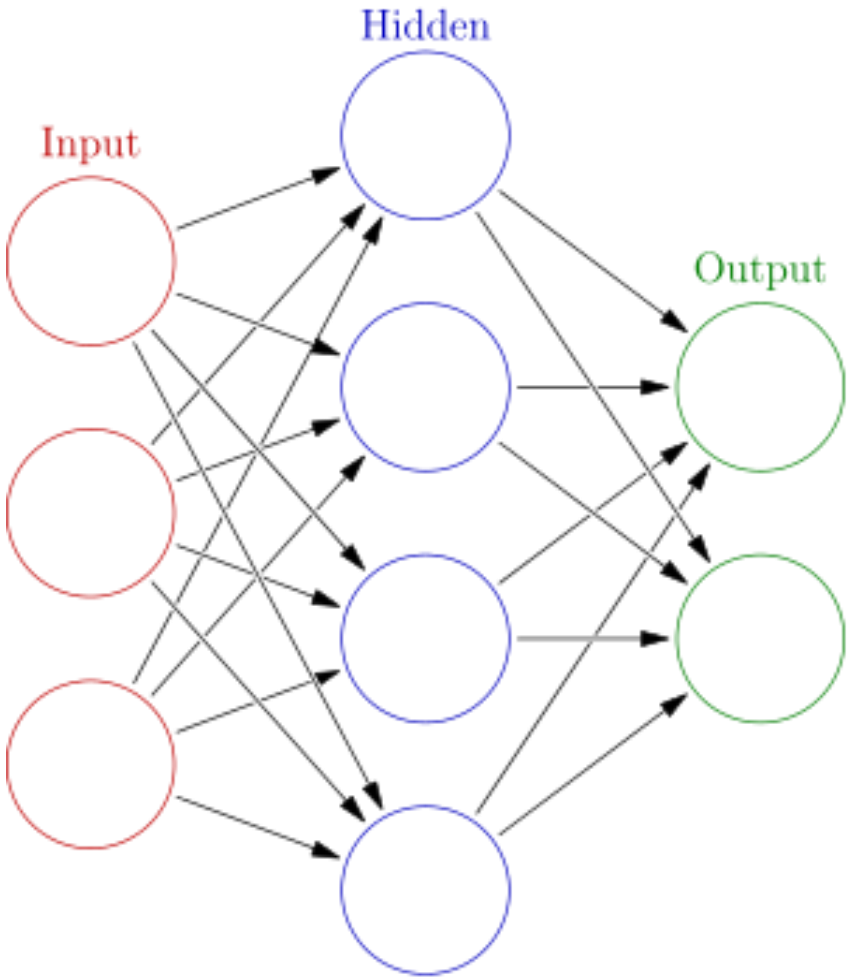
Deep learning is so cool for so many problems…


OLD CROW
MODERN CROW
Update Yourself - It saves a lot of extra effort

A Brief Introduction to Deep Learning

- Artificial Neural Network
- Back-propagation
- Fully Connected Layer
- Convolutional Layer
- Overfitting

Artificial Neural Network



1. Activation function
2. Weights
3. Cost function
4. Learning algorithm

Live Demo

Neurons are functions

- Let's start with a complex one!

$$f(x, y) = x + y$$

- Given $x = a, y = b$, how to update x *and* y to make $f(x, y)$ larger?
- Follow gradient directions!

$$f(x, y) = x + y \qquad \rightarrow \qquad \frac{\partial f}{\partial x} = 1 \qquad \frac{\partial f}{\partial y} = 1$$

$$x = a + 0.01 * 1,$$
$$y = b + 0.01 * 1$$
$$f(x, y): a + b \rightarrow \mathrm{a} + \mathrm{b} + 0.02$$

Neurons are functions

- A more complex one!

$$f(x, y) = x * y$$

- Given $x = a, y = b$, how to update x *and* y to make $f(x, y)$ larger?
- Follow gradient directions!

$$f(x, y) = xy \qquad \rightarrow \qquad \frac{\partial f}{\partial x} = y \qquad \frac{\partial f}{\partial y} = x$$

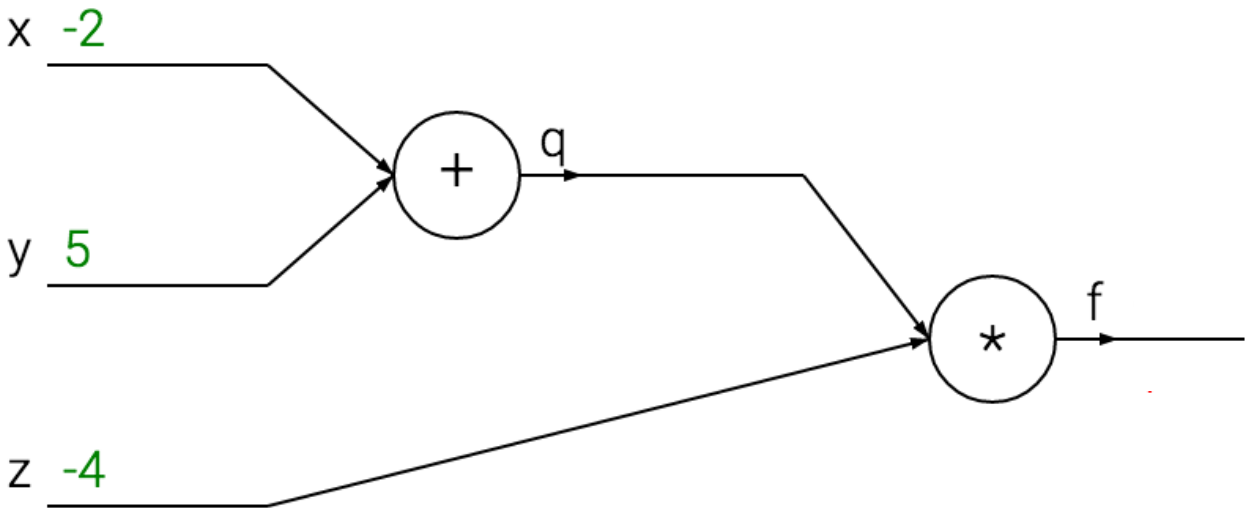
$$x = a + 0.01 * b,$$
$$y = b + 0.01 * a$$
$$f(x, y): a * b \rightarrow (\mathrm{a} + 0.01 * \mathrm{b})(\mathrm{b} + 0.01 * \mathrm{a})$$
$$f(x, y): 4 * (-3) \rightarrow 3.97 * (-2.96)$$

Back-propagation

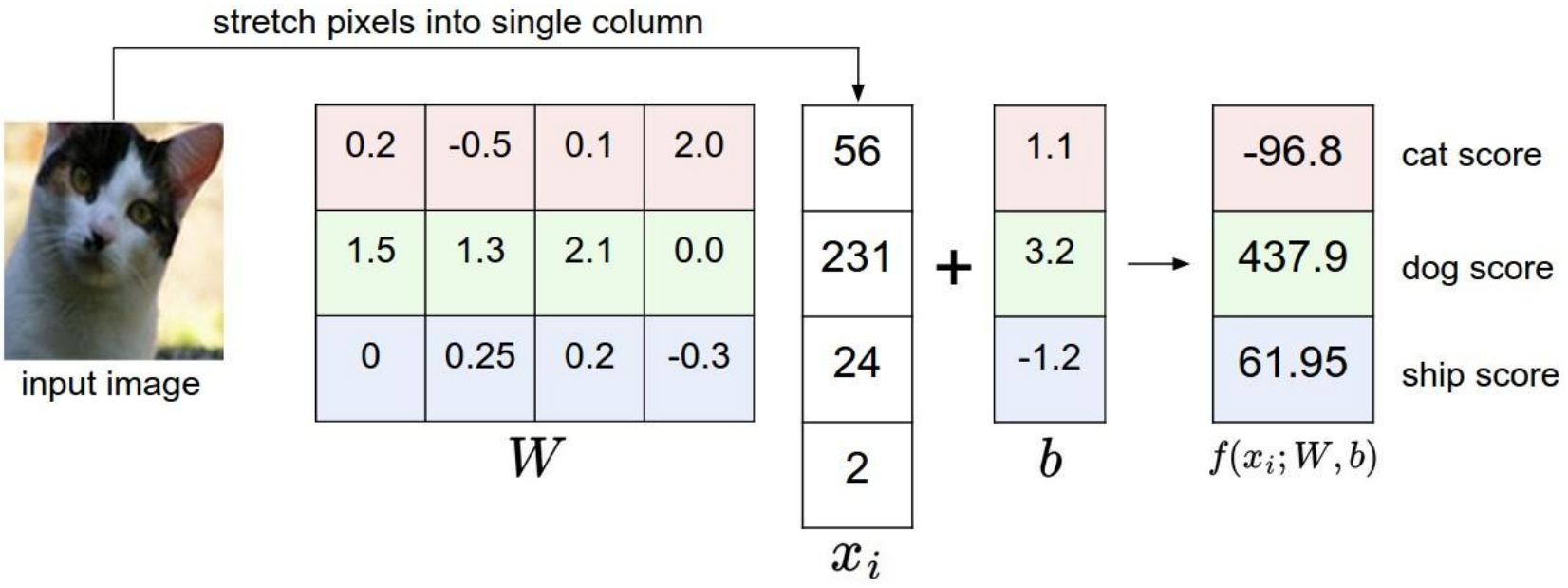
- An extremely complex one!

$$f(x, y, z) = (x + y) * z$$

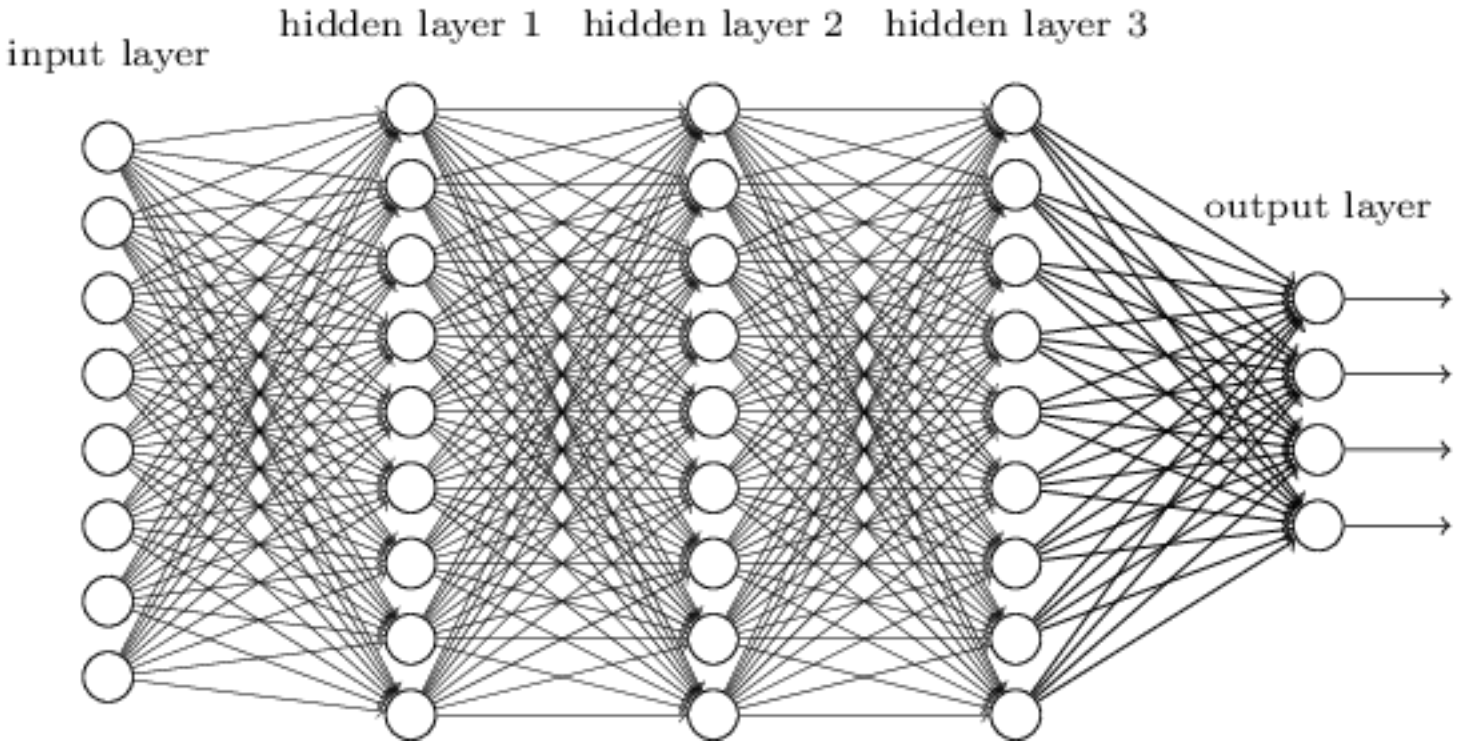
- Let $q(x, y) = (x + y)$, then $f(x, y, z) = q(x, y) * z$
- Chain rule: $\frac{\partial f}{\partial x} = \frac{\partial f}{\partial q} \frac{\partial q}{\partial x}$



Now, serious stuff, a bit…



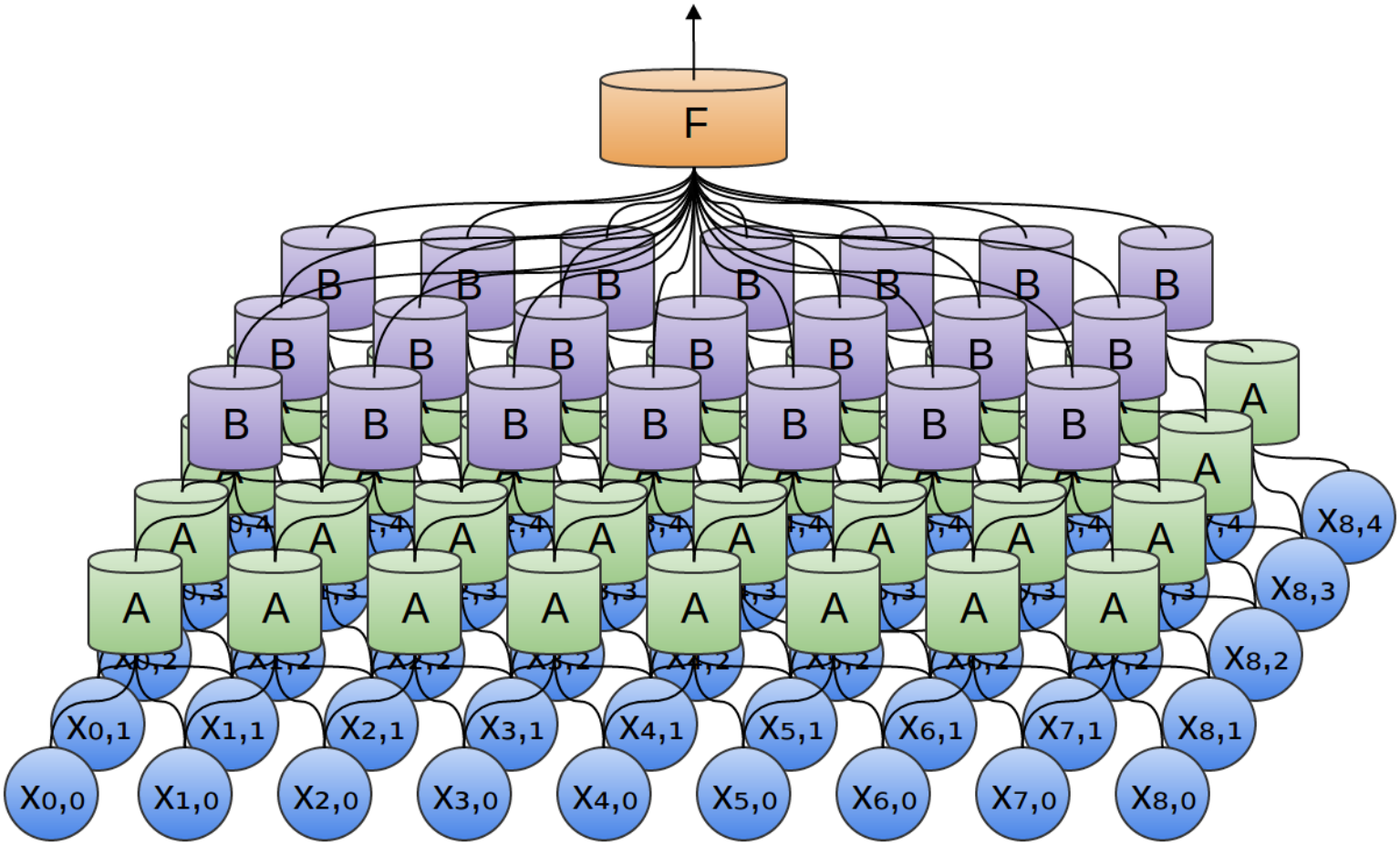
Fully Connected Layers



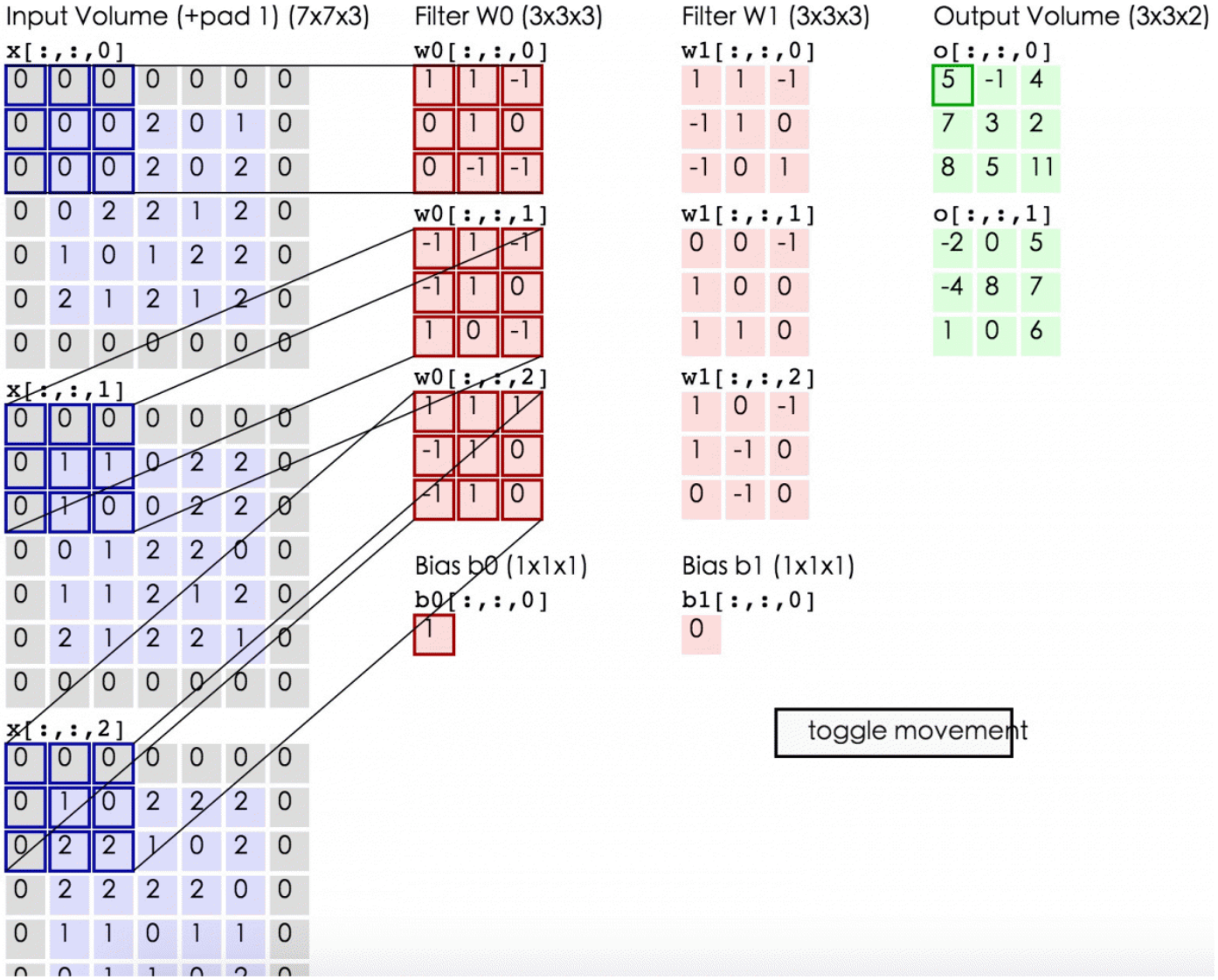
“When in doubt, use brute force.”
--Ken Thompson

“If brute force is possible...”
--Yangyan Li

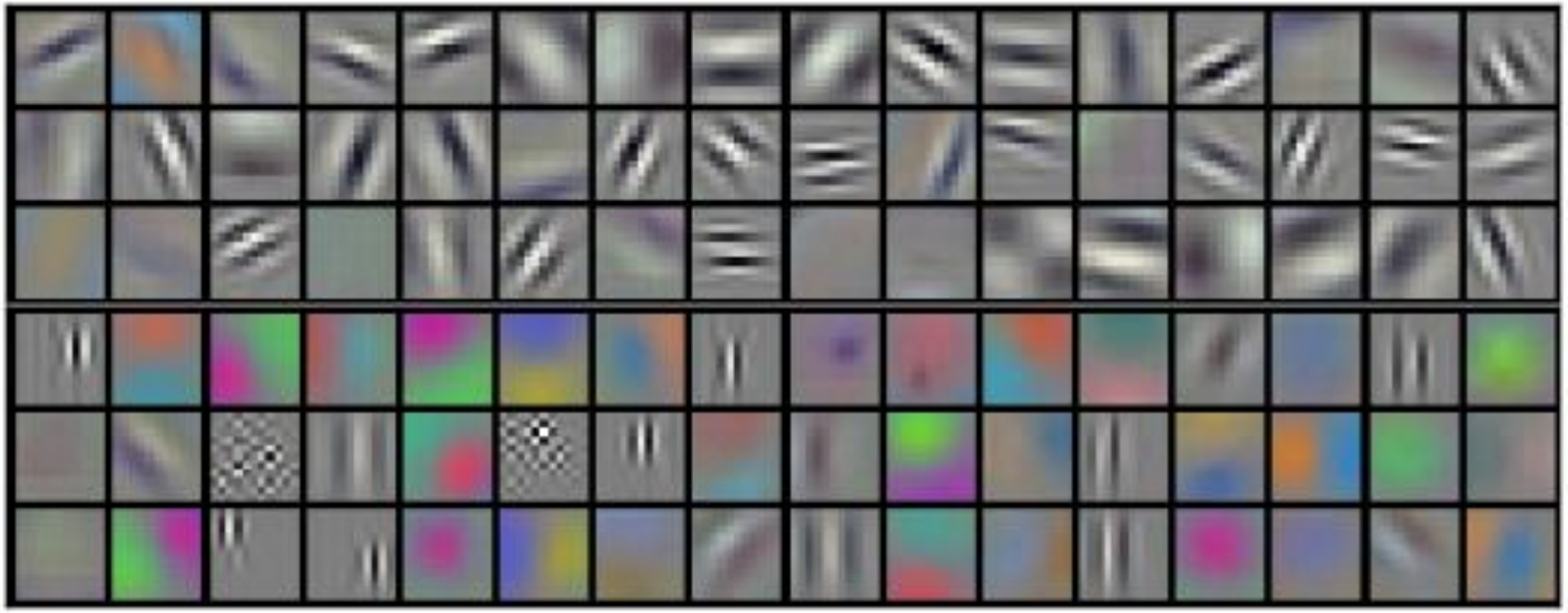
Convolutional Layers



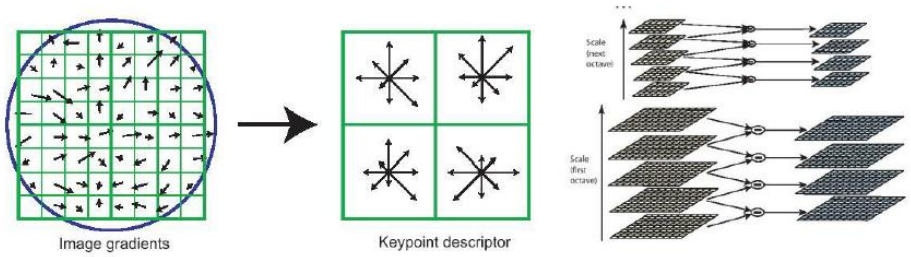
Convolutional Layers



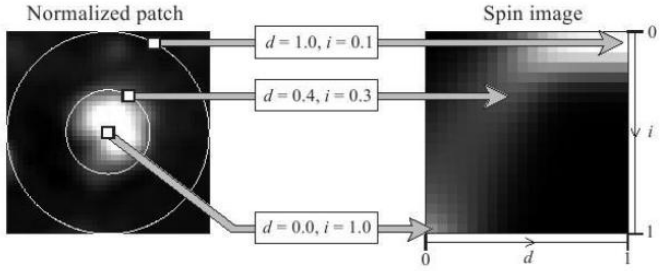
Convolution Filters



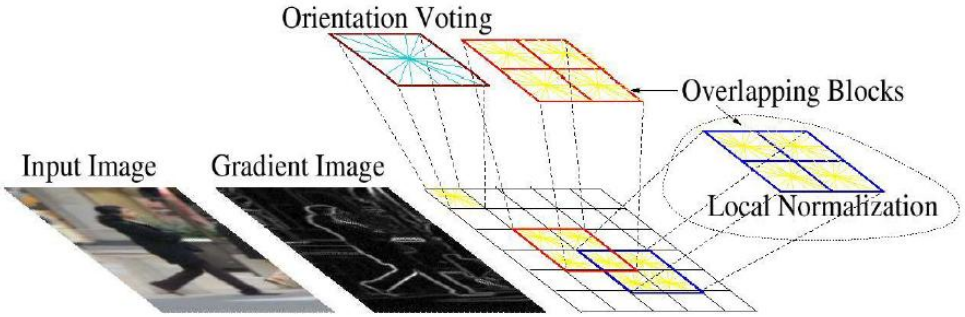
Computer vision features



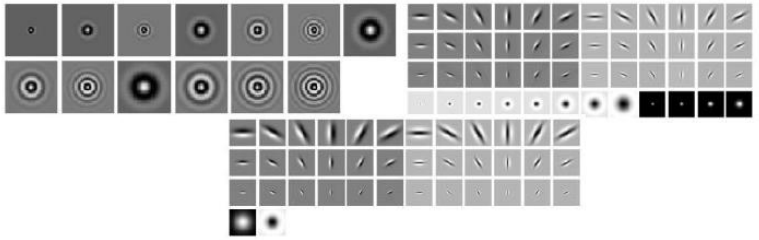
SIFT



Spin image



HoG



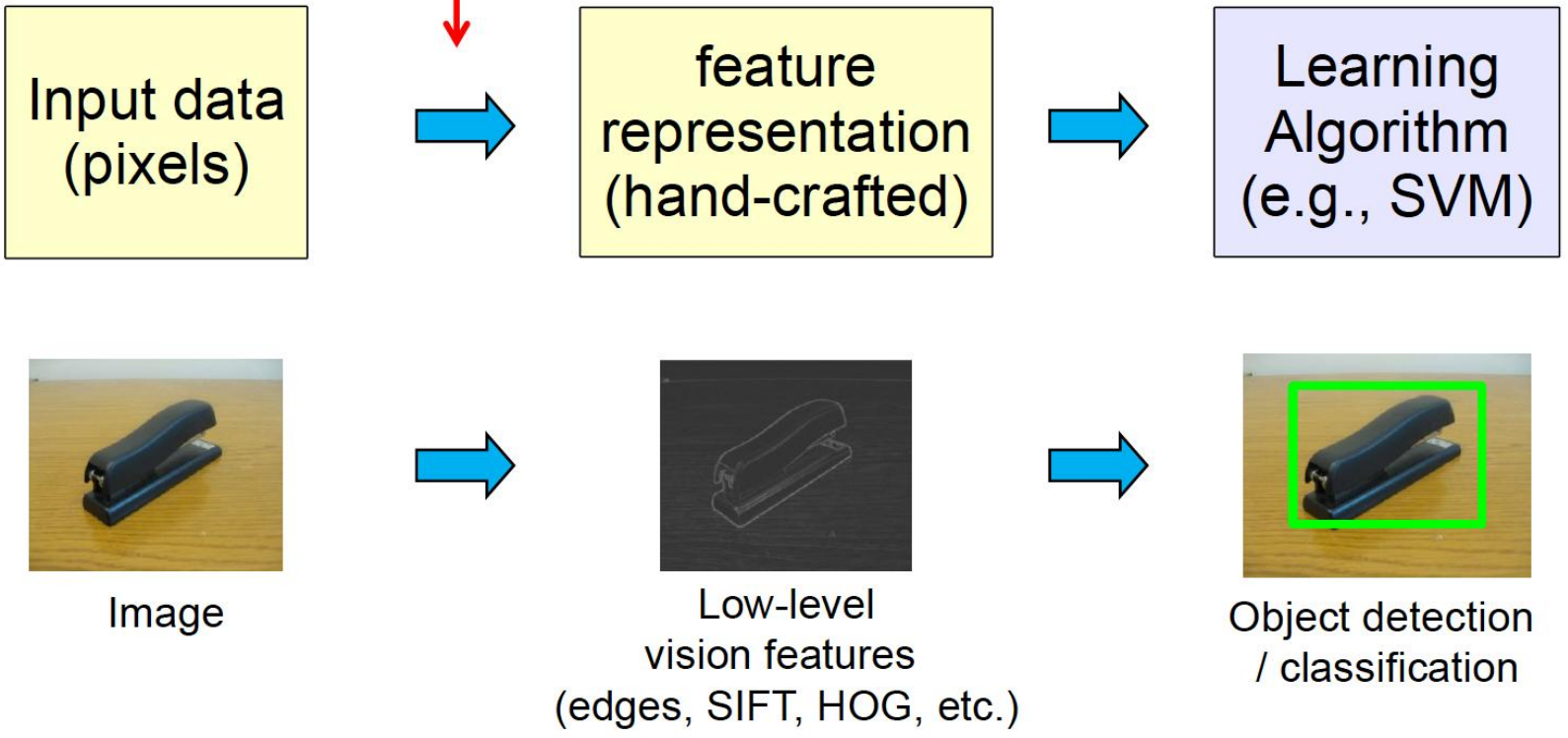
Textons

and many others:

SURF, MSER, LBP, Color-SIFT, Color histogram, GLOH, …..

Traditional Recognition Approach

Features are not learned



Feature Engineering vs. Learning

- Feature engineering is the process of using domain knowledge of the data to create features that make machine learning algorithms work.
- “When working on a machine learning problem, feature engineering is manually designing what the input x's should be.”

-- Shayne Miel

- “Coming up with features is difficult, time-consuming, requires expert knowledge.”

--Andrew Ng

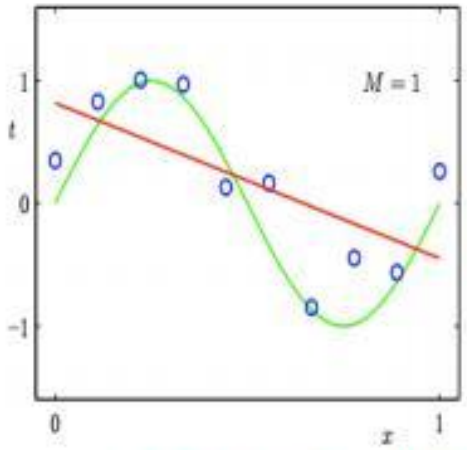
With four parameters I can fit an elephant, and with five I can make him wiggle his trunk.

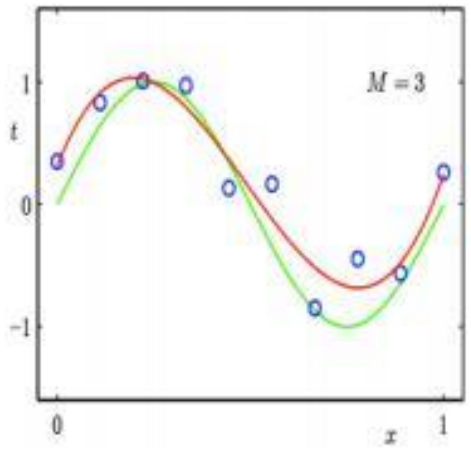
— John von Neumann —

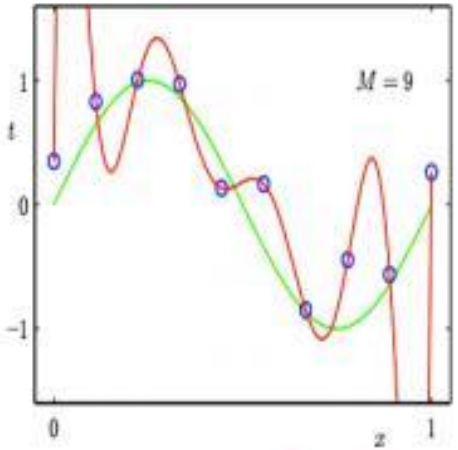
AZ QUOTES

Under- and Over-fitting examples

Regression:



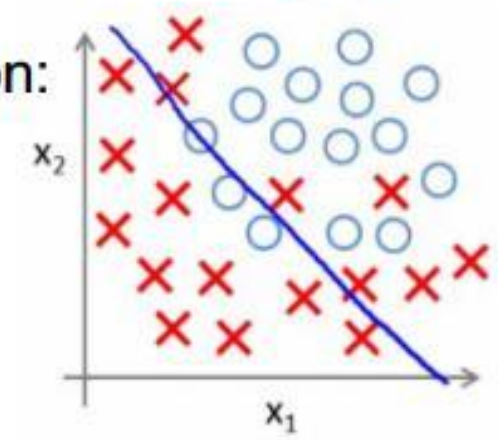


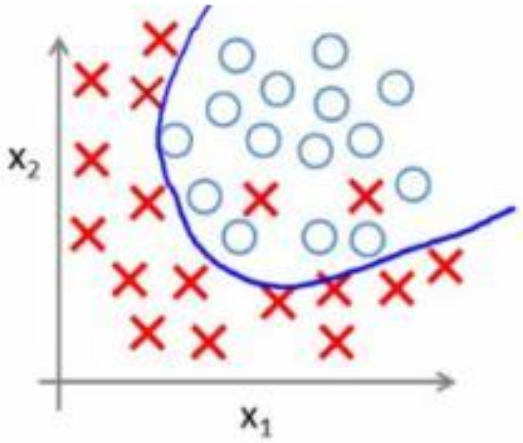


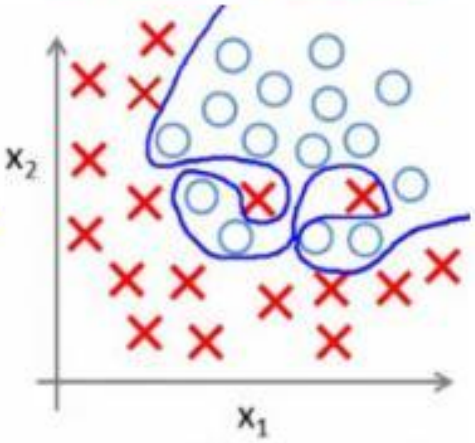
predictor too inflexible: cannot capture pattern

predictor too flexible: fits noise in the data

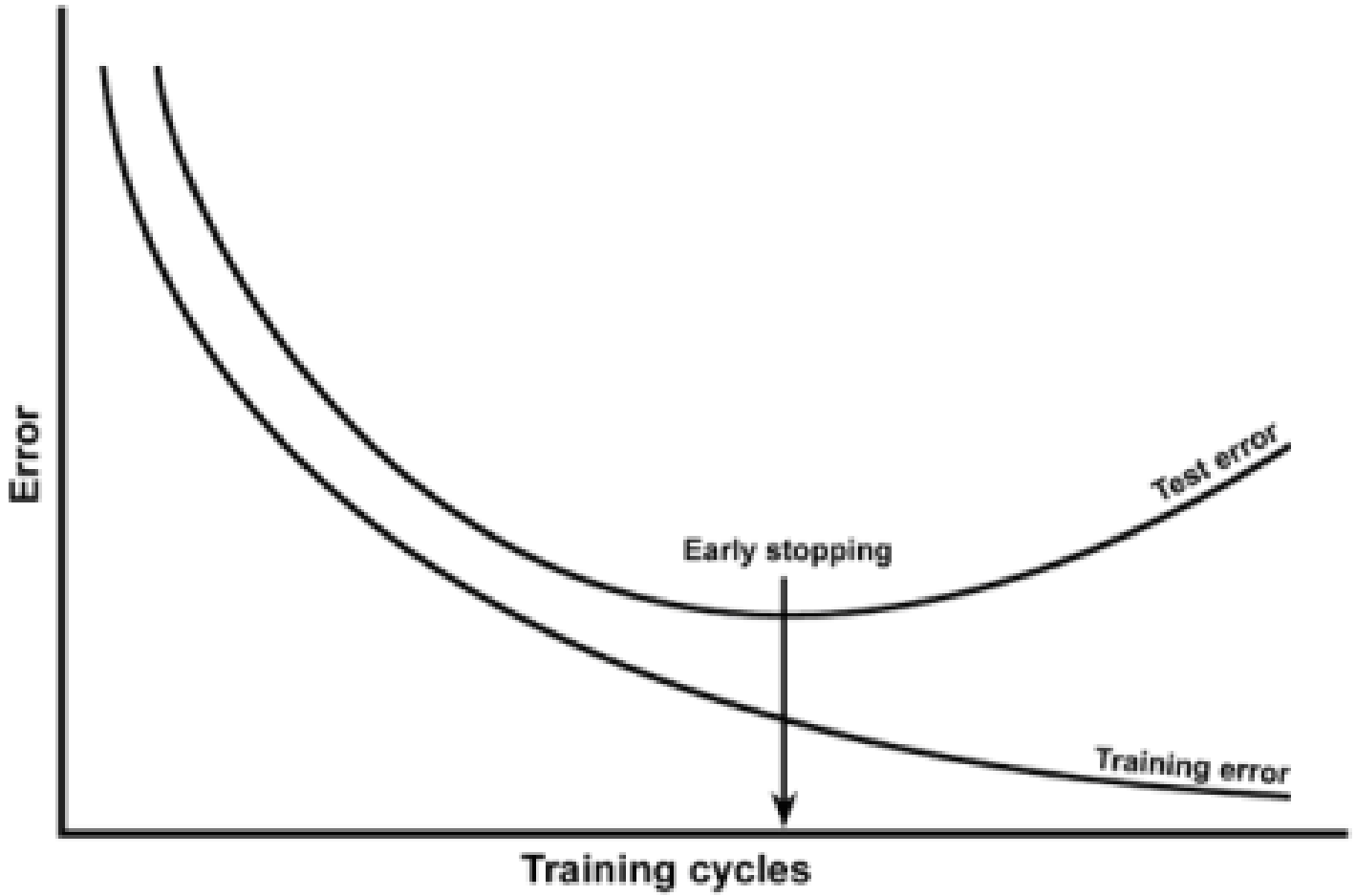
Classification:



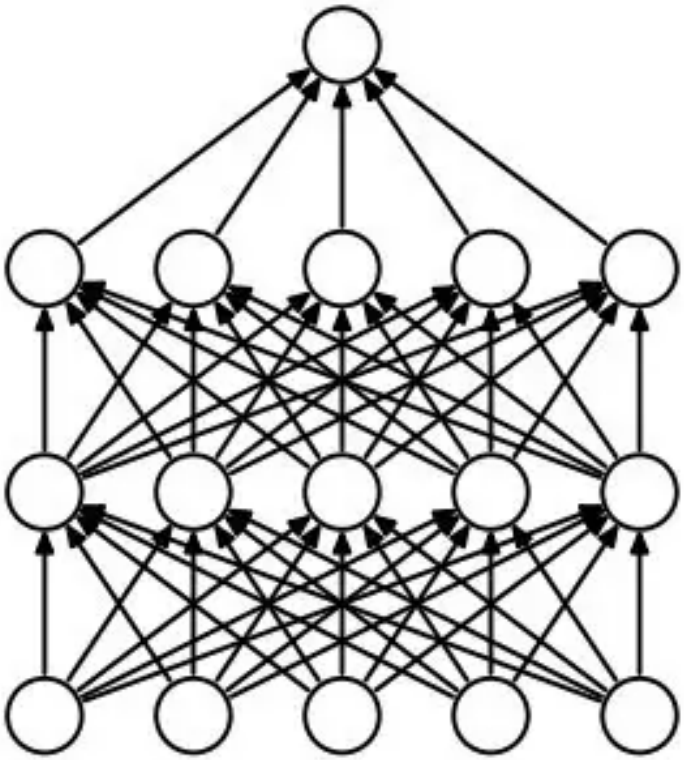


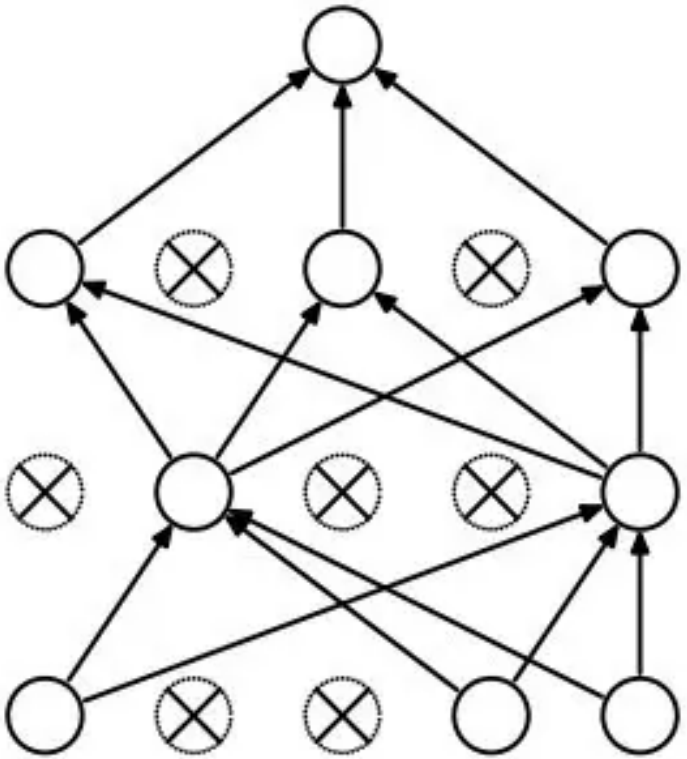


How to detect it in training process?

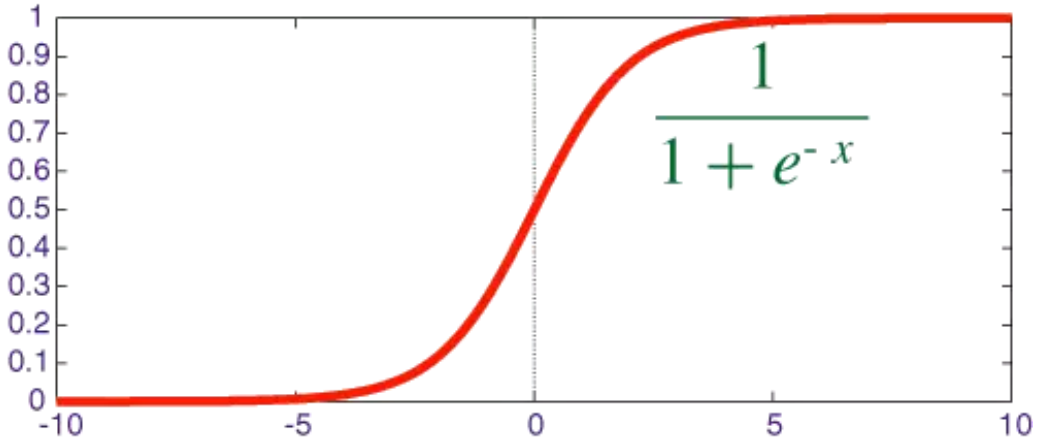


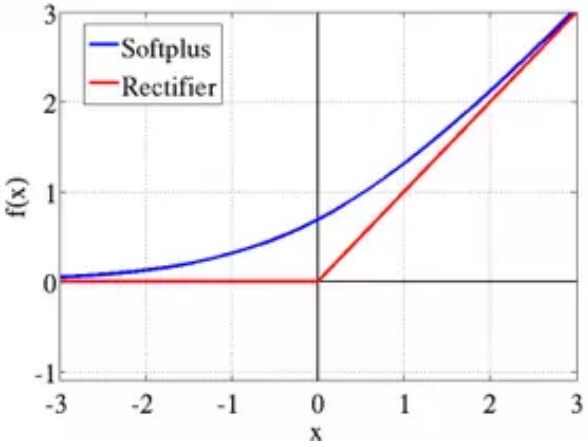
Dropout



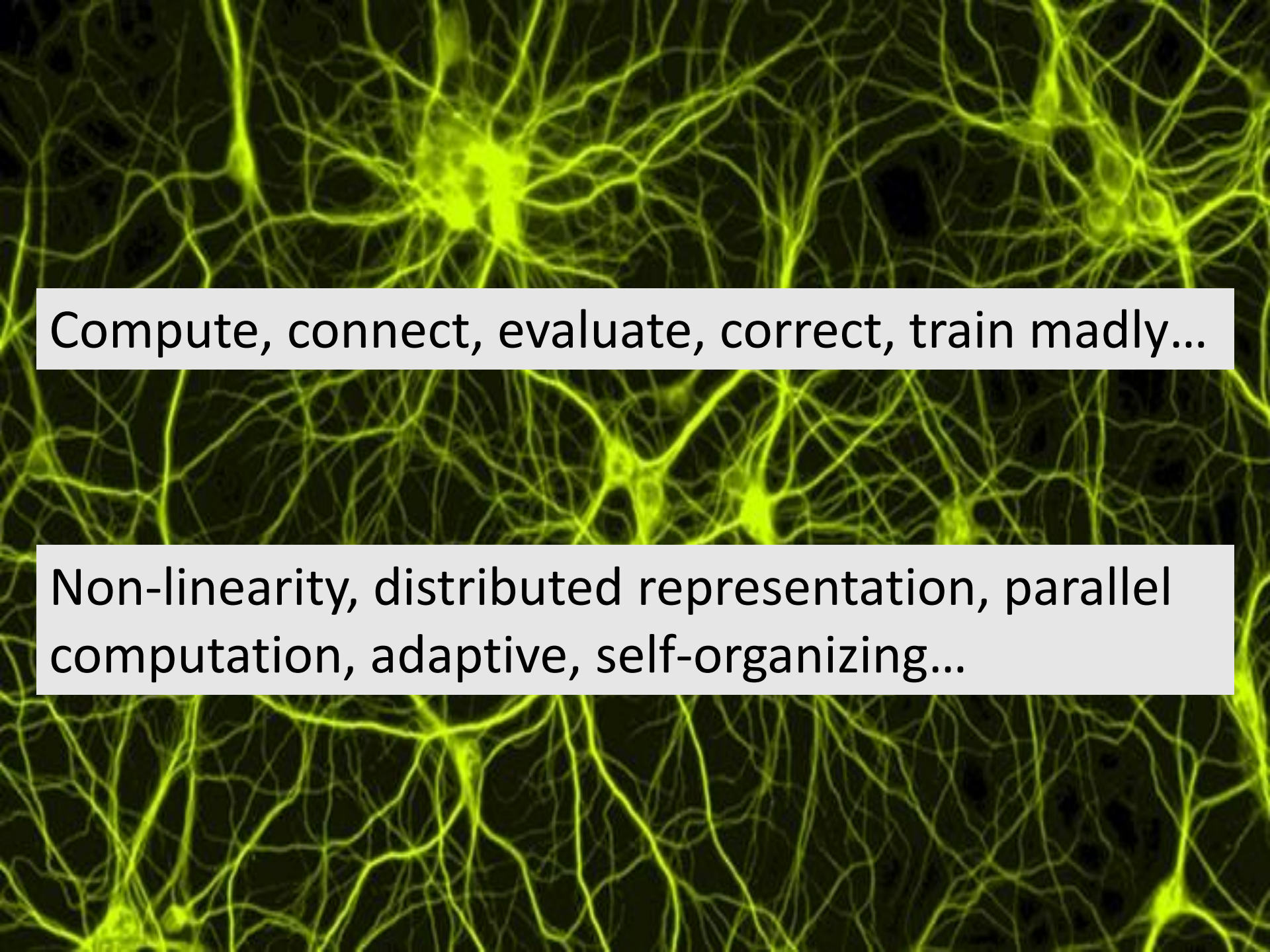


Sigmod → ReLU


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-10
-5
0
5
10
$\frac{1}{1+e^{-x}}$


Softplus
Rectifier
f(x)
x
3
2
1
0
-1
-3
-2
-1
0
1
2
3

Sigmod → ReLU


Compute, connect, evaluate, correct, train madly…
Non-linearity, distributed representation, parallel computation, adaptive, self-organizing…

A brief history

- McCulloch, Warren S., and Walter Pitts. "A logical calculus of the ideas immanent in nervous activity." *The bulletin of mathematical biophysics* 5.4 (1943): 115-133.
- Rosenblatt, Frank. "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological review* 65.6 (1958): 386.
- Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams. "Learning representations by back-propagating errors." *Cognitive modeling* 5.3 (1988): 1.
- LeCun, Yann, et al. "Backpropagation applied to handwritten zip code recognition." *Neural computation* 1.4 (1989): 541-551.
- 1993: Nvidia started…
- Hinton, Geoffrey E., Simon Osindero, and Yee-Whye Teh. "A fast learning algorithm for deep belief nets." *Neural computation* 18.7 (2006): 1527-1554.
- Raina, Rajat, Anand Madhavan, and Andrew Y. Ng. "Large-scale deep unsupervised learning using graphics processors." *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009.
- Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database."*Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009.
- 2010: “GPUS ARE ONLY UP TO 14 TIMES FASTER THAN CPUS” SAYS INTEL –Nvidia
- Glorot, Xavier, Antoine Bordes, and Yoshua Bengio. "Deep sparse rectifier neural networks." *International Conference on Artificial Intelligence and Statistics*. 2011.
- Hinton, Geoffrey E., et al. "Improving neural networks by preventing co-adaptation of feature detectors." *arXiv preprint arXiv:1207.0580* (2012).
- Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.

“Now this is not the end. It is not even the beginning of the end. But it is, perhaps, the end of the beginning.”

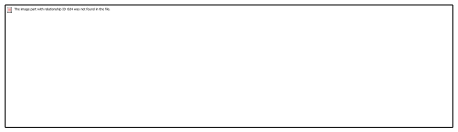
--Winston Churchill

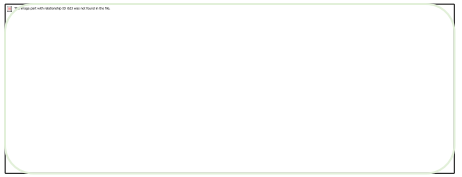
Is Deep Learning Taking Over the World?

- What applications are likely/unlikely to benefit from DL? Why?

Deep learning, yay or nay?

A piece of cake, elementary math…





It eats, a lot!