Active Appearance Models applied to Human Faces

VisHCI 2007, Adelaide – Part 2

David Cristinacce
Tim Cootes, Chris Taylor, Ian Scott, Carole Twining,
Vladimir Petrović, John Kang, Gavin Wheeler, Gareth Edwards

Imaging Science and Biomedical Engineering,
The University of Manchester, UK

Tutorial Part 2

- Constrained Local Model (CLM)
- Demo of Face Tracking System
- Components of Demo
- Automatic Model Building
- Future Work

Constrained Local Model

- Similar building method to Active Appearance Model (AAM)
- Different Search Algorithm
  - Relies on feature detection
  - **Not** difference between model+ underlying image

MANCHESTER 1824

Constrained Local Model (CLM)

- Mode 1

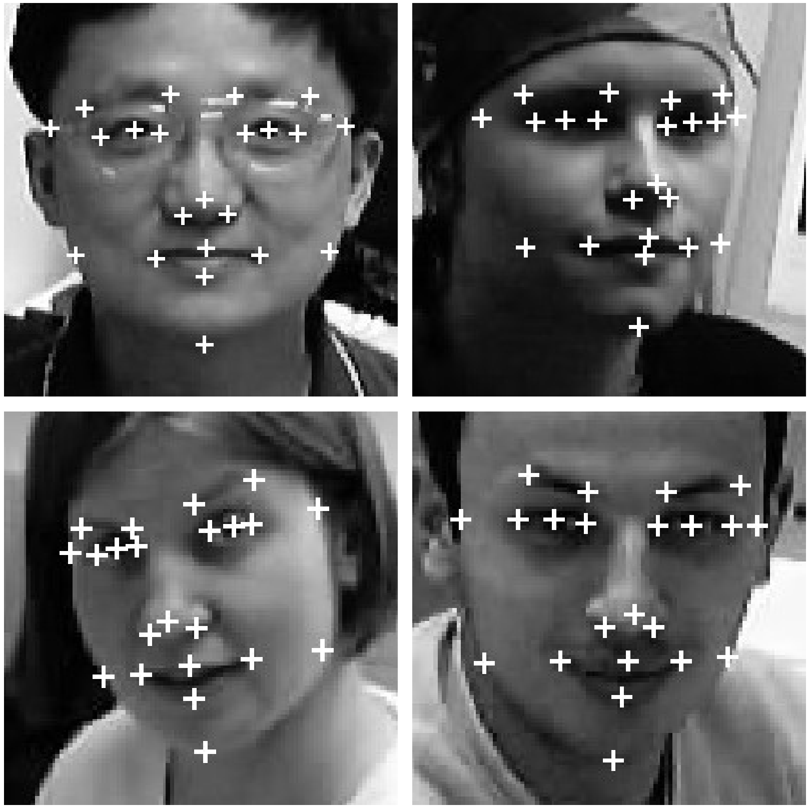
$-3\sqrt{\lambda_1}$ ← b_1 → $3\sqrt{\lambda_1}$

- Mode 2

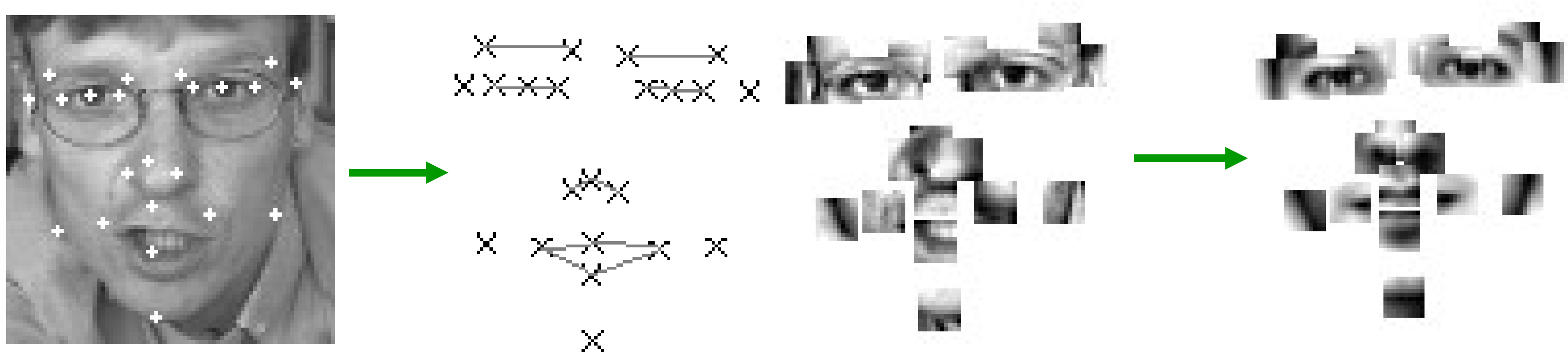
$-3\sqrt{\lambda_2}$ ← b_2 → $3\sqrt{\lambda_2}$

CLM Training

- 1000 manually labelled faces
- Build statistical shape model of point variation
- Sample regions around 22 landmark points
- Build appearance model of texture variation



CLM Template Generation



Current Points

Current Points and Texture

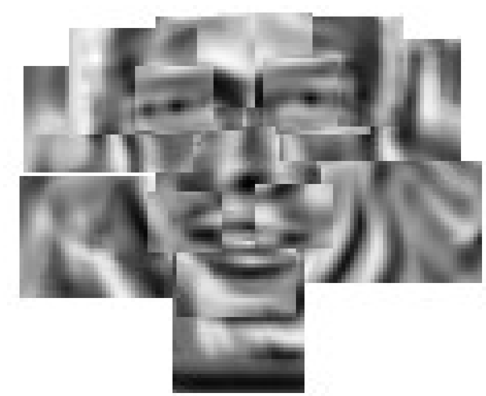
Generated Templates

- Generate new templates by fitting appearance model to current points.

CLM Shape Constrained Search 1

- Compute response surface for each feature detector (using normalised correlation)

&



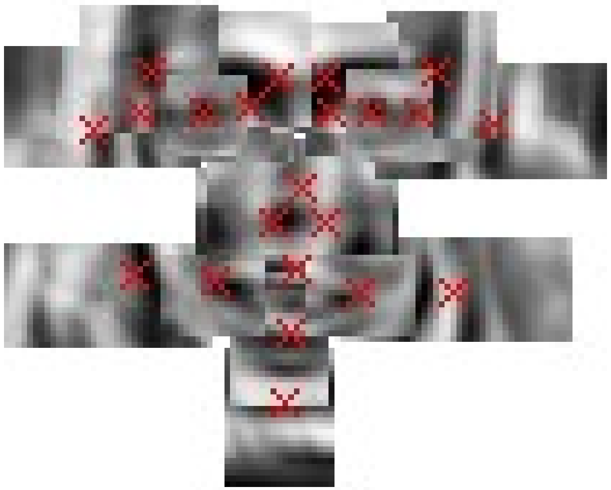
Current Templates Target Image Response Surfaces

Nb Dark regions => Higher Correlation response

CLM Shape Constrained Search 2

- Optimise sum of response images subject to objective function:-

$$f(p)=\sum_{i=1}^{n} I_i(x_i,y_i)+K\sum_{j=1}^{s}\frac{-b_j^2}{\lambda_j}$$



b_j are shape model parameters

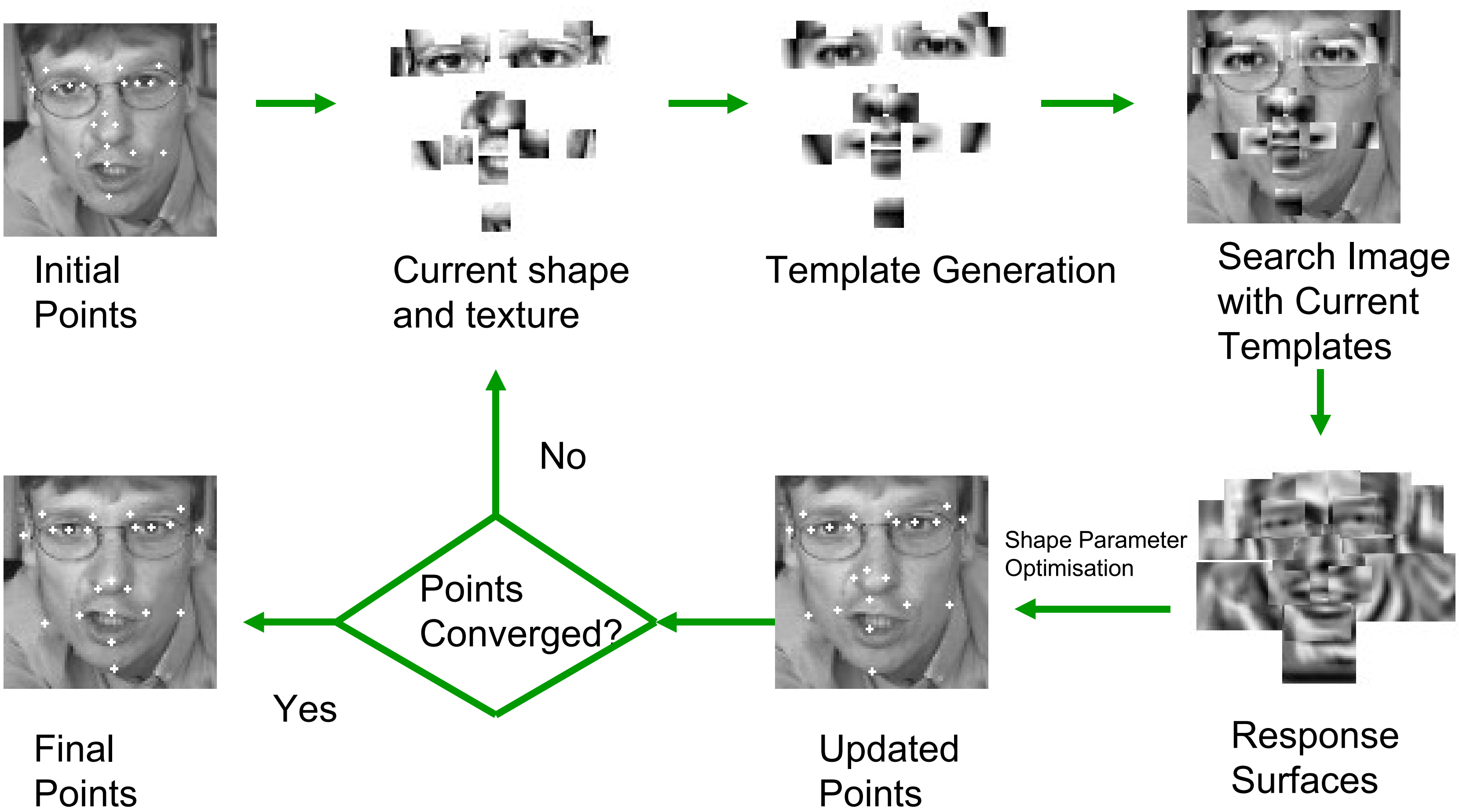
K is a shape and texture weight

p is a combined set of shape model and transform parameters

λ_j are the eigenvalues of the shape model

$I_i(x_i, y_i)$ is the location in the i^{th} response image

CLM Search Algorithm



CLM – Example Search

Start | After Iteration 1 | After Iteration 2 | Final Points

Constrained Local Model - Overview

- CLM generates likely feature templates
- Compute whole response surface for each detector
- Uses non-linear optimisation, subject to shape model constraints
- Iterative method

Ref: Cristinacce & Cootes – Detection and tracking with constrained local models –British Machine Vision Conference 2006

Displacement Experiments CLM vs AAM

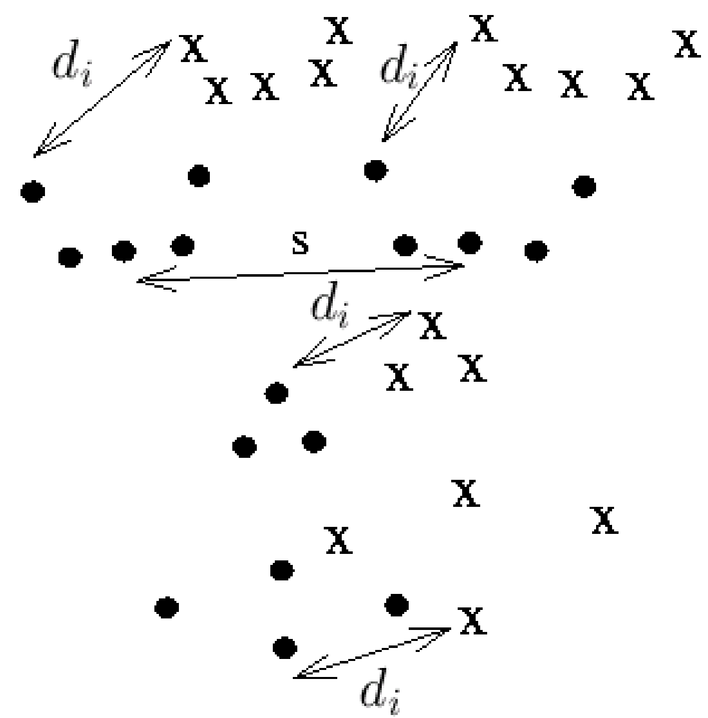
- Place model at correct points
- Apply displacement from correct points(8 possible directions)
- Set to mean shape
- Search with model to improve point locations

Point to Point Error Measure

● = Manually labelled points

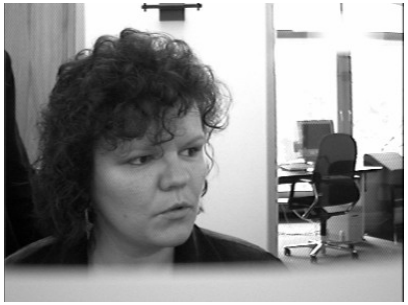
X = Automatically predicted points

$$error = \frac{1}{n * s} \sum_{i=1}^{n} d_i$$

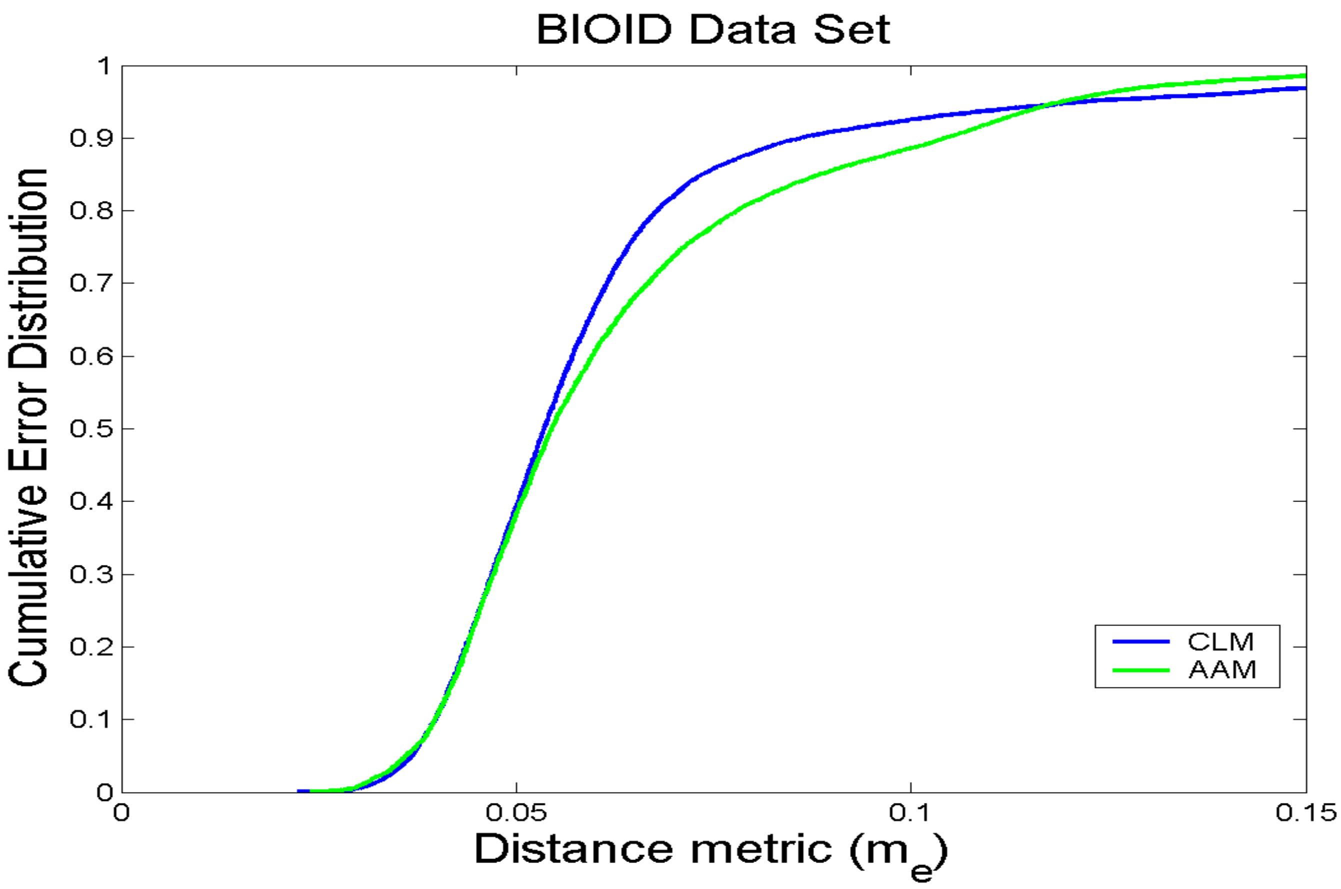


Faces Test Set - BIOID

- 1522 images
- 22 points human labelled ground truth
- Independent from training set

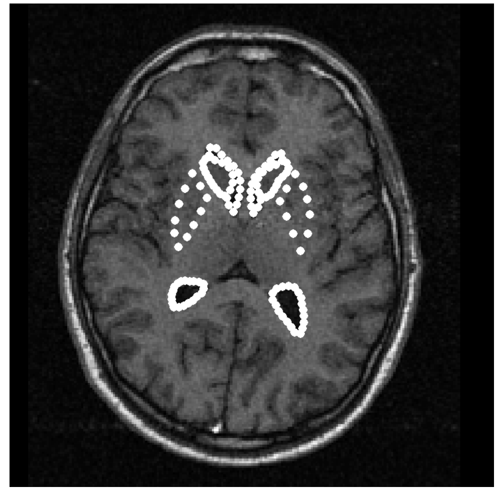
Faces Displacement Results

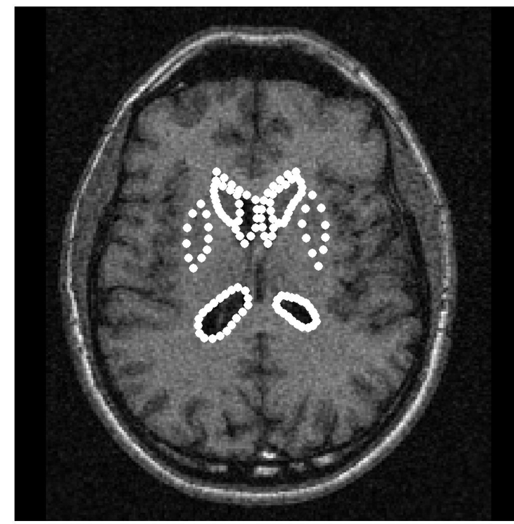


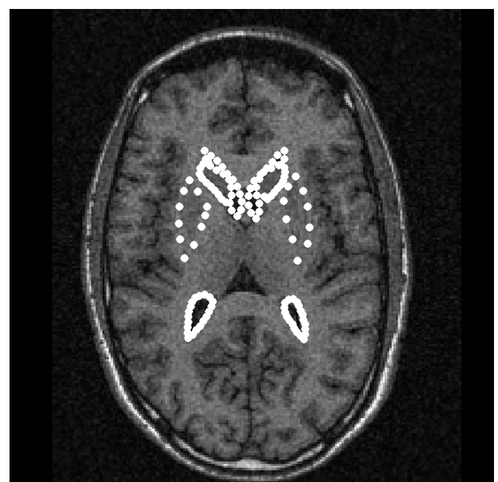
Initial displacement =0.15

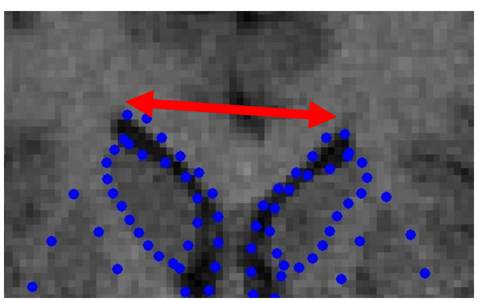
Example Brain Images

- 123 points
- 69 MRI Brain Slices
  - 35 training
  - 34 test







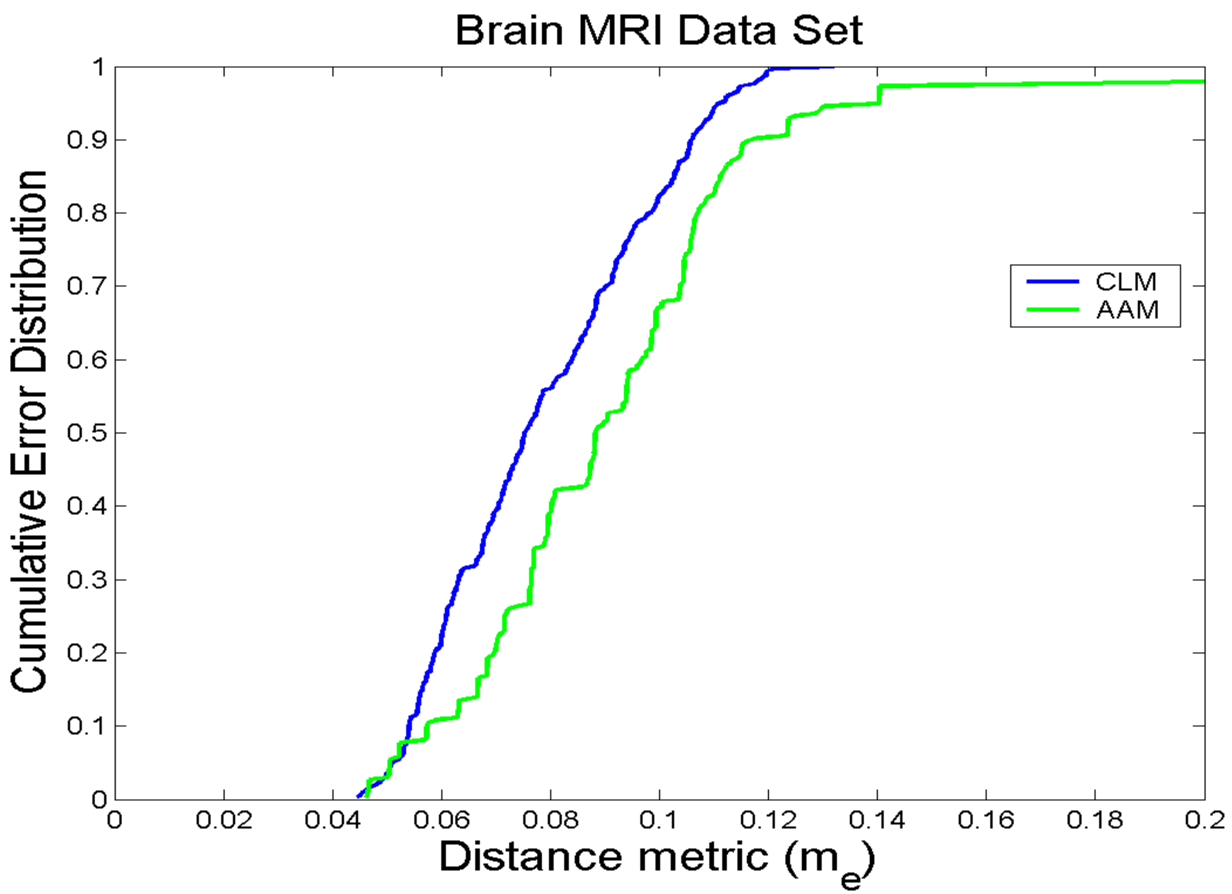


Ref Sep

MANCHESTER 1824

The University of Manchester

Brain Displacement Results



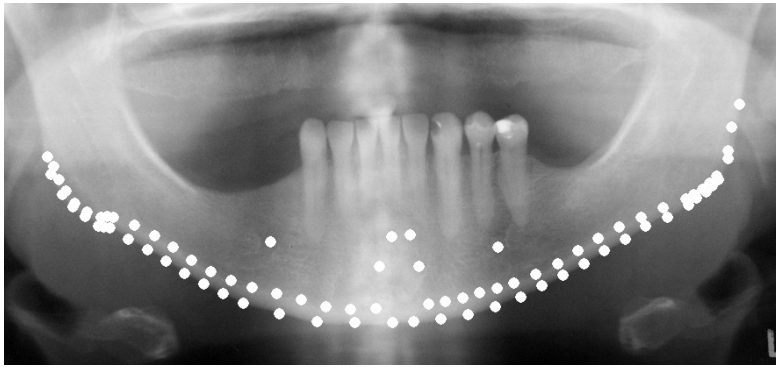
Initial displacement =0.2

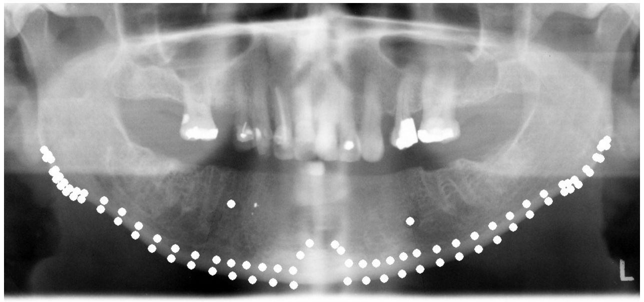
MANCHESTER
1824

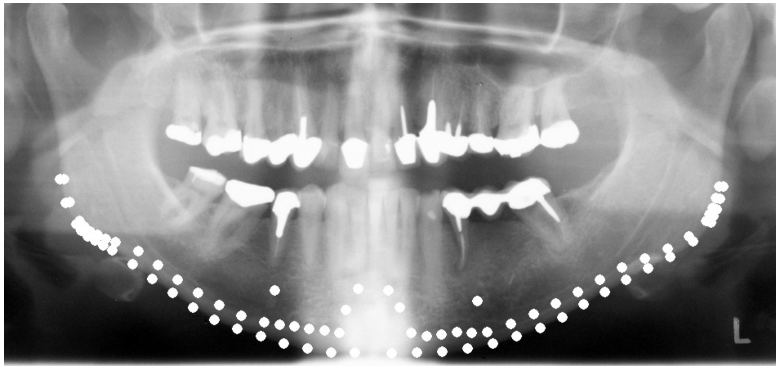
The University of Manchester

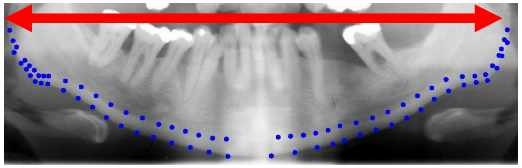
Example Dental Tomograms

- 78 points
- 134 Panoramic Tomograms
  - 67 training
  - 67 testing



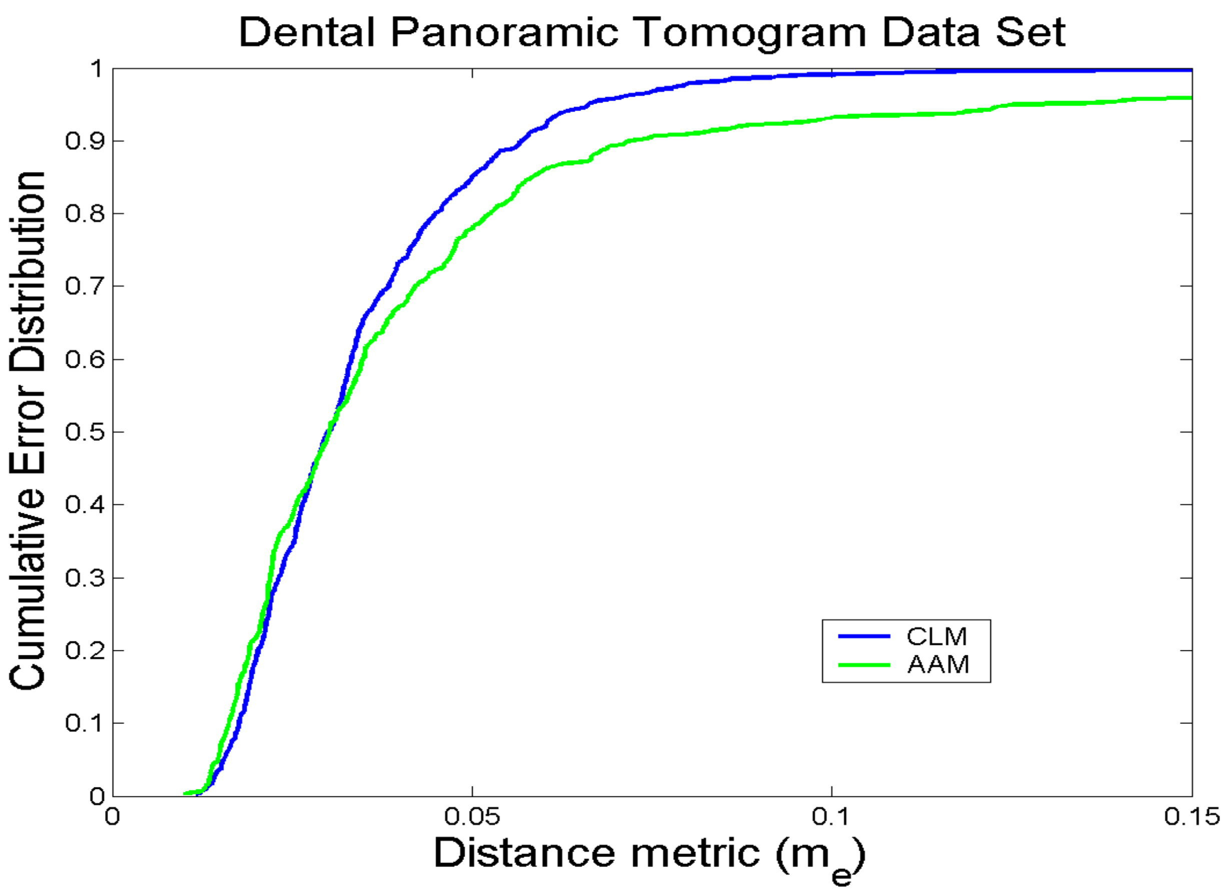






Ref Sep

Dental Displacement Results



Initial displacement =0.1

Constrained Local Models Summary

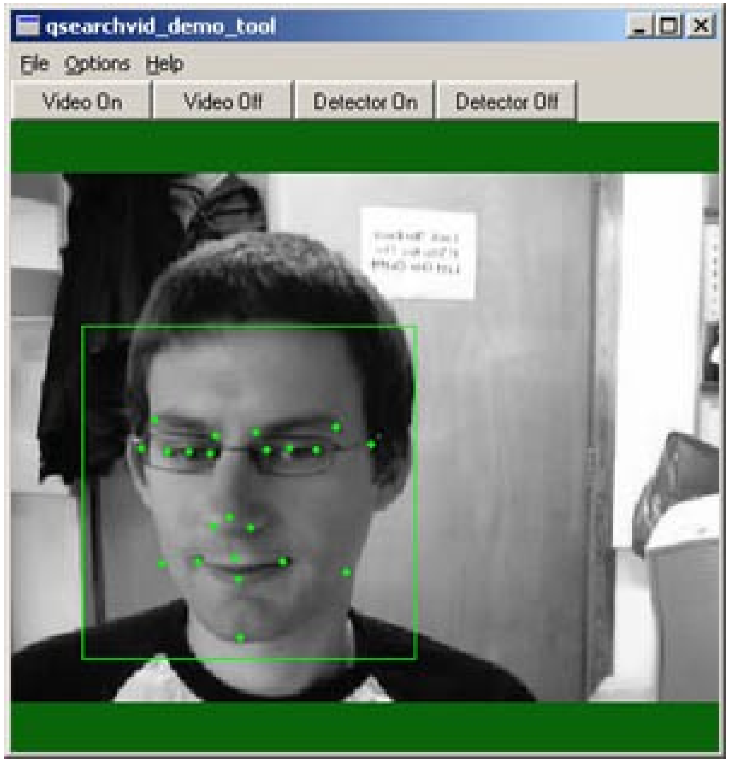
- Wider radius of convergence compared to AAM
- More accurate results
- Faster training time

Full Face Tracking System

- AAM/CLM provide **LOCAL** search results
- Need initialisation with other methods:-
  - Automatically find features in first frame
  - Track feature in subsequent frames
  - Re-initialise when necessary

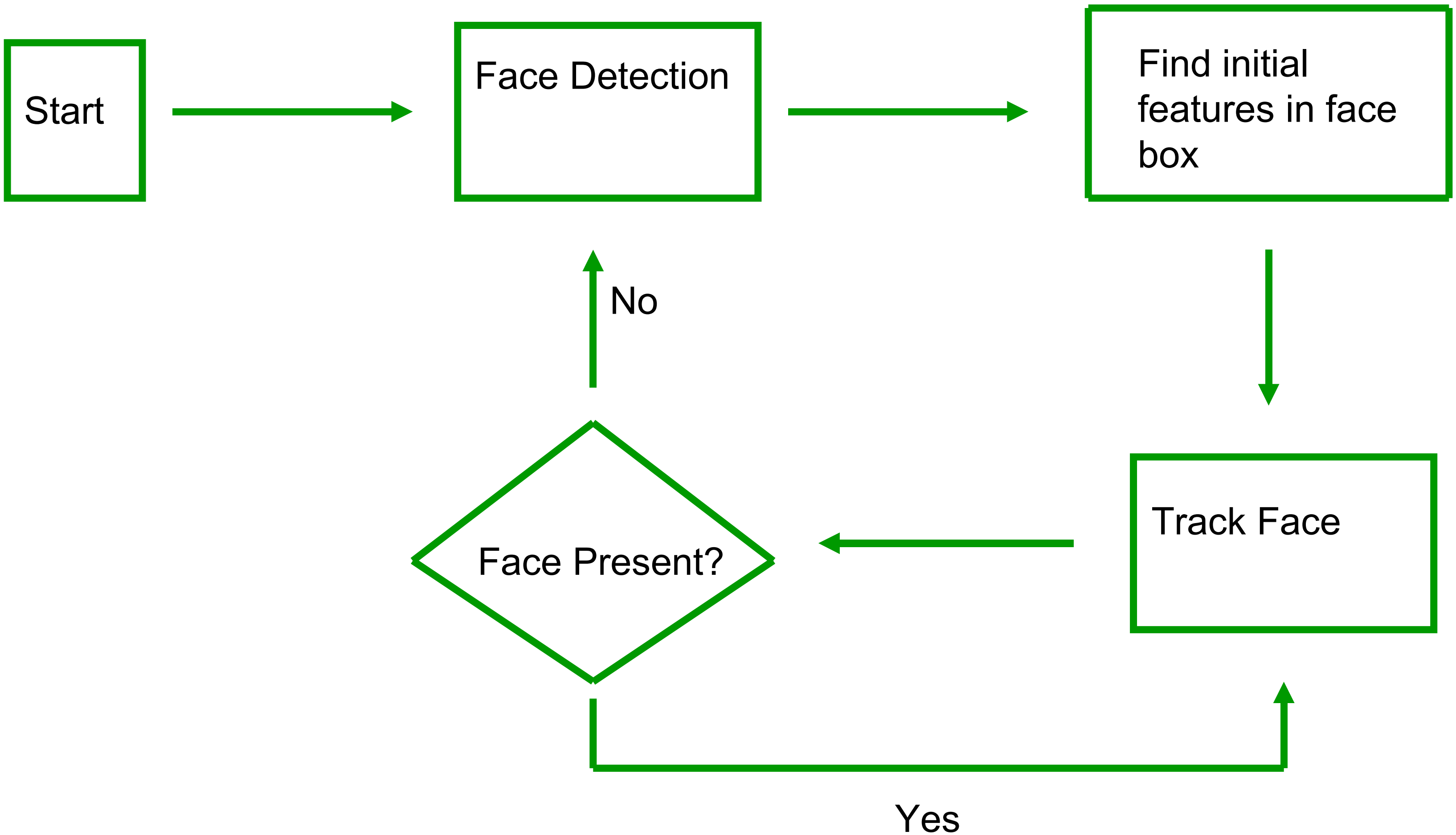
Face Tracking Demo

- Input from webcam
- Applies CLM to update points on each frame
- Windows Binary version available on web



http://mimban.smb.man.ac.uk/downloads/face_demo.php

Detect/Track Framework



Face Tracking Components

- Face Detection (Viola-Jones)
- Finding Initial Features (Pictorial Structures)
- Local Model Fitting (CLM)
- Face Region Verification (Viola-Jones)

Face Detection

- Good implementation of Viola-Jones in Intel OpenCV library

  http://sourceforge.net/projects/opencvlibrary/

- Created own version using VXL library
  - Internal code

Viola-Jones Overview

- Grey Scale Template method
- Three parts-

(1) Integral image structure

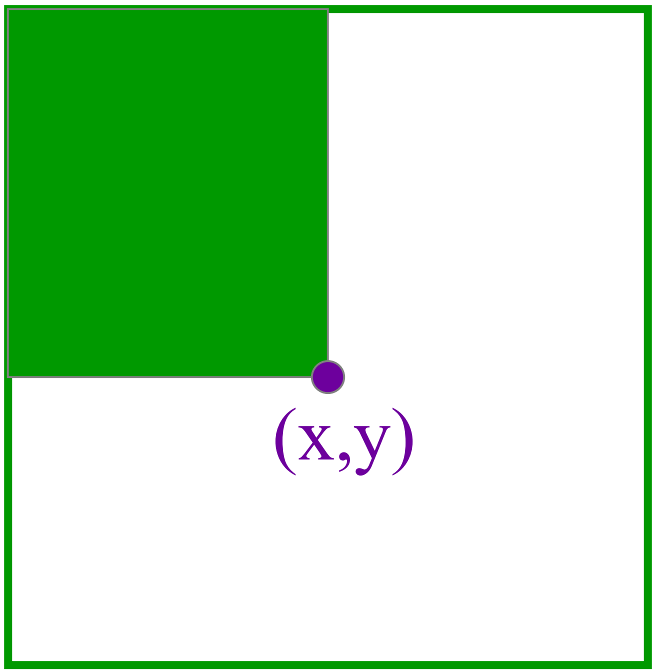
(2) Template Feature Selection – using Adaboost

(3) Cascade of classifiers –speed up!

Ref: Viola & Jones – Rapid Object Detection using a Boosted Cascade of Simple Features –Computer Vision and Pattern Recognition Conference, Honolulu, Hawaii 2001

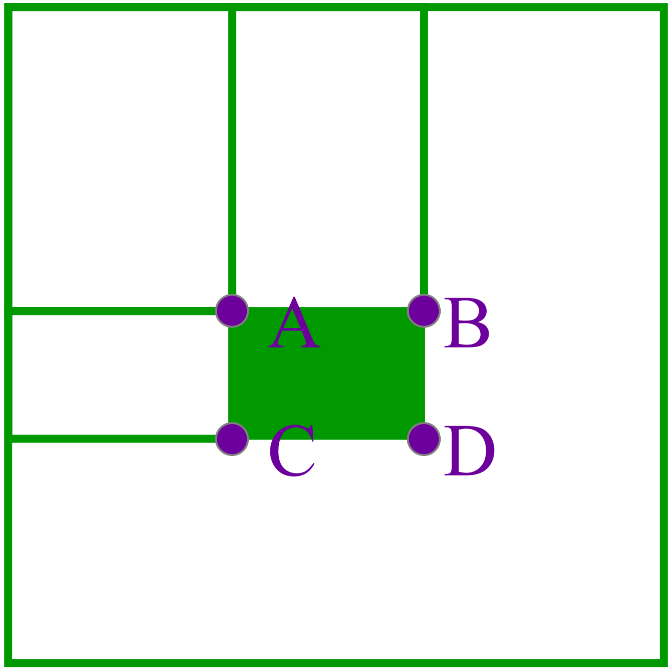
Integral Image

- Pre-compute sum above
  and to left of point (x,y)
- Store values as a new image
- Requires one scan of original image!



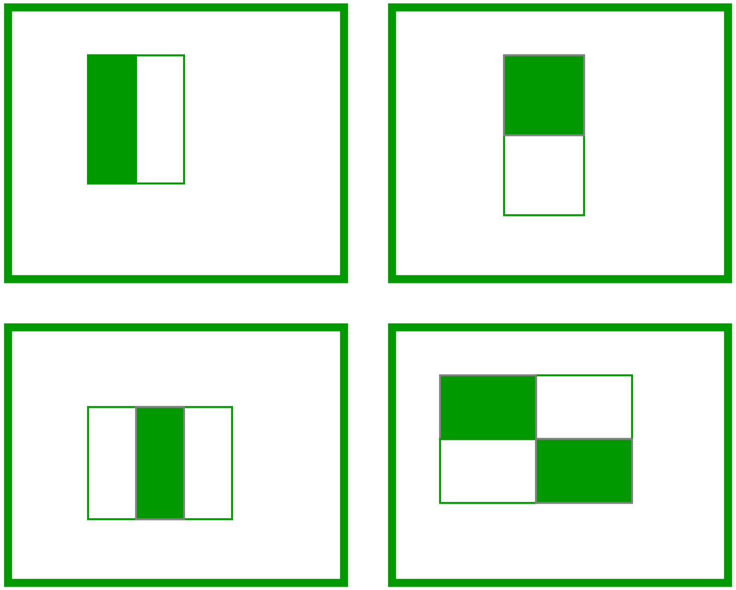
Integral Image 2

- Using integral image can compute sum of grey values in any region of the original image very quickly.
- i.e. 4 operations (A +D- B-C)



Integral Image Features

- Classifier consists of a set of features
- 4 different types
- Based on Haar Wavelets
- Forms massively over complete basis



~30,000 possible features vs 576 pixels for 24*24 size image

Selecting the features

- Need to select useful subset of ~30,000 possible features
- i.e. features that can differentiate face/background
- Use AdaBoost to select features!
- AdaBoost weights the features to produce an effective classifier

Adaboost classifier

Feature template score:-

$$s = \sum_{i=1}^{n} w_i f_i(X)$$

n = number of features

w_i = weight of feature i

X = image window

classification of feature i

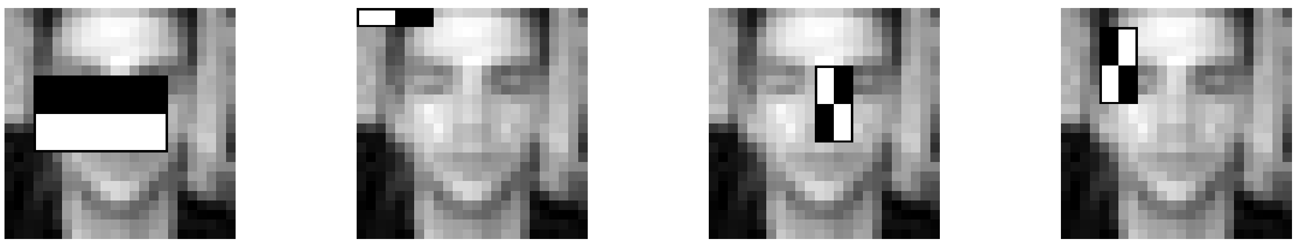
$$f_i(X) = \begin{cases} 0 \\ 1 \end{cases}$$

Face Training set

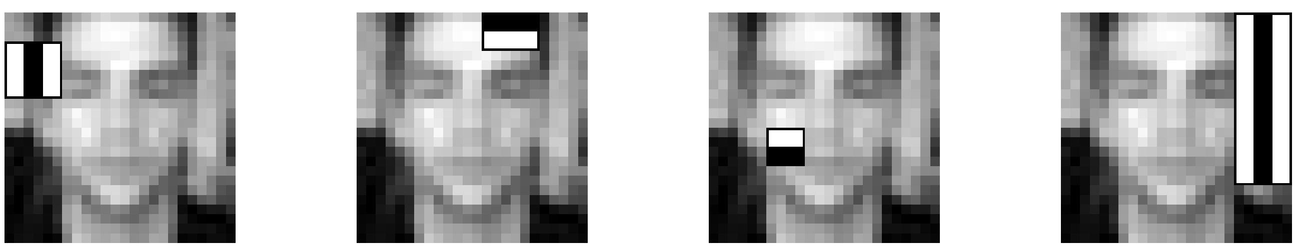
- 5,128 face examples
- 5,782 non-face examples
- each face 24*24 pixels

Features selected

First four features selected by AdaBoost

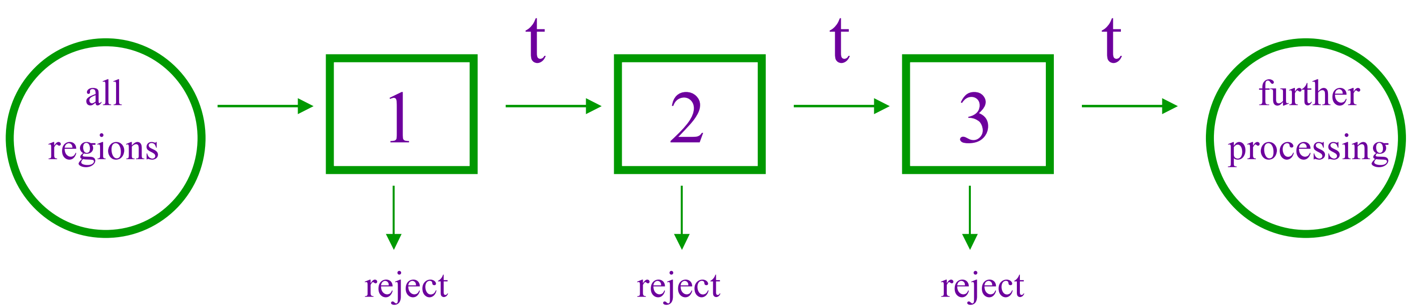


Next four features selected by AdaBoost



Cascade of Classifiers

- Early simple classifiers (Small no. features/Fast)
- Later powerful classifiers (Large no. features/Slow)
- All classifiers have - LOW false rejection rate
  - HIGH false acceptance rate
- Region must pass all classifiers to be declared a face



Training the cascade

- Main parameters:-
  - -Number of features in each level
  - -False rejection rate at each level
- Trial and error process
- Time consuming ~ 2/3 days on 2.0GHz machine for single cascade
- Features chosen dependent on training set
- Threshold at each level dependent on a verification set

Cascade parameters

- Number face examples - 3,076 train
  - 2,052 verification
- Number of non-face examples – 3,470 train
  2,314 verification
- Number of features in each level 10-10-20-20-20-50-50-100-100-100-100-100-200-200-200-200
- False rejection rate at each level 0.01

MANCHESTER 1824

The University of Manchester

Viola-Jones Summary

- Training time is a few days, using 2.0Ghz machine!
- This inhibits ability to vary training parameters
- Final detector very quick

  ~ 200ms for 320*240 image on 500Mhz PII

MANCHESTER
1824

The University of Manchester

Static Image Test Set - BIOID

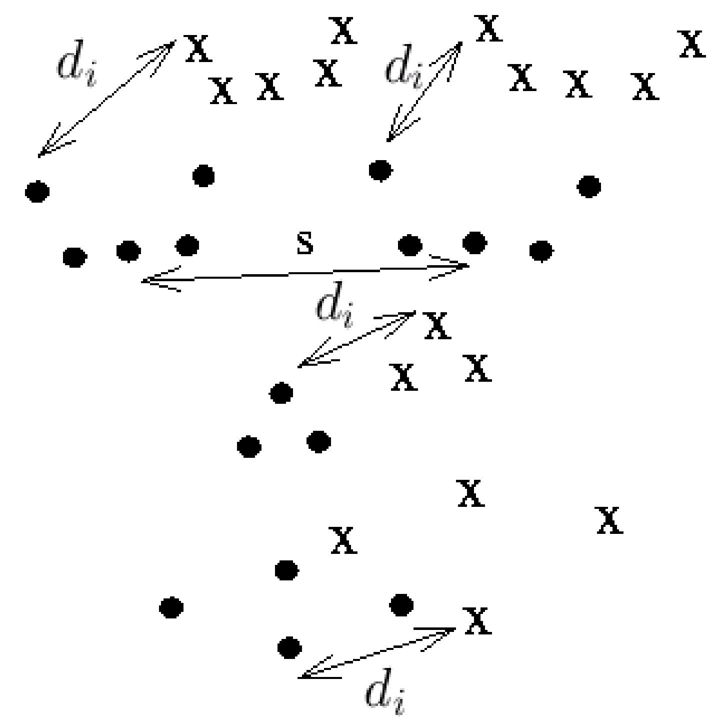
- 1522 images
- Size 384*286
- 23 identities
- Face large in image
- Low quality camera
- Some background variation
- Lighting variation

Point to Point Error Measure

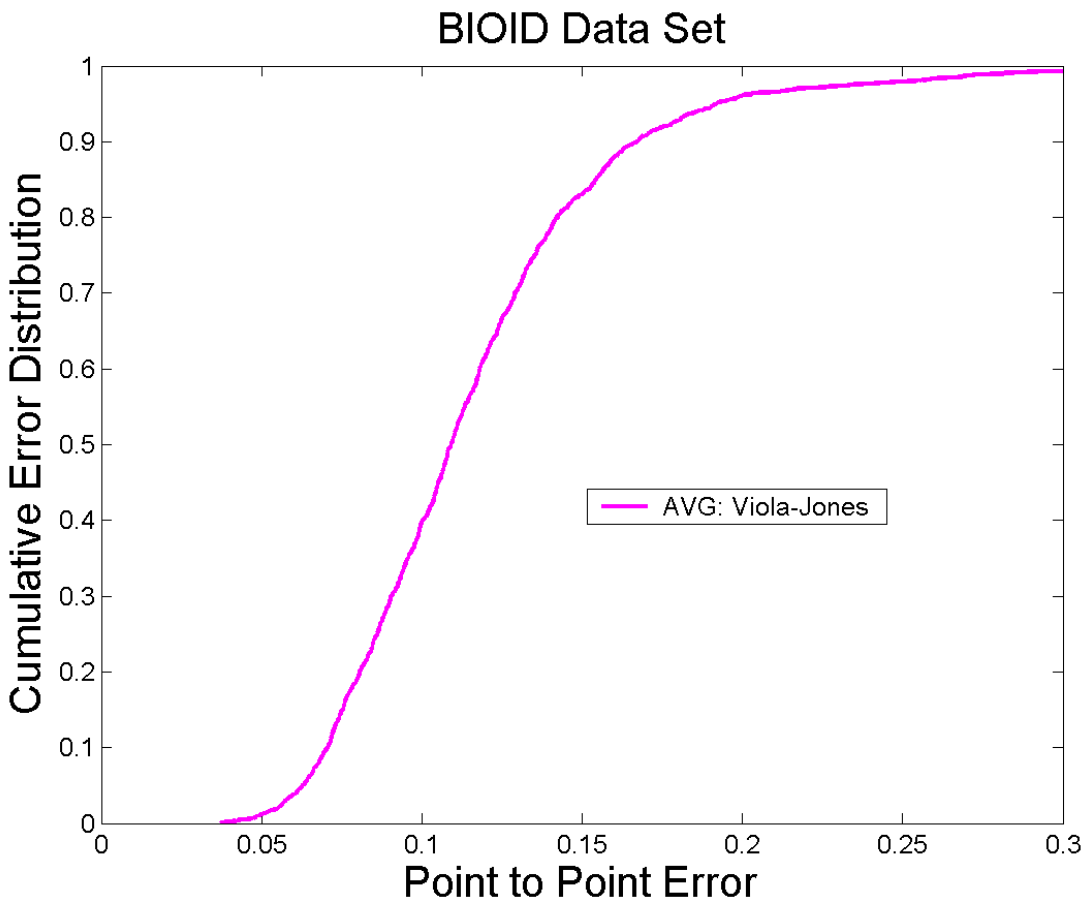
● = Manually labelled points

X = Automatically predicted points

$$error = \frac{1}{n*s} \sum_{i=1}^{n} d_i$$

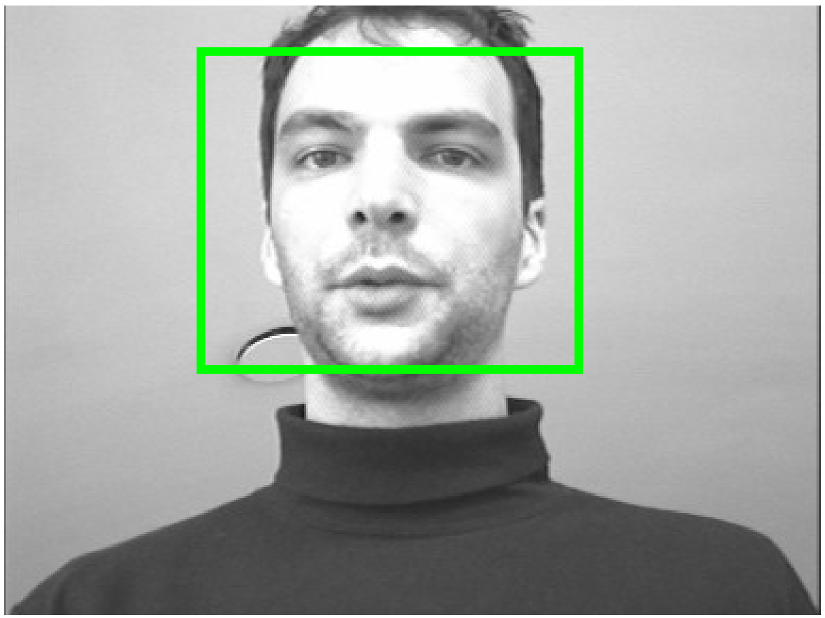


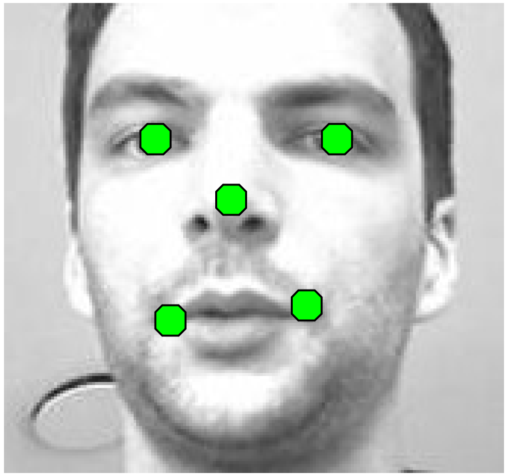
BIOID Full Search



Face detection only – no local search

Finding Initial Features

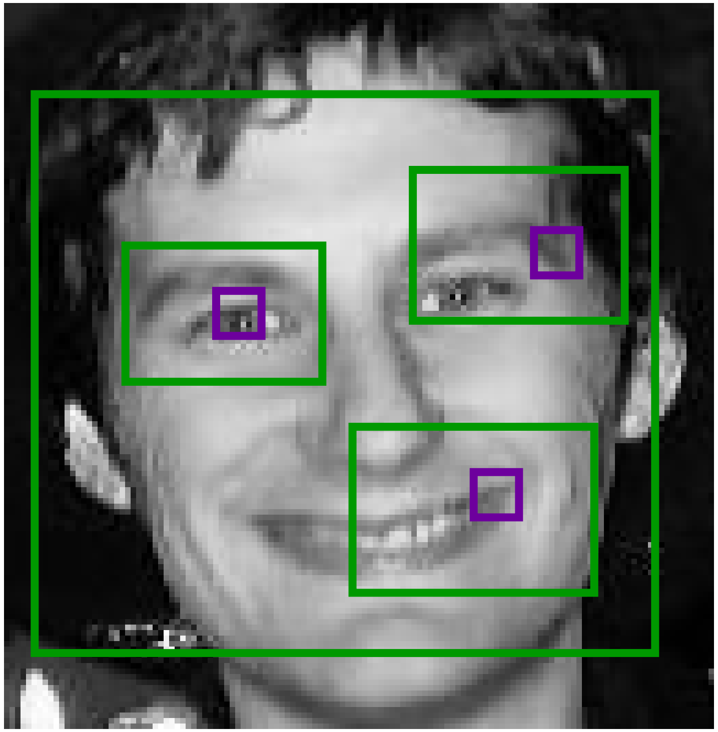




- Have box around face from face detector
  - need initial feature points

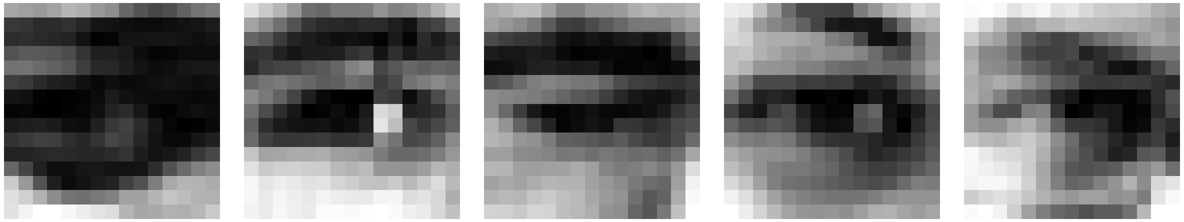
Local Feature Detectors

- One detector for each feature
- Learn range of features within face detection region
- Unreliable feature detectors

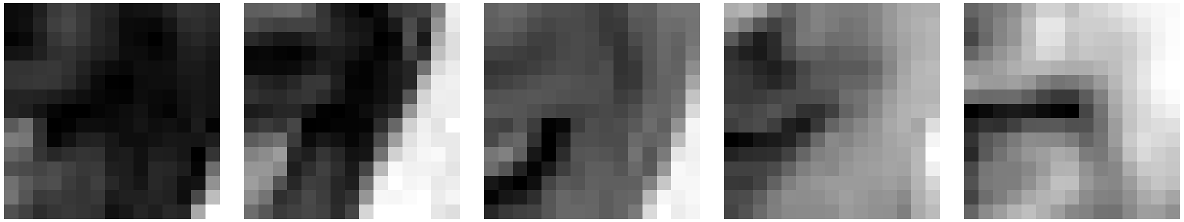


Example Training Features

- Right Eye



- Left Mouth Corner



Pictorial Structures

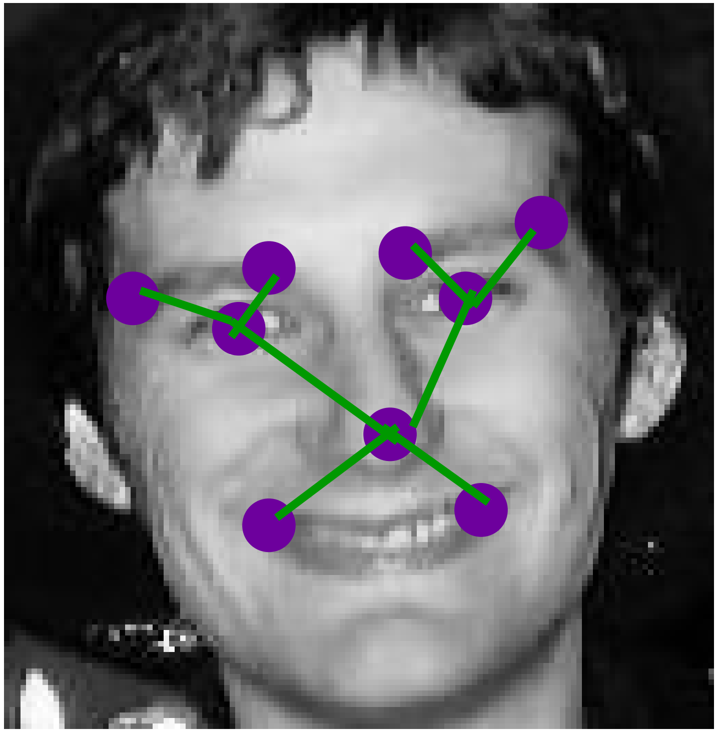
- Need Shape Constraints to cope with feature outliers
- Pictorial Structures

  - Efficient method of combining response images of feature detectors

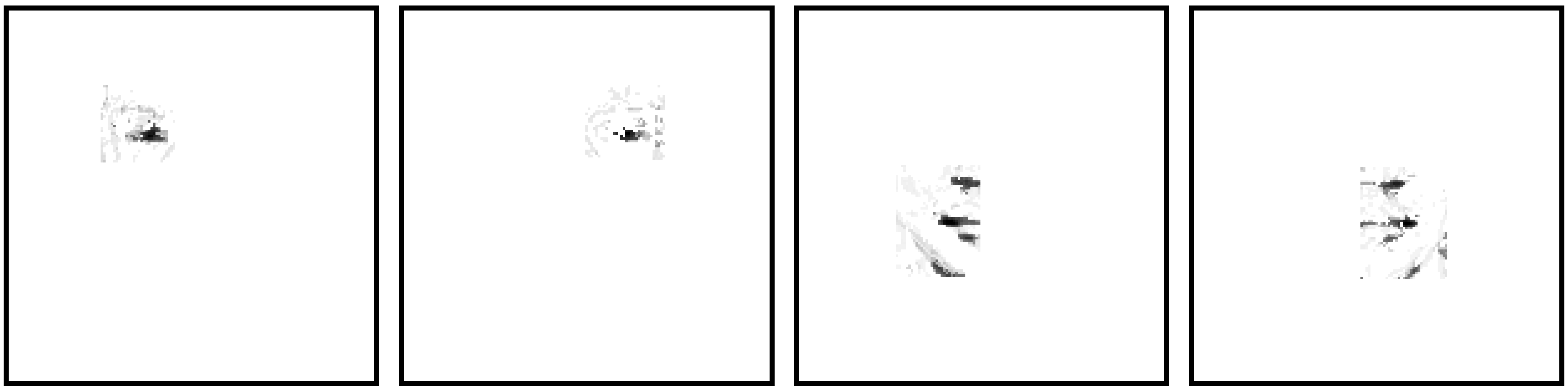
Ref: Felzenszwalb & Huttenlocher – Efficient Matching of Pictorial Structures–Computer Vision and Pattern Recognition, 2000

Pictorial Tree Structure

- Pairwise relationships between nearby points
- Cost of deformation is distance from mean separation of each pair
- Deformation Cost Learnt from Labelled Training set



Example Response Images

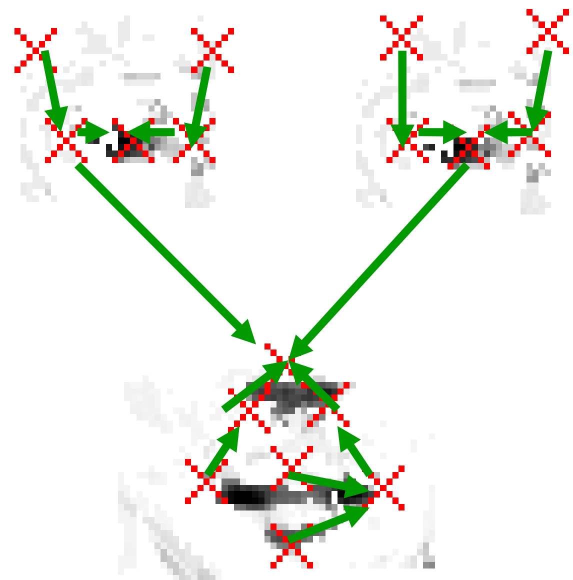


- Right eye
- Left eye
- Right mouth corner
- Left mouth corner

Dark => good response

Dynamic Programming Search

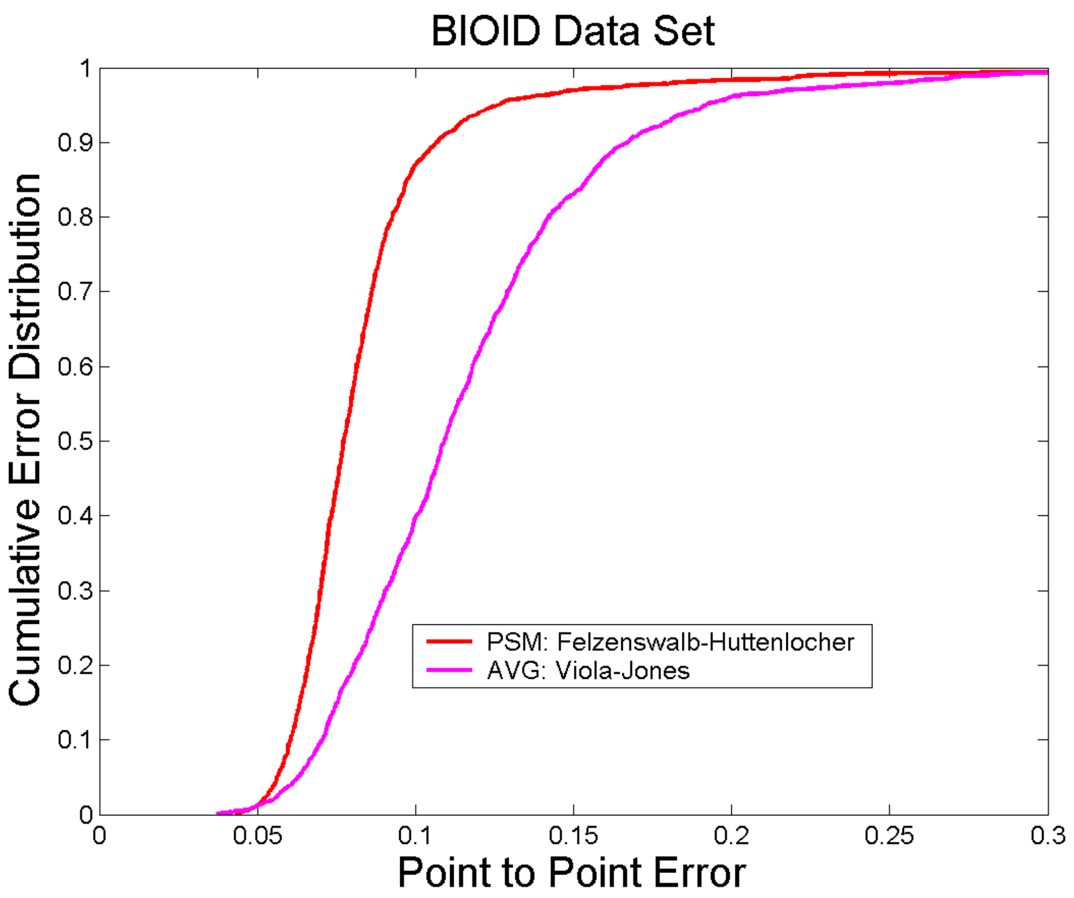
- Compute response surface for each feature detector
- Apply distance transform to encode uncertainty
- Use to predict location of parent node
- Once found root node – work backwards
- Finds global optimum!



Summary of Pictorial Structures

- Soft method (uses whole response surface)
- Efficient linear search
- Relatively weak shape constraints
- Used to initialise AAM/CLM local search

BIOID Full Search



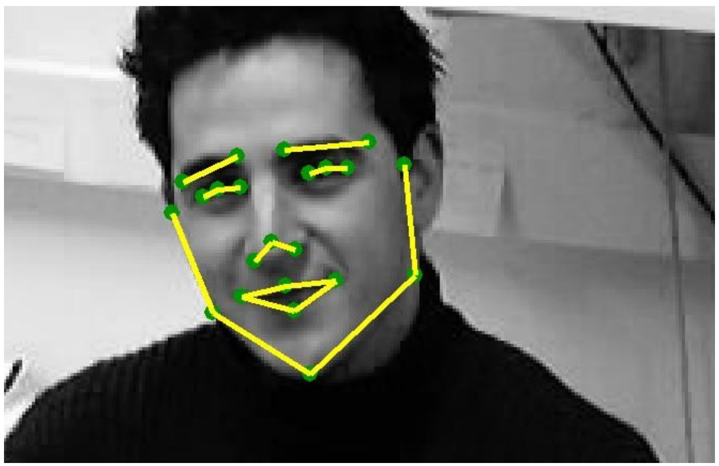
Face Detection + PSM local search

Local Model Fitting

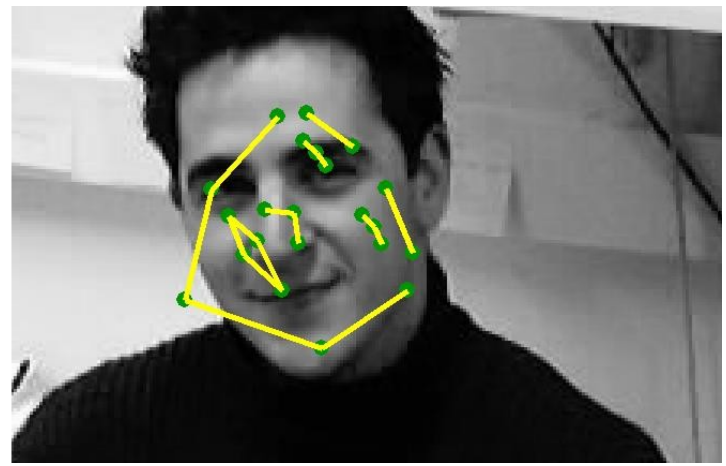
- Take initial feature points + improve local fit
- Can use any local search method
  - Active Shape Model (ASM)
  - Active Appearance Model (AAM)
  - Constrained Local Model (CLM)

Face Verification

- Did search converge correctly?



Correct



Wrong

Adaboost Patch Classifier

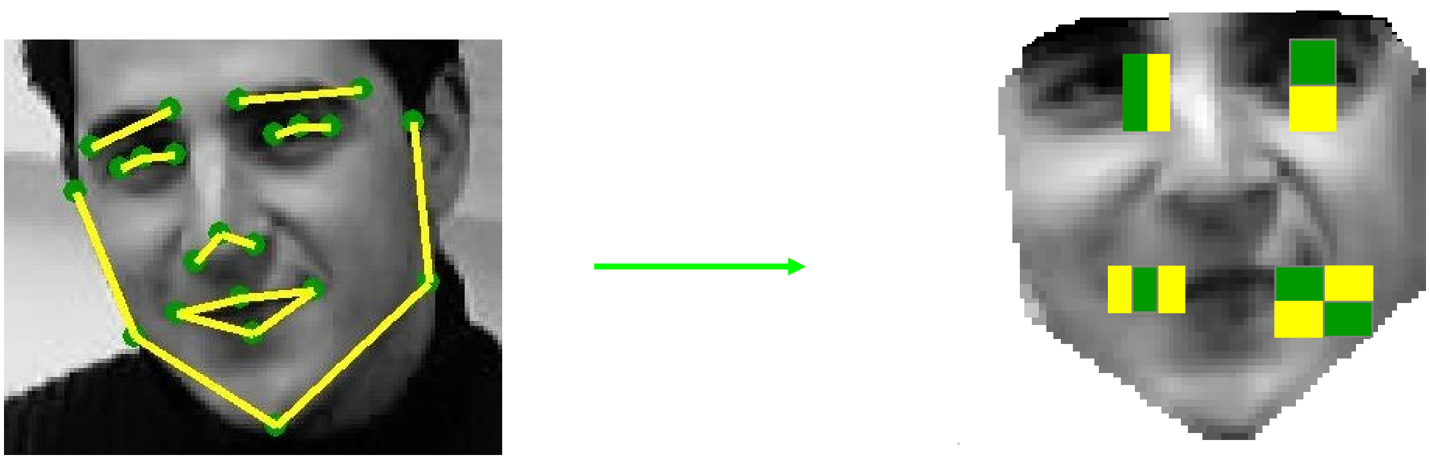


Image Points

Mean Shape Texture

Apply Viola-Jones style features to mean shape texture captured from image

Full Search Method

Global search
VJD: Viola-Jones Face Detector

→

Point Prediction
PSM: Pictorial Structure Matching

→

Point Refinement
CLM: Constrained Local Model

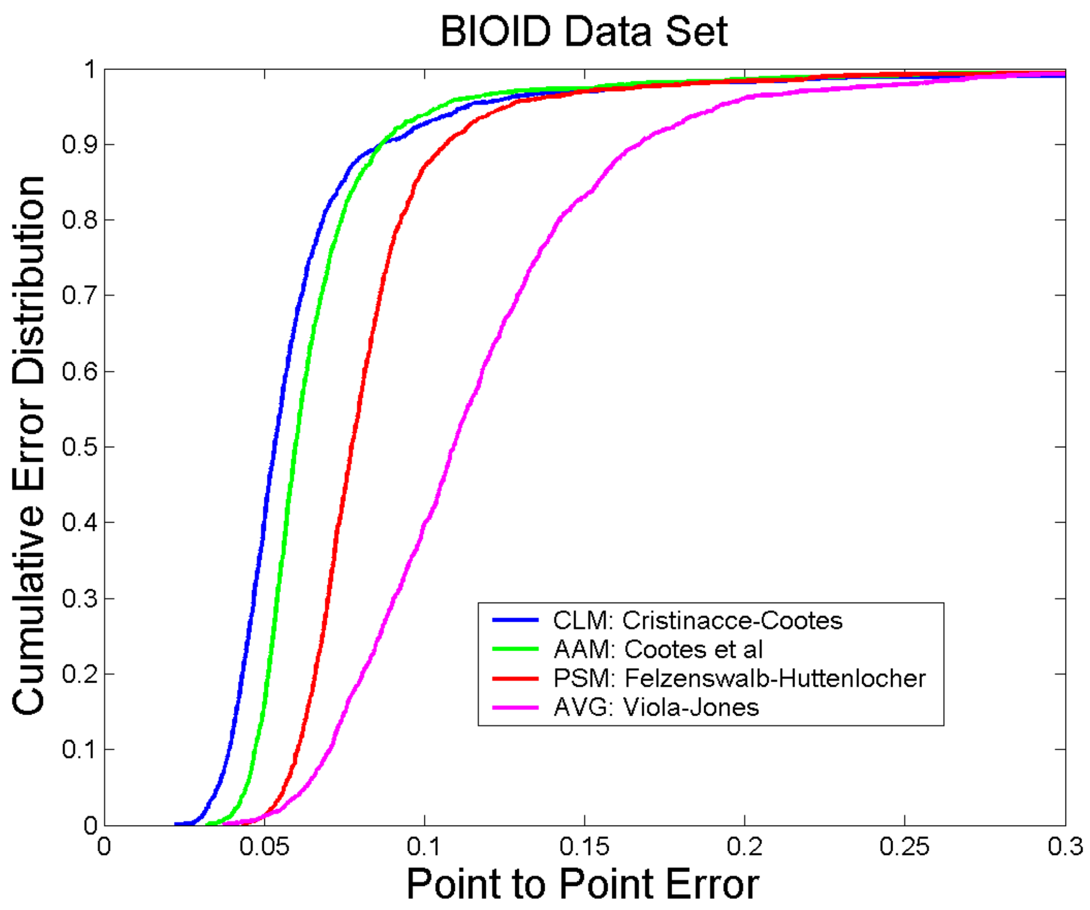
or

AAM: Active Appearance Model

MANCHESTER
1824

The University
of Manchester

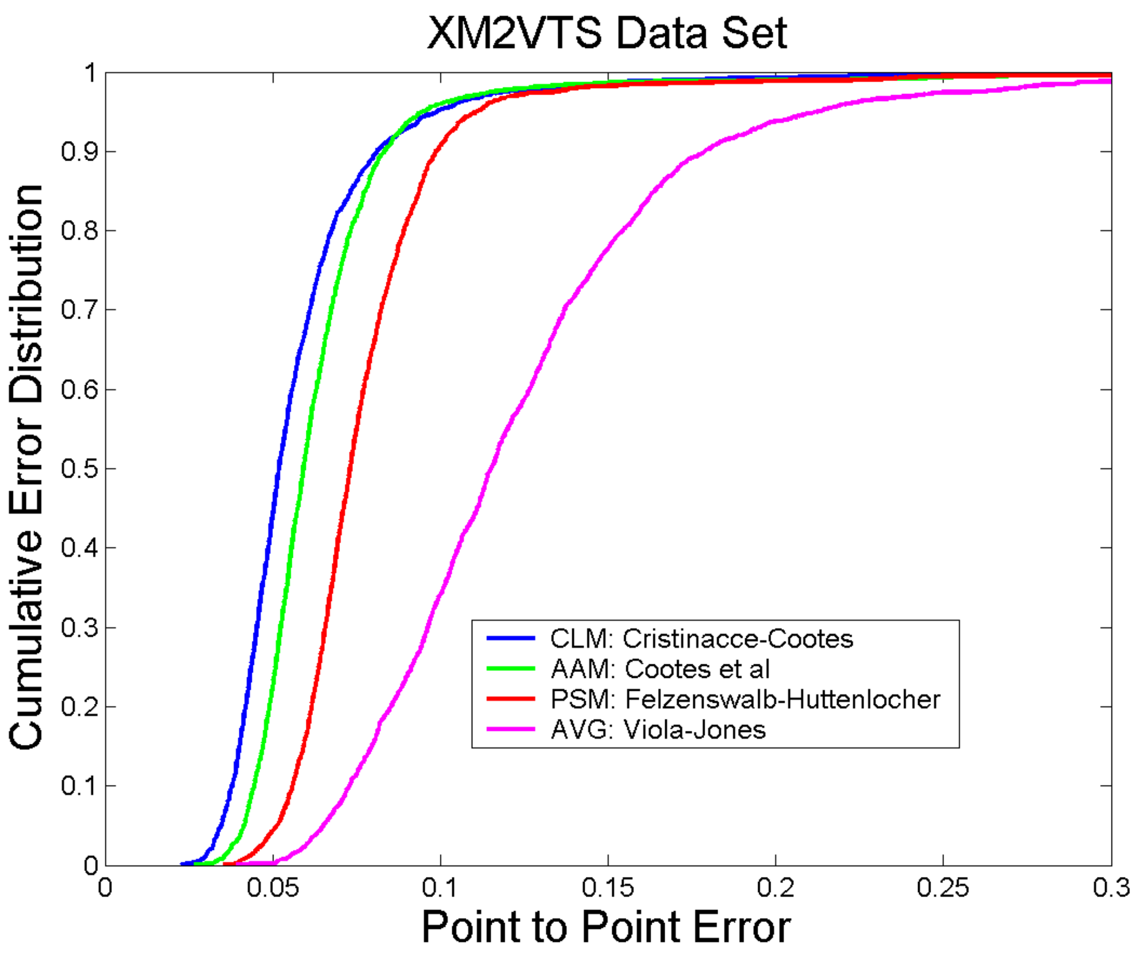
BIOID Full Search Results



XM2VTS Test Set

- 1817 images
- Size 720*576
- ~200 identities
- Face large in image
- Little Background Variation
- Glasses + Beards

XM2VTS Full Search Results



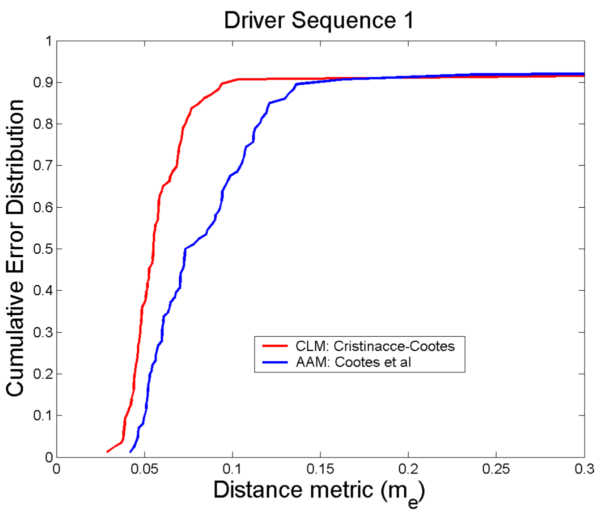
MANCHESTER 1824

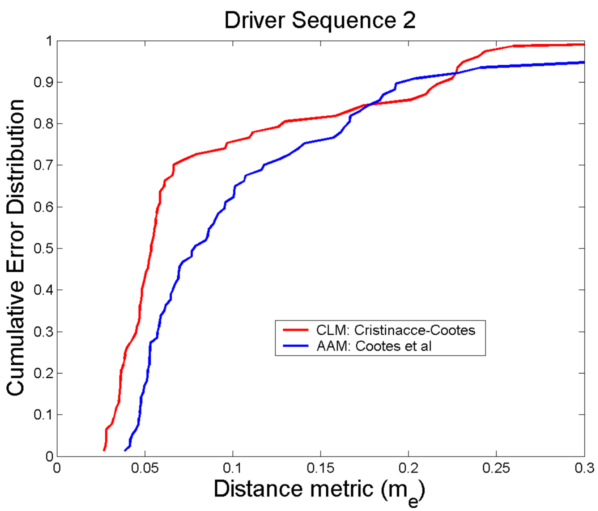
The University of Manchester

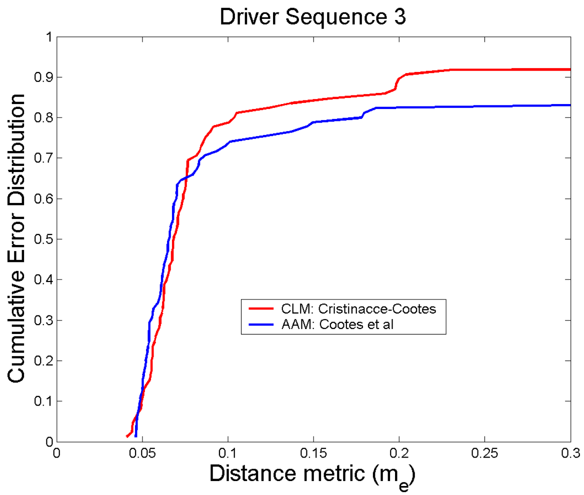
Tracking Test Set

- 3 video sequences
- Lighting variation
- Face large in image
- Head rotation

Tracking Results – Driver Sequences







Face Tracking Summary

- Improved results when using CLM vs edge/corner AAM
- Tracking results very dependent on face detector/verifier

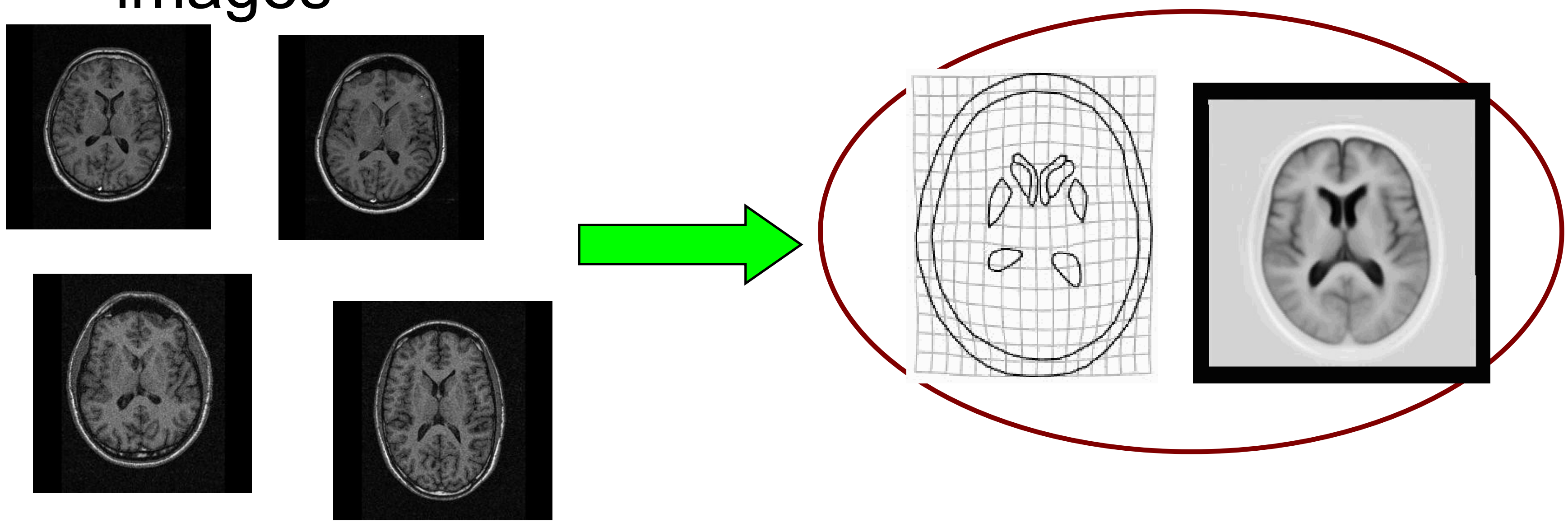
End of face tracking section

Automatic Model Building

- Wish to avoid requirement for manually labelled data
- Problem with human annotation:-
  - Error prone
  - Sometimes inconsistent
  - Tedious

Goal: Fully Automatic System

- Build models from unlabelled images

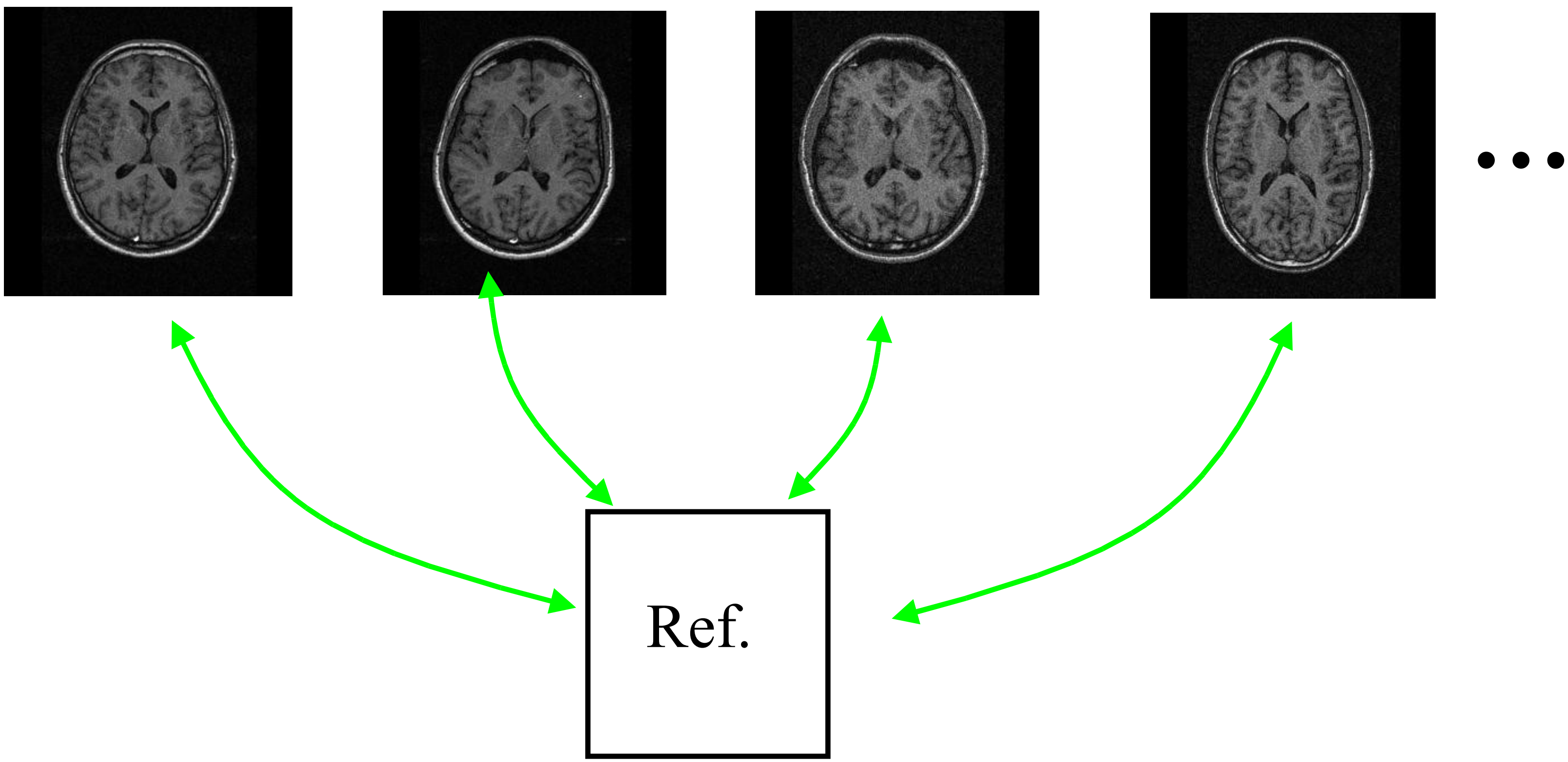


• • •

Model

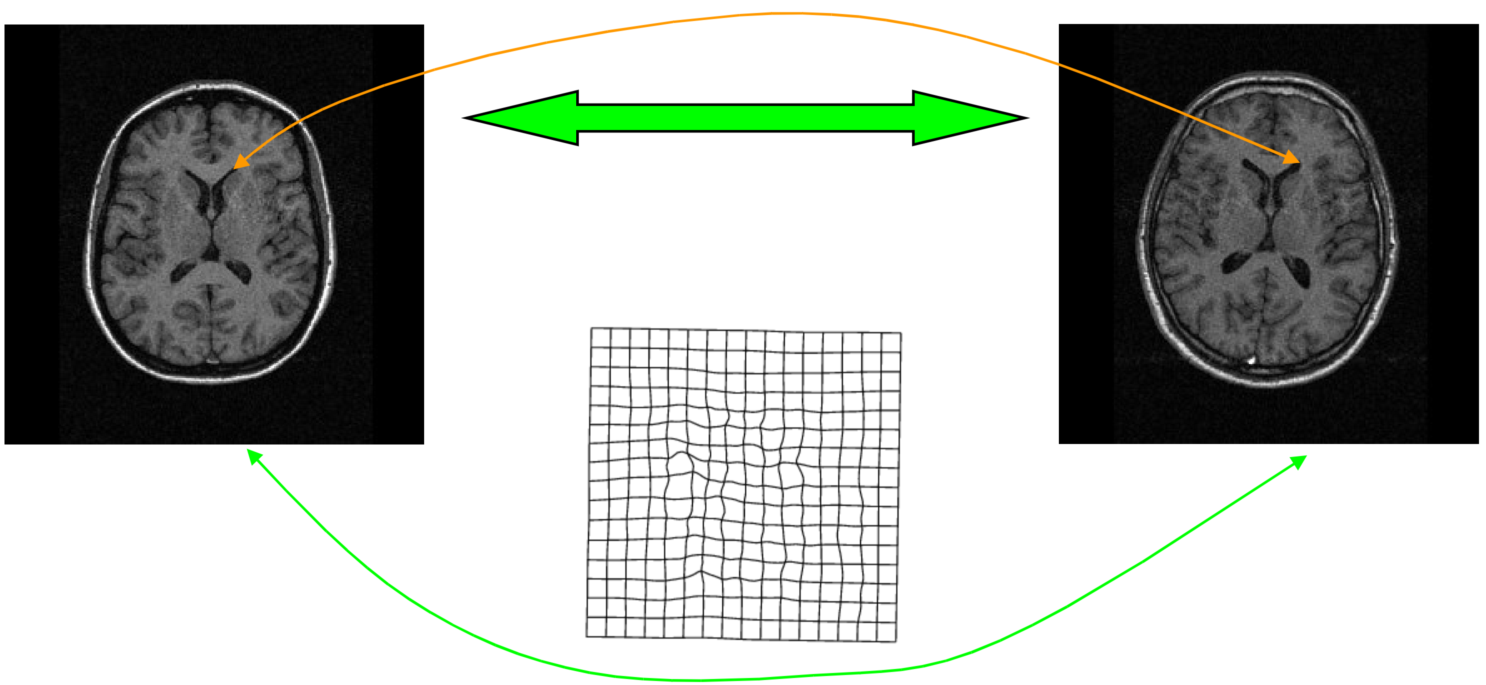
Approach

Find the correspondences across the set of images



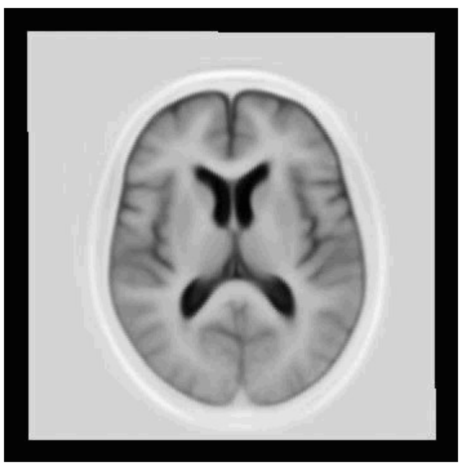
Assumptions

- For every point on one image there is a corresponding point in every other image
- Deformation field is invertible

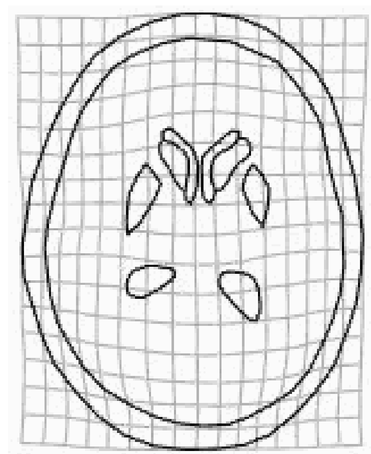


Assumptions

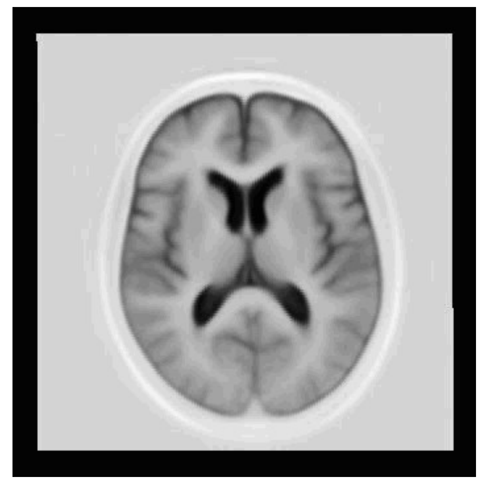
- Images can be generated by
    1. Creating texture in reference frame
    2. Applying a shape deformation



Texture



Shape



Shape and texture

MDL/Coding Approach

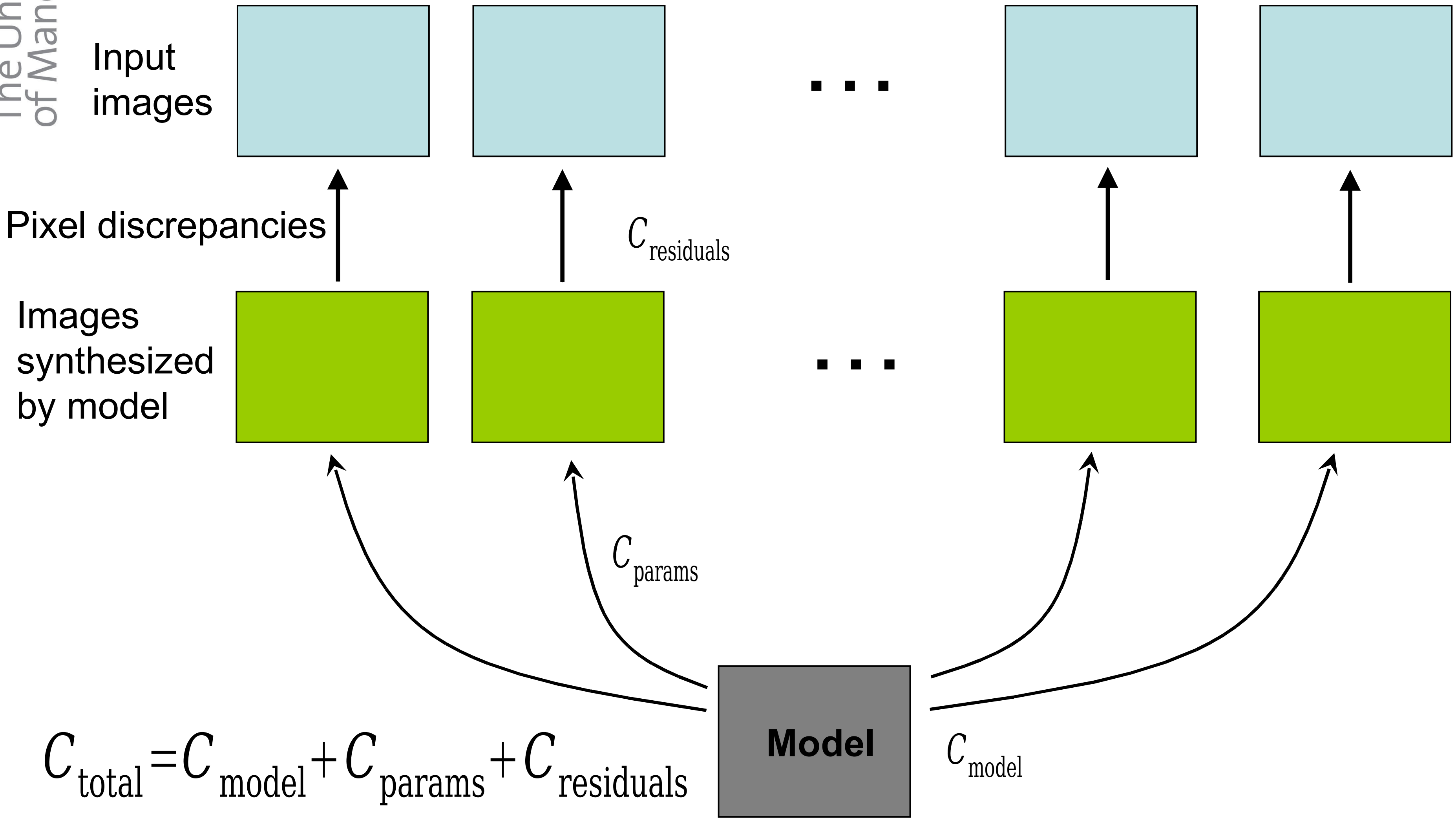
Aim: Construct the “best” model

- Model which can encode data most compactly
- Manipulate correspondences to optimise description length of training set

Combines registration with model-building

Ref: Cootes & Twining et al – Groupwise Construction of Appearance Models using Piece-wise affine deformations. BMVC 2005

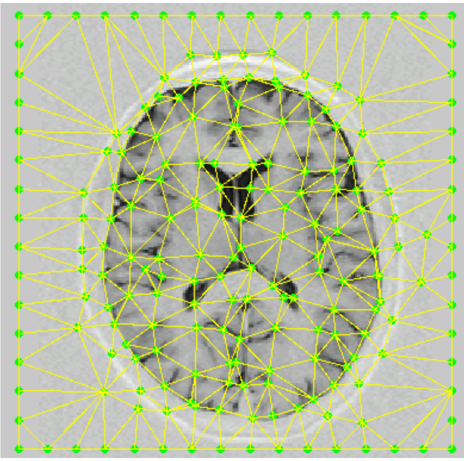
MDL Framework



$$C_{\mathrm{total}} = C_{\mathrm{model}} + C_{\mathrm{params}} + C_{\mathrm{residuals}}$$

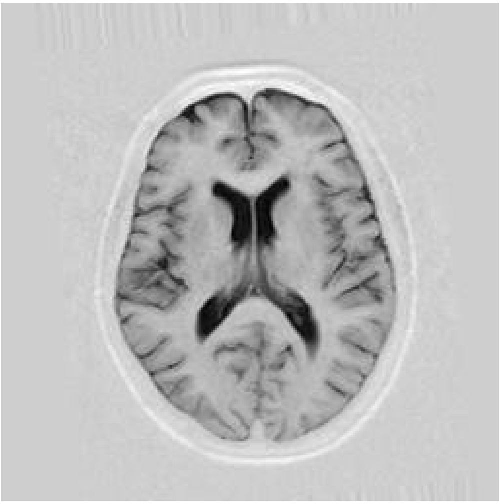
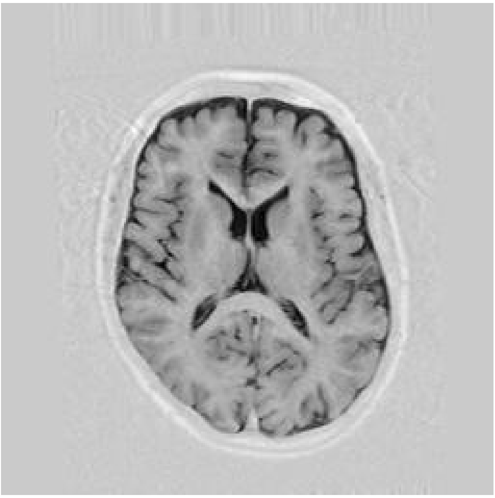
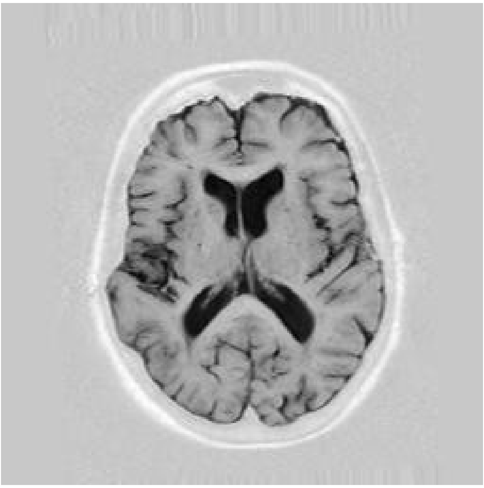
Groupwise Algorithm

- Initialisation: affine registration
- Generate control points
- Repeat
  - Select image
  - Build model from all others
  - Compute coding cost of current image
    - Optimise positions of control points
- Until happy

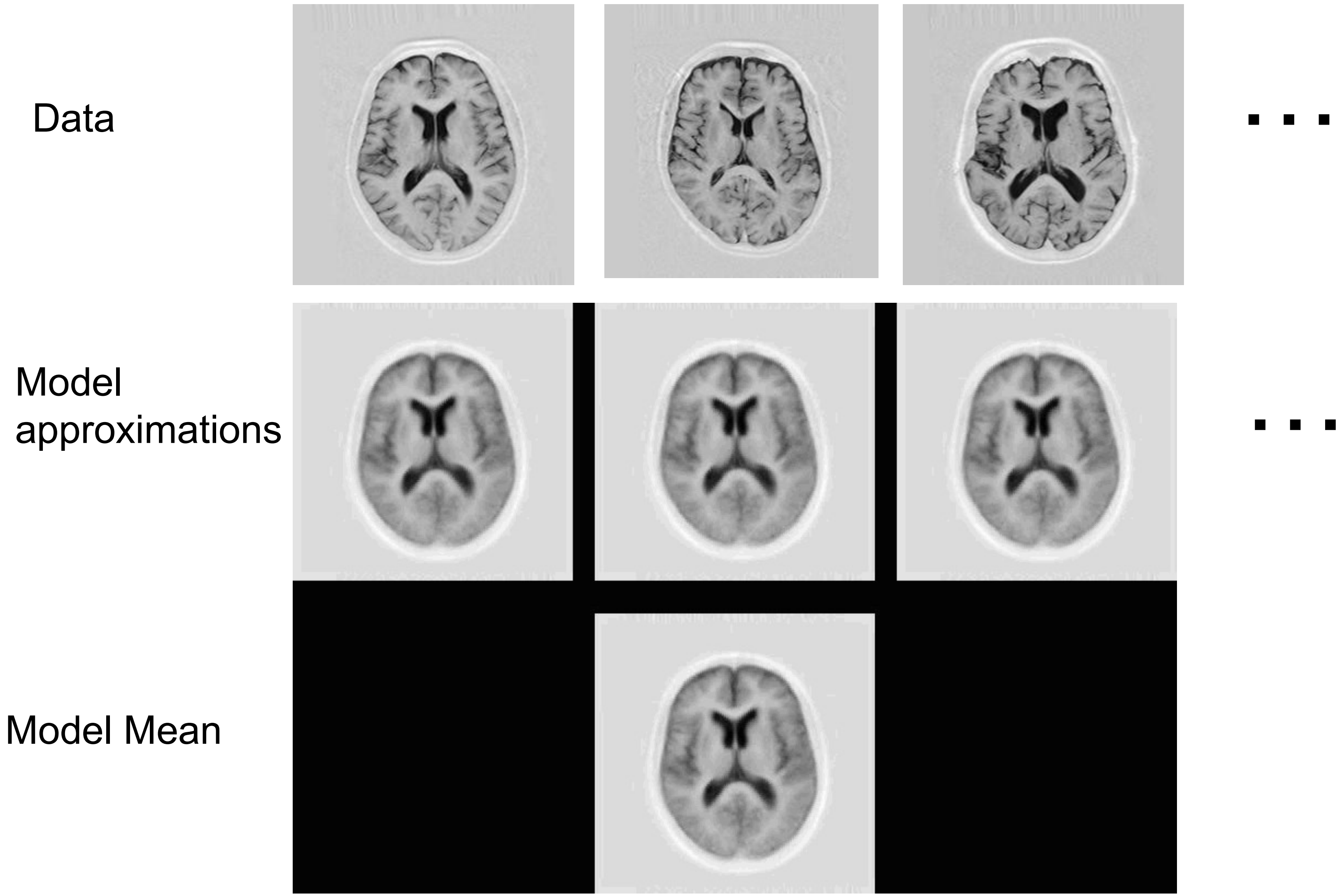


Examples (2D)

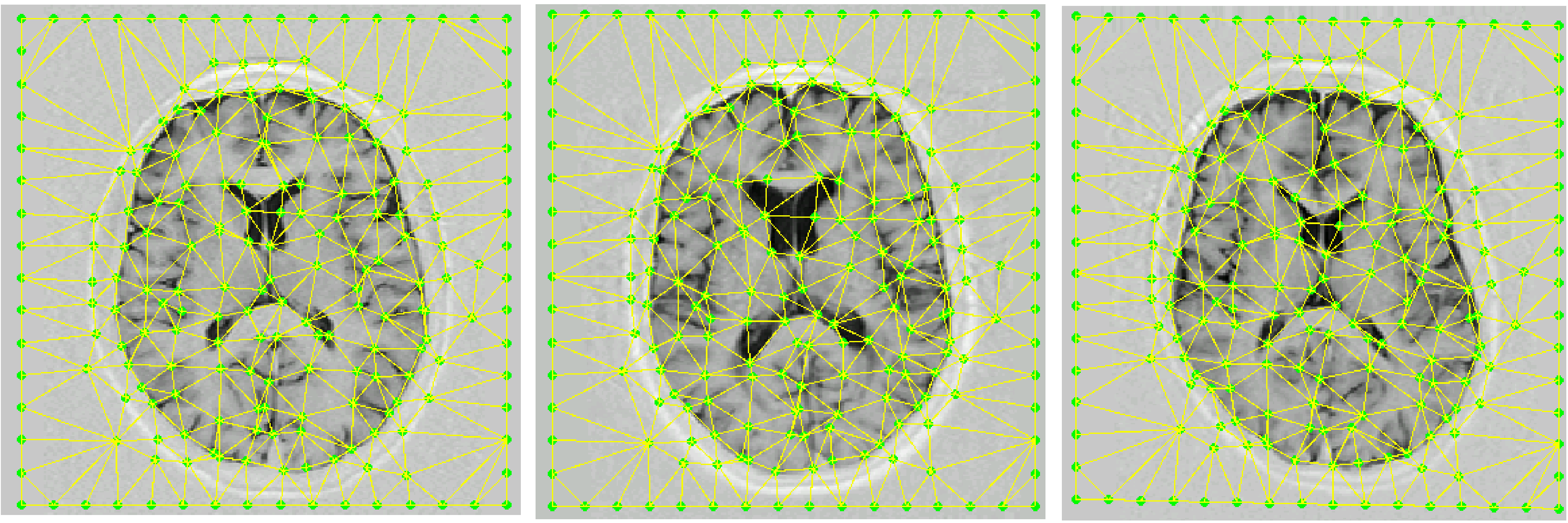
- 104 3D MR images of `Normal' brains
- Affine align 3D images first
- Select equivalent 2D slices

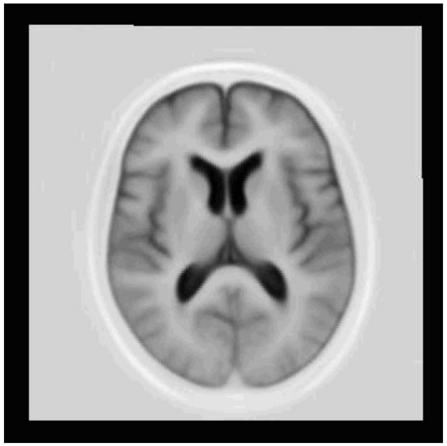
2D Registration



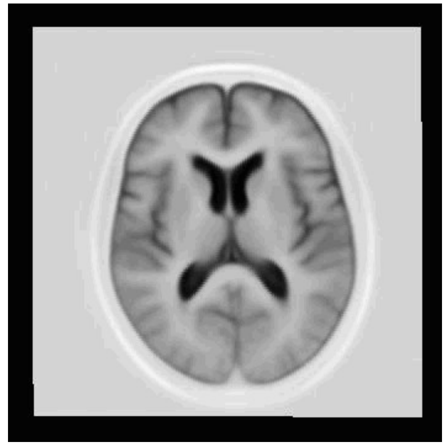
Correspondences:



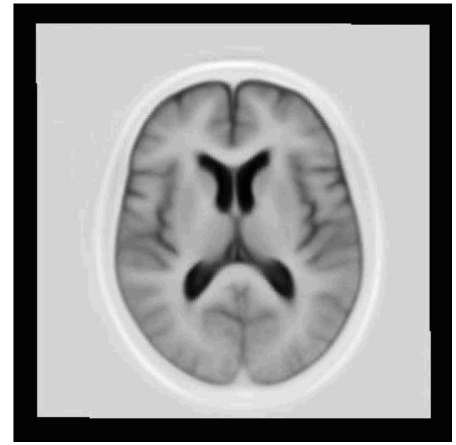
Brain Model Modes



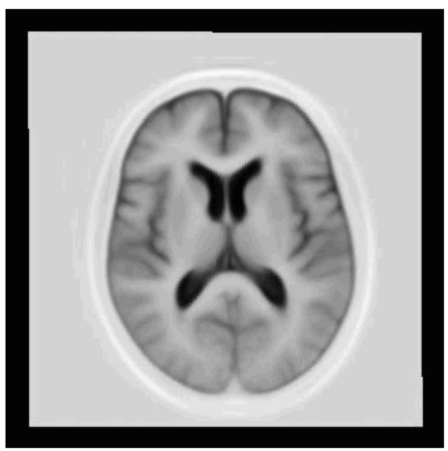
Shape Mode 1



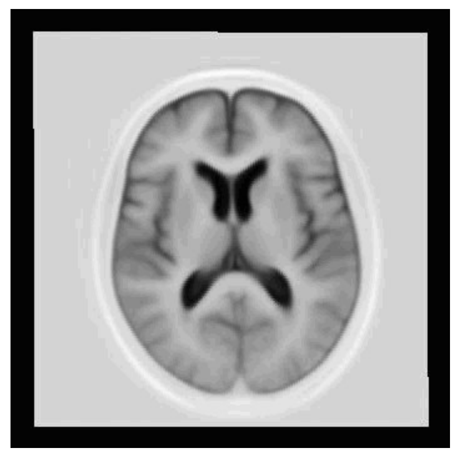
Shape Mode 2



Shape Mode 3



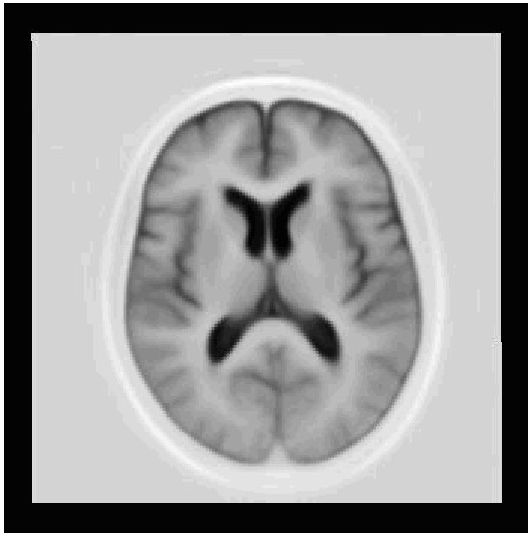
Texture Mode 1



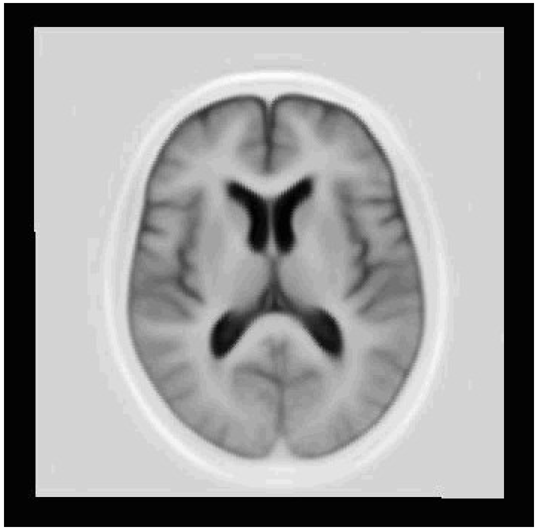
Texture Mode 2

Brain Appearance Modes

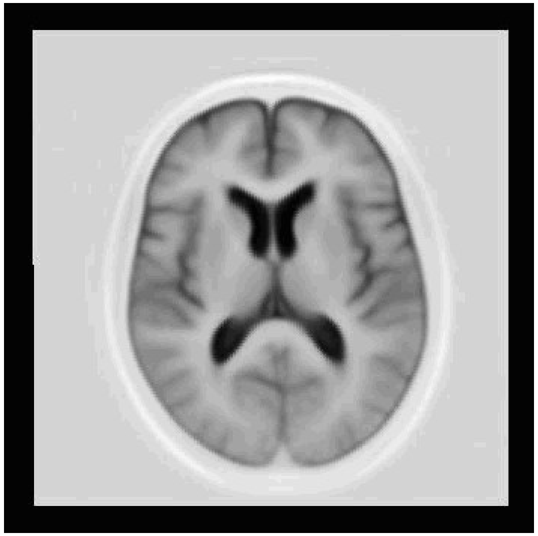
- Combined shape/texture modes



Combined Mode 1



Combined Mode 2



Combined Mode 3

Model of an individual

Part of the training data:

Some of the model modes:

Population models:

- Model built from of 161 different faces

Mode 1

Mode 2

Mode 3

Summary

- Lots of progress in recent years in model based computer vision
- HCI applications in appearance model space (face recognition etc)
- Automatic model building has great potential, but lots of work left to do

Future Work

- Further improve tracking methods
- Extend to 3D model (more view angles)
- Apply improved model matching to face recognition+ fatigue detection

End of tutorial!

MANCHESTER
1824

The University
of Manchester