Computer Vision

Segmentation: boundary based techniques, continued

Model based approaches in boundary detection

Boundary based segmentation

- Deformable templates
- Snakes
- Active Countours
- Active Shape Models

Deformable templates

- Parametric shape models with few degrees of freedom r(s,X)
- Match the template to an image by searching for the value of X that minimizes an external energy term, $E_{ext}(X)$
- An internal energy term, $E_{int}(X)$ can be included as a "regularizer" .

Snakes

- We are given a feature map, F(r)
- Treat - gradient of F(r) as a landscape on which the snake, r(s) 0<s<1 can slither
- Maximize grad F(r(s)) while retaining smoothness of the snake

$$\underbrace{\left(\frac{w_1\partial r}{\partial s}-\frac{w_2\partial^2 r}{\partial s^2}\right)}_{\text{Internal forces}}+\underbrace{\nabla F}_{\text{external force}}=0$$

∇F : spatial gradient of F

w_1 : elasticity

w_2 : stiffness

Snakes - Discretization

- Sparse set of equations:

$$w_1 r_s(s_1) - w_2 r_{ss}(s_1) + \nabla F(s_1) = 0$$

$$w_1 r_s(s_2) - w_2 r_{ss}(s_2) + \nabla F(s_2) = 0$$

$$\vdots$$

- To find out what the curve looks like in between, use polygonal approximation or splines
- If B-splines are used, the smoothness term may be omitted

Active Contours

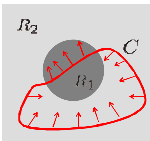
- Feature map F(t) and active contour r(s,t) both vary in time

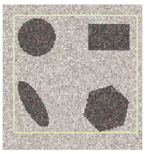
$$\rho r_{tt} = -\left(\gamma r_t - w_1\frac{\partial r}{\partial s} + w_2\frac{\partial^2 r}{\partial s^2}\right) + \nabla F$$

ρ : mass density

γ : viscous resistance from the medium surrounding the snake

Curve Evolution (Active Contours formulated as a Bayesian problem)





- Formulate as an optimization problem for the curve C
$$\hat{C} = \arg\min_{C} \{E(C)\}$$
- Solve iteratively by "evolving" the curve
$$\hat{C}^{(n+1)} = \mathcal{F}(\hat{C}^{(n)})$$
- How to choose the objective functional $E(C)$?

Active Shape Models Overview

- The algorithm

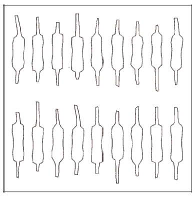
Training
- – Manually landmark shape points
- – Align the training set
- – Learn the shape variability
  - » May now generate new shapes

ASM
- – Iterate for optimizing the translation, rotation, scale parameters and also the shape parameters with an optimization algorithm (based for example on gradients)

ASM Overview

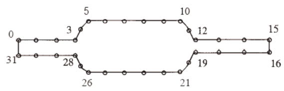
- Training set of shapes

Different resistor shapes



ASM Overview

- We wish to learn the variability of the shapes.
  - ➔ Define Landmark Points (N)



Each landmark point should be in correspondence!!

ASM Overview

- The shapes in the training set may be of different rotation, scale and translation.

➔They must be aligned to the same coordinate frame.

Generalized Procrustes Method

Iterative Least Square Optimisation for translation,rotation,scale parameters.

PROCRUSTES STEPS

1. Find centers of gravity $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$.
2. Form displacements

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}}, \quad \tilde{\mathbf{y}}_i = \mathbf{y}_i - \bar{\mathbf{y}}.$$

3. Form the matrix $\mathbf{A} = \sum_{i=1}^{n} \tilde{\mathbf{x}}_i \tilde{\mathbf{y}}_i^T$
4. Obtain SVD $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$
5. Take $\mathbf{Q} = \mathbf{V}\mathbf{U}^T$
6. Take $\mathbf{t} = \bar{\mathbf{y}} - \mathbf{Q}\bar{\mathbf{x}}$

ASM Overview

- Apply PCA to the training set.
  - Consider each training shape as a point in 2N space.
  - Apply PCA to the points (training shapes) in this 2N dimensional space to learn the shape variability.
  - Generate new shapes by

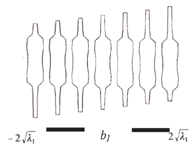
$$x = \bar{x} + Pb$$

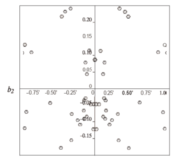
where P: matrix of eigenvectors,
b: weights

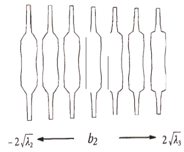
ASM Overview

- New shapes

Eigenvalue	$\frac{\lambda_i}{\lambda_T} \times 100\%$
λ_1	66%
λ_2	8%
λ_3	5%
λ_4	4%
λ_5	3%
λ_6	3%

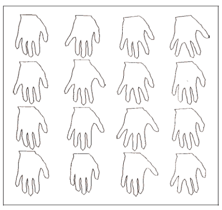
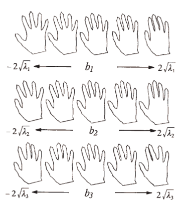






ASM Overview

- Another example

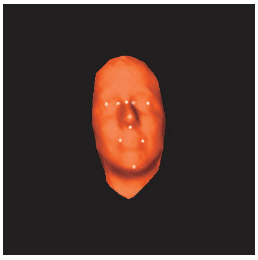
Training Set	Generated Shapes
	

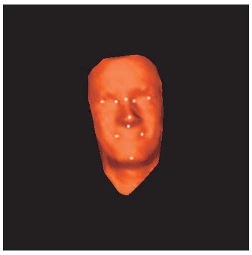
ASM Overview

- Image search (Iterative)
  - Optimize translation,rotation,scale parameters
  - Optimize shape parameters
  - Constrain shape parameters according to training set
  - Until the process converges

Automated Registration of 3D Faces using Dense Surface Models

- Learning algorithm
  - 1) Manually landmark 10 facial points.





Automated Registration of 3D Faces using Dense Surface Models

2) Compute the mean landmarks using Generalised Procrustes analysis.

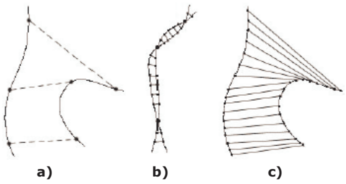
3) Warp the surfaces using their 10 landmark points onto the mean landmark points using the *"Thin Plate Spline Warping"* algorithm.

» TPS is a type of spline interpolation powerful in modelling biological shape changes.

» It simulates the shape changes of a thin metal sheet and therefore minimizes the bending energy of the transform, keeping the distortion to a minimum.

Automated Registration of 3D Faces using Dense Surface Models

4) Each Warped surface is resampled using a base mesh and returned to its original position.



The TPS alignment step illustrated in 2D. The surfaces are landmarked (a), then TPS warped onto a mean set of landmarks and a dense correspondence is established(b). Finally the surfaces are unwarped back to their original locations(c).

Automated Registration of 3D Faces using Dense Surface Models

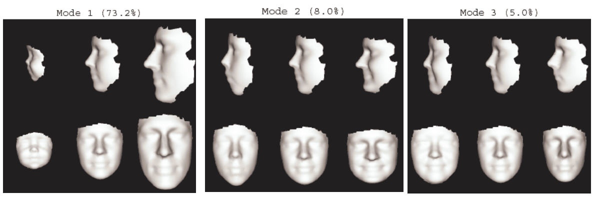
5) If the resampling distance exceeds a threshold, do not include these points in the landmark set.

➔Only points that appear in all shapes are included in the landmark set.

6) Procrustes as in ASM

7) PCA as in ASM.

Automated Registration of 3D Faces using Dense Surface Models



420 people. Aging from 1 to 80. Both genders. Neutral expressions.

Automated Registration of 3D Faces using Dense Surface Models

- Fitting Algorithm
  - 1) Initial template x(0) (mean template), target template y (test data)
  - 2) ICP (Iterative Closest Point) to fit x(0) to y ➔ x(1)
  - 3) Closest point mapping x(1) to y ➔ x(2)
  - 4) Procrustes x(2) onto $\bar{x}$ ➔ x(3)
  - 5) Extract the shape coefficients b

$$b = W^{-1}\Phi^T(x(3) - \bar{x})$$

  - 6) The shape parameters are limited to learned variances from the training set ➔ b'

Automated Registration of 3D Faces using Dense Surface Models

- 7) The best guess surface is computed ➔ x(4)

$$x(4) = \bar{x} + \Phi W b'$$

- 8) x(4) aligned with x(2) ➔ x(5)
- 9) Iterate with x(1) new = x(5)

ASM for facial expression recognition

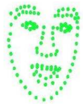
- İsmail Arı

Appearance

- *Appearance* = *Shape* + *Texture*
- *Shape*: tuple of characteristic locations in the image, up to allowed transformation
  - Example: contours of the face up to 2D similarity transformation (translation, rotation, scaling)
- *Texture*: intensity (or color) patch of an image in the shape-normalized frame, up to scale and offset of values

Shape

- Configuration of landmarks
  - Good landmarks – points, consistently located in every image. Add also intermediate points
  - Represent by vector of the coordinates: e.g. $x=(x^1,\dots,x^n, y^1,\dots,y^n)^T$ for n 2D landmarks

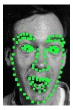


- Configurations x and x' are considered to have the same *shape*[*] if they can be merged by an appropriate transformation T (registration)
- *Shape distance* – the distance after registration:

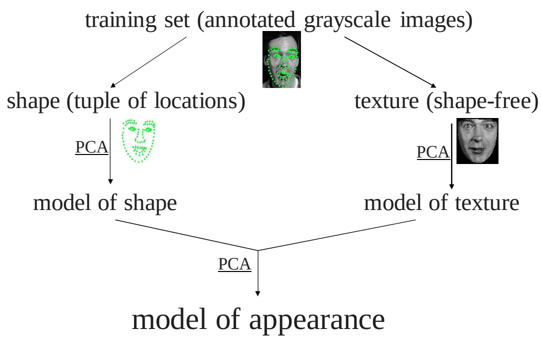
$$dist_{shape}(x, x') = \inf_T \; dist(x, Tx')$$

Shape-free texture

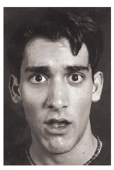
- An attempt to eliminate texture variation due to different shape ("orthogonalization")
- Given shape x and a target "normal" shape x' (typically the average one) we warp our image so that points of x move into the corresponding points of x'

 warped to  becomes 

Modeling Appearance



Training set



- Annotated images
- Done manually, it is the most human time consuming and error prone part of building the models
- (Semi-) automatic methods are being developed[**]

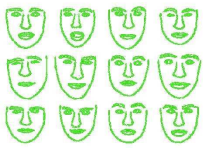
[*] Example from: *Active Shape Model Toolkit* (for MATLAB), Visual Automation Ltd.
[**] A number of references is given in: T.F.Cootes and C.J.Taylor. *"Statistical Models of Appearance for Computer Vision"*, Feb 28 2001; pp. 62-65

Training sets for shape and texture models

- From the initial training set (annotated images) we obtain $\{x_1,\dots,x_n\}$ – set of shapes, $\{g_1,\dots,g_n\}$ – set of shape-free textures.
- We allow the following transformations:
  - S for the shape: translation (t_x,t_y), rotation θ, scaling s.
  - T for the texture: scaling α, offset β ($T\mathbf{g} = (\mathbf{g} - \beta\mathbf{1})/\alpha$).
- Align both sets using these transformations, by minimizing distance between shapes (textures) and their mean
  - Iterative procedure: align all x_i (g_i) to the current $\bar{x}(\bar{g})$, recalculate $\bar{x}$ $(\bar{g})$ with new x_i (g_i), repeat until convergence.

Examples of training sets

Shapes



The mean shape

Textures

*

* From the work of Mikkel B. Stegmann, Section for Image Analysis, The Technical University of Denmark

Model of Shape

- Training set $\{x_1,\ldots,x_n\}$ of aligned shapes
- Apply PCA to the training set
  - Model of shape: $x = \bar{x} + P_s b_s$
    where $\bar{x}$ (the mean shape) and P_s (matrix of eigenvectors) define the model;
    b_s is a vector of parameters of the model.
  - Range of variation of parameters: $|b_s^i| \le 3\sqrt{\lambda_i}$
    determined by the eigenvalues, e.g.

Example: 3 modes of a shape model

Model of Texture

- Training set $\{g_1,\ldots,g_n\}$ of shape-free normalized image patches
- Apply PCA
  - Model of texture: $g = \bar{g} + P_g b_g$
  - Range of variation of parameters

Example: 1st mode of a texture model:

Combining two models

- Joint parameter vector $b = \begin{pmatrix} W_s b_s \\ b_g \end{pmatrix}$
  where the diagonal matrix W_s accounts for different units of shape and texture parameters.
- Training set
  - For every pair (x_i, g_i) we obtain:
    $$b_i = \begin{pmatrix} W_s b_{si} \\ b_{gi} \end{pmatrix} = \begin{pmatrix} W_s P_s^T (x_i - \bar{x}) \\ P_g^T (g_i - \bar{g}) \end{pmatrix}$$
- Apply PCA to the training set $\{b_1,\ldots,b_n\}$
  - Model for parameters: $b = P_c c$, $P_c = [P_{cs} | P_{cg}]^T$
  - Finally, the combined model:
    $$x = \bar{x} + Q_s c, \; g = \bar{g} + Q_g c$$
    where $Q_s = P_s W_s^{-1} P_{cs}$, $Q_s = P_g P_{cg}$.

Examples (combined model)

$$x = \bar{x} + Q_s c, \; g = \bar{g} + Q_g c$$

- Self-portrait of the inventor

Tim Cootes

His shape

A mode of the model

- Color model (by Gareth Edwards)

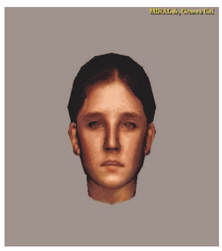
Several modes

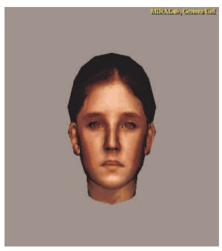
Generating synthetic images: example

By varying parameters c in the appearance model

$$x = \bar{x} + Q_s c, \; g = \bar{g} + Q_g c$$

we obtain synthetic images:





Active Appearance Model (AAM)

Given:	*Find:*
1) an appearance model,	the best matching
2) a new image,	synthetic image
3) a starting approximation	

Approach:

- Difference vector: $\delta I = I_i - I_m$
  - I_i – input (new) image;
  - I_m – model-generated (synthetic) image for the current estimation of parameters.
- Search for the best match
  - Minimize $\Delta = |\delta I|^2$, varying parameters of the model

Predicting difference of parameters

- Knowing the matching error δI, we want to obtain information how to improve parameters c
- Approximate this relation by $\delta c = \mathbf{A}\delta I$
- Precompute **A**:
  - Include into δc extra parameters: translations, rotations and scaling of shape; scaling and offset of gray levels (texture)
  - Take δI in the shape-normalized frame i.e. $\delta I = \delta g$ where textures are warped into the same shape
  - Generate pairs $(\delta c, \delta g)$ and estimate **A** by linear regression.

AAM search: examples

Model of face

Model of hand

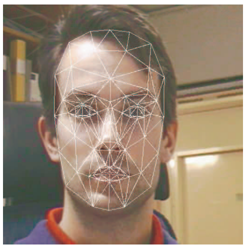
From the work of Mikkel B. Stegmann, Section for Image Analysis, The Technical University of Denmark

AAM: tracking experiments

AAM

Done with AAM-API (Mikkel B. Stegmann)

Extension of AAM



By Jörgen Ahlberg, Linköping University