Segmentation: Shape Analysis

Bulat Ibragimov

bulat@di.ku.dk

Department of Computer Science
University of Copenhagen

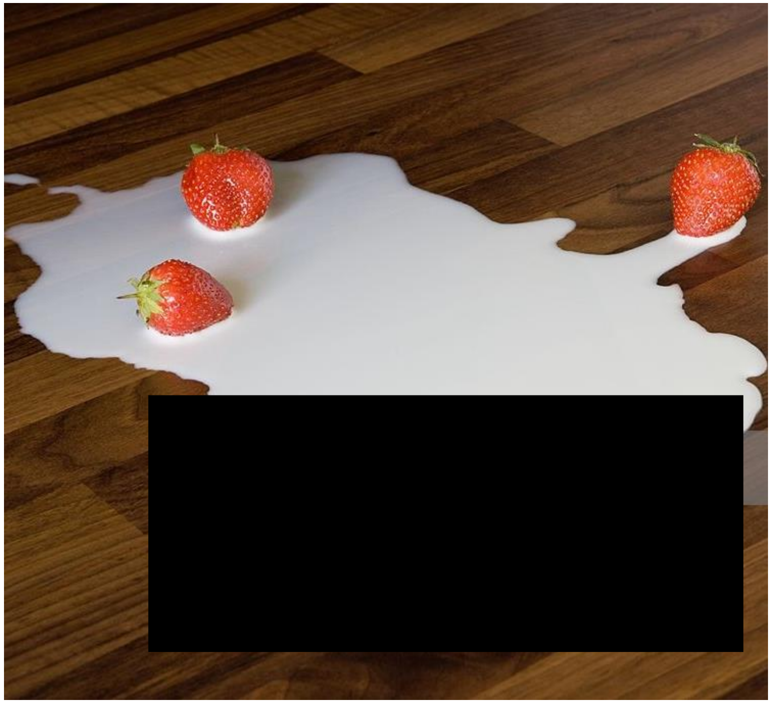
UNIVERSITY OF COPENHAGEN

Learning Objectives

- How shape can be used in medical imaging
- Procrustes analysis
- Active shape models
- What are the advantages of shape-based segmentation

What is shape

Which one is easier to recover?

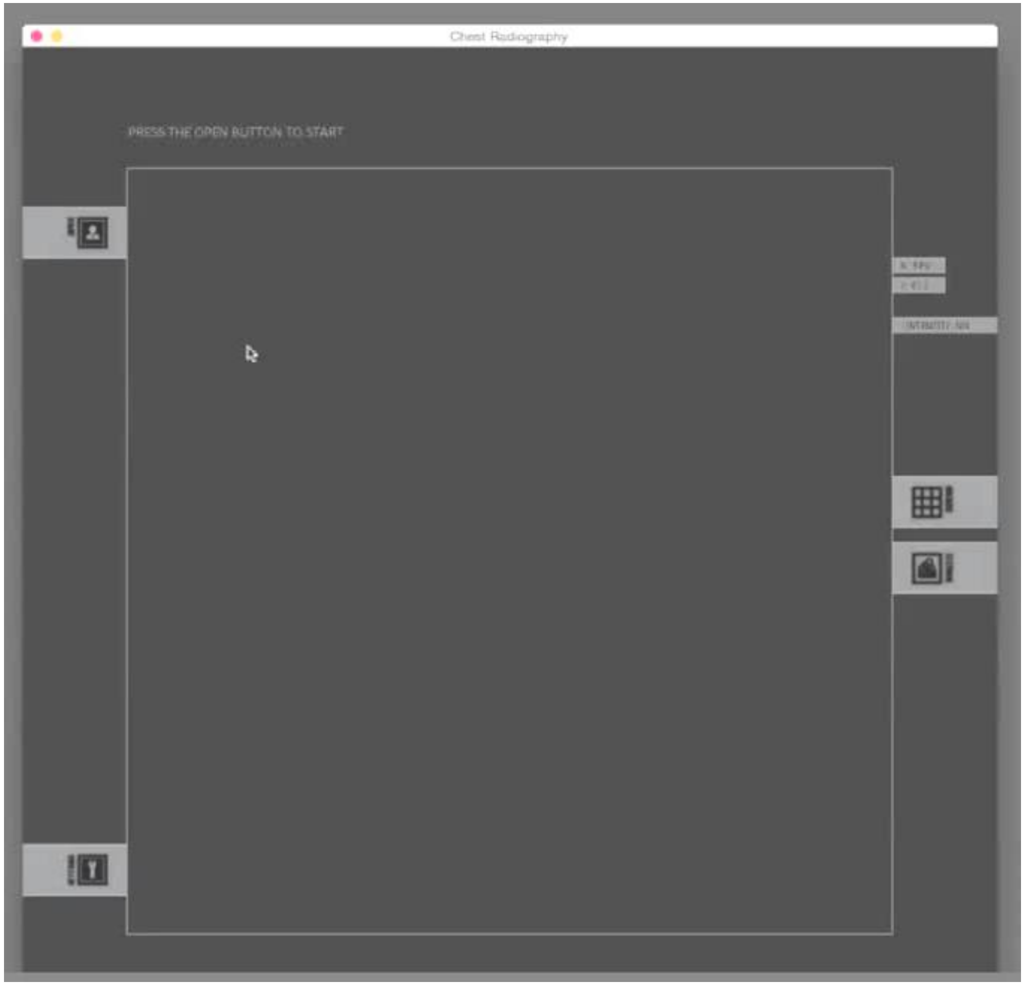


Border of the milk puddle

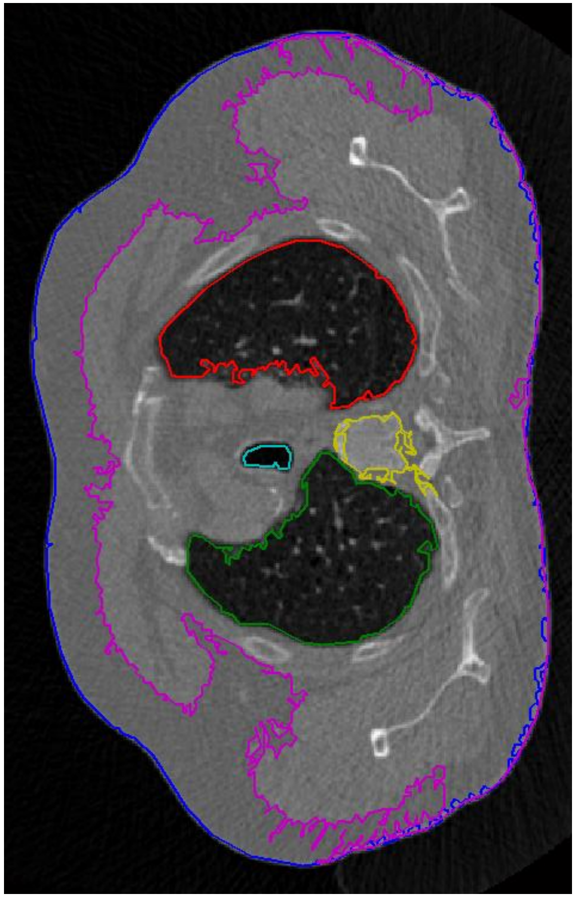
Border of the left lung

What is shape

Is intensity information sufficient to segment medical images?



What is shape



- Can we use something except the fact that lung fields are darker that surrounding structures?
- Even without having intensities and good boundaries, we can visually recognize lung fields. What do we rely on?

What is shape

Different shape definitions:

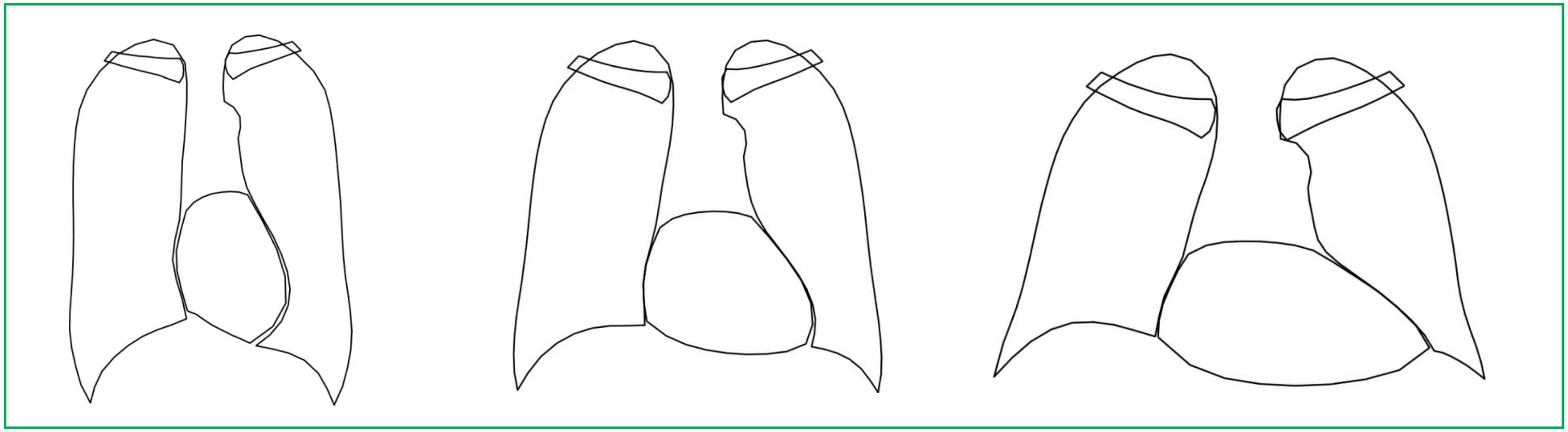
- The form of an object or its external boundary, outline, or external surface
- All geometrical properties of an object that remain after excluding location, scale and translation

Shape samples of the same object type:

- How do we understand that two objects have similar shapes?
- Can we measure how similar is the shape of an observed object and the target object type?
- Can we automatically generate new samples of the object?

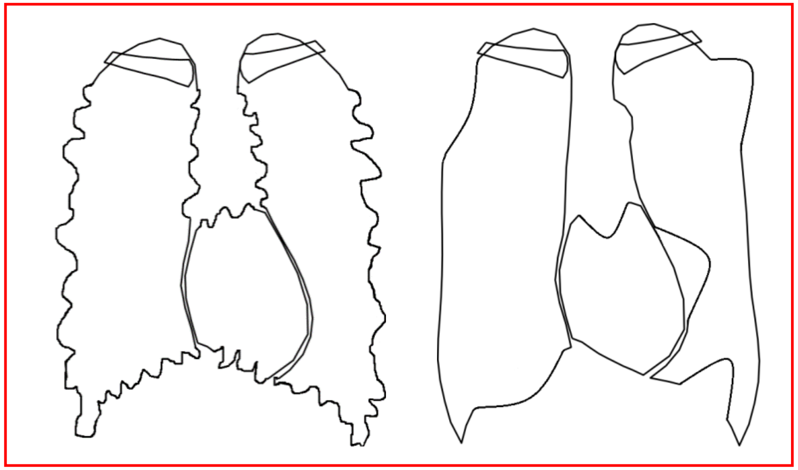
What is shape

How do we understand that two objects have similar shapes?



Shape similarity **cannot** be solely defined by:

- Boundary-to-boundary distance
- Counting shape corners
- Measuring shape angles
- Measuring shape depressions

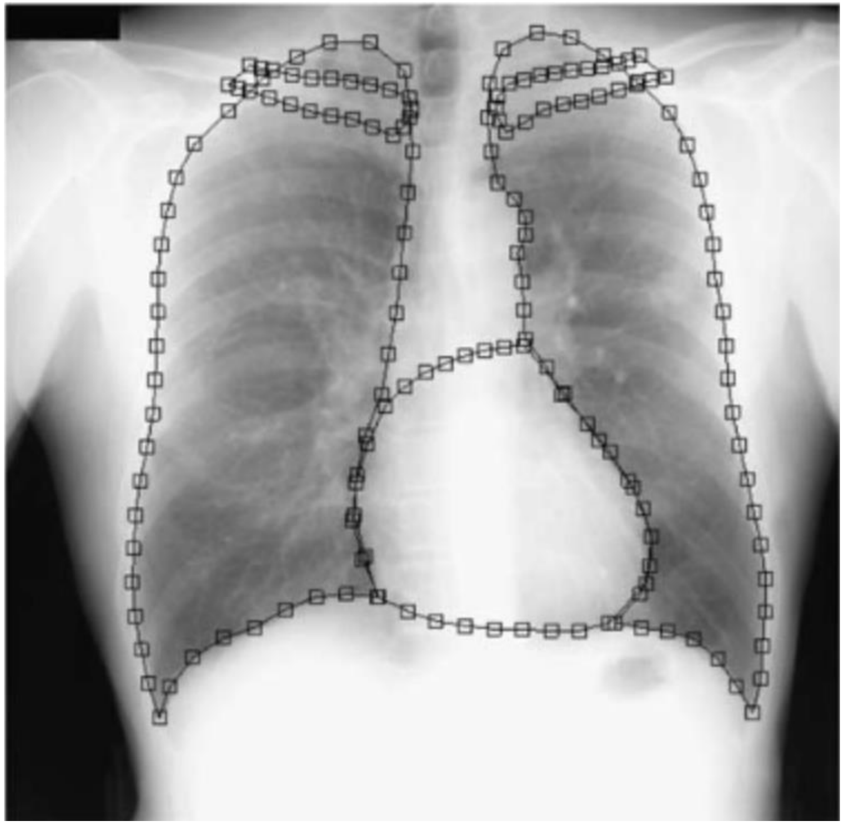


Landmarks for shape definition

Let's say lung field shape is defined with 100 points in 2D X-rays.

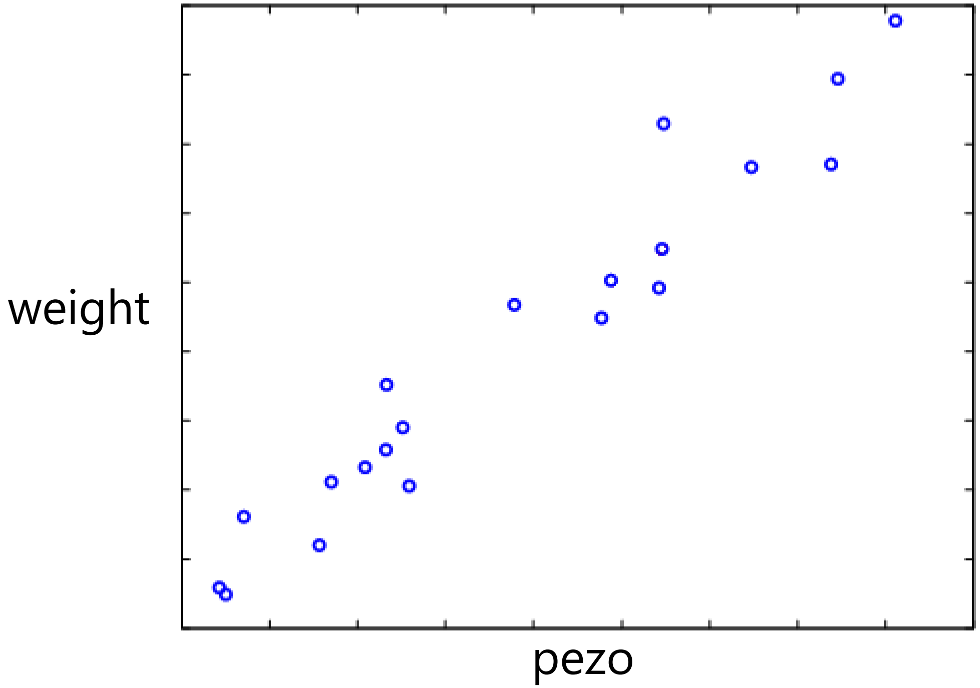
Does this mean that lung shape is 200-dimensional?

Will we get a lung shape if we randomly sample 100 2D points?



Dimensionality reduction

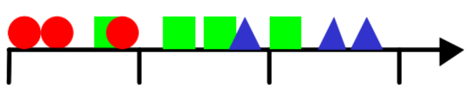
- Let's say you got a set of observations with two features {weight, pezo}.
- What is the dimensionality of this set, considering its plot?



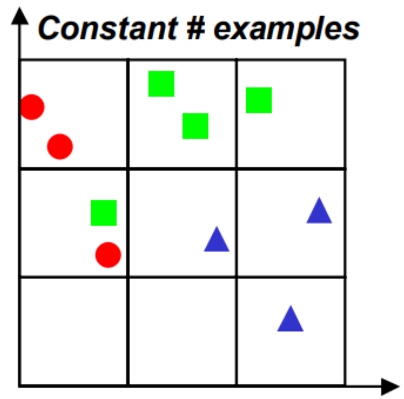
- The true dimensionality is one, and there was just some imperfection in measurements (pezo = weight in Esperanto)

Curse of dimensionality

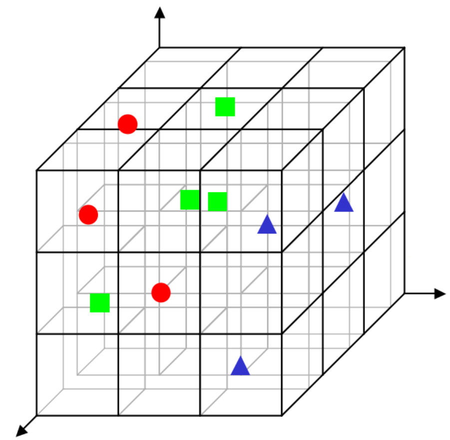
- Why do we need to estimate the true dimensionality?
- Machine learning relies very much on statistics:
  - As dimensionality grows, the samples become sparser



Sample in 1D



Sample in 2D



Sample in 3D

- We will not be able to generalize if the number of samples is not appropriate for problem dimensionality:
  - One person with a rare name X is carpenter, dose this mean

$$P(job = carpenter \mid name = X) = 1$$

Figure from http://research.cs.tamu.edu/prism/lectures/iss/iss_l10.pdf

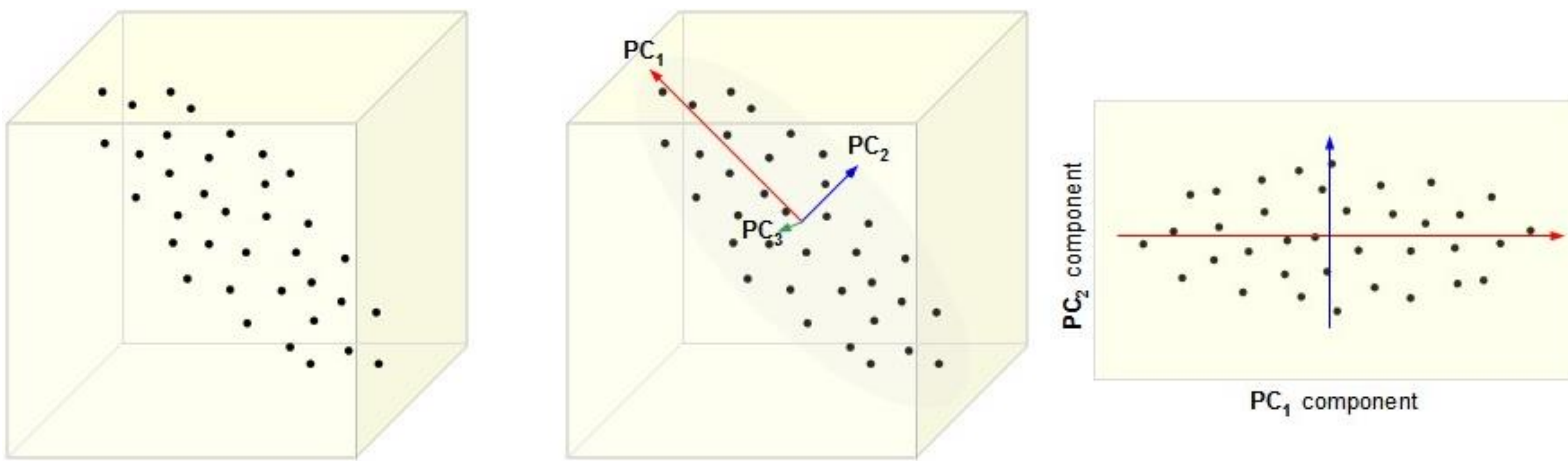
Dimensionality reduction

- Use prior knowledge:
  - We know that some landmarks are located between others, so maybe we can remove them.
- Use feature selection:
  - Go through all features and compute its importance for the prediction
  - Gini coefficient, entropy
- Use feature extraction:
  - Construct new set of features $Y = \{y_1, y_2, y_K\}$ from the original set $X = \{x_1, x_2, x_N\}$, where $K << N$
  - $y_i = f_i(x_1, x_2, x_N)$
- Principal component analysis is based on the idea of feature extraction

Principal component analysis (PCA)

- Data is defined with principal components:
  - 1st component captures the direction of the greatest data variability
  - 2nd component is orthogonal to 1st and captures the greatest variability of what is left
  - …

- First m components form $f_i()$ to generate new data features:

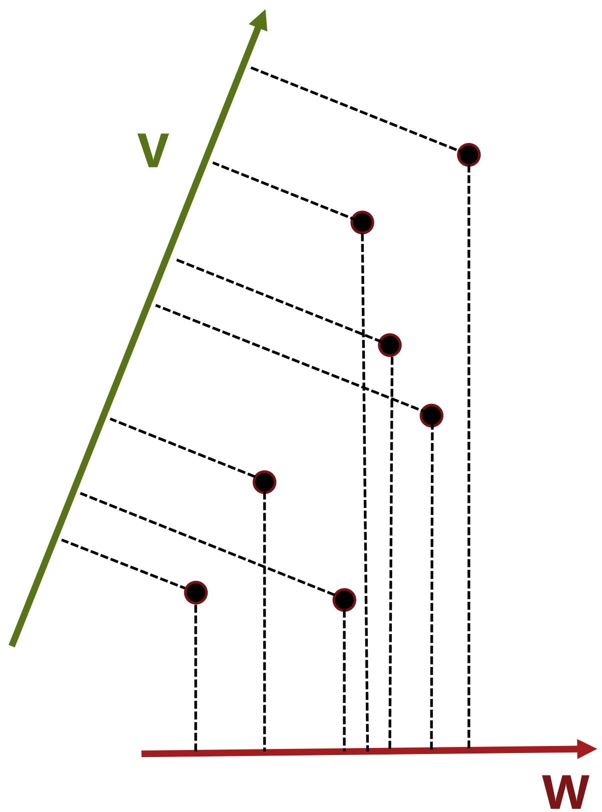


https://cnx.org/resources/269e78e2506cdbdc7f45ca17bb7d83491ce0c644/pca.jpg

PCA: maximal variance formulation

- Idea is to iteratively project the data to the direction of the maximum variance. Why do we care about maximizing it?

- Example: reduce dimensionality of 2D points to 1D:
  - Projection to V results in higher variance than projection to W
  - Projection to V preserve distances between way better
  - We would like distant objects to remain distant to preserve the logical differences between them



PCA: maximal variance formulation

- Center all the data to zero mean:

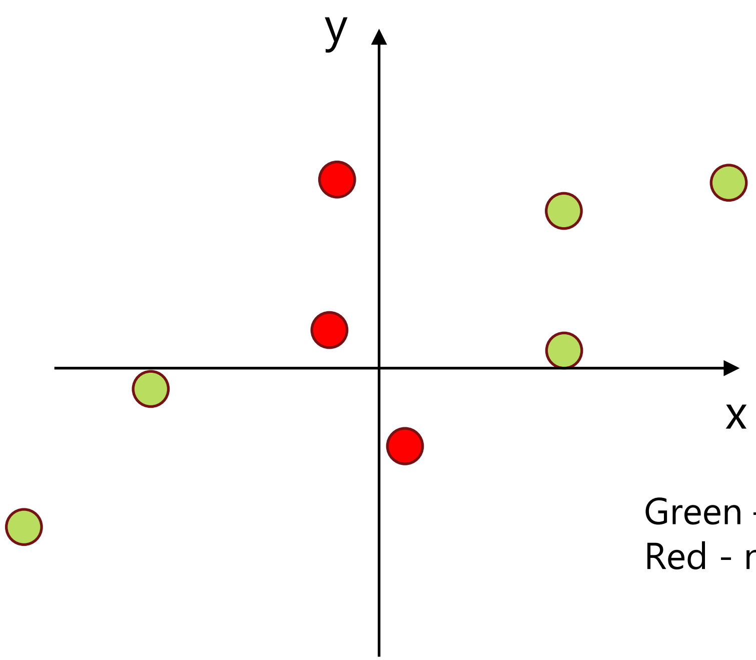
$$\mathbf{p}_\mathrm{i} = \mathbf{r}_\mathrm{i} - \bar{\mathbf{r}}$$

- Compute covariance matrix:
  - Covariance matrix shows if different dimensions increase/decrease together

$$\boldsymbol{\Sigma} = \mathbf{cov}\{\mathbf{r}\} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{p}\mathbf{p}^\mathbf{T}$$

$$\begin{array}{ccc} & \mathrm{x} & \mathrm{y} \\ \mathrm{x} & \multicolumn{2}{c}{\multirow{2}{*}{$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}$}} \\ \mathrm{y} & & \end{array}$$

Positive; x and y increase and decrease together



Green – points that contribute positively to covariance
Red - negatively

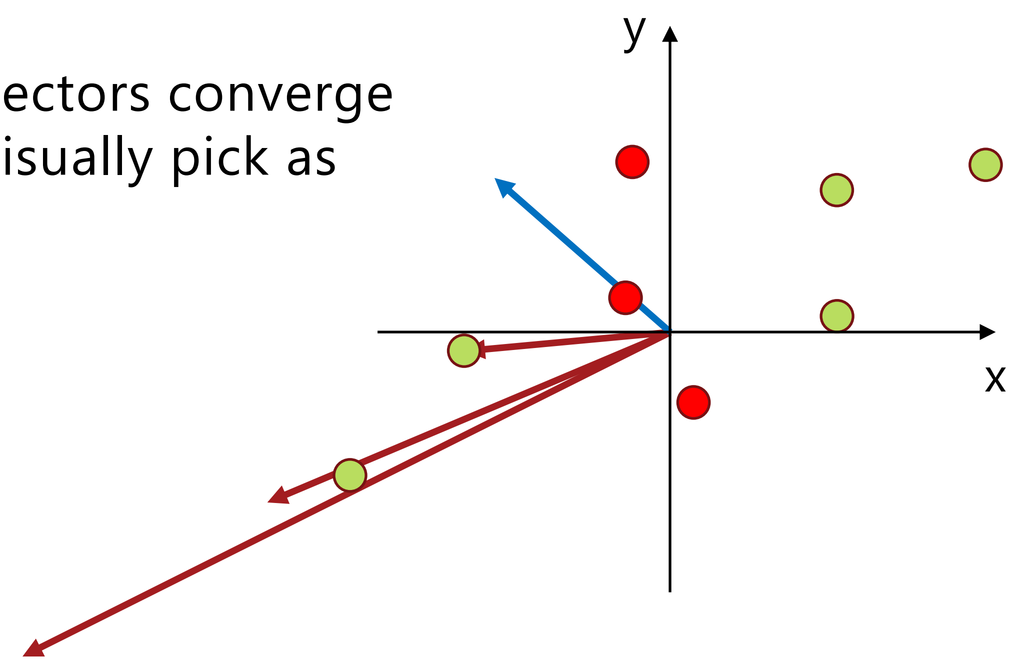
PCA: maximal variance formulation

- Let's take an arbitrary vector (-1, 1) and multiply with $\mathbf{\Sigma}$: $\begin{pmatrix} 2 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}$

$$\begin{pmatrix} 2 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \begin{pmatrix} -1 \\ 1 \end{pmatrix} = \begin{pmatrix} -1.2 \\ -0.2 \end{pmatrix}$$

- Let's multiply the result with $\mathbf{\Sigma}$ several times:
  - 1^{st} multiplication = $\begin{pmatrix} -1.2 \\ -0.2 \end{pmatrix}$; 2^{nd} = $\begin{pmatrix} -2.5 \\ -1.0 \end{pmatrix}$; 3^{rd} = $\begin{pmatrix} -6.0 \\ -2.7 \end{pmatrix}$; 4^{th} = $\begin{pmatrix} -14.1 \\ -6.4 \end{pmatrix}$; 5^{the} = $\begin{pmatrix} -33.3 \\ -15.1 \end{pmatrix}$

- The slopes of the resulting vectors converge to the direction you would visually pick as the 1^{st} principal component:
  - 1^{st} slope = 0.17
  - 2^{nd} = 0.4
  - 3^{rd} = 0.45
  - 4^{th} = 0.454
  - 5^{th} = 0.454



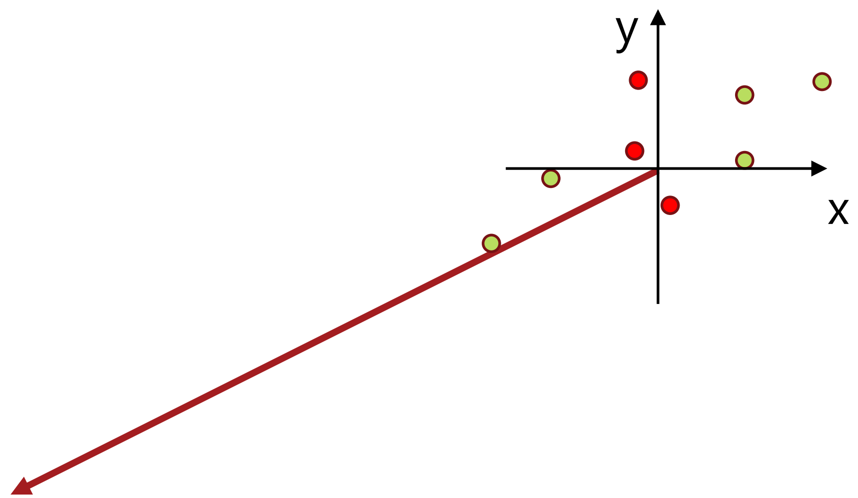
PCA: maximal variance formulation

- The eigenvectors $\mathbf{e}$ do not turn when multiplied by $\boldsymbol{\Sigma}$:

$$\boldsymbol{\Sigma}\mathbf{e} = \lambda\mathbf{e}; \qquad \|\mathbf{e}\| = 1$$

- The weighting coefficients λ are called eigenvalues; they encode contribution of the corresponding eigenvector

- From previous example
  - Vector at 5th multiplication = $\begin{pmatrix} -33.3 \\ -15.1 \end{pmatrix}$
  - After normalization $\begin{pmatrix} -0.91 \\ -0.41 \end{pmatrix}$
  - $\begin{pmatrix} 2 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \begin{pmatrix} -0.91 \\ -0.41 \end{pmatrix} = \begin{pmatrix} -2.15 \\ -0.97 \end{pmatrix}$
  - $\begin{pmatrix} -2.15 \\ -0.97 \end{pmatrix} \approx \lambda \begin{pmatrix} -0.91 \\ -0.41 \end{pmatrix}$; λ=2.38



- The true first eigenvalue = 2.36

PCA: maximal variance formulation

- To find eigenvectors and eigenvalues, first solve $\det(\boldsymbol{\Sigma} - \lambda \mathrm{I}) = 0$:

$$\det\begin{pmatrix} 2.0-\lambda & 0.8 \\ 0.8 & 0.6-\lambda \end{pmatrix} = (2-\lambda)(0.6-\lambda) - 0.8 \cdot 0.8 = \lambda^2 - 2.6\lambda + 0.56 = 0$$

$$\{\lambda_1, \lambda_2\} = \{2.36, 0.23\}$$

- Find eigenvectors by solving $\boldsymbol{\Sigma}\mathbf{e} = \lambda\mathbf{e}$:

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}\begin{pmatrix} e_{1,1} \\ e_{1,2} \end{pmatrix} = 2.36\begin{pmatrix} e_{1,1} \\ e_{1,2} \end{pmatrix} \rightarrow \begin{matrix} 2.0e_{1,1} + 0.8e_{1,2} = 2.36e_{1,1} \\ 0.8e_{1,1} + 0.6e_{1,2} = 2.36e_{1,2} \end{matrix}$$

What is the problem with this system?

This system is redundant so many solutions exists Simply multiply a solution $\mathbf{e}$ by a constant and get a new one. We can only get proportion:

$$e_{1,1} = 2.2e_{1,2}$$

- Impose condition of unit norm $||\mathbf{e}|| = 1$, and the proportion:

$$\mathbf{e} = [0.91, 0.41]$$

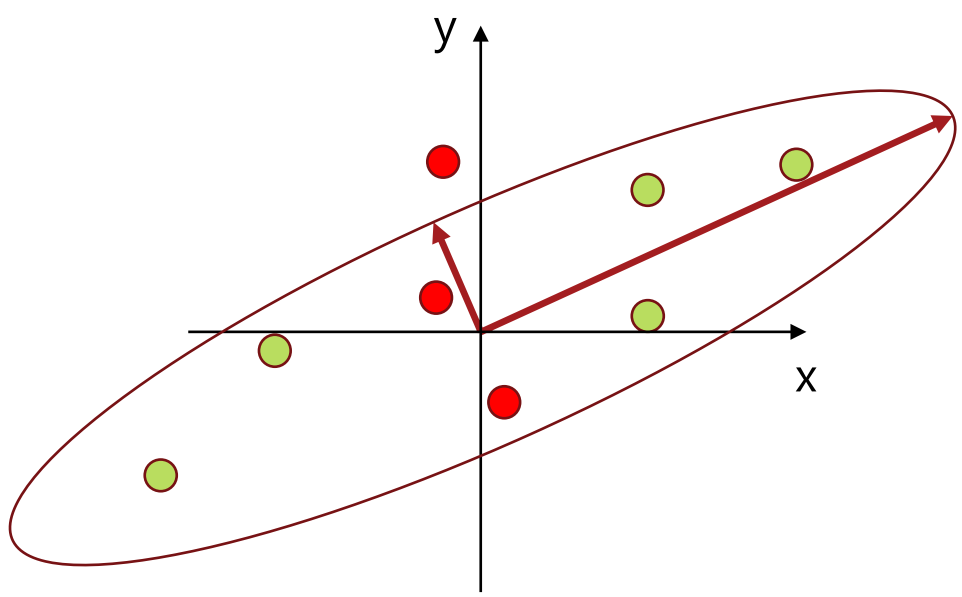
PCA: maximal variance formulation

- The second eigenvector is:

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \begin{pmatrix} e_{2,1} \\ e_{2,2} \end{pmatrix} = 0.23 \begin{pmatrix} e_{2,1} \\ e_{2,2} \end{pmatrix} \rightarrow \begin{matrix} 2.0e_{2,1} + 0.8e_{2,2} = 0.23e_{2,1} \\ 0.8e_{2,1} + 0.6e_{2,2} = 0.23e_{2,2} \end{matrix}$$

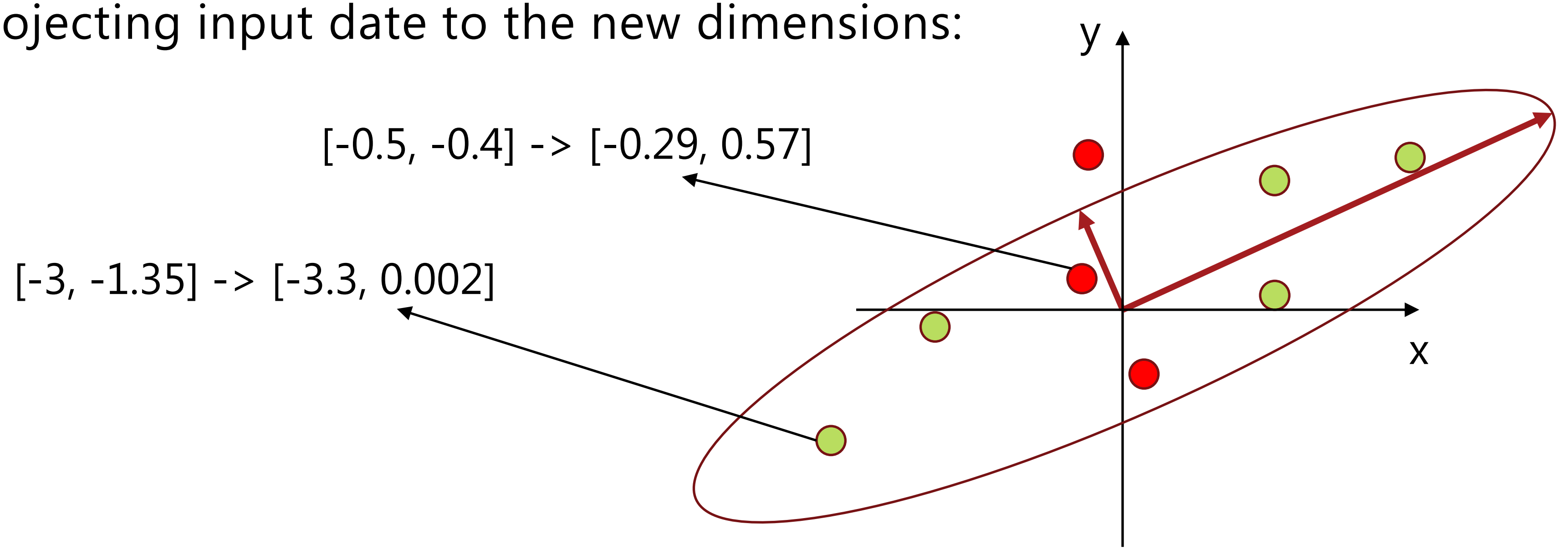
$$\mathbf{e} = [-0.41, 0.91]$$

- The eigenvectors are orthogonal so they define a 2D basis:



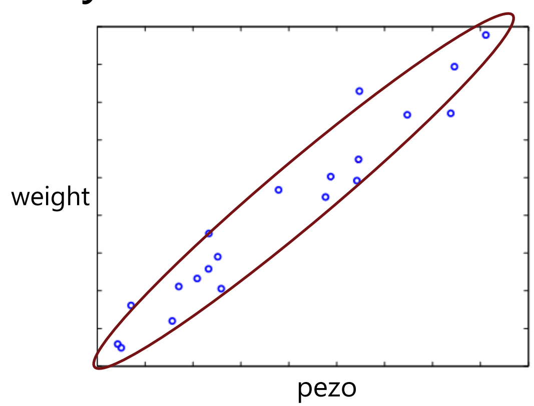
PCA: maximal variance formulation

- Projecting input date to the new dimensions:



- Do eigenvectors and eigenvalues give us any useful information?

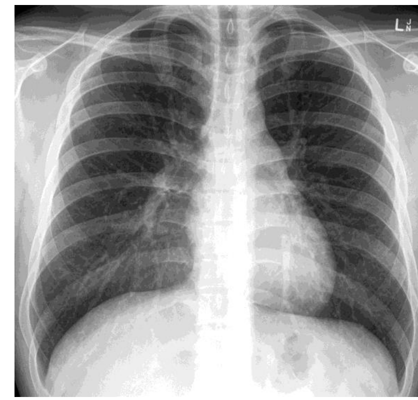
For weight/pezo example, the second eigenvalue is very small, so the data is almost 1-dimensional

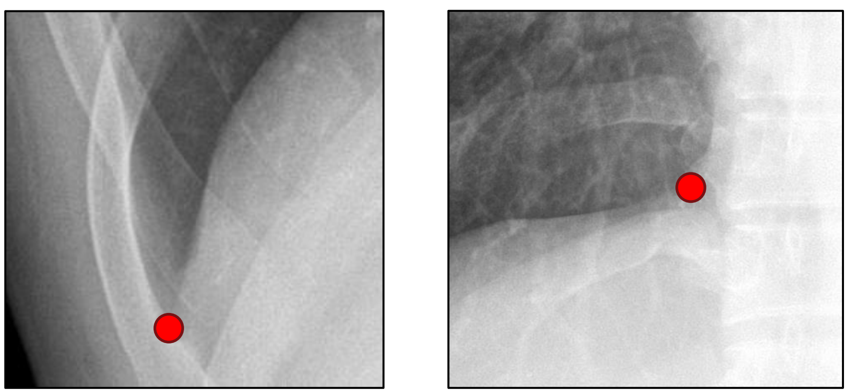


Shape: landmarks

Landmark in medical imaging:

- A point in the image that can be uniquely located according to certain predefined rules:
    - Curvature extrema
    - Terminal points
    - Centerline intersections





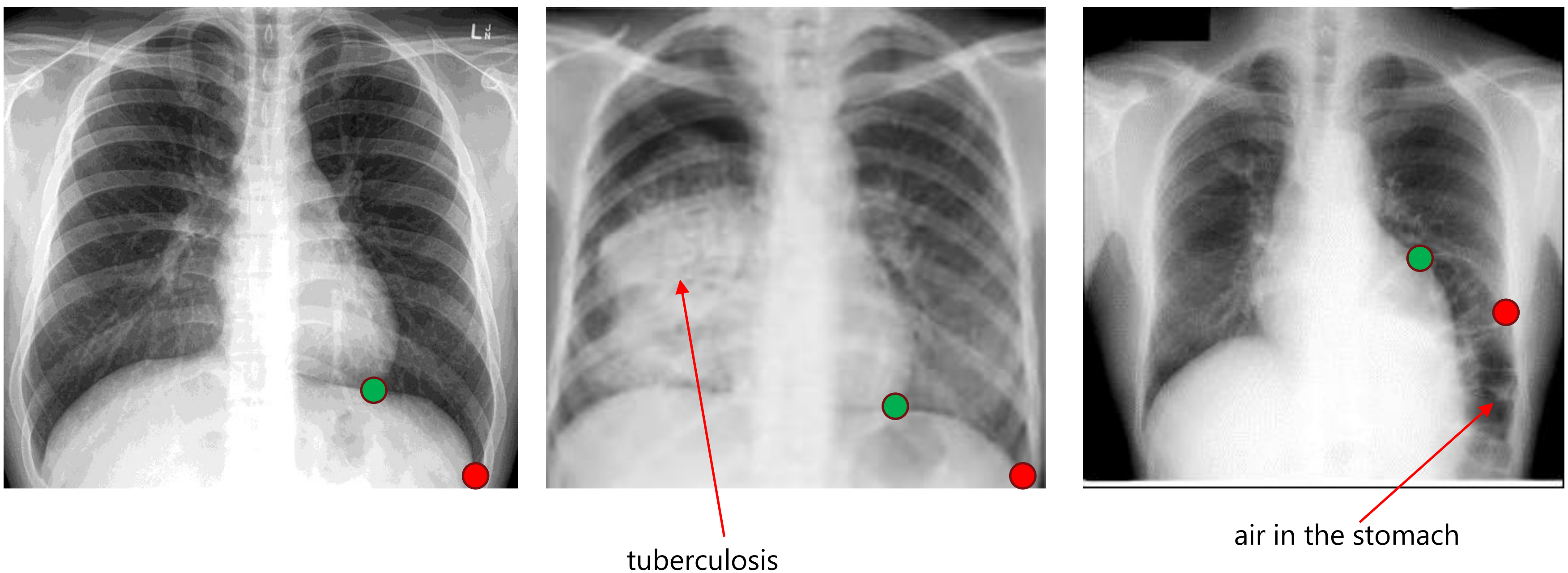
Visually-recognizable points

Evenly distributed points

Shape: landmarks

Landmark establish correspondence between images of the same type:

- Landmarks are defined is such a way that they can be found in any object



Break

How can we use PCA computed on lung shapes (100 landmarks each) to:

a) Check if an object defined with 100 landmarks represent an example of lungs

b) Generate new examples of lung shapes

BREAK

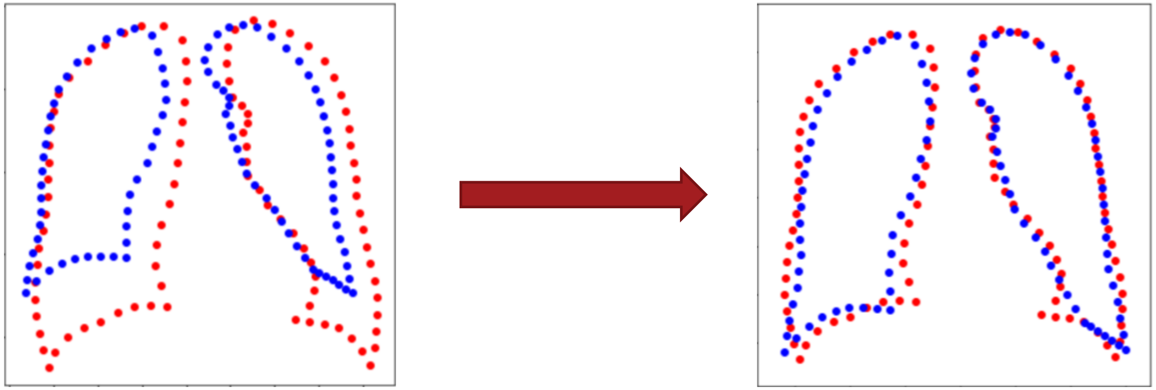
10 minutes

PCA: applications

- i-th lung field shape can be described as:

$$S_i = [[x_1, y_1], [x_2, y_2], \dots, [x_{92}, y_{92}]]$$

- Perform normalization:
  - Subtract mean from each lung field shape $\bar{S}_i = [\bar{x}, \bar{y}]$
  - It is also important to normalize to scale and rotation (check Procrustes Analysis)



Two shapes before normalization   Two shapes after normalization

- We will get an array of N = 256(# images) of M = 200 dimensional samples:

$$\mathbf{p} = \begin{bmatrix} p_{1,1} & \cdots & p_{1,M} \\ \vdots & \ddots & \vdots \\ p_{N,1} & \cdots & p_{N,M} \end{bmatrix}$$

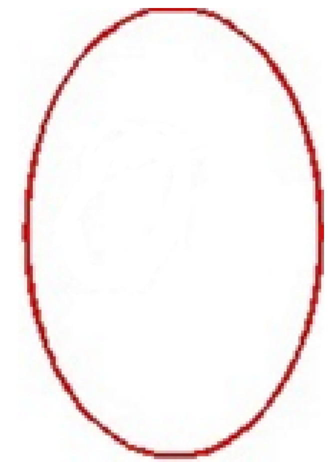
Shape: Procrustes

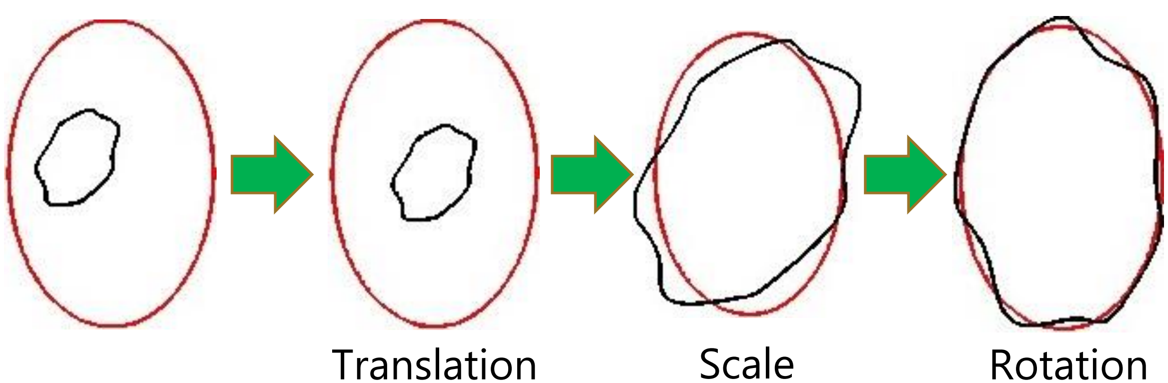
Shape: Procrustes analysis

Let we have an objects **X** and with **Y** with the same shape:

- $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]$
- $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n]$

X= 

Y= 



Shape: Procrustes analysis

Translation:

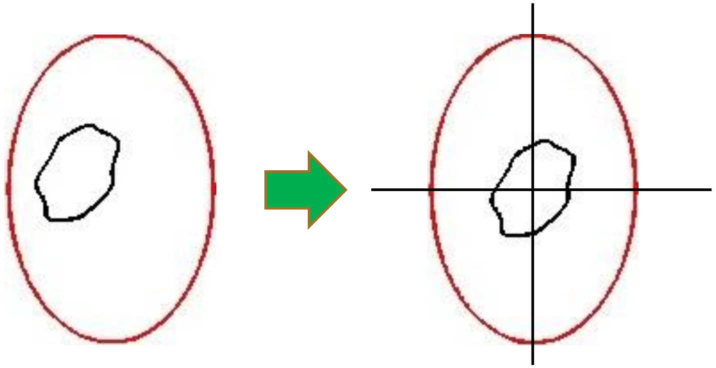
- Calculate the centers of both **X** and **Y**:

$$\bar{\mathbf{x}} = \frac{\mathbf{x}_1 + \mathbf{x}_2 + \ldots + \mathbf{x}_n}{n} \qquad \bar{\mathbf{y}} = \frac{\mathbf{y}_1 + \mathbf{y}_2 + \ldots + \mathbf{y}_n}{n}$$

- Move both **X** and **Y** to the coordinate system center:

$$\mathbf{X} \leftarrow [\mathbf{x}_1 - \bar{\mathbf{x}}, \mathbf{x}_2 - \bar{\mathbf{x}}, \ldots, \mathbf{x}_\mathrm{n} - \bar{\mathbf{x}}]$$

$$\mathbf{Y} \leftarrow [\mathbf{y}_1 - \bar{\mathbf{y}}, \mathbf{y}_2 - \bar{\mathbf{y}}, \ldots, \mathbf{y}_\mathrm{n} - \bar{\mathbf{y}}]$$



Shape: Procrustes analysis

Scale:

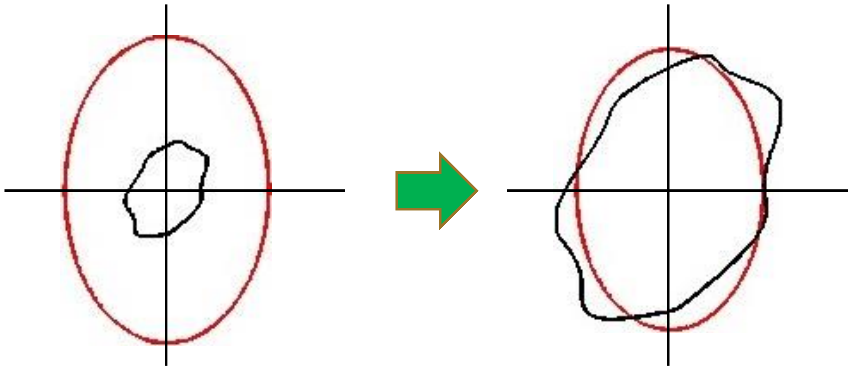
- Calculate the “size” of both **X** and **Y**:

$$s_{\mathrm{X}} = \sqrt{\frac{\|\mathbf{x}_1\| + \|\mathbf{x}_2\| + \ldots + \|\mathbf{x}_n\|}{n}} \qquad s_{\mathrm{Y}} = \sqrt{\frac{\|\mathbf{y}_1\| + \|\mathbf{y}_2\| + \ldots + \|\mathbf{y}_{\mathrm{n}}\|}{\mathrm{n}}}$$

- Scale both **X** and **Y** to the unit space:

$$\mathbf{X} \leftarrow \left[\frac{\mathbf{x}_1}{s_{\mathrm{X}}}, \frac{\mathbf{x}_2}{s_{\mathrm{X}}}, \ldots, \frac{\mathbf{x}_{\mathrm{n}}}{s_{\mathrm{X}}}\right]$$

$$\mathbf{Y} \leftarrow \left[\frac{\mathbf{y}_1}{s_{\mathrm{Y}}}, \frac{\mathbf{y}_2}{s_{\mathrm{Y}}}, \ldots, \frac{\mathbf{y}_{\mathrm{n}}}{s_{\mathrm{Y}}}\right]$$



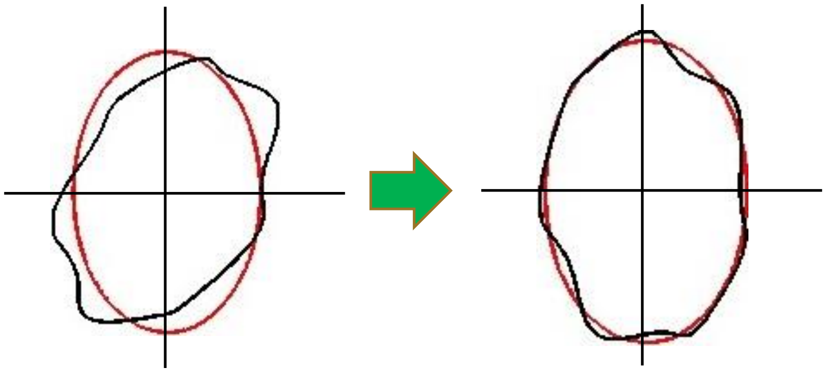
Shape: Procrustes analysis

Rotation:

- We need to chose one object as a reference, e.g. $\mathbf{X}$. The goal is to find the optimal θ:

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_n] \qquad \begin{aligned} \mathbf{u}_i[0] &= \cos(\theta)\,\mathbf{y}_i[0] - \sin(\theta)\,\mathbf{y}_i[1] \\ \mathbf{u}_i[1] &= \sin(\theta)\,\mathbf{y}_i[0] + \cos(\theta)\,\mathbf{y}_i[1] \end{aligned}$$

$$\boldsymbol{f} = \min_{\theta}\left(\sum_{i=1}^{n} \|\mathbf{u}_i - \mathbf{x}_i\|^2\right)$$



Shape: Principal component analysis

1. Compute the mean of the data

$$\overline{\mathbf{q}} = \frac{1}{M} \sum_{i=1..M} \mathbf{q}_i$$

2. Compute the covariance of the data

$$\mathbf{S} = \frac{1}{M-1} \sum_i (\mathbf{q}_i - \overline{\mathbf{q}})(\mathbf{q}_i - \overline{\mathbf{q}})^T$$

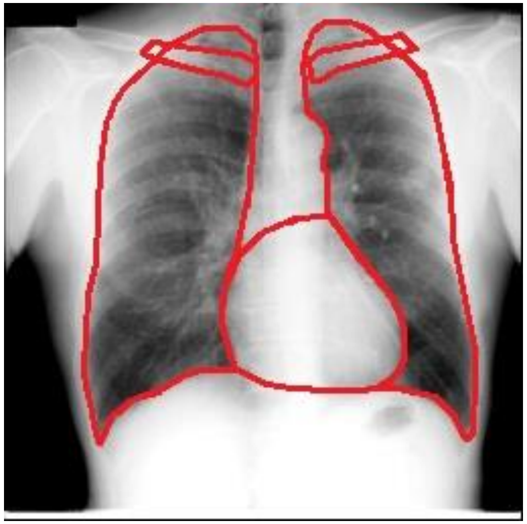
3. Compute the eigenvectors $\mathbf{u}_i$ and eigenvalues λ_i of the covariance matrix, sorted in decreasing order of eigenvalue size

4. Remove the small eigenvalues, retaining “most” (eg 98%) of the variation

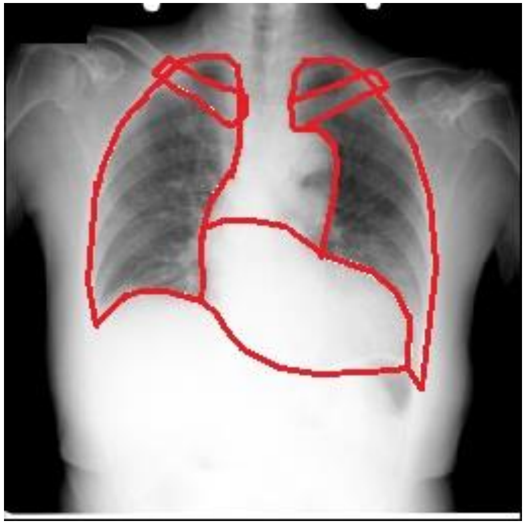
$$\text{Choose } t \text{ so that } \sum_{i=1}^{t} \lambda_i \geq 0.98 * \sum_{i=1}^{M} \lambda_i$$

How does PCA help?

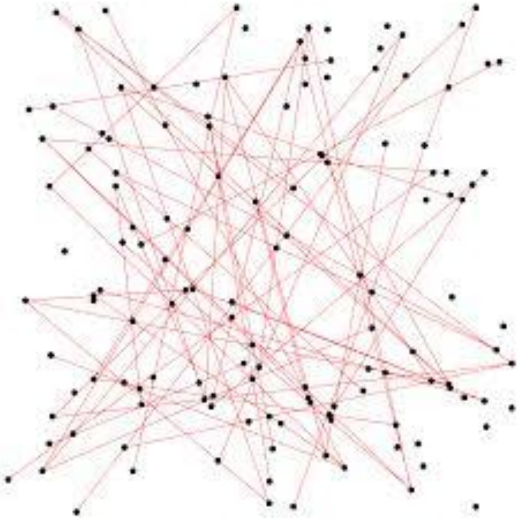
Let's describe three objects using n points and the system of principal components (eigenvectors):



Chest organs 1



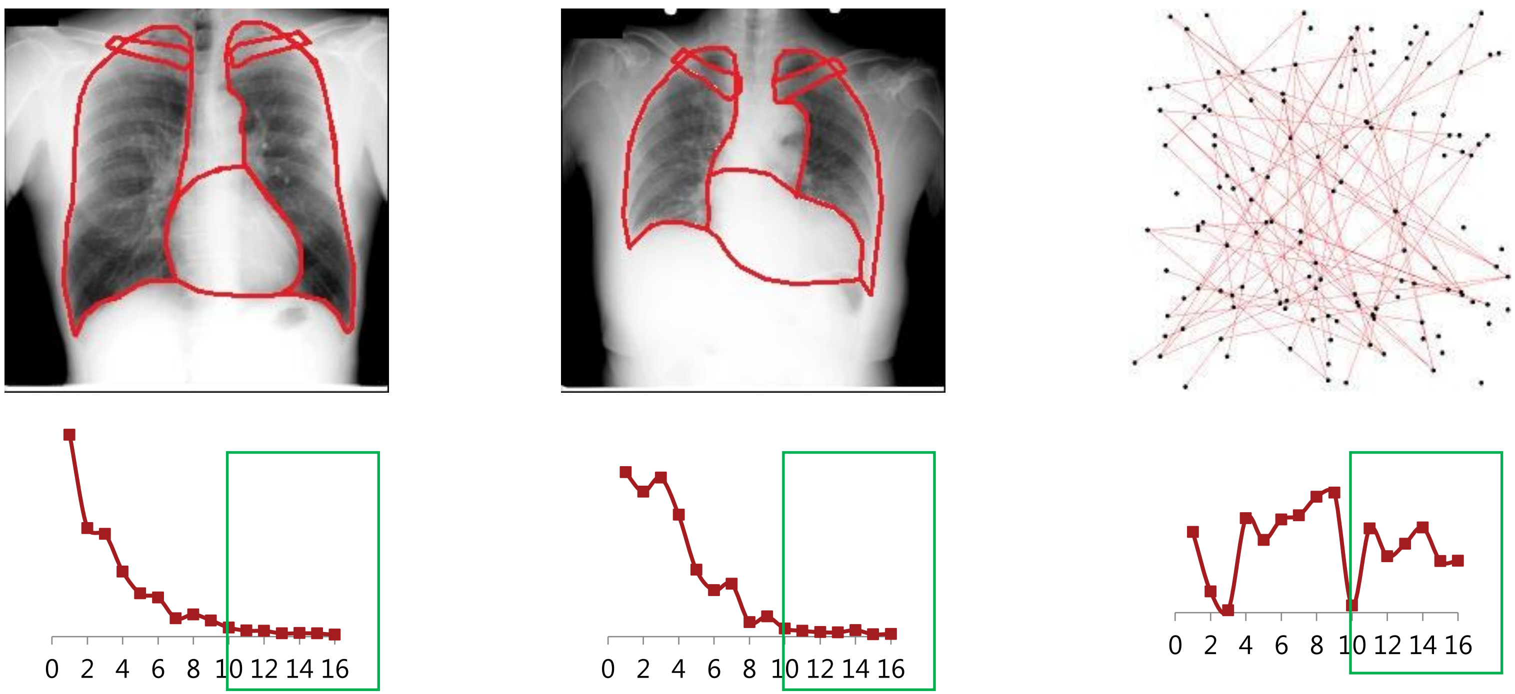
Chest organs 2



Random set of *n* points

How does PCA help?

- For random set of points, the coefficients will be random

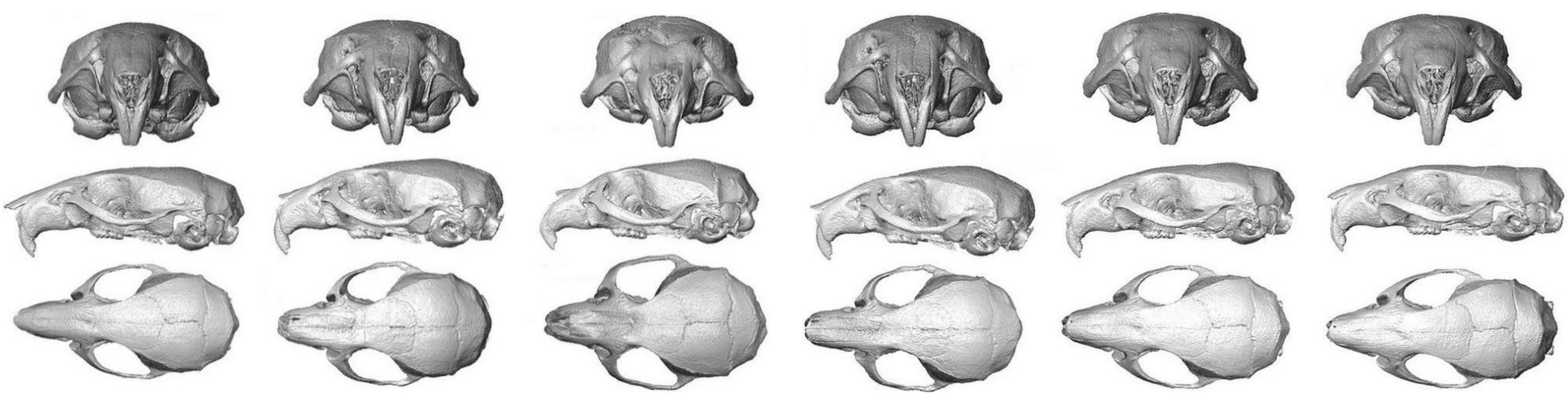


- If we remove last eigenvectors, we will not be able to accurately describe the random set of points

PCA in medical imaging

- Each eigenvector describe smooth object shape variations. So there will be not spikes that appear at random places. However, such spikes may be generated if a combination of eigenvectors with low eigenvalues are used.
- First eigenvectors usually correspond to meaningful shape variations. For example, changing an eigenvector may transform male pelvis towards female, systolic ventricle towards diastolic, etc.

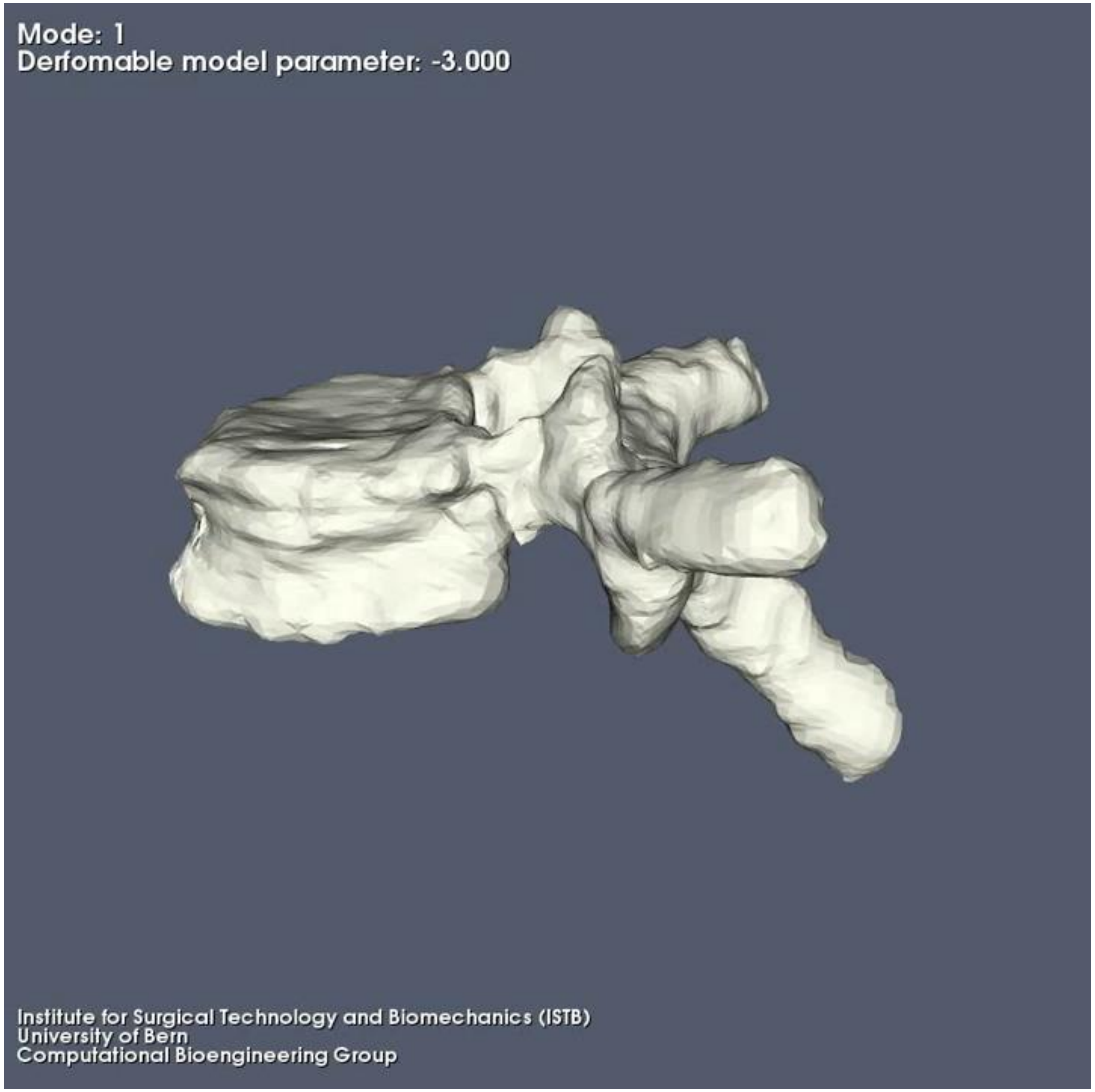
Eigenvector 1 Eigenvector 2



PCA in medical imaging

Mean

PCA in medical imaging



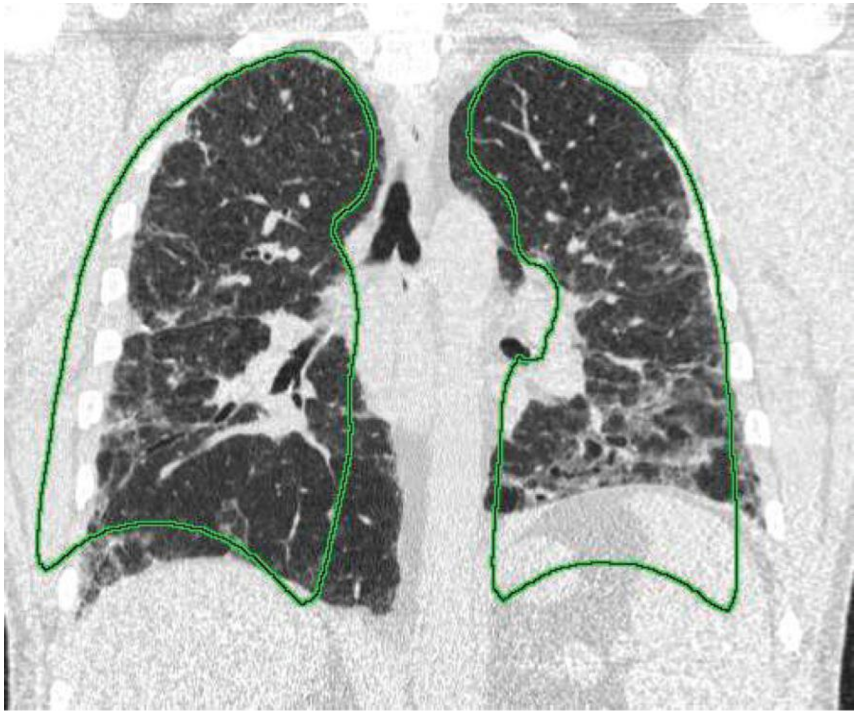
Active shape models

Supervised segmentation:

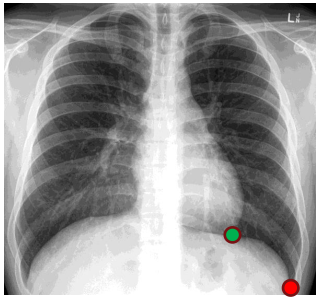
- Database of images
- All images are manually annotated

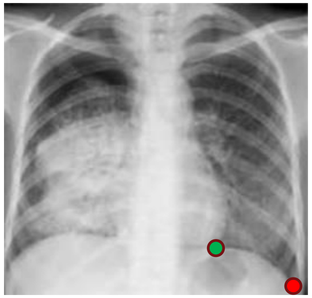
Active shape models:

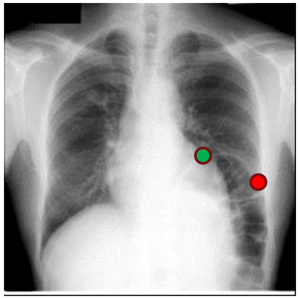
- Annotations consist of landmarks with correspondences
- Use PCA to construct the object shape model



Mean shape positioned at the testing image

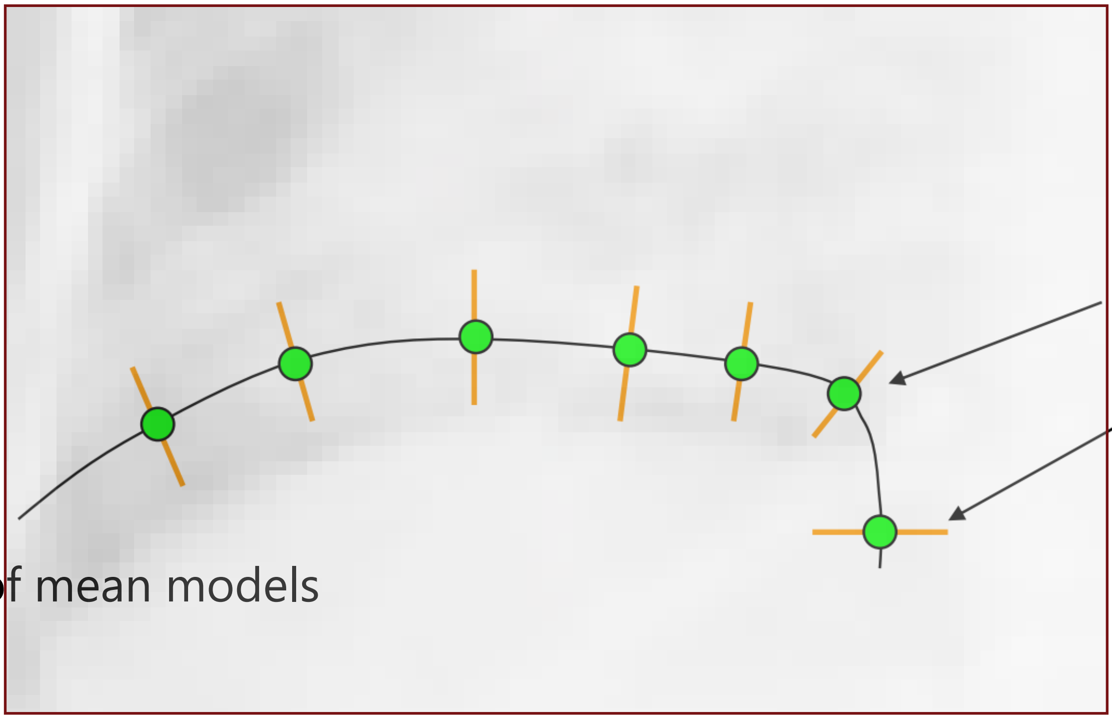






Active shape models

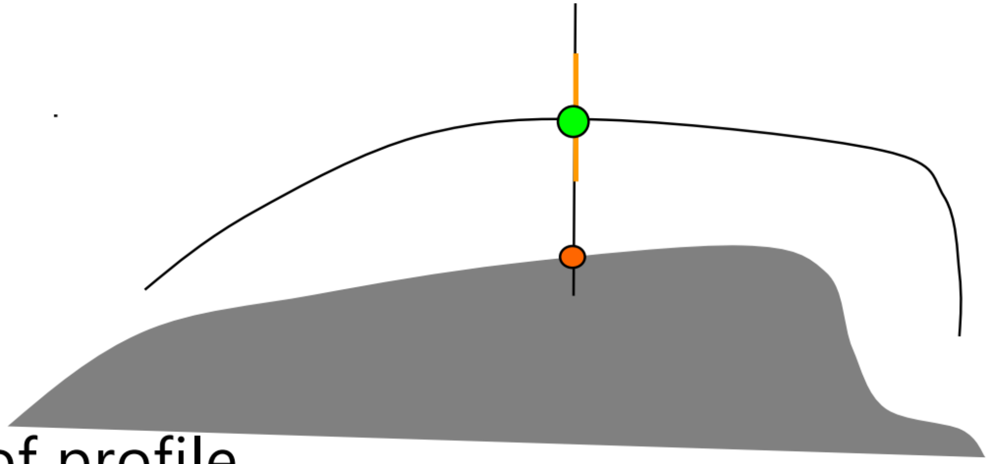
- Match shape model to new image
- Require:
  - Statistical shape model
  - Model of image structure at each point



Active shape models

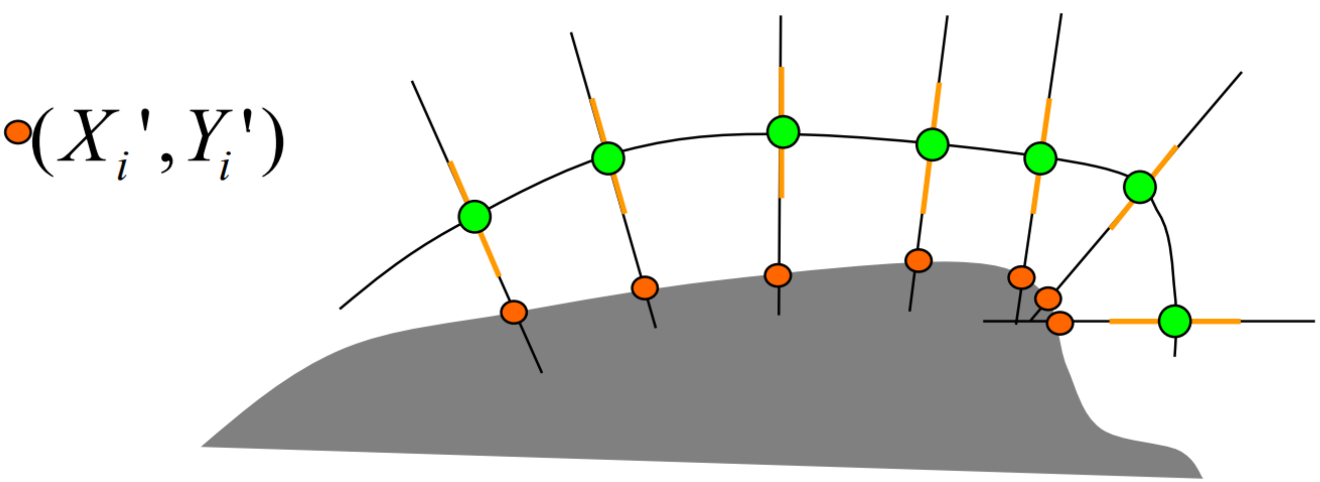
- Need to search for local match for each point

- Model
  - Strongest edge
  - Correlation
  - Statistical model of profile



Active shape models

- Local optimisation
- Initialise near target
  - Search along profiles for best match, **X**′
  - Update parameters to match to **X’**.

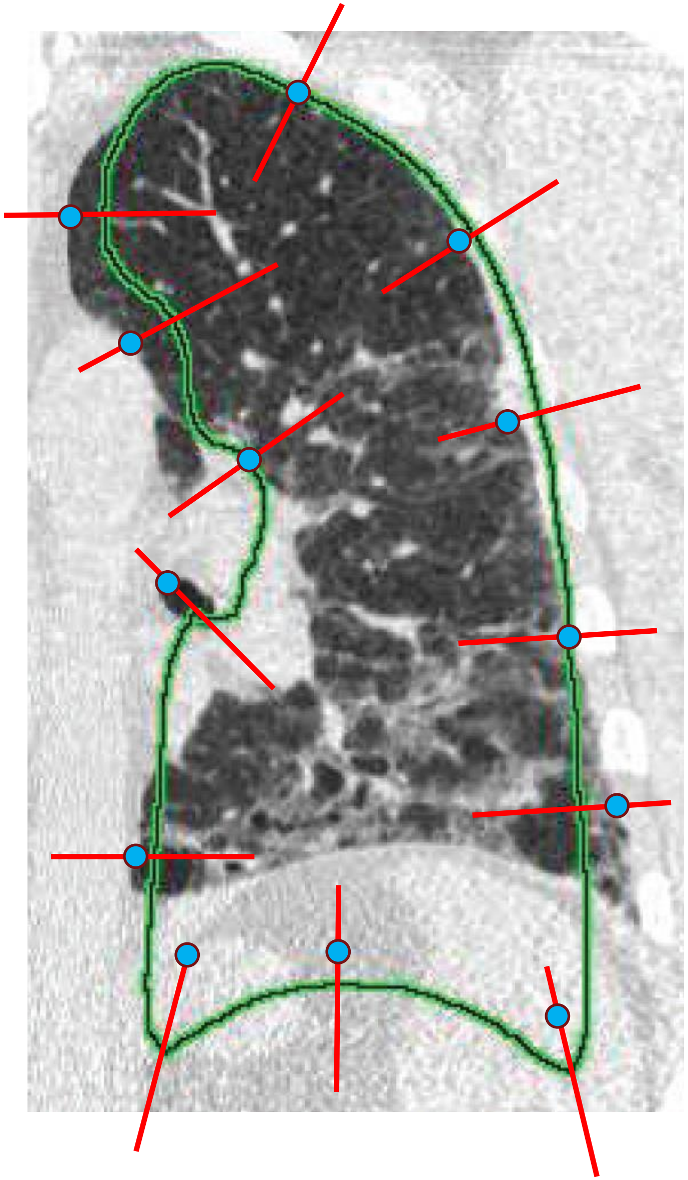


Active shape models

Supervised segmentation:

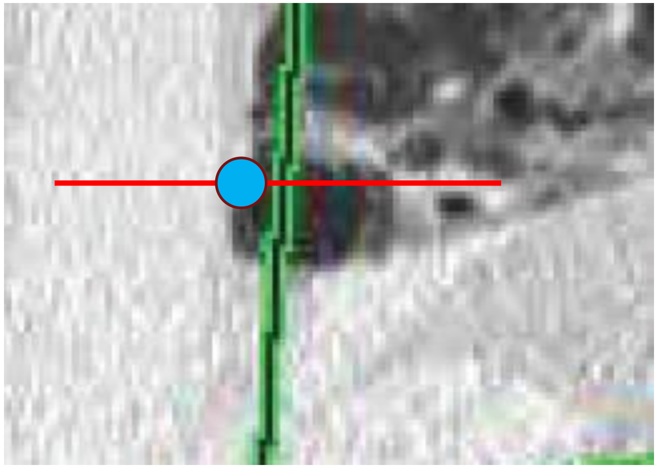
- Get new optimal points $\mathbf{X}'$
- Align current model $\mathbf{M}'$ with $\mathbf{X}'$ against translation, scaling and rotation
- Fit the aligned model $\mathbf{M}'$ with $\mathbf{X}'$ using PCA of the object
- Update $\mathbf{M}'$ and search for new along new points normal from $\mathbf{M}'$

- PCA protects segmentation from noise points in $\mathbf{X}'$ because the main eigenvectors will not fit to the noise points

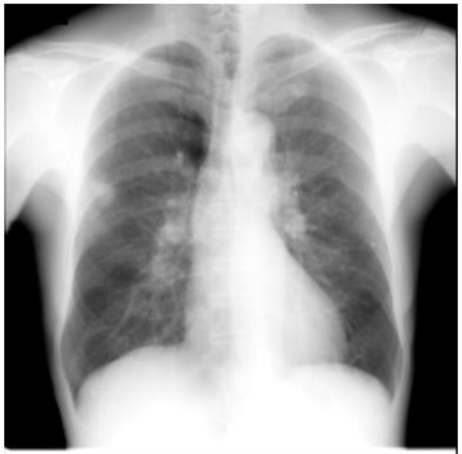


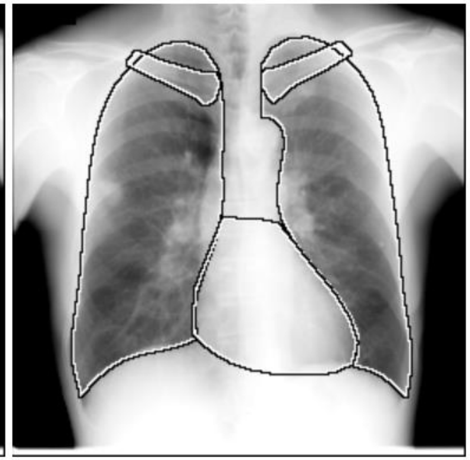
Active shape models: searching for candidate points

- A point with strongest gradient
- Using some prior knowledge:
  - Intensity model of each point

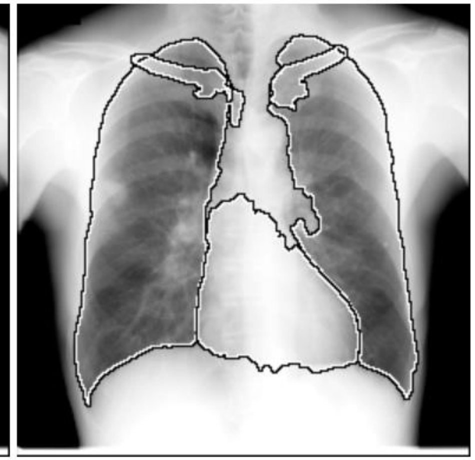


Active shape models vs pixel-based segmentation

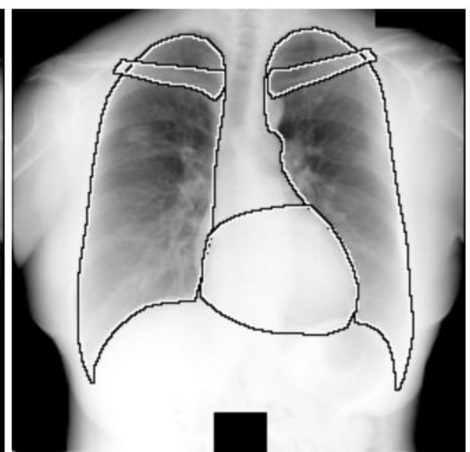




Lung Overlap 0.93



Lung Overlap 0.95

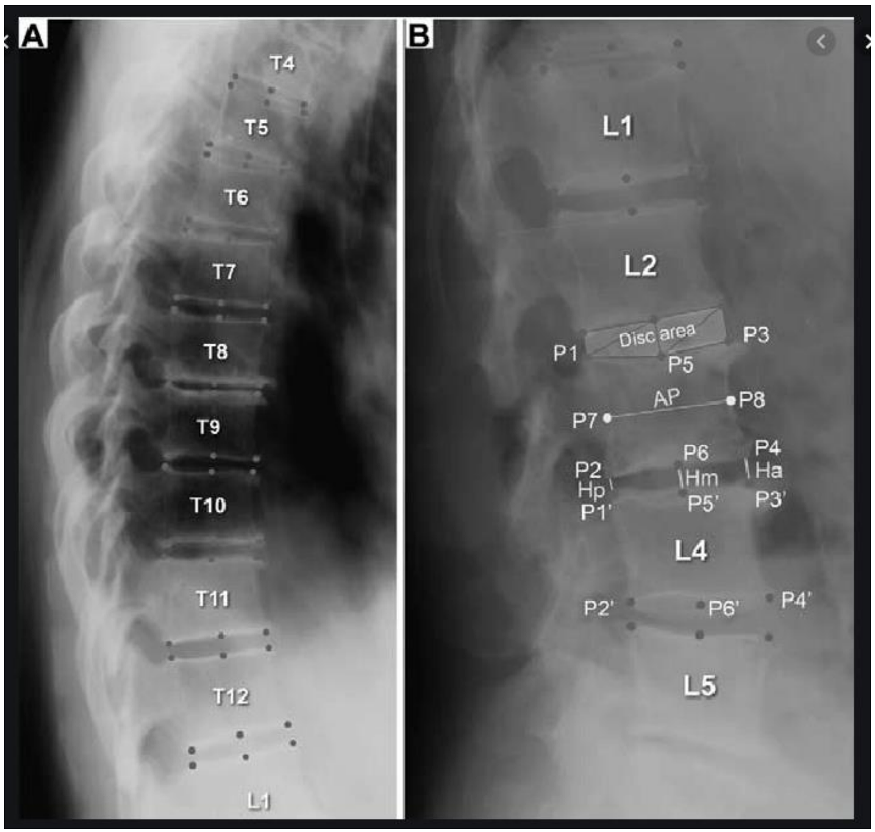


Lung Overlap 0.92

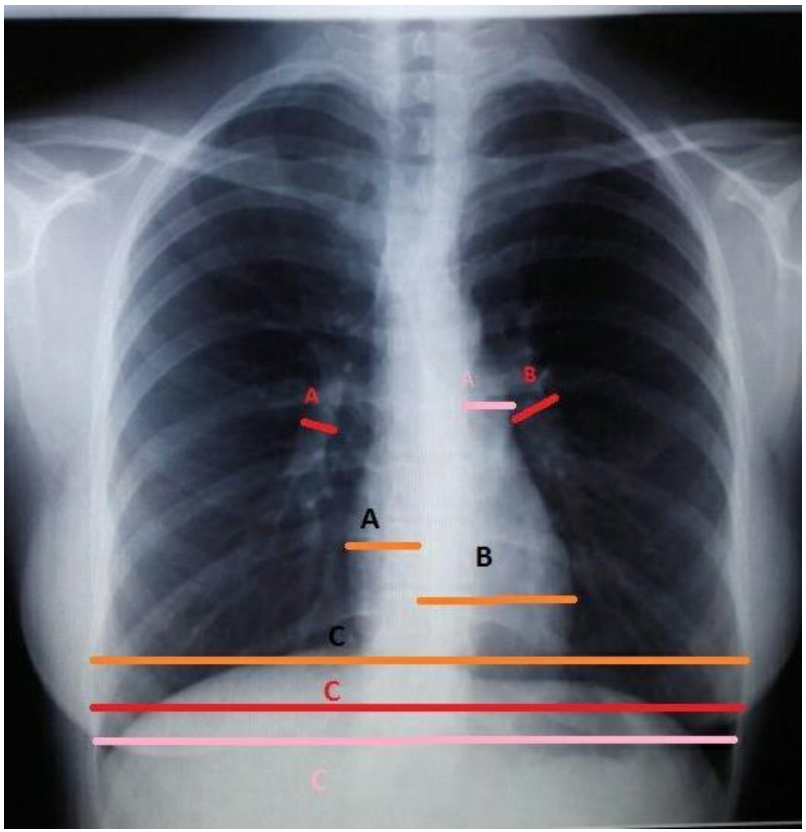
Lung Overlap 0.89

Shape models: beyond segmentation

Diagnosis of many diseases is based on organ morphometry:



Vertebral fracture diagnosis

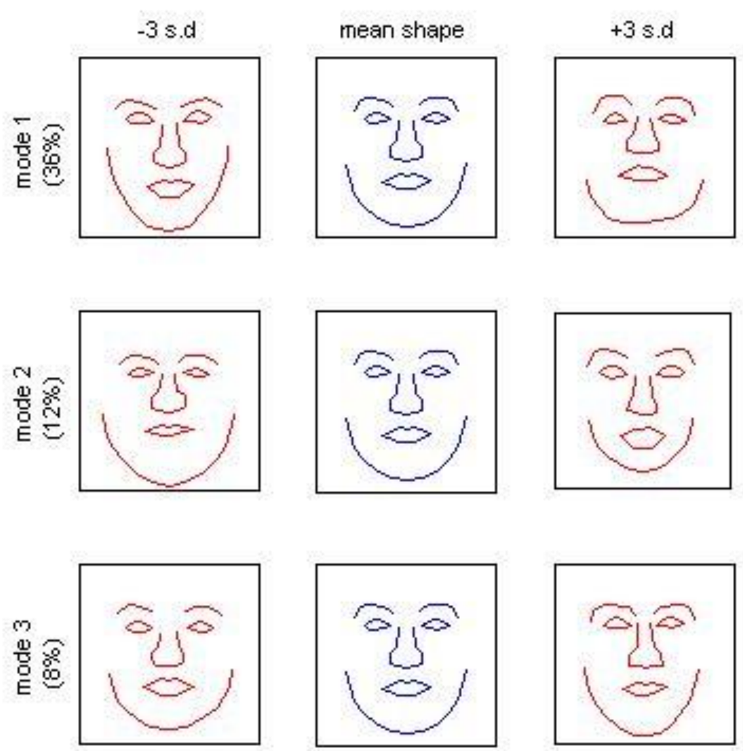


Cardiomegaly diagnosis

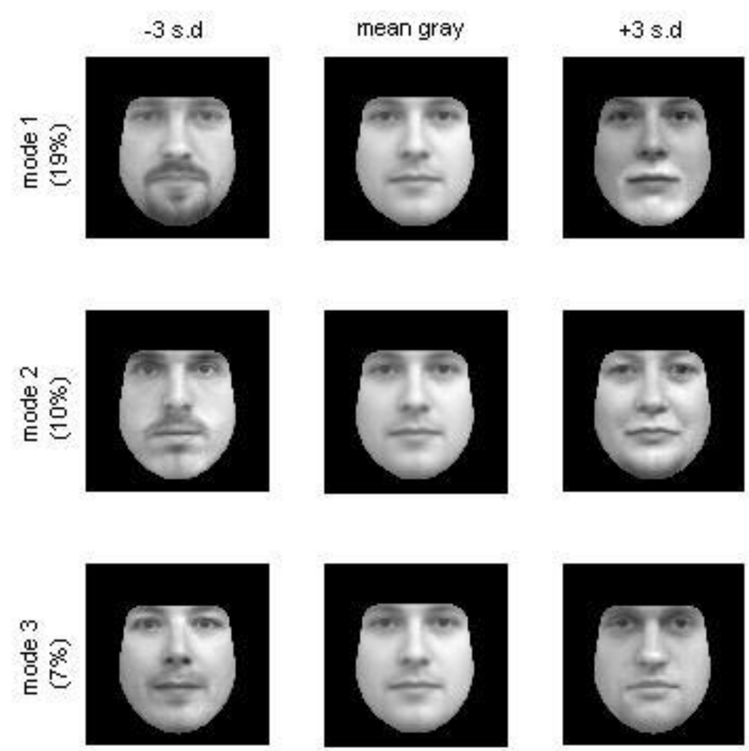
Teasers: active appearance models

PCA is not limited to landmarks, we can model both landmark positions and object intensities

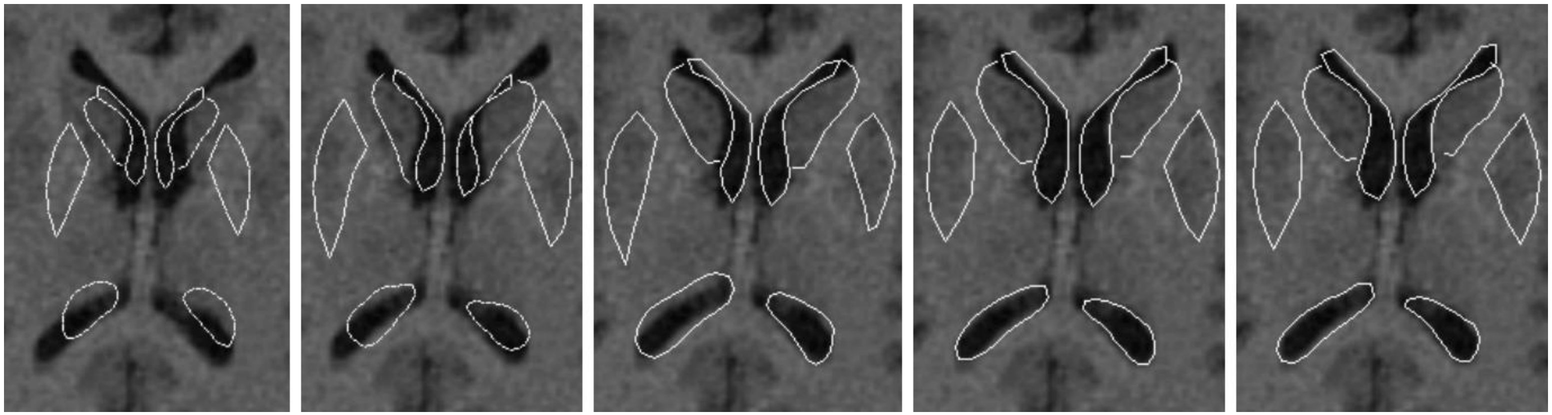
Shape PCA



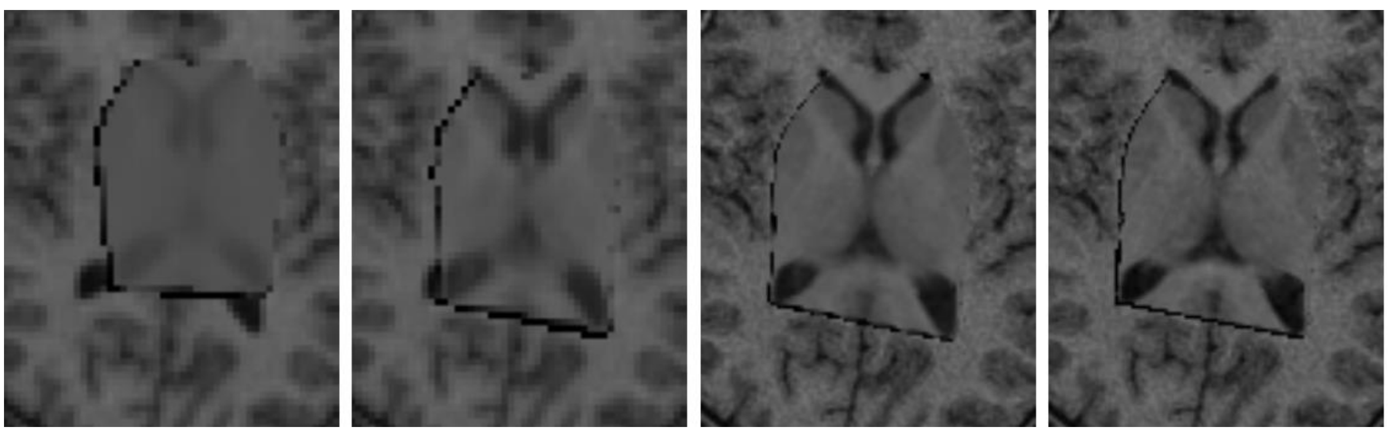
Intensity PCA



Teasers: active appearance models



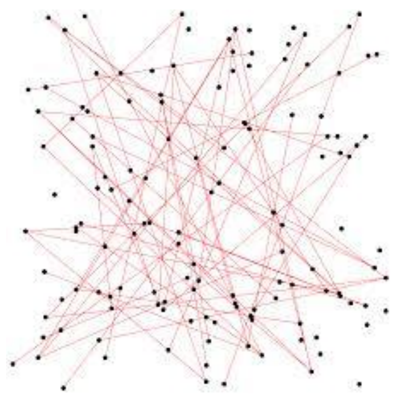
Segmentation with active shape models

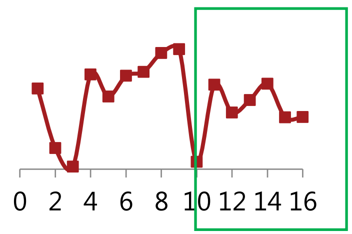


Segmentation with active appearance models

Teasers: active appearance models

As you remember, the PCA-based shape model can find a most similar shape example for every combination of points



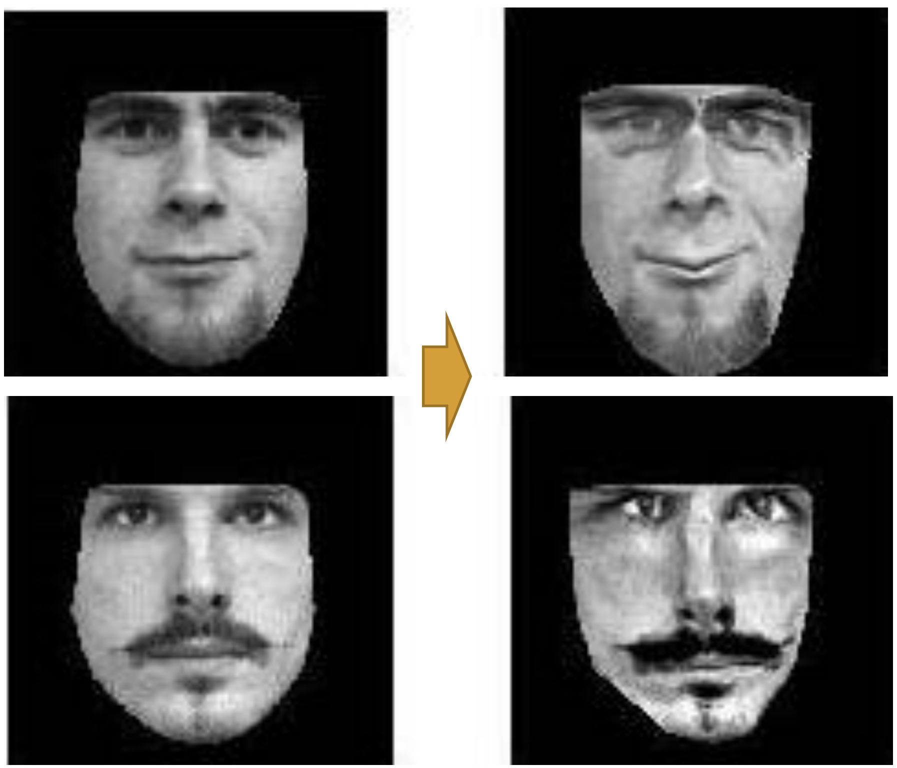


What if we try to fit PCA-based appearance model to an image with previously unseen appearance?

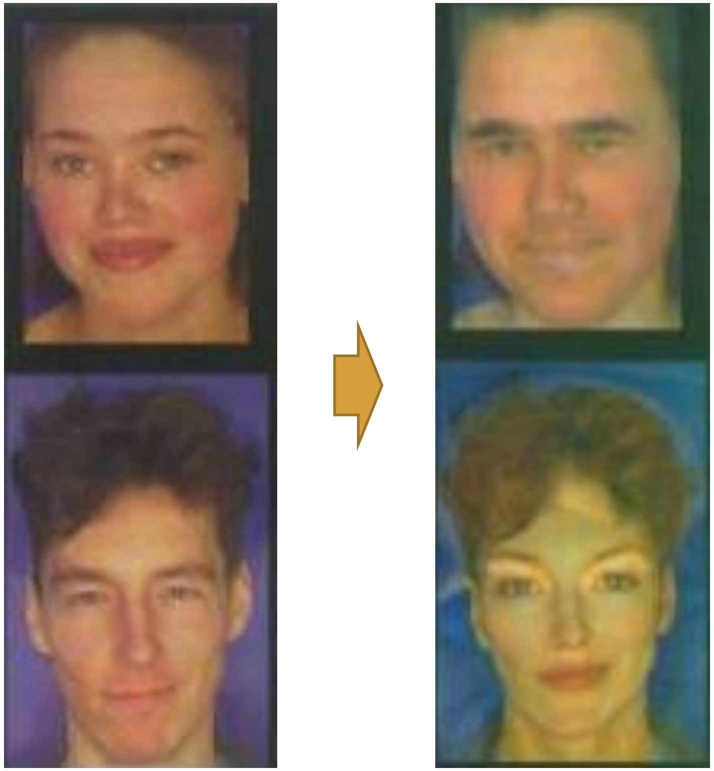
Active appearance models for style transform

Teasers: active appearance models

AAM for caricature picture



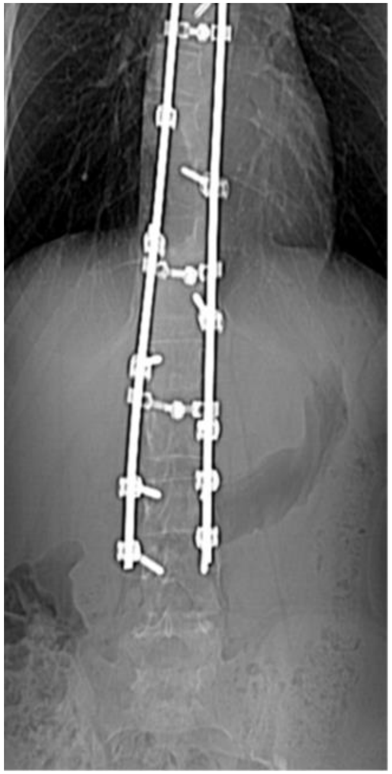
AAM for gender changing



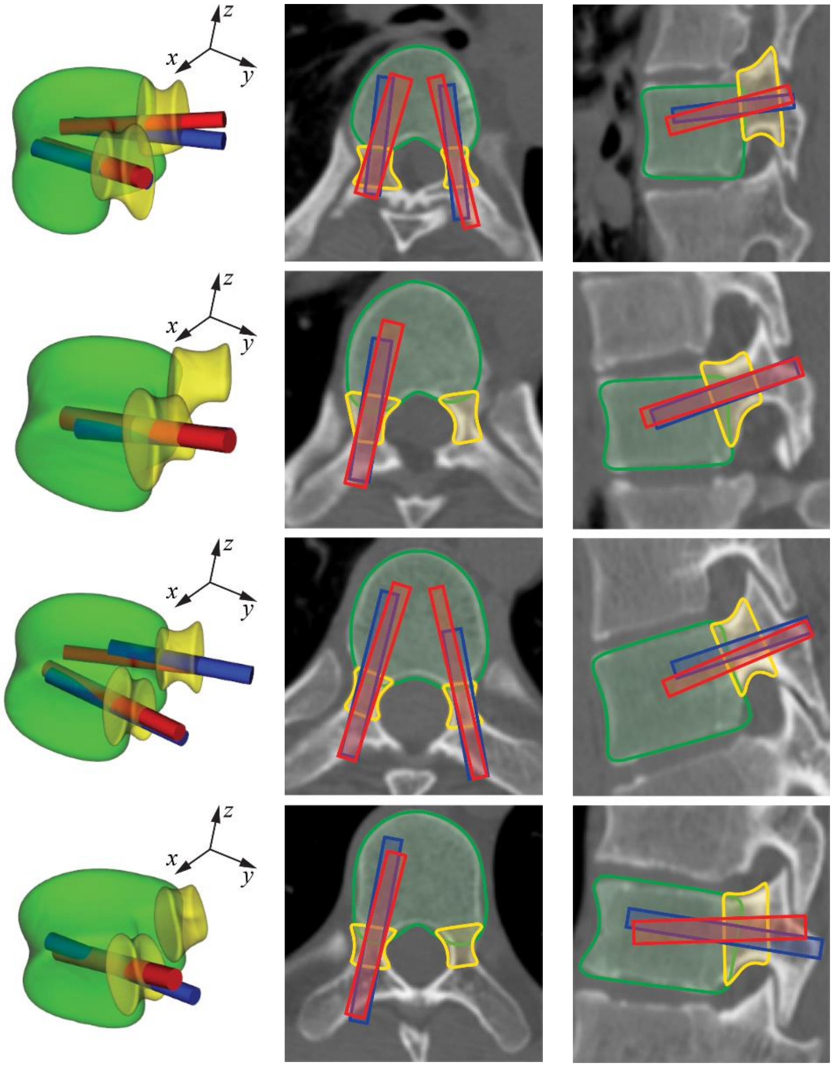
Teasers: parametric models

Pedicle screw size and trajectory planning using superquadrics

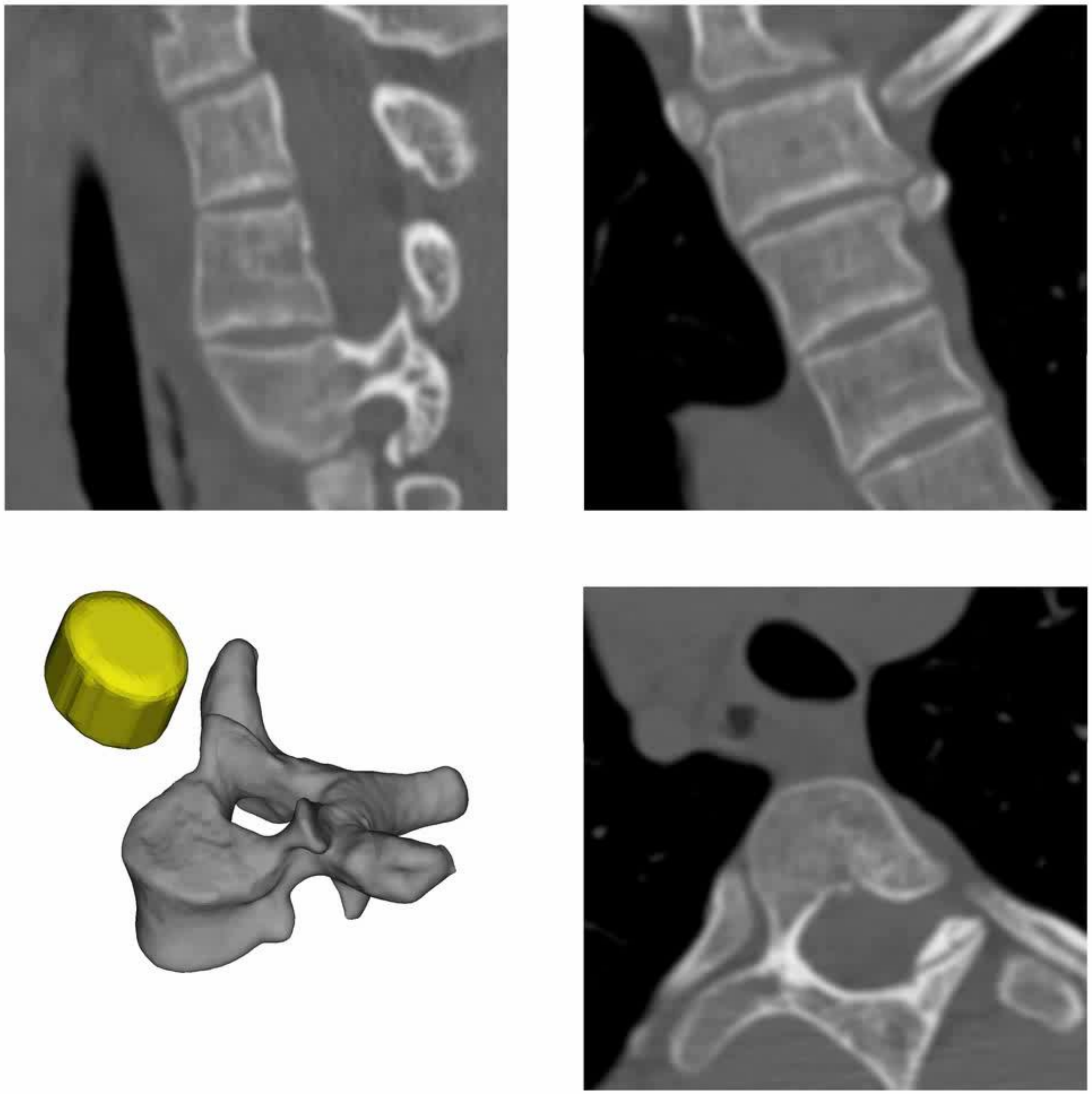
Preoperative

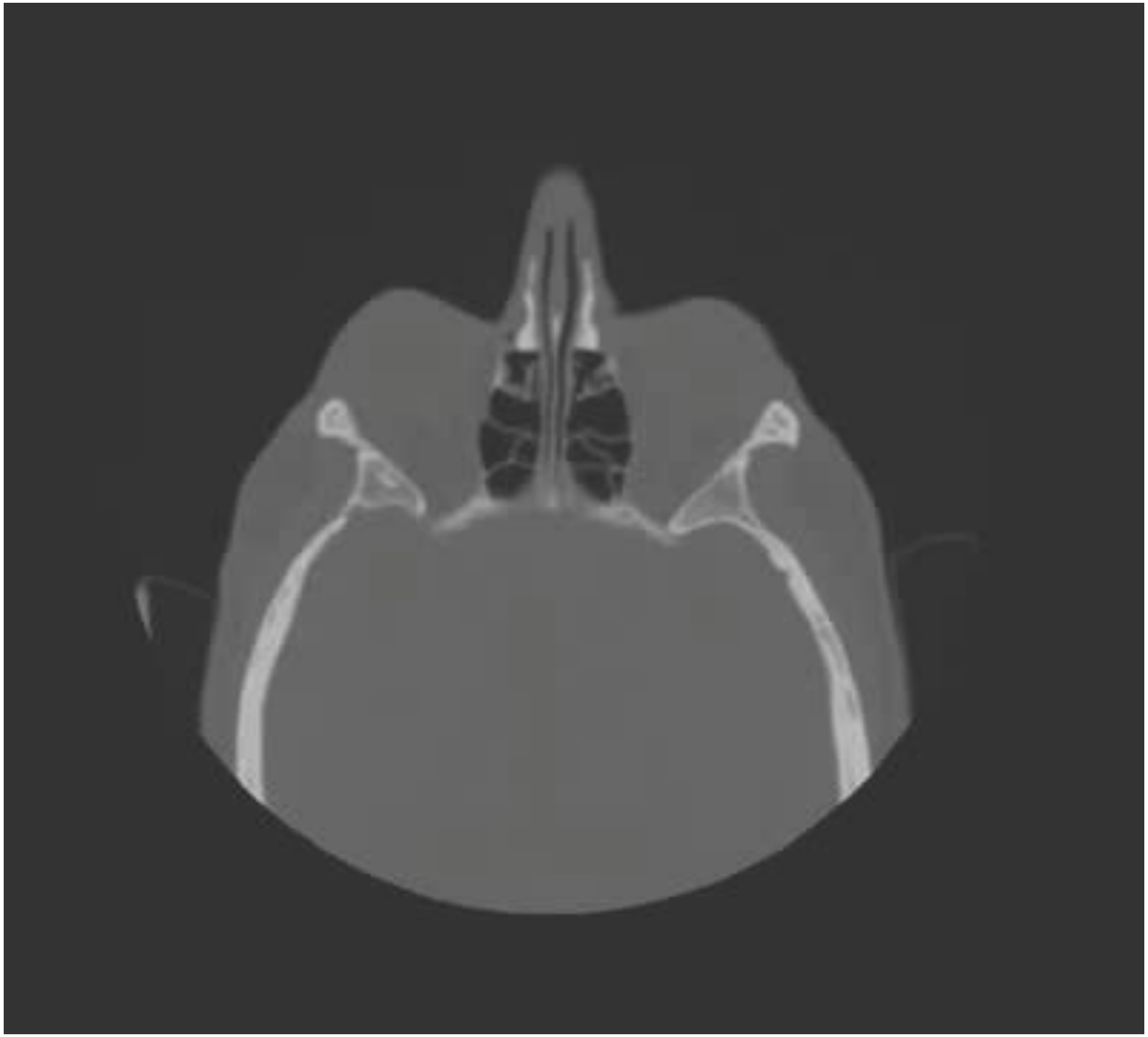


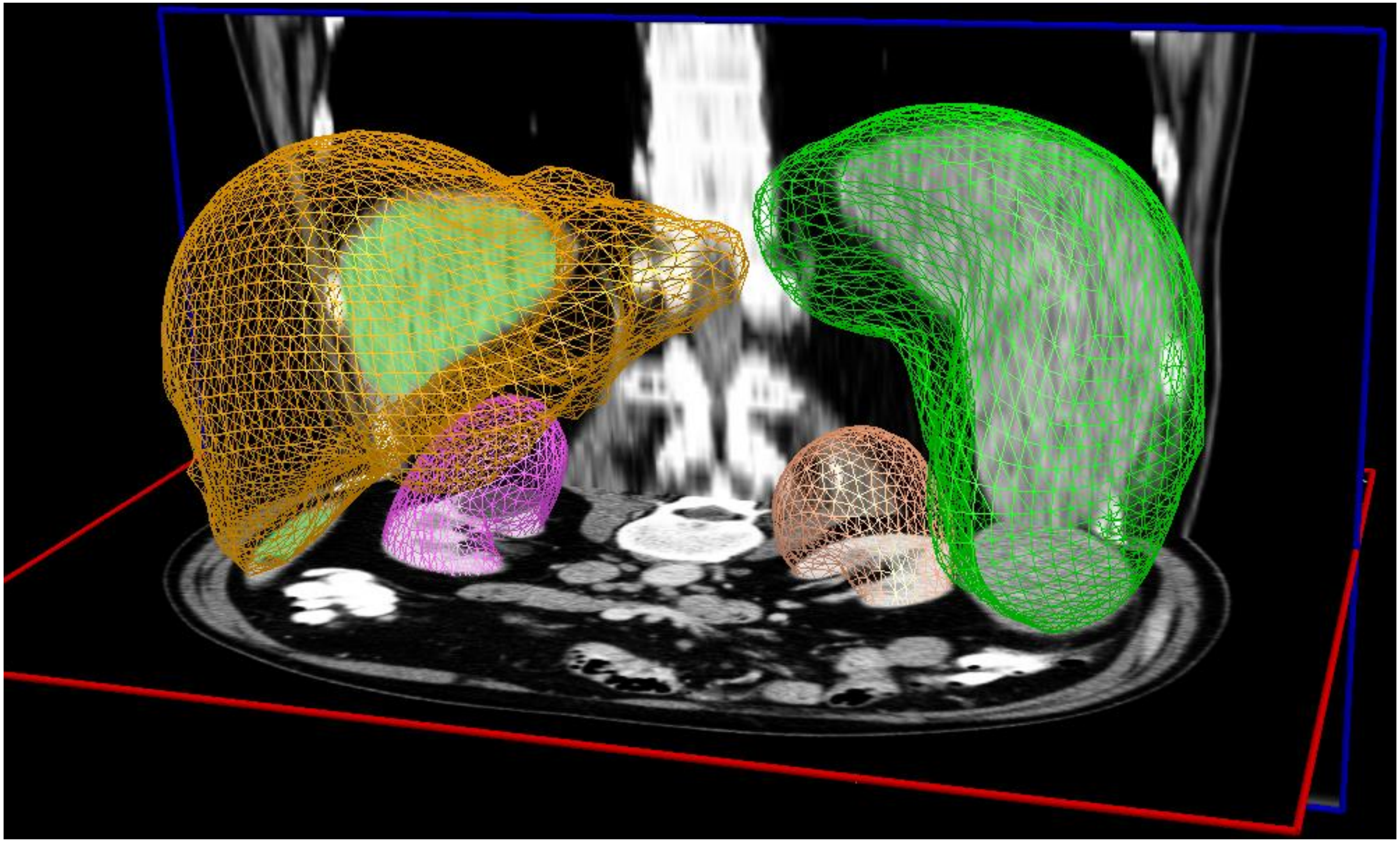
Postoperative

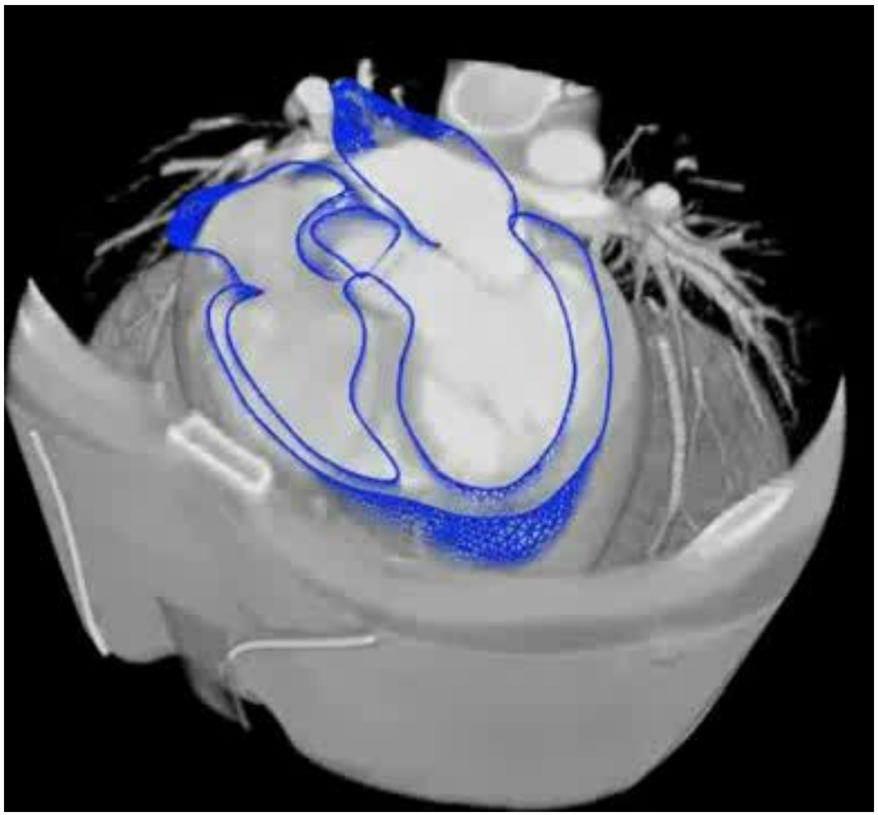


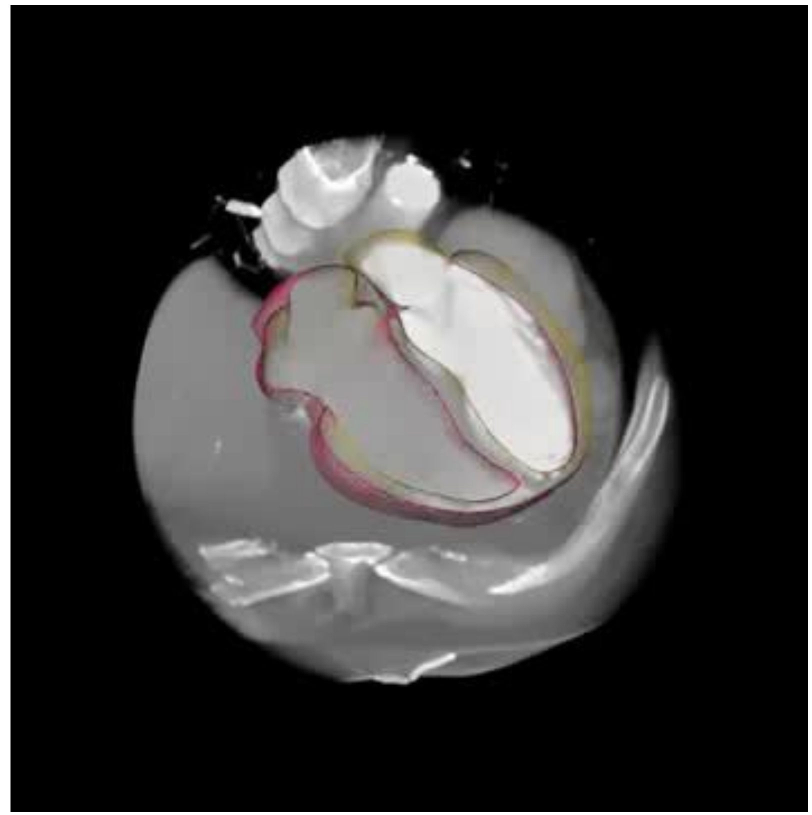
Teasers: parametric models











Questions?