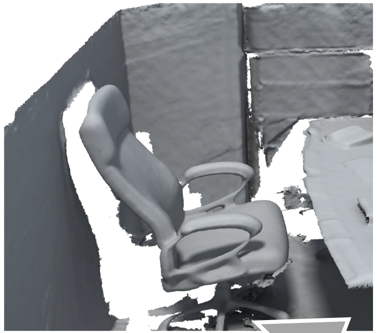
UC San Diego

Lecture 1: Introduction

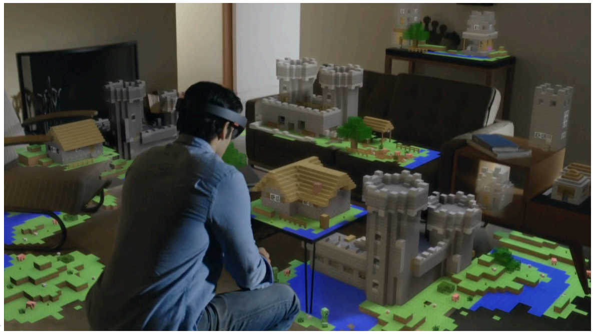
Instructor: Hao Su

Jan 9, 2018

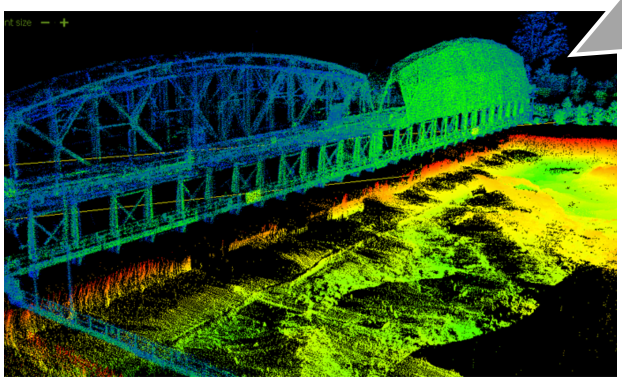
Welcome to CSE291-I!



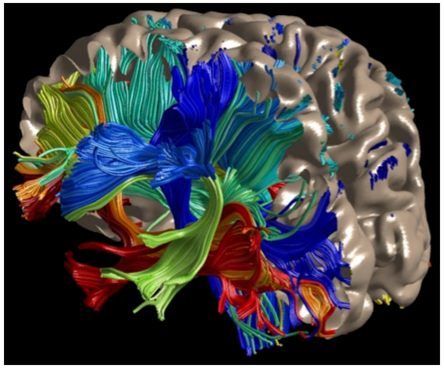
Robotics



Augmented Reality



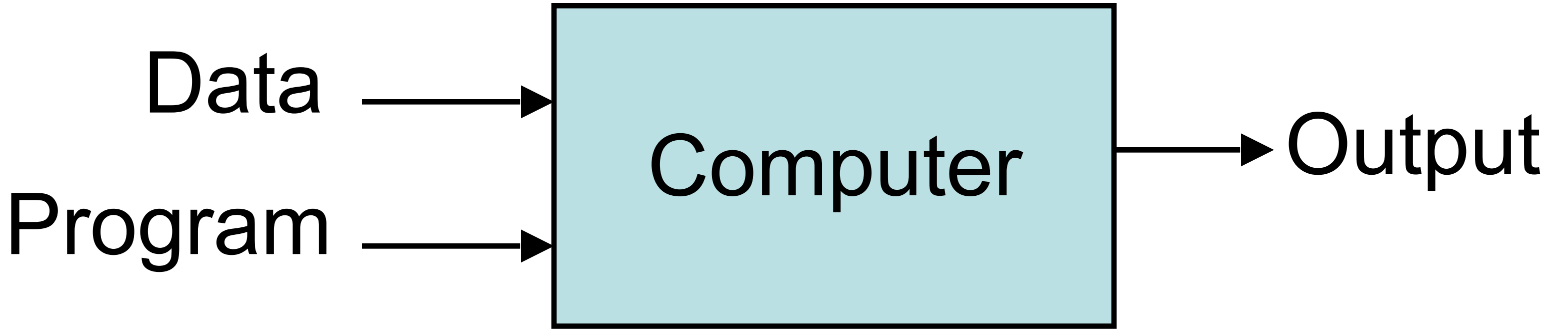
Autonomous driving



Medical Image Processing

Machine Learning

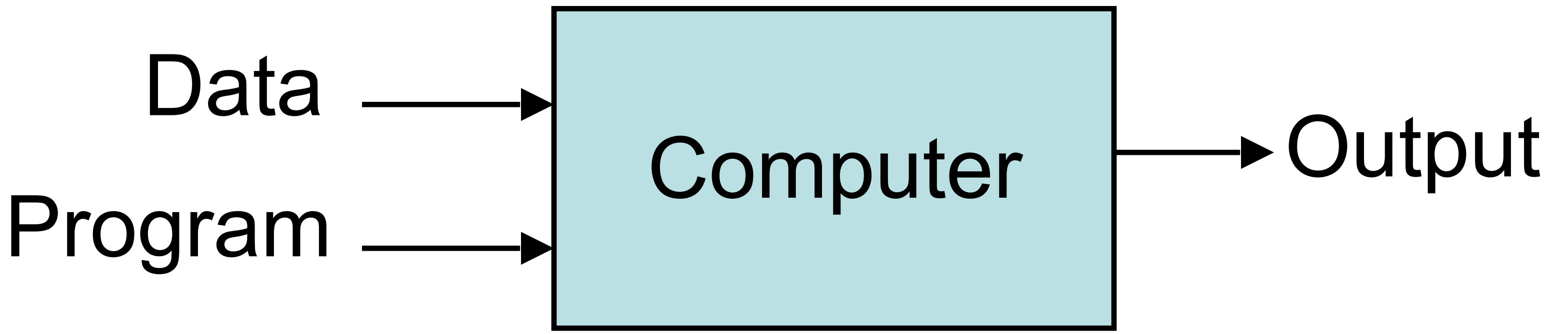
Traditional Programming



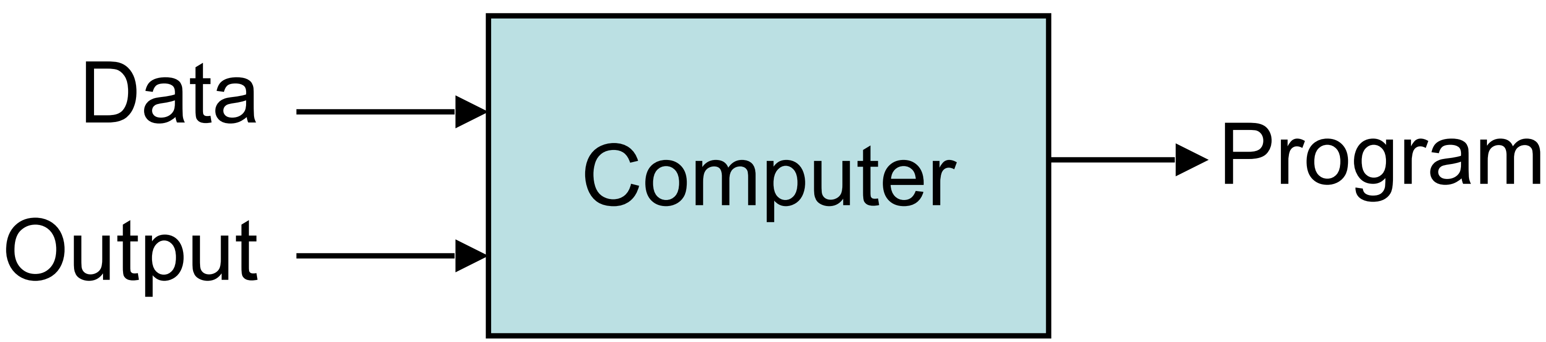
[Slide ack: P. Domingos]

Machine Learning

Traditional Programming



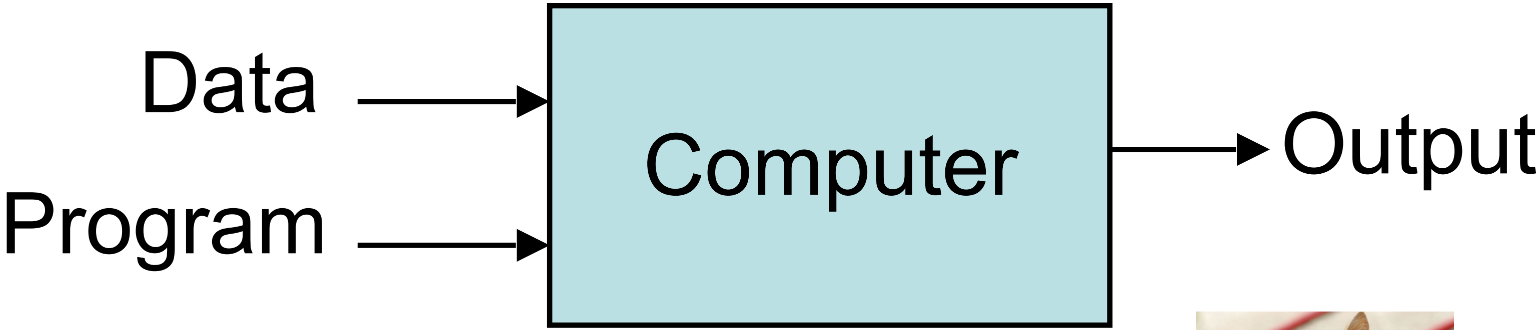
Machine Learning



[Slide ack: P. Domingos]

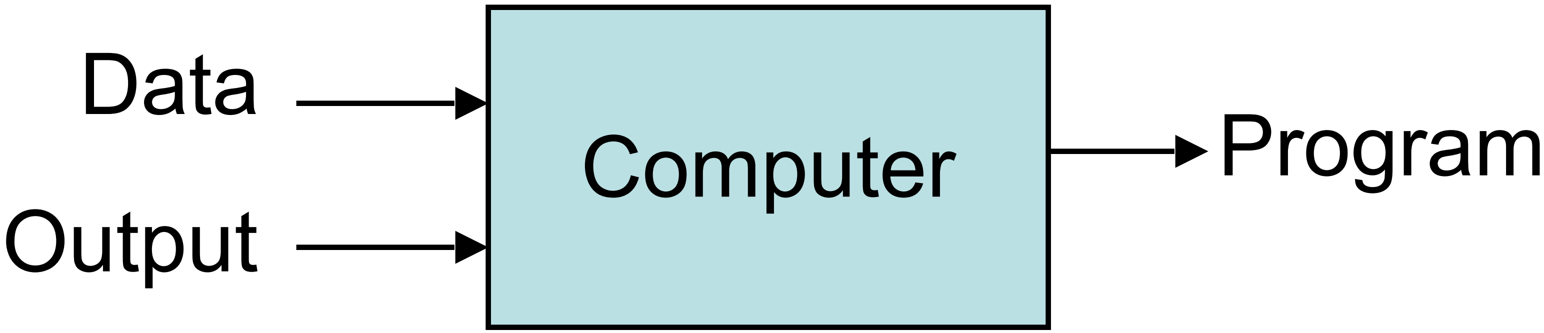
Machine Learning

Traditional Programming



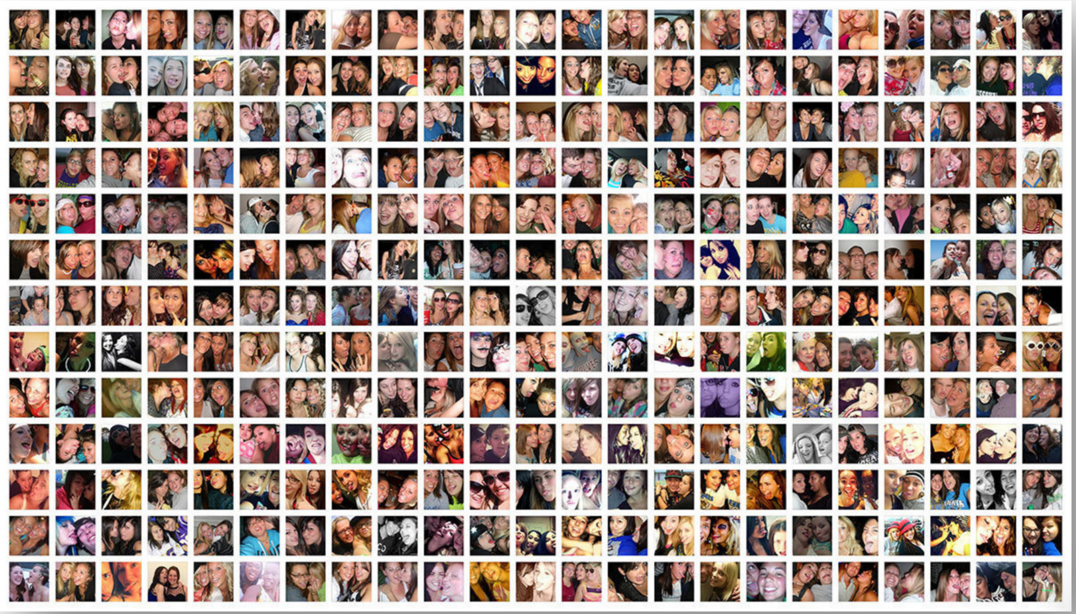
f()=cat
?

Machine Learning



[Slide ack: P. Domingos]

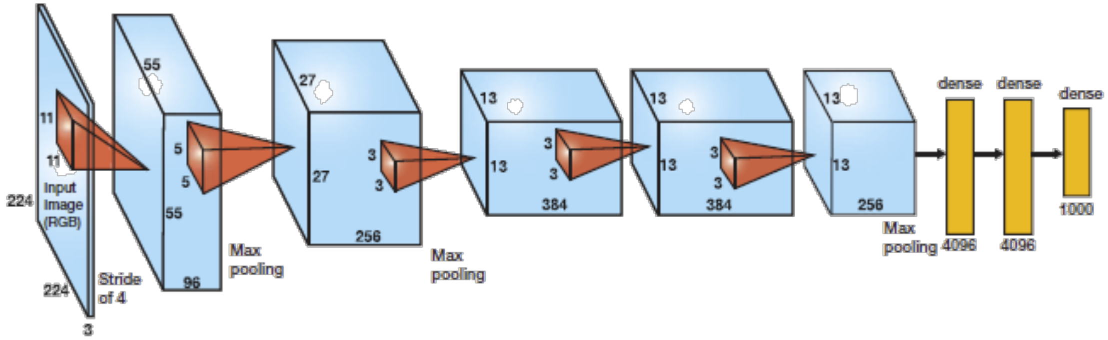
Made Possible By ....



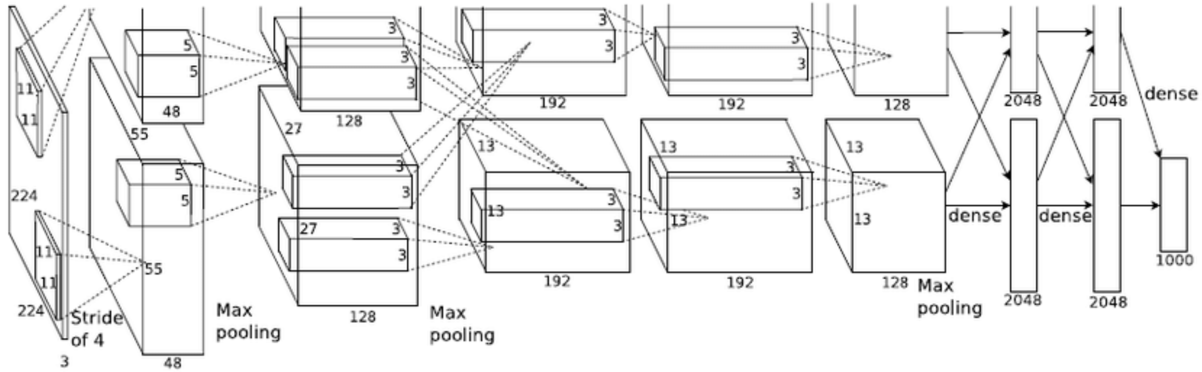
Lots of data

Lots of computing power

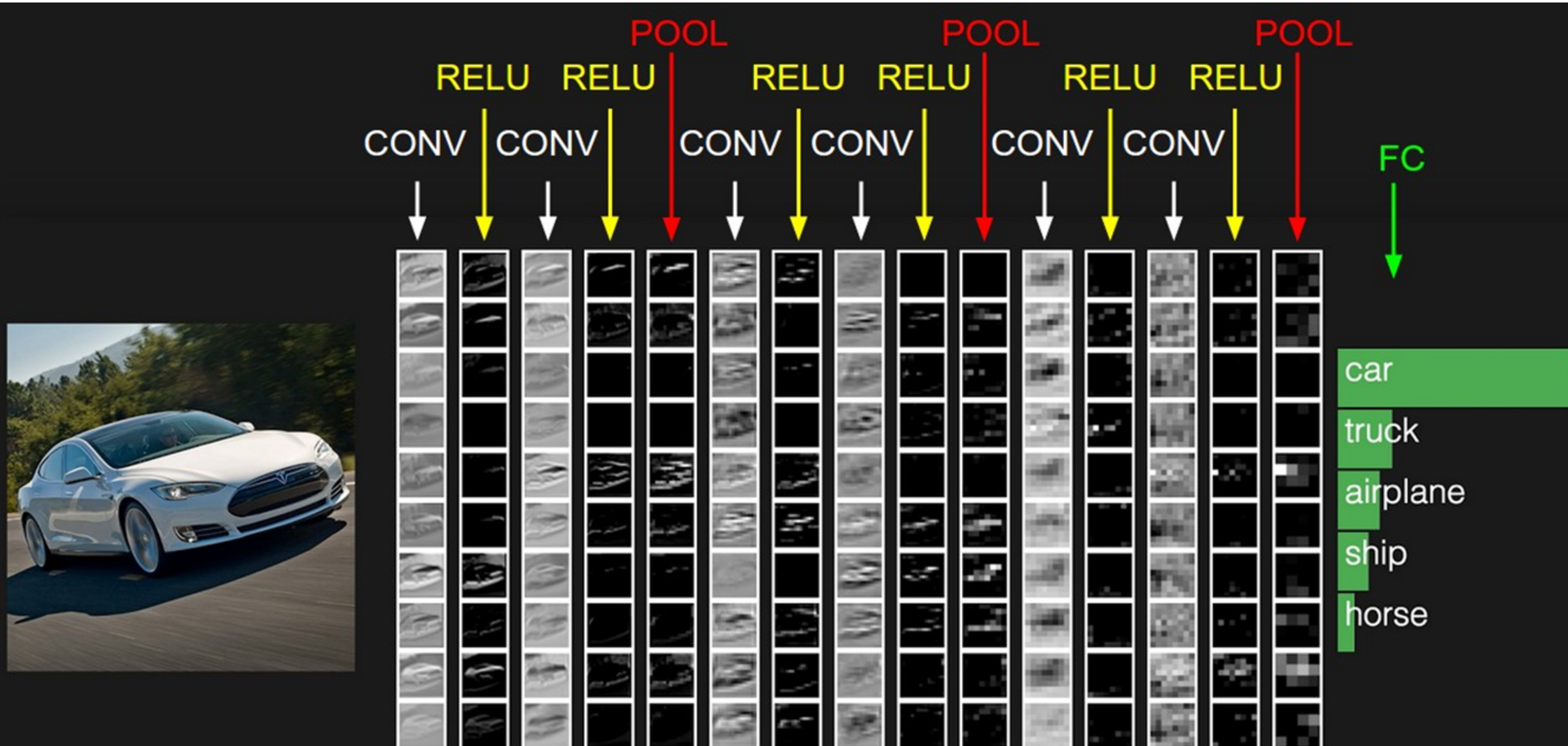
Many Successes:
Image Classification



Alexnet



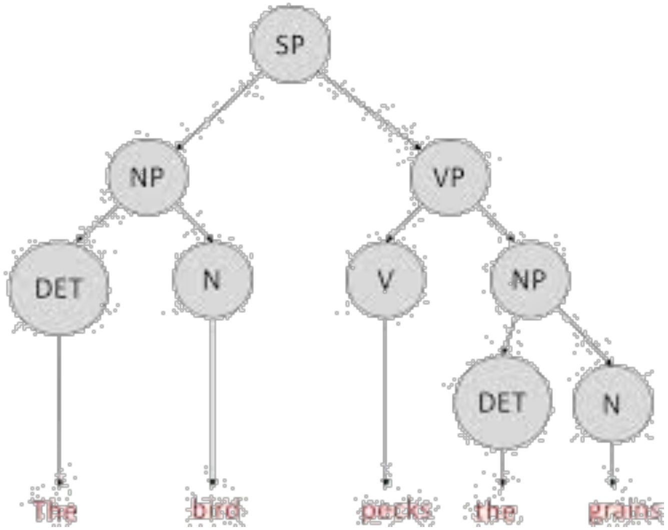
Stride of 4
Max pooling
Max pooling
Max pooling
dense
dense
dense
224
11
55
27
13
128
192
48
2048
1000



CONV
RELU
POOL
FC
car
truck
airplane
ship
horse

But Also …

- Speech understanding
- Natural language processing
- …



3D Perception is important for AI

Cosimo Alfredo Pina, "The domestic robots are getting closer"

3D Perception is important for AI

3D Perception is important for AI

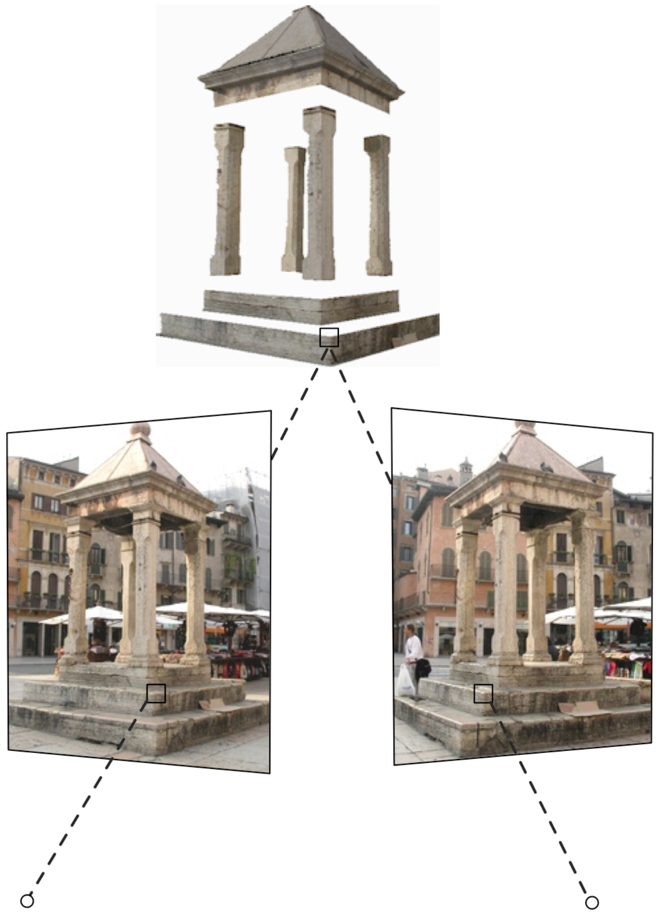
3D Perception is important for AI

3D Perception is important for AI

3D Perception requires "**Knowledge**" of 3D World

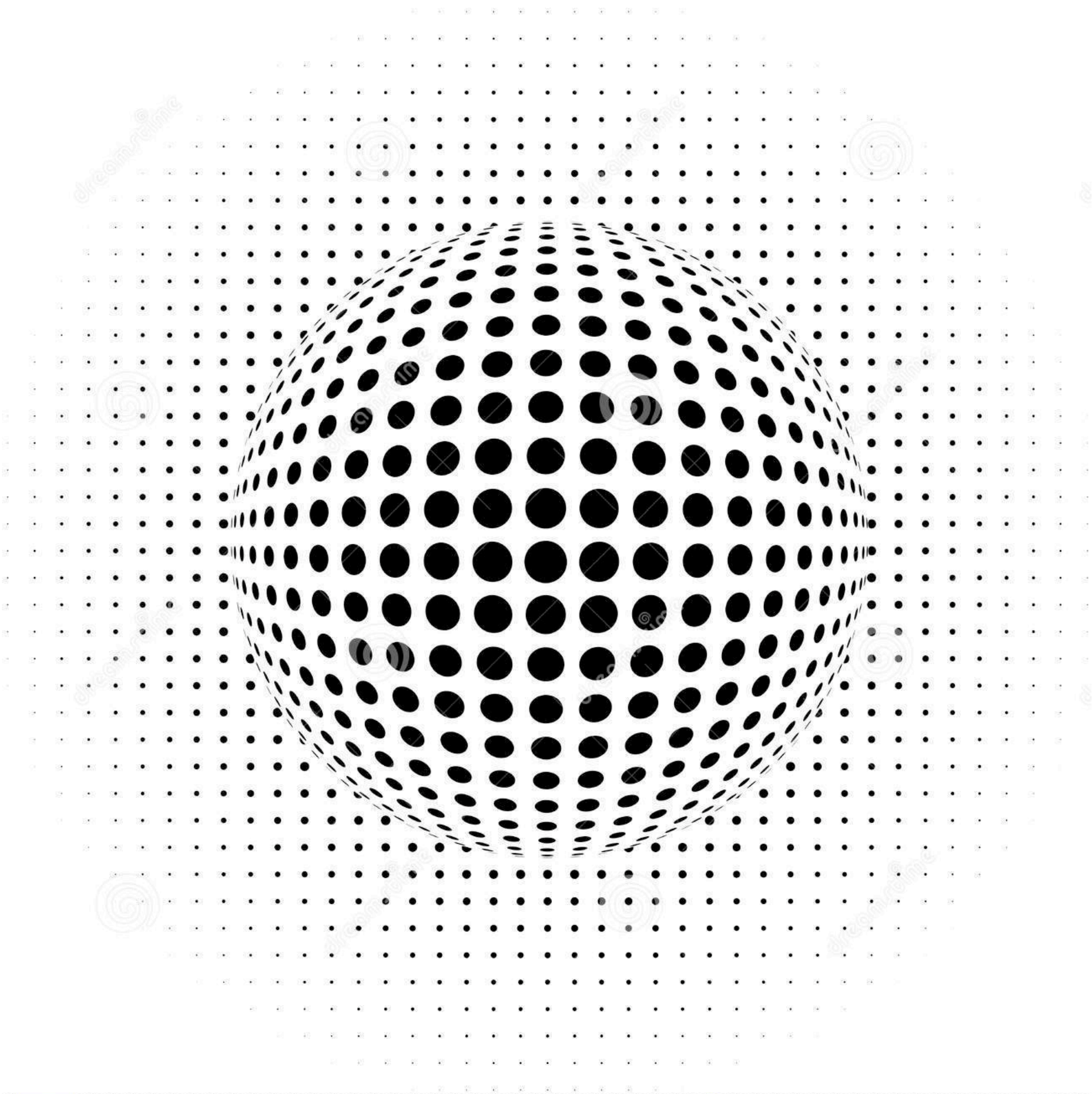
Traditional 3D Vision

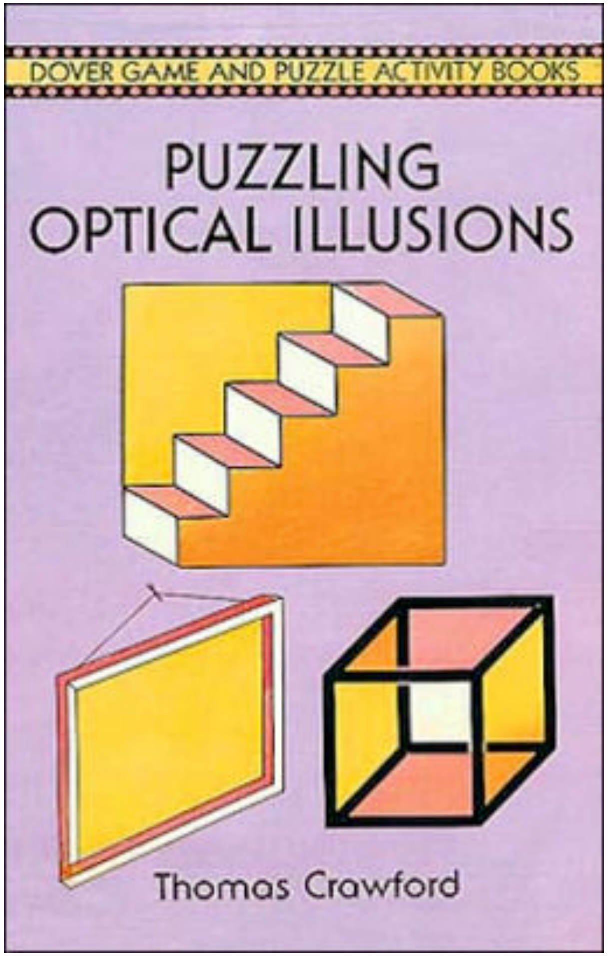
Multi-view Geometry: Physics based



3D Learning: Knowledge Based
Example I: Monocular Reconstruction

3D Learning: Knowledge Based
Example I: 3D Illusion





Visual Cues are Complicated

contrast

color

texture

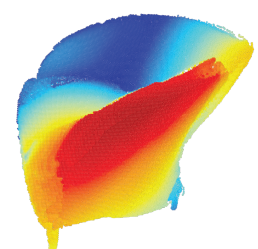
motion

symmetry

part

category-specific 3D knowledge

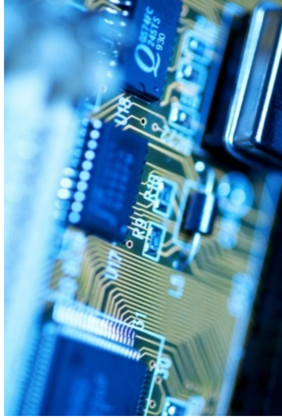
……



Acquire Knowledge of 3D World by **Learning**

A priori knowledge of the 3D world

Bridge Human and Computer Understanding



Human understanding and situational awareness comes from

- having **models** of the world
- **relating** current **sensory** inputs to past experience

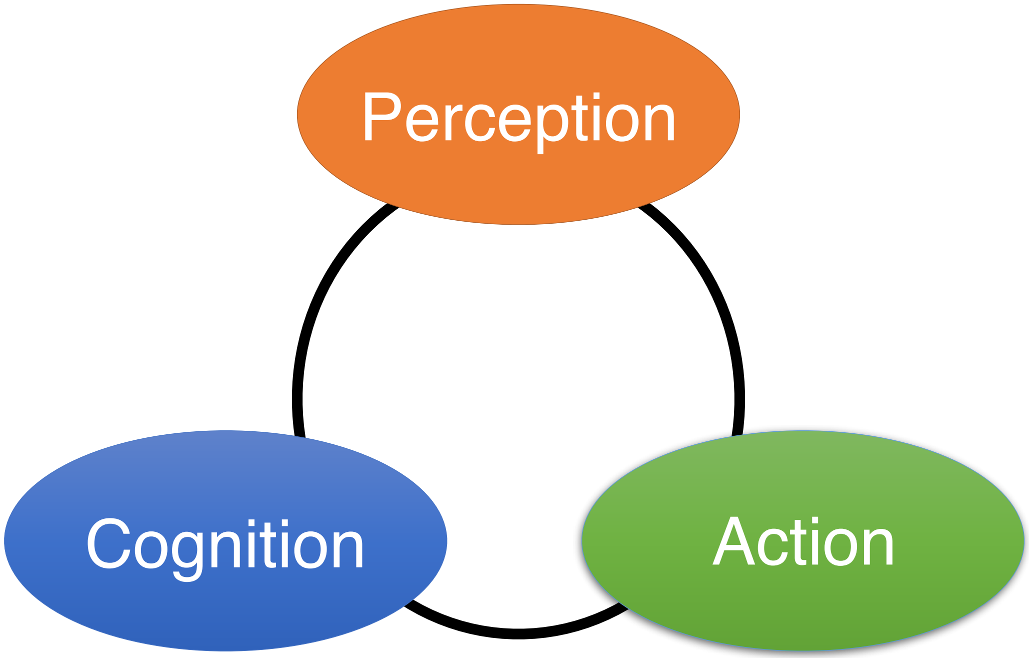
Both assistive and independent intelligent agents need to have a deep understanding of the physical world

AI Perspective of 3D Understanding

Discover the world

Understand the world

Transform the world

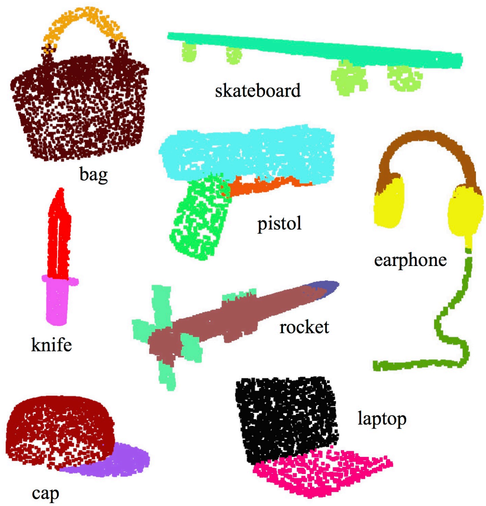


Towards **interaction** with the physical world,
3D is the key!

3D Learning Tasks

3D Analysis

Classification



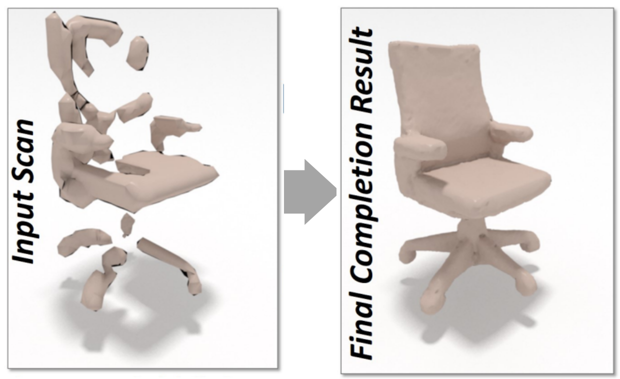
Segmentation (object/scene)

Correspondence

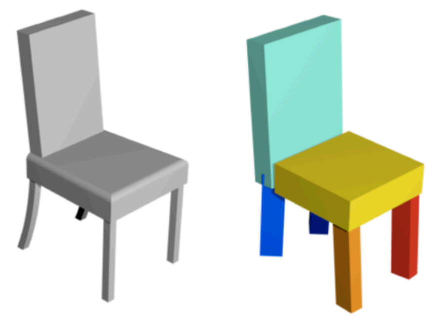
3D Learning Tasks

3D Synthesis

Monocular
3D reconstruction



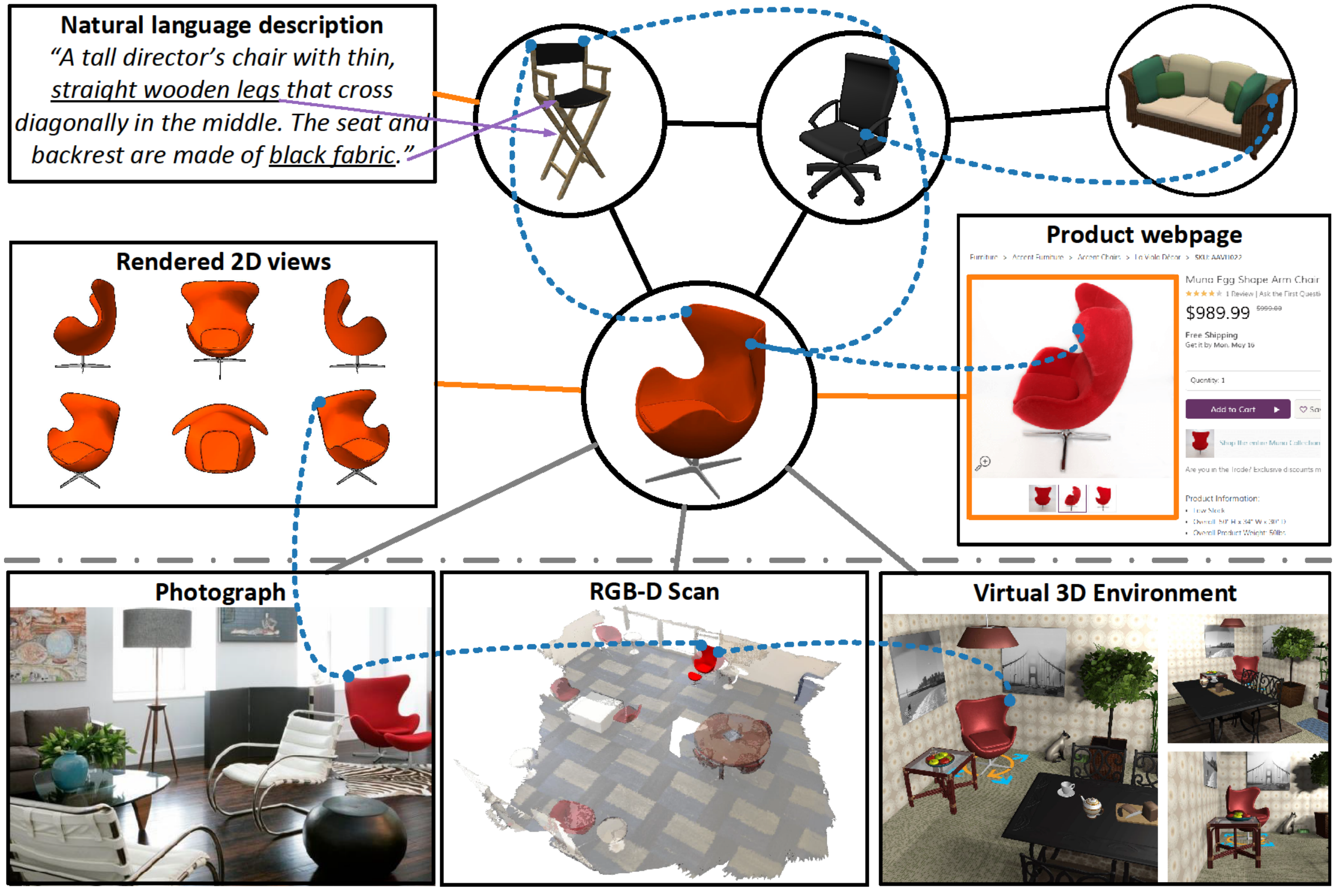
Shape completion



Shape modeling

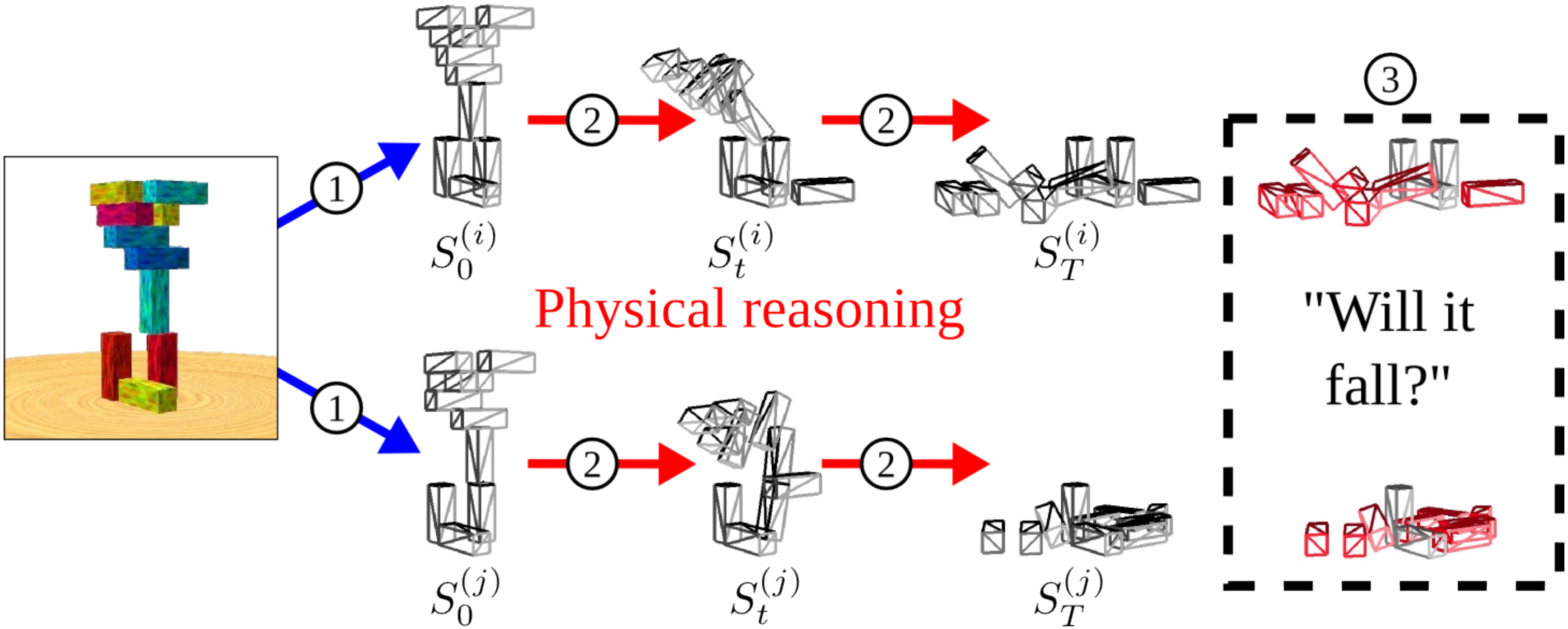
3D Learning Tasks

3D-based Knowledge Transportation

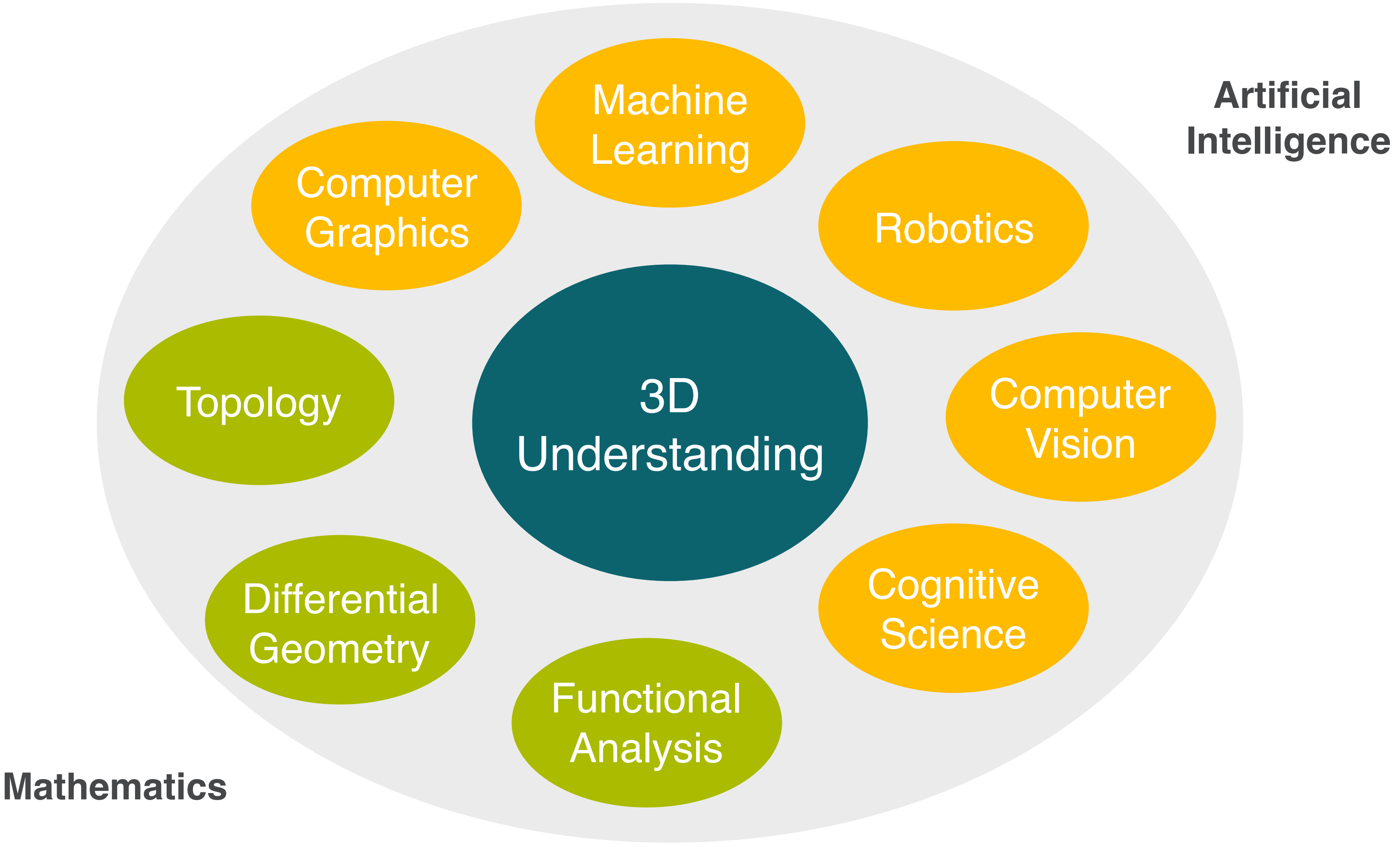


3D Learning Tasks

Intuitive Physics based on 3D Understanding



Machine Learning on 3D: A New Rising Field



Related Courses @ UCSD

- CSE258: Machine Learning and Recommender Systems
- CSE253: Neural Networks
- CSE250A: Probabilistic Models
- CSE250B: Learning Algorithms

- Let me know more!

Syllabus

- Introduction to Machine Learning and Deep Learning
- Shape Representation, Geometry Parameterization, Machine Learning on Extrinsic Geometry
- Simulation for Learning
- Graph Laplacian and Intrinsic Geometry
- Map Networks
- Topological Data Analysis

Syllabus (Week 1-2)

Introduction to Machine **Learning** and Deep Learning

Machine learning paradigm

- Optimization, Gradient Descent, Back Propagation
- Linear Classifier, k-NN Classifier
- Multi-layer Perceptron, Convolutional Neural Network (CNN)
- Why deep learning is more effective?
- Non-convex Optimization Issues in Deep Learning

Syllabus (Week 3-4)

Shape **Representation**, Geometry Parameterization, Machine Learning on Extrinsic Geometry

- Multi-view Representation, the Novel-view Synthesis Problem
- Volumetric Representation, 3D CNN
- Parametric Representation of Surfaces, Differential Geometry
- Point Cloud Representation, EMD, PointNet
- Shape Grammar, Procedural Modeling

Syllabus (Week 5)

Simulation for Learning

- Render for CNN
- Domain Adaptation, **GAN**
- Intuitive Physics

Syllabus (Week 6-7)

Graph Laplacian and Intrinsic Geometry

- **Spectral Graph Theory**
- Extrinsic vs **Intrinsic Geometry**
- Laplacian-Bertrami operator
- Heat Kernel and Wave Kernel signatures
- Graph CNNs, Deep Learning on Manifolds

Syllabus (Week 8-9)

Map Networks (Analysis of Shape Collection)

- Rigid and Non-rigid Shape **Registration**
- Functional Map, Canonical Correlation Analysis
- Cycle Consistency
- Functional Map Network, Deep Functional Map Network
- Unsupervised Learning by Map Networks, Cycle GAN
- Map Networks on Shape Parts
- Map Networks across Data Domains (Image Captioning, Joint image-shape Networks)

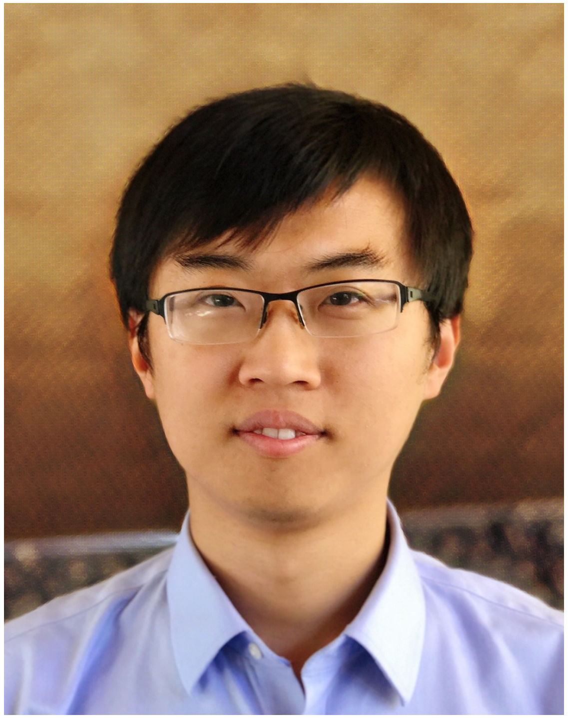
Syllabus (Week 10)

Topological Data Analysis

- **Topology** Review, Complexes
- Homology Groups, Persistent Homology

Who we are?

Instructor: Hao Su



Teaching Assistant: Vignesh Gokul

Logistics

Course website announced: Link

- Related materials
- Announcements
- Lecture Slides
- Office hour and location

Questions? Piazza

Philosophy

- State-of-the-art
  - We hope that you will find opportunities to publish by taking this course
- Practical
  - Bring you first-hand experience in 3D learning by programming assignments
- Foundational
  - Introduce knowledge of broad related fields in a friendly language

Grading policy

Grading (tentative)

- In-class quiz (~10 times): 10%
- Homework (4 assignments): 50%
- Course project presentation: 20%
- Course project writeup: 20%
- Best paper award (Top 3): 10% bonus
- There will not be a final exam

Late days: in total 7 days

Pre-requisite

- Try to be as self-contained as possible
- Proficiency in Python and Matlab
- Calculus, Linear Algebra
- Machine learning
  - Classification
  - Optimization

Introduction to Machine Learning and Deep Learning

Common Task Framework (1980's)

Crucial methodology driving predictive modeling's success

An instance has the following ingredients:

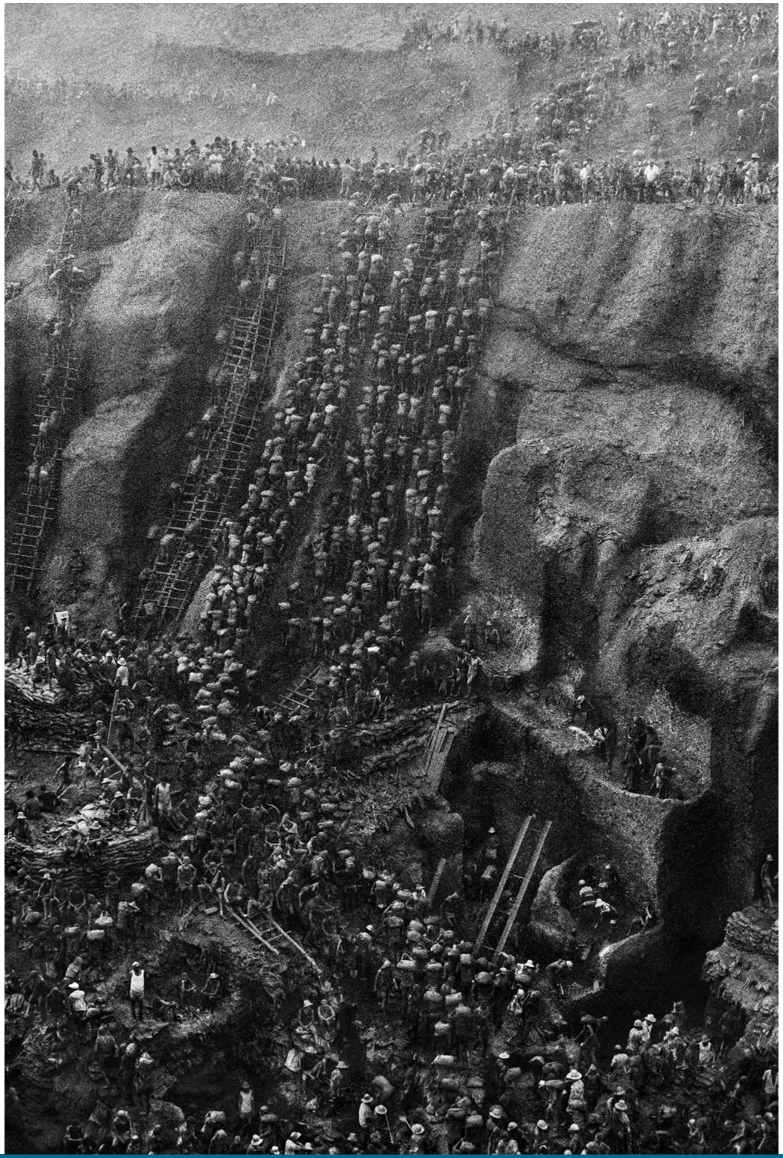
- Training **dataset,** a list of observations — (feature, label) pairs
- Competitors whose goal is to learn a predictor from the training set (**learner**)
- Scoring **referee**

See Mark Liberman's description (Liberman, 2009).

CTF Lifestyle - 1

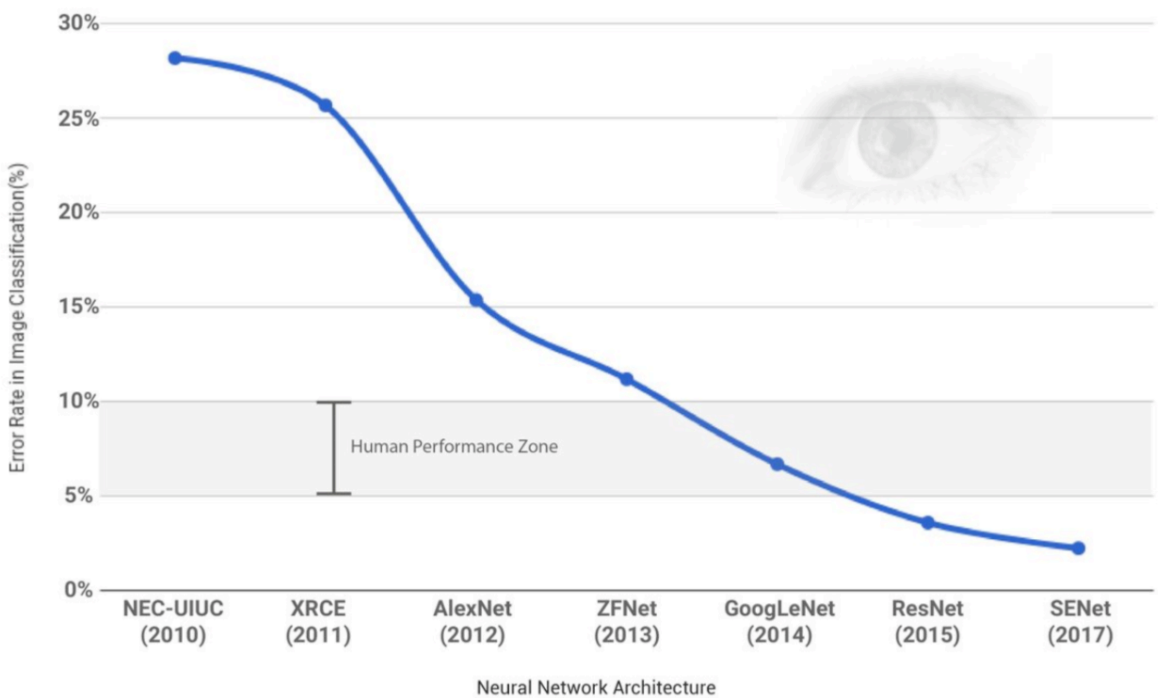
1. Researchers set up local copies of Challenge
   - Data – Training, Test carved out of public dataset
   - Scoring – same as challenge scoring rule
2. Researcher's job: *'tuning models'*
   - Think up a family of model variations – *'tweak's*
   - Run a full *'experiment'* – suite of tweaks – *'grid'*
   - Score each tweak
   - Submit best-scoring result to central authority
3. Successful researchers perpetually motivated by *Game-ification*: tweaking, scoring, winning.
4. Researchers who tweak more often, win more often!.
5. If easier to implement tweaks and faster to evaluate them, more likely to win!.

CTF Lifestyle - 2



Instance of Common Task Framework

- ImageNet (subset):
  - 1.2 million training images
  - 100,000 test images
  - 1000 classes
- ImageNet large-scale visual recognition Challenge



Formally,

Dataset: $\mathfrak{T} = \{(x_i, y_i)\}_{i=1}^{n}$

Learner: $y = f(x;\theta)$ e.g. $y = w^T x$

Referee (loss function): $L(y, y')$

e.g. $L(y, y') = (y - y')^2$

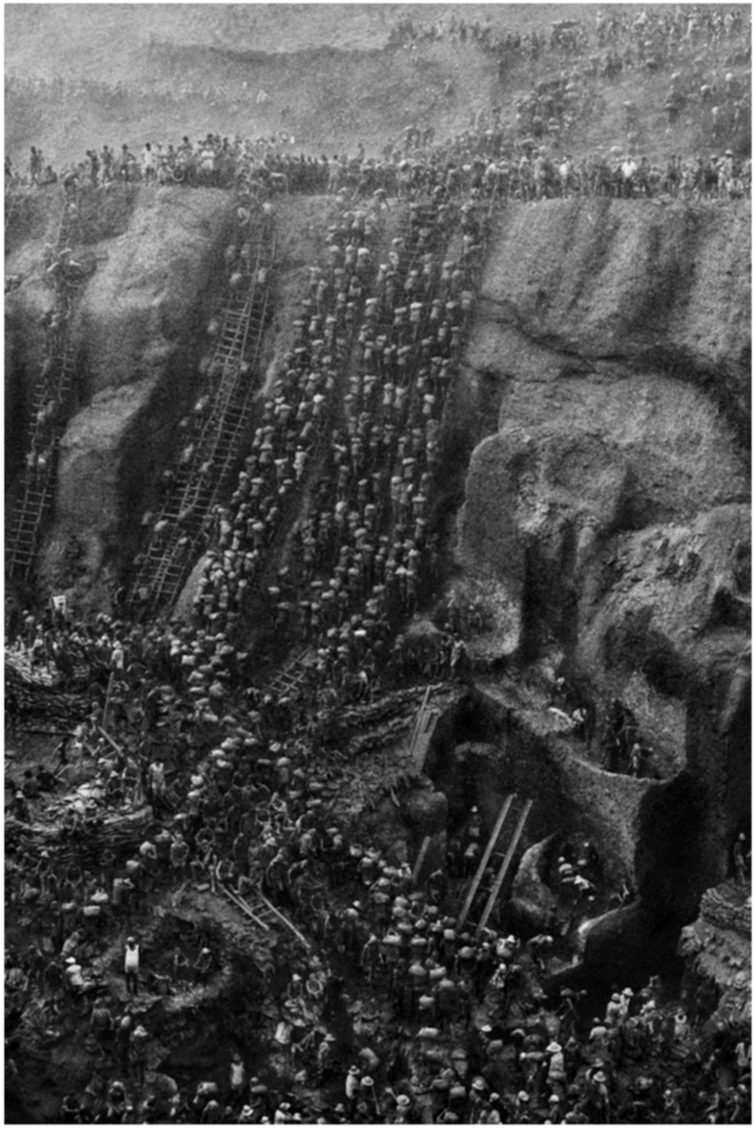
Learning as an Optimization Problem

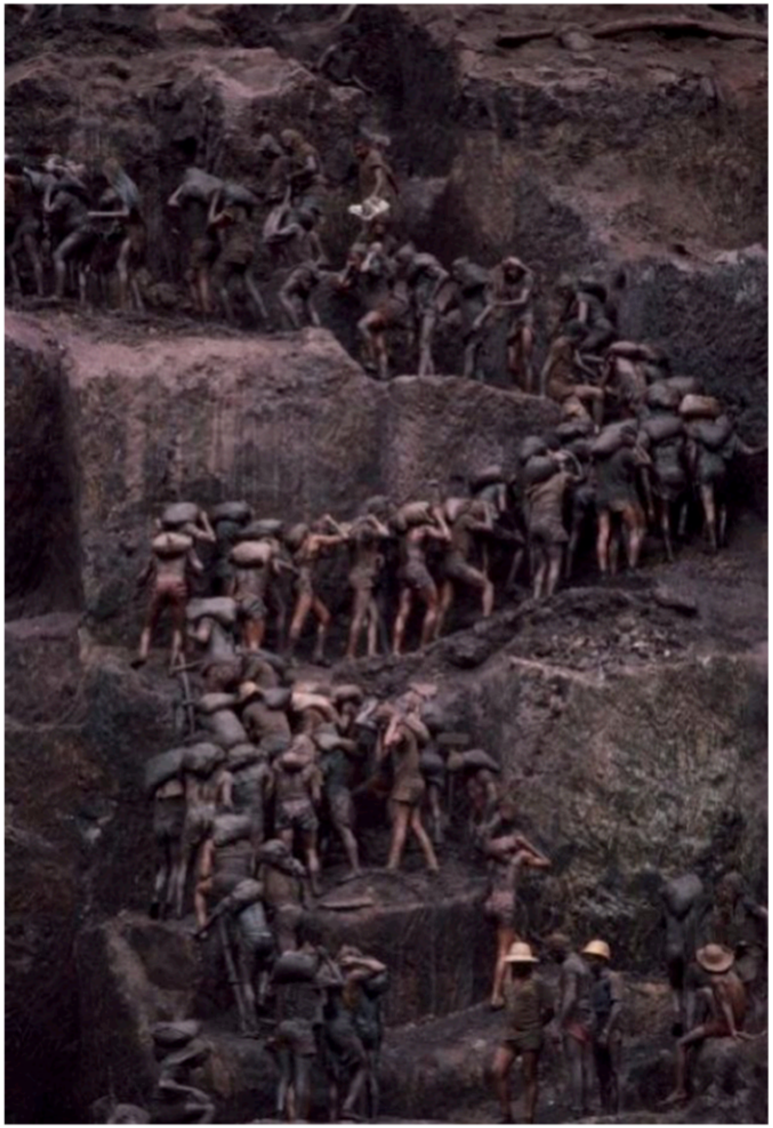
$$\underset{\theta}{\text{minimize}} \sum_i L(f(x_i;\theta), y_i)$$

In the context of machine learning, “learning” means to the search for the optimal prediction rule. e.g.,

- In classification trees, we search for the tree structure, decision variable at each node, and split value
- In linear regressor, we search for the optimal weights by convex opt

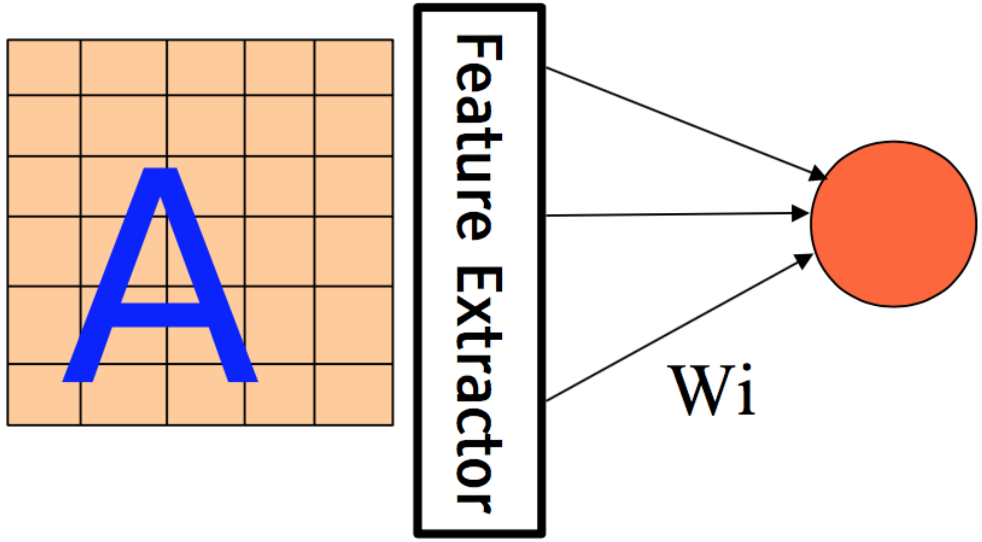
CTF Lifestyle





FROM PERCEPTRON TO DEEP LEARNING

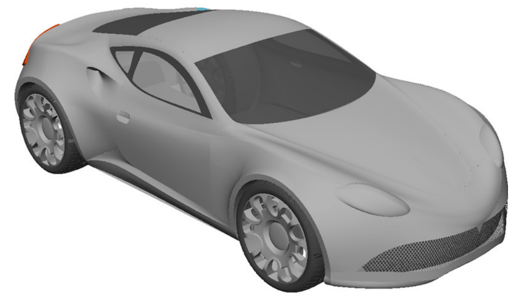
Perceptron



$$y = sign\left(\sum_{i=1}^{N} W_i F_i(X) + b\right)$$

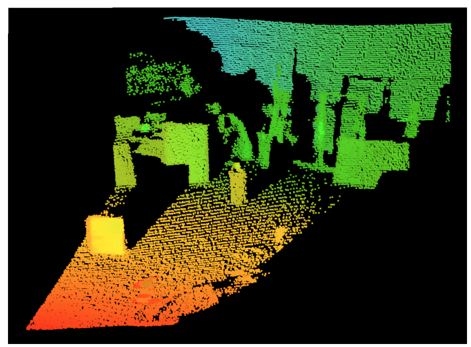
Imag

Video

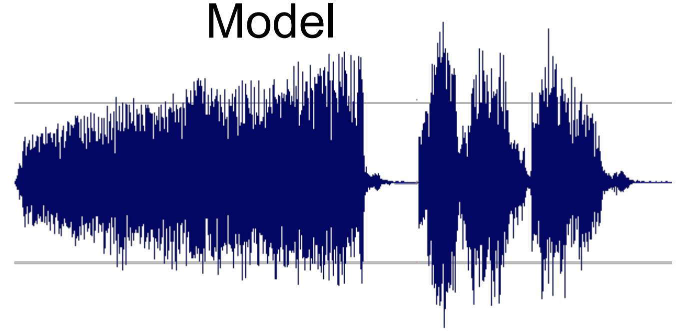


3D CAD Model

Thermal Infrared



Depth Scan



Audio

Can we automatically learn “good” feature representations?

- The traditional model of pattern recognition (since the late 50's)
  - Fixed/engineered features (or fixed kernel) + trainable classifier

→ hand-crafted Feature Extractor → "Simple" Trainable Classifier →

- End-to-end learning / Feature learning / Deep learning
  - Trainable features (or kernel) + trainable classifier

→ Trainable Feature Extractor → Trainable Classifier →

From Y. LeCun's Slides

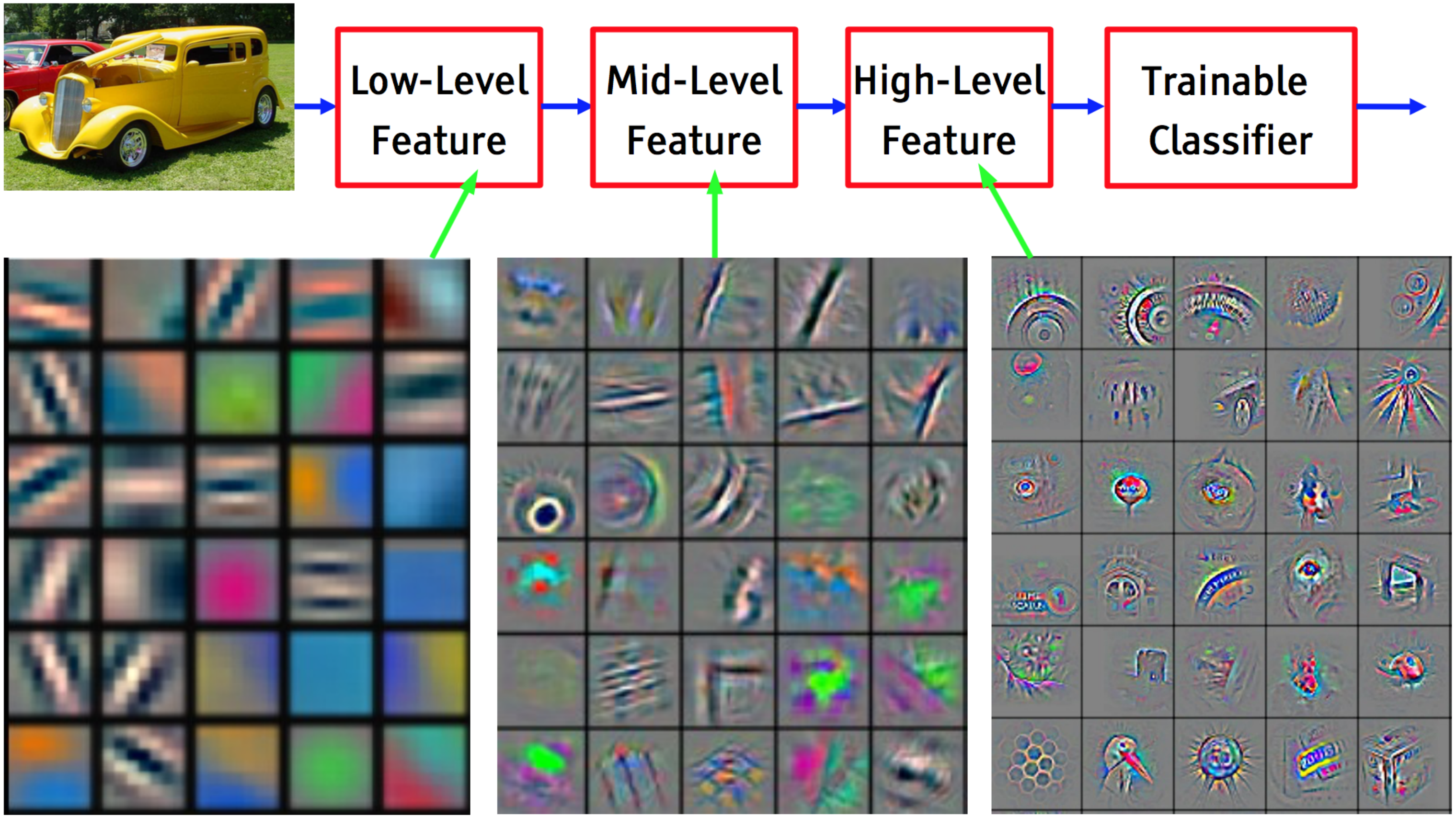
Traditional Pattern Recognition: Fixed/Handcrafted Feature Extractor

Mainstream Modern Pattern Recognition: Unsupervised mid-level features

Deep Learning: Representations are hierarchical and trained

From Y. LeCun's Slides

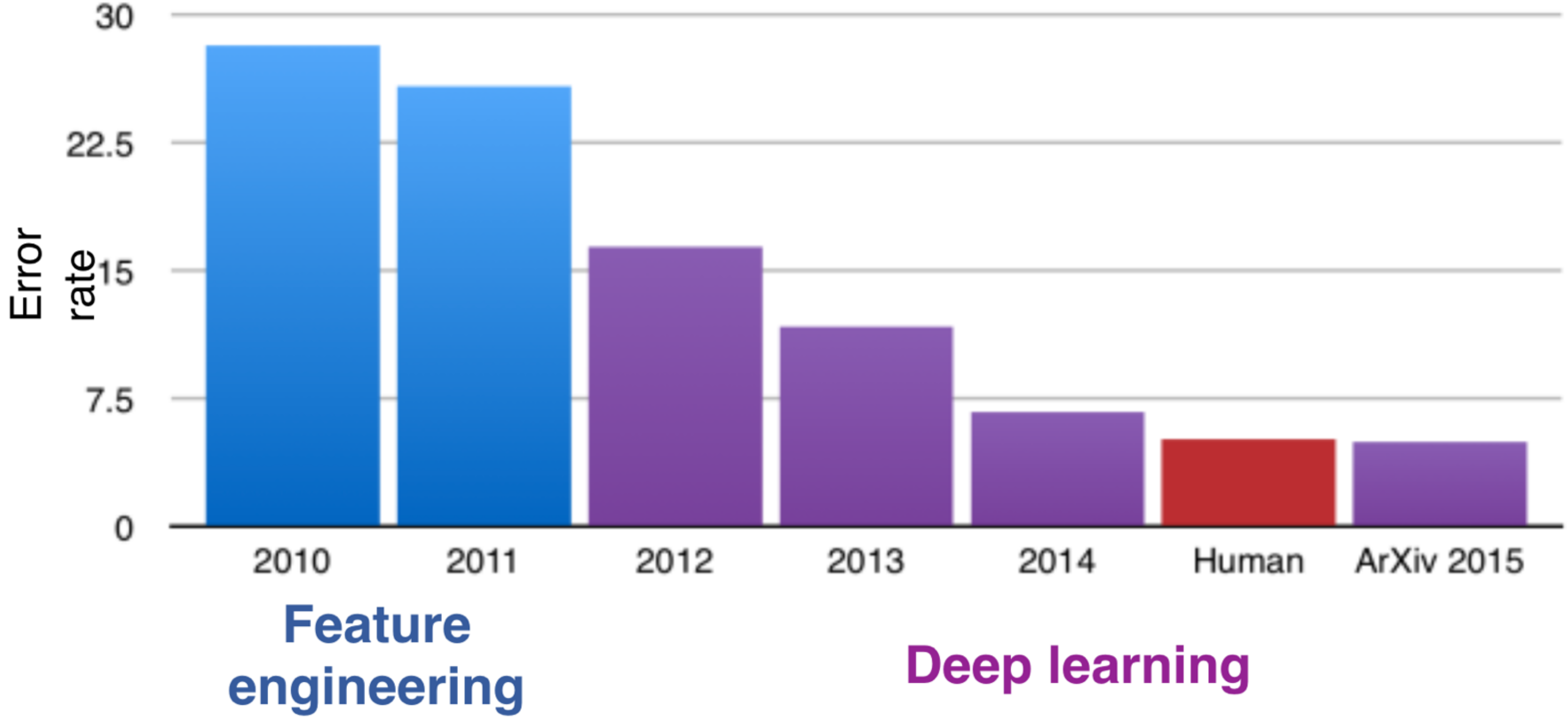
- It's deep if it has more than one stage of non-linear feature transformation



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

From Y. LeCun's Slides

ImageNet 1000 class image classification accuracy



A SIMPLE NEURAL NETWORK

A Simple Neural Network

Use recent three days' average temperature to predict tomorrow's average temperature.

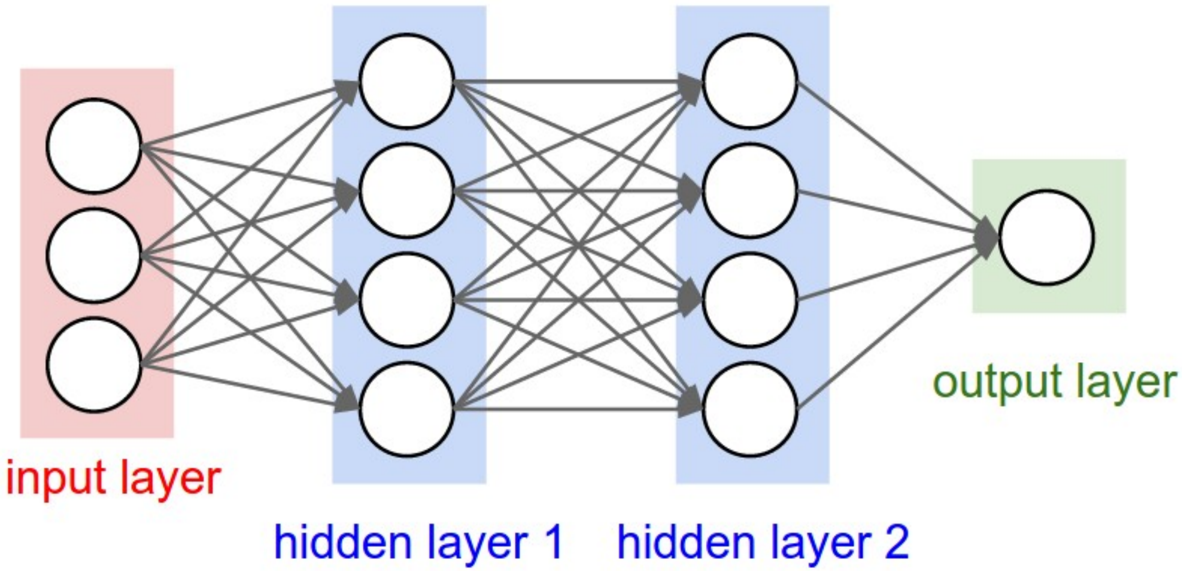
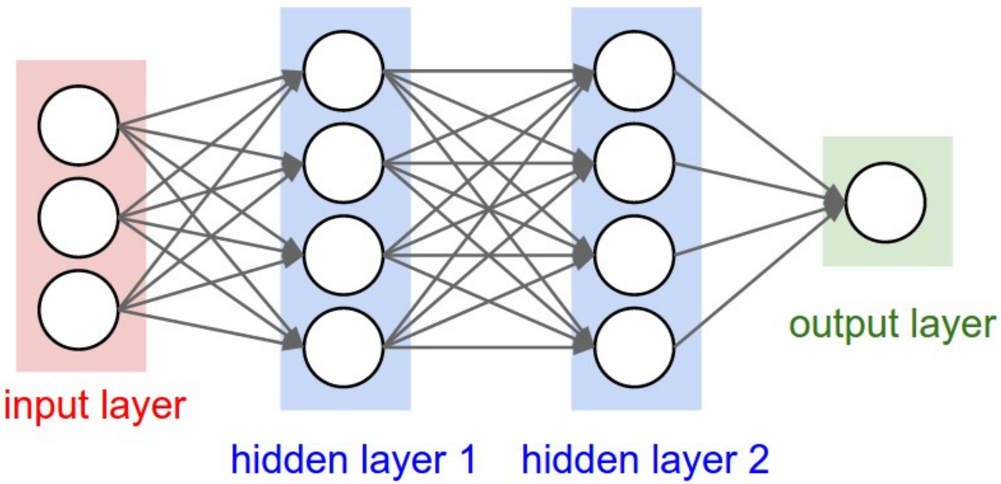
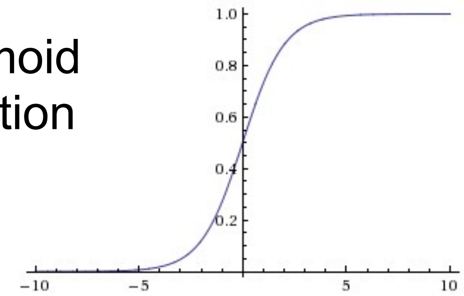


Image from CS231N

A Simple Neural Network



Sigmoid function

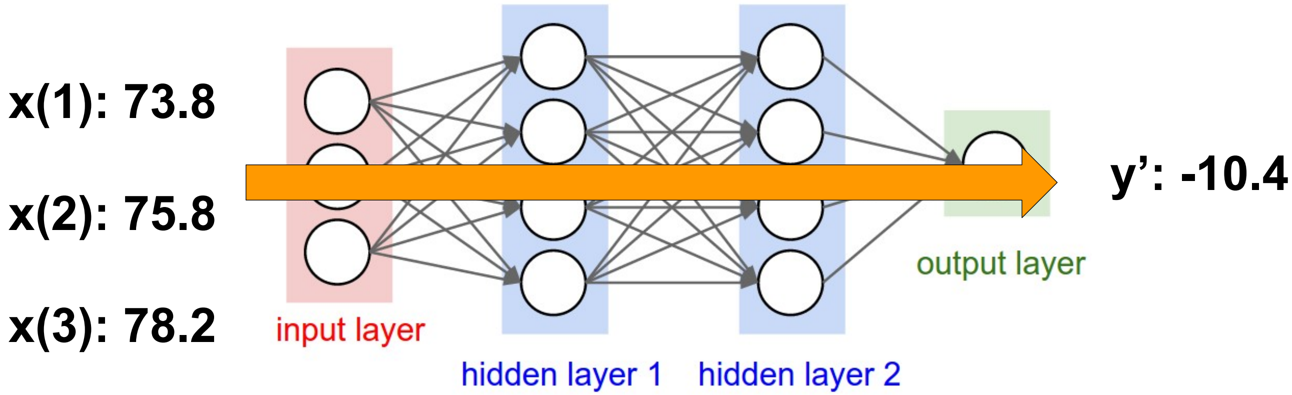


W1, b1, W2, b2, W3, b3 are network parameters that need to be learned.

```
# forward-pass of a 3-layer neural network:
f = lambda x: 1.0/(1.0 + np.exp(-x)) # activation function (use sigmoid)
x = np.random.randn(3, 1) # random input vector of three numbers (3x1)
h1 = f(np.dot(W1, x) + b1) # calculate first hidden layer activations (4x1)
h2 = f(np.dot(W2, h1) + b2) # calculate second hidden layer activations (4x1)
out = np.dot(W3, h2) + b3 # output neuron (1x1)
```

From CS231N

Neural Network: Forward Pass



$$y' = W_3 f(W_2 f(W_1 x + b_1) + b_2) + b_3)$$

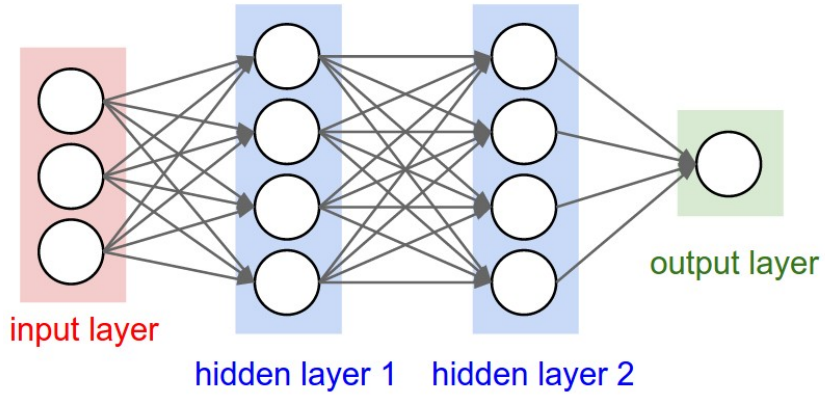
From CS231N

Neural Network: Backward Pass

73.8

75.8

78.2



Prediction: -10.4

Ground truth: 80.8

L2 error = (80.8 - (-10.4))^2

Update Network Parameters

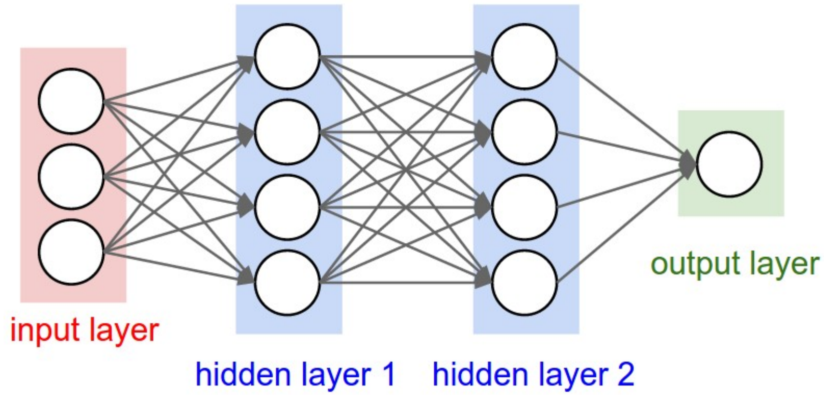
From CS231N

Neural Network: Backward Pass

73.8

75.8

78.2



Prediction: -10.4

Ground truth: 80.8

L2 error = (80.8 - (-10.4))^2

Update Network Parameters

Minimize: $L(x, y; W, b) = \sum_{i=1}^{N} (W_3 f(W_2 f(W_1 x_i + b_1) + b_2) + b_3) - y_i)^2$

Given N training pairs: $\{x_i, y_i\}_{i=1}^{N}$

Neural Network: Backward Pass

Minimize: $L(x, y; W, b) = \sum_{i=1}^{N} (W_3 f(W_2 f(W_1 x_i + b_1) + b_2) + b_3) - y_i)^2$

Given N training pairs: $\{x_i, y_i\}_{i=1}^{N}$

Non-convex optimization :(

Neural Network: Backward Pass

Minimize: $L(x, y; W, b) = \sum_{i=1}^{N} (W_3 f(W_2 f(W_1 x_i + b_1) + b_2) + b_3) - y_i)^2$

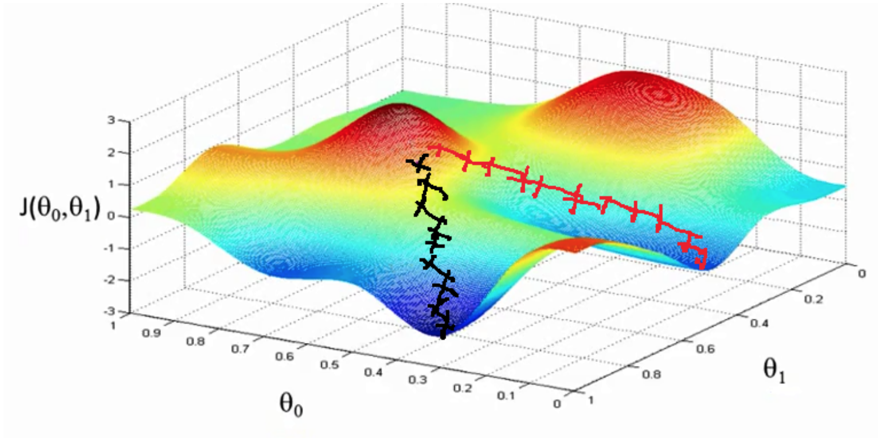
Given N training pairs: $\{x_i, y_i\}_{i=1}^{N}$

Non-convex optimization :(

Use gradient descent!

Parameter update example:

$$W_3 = W_3 - \eta \frac{\partial L}{\partial W_3}$$



Neural Network: Backward Pass

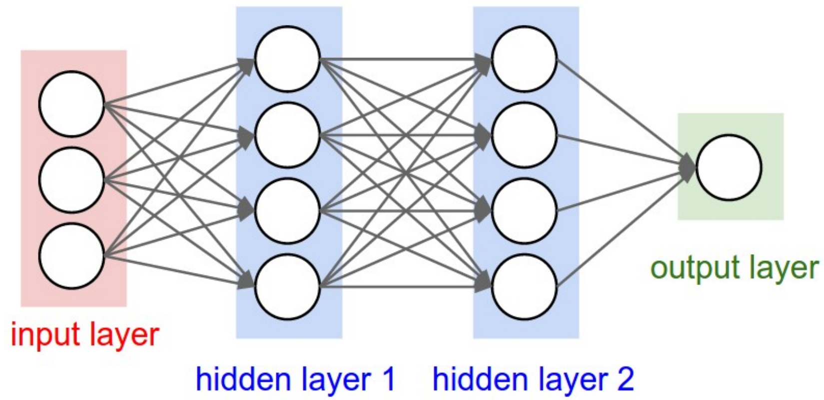
Minimize: $L(x,y;W,b)=\sum_{i=1}^{N}(W_3f(W_2f(W_1x_i+b_1)+b_2)+b_3)-y_i)^2$

$$W_3 = W_3 - \eta\frac{\partial L}{\partial W_3}$$

Let $x_i^{(3)} = f(W_2f(W_1x_i+b_1)+b_2)$, cached after the forward pass,
What's the prediction?
$\hat{y}_i = W_3x_i^{(3)} + b_3$
Then $L = \sum L_i = \sum_i(\hat{y}_i - y_i)^2$
$\frac{\partial L_i}{\partial W_3} = \frac{\partial L}{\partial \hat{y}_i}\frac{\partial \hat{y}_i}{\partial W_3} = 2(\hat{y}_i - y_i)x_i^{(3)}$

Back-propagation is equivalent to gradient descent by chain rule

A Simple Neural Network



Model: Multi-Layer Perceptron (MLP) $y' = W_3 f(W_2 f(W_1 x + b_1) + b_2) + b_3)$

Loss function: L2 loss $l(y, y') = (y - y')^2$

Optimization: Gradient descent $W = W - \eta \frac{\partial L}{\partial W}$