Efficient Shape Matching using Vector Extrapolation

Emanuele Rodolà[1,2], Tatsuya Harada[1],
Yasuo Kuniyoshi[1], and Daniel Cremers[2]

1 THE UNIVERSITY OF TOKYO

2 TUM Technische Universität München

Shape matching

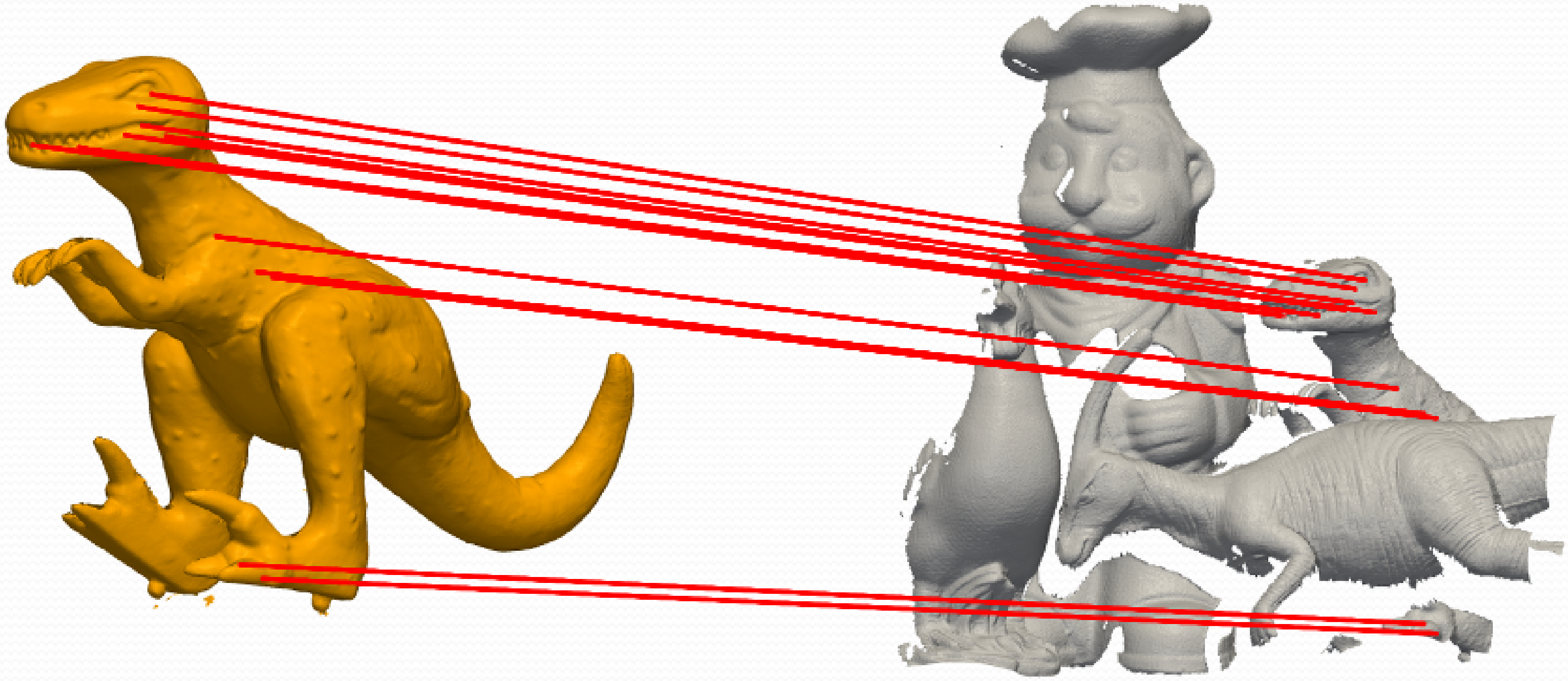
- Shape matching is a pervasive problem in computer vision

Shape matching

- Shape matching is a pervasive problem in computer vision

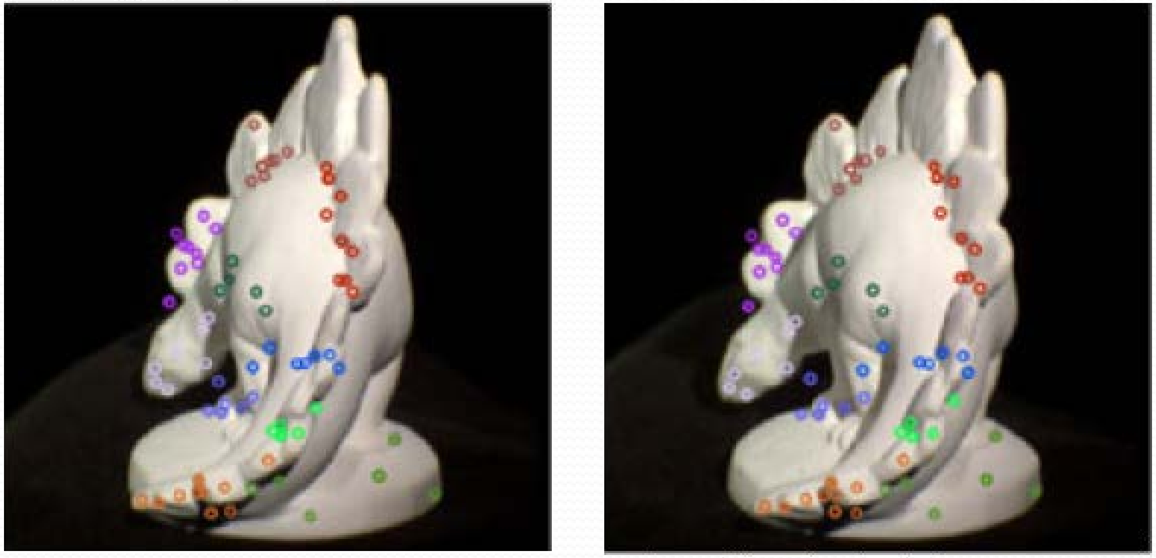
Shape matching

- Shape matching is a pervasive problem in computer vision



Shape matching

- Shape matching is a pervasive problem in computer vision



Shape matching

- Devising specific techniques for reaching a solution can be very challenging

Shape matching

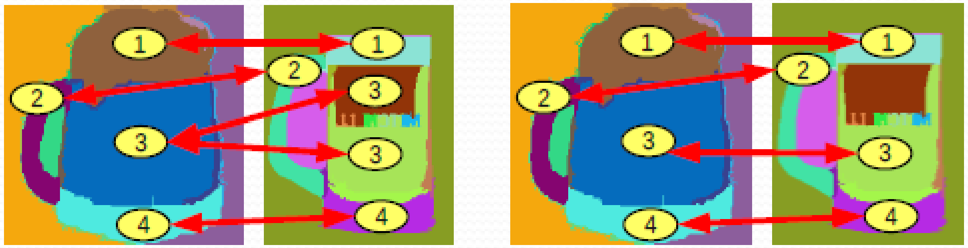
- Devising specific techniques for reaching a solution can be very challenging
- More general optimization algorithms need to be considered (*e.g.*, QAP+gradient methods)

Shape matching

- Devising specific techniques for reaching a solution can be very challenging
- More general optimization algorithms need to be considered (*e.g.*, QAP+gradient methods)
- These algorithms tend to be slow in large-scale problems and get stuck at local minima (*e.g.*, parallel tangents)

Shape correspondence

- Let $X, Y \subseteq \mathbf{R}^n$ be two given shapes
- A point-to-point *correspondence* between X and Y can be defined as a *fuzzy* function $u : X \times Y \to [0,1]$ satisfying certain *mapping constraints*

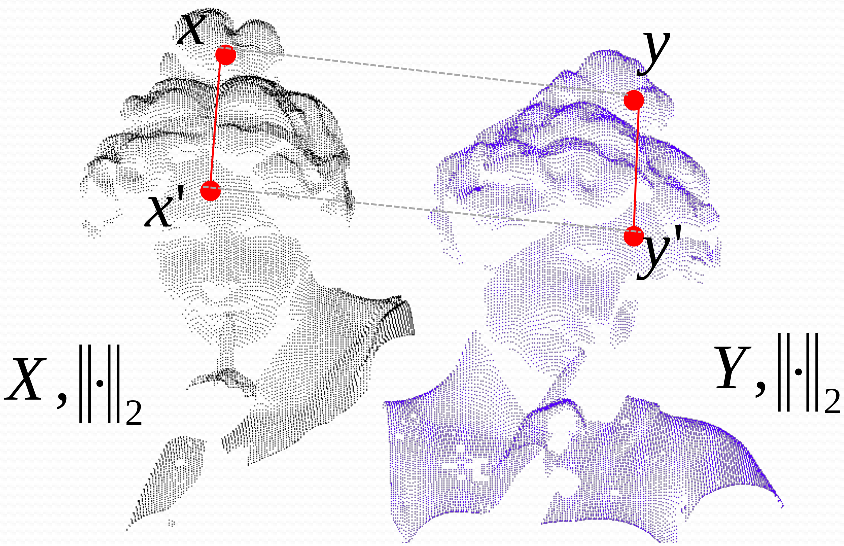


one-to-many one-to-one

- We will alternatively denote as correspondence the subset $U \subset X \times Y$ satisfying the constraints above

Local metric distortion

- Recall that a map $\varphi : (X, d_X) \to (Y, d_Y)$ is an *isometry* if for any $x, x' \in X$ one has
$$d_Y(\varphi(x), \varphi(x')) = d_X(x, x')$$
- Suppose $(x, y), (x', y') \in U$ are in correspondence
$$\varepsilon(x, y, x', y') = \left| d_X(x, x') - d_Y(y, y') \right|$$



Similarity-based matching

- Equivalently, one may define a measure *s* of similarity among pairs of matches
- We can then define the total similarity of a correspondence *U* as the weighted sum

$$\sum_{(x,y),(x',y')\in U} u(x,y)u(x',y')s\big((x,y),(x',y')\big)$$

which we aim at maximizing.

Quadratic assignment problem

$$\max_{\mathbf{x}} \ \mathbf{x}^T \mathbf{S} \mathbf{x}$$

$$\text{s.t.} \quad \Pi(\mathbf{x}) \leq \mathbf{b}$$

where $\mathbf{x} \in [0,1]^{|U|}$ is a vector representation of the correspondence function, $\mathbf{S}$ encodes the similarity terms, and $\Pi(\mathbf{x})$ specifies the mapping constraints on $\mathbf{x}$

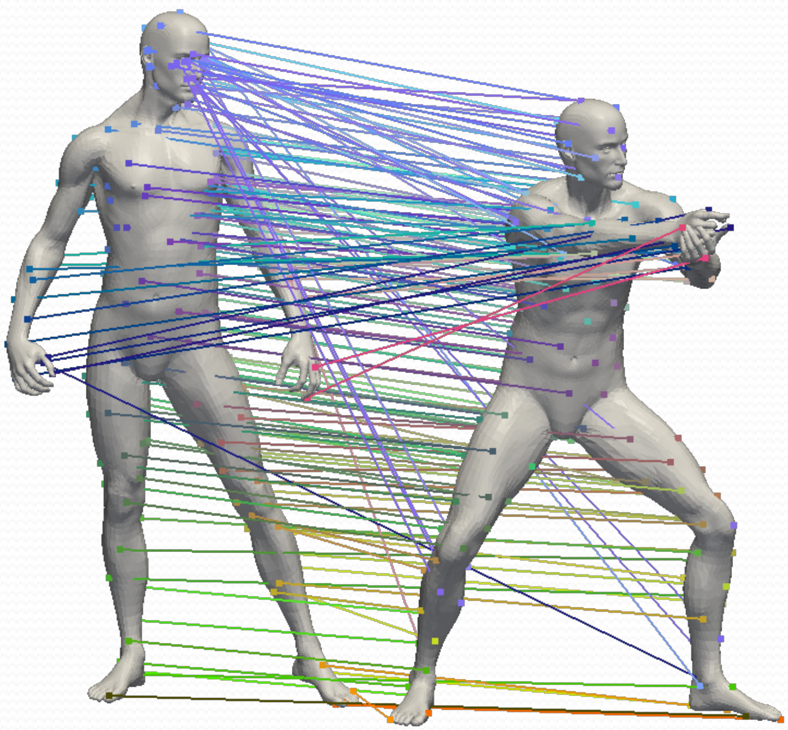
Spectral matching

- A well known relaxation to the QAP is given by:

$$\max \ \mathbf{x}^T \mathbf{S} \mathbf{x}$$

$$\text{s.t.} \quad \|\mathbf{x}\|_2^2 = 1, \ \ \mathbf{x} \in [0,1]^{|U|}$$

which is then globally maximized by the main eigenvector of **S**



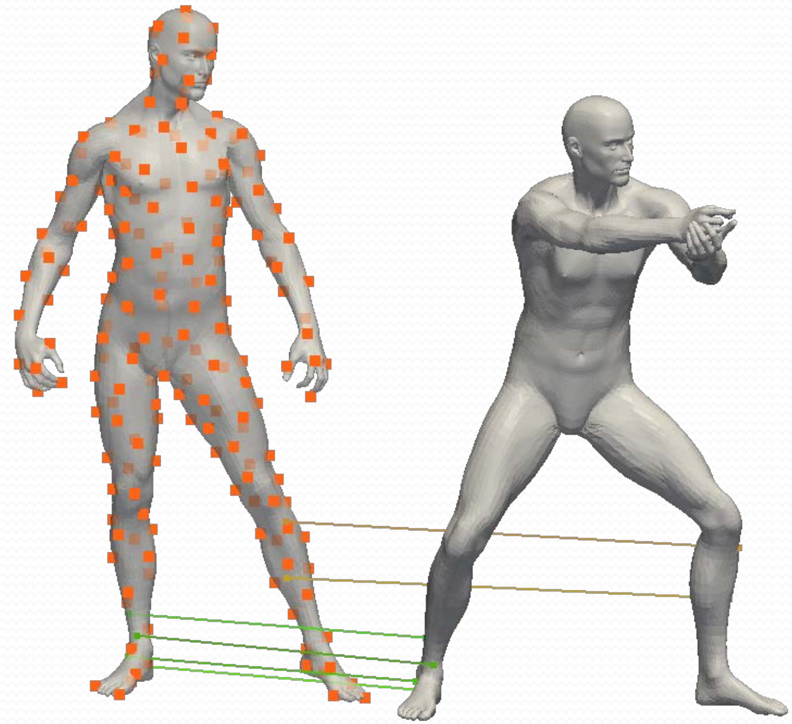
A spectral technique to correspondence problems using pairwise constraints. M.Leordeanu and M.Hebert. ICCV 2005.

Game-theoretic matching

- Replacing the *ridge*-like relaxation of the spectral approach with a simplex constraint

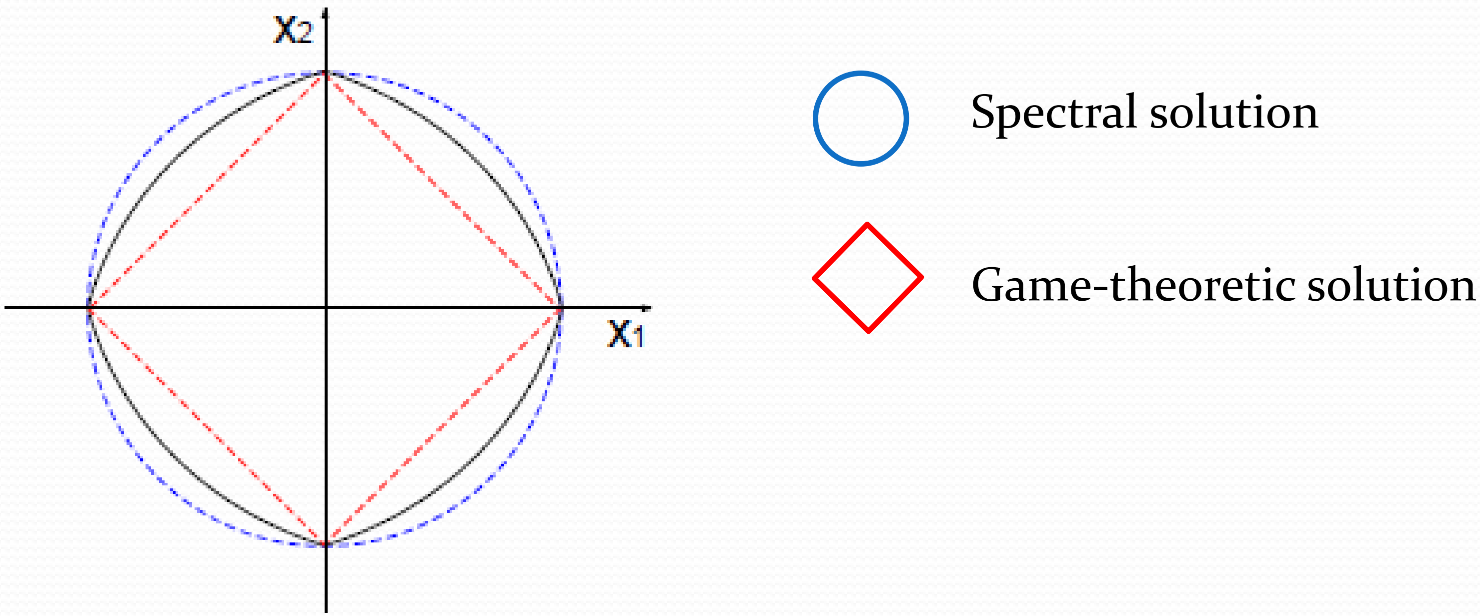
$$\max\ \mathbf{x}^T\mathbf{S}\mathbf{x}$$
$$\text{s.t.}\quad \|\mathbf{x}\|_1 = 1,\ \ \mathbf{x} \geq 0$$

This formulation tends to favor sparse solutions



A game-theoretic approach to deformable shape matching. E.Rodolà, A.M.Bronstein, A.Albarelli, F.Bergamasco, and A.Torsello. CVPR 2012.

Elastic Net constraints



$$(1-\alpha)\|\mathbf{x}\|_1 + \alpha\|\mathbf{x}\|_2^2 = 1, \quad \alpha \in [0,1]$$

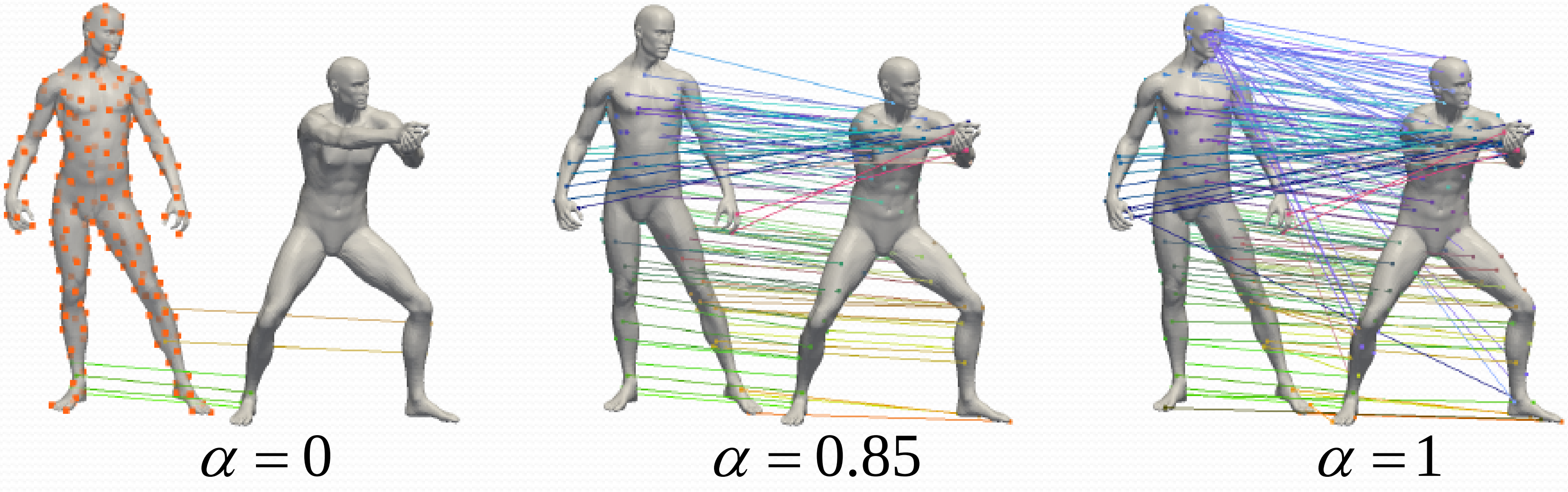
Regularization and variable selection via the elastic net. H.Zou and T.Hastie. Journal of the Royal Statistical Society Series B, 2005.

Elastic Net matching

$$\max\ \mathbf{x}^{\mathrm{T}}\mathbf{S}\mathbf{x}$$

$$\text{s.t.}\quad (1-\alpha)\|\mathbf{x}\|_1 + \alpha\|\mathbf{x}\|_2^2 = 1, \quad \alpha \in [0,1]$$

- Strict convexity guarantees the *grouping effect*
- The trade-off between size and error is regulated by α

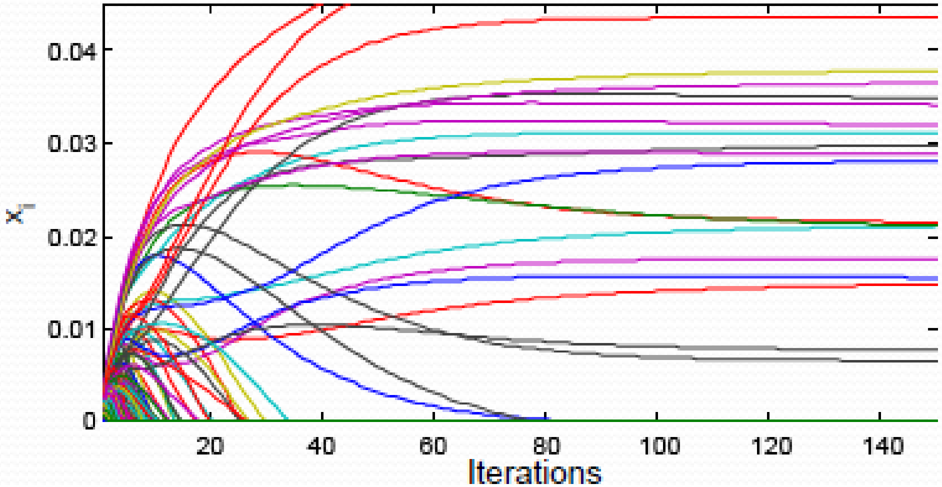


Optimization

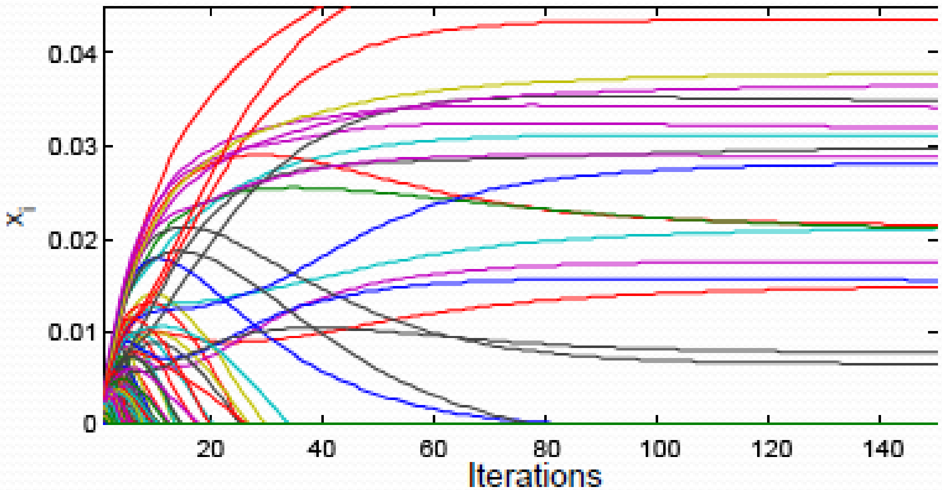
- We follow a projected gradient approach

$$\mathbf{x}^{(t+1)} = P_\alpha\left(\mathbf{x}^{(t)} + \delta \mathbf{S}\mathbf{x}^{(t)}\right)$$

where $P_\alpha : \mathbf{R}^{|U|} \to \mathbf{R}^{|U|}$ is a projection operator onto the elastic net ball with convexity α.



Accelerating the process



- Can we infer the general direction of the convergence process from past iterates?

Reduced rank extrapolation

- Consider the N-dimensional vector sequence $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots$ as generated by the linear process

$$\mathbf{x}_{i+1} = \mathbf{A}\mathbf{x}_i + \mathbf{b}$$

- Assume that neither $\mathbf{A}$ nor $\mathbf{b}$ are known, and only the sequence $\{\mathbf{x}_i\}$ or the generating process are given
- If the sequence converges, it does so to the unique solution to the system $\mathbf{s} = \mathbf{A}\mathbf{s} + \mathbf{b}$

Reduced rank extrapolation

- Consider the N-dimensional vector sequence $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots$ as generated by the linear process

$$\mathbf{x}_{i+1} = \mathbf{A}\mathbf{x}_i + \mathbf{b}$$

- The process often requires many iterations
- The individual terms $\mathbf{x}_i$ might be expensive to compute

$$\mathbf{x}^{(t+1)} = P_\alpha\left(\mathbf{x}^{(t)} + \delta \mathbf{S}\mathbf{x}^{(t)}\right)$$

projection onto mapping constraints set

Reduced rank extrapolation

- We would like to compute the limit $\lim_{i\to\infty} \mathbf{x}_i$ using as few terms as possible (denote this number by k)

$$\mathbf{s}^* = \sum_{i=0}^{k} \gamma_i \mathbf{x}_i \quad \text{s.t.} \sum_{i=0}^{k} \gamma_i = 1$$

- Define the first-order difference vectors for $i = 0,1,2,\ldots$

$$\mathbf{u}_i = \Delta \mathbf{x}_i = \mathbf{x}_{i+1} - \mathbf{x}_i$$

- We can solve for γ_i efficiently by minimizing

$$\min_{\gamma \in \mathbf{R}^{k+1}} \left\| \sum_{i=0}^{k} \gamma_i \mathbf{u}_i \right\|_2 \quad \text{s.t.} \sum_{i=0}^{k} \gamma_i = 1$$

Nonlinear equations

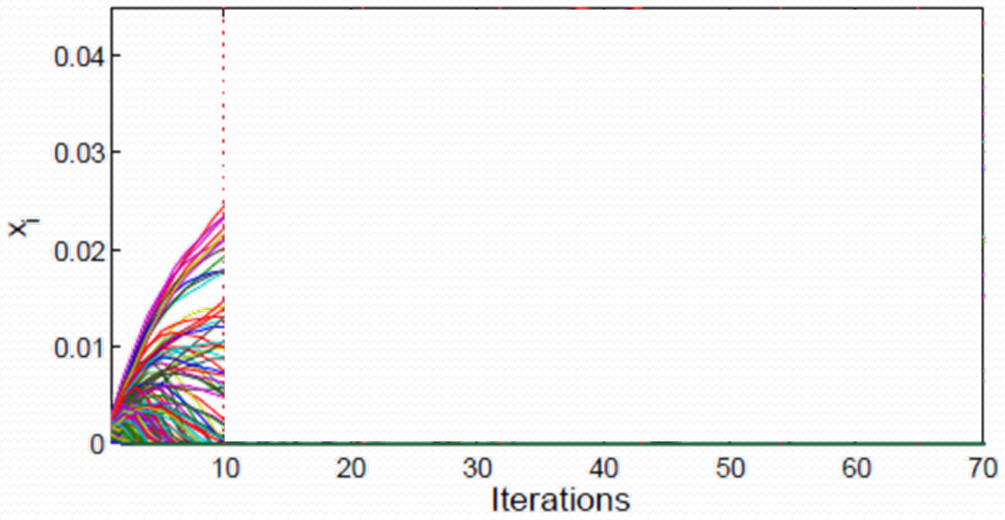
- The extrapolation technique provides the provably optimal solution when the input sequence is linearly generated
- In our setting, the sequence is generated by a non-linear process

$$\mathbf{x}^{(t+1)} = P_{\alpha}\left(\mathbf{x}^{(t)} + \delta \mathbf{S}\mathbf{x}^{(t)}\right)$$

- This scenario is representative of a whole range of minimum-distortion matching methods

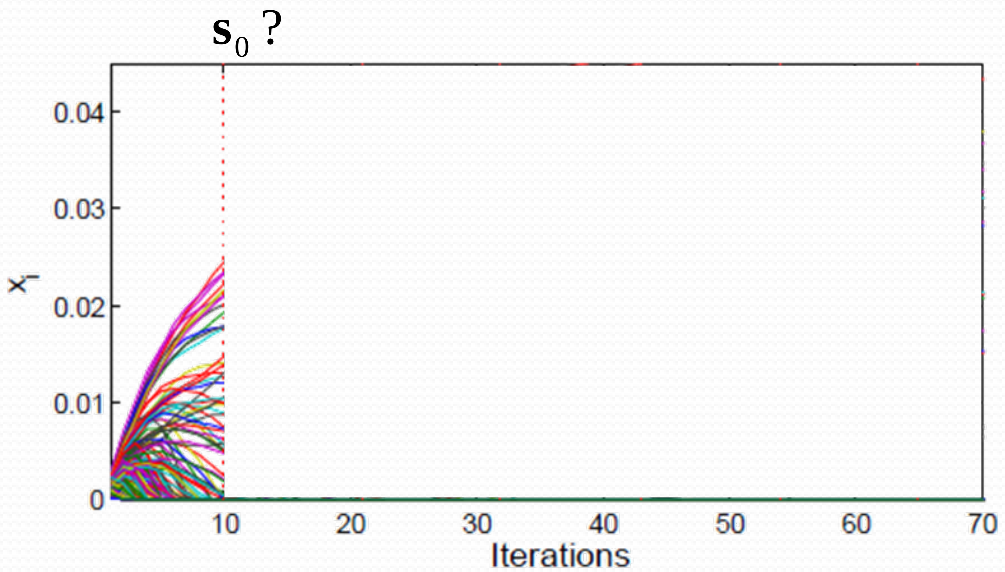
Cycling

- Choose parameters n, k
  1. Perform $n+k+1$ standard projected gradient steps, saving the last $k+1$ vectors as $\{\mathbf{x}_i\}$



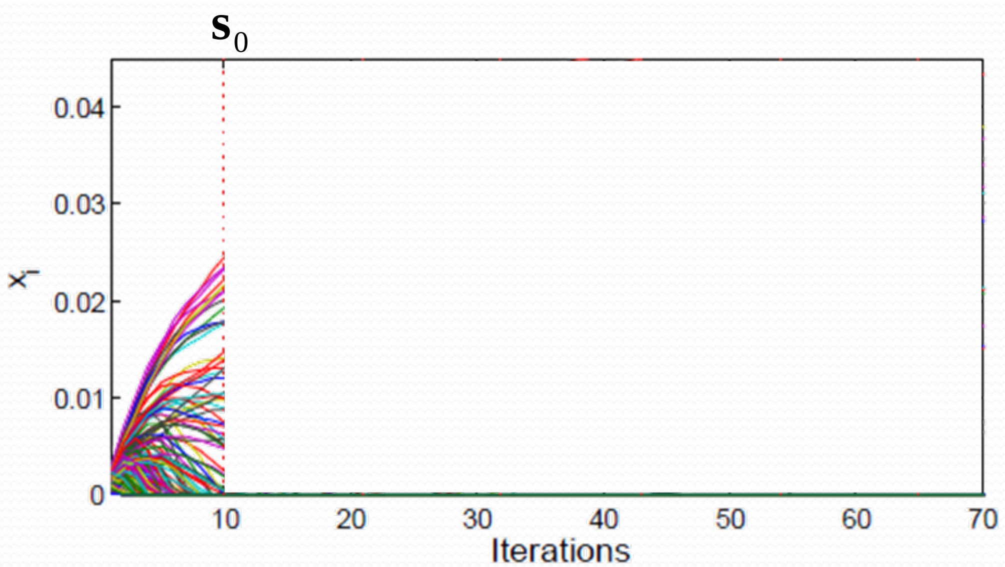
Cycling

- Choose parameters n, k
  1. Perform $n+k+1$ standard projected gradient steps, saving the last $k+1$ vectors as $\{\mathbf{x}_i\}$
  2. Apply RRE to the $\{\mathbf{x}_i\}$ and obtain the estimate $\mathbf{s}_{n,k}$



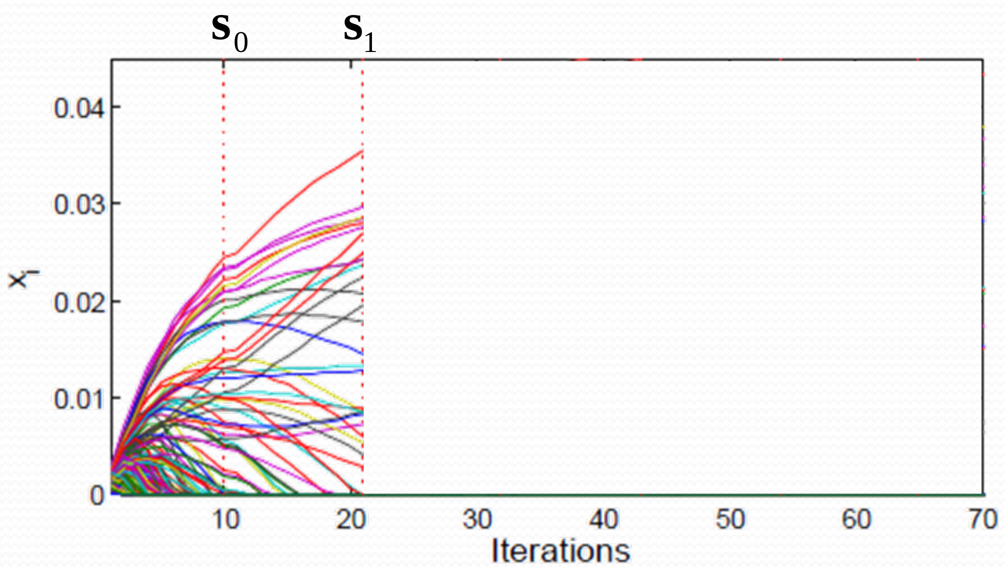
Cycling

- Choose parameters n, k
  1. Perform $n+k+1$ standard projected gradient steps, saving the last $k+1$ vectors as $\{\mathbf{x}_i\}$
  2. Apply RRE to the $\{\mathbf{x}_i\}$ and obtain the estimate $\mathbf{s}_{n,k}$
  3. Project back onto the feasible set, *i.e.* compute $P_\alpha(\mathbf{s}_{n,k})$



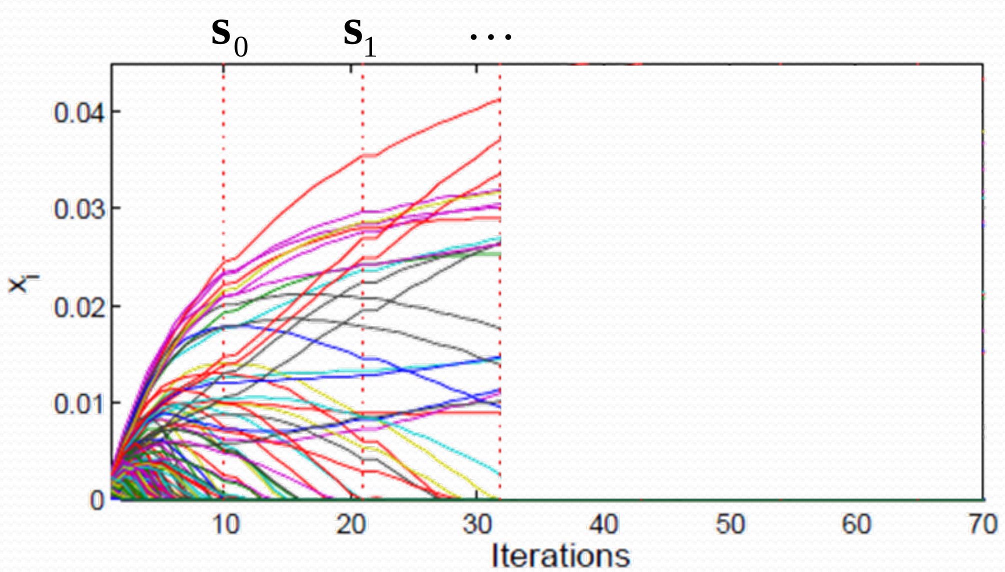
Cycling

- Choose parameters n, k
  1. Perform $n+k+1$ standard projected gradient steps, saving the last $k+1$ vectors as $\{\mathbf{x}_i\}$
  2. Apply RRE to the $\{\mathbf{x}_i\}$ and obtain the estimate $\mathbf{s}_{n,k}$
  3. Project back onto the feasible set, *i.e.* compute $P_\alpha(\mathbf{s}_{n,k})$
  4. Check objective value of the projected solution and repeat



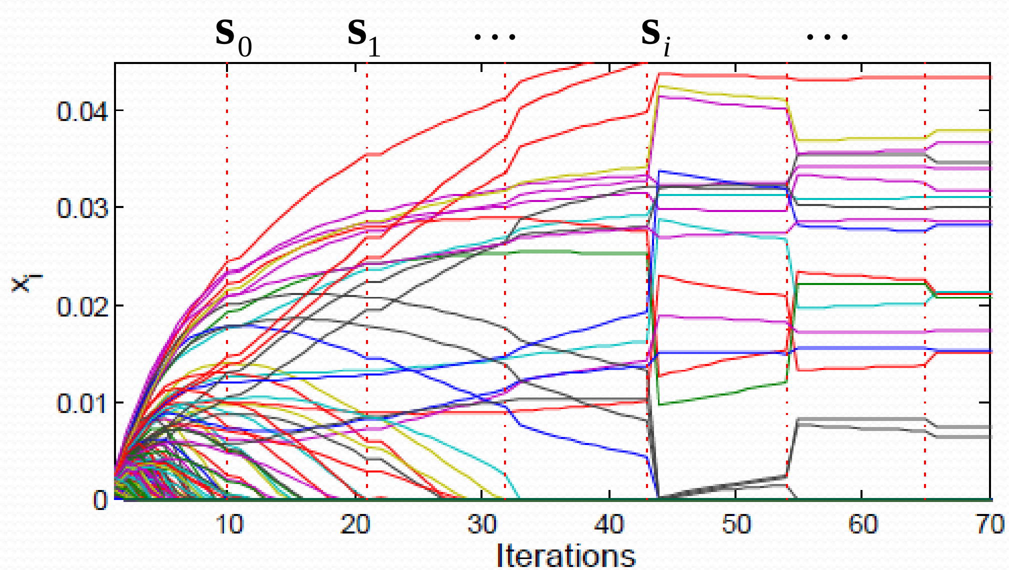
Cycling

- Choose parameters n, k
  1. Perform $n+k+1$ standard projected gradient steps, saving the last $k+1$ vectors as $\{\mathbf{x}_i\}$
  2. Apply RRE to the $\{\mathbf{x}_i\}$ and obtain the estimate $\mathbf{s}_{n,k}$
  3. Project back onto the feasible set, *i.e.* compute $P_\alpha(\mathbf{s}_{n,k})$
  4. Check objective value of the projected solution and repeat



Cycling

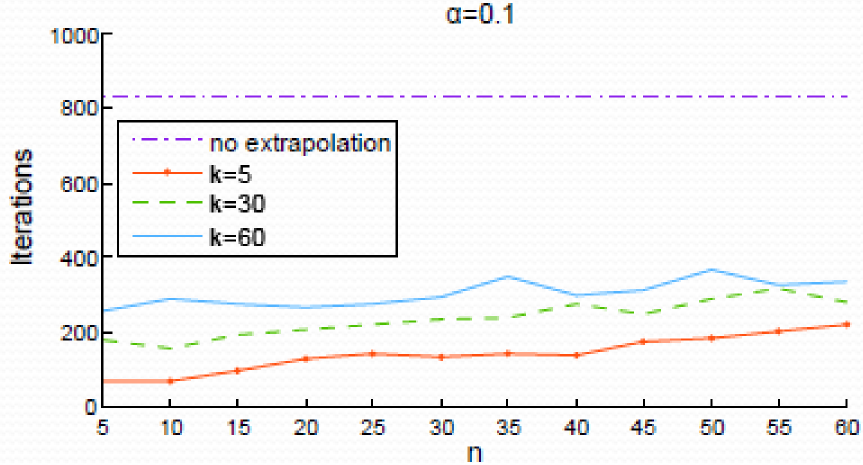
- Choose parameters n, k
  1. Perform $n+k+1$ standard projected gradient steps, saving the last $k+1$ vectors as $\{\mathbf{x}_i\}$
  2. Apply RRE to the $\{\mathbf{x}_i\}$ and obtain the estimate $\mathbf{s}_{n,k}$
  3. Project back onto the feasible set, *i.e.* compute $P_\alpha(\mathbf{s}_{n,k})$
  4. Check objective value of the projected solution and repeat

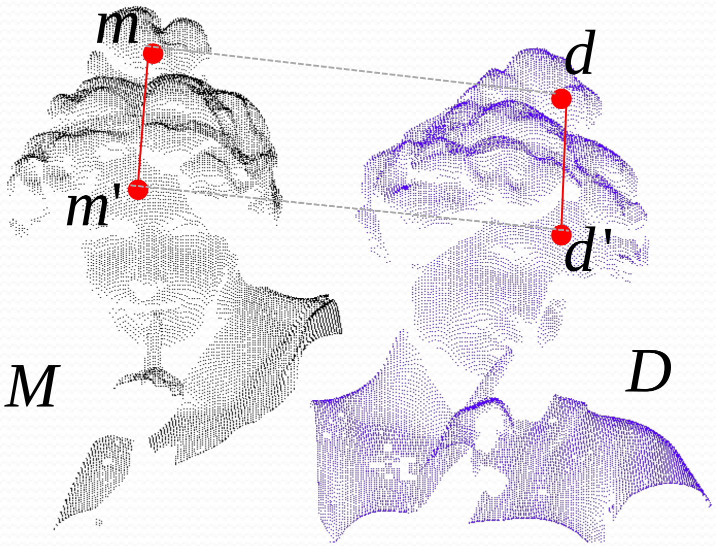


Rigid matching of point clouds

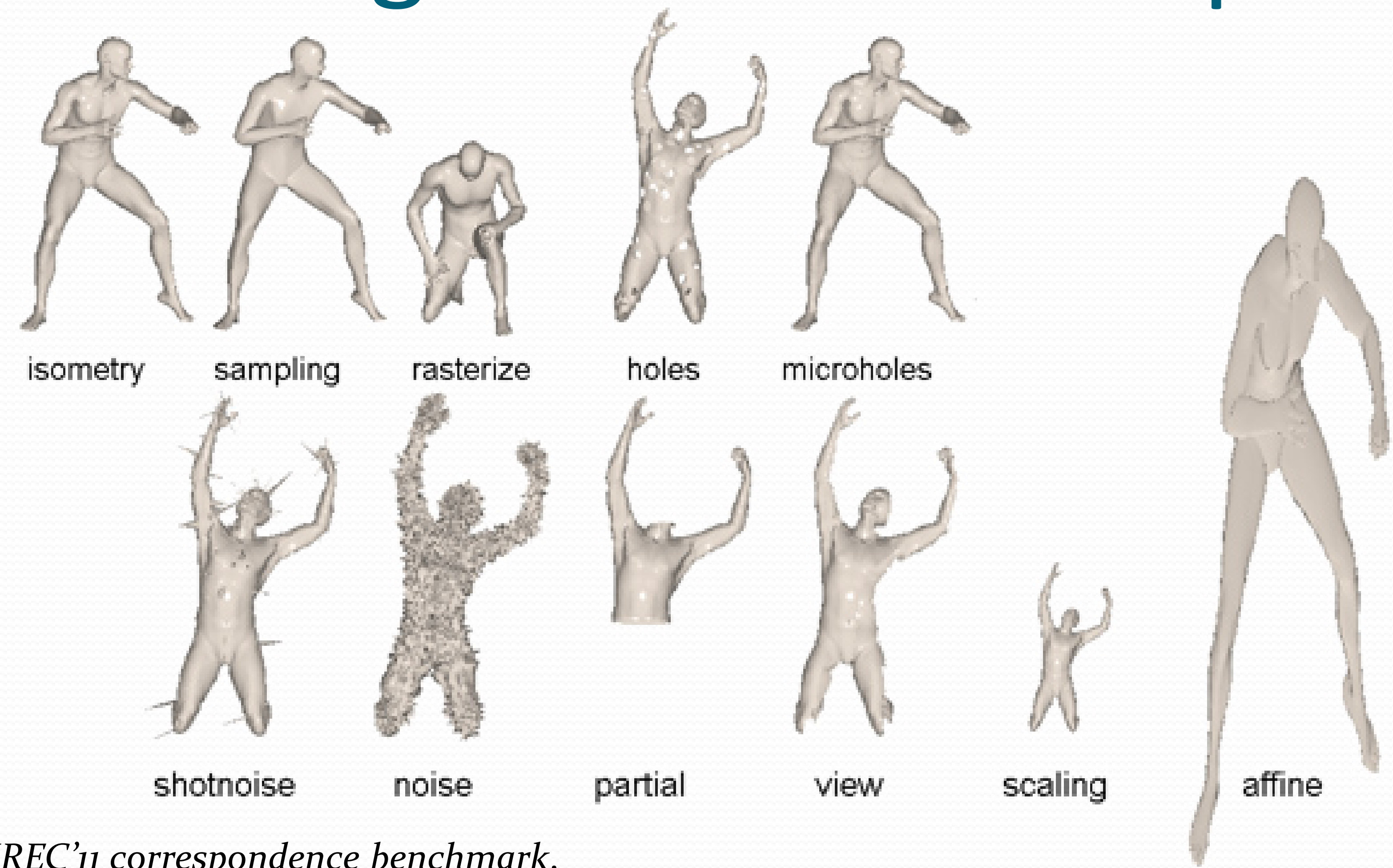
- Enforce rigid isometries among two point clouds $M, D \subseteq \mathbf{R}^3$
  i.e., define the similarity among two pairs of matches $(m,d),(m',d') \in M \times D$ as

$$s((m,d),(m',d')) = \frac{\min\{\|m-m'\|_2, \|d-d'\|_2\}}{\max\{\|m-m'\|_2, \|d-d'\|_2\}}$$





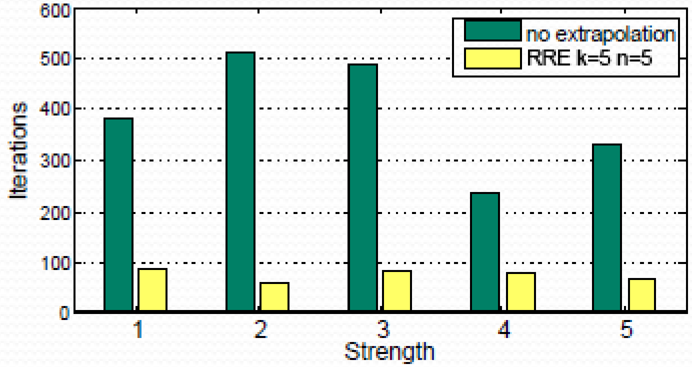
Matching deformable shapes



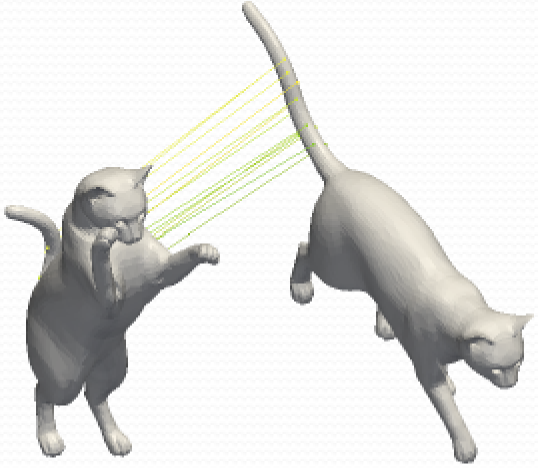
SHREC'11 correspondence benchmark.
http://tosca.cs.technion.ac.il/book/shrec_correspondence.html

Matching deformable shapes

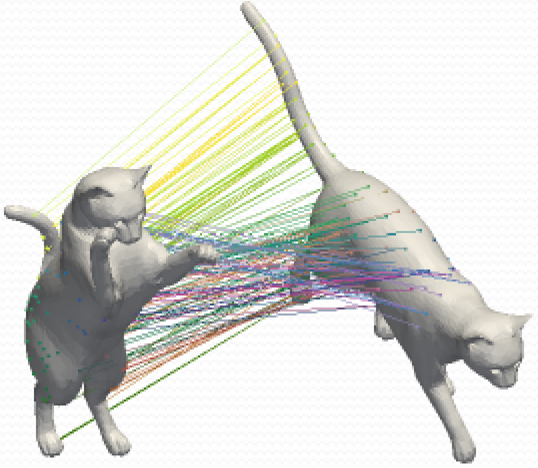
- Replace the Euclidean metric with an intrinsic metric
- A very challenging problem!



$\alpha = 0.1$



$\alpha = 0.85$



•Almost one order of magnitude of improvement

•State-of-the-art results on SHREC'10 dataset, isometry class

Multiple-view stereo

- In order to meet the strong selectivity requirements of this class of problems, we set $\alpha = 0.2$

- Only a moderate improvement (~ 30% less iterations)
- Using short cycles led to premature convergence
- The presence of repeated structure and textures makes the whole setting more unstable

Middlebury multi-view stereo dataset. http://vision.middlebury.edu/mview/data/