Mesh Alignment Algorithms

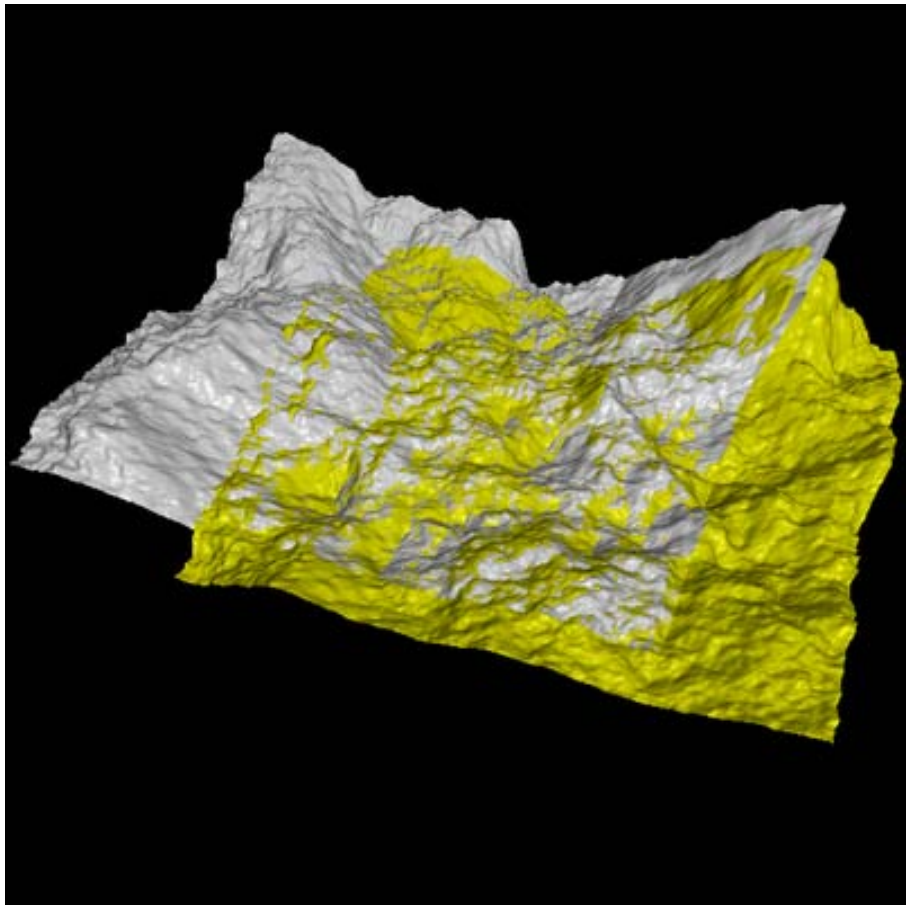
General and the Craniofacial Standard

Outline

- The Problem
- Iterative Closest Point Algorithm
- Deformable Mesh Alignment
- The Procrustes Methodology
- Examples

The Problem

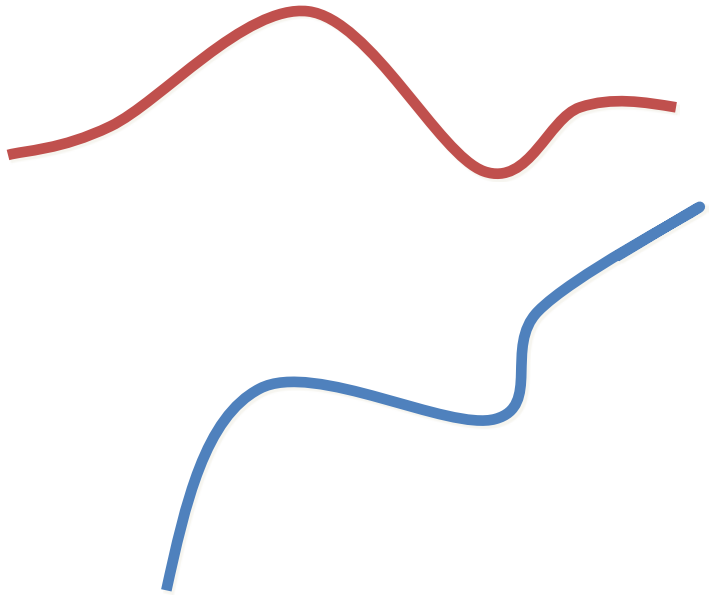
- Align two partially-overlapping meshes given initial guess for relative transform



Motivation

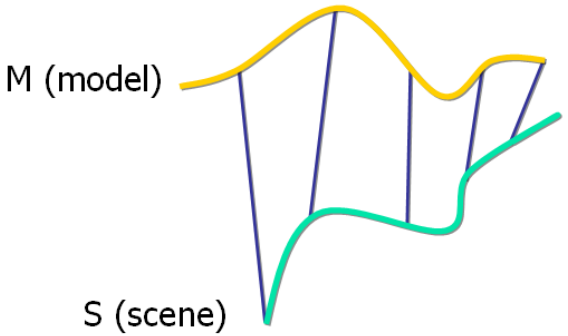
- Shape inspection
- Motion estimation
- Appearance analysis (what we are doing)
- Texture mapping
- Tracking

Aligning 3D Data:
Iterative Closest Point Algorithm



Corresponding Point Set Alignment

- Let M be a model point set.
- Let S be a scene point set.

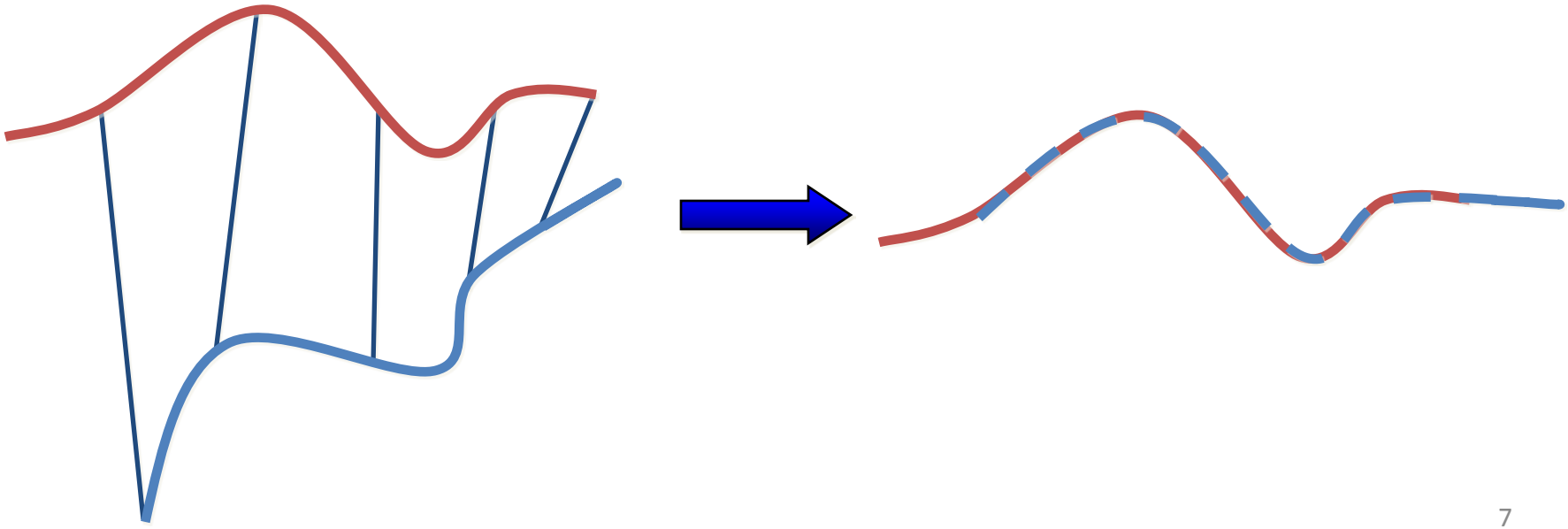


We assume :

1. $N_M = N_S$.
2. Each point S_i correspond to M_i .

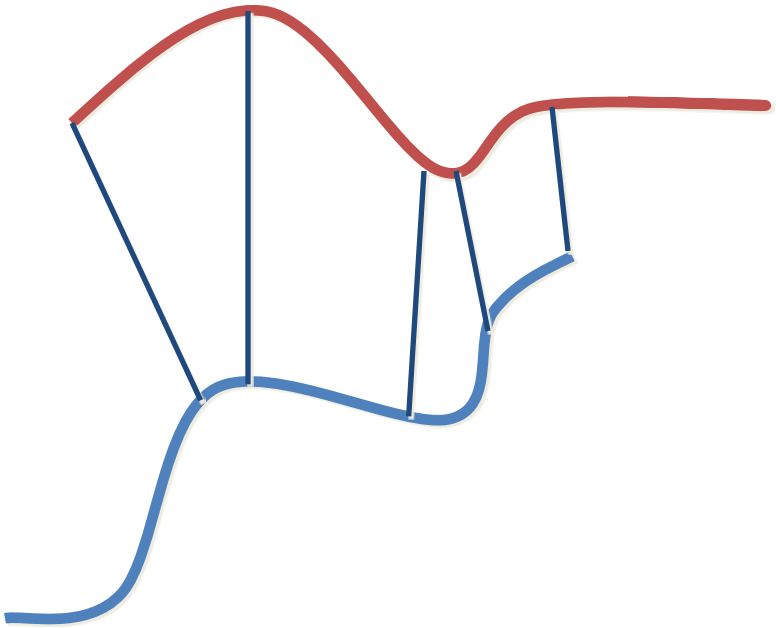
Aligning 3D Data

- If correct correspondences are known, can find correct relative rotation/translation



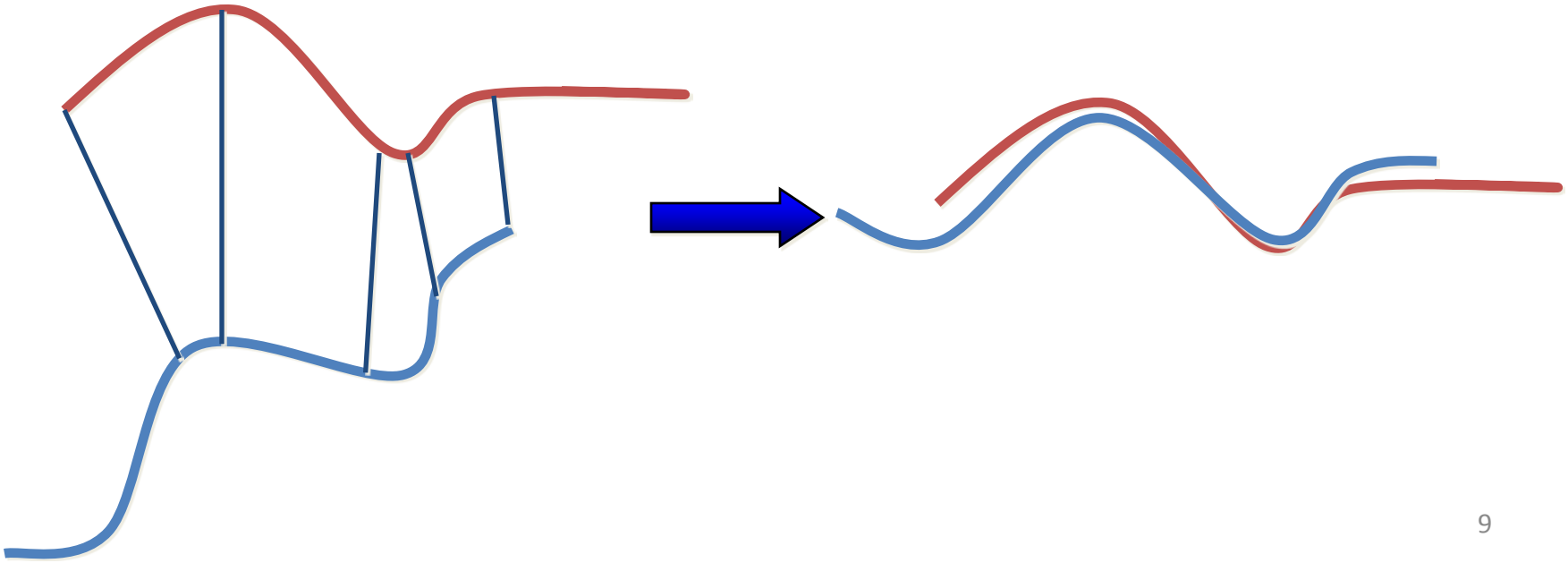
Aligning 3D Data

- How to find correspondences: User input? Feature detection? Signatures?
- Alternative: assume closest points correspond



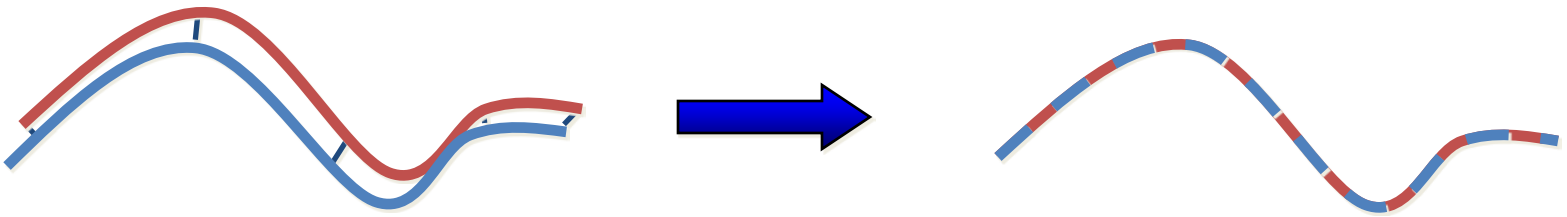
Aligning 3D Data

- How to find correspondences: User input? Feature detection? Signatures?
- Alternative: assume closest points correspond



Aligning 3D Data

- Converges if starting position “close enough“



Closest Point

- Given 2 points r_1 and r_2 , the Euclidean distance is:

$$d(r_1, r_2) = \|r_1 - r_2\| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

- Given a point r_1 and set of points A , the Euclidean distance is:

$$d(r_1, A) = \min_{i \in 1..n} d(r_1, a_i)$$

Finding Matches

- The scene shape S is aligned to be in the best alignment with the model shape M.
- The distance of each point s of the scene from the model is :

$$d(s, M) = \min_{m \in M} d\left\| m - s \right\|$$

Finding Matches

$$d(s,M) = \min_{m \in M} d\left\|m - s\right\| = d(s,y)$$

$$y \in M$$

$$Y = C(S,M)$$

$$Y \subseteq M$$

C – the closest point operator

Y – the set of closest points to S

Finding Matches

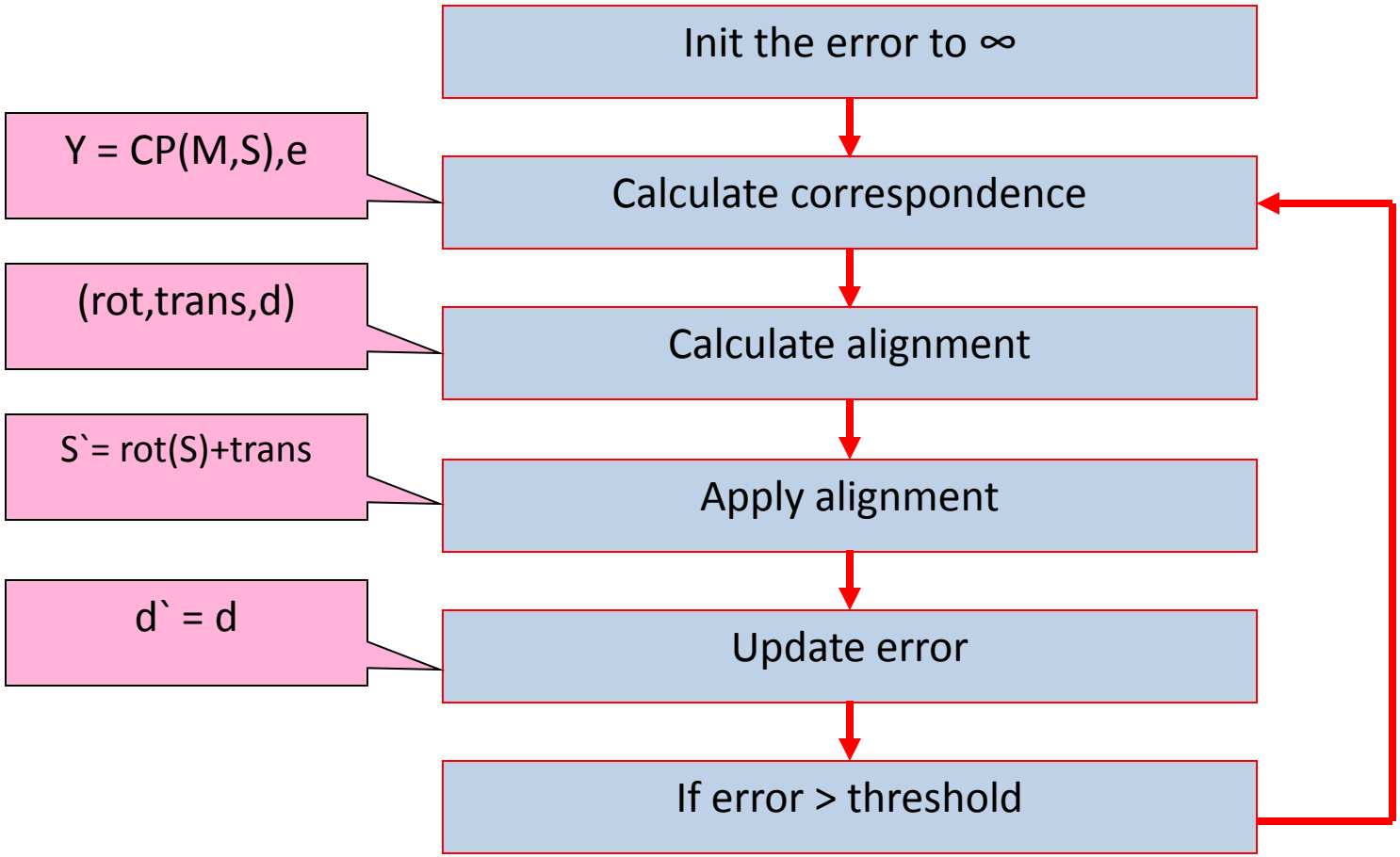
- Finding each match is performed in $O(N_M)$ worst case.
- Given Y we can calculate alignment

$$(rot, trans, d) = \Phi(S, Y)$$

- S is updated to be :

$$S_{new} = rot(S) + trans$$

The Algorithm



The Algorithm

```
function ICP(Scene,Model)
begin
E` ← + ∞;
(Rot,Trans) ← In Initialize-Alignment(Scene,Model);
repeat
    E ← E`;
    Aligned-Scene ← Apply-Alignment(Scene,Rot,Trans);
    Pairs ← Return-Closest-Pairs(Aligned-Scene,Model);
    (Rot,Trans,E`) ← Update-Alignment(Scene,Model,Pairs,Rot,Trans);
Until |E`- E|  < Threshold
return (Rot,Trans);
end
```

Convergence Theorem

- The ICP algorithm always converges monotonically to a local minimum with respect to the MSE distance objective function.

Convergence Theorem

- Correspondence error :

$$e_k = \frac{1}{N_S} \sum_{i=1}^{N_S} \left\| y_{ik} - s_{ik} \right\|^2$$

- Alignment error:

$$d_k = \frac{1}{N_S} \sum_{i=1}^{N_S} \left\| y_{ik} - Rot_k(s_{io}) - Trans_k \right\|^2$$

Time analysis

Each iteration includes 3 main steps

A. Finding the closest points :

   $O(N_M)$ per each point

   $O(N_M * N_S)$ total.

B. Calculating the alignment: $O(N_S)$

C. Updating the scene: $O(N_S)$

Optimizing the Algorithm

The best match/nearest neighbor problem :

Given **N** records each described by **K** real values (attributes) , and a dissimilarity measure **D** , find the **m** records closest to a query record.

Optimizing the Algorithm

- K-D Tree :

  Construction time: O(knlogn)

  Space: O(n)

  Region Query : $O(n^{1-1/k}+k)$

ICP Variants

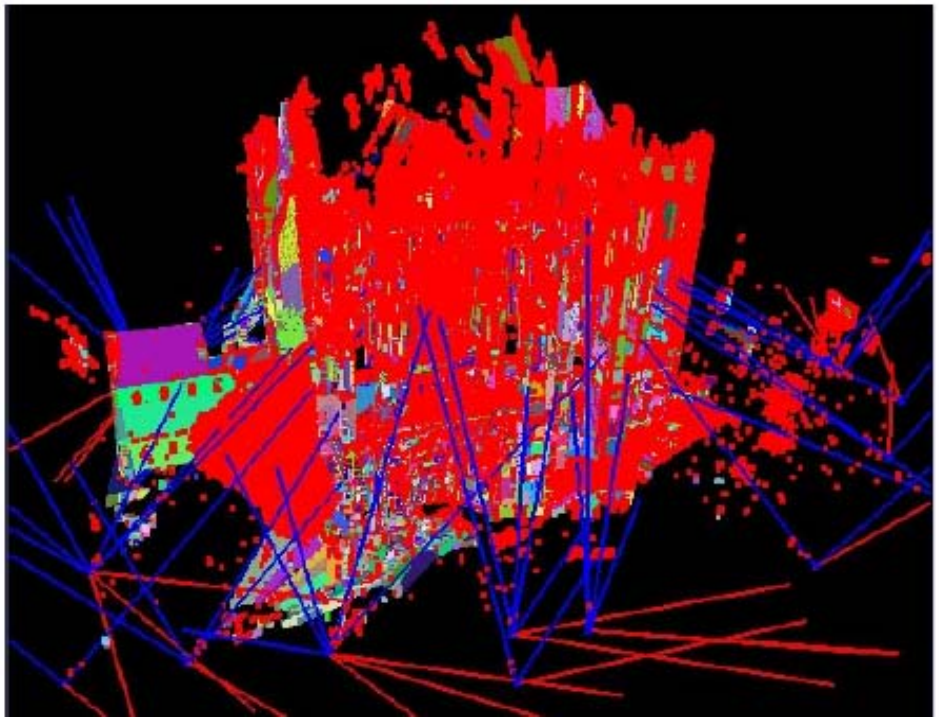
- Variants on the following stages of ICP have been proposed:

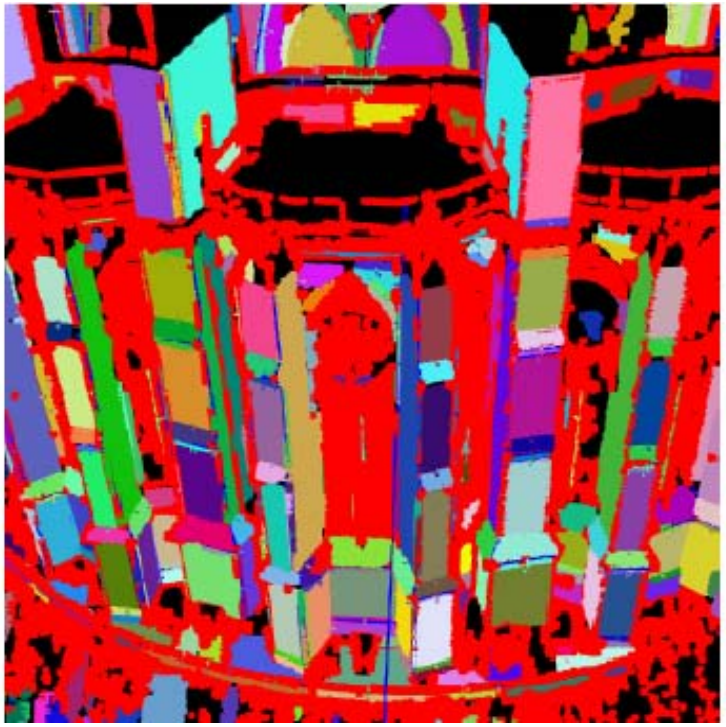
1. Selecting sample points (from one or both meshes)
2. Matching to points in the other mesh
3. Weighting the correspondences
4. Rejecting certain (outlier) point pairs
5. Assigning an error metric to the current transform
6. Minimizing the error metric w.r.t. transformation

Robust Simultaneous Alignment of Multiple Range Images

Figure 1: *Cathedral of Ste. Pierre.*

Motivation





Motivation

Why not just use ICP?

It looks for a 3D rigid body transformation involving only rotation and translation.

Not good enough for heads and human bodies.

Three-Dimensional Correspondence

Christian R. Shelton
M.S. Thesis
M.I.T.
1998

(mentioned in Chia-Chi Teng's Work)

Problem Statement

- An object X is a mesh of triangles (X_K, X_V) where X_V is an ordered set of vertices and X_K is the mesh connectivity.

- A correspondence algorithm will take two meshes A and B and produce a displacement set D of vectors that warp object A by adding each member of D to the corresponding vertex of A. The warped object is called $\vec{D}(A)$.

Energy Minimization

- The problem is to find the best D to warp A into correspondence with B.
- The method is to minimize an energy function E(D).
- E(D) will have two terms
  - The similarity term will measure the similarity of $D(\overrightarrow{A})$ to B.
  - The structure term will measure the changes in in structure to A.

Similarity Term

distance from point p to manifold X

$$\mathrm{d}_X(p) \triangleq \min_{x \in X} \|p - x\|^2 \tag{2.1}$$

sum over sampled points of D(A) of distance^2 to closest point of B

other way (B to D(A))

$$\sum_{s \in S_n(\vec{D}(A))} \mathrm{d}_B(s) + \sum_{s \in S_n(B)} \mathrm{d}_{\vec{D}(A)}(s) \tag{2.2}$$

add in the vertices to the sampled points

$$E_{sim}(D) = \sum_{s \in S'_n(\vec{D}(A))} \mathrm{d'}_B(s) + \sum_{s \in S'_n(B)} \mathrm{d'}_{\vec{D}(A)}(s) \tag{2.3}$$

$$\mathrm{d'}_X(y) \triangleq \min_{x \in X} \|y - x\|^2 + \rho(1 - (N_Y(y)^T N_X(x))^2) \tag{2.4}$$

this new part of the distance penalizes matching 2 points whose orientations do not match well

e.g. what if they differ by 90 degrees?

Structure Term

- Uses the model of directional springs.
- Place a directional spring from each vertex in A to all adjacent vertices in A and every other vertex closer than the most distant adjacent one.
- Let C_A be the set of all pairs of vertices of A connected by directional springs
- Let C_A^a be that intersected just with vertex a

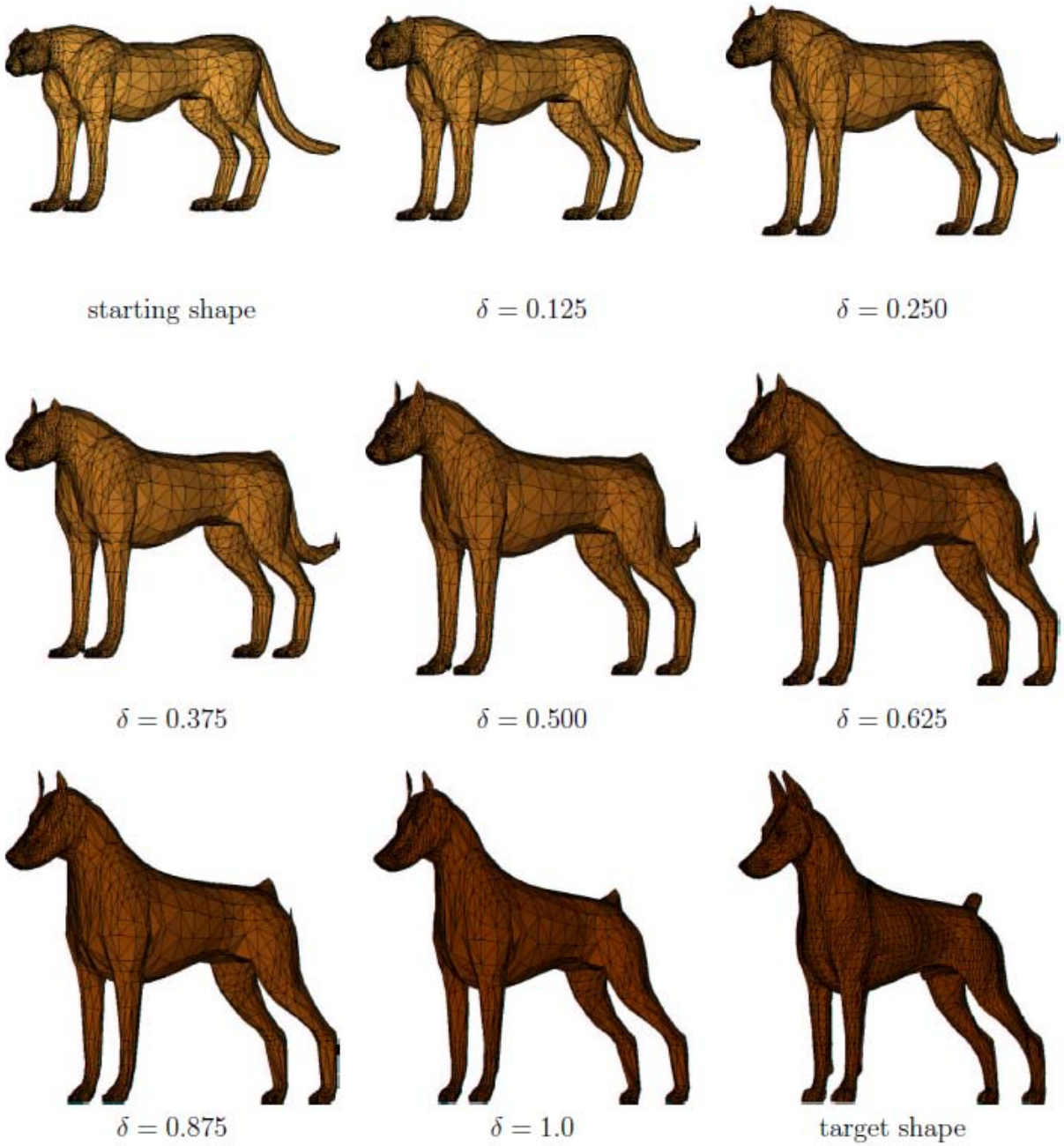
$$E_{str}(D) = k \sum_{(a,b)\in C_A} \frac{\left\| (a-b) - (\vec{D}(a) - \vec{D}(b)) \right\|^2}{\|a-b\| \, |C_A^a|} \qquad (2.8)$$

How much did the spring from a to b stretch?

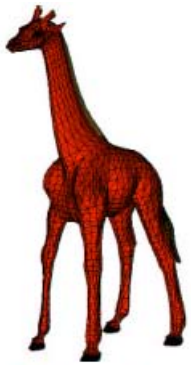
Energy Function

- $E(D) = E_{sim}(D) + \alpha E_{str}(D)$
- Method uses gradient descent to minimize E(D.)
- Algorithm uses a pyramid to speed it up.

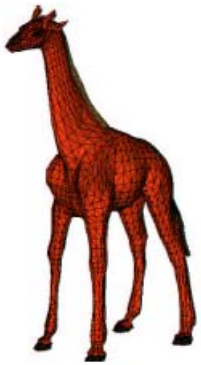
A


starting shape
$\delta = 0.125$
$\delta = 0.250$
$\delta = 0.375$
$\delta = 0.500$
$\delta = 0.625$
$\delta = 0.875$
$\delta = 1.0$
target shape

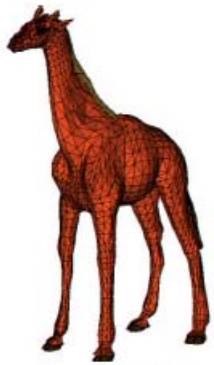
B



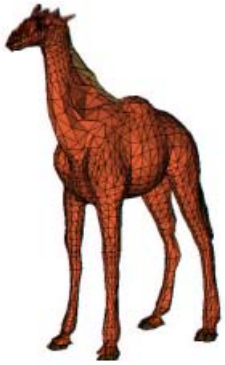
starting shape



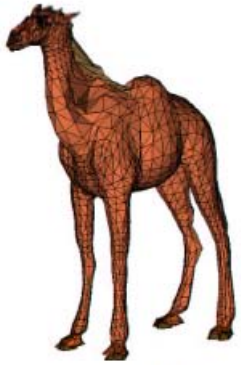
$\delta = 0.125$



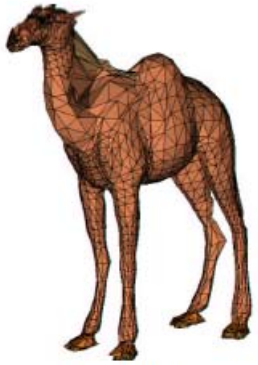
$\delta = 0.250$



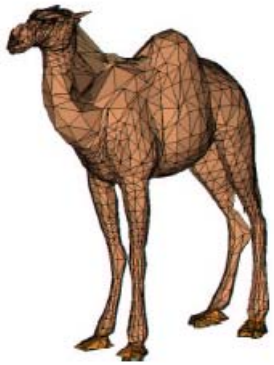
$\delta = 0.375$

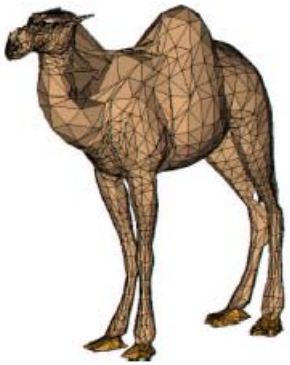


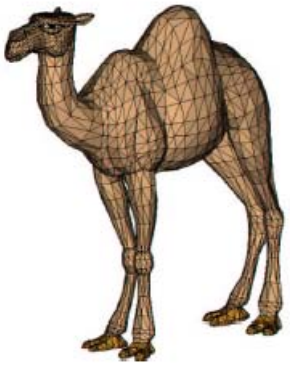
$\delta = 0.500$



$\delta = 0.625$





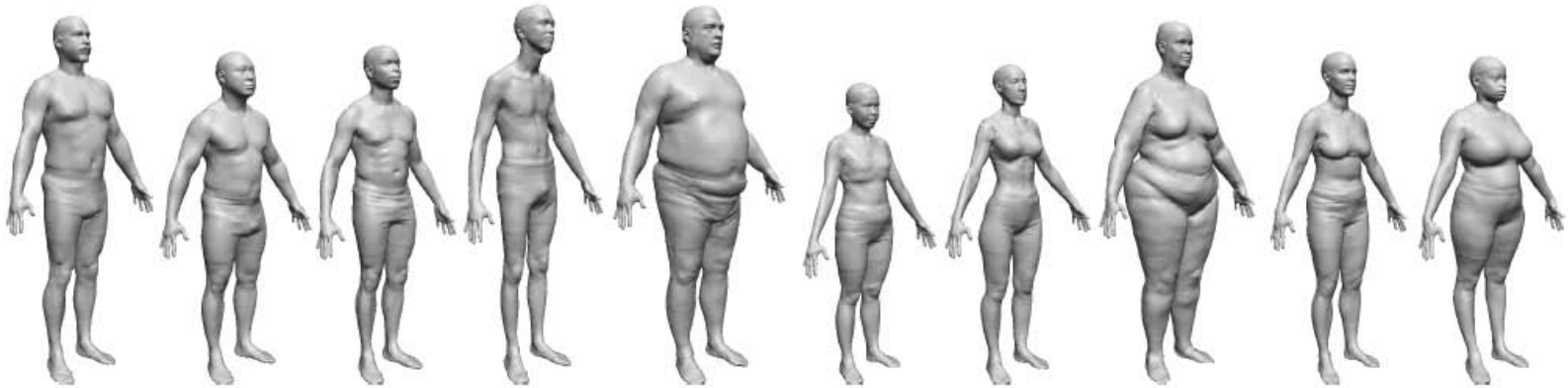


Deformable Mesh Alignment: Brett Allen's 2003 SIGGRAPH Paper

The space of human body shapes: reconstruction and parameterization from range scans

Brett Allen Brian Curless Zoran Popović

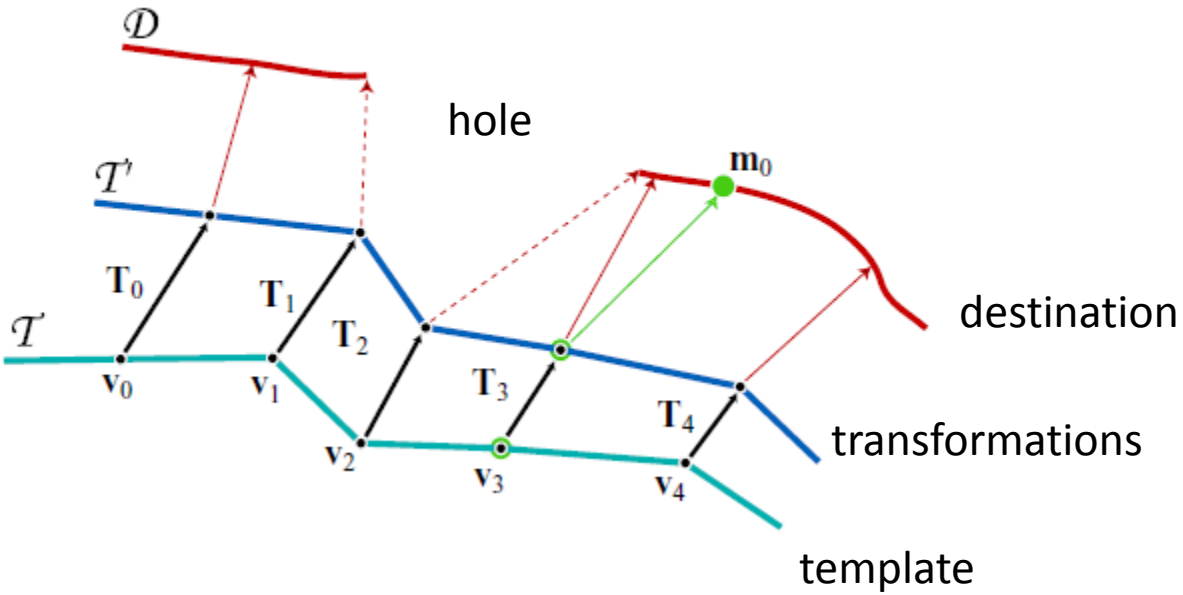
University of Washington



Overview

- Goal: to fit a template surface $\mathcal{T}$ to a scanned example surface $\mathcal{D}$.
- Each surface is represented as a triangular mesh
- The matching is accomplished by an optimization framework
- Each vertex v_i is influenced by a 4 x 4 affine transformation matrix T_i.
- The algorithm must find a set of transformations that move all of the points in $\mathcal{T}$ to a deformed surface $\mathcal{T}'$ such that $\mathcal{T}'$ matches well with $\mathcal{D}$.

Match in Progress



Energy Function: $E = \alpha E_d + \beta E_s + \gamma E_m$

data error

$$E_d = \sum_{i=1}^{n} w_i \, \mathrm{dist}^2(\mathbf{T}_i \mathbf{v}_i, \mathcal{D}), \qquad (1)$$

where dist() computes the distances from $T_i v_i$ to the closest compatible point on $\mathcal{D}$, where compatible means the surface normals are no more than 90 degrees apart, and the distance is less than a threshold.

smoothness error

$$E_s = \sum_{\{i,j|\{\mathbf{v}_i,\mathbf{v}_j\} \in edges(\mathcal{T})\}} ||\mathbf{T}_i - \mathbf{T}_j||_F^2 \qquad (2)$$

where $||\ ||_F$ is the Frobenius norm and measures the distance between transformations.

marker error (if markers)

$$E_m = \sum_{i=1}^{m} ||\mathbf{T}_{\kappa_i} \mathbf{v}_{\kappa_i} - \mathbf{m}_i||^2 \qquad (3)$$

Procedure

At low resolution

1. Fit the markers first: α=0, β=1, γ=10
2. Allow the data term to contribute: α=1, β=1, γ=10

At high resolution

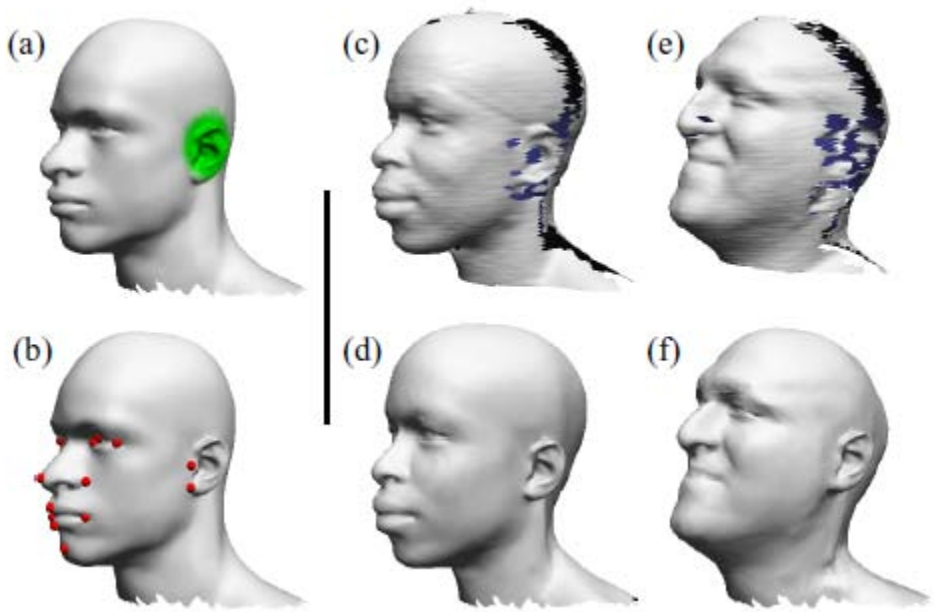
3. Continue the optimization: α=1, β=1, γ=10
4. Allow the data term to dominate: α=10, β=1, γ=1

There is also a hole-filling procedure.

Example

two data heads

template with
zero data-fitting
term in green



with markers

template mapped to data heads
with holes filled and ear fixed

Video

SIGGRAPH 2003 Video

The Procrustes Approach

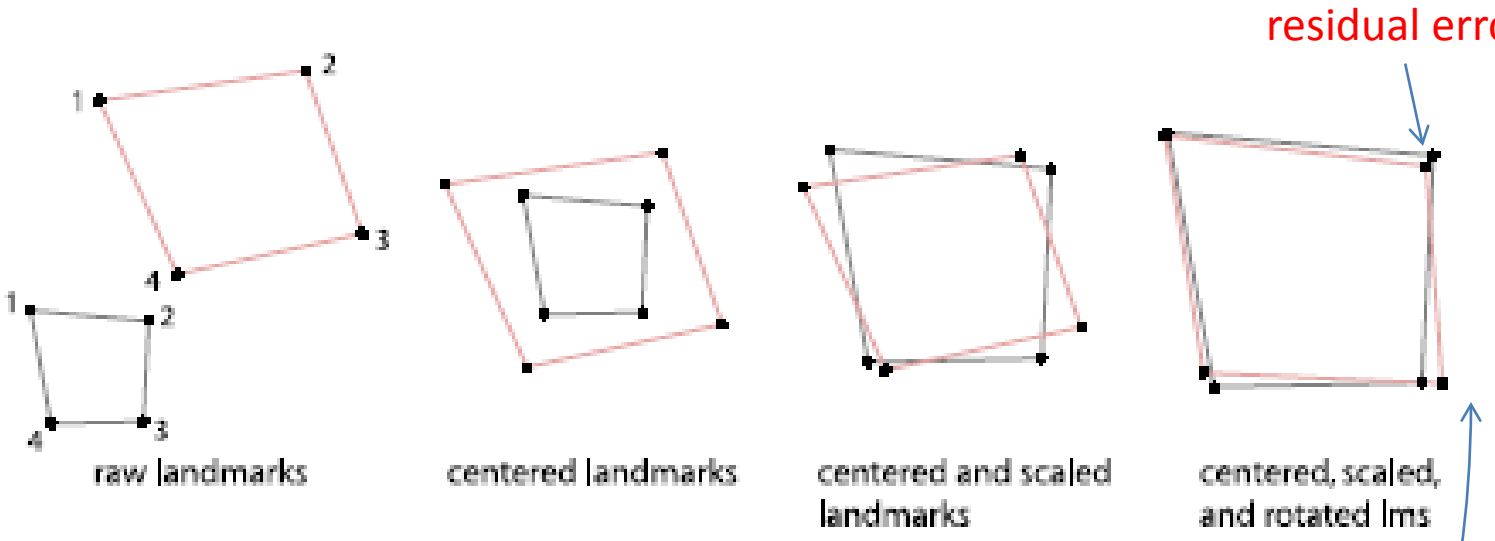
- In Greek mythology **Procrustes** or "the stretcher [who hammers out the metal]" was a rogue smith and bandit from Attica who physically attacked people by stretching them or cutting off their legs, so as to force them to fit the size of an iron bed.

- In general, when something is Procrustean, different lengths or sizes or properties are fitted to an arbitrary standard.

Procrustes Superimposition in Geometric Morphometrics

- Given a group of similar objects (ie. heads, feet, livers) each represented by a set of landmark points.
  1. Translate the landmark configuration of each object so that they have the same centroid (0,0,0)
  2. Scale each landmark configuration so they have the same centroid size (square root of summed squared distances of the landmarks to the centroid).
  3. When superimposing two objects, one of the two centered, scaled configurations is rotated around its centroid until the sum of squared Euclidean distances between the corresponding landmarks is minimized.

Example with 2 Shapes



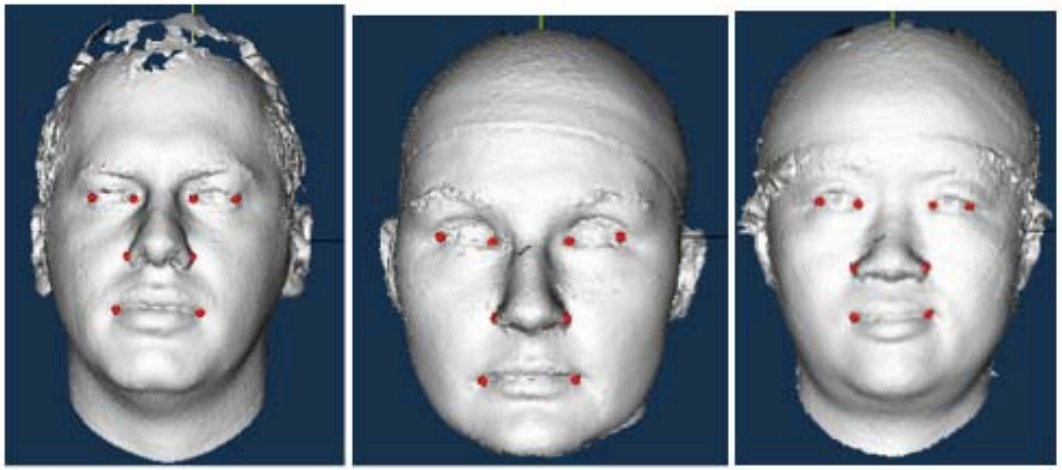
- Note that the scaling process has removed the size difference between the 2 shapes, which is some applications may be important.
- The resultant coordinates are called the Procrustes shape coordinates. The shapes are aligned as they best can be under these transformations.
- This is like least squares, but not identical.

Extension to N Shapes: Generalized Procrustes Analysis

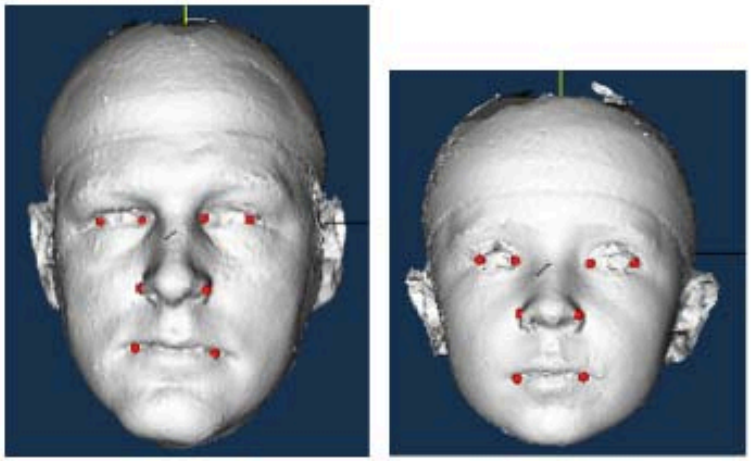
- Perform the translation and scaling steps.
  convergence is set to *false*.
- Choose one sample object arbitrarily from the set
  and call it the consensus.
- while *not* convergence
  - rotate each object to best fit the consensus
  - if total Procrustes distance* from objects to consensus
    is small enough, convergence becomes *true*
  - set consensus to the average of the newly
    rotated objects

* Procrustes distance between 2 shapes is the Euclidian distance between the two sets of Procrustes shape coordinates.

Demo – using our subset



Seattle.stl Kasia.stl Dingding.stl



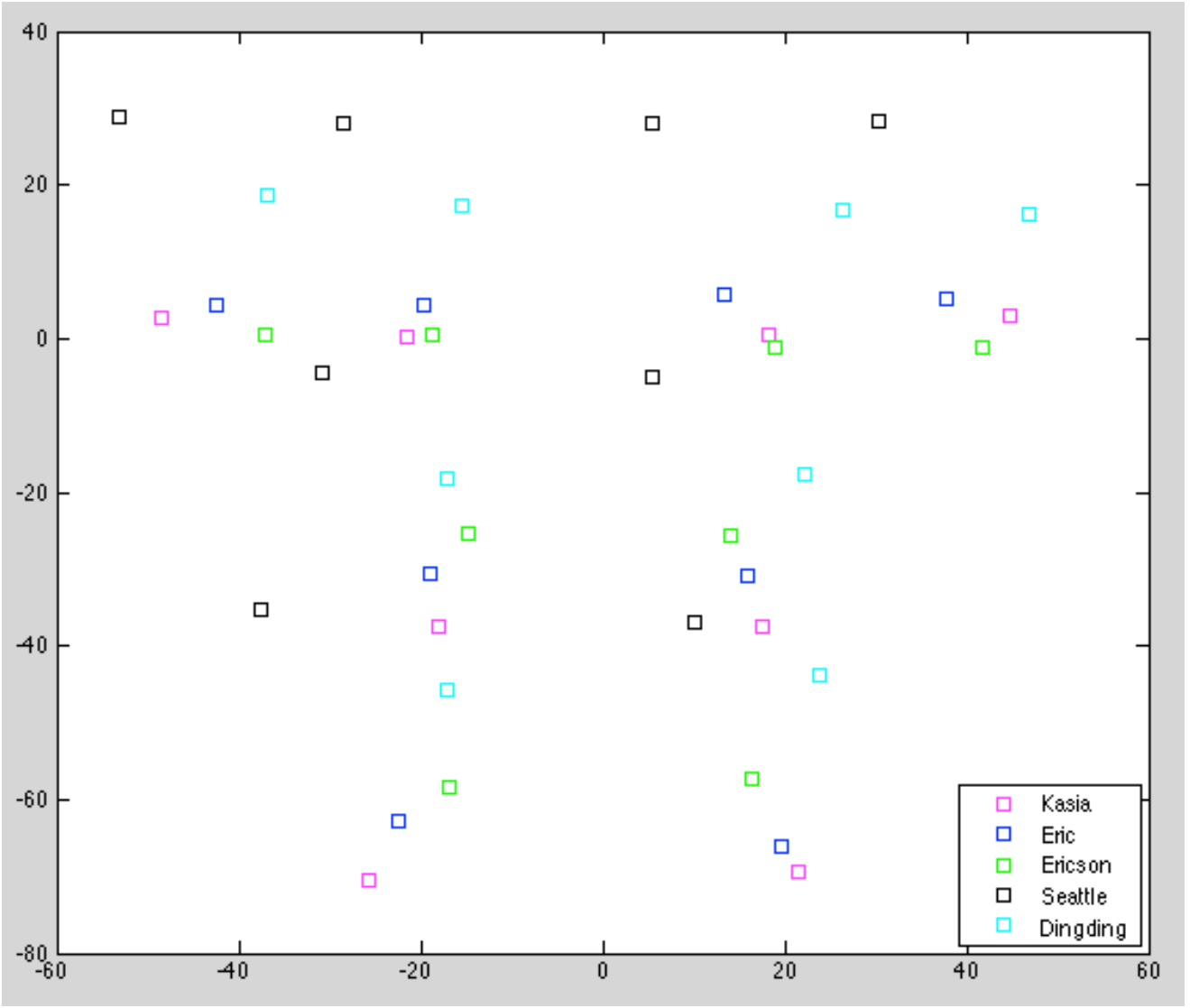
Eric.stl Ericson.stl

8 landmarks each

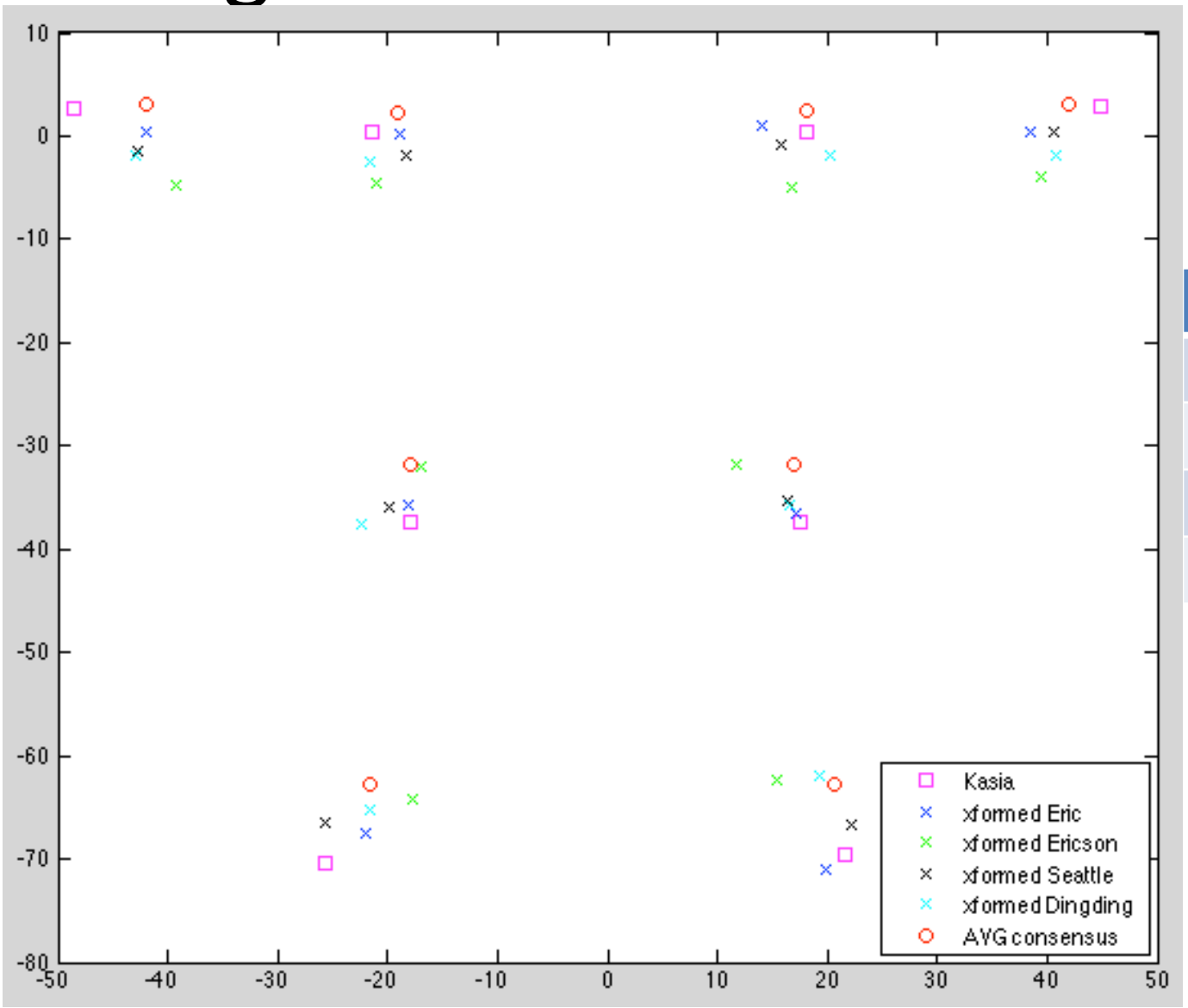
Eyes

Nose

mouth



Avg consensus Procrustes shape

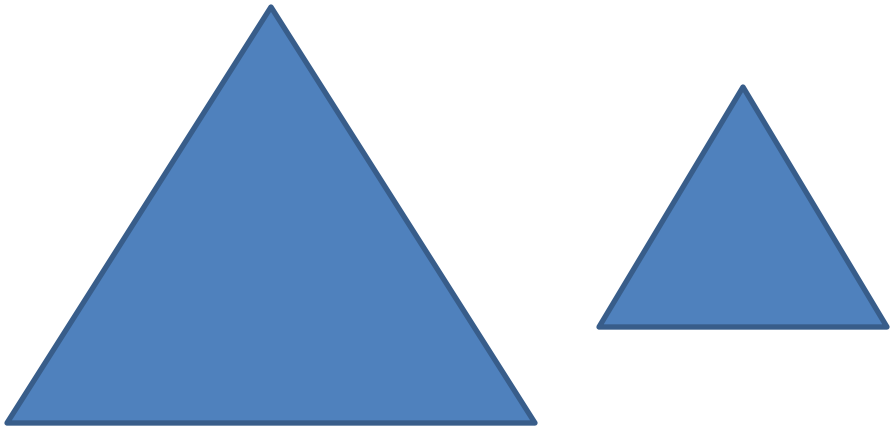


	d (to Kasia)
Eric	0.0119
Ericson	0.0414
Seattle	0.0122
Dingding	0.0307

Shape vs. Form

- Mitteroecker makes a strong point of differentiating between shape and form.
- Shape refers to the geometric properties of an object that are invariant to scale, rotation. and translation.
- Form refers to the geometric properties of an object that are invariant to rotation and translation, but not scale.

same shape



not the same form

Can we adapt Procrustes to form?

- Yes, there are several possibilities.
- Mitteroecker says to augment the Procrustes shape coordinates by the natural logarithm of centroid size to construct a Procrustes form space.
- Hammond and Hutton use Procrustes as the first step in a different approach they call a dense surface model.

Dense Surface Approach

- Start with a set of already landmarked head meshes.
- Apply the generalized Procrustes algorithm to obtain a mean landmark set.
- Warp each full surface mesh to the mean using the landmarks and Bookstein's thin-plate spline technique. NOW THEY ARE ALL ALIGNED IN ONE SPACE.

Continued

- Choose one mesh to be the base mesh, and for each head mesh, find a dense correspondence between the vertices of the base mesh and its vertices using closest point.
  NOW THEY ARE ALL IN FULL, DENSE CORRESPONDENCE
- Transfer the mesh connectivity of the base mesh to each of the others, throwing out original meshes.
- Apply the inverse of the warp to send each mesh back to its original space,
  THUS REGAINING THE ORIGINAL FORM.

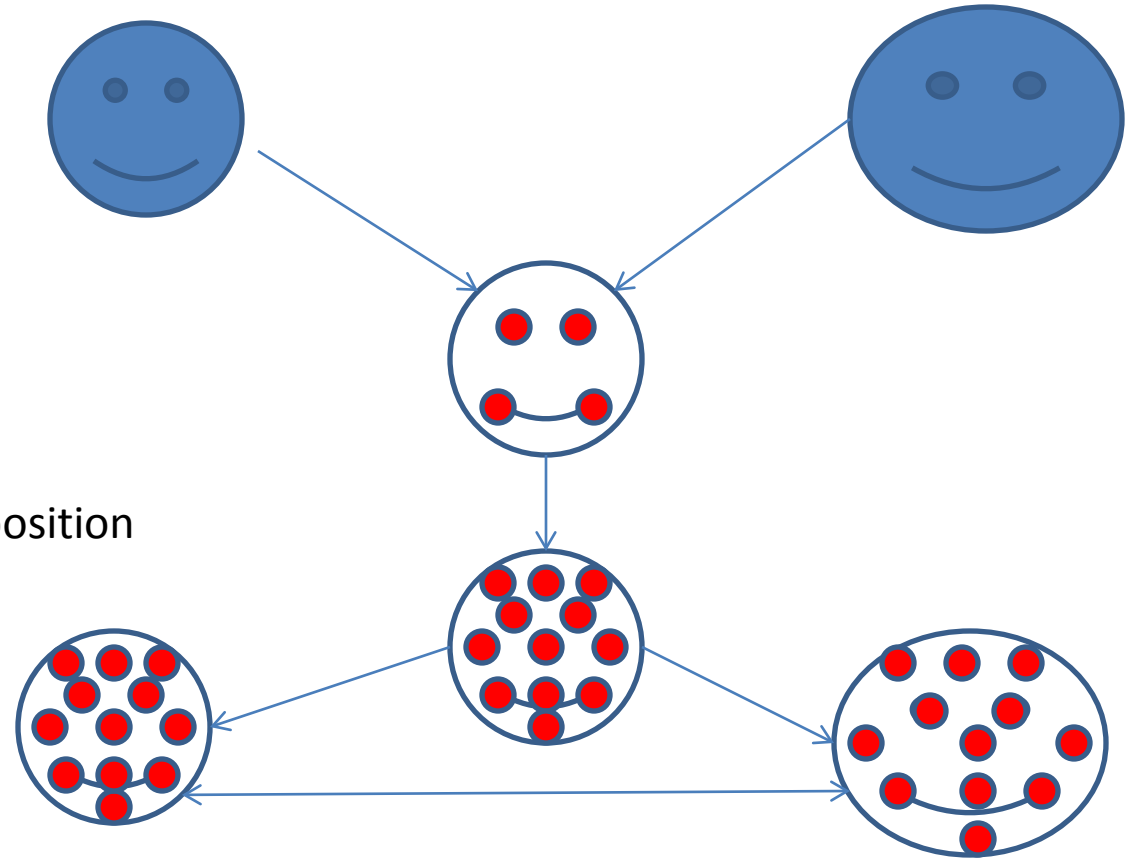
Conceptually

original shapes

Procrustes superimposition

Dense superimposition

Dense correspondence



Procrustes in Classification

Vector of PCA Mode Values, Nearest Mean Classifier

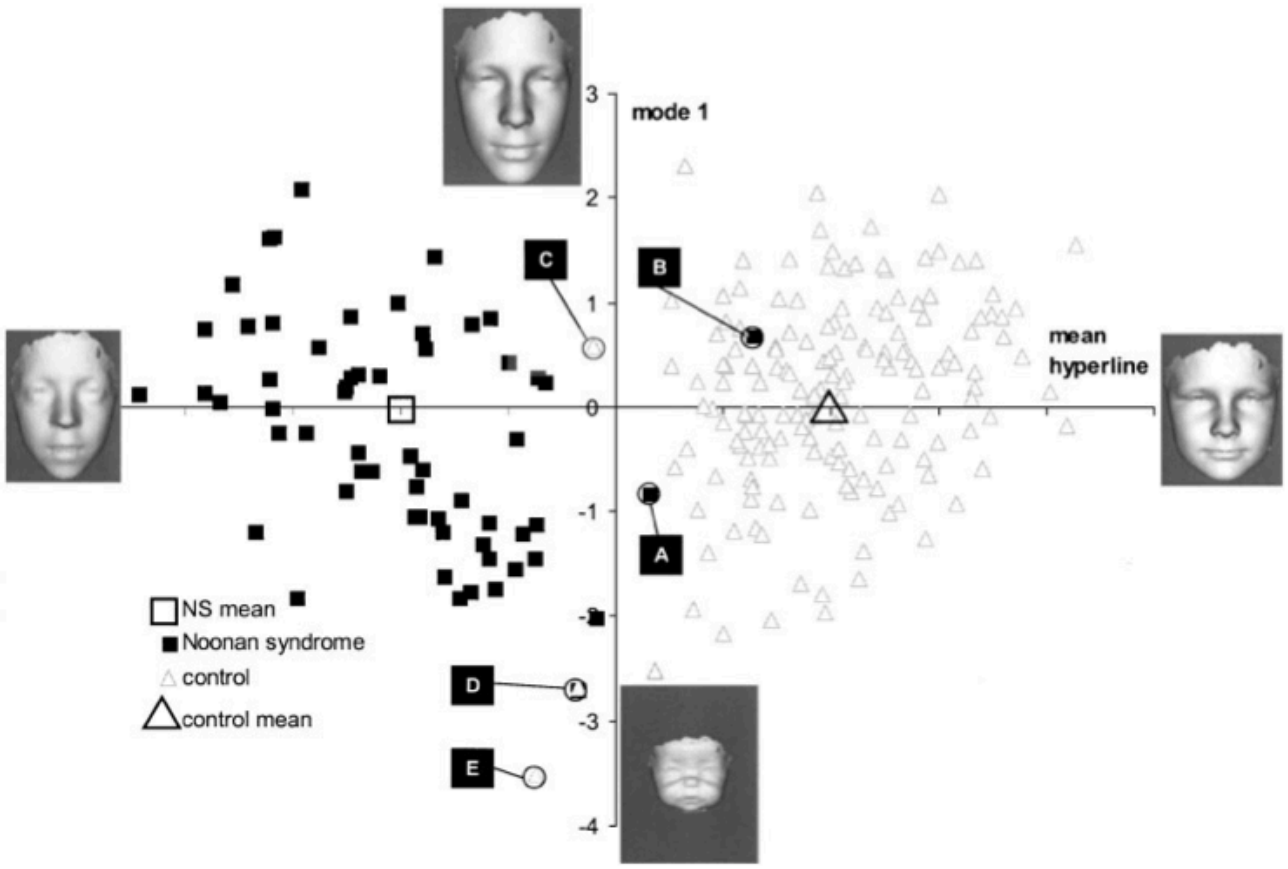


Fig. 7. Scatter plot of mean hyperline against mode 2 for non-adult controls (n = 160) and individuals with NS (n = 60). The labeled faces are misclassified according to the nearest mean discrimination test. Faces **D** and **E** are of very young babies and given the small number of such children in the database, their misclassification is not surprising. Faces **C** is just closer to the NS mean in this model. Faces **A** and **B** are misclassified as controls, but it happens that some of their features are quite uncharacteristic of NS. For ethical reasons, the faces of these individuals are not shown.

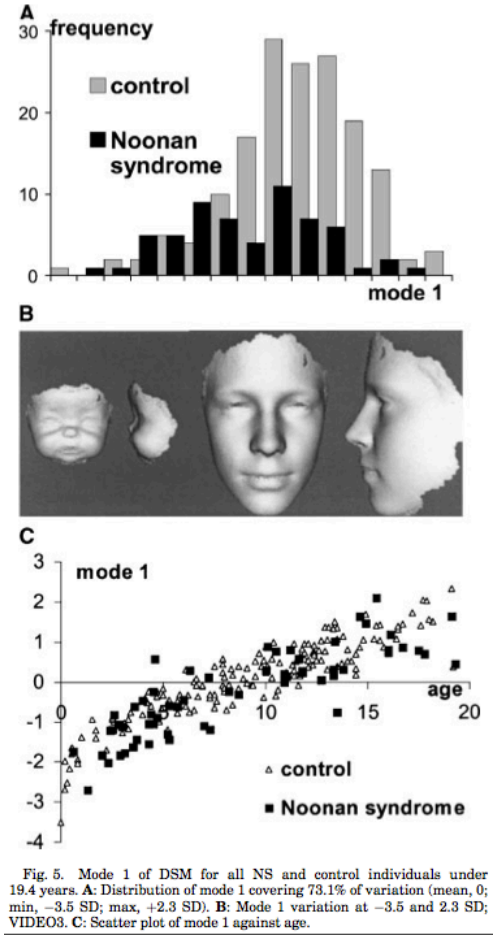


Fig. 5. Mode 1 of DSM for all NS and control individuals under 19.4 years. **A**: Distribution of mode 1 covering 73.1% of variation (mean, 0; min, −3.5 SD; max, +2.3 SD). **B**: Mode 1 variation at −3.5 and 2.3 SD; VIDEO3. **C**: Scatter plot of mode 1 against age.

Peter Hammond et al., "3D Analysis of Facial Morphology", American Joural of Medical Genetics 126A:33-348, 2004