Monocular Head Pose Estimation

Pedro Martins, Jorge Batista

Institute for Systems and Robotics
http://www.isr.uc.pt
Department of Electrical Engineering and Computers
University of Coimbra
Portugal

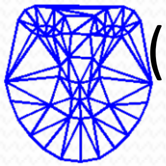
AIFI
Advance Interaction
using Facial Information

AI_FI – Advance Iteraction using Facial Information
FCT Project POSC/EEA-SRI/61150/2004
http://aifi.isr.uc.pt

Introduction Agenda

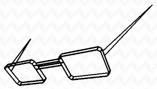
- Single View 6DOF Pose Estimation
  - Human Computer Interface (HCI)
  - Face Recognition Systems
  - Knowledge about gaze direction
  - Video Compression

- Active Appearance Models (AAM)



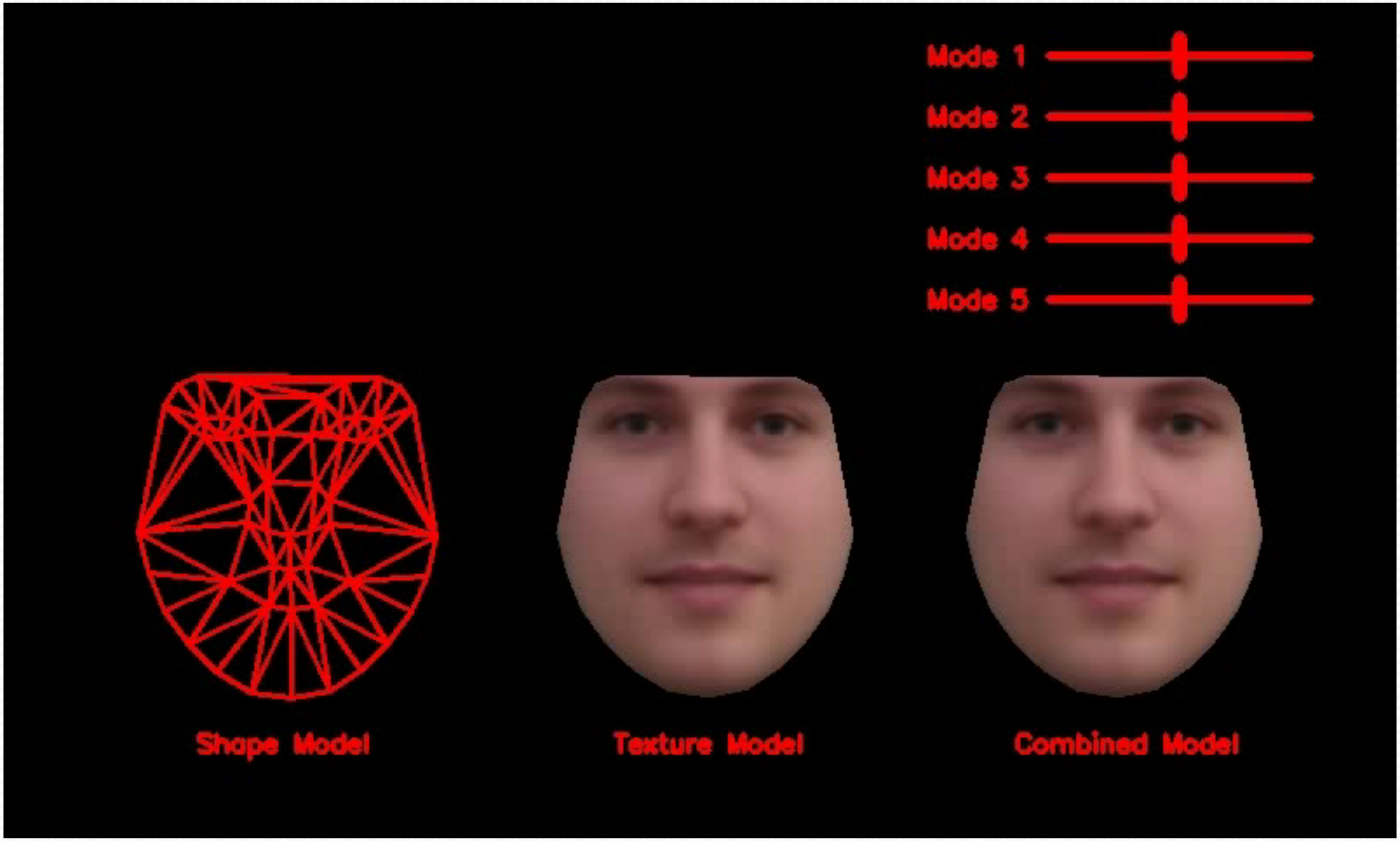
  - Shape, Texture and Combined Models
  - Model Training
  - Model Fitting
- Monocular Pose Estimation

  - Pose from Orthography and Scaling with ITerations (POSIT)
  - Anthropometric 3D Model
  - Pose Evaluation
  - Augmented Reality



Face Model

A set of input parameters generate a face image output



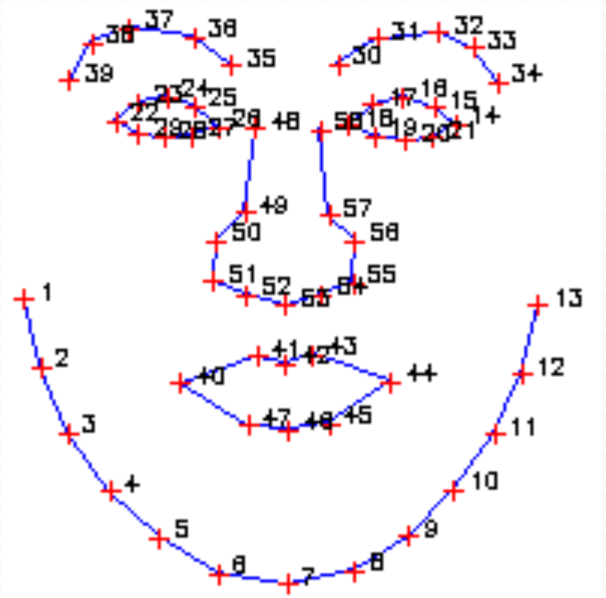
Active Appearance Models

- Active Appearance Models (AAM) is a statistical based template matching method, where the variability of shape and texture is captured from a representative training set.
- Able to extract relevant face information without background interference
- Describes facial characteristics in a reduced model

Shape Model

- Shape is defined as a Set of Landmarks Points
  - Invariant over Euclidian Similarity transformations
  - No landmark connectivity information is given

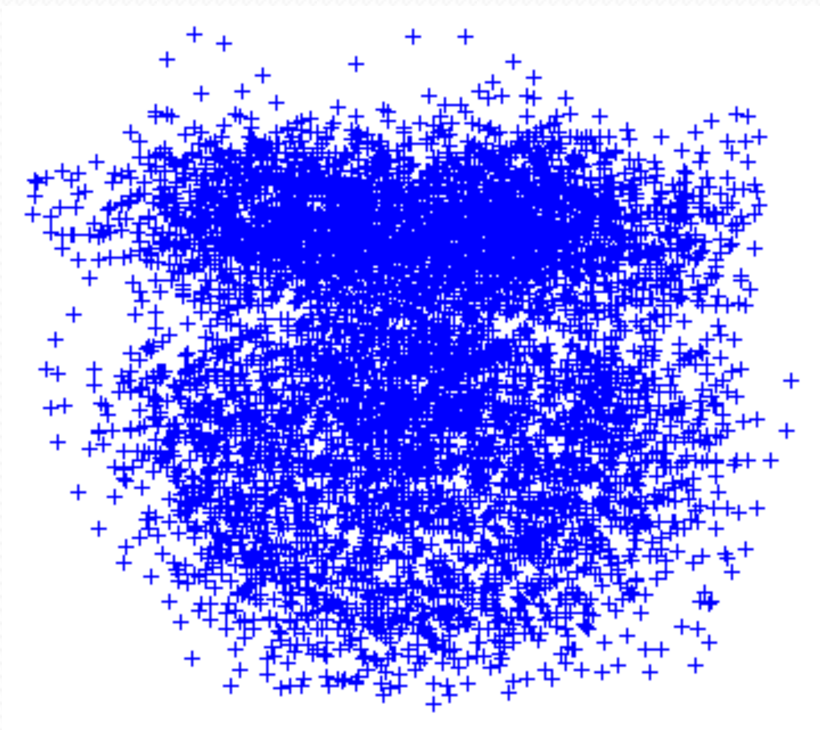
$$x = (x_1, y_1, x_2, y_2, \ldots, x_n, y_n)^T$$



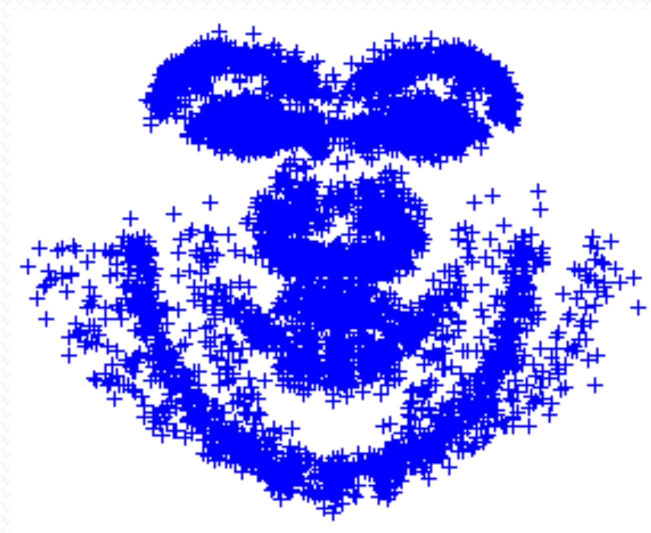
58 Landmark Points Used

Shape Model - Generalized Procrustes Analysis

- Remove location, scale and rotation effects



Raw Data



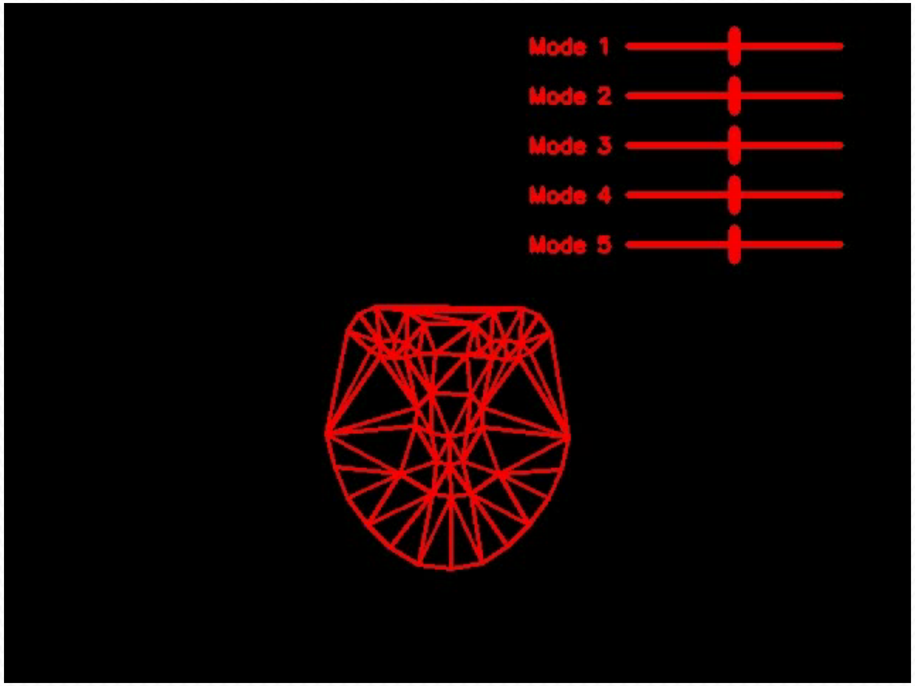
Aligned Data

Shape Model

- Applying a PCA

$$x = \bar{x} + \Phi_s b_s$$

  - x is the sysnthesized shape
  - $\bar{x}$ is the mean shape
  - Φs contains the highest covariance shape eigenvectors
  - bs is a vector of shape parameters representing the weights

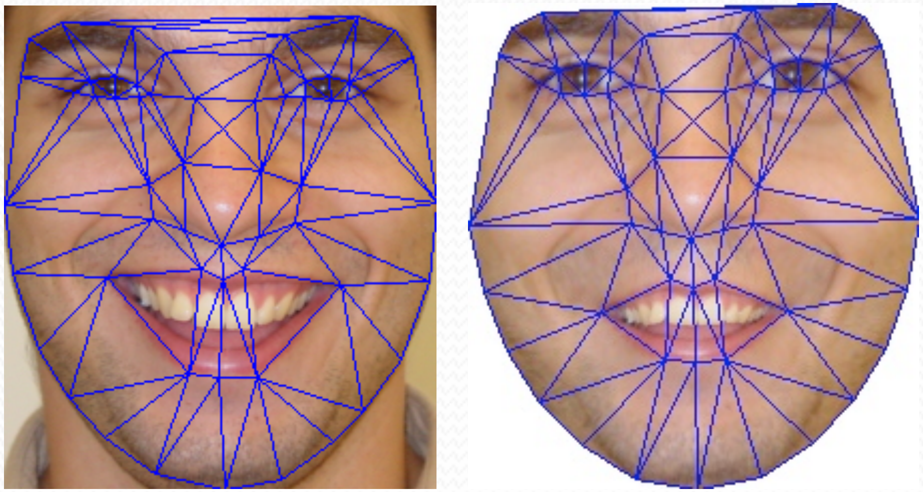


Texture Model

- For *m* pixels sampled, the texture is represented by:

$$g = (g_1, g_2, ..., g_{m-1}, g_m)^T$$

- Required warping each image to a common reference frame



- Delaunay Triangulation
- Each pixel is mapped by barycentric coordinates

Hardware Assisted Texture

- Modern graphics cards provide hardware based solutions
- Texture mapping using OpenGL API
- Delaunay Triangles
- Orthographic Projection Model
- Load warped image from the FrameBuffer

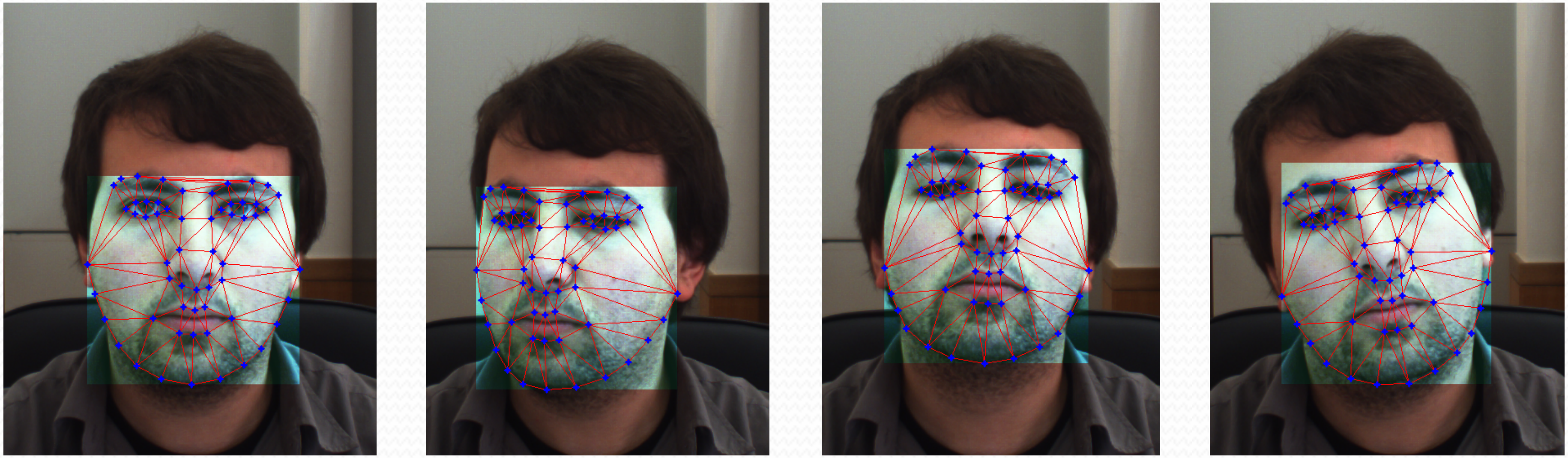
	MatLab	C/C++	OpenGL
Time	2.7 s	200 ms	5 ms

Texture Mapping Video

Texture Mapping Examples

Photometric Normalization

- Histogram Equalization in each of the 3 Color Channels

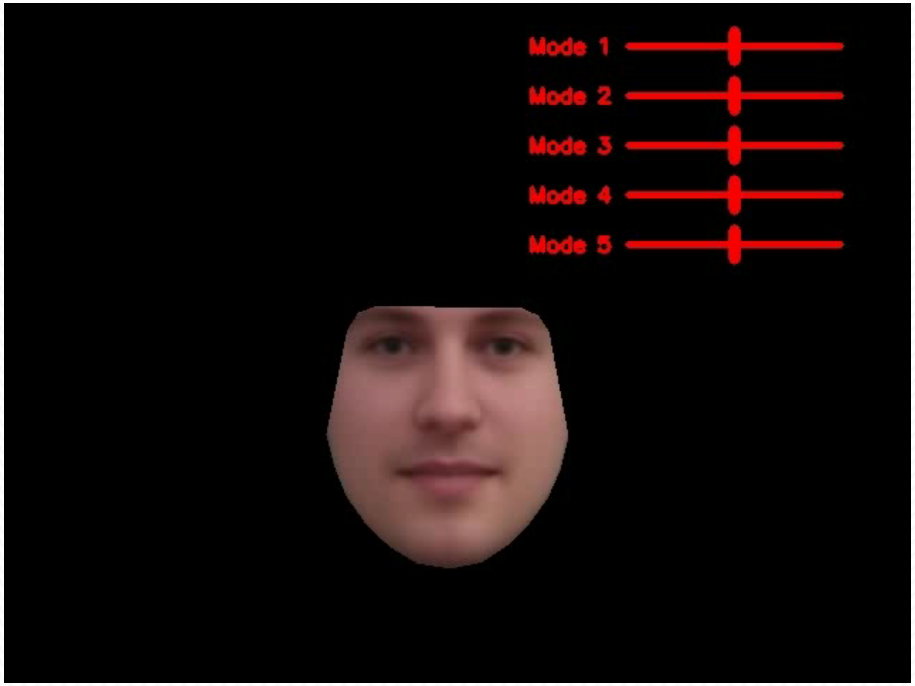


Texture Model

- Applying a LowMemory PCA

$$g = \overline{g} + \Phi_g b_g$$

  - g is the sysnthesized texture
  - $\overline{g}$ is the mean texture
  - Φg contains the highest covariance texture eigenvectors
  - bg is a vector of texture parameters representing the weights



Combined Shape + Texture Model

- To remove correlations between bs and bg a third PCA is performed

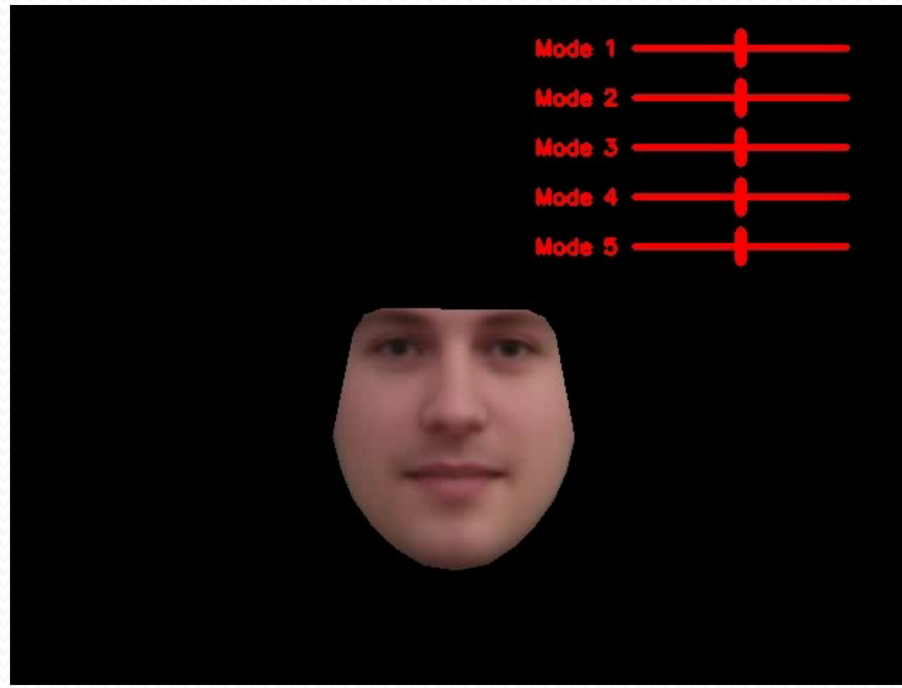
$$b = \begin{pmatrix} W_s b_s \\ b_g \end{pmatrix} = \begin{pmatrix} W_s \Phi_s^T (x - \bar{x}) \\ \Phi_g^T (g - \bar{g}) \end{pmatrix}$$

- Uniformly weight with ratio r

$$W_s = rI \qquad r = \frac{\sum_i \lambda_{gi}}{\sum_j \lambda_{sj}}$$

- Combined model

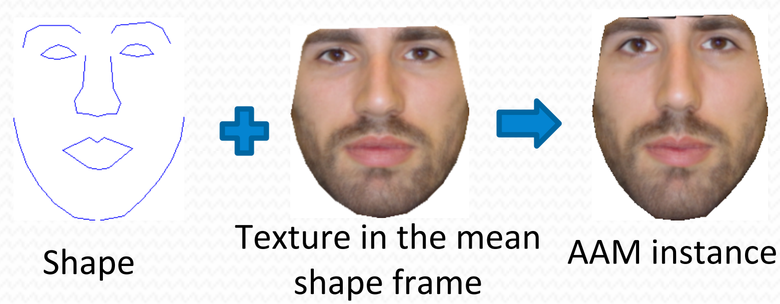
$$b = \Phi_c c$$



Shape: $x = \bar{x} + \Phi_s W^{-1} \Phi_{cs} c$

Texture: $g = \bar{g} + \Phi_g \Phi_{cg} c$

$$\Phi_c = \begin{pmatrix} \Phi_{cs} \\ \Phi_{cg} \end{pmatrix}$$



AAM Instance Examples

AAM Model Training

$$\delta g = r(p)$$

$$\arg\min_c \sum_{pixels} \left(\text{[image 1]} - \text{[image 2]} \right)^2$$

- Optimization Problem
  - Mimimize texture difference between mode and the beneath part of the target image that it covers
- Include pose parameters $t = \begin{bmatrix} S_x & S_y & T_x & T_y \end{bmatrix}^T, S_x = s\cos(\theta) - 1, S_y = s\sin(\theta)$
- Full parameters $p = \begin{bmatrix} c^T & t^T \end{bmatrix}^T$
- Learning the correlations between AAM model instances and texture residuals

$$\Delta p = \begin{bmatrix} \vdots & & \vdots \\ \delta p_1 & \cdots & \delta p_s \\ \vdots & & \vdots \end{bmatrix}_{t_p \times s} \quad \Delta g = \begin{bmatrix} \vdots & & \vdots \\ \delta g_1 & \cdots & \delta g_s \\ \vdots & & \vdots \end{bmatrix}_{m \times s}$$

Find the optimal predition matrix

$$\delta p = R \delta g$$

$$\Delta p = R \Delta g$$

AAM Model Training(2)

Parameter p	Perturbation
c	±0.25σ, ±0.5σ
Scale	90%, 110%
θ	±5, ±10 deg
Tx, Ty	±5%, ±10%

- Residual $r(p) = g_{image} - g_{model}$
- Minimize $|r(p)|^2$
- Expanding in Taylor Series

$$r(p+\delta p) \approx r(p) + J\delta p$$

- So $|r(p+\delta p)|^2$ leads to

$$\delta p = -(J^T J)^{-1} J^T r$$

$$J = \frac{\delta r(p)}{\delta p} = \begin{bmatrix} \frac{\delta r_1}{\delta p_1} & \cdots & \frac{\delta r_1}{\delta p_{t_p}} \\ \vdots & & \vdots \\ \frac{\delta r_m}{\delta p_1} & \cdots & \frac{\delta r_m}{\delta p_{t_p}} \end{bmatrix}_{m \times t_p}$$

$$J = \frac{\delta r}{\delta p} = \Delta g . \Delta p^{-1}$$

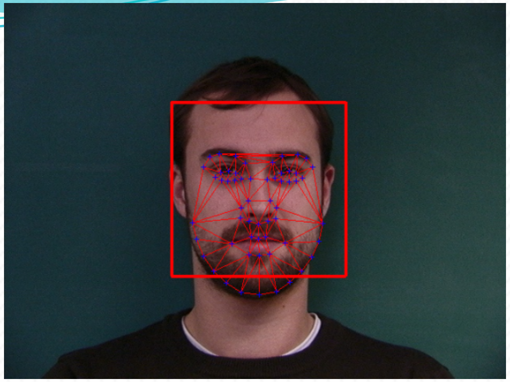
Parameters Displacements

Appearance Displacement

$$\delta p = \begin{bmatrix} \delta c_1 \\ \delta c_2 \\ \vdots \\ \delta c_{t_c - 1} \\ \delta c_{t_c} \\ \delta S_x \\ \delta Sy \\ \delta T_x \\ \delta T_y \end{bmatrix}$$

Pose Displacement

AAM Model Fitting



- AdaBoost Initial Location Estimate
- Damped Gauss-Newton Steepest Descend method

Until No Improvement is made to the error

Sample Image

$(x, y) \text{ ® } g_{image}$

Build AAM Instance

$AAM(p) \text{ ® } (x_{model}, y_{model}, g_{model})$

Compute Texture Residual

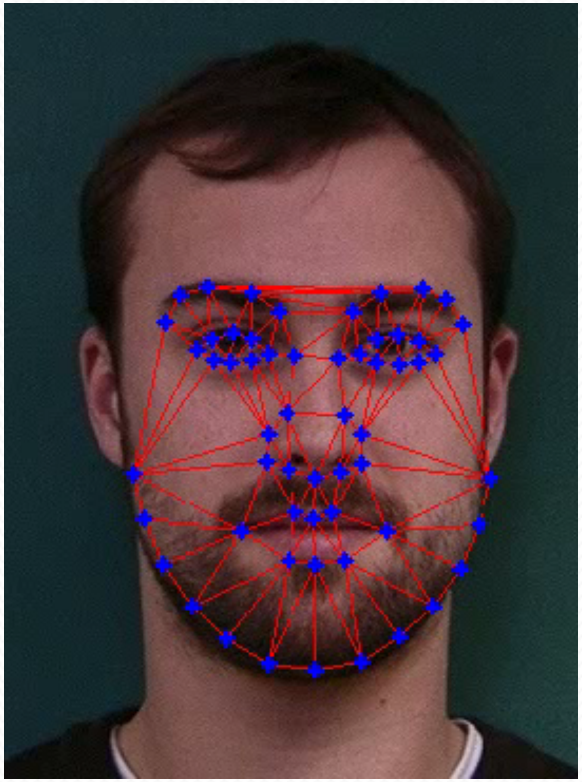
$\partial g = g_{image} - g_{model}$

Update Model Displacements

$p_{k+1} = p_k + \alpha (J^T J)^{-1} J^T \delta g$

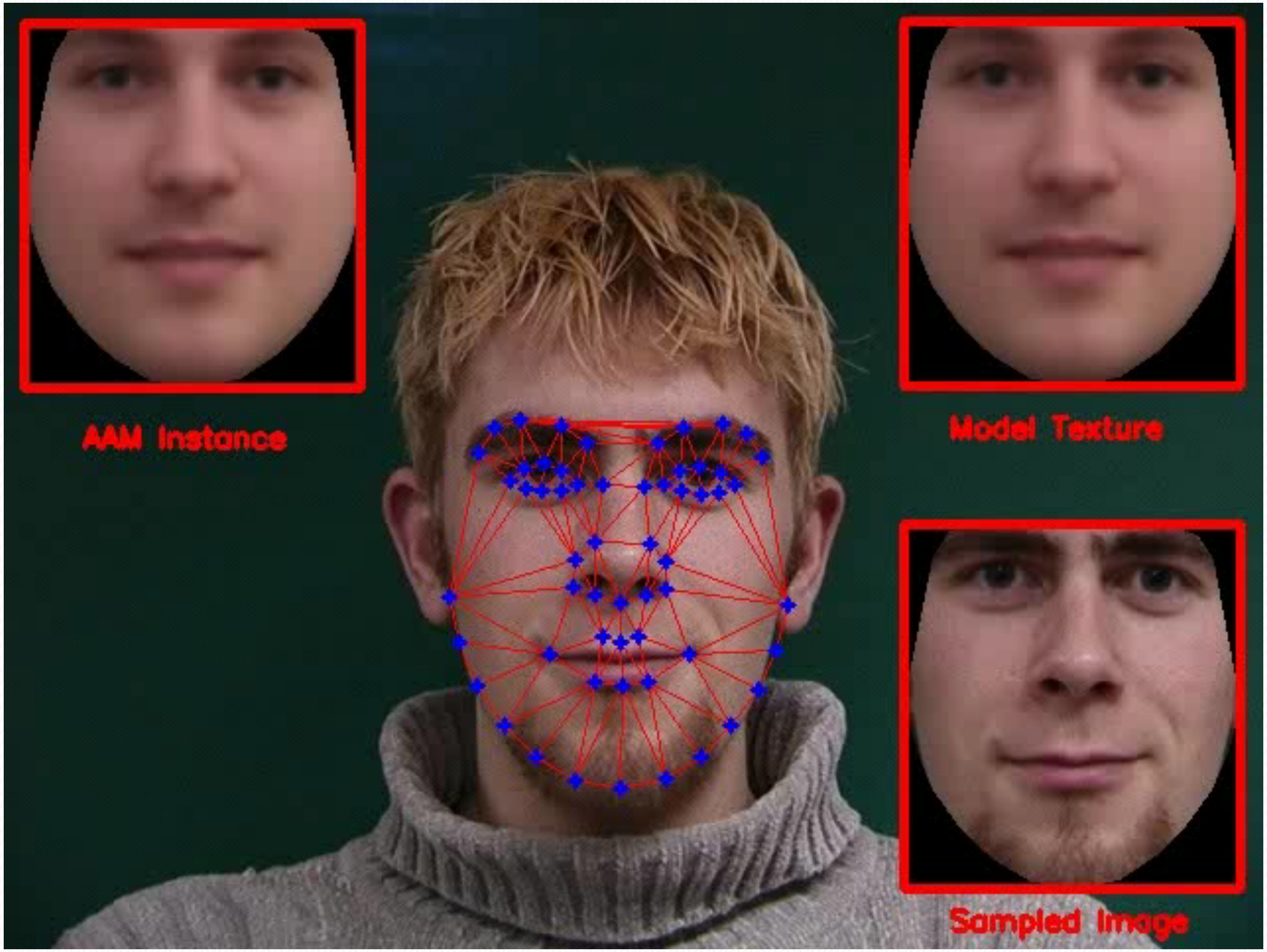
Sampled Instace

Current AAM Instance



Current (x,y) Control Points

IMM Database AAM Fitting

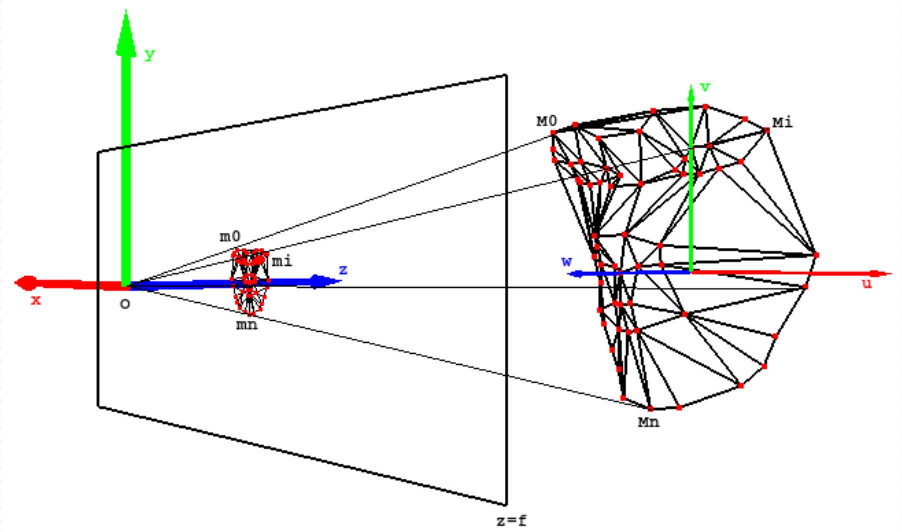


AAM Model Fitting

AAM Model Fitting Failure

Monocular Head Pose Estimation

- Single View Head Pose Estimation
- POSIT - Pose from Orthography and Scaling with ITerations
- Rigid 3D Face Surface Model

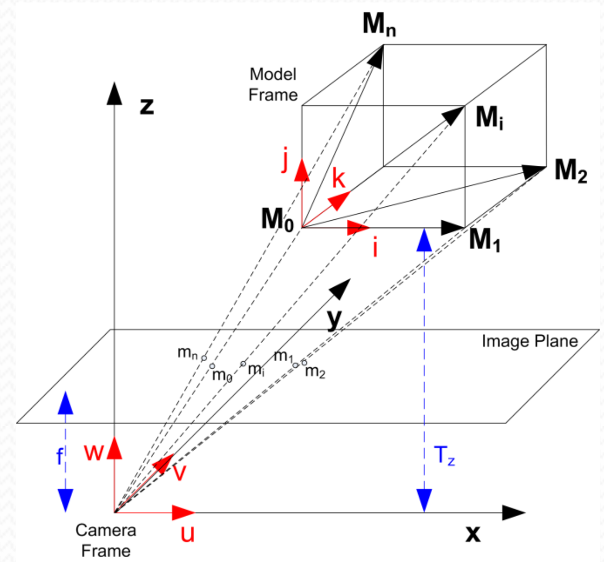


POSIT - Pose from Orthography and Scaling with ITerations

- Perspective Projection Model

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = K \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

(R: the r_{ij} block; T: the T_x, T_y, T_z column)



- Using normalized image coordinates

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = K^{-1} \begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} \Rightarrow \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} R & T \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Defining **r1**, **r2** and **r3** as

$$r_1 = \begin{bmatrix} r_{11} \\ r_{12} \\ r_{13} \end{bmatrix}, r_2 = \begin{bmatrix} r_{21} \\ r_{22} \\ r_{23} \end{bmatrix}, r_3 = \begin{bmatrix} r_{31} \\ r_{32} \\ r_{33} \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} r_1^T & T_x \\ r_2^T & T_y \\ r_3^T & T_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

POSIT - Pose from Orthography and Scaling with Iterations(2)

Dividing all elements by **Tz**

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} r_1^T / T_z & T_x / T_z \\ r_2^T / T_z & T_y / T_z \\ r_3^T / T_z & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$w_i = 1 + \frac{r_3}{T_z}(X_i, Y_i, Z_i)$$

Applying the transpose on the remaining eqs

$$\begin{bmatrix} u & v \end{bmatrix} = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix} \begin{bmatrix} r_1 / T_z & r_2 / T_z \\ T_x / T_z & T_y / T_z \end{bmatrix}$$

Extending for n points

$$\begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \\ \vdots & \vdots \\ u_{n-1} & v_{n-1} \\ u_n & v_n \end{bmatrix} = \underbrace{\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 \\ X_2 & Y_2 & Z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ X_{n-1} & Y_{n-1} & Z_{n-1} & 1 \\ X_n & Y_n & Z_n & 1 \end{bmatrix}}_{M} \begin{bmatrix} r_1 / T_z & r_2 / T_z \\ T_x / T_z & T_y / T_z \end{bmatrix}$$

Until Pose Converge

POSIT Algorithm

Normalize Image Coordinates $u_i = u_i - \frac{c_x}{f}, v_i = v_i - \frac{c_y}{f}$

Compute Model inverse M^{-1}

Assume $w_i = 1$

Get Scaled Orthographic coordinates $(u_i, v_i) = w_i (u_i, v_i)$

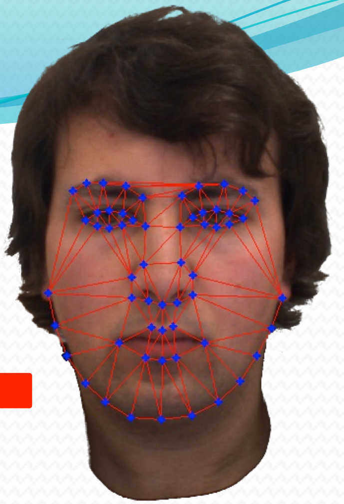
Compute $\begin{bmatrix} r_1 / T_z & r_2 / T_z \\ T_x / T_z & T_y / T_z \end{bmatrix} = M^{-1} \begin{bmatrix} u_1 & v_1 \\ \vdots & \vdots \\ u_n & v_n \end{bmatrix}$

Find **Tz**, **Tx**, **Ty**, **r1** and **r2**

Compute **r3** by the cross product $r_3 = r_1 \times r_2$

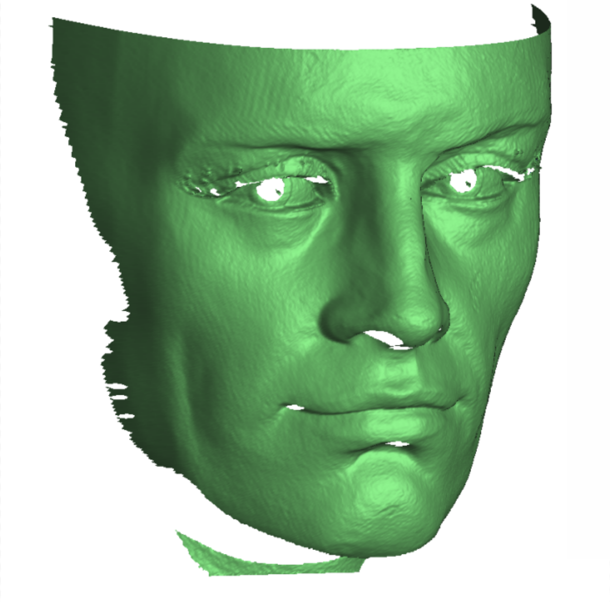
Update $w_i = 1 + \frac{r_3}{T_z}(X_i, Y_i, Z_i)$

3D Anthropometric Model

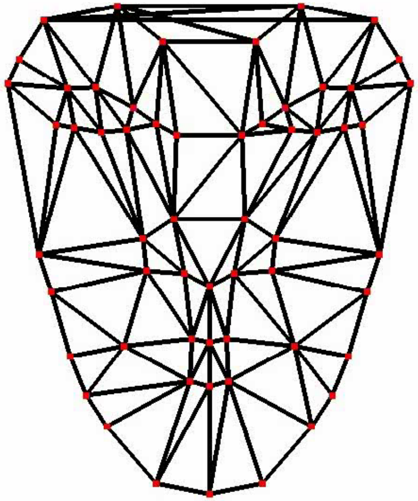


One-to-One 2D/3D
Correspondences

Physical Anthropometric
Model

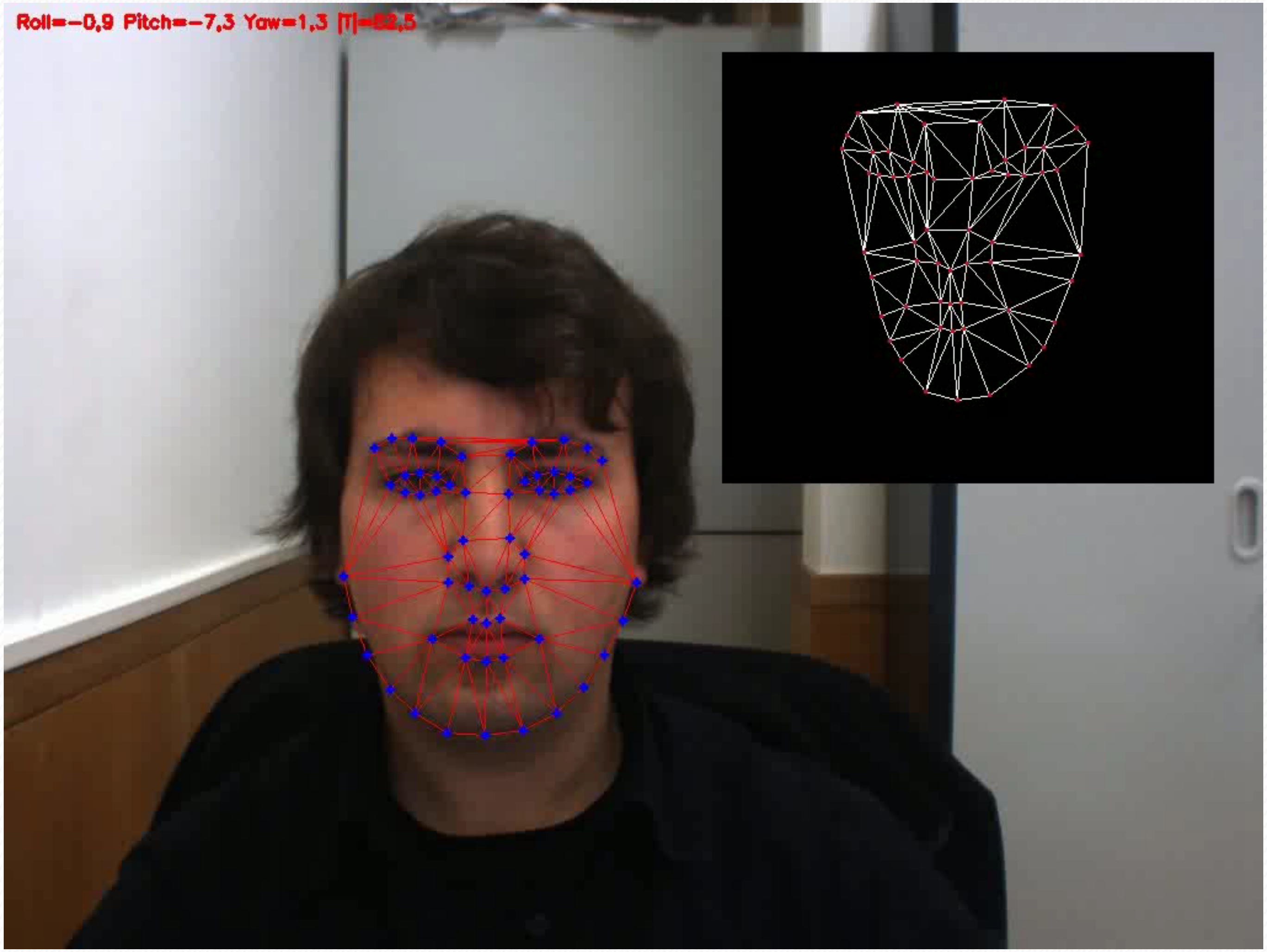


3D laser scan data



Sparse 3D model
(OpenGL)

Head Pose Estimation - Demo

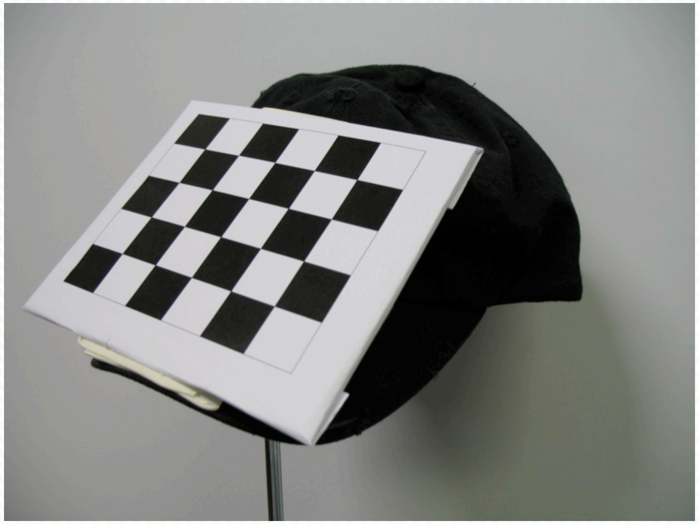


Pose Evaluation – Pose From a Plane

- Knowing the camera matrix, **K**, the Homography holds,

$$H = K[R_1 \mid R_2 \mid T]$$

  - **R1**, **R2** – first 2 columns of rotation matrix **R**
  - **T** – translation vector
- The full pose can be retrieved using the following normalization



The vectors **c**, **p** and **d** are defined as

$$c = R_1 + R_2 \qquad p = R_1 \times R_2 \qquad d = c \times p$$

$$R_1' = \frac{1}{\sqrt{2}}\left(\frac{c}{|c|} + \frac{d}{|d|}\right) \qquad R_2' = \frac{1}{\sqrt{2}}\left(\frac{c}{|c|} - \frac{d}{|d|}\right) \qquad R_3' = R_1' \times R_2'$$

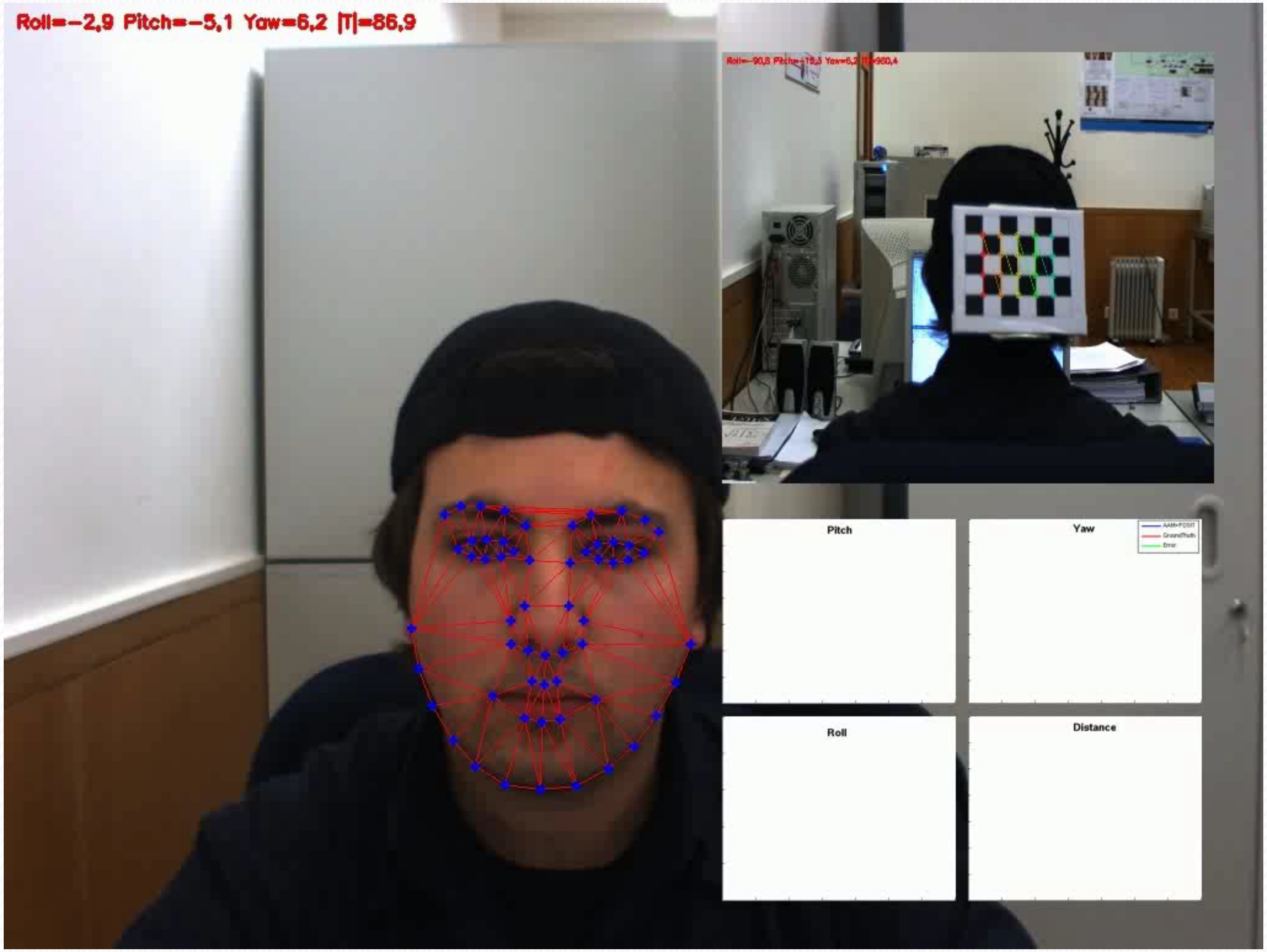
$$R = [R_1' \mid R_2' \mid R_3']$$

Compute $W = K^{-1}H$

$$R_1 = \frac{W_1}{l} \qquad R_2 = \frac{W_2}{l} \qquad T = \frac{W_3}{l}$$

$$l = \sqrt{|W_1||W_2|}$$

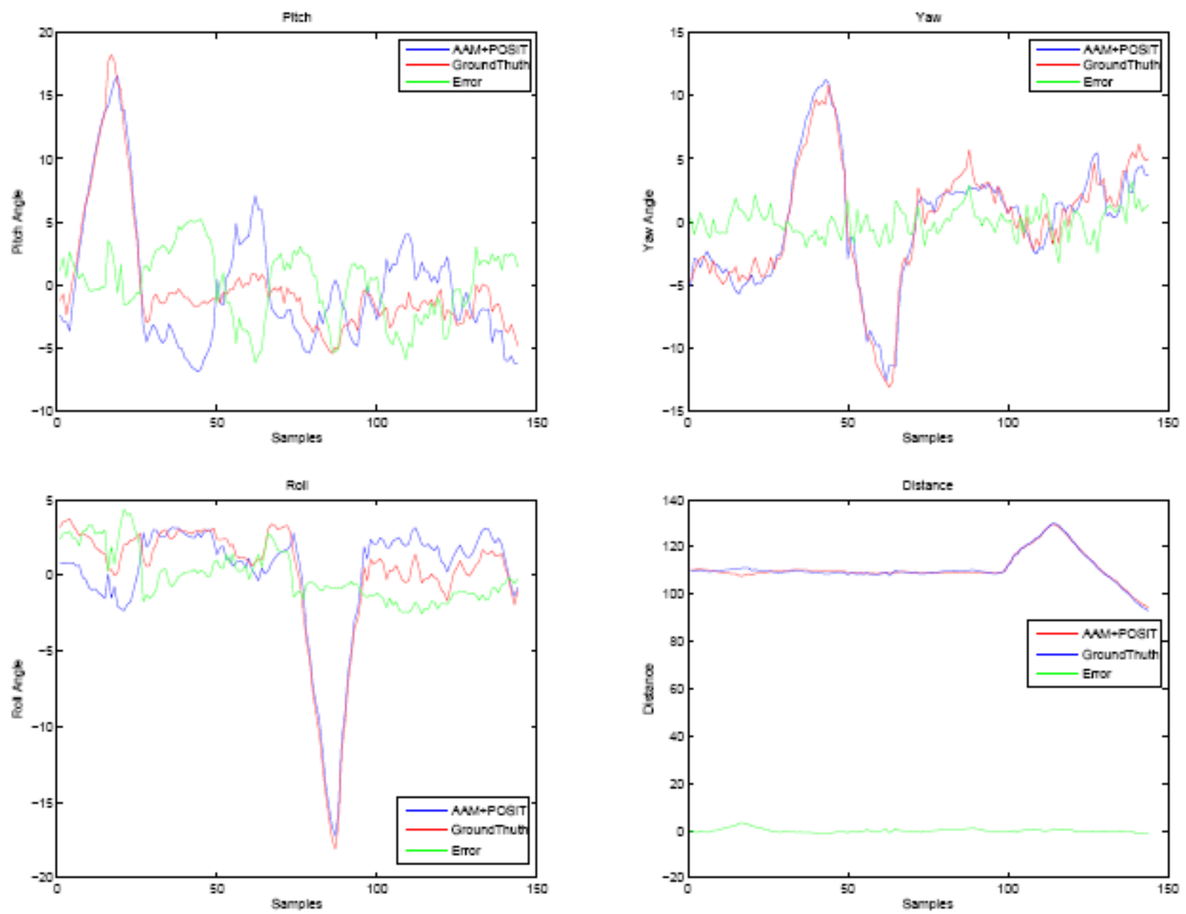
Pose Estimation Evaluation - Demo



Pose Estimation Evaluation

- AAM+POSIT Head Pose Compared with a planar checkboard pose

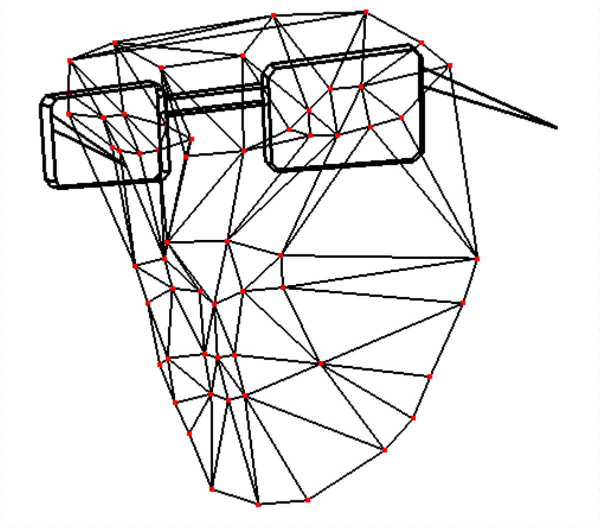
Parameters	Avg std
Roll	1.94 deg
Pitch	2.57 deg
Yaw	1.7 deg
Distance	1.33cm



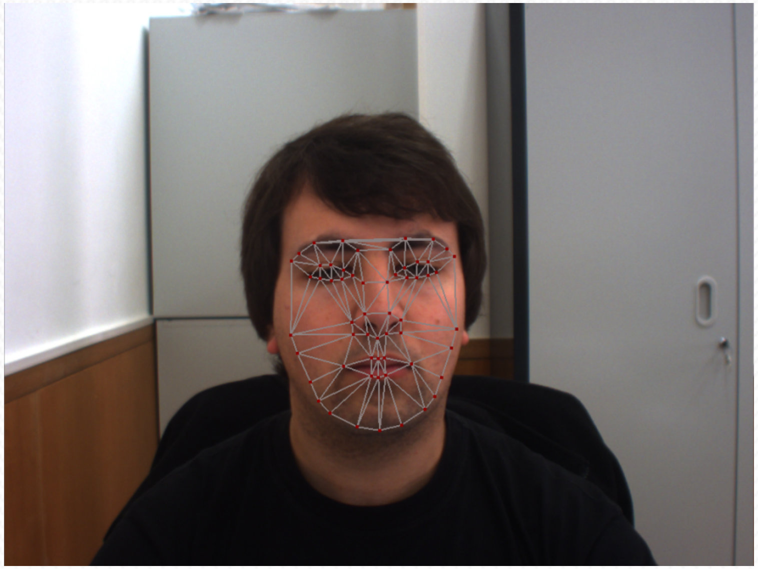
Correlations between Pitch and Yaw angles

3D Glasses Augmentation

- Augmented Reality (AR) is the overlay of artificial computer graphics images on the physical world



3D Glasses drawn with Respect to the Head Model



3D anthropometric model overlaid

3D Glasses Augmentation - Demo

Final Notes

Advantages

- Single View Solution to estimate the 6DOF Head Pose
- Combines AAM Features Extracting + POSIT Pose Estimation
- Easy 2D/3D image registration
- Average std errors in about 2 degree in orientation and 1 cm in position

Weeknesses

- Rigid 3D Head Model
- Identity Differences
- Facial Expression Influence