Statistical models of shape and apperance

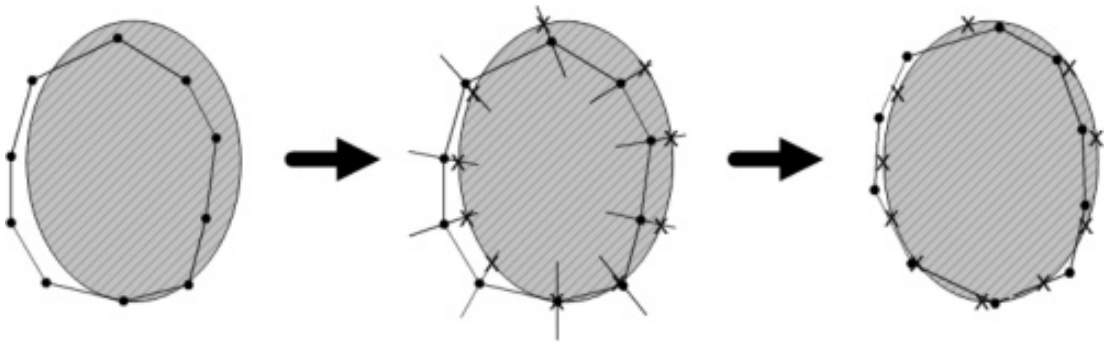
Ing. Marek Hrúz, Ph.D.

Katedra Kybernetiky
Fakulta aplikovaných věd
Západočeská univerzita v Plzni

DEPARTMENT OF CYBERNETICS

Statistical models of shape

- they are used for object representation in images
- statistical analysis is applied to data representing a certain shape with the goal of finding its model
- there are two main methods that differ in the approach to image data
  - Active Shape Model (ASM)... uses only information about the shape of the object
  - Active Appearance model (AAM)... adds information about texture

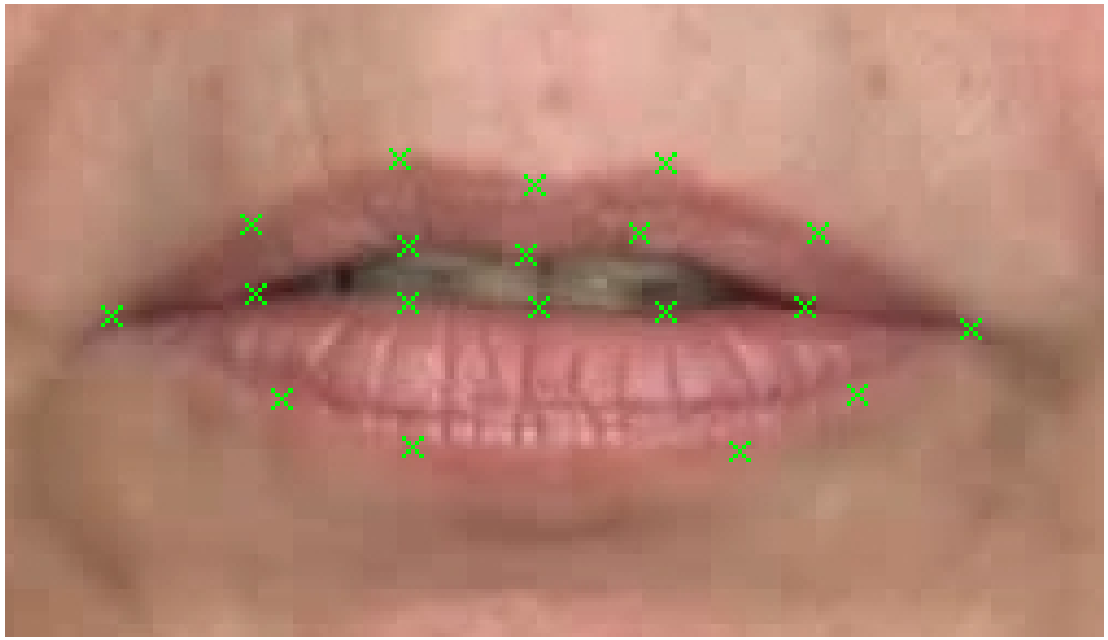


DEPARTMENT OF CYBERNETICS

Landmarks - interesting points

- there are several possibilities to represent a shape in 2D image - one of them is a list of **landmarks** - corners, strong edges
- for statistical models of shape it's good to consider discriminative locations (landmarks) in the image and the locations should be visible in the whole training dataset
- eg. tips of fingers, corners of mouth
- the positions (x, y) are used for statistical analysis
- the shape of the object is obtained by suitably connecting the positions
- to determine the dependence of the movement of individual landmarks ... Principal Component Analysis - PCA

Example of landmark selection for the shape of lips



Principal Component Analysis (PCA)

- PCA is an orthogonal transformation of features (that are correlated) into new features that are expressed as a linear combination of uncorrelated **principal components**
  1. used for data decorrelation, known from year 1901
  2. used for dimension reduction with the ability to control the amount of lost information
- the calculation is based on the covariation of measured data $\mathbf{x_1}, \mathbf{x_2}, \ldots \mathbf{x_n}$, with dimension p

$$C(i,j) = \mathrm{cov}(x_i, x_j) = \mathrm{E}\left[(x_i - \mu_i)(x_j - \mu_j)\right] \qquad (1)$$

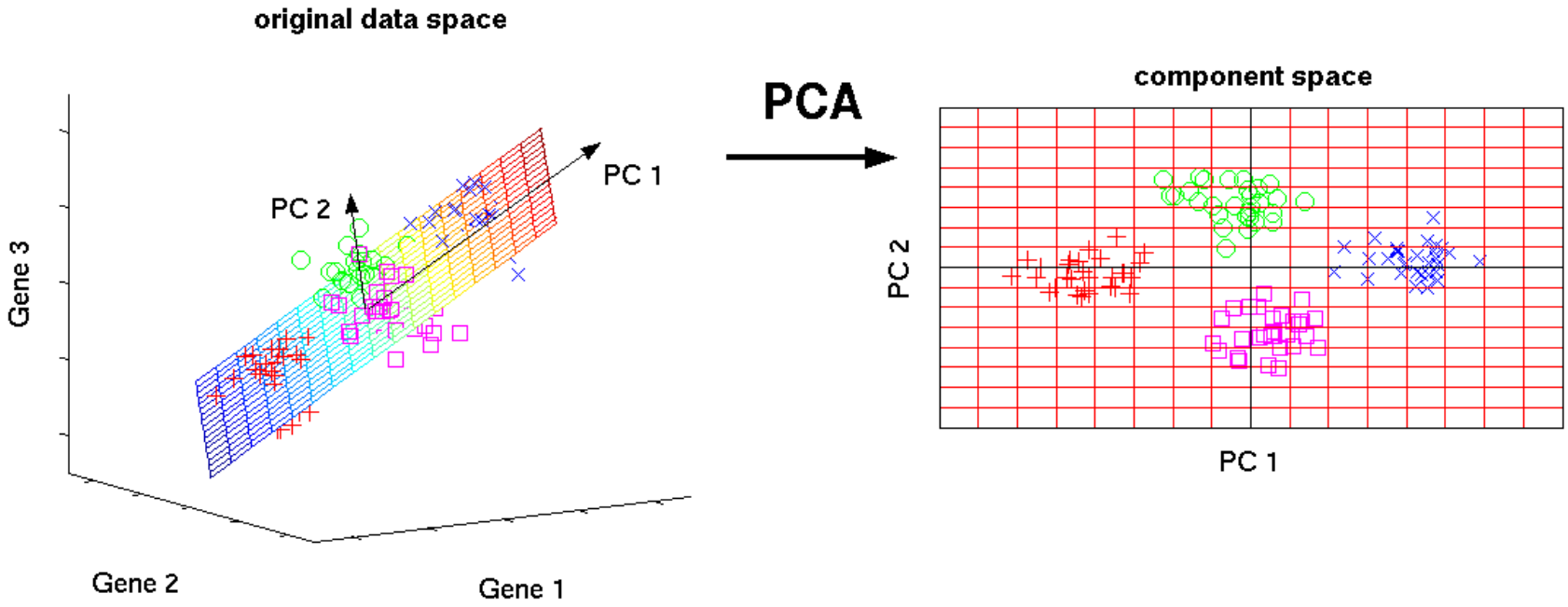
- $C(i,j)$ is an element of the covariance matrix $\mathbf{C}$, x_i and x_j are i-th and j-th dimension of the data, and μ_i and μ_j are the mean values in the dimensions

DEPARTMENT OF CYBERNETICS

- from the covariance matrix we can compute a matrix of eigenvectors $\mathbf{V}$ and vector of eigenvalues using the eigenvalue decomposition
- $\mathbf{VDV}^{-1} = \mathbf{C}$, where $\mathbf{D}$ is a diagonal matrix with eigenvalues on the diagonal
- if the values in $\mathbf{C}$ are real (and by definition the matrix is symmetric) we can write $\mathbf{VDV}^{\mathbf{T}} = \mathbf{C}$
- the principal components are the columns of matrix $\mathbf{V}$
- the eigenvalues represent the importance of the components
- the relative value of the eigenvalue is a percentual expression of the importance
- we can reduce the dimensionality of the data and retain a certain percentage of variance of the data (eg. 99% or 95%)

Note - the percentual values of the eigenvalues are added up (from highest to lowest) until we get the desired percentage and the rest of the eigenvectors are not considered

Example of dimension reduction from 3 to dimension 2



Summary - properties of transformed data

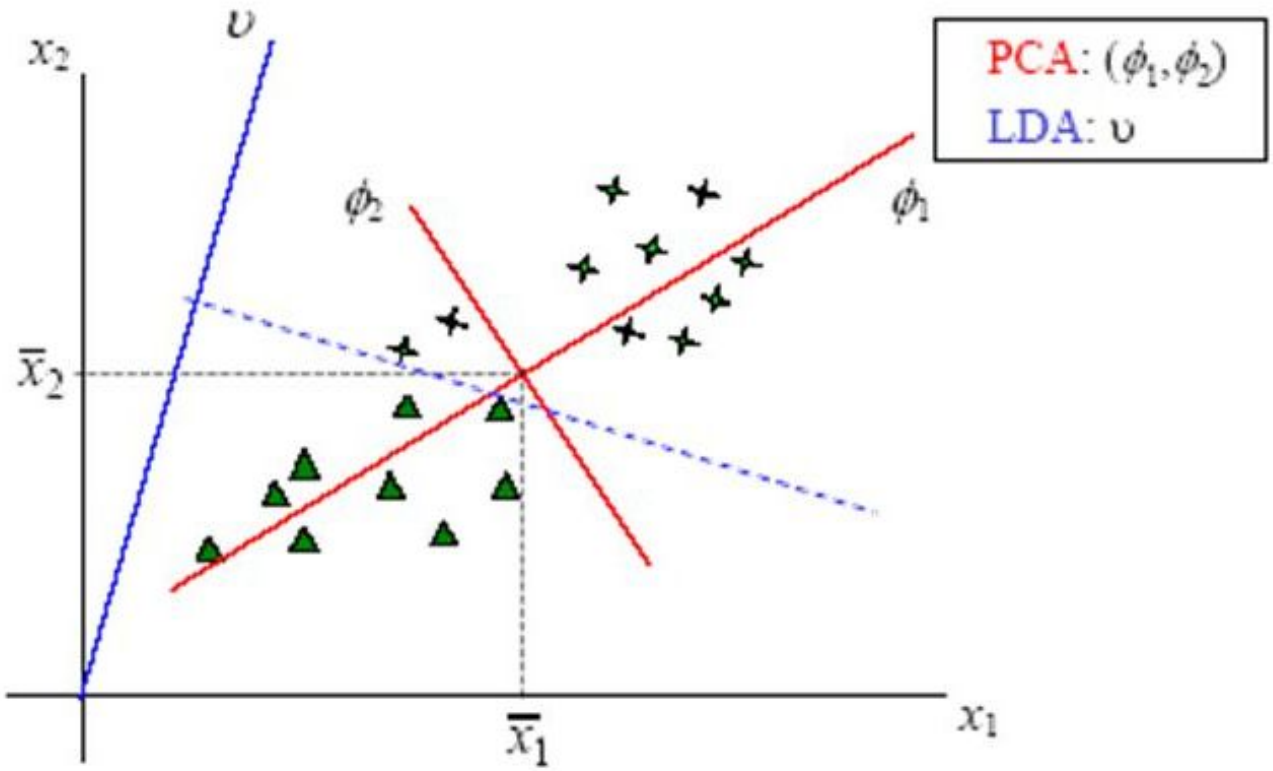
- most of the information about the variability of data is concentrated in the first component and the least information is in the last component
- we apply dimension reduction without loosing too much information $\rightarrow$ we use only several leading eigenvectors
- the unused components contain small amount of information (eg. noise) because their variability is small
- mathematically, the reduction is applied by ignoring the last columns of matrix $\mathbf{V}$, thus obtaining a rectangular matrix from a square matrix and transforming the data with this reduced matrix

Linear discriminative analysis (LDA)

- LDA is a method of multidimensional statistical analysis
- it is applied to detect a linear combination of features that discriminate a given labeled dataset
- LDA is also used for dimension reduction
- comparison:
  - **PCA** defines directions with **largest variance** from the original space (the ones that describe the data the best)
  - **LDA** defines direction that **best discriminate** the data given their classes

Note - LDA is used frequently in tasks like face recognition - reduction of dimensionality

Example of LDA and PCA reduction (in 2D)



Linear model of shape

- is based on PCA
- the shape of the object in the image is defined via landmarks
- the shape can have different representations eg. - $\mathbf{x_i} = [x_1, x_2, \ldots, x_n, y1, y2, \ldots, y_n]$, where $[x_i, y_i]$ is the location of i-th landmark
- the shape is generated from the model:

$$x = \overline{x} + \Phi_s b_s \tag{2}$$

- Φ is a matrix of eigenvectors obtained from covariance matrix of training shapes
- $\overline{x}$ is the mean of training data
- b_s is a parameter vector of the model
- x is the generated shape
- $\rightarrow$ the generated shape is controlled by the shape parameters

Fitting of the model to the image

- the shape generated via the linear model has only local characteristics (it has zero mean)
- the global positions of the shape use another set of parameters - pose parameters - translation, rotation, and scale
- all together can be written into a transformation matrix T

$$x = T_{X_t, Y_t, s, \theta}(\overline{x} + \Phi_s b_s) \tag{3}$$

- for a single landmark (x, y) we can write the transformation as:

$$T_{X_t, Y_t, s, \theta}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} X_t \\ Y_t \end{pmatrix} + \begin{pmatrix} s\cos\theta & s\sin\theta \\ -s\sin\theta & s\cos\theta \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} \tag{4}$$

- we are looking for the pose and shape parameters of the model so that we generate a shape **x** that is correspond to the object's point Y

$$\min_{X_t, Y_t, s, \theta, b_s} ||Y - T_{X_t, Y_t, s, \theta}(\overline{x} + \Phi_s b_s)||^2 \tag{5}$$

DEPARTMENT OF CYBERNETICS

The algorithm of the fitting

1. Set parameters $b_s = 0$
2. Compute the shape using formula $x = \overline{x} + \Phi_s b_s$
3. Find the pose parameters $t = (X_t, Y_t, s, \theta)$ that minimize $||Y - T_{X_t, Y_t, s, \theta}(\overline{x} + \Phi_s b_s)||^2$
4. Project the points Y into the local coordinate system of the shape

$$y = T^{-1}_{X_t, Y_t, s, \theta}(Y) \tag{6}$$

5. Update the shape parameters

$$b'_s = \Phi_s^T (y - \overline{x}) \tag{7}$$

6. If the distance of b'_s and b_s are large, set b_s to b'_s and go to step 2

Linear models of appearance

- it is en extension of shape models by the description of appearance (texture) inside the shape
- it is necessary to know the landmarks beforehand
- the texture inside the annotated shape is transformed into a texture of the mean shape using *warping* $\rightarrow$ we obtain a shape independent texture
- PCA can be applied to the texture similarly as to the shape

DEPARTMENT OF CYBERNETICS

Image Warping

- warping is a mapping of texture into a desired shape
  . . . modification of the shape of the texture
- the easiest way is so called *triangulation of coordinates*
- both shapes are first divided into same triangles (vertexes of the triangles are the same landmarks in both shapes)
- for every triangle in the target shape we find a corresponding triangle in the source shape
- for every pixel of the target triangle we find a corresponding pixel in the source triangle
- the location of a pixel in the triangle can be expressed in relation to its vertexes

$$\begin{bmatrix} x \\ y \end{bmatrix} = \alpha \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \beta \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} + \gamma \begin{bmatrix} x_3 \\ y_3 \end{bmatrix} \tag{8}$$

- x_1,x_2 a x_3 are the vertexes of the triangle and x is the location of the desired pixel

DEPARTMENT OF CYBERNETICS

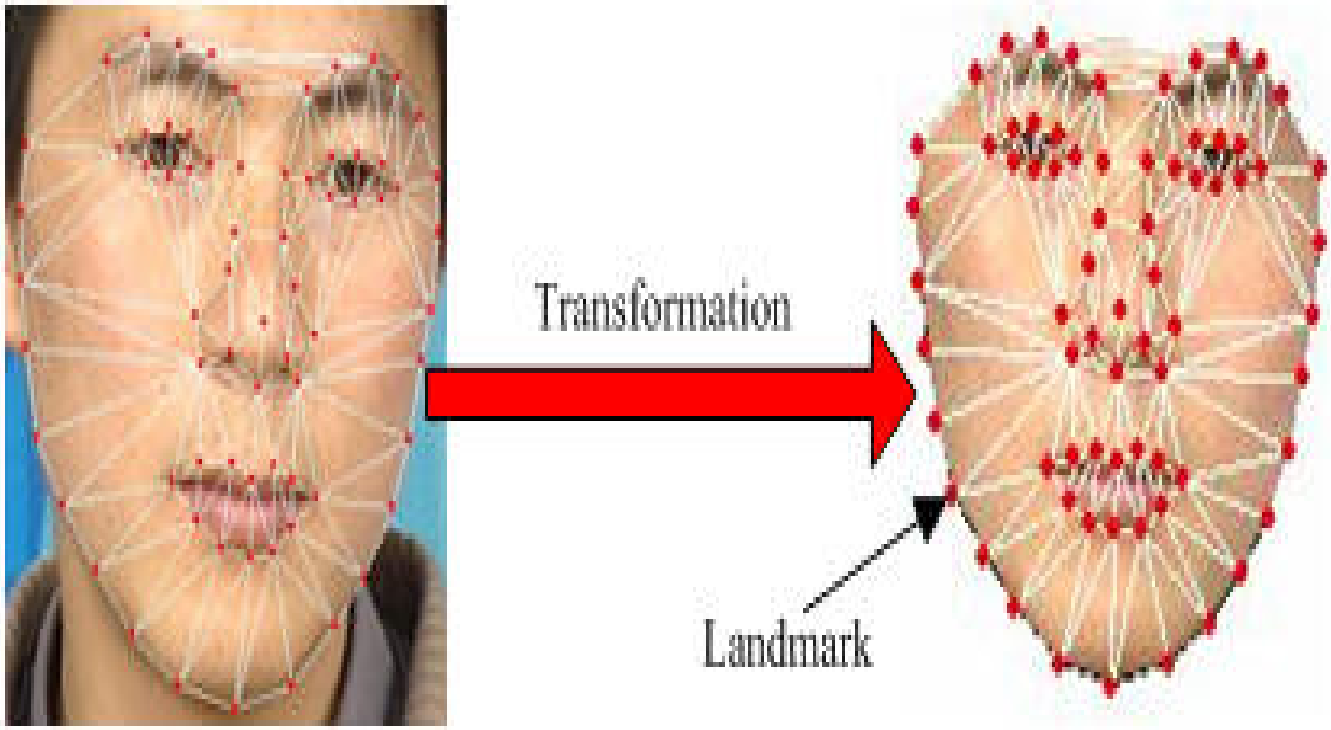
- the parameters α,β a γ can be computed from equations:

$$\alpha = 1 - (\beta + \gamma) \tag{9}$$

$$\beta = \frac{yx_3 - yx_1 - x_3y_1 - xy_3 + x_1y_3 + xy_1}{-x_2y_3 + x_2y_1 + x_1y_3 + x_3y_2 - x_3y_1 - x_1y_2} \tag{10}$$

$$\gamma = \frac{xy_2 - xy_1 - x_1y_2 - yx_2 + x_2y_1 + yx_1}{-x_2y_3 + x_2y_1 + x_1y_3 + x_3y_2 - x_3y_1 - x_1y_2} \tag{11}$$

- x and y are the coordinates of the target pixel (from the mean shape), x_i and x_j are the coordinates of the triangle (in the mean shape)
- we find the location of the pixel in the input image by applying the computed α, β, γ into the Eq. 8


Transformation
Landmark
Labeled Face
Mean Shape

Linear model of texture

- by applying PCA to the warped texture we can generate the linear model of texture:

$$g = \overline{g} + \Phi_g b_g \tag{12}$$

- Φ_g is the matrix of eigenvectors from the training textures
- $\overline{g}$ is the mean vector of texture
- b_g is a vector of the texture parameters and g is the computed texture

DEPARTMENT OF CYBERNETICS

Combined model

- to create the combination model we define the dependence between shape and texture parameters
- first we compute parameters for the individual parameters

$$b = \begin{pmatrix} W_s b_s \\ b_g \end{pmatrix} = \begin{pmatrix} W_s {\Phi_s}^T (x - \overline{x}) \\ {\Phi_g}^T (g - \overline{g}) \end{pmatrix} \tag{13}$$

- b is a vector composed of shape and texture parameters
- W_s is a weighting matrix . . . normalization of shape parameters against the texture paramters (shape vs. texture)
- Φ_s a Φ_g are the eigenvector matrices of the training covariance matrices
- x are the real coordinates of landmarks in the current image
- g is the shape independent texture from a given image

- then by applying another PCA on the combined parameter vector b we obtain the combined model

$$b = \Phi_c c \tag{14}$$

- Φ_c are the eigenvectors of the covariance matrix of the combined model
- c is the vector of combined model parameters
- using the combined parameters c we can control both the shape and texture of an object

Active Shape Model (ASM)

- **ASM** is the most popular algorithm for fitting the statistical shape models
- ASM iteratively deforms its shape $\rightarrow$ **finding the best fit** between the model and the observed image
- at the beginning there is a coarse shape estimation - we use the mean shape
- an instance of the shape is generated

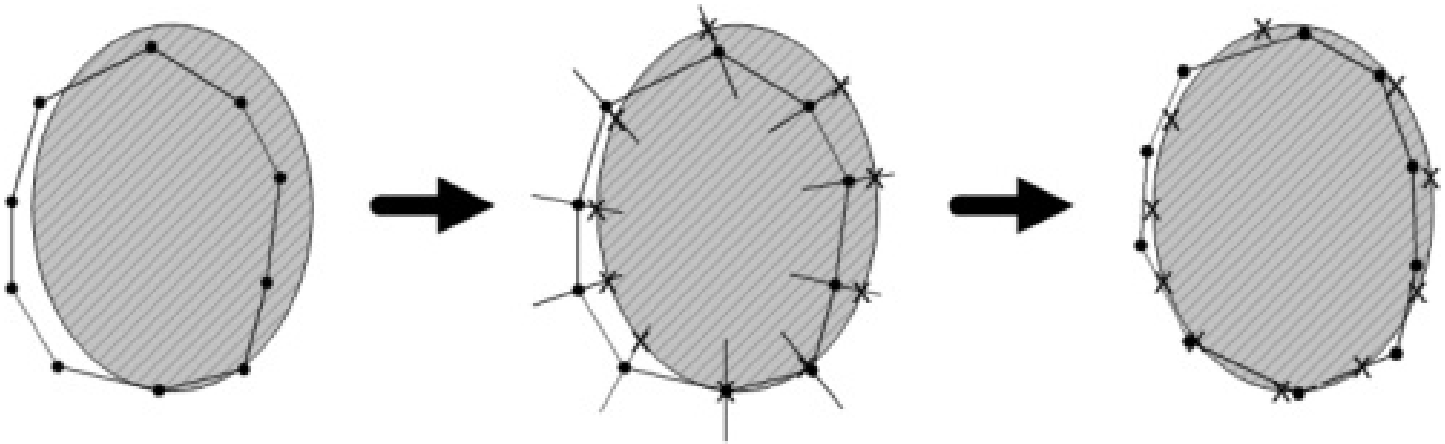
$$x = T_{X_t, Y_t, s, \theta}(\overline{x} + \Phi b) \tag{15}$$

- for the fitting we use following iterative algorithm:

DEPARTMENT OF CYBERNETICS

Algorithm

1. Analyze the neighborhood of each landmark $\mathbf{x_i}$ and find the best possible location $\mathbf{x_i'}$ where the point $\mathbf{x_i}$ should move - eg. the profile modeling
2. Update the parameters (X_t, Y_t, s, θ, b) so that they fit the best on the new points $\mathbf{X}'$
3. Repeat the process until convergence



DEPARTMENT OF CYBERNETICS

ASM for tracking movement of a human face, example of convergence for one image

Active Appearance Model (AAM)

- **AAM** is the most popular algorithm for fitting the combined model (shape+texture)
- the criterion for the fitting is given by the difference:

$$\delta g = g_s - g_m \tag{16}$$

- where g_s is a vector of image brightness transformed to the texture model frame and g_m is a vector of brightness of the generated shape and texture

DEPARTMENT OF CYBERNETICS

- to find the best fit we minimize the quantity $\Delta = |\delta g|^2$ by updating the parameters of combined model c
- the problem might seem hard because of the large parametric space
- the solution is that we train a model of the fitting - for different values of the vector δg we can express the desired update of the parameters c from the training set

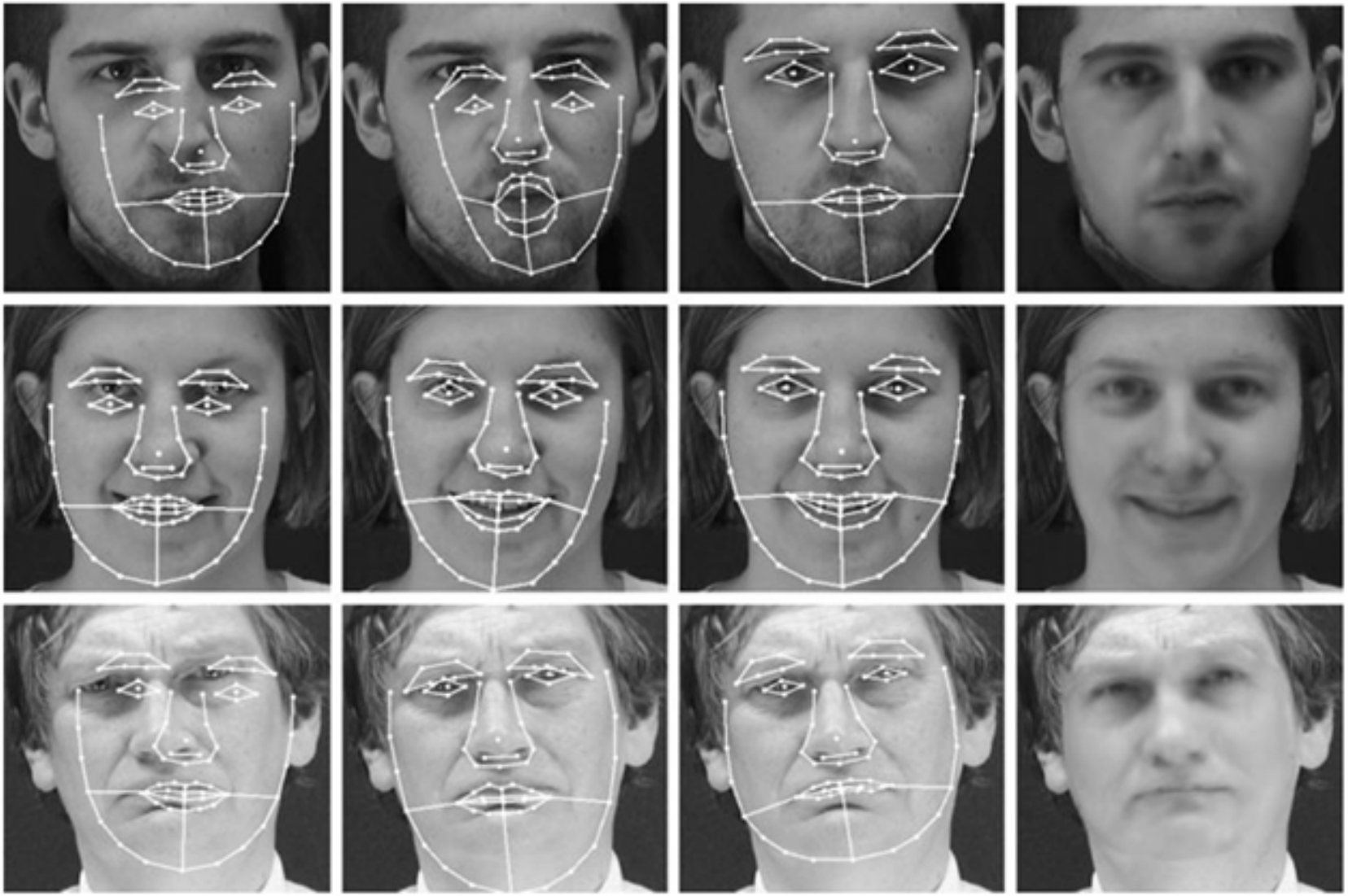
During training we systematically generate the vector g_m by displacing ground truth model parameters c^* and pose parameters (X_t, Y_t, s, θ) - jointly denoted $\mathbf{p}$. Using LSM we compute a transformation matrix R that is needed for fitting.

DEPARTMENT OF CYBERNETICS

Optimizing AAM parameters

- Project the texture sample into the model frame obtaining g_s
- Compute the error vector $\delta g = g_s - g_m$
- Compute the error $E = |\delta g|^2$
- Compute the predicted displacements, $\delta p = R\delta g$
- Update the model parameters $p = p + k\delta p$, where initially $k = 1$
- Calculate the new points $\mathbf{X}'$ and model frame texture g'_m
- Sample and project the new texture into model frame g'_s
- Calculate a new error vector $\delta g' = g'_s - g'_m$
- If $|\delta g'|^2 < E$, then accept new estimate; otherwise try at $k = 0.5, k = 0.25$, etc

AAM for face detection



DEPARTMENT OF CYBERNETICS