# Computer Vision / Robot Vision

- Lecturer: Takayuki Okatani, Prof. at GSIS
- Room: Here
- Date: Every Monday in the 3rd quarter
  - 10/7, 14(holiday), 21, 28, 11/11, 18, 25
- Time 8:50-10:20, 10:30-12:00 → 9:00-12:00 w/ 10 min break

# Requirements: knowledge and skills for students

- Basic knowledge of linear algebra & statistics
  - E.g., you need to understand the followings:
    - Eigenvalues/vectors of a matrix
    - Pseudo inverse of a matrix
    - Solving an overdetermined/underdetermined system of linear equations
    - Newtons' method
    - Joint probability, conditional probability etc.
    - Random variables and their expected values
    - Bayes theorem
    - Maximum likelihood/maximum posteriori inference
- Can (learn to) write code in Python
  - You need to be at least ready to learn
  - You will be using Google Colaboratory for your assinments

# Grading

- Grading: Several assignments
  - All need to be completed and submitted properly
  - All assignments will be able to be completed in *Google Colaboratory*
  - Brief intro to Google Colaboratory will be given but no further explanation on Python etc. will not be given
    - You must learn how to use Python and relevant modules(OpenCV & DL/ML frameworks) by yourself!

# Outline of this course

1. Introduction
2. Physics of Imaging
3. Camera Model and Projective Transform
4. Multi-view Geometry
5. Basic Design of Neural Networks
6. Training of Neural Networks
7. Convolutional Neural Networks
8. Object Recognition
9. Object Detection/Estimation of Position and Pose
10. Various Learning Methods
11. Visualization and Understanding of Networks
12. Generative Models and Image Generation
13. Networks for Sequential Data
14. Various Applications
15. Summary and Conclusion

# Updated version of course outline (*still under construction*)

1. Introduction to Computer Vision
2. Basic Design of Neural Networks
3. Training of Neural Networks
4. Convolutional Neural Networks I
5. Convolutional Neural Networks II
6. Networks for Sequences, Sets, and Graphs
7. Apps to Standard CV Problems
8. Various Learning Methods
9. Explanability of Deep Learning Methods
10. Efficient Models
11. Generative Models and Image Generation
12. Model-based Computer Vision I
13. Model-based Computer Vision II
14. Summary and Conclusion

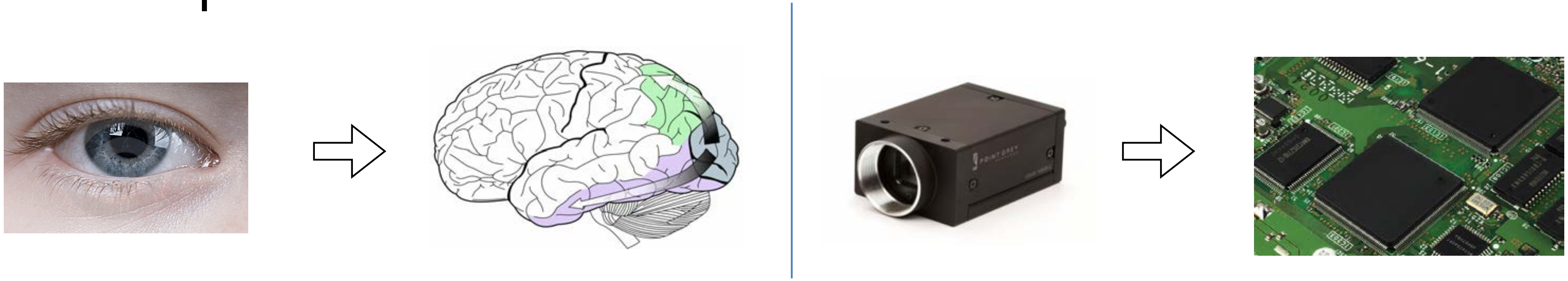
# INTRODUCTION TO COMPUTER VISION

# History of Computer Vision

- Math and physics-based model (1980–)
  - Multi-view geometry & physics-based vision
  - Apps: computer graphics, augmented reality, etc.
  - Examples
    - Photometric stereo, Optic flow estimation, SIFT, Structure-from-Motion, Blind deblurring
- Introduction of machine learning (2000–)
  - Apps: surveillance camera, driver assistance, etc.
  - Examples
    - Face detection (cascaded classifier), Mocap for Kinnect (random forest)
- Paradigm shift: deep learning (2010–)
  - Explosive developments
  - Apps: potentially every problem

# Computer Vision ~ The Early Days

- Goal: Implement high-level functions of human vision on a computer



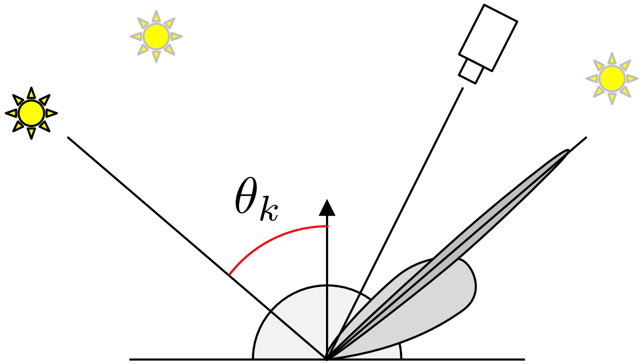
- Background: Scientific interests in human vision
  - Why vision? Because it is a good initial step for understanding intelligence

- D. Marr --- Three levels of analysis
  - Computational level (What problem is solved?)
  - Algorithmic level (How is it solved?)
  - Implementation level (What hardware?)

# Photometric stereo [Woodham1980]

- Surface shape can be recovered from multiple images of a single object under different illumination directions
  - Brightness of a surface point is modeled by: $x_k = \rho \cos \theta_k = \rho(\mathbf{n}^\top \mathbf{l}_k)$

(An ideal case: Lambertian reflectance)



$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_L \end{bmatrix} = \rho \begin{bmatrix} \mathbf{l}_1^\top \\ \mathbf{l}_2^\top \\ \vdots \\ \mathbf{l}_L^\top \end{bmatrix} \mathbf{n}$$

Assuming knowledge of illuminant directions $\mathbf{l}_1, \ldots, \mathbf{l}_L$, it's easy to compute $\mathbf{n}$

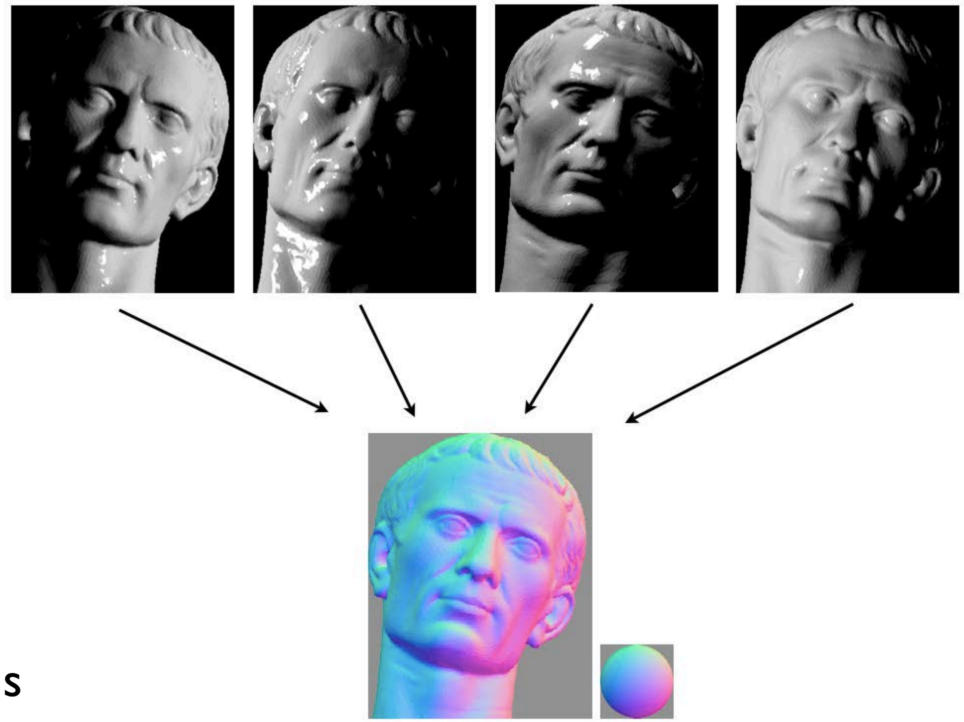
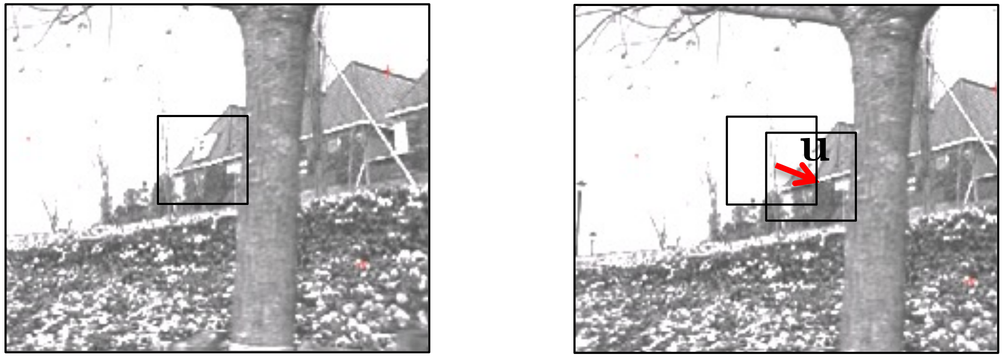


Image by Meekohi

# Lucas-Kanade tracker [Lucas-Kanade1982]

- Want to estimate displacement of a scene point from $I_0$ to $I_1$



- Minimize squared difference between a fixed patch in $I_0$ and its displaced patch in $I_1$ by Newton's *method*
  – Starting from an initial guess, update a solution iteratively solving a linear equation

$$
\begin{aligned}
E_{\mathrm{LK-SSD}}(\boldsymbol{u}+\Delta\boldsymbol{u}) &= \sum_i [I_1(\boldsymbol{x}_i+\boldsymbol{u}+\Delta\boldsymbol{u}) - I_0(\boldsymbol{x}_i)]^2 \\
&\approx \sum_i [I_1(\boldsymbol{x}_i+\boldsymbol{u}) + \boldsymbol{J}_1(\boldsymbol{x}_i+\boldsymbol{u})\Delta\boldsymbol{u} - I_0(\boldsymbol{x}_i)]^2 \\
&= \sum_i [\boldsymbol{J}_1(\boldsymbol{x}_i+\boldsymbol{u})\Delta\boldsymbol{u} + e_i]^2,
\end{aligned}
$$

$$\boldsymbol{J}_1(\boldsymbol{x}_i+\boldsymbol{u}) = \nabla I_1(\boldsymbol{x}_i+\boldsymbol{u}) = (\frac{\partial I_1}{\partial x}, \frac{\partial I_1}{\partial y})(\boldsymbol{x}_i+\boldsymbol{u})$$

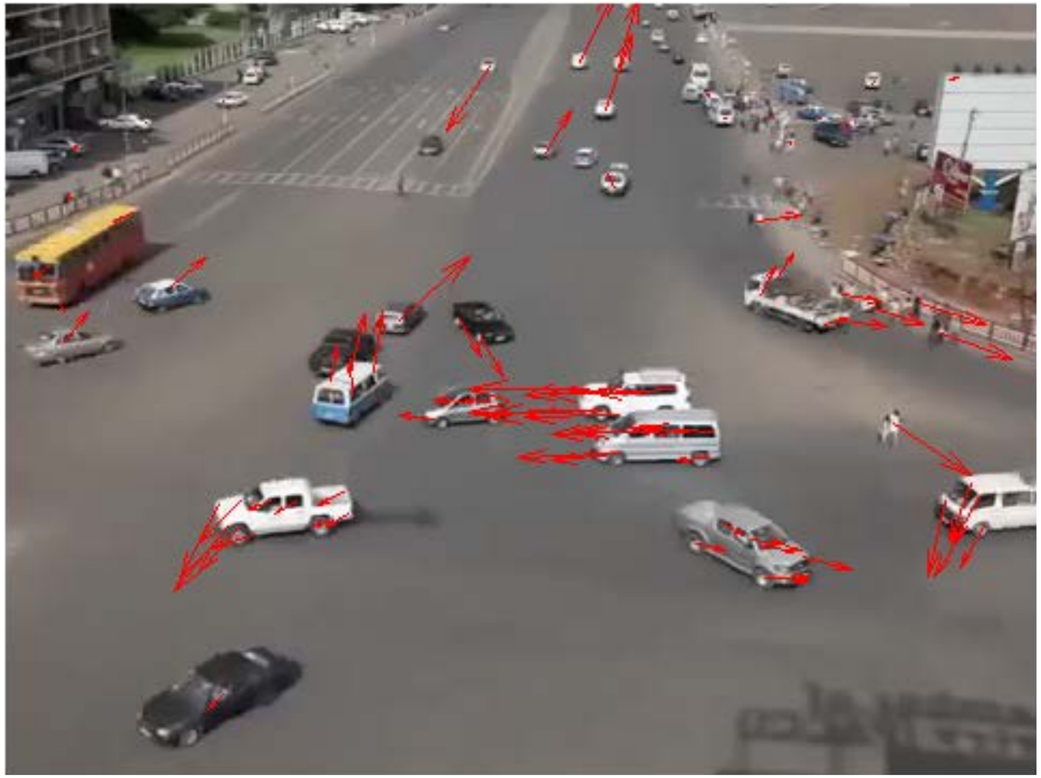
$$e_i = I_1(\boldsymbol{x}_i+\boldsymbol{u}) - I_0(\boldsymbol{x}_i)$$

$$\boldsymbol{A}\Delta\boldsymbol{u} = \boldsymbol{b}$$

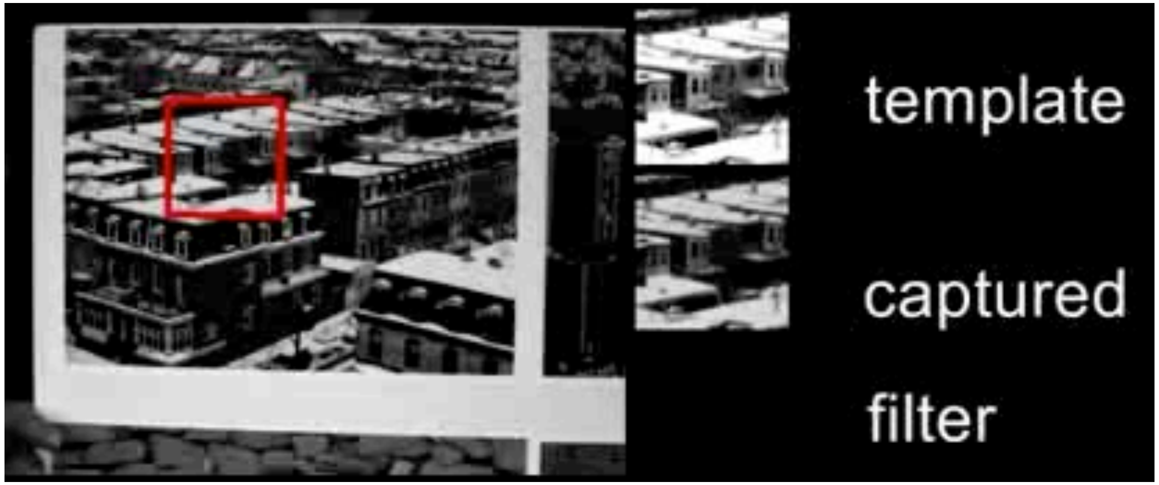
$$\boldsymbol{A} = \sum_i \boldsymbol{J}_1^T(\boldsymbol{x}_i+\boldsymbol{u})\boldsymbol{J}_1(\boldsymbol{x}_i+\boldsymbol{u})$$

$$\boldsymbol{b} = -\sum_i e_i \boldsymbol{J}_1^T(\boldsymbol{x}_i+\boldsymbol{u})$$

# Lucas-Kanade tracker [Lucas-Kanade1982]



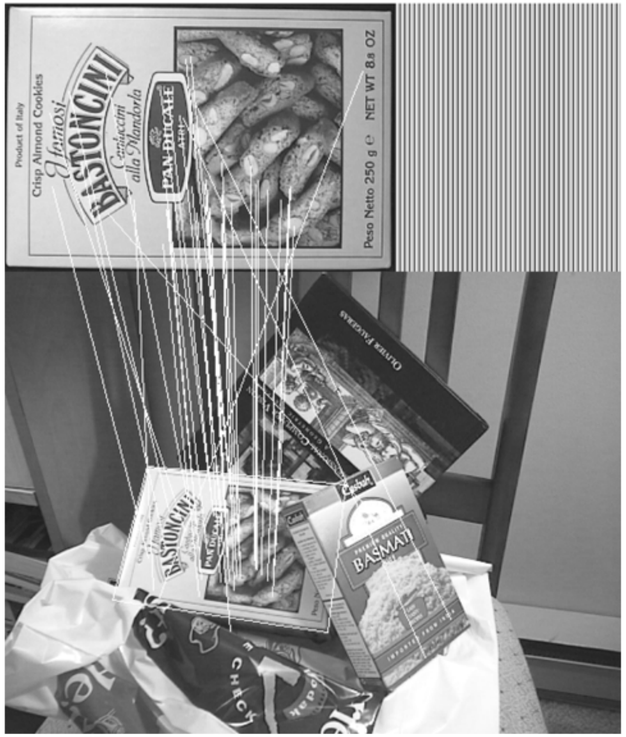
Selected points/windows are tracked between two images from a video clip

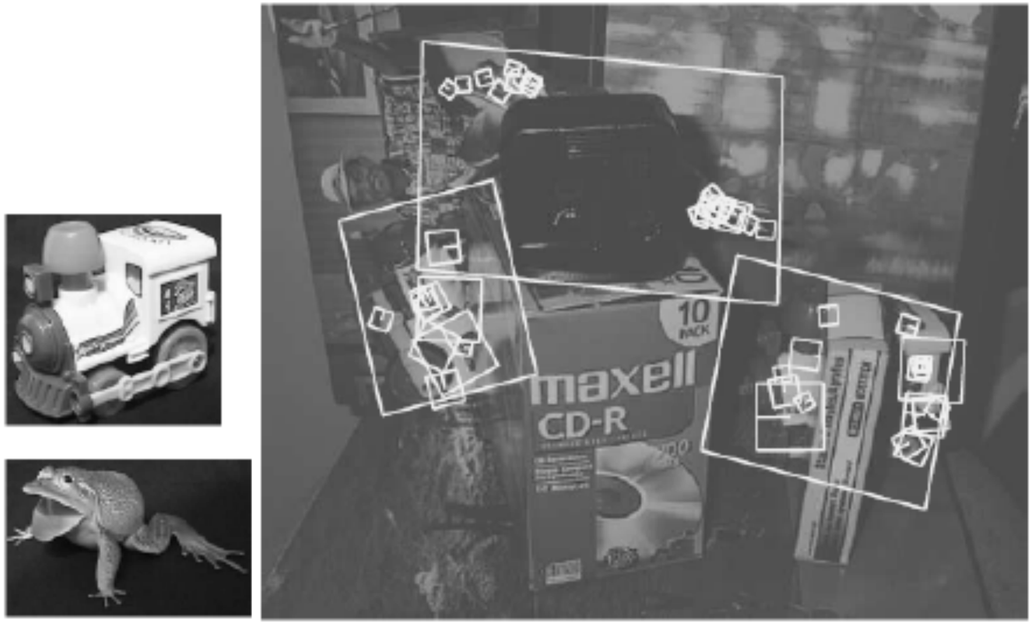


Target plane is parametrized by 2D homography (8DOF), for which similar iterative optimization is performed [Ito-Okatani, 2011]

# SIFT (Scale Invariant Feature Transform) [Lowe 1999]

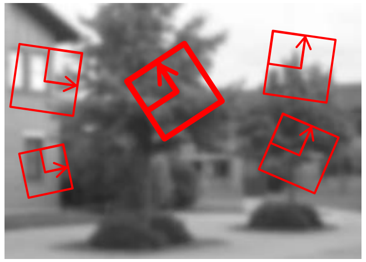
- We wish to match points of an object between its two images captured at different viewpoints
  - Difficulties: Changes in scale, rotation, etc.
- Originally developed for estimating pose of objects; later applied to multi-view geometry

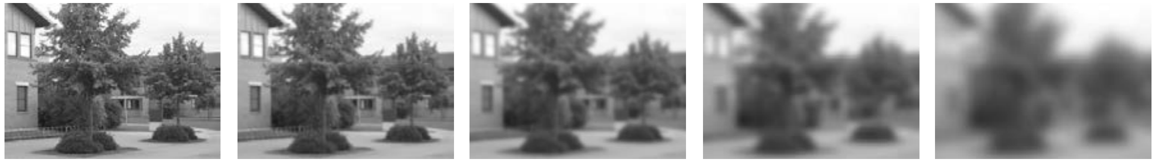




# SIFT (Scale Invariant Feature Transform) [Lowe 1999]

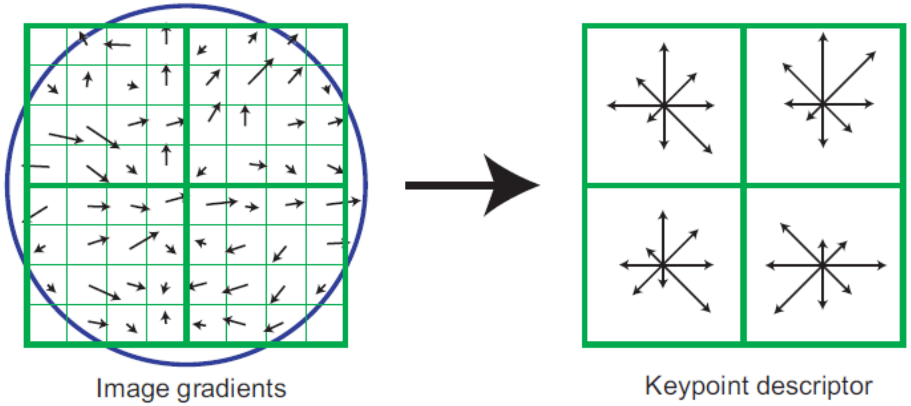
- Good points for matching are chosen; *key points* or interest points
  - Salient points are chosen with their inherent orientation and scale





Scale space

- Local appearance of each key point is encoded; *descriptors*
  - Orientation histogram of brightness gradients



# Structure-from-Motion

- Given m images of n scene points captured from different viewpoints, we want to estimate the 3D coordinates of the n points and the camera matrices of the m views

Geometric imaging model

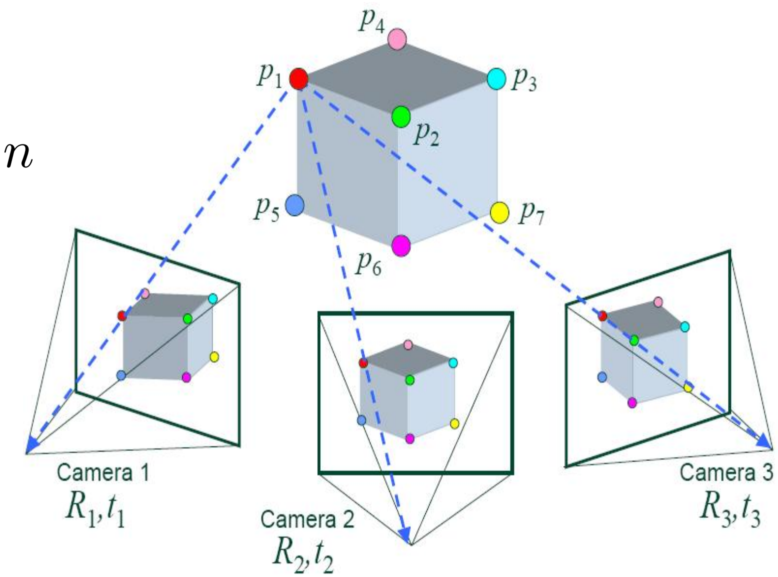
Input

$$\mathbf{x}_j^{(i)} = \begin{bmatrix} x_j^{(i)} & y_j^{(i)} & 1 \end{bmatrix}^\top$$

$$i = 1, \ldots, m, \quad j = 1, \ldots, n$$

$$2mn$$

$$\mathbf{x}_j^{(i)} \propto \mathtt{P}_i \mathbf{X}_j$$



Output

$$\mathbf{X}_j = \begin{bmatrix} X_j & Y_j & Z_j & 1 \end{bmatrix}^\top$$

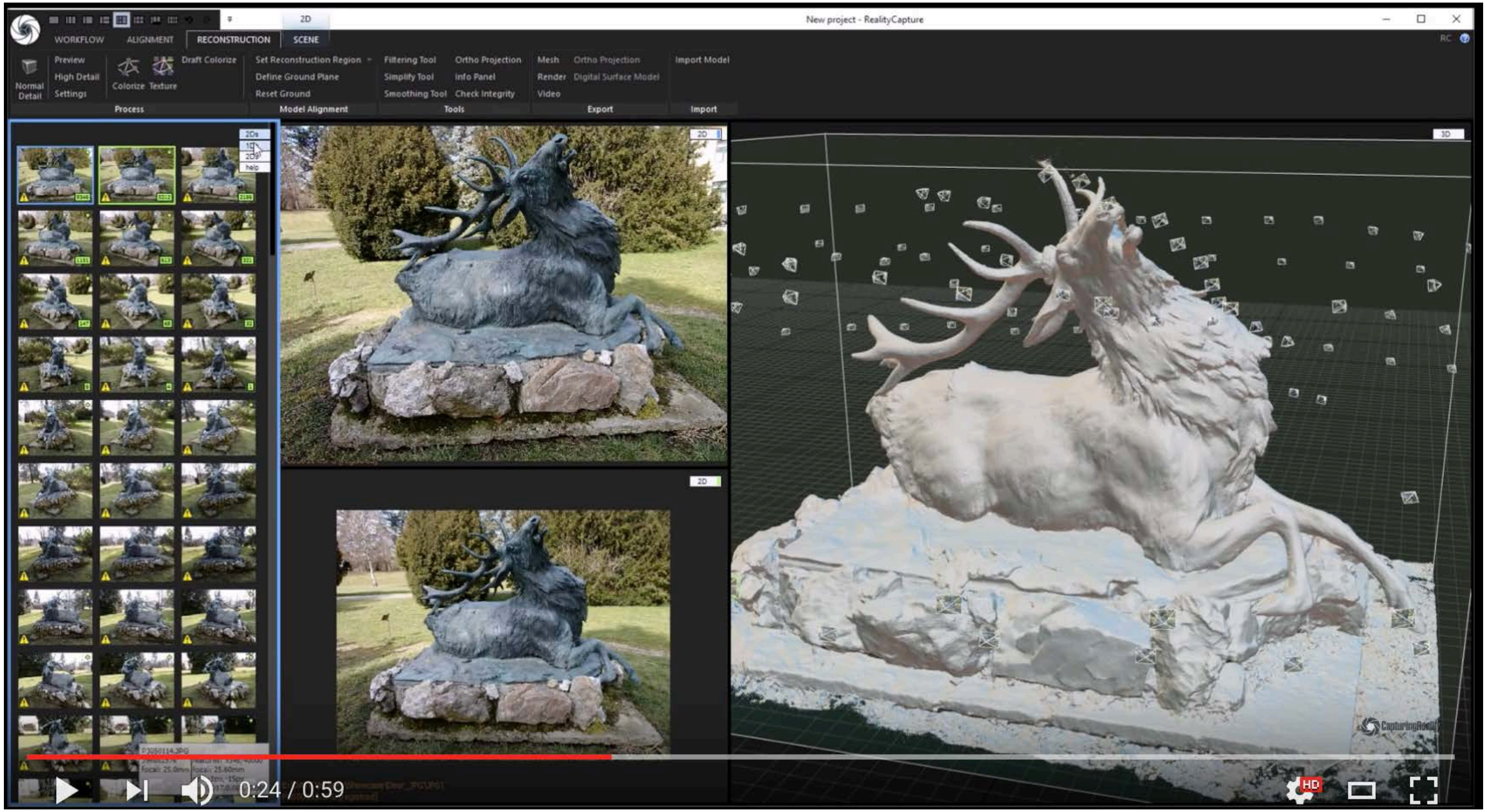
$$3n$$

$$\mathtt{P}_i = \mathtt{K}_i \begin{bmatrix} \mathtt{R}_i & \mathbf{t}_i \end{bmatrix}$$

$$11m$$

# Structure-from-Motion: Example

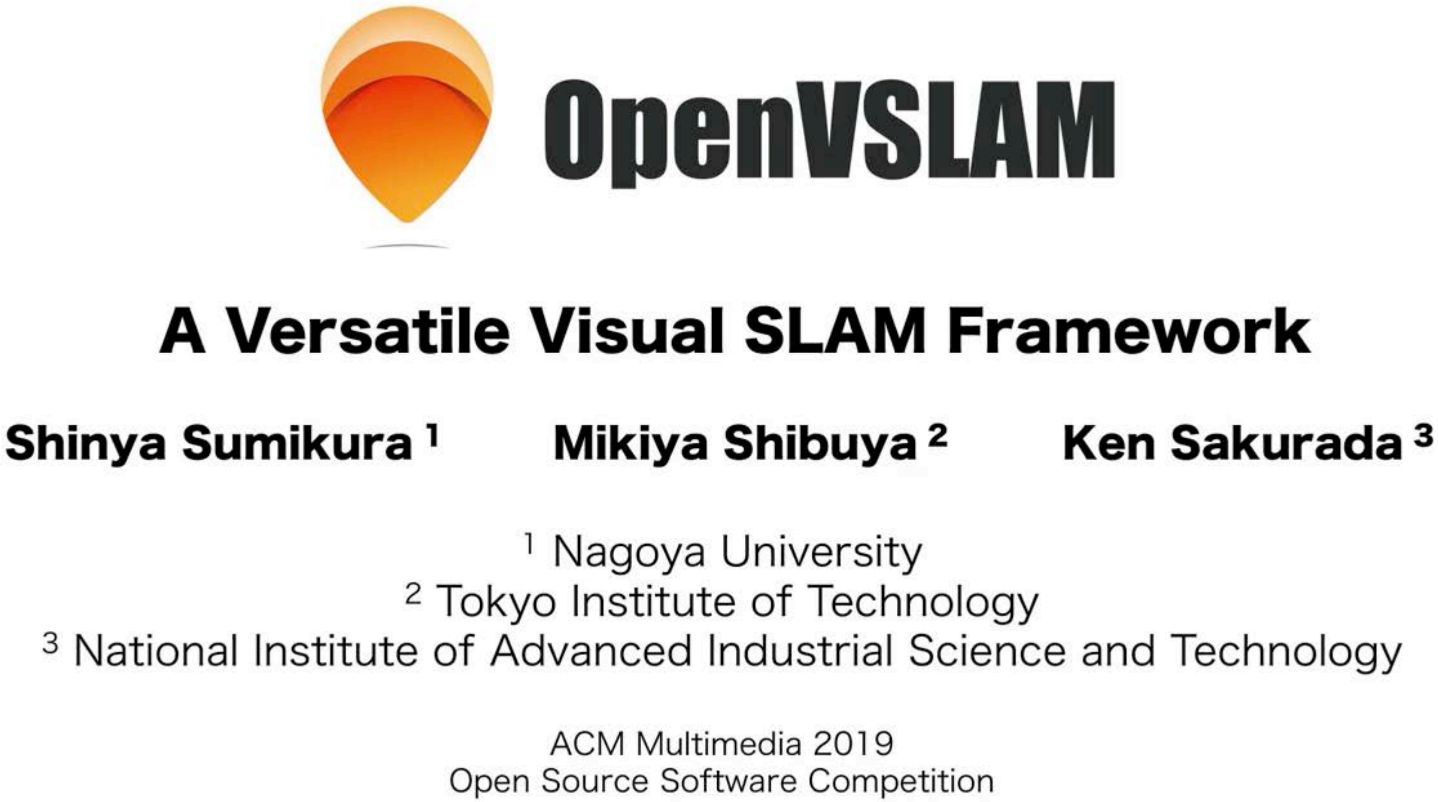
- SfM = Recovering Camera pose + 3D position of interest points
  - Visual SLAM (Simultaneous Localization And Mapping): Does this in real time
- MVS (Multi-View Stereo) = Recovering dense surface shape



www.capturingreality.com

# Structure-from-Motion: Example

- SfM = Recovering Camera pose + 3D position of interest points
  - Visual SLAM (Simultaneous Localization And Mapping): Does this in real time
- MVS (Multi-View Stereo) = Recovering dense surface shape



OpenVSLAM

A Versatile Visual SLAM Framework

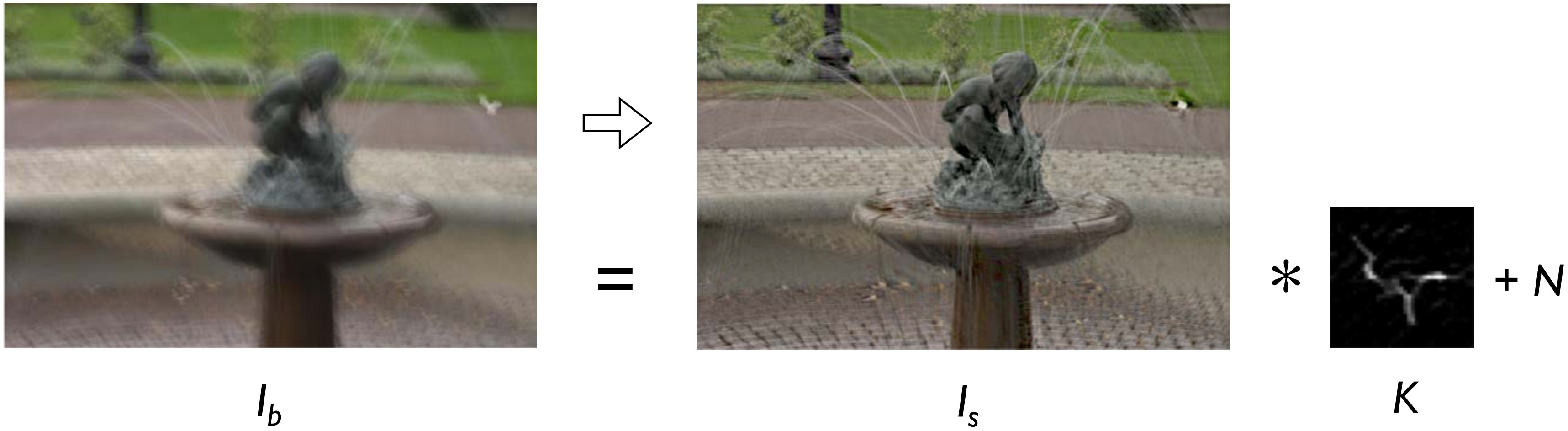
Shinya Sumikura [1] Mikiya Shibuya [2] Ken Sakurada [3]

[1] Nagoya University
[2] Tokyo Institute of Technology
[3] National Institute of Advanced Industrial Science and Technology

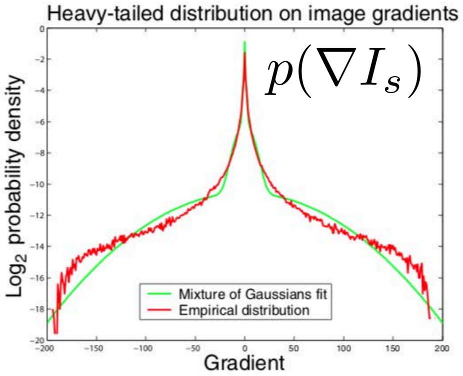
ACM Multimedia 2019
Open Source Software Competition

# Blind image deblurring [Fergus+2006]

- Problem: Given an image $I_b$ of a scene with (motion) blur, we want to recover its sharp image $I_s$



- Model of blurred images: $I_b(x, y) = I_s(x, y) * K(x, y) + N(x, y)$
  - We use a prior knowledge on (statistics) of sharp images and $K$



$$p(\nabla I_s, K \mid \nabla I_b)$$

$$\propto p(\nabla I_b \mid \nabla I_s, K) p(\nabla I_s) p(K)$$

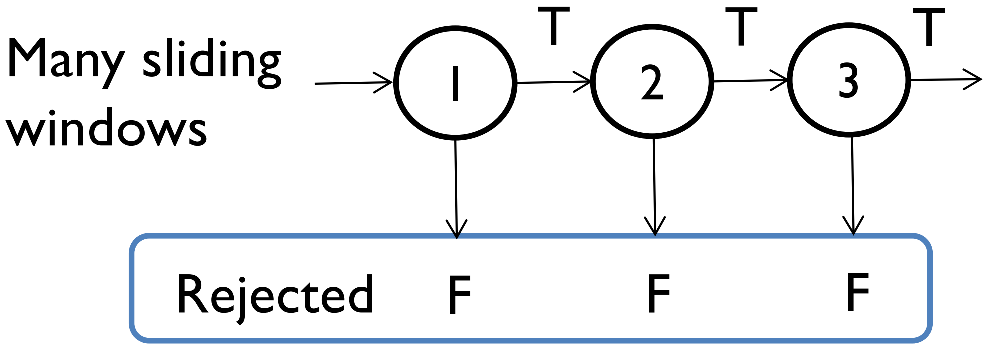
MAP (maximum a posteriori) estimation is conducted

# History of Computer Vision

- Math and physics-based model (1980–)
  - Multi-view geometry & physics-based vision
  - Apps: computer graphics, augmented reality, etc.
  - Examples
    - Photometric stereo, Optic flow estimation, SIFT, Structure-from-Motion, Blind deblurring
- Introduction of machine learning (2000–)
  - Apps: surveillance camera, driver assistance, etc.
  - Examples
    - Face detection (cascaded classifier), Mocap for Kinnect (random forest)
- Paradigm shift: deep learning (2010–)
  - Explosive developments
  - Apps: potentially every problem

# Object detection: Viola-Jones framework [Viola-Jones01]

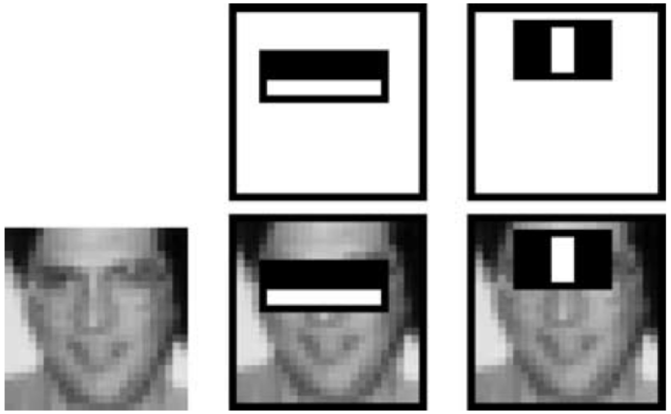
- An approach to object detection: Sliding window w/ binary classification
  - Scan the input image w/ a small window and judge if a face resides inside it
  - Large computational cost → How to cope with it?
- Simple image feature that can be computed quickly
  - *Haar-like* feature; Efficient computation using *integral images* [Papageorgiou+98]
- Strong learning algorithm: *Boosting*, e.g., AdaBoost
  - Boosting is a machine learning method that builds a strong classifier using many weak classifiers
- Fast decision making using cascaded classifiers



# Object detection: Viola-Jones framework [Viola-Jones01]

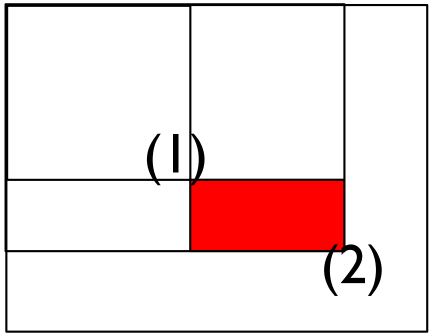
## Haar-like feature

- Binary filters with rect. shapes



[Viola-Jones04]

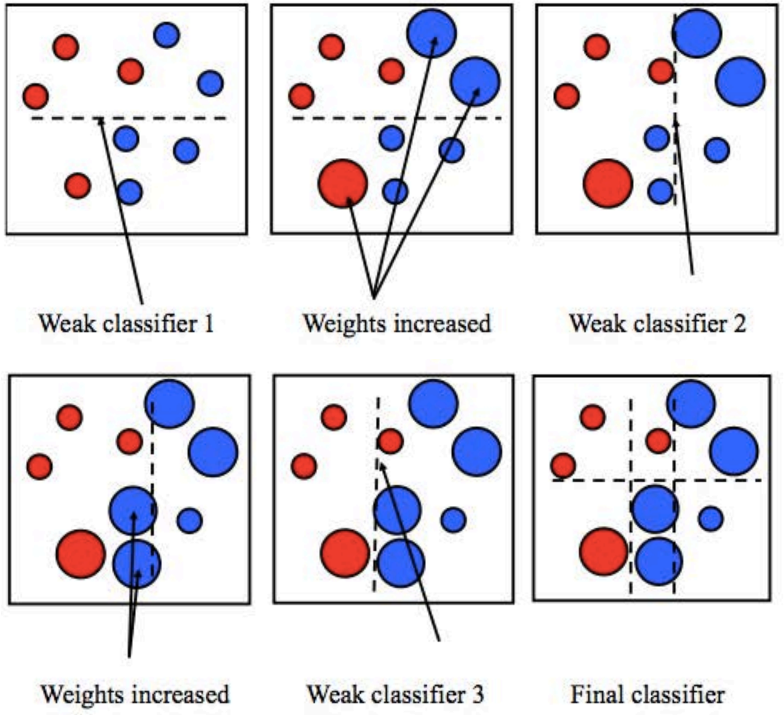
- Integral image
  - Input image is integrated from the top-left to (x,y)



(2)-(1) gives the filter response

## Boosting

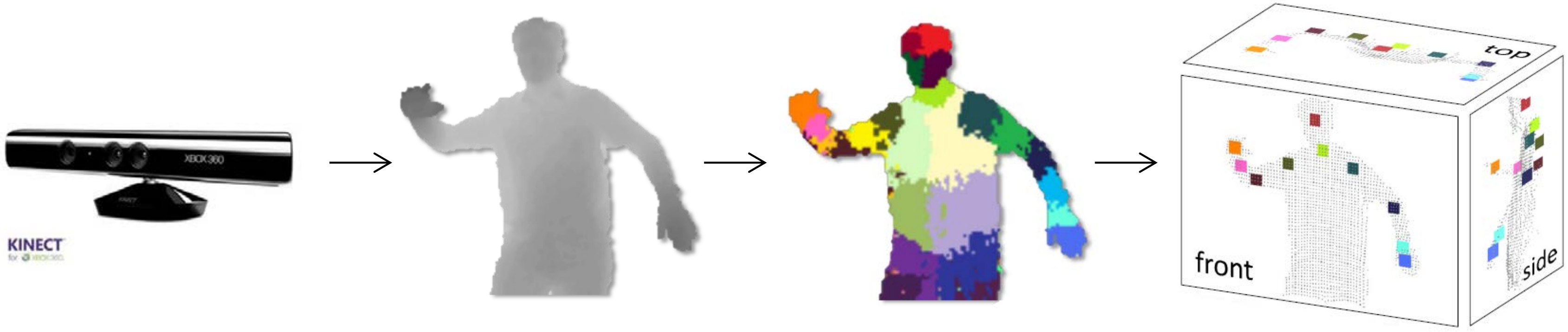
- Each weak classifier builds a *decision boundary* by thresholding a feature score
- A set of such classifiers generates a complex decision boundary

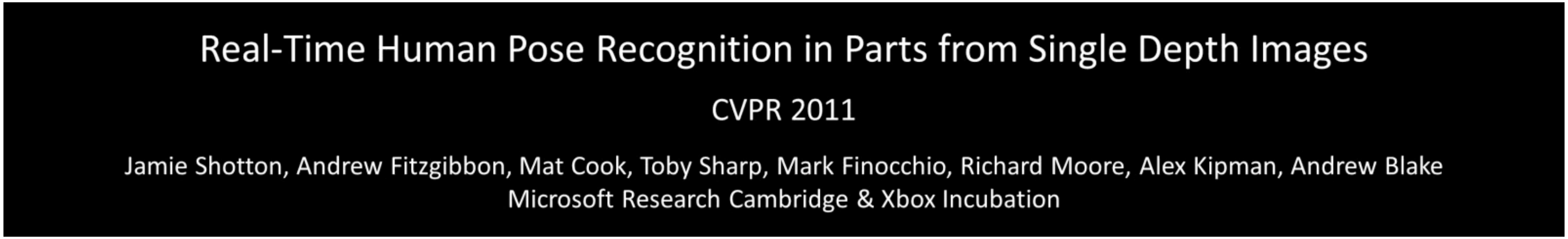


[Szeliski10]

# Mocap for Kinnect [Shotton+2011]

- 1st marker-less motion capture in the history
  - Developed for Microsoft Xbox
- Problem formulation
  - Input: a single depth image
  - Output: class-label of each pixel representing which body part it belongs to; 31 body parts





Real-Time Human Pose Recognition in Parts from Single Depth Images

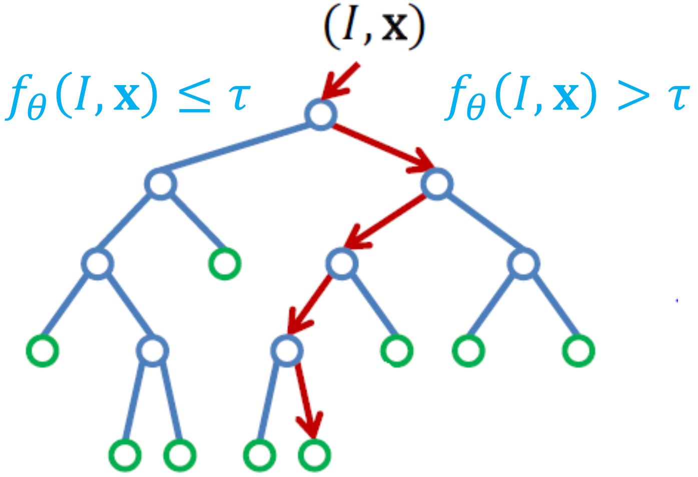
CVPR 2011

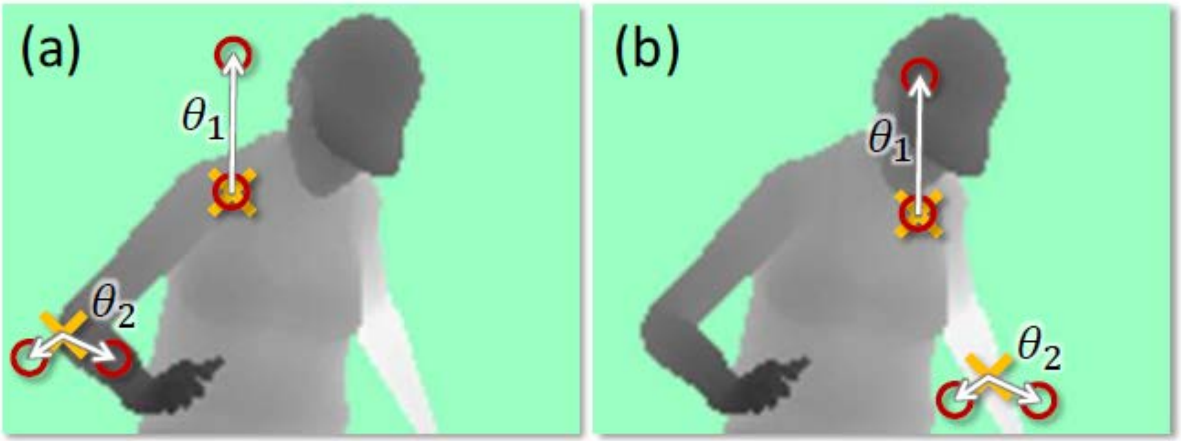
Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, Andrew Blake
Microsoft Research Cambridge & Xbox Incubation

# Mocap for Kinnect [Shotton+2011]

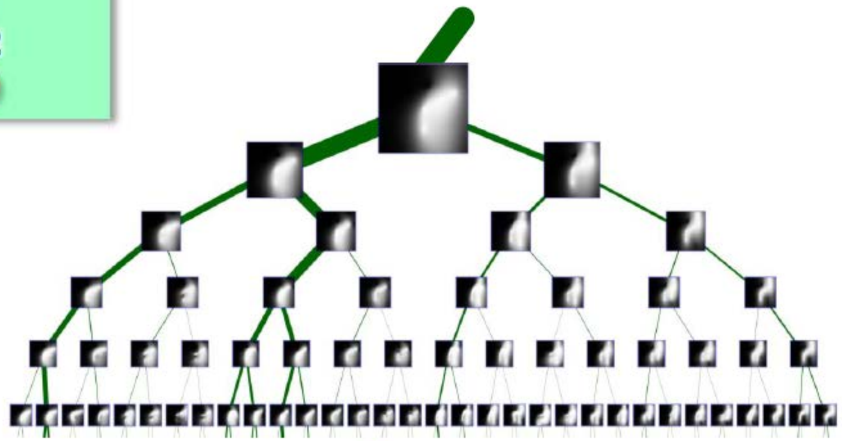
- For each pixel **x**, its class is predicted by a decision tree
- At each node, difference in depth at two chosen points **u** and **v** is computed and used as a feature

$$f_\theta(I,\mathbf{x}) = d_I\left(\mathbf{x} + \frac{\mathbf{u}}{d_I(\mathbf{x})}\right) - d_I\left(\mathbf{x} + \frac{\mathbf{v}}{d_I(\mathbf{x})}\right)$$


$(I,\mathbf{x})$
$f_\theta(I,\mathbf{x}) \leq \tau$
$f_\theta(I,\mathbf{x}) > \tau$



Ensemble of such decision trees are used: *Random forest*



# History of Computer Vision

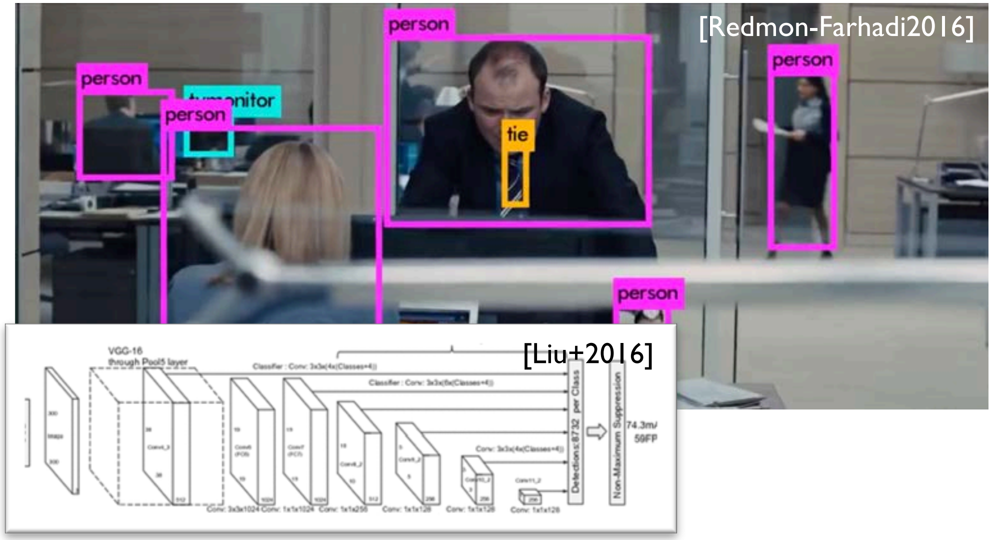
- Math and physics-based model (1980–)
  - Multi-view geometry & physics-based vision
  - Apps: computer graphics, augmented reality, etc.
  - Examples
    - Photometric stereo, Optic flow estimation, SIFT, Structure-from-Motion, Blind deblurring
- Introduction of machine learning (2000–)
  - Apps: surveillance camera, driver assistance, etc.
  - Examples
    - Face detection (cascaded classifier), Mocap for Kinnect (random forest)
- Paradigm shift: deep learning (2010–)
  - Explosive developments
  - Apps: potentially every problem

# VI. Paradigm shift: deep learning

- Deep learning has been applied to almost every problem
- Every attempt has achieved a great success almost without any exception!
  - Problems that were (considered to be) too hard to solve in the past
    - Object recognition, semantic segmentation, human pose estimation, monocular depth estimation, image captioning and many other tasks related to "image understanding", etc.
  - Problems for which good solutions were (considered to be) already found
    - Improved performance in terms of accuracy or computational speed
    - Face detection/recognition, object detection, optical flow estimation, stereo matching, super-resolution, etc.
  - Unclear yet if DL approaches surpass old solutions
    - Structure-from-motion, visual SLAM, etc.

# Recognition

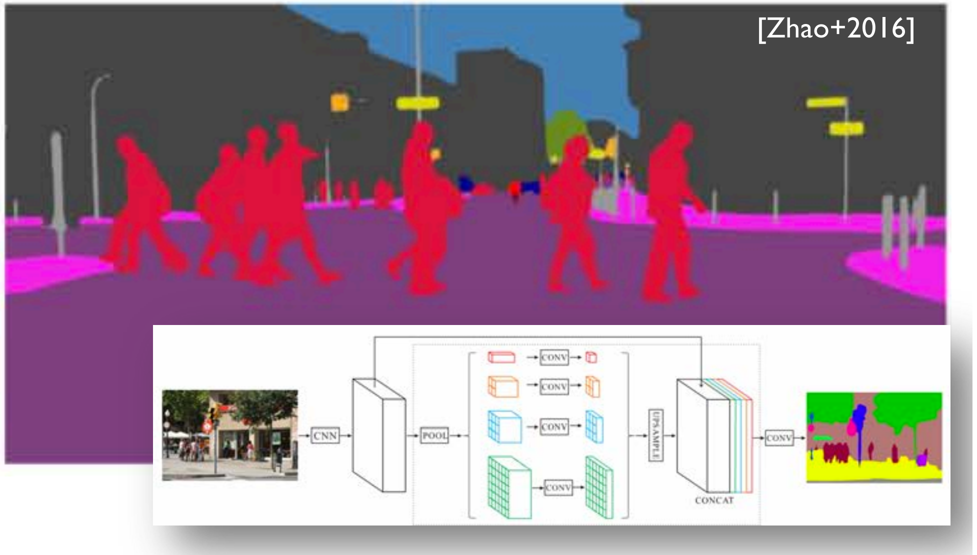
## Object detection



[Redmon-Farhadi2016]

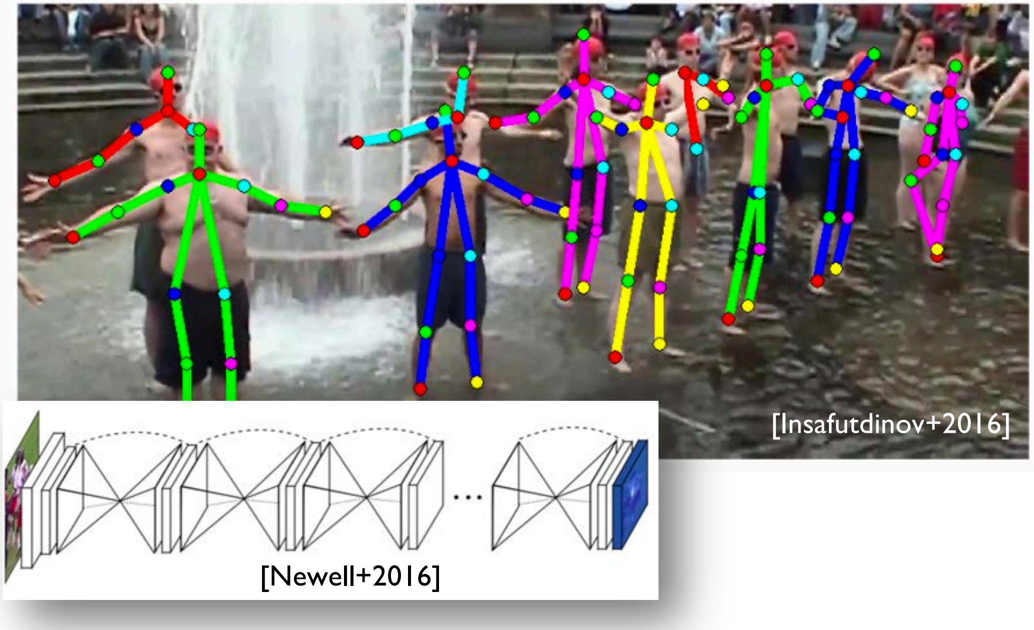
[Liu+2016]

## Semantic segmentation



[Zhao+2016]

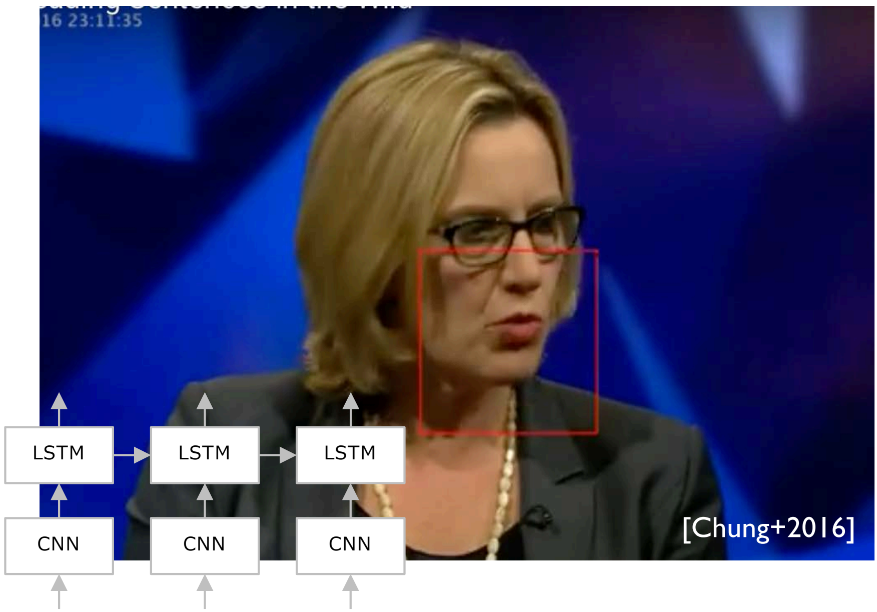
## Human pose detection



[Insafutdinov+2016]

[Newell+2016]

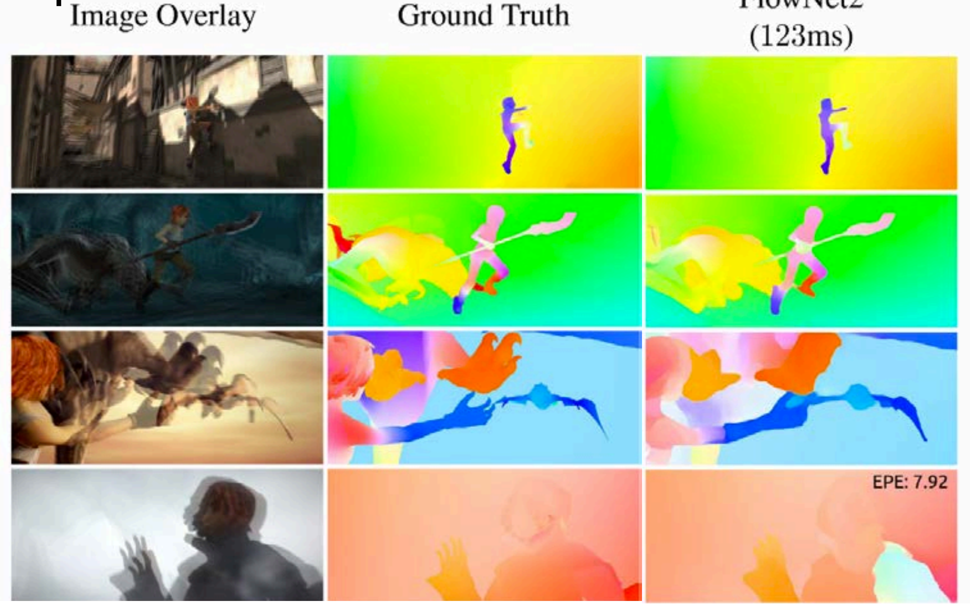
## Lip reading

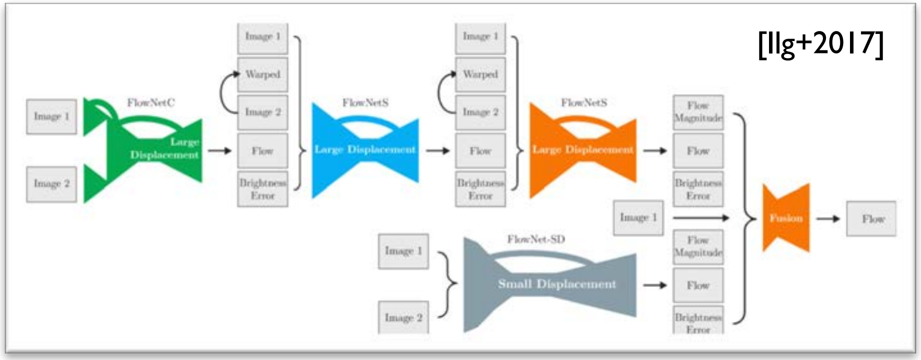


[Chung+2016]

# Motion and geometry

## Optical flow

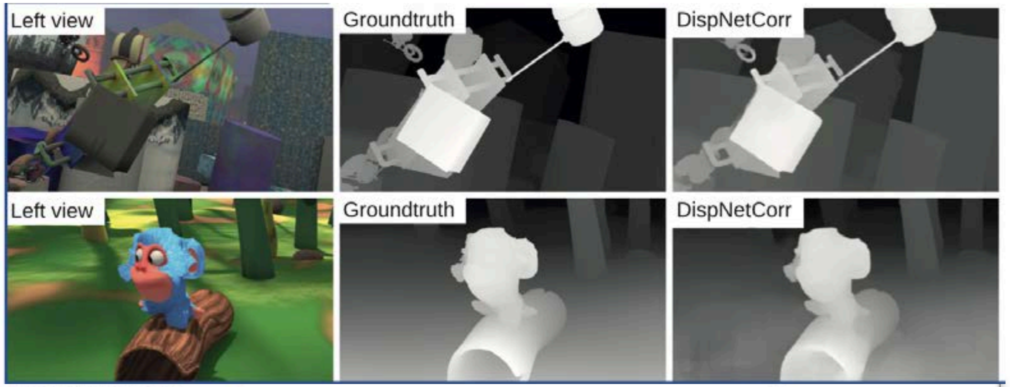


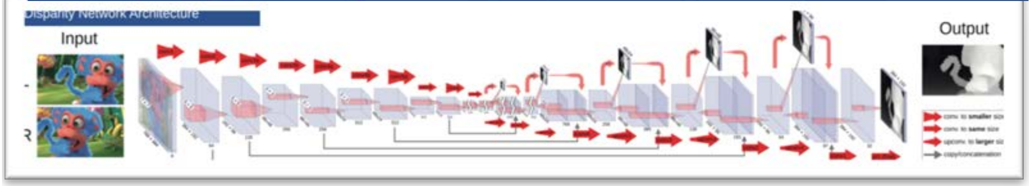


[Ilg+2017]

## Stereo matching

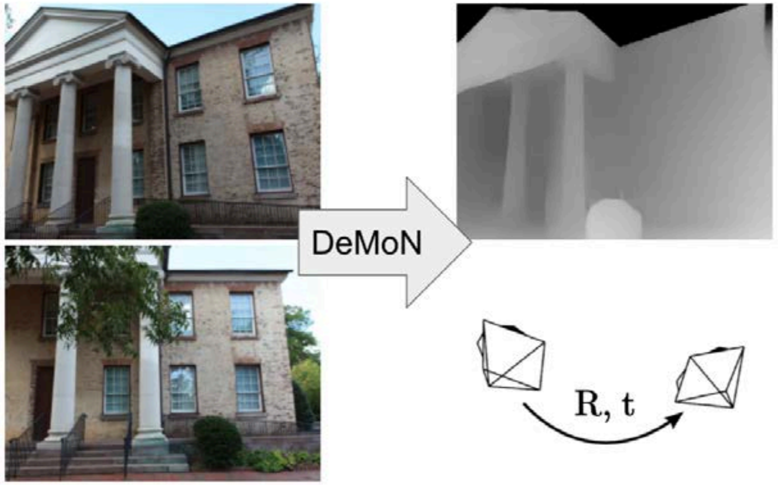
[Mayer+2017]

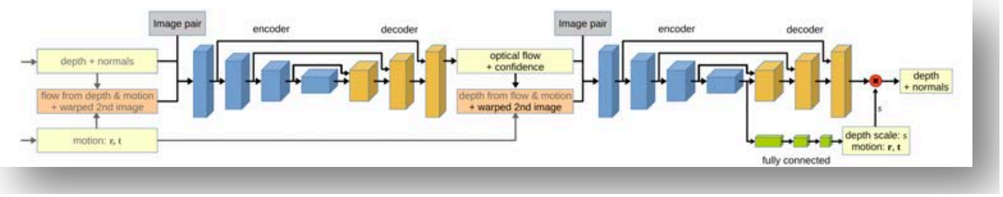




## Depth & cam. pose

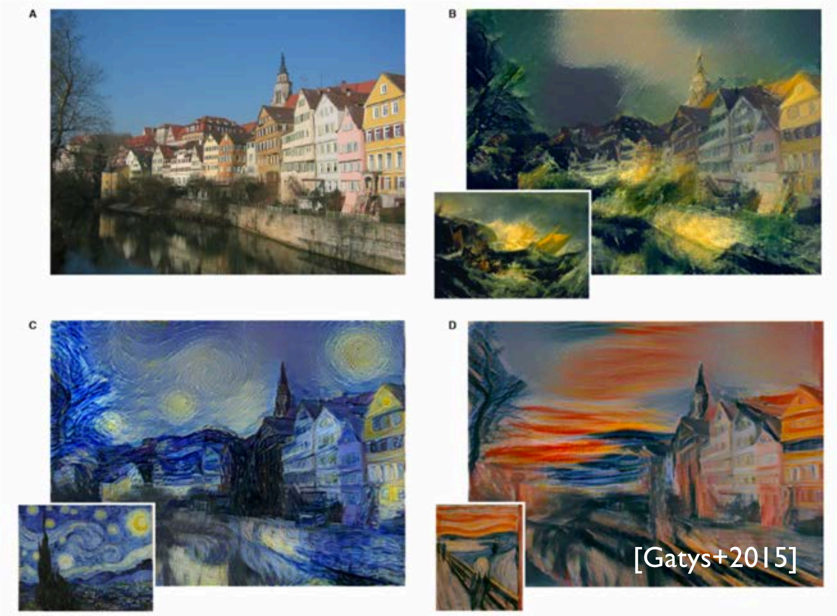
[Ummenhofer+2017]





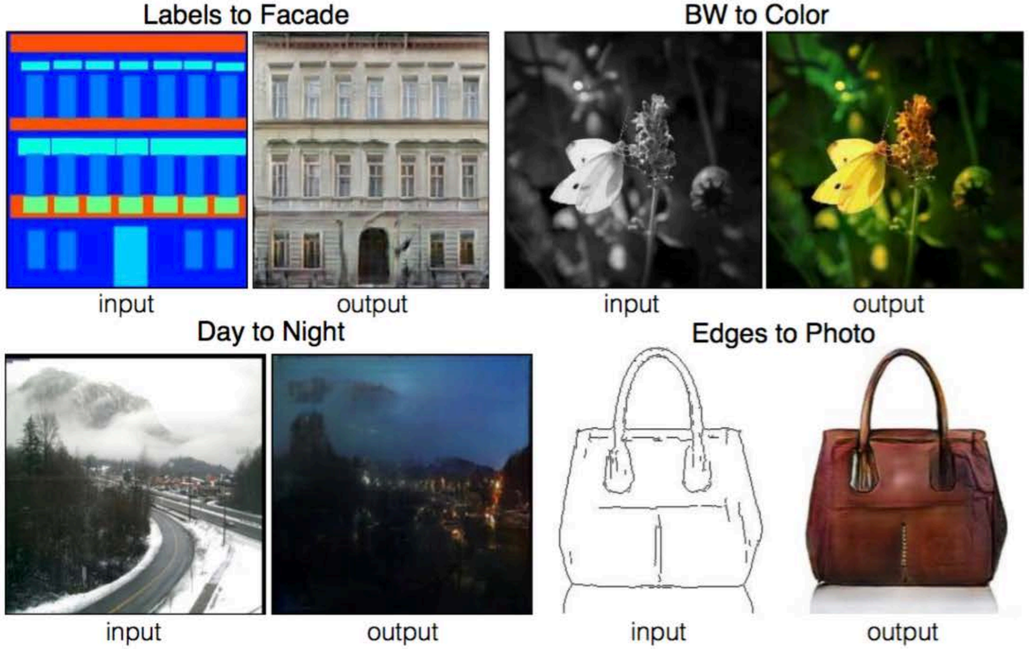
# Image synthesis & conversion

## Style transfer

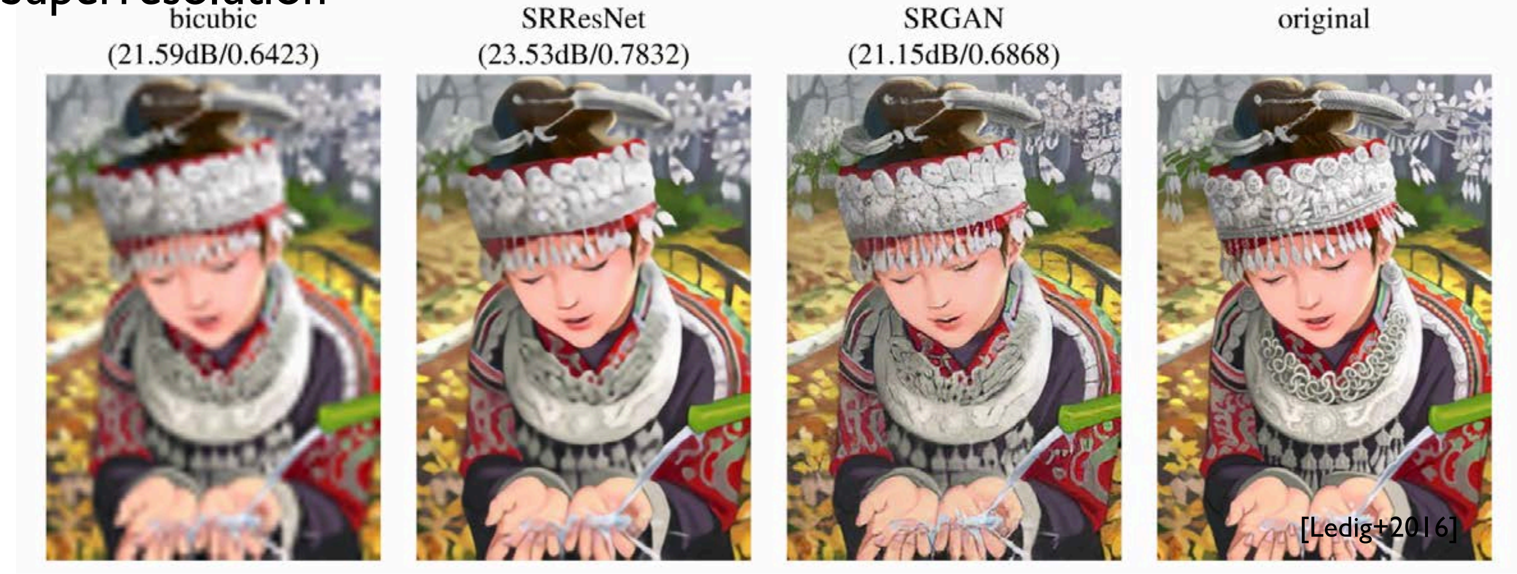


[Gatys+2015]

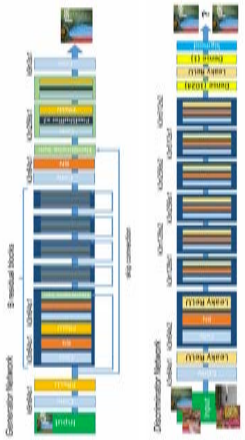
## Colorization, etc.



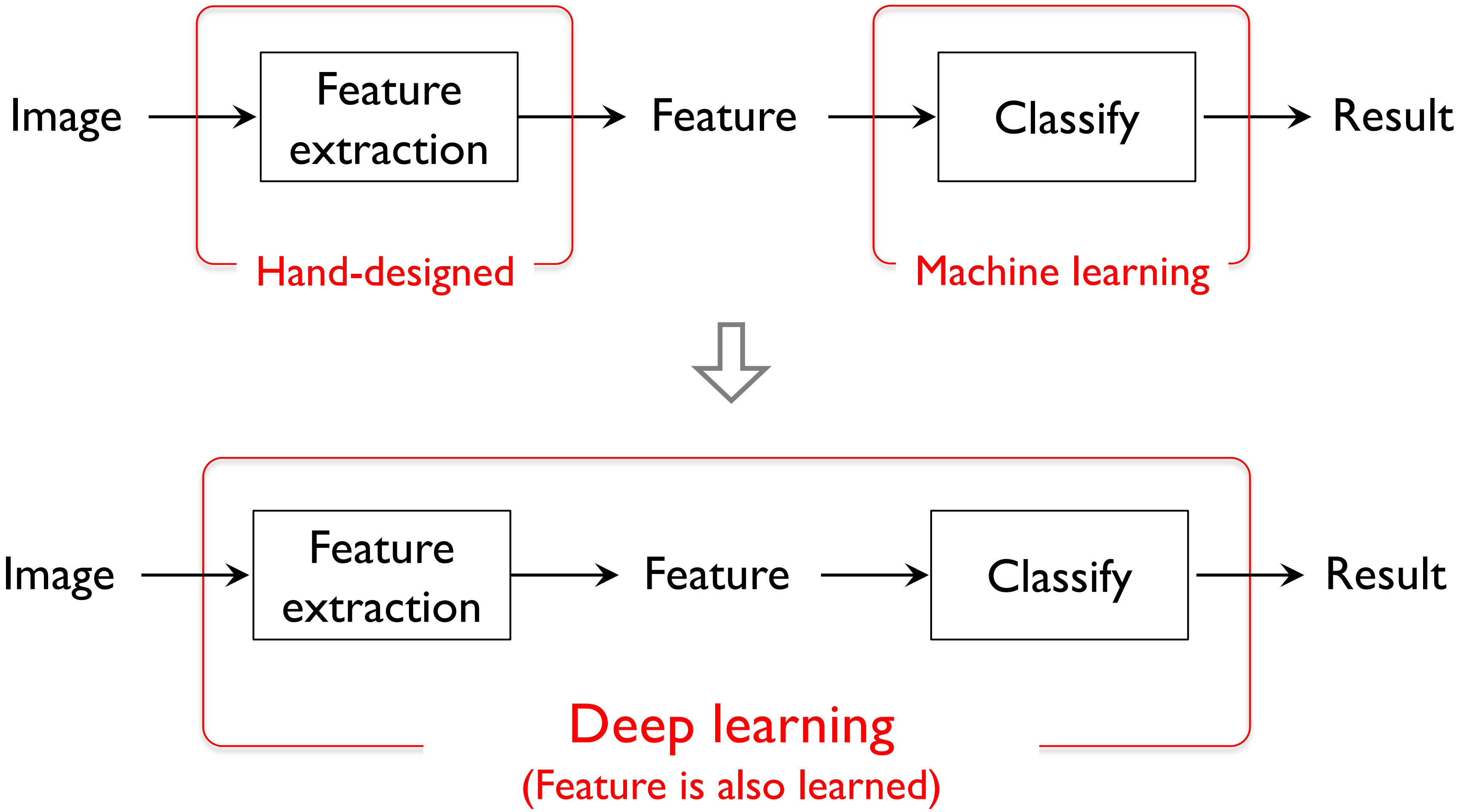
## Superresolution



[Ledig+2016]



# Paradigm has shifted…

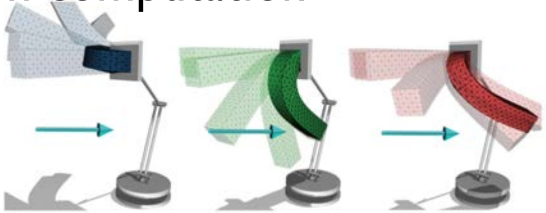


# Deep learning is beginning to go beyond CV/AI

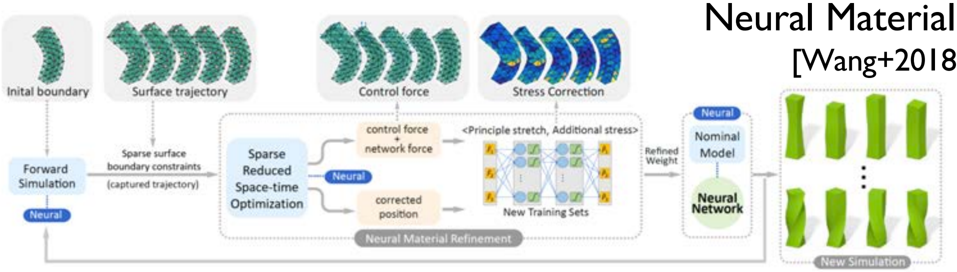
- Problems in all engineering/science fields → To solve unsolved problems
- Toward faster computation than simulation → Forward propagation is quick

## Structural deformation computation

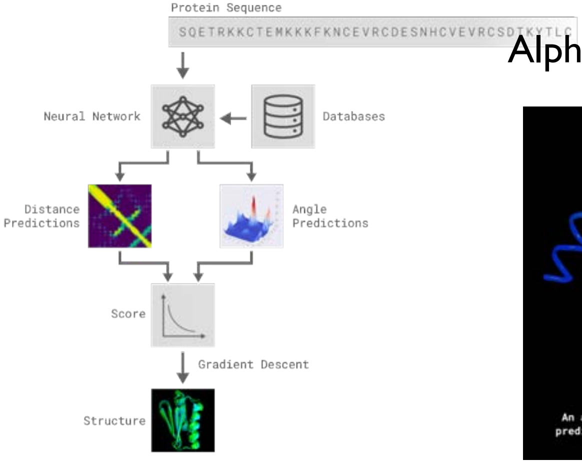
DeepWarp
[Luo+2018]



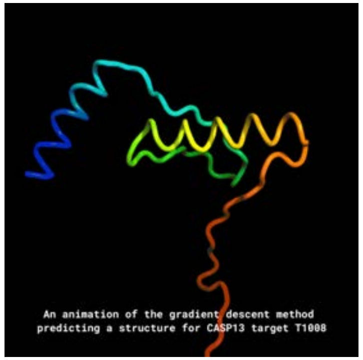
Neural Material
[Wang+2018]



## Protein folding

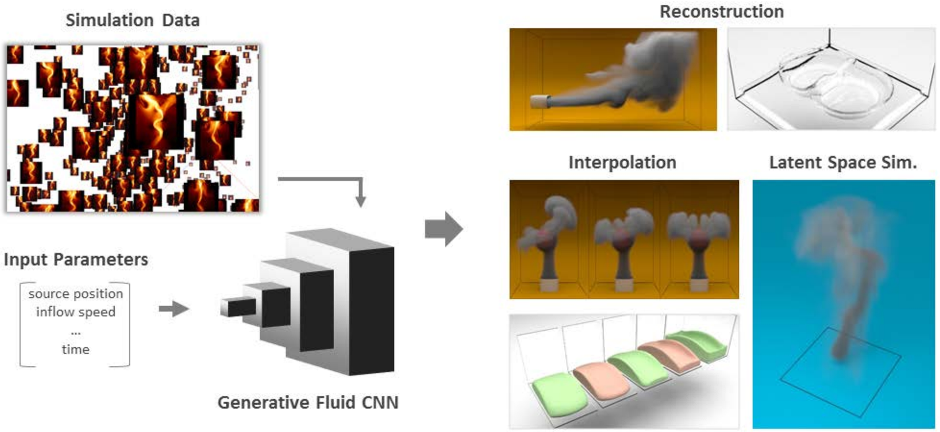


AlphaFold [DeepMind2018]


An animation of the gradient descent method predicting a structure for CASP13 target T1008

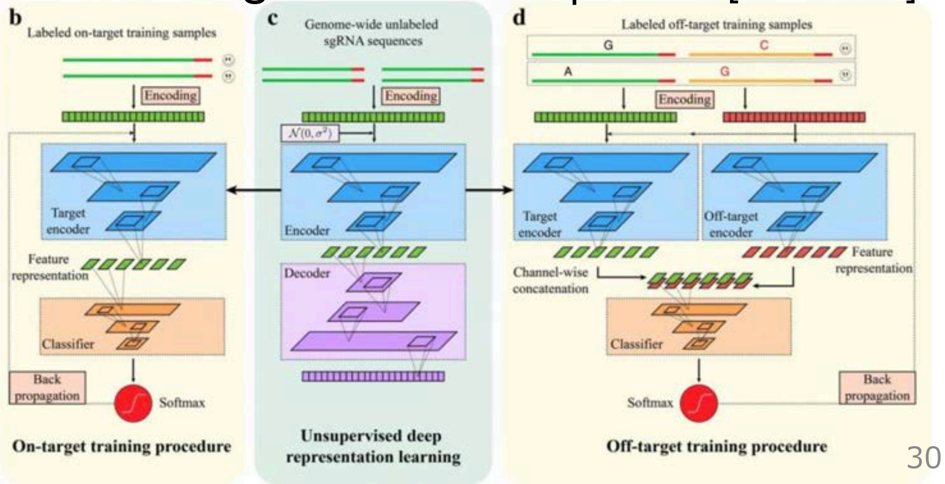
## Fluid mechanics

Deep Fluids [Kim+2018]



## Gene editing

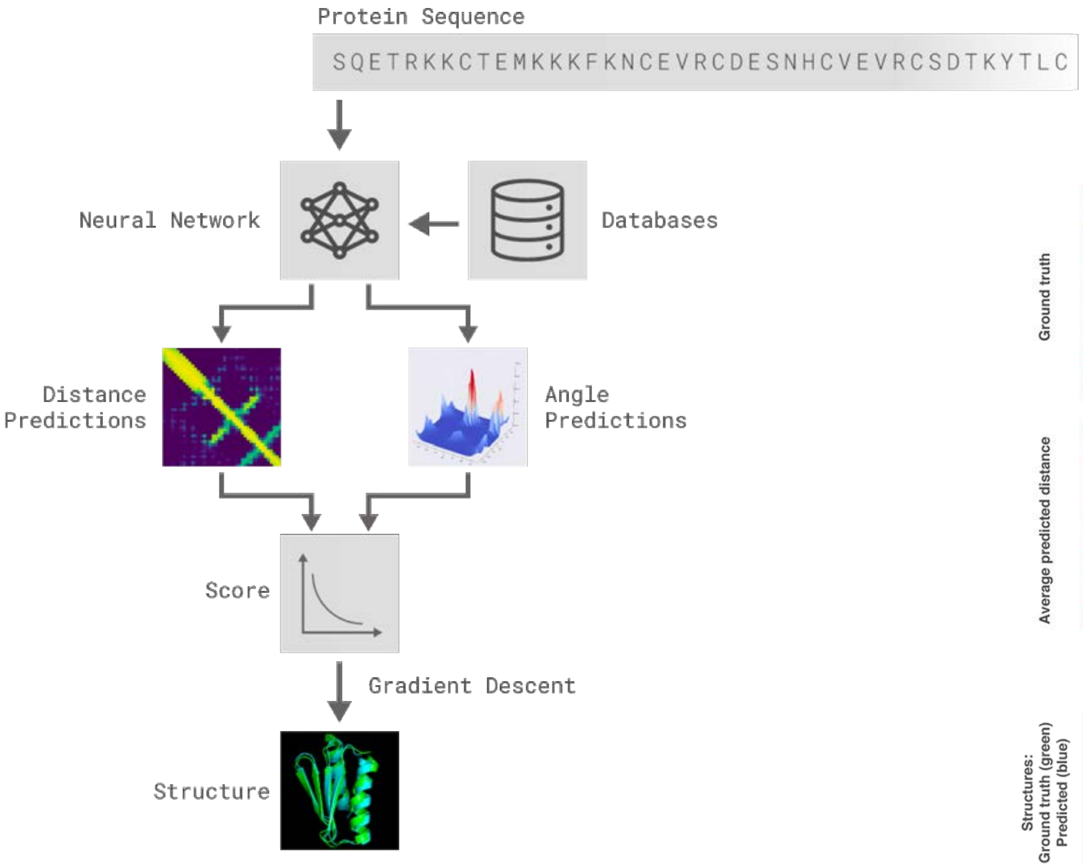
DeepCRISPR [Chuai+2018]


b
Labeled on-target training samples
Encoding
Target encoder
Feature representation
Classifier
Back propagation
Softmax
On-target training procedure
c
Genome-wide unlabeled sgRNA sequences
Encoding
$\mathcal{N}(0,\sigma^2)$
Encoder
Decoder
Unsupervised deep representation learning
d
Labeled off-target training samples
G
C
A
G
Encoding
Target encoder
Off-target encoder
Feature representation
Channel-wise concatenation
Classifier
Softmax
Back propagation
Off-target training procedure

# Protein folding problem

DeepMind, https://deepmind.com/blog/alphafold/

- Want to predict 3D structure of a protein from its genetic sequence
  - Distance and angle between pairs of amino acids are predicted
  - A score function based on these is minimized, yielding 3D structure



Protein Sequence

SQETRKKCTEMKKKFKNCEVRCDESNHCVEVRCSDTKYTLC

Neural Network

Databases

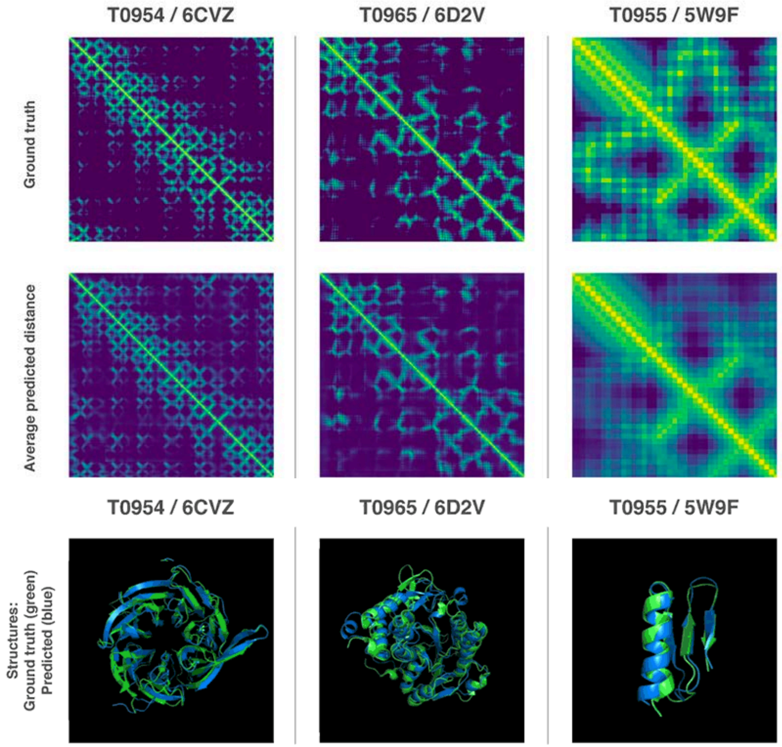
Distance Predictions

Angle Predictions

Score

Gradient Descent

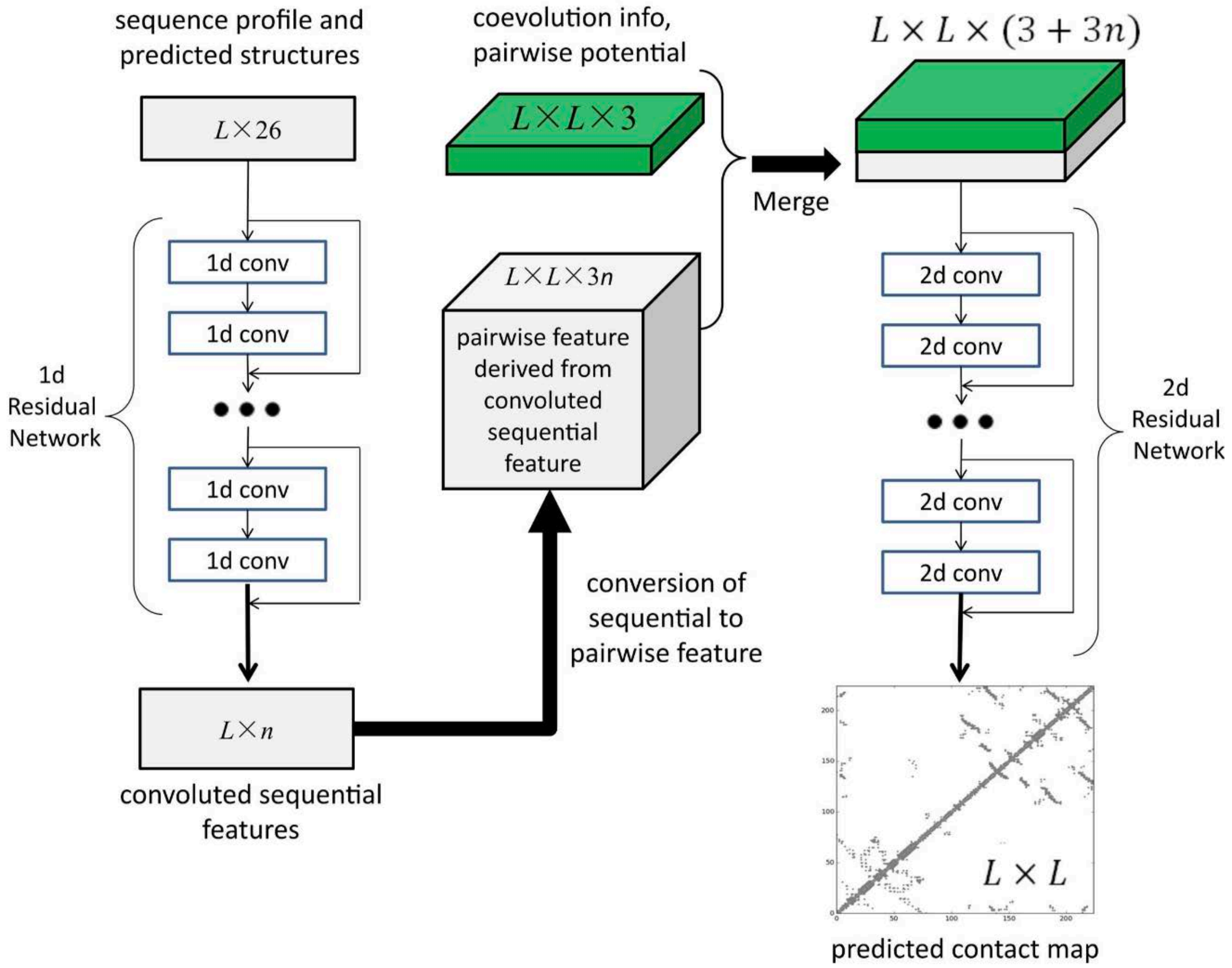
Structure



# Protein folding--- 1$^{st}$ successful application of DNNs

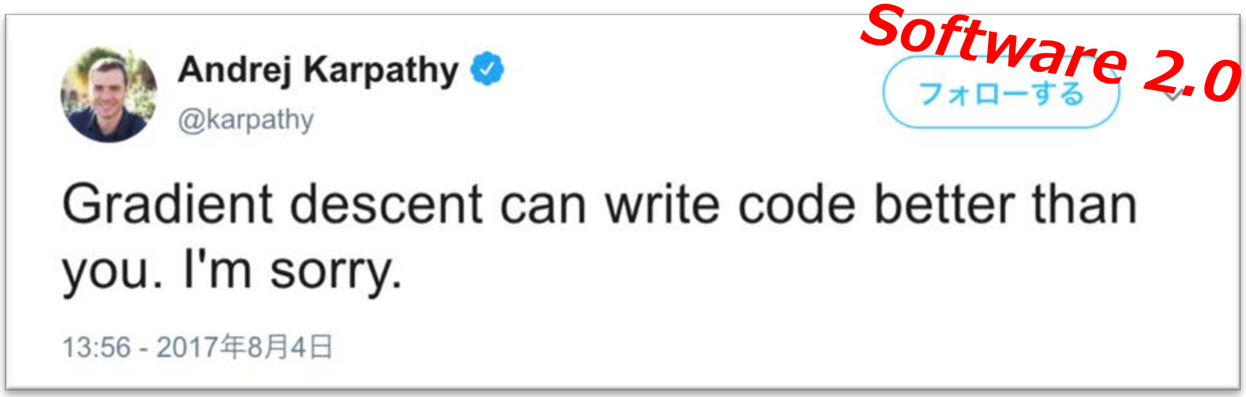
Wang, … Xu, Accurate DeNovo Prediction of Protein Contact Map by Ultra-Deep Learning Model, PLOS Computational Biology, 2017

- A DNN is applied to predict contacting probability of pairs of amino acids


sequence profile and predicted structures
$L \times 26$
1d Residual Network
1d conv
1d conv
1d conv
1d conv
$L \times n$
convoluted sequential features
coevolution info, pairwise potential
$L \times L \times 3$
$L \times L \times 3n$
pairwise feature derived from convoluted sequential feature
conversion of sequential to pairwise feature
Merge
$L \times L \times (3 + 3n)$
2d conv
2d conv
2d conv
2d conv
2d Residual Network
$L \times L$
predicted contact map

# Deep learning in practice

- To apply deep learning to solve a particular task…
  1. Design a network (including determination of inputs/outputs)
  2. Collect (a large, sometimes huge, amount of) training data



Software 2.0

Andrej Karpathy

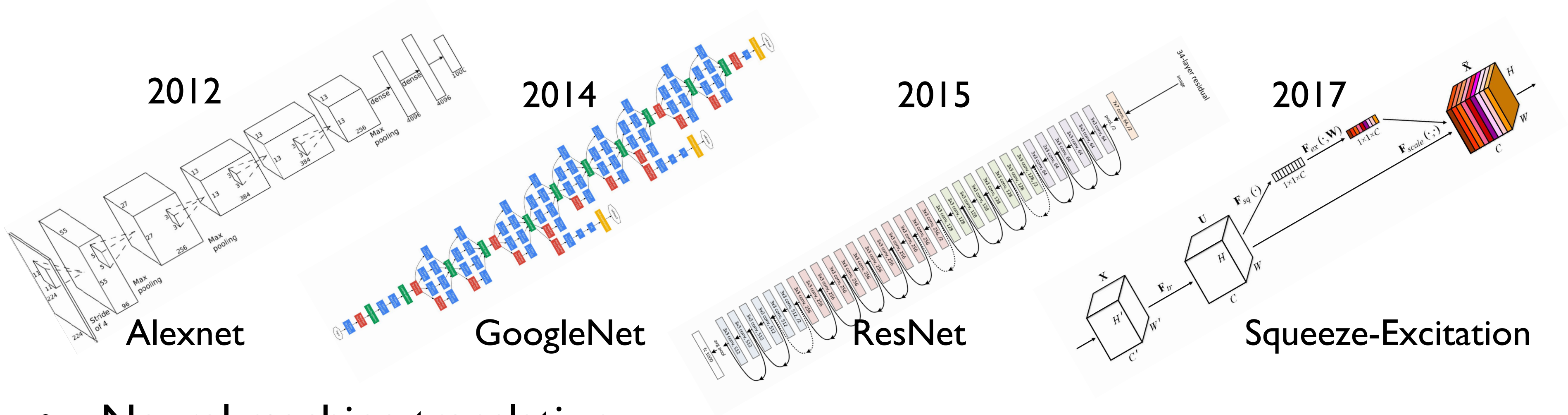
@karpathy

Gradient descent can write code better than you. I'm sorry.

13:56 - 2017年8月4日

# Advancements of network design

- ILSVRC Winners



2012 Alexnet

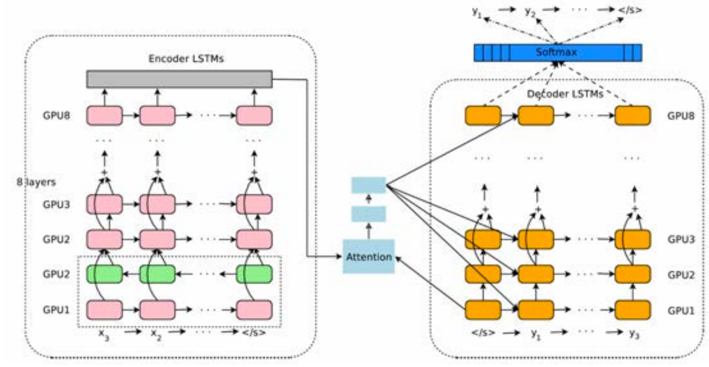
2014 GoogleNet

2015 ResNet

2017 Squeeze-Excitation

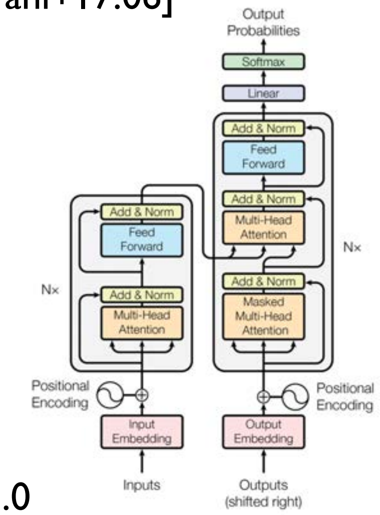
- Neural machine translation

Google's NMT
[Wu+16.09]



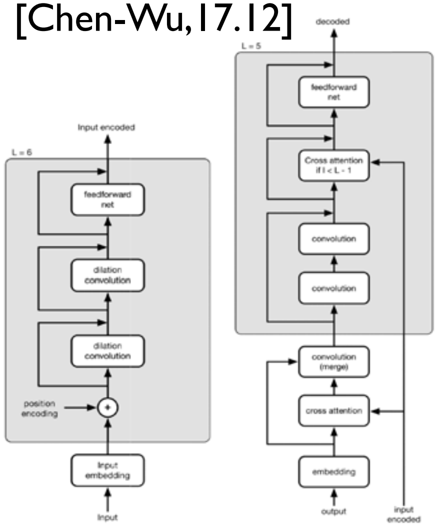
EN-FR BLUE: 39.92

Attention is all you need
[VasWani+17.06]



41.0

CNN is all you need
[Chen-Wu, 17.12]



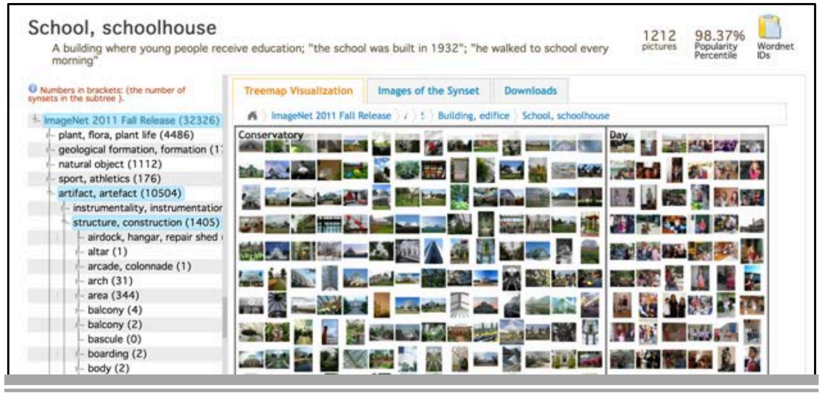
45.54

# Examples of datasets

## IMAGENET

http://image-net.org/index

- Object category recognition
- 21841 classes・14,197,122 images
- Stanford U, Princeton U



School, schoolhouse

A building where young people receive education; "the school was built in 1932"; "he walked to school every morning"

1212 pictures · 98.37% Popularity Percentile · Wordnet IDs

Numbers in brackets: (the number of synsets in the subtree ).

- ImageNet 2011 Fall Release (32326)
  - plant, flora, plant life (4486)
  - geological formation, formation (1
  - natural object (1112)
  - sport, athletics (176)
  - artifact, artefact (10504)
    - instrumentality, instrumentatior
    - structure, construction (1405)
      - airdock, hangar, repair shed
      - altar (1)
      - arcade, colonnade (1)
      - arch (31)
      - area (344)
      - balcony (4)
      - balcony (2)
      - bascule (0)
      - boarding (2)
      - body (2)

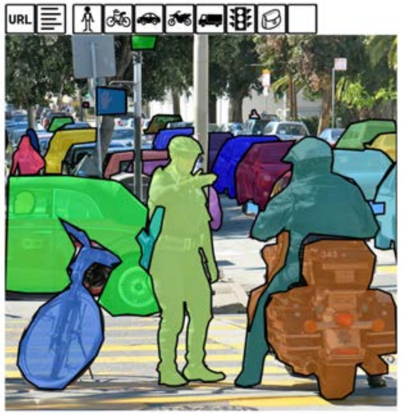
Treemap Visualization | Images of the Synset | Downloads

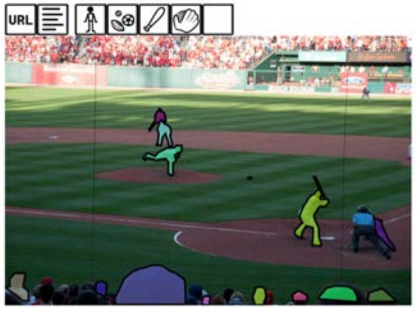
ImageNet 2011 Fall Release > Building, edifice > School, schoolhouse

## COCO

http://cocodataset.org

- Object classes and segmentation
- 80 objects + 91 stuffs; 330,000 images
- Cornel U, Microsoft

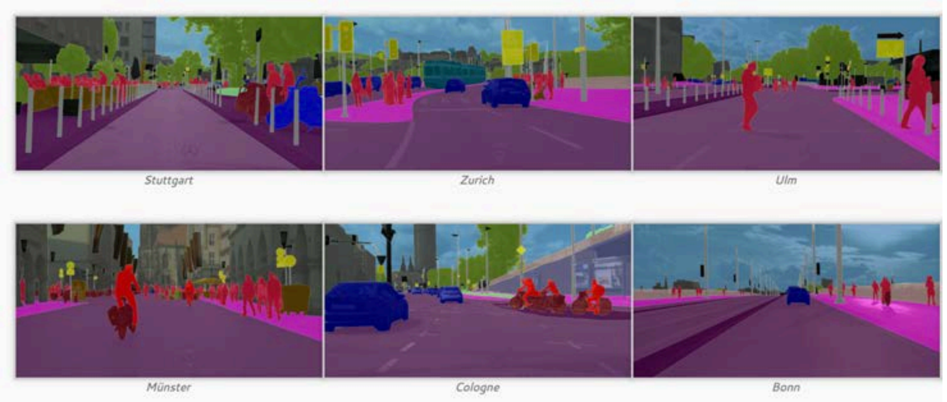




## CITYSCAPES

https://www.cityscapes-dataset.com/examples/

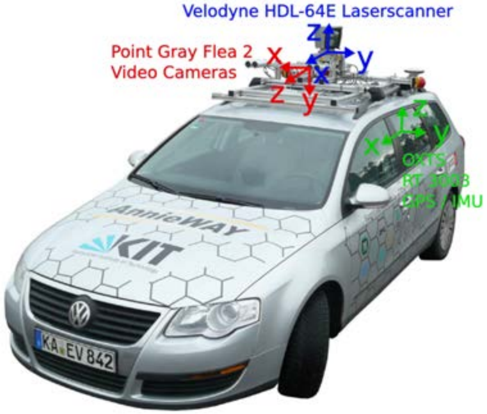
- Semantic segmentation
- 30 classes・50 cities・ 5 ,000/20,000 images



## KITTI Vision

http://www.cvlibs.net/datasets/kitti/index.php

- Stereo vision, optical flow etc.>100GB
- Karlsruhe Institute of Tech., TTI Chicago

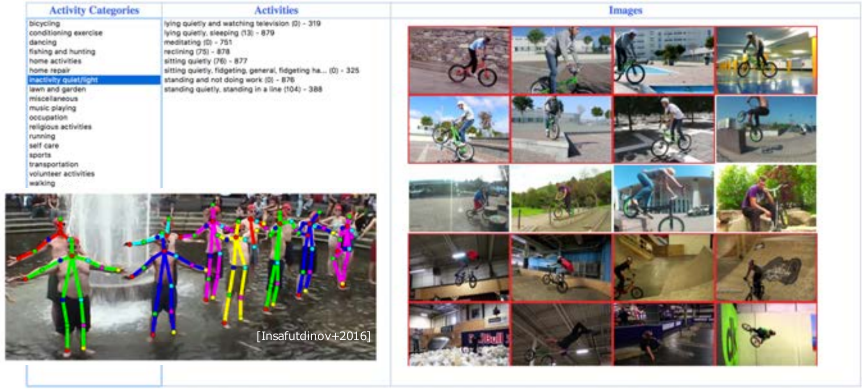


# Examples of datasets

## MPII Human Pose Dataset

http://human-pose.mpi-inf.mpg.de

- Human pose
- All joints; 40,000 persons; 25,000 images
- Max Planck Institute Informatik



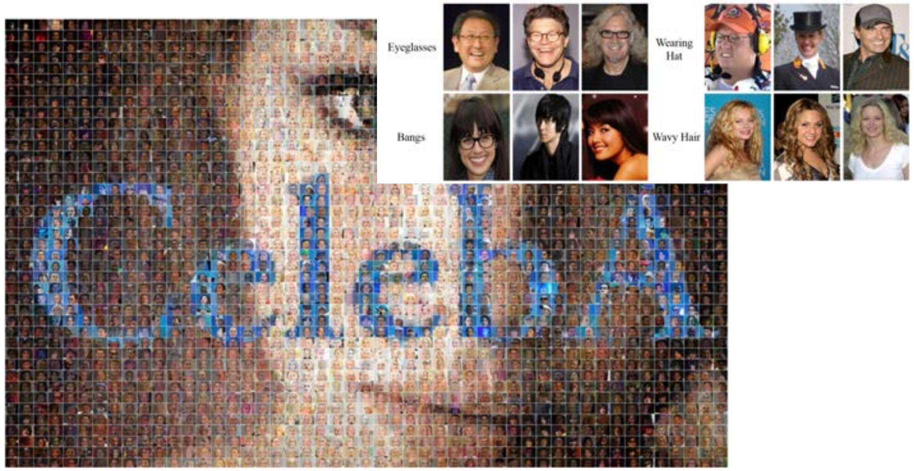
| Activity Categories | Activities |
|---|---|
| bicycling<br>conditioning exercise<br>dancing<br>fishing and hunting<br>home activities<br>home repair<br>inactivity quiet/light<br>lawn and garden<br>miscellaneous<br>music playing<br>occupation<br>religious activities<br>running<br>self care<br>sports<br>transportation<br>volunteer activities<br>walking | lying quietly and watching television (0) - 319<br>lying quietly, sleeping (13) - 879<br>meditating (0) - 751<br>reclining (75) - 878<br>sitting quietly (76) - 877<br>sitting quietly, fidgeting, general, fidgeting ha... (0) - 325<br>standing and not doing work (0) - 876<br>standing quietly, standing in a line (104) - 388 |

[Insafutdinov+2016]

## CelebA

http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

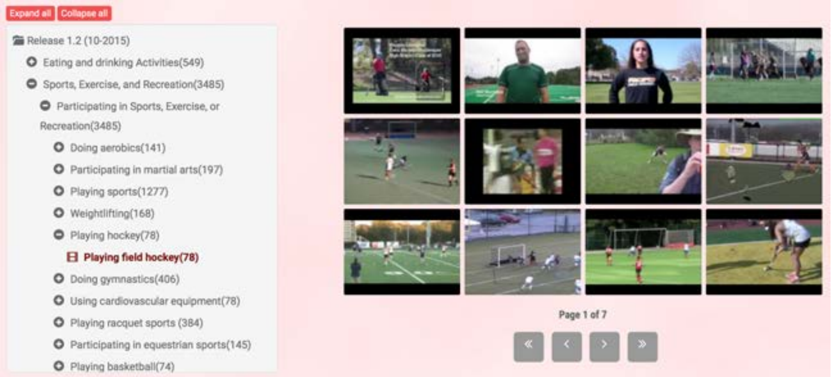
- Face images
- 40 attributes・10,177 persons・202,599 images
- 香港中文大学



## ACTIVITYNET

http://activity-net.org

- Human activity in video
- 200 classes; 20,000 video clips (648 hours)
- サウジ王立科技大・ノルテ大（コロンビア）



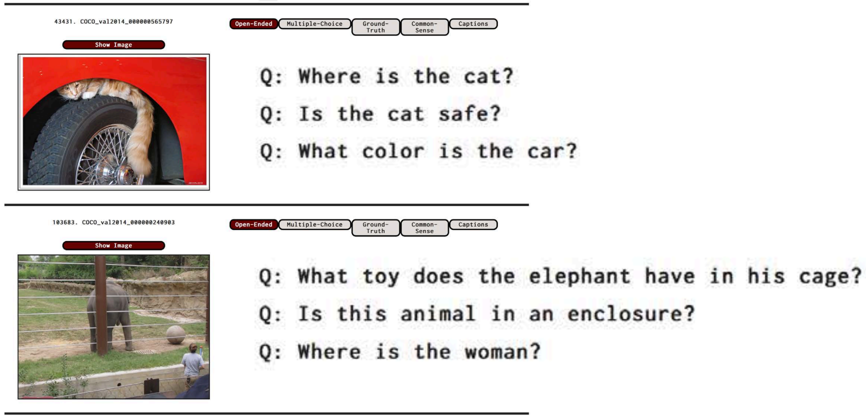
Release 1.2 (10-2015)
- Eating and drinking Activities(549)
- Sports, Exercise, and Recreation(3485)
  - Participating in Sports, Exercise, or Recreation(3485)
    - Doing aerobics(141)
    - Participating in martial arts(197)
    - Playing sports(1277)
    - Weightlifting(168)
    - Playing hockey(78)
      - **Playing field hockey(78)**
    - Doing gymnastics(406)
    - Using cardiovascular equipment(78)
    - Playing racquet sports (384)
    - Participating in equestrian sports(145)
    - Playing basketball(74)

Page 1 of 7

## VQA

http://visualqa.org/index.html

- Visual question and answering
- 5.4 questions per image; 10 ans; 265,016 images
- Virginia Tech., Georgia Tech.



43431. COCO_val2014_000000565797

Q: Where is the cat?

Q: Is the cat safe?

Q: What color is the car?

103683. COCO_val2014_000000240903

Q: What toy does the elephant have in his cage?

Q: Is this animal in an enclosure?

Q: Where is the woman?

# Summary

- Math and physics-based model (1980–)
  - Multi-view geometry & physics-based vision
  - Apps: computer graphics, augmented reality, etc.
  - Examples
    - Photometric stereo, Optic flow estimation, SIFT, Structure-from-Motion, Blind deblurring
- Introduction of machine learning (2000–)
  - Apps: surveillance camera, driver assistance, etc.
  - Examples
    - Face detection (cascaded classifier), Mocap for Kinnect (random forest)
- Paradigm shift: deep learning (2010–) ← We are here!
  - Explosive developments
  - Apps: potentially every problem

What is the next?