An overview of face recognition

Presenter: Trần Tuấn Anh

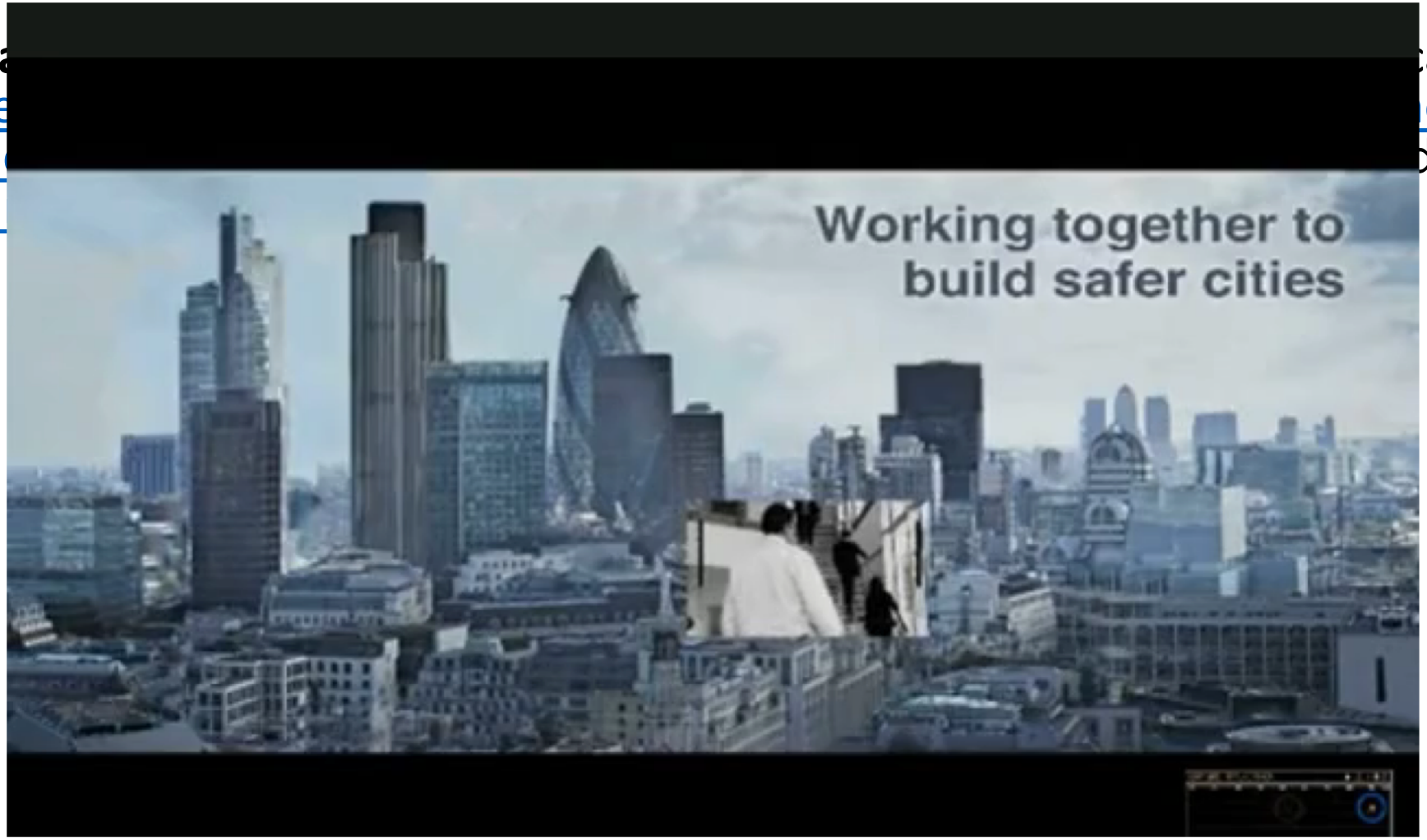
Site: https://sites.google.com/site/trtanh1988/home

Face recognition, an introduction

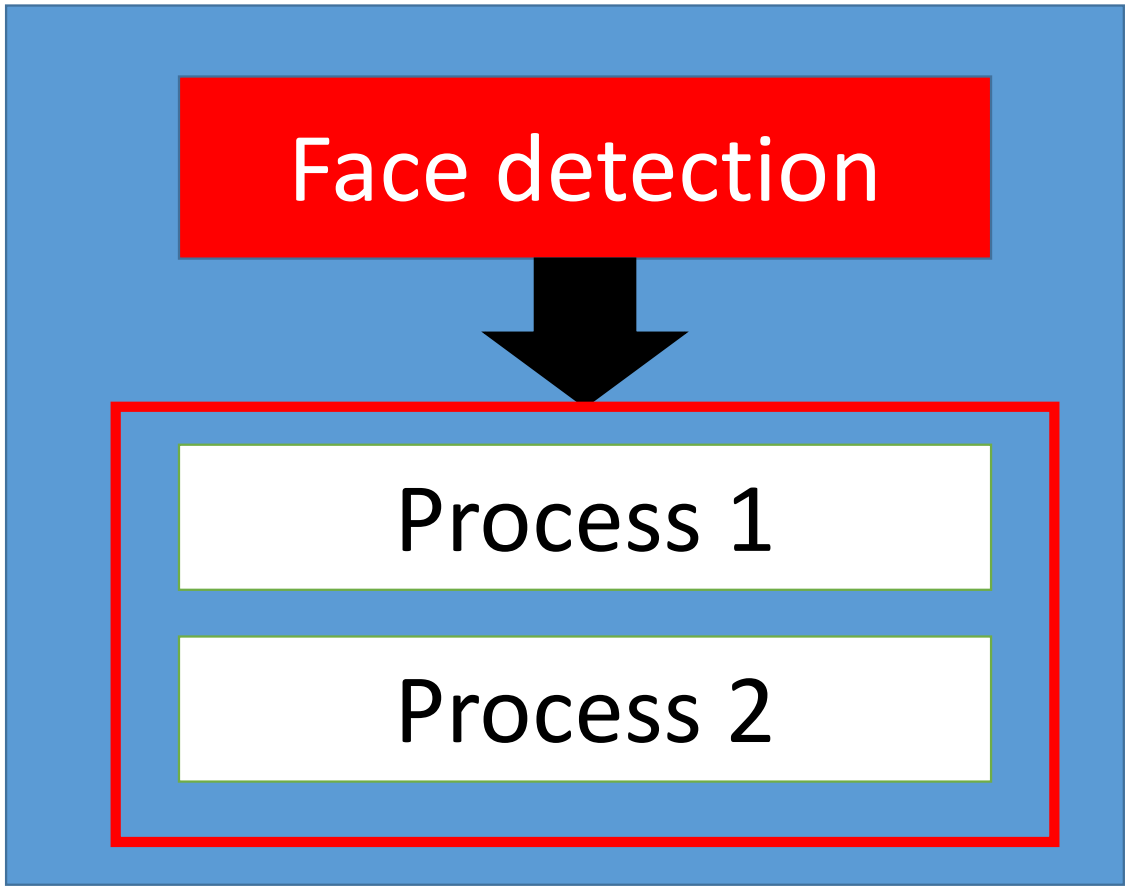
- It is a century problem in computer vision
- Application of face recognition is really wide: criminal searching; monitoring; security; VIP identification

1. What is face recognition?

- A **fa** capable of ide e from a vid d facial featu



1. What is face recognition?



Face recognition

2. Face detection

Find faces in pictures

Find all the faces that appear in a picture:

Input

Output

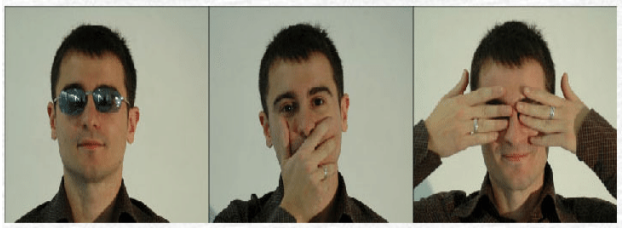
2.1 Motivation of face detection

- Face Detection: A Solved Problem?

Different sized faces

Different poses faces

Different illumination in same face



Different occluded faces

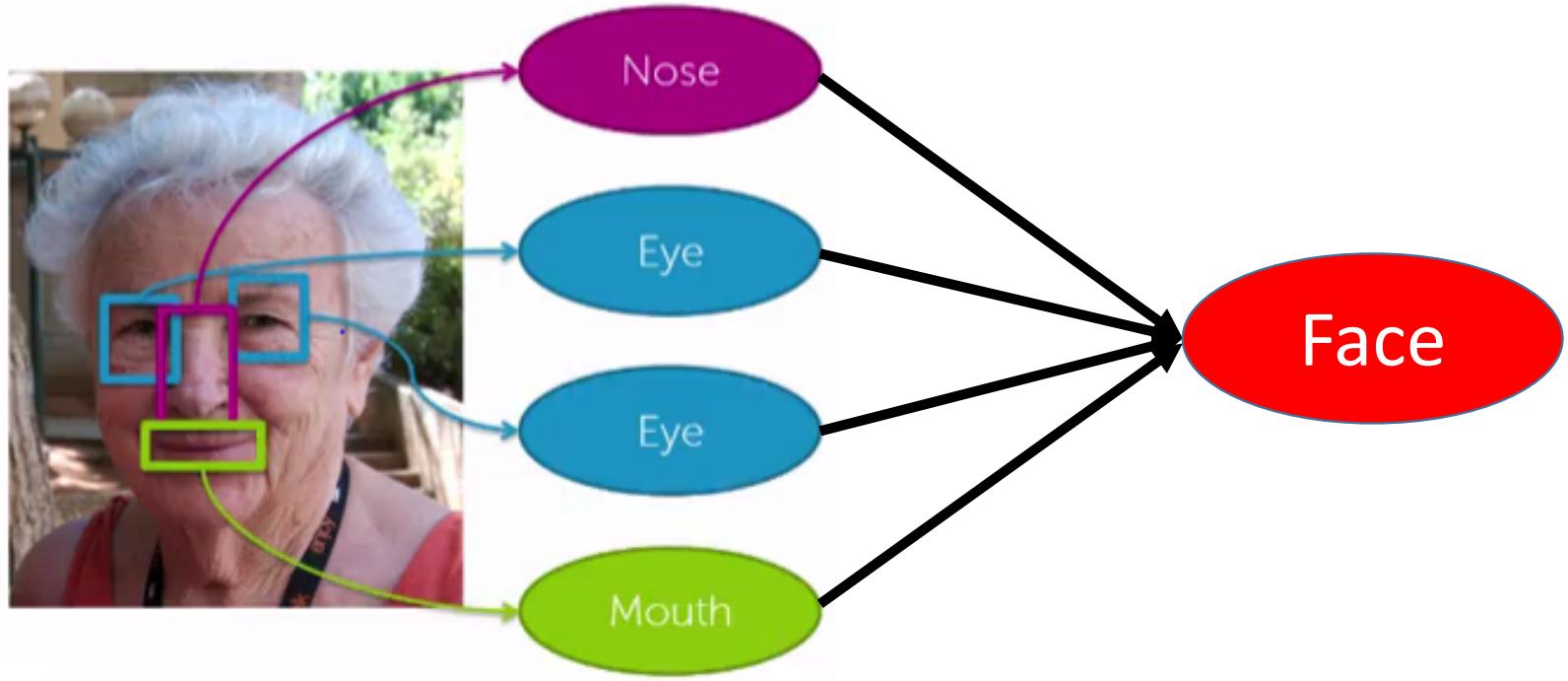
Faces in clutter

Different expressions in same face

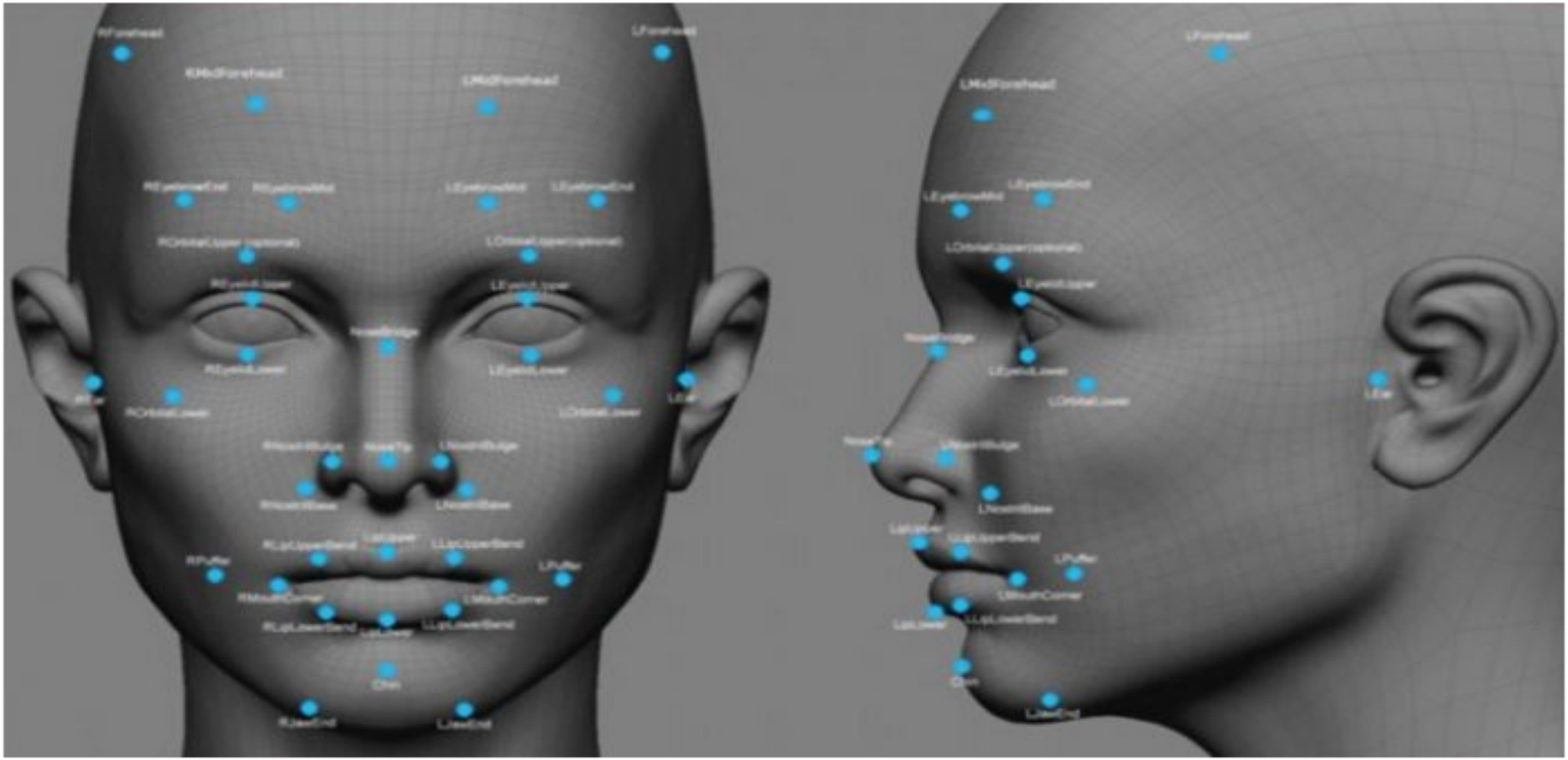
2.2 Face detection model

Image features

Features = local detectors
- Features are combined to make prediction
- In reality, features are more low-level

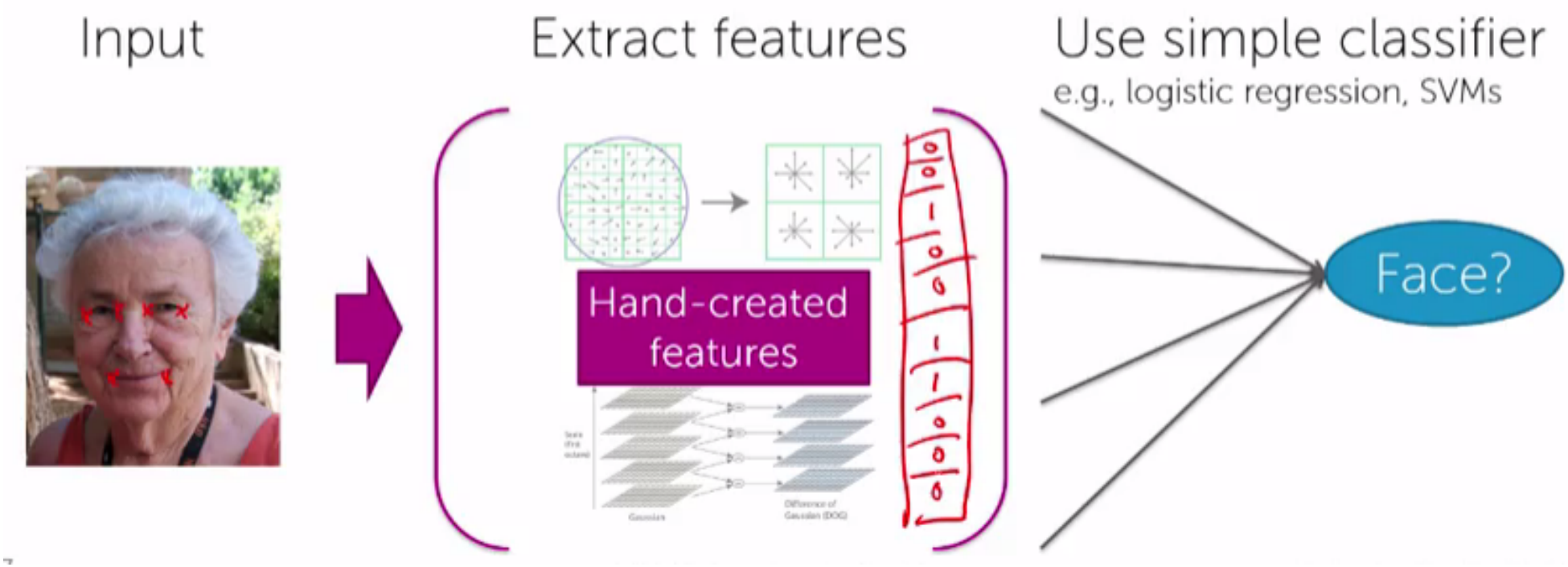


2.2 Face detection model



2.2 Face detection model

Standard image classification approach



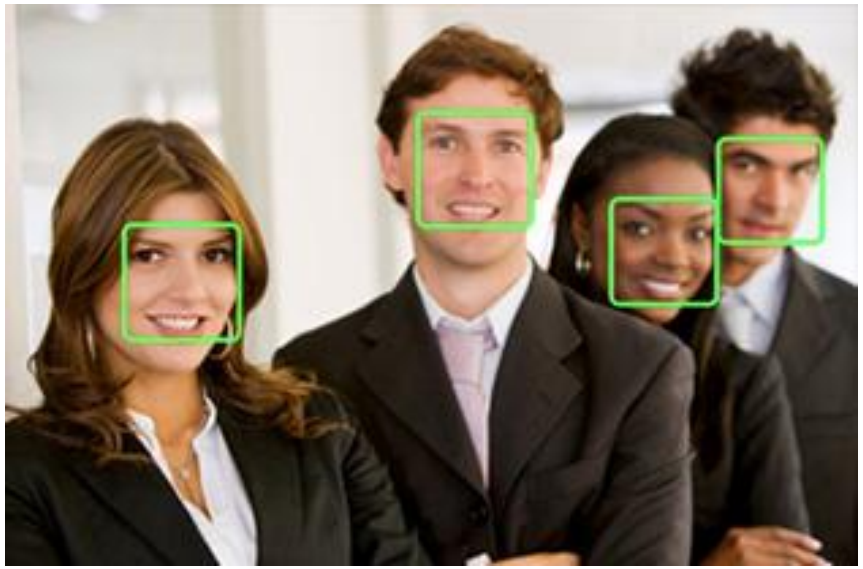
2.2 Face detection model

Until 2000, there are many different techniques was proposed to find the face, but all are either slow or not trusting or both.

A major change occurred in 2001 when Viola and Jones invented the Haar-based cascade classifier for object identification, and in 2002 it was improved by Lienhart and Maydt.

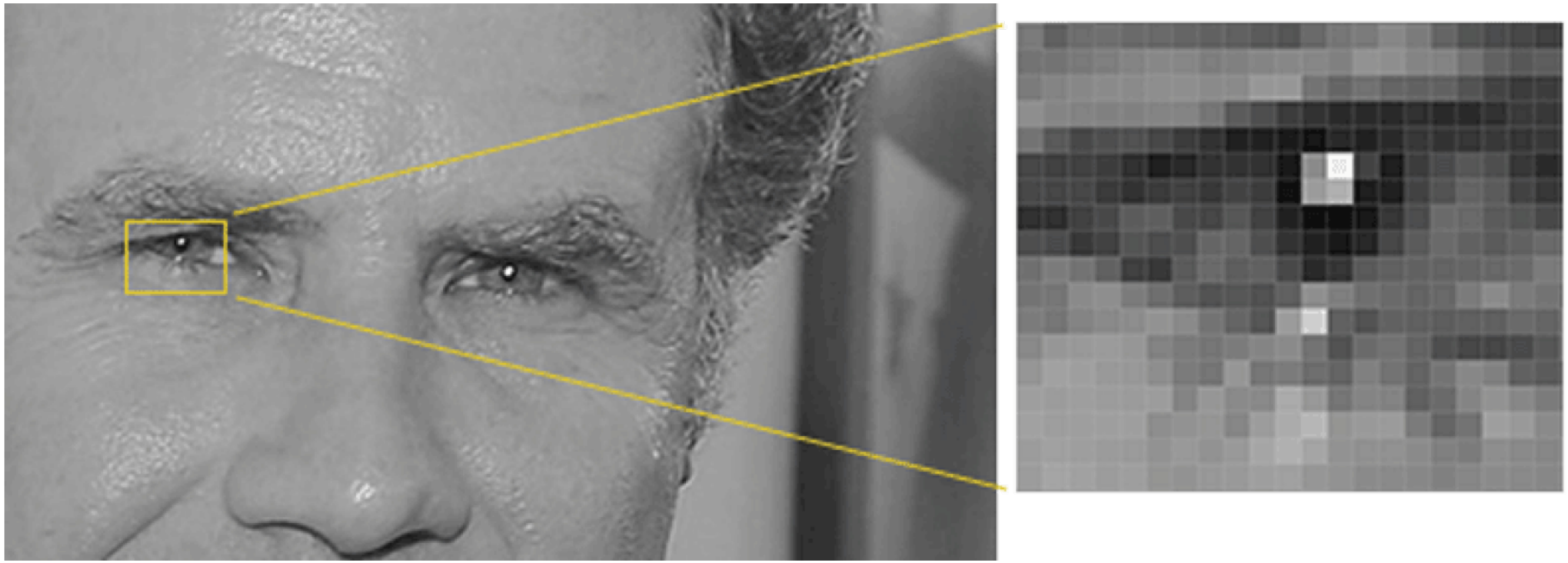
As a result, object identification is fast enough (real-time detection on web-enabled computers) and reliable (95% accuracy).

2.3 An example of face detection



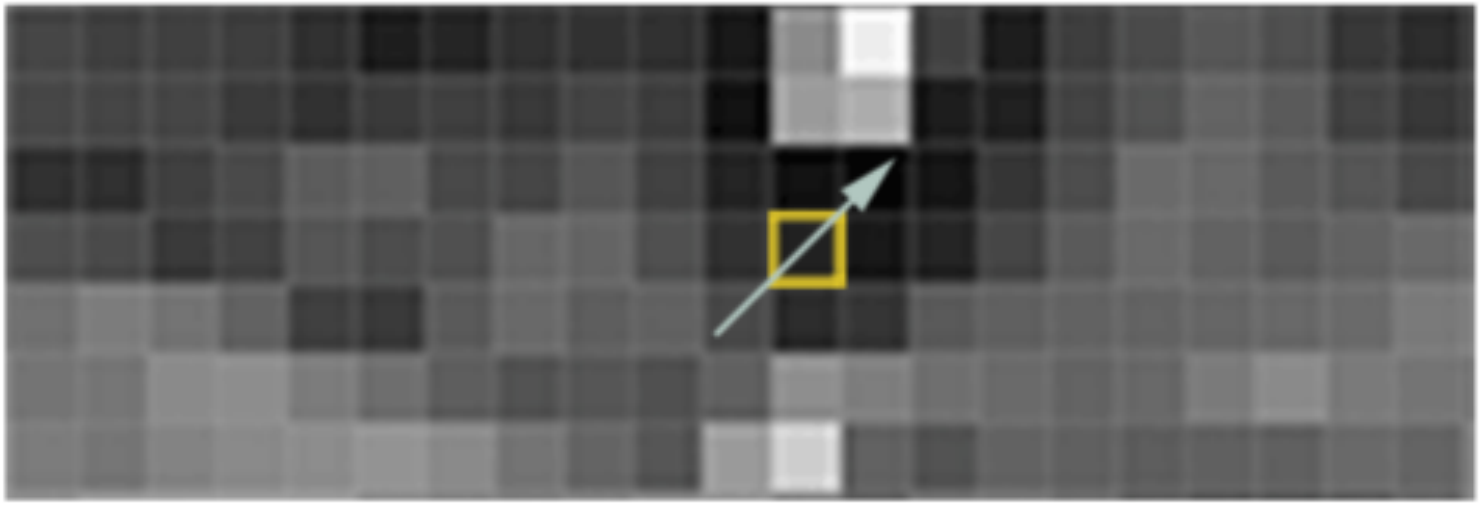
Gray image

2.3 An example of face detection



Consider every pixel and its neighbor

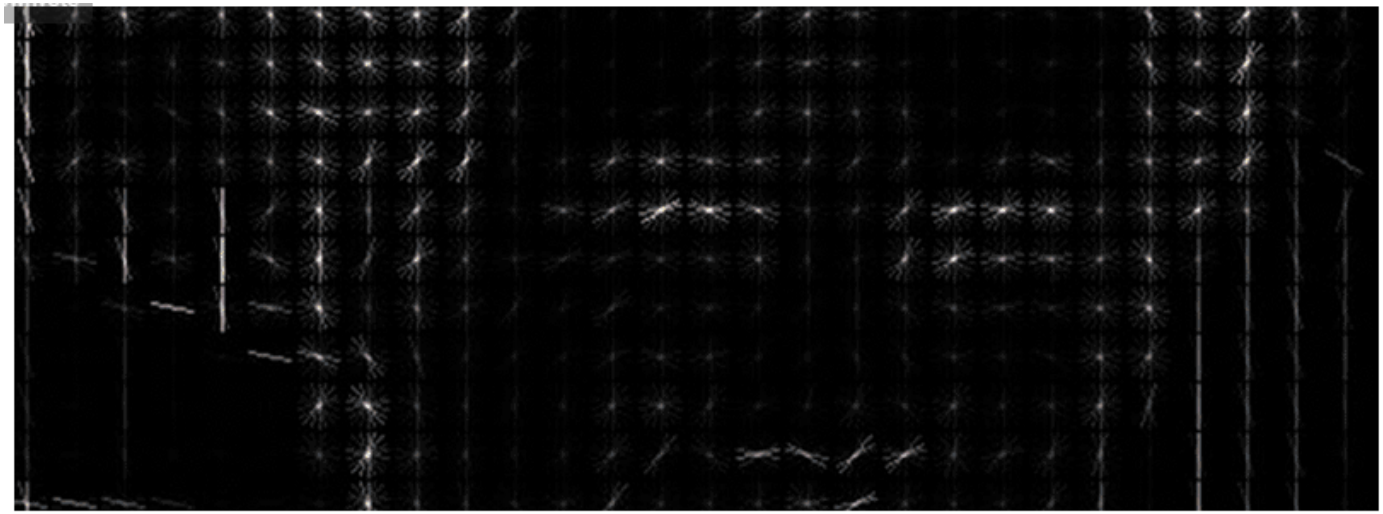
2.3 An example of face detection



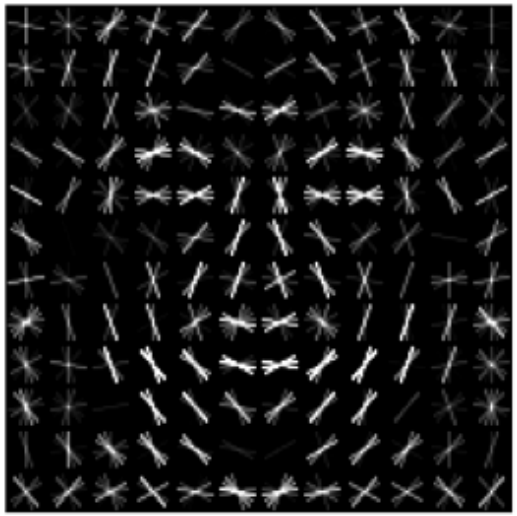
Gradient: a flow from bright to dark pixel

2.3 An example of face detection

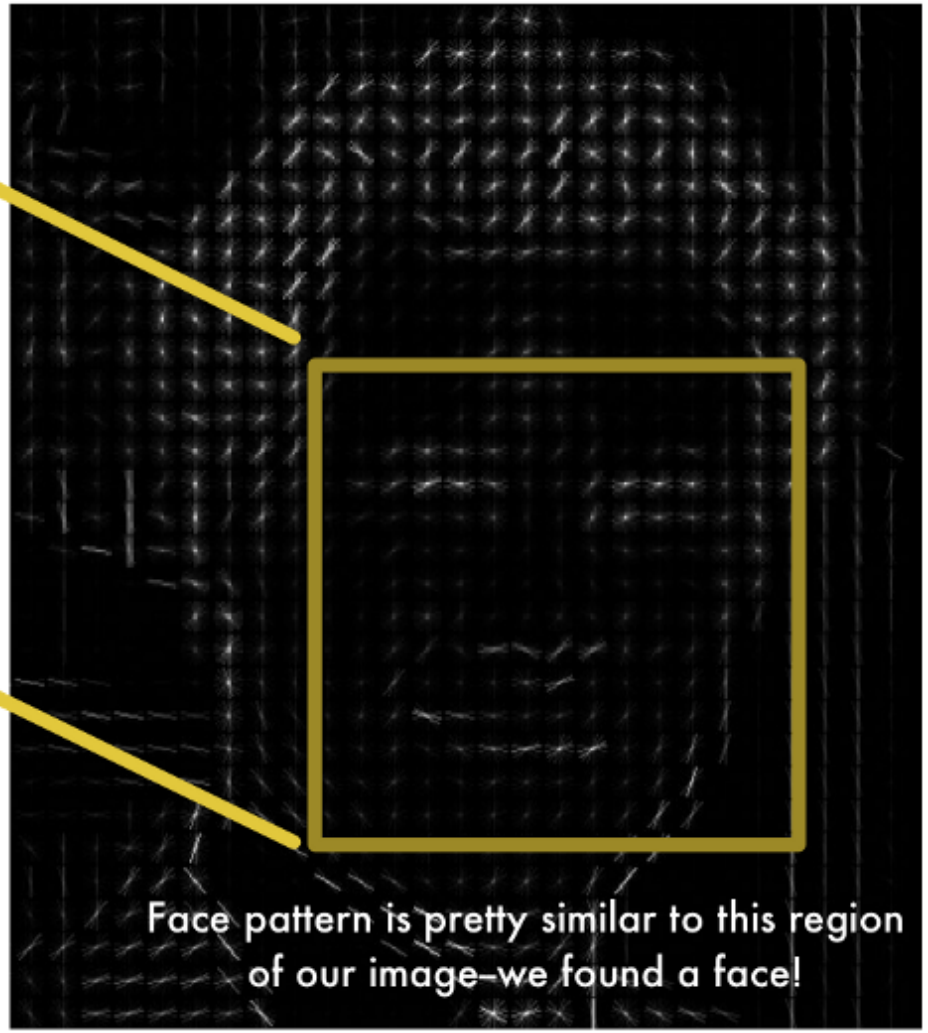
16x16 scale window size



2.3 An example of face detection

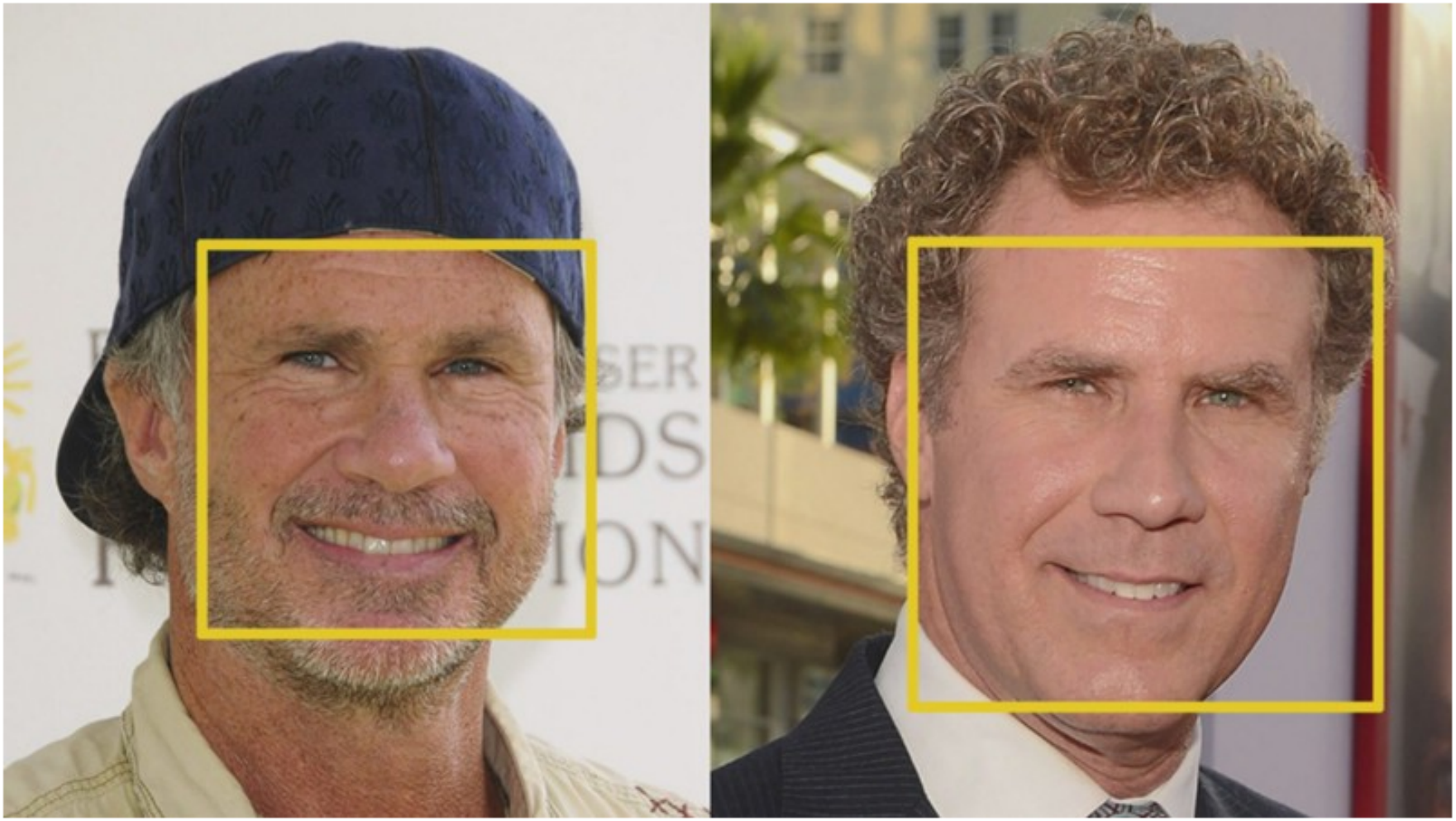


HOG face pattern generated from lots of face images

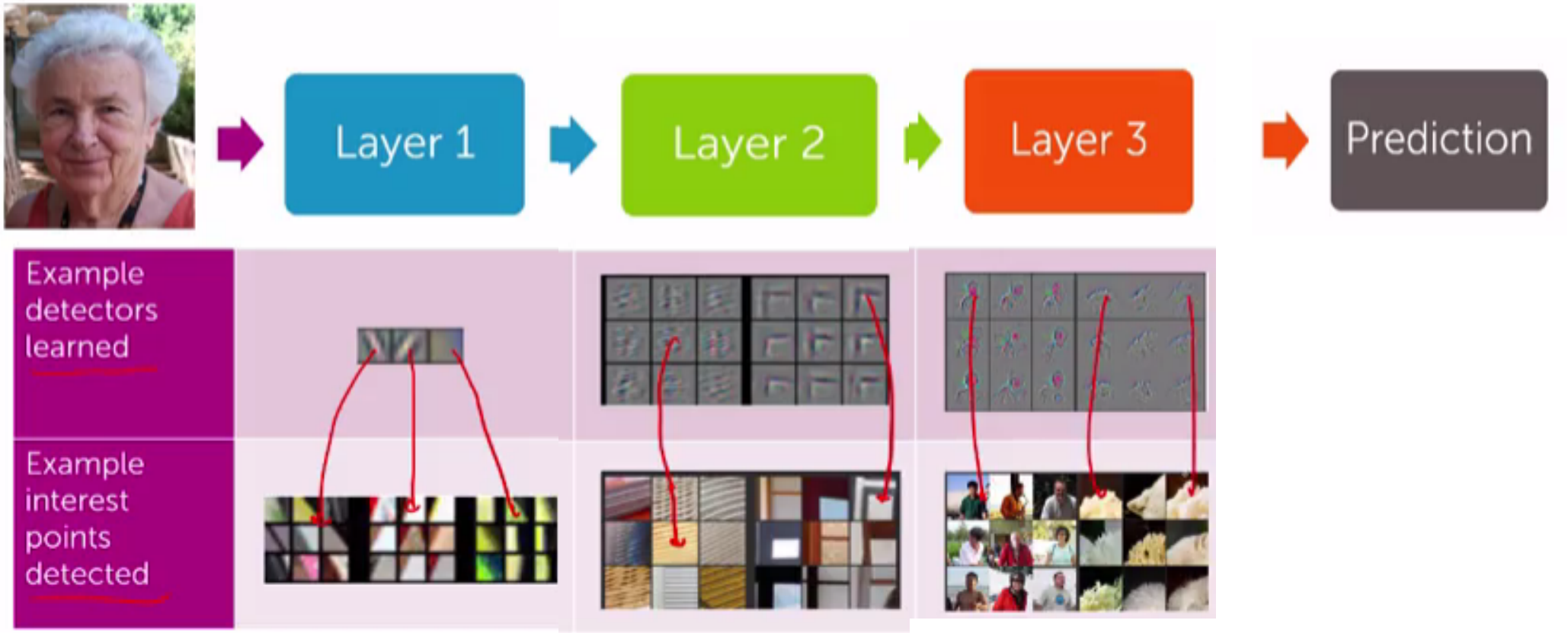


HOG version of our image

2.3 An example of face detection

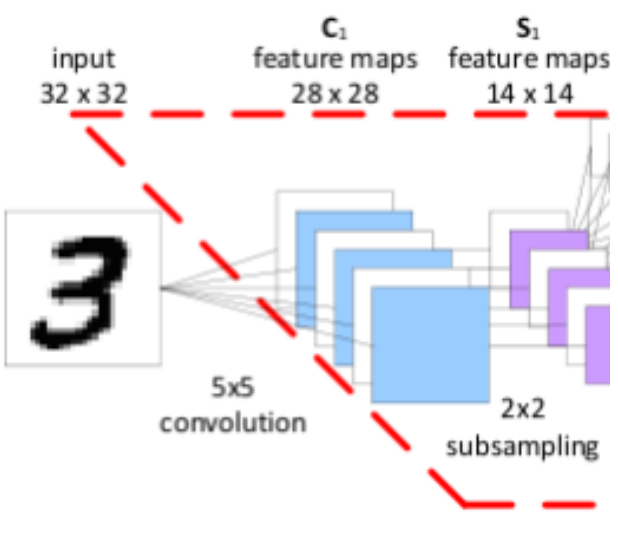


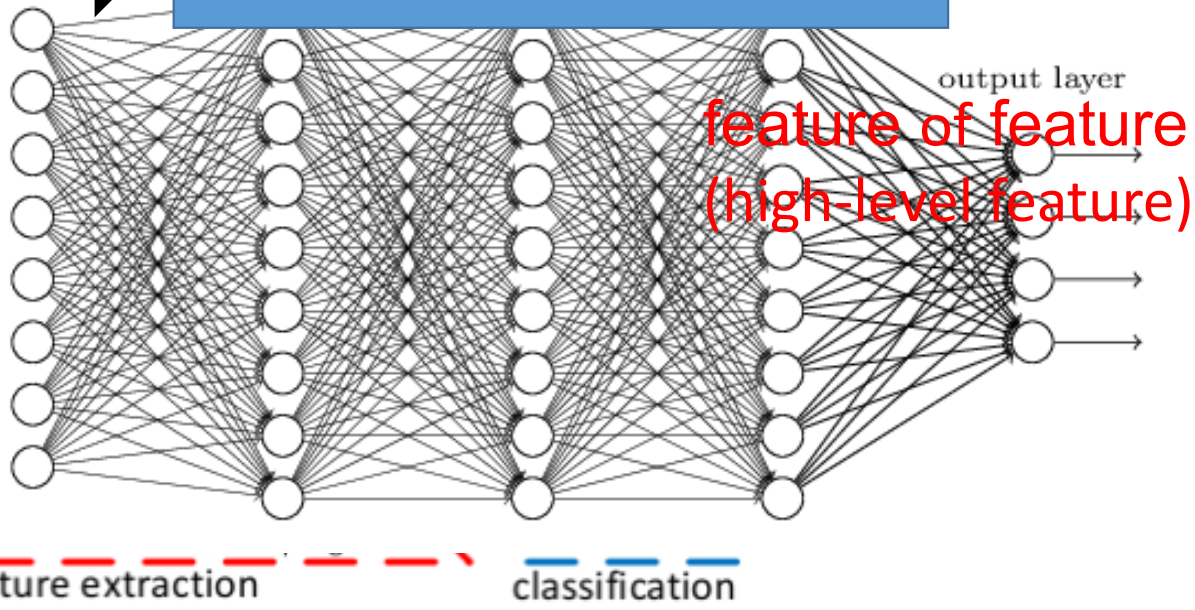
2.4 Face detection using deep learning



2.4 Why Deep Learning?

Machine learning	Deep learning
The network structure	Complex neural network
Manual feature extraction	Automatic feature extraction





Convolutional Neural Networks

2.5 Survey in recently years

1. Li, Haoxiang and Lin, Zhe and Shen, Xiaohui and Brandt, Jonathan and Hua, G. (2015). ***A Convolutional Neural Network Approach for Face Detection***. *Cvpr*, 5325–5334. https://doi.org/10.1109/CVPR.2015.7299170

Main contribution:

A hierarchical CNN network structure is proposed for high-speed face detection.

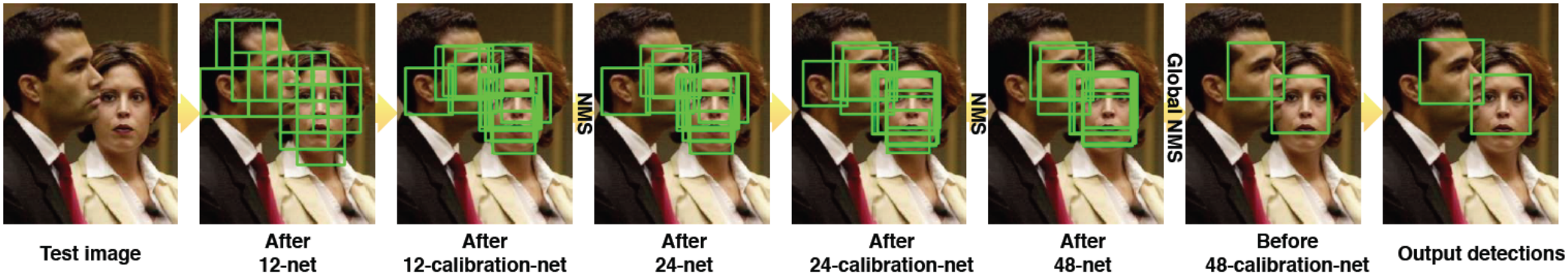
A boundary correction network is designed to better position the face.

Proposed ***a multi-resolution CNN network structure***, has a stronger ability to identify than a single network structure, and a small additional overhead.

In the FDDB reached the ***highest score at the time***.

2.5 Survey in recently years

Implementation:
https://github.com/anson0910/CNN_face_detection



2.5 Survey in recently years

2. Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). ***Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks***. *IEEE Signal Processing Letters*, *23*(10), 1499–1503. https://doi.org/10.1109/LSP.2016.2603342

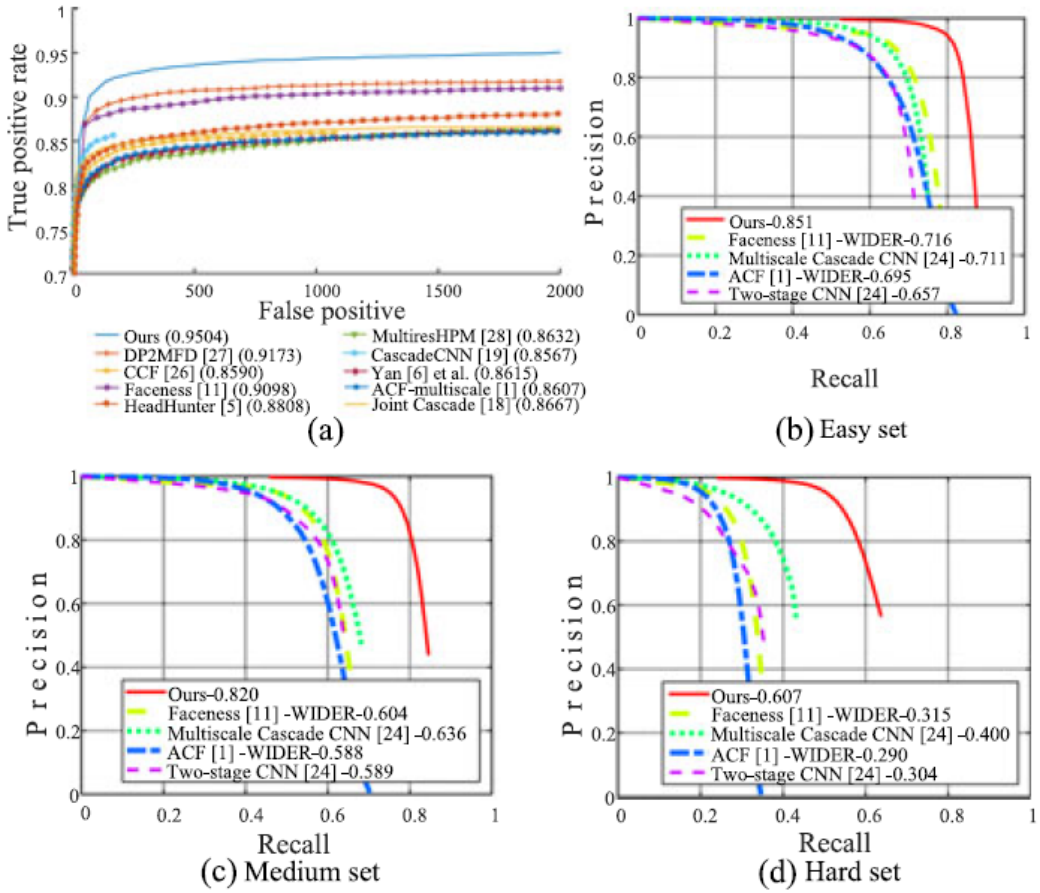
Main contribution:

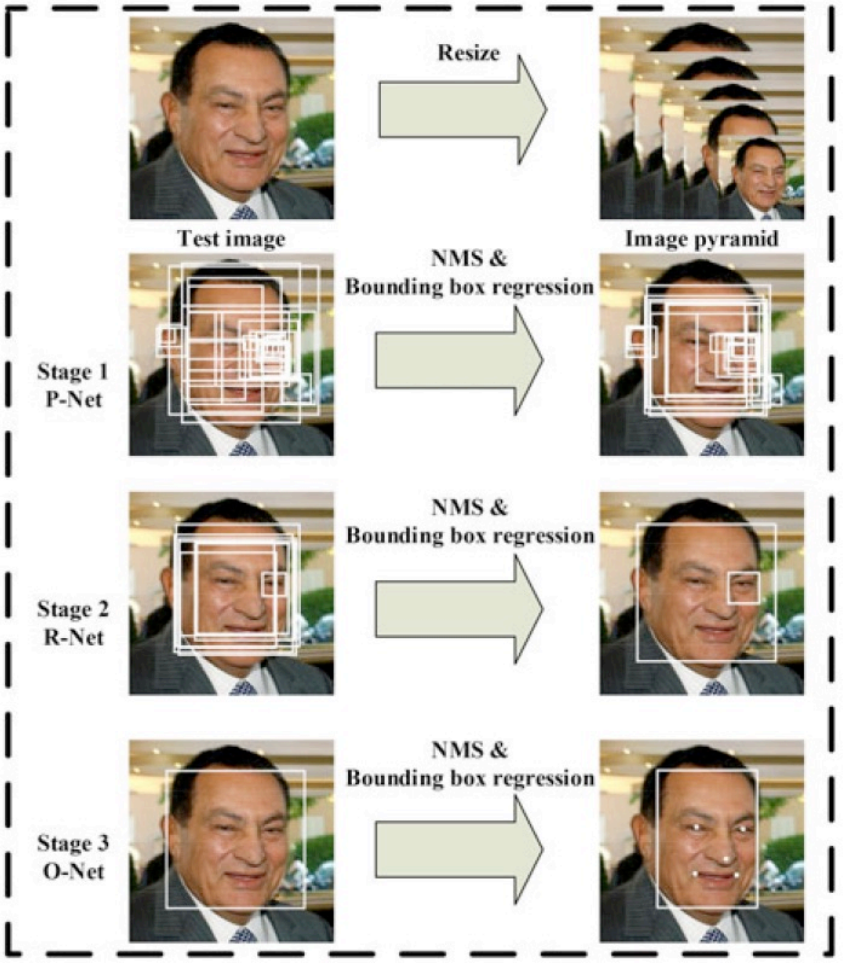
Exploits the ***inherent correlation between detection and alignment*** to boost up their performance.

Third-order cascade convolution neural network on the task from coarse to fine processing.

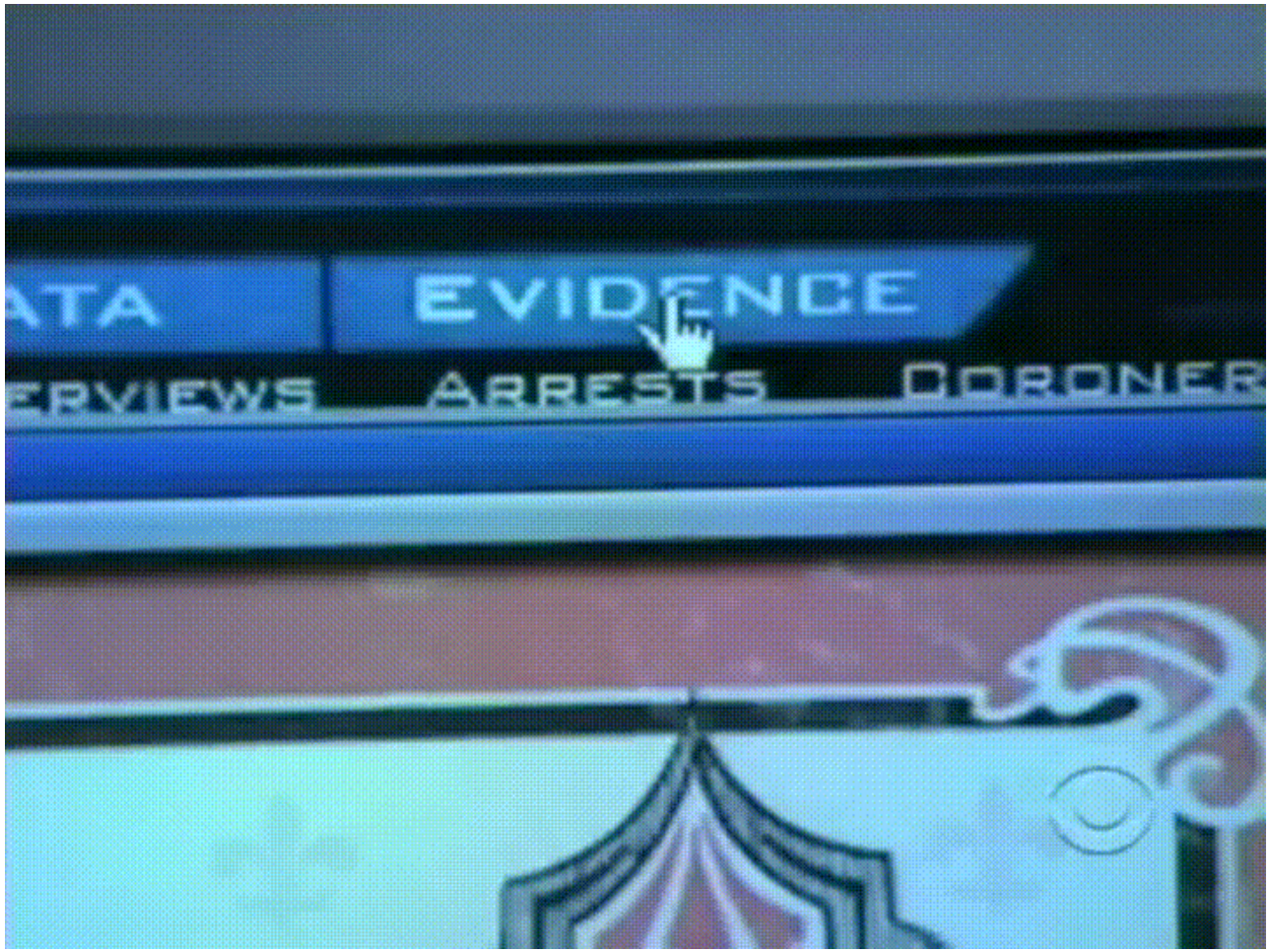
A new ***online difficult sample generation strategy*** can further improve performance.

2.5 Survey in recently years

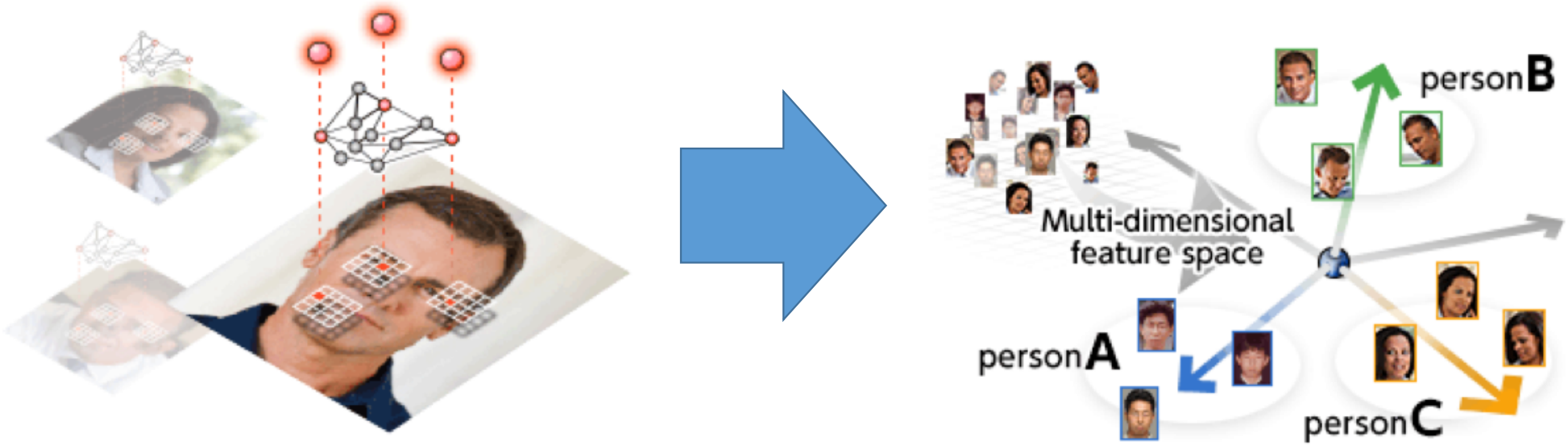




3. Face recognition

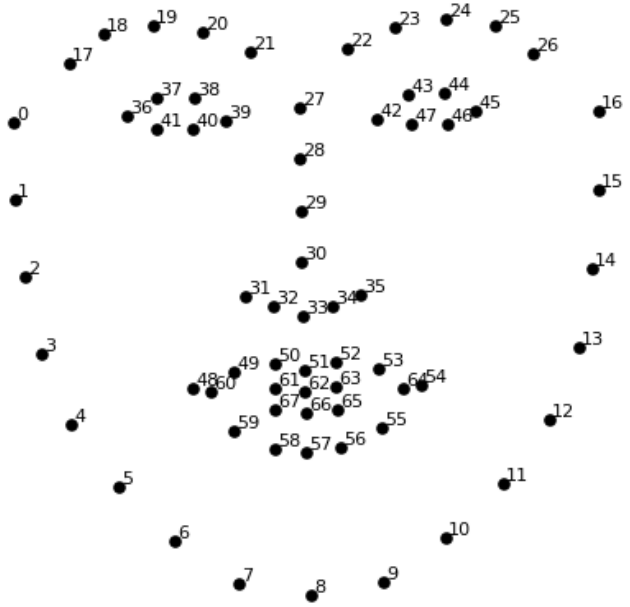


3. Face recognition

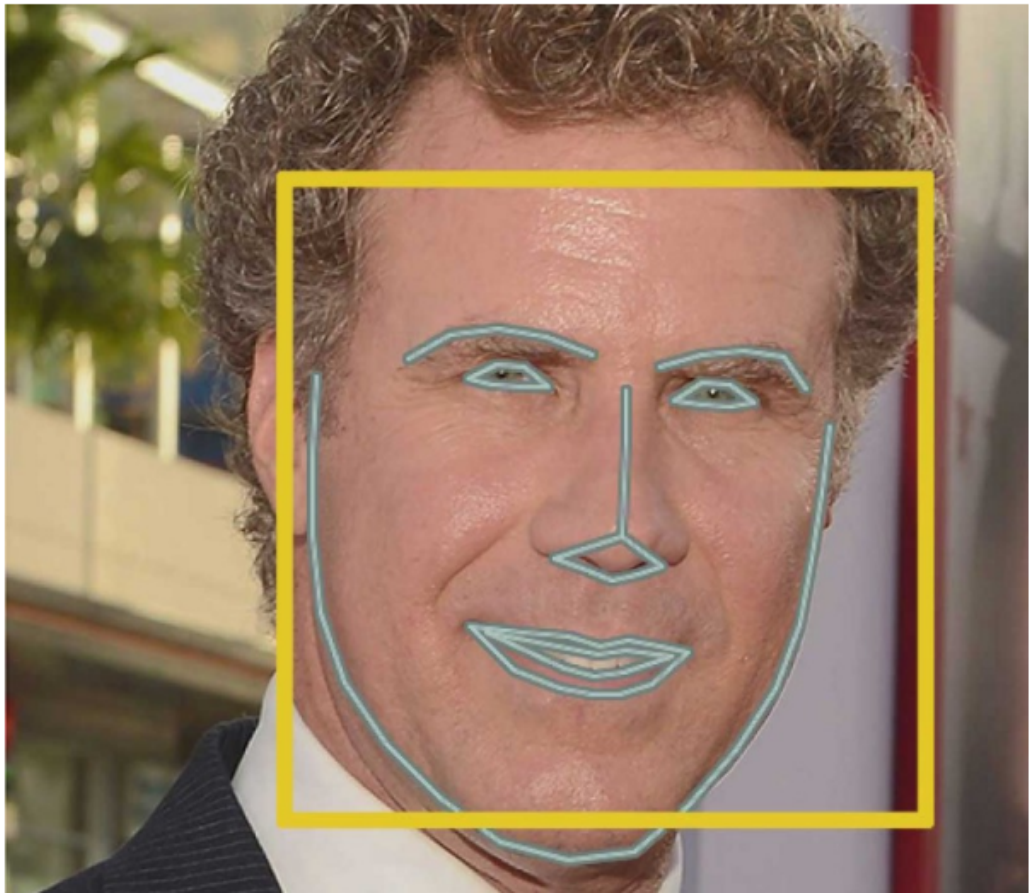


3.1 Posing and Projecting Faces

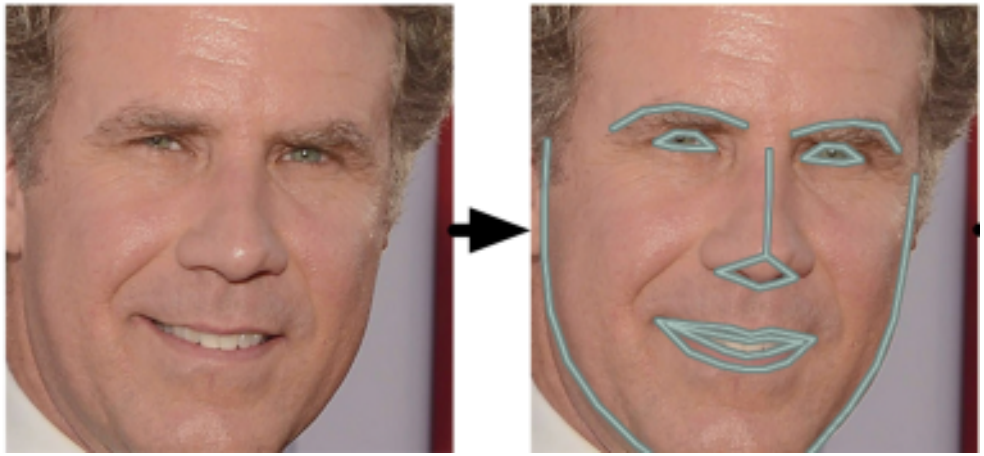
Face landmark estimation



Extract 68 landmarks (points) on the face



3.1 Posing and Projecting Faces



Face area detected in image

Face landmarks detected

Face transformed to be a close as possible to perfectly cetered

The perfectly centered result we want

3.2 Encoding faces

Comparison

How to

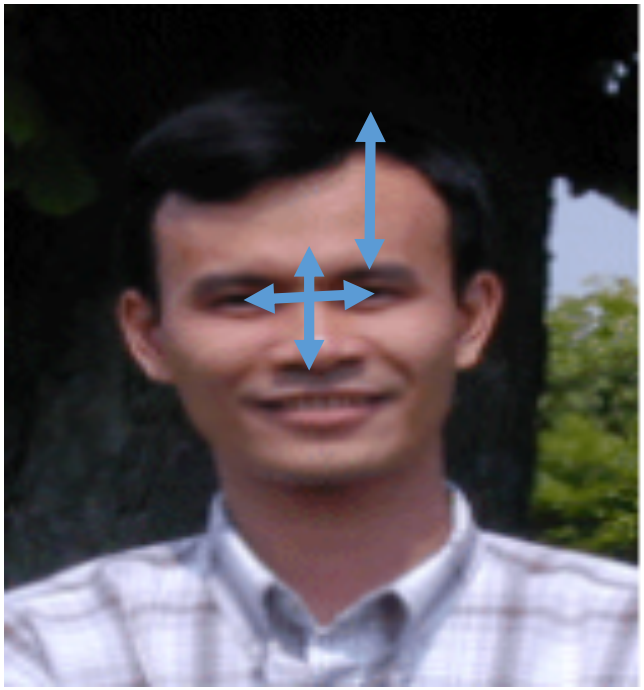
Metric

these faces?

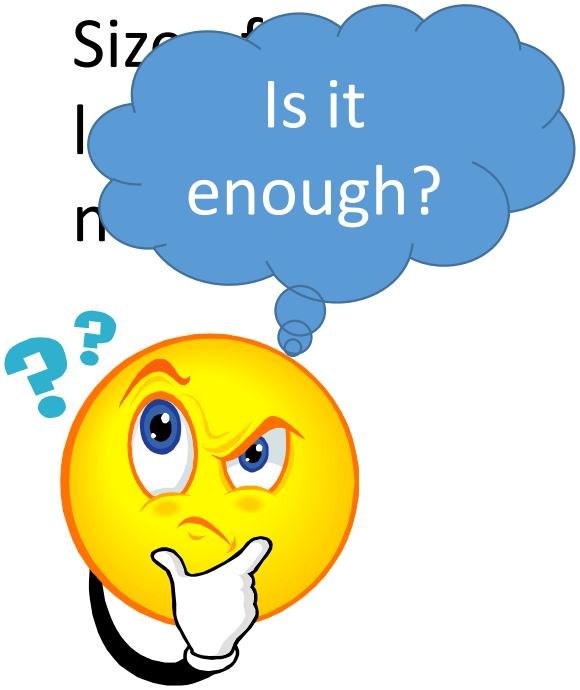
3.2 Encoding faces

directly compare the unknown face we found in Step 2 wit[illegible] pic[illegible] of [illegible] have alre[illegible]dy be[illegible]ed

3.2 Encoding faces



Distance between eyes
Size of nose,
Siz
l
n



3.2 Encoding faces

Deep Convolutional Neural Network, we are going to train it to generate **128 measurements** for each face.

The training process works by looking at 3 face images at a time:

1.Load a training face image of a known person
2.Load another picture of the same known person
3.Load a picture of a totally different person

3.2 Encoding faces

Picture of Will Ferrell

Another picture of Will Ferrell

Picture of Chad Smith

128 measurements generated by neural net

128 measurements generated by neural net

128 measurements generated by neural net

Compare results

Measurements

Measurements for the two will Farrell pictures are closer and the Chad Smith measurements are further away

3.2 Encoding faces

Input Image

128 Measurements Generated from Image

0.097496084868908	0.045223236083984	-0.1281466782093	0.032084941864014
0.12529824674129	0.060309179127216	0.17521631717682	0.020976085215807
0.030809439718723	-0.01981477253139	0.10801389068365	-0.00052163278451189
0.036050599068403	0.065554238855839	0.0731306001544	-0.1318951100111
-0.097486883401871	0.1226262897253	-0.029626874253154	-0.0059557510539889
-0.0066401711665094	0.036750309169292	-0.15958009660244	0.043374512344599
-0.14131525158882	0.14114324748516	-0.031351584941149	-0.053343612700701
-0.048540540039539	-0.061901587992907	-0.15042643249035	0.078198105096817
-0.12567175924778	-0.10568545013666	-0.12728653848171	-0.076289616525173
-0.061418771743774	-0.074287034571171	-0.065365232527256	0.12369467318058
0.046741496771574	0.0061761881224811	0.14746543765068	0.056418422609568
-0.12113650143147	-0.21055991947651	0.0041091227903962	0.089727647602558
0.061606746166945	0.11345765739679	0.021352224051952	-0.0085843298584223
0.061989940702915	0.19372203946114	-0.086726233363152	-0.022388197481632
0.10904195904732	0.084853030741215	0.09463594853878	0.020696049556136
-0.019414527341723	0.0064811296761036	0.21180312335491	-0.050584398210049
0.15245945751667	-0.16582328081131	-0.035577941685915	-0.072376452386379
-0.12216668576002	-0.0072777755558491	-0.036901291459799	-0.034365277737379
0.083934605121613	-0.059730969369411	-0.070026844739914	-0.045013956725597
0.087945111095905	0.11478432267904	-0.089621491730213	-0.013955107890069
-0.021407851949334	0.14841195940971	0.078333757817745	-0.17898085713387
-0.018298890441656	0.049525424838066	0.13227833807468	-0.072600327432156
-0.011014151386917	-0.051016297191381	-0.14132921397686	0.0050511928275228
0.0093679334968328	-0.062812767922878	-0.13407498598099	-0.014829395338893
0.058139257133007	0.0048638740554452	-0.039491076022387	-0.043765489012003
-0.024210374802351	-0.11443792283535	0.071997955441475	-0.012062266469002
-0.057223934680223	0.014683869667351	0.05228154733777	0.012774495407939
0.023535015061498	-0.081752359867096	-0.031709920614958	0.069833360612392
-0.0098039731383324	0.037022035568953	0.11009479314089	0.11638788878918
0.020220354199409	0.12788131833076	0.18632389605045	-0.015336792916059
0.0040337680839002	-0.094398014247417	-0.11768248677254	0.10281457751989
0.051597066223621	-0.10034311562777	-0.040977258235216	-0.082041338086128

3.3 Finding the person's name from the encoding

We can do that by using any basic machine learning classification algorithm. No fancy deep learning tricks are needed. We'll use a simple linear SVM classifier, but lots of classification algorithms could work.

a classifier with the embeddings of about 20 pictures each of Will Ferrell, Chad Smith and Jimmy Falon

3.3 Finding the person's name from the encoding

3.4 Survey in recent years

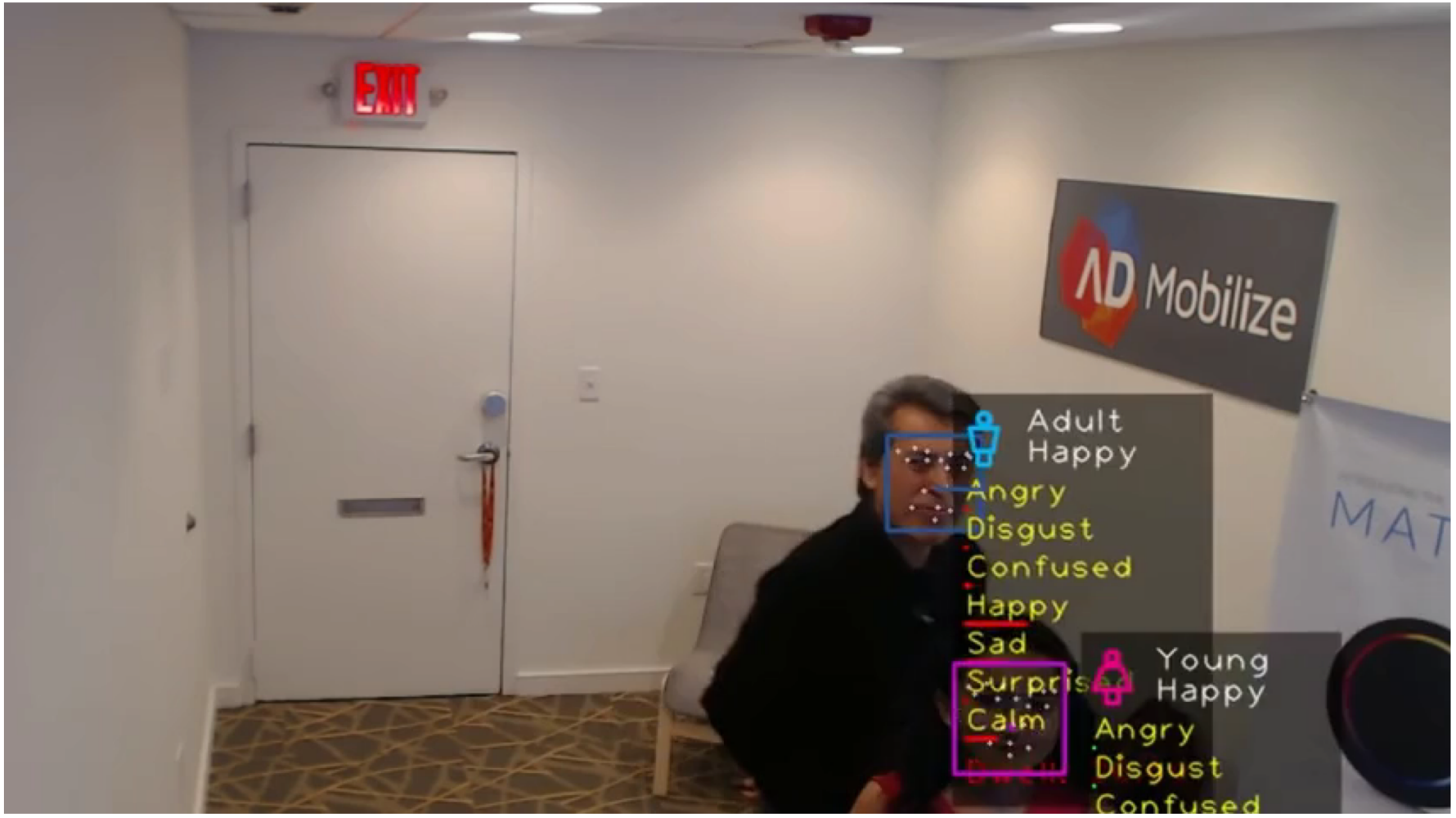
Facenet: https://github.com/davidsandberg/facenet

Openface: https://github.com/cmusatyalab/openface

4. Futher works with face recognition

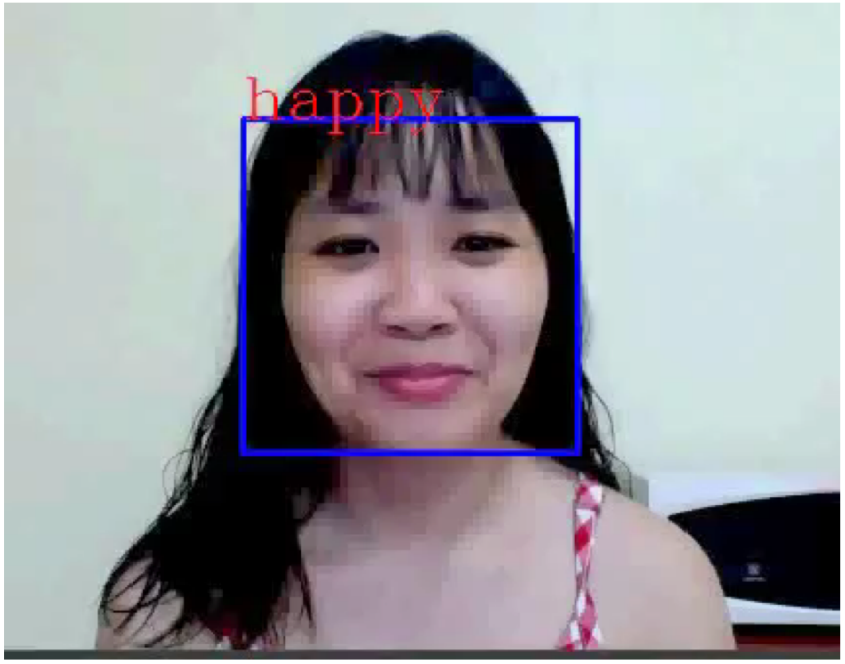
- Face tracking
- Facial Expression Recognition

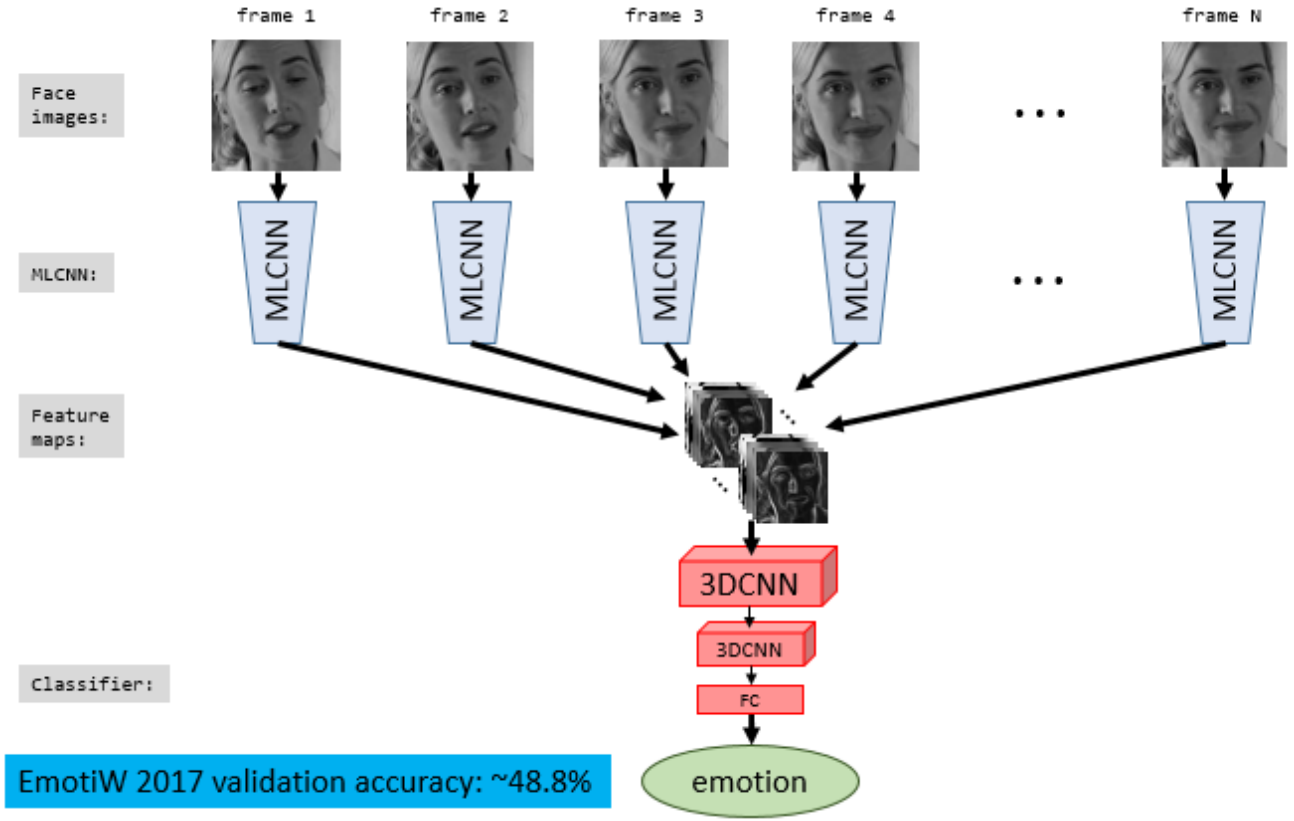
4.1 Face tracking



Facial Expression Recognition

Architecture: MLCNN + 3DCNN





EmotiW 2017 validation accuracy: ~48.8%

References

- http://aikorea.org/blog/rnn-tutorial-1/
- http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/
- http://ir.hit.edu.cn/~jguo/docs/notes/bptt.pdf
- http://minds.jacobs-university.de/sites/default/files/uploads/papers/ESNTutorialRev.pdf
- https://www.coursera.org/learn/ml-foundations/lecture/dZMX4/other-examples-of-deep-learning-in-computer-vision
- Deep Learning, Y. LeCun, Y. Bengio, G. Hinton, Nature 14539, May 2015
- CS231n: CNN for Visual Recognition, Stanford Univ., A. Karpathy, et al., 2016, http://cs231n.github.io/convolutional-networks/
- UFLDL Tutorial, Andrew Ng, et al., http://deeplearning.stanford.edu/tutorial/
- Convolutional Neural Networks, A. Gibiansky, 2014, http://andrew.gibiansky.com/blog/machine-learning/convolutional-neural-networks/
- Neural Networks and Deep Learning, M. Nielsen, 2016, http://neuralnetworksanddeeplearning.com/
- Principles of Backpropagation, M. Bernacki, et al., 2004, http://galaxy.agh.edu.pl/~vlsi/AI/backp_t_en/backprop.html

References

- https://pjreddie.com/darknet/yolo/
- https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78
- https://www.youtube.com/watch?v=PL3xJErjEgU

Thank you