(4th Supplementary Slides: Computer Vision)

Professor John Daugman

University of Cambridge

Computer Science Tripos, Part II
Lent Term 2015/16

Face Detection, Recognition, and Interpretation

Some variations in facial appearance (L.L. Boilly: *Réunion de Têtes Diverses)*

(Face Detection, Recognition, and Interpretation, con't)

Detecting faces and recognising their identity is a "Holy Grail" problem in computer vision. It is difficult for all the usual reasons:

- Faces are surfaces on 3D objects (heads), so facial images depend on pose and perspective angles, distance, and illumination
- Facial surfaces have relief, so some parts (e.g. noses) can occlude other parts. Hair can also create random occlusions and shadows
- Surface shape causes shading and shadows to depend upon the angle of the illuminant, and whether it is an extended or a point source
- Faces have variable specularity (dry skin may be Lambertian, whereas oily or sweaty skin may be specular). As always, this confounds the interpretation of the reflectance map
- Parts of faces can move around relative to other parts (eye or lip movements; eyebrows and winks). We have 7 pairs of facial muscles. People use their faces as communicative organs of expression
- People put things on their faces (e.g. glasses, cosmetics, cigarettes), change their facial hair (moustaches, eyebrows), and age over time

(Face Detection, Recognition, and Interpretation, con't)

Classic problem: within-class variation (same person, different conditions) can exceed the between-class variation (different persons).

These are different persons, in genetically identical (monozygotic) pairs:

(Face Detection, Recognition, and Interpretation, con't)

Classic problem: within-class variation (same person, different conditions) can exceed the between-class variation (different persons).

Persons who share 50% of their genes (parents and children; full siblings; double cousins) sometimes look almost identical (apart from age cues):

(Face Detection, Recognition, and Interpretation, con't)

Classic problem: within-class variation (same person, different conditions) can exceed the between-class variation (different persons).

...and these are *completely unrelated people, in Doppelgänger pairs:*









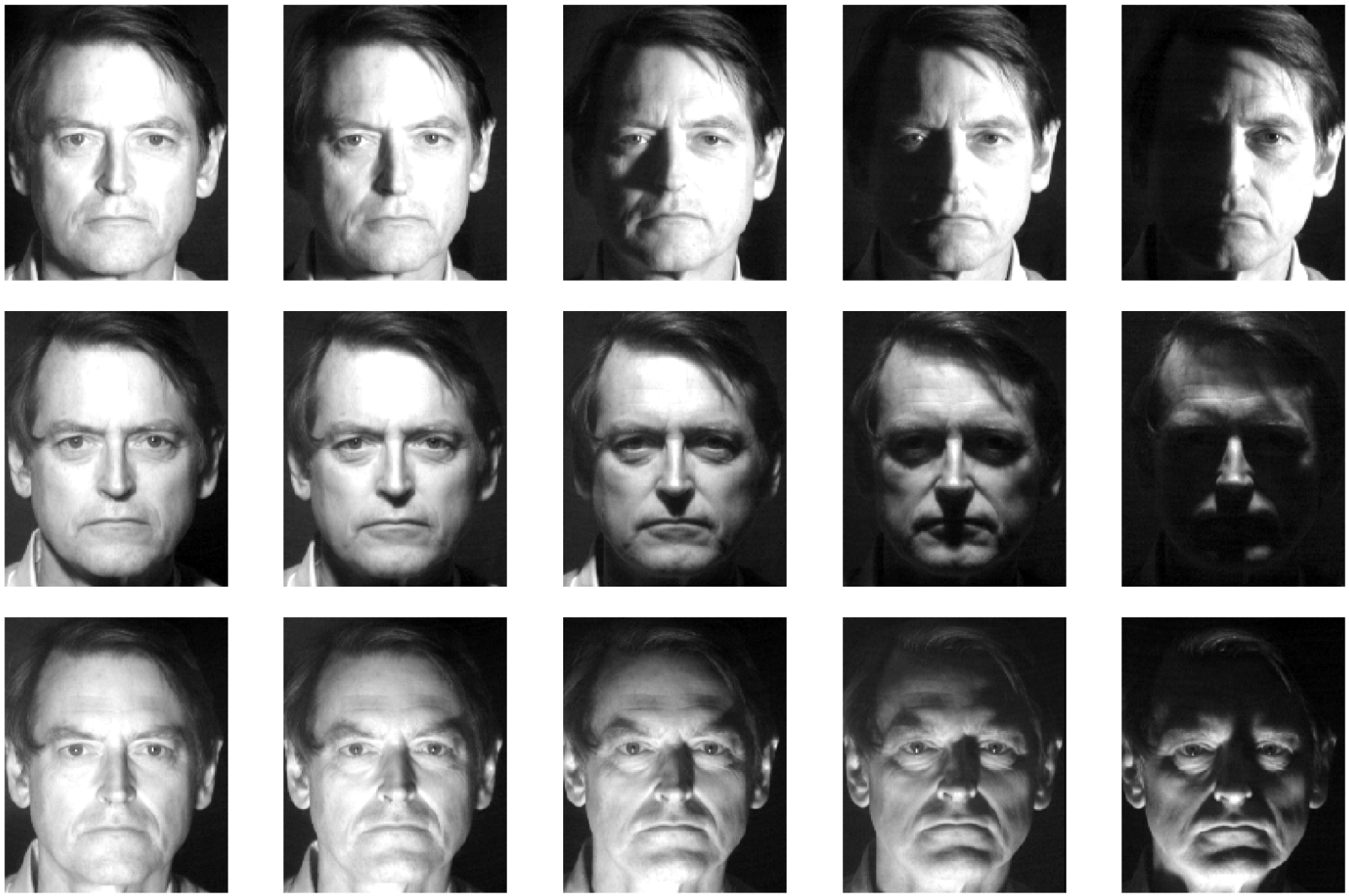




(Face Detection, Recognition, and Interpretation, con't)

Classic problem: within-class variation (same person, different conditions) can exceed the between-class variation (different persons).

Same person, fixed pose and expression; varying illumination geometry:



(Face Detection, Recognition, and Interpretation, con't)

Classic problem: within-class variation (same person, different conditions) can exceed the between-class variation (different persons).

Effect of variations in pose angle (easy and hard), and distance:

(Face Detection, Recognition, and Interpretation, con't)

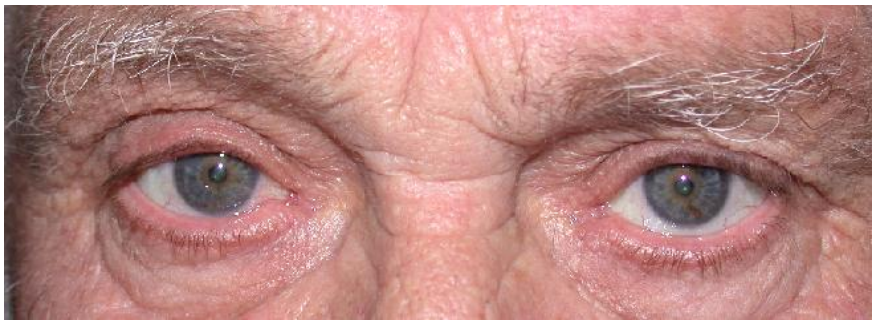
Classic problem: within-class variation (same person, different conditions) can exceed the between-class variation (different persons).

Changes in appearance over time (sometimes artificial and deliberate)

Paradox of Facial Phenotype and Genotype

Facial appearance (phenotype) of everyone changes over time with age; but monozygotic twins (identical genotype) track each other as they age.

Therefore at any given point in time, they look more like each other than they look like themselves at either earlier or later periods in time

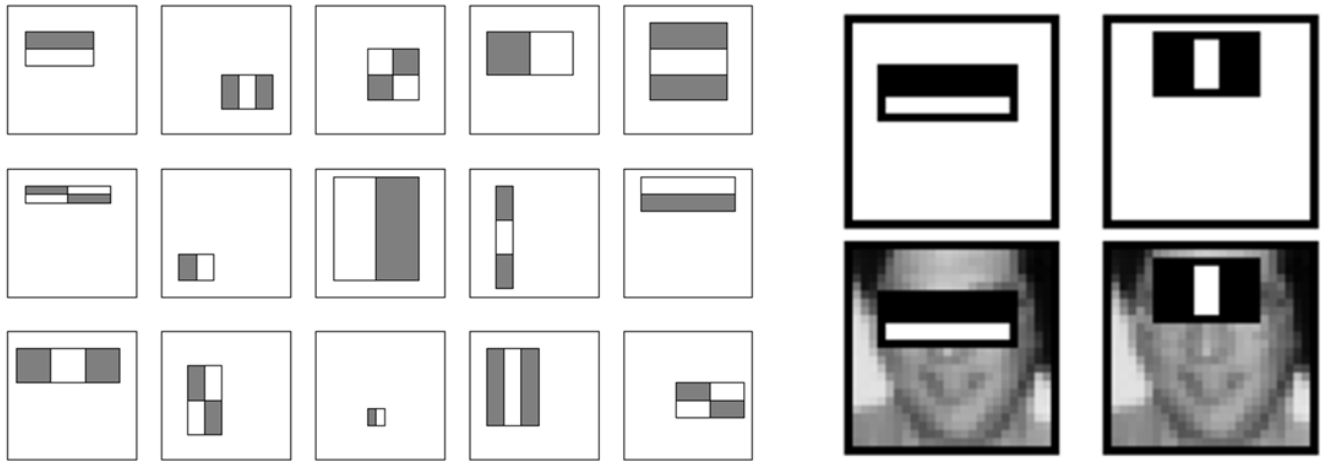
(Face Detection, Recognition, and Interpretation, con't)

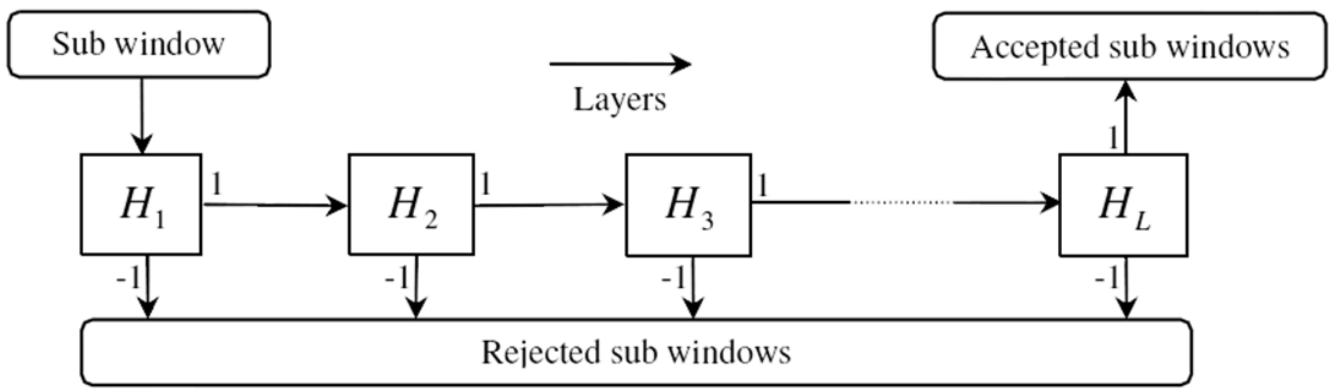
Detecting and recognising faces raises all the usual questions encountered in other domains of computer vision:

- What is the best representation to use for faces?
- Should this be treated as a 3D problem (object-based, volumetric), or a 2D problem (image appearance-based)?
- How can invariances to size (hence distance), location, pose, and illumination be achieved? (A given face should acquire a similar representation under such transformations, for matching purposes.)
- What are the generic (i.e. universal) properties of all faces that we can rely upon, in order to reliably detect the presence of a face?
- What are the particular features that we can rely upon to distinguish among faces, and thus determine the identity of a given face?
- What is the best way to handle "integration of evidence", and incomplete information, and to make decisions under uncertainty?
- How can machine learning develop domain expertise, either about faces in general (e.g. pose transformations), or facial distinctions?

Viola-Jones Face Detection Algorithm

Paradoxically, face detection is a harder problem than recognition, and performance rates of algorithms are poorer. (It seems paradoxical since detection precedes recognition; but recognition performance is measured only with images already containing faces.) The best known way to find faces is the cascade of classifiers developed by Viola and Jones (2004).





(Viola-Jones Face Detection Algorithm, con't)

Key idea: build a strong classifier from a cascade of many weak classifiers – all of whom in succession must agree on the presence of a face

- A face (in frontal view) is presumed to have structures that should trigger various local "on-off" or "on-off-on" feature detectors
- A good choice for such feature detectors are 2D Haar wavelets (simple rectangular binary alternating patterns)
- There may be 2, 3, or 4 rectangular regions (each $+1$ or -1) forming feature detectors f_j, at differing scales, positions, and orientations
- Applying Haar wavelets to a local image region only involves adding and subtracting pixel values (no multiplications; hence very fast)
- A given weak classifier $h_j(x)$ consists of a feature f_j, a threshold θ_j and a polarity $p_j \in \pm 1$ (all determined in training) such that

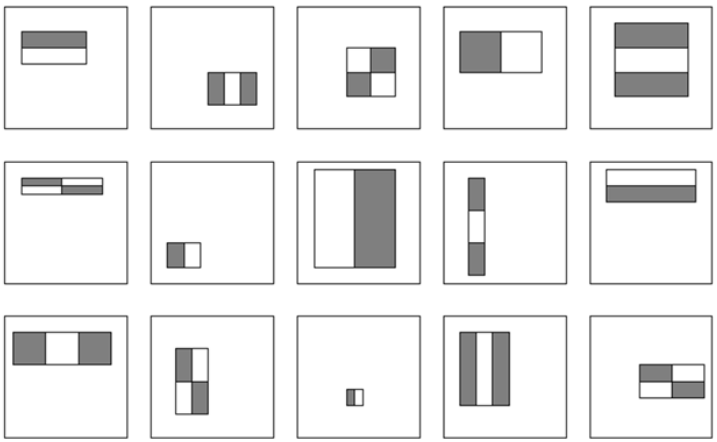
$$h_j(x) = \begin{cases} -p_j & \text{if } f_j < \theta_j \\ p_j & \text{otherwise} \end{cases}$$

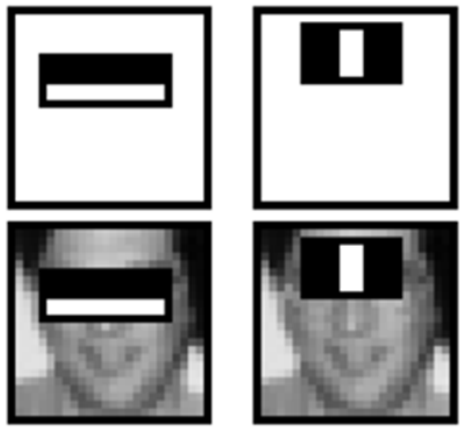
- A strong classifier $h(x)$ takes a linear combination of weak classifiers, using weights α_j learned in a training phase, and considers its sign:

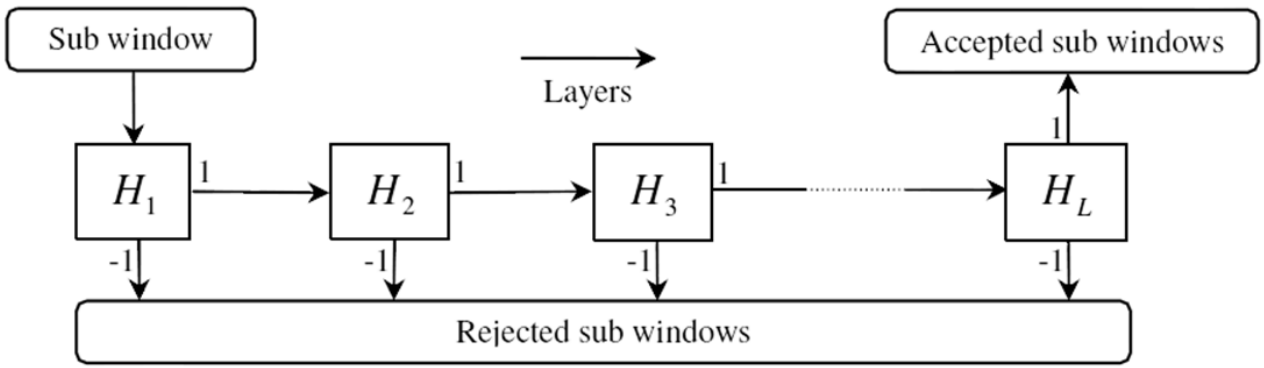
$$h(x) = \text{sign}(\sum_j \alpha_j h_j)$$

(Viola-Jones Face Detection Algorithm, con't)

- At a given level of the cascade, a face is "provisionally deemed to have been detected" at a certain position if $h(x) > 0$
- Only those image regions accepted by a given layer of the cascade ($h(x) > 0$) are passed on to the next layer for further consideration
- A face detection cascade may have 30+ layers, yet the vast majority of candidate image regions will be rejected early in the cascade.







(Viola-Jones Face Detection Algorithm, con't)

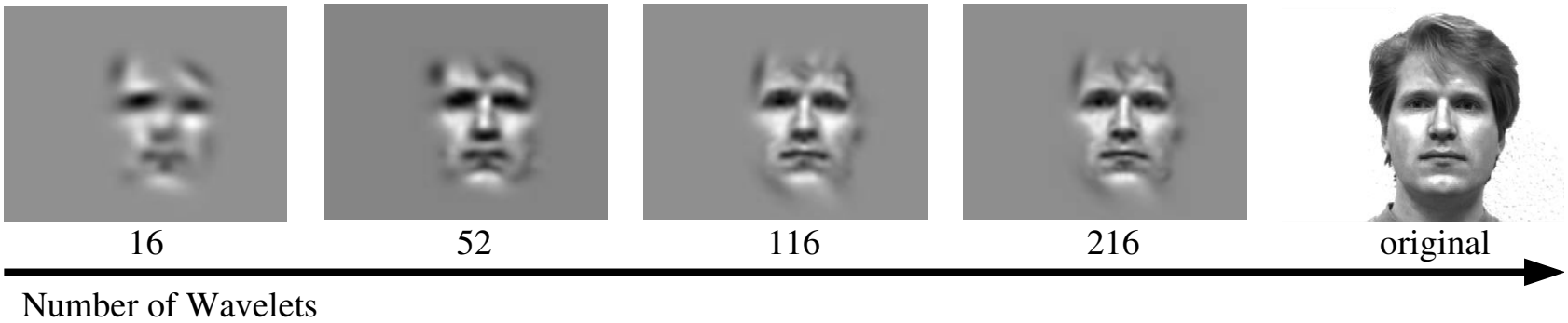
- Training uses the AdaBoost ("Adaptive Boosting") algorithm
- This supervised machine learning process adapts the weights α_j such that early cascade layers have very high true accept rates, say 99.8% (as *all* must detect a face; hence high false positive rates, say 68%)
- Later stages in the cascade, increasingly complex, are trained to be more discriminating and therefore have lower false positive rates
- More and more 2D Haar wavelet feature detectors are added to each layer and trained, until performance targets are met
- The cascade is evaluated at different scales and offsets across an image using a sliding window approach, to find any (frontal) faces
- With "true detection" probability d_i in the i^{th} layer of an N-layer cascade, the overall correct detection rate is: $D = \prod_{i=1}^{N} d_i$
- With "erroneous detection" probability e_i at the i^{th} layer, the overall false positive rate is $E = \prod_{i=1}^{N} e_i$ (as every layer must falsely detect)
- Example: if we want no false detections, with 10^5 image subregions so $E < 10^{-5}$, in a 30-layer cascade we train for $e_i = 10^{-5/30} \approx 0.68$ which shows why each layer can use such weak classifiers!
- Likewise, to achieve a decent overall detection rate of $D = 0.95$ requires $d_i = 0.95^{1/30} \approx .9983$ (very happy to call things "faces")

(Viola-Jones Face Detection Algorithm, con't)

Performance on a local group photograph:

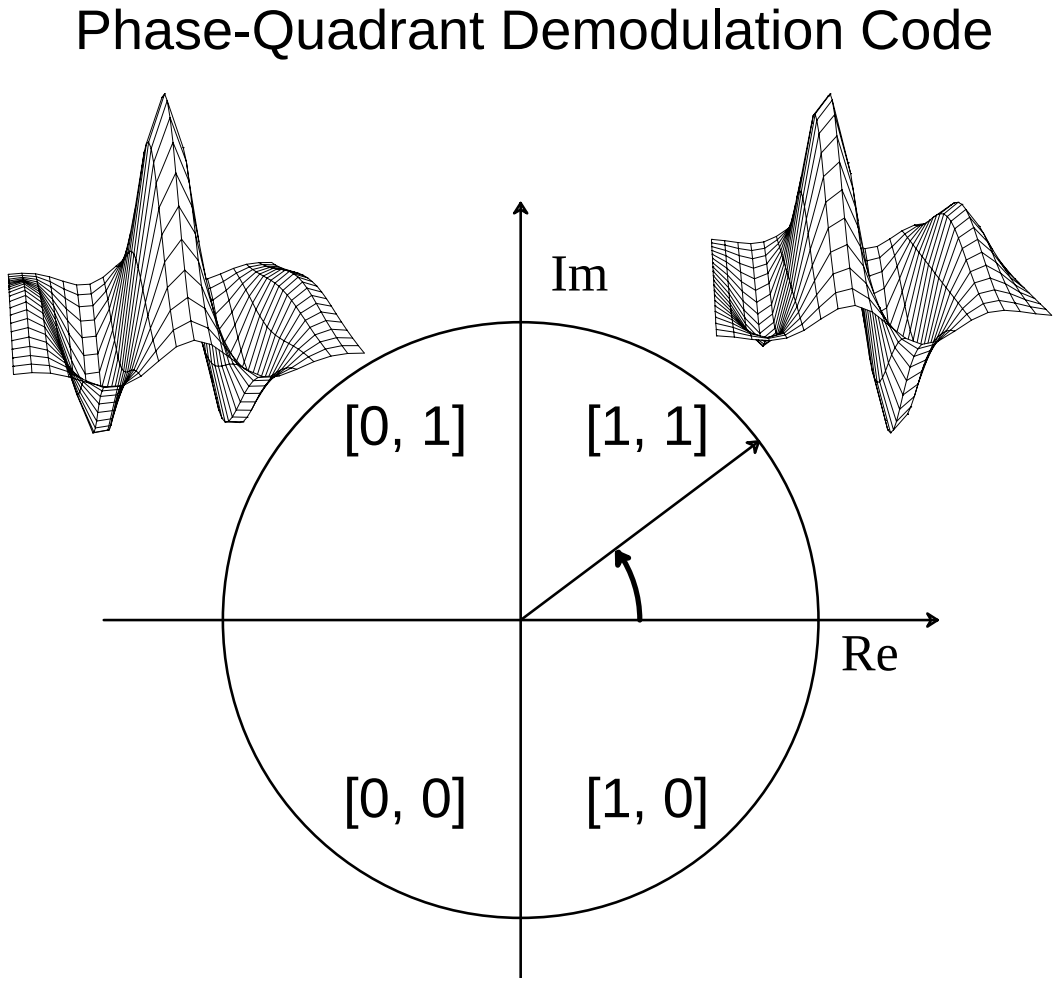
2D Appearance-based Face Recognition: Gabor Wavelets

We saw that 2D Gabor wavelets can make remarkably compact codes for faces, among many other things. In this sequence, even using only about 100 Gabor wavelets, not only the presence of a face is obvious, but also its gender, rough age, pose, expression, and perhaps even identity:



- Gabor wavelets capture image structure as combined undulations
- Parameterisation: 2D positions, sizes, orientations, and phases
- Facial features like eyes, lips, and noses are represented with just a handful of wavelets, without requiring explicit *models* for such parts
- Can track changes of expression locally. Example: gaze = phase
- A deformable elastic graph made from such an encoding can preserve matching, while tolerating *some* changes in pose and expression

(2D Appearance-based Face Recognition: Gabor Wavelets)



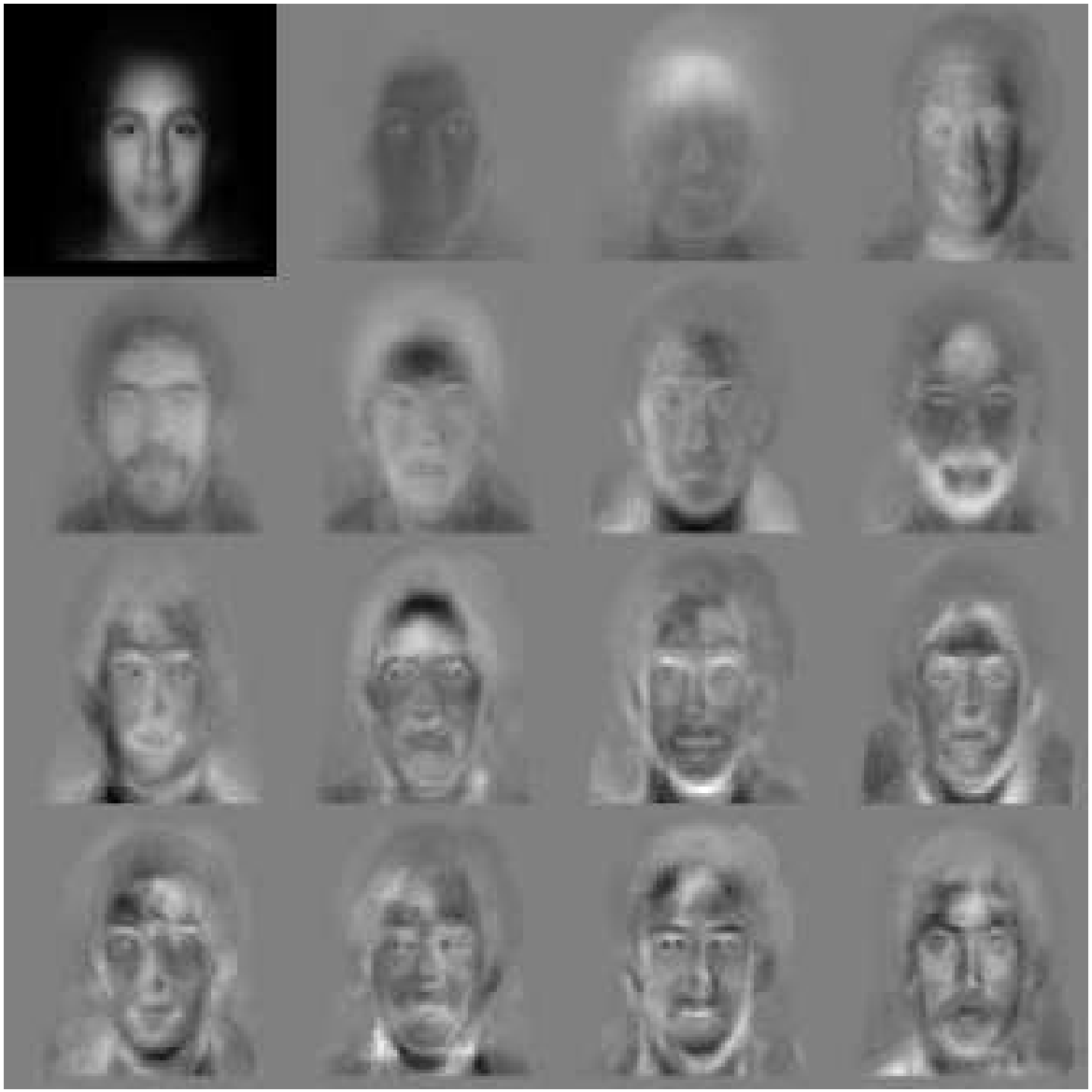
Computed feature vectors in a face code can be local 2D Gabor wavelet amplitude or phase information. Bits in the "face code" are set by the quadrant in which the phasor lies, for each aspect of facial structure.

2D Appearance-based Face Recognition: "Eigenfaces"

An elegant method for 2D appearance-based face recognition combines Principal Components Analysis (PCA) with machine learning and algebra, to compute a linear basis (like the Fourier basis) for representing any face as a combination of empirical eigenfunctions, called eigenfaces.

- A database of face images (at least 10,000) that are pre-normalised for size, position, and frontal pose is "decomposed" into its Principal Components of statistical variation, as a sequence of orthonormal eigenfunctions whose eigenvalues are in descending order
- This is a classical framework of linear algebra, associated also with the names Karhunen-Loève Transform, or the Hotelling Transform, or Dimensionality Reduction and subspace projection
- Optimised for truncation: finding the best possible (most accurate) representation of data using any specified finite number of terms
- Having extracted from a face gallery the (say) 20 most important eigenfaces of variation (in sequence of descending significance), any given presenting face is projected onto these, by inner product
- The resulting (say) 20 coefficients then constitute a very compact code for representing, and recognising, the presenting face
- 15 such representational eigenfaces are shown in the next slide

(2D Appearance-based Face Recognition: "Eigenfaces")



The top left face is a particular linear combination of the eigenfaces

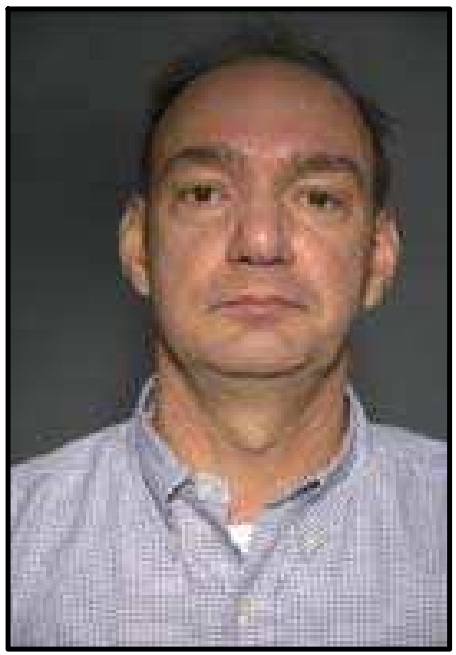
(2D Appearance-based Face Recognition: "Eigenfaces")

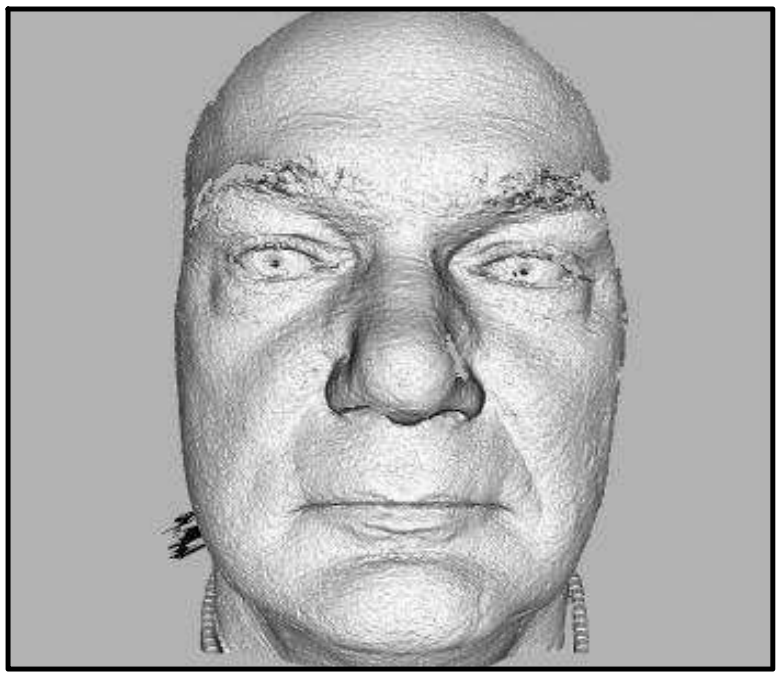
- Performance is often in the range of 90% to 95% accuracy
- Databases can be searched very rapidly, as each face is represented by a very compact feature vector of only about 20 numbers
- A major limitation is that significant (early, low-order) eigenfaces emerging from the statistical analysis arise just from normalisation errors of size (head outlines), or variations in illumination angle
- Like other 2D representations for faces, the desired invariances for transformations of size (distance), illumination, and pose are lacking
- Both the Viola-Jones face detection algorithm, and these 2D appearance-based face recognition algorithms, sometimes deploy "brute force" solutions (say at airport Passport control) such as acquiring images from a large (3×3) or (4×4) array of cameras for different pose angles, each allowing some range of angles

Three-Dimensional Approaches to Face Recognition

Face recognition algorithms now aim to model faces as three-dimensional objects, even as dynamic objects, in order to achieve invariances for pose, size (distance), and illumination geometry. Performing face recognition in object-based (volumetric) terms, rather than appearance-based terms, unites vision with model-building and graphics.

To construct a 3D representation of a face, it is necessary to extract both a shape model (below right), and a texture model (below left). The term "texture" here encompasses albedo, colouration, and 2D surface details.





(Three-Dimensional Approaches to Face Recognition)

Extracting the 3D shape model can be done by various means:

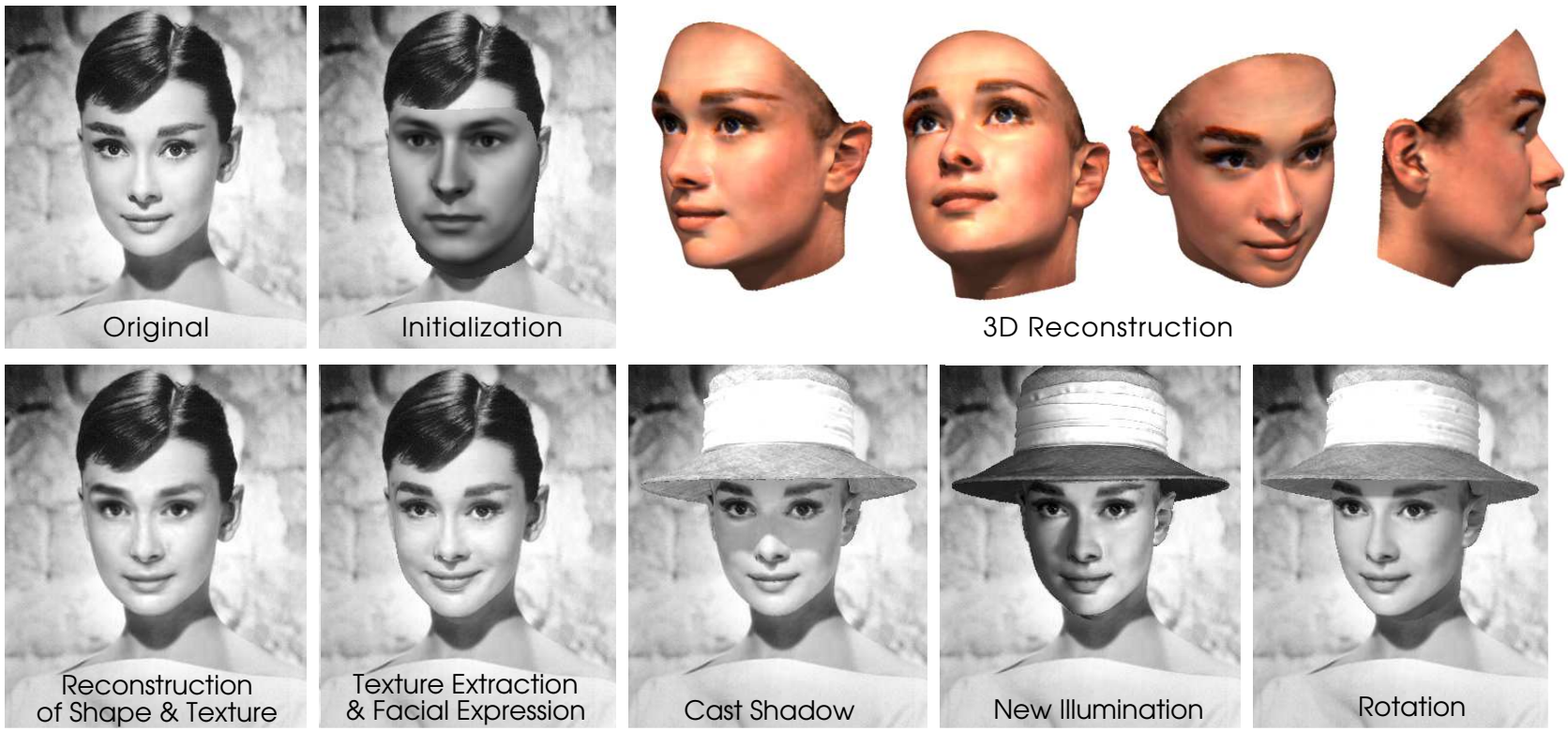
- laser range-finding, even down to millimetre resolution
- calibrated stereo cameras
- projection of structured IR light (grid patterns whose distortions reveal shape, as with Kinect)
- extrapolation from multiple images taken from different angles

The size of the resulting 3D data structure can be in the gigabyte range, and significant time can be required for the computation.

Since the texture model is linked to coordinates on the shape model, it is possible to "project" the texture (tone, colour, features) onto the shape, and thereby to generate predictive models of the face in different poses.

Clearly sensors play an important role here for extracting shape models, but it is also possible to do this even from just a single photograph if sufficiently strong Bayesian priors are also marshalled, assuming an illumination geometry and some universal aspects of head and face shape.

(Three-Dimensional Approaches to Face Recognition)



An impressive demo of using a single 2D photograph (top left) to morph a 3D face model after manual initialisation, building a 3D representation of the face that can be manipulated for differing pose angles, illumination geometries, and even expressions, can be seen here:

`http://www.youtube.com/watch?v=nice6NYb_WA`

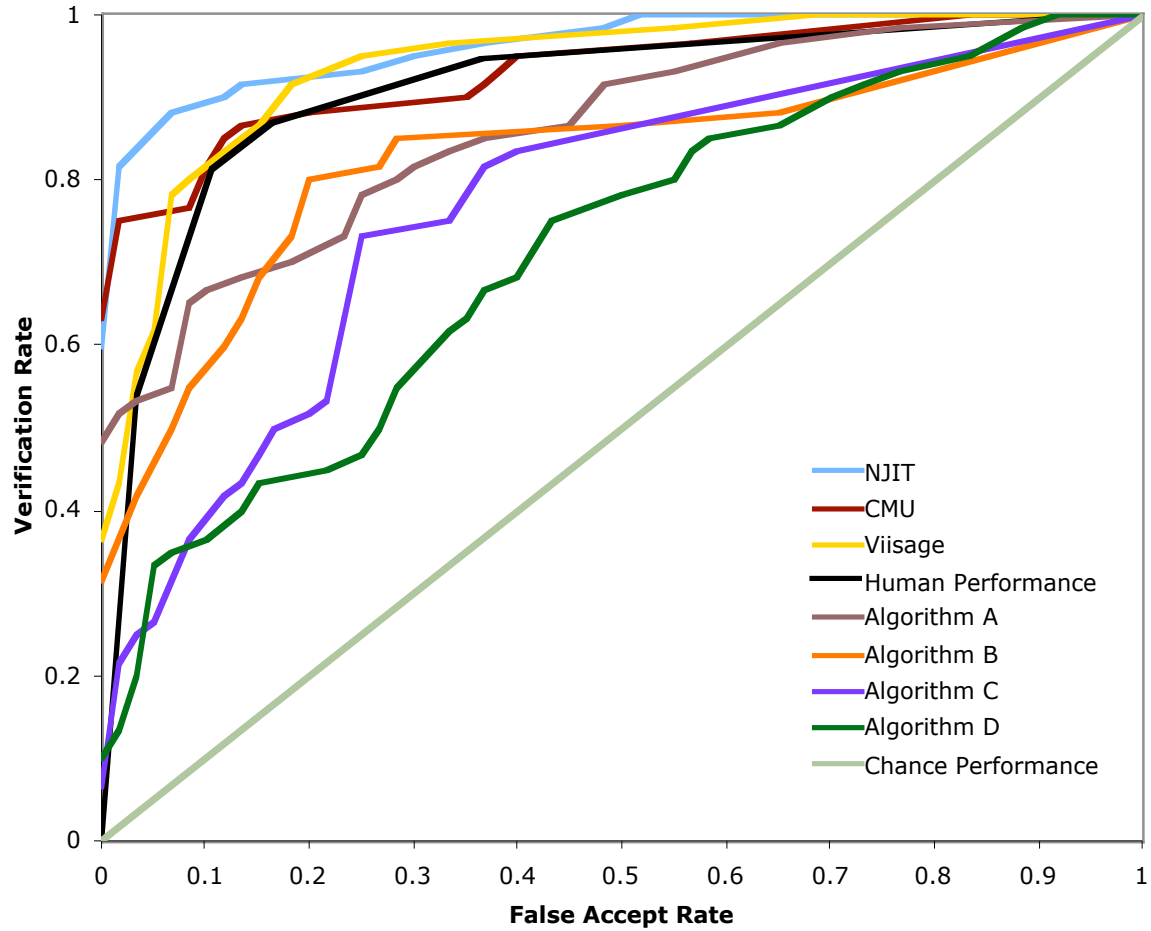
(Three-Dimensional Approaches to Face Recognition)

Description from the Blanz and Vetter paper,
Face Recognition Based on Fitting a 3D Morphable Model:

"...a method for face recognition across variations in pose, ranging from frontal to profile views, and across a wide range of illuminations, including cast shadows and specular reflections. To account for these variations, the algorithm simulates the process of image formation in 3D space, using computer graphics, and it estimates 3D shape and texture of faces from single images. The estimate is achieved by fitting a statistical, morphable model of 3D faces to images. The model is learned from a set of textured 3D scans of heads. Faces are represented by model parameters for 3D shape and texture."

Face Algorithms Compared with Human Performance

The US National Institute for Standards and Technology (NIST) runs periodic competitions for face recognition algorithms, over a wide range of conditions. Uncontrolled illumination and pose remain challenging. But in a 2007 test, three algorithms had ROC curves above (better than) human performance at non-familiar face recognition (the black curve):



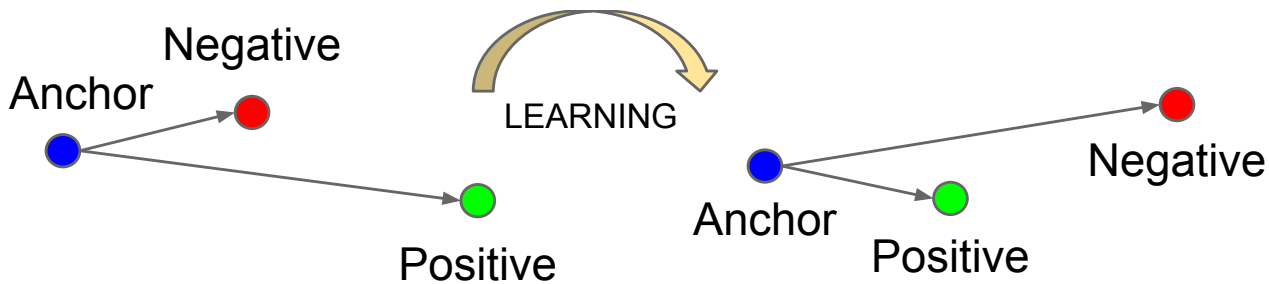
Major Breakthrough in 2015: Deep-Learning "FaceNet"

Machine learning approaches focused on scale ("Big Data") are having a profound impact in Computer Vision. In 2015 Google demonstrated large reductions in face recognition error rates (by 30%) on two very difficult databases: YouTube Faces (95%), and Labeled Faces in the Wild (LFW) database (99.63%), which are new accuracy records.

(Major Breakthrough in 2015: Deep-Learning "FaceNet")

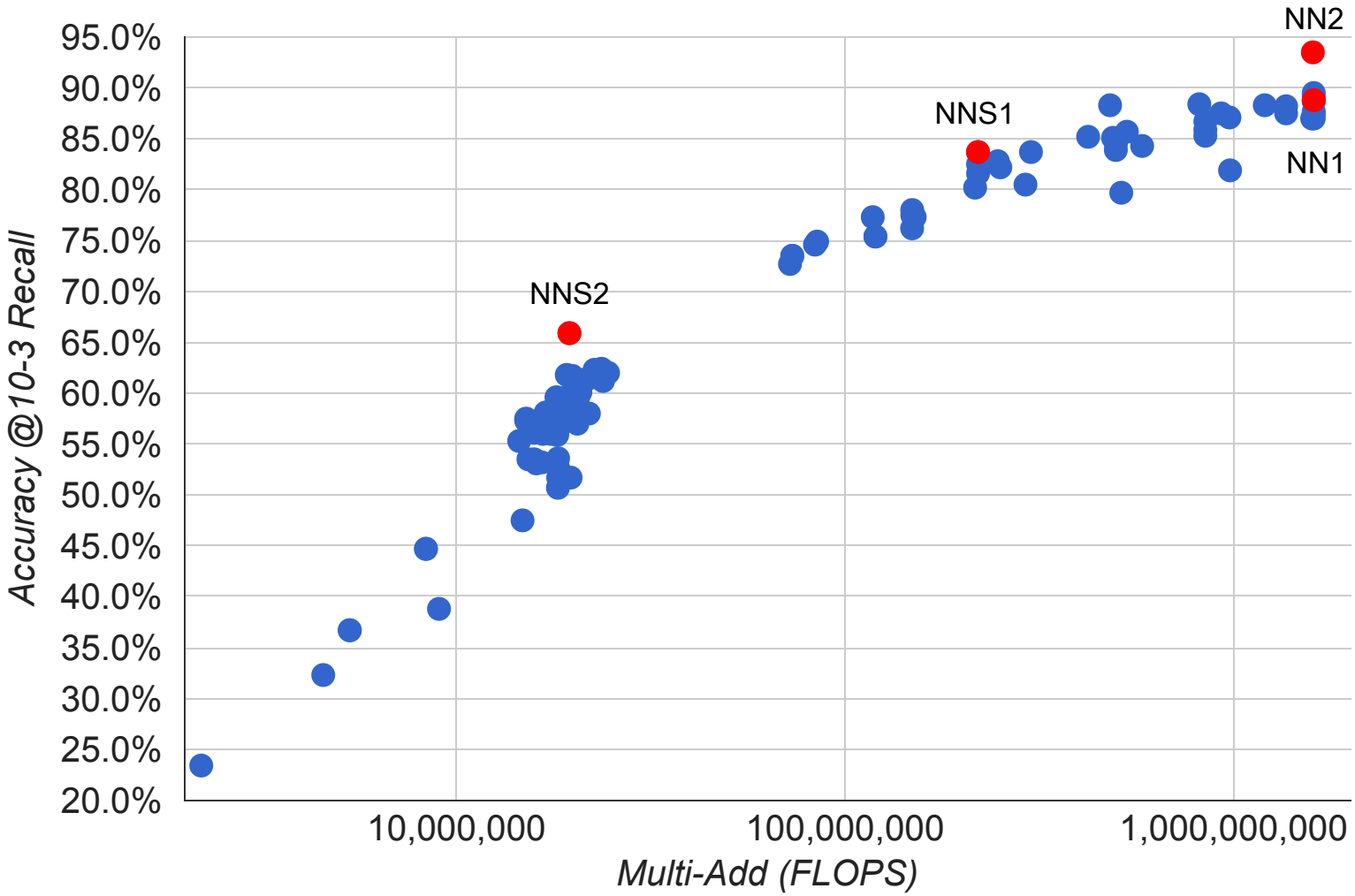
- Convolutional Neural Net with 22 layers and 140 million parameters
- Big dataset: trained on 200 million face images, 8 million identities
- 2,000 hours training (clusters); about 1.6 billion FLOPS per image
- Euclidean distance metric (L2 norm) on embeddings $f(x_i)$ learned for cropped, but not pre-segmented, images x_i using back-propagation
- Used triplets of images, one pair being from the same person, so that both the **p**ositive (same face) and **n**egative (different person) features were learned by minimising a loss function L:

$$L = \sum_i \left[\| f(x_i^a) - f(x_i^p) \|^2 - \| f(x_i^a) - f(x_i^n) \|^2 \right]$$



- The embeddings create a compact (128 byte) code for each face
- Simple threshold on Euclidean distances among these embeddings then gives decisions of "same" vs "different" person

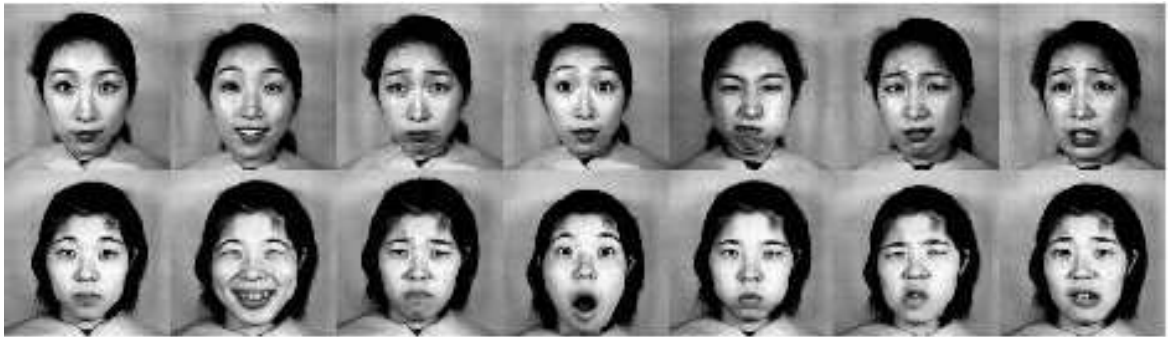
(Major Breakthrough in 2015: Deep-Learning "FaceNet")



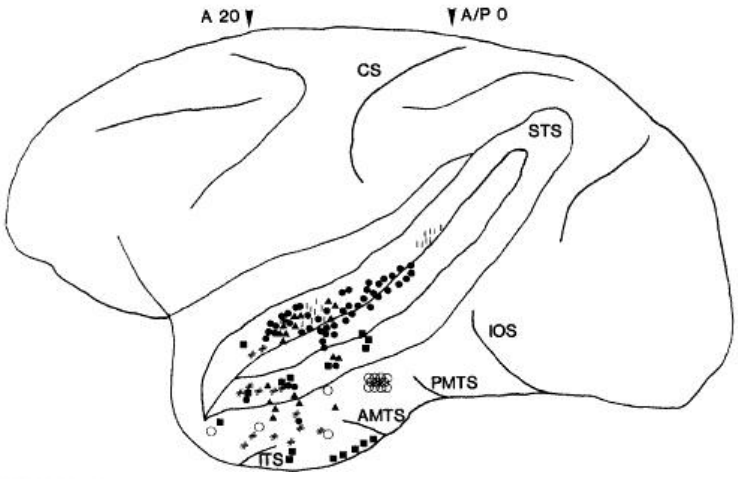
Different variants of the Convolutional Neural Net and model sizes were generated and run, revealing the trade-off between FLOPS and accuracy for a particular point on the ROC curve (False Accept Rate = 0.001)

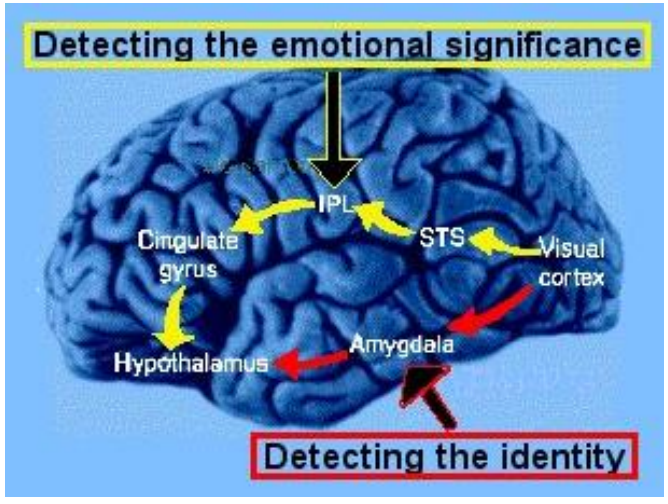
Affective Computing: Interpreting Facial Emotion

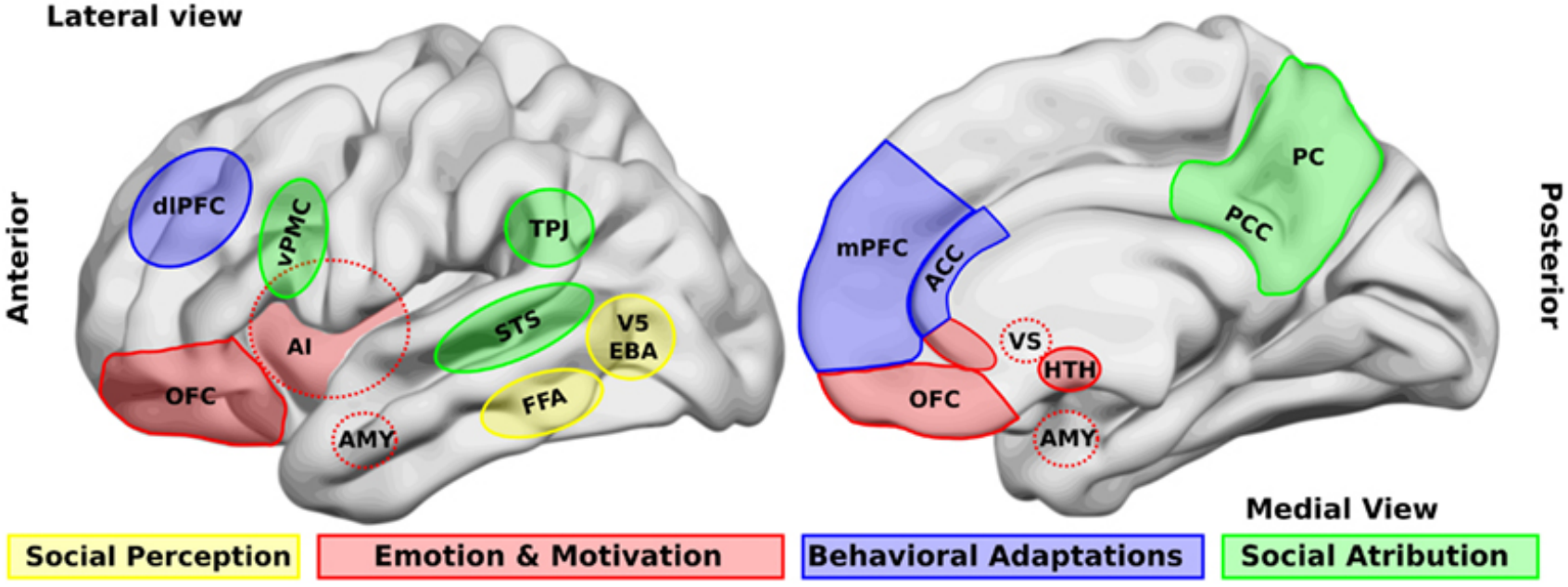
Humans use their faces as visually expressive organs, cross-culturally



Many areas of the human brain are concerned with recognising and interpreting faces, and social computation is believed to have been the primary computational load in the evolution of our brains, because of its role in reproductive success



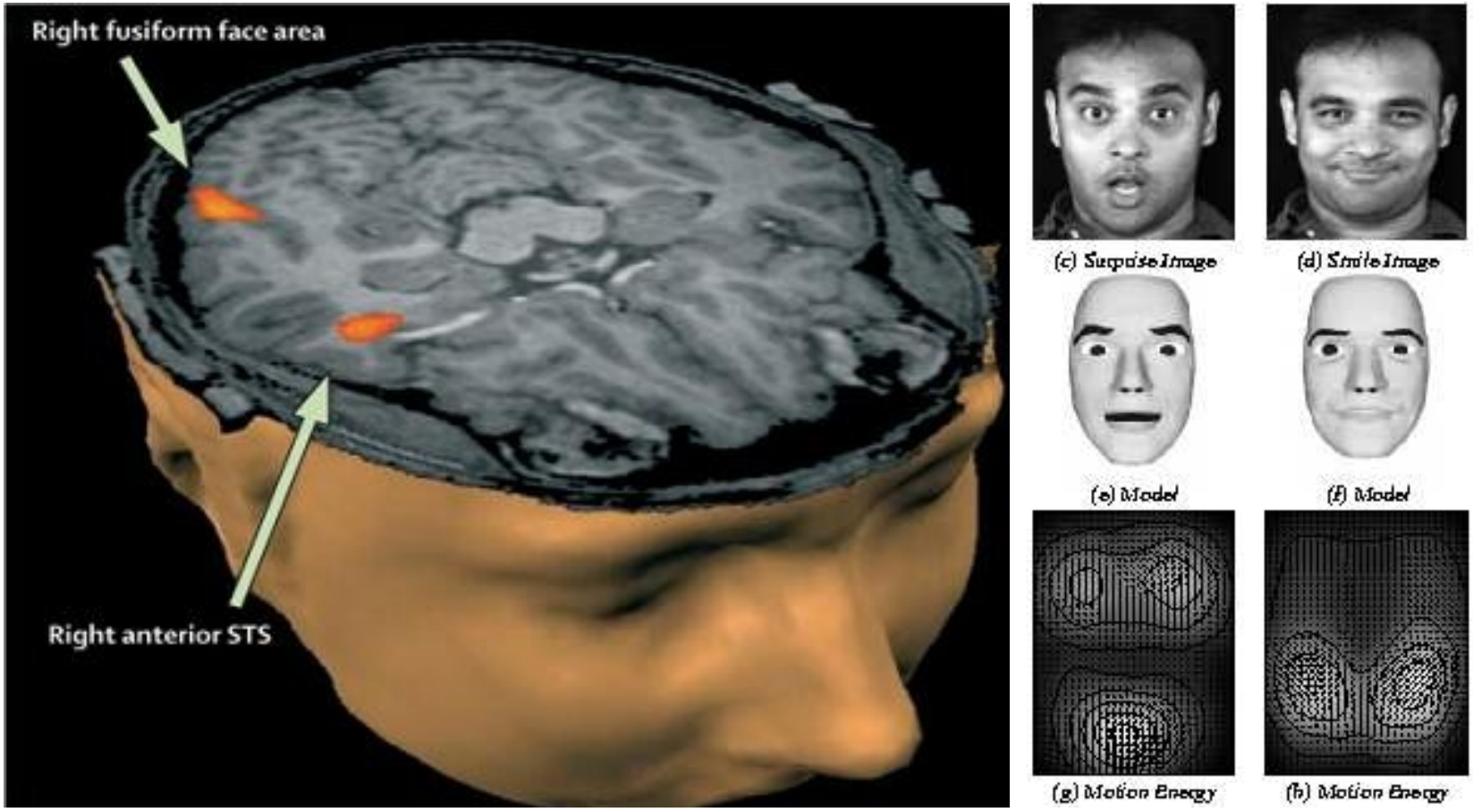




Affective Computing: Classifying Identity *and* Emotion

- Target stimulus
- Same identity / different emotion
- Same emotion / different identity
- Different identity / different emotion

(Affective Computing: Interpreting Facial Emotion)



MRI scanning has revealed much about brain areas that interpret facial expressions. Affective computing aims to classify visual emotions as articulated sequences using Hidden Markov Models of their generation. Mapping the visible data to action sequences of the facial musculature becomes a generative classifier of emotions.