Big Data and Deep Learning
Case Study: Face Recognition

October 25, 2018

Chantana Chantrapornchai
HPCNC Laboratory, Dept. of Computer Engineering
Kasetsart University

KU KASETSART UNIVERSITY

NVIDIA GPU EDUCATION CENTER

HPCNC High Performance Computing and Networking Center

Outline

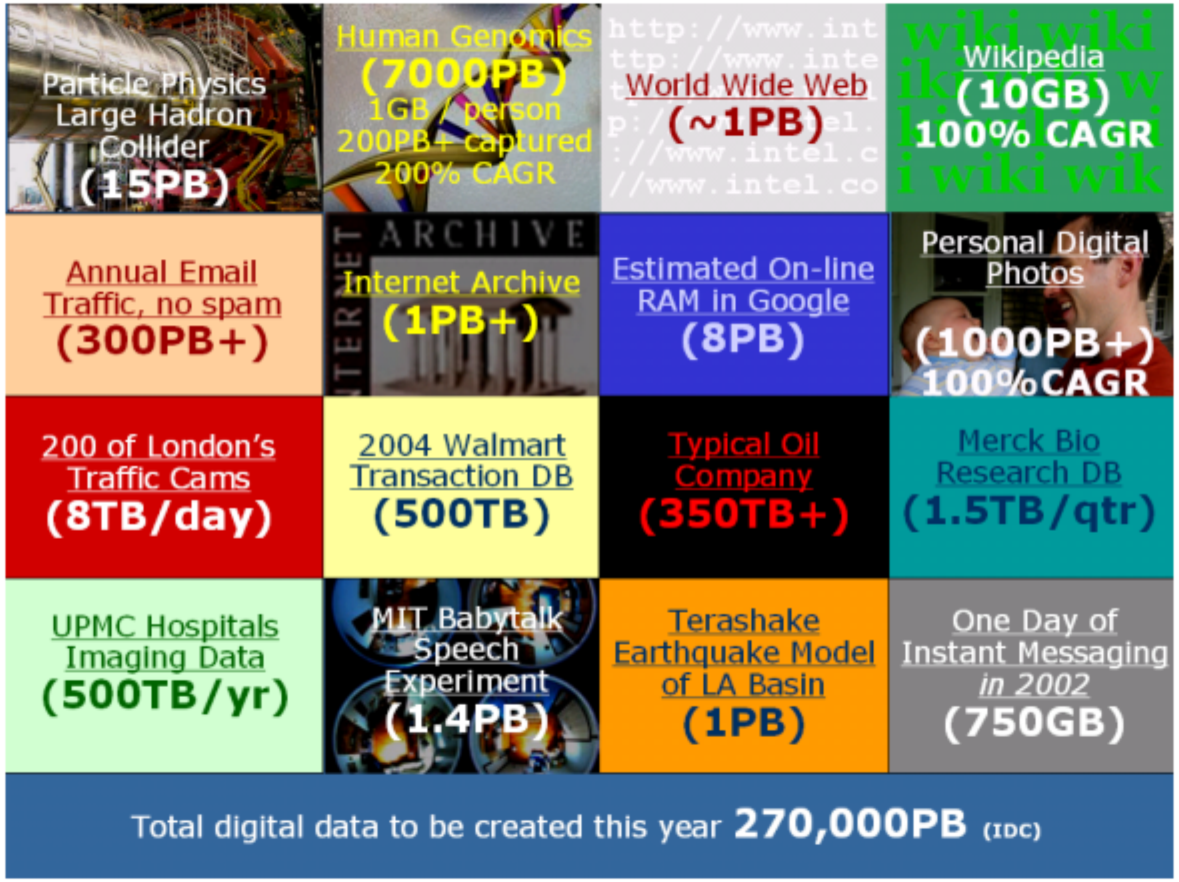
- Introduction to big data, deep learning, HPC
- Top applications of deep learning
- Case study: Hands on Face recognition

What is big data? **- 3 V's**

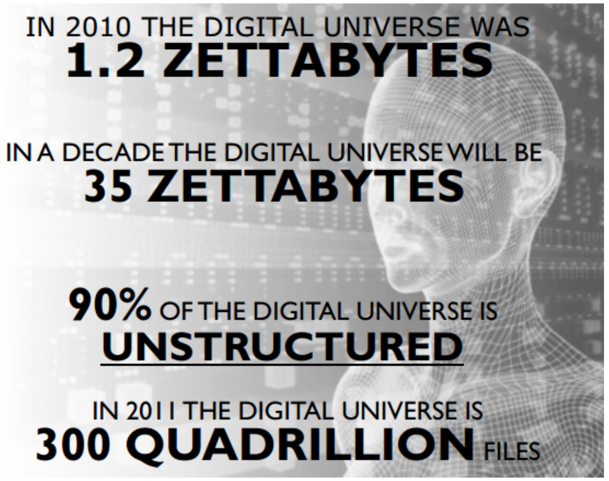
- Big Volume
  - With simple (SQL) analytics
  - With complex (non-SQL) analytics
- Big Velocity
  - How fast data is processed
- Big Variety
  - Large number of diverse data sources to integrate

http://whatis.techtarget.com/definition/3Vs

http://www.dummies.com/how-to/content/defining-big-data-volume-velocity-and-variety.html



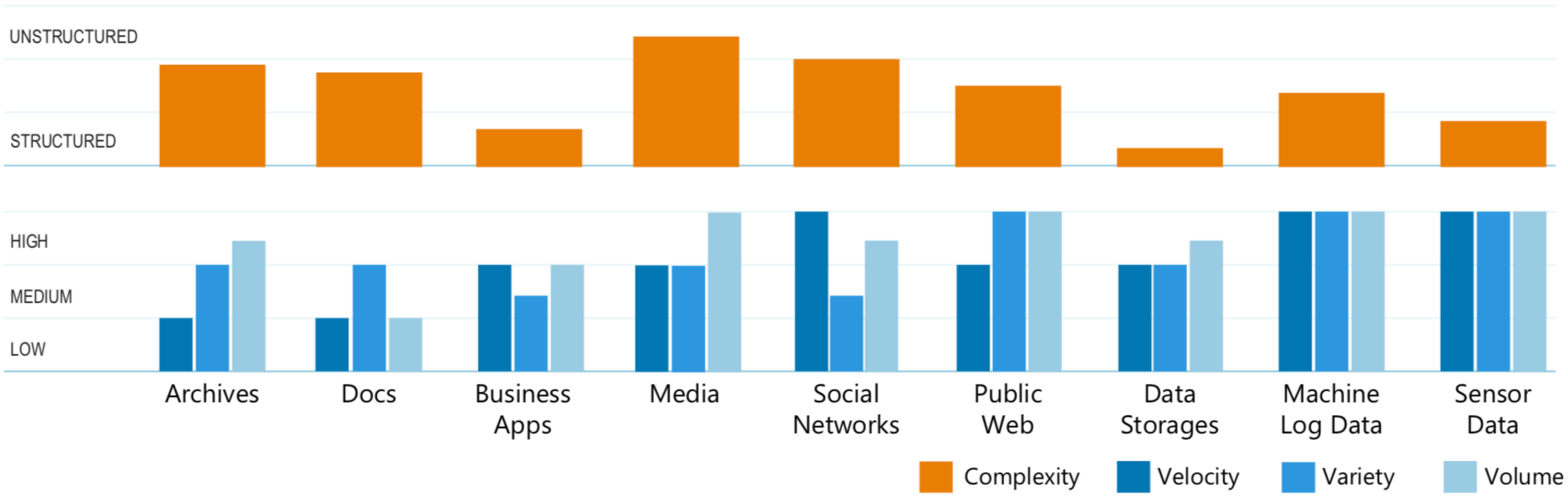
The Economist, Feb 25, 2010



WIRED The New York Times Bloomberg Businessweek Forbes WALL STREET JOURNAL

1 ZB = 1000^7bytes = 10^{21} bytes = 1000000000000000000000 bytes = 1000 exabytes = 1 billion tera bytes.

http://www.snia.org/sites/default/files2/ABDS2012/Tutorials/RobPeglar_Introduction_Analytics%20_Big%20Data_Hadoop.pdf



Archives
Scanned documents, statements, medical records, e-mails etc..

Docs
XLS, PDF, CSV, HTML, JSON etc.

Business Apps
CRM, ERP systems, HR, project management etc.

Media
Images, video, audio etc.

Social Networks
Twitter, Facebook, Google+, LinkedIn etc.

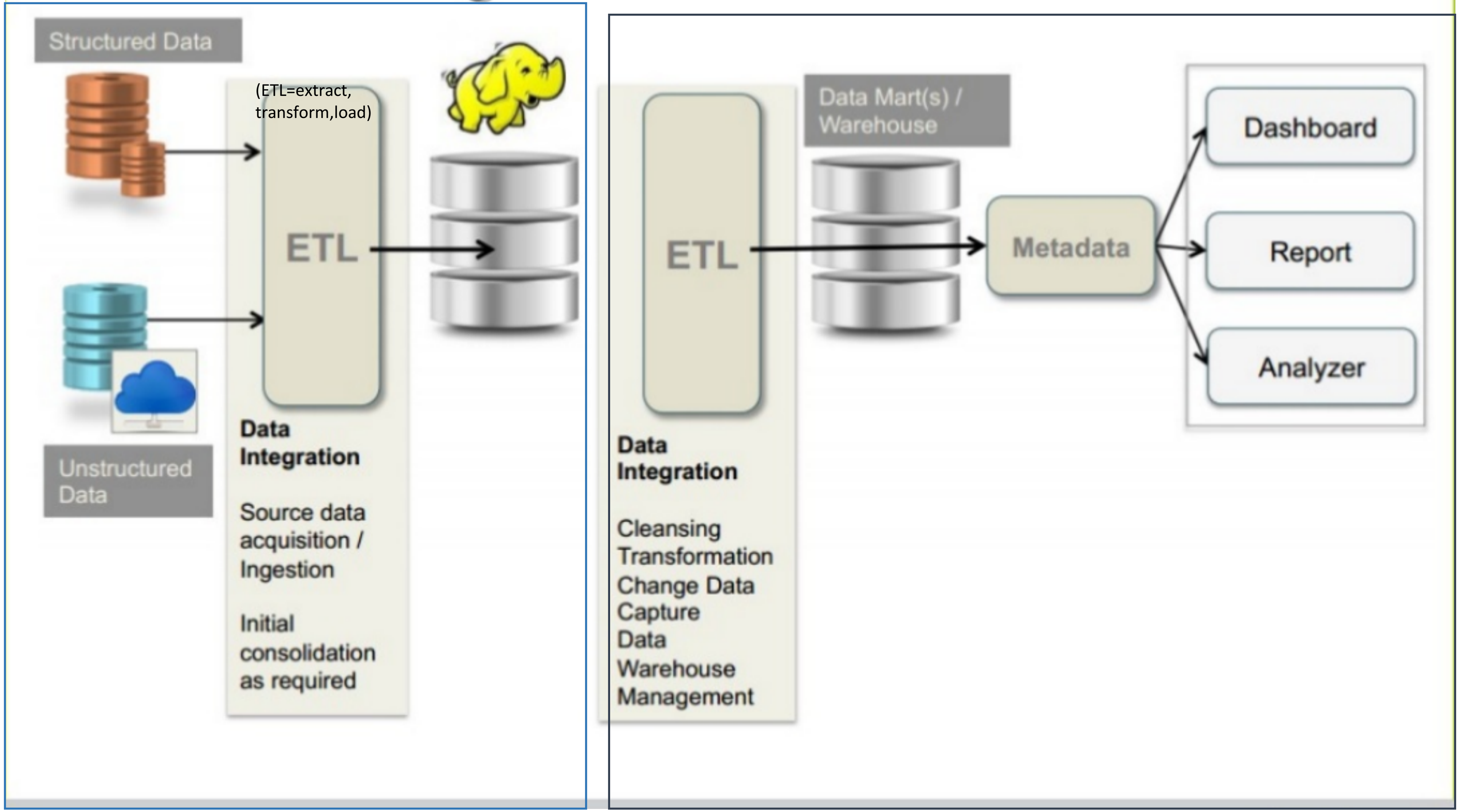
Public Web
Wikipedia, news, weather, public finance etc

Data Storages
RDBMS, NoSQL, Hadoop, file systems etc.

Machine Log Data
Application logs, event logs, server data, CDRs, clickstream data etc.

Sensor Data
Smart electric meters, medical devices, car sensors, road cameras etc.

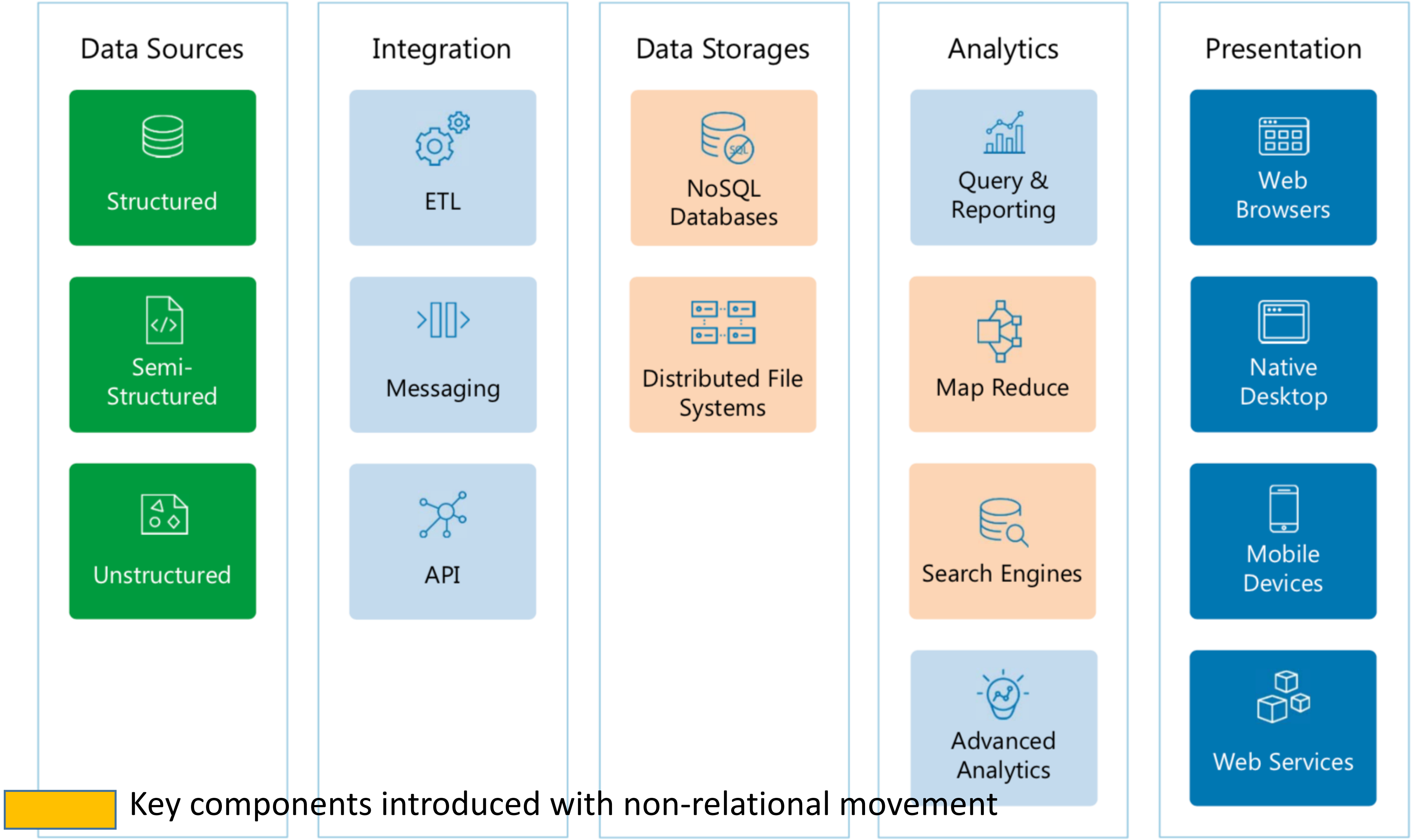
Big Data Architecture

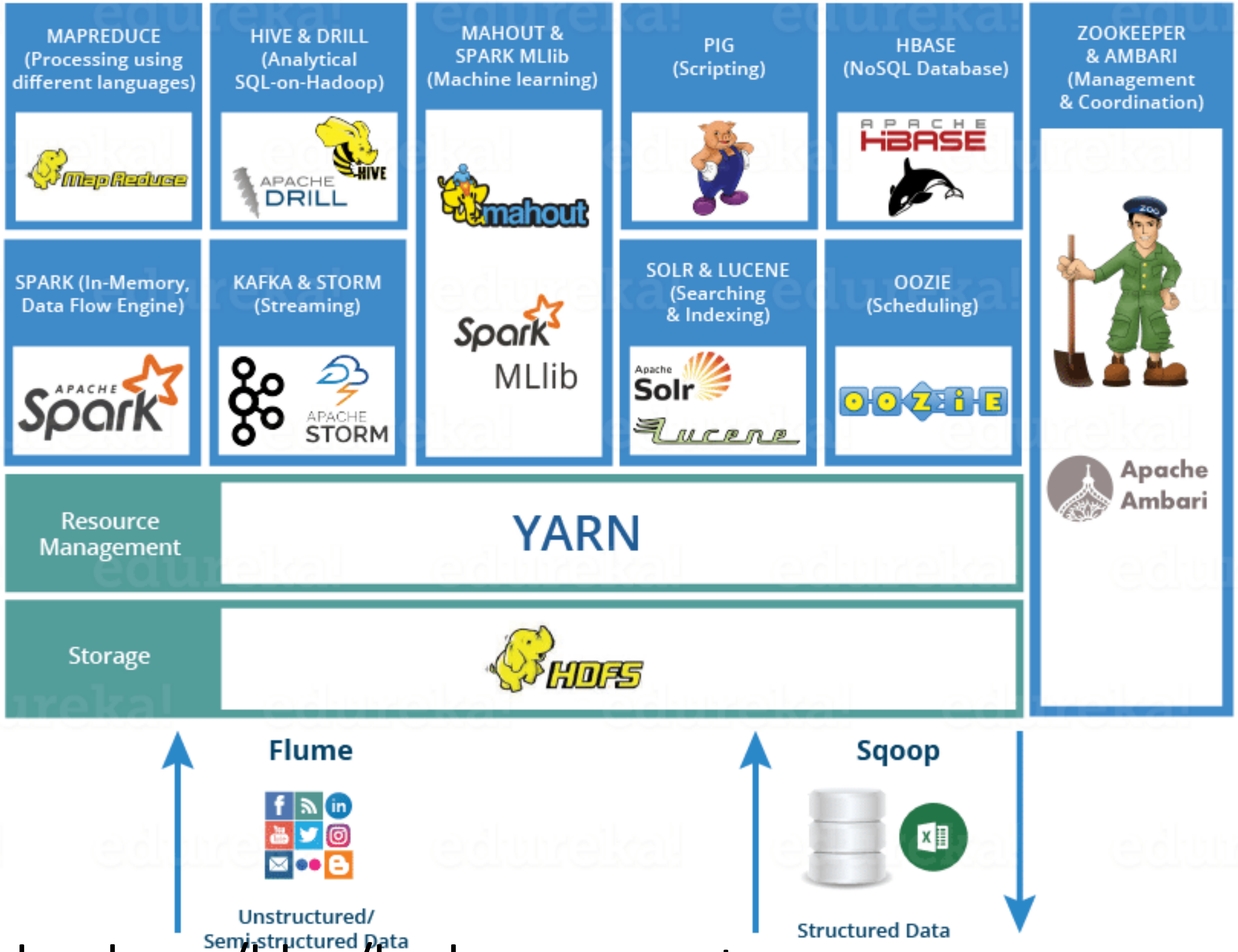


	Traditional Analytics (BI)	vs Big Data Analytics
Focus on	• Descriptive analytics • Diagnosis analytics	• **Predictive analytics** • **Data Science**
Data Sets	• Limited data sets • Cleansed data • Simple models	• Large scale data sets • More types of data • Raw data • Complex data models
Supports	**Causation:** what happened, and why?	**Correlation**: new insight More accurate answers

https://resources.sei.cmu.edu/asset_files/Presentation/2014_017_101_89659.pdf

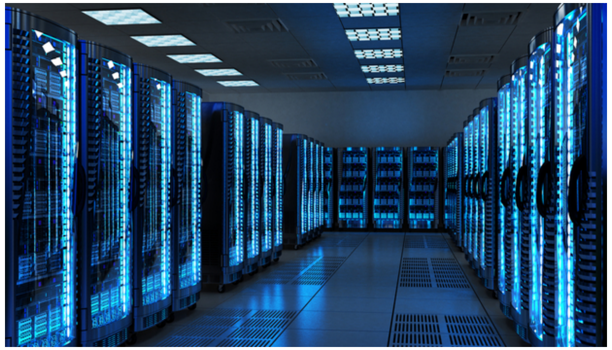
Non-Relational Reference Architecture

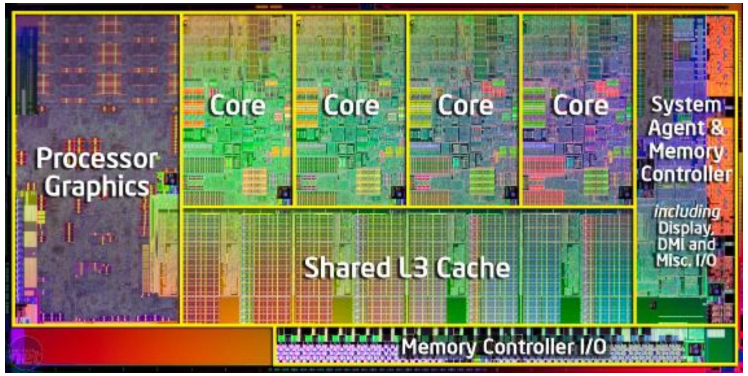


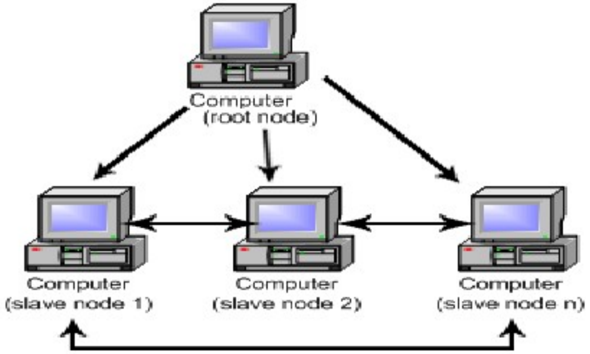


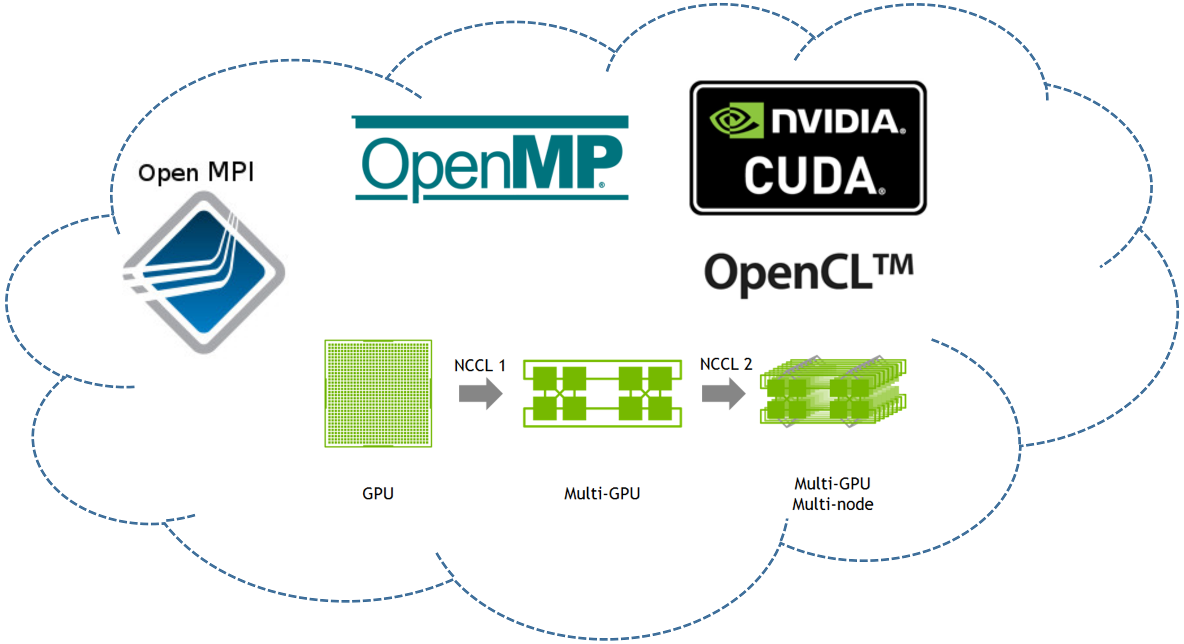
https://www.edureka.co/blog/hadoop-ecosystem

HPC- High performance computing

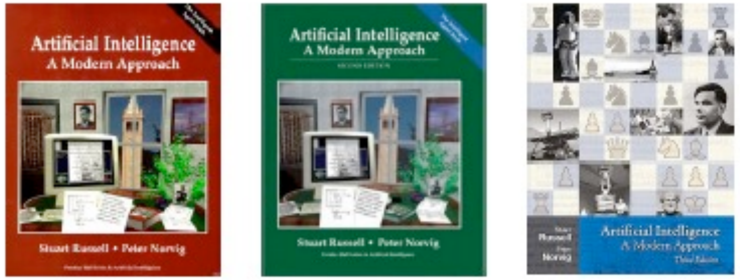




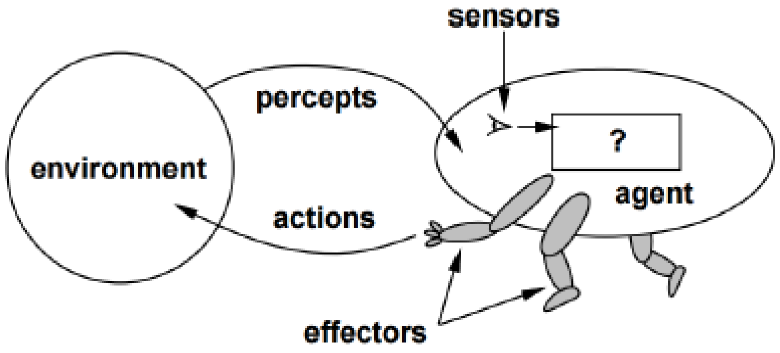




Artificial Intelligence



(1995) (2002) (2009)



-Special purpose task

*limited set of constrained

-General purpose task

*not well-defined constrained

What tasks that AI can do?

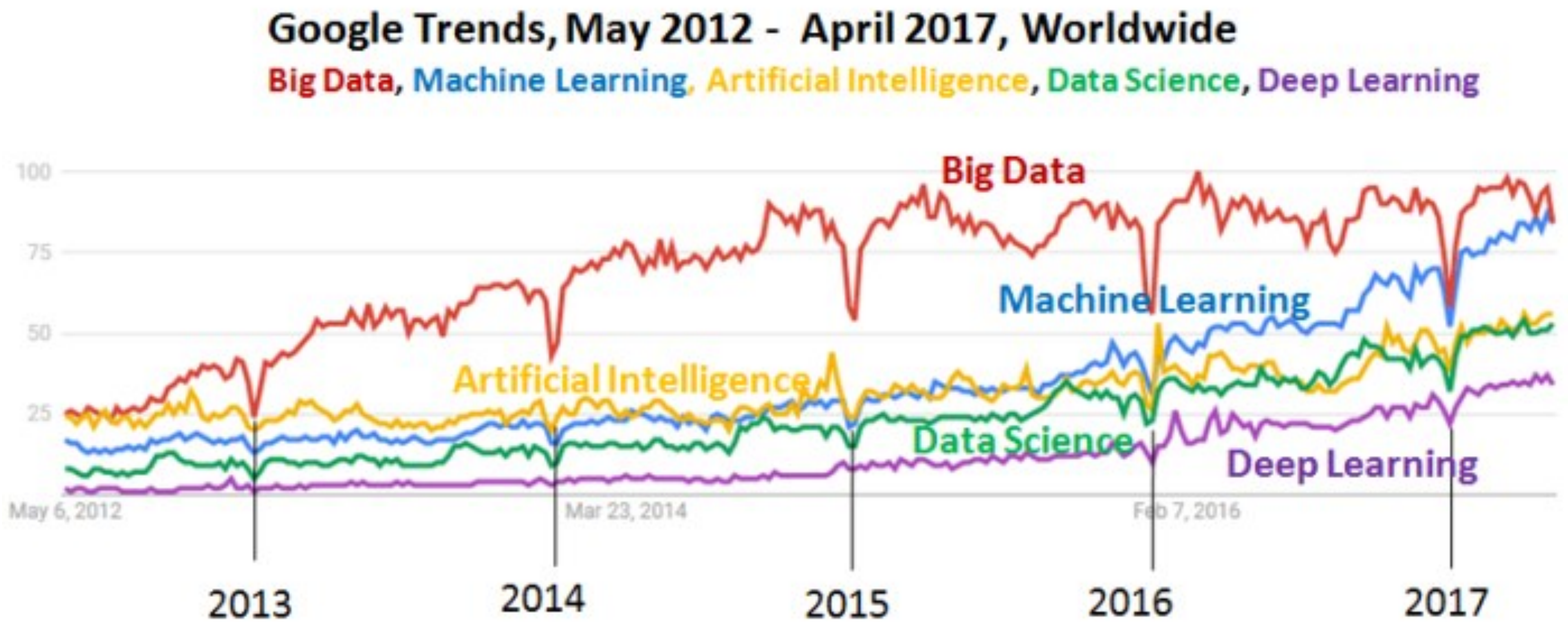
-Games

-Expert tasks

-Everyday tasks

Google Trends

1 Machine Learning is rising like a true champion, 2) Deep Learning is new and rising fast.



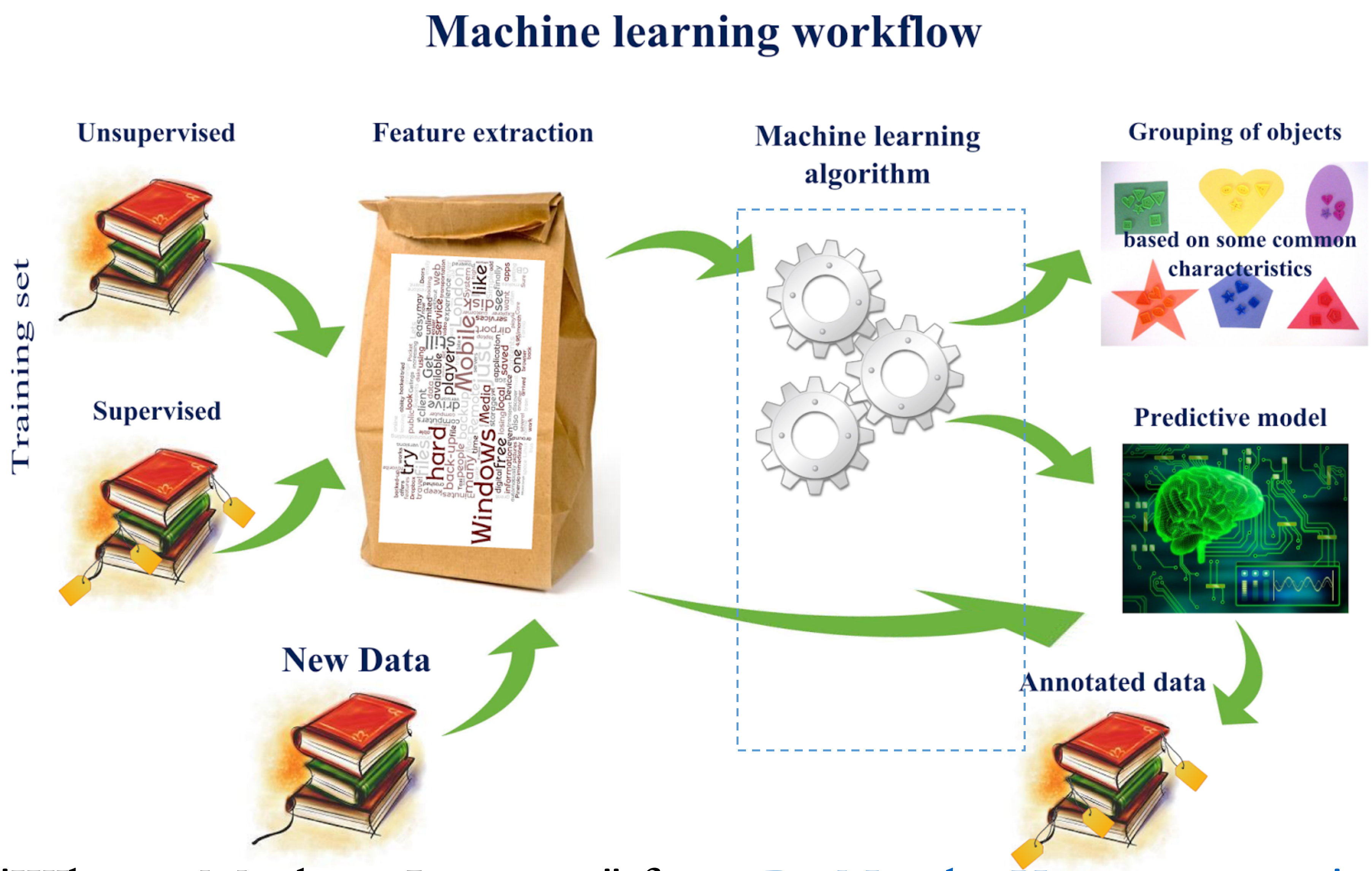
https://www.kdnuggets.com/2017/05/machine-learning-overtaking-big-data.html

Rising from Pattern Recognition

- Getting a computer program to do something “smart” like recognize the character "3”
- Optical Character Recognition grew out of this community (OCR)
- It is fair to call “Pattern Recognition” as the “Smart" Signal Processing of the 70s, 80s, and early 90s.
- Decision trees, heuristics, quadratic discriminant analysis, etc. all came out of this era. Pattern Recognition become something CS people did like, and not EE folks. – e.g. template matching

Machine learning

- A powerful way to build pattern recognition algorithms is to replace an expert (who probably knows way too much about pixels) with data (which can be mined from cheap laborers).
- Collect lots of face images and non-face images, choose an algorithm, and wait for the computations to finish.



"What is Machine Learning" from Dr Natalia Konstantinova's Blog. The most important part of this diagram are the "Gears" which suggests that crunching/working/computing is an important step in the ML pipeline

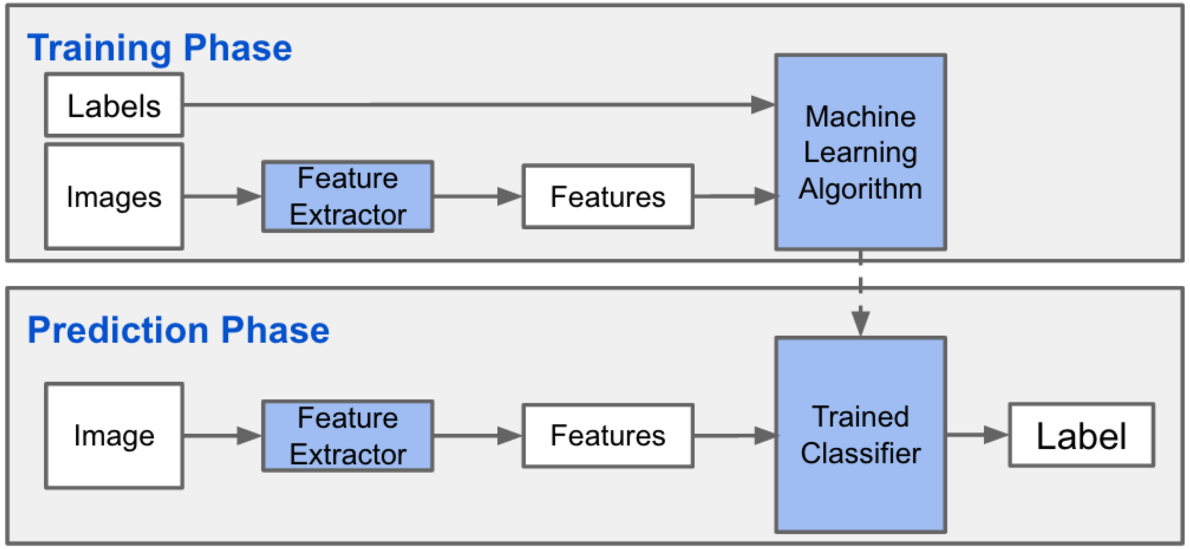
Application Areas

- Researchers apply Machine Learning to Robotics (reinforcement learning, manipulation, motion planning, grasping, vision), to genome data, to remote sensing data as well as to predict financial markets.
- One of the best application areas for machine learning for many years was computer vision.

Until recently neural networks were all but shunned by the AI research community. The problem is the most basic computation of neural networks were very computationally intensive, it just wasn't a practical approach.
But it wasn't until GPUs were deployed in the effort that the promise was realized.

"Andriew Ng 's breakthrough was to take these neural networks, and essentially make them huge, increase the layers and the neurons, and then run massive amounts of data through the system to train it. In Ng's case it was images from 10 million YouTube videos. Ng put the "deep" in deep learning, which describes all the layers in these neural networks."

--Google's AlphaGo



Machine Learning Phases

Traditional Machine Learning Flow

Deep Learning Flow

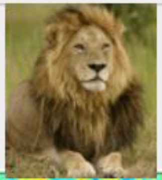
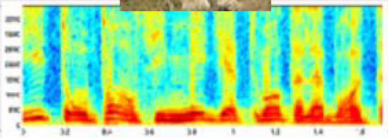
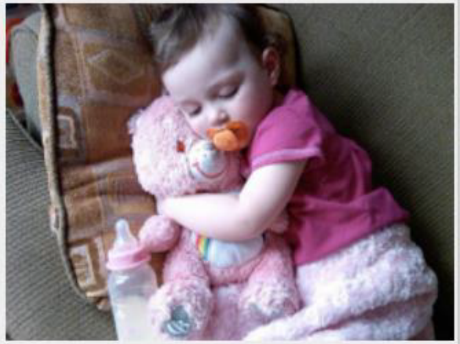
"What is Machine Learning" from Dr Natalia Konstantinova's Blog. The most important part of this diagram are the "Gears" which suggests that crunching/working/computing is an important step in the ML pipeline.

Deep Learning

- Deep Learning has become the most popular approach to developing Artificial Intelligence (AI) –machines that perceive and understand the world
- The very first focus is currently on specific perceptual tasks, and there are many successes.
- GPUs have become the parallel platform for for deep learning in research and production.

Jeff Dean, GTC 2015

Examples

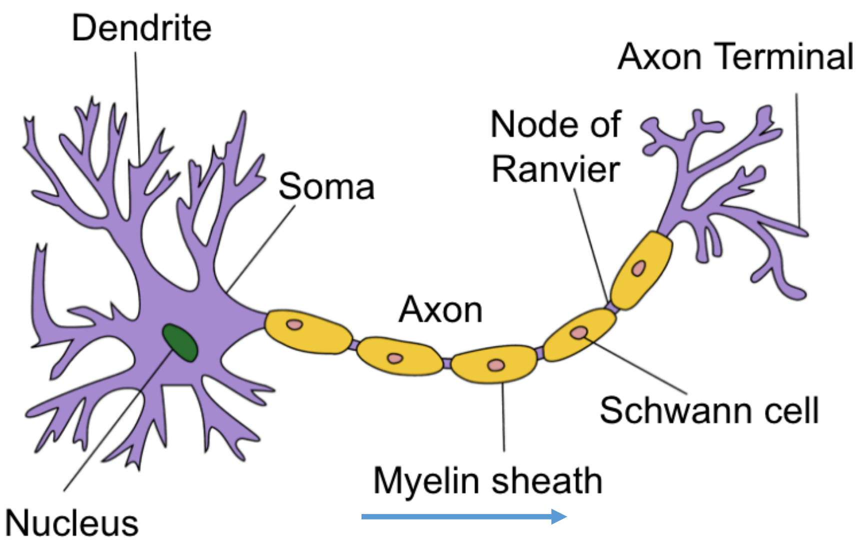
Input	Output
Pixels: 	"lion"
Audio: 	"see at tuhl res taur aun ts"
<query, doc>	P(click on doc)
"Hello, how are you?"	"Bonjour, comment allez-vous?"
Pixels: 	"A close up of a small child holding a stuffed animal"

Some Review of ANNs (Artificial Neural Networks)

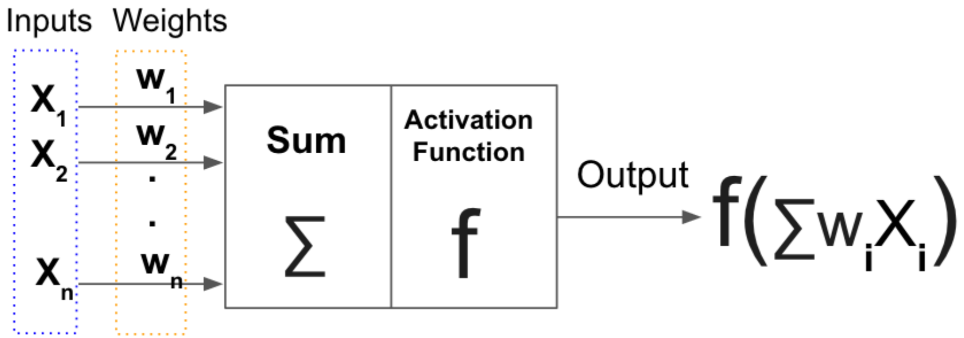
Note: The tutorial slides are from many places.

- It is inspired by biological neural networks: Biological Neurons
- It processes and transmit information to other neurons by emitting electrical signals. Each neuron receives input signals from its dendrites and produces output signals along its axon. The axon branches out and connects via synapses to dendrites of other neurons.
- Artificial neurons are originated by biological neurons, and formulate the model in a computational form
    - has a finite number of inputs with weights associated to them,
    - an activation function (also called transfer function)
    - the output of the neuron is the result of the activation function applied to the weighted sum of inputs.
    - Artificial neurons are connected with each others to form artificial neural networks.

• y=



Structure of a typical neuron
(source: Wikipedia)



Structure of artificial neuron

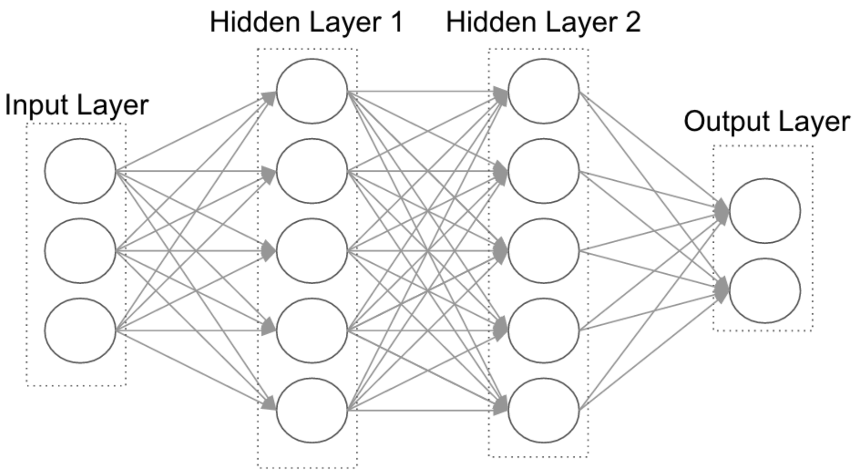
e.g. $f(x) = Max(0,x)$

Biological Neurons are the core components of the human brain. A neuron consists of a cell body, dendrites, and an axon.

Human brain contains about 10,000 computational power than the computer's

Feedforward Neural Networks

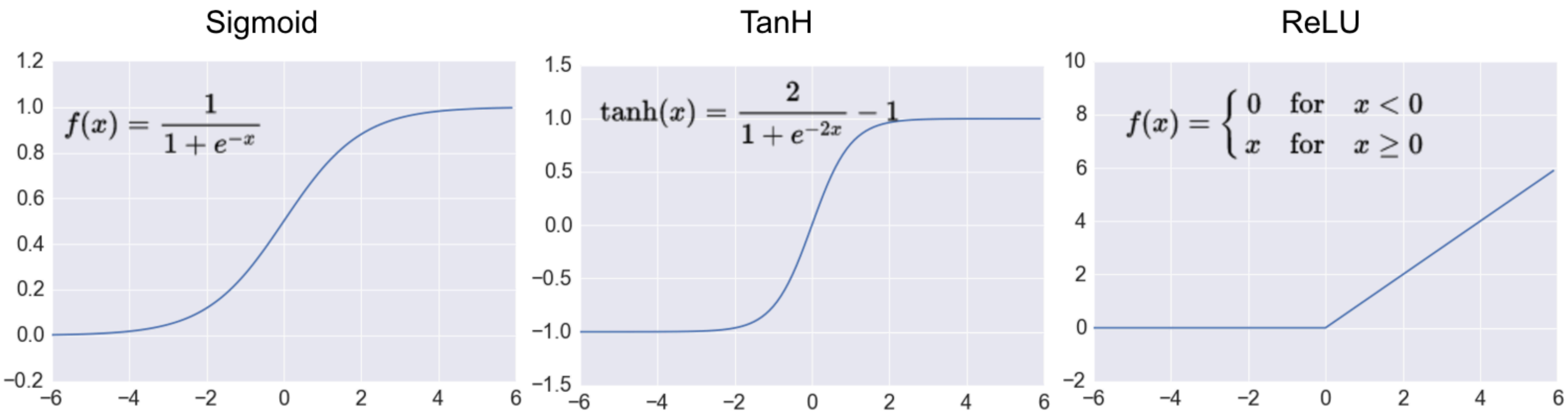
- These networks have 3 types of layers: Input layer, hidden layer and output layer. In these networks, data moves from the input layer through the hidden nodes (if any) and to the output nodes.
- "Fully-connected" means that each node is connected to all the nodes in the next layer.



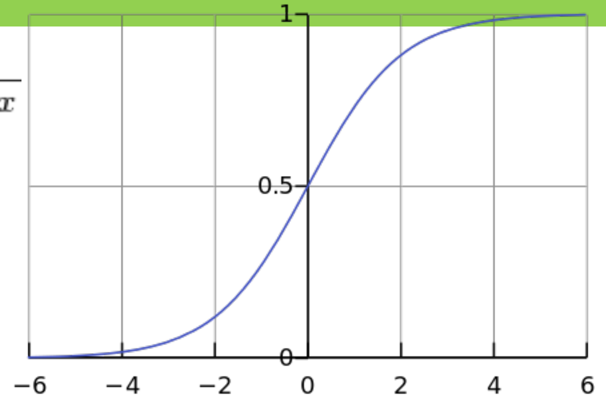
Feedforward neural network with 2 hidden layers

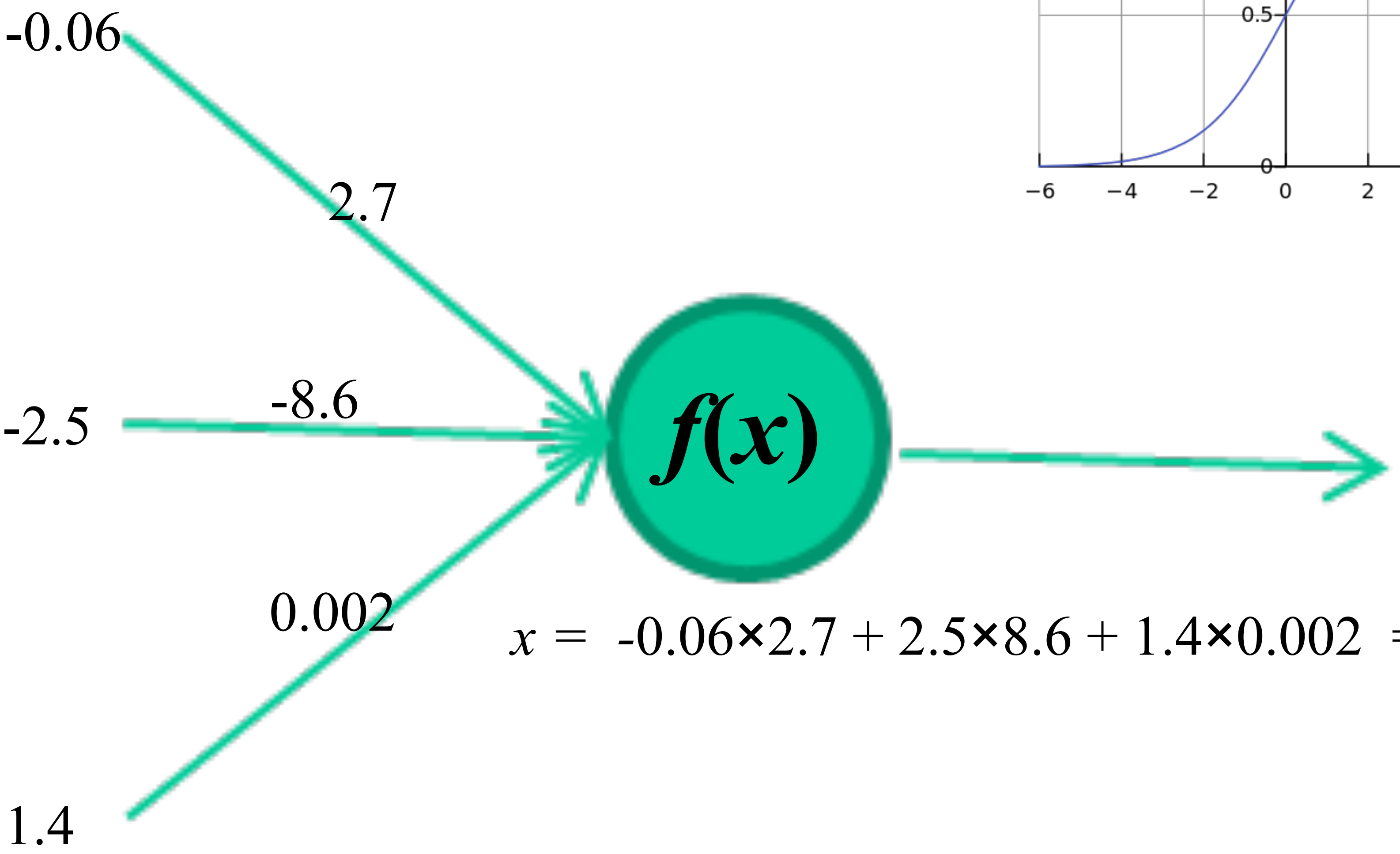
Activation Functions

- Activation functions transform the weighted sum of inputs that goes into the artificial neurons.
- The most popular activation functions are Sigmoid, Tanh and ReLU (rectified linear unit). ReLU is the most popular activation function in deep neural networks.



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





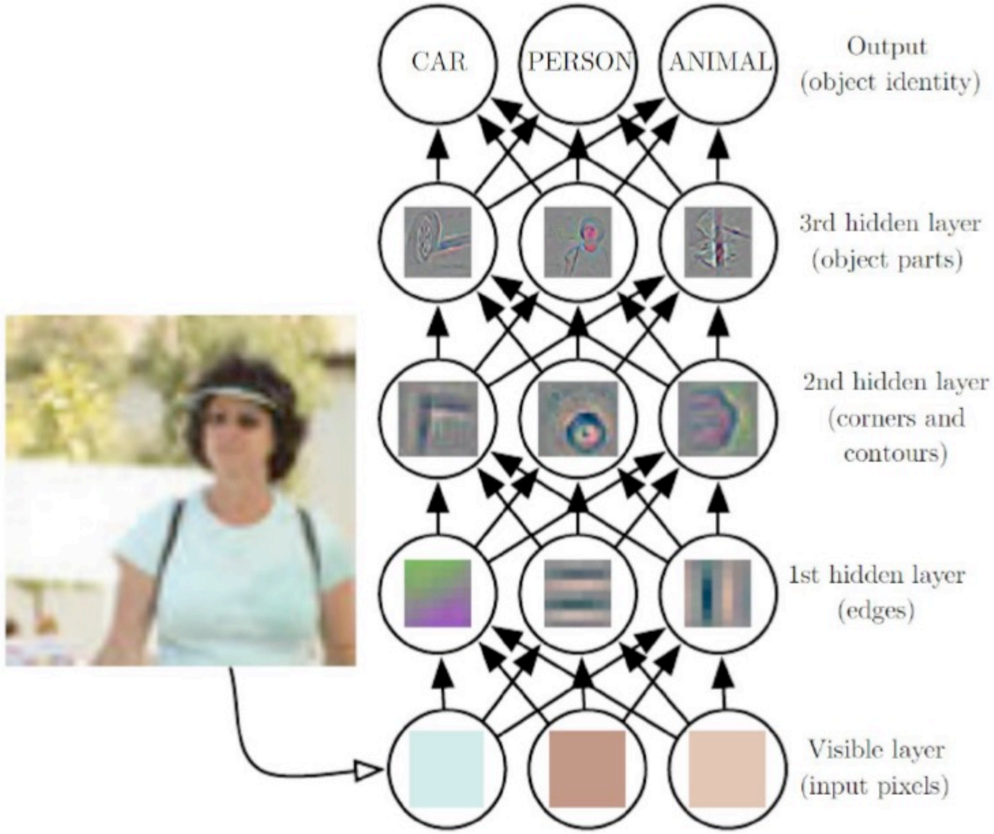
$$x = -0.06\times2.7 + 2.5\times8.6 + 1.4\times0.002 = 21.34$$

Training Artificial Neural Networks

- The goal of the training phase is to learn the network's weights.
- 2 elements to train an artificial neural network:
  - Training data: In the case of image classification, the training data is composed of images and the corresponding labels.
  - Loss function: A function that measures the inaccuracy of predictions.
- We train the ANN using an algorithm called backpropagation together with gradient descent (or one of its derivatives).

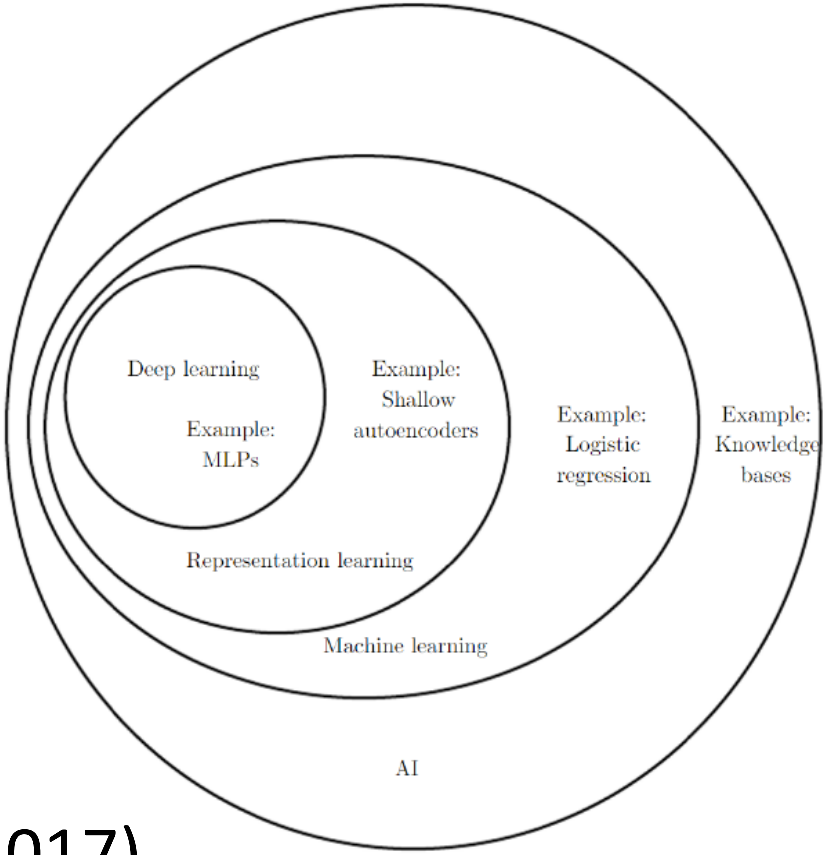
https://mattmazur.com/2015/03/17/a-step-by-step-backpropagation-example/

Deep learning is a representation learning

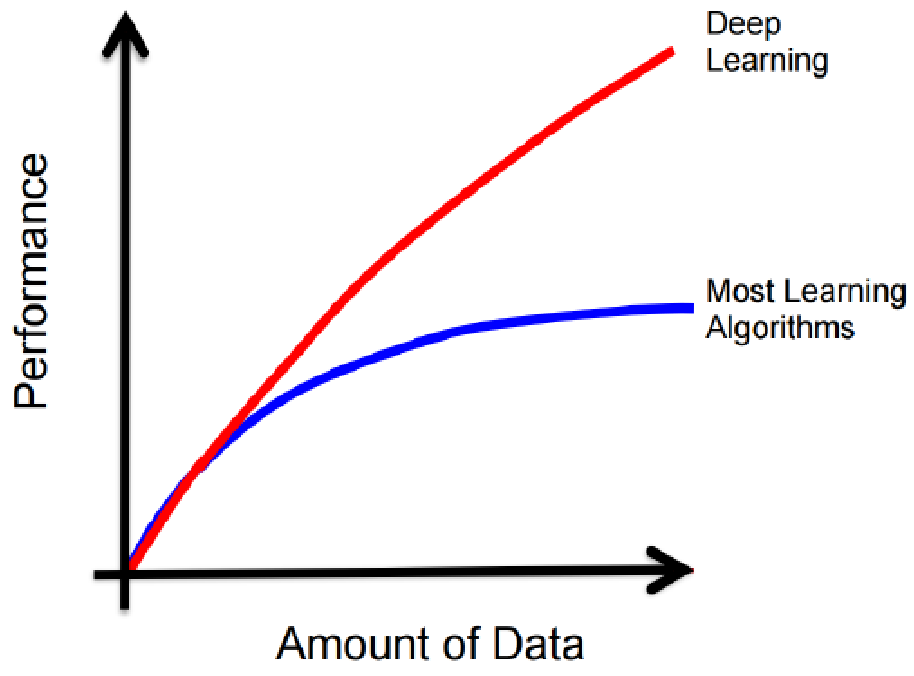


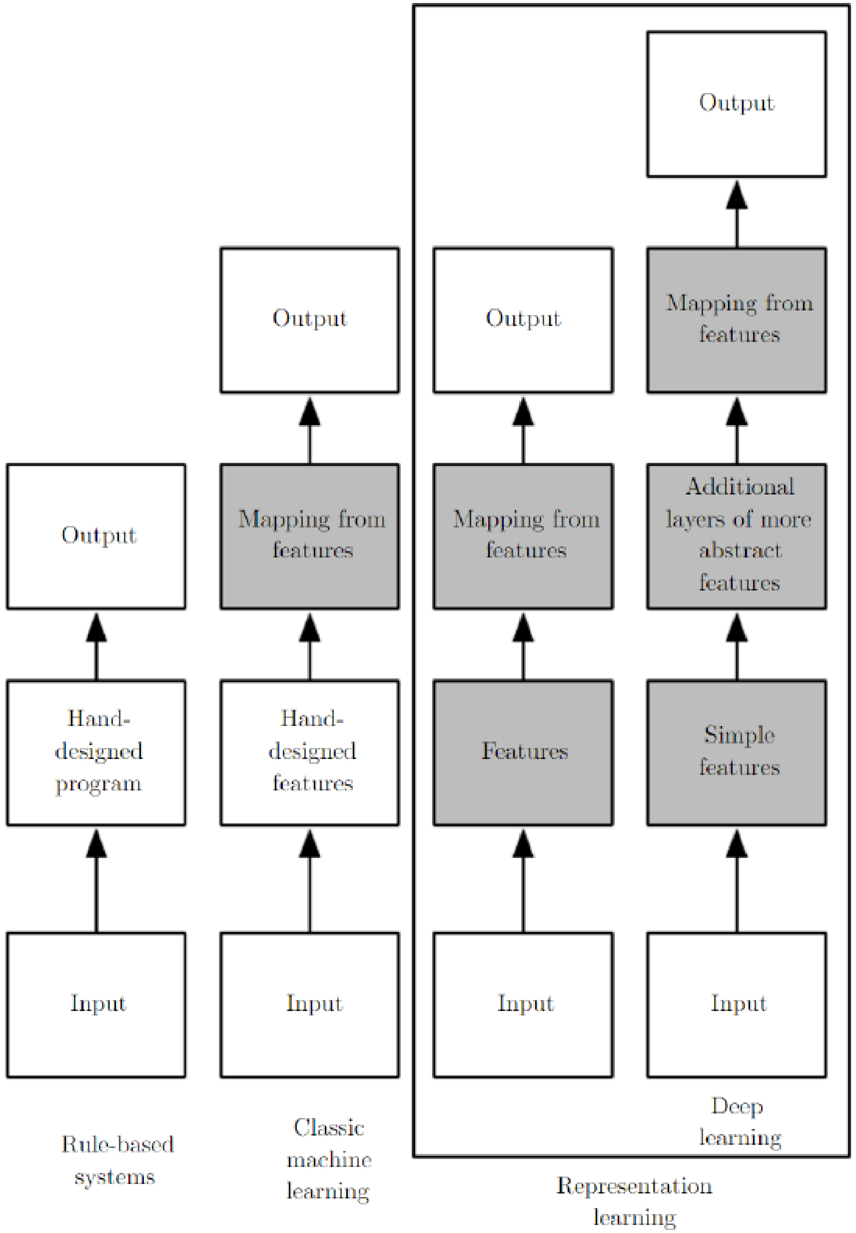
Goodfellow et al. "Deep learning." (2017).

Deep learning is a representation learning



Ian Good Fellow et.al. (2017)
Deep learning.

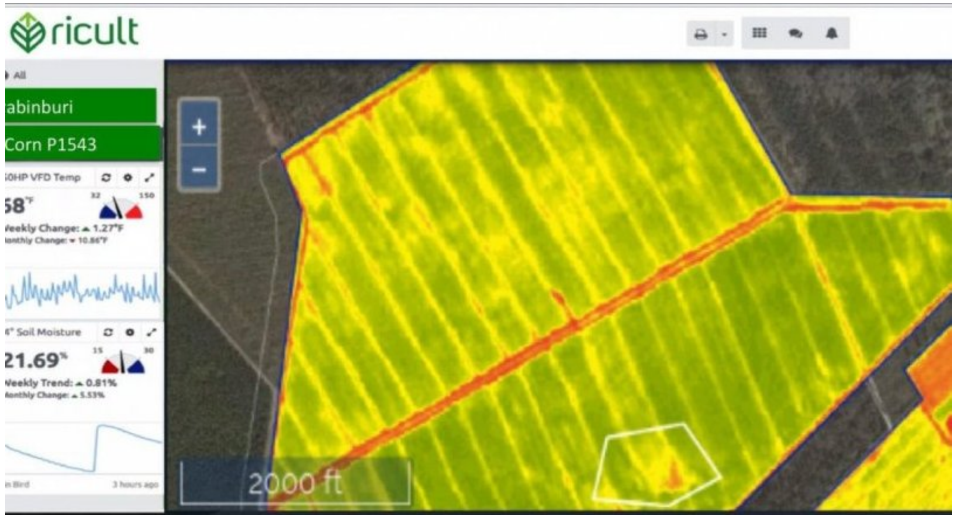




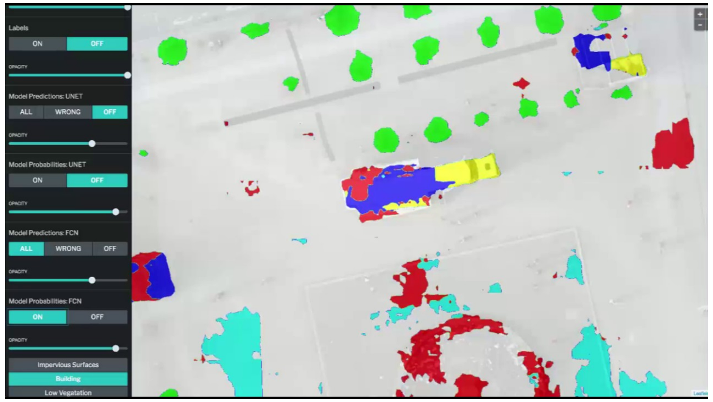
The more data…

Agriculture/ Land use

- Weed/Plant classification
- Productivity prediction
- Fruit inspection: NIR images
- Crop disease



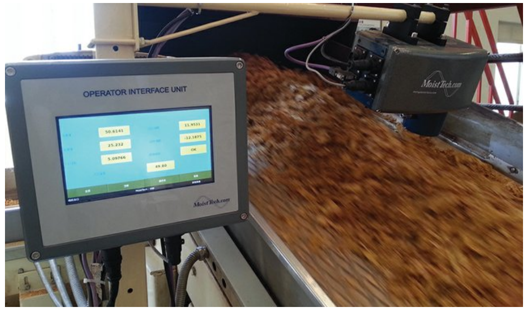
RGB Image NIR Image NDVI Image

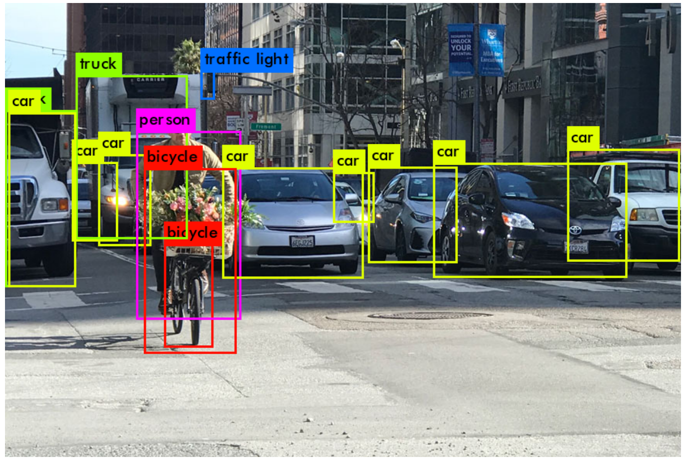


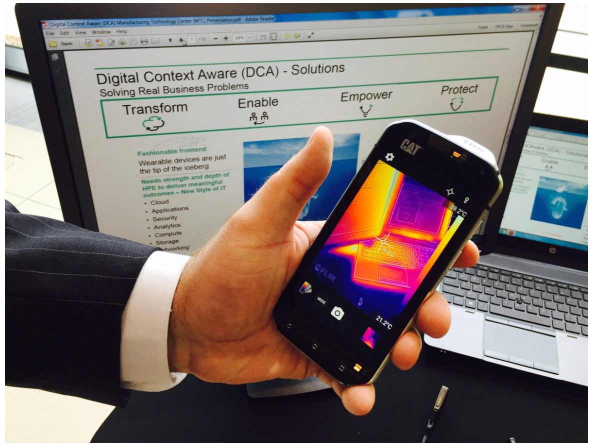
Manufacturing

Automatic detection

- Detecting overheating (Equipment)
- QC of Finished products
- Glass defect detection



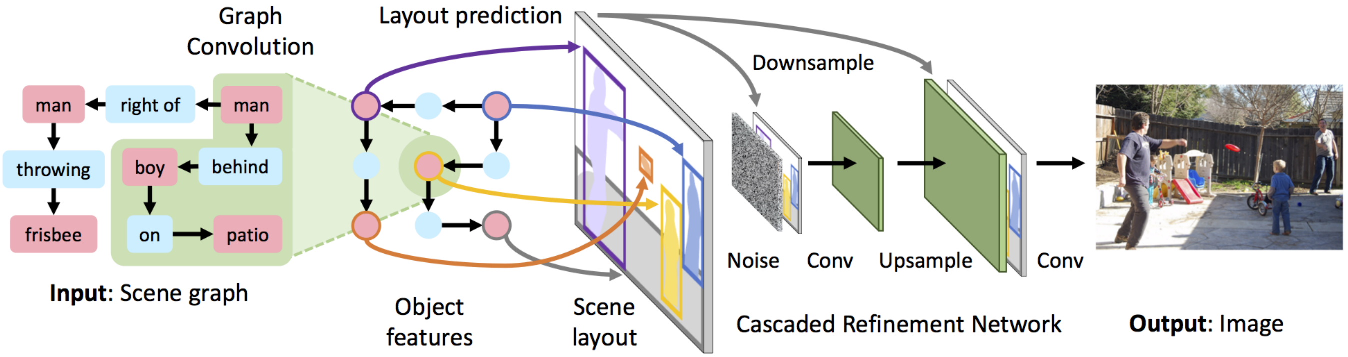




Video Classification/Semantics

Majorly use CNN: Describe event/ semantic

- Have temporal information: learning a global description of the video's temporal evolution is important.
- Two approaches:
  - Feature pooling
    - Process each frame using CNN and combine using pooling layers.
  - Long Short Term Memory (LSTM)
    - Use memory cells to store, modify, access internal states. Operate on each frame using CNN but integrate information overtime.



Application areas

- Data security: Predict malware, suspicious attack.
- Personal security: security screenings at airports, stadium.
- Financial trading
- Healthcare: compute assisted diagnosis
- Marketing personalization
- Fraud detection: suspicious transfer transaction
- Recommendations: TV program, Song
- NLP
- Self driving cars

https://www.forbes.com/sites/bernardmarr/2016/09/30/what-are-the-top-10-use-cases-for-machine-learning-and-ai/#74ff4c2f94c9

Face recognition

- Face recognition applications in military

- Integrate deep learning technology to industries /real-world applications.

Criminal

UK, the world's most surveilled state, begins using automated face recognition to catch criminals

Drone

video drones with facial recognition target persons

Retailers Increasingly Using Facial Recognition To Spot Thieves

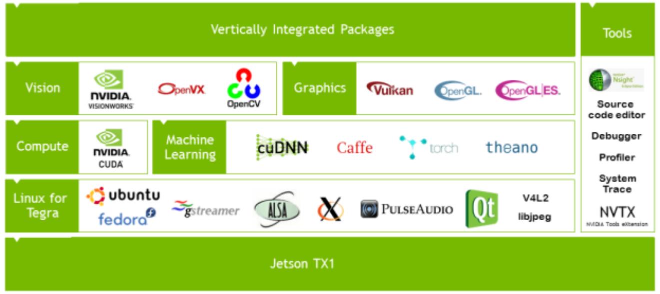
Literature Review

- Deep Face, FaceNet, Parkhi et.al (2015), Sarkar et.al. (2016)....
- Open source githubs

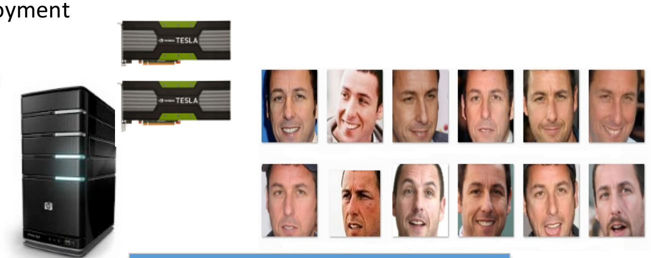
Methodology

http://www.iis.ee.ic.ac.uk/cxiong/database.html

JETSON SDK

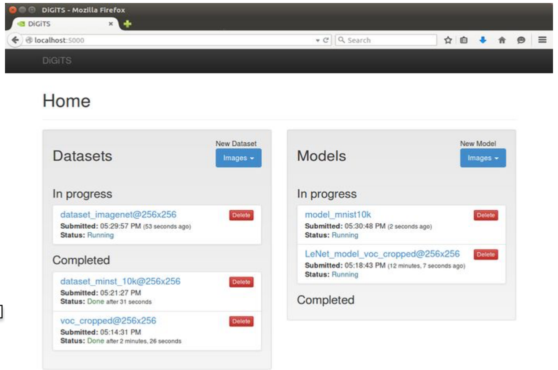


Model Deployment



Linux Centos 7.0

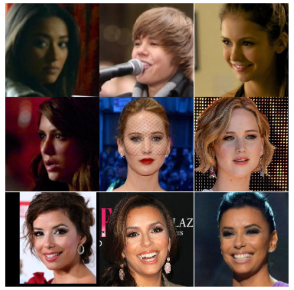
Deployment



Explore public data sets

- LFW
- FaceScrub
- CASIA
- MegaFace
- MS-Celeb
- WIDER

(a) LFW



(b) CASIA

(c) FaceScrub

(d) MegaFace

Available pretrained nets

- AlexNet
- CaffeNet
- GoogleNet
- SqueezeNet
- ResNet
- VGG

Etc

Find more at model zoo.

Type	AlexNet	GoogLe	Caffe	SqueezeNet	ResNet10	VGG16	RCNN
Conv	5	59	5	26	12	13	5
InnerProd	3	5	3	-	1	3	3
Pooling	3	16	3	3	2	5	3
ReLU	7	61	7	26	9	15	7
LRN	2	2	2	-	-	-	2
Dropout	2	3	2	1	-	2	2
Misc.					10 BN 10 Scale 4 Eltwise		

Network comparison

Type	AlexNet	GoogLe	Caffe	SqueezeNet	ResNet10	VGG16	RCNN
Conv	5	59	5	26	12	13	5
InnerProd	3	5	3	-	1	3	3
Pooling	3	16	3	3	2	5	3
ReLU	7	61	7	26	9	15	7
LRN	2	2	2	-	-	-	2
Dropout	2	3	2	1	-	2	2
Misc.					10 BN 10 Scale 4 Eltwise		

On the Performance Issues of Various Convolutional Neural Networks (CNN) for Face Recognition Tasks

Chantana Chantrapornchai
Dept. of Computer Engineering
Kasetsart University
Bangkok, Thailand
fengcnc@ku.ac.th

Samrid Duangkaew
Dept. of Computer Engineering
Kasetsart University
Bangkok, Thailand
samrid.d@ku.th

***Abstract*—In this paper, we are interested in performance issues of various convolutional neural networks (CNN) for face recognition tasks. Various well-known CNN models exist and most of them are well trained for a large data set such as ImageNet. These models are easier for fine tuning with the given pretrained weights. However, when focusing on a different task like face recognition, it is questionable how well these pretrained nets can be utilized, how well the nets can perform, whether or not they are easy to fine tune or retrain, etc? These usually are the starting issues when one would like to employ a CNN to solve his problem in practice. In this paper, we consider the training time, accuracy, and fine tuning different layers of existing nets. The performance of existing nets: AlexNet, CaffeNet, GoogLeNet, SqueezeNet, VGG16, ResNet10, are compared and the usability is discussed, against a face recognition task for four common face data sets: LFW, CASIA, FaceScrub, MegaFace.**

***Index Terms*—CNN, Deep learning, Fine Tuning, Face Recognition.**

I. Introduction

Deep learning has been one of the popular techniques there are lots of existing networks to explore and finding the most suitable one requires extensive experiments. Since the goal of using deep learning approach is to extract the important features of faces, the network that yields the features can represent the faces would be the good choices. We show the experimental approach to fine tuning existing network for this task. The discussions about the network complexity, the training time, preprocessing, as well as the accuracy are presented.

II. Related Work

Many existing works demonstrate the use of deep learning in face recognition. Some of the works deploy deep network for face detection. Some work deploys faster RCNN to face detection [6]. Note that RCNN is a network that is aimed to detect the bounding box of the objects. The improved versions, fast RCNN and faster RCNN, reorganizes the computation to make the object detection fast [7], [8]. There are two steps

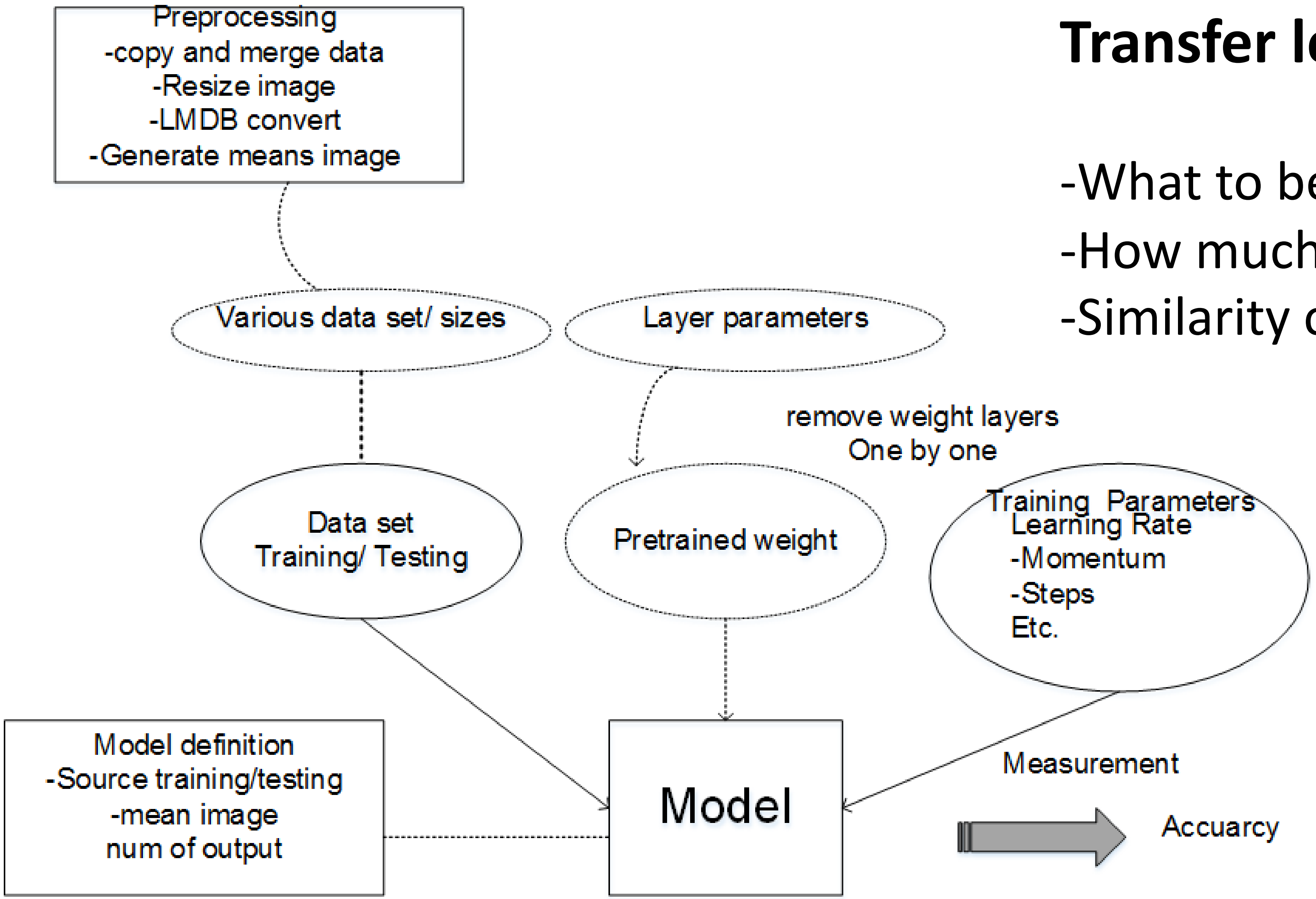
http://miticon.org/

Transfer learning

-What to be transferred?

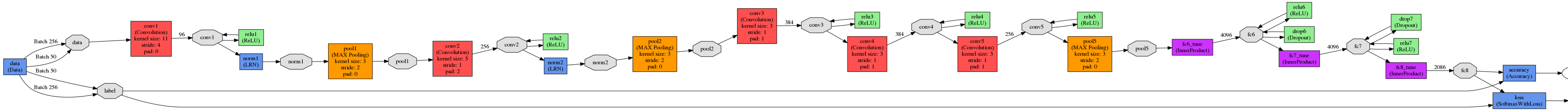
-How much?

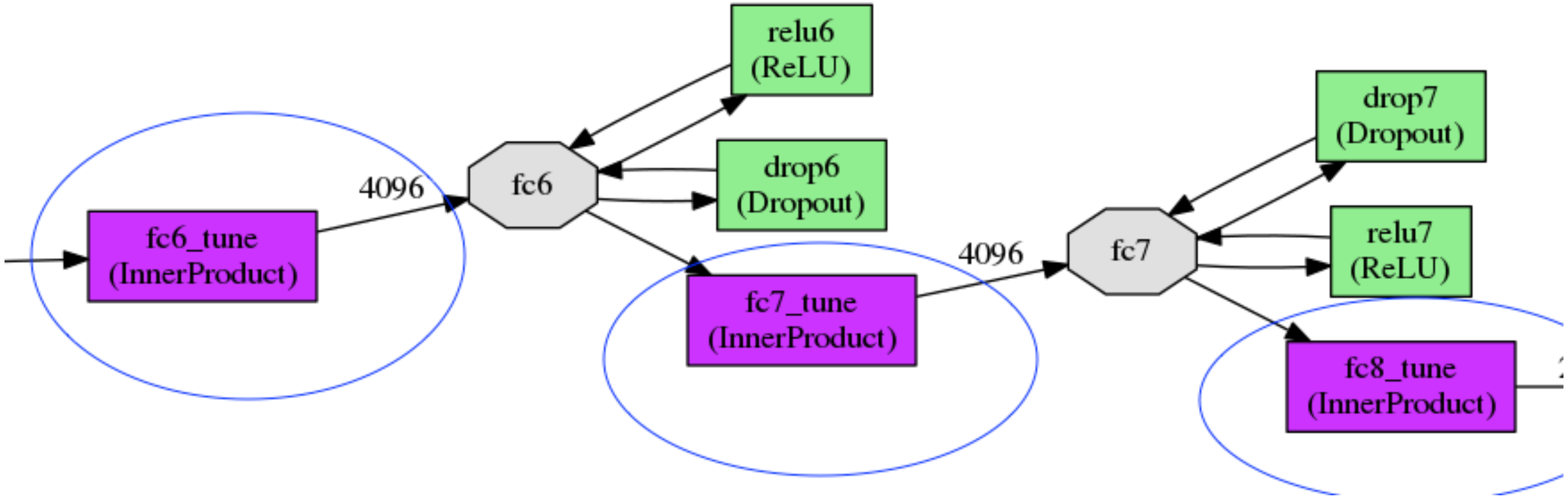
-Similarity of data set?



Fine tune network

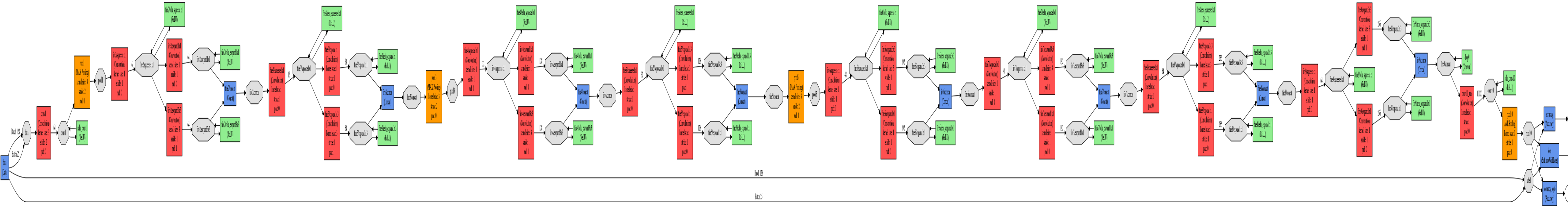
- AlexNet

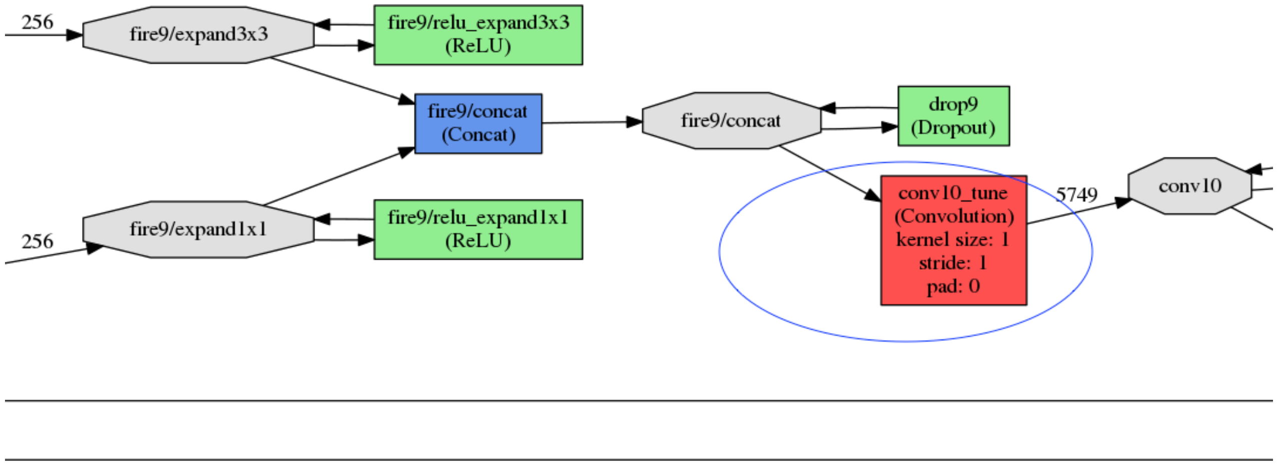




Fine tune network

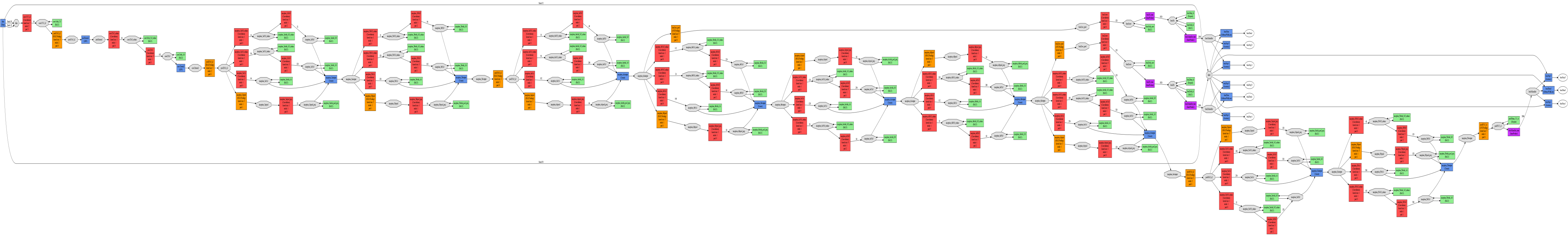
- SqueezeNet

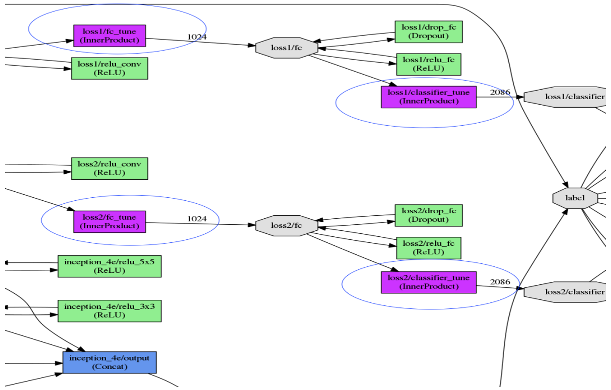




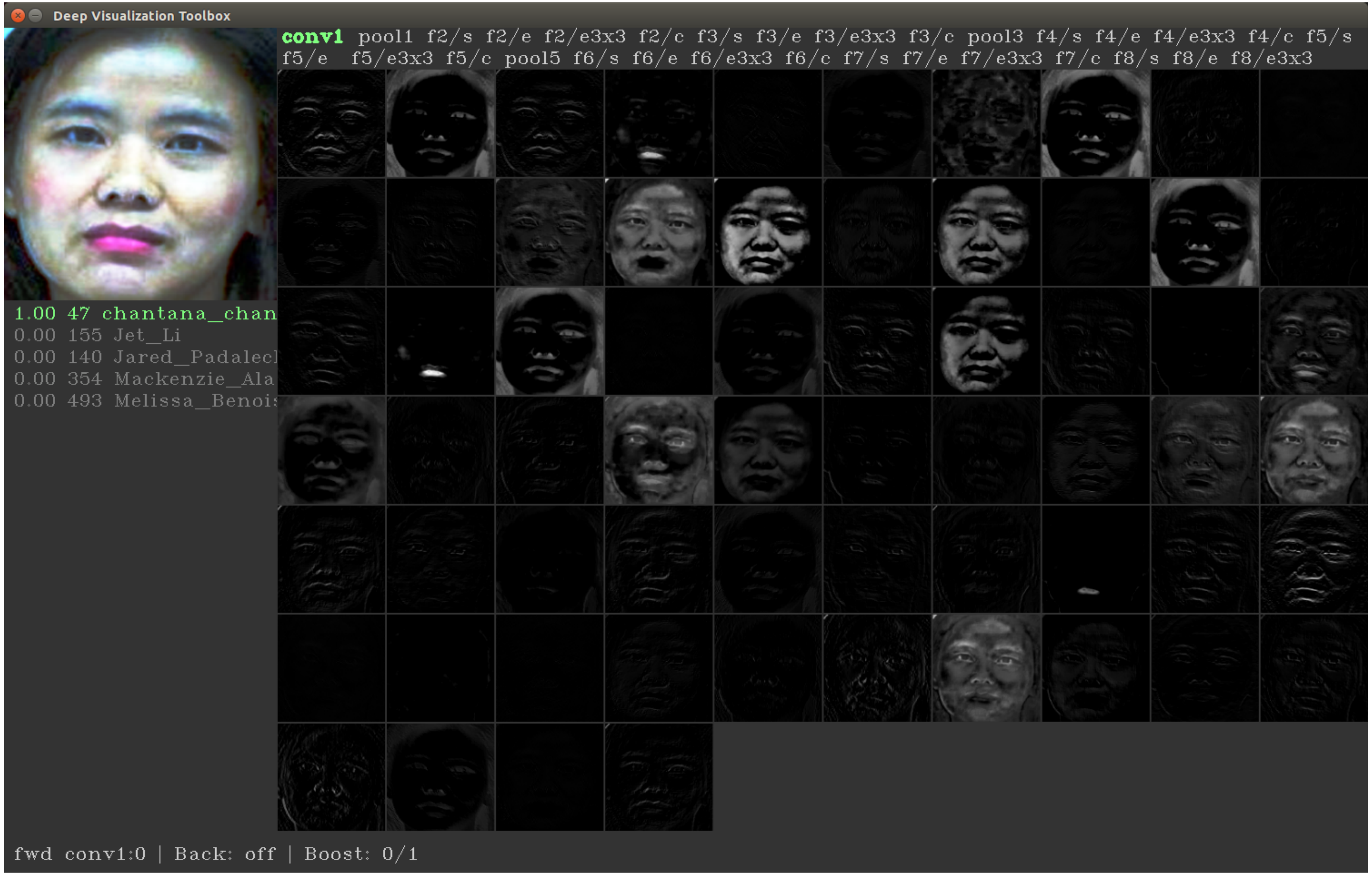
Fine tune network

- GoogLeNet



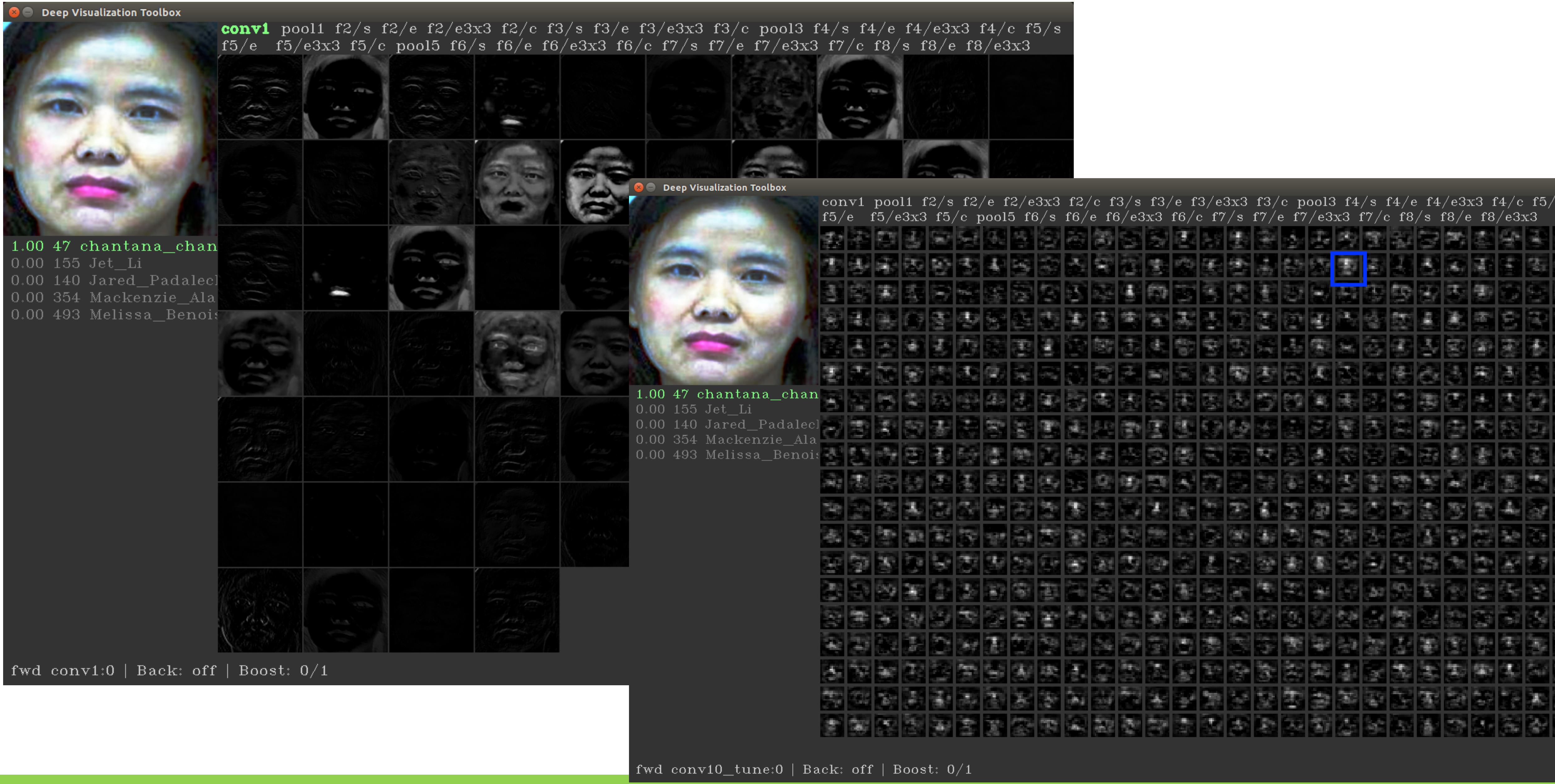


Visualize weights



Deep visualization toolbox

Visualize weights



Comparison to other models

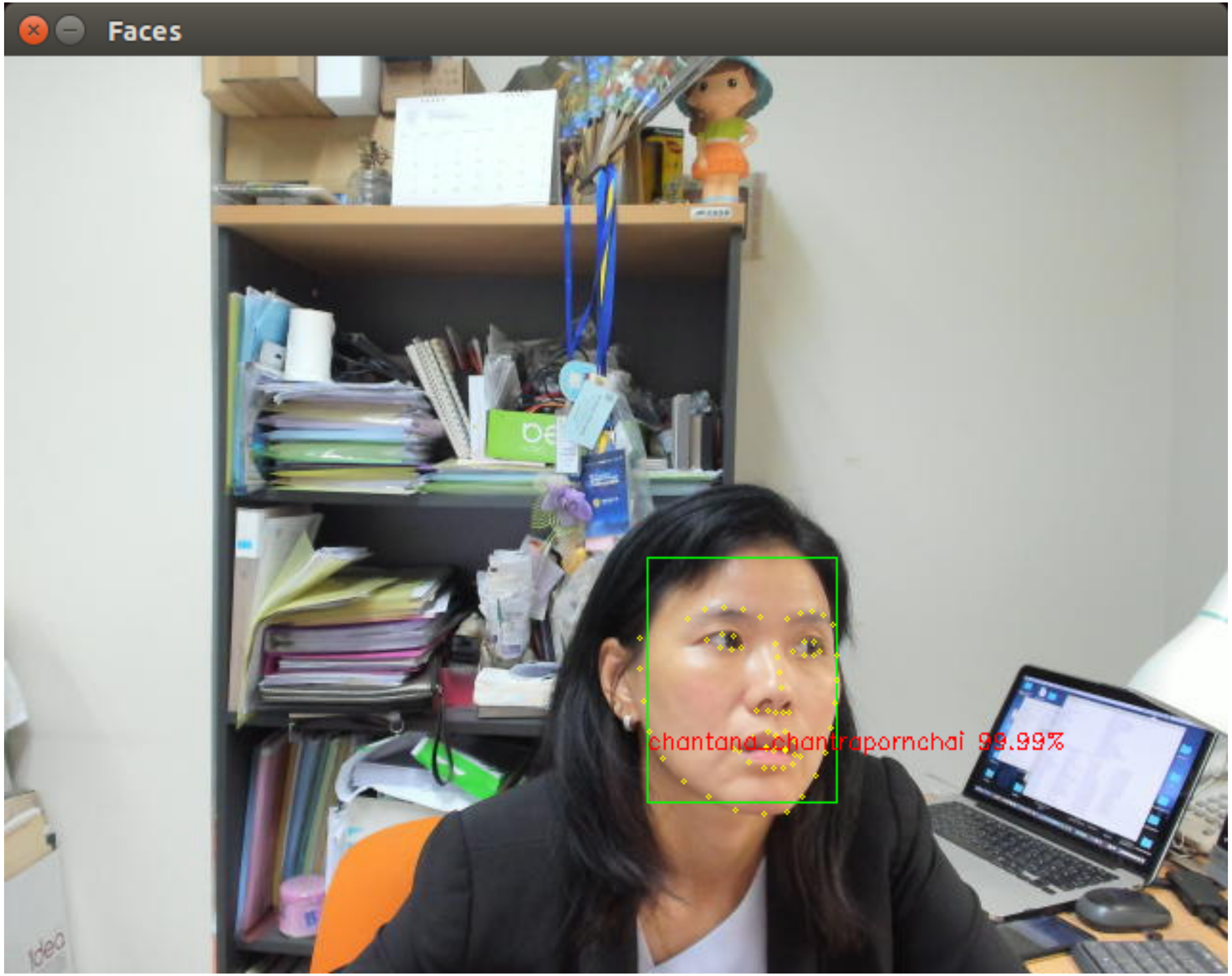
- Openface (CMU) (2016)

https://cmusatyalab.github.io/openface/

- https://github.com/ageitgey/face_recognition/blob/master/examples/face_recognition_knn.py
- FaceNet (Triplet loss)

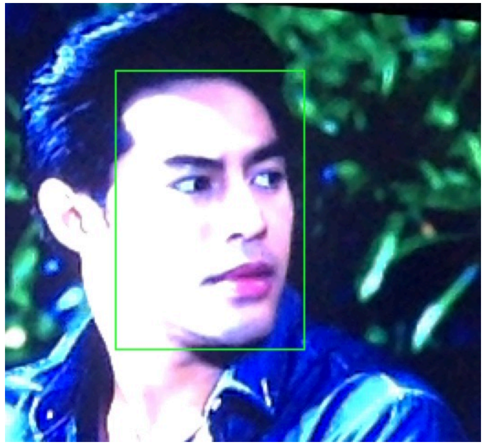
https://github.com/davidsandberg/facenet/wiki/Triplet-loss-training

- Faster RCNN with face recognition (2016)
  https://github.com/playerkk/face-py-faster-rcnn


Faces
chantana.chantrapornchai 99.99%

Software Process

Face detection



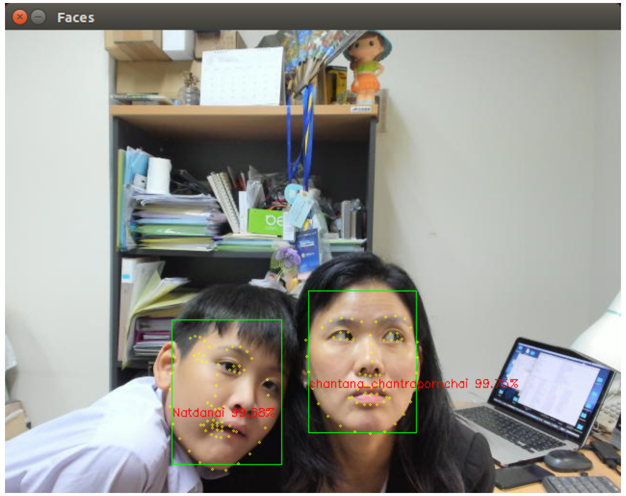
Preprocess
-alignment
-pose adjustment
-augmentation

Recog. Model?

Image means

Recognition

Classes



MTCNN
https://github.com/DuinoDu/mtcnn

To query database

Preprocessing Done

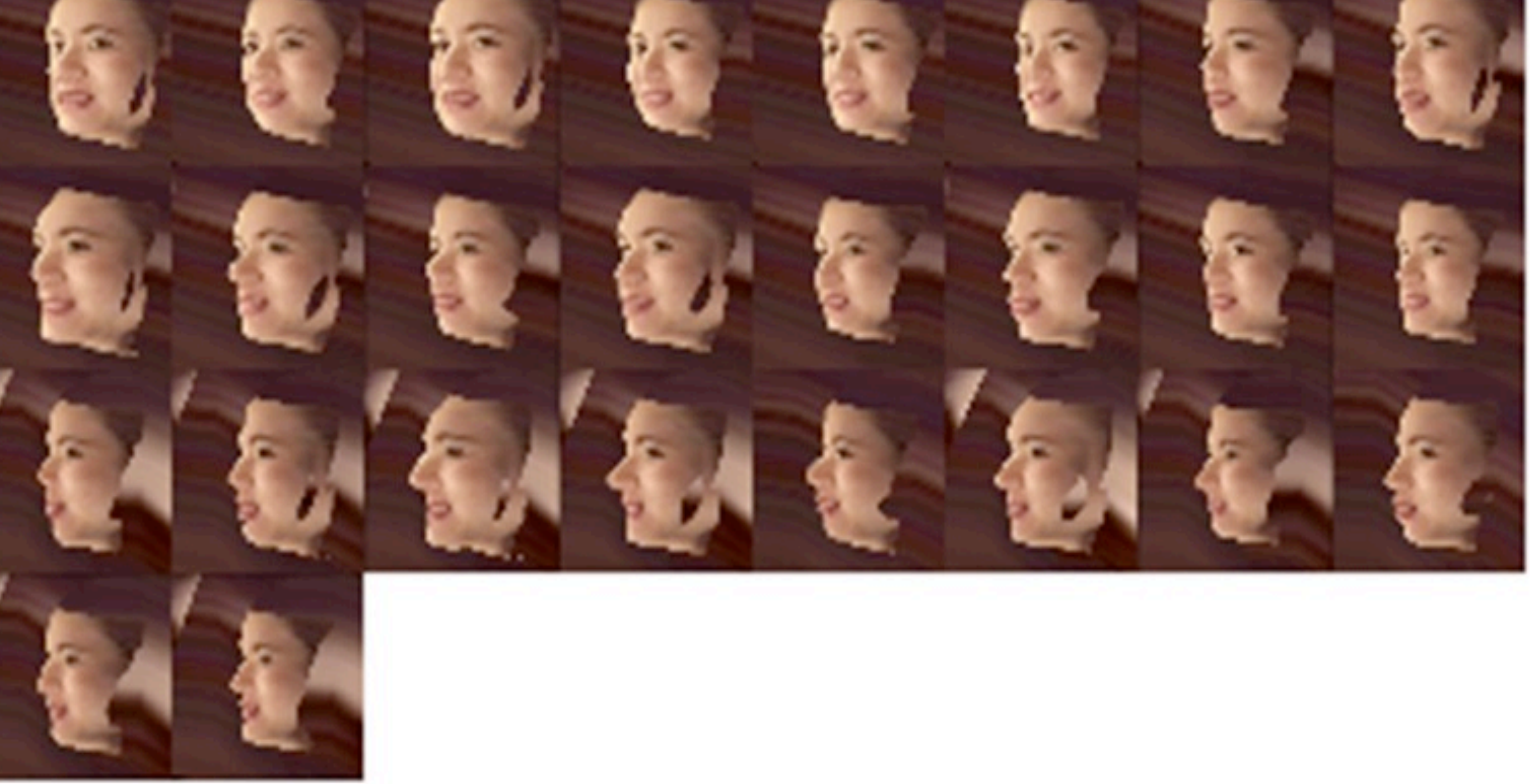
- Data augmentation -- https://github.com/aleju/imgaug

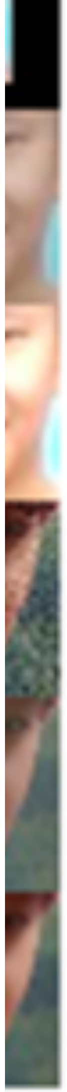
http://www.openu.ac.il/home/hassner/projects/augmented_faces/Masietal2016really.pdf

- Face Alignment (dlib) –Openface
- Pose alignment -- http://cvlab.cse.msu.edu/project-pifa.html

https://www.cv-foundation.org/openaccess/content_iccv_2015/papers/Jourabloo_Pose-Invariant_3D_Face_ICCV_2015_paper.pdf

- Compute image means
- Image resizing





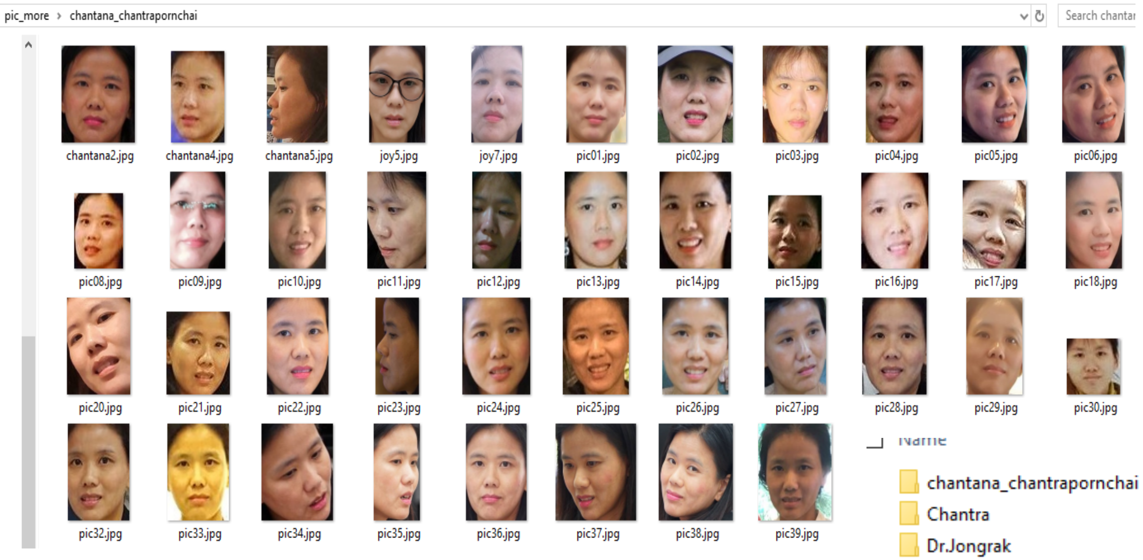
https://github.com/iacopomasi/face_specific_augm

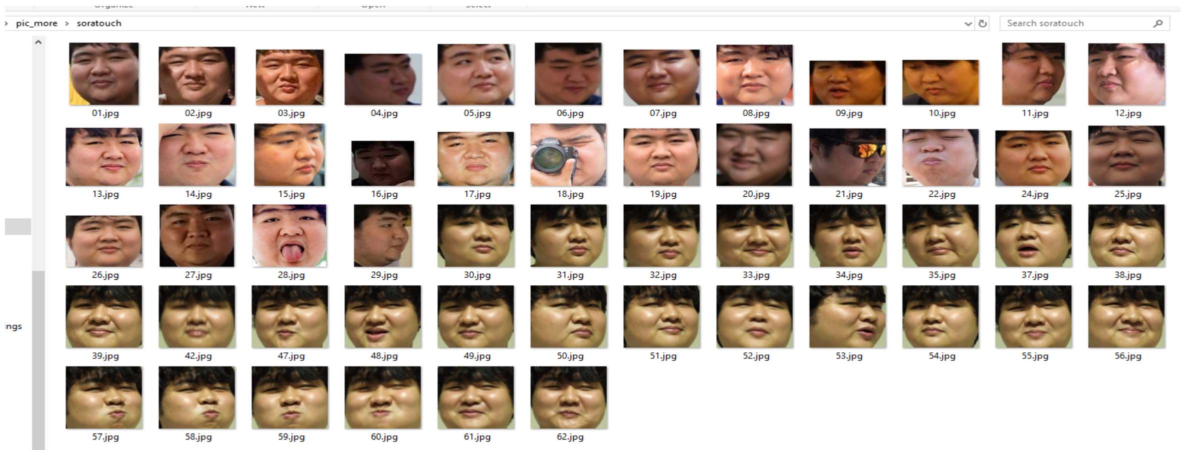
https://github.com/aleju/imgaug

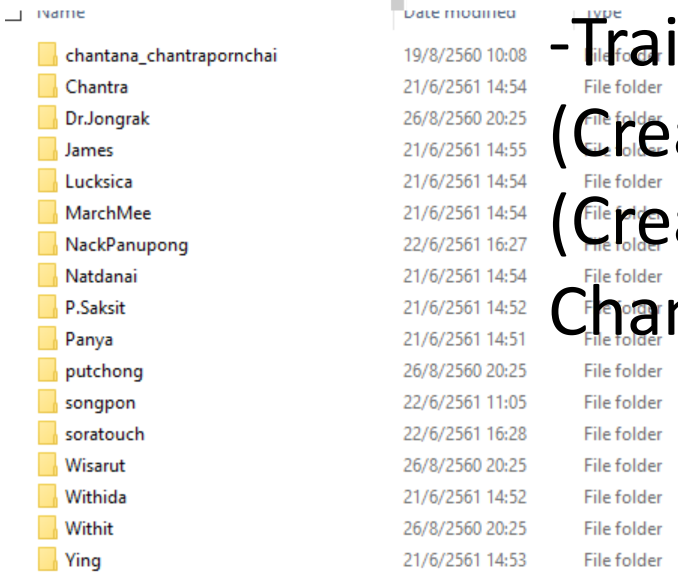
Transfer learning parameters

- Number of classes
- Number of layers
- Layer parameters: filter size, pretrained weight, module, batch size, number of filters, etc.

Adding new classes/data images







-Training

(Create lmdb for training/testing)

(Create mean file)

Change number pf classes

-Deployment

Adding prob layer

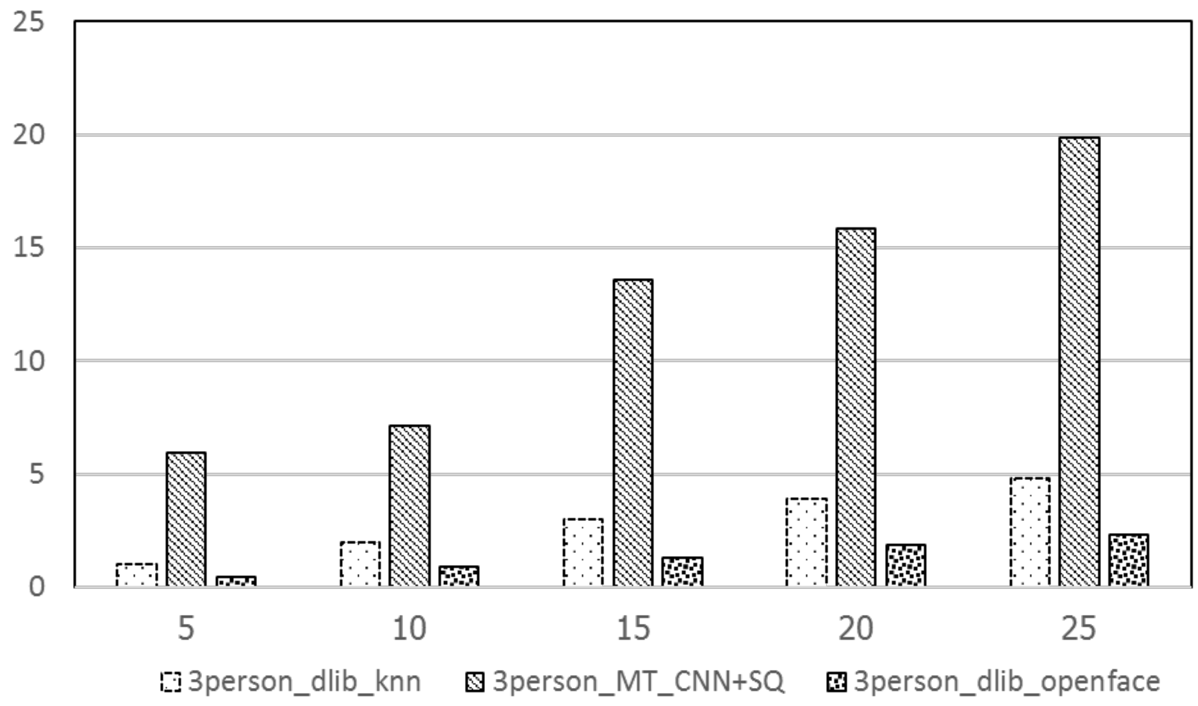
Remove train/test data s

Testing choices Examples

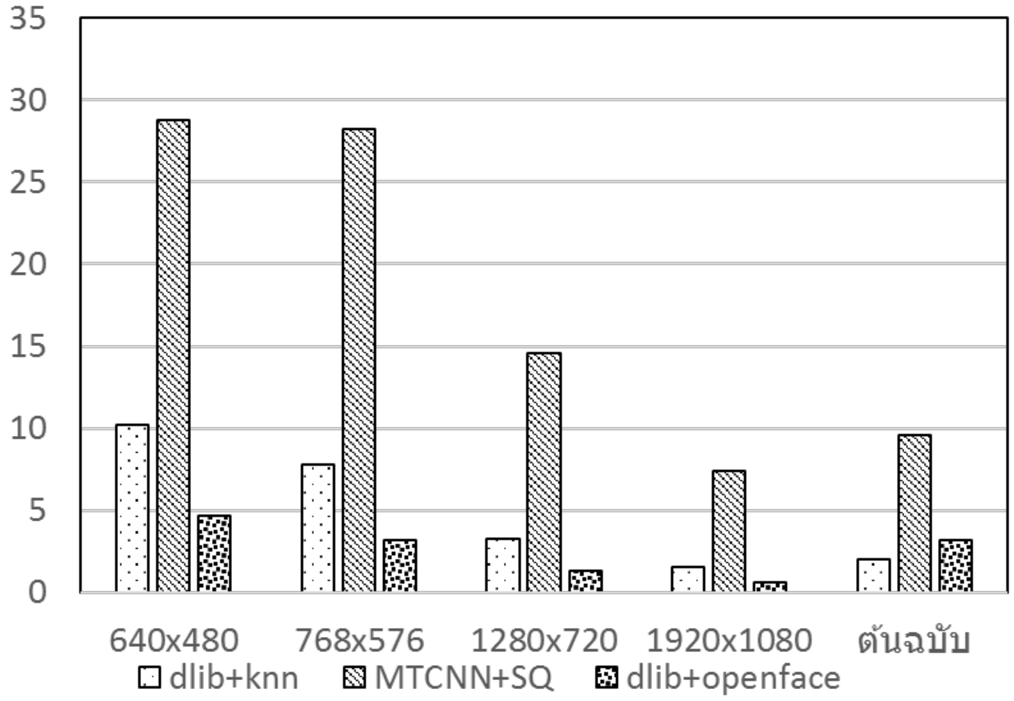
- MTCNN+SqueezeNet
- OpenCV + SqueezeNet
- Dlib+KNN
- OpenCV+OpenFace

Frame rate comparison

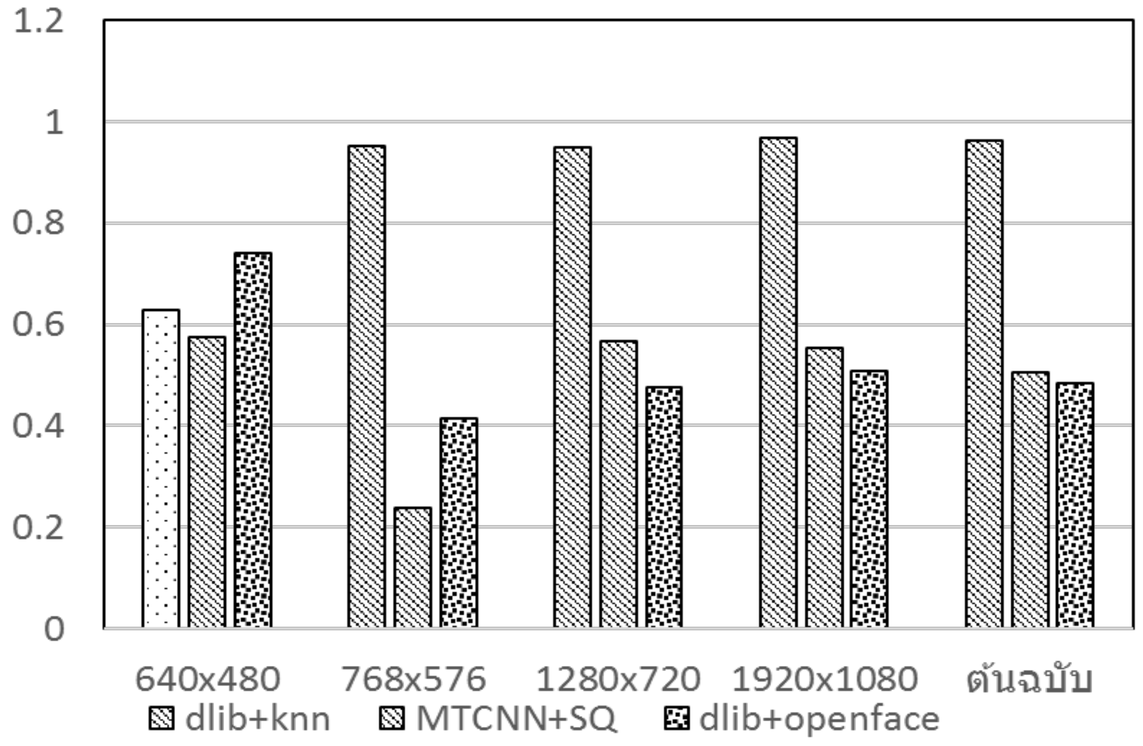
Different sampling rate

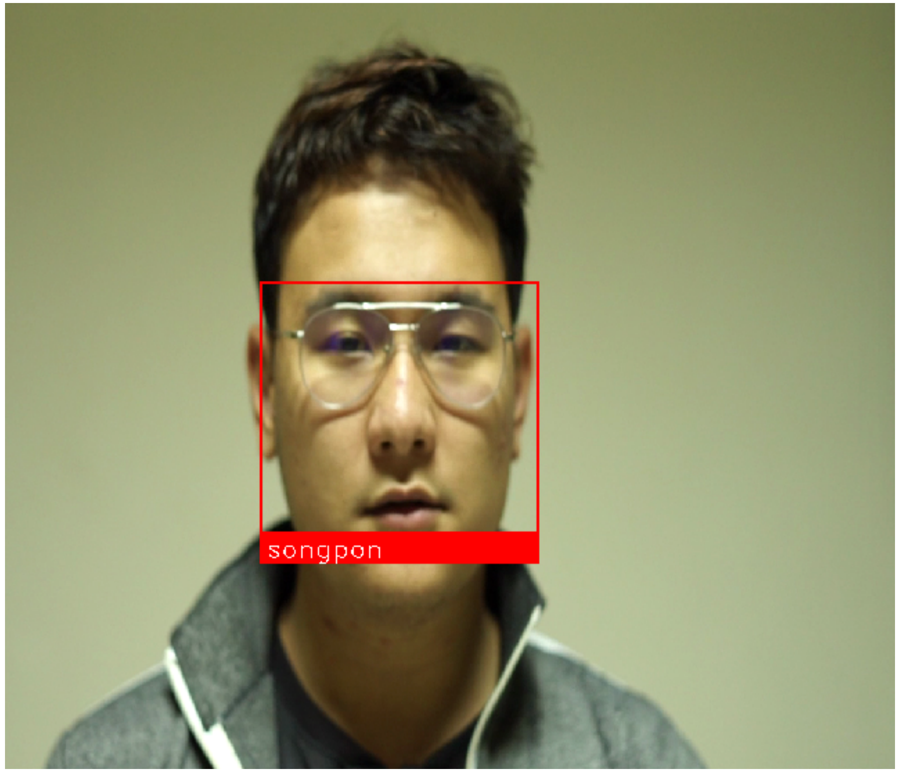


Different image size



Accuracy


1.2
1
0.8
0.6
0.4
0.2
0
640x480
768x576
1280x720
1920x1080
ต้นฉบับ
dlib+knn
MTCNN+SQ
dlib+openface


songpon


NackPanupong
soratouch
songpon


NackPanupong 98.47%
status = Undergrad
GPA = 2.95
soratouch 80.08%
status = Grad
GPA = 3.57
songpon 100.0%
status = Undergrad
GPA = 3.2

Discussions

Method	Accuracy	Size	Simplicity	Speed	Scalability	Sensitivit y
Deep learning (MTCNN+SQ)	Medium	Constant	Medium	H	High	Med
Openface	High	Varied	High	Low	Low	High
Dlib+knn	High	Varied	High	Low/ Med	Low	High

-Simplicity in retraining
-Speed of inference
-Scale to large data sets
-Sensitivity to smaller size/ Correctness

In summary

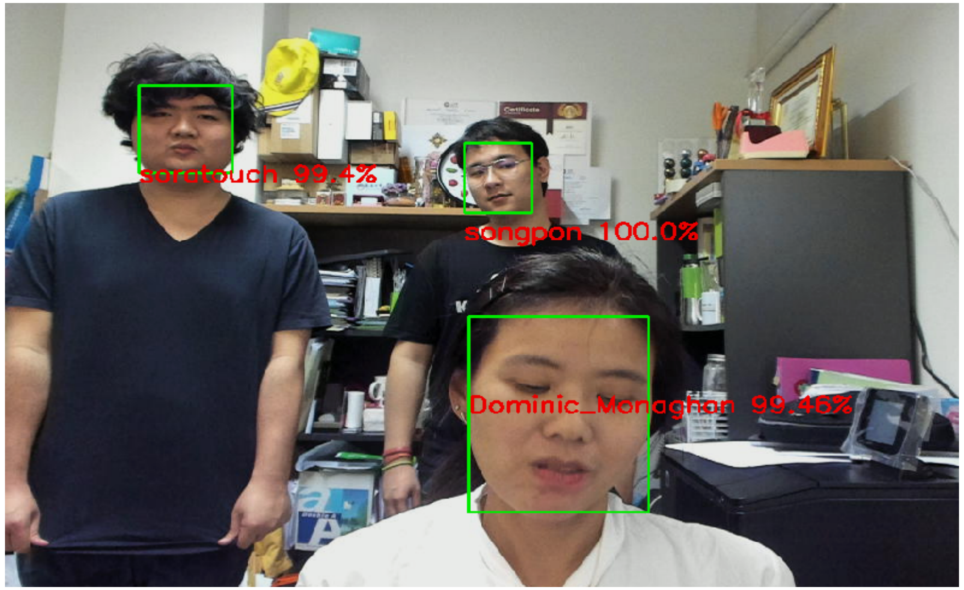
- Explore various face recognition tools including deep learning
- Study preprocessing method
- Study transfer learning approach

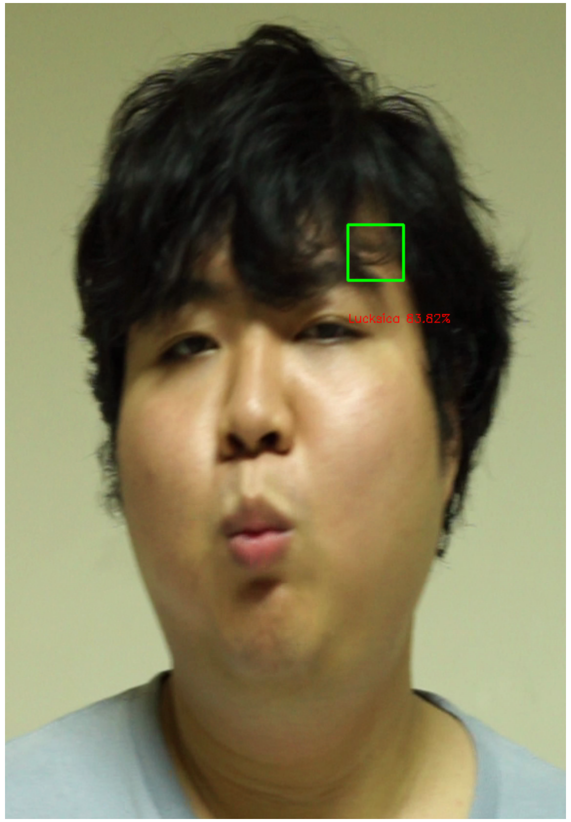
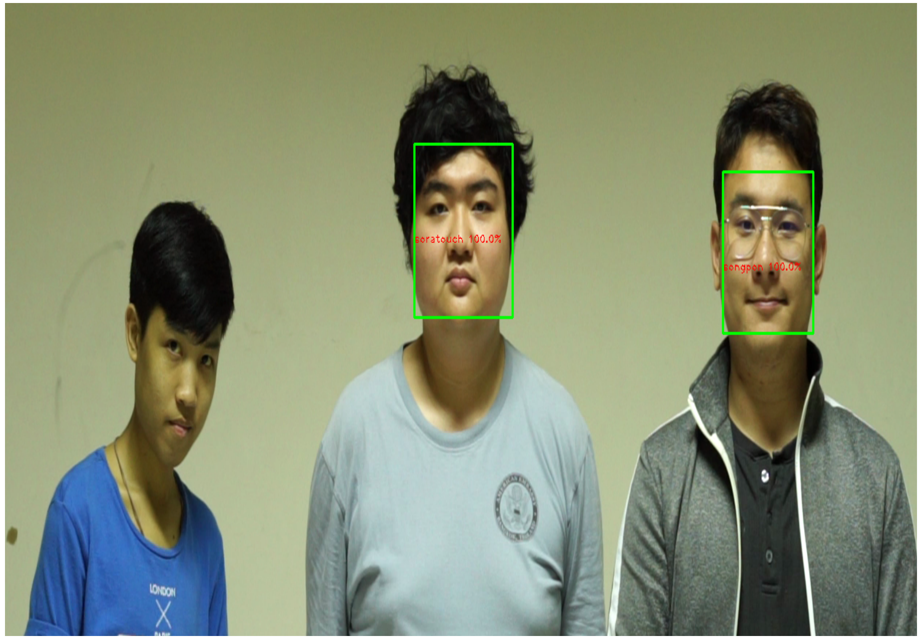
Deep learning approach can scale very well with large data sets.
-The more training data/ the more accuracy
OpenFace/dlib is limited to number of training images. High accuracy is obtained with small training set.

Suggestion to work with

- Fixed environment; lights and path
- Fixed pose for seconds, acceptable resolution.
- More training images, more accuracy
- Drawback: unclear image (blur/resolution), image with less landmark for detection,
  - False face detection (size threshold)
- In case with accuracy less than threshold: human judgement
  - In case of video, average predictions over frames

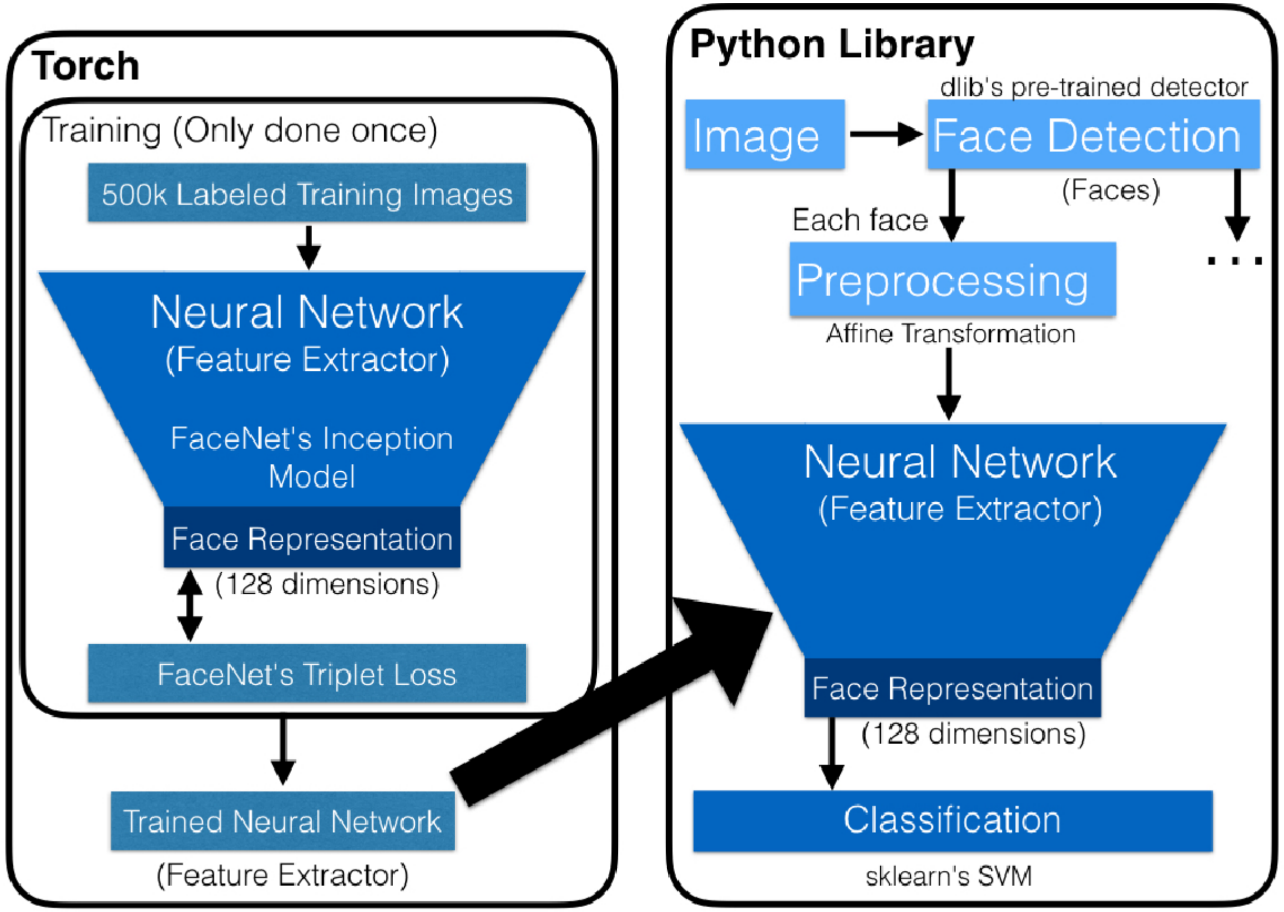
In case of unknown class: open set training approach


soratouch 99.4%
songpon 100.0%
Dominic_Monaghan 99.46%


Lucksica 83.82%

soratouch 100.0%
songpon 100.0%

Openface

https://blog.algorithmia.com/understanding-facial-recognition-openface/

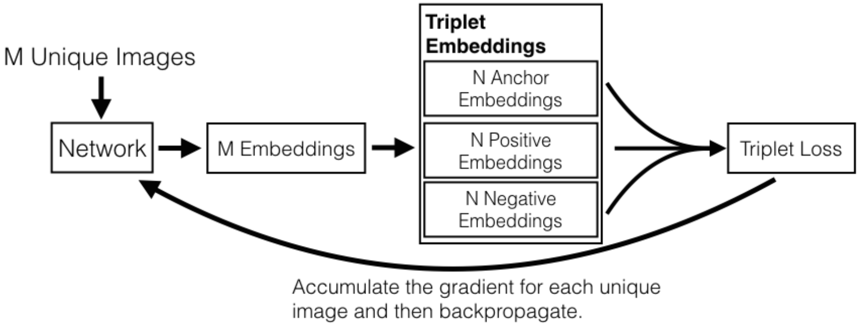


Use dlib face detection

Use openCV affine transformation (alig

Feature extraction (128 embeddings)

SVM training (clustering)



https://www.cs.cmu.edu/~satya/docdir/CMU-CS-16-118.pdf

Triplet Loss (FaceNet)

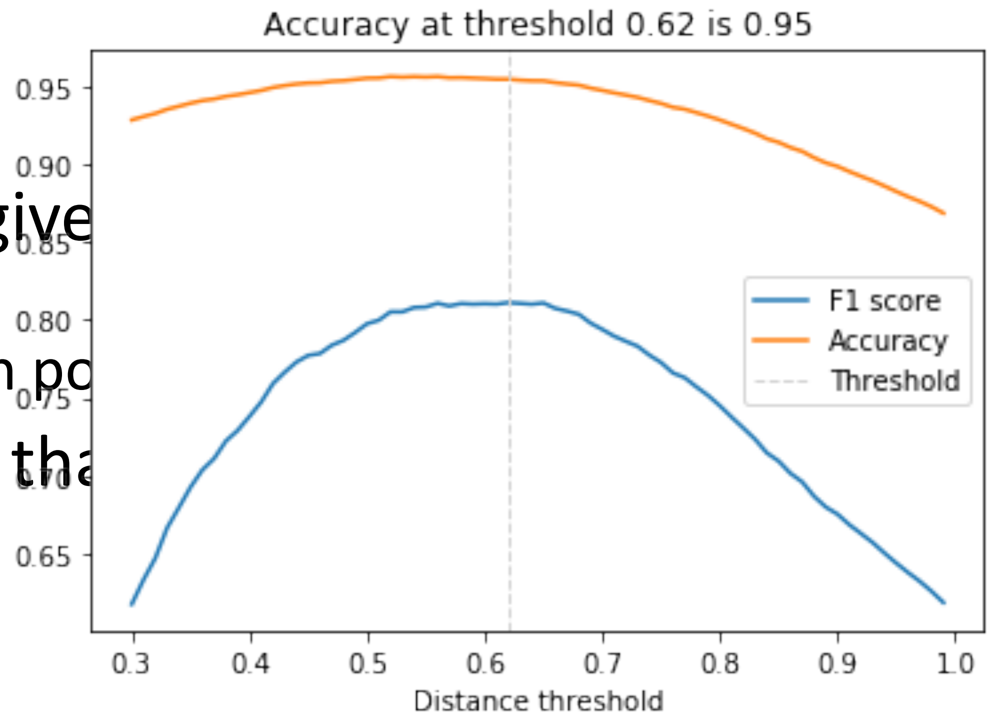
- from anchor, positive and negative embedding vectors
  - Squared L2 distance between all faces of the same identity is small and the distance between a pair of faces from different identities is large.
  - when the distance between an anchor image x_i^a and a positive image x_i^p (same identity) in embedding space is smaller than the distance between that anchor image and a negative image x_i^n (different identity) by at least a margin α.

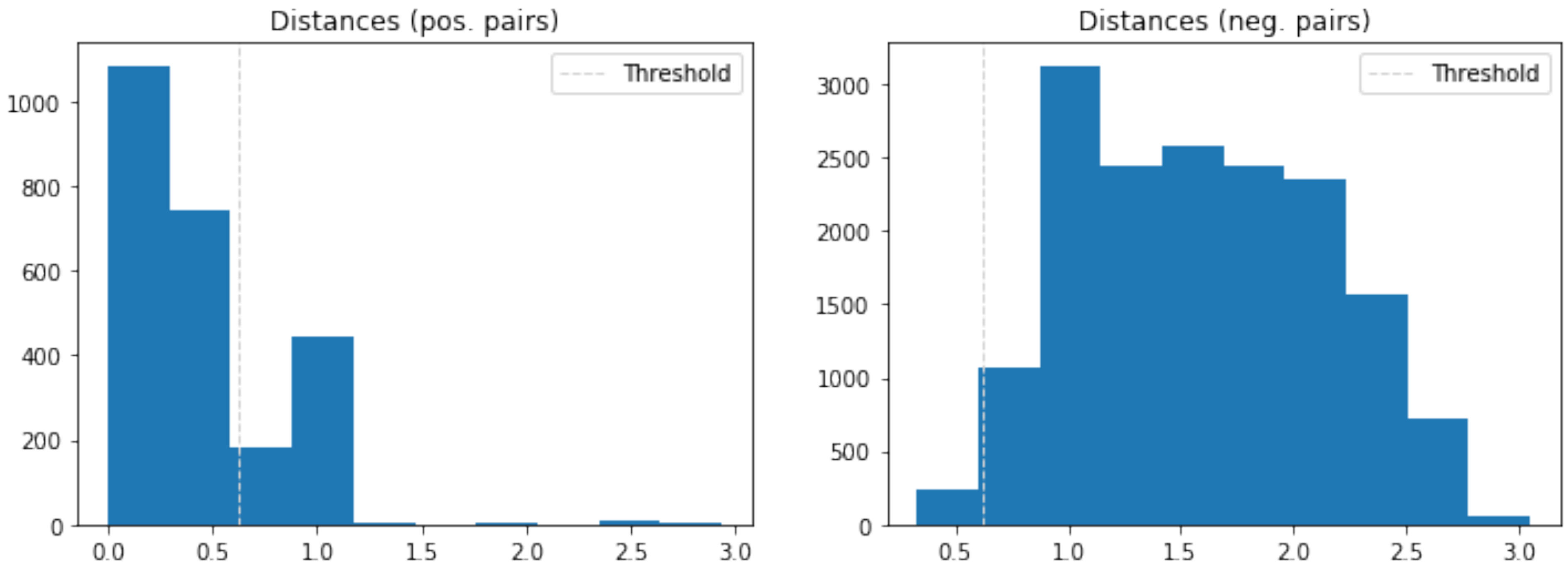
- Minimize loss $$L = \sum_{i=1}^{m} \left[\| f(x_i^a) - f(x_i^p)) \|_2^2 - \| f(x_i^a) - f(x_i^n)) \|_2^2 + \alpha \right]_+$$

Steps

- Align faces
- Find embeddings
- Find distances between pairs for all possible pairs
  - Find list of the same images (identical images)

- Find F1 for identical images with less than give
  threshold)
  - skewed classes (much more negative pairs than po
- Find accuracy for identical images with less tha
  (Varying threshold)





Distance distribution between positive pairs and negative pairs

- Given an estimate of the distance threshold τ, face recognition is as calculating the distances between an input embedding vector and all embedding vectors in a database.
- The input is assigned the label (i.e. identity) of the database entry with the smallest distance if it is less than τ or label *unknown* otherwise.

https://github.com/krasserm/face-recognition

- face-recognition1.ipynb
- github.com/cchantra/face_training

https://github.com/habrman/FaceRecognition

- python3 main.py ./models/models.pb ./picture_train/

https://cmusatyalab.github.io/openface/

demo

OpenCV face recognition helper libraries

- Dlib face detection (HOG Histogram of Oriented Gradient, Support Vector Machine or OpenCV's Haar cascade classifier)
- Find face landmark

Affine transform