ACTIVE SHAPE MODELS

Yogesh Singh Rawat
SOC NUS
September 19, 2012

Active Shape Models

- T.F.Cootes, C.J.Taylor, D.H.Cooper, J.Graham, “Active Shape Models: Their Training and Application.” Computer Vision and Image Understanding, V16, N1, January, pp. 38-59, 1995.

What we will talk about?

- Modeling of objects which can change shape.

Example

Image Source - Google Images

Example

Image Source - Google Images

Example

Image Source - Google Images

Example

Image Source - Google Images

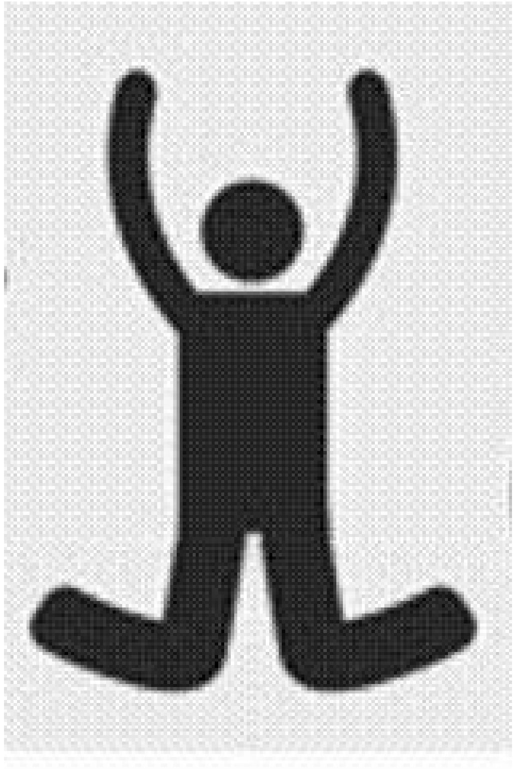
Problem

- Not a new problem, has been solved before.
- New method to solve the problem.
- Better then earlier methods.

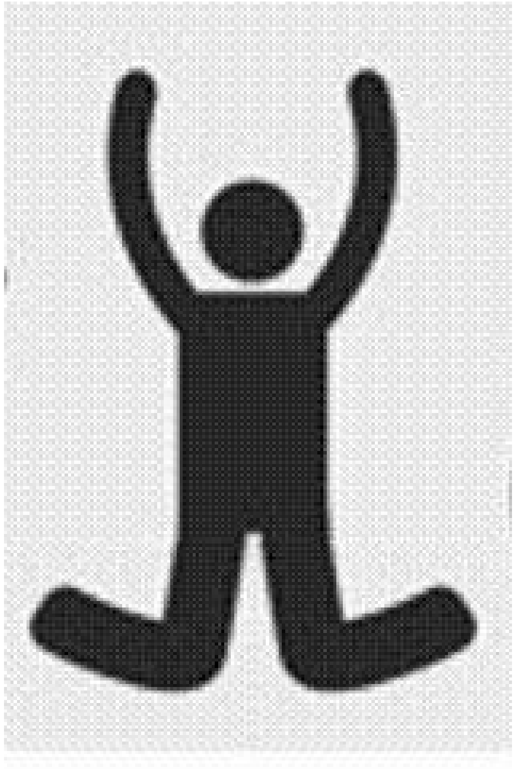
Possible Shapes of Human Body

Image Source - Google Images

Is This Possible?

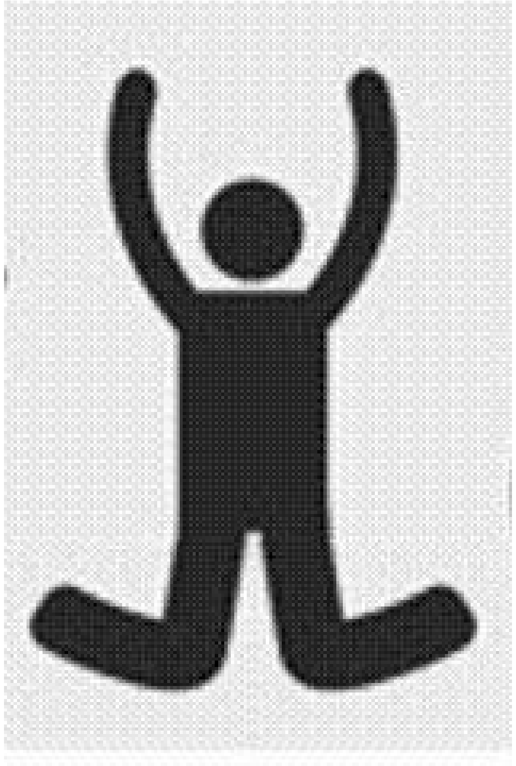


Is This Possible?



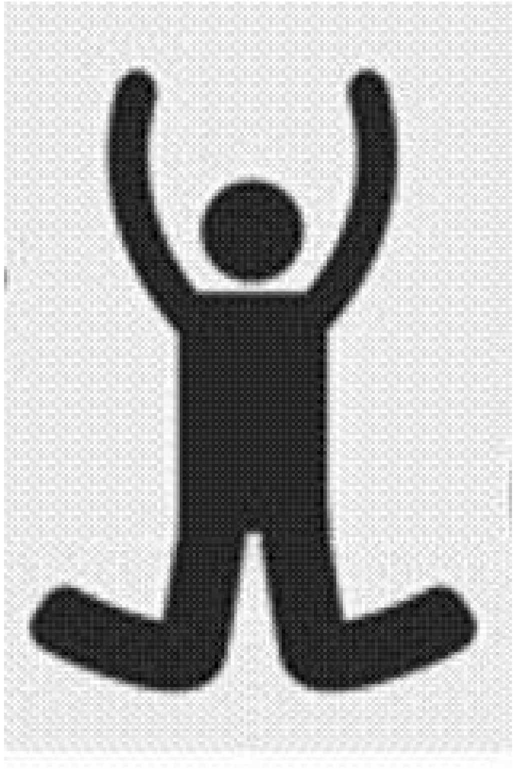
Model - YES

Is This Possible?



Real Life - NO

Is This Possible?



Motivation for this work

Existing Models

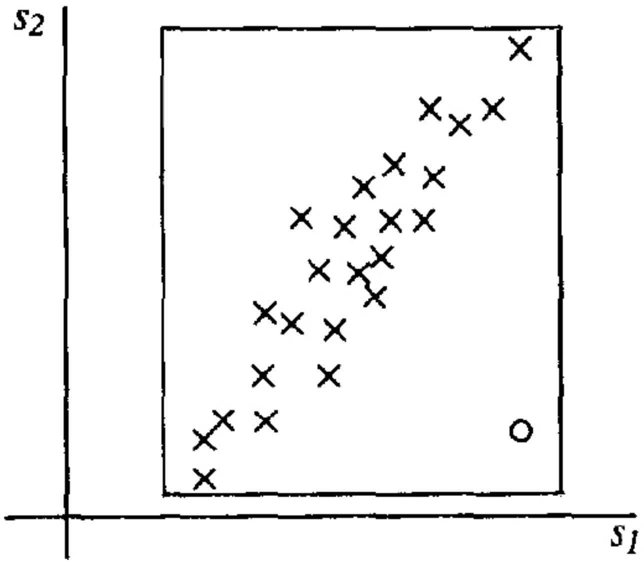
- “Hand Crafted” Models
- Articulated Models
- Active Contour Models – “Snakes”
- Fourier Series Shape Models
- Statistical Models of Shape
- Finite Element Models

Problem with Existing Models

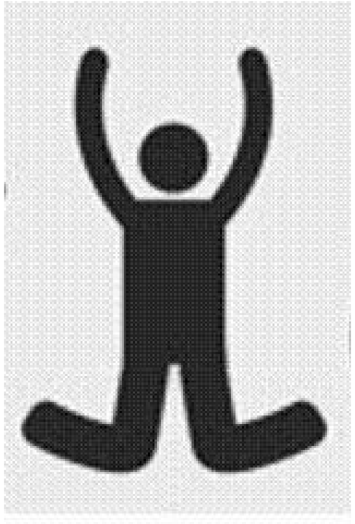
- Nonspecific class deformation
  - An object should transform only as per the characteristics of the class.

Problem with Existing Models

- If two shape parameters are correlated over a set of shapes then their variation does not restrict shapes to any set of class.



Problem with Existing Models



No restriction on deformation
Not a robust model

Goals

- Deform to characteristics of the class represented
- “Learn” specific patterns of variability from a training set
- Specific to ranges of variation
- Searches images for represented structures
- Classify shapes
- Robust (noisy, cluttered, and occluded image)

Point Distribution Model

- Captures variability of training set by calculating mean shape and **main modes of variation**
- Each mode changes the shape by moving landmarks along **straight lines** through mean positions
- New shapes created by modifying mean shape with weighted sums of modes

PDM Construction

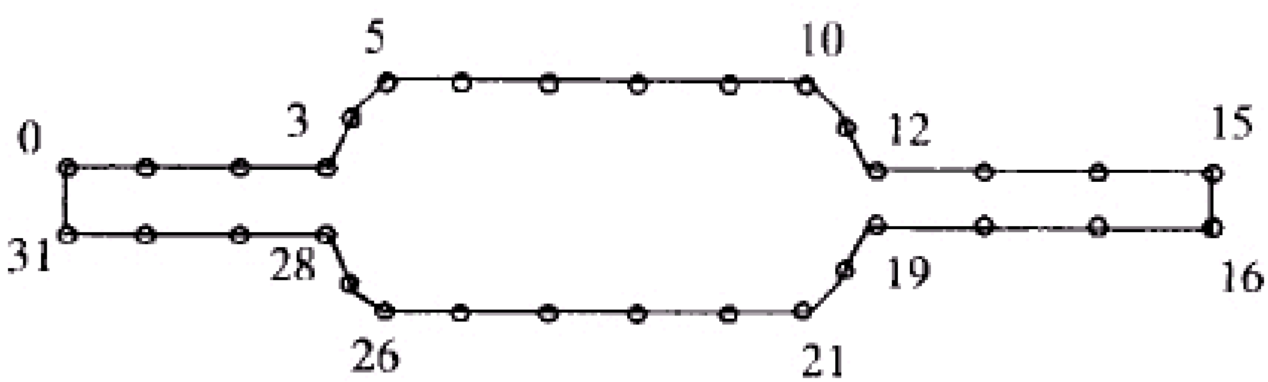
Manual Labeling

Alignment

Statistical Analysis

Labeling the Training Set

- Represent shapes by **points**
- Useful points are marked called "**landmark points**"



- Manual Process

Aligning the Training Set

- $\mathbf{x}_i$ is a vector of n points describing the the ith shape in the set:

$$\boldsymbol{x}_i=(x_{i0}, y_{i0}, x_{i1}, y_{i1},\ldots\ldots, x_{ik}, y_{ik},\ldots\ldots,x_{in-1}, y_{in-1})^T$$

- Minimize:

"weighted sum of squares of distances between equivalent points"

Aligning the Training Set

- Minimize:

$$E_j = (\mathbf{x}_i - M(s_j, \theta_j)[\mathbf{x}_k] - \mathbf{t}_j)^T \mathbf{W} (\mathbf{x}_i - M(s_j, \theta_j)[\mathbf{x}_k] - \mathbf{t}_j)$$

- Weight matrix used:

$$w_k = \left(\sum_{l=0}^{n-1} V_{R_{kl}} \right)^{-1}$$

- More significance is given to those points which are stable over the set.

Alignment Algorithm

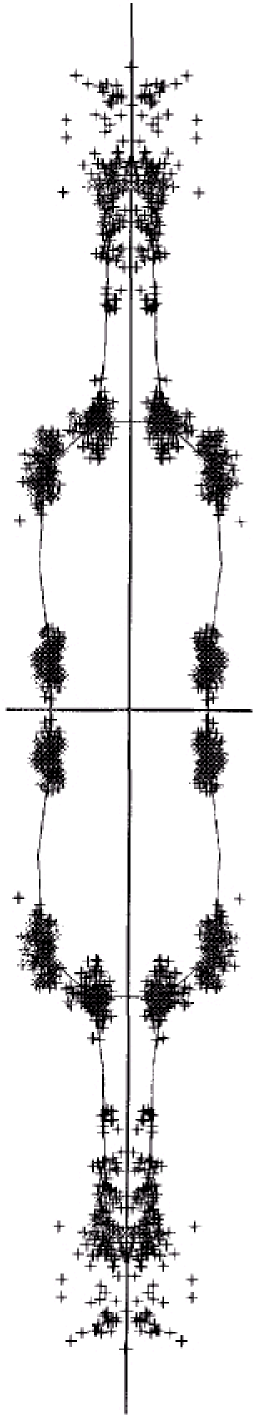
- Align each shape to first shape by rotation, scaling, and translation
- **Repeat**
  - Calculate the mean shape
  - Normalize the orientation, scale, and origin of the current mean to suitable defaults
  - Realign every shape with the current mean
- **Until** the process **converges**

Mean Normalization

- Guarantees convergence
- Not formally proved
- Independent of initial shape aligned to

Aligned Shape Statistics

- PDM models "**cloud**" variation in 2n space
- Assumptions:
  - Points lie within "**Allowable Shape Domain**"
  - Cloud is **ellipsoid** (2n-D)



Statistics

- Center of ellipsoid is mean shape

$$\bar{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x_i}$$

- Axes are found using PCA
  - Each axis yields a **mode of variation**
  - Defined as p_k, the eigenvectors of covariance matrix

$$\mathbf{S} = \frac{1}{N}\sum_{i=1}^{N} d\mathbf{x_i} d\mathbf{x}_i^T$$, such that

$$\mathbf{S}\mathbf{p}_k = \lambda_k \mathbf{p}_k$$,where λ_k is the kth **eigenvalue** of **S**

Approximation

- Most variation described by *t*-modes
- Choose *t* such that a small number of modes accounts for most of the total variance
- If total variance = $\lambda_T = \sum_{k=1}^{2n} \lambda_k$ and the approximated variance = $\lambda_A = \sum_{i=1}^{t} \lambda_i$, then

$$\lambda_A \cong \lambda_T$$

Generating New Example Shapes

- Shapes of training set approximated by:

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b}$$

Generating New Example Shapes

- Shapes of training set approximated by:

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b}$$

where $\mathbf{P} = (\mathbf{p}_1 \mathbf{p}_2 \ldots \mathbf{p}_t)$ is the matrix of the first *t* eigenvectors and $\mathbf{b} = (b_1 b_2 \ldots b_t)^T$ is a vector of weights

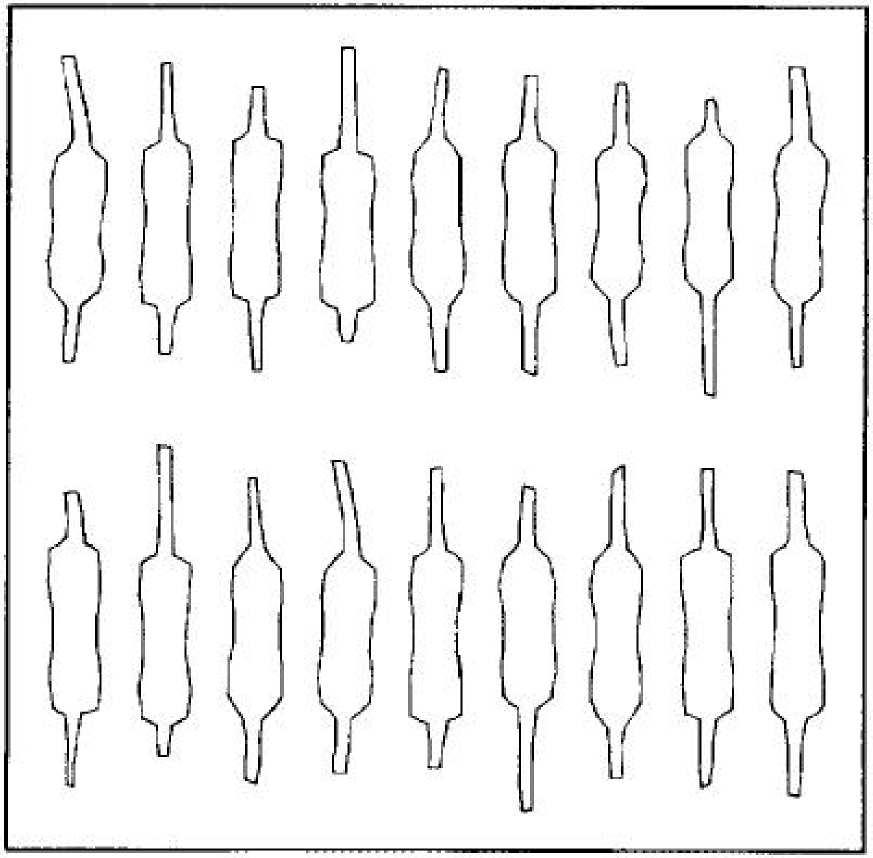
- Vary b_k within suitable **limits** for similar shapes

$$-3\sqrt{\lambda_k} \le b_k \le 3\sqrt{\lambda_k}$$

Experiments

- Applied to:
  - Resistors
  - “Heart”
  - Hand
  - “Worm” model

Resistor Example



Training Set

Resistor Example

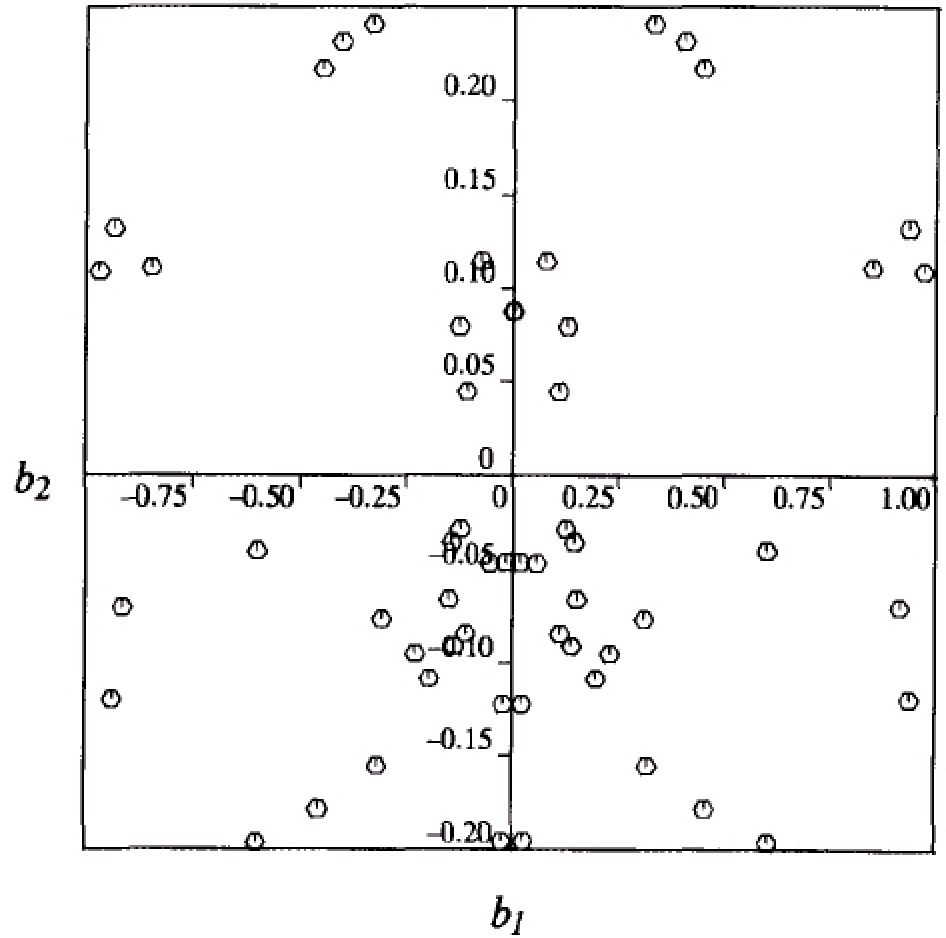
- 32 points
- 3 parameters capture variability

Eigenvalues of the Covariance Matrix Derived from a Set of Resistor Shapes

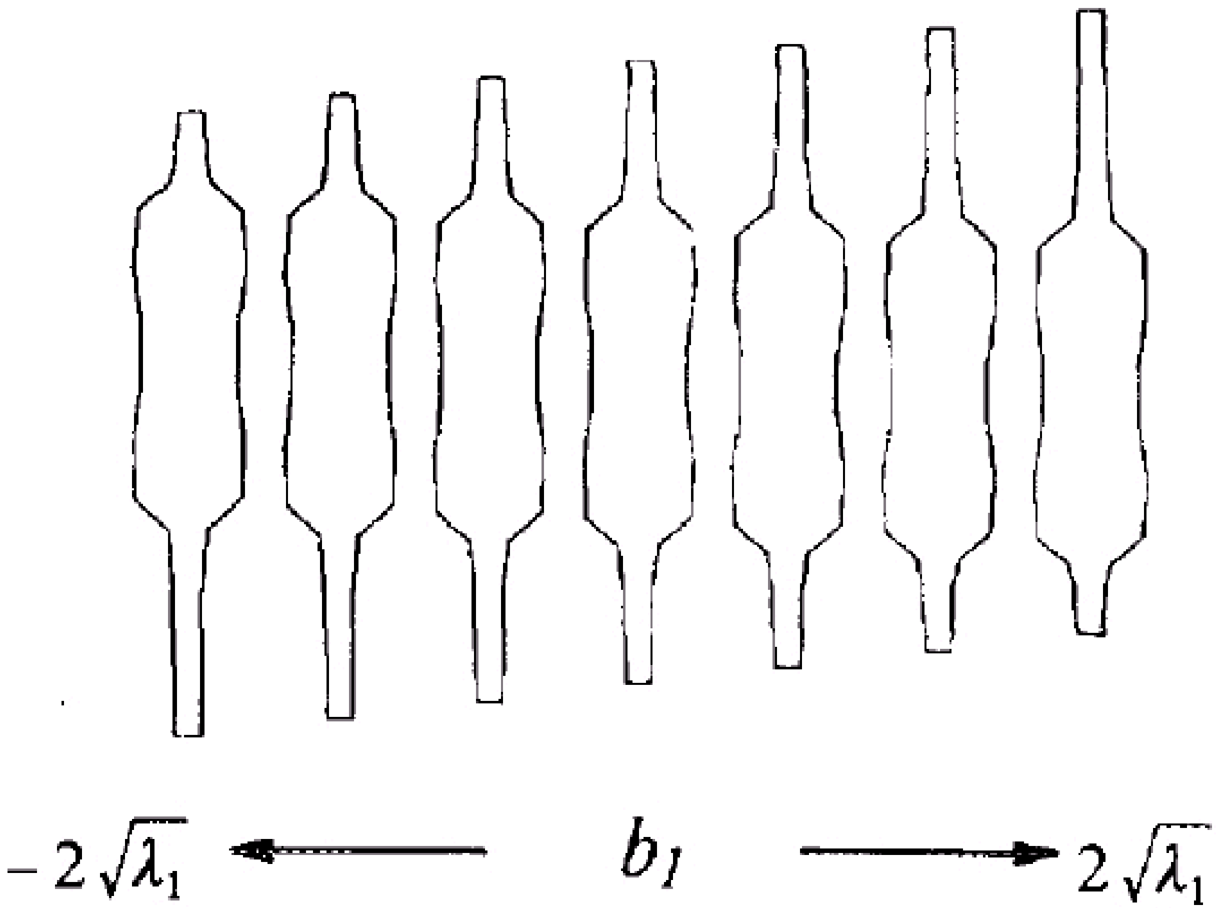
Eigenvalue	$\frac{\lambda_i}{\lambda_T} \times 100\%$
λ_1	66%
λ_2	8%
λ_3	5%
λ_4	4%
λ_5	3%
λ_6	3%

Resistor Example (cont.'d)

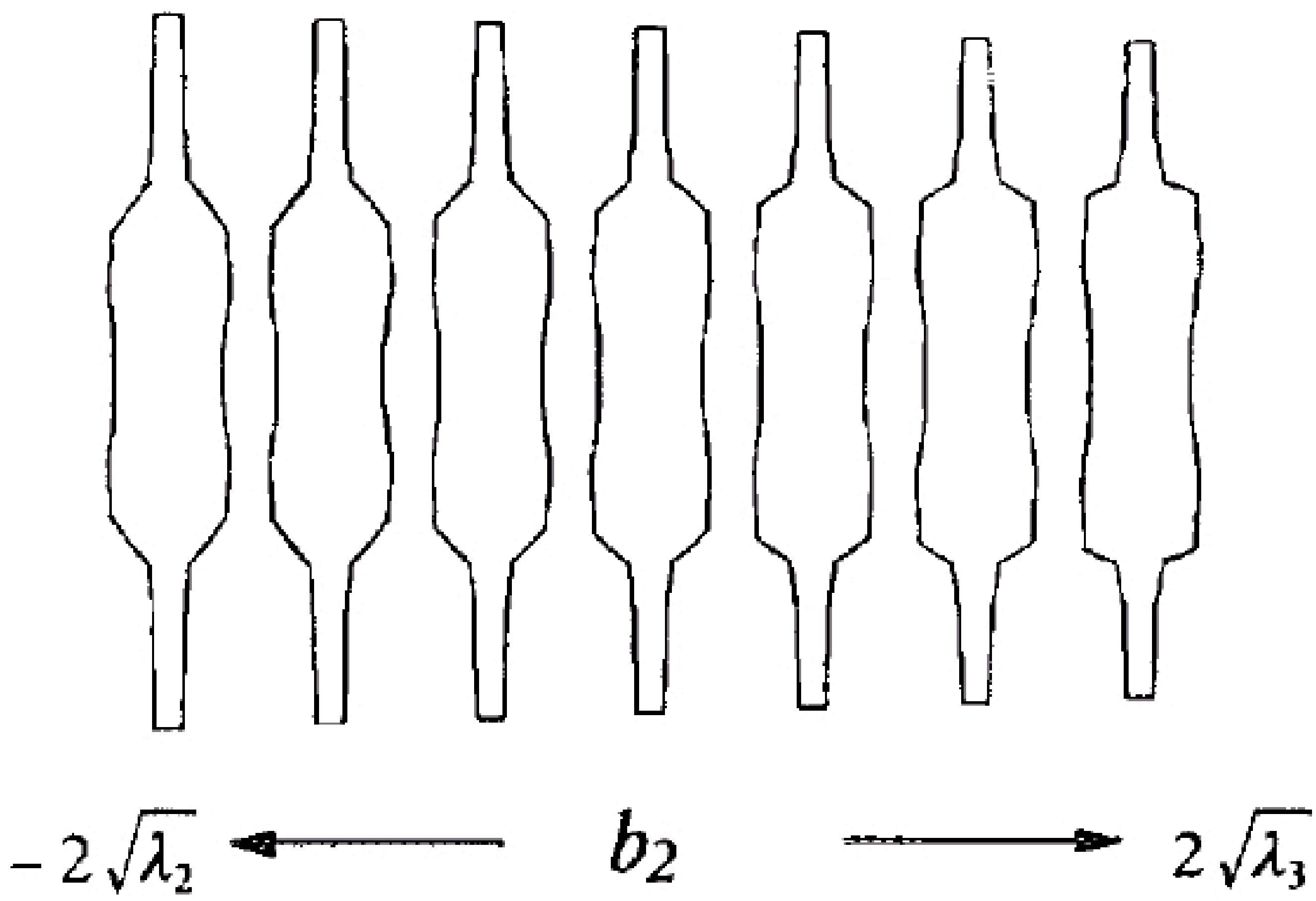
- Lacks structure
- **Independence** of parameters b_1 and b_2
- Will generate "legal" shapes



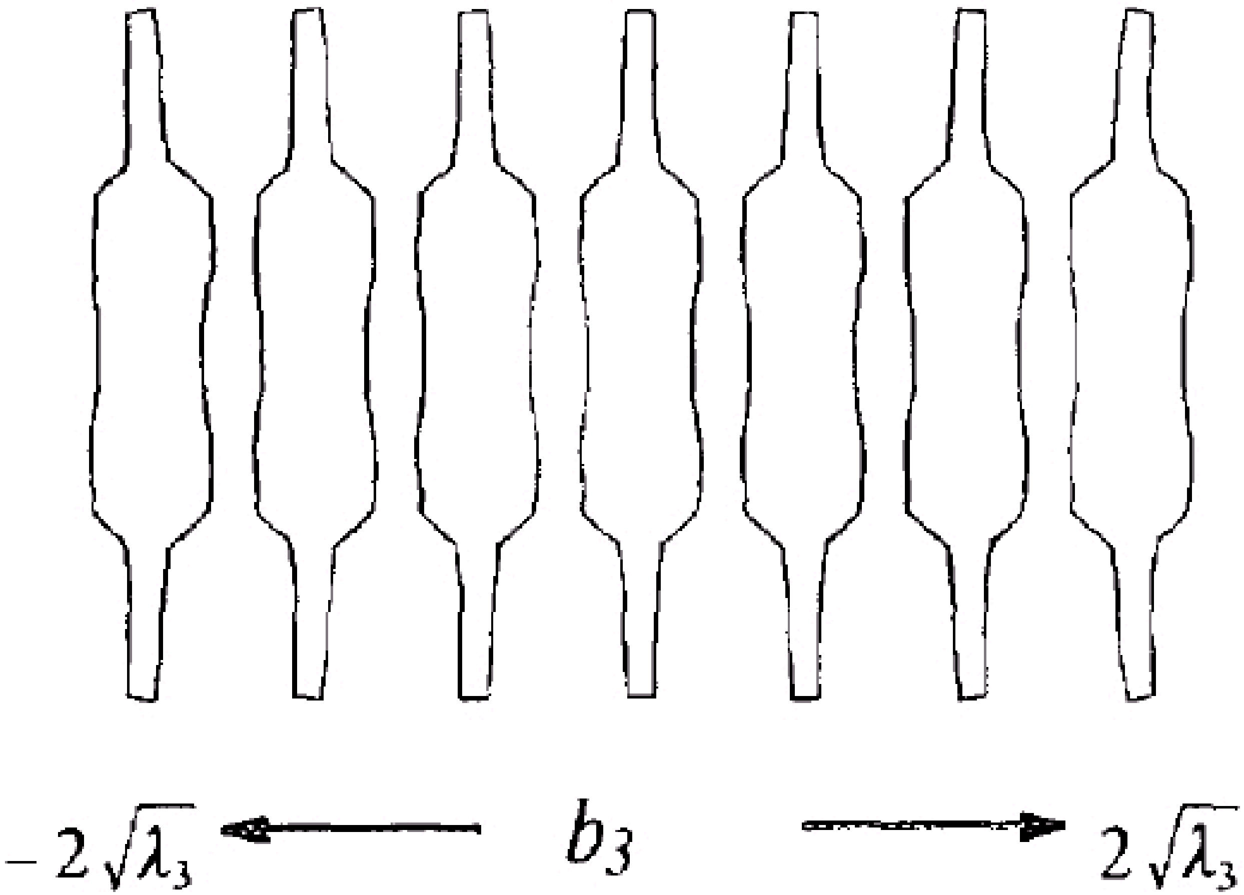
Resistor Example (cont.'d)



Resistor Example (cont.'d)

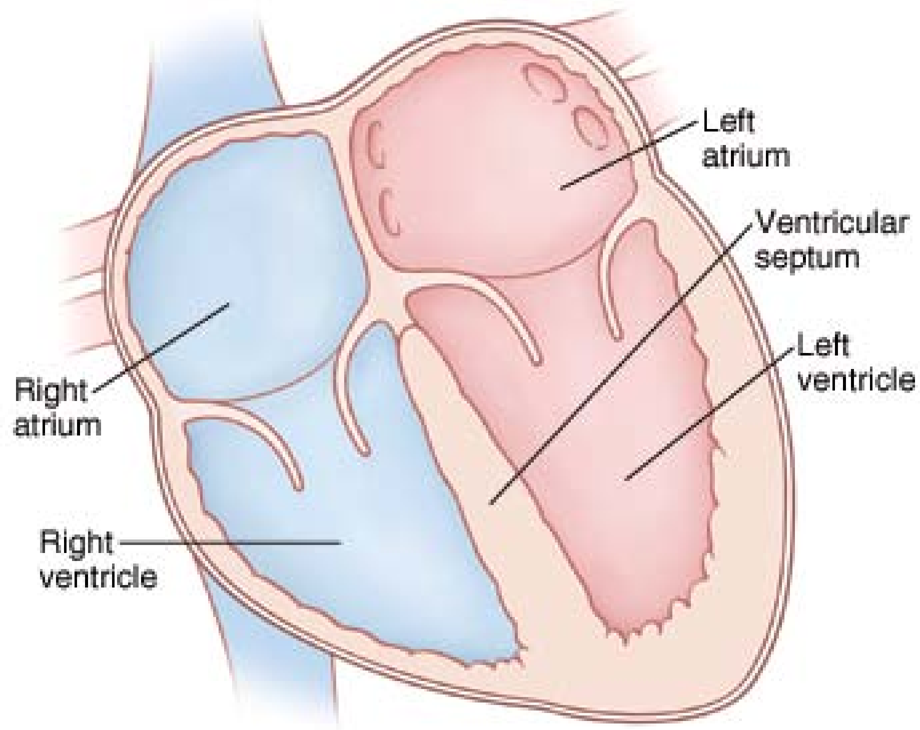


Resistor Example (cont.'d)

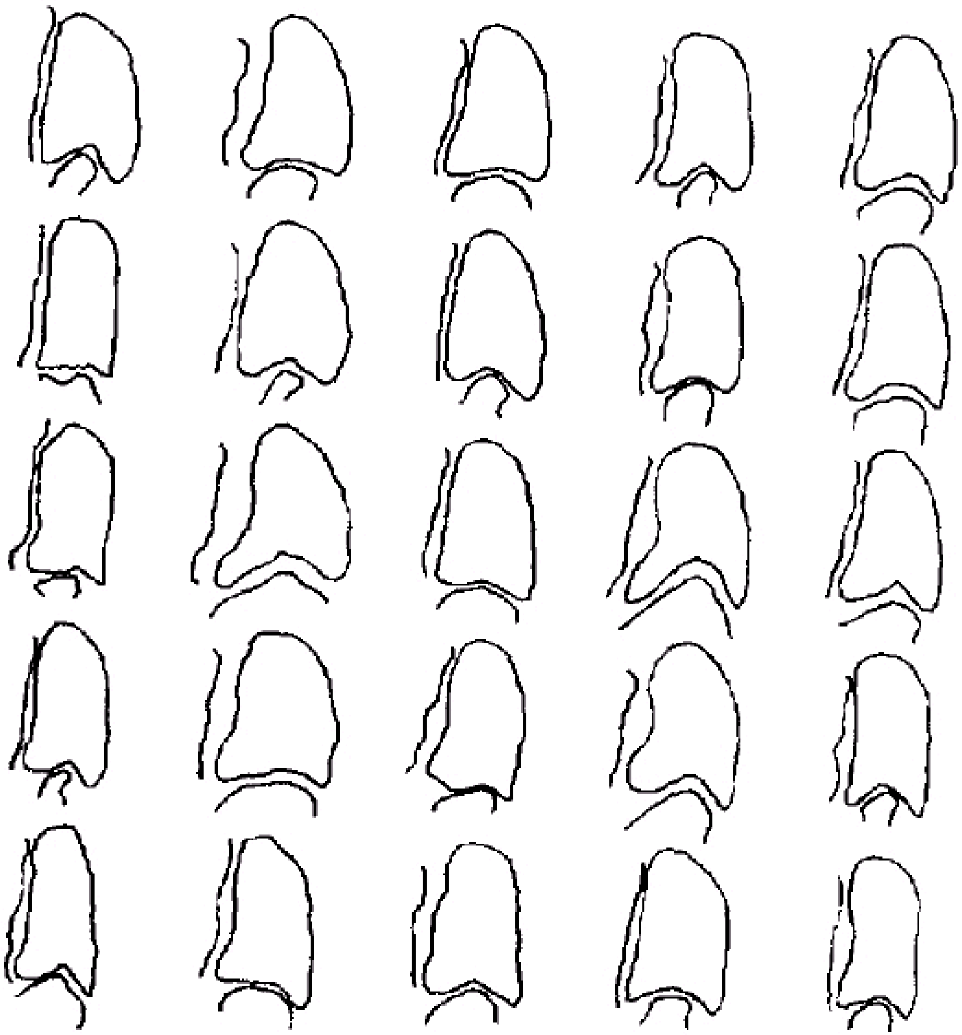


"Heart" Example

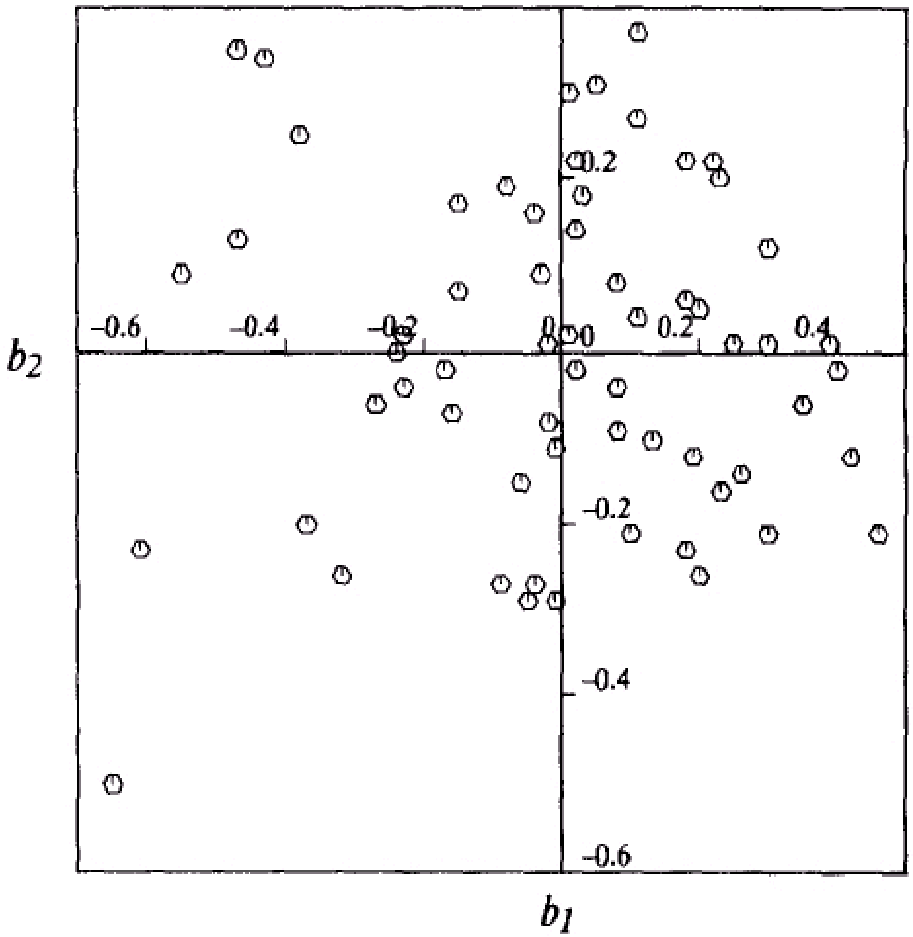
- 66 examples
- 96 points
  - Left ventricle
  - Right ventricle
  - Left atrium
- Traced by cardiologists



“Heart” Example



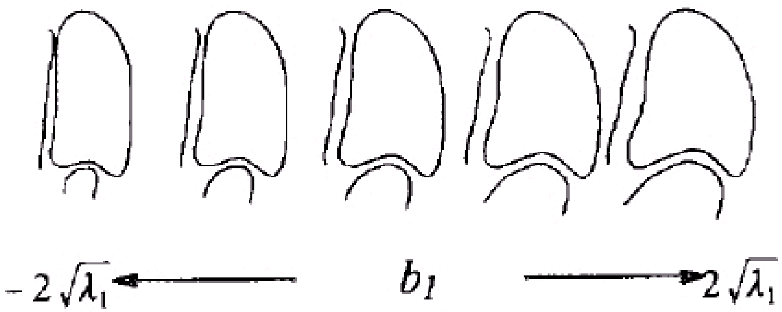
"Heart" Example (cont.'d)



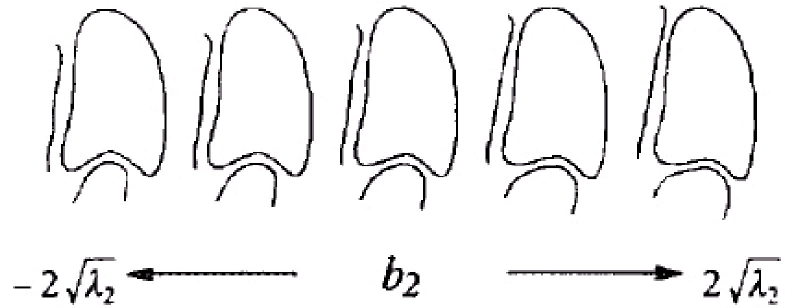
Eigenvalues of the Covariance Matrix Derived from a Set of Heart Ventricle Shapes

Eigenvalue	$\frac{\lambda_i}{\lambda_T} \times 100\%$
λ_1	37%
λ_2	17%
λ_3	13%
λ_4	7%
λ_5	6%
λ_6	4%

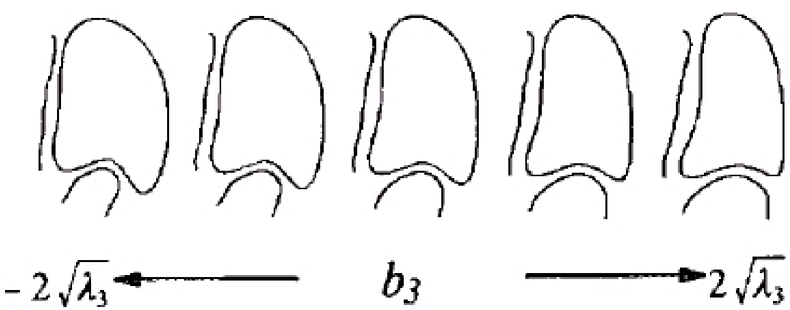
"Heart" Example (cont.'d)



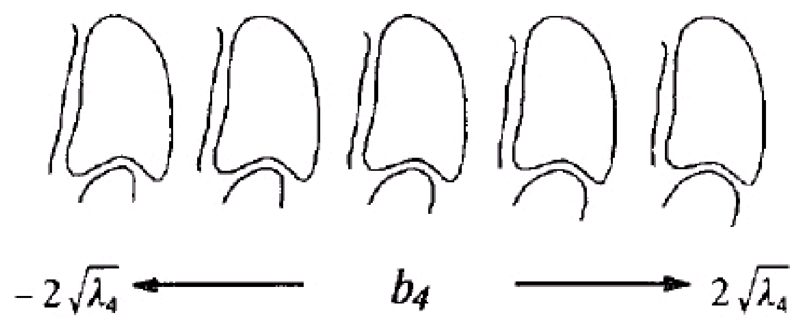
- Varies Width



- Varies Septum



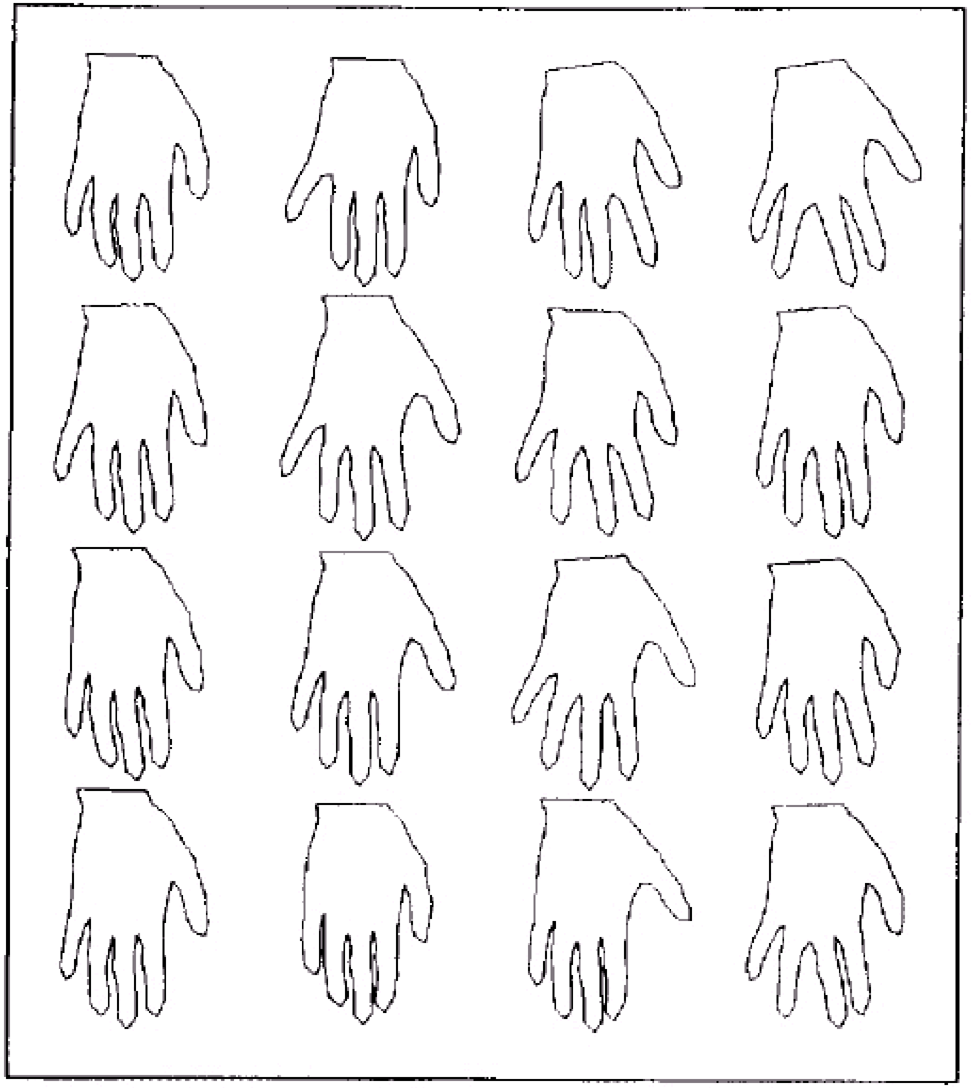
- Varies LV



- Varies Atrium

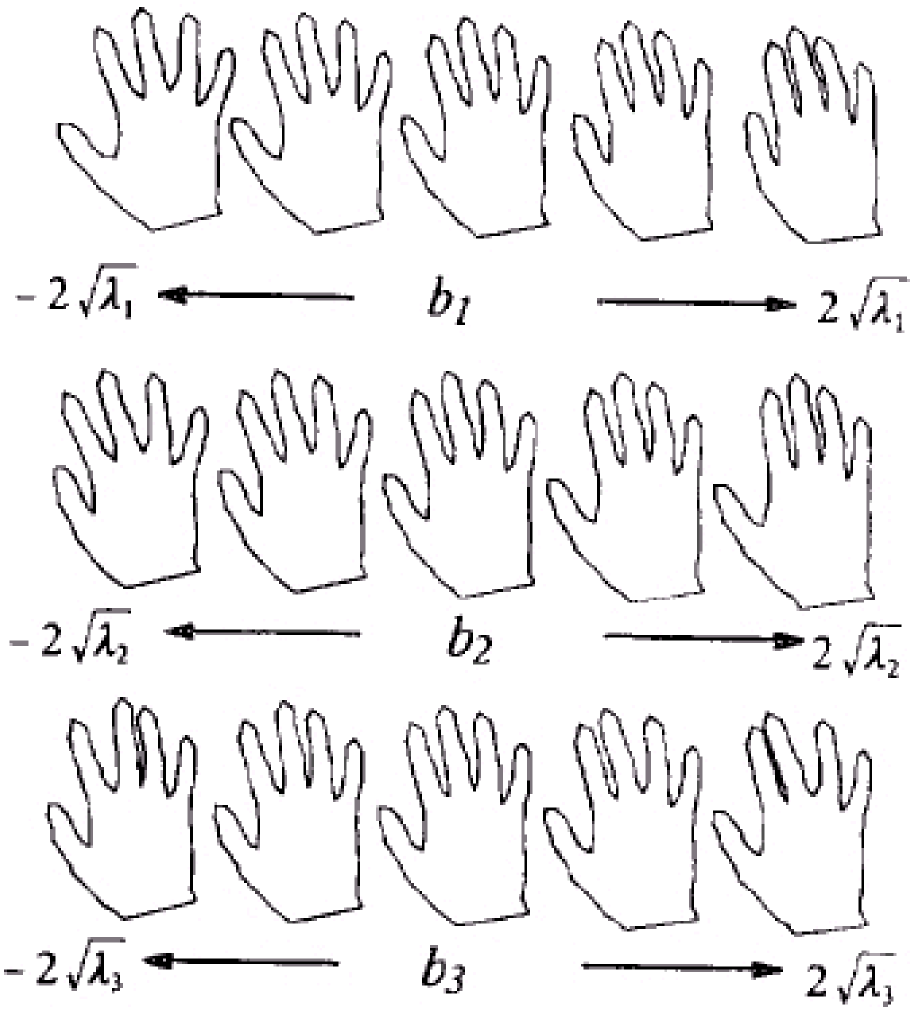
Hand Example

- 18 shapes
- 72 points
- 12 landmarks at fingertips and joints



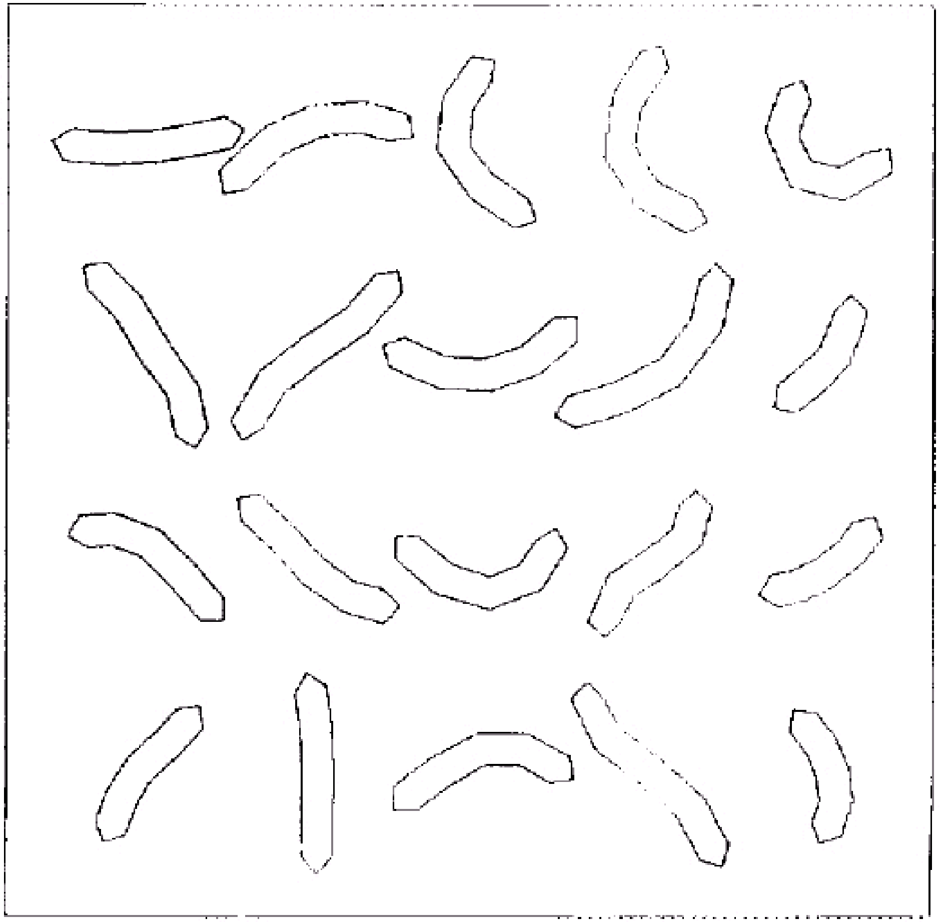
Hand Example (cont.'d)

- 96% of variability due to first 6 modes
- First 3 modes
- Vary finger movements



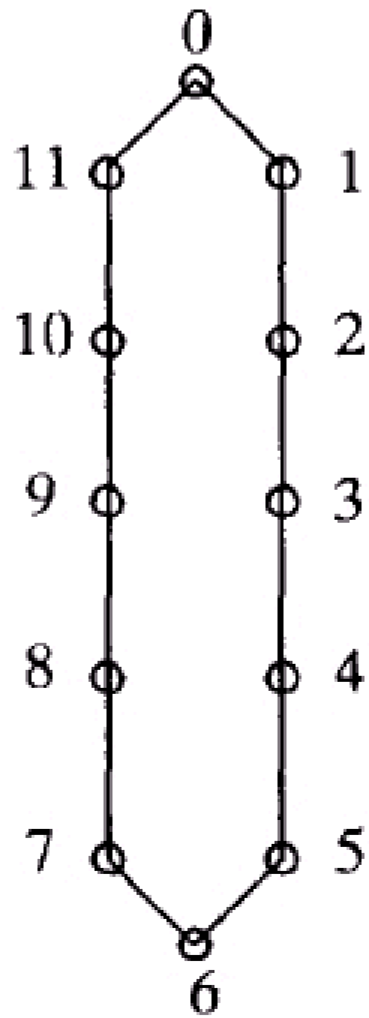
“Worm” Example

- 84 shapes
- **Fixed** width
- Varying curvature and length



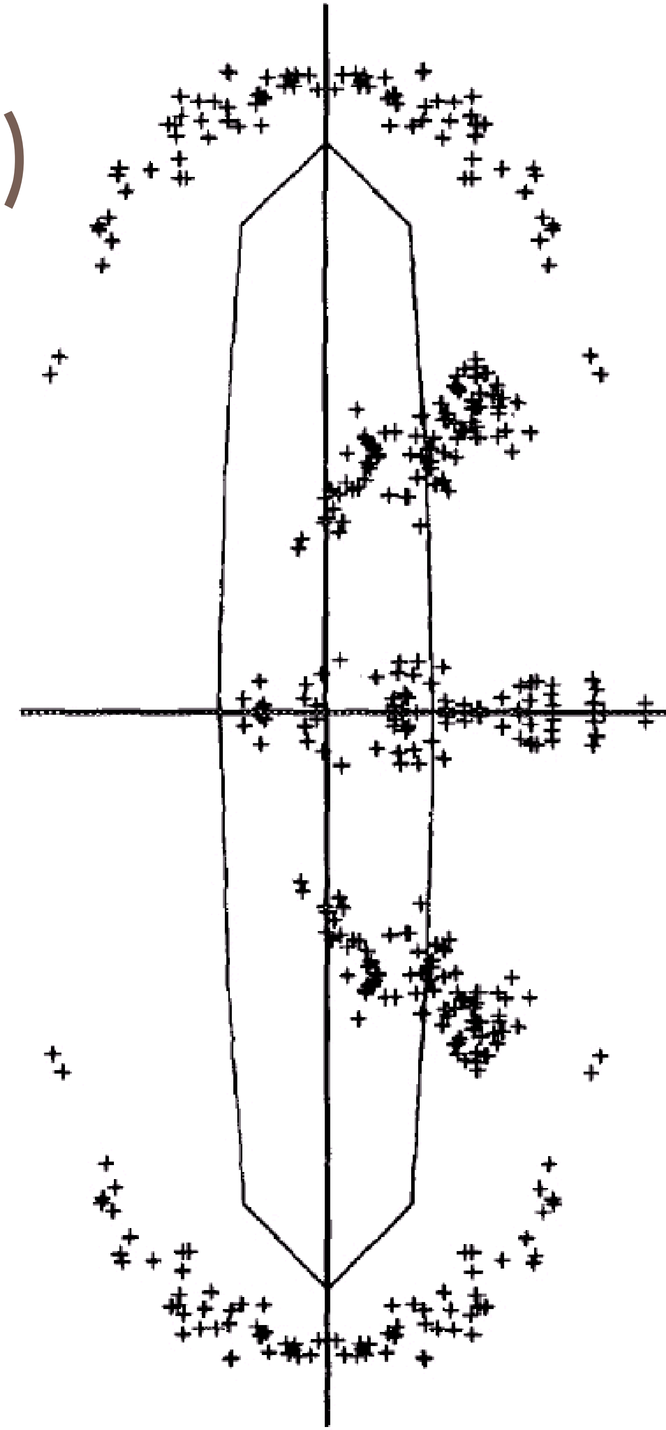
“Worm” Example (cont.’d)

- Represented by 12 point
- Breakdown of PDM



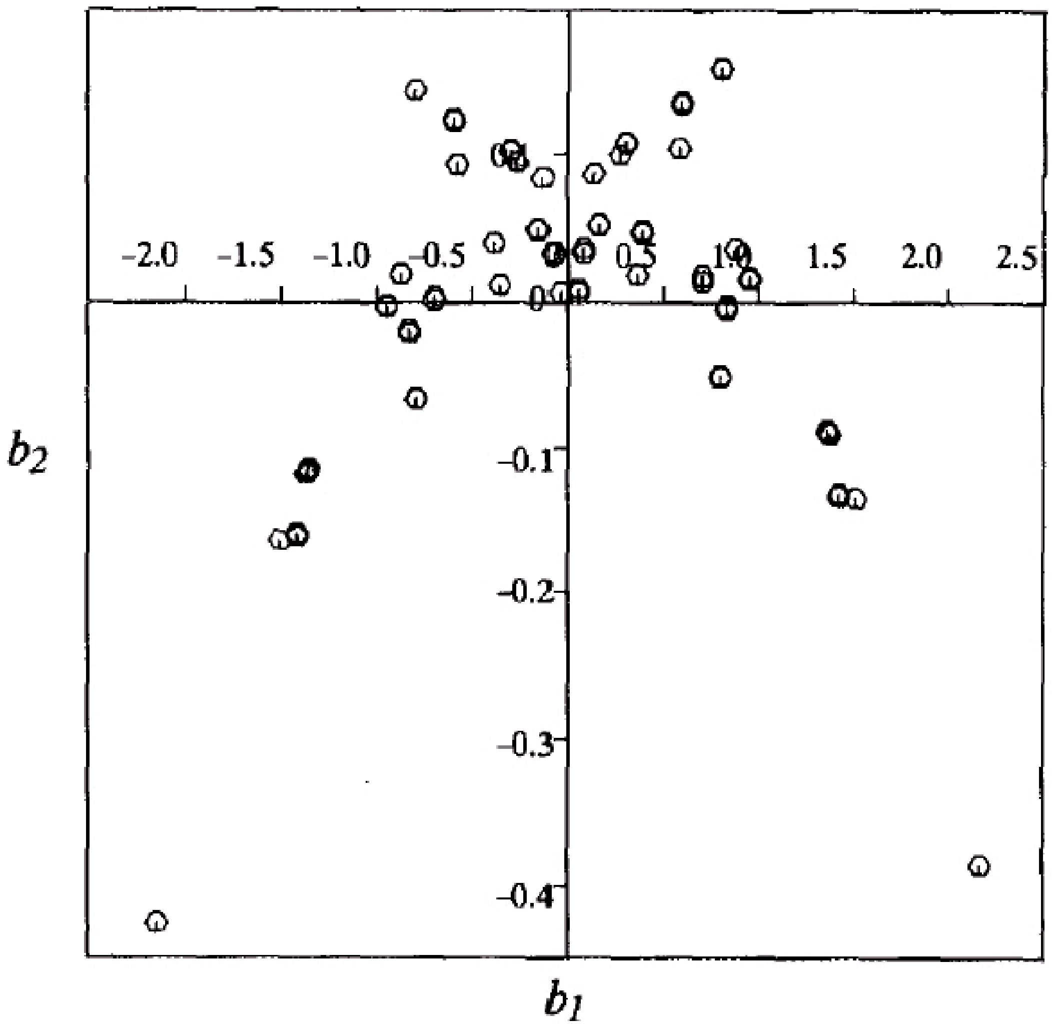
“Worm” Example (cont.’d)

- Curved cloud
- Mean shape:
  - Varying width
  - Improper length



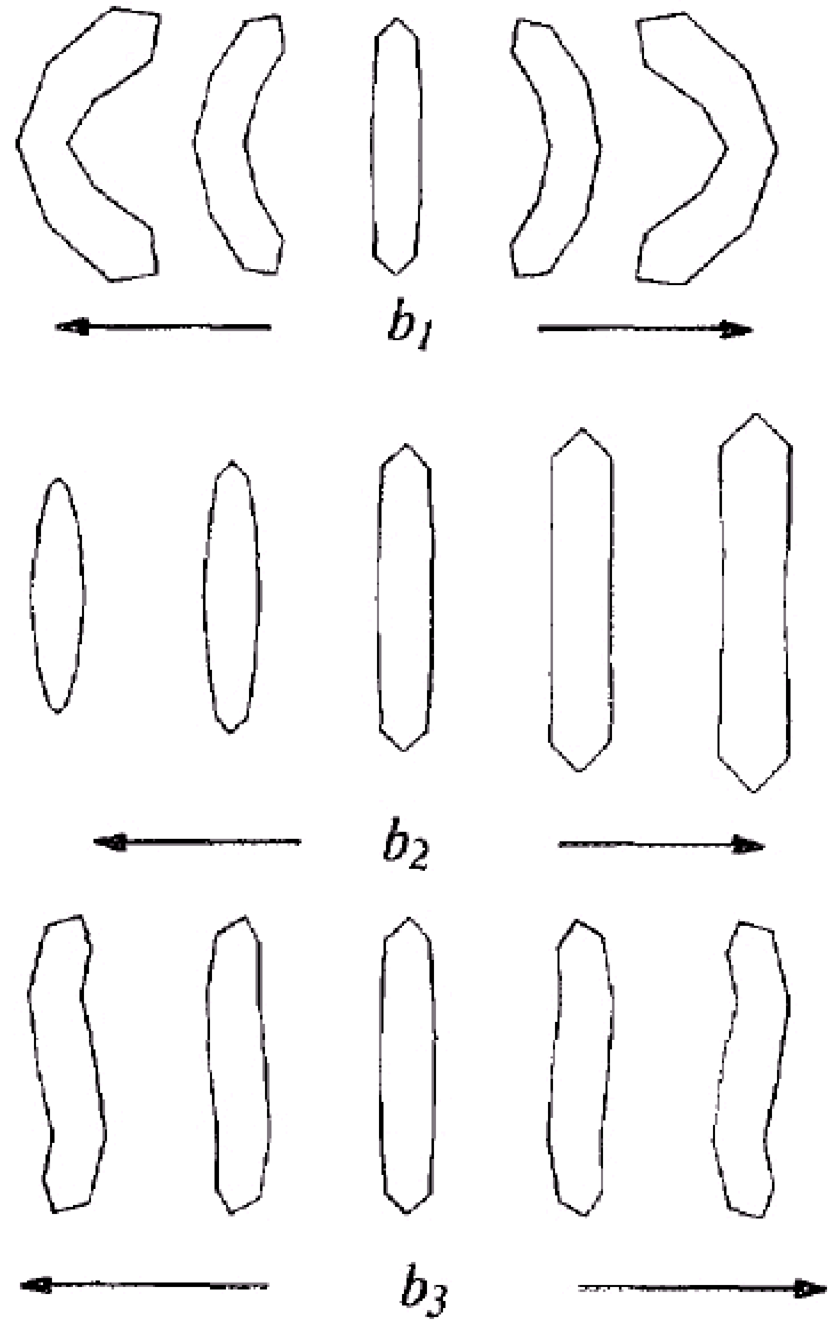
"Worm" Example (cont.'d)

- Linearly independent
- Nonlinear dependence



"Worm" Example

- Effects of varying first 3 parameters:
- 1st mode is **linear** approximation to **curvature**
- 2nd mode is correction to poor linear approximation
- 3rd approximates 2nd order bending



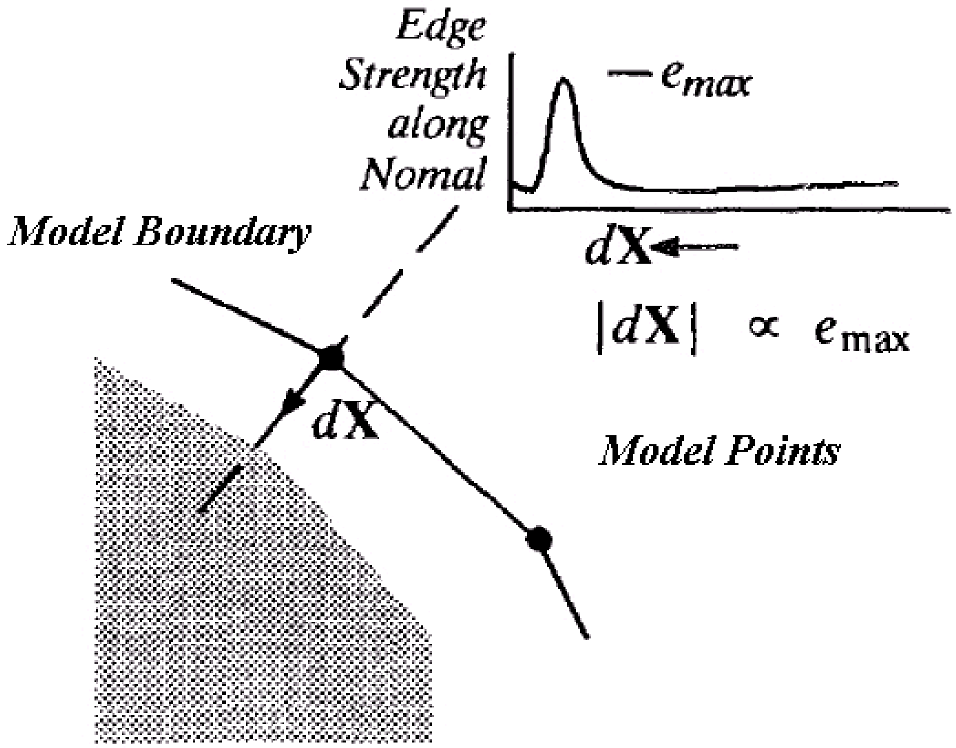
PDM Improvements

- Automated labeling
- 3D PDMs
- Multi-layered PDMs
- Chord Length Distribution Model

PDMs to Search an Image - ASMs

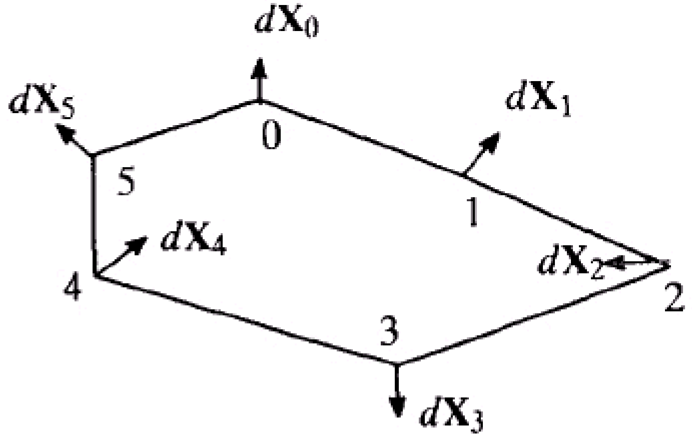
- Estimate initial position of model
- Displace points of model to “better fit” data
- Adjust model parameters
- Apply global constraints to keep model “legal”

Adjusting Model Points



- Along normal to model boundary proportional to edge strength

Adjusting Model Points



- Vector of adjustments:

$$d\mathbf{X} = (dX_0, dY_0, \ldots, dX_{n-1}, dY_{n-1})^T$$

Calculating Changes in Parameters

- Initial position: $\mathbf{X} = M(s,\theta)[\mathbf{x}] + \mathbf{X}_c$
- Move ***X*** as close to new position (***X*** + *d**X***)
- Calculate *dx* to move ***X*** by *d**X***

$$M(s(1+ds),(\theta,d\theta)[\mathbf{x}+d\mathbf{x}]+(\mathbf{X}_c + d\mathbf{X}_c) = (\mathbf{X}+d\mathbf{x})$$

$$d\mathbf{x} = M((s(1+ds))^{-1},-(\theta,d\theta))[\mathbf{y}]-\mathbf{x}, \text{ where } \mathbf{y} = M(s,\theta)[\mathbf{x}\} + d\mathbf{X} - d\mathbf{X}_c$$

- Update parameters to better fit image
- Not usually consistent with model constraints
- Residual adjustments made by deformation

Model Parameter Space

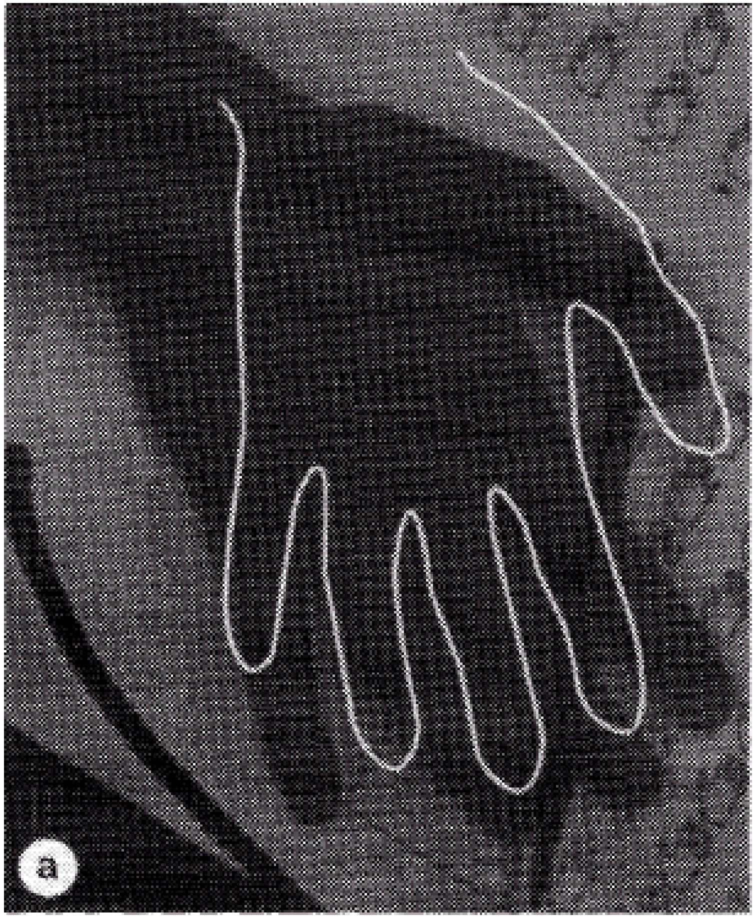
- Transforms *d**x*** to parameter space giving **allowable** changes in parameters, *d**b***
- Recall: $\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b}$
  - Find *d**b*** such that $\mathbf{x} + d\mathbf{x} \approx \bar{\mathbf{x}} + \mathbf{P}(\mathbf{b} + d\mathbf{b})$
  - $\bar{\mathbf{x}} + \mathbf{P}\mathbf{b} = (\bar{\mathbf{x}} + \mathbf{P}(\mathbf{b} + d\mathbf{b}) - d\mathbf{x})$ yields

$$d\mathbf{b} = \mathbf{P}^T d\mathbf{x}$$

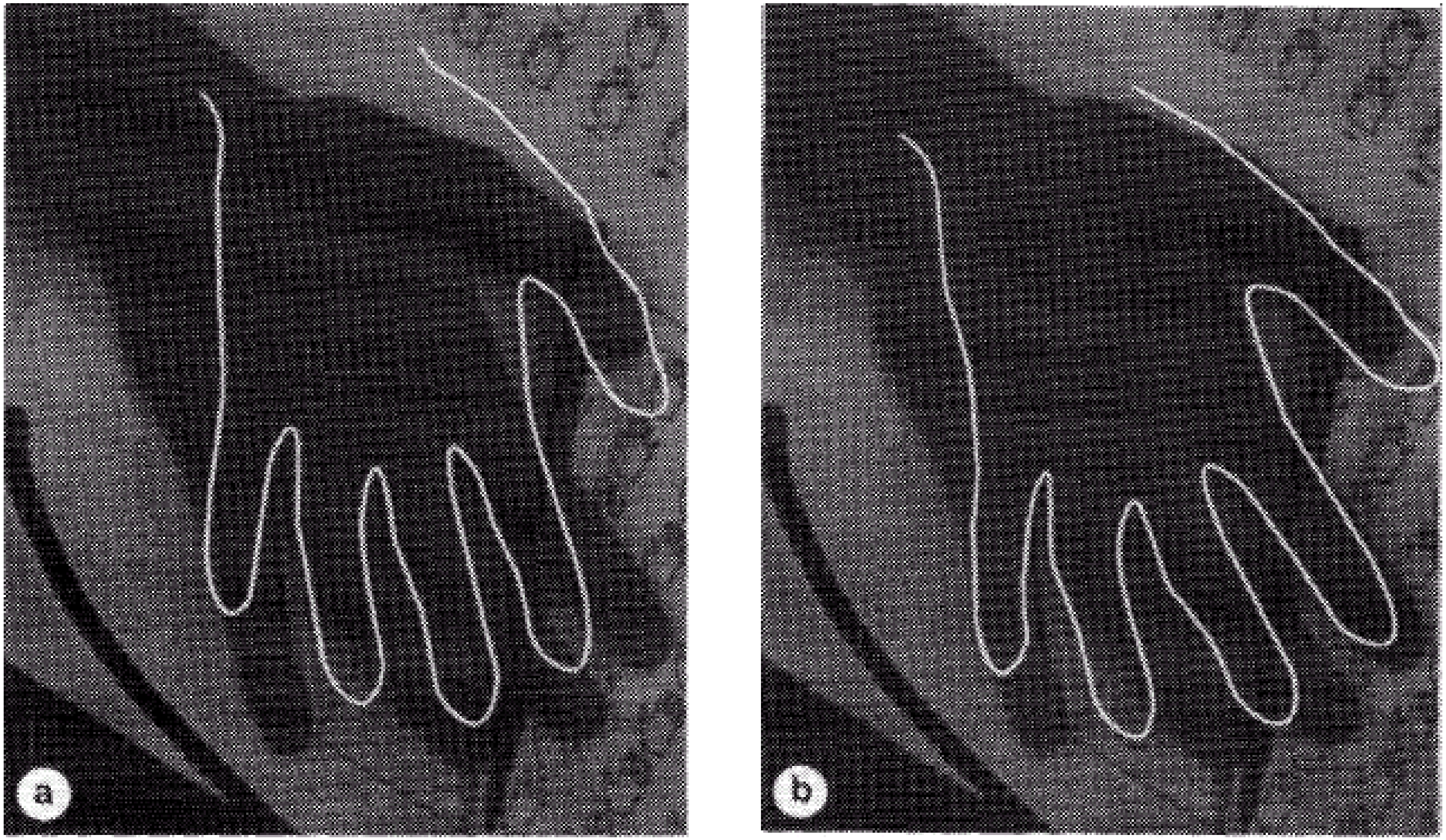
- Update model parameters within limits

ASM Application to Hand

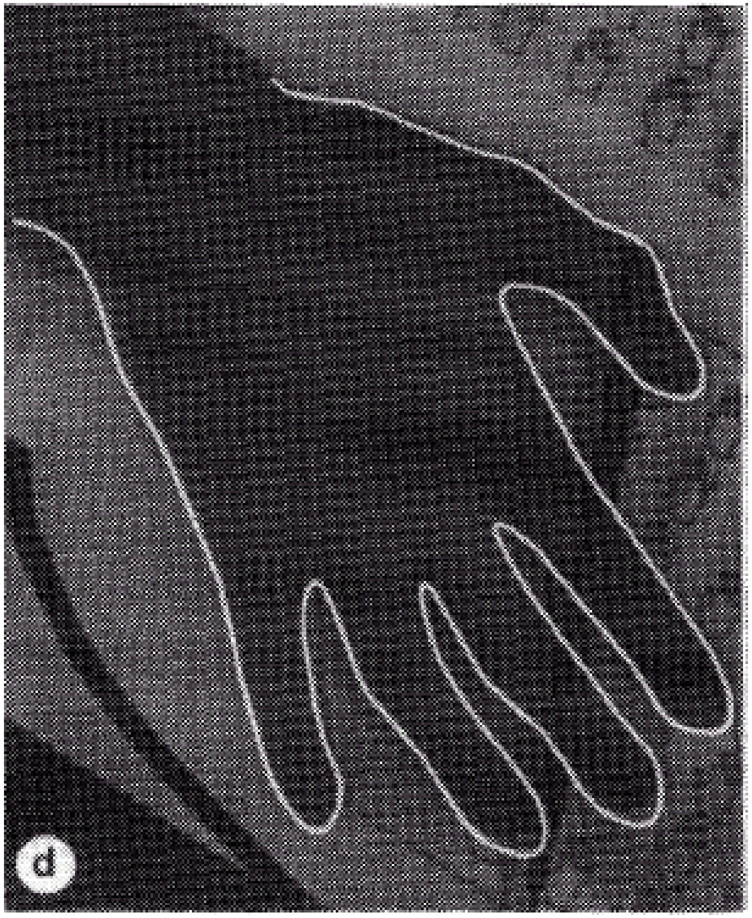
- 72 points
- Clutter and occlusions
- 8 degrees of freedom
- Adjustments made finding strongest edge
- 100, 200, 350 iterations



ASM Application to Hand



ASM Application to Hand



Applications

- Medical
- Industrial
- Surveillance
- Biometrics

Conclusions

- Object identification and location is robust.
- Constraint to be similar to shapes of the training sets.

Extension

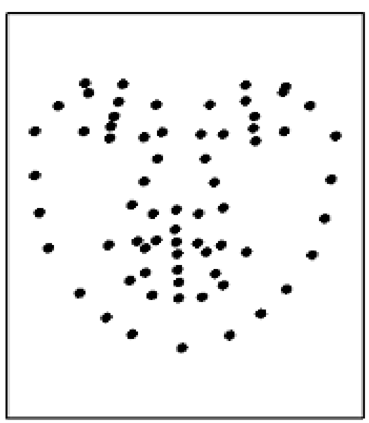
Active Appearance Model

1. T.F.Cootes, G.J. Edwards and C.J.Taylor. "Active Appearance Models", in Proc. European Conference on Computer Vision 1998 (H.Burkhardt & B. Neumann Ed.s). Vol. 2, pp. 484-498, Springer, 1998
2. T.F.Cootes, G.J. Edwards and C.J.Taylor. "Active Appearance Models", IEEE PAMI, Vol.23, No.6, pp.681-685, 2001

Active Appearance Model

Model

“Shape”



“texture”

Active Appearance Model

Initial 2 its 8 its 14 its 20 its converged

Thank You

References

1. Cootes, Taylor, Cooper, Graham, “Active Shape Models: Their Training and Application.” Computer Vision and Image Understanding, V16, N1, January, pp. 38-59, 1995.
2. T.F.Cootes, G.J. Edwards and C.J.Taylor. "Active Appearance Models", in Proc. European Conference on Computer Vision 1998 (H.Burkhardt & B. Neumann Ed.s). Vol. 2, pp. 484-498, Springer, 1998
3. T.F.Cootes, G.J. Edwards and C.J.Taylor. "Active Appearance Models", IEEE PAMI, Vol.23, No.6, pp.681-685, 2001