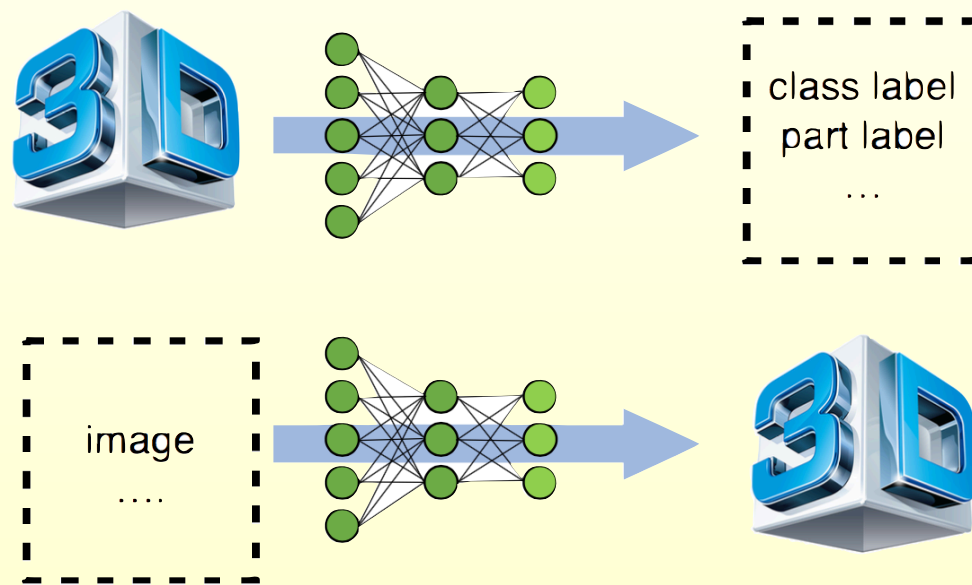


CS233: Geometric and Topological Data Analysis

Deep Learning Intro
3D Deep Learning
Multi-view CNNs
Volumetric CNNs

Kaichun Mo
May 16, 2018



Slides Credits: Hao Su, Charles Qi, Leo Guibas

Agenda

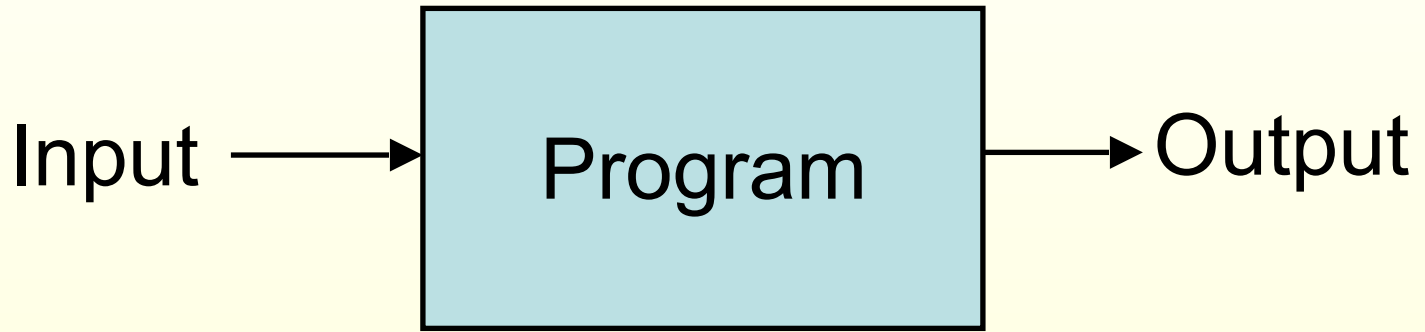
- ◆ Today
 - ◆ Deep Learning Intro
 - ◆ 3D Deep Learning
 - ◆ Multi-view CNNs
 - ◆ Volumetric CNNs
- ◆ Later lectures
 - ◆ Point-cloud NNs (May 21)
 - ◆ Graph CNNs (May 24)

Agenda

- ◆ Today
 - ◆ Deep Learning Intro
 - ◆ 3D Deep Learning
 - ◆ Multi-view CNNs
 - ◆ Volumetric CNNs
- ◆ Later lectures
 - ◆ Point-cloud NNs (May 21)
 - ◆ Graph CNNs (May 24)

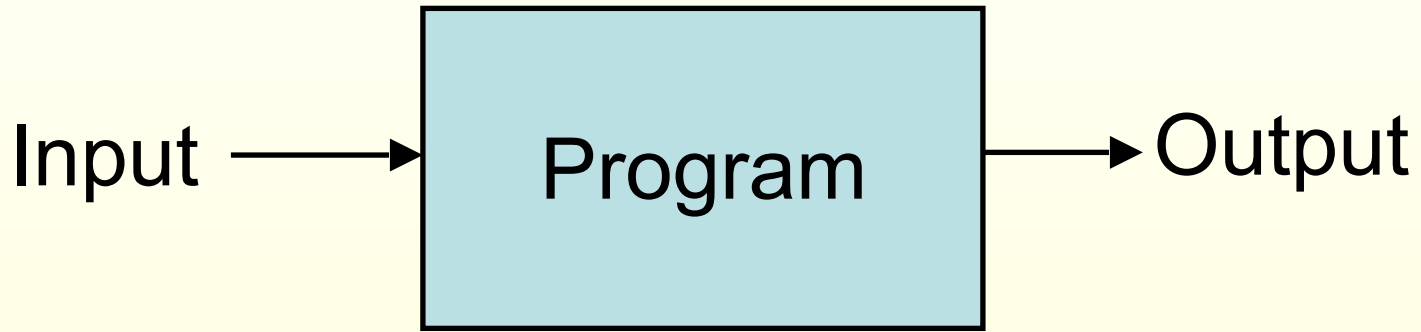
Machine Learning

◆ Traditional Programming

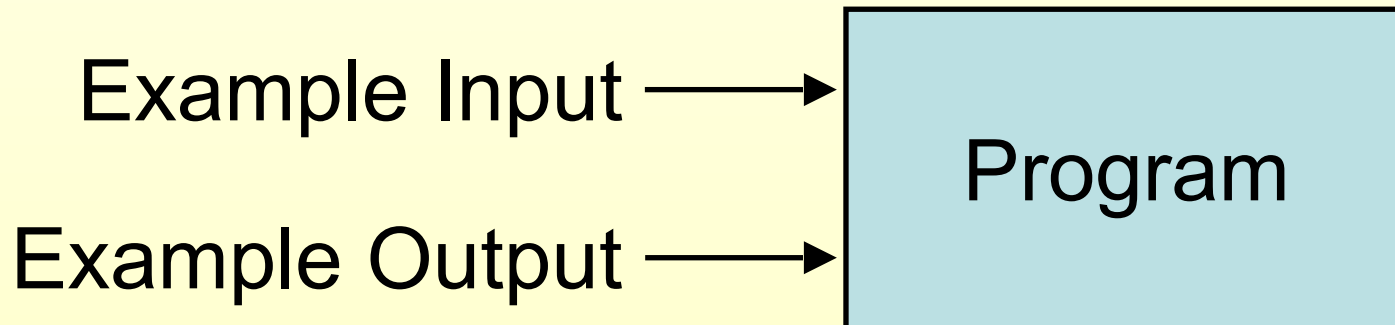


Machine Learning

◆ Traditional Programming



◆ Machine Learning



Machine Learning

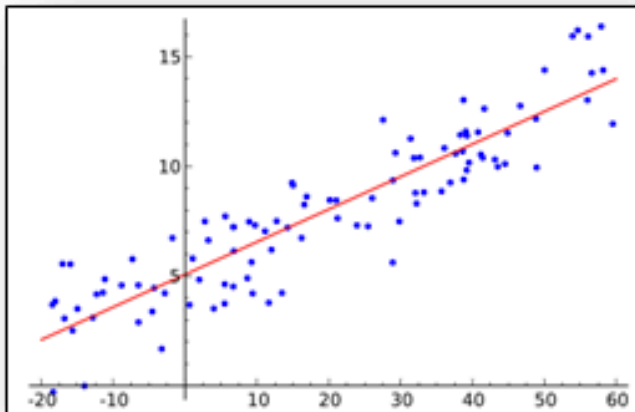
Parametrized by many
learnable parameters

$f(\$

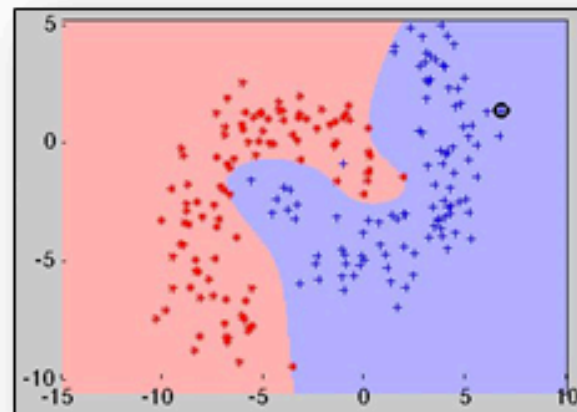


$) = \text{cat}$

Model Fitting



Regression

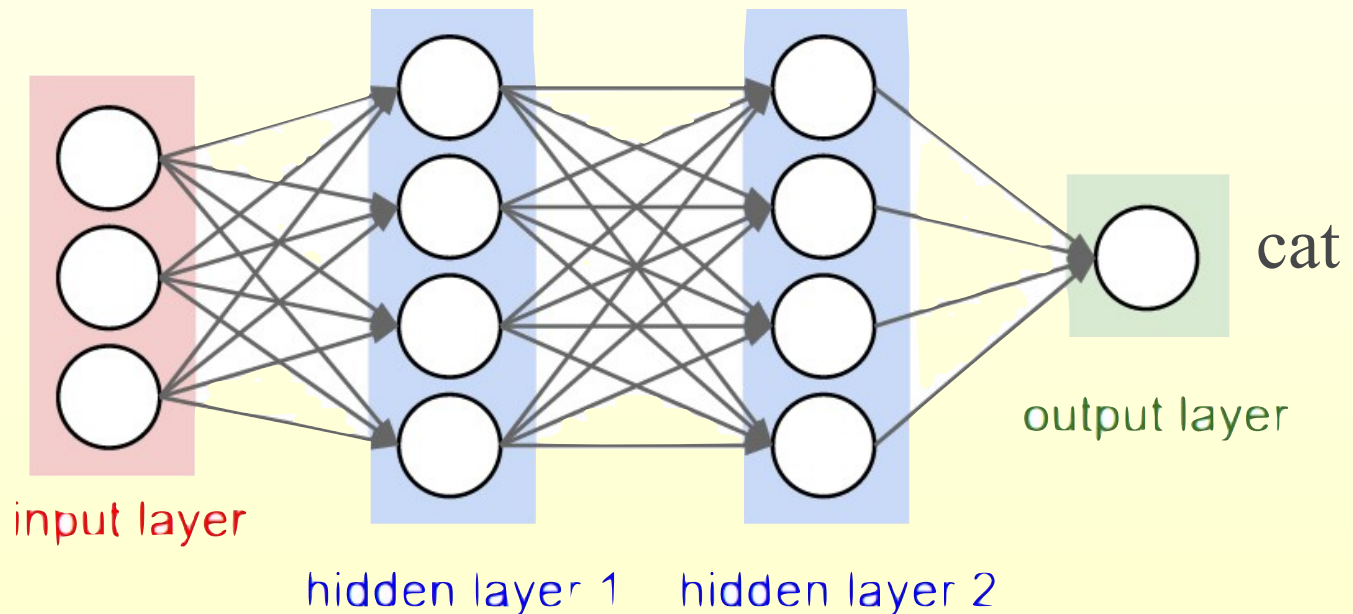


Classification

Deep Learning

- ◆ Deep learning allows computational models that are composed of **multiple processing layers to learn representations of data** with **multiple levels of abstraction**.

Deep Learning by Y. LeCun et al. Nature 2015



Neural Networks as Feature Extractor

- ◆ Neural networks extract powerful features from data

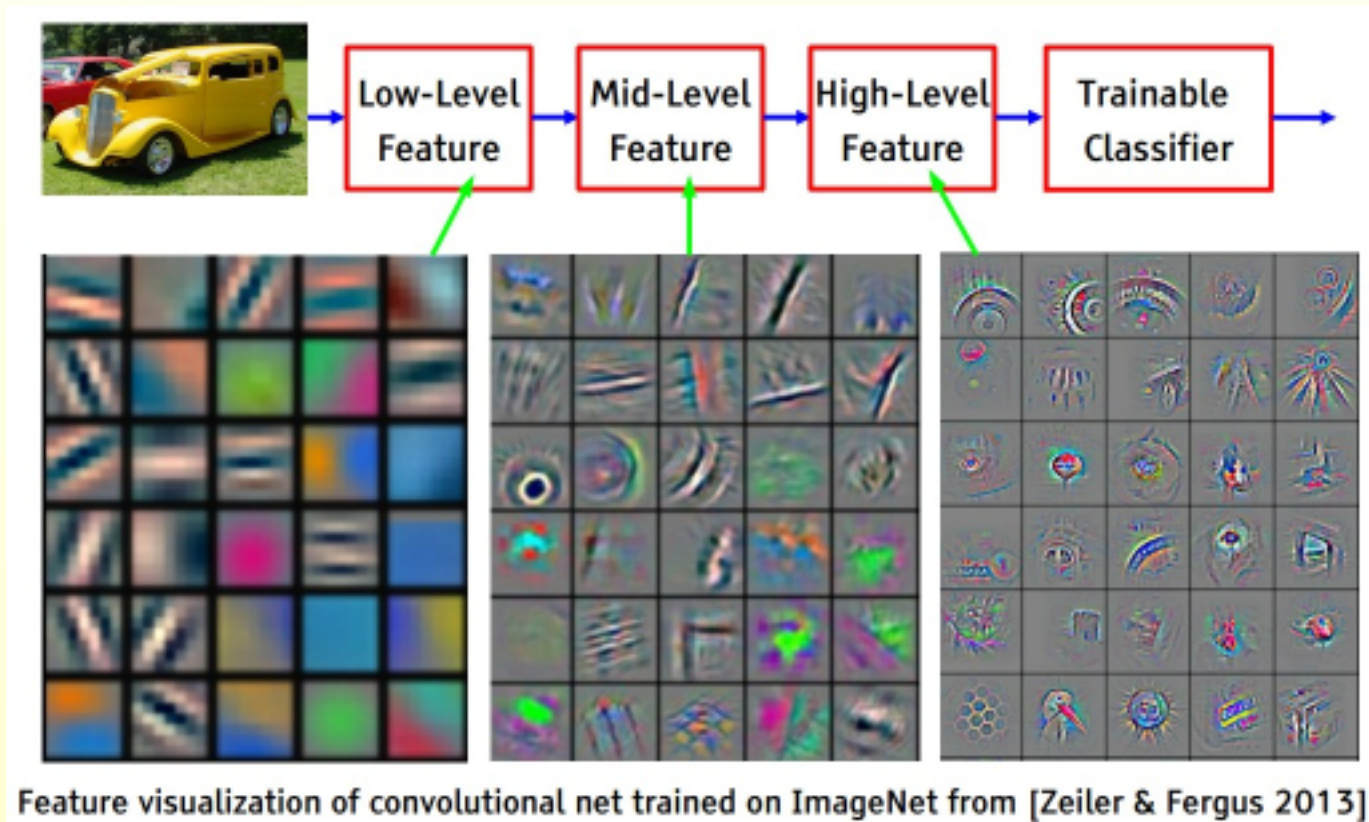
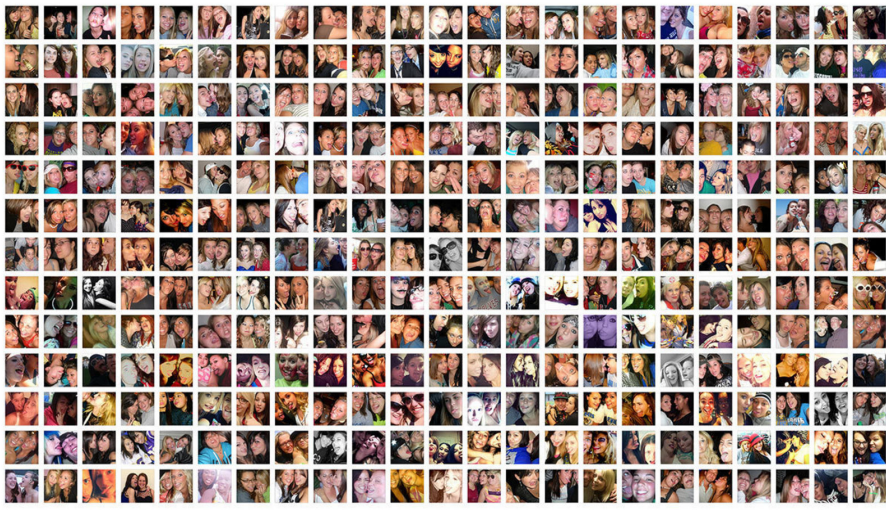


Image Credits: Yan LeCun

Deep Learning: Powered By

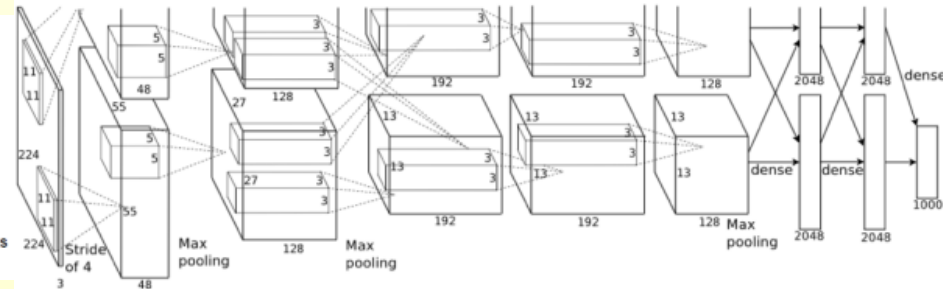
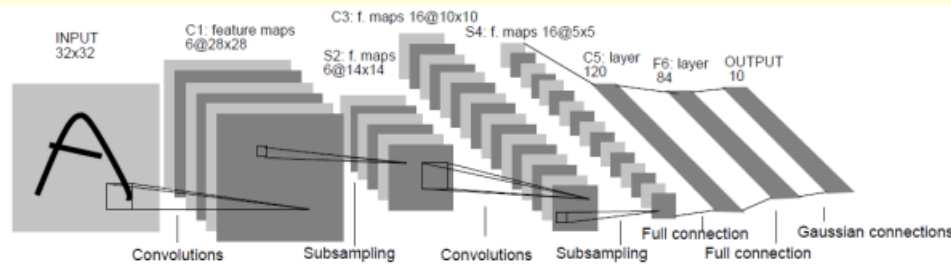


Lots of data



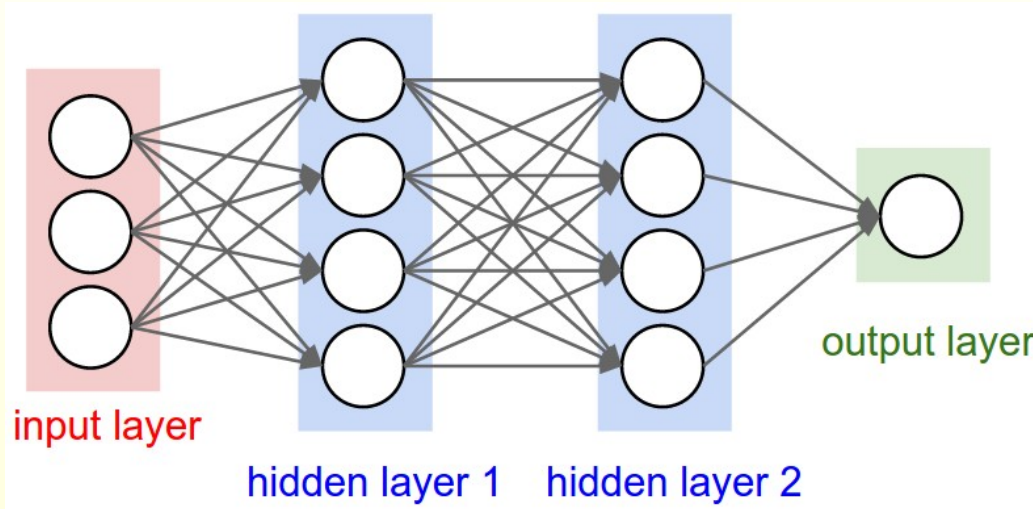
Lots of computing power

+



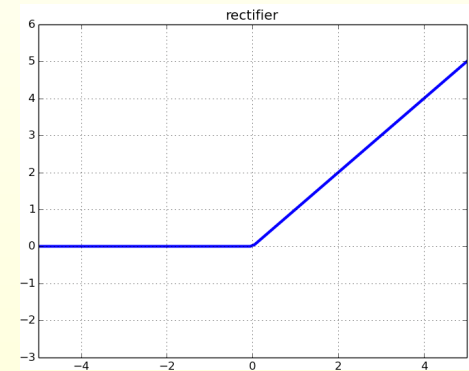
Powerful Neural Network Architectures

Neural Networks



f: non-linear activation function

$$\text{ReLU}(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$$



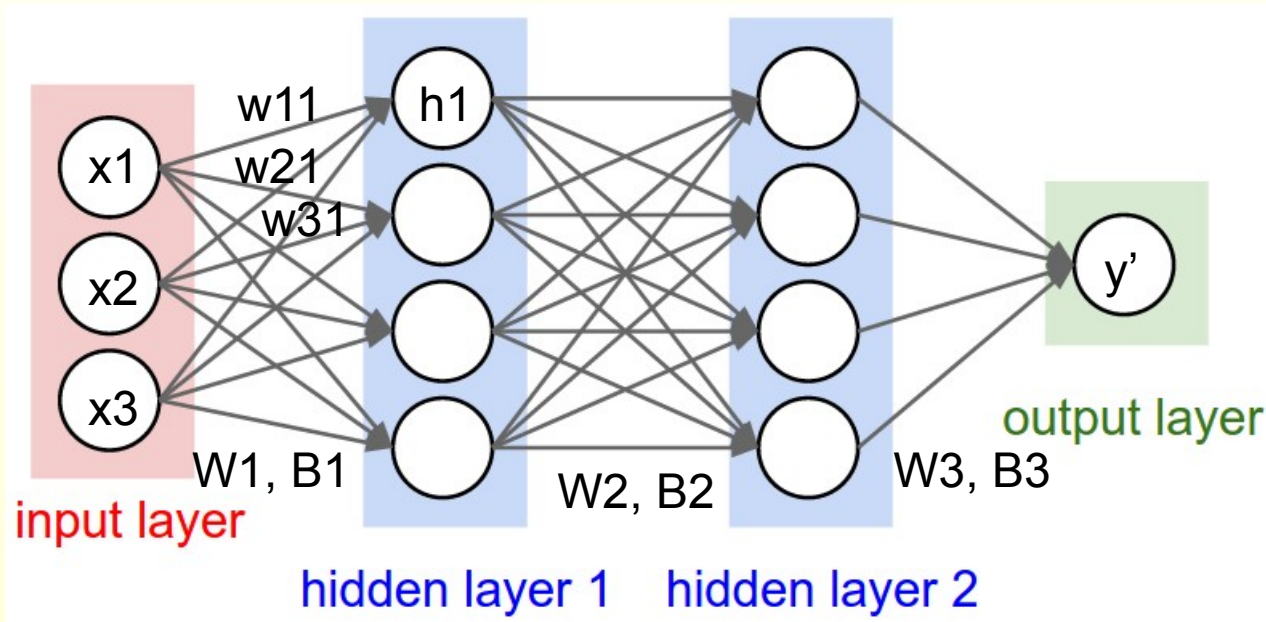
Model: Multi-Layer Perceptron (MLP)

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

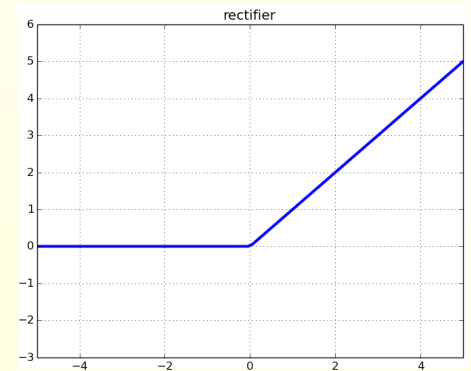
Neural Networks

$$h_1 = f(w_{11} * x_1 + w_{21} * x_2 + w_{31} * x_3 + b_1)$$

f: non-linear activation function



$$\text{ReLU}(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$$



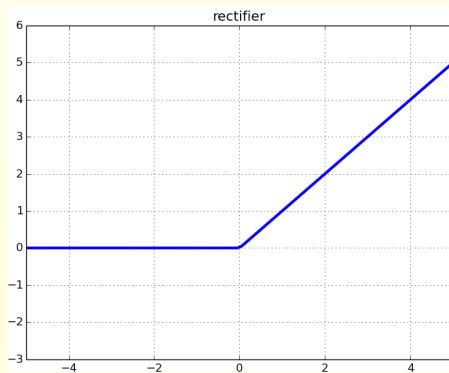
Model: Multi-Layer Perceptron (MLP)

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

Neural Networks

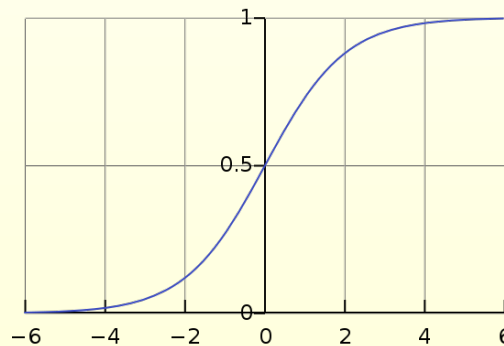
f: non-linear activation function

$$\text{ReLU}(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$$

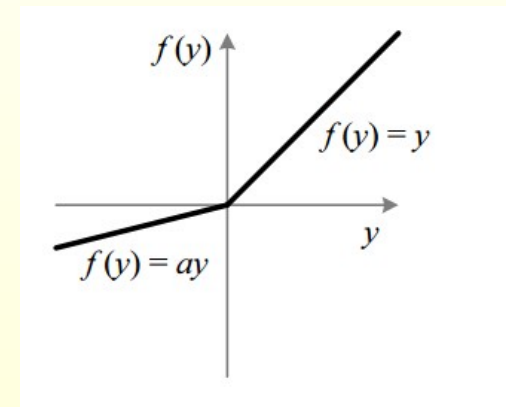


Sigmoid Function

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



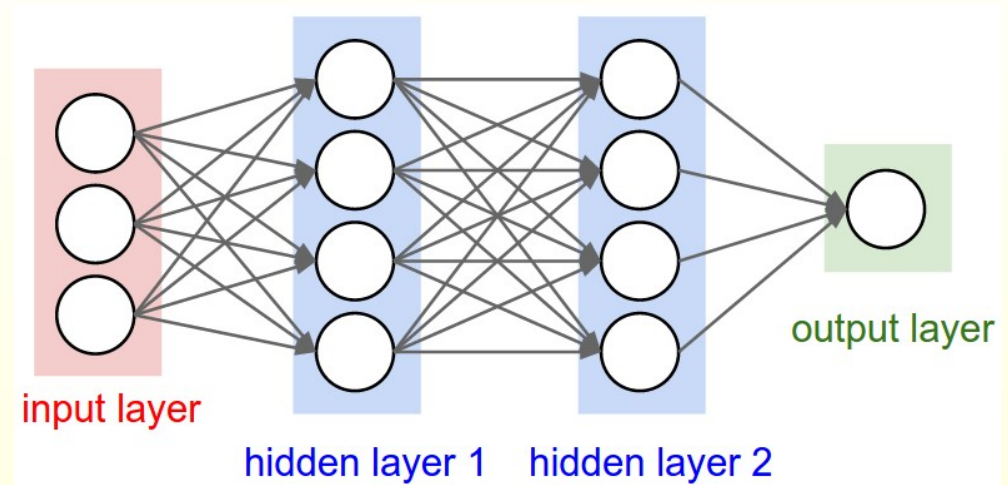
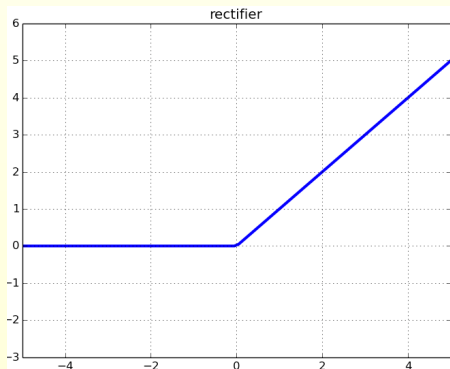
Leaky ReLu



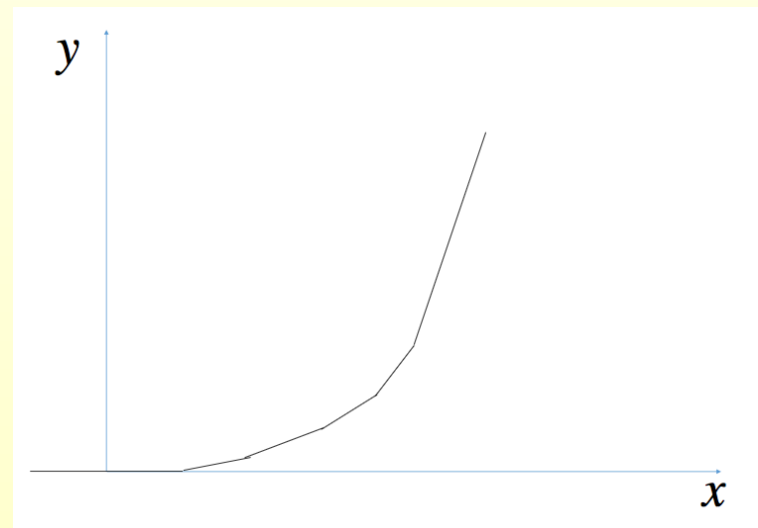
Neural Networks

f: non-linear activation function

$$\text{ReLU}(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$$



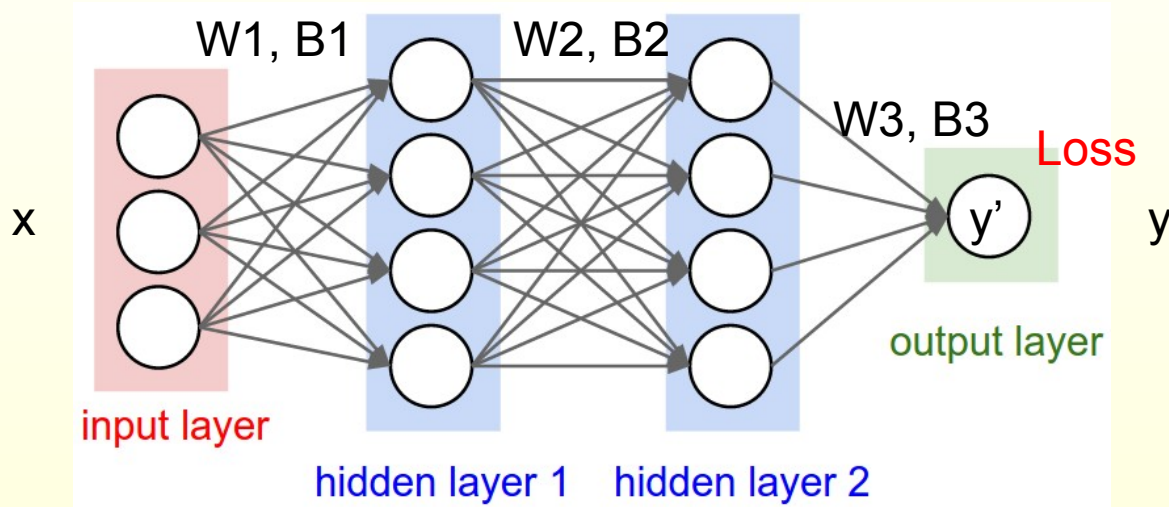
Piece-wise Approximation
 $y = x^2$



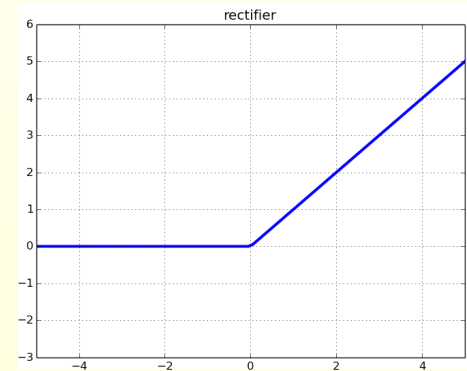
Neural Networks

$$W = \{W1, W2, W3, B1, B2, B3\}$$

f: non-linear activation function



$$\text{ReLU}(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$$



Model: Multi-Layer Perceptron (MLP)

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

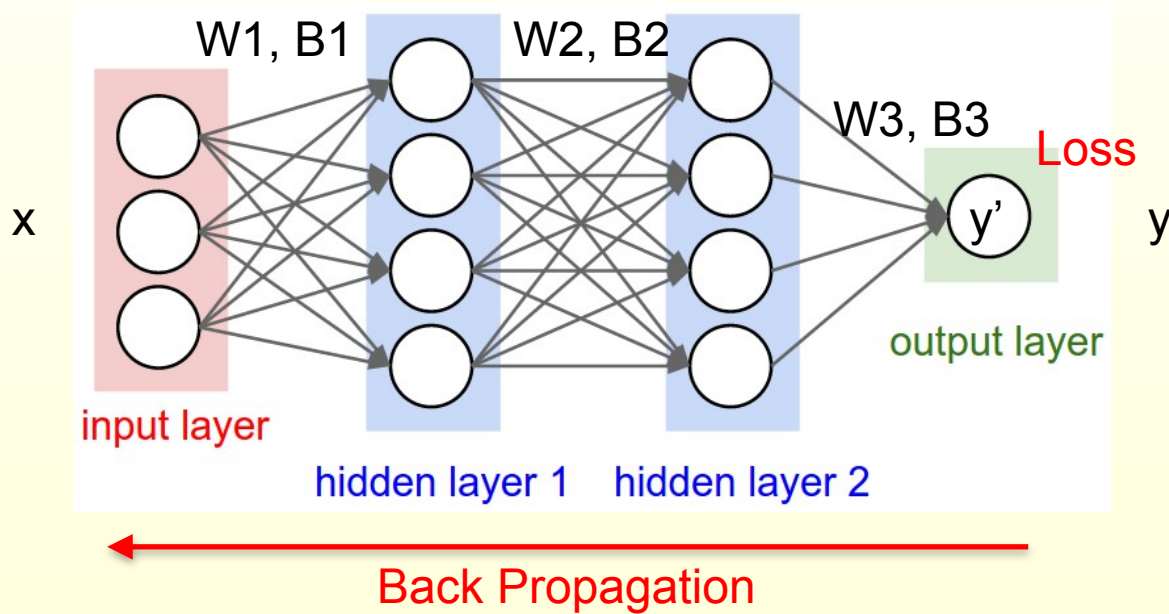
Loss function: L2 loss

$$l(y, y') = (y - y')^2$$

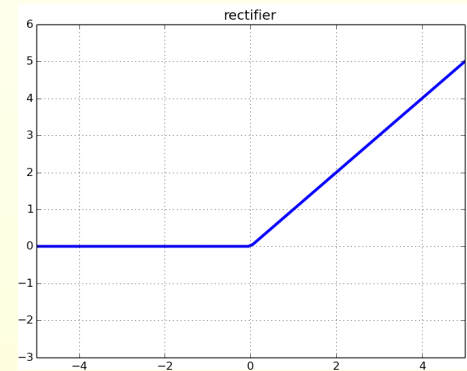
Neural Networks

$$W = \{W1, W2, W3, B1, B2, B3\}$$

f: non-linear activation function



$$\text{ReLU}(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$$



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

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

Neural Networks

A three-layer network approximates any continuous function

Let $\varphi(\cdot)$ be a nonconstant, **bounded**, and **monotonically-increasing continuous** function. Let I_m denote the m -dimensional **unit hypercube** $[0, 1]^m$. The space of continuous functions on I_m is denoted by $C(I_m)$. Then, given any function $f \in C(I_m)$ and $\varepsilon > 0$, there exists an integer N , real constants $v_i, b_i \in \mathbb{R}$ and real vectors $w_i \in \mathbb{R}^m$, where $i = 1, \dots, N$, such that we may define:

$$F(x) = \sum_{i=1}^N v_i \varphi(w_i^T x + b_i)$$

as an approximate realization of the function f where f is independent of φ ; that is,

$$|F(x) - f(x)| < \varepsilon$$

for all $x \in I_m$. In other words, functions of the form $F(x)$ are **dense** in $C(I_m)$.

Neural Networks

A three-layer network approximates any continuous function

Let $\varphi(\cdot)$ be a nonconstant, **bounded**, and **monotonically-increasing continuous** function. Let I_m denote the m -dimensional **unit hypercube** $[0, 1]^m$. Then, given any continuous function $f: I_m \rightarrow \mathbb{R}$, there exists a function $F: I_m \rightarrow \mathbb{R}$ such that $F \in C(I_m)$. Then, given any $f \in C(I_m)$ and $\epsilon > 0$, there exist weights $w_i \in \mathbb{R}^m$, where $i = 1, \dots, N$,

$$F(x) = \sum_{i=1}^N v_i \varphi(w_i \cdot x)$$

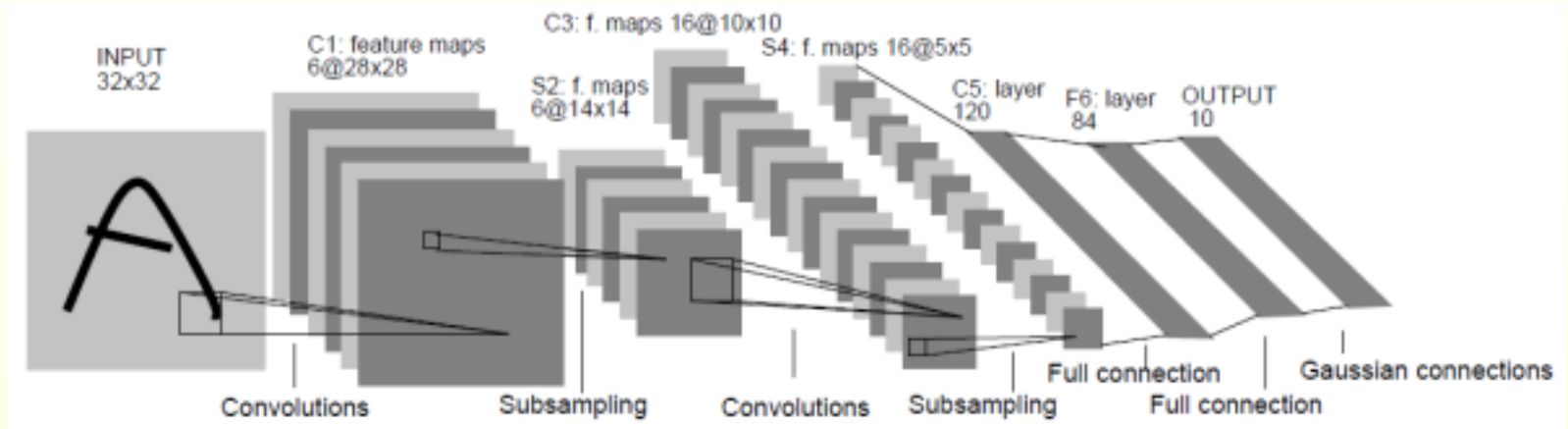
as an approximate

$$|F(x) - f(x)| < \epsilon$$

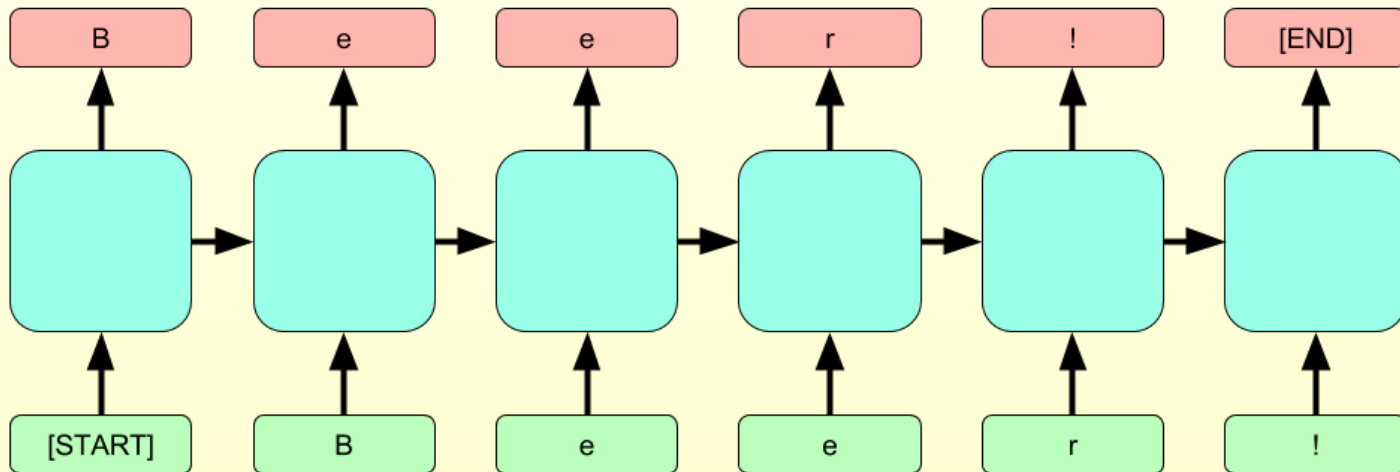
for all $x \in I_m$. In other words, functions of the form $F(x)$ are **dense** in $C(I_m)$.

At the cost of many parameters
and much difficulty to fit

Neural Networks



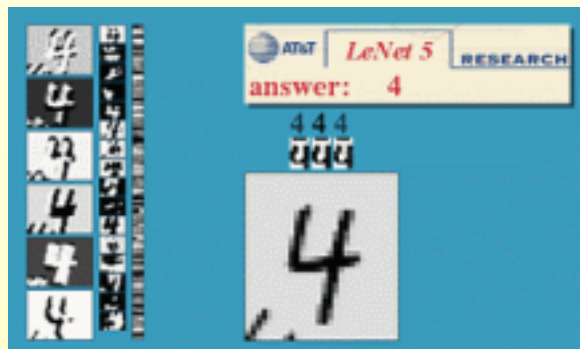
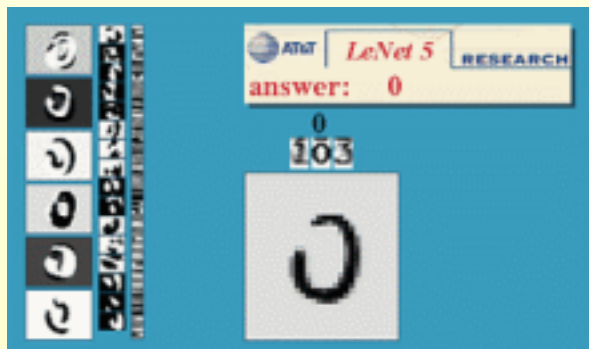
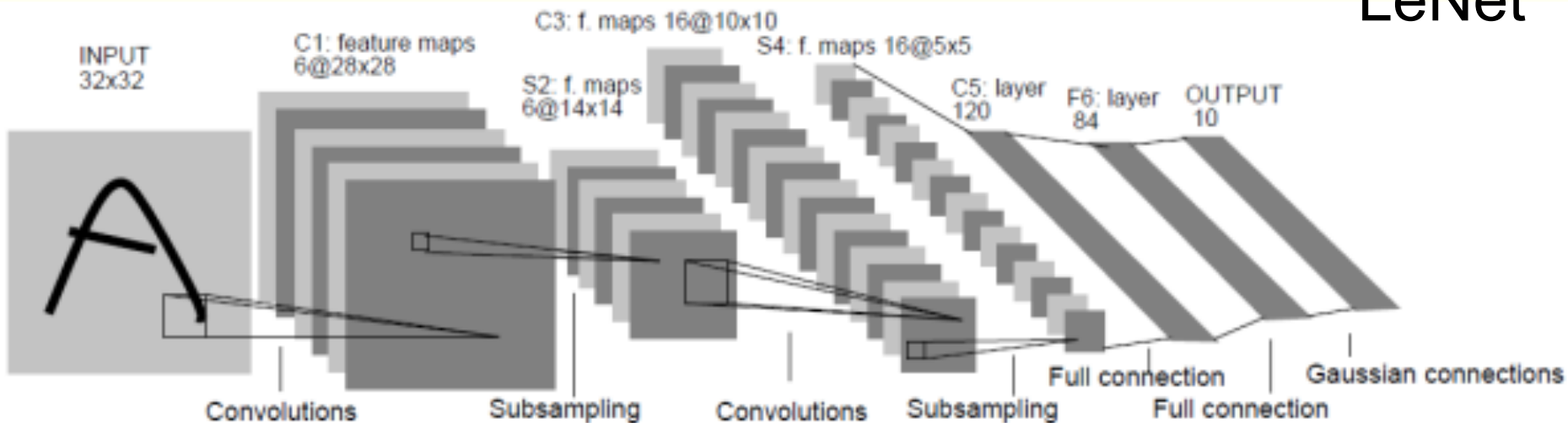
CNN



RNN

Convolutional Neural Networks

LeNet



One of the first successful applications of CNN.

Convolutional Neural Networks

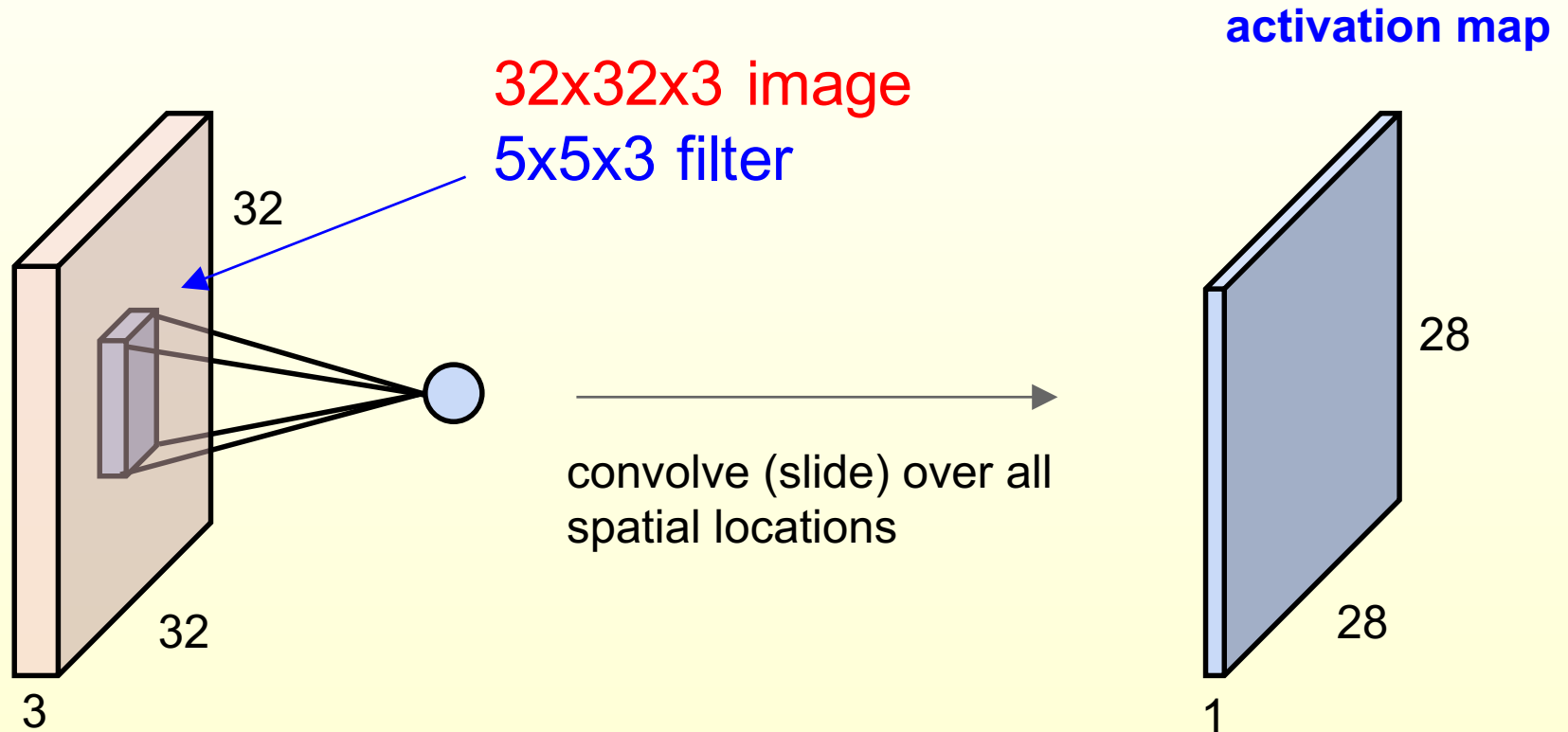


Image Credits: Andrej Karpathy

Convolutional Neural Networks

- ◆ Filters are doing pattern matching

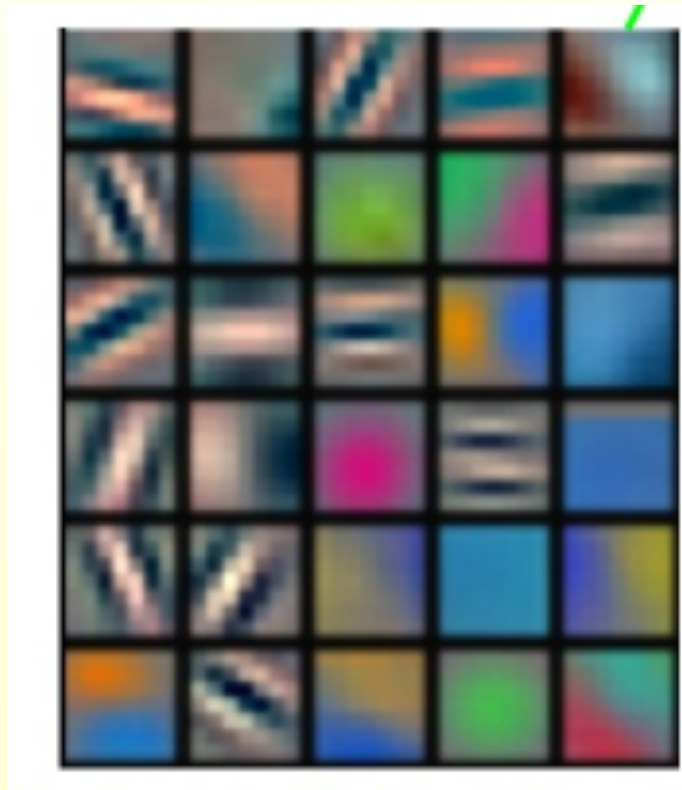
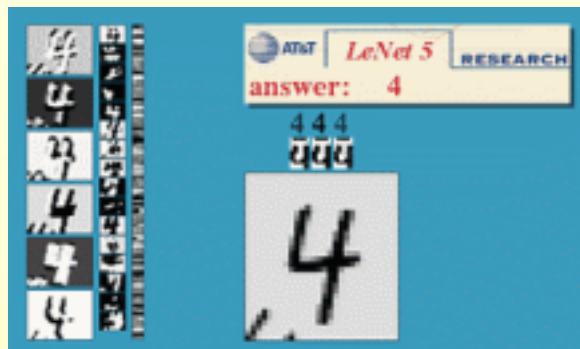
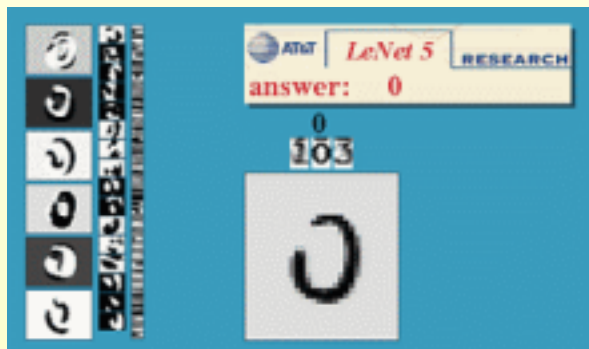
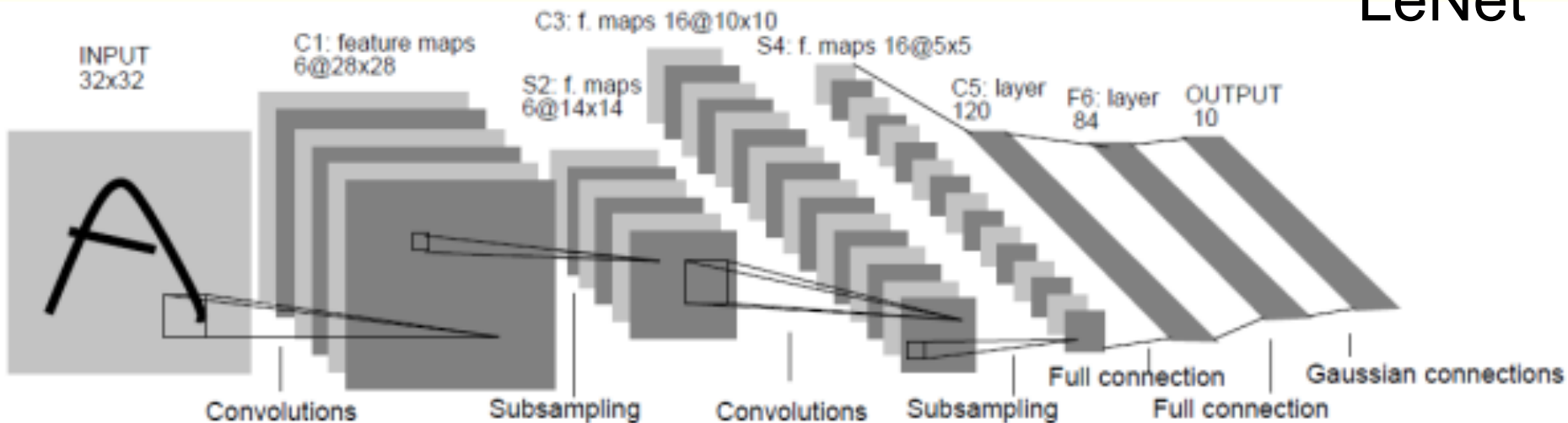


Image Credits: Yan LeCun

Convolutional Neural Networks

LeNet

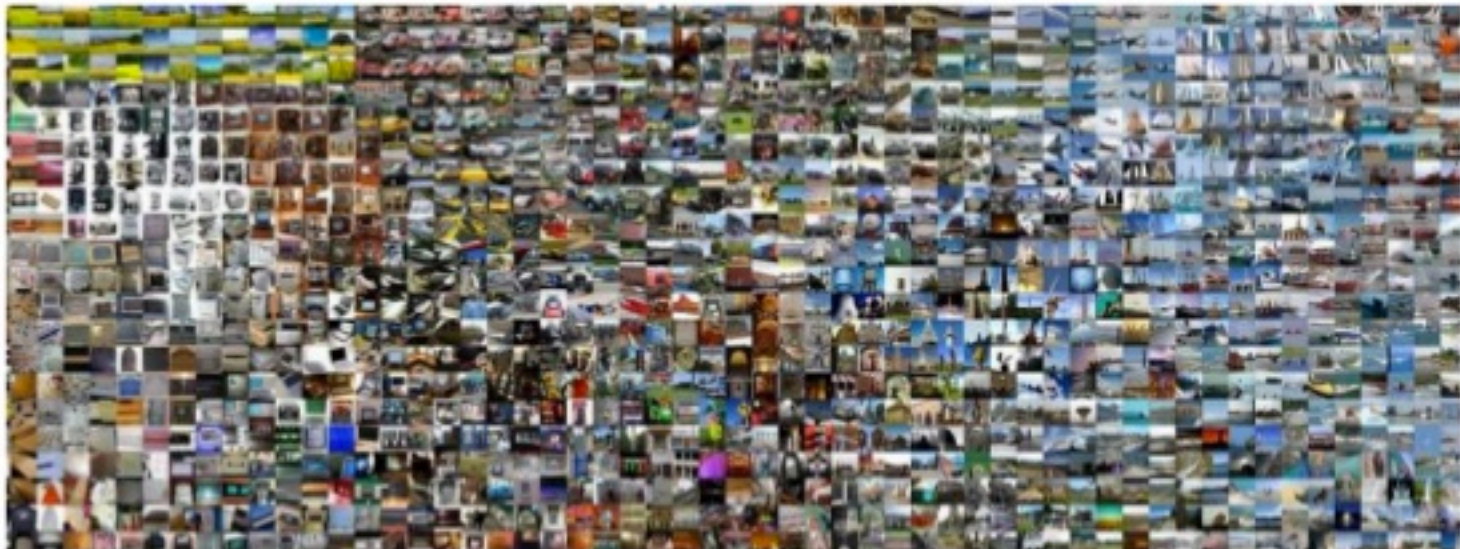


One of the first successful applications of CNN.

ImageNet Challenge

ImageNet Dataset

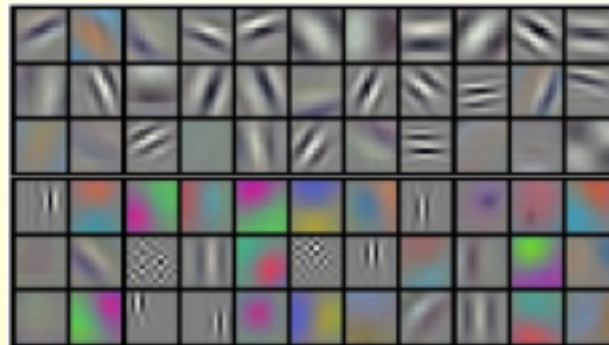
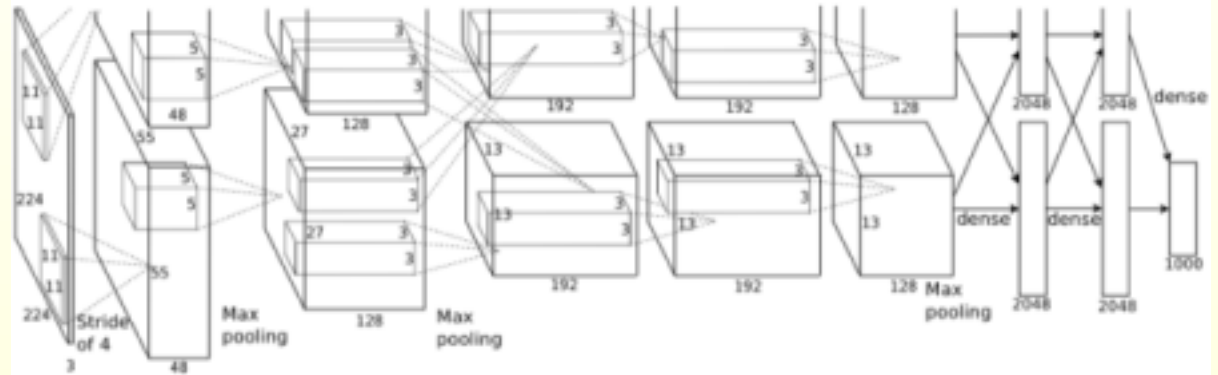
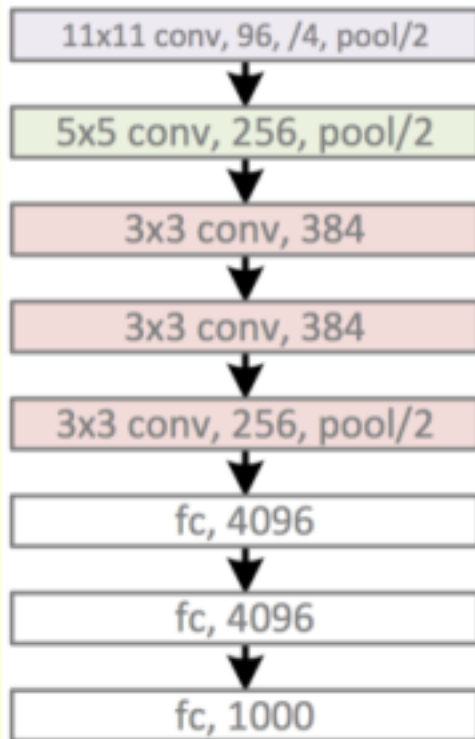
IMAGENET



Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang et al. ["Imagenet large scale visual recognition challenge."](#) International Journal of Computer Vision 115, no. 3 (2015): 211-252. [\[web\]](#)

Convolutional Neural Networks

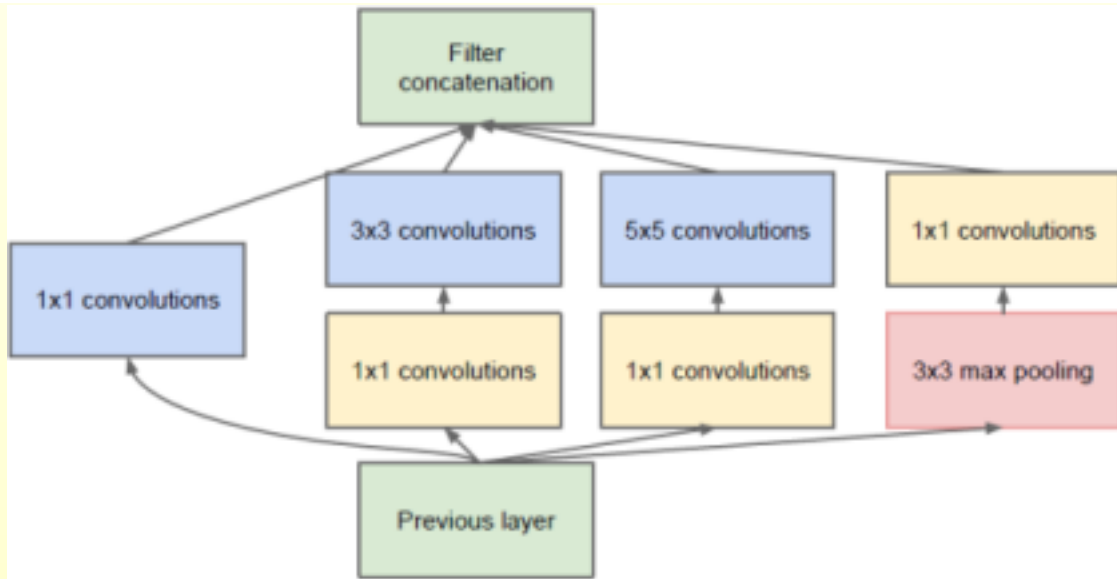
AlexNet



The first work that popularized Convolutional Networks in Computer Vision

Convolutional Neural Networks

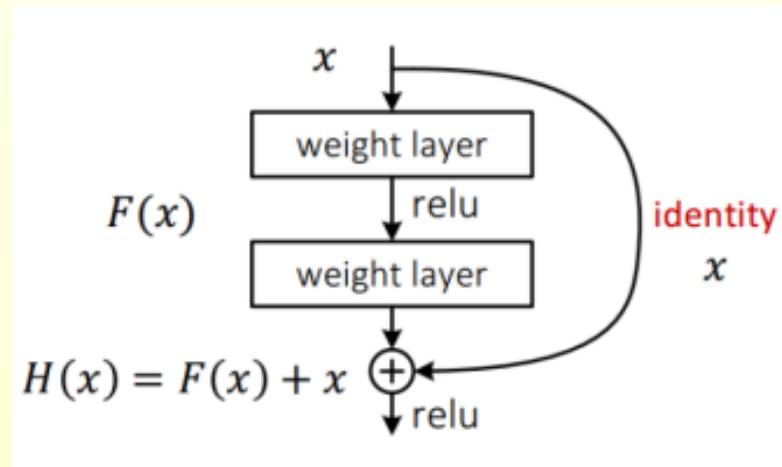
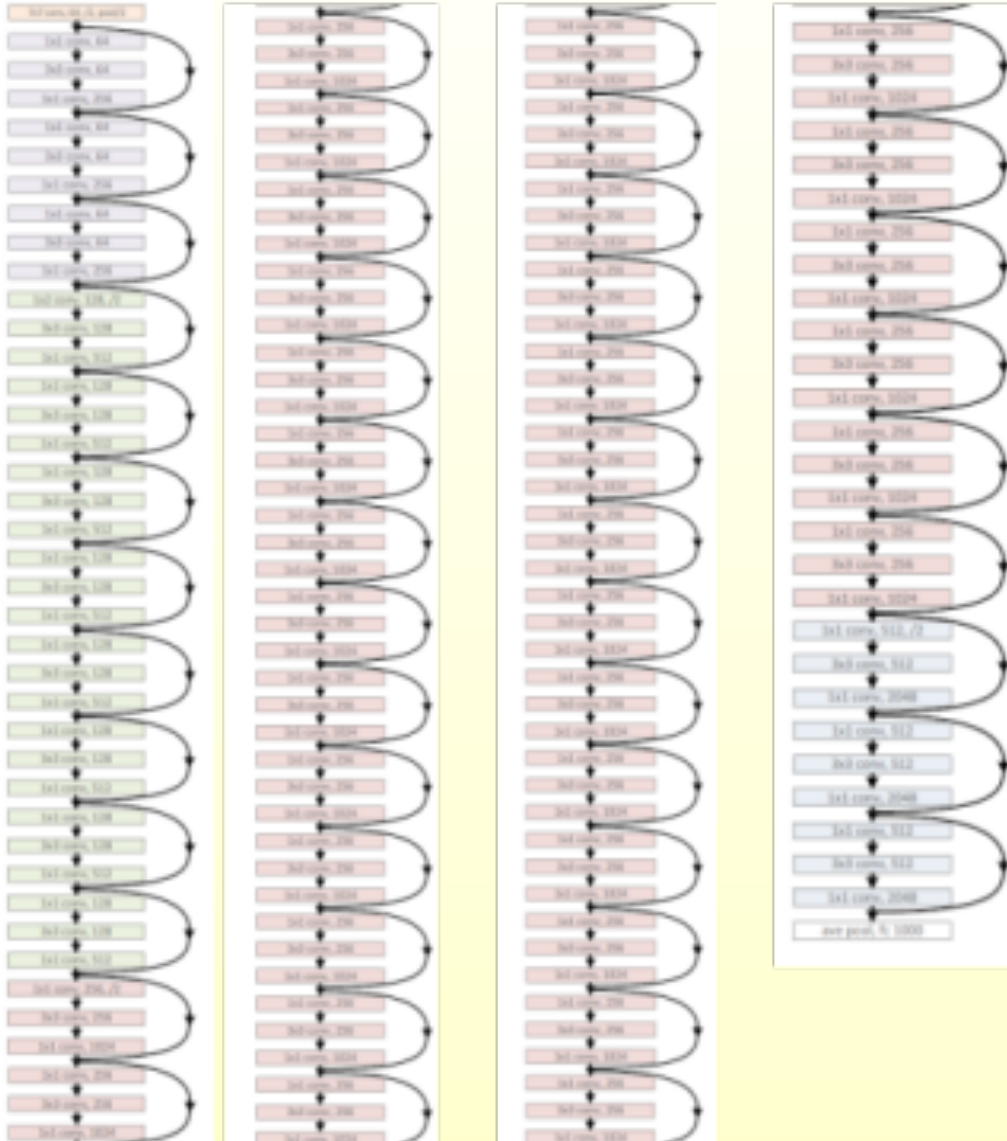
GoogleNet



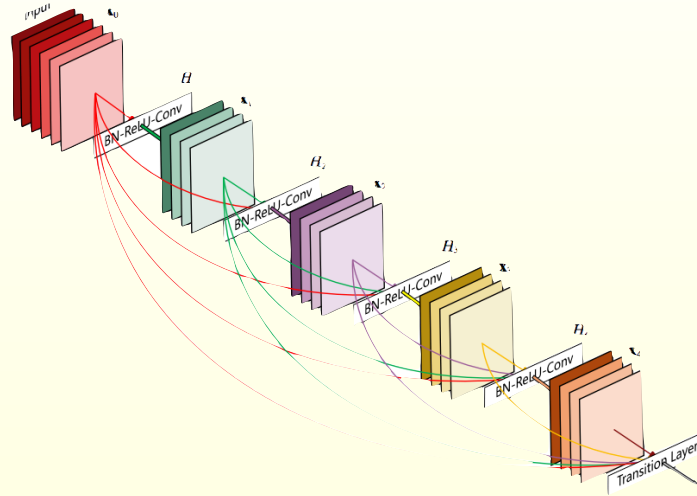
An Inception Module: a new building block..

Convolutional Neural Networks

ResNet



Convolutional Neural Networks



DenseNet

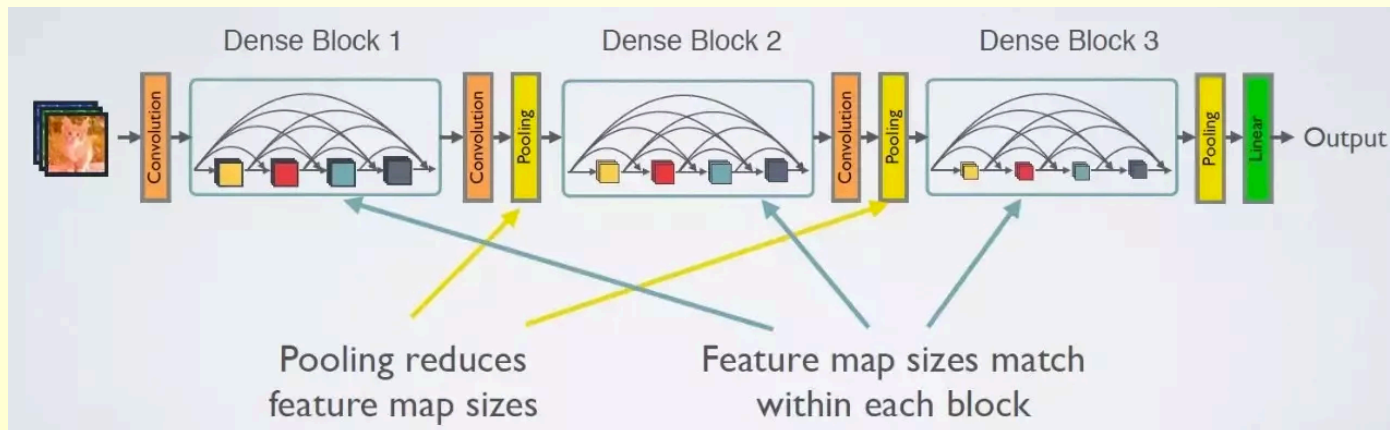
DenseNet-264

$$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$$

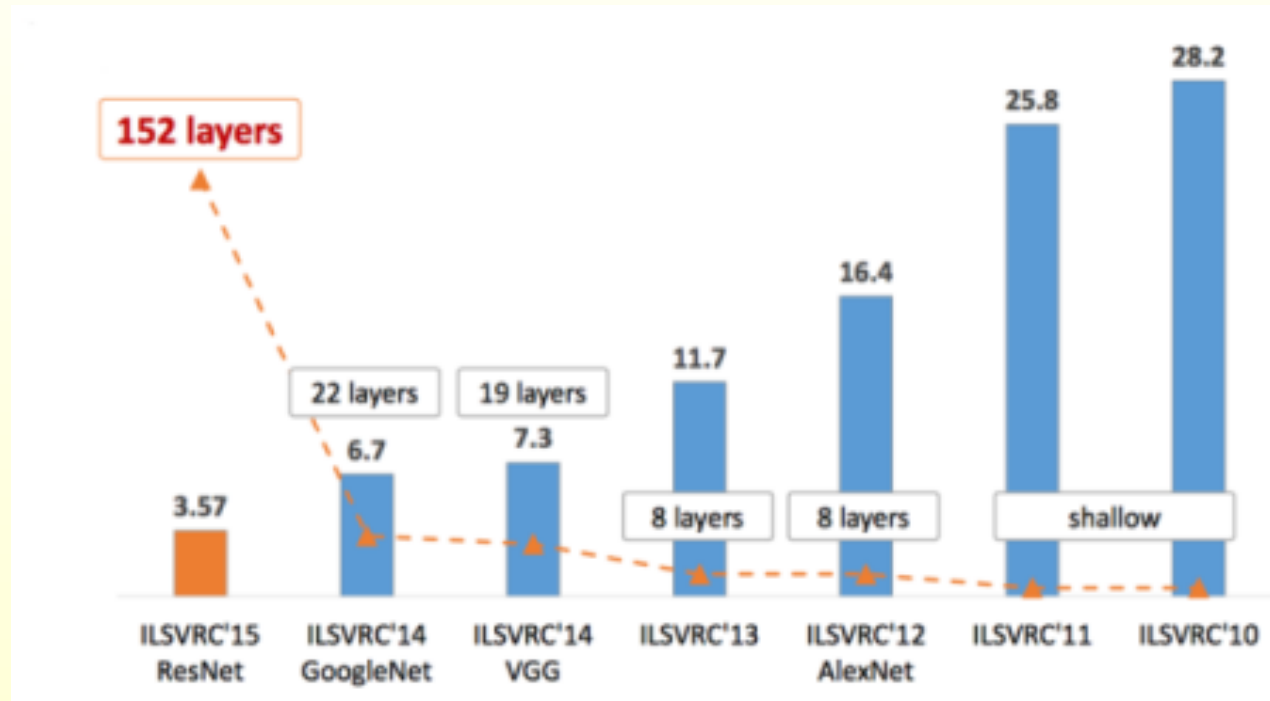
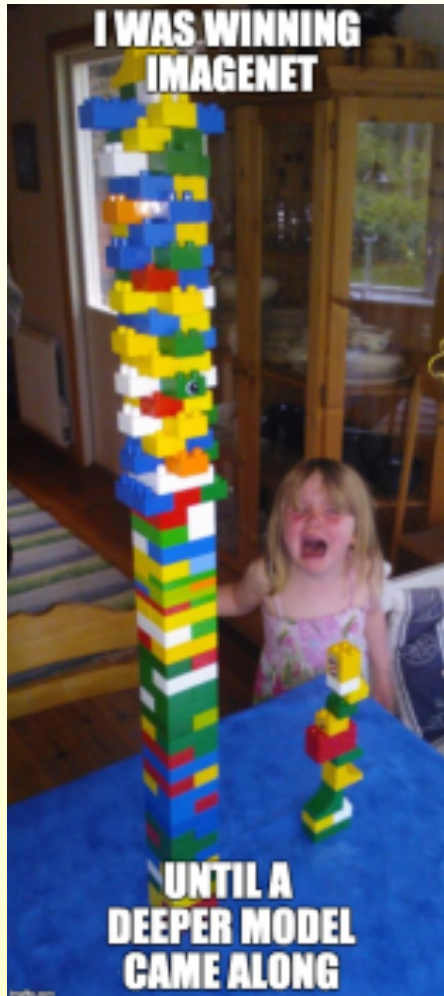
$$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$$

$$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$$

$$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$$



Convolutional Neural Networks



Classification Error: the lower, the better

Neural Networks as Feature Extractor

◆ How do neural networks work?

Neural Networks as Feature Extractor

◆ Traditional methods hand-engineer features

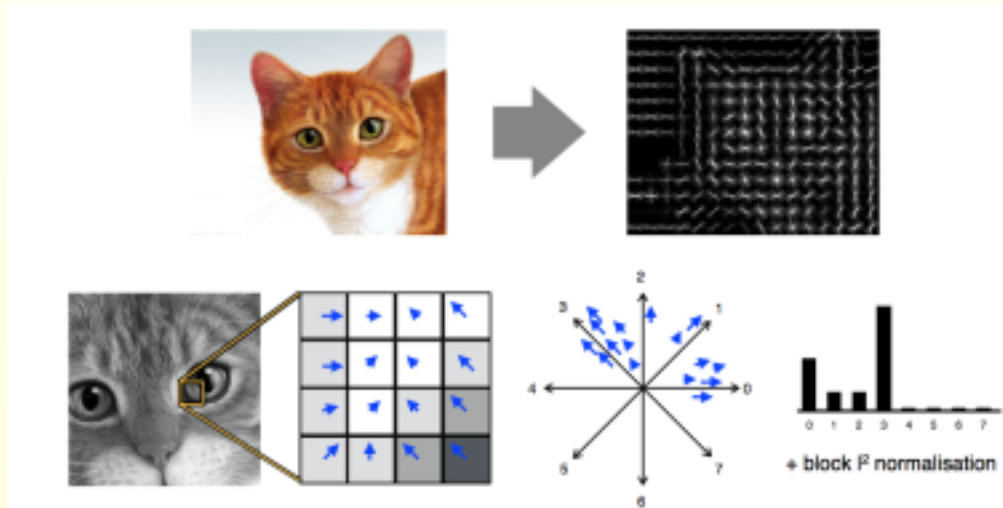
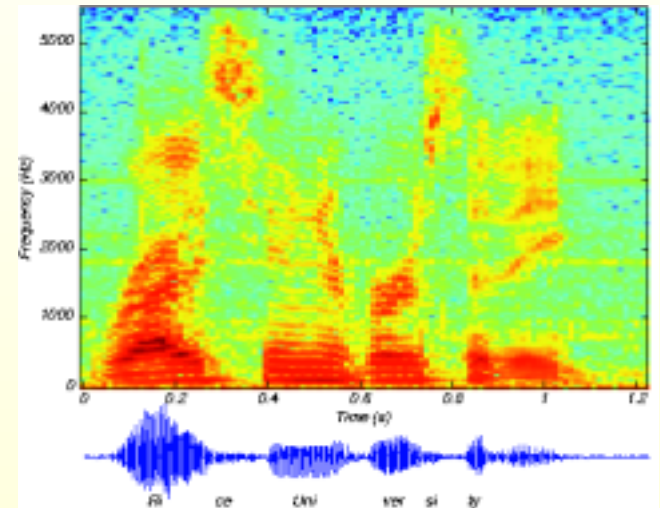


Image: HoG



Audio: Spectrogram

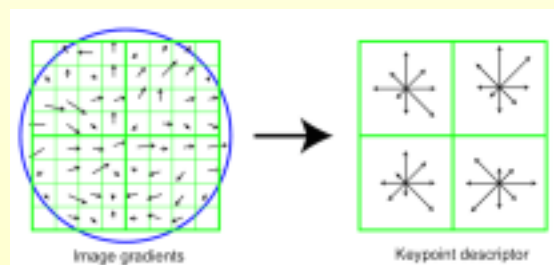
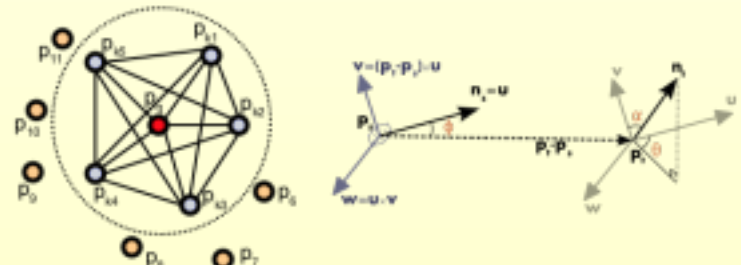
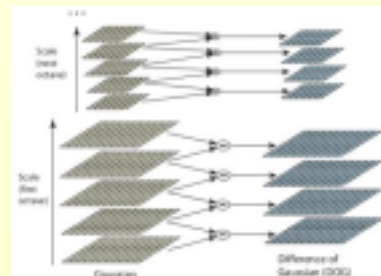


Image: SIFT



Point Cloud: PFH

Neural Networks as Feature Extractor

- ◆ Neural networks extract powerful features from data

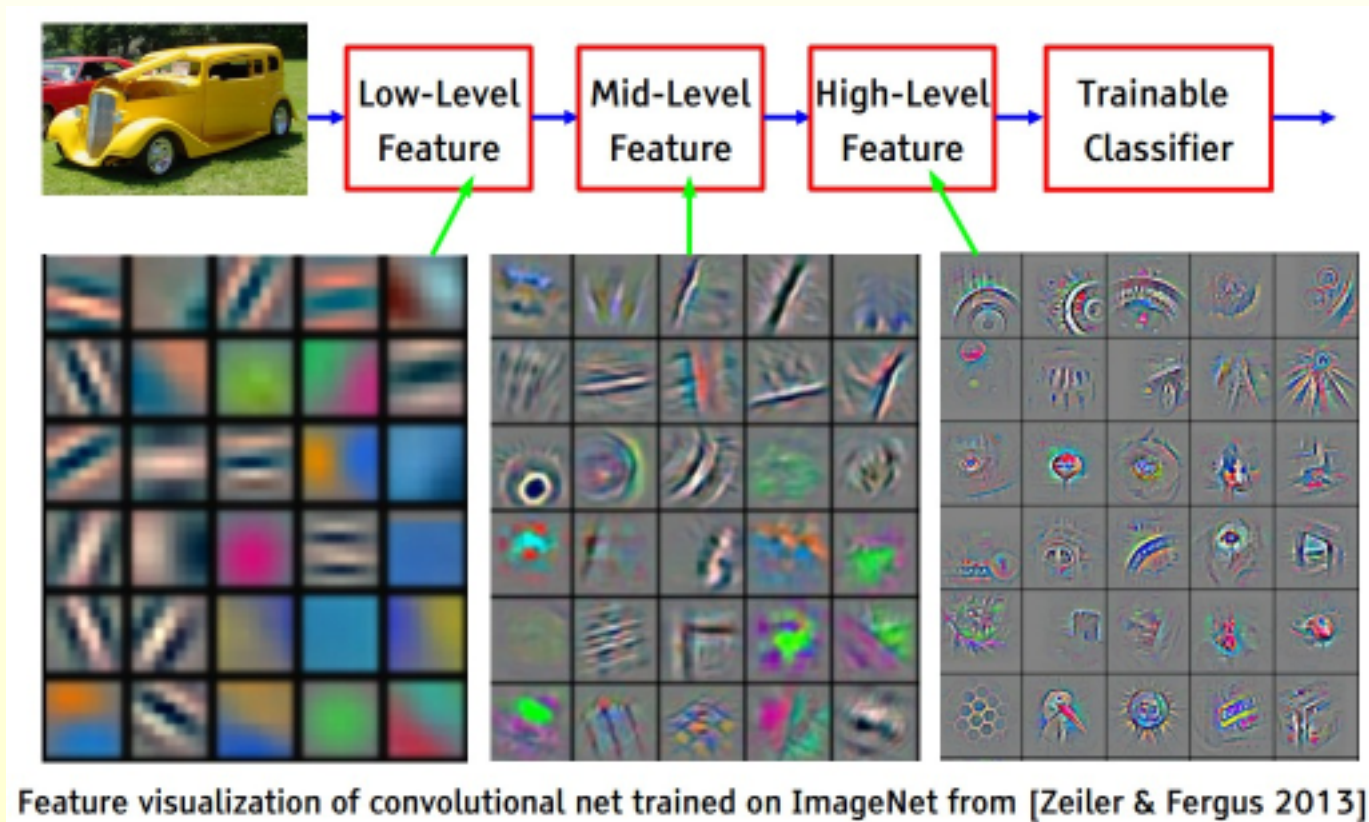
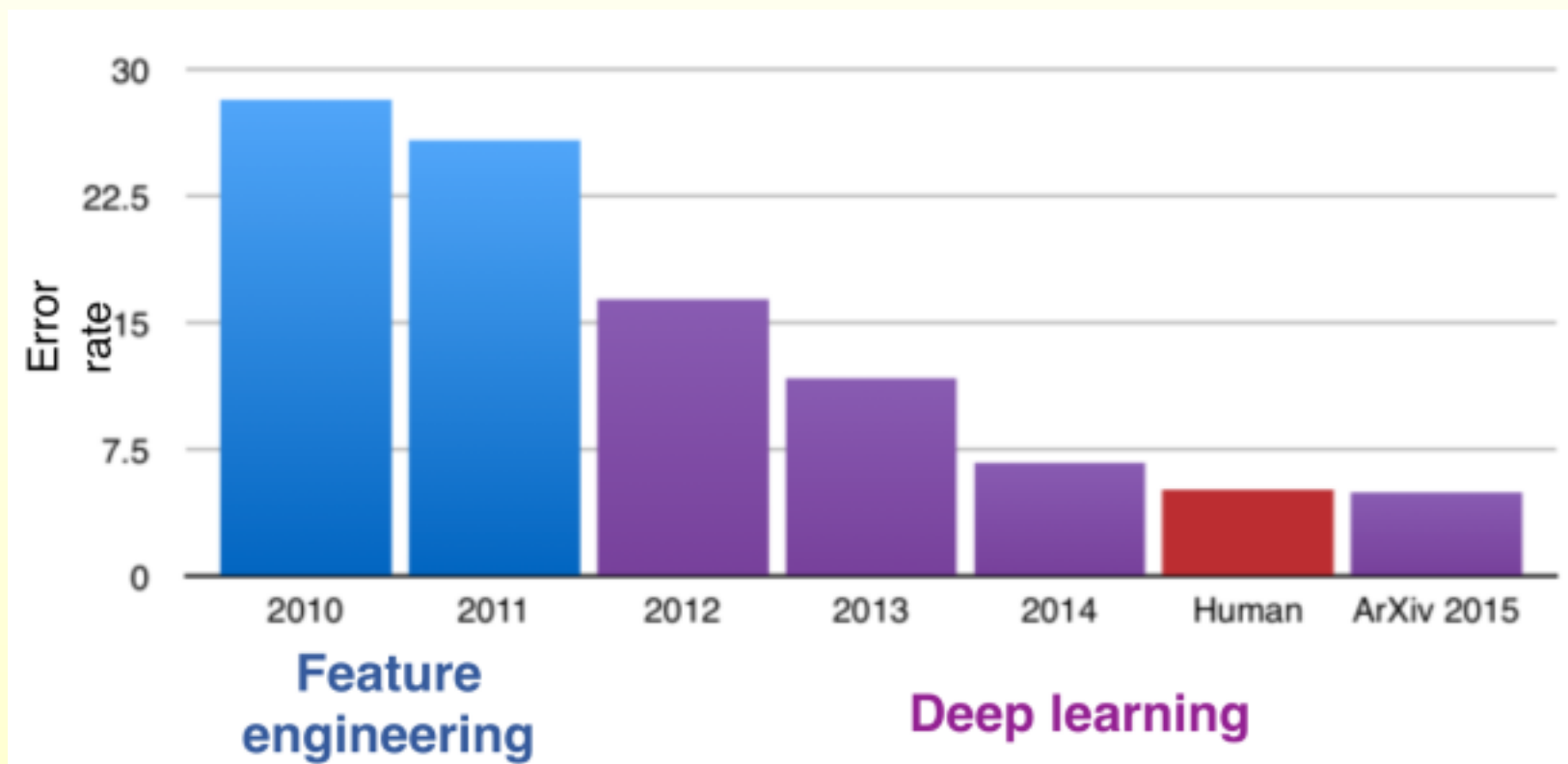


Image Credits: Yan LeCun

Neural Networks as Feature Extractor

ImageNet 1000 class image classification accuracy



Useful Courses

- ◆ CS 231n: <http://cs231n.stanford.edu/>
 - ◆ Deep Learning for Visual Data
- ◆ CS 224n: <http://cs224n.stanford.edu/>
 - ◆ Deep Learning for Language / Sequential Data
- ◆ CS 234: <http://cs234.stanford.edu/>
 - ◆ Reinforcement Learning

Programming Neural Networks

◆ TensorFlow, Python, Google

◆ <https://www.tensorflow.org/>

The TensorFlow logo consists of the word "TensorFlow" in white text on an orange rectangular background, with a small "TM" trademark symbol to the right.

◆ PyTorch, Python, Facebook

◆ <https://pytorch.org/>

The PyTorch logo features the word "PYTORCH" in white text on a dark gray rectangular background. The letter "O" is replaced by a stylized orange flame icon.

◆ Caffe, Berkeley

◆ <http://caffe.berkeleyvision.org/>

Programming Neural Networks



```
y_train = keras.utils.to_categorical(y_train, num_classes)
y_test = keras.utils.to_categorical(y_test, num_classes)
```

```
model = Sequential()
model.add(Conv2D(32, kernel_size=(3, 3),
                activation='relu',
                input_shape=input_shape))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(num_classes, activation='softmax'))
```

```
model.compile(loss=keras.losses.categorical_crossentropy,
              optimizer=keras.optimizers.Adadelta(),
              metrics=['accuracy'])

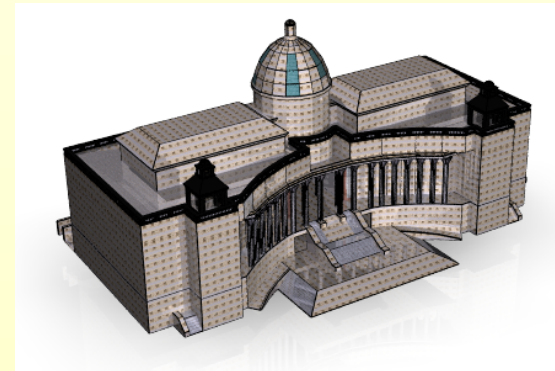
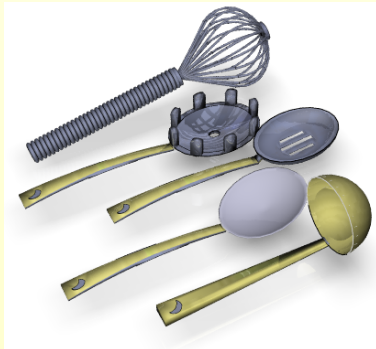
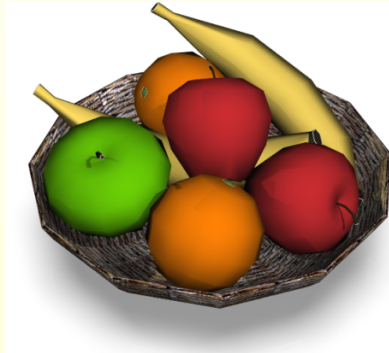
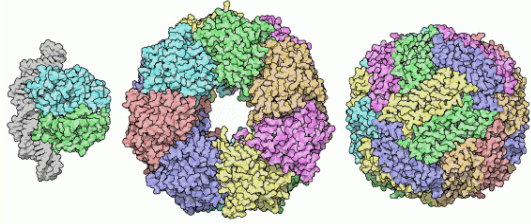
model.fit(x_train, y_train,
          batch_size=batch_size,
          epochs=epochs,
          verbose=1,
          validation_data=(x_test, y_test))

score = model.evaluate(x_test, y_test, verbose=0)
print('Test loss:', score[0])
print('Test accuracy:', score[1])
```

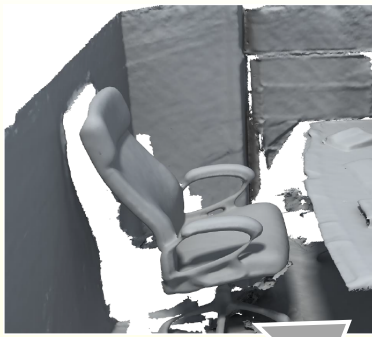
Agenda

- ◆ Today
 - ◆ Deep Learning Intro
 - ◆ **3D Deep Learning**
 - ◆ Multi-view CNNs
 - ◆ Volumetric CNNs
- ◆ Later lectures
 - ◆ Point-cloud NNs (May 21)
 - ◆ Graph CNNs (May 24)

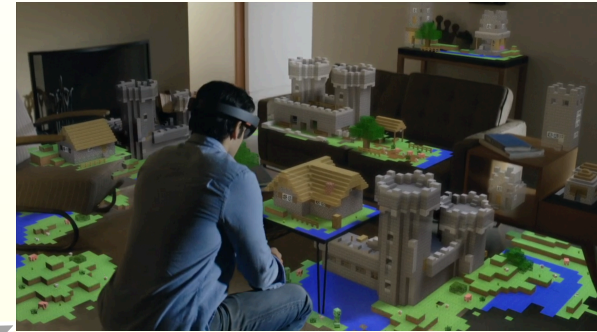
The World is 3D



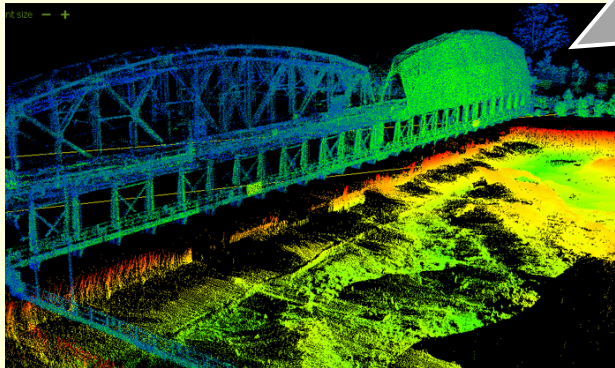
3D Applications



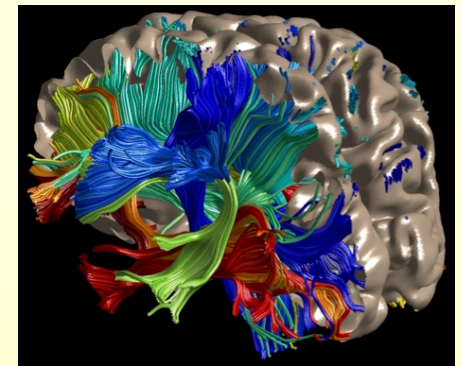
Robotics



Augmented Reality



Autonomous driving



Medical Image Processing

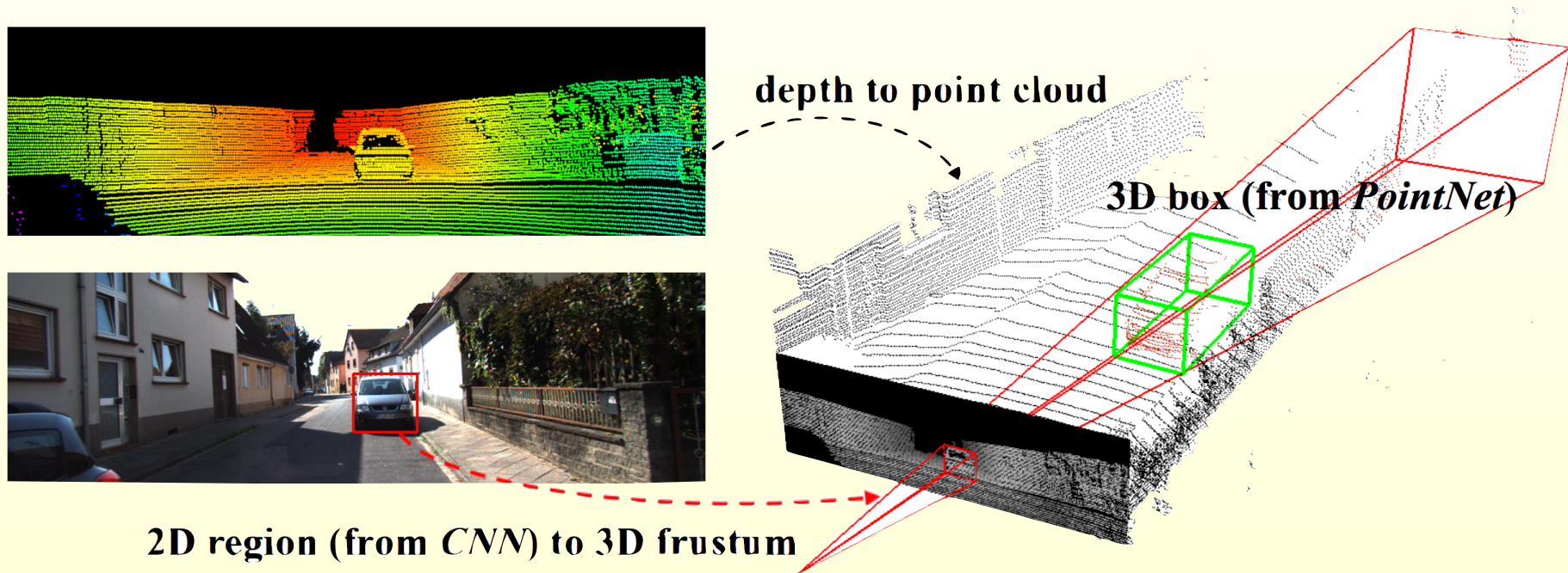
3D Perception is important for AI



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

Understanding in 3D is “Easier”!

- ◆ Depth Data really helps (e.g. laser scan)



Frustum PointNets for 3D Object Detection from RGB-D Data, Charles Qi et. al.

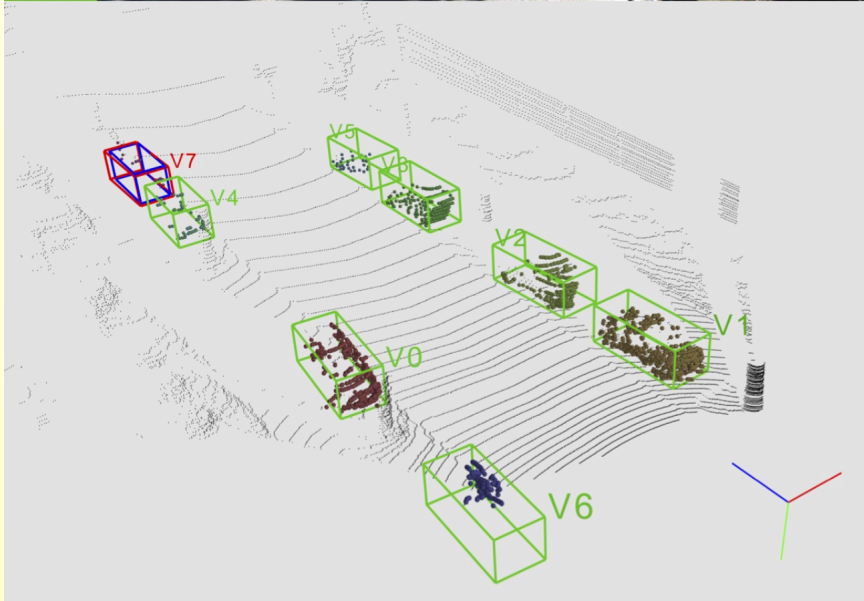
Understanding in 3D is “Easier”!

- ◆ 3D scanners even can give us full 3D models!



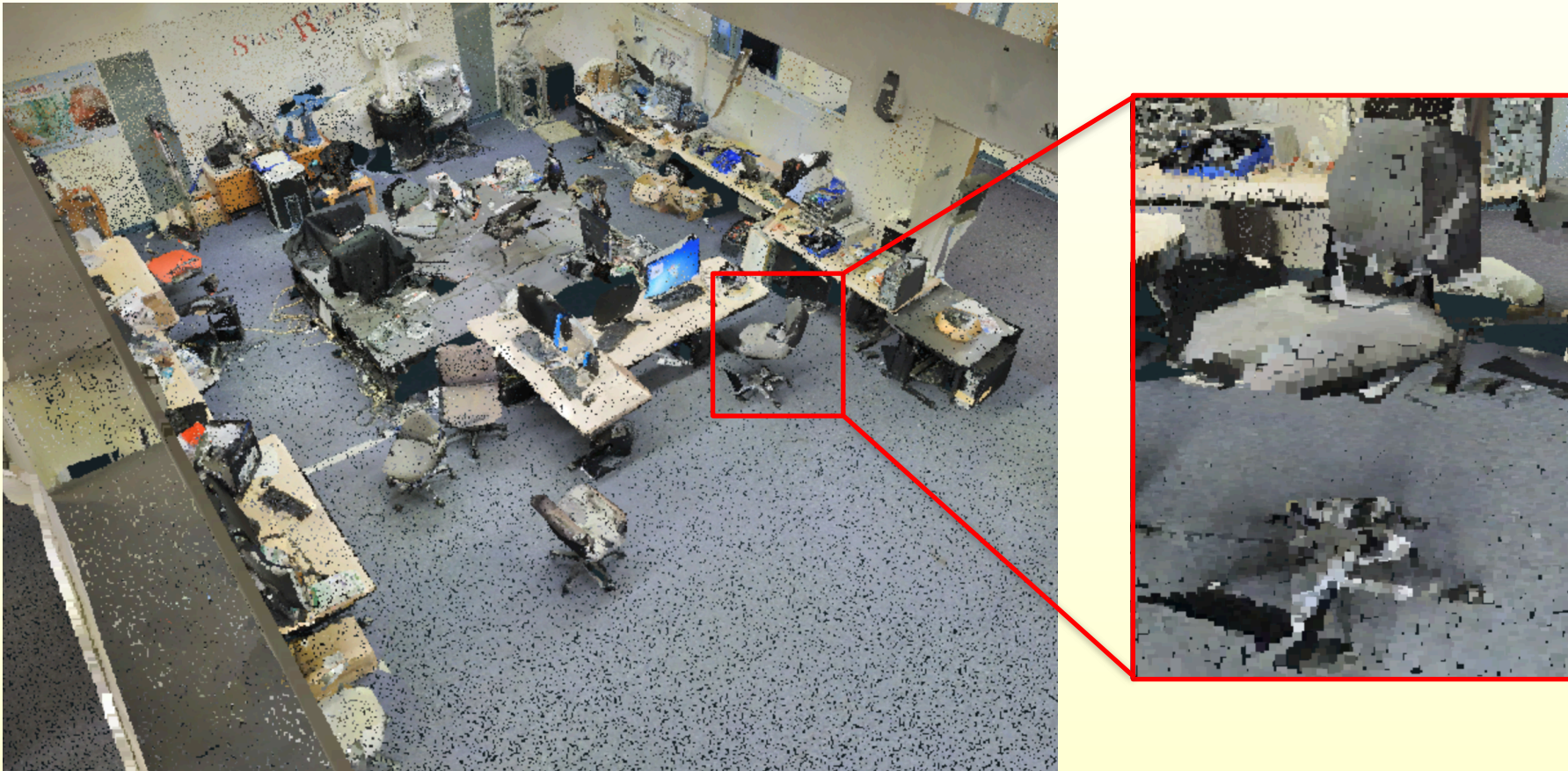
Understanding in 3D is “Easier”!

- ◆ 3D data is less vulnerable to different occlusions, different lighting conditions, etc.



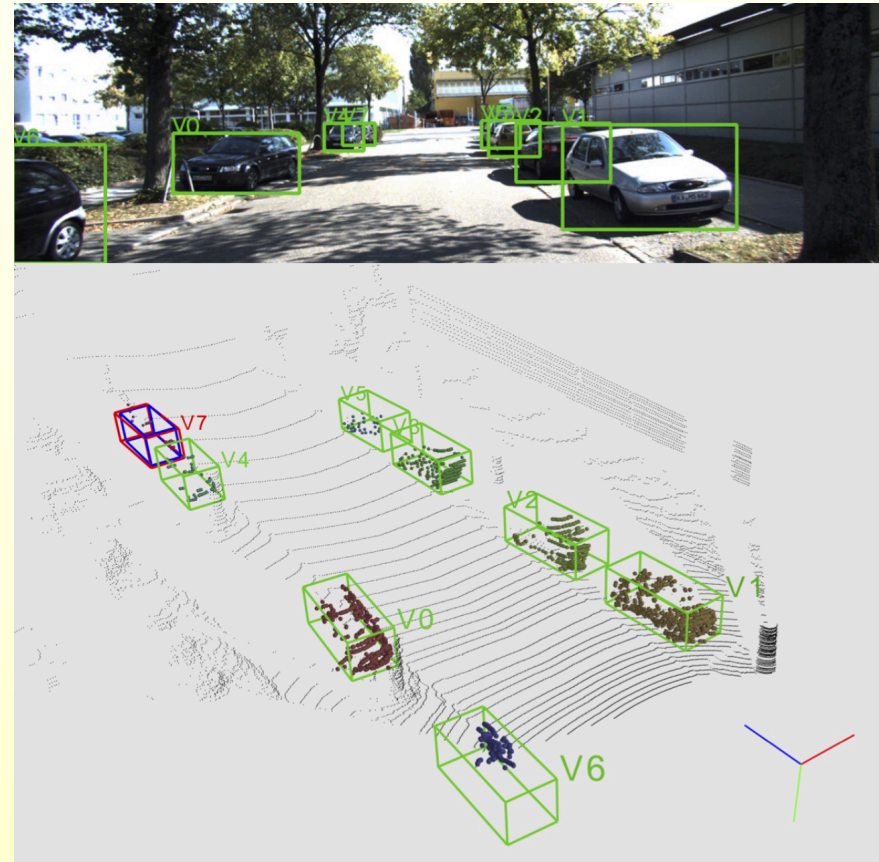
Understanding in 3D is Harder!

- ◆ 3D data is often noisy and of low quality.



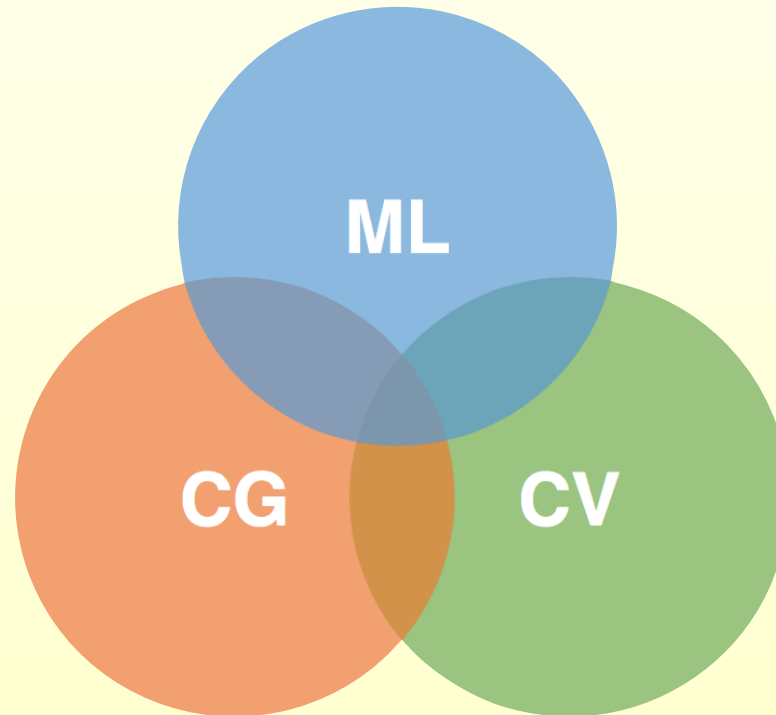
Understanding in 3D is Harder!

- ◆ 3D data is often incomplete.



3D Deep Learning

- ◆ A field with very short history — starting from 2015
- ◆ But very active due to huge industry interests!



Big Data in 3D

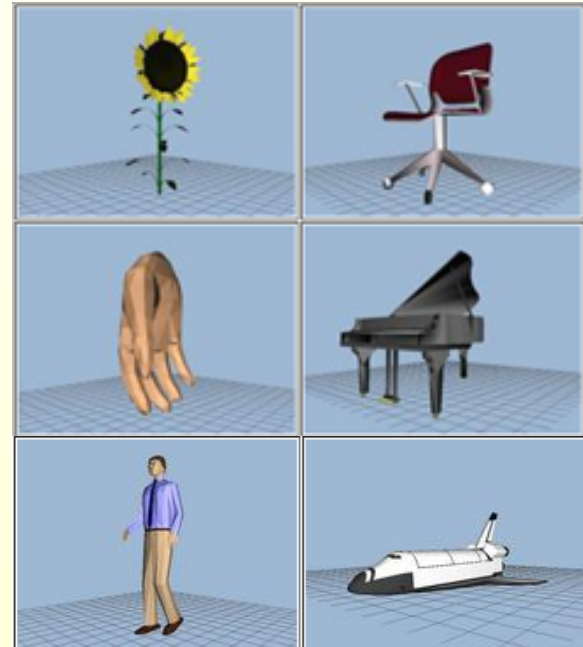
Status as of 2010:



Stanford bunny



1800 models in **90** categories



Princeton shape benchmark
[Shilane et al. 04]

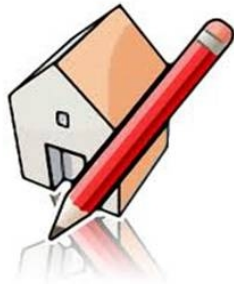
Big Data in 3D

The screenshot displays the 3D Warehouse interface. At the top, the logo '3D Warehouse' is on the left, and 'Upload Model' is on the right. Below the logo, the search term 'chair' is entered in a search bar, with a 'Search' button and a 'Sign In' link. The results section shows '37,091 Results' and filters for 'ALL', 'Results Per Page', 'Models', and 'Collections'. The main area is a grid of 3D model thumbnails, each with a title and a download icon. The models include: 'The Art House - Interior 01' by lionmessage, 'SWAN Miami Chair' by Danir, 'Mission-style club chair' by davidheim, '78413 Chair with Armrest Foru...' by KARE SketchUp, 'Herman Miller Ergon 3-Work Ch...' by Herman Miller, 'Chair Table' by Nurul K, 'A Modern-ized Classic Library D...' by Tim, 'Bürostuhl / Office Chair' by Adrian Q., and 'CHAIRS STACKING SHOWING S...' by JAW Functional Sculptures. The bottom row shows partial thumbnails of a rocking chair and a patterned chair.

Nowadays millions of 3D models in online repositories

Big Data in 3D

Growing crowd-sourced market for 3D models



.....

Big Data in 3D

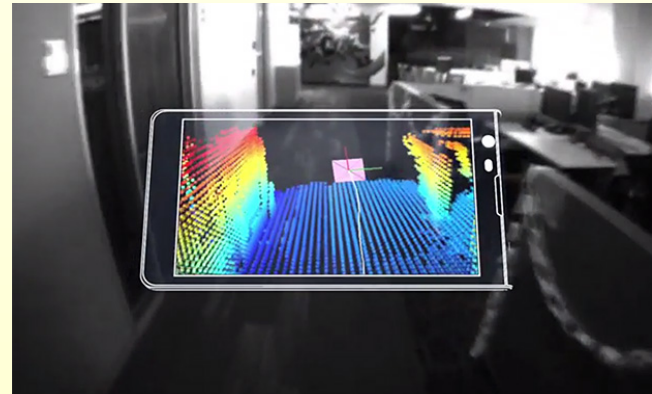
Growing crowd-sourced market for 3D models



**An opportunity for data-driven
3D visual computing**



Big Data in 3D



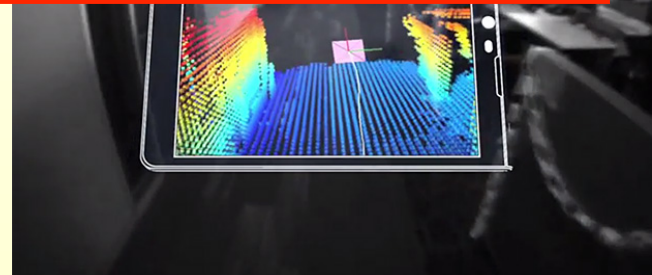
Big Data in 3D



KINECT™



Over 30,000,000 sold



airplane,aeroplane,plane

an aircraft that has a fixed wing and is powered by propellers or jets; 'the flight was delayed due to trouble with the airplane'

[ImageNet](#) [MetaData](#)

Choose a taxonomy:

ShapeNetCore ▾

- airplane,aeroplane,plane(11,4045)
- ashcan,trash can,garbage can,wastebin,...
- bag,traveling bag,traveling bag,grip,suitc...
- basket,handbasket(2,113)
- bathtub,bathing tub,bath,tub(0,856)
- bed(13,233)
- bench(5,1813)
- bicycle,bike,wheel,cycle(0,59)
- birdhouse(0,73)
- bookshelf(0,452)
- bottle(6,498)
- bowl(1,186)
- bus,autobus,coach,charabanc,double-de...
- cabinet(9,1571)
- camera,photographic camera(4,113)
- can,tin,tin can(2,108)
- cap(4,56)
- car,auto,automobile,machine,motorcar(18...
- chair(23,6778)
- clock(3,651)
- computer keyboard,keypad(0,65)
- dishwasher,dish washer,dishwashing ma...









































Synset Models

TreeMap

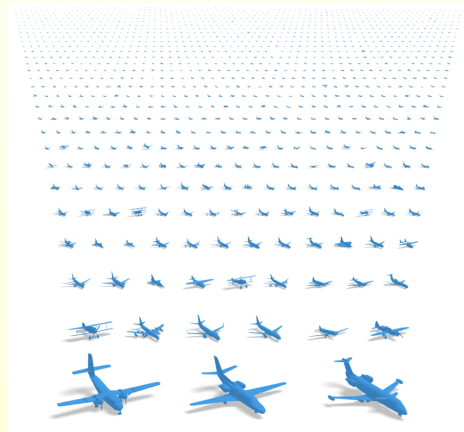
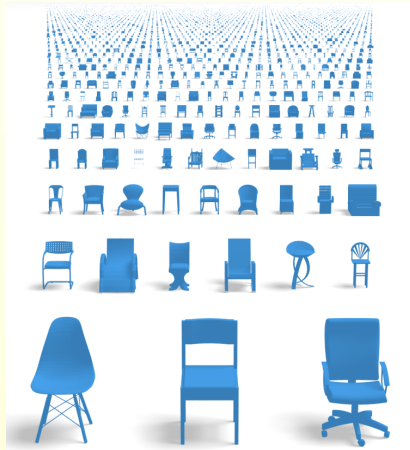
Stats

Displaying 1 to 160 of 4045

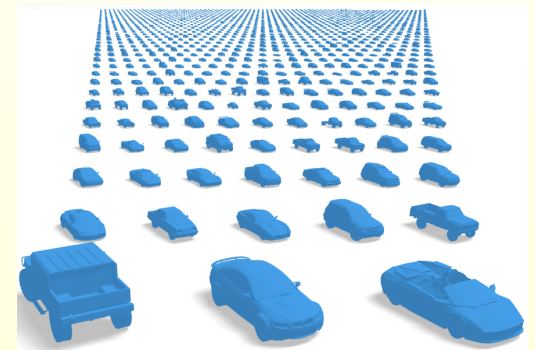
...

							
airplane	airplane	airplane	airplane	bomber	fighter	airliner	straight wing
							
airplane	airplane	airplane	propeller plane	airplane	airplane	airplane	bomber
							
airplane	airplane	airliner	delta wing	airplane	airplane	jet	jet
							
airplane	airplane	airplane	airplane	airplane	propeller plane	airplane	airplane
							
airplane	jet	airliner	fighter	fighter	airplane	airplane	airplane

ShapeNet: A Large-scale 3D Model Database



...

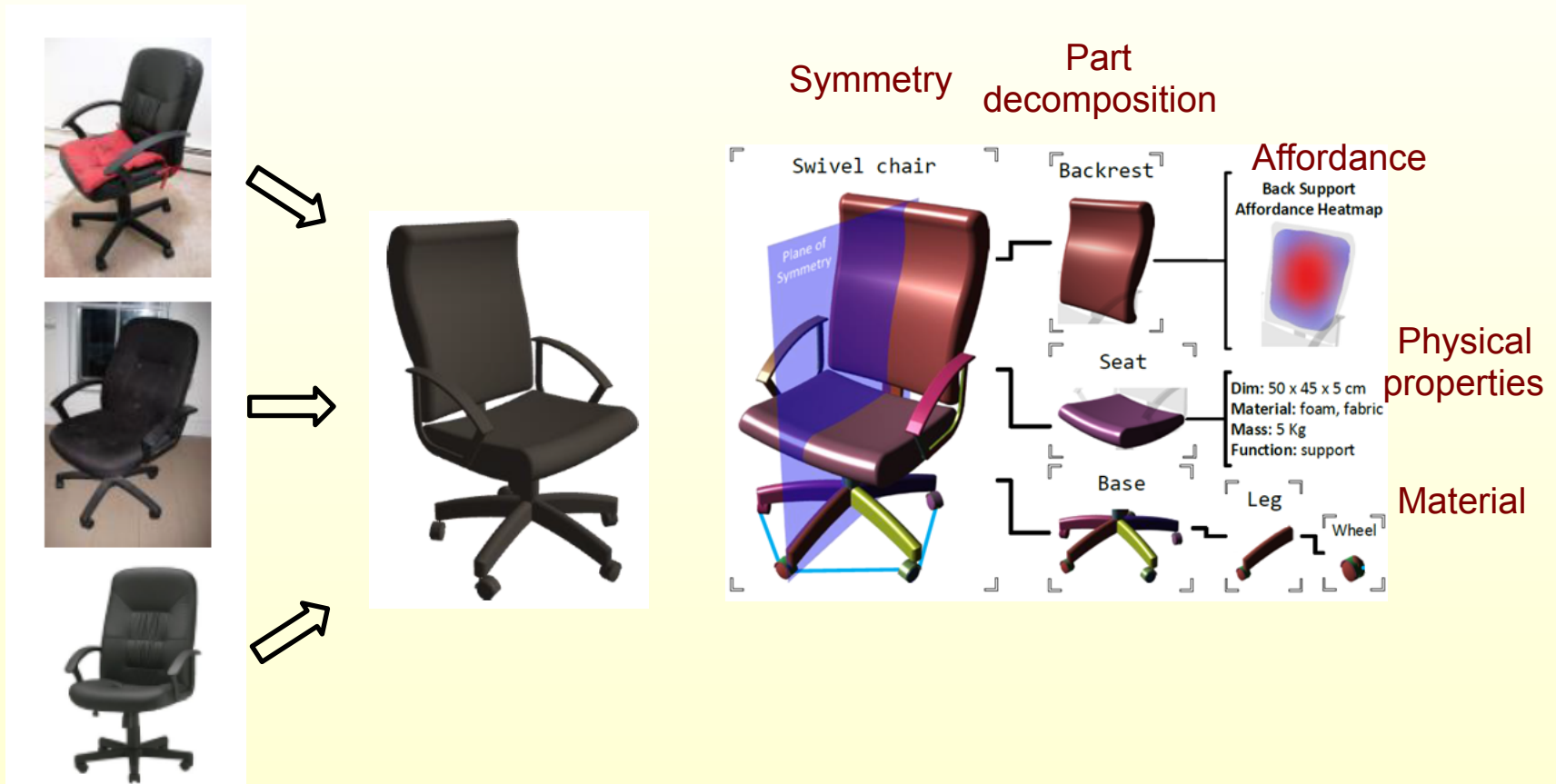


~**3 million** models in total

~**2,000** classes

Rich annotations

Unified Knowledge Representation in 3D

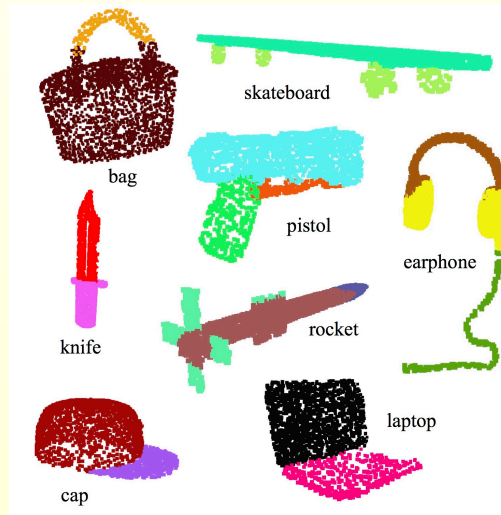


3D Deep Learning Tasks

3D geometry analysis



Classification



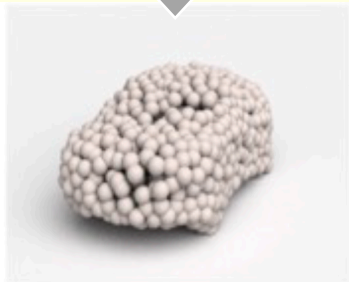
Parsing
(object/scene)



Correspondence

3D Deep Learning Tasks

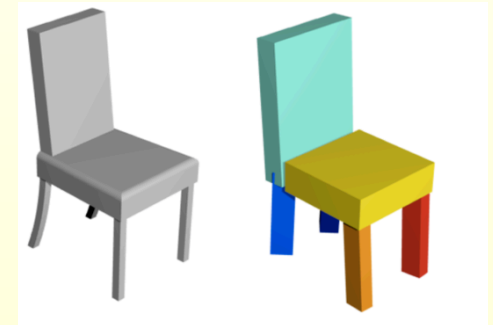
3D Synthesis



Monocular
3D reconstruction



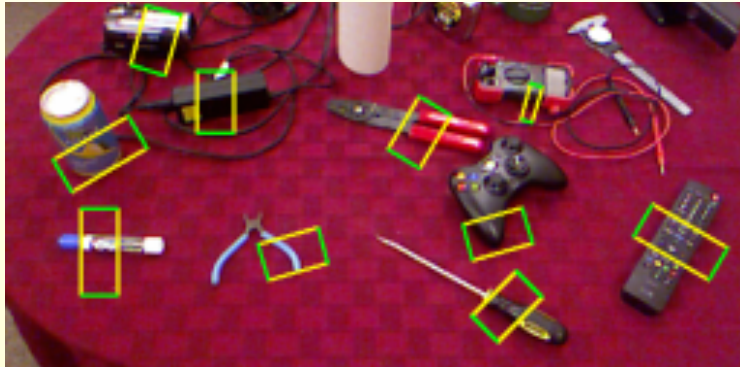
Shape completion



Shape modeling

3D Deep Learning Tasks

Robotics Applications



Robot Grasping Affordance Map

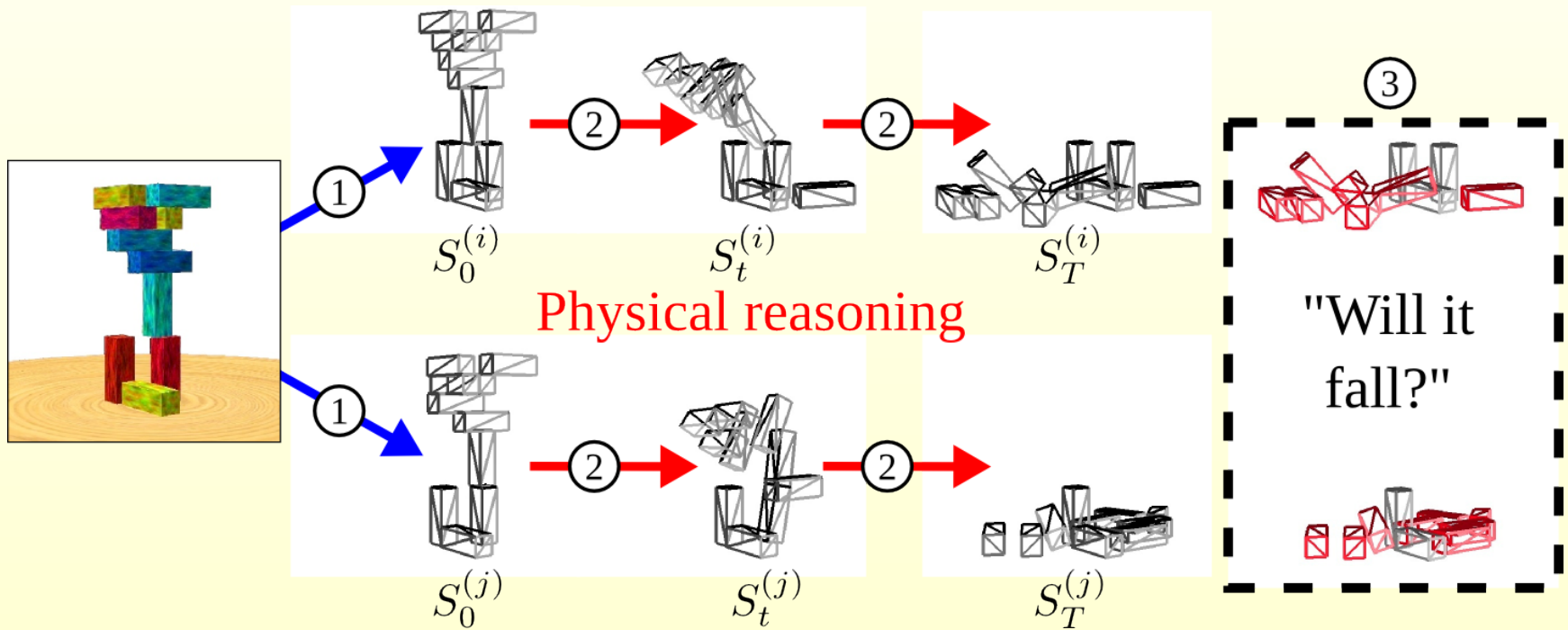


Human Interaction Affordance Map



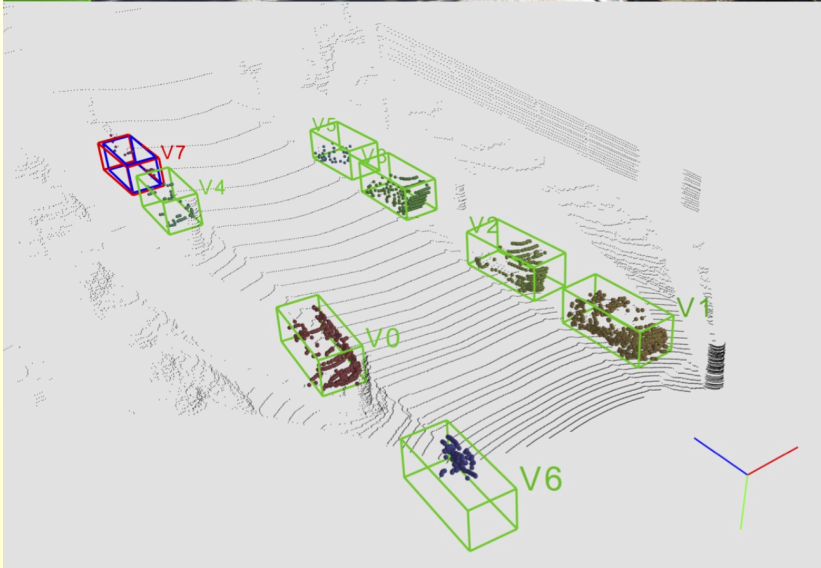
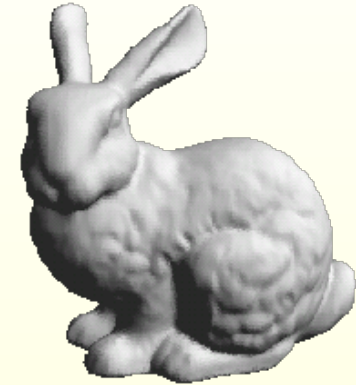
3D Deep Learning Tasks

Intuitive Physics based on 3D Understanding



3D Representation

◆ How to represent 3D data?



2D Representation

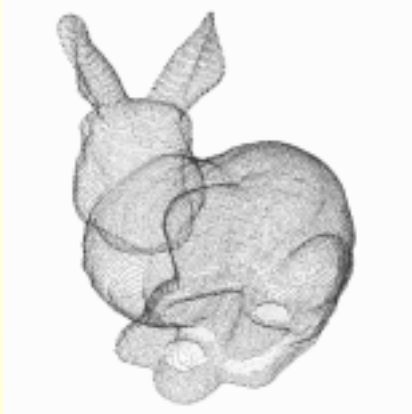
Images: canonical representation with regular data structure



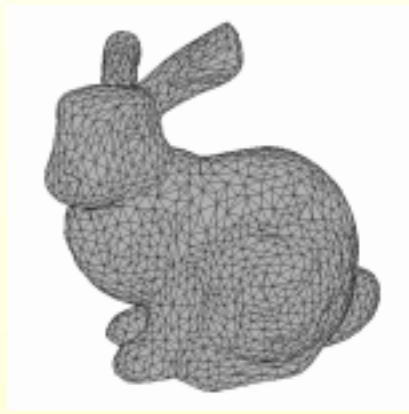
1	44	33	12	20	23	35	14
51	16	40	32	46	48	28	17
29	60	3	63	49	55	36	7
52	22	26	41	38	10	61	53
2	24	19	11	34	43	5	8
57	9	37	42	25	21	27	18
30	56	50	64	4	59	6	13
58	47	45	31	39	15	62	54

3D Representation

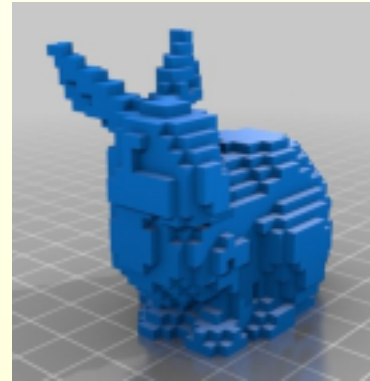
- ◆ Design good 3D representation for NN to consume



Point Cloud



Mesh



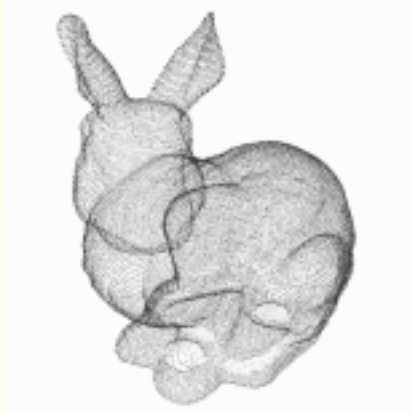
Volumetric



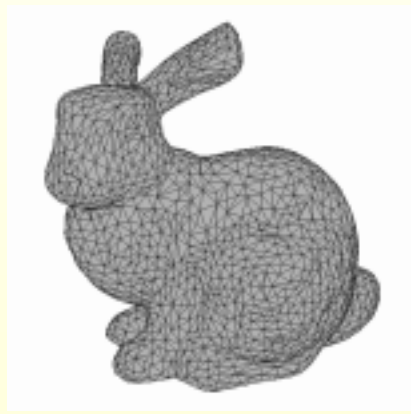
Image

3D Representation

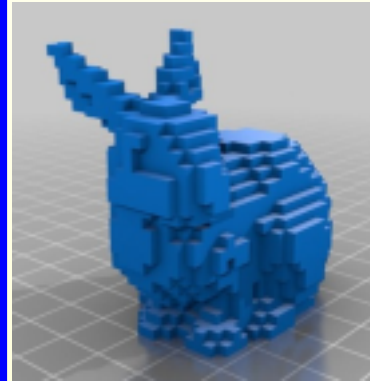
- ◆ Design good 3D representation for NN to consume



Point Cloud



Mesh



Volumetric

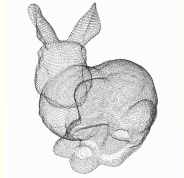


Image

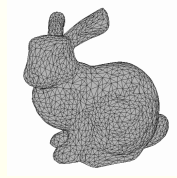
Irregular

Regular

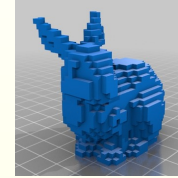
3D Representation



Point Cloud



Mesh

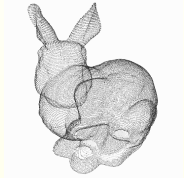


Volumetric

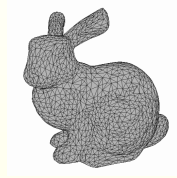


Image

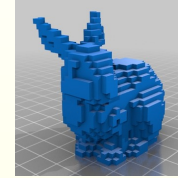
3D Representation



Point Cloud



Mesh



Volumetric



Image

Rawness

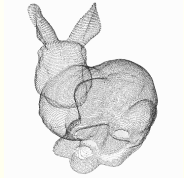
High

Low

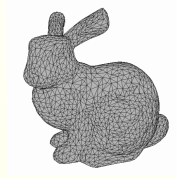
Medium

High

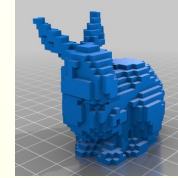
3D Representation



Point Cloud



Mesh



Volumetric



Image

Rawness

High

Low

Medium

High

**3D
Geometry**

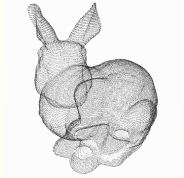
Yes

Yes

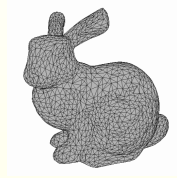
Yes

No

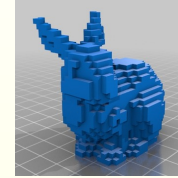
3D Representation



Point Cloud



Mesh



Volumetric



Image

Rawness

High

Low

Medium

High

**3D
Geometry**

Yes

Yes

Yes

No

**Compact-
ness**

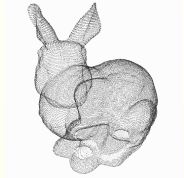
Yes

Yes

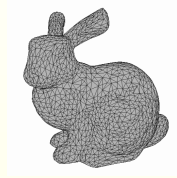
No

Medium

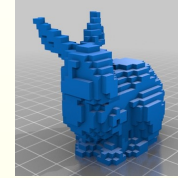
3D Representation



Point Cloud



Mesh



Volumetric



Image

Rawness

High

Low

Medium

High

**3D
Geometry**

Yes

Yes

Yes

No

**Compact-
ness**

Yes

Yes

No

Medium

**Data
Structure**

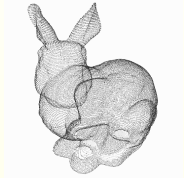
Set
{ X_i }

Graph
(V , E)

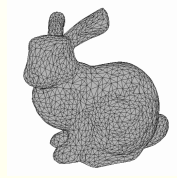
Array
 X_{ijk}

Array
 X_{ij}

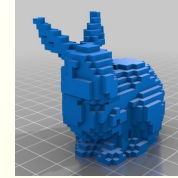
3D Representation



Point Cloud



Mesh



Volumetric



Image

Rawness

High

Low

Medium

High

3D
Geometry

Yes

Yes

Yes

No

Compact-
ness

Yes

Yes

No

Medium

Data
Structure

Set
{ X_i }

Graph
(V, E)

Array
 X_{ijk}

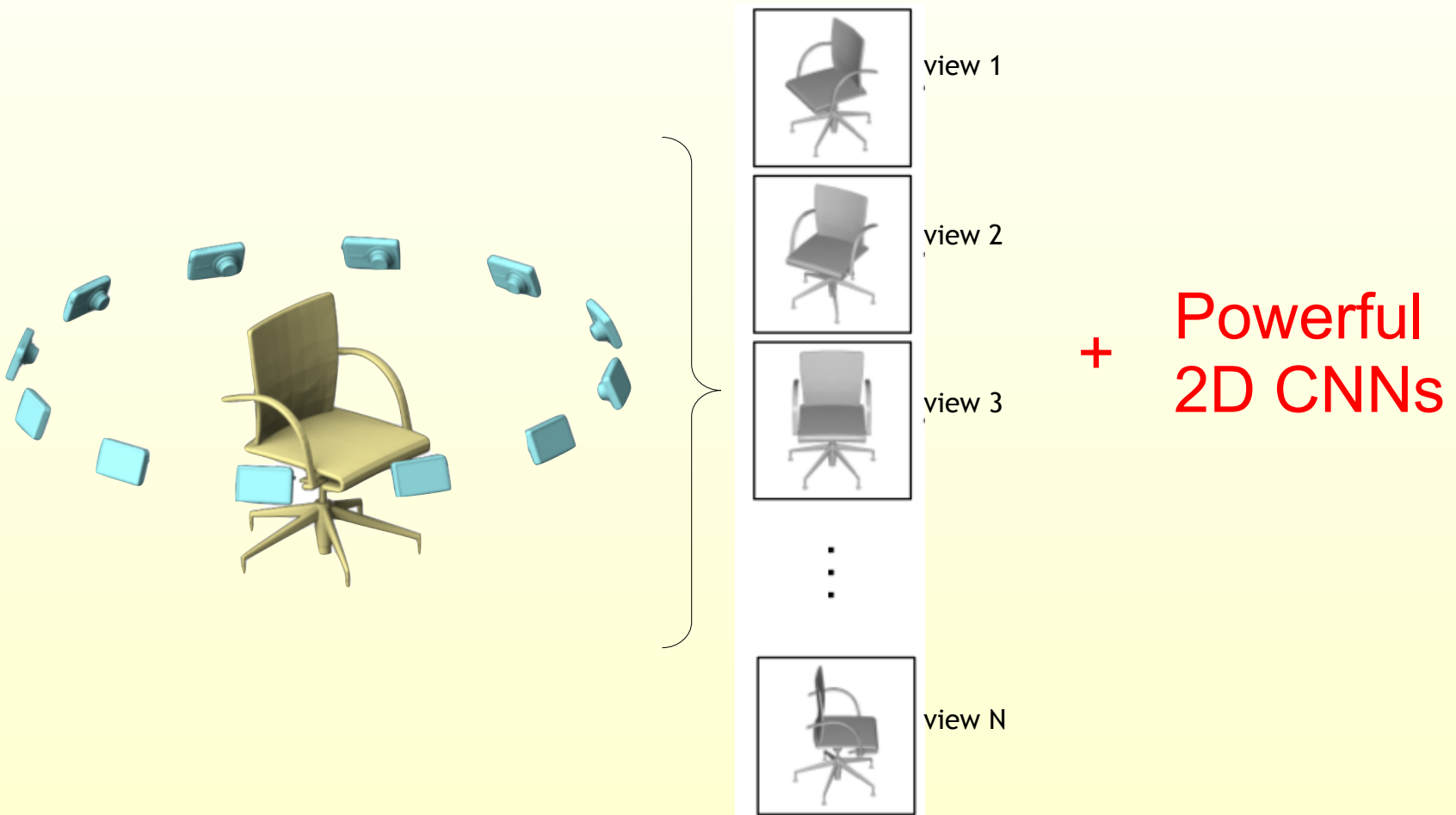
Array
 X_{ij}

Again, no free lunch!

Agenda

- ◆ Today
 - ◆ Deep Learning Intro
 - ◆ 3D Deep Learning
 - ◆ **Multi-view CNNs**
 - ◆ Volumetric CNNs
- ◆ Later lectures
 - ◆ Point-cloud NNs (May 21)
 - ◆ Graph CNNs (May 24)

Multi-view Representation



Multi-view Representation

◆ Pros

- ◆ Regular grids representation
- ◆ Easy to obtain the input images
- ◆ Easy to input to Neural Nets
- ◆ Leverage state-of-the-art 2D CNNs

◆ Cons

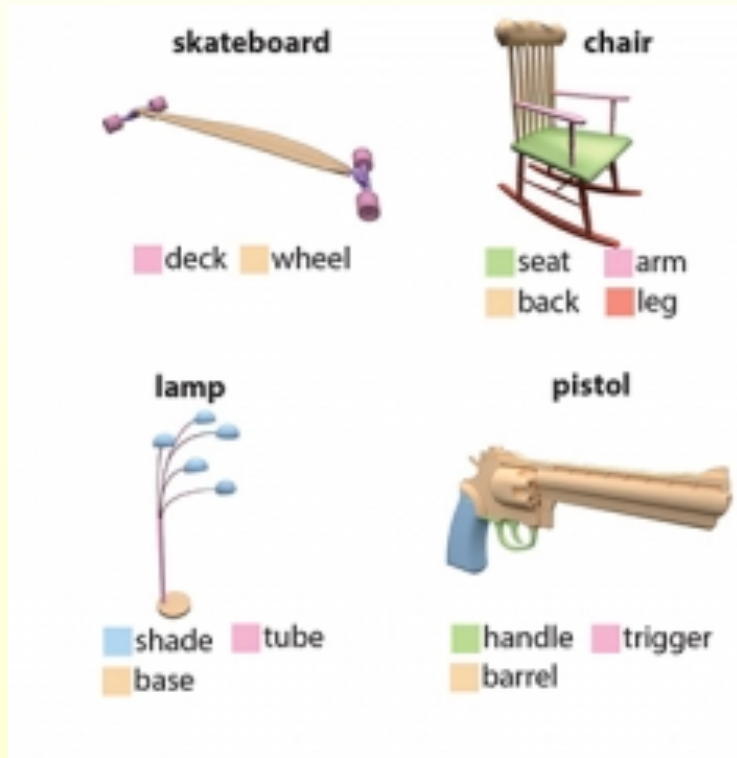
- ◆ Background clutters / lightings / etc.
- ◆ No access to the original 3D content
- ◆ Each view only contains partial information and aggregation is needed

Multi-view Representation

- ◆ Classification
- ◆ Segmentation



This is a chair!



Multi-view CNNs: Classification



This is a chair!

Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

Image Credits: Hang Su

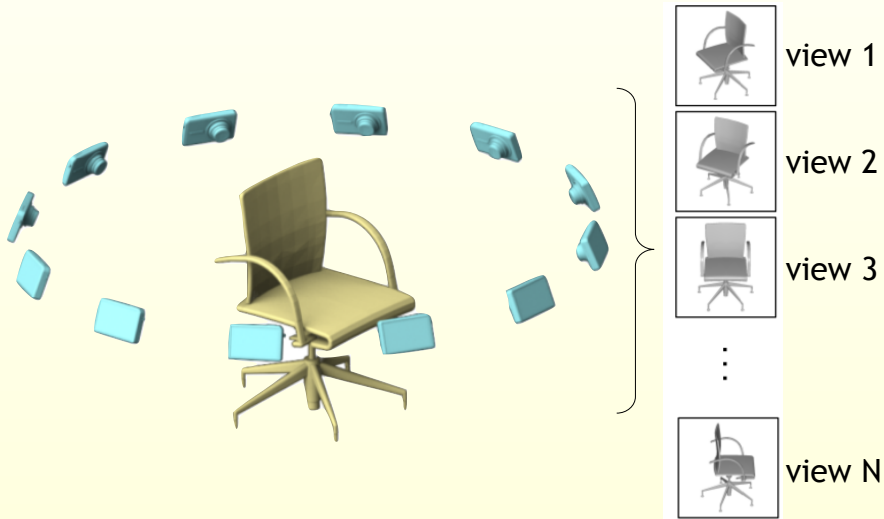
Multi-view CNNs: Classification



Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

Image Credits: Hang Su

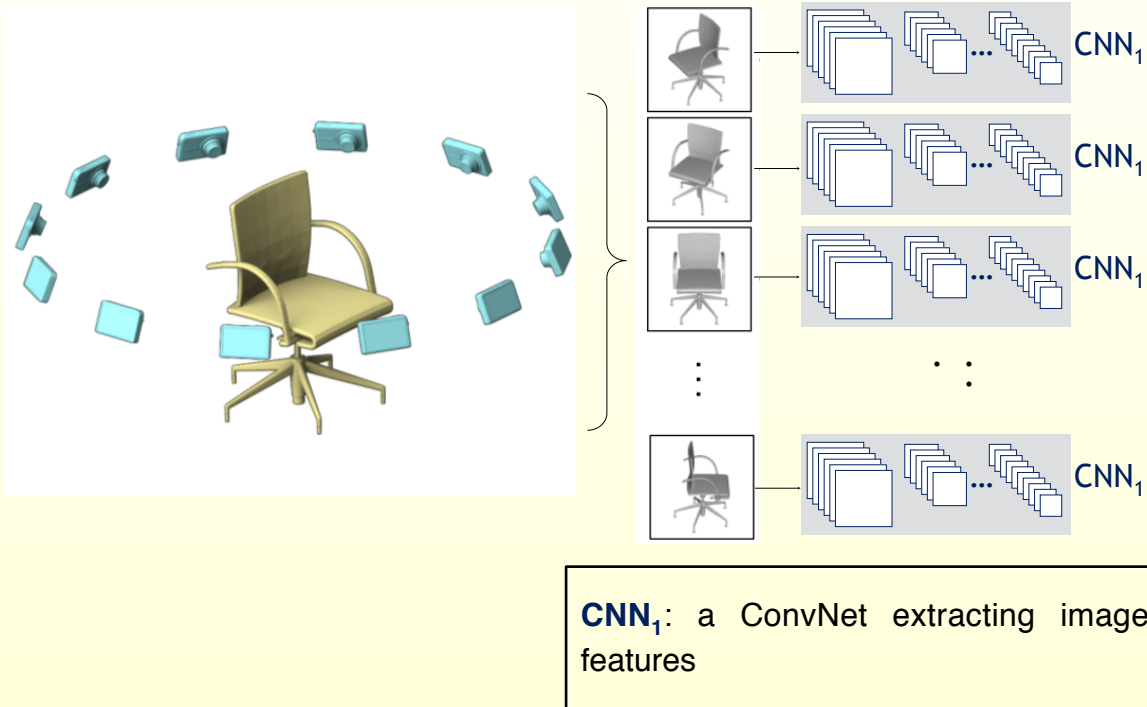
Multi-view CNNs: Classification



Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

Image Credits: Hang Su

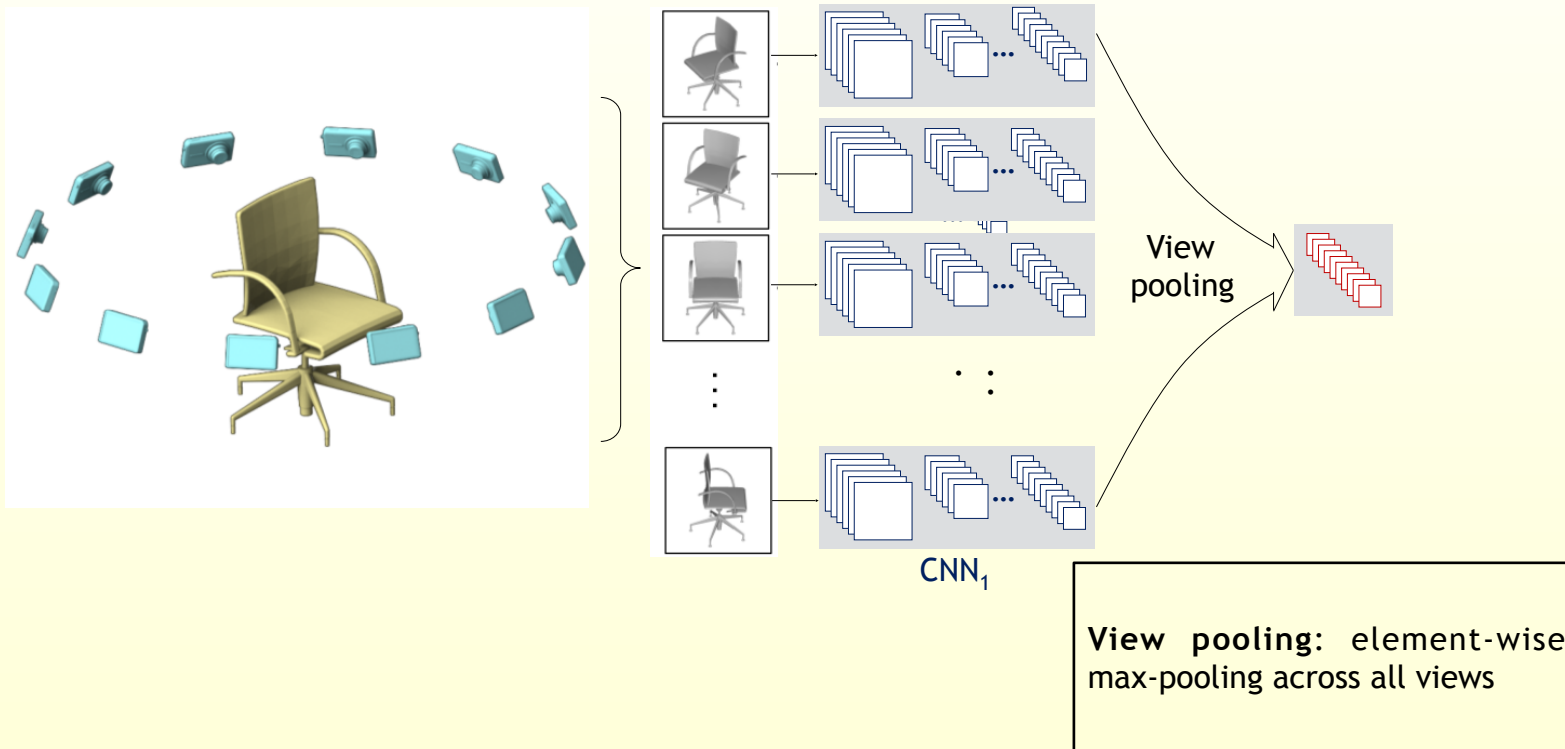
Multi-view CNNs: Classification



Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

Image Credits: Hang Su

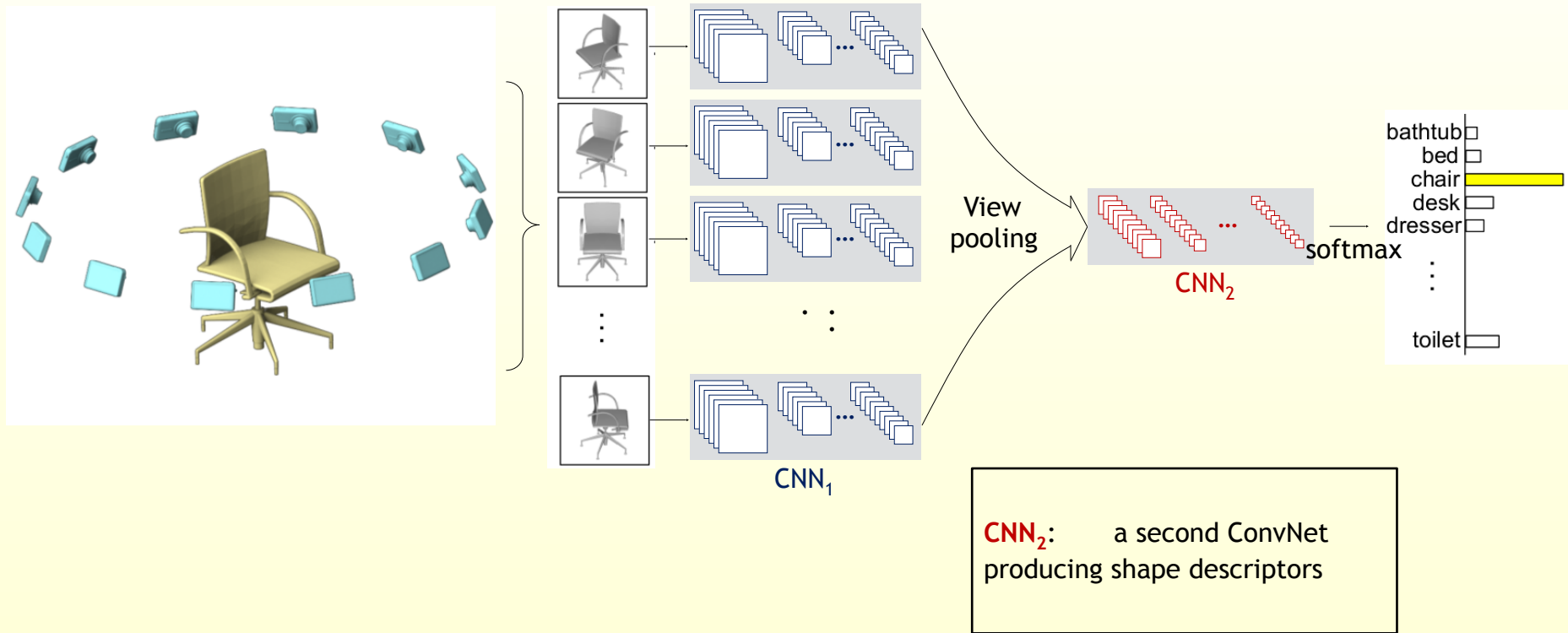
Multi-view CNNs: Classification



Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

Image Credits: Hang Su

Multi-view CNNs: Classification



Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, "**Multi-view Convolutional Neural Networks for 3D Shape Recognition**", *Proceedings of ICCV 2015*

Image Credits: Hang Su

Multi-view CNNs: Classification

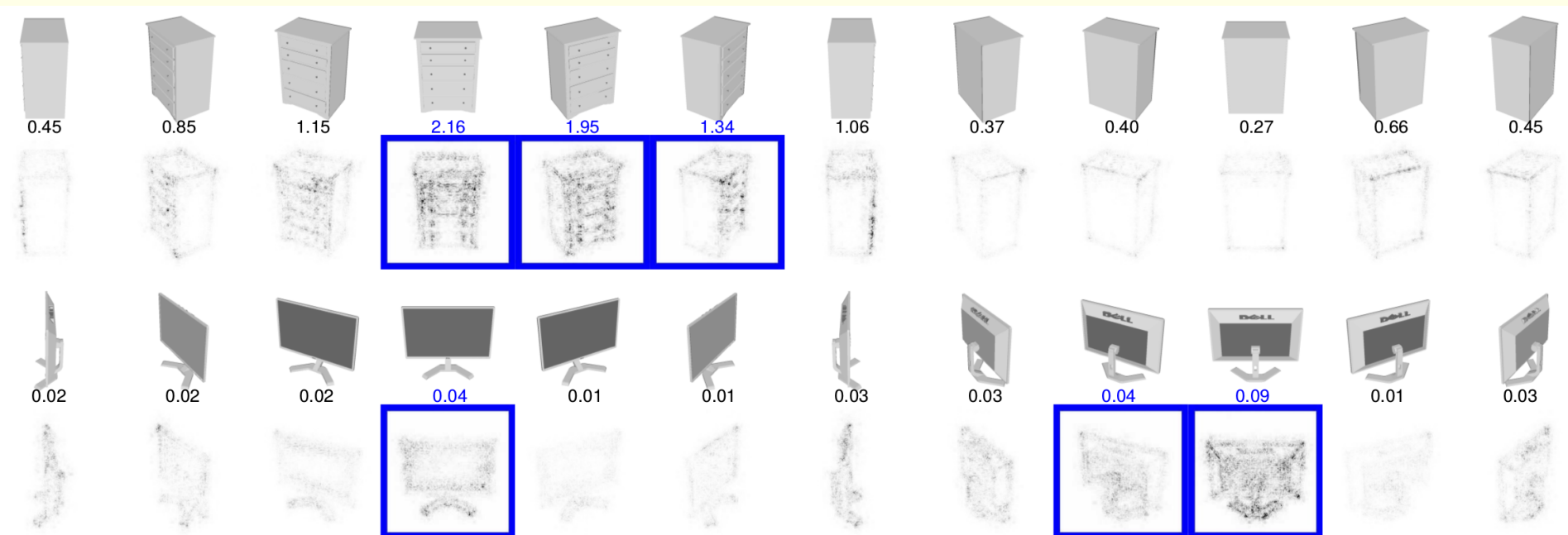
On **ModelNet40**, compared against:

- 3 existing methods:
SPH, LFD, 3D ShapeNets
- 2 strong baselines:
Fisher vectors, CNN

Method	Classification (Accuracy)	Retrieval (mAP)
SPH [16]	68.2%	33.3%
LFD [5]	75.5%	40.9%
3D ShapeNets [37]	77.3%	49.2%
FV, 12 views	84.8%	43.9%
CNN, 12 views	88.6%	62.8%
MVCNN, 12 views	89.9%	70.1%
MVCNN+metric, 12 views	89.5%	80.2%
MVCNN, 80 views	90.1%	70.4%
MVCNN+metric, 80 views	90.1%	79.5%

Multi-view CNNs: Classification

$$[\omega_1, \omega_2 \dots \omega_K] = \left[\frac{\partial F_c}{\partial I_1} \Big|_S, \frac{\partial F_c}{\partial I_2} \Big|_S \dots \frac{\partial F_c}{\partial I_K} \Big|_S \right]$$



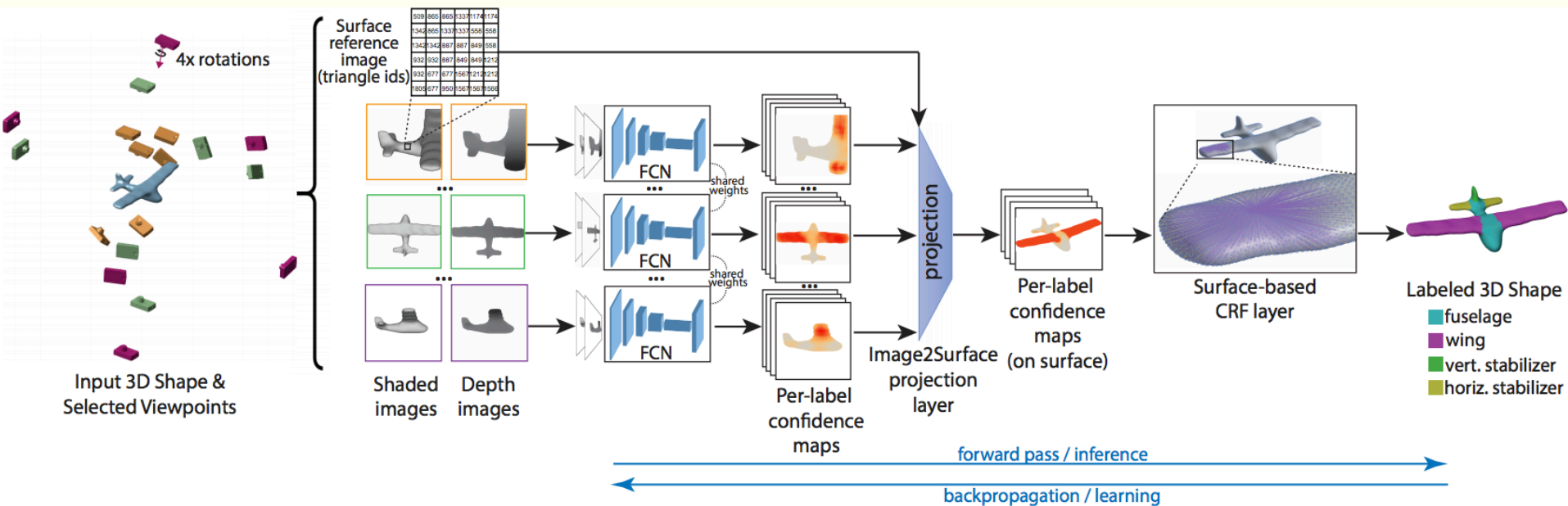
Multi-view CNNs: Segmentation



Li Yi, Vladimir G. Kim, Duygu Ceylan, I-Chao Shen, Mengyuan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, Leonidas J. Guibas, "A Scalable Active Framework for Region Annotation in 3D Shape Collections", *SIGGRAPH Asia 2016*

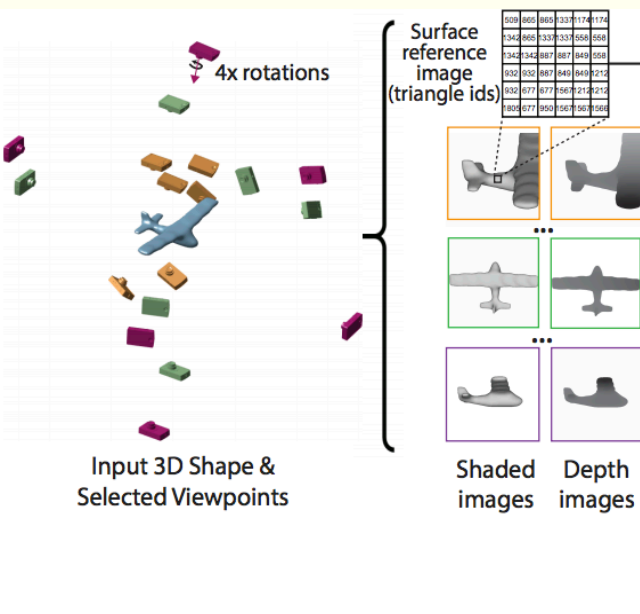
Image Credits: Eric Yi

Multi-view CNNs: Segmentation



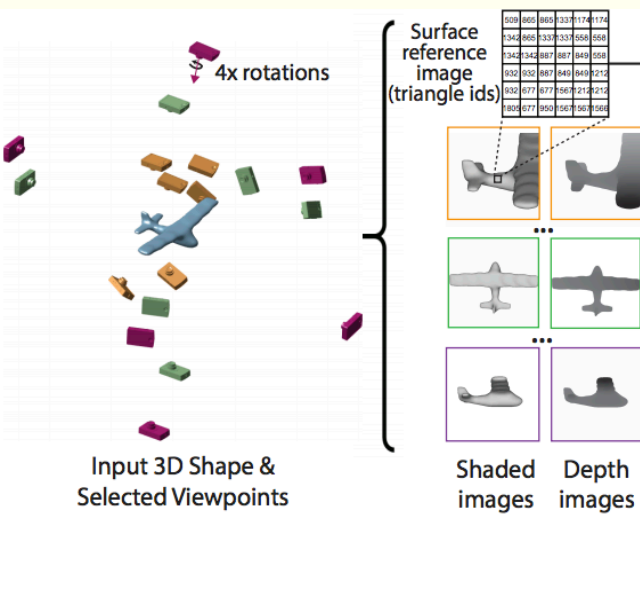
Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
 “3D Shape Segmentation with Projective Convolutional Networks”,
 CVPR2017

Multi-view CNNs: Segmentation



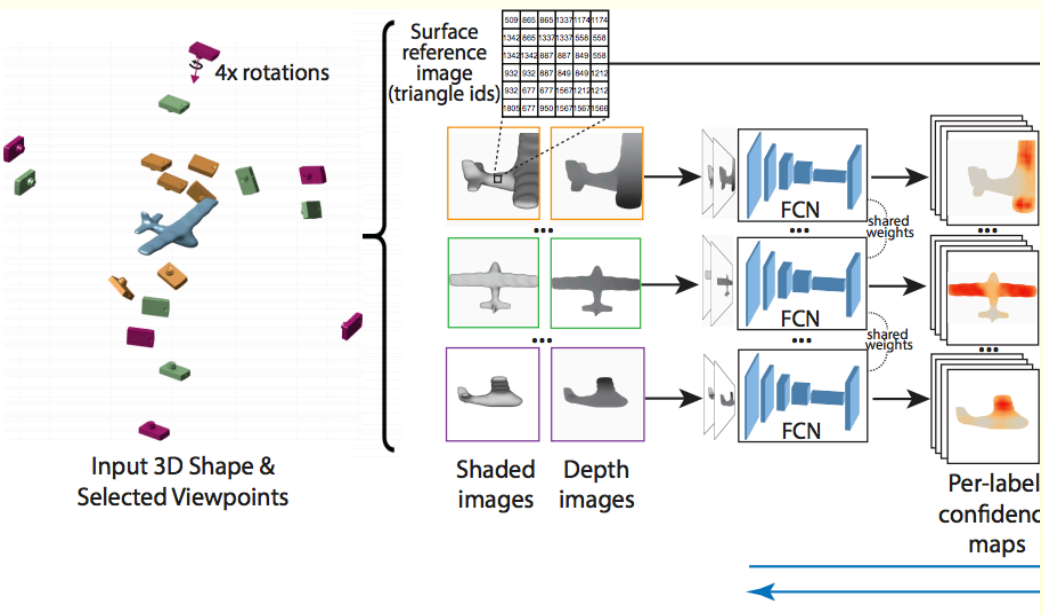
Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
“3D Shape Segmentation with Projective Convolutional Networks”,
CVPR2017

Multi-view CNNs: Segmentation



- ◆ Parts have different sizes
- ◆ Sample from different distances
- ◆ Make sure all points are visible
- ◆ Compute points coverage
- ◆ Make the boundaries sharp
- ◆ Use depth images

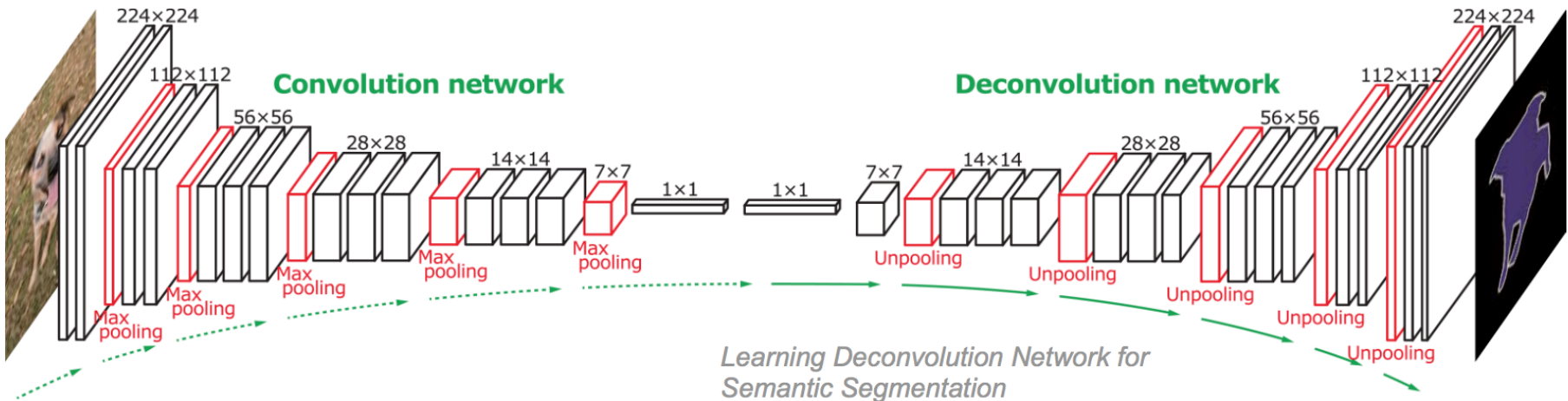
Multi-view CNNs: Segmentation



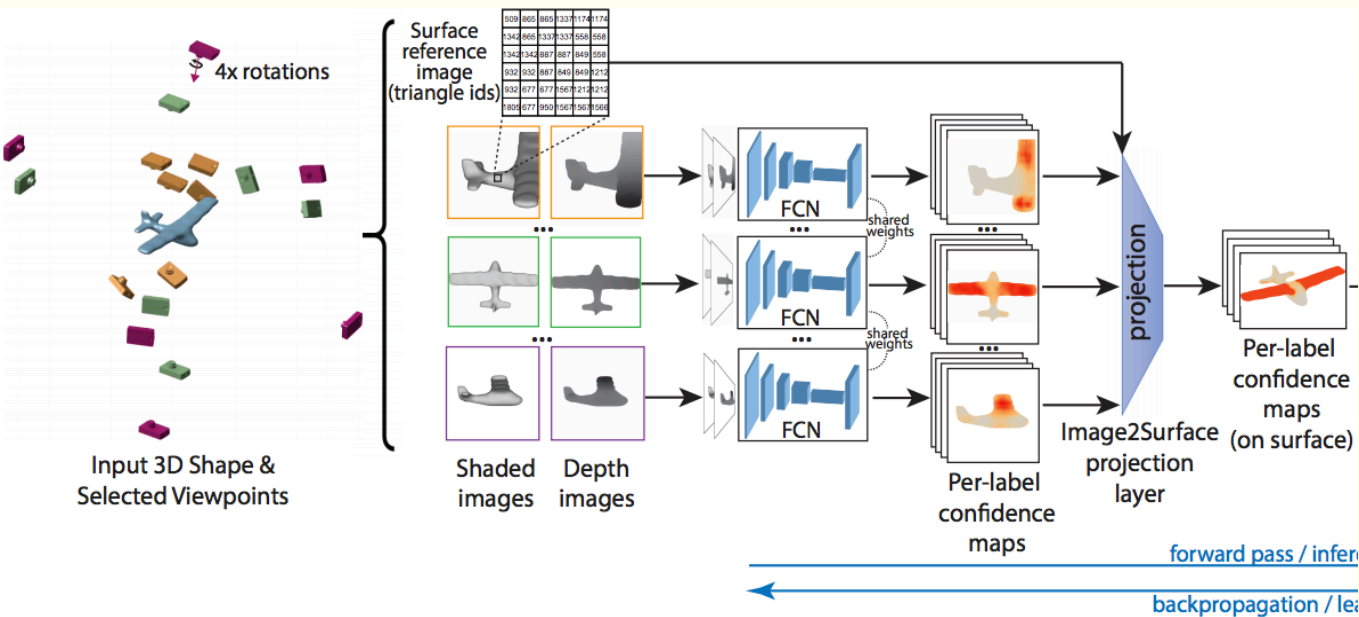
Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
“3D Shape Segmentation with Projective Convolutional Networks”,
CVPR2017

Fully Convolutional Network

Segmentation:

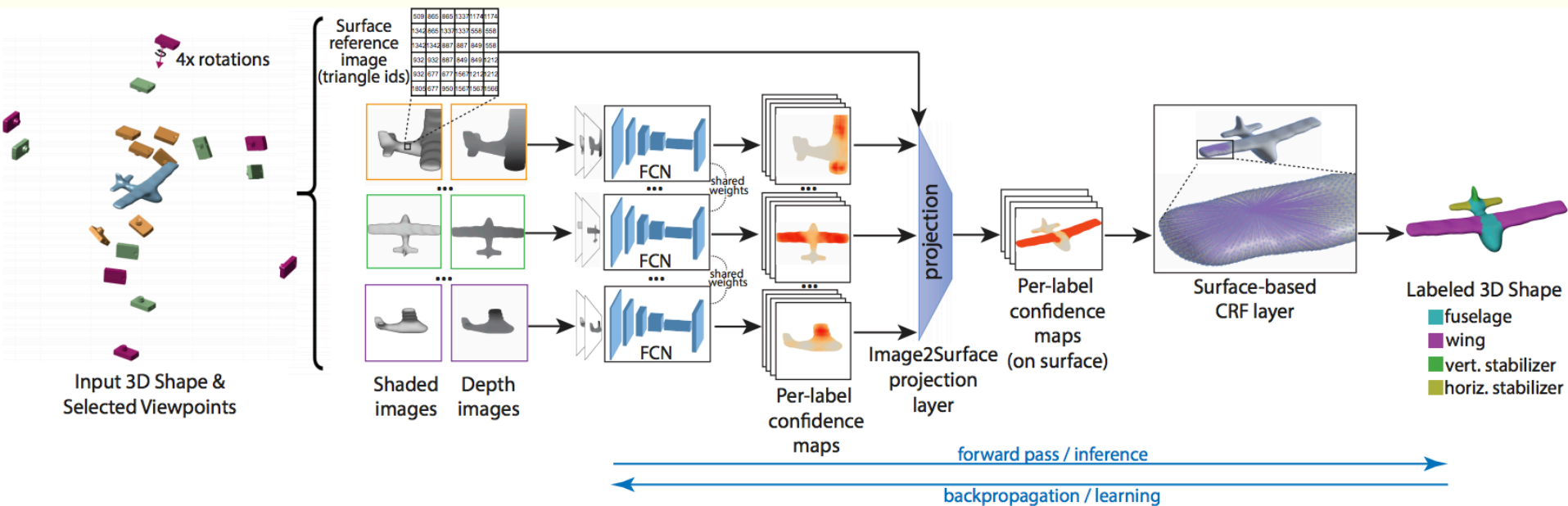


Multi-view CNNs: Segmentation



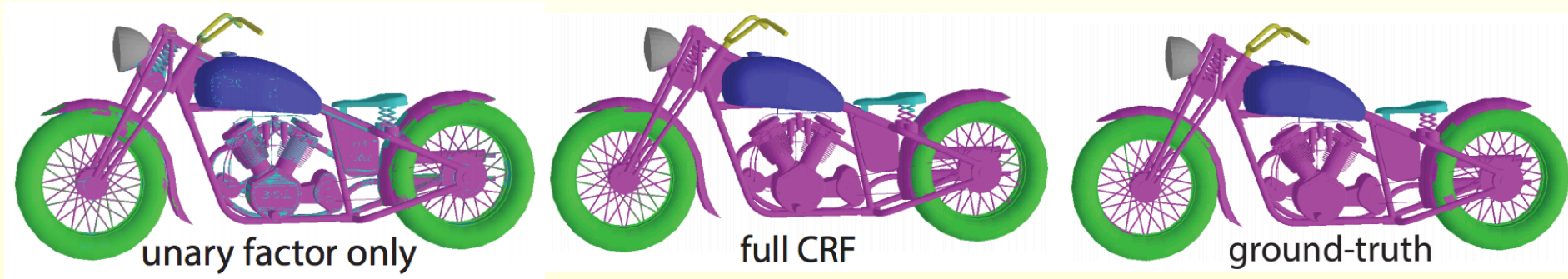
Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
 “3D Shape Segmentation with Projective Convolutional Networks”,
 CVPR2017

Multi-view CNNs: Segmentation



Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
 “3D Shape Segmentation with Projective Convolutional Networks”,
 CVPR2017

Multi-view CNNs: Segmentation



Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, Siddhartha Chaudhuri,
“3D Shape Segmentation with Projective Convolutional Networks”,
CVPR2017

Multi-view Representation Issues

- ◆ Cannot process invisible points
- ◆ View-based representations have redundancy
- ◆ Aggregation of view representations via max-pooling may lose information
- ◆ Properly fusing information across viewpoints is not incorporated in the network (not trivial).

Agenda

- ◆ Today
 - ◆ Deep Learning Intro
 - ◆ 3D Deep Learning
 - ◆ Multi-view CNNs
 - ◆ **Volumetric CNNs**
- ◆ Later lectures
 - ◆ Point-cloud NNs (May 21)
 - ◆ Graph CNNs (May 24)

Volumetric Representation

Images: canonical representation with regular data structure



1	44	33	12	20	23	35	14
51	16	40	32	46	48	28	17
29	60	3	63	49	55	36	7
52	22	26	41	38	10	61	53
2	24	19	11	34	43	5	8
57	9	37	42	25	21	27	18
30	56	50	64	4	59	6	13
58	47	45	31	39	15	62	54

Volumetric Representation

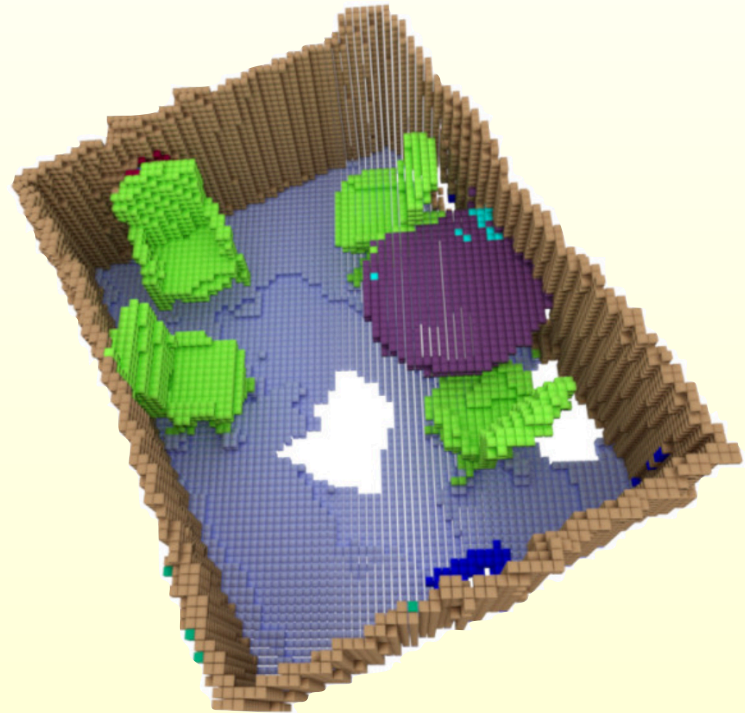
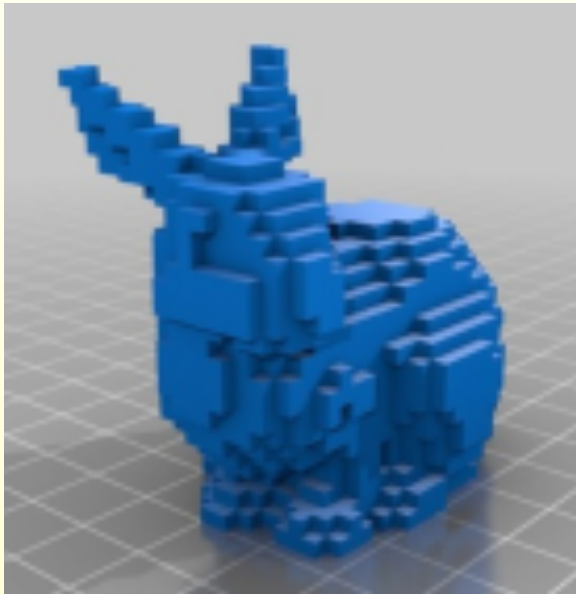
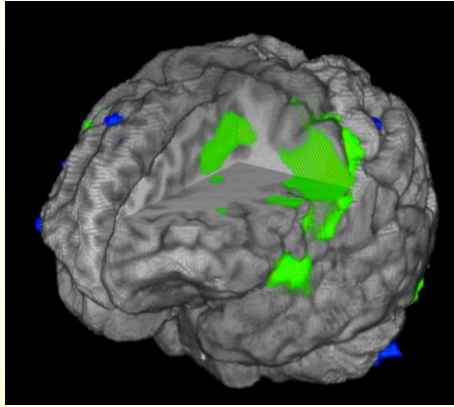
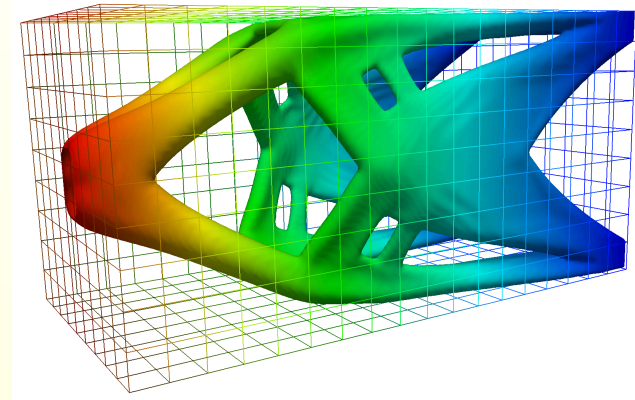


Image Credits: Scannet

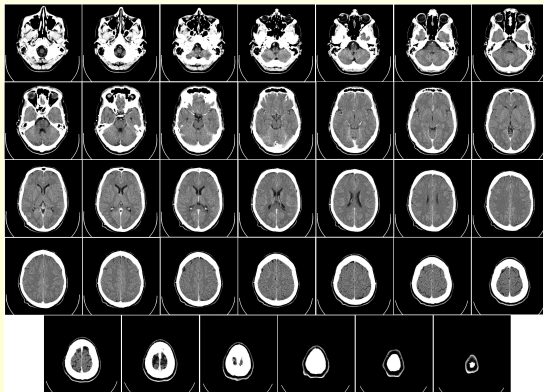
Volumetric Representation



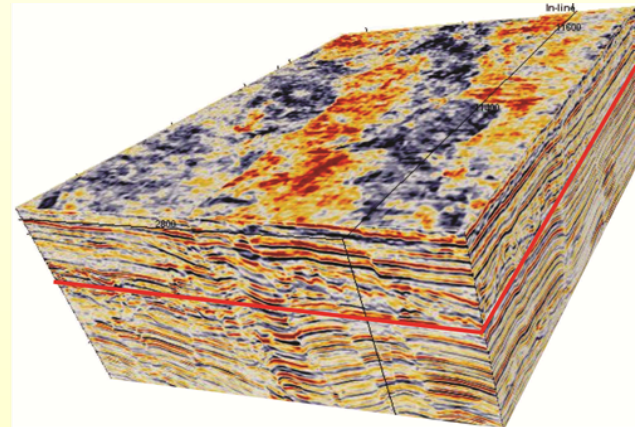
fMRI



Manufacturing
(finite-element analysis)

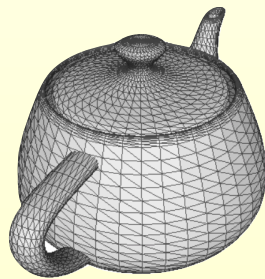
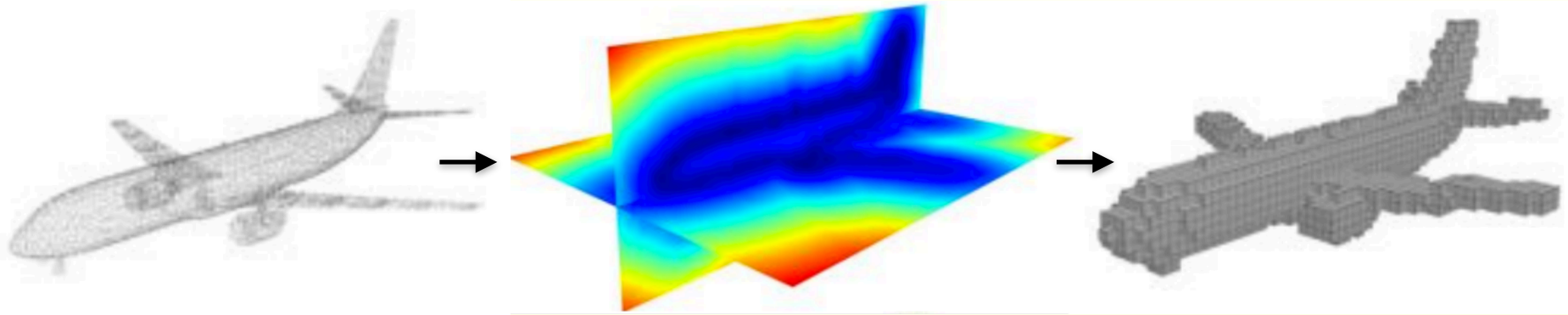


CT

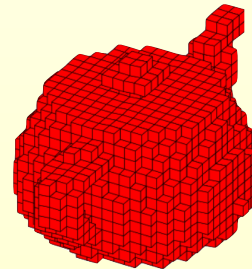


Geology

Volumetric Representation



Polygon Mesh



Occupancy Grid
30x30x30

Volumetric Representation

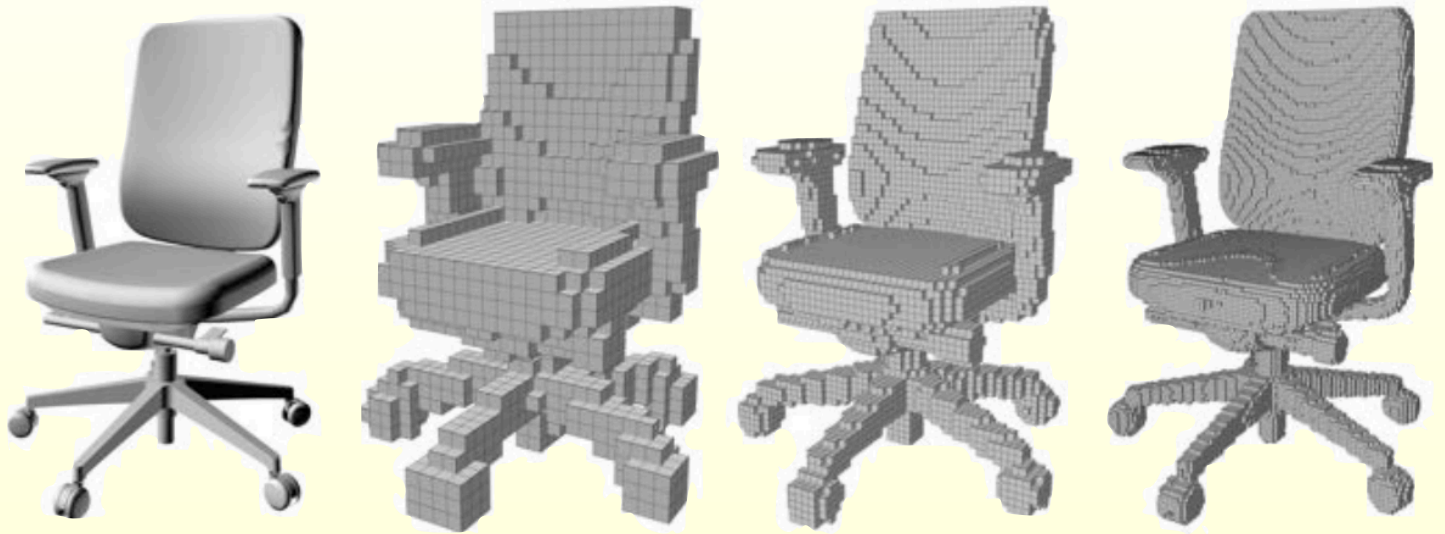
◆ Pros

- ◆ Regular grids representation
- ◆ Intuitive extension of images
- ◆ Easy to input to Neural Nets
- ◆ Each grid can have many feature inputs

◆ Cons

- ◆ Need to convert from point clouds scans
- ◆ Surface voxels? / solid voxels?
- ◆ Space / Time Complexities

Volumetric Representation



$\frac{\#occupied\ grid}{\#total\ grid}$

Occupancy: 10.41%

5.09%

2.41%

Resolution: 32

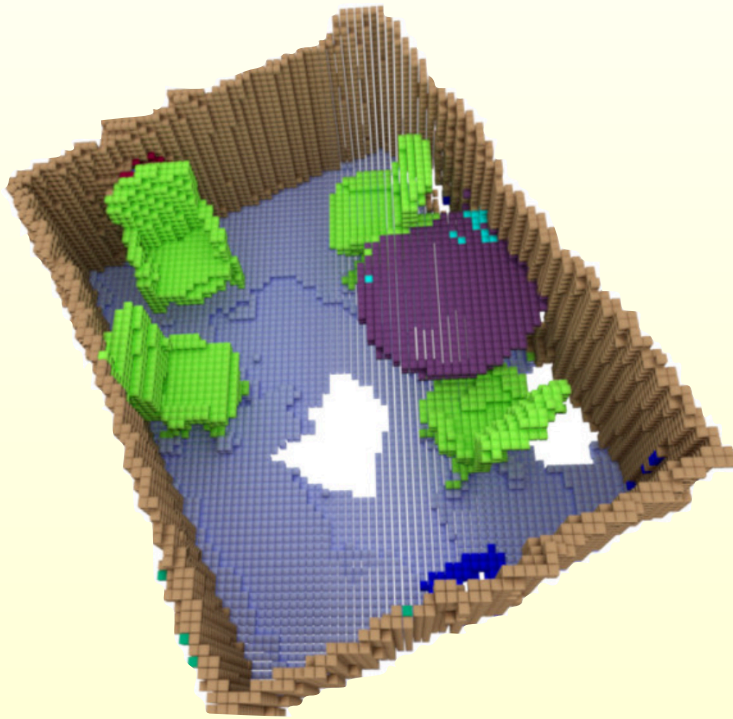
64

128

Resolution N

$O(N^3)$

Volumetric Representation

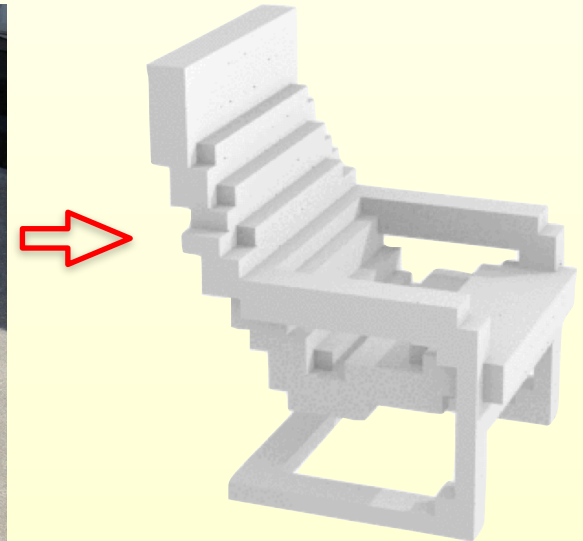
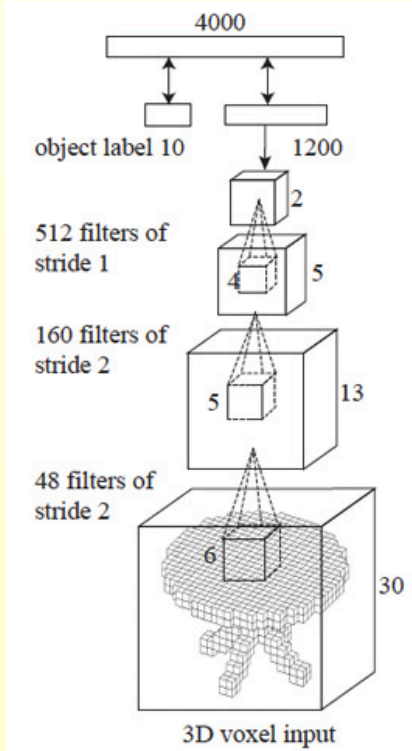


< 10%

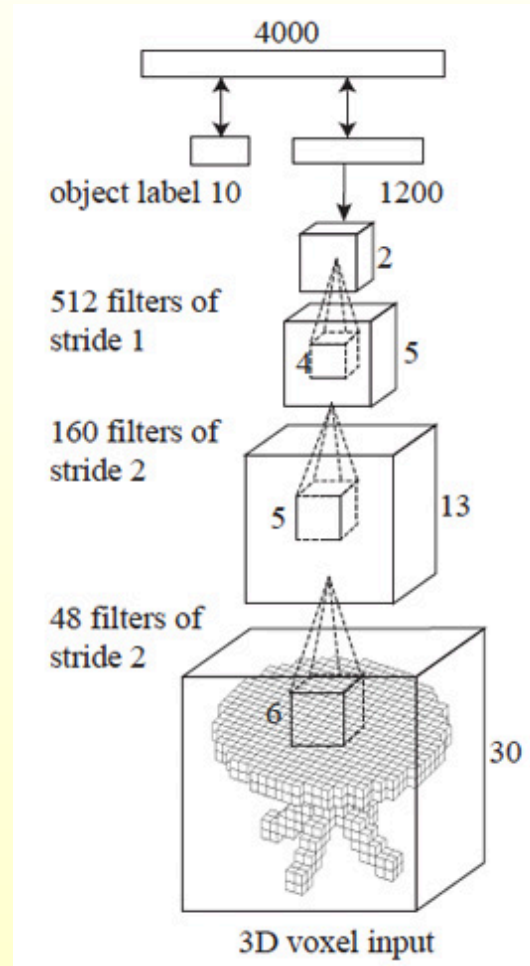
Image Credits: Scannet

Multi-view Representation

- ◆ Classification
- ◆ Reconstruction

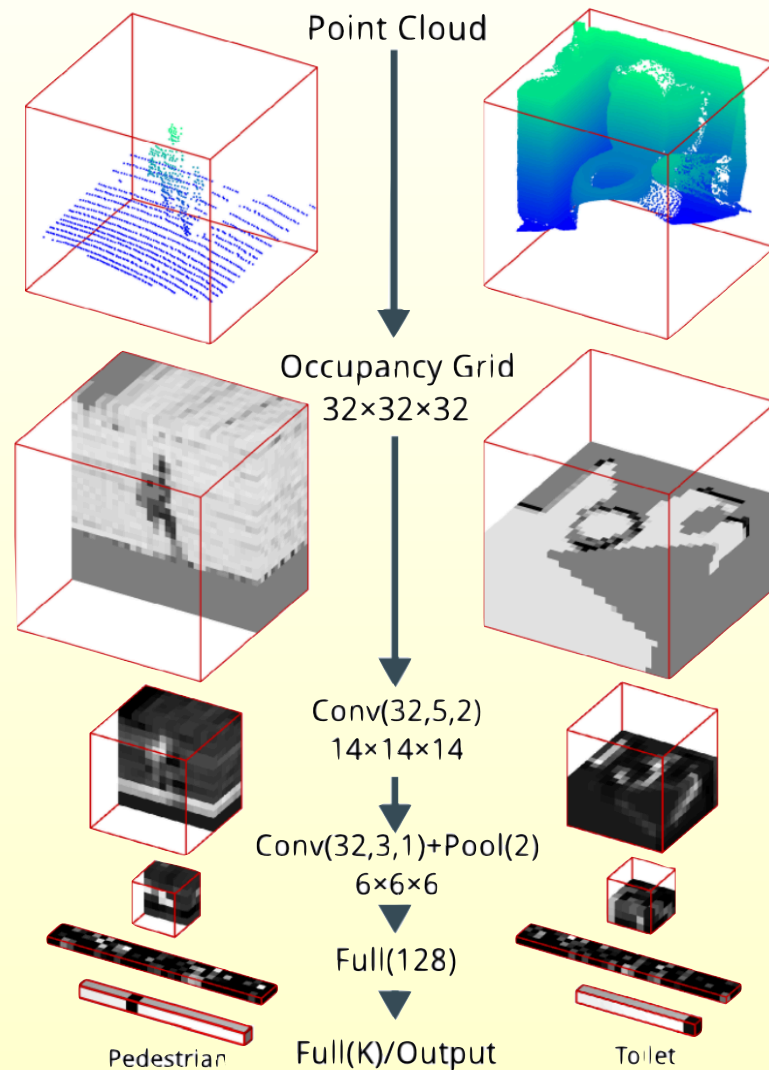


Volumetric Representation: Classification



Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang and J. Xiao,
“3D ShapeNets: A Deep Representation for Volumetric Shape Modeling”,
CVPR2015

Volumetric Representation: Classification

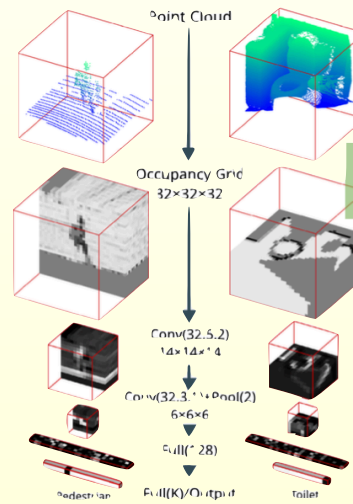
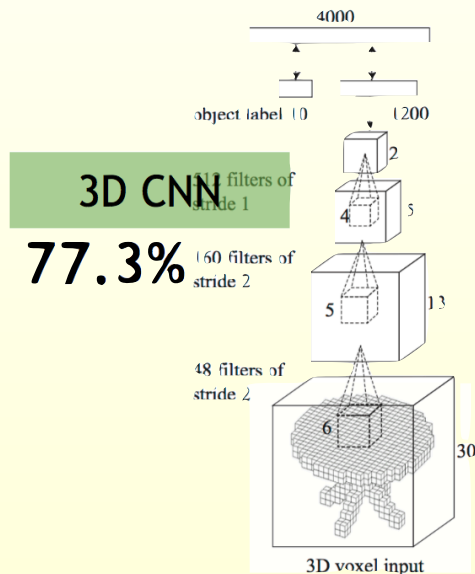


Daniel Maturana
and Sebastian
Scherer,
“**VoxNet: A 3D
Convolutional
Neural Network for
Real-Time Object
Recognition**”,
IROS2015

Volumetric Representation: Classification

3DShapeNets from Princeton
CVPR 2015 Oral

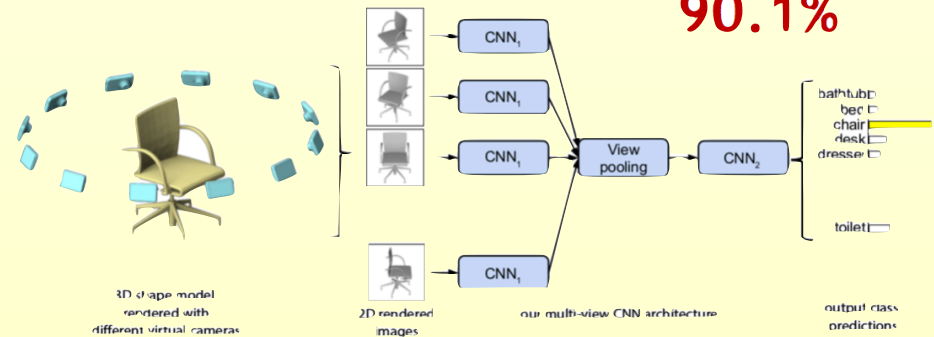
VoxNet from CMU Robotics
IEEE/RSJ 2015



Rendering +
2D CNN

90.1%

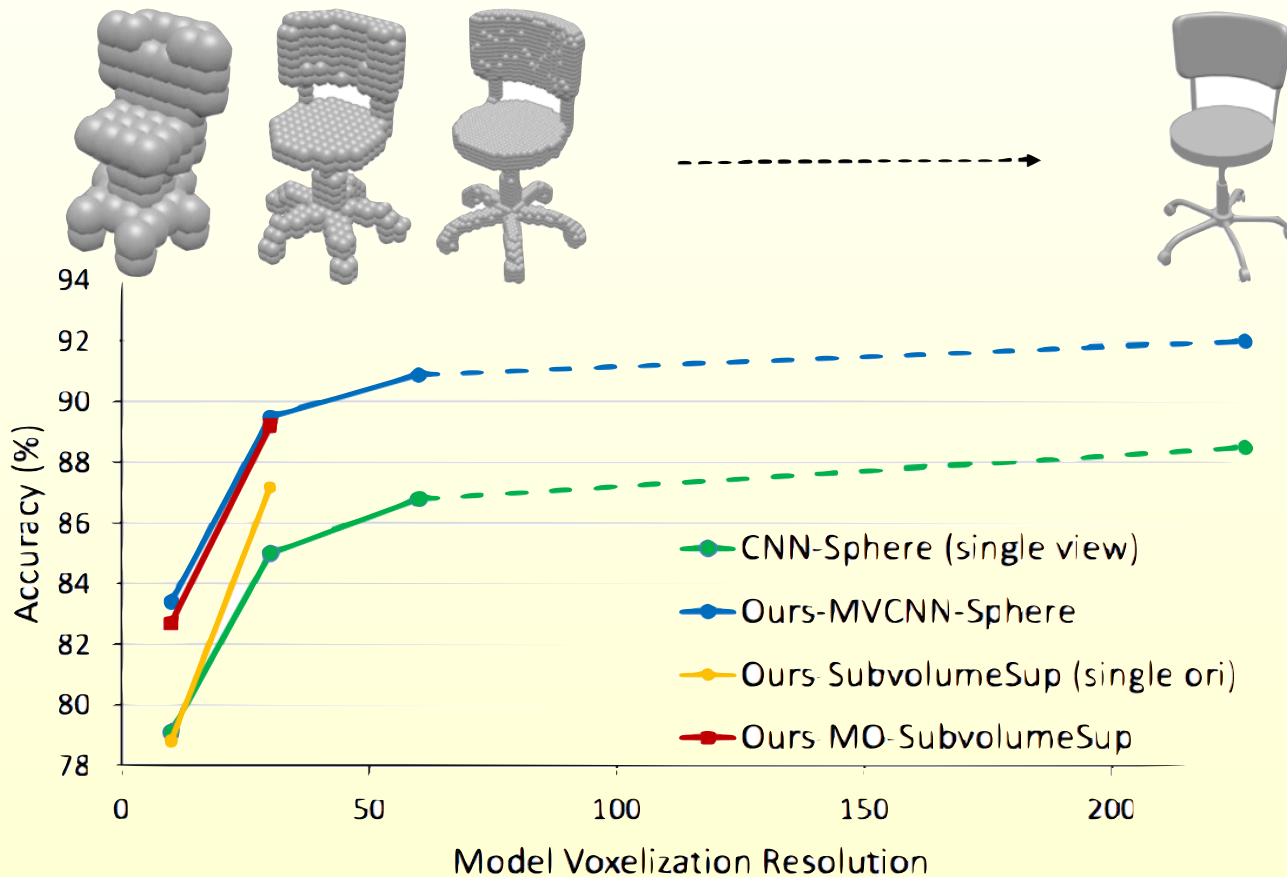
MVCNN from UMass
ICCV 2015



Volumetric Representation: Classification

- ◆ Vanilla 3D CNN methods perform worse than Multi-view CNNs
 - ◆ Resolution is limited (information loss)
 - ◆ Computation is more expensive than 2D
 - ◆ Have no colors / shading effects
 - ◆ Aliasing

Volumetric Representation: Classification



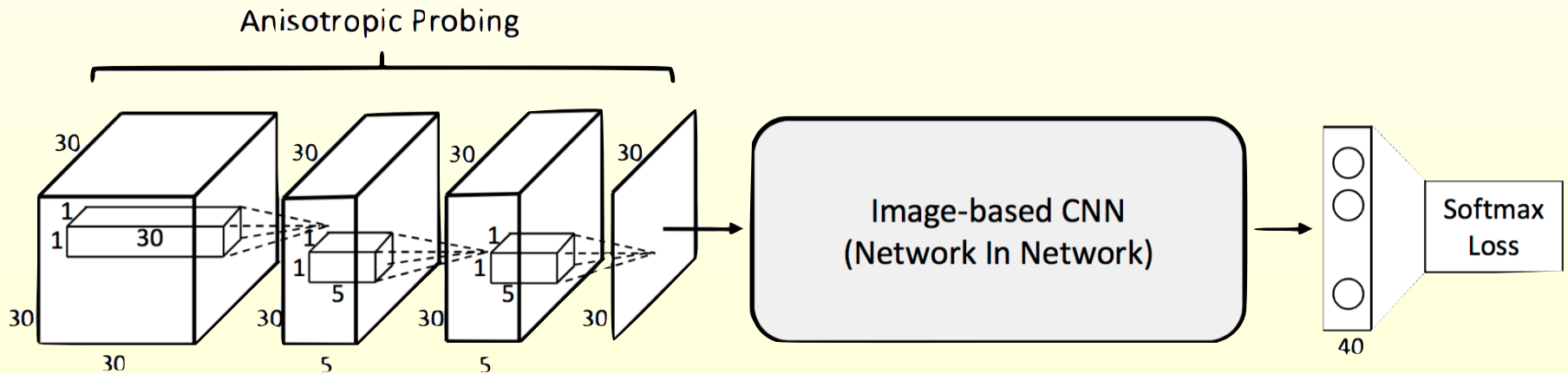
Volumetric and Multi-View CNNs for Object Classification on 3D Data

Hao Su*, **Charles Qi***, **Matthias Niessner**, **Angela Dai**, **Mengyuan Yan**, **Leonidas Guibas**

CVPR 2016 (spotlight oral)

Volumetric Representation: Classification

*Idea: “X-ray” rendering + Image (2D) CNNs
very low #param, very low computation*



Volumetric and Multi-View CNNs for Object Classification on 3D Data

Hao Su*, **Charles Qi***, **Matthias Niessner**, **Angela Dai**, **Mengyuan Yan**, **Leonidas Guibas**

CVPR 2016 (spotlight oral)

Volumetric Representation: Classification

Network	Single-Ori	Multi-Ori
E2E-[33]	83.0	87.8
VoxNet[24]	83.8	85.9
3D-NIN	86.1	88.5
Ours-SubvolumeSup	87.2	89.2
Ours-AniProbing	85.9	89.9

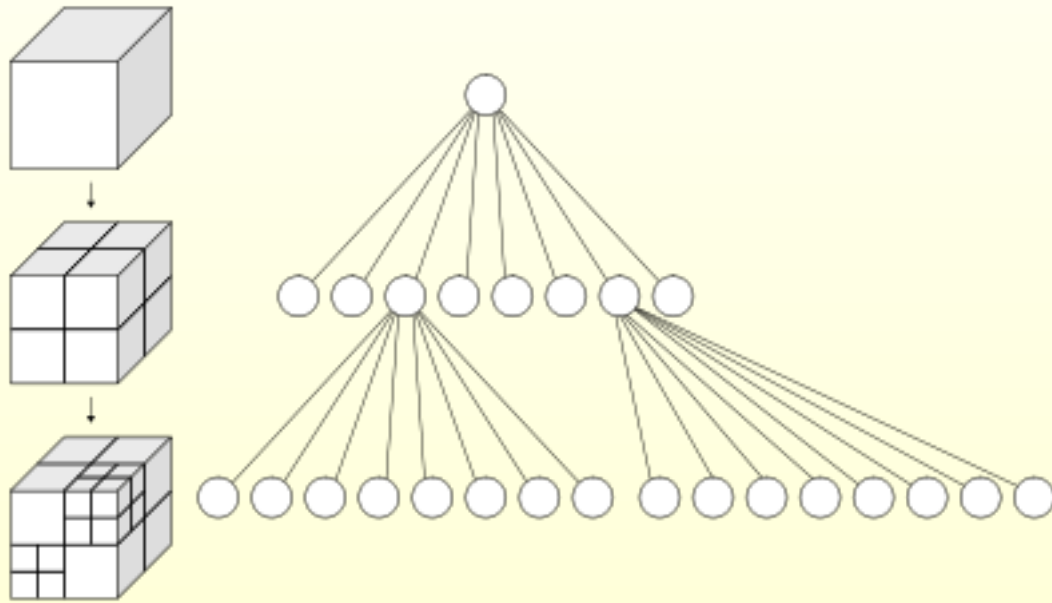
Multi-view: 90.1%

Volumetric and Multi-View CNNs for Object Classification on 3D Data

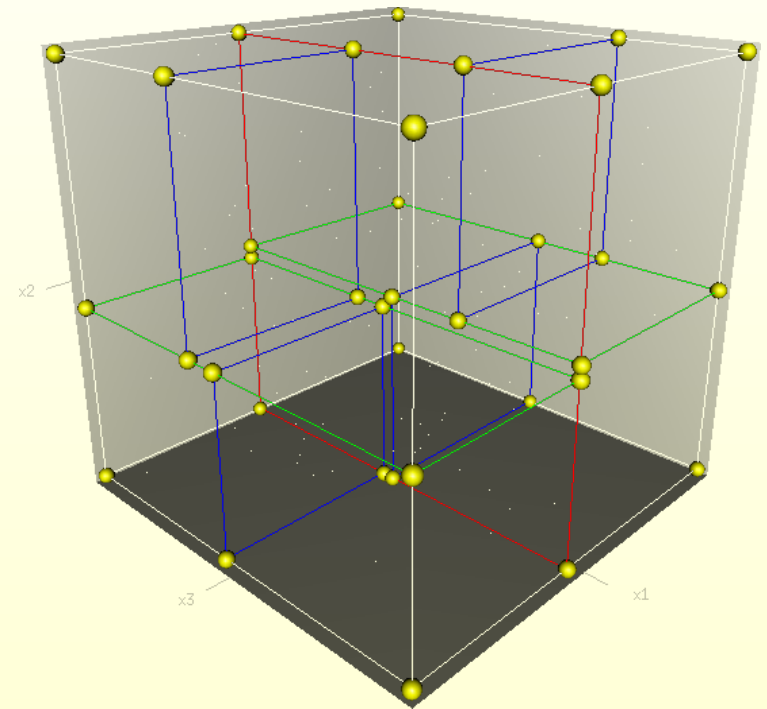
Hao Su*, Charles Qi*, Matthias Niessner, Angela Dai, Mengyuan Yan, Leonidas Guibas

CVPR 2016 (spotlight oral)

Sparse Volumetric Representation



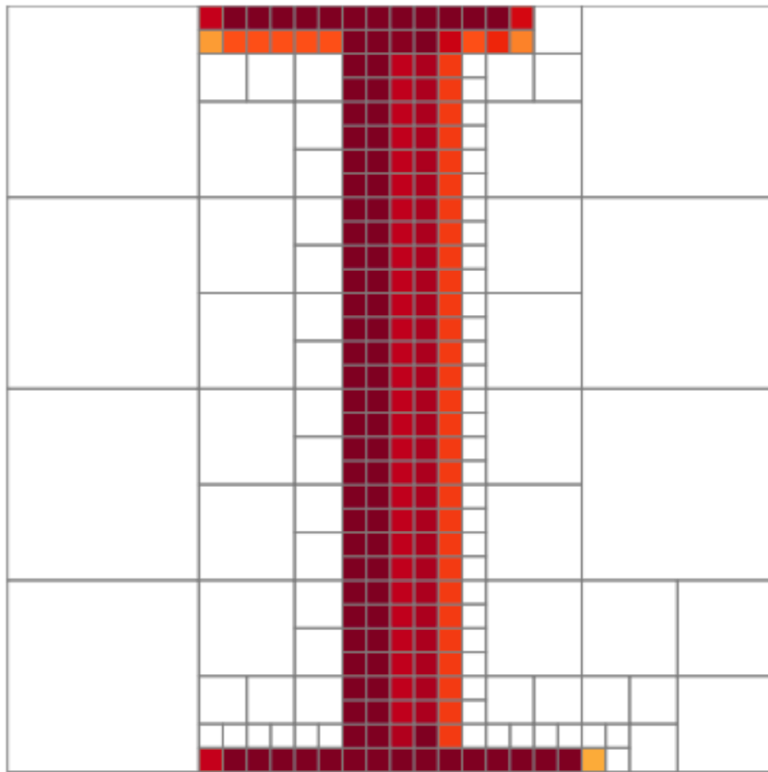
Octree



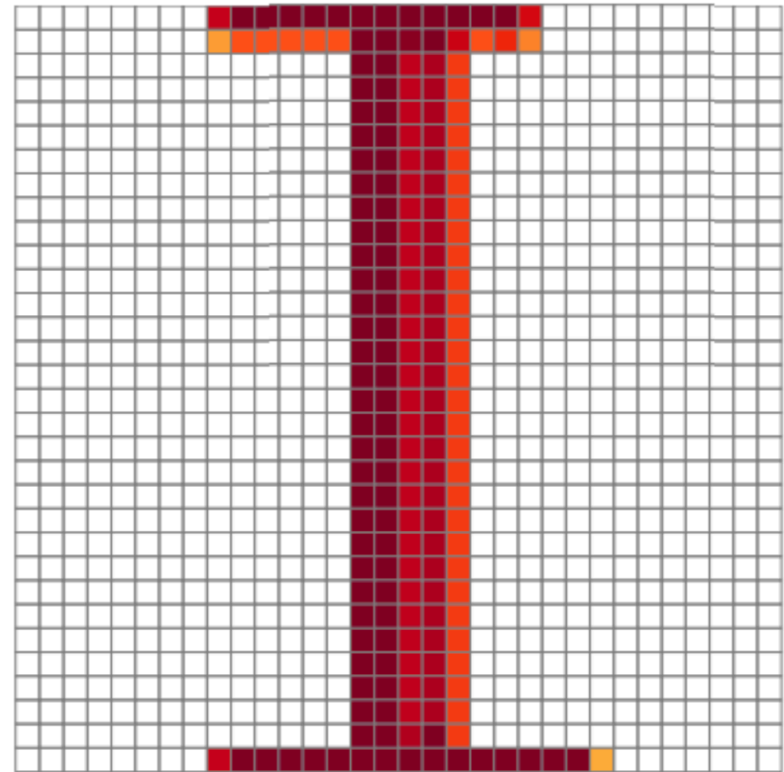
3D KD-Tree

Sparse Volumetric Representation

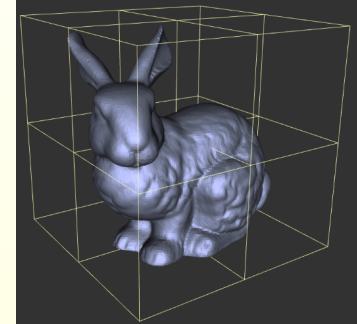
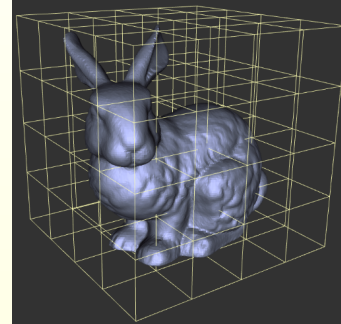
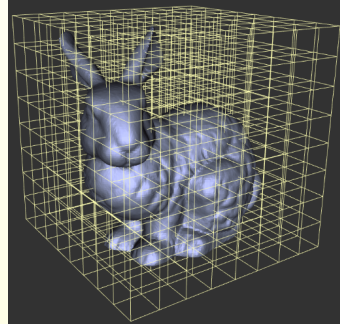
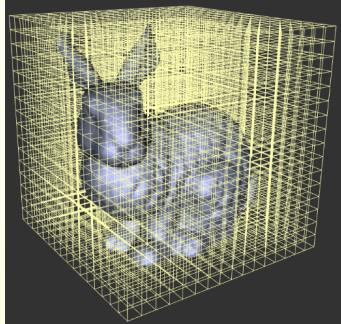
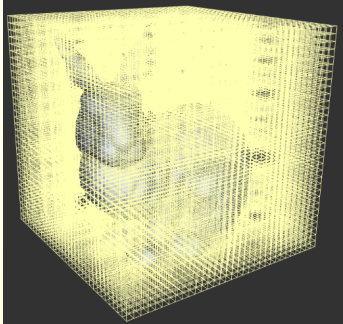
OctNet



Dense 3D ConvNet



Sparse Volumetric Representation



Sparse Volumetric Representation

O-CNN: Octree-based Convolutional Neural Networks for 3D Shape Analysis

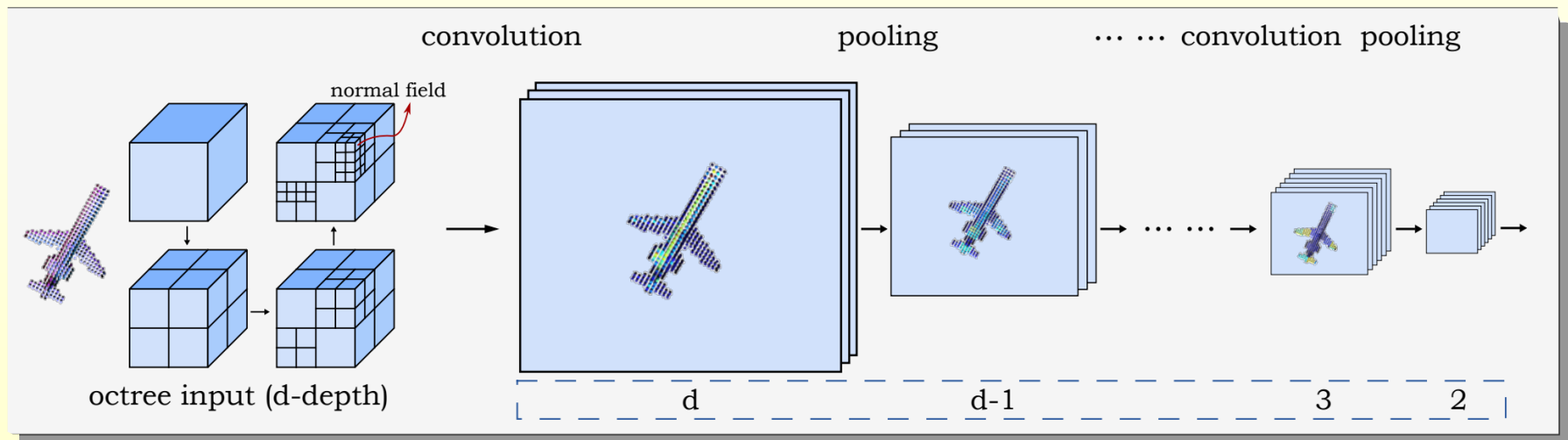
PENG-SHUAI WANG, Tsinghua University and Microsoft Research Asia

YANG LIU, Microsoft Research Asia

YU-XIAO GUO, University of Electronic Science and Technology of China and Microsoft Research Asia

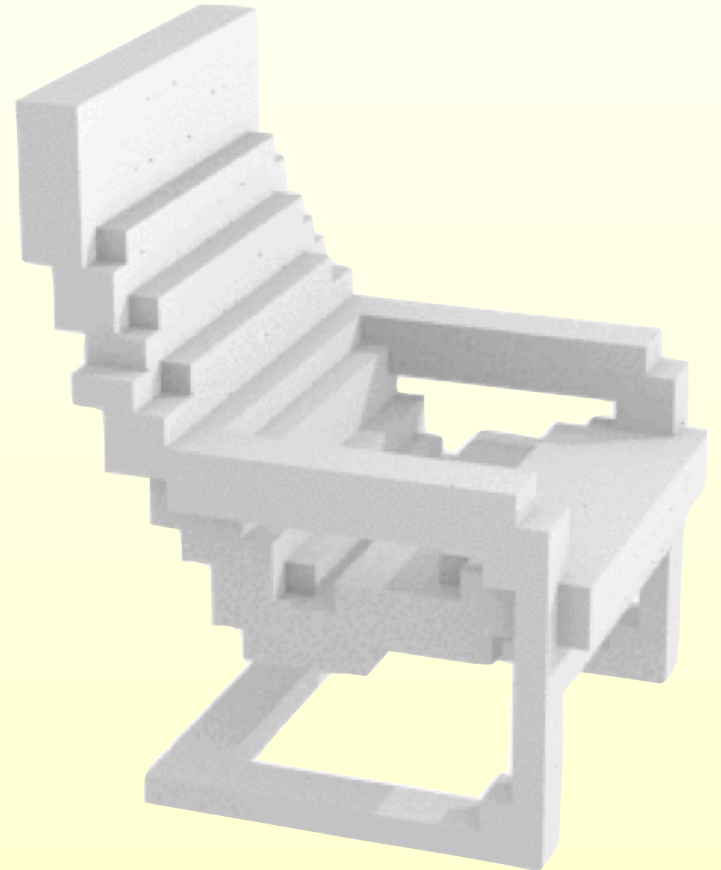
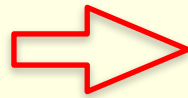
CHUN-YU SUN, Tsinghua University and Microsoft Research Asia

XIN TONG, Microsoft Research Asia

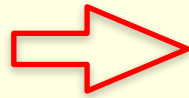


Classification Accuracy: 89.9%

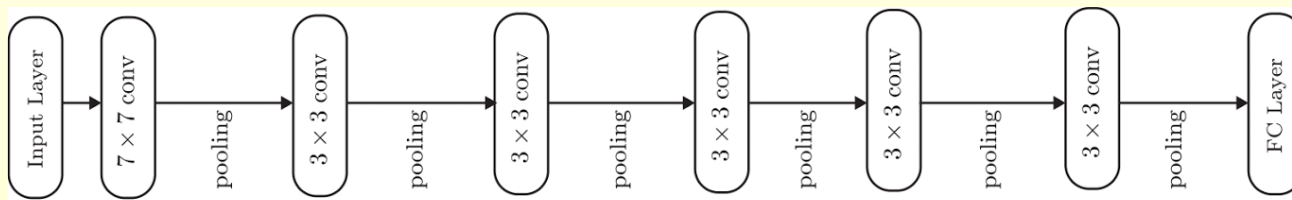
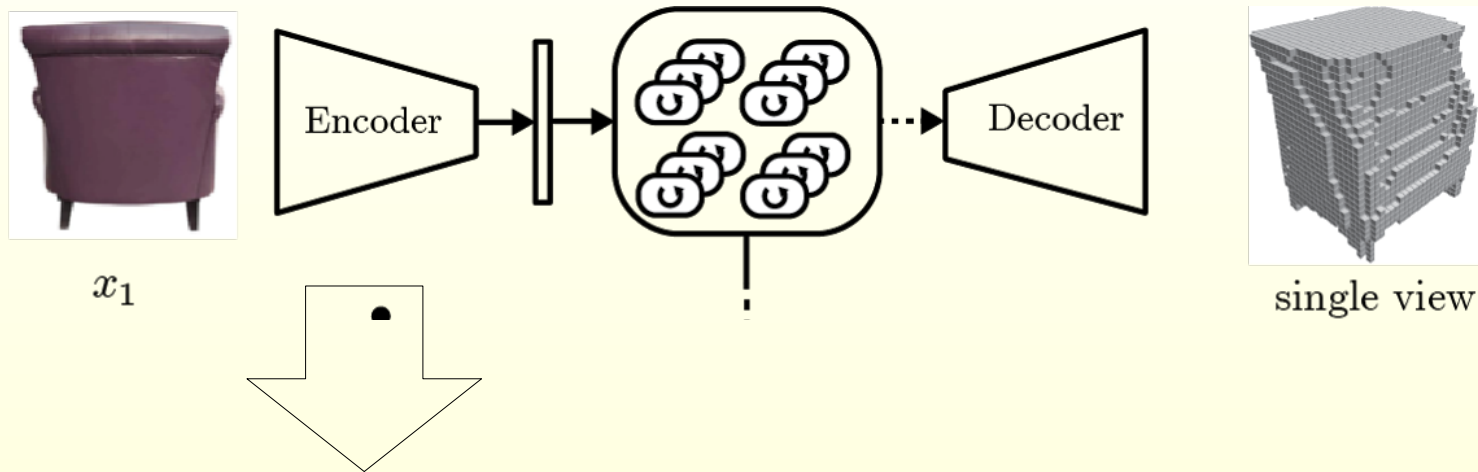
Volumetric Representation: Reconstruction



Volumetric Representation: Completion

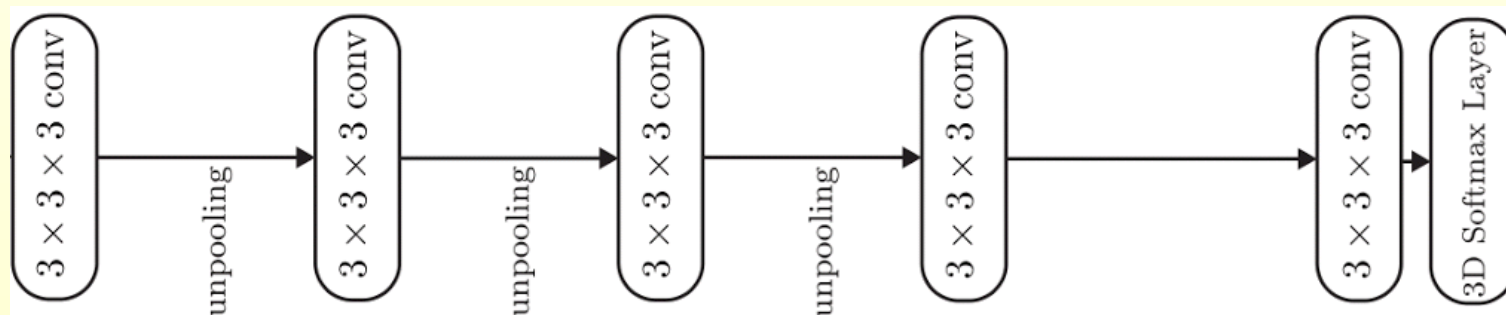
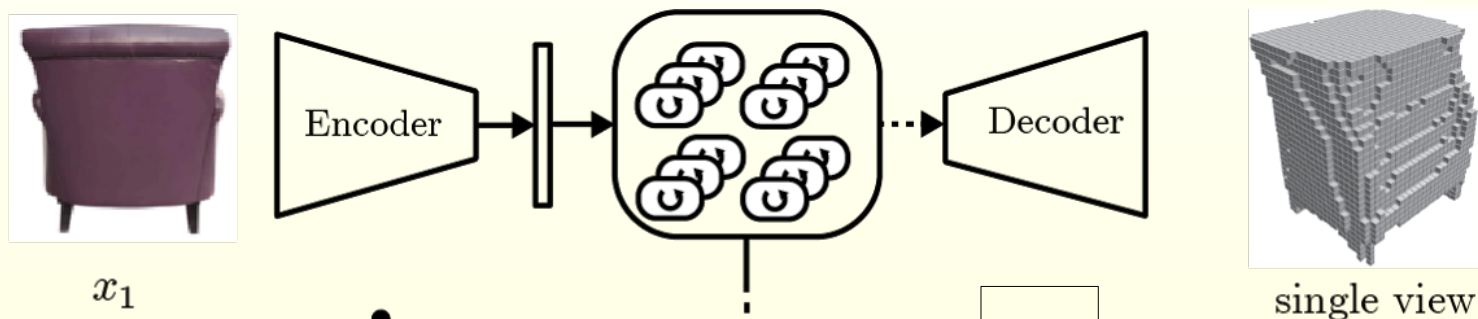


Volumetric Representation: Reconstruction



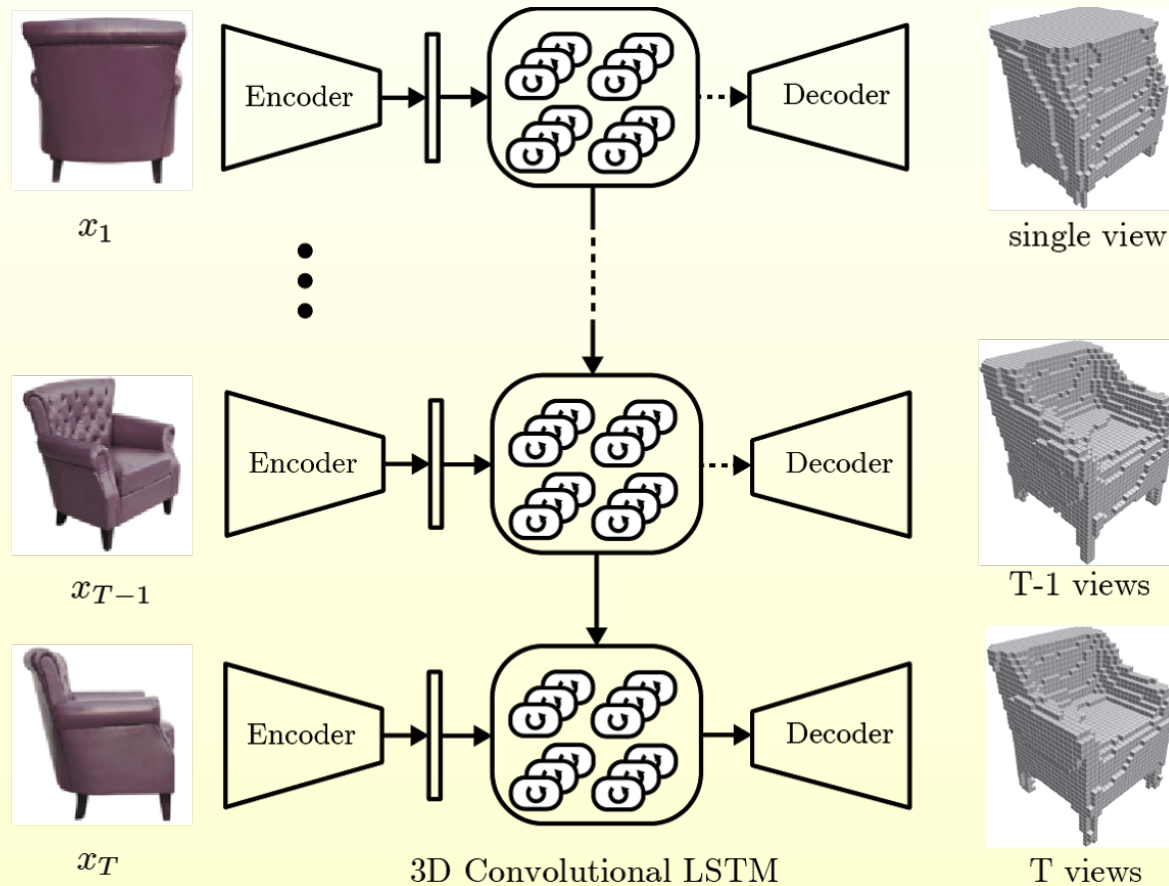
3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction
Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, Silvio Savarese
ECCV 2016

Volumetric Representation: Reconstruction

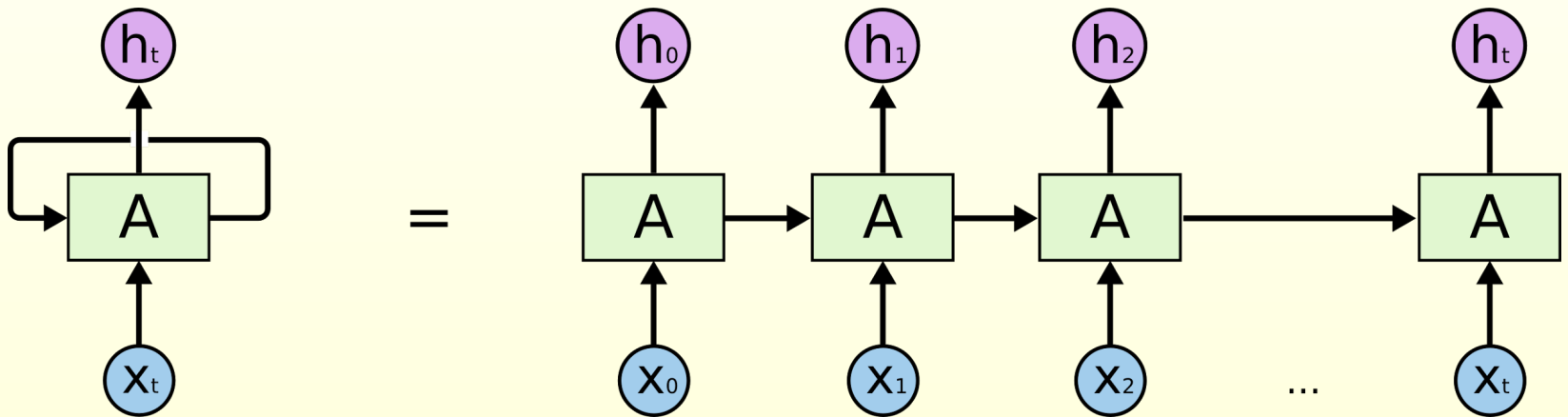


3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction
Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, Silvio Savarese
ECCV 2016

Volumetric Representation: Reconstruction

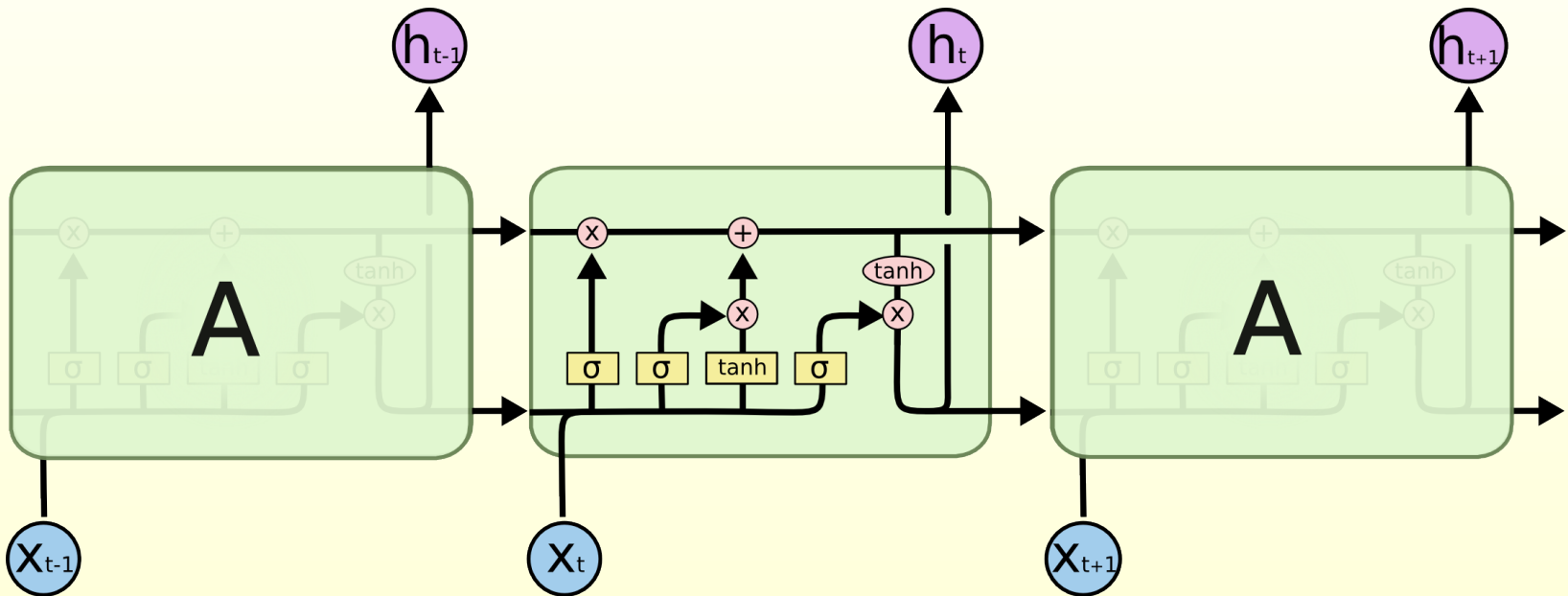


RNN



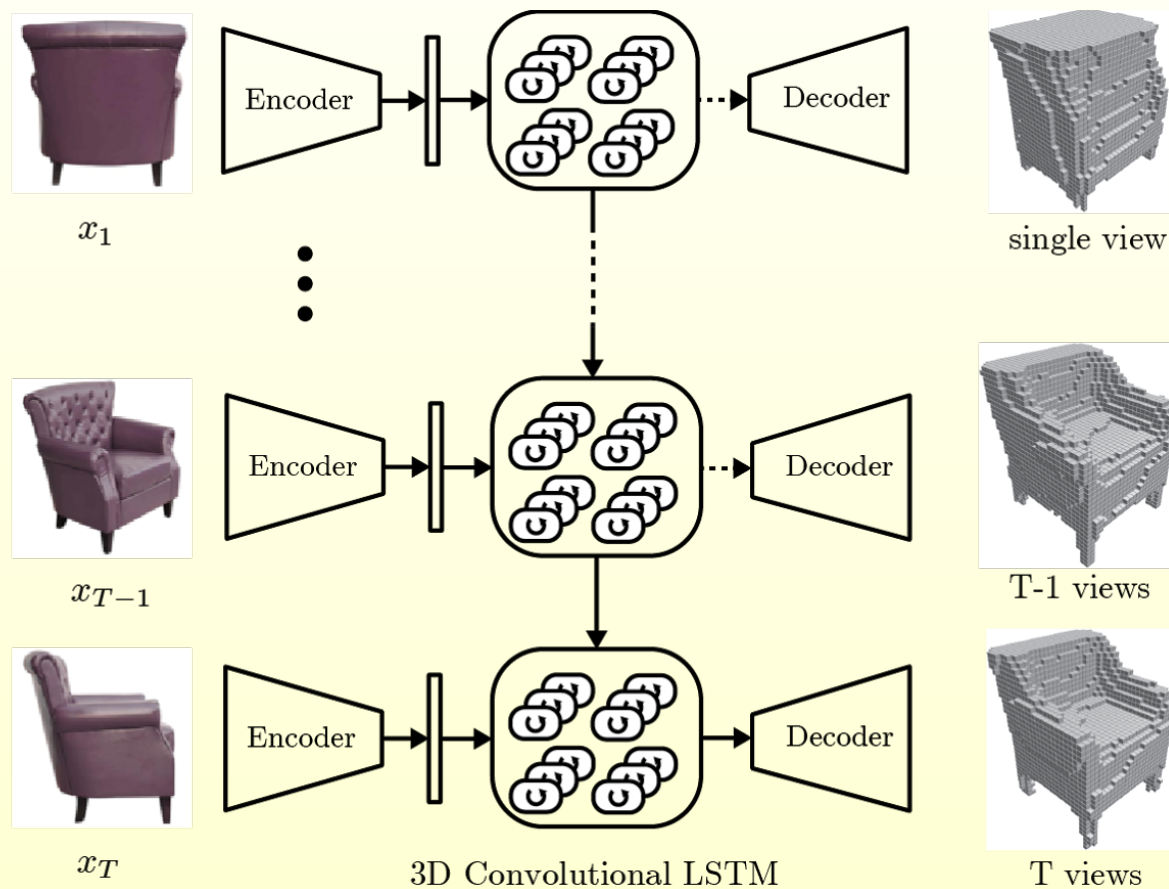
[Christopher Olah] Understanding LSTM Networks, <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

LSTM



[Christopher Olah] Understanding LSTM Networks, <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

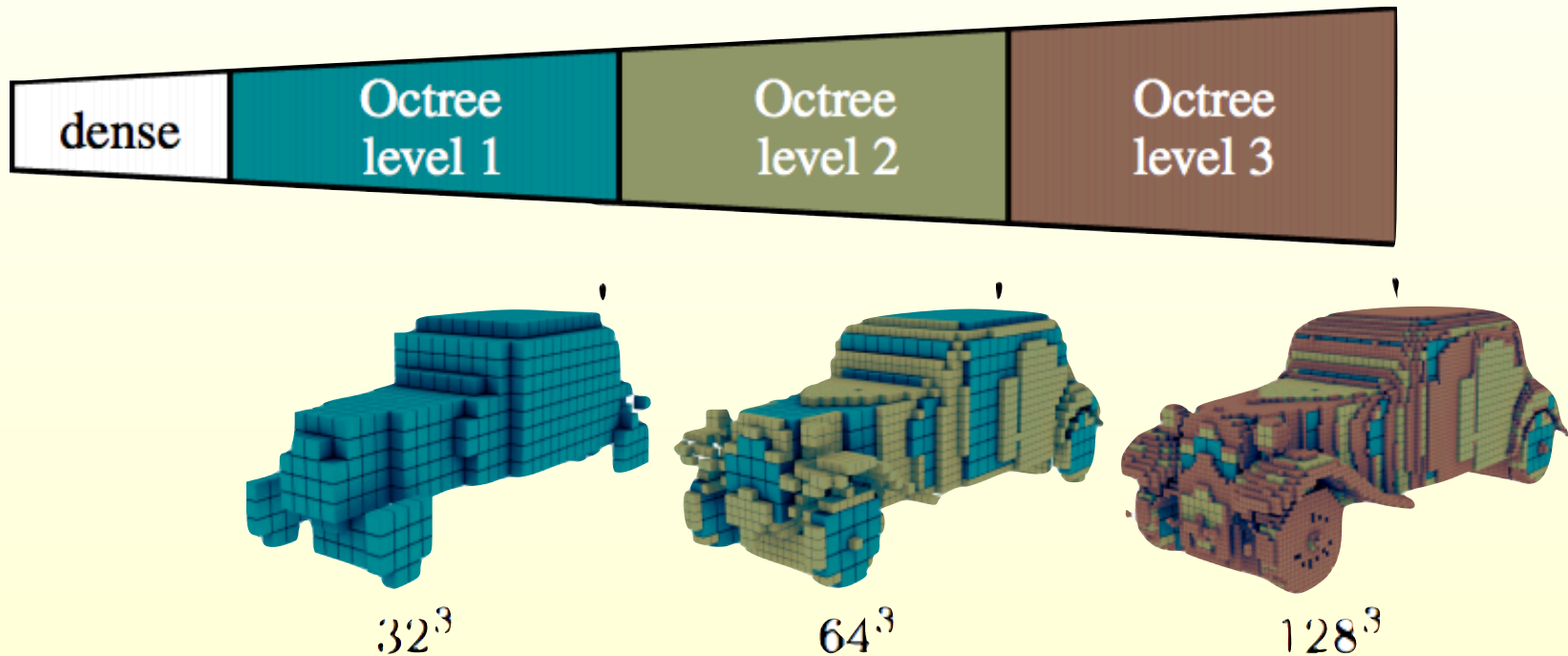
Volumetric Representation: Reconstruction



Volumetric Representation: Reconstruction



Volumetric Representation: Reconstruction

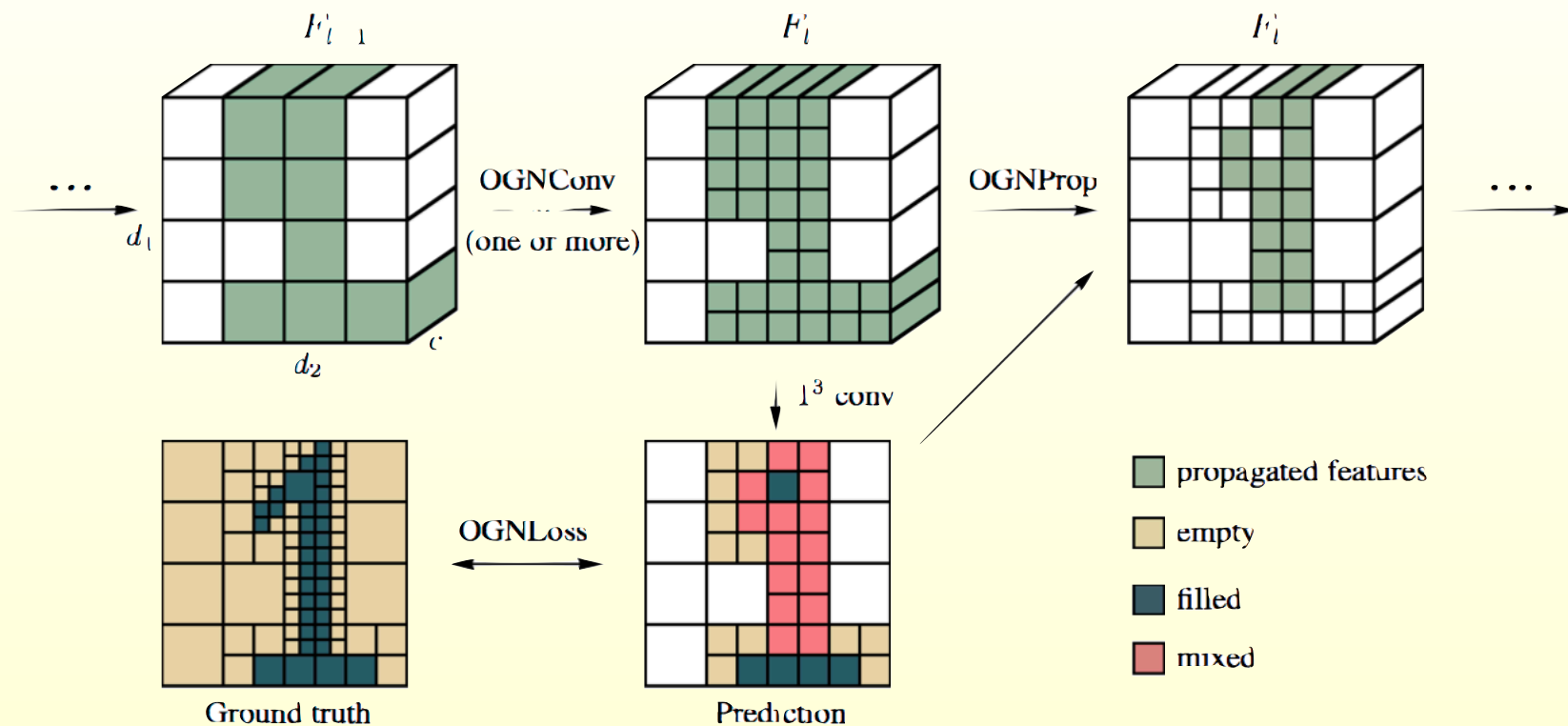


Maxim Tatarchenko, Alexey Dosovitskiy, Thomas Brox

“Octree Generating Networks: Efficient Convolutional Architectures for High-resolution 3D Outputs”

ICCV, 2017

Volumetric Representation: Reconstruction

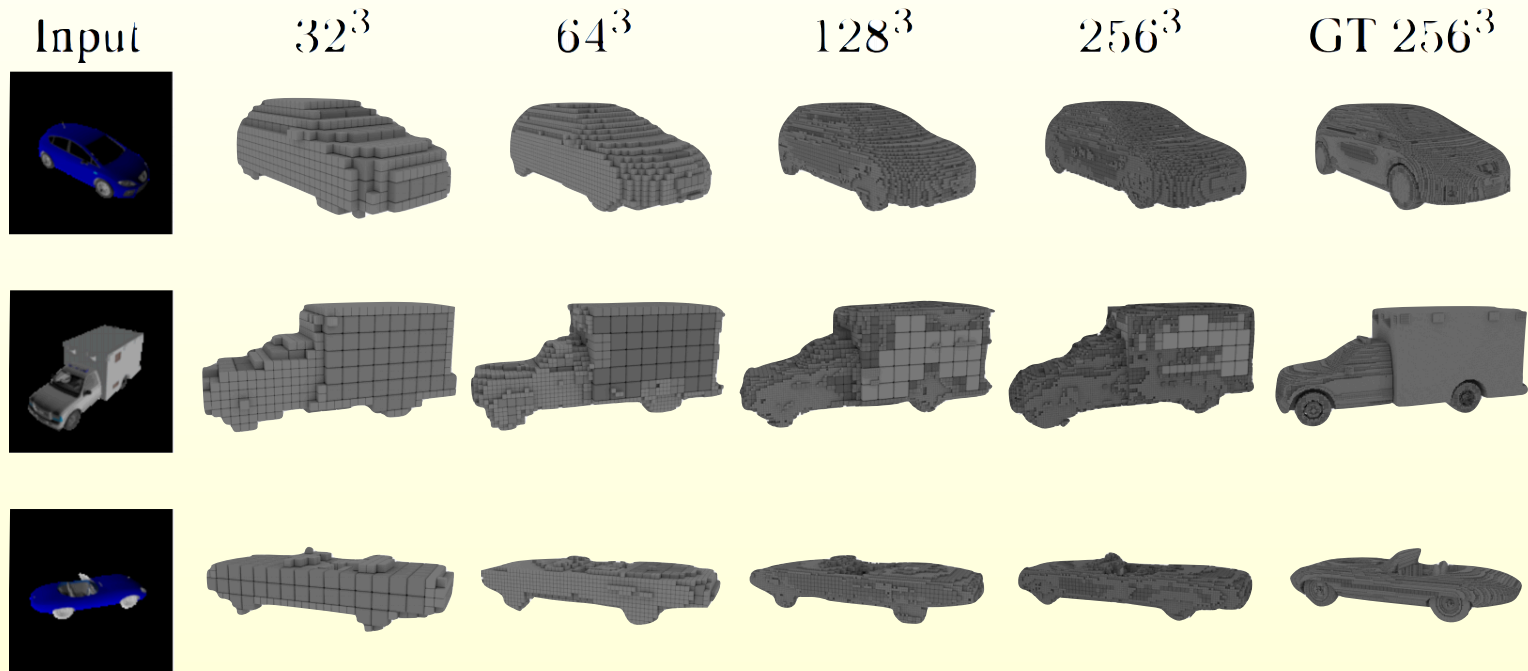


Maxim Tatarchenko, Alexey Dosovitskiy, Thomas Brox

“Octree Generating Networks: Efficient Convolutional Architectures for High-resolution 3D Outputs”

ICCV, 2017

Volumetric Representation: Reconstruction



Maxim Tatarchenko, Alexey Dosovitskiy, Thomas Brox

“Octree Generating Networks: Efficient Convolutional Architectures for High-resolution 3D Outputs”

ICCV, 2017

Reconstruction: Point Cloud Based



Input

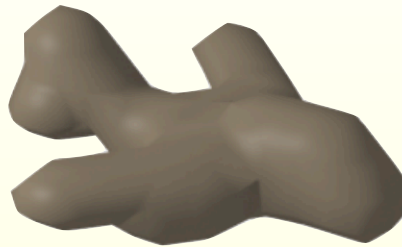
Reconstructed 3D point cloud

Haoqiang Fan, Hao Su, Leonidas Guibas,
“A Point Set Generation Network for 3D Object Reconstruction from a Single Image”
CVPR, 2017

Reconstruction: Surface Based



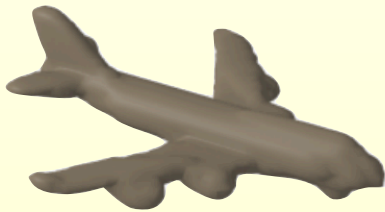
(a) Input Image



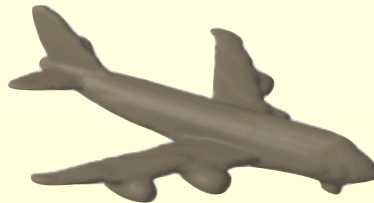
(b) 16^3



(c) 32^3



(d) 64^3



(e) 128^3



(f) 256^3

Conclusion

- ◆ 3D Deep Learning is a new and active research direction
- ◆ 3D data have different representation
- ◆ Multi-view CNNs take advantage of the state-of-the-art 2D CNNs
- ◆ Volumetric 3D CNNs suffer from the “curse of dimensionality”

Agenda

- ◆ Today
 - ◆ Deep Learning Intro
 - ◆ 3D Deep Learning
 - ◆ Multi-view CNNs
 - ◆ Volumetric CNNs
- ◆ Later lectures
 - ◆ Point-cloud NNs (May 21)
 - ◆ Graph CNNs (May 24)

Useful Courses

- ◆ Deep Learning

- ◆ CS 231n: <http://cs231n.stanford.edu/>

- ◆ CS 224n: <http://cs224n.stanford.edu/>

- ◆ 3D Deep Learning

- ◆ CS 468: <http://cs468.stanford.edu/>

- ◆ UCSD CSE 291-100: <https://cse291-i.github.io/>

End