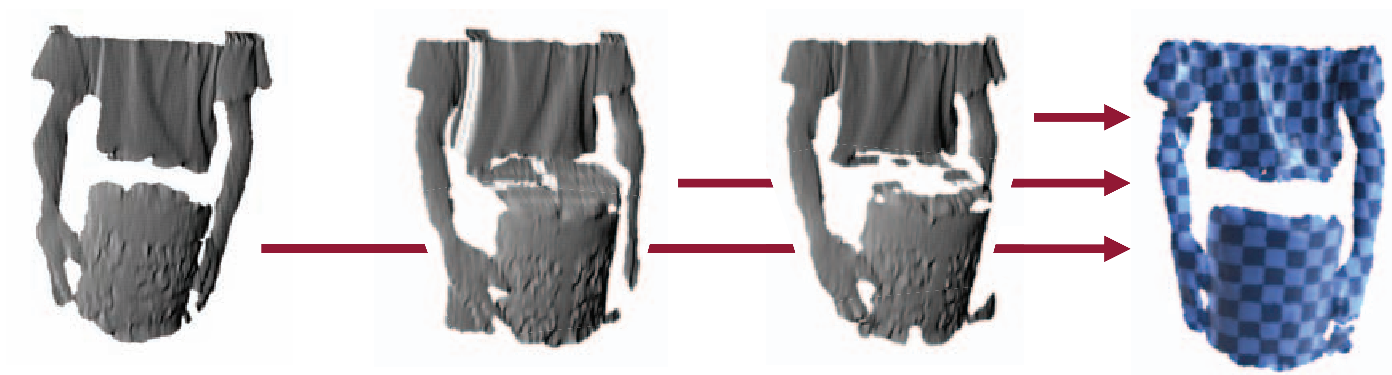
Reconstruction techniques for shape and motion



DAGM 2009 Tutorial

Visual 3D Scene Analysis and Synthesis

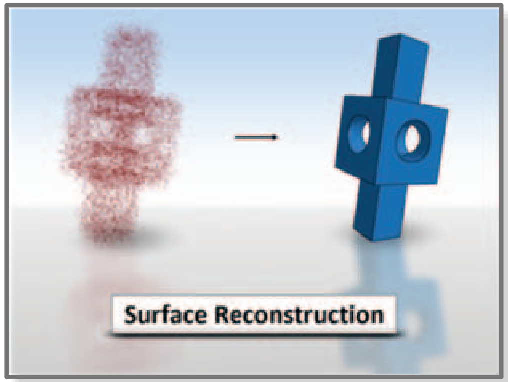
Michael Wand

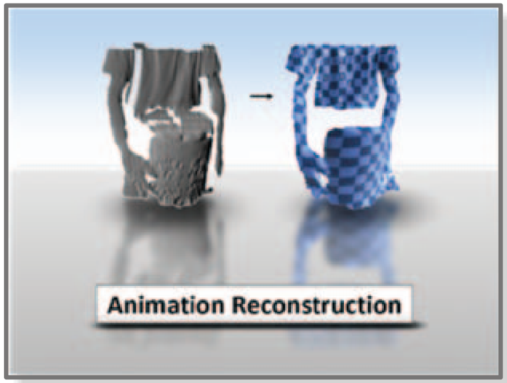
Universität des Saarlandes,
Max-Planck Institut Informatik Saarbrücken

Reconstructing Shape and Motion

Overview

- Introduction
- Surface Reconstruction
  - Geometric reconstruction
  - Bayesian smoothing
- Animation Reconstruction
  - Trajectory fitting
  - Improved factorization algorithm
- Global Deformable Matching
- Conclusions



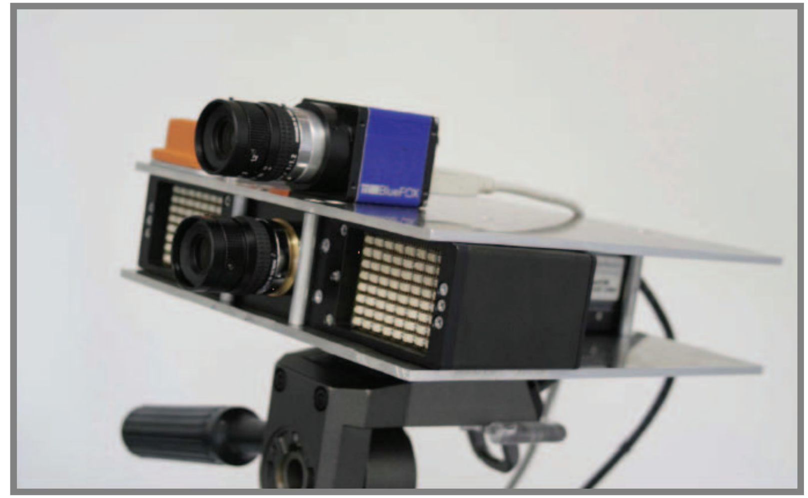


Introduction

3D Animation Scanner

New technology

- 3D animation scanners
- Record 3D video
- Active research area



Ultimate goal

- 3D movie making
- New creative perspectives

Structured Light Scanners

space-time stereo

courtesy of James Davis, UC Santa Cruz

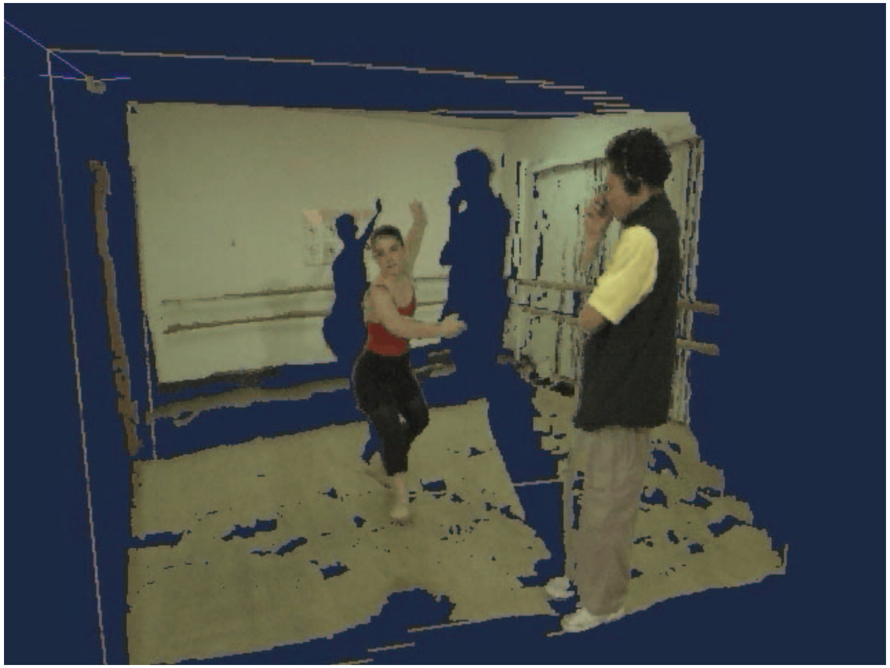
color-coded structured light

courtesy of Phil Fong, Stanford University

motion compensated structured light

courtesy of Sören König, TU Dresden

Passive Multi-Camera Acquisition



segmentation & belief propagation

[Zitnick et al. 2004]
Microsoft Research

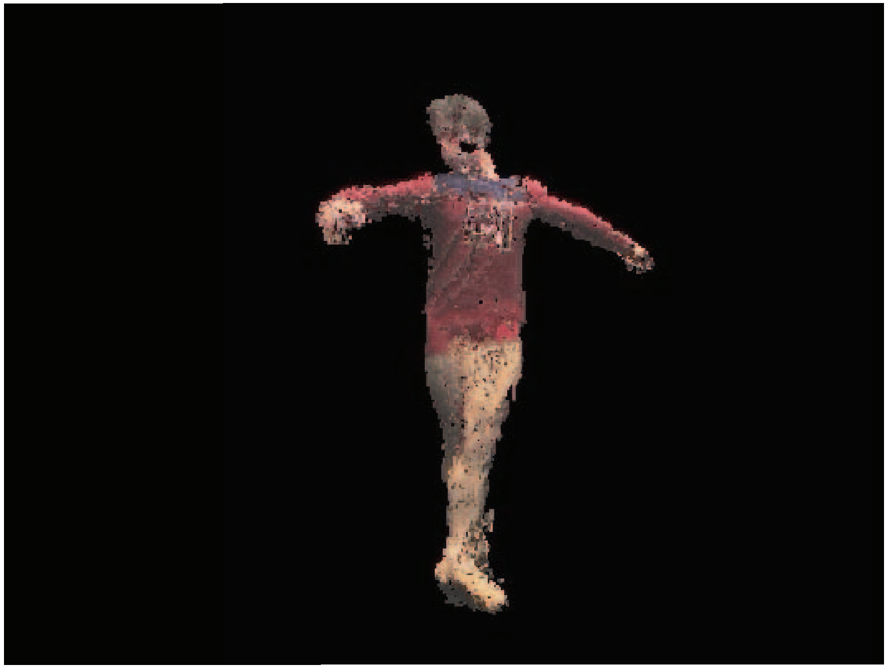
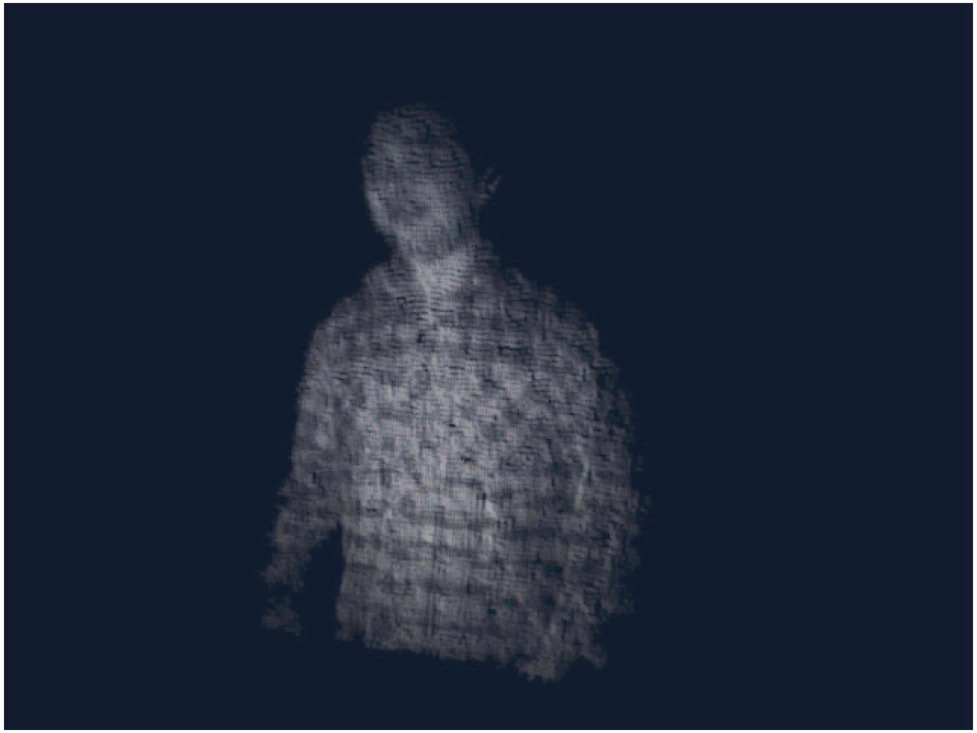
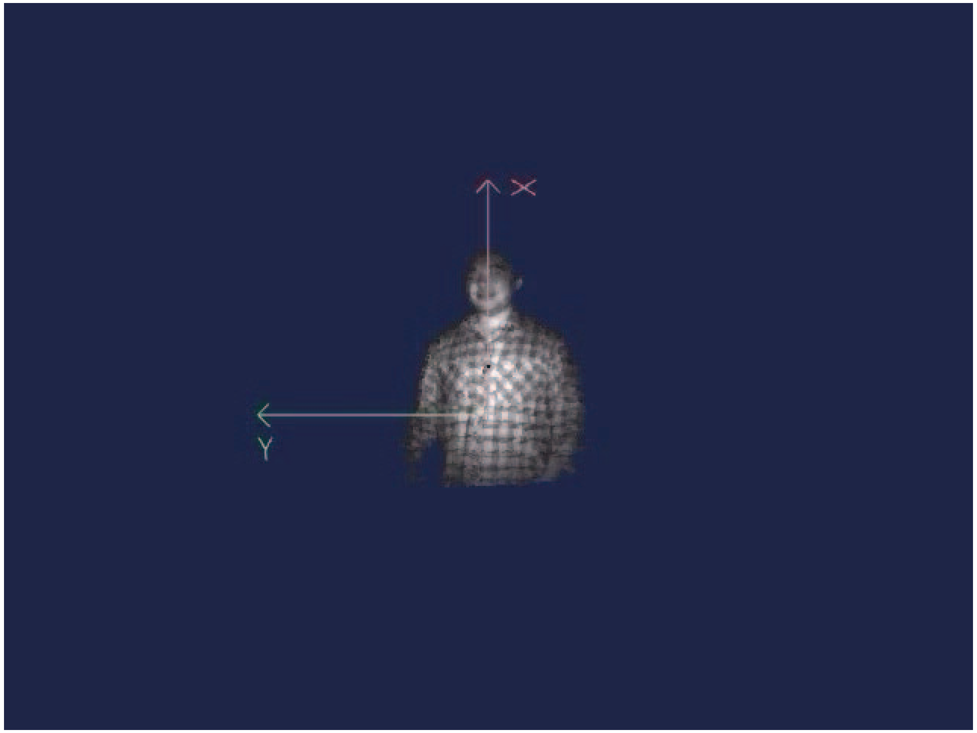


photo-consistent space carving

Christian Theobald
MPI-Informatik

Time-of-Flight / PMD Devices





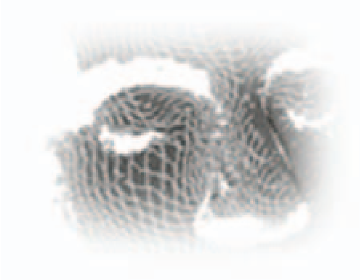
Swiss Ranger Time-of-flight camera

Animation Reconstruction

Problems

- Noisy data
- Incomplete data (acquisition holes)
- No correspondences

noise



holes

missing correspondences

Animation Reconstruction

Remove noise, outliers

Fill-in holes
(from all frames)

Dense correspondences

Geometric Reconstruction

This part of the tutorial

- Geometric reconstruction
- Assumptions
  - A set of point clouds
  - One for each time step

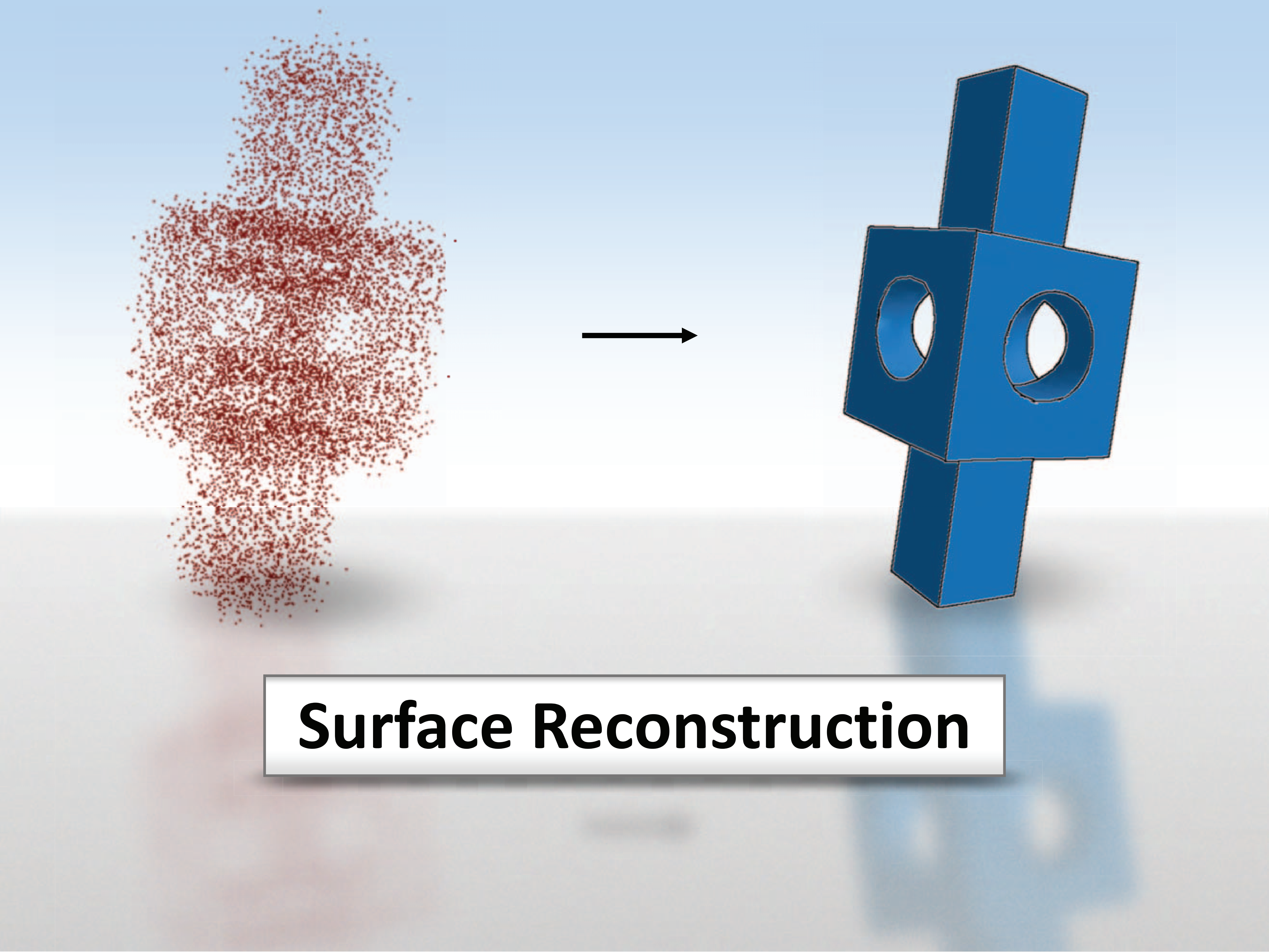
Motivation

- Most techniques output point clouds
- Direct reconstruction from video $\rightarrow$ next session

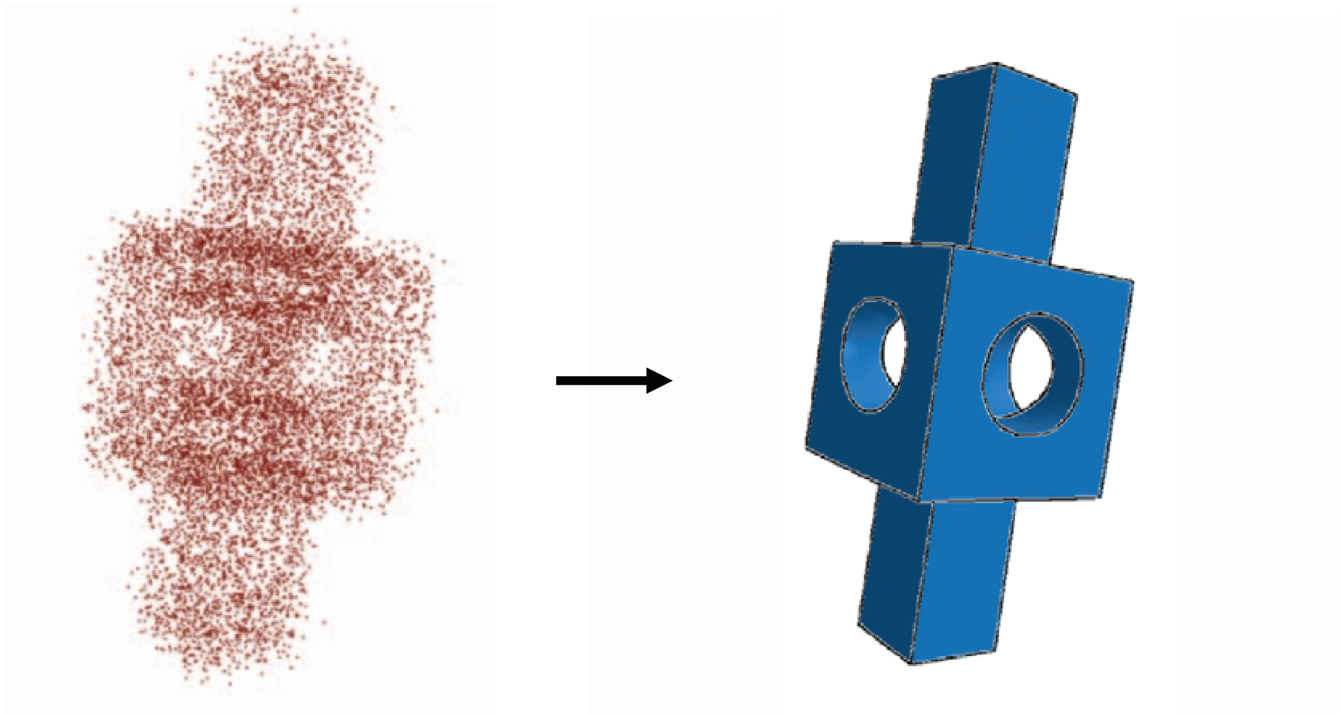
Roadmap...

Addressing the problem in four steps:

- Surface reconstruction
  - Surfaces from unorganized points
- Animation reconstruction
  - Handling time sequences
- Factorization model
  - More efficient algorithm
- Global deformable matching
  - No coherency assumptions
  - Open problems...


Surface Reconstruction

Surface Reconstruction



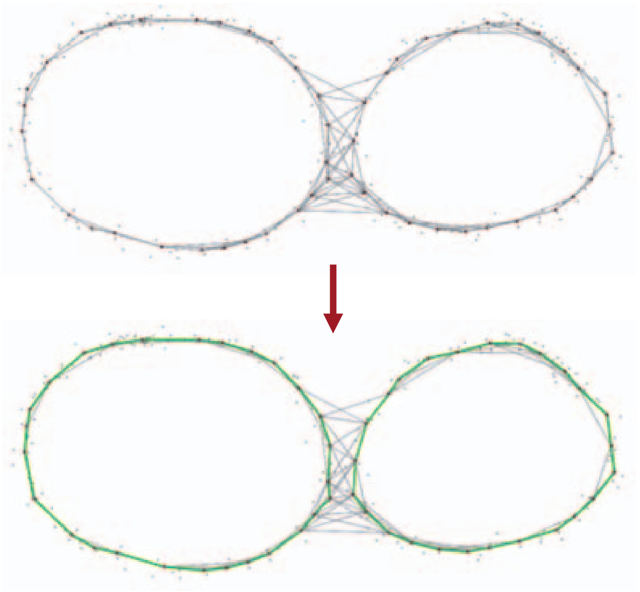
Surface Reconstruction

- Input: Noisy raw scanner data
- Output: “Nice” surface

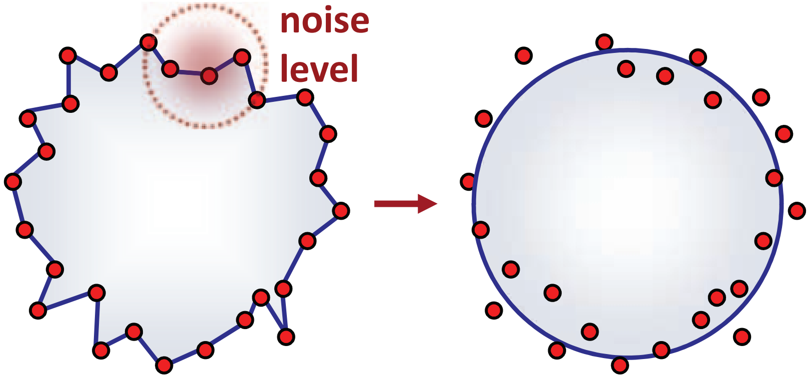
Surface Reconstruction

Subproblems:

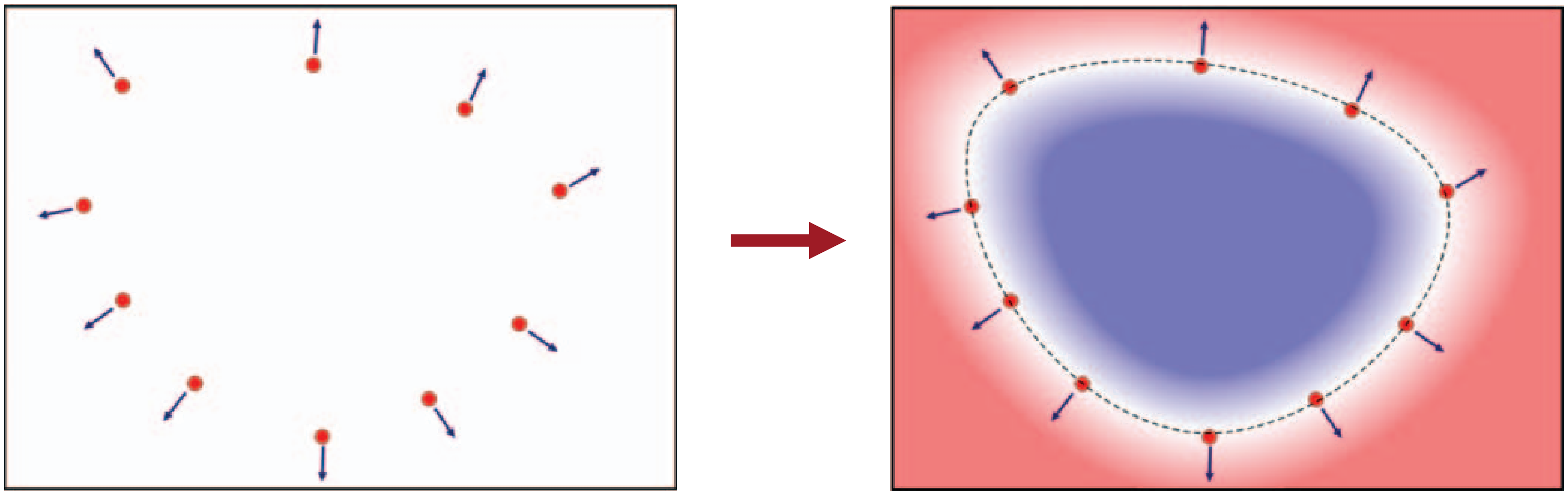
- Topology Reconstruction
  - How to “connect the dots”?
  - Compute a surface that is of the same topology as the (unknown) original surface.



- Geometry Smoothing
  - Remove noise
  - Compute a surface close to the data that does not contain noise artifacts



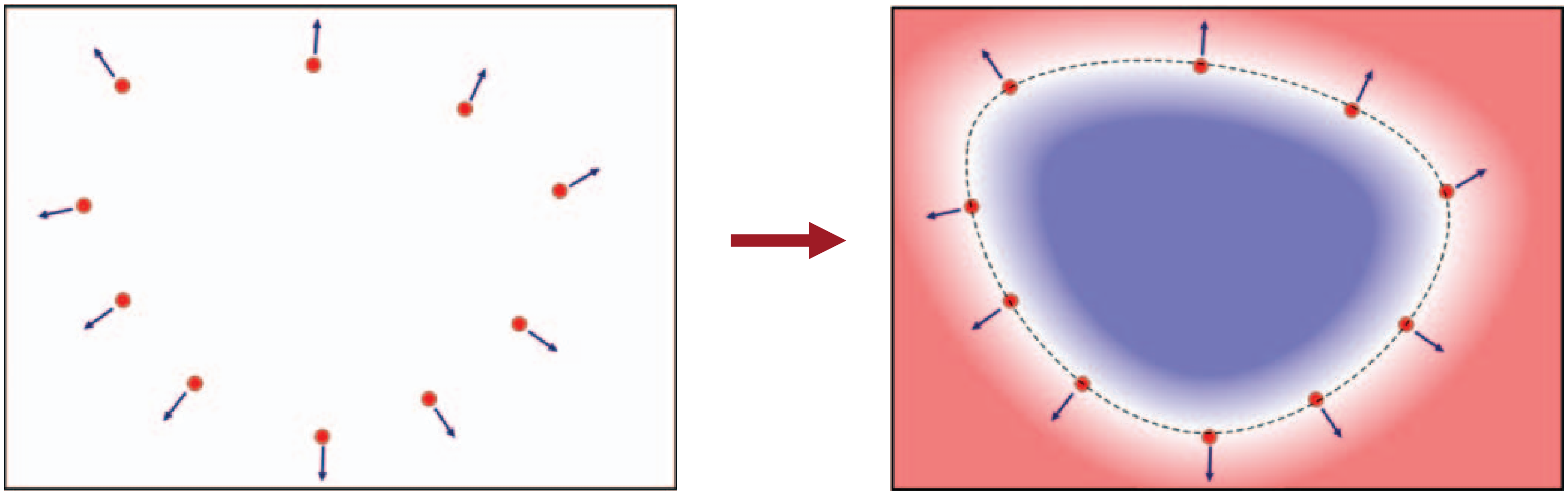
Surface Reconstruction Techniques



Implicit Functions

- Global: [*Hoppe et al. 92*], [*Turk et al. 99*], [*Carr et al. 01*], [*Ohtake et al. 03*], ...
- Local – moving least squares: [*Levin 03*], [*Alexa et al. 01 ff.*], [*Shen et al. 04*], [*Fleishman et al. 05*]

Surface Reconstruction Techniques



Implicit Functions

- Fit an implicit function to data points
- Regularization (smoothness of implicit function)
- Topology: Estimated implicitly
- Geometry: Smoothness depends on regularizer

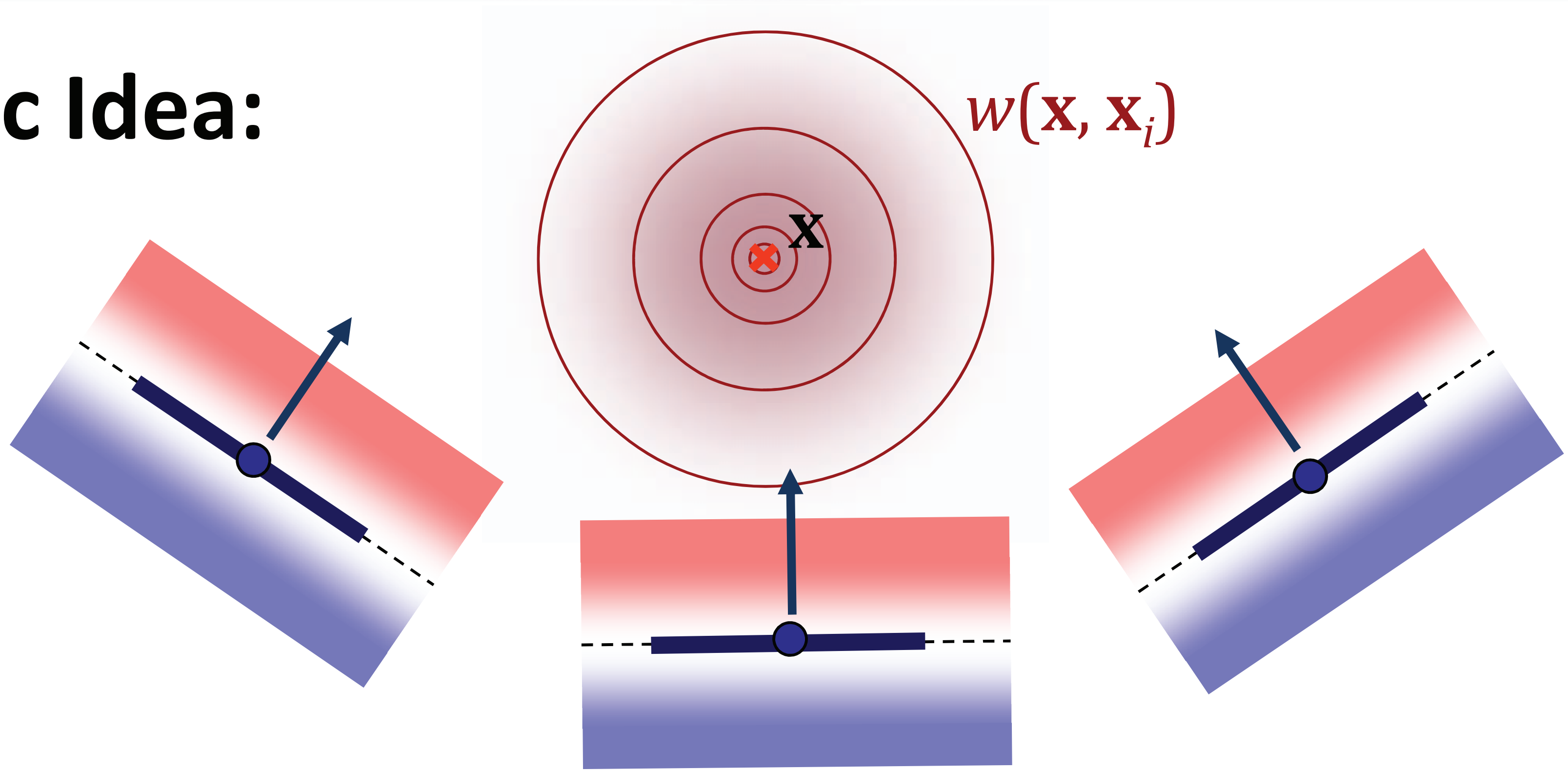
Example

Example: Moving Least Squares reconstruction

- Assume points with *consistently oriented* normals
- Idea:
  - Each point describes a half space
  - Blend between these by distance
  - Using a blending kernel
  - Reference: [Shen et al. 04]

Normal Constraints

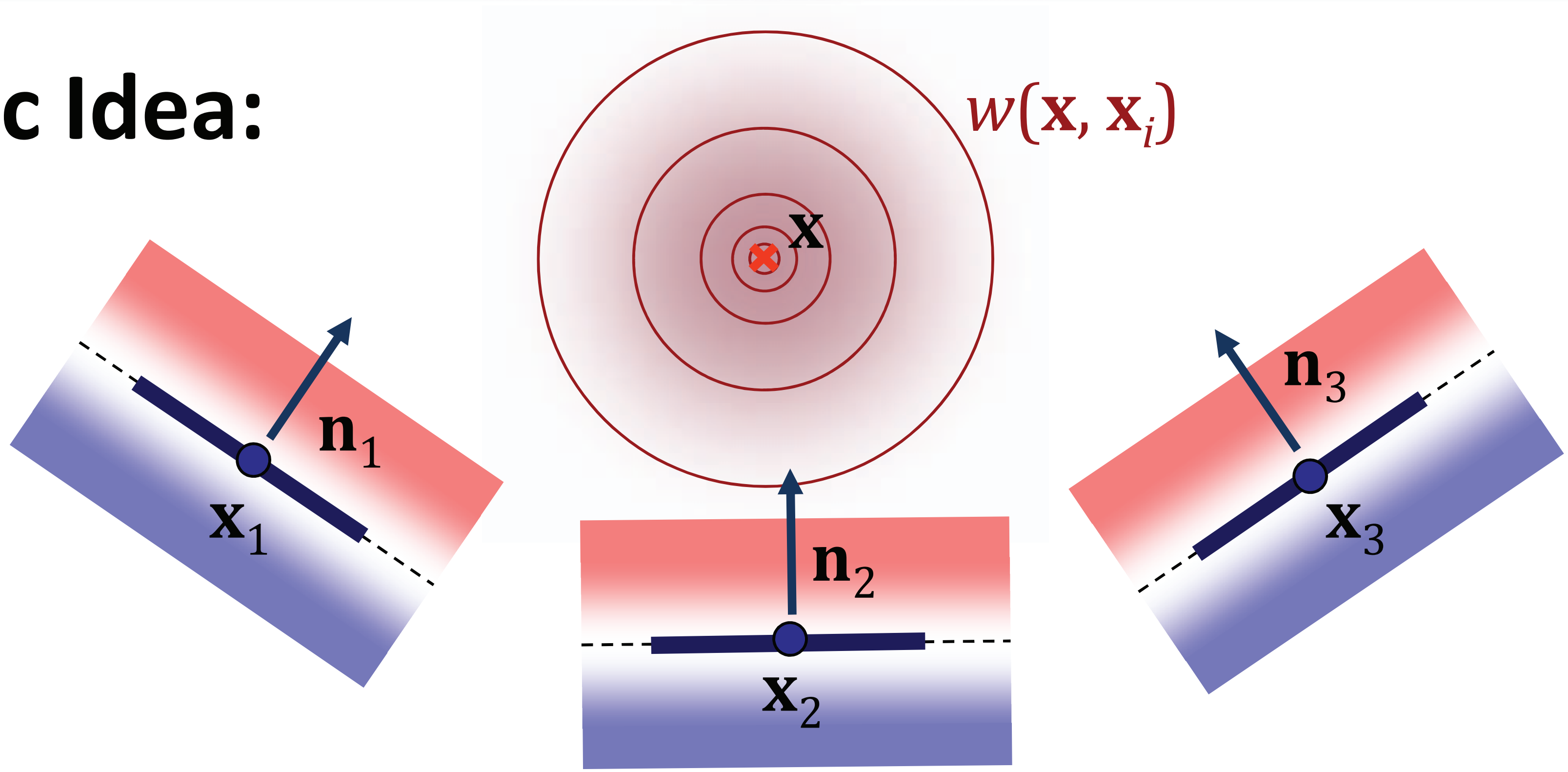
Basic Idea:



- Oriented plane, signed distance function
- Blend these distance functions with weights from a kernel function (e.g. Gaussian window)

Normal Constraints

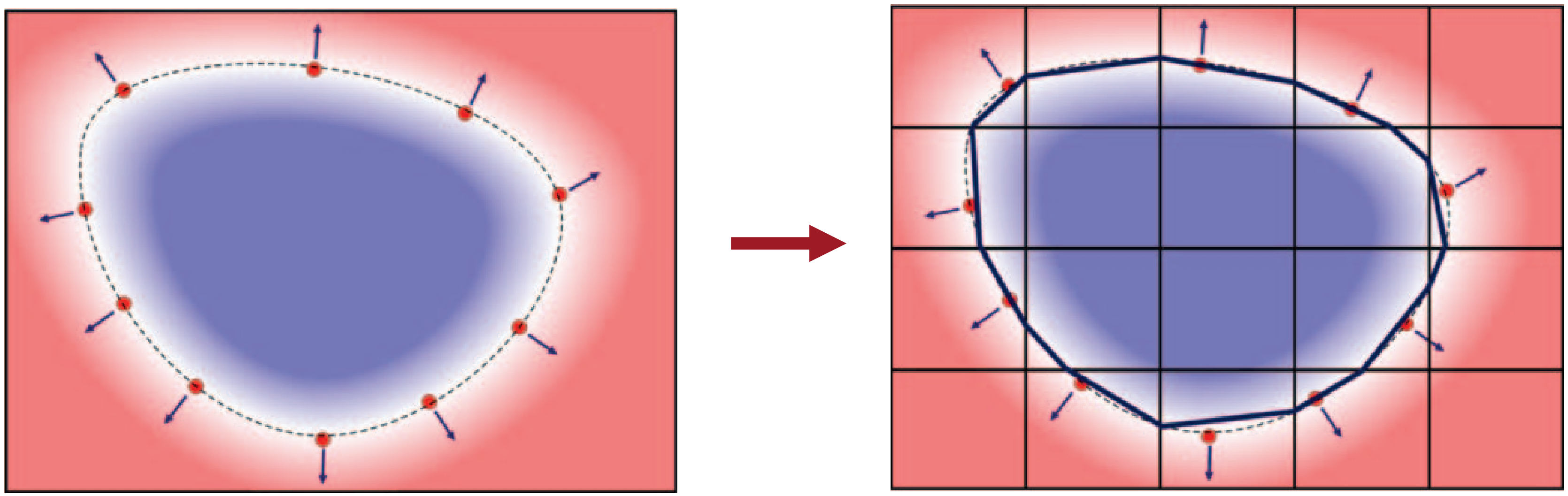
Basic Idea:



$$f(\mathbf{x}) = \frac{\sum_{i=1}^{n} \langle \mathbf{n}_i, \mathbf{x} - \mathbf{x}_i \rangle w(\|\mathbf{x} - \mathbf{x}_i\|)}{\sum_{i=1}^{n} w(\|\mathbf{x} - \mathbf{x}_i\|)}$$

(partition of unity weights)

Mesh Extraction



Optional: Mesh extraction

- Using marching cubes
- Alternative: Use point-based model directly
  - Project points to zero level set
  - Add additional points if necessary (sampling density)

Surface Reconstruction Techniques

Voronoi / Delauny techniques:

- [*Amenta et al. 98*], [*Amenta et al. 01*], [*Mederos et al. 05*], [*Kolluri et al. 04*], [*Alliez et al. 07*], ...
- Estimate "restricted Delauny triangulation"
- Same topology as original surface for dense enough sampling
- "Dense enough": ε-fraction of medial axis
- Basic technique computes *topology* and *coarse geometry*.

Surface Reconstruction Techniques

In this tutorial

- Statistically motivated techniques
- Reason: Easy to generalize to *animated* data

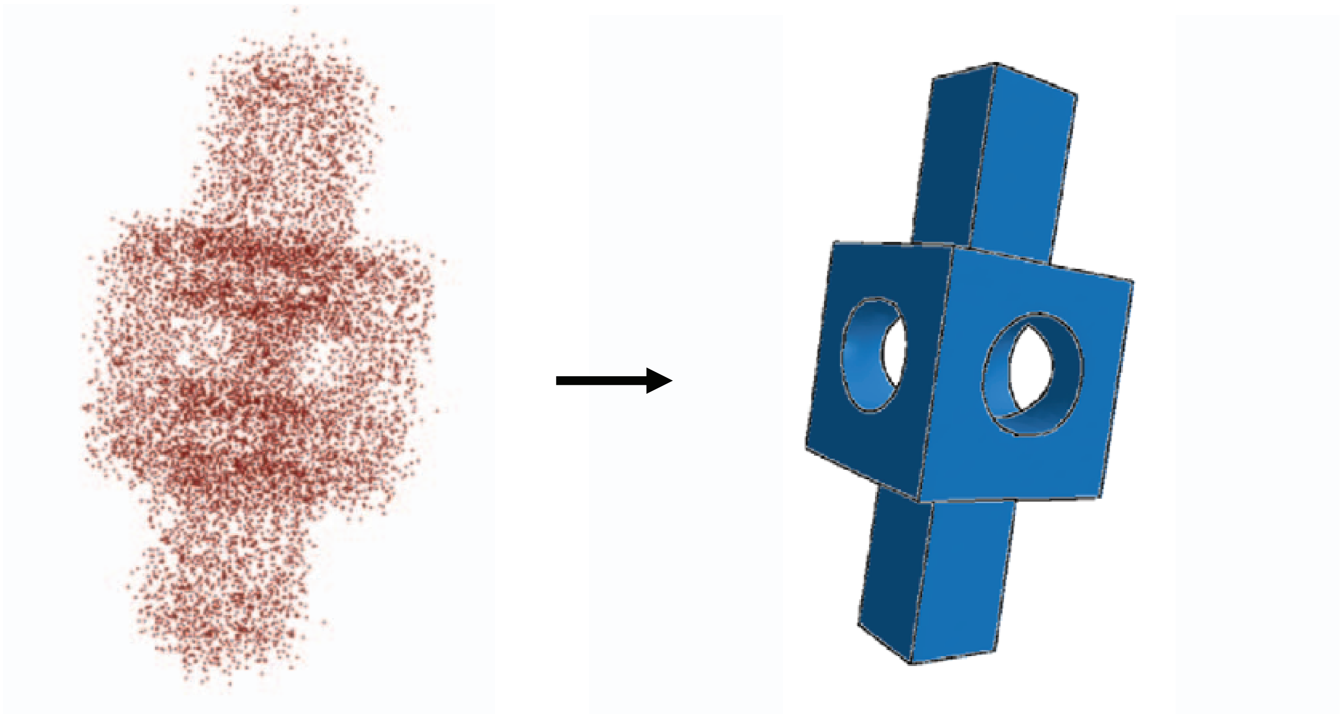
Statistically Motivated

- [*Ivrissimtzis et al. 03*], [*Pauly et al. 04*], [*Steinke et al. 05*]

Bayesian Smoothing

- [*Diebel et al. 06*], [*Jenke et al. 06*]

Surface Reconstruction



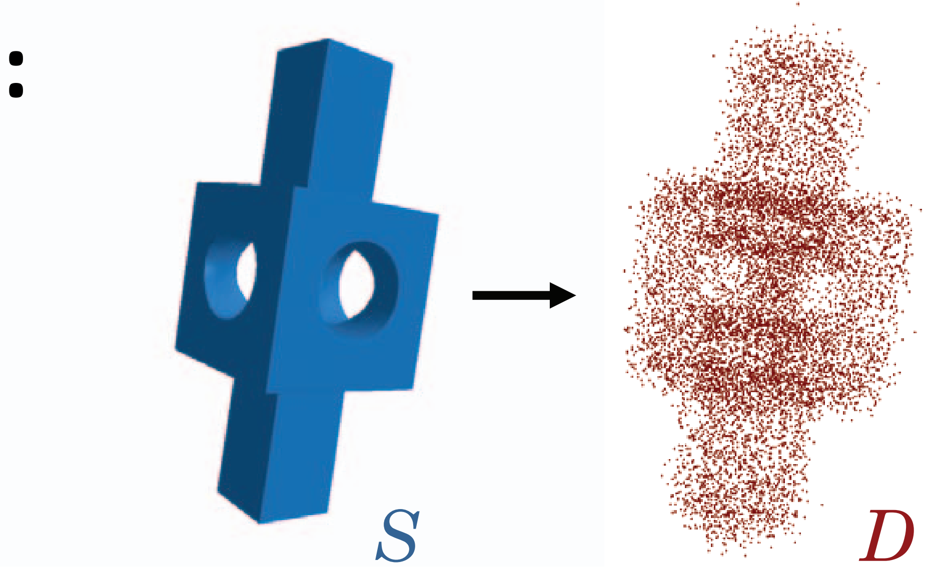
Surface Reconstruction

- Input: Noisy raw scanner data
- Output: "Nice" surface

Bayesian Approach

Bayesian reconstruction:

- Probability space
  $\Omega = \Omega_S \times \Omega_D$
- S – original model
  D – measurement data



- Bayes' rule:

$$P(S \mid D) = \frac{P(D \mid S)\, P(S)}{P(D)}$$

- Find most likely S

Bayesian Approach

measurement model ("likelihood")

prior assumptions

$$P(S \mid D) = \frac{P(D \mid S)\, P(S)}{\cancel{P(D)}}$$

optimize (best S)

Log Space:

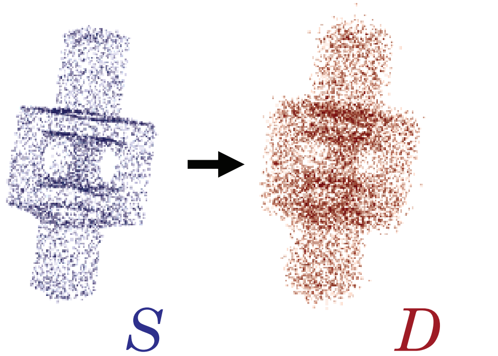
$$E(S \mid D) \sim E(D \mid S) + E(S)$$

measurement potential

prior potential

The Space of All Scenes

What is the space of all scenes?

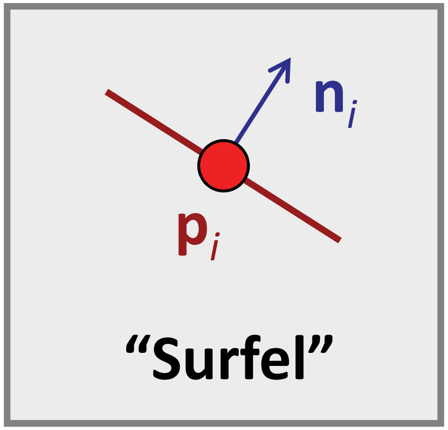


- Discretized model
- Pretend that the original scene has been a point cloud, too.
- $\Omega_S = \mathbb{R}^{3n}$, $\Omega_D = \mathbb{R}^{3m}$
- Define probability density $p(D, S)$ on Ω.
- Truncate p to make it well defined (bounding box support).

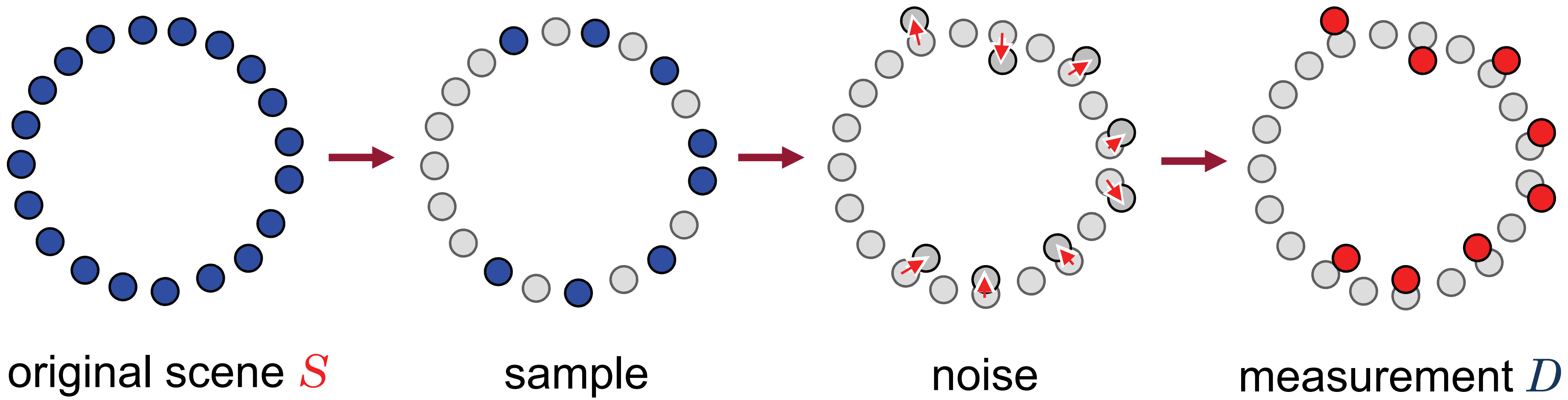
Implementation...

Implementation: Point-based model

- Our model is a set of points
- "Surfels": Every point has a latent surface normal
- We want to estimate *position* and *normals*



Measurement Model – P(D|S)

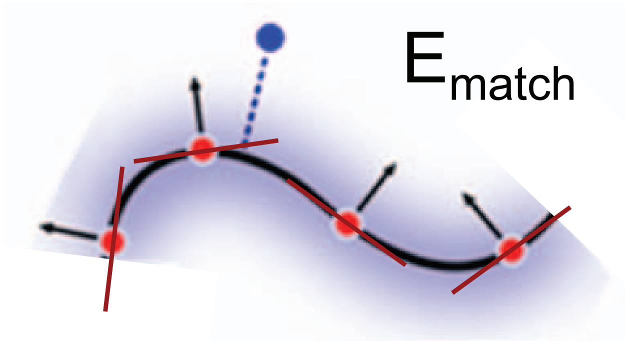


Generative Model:

- **Subsamling:** according to (known) $p_{sampl}^{(i)}$
- **Noise:** according to (known) $p_{noise}(x_1,\ldots,x_m)$
(currently assuming independent, Gaussian noise)

Data Term – E(D|S)

Data fitting term:



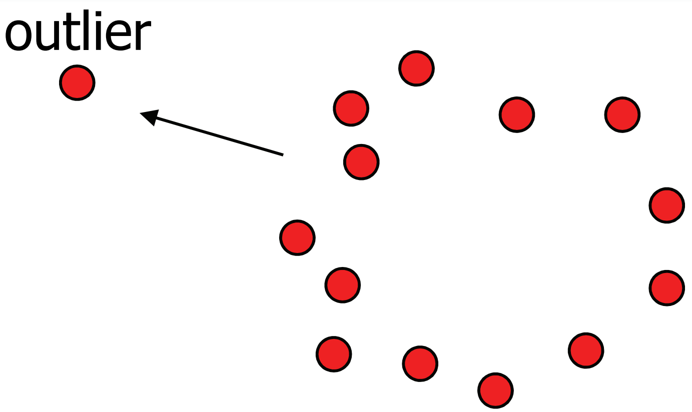
- Surface should be close to data
- Truncated squared distance function

$$E_{match}(D,S) = \sum_{data\,pts} trunc_{\delta}(dist(S, \mathbf{d}_i)^2)$$

- Sum of distances² of data points to surfel planes
- Point-to-plane: No exact 1:1 match necessary
- Truncation (M-estimator): Robustness to outliers

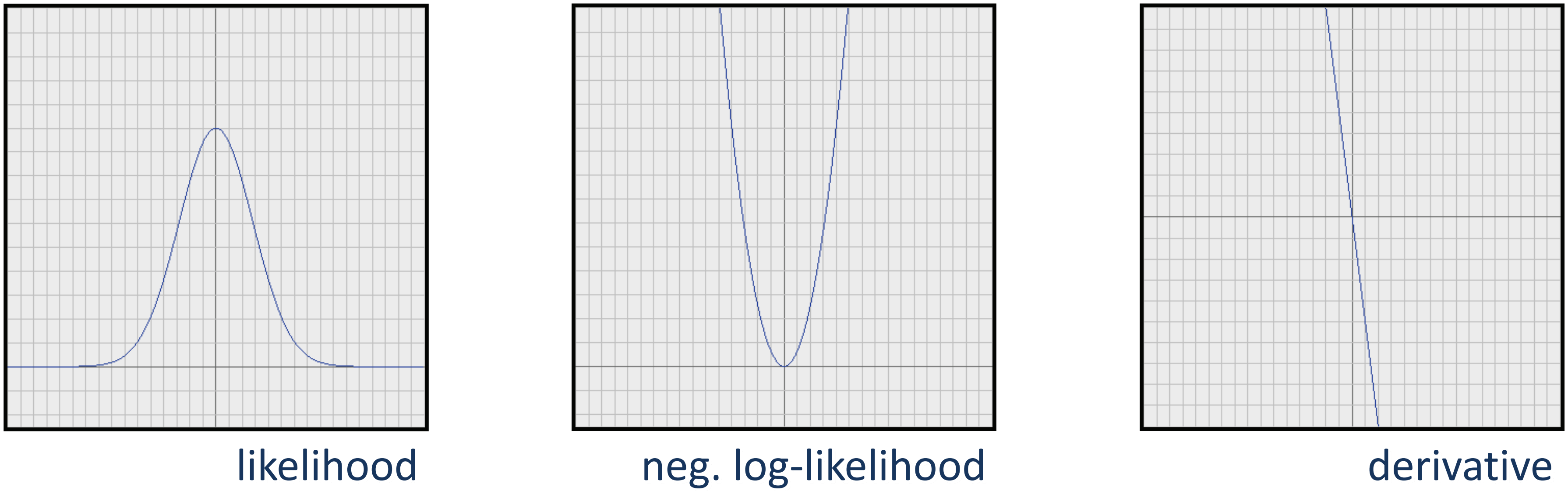
[Pottmann et al. 03]

Robustness



Problem:

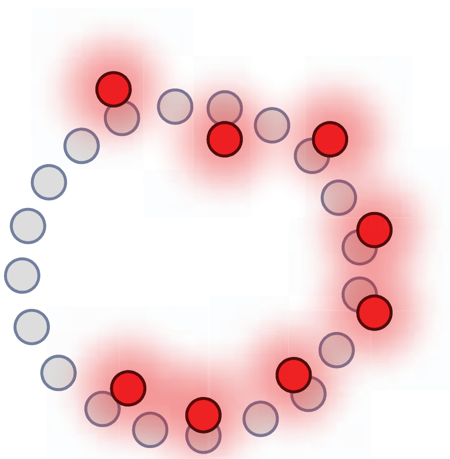
- Quadratic potential – strong influence of outliers
- Solution: truncate neg. log-likelihood



Why do We Need Priors?

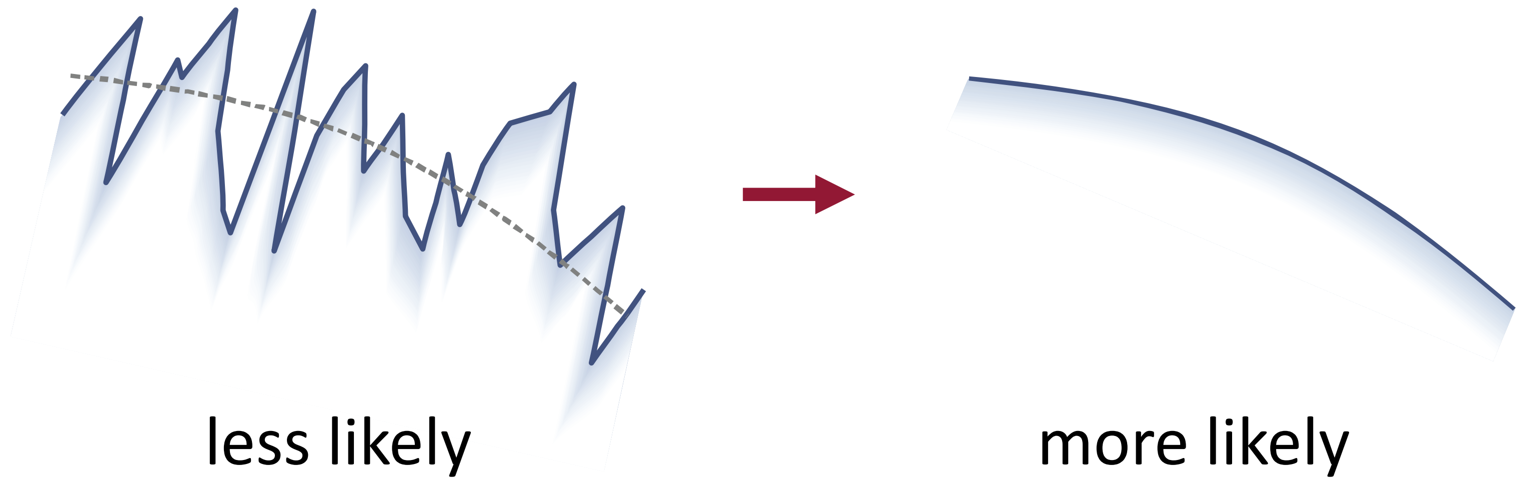
No Reconstruction without Priors

- Non-measurement points unconstrained
- For the rest: Measurement itself has largest probability density



measurement D

Priors – P(S)



Canonical assumption: smooth surfaces

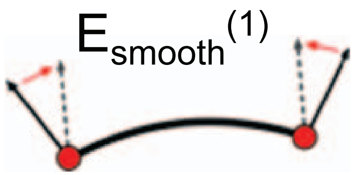
- Correlations between neighboring points

Point-based Model

Simple Smoothness Priors:

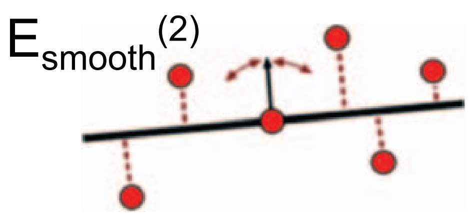
- Similar surfel normals:

$$E^{(1)}_{smooth}(S) = \sum_{surfels} \sum_{neighbors} \left(n_i - n_{i_j}\right)^2, \quad \|n_i\| = 1$$



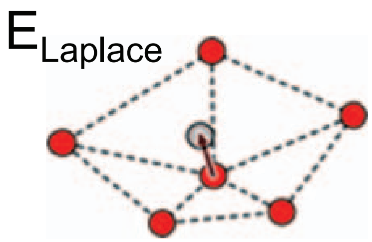
- Surfel positions – flat surface:

$$E^{(2)}_{smooth}(S) = \sum_{surfels} \sum_{neighbors} \left\langle \mathbf{s}_i - \mathbf{s}_{i_j}, \mathbf{n}(\mathbf{s}_i) \right\rangle^2$$



- Uniform density:

$$E_{Laplace}(S) = \sum_{surfels} \sum_{neighbors} \left(\mathbf{s}_i - average\right)^2$$



[c.f. Szeliski et al. 93]

Nasty Normals

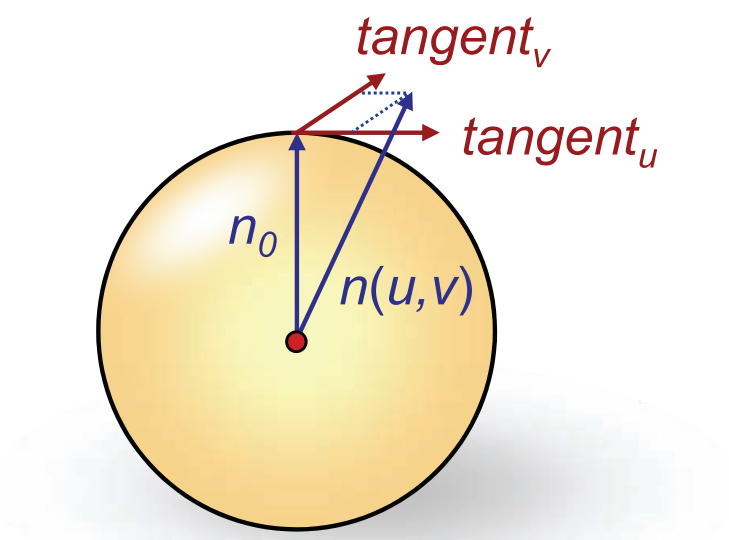
Optimizing Normals

- Problem: $E^{(1)}_{smooth}(S) = \sum_{surfels} \sum_{neighbors} \left(n_i - n_{i_j}\right)^2, \;\; s.t. \; \|n_i\| = 1$
- Need unit normals: constraint optimization
- Unconstraint: trivial solution (all zeros)

Nasty Normals

Solution: Local Parameterization

- Current normal estimate
- Tangent parameterization
- New variables *u*, *v*
- Renormalize
- Non-linear optimization
- No degeneracies



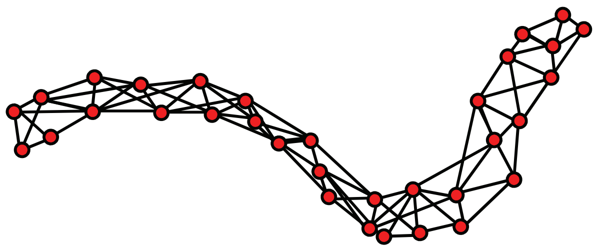
$$n(u,v) = n_0 + u \cdot tangent_u + v \cdot tangent_v$$

[Hoffer et al. 04]

Neighborhoods?

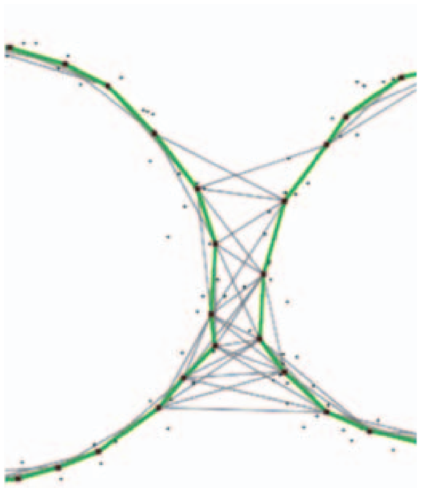
Topology estimation

- Domain of *S*, base shape (topology)
- Here, we assume this is easy to get
- In the following
  - *k*-nearest neighborhood graph
  - Typically: *k* = 6..20



Limitations

- This requires dense enough sampling
- Does not work for undersampled data



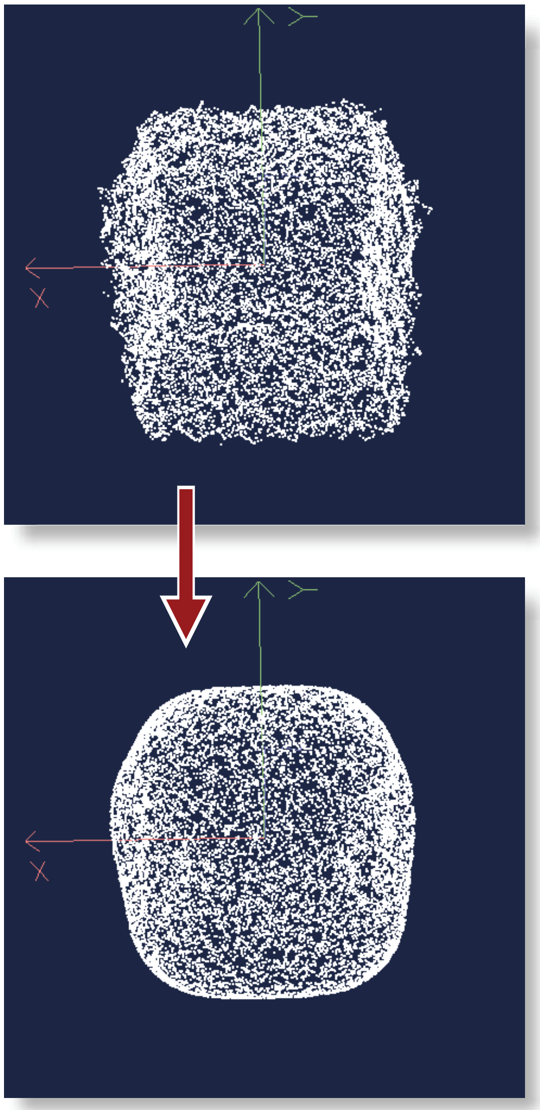
Numerical Optimization

Task:

- Compute most likely "original scene" S
- Nonlinear optimization problem

Solution:

- Create initial guess for S
  - Close to measured data
  - Use original data
- Find local optimum
  - (Conjugate) gradient descent
  - (Gauss-) Newton descent



Overall Objective Function

Overall Objective Function

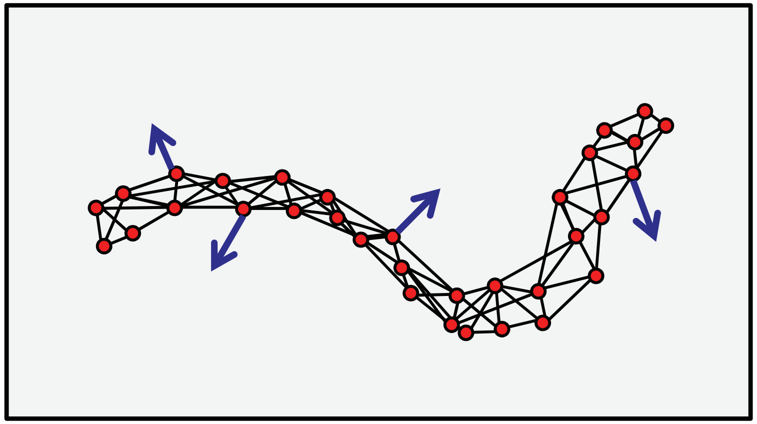
$$\underset{S\in \mathbb{R}^{3n}}{\arg\min}\, E(D,S) = E_{match}(D,S) + E^{(1)}_{smooth}(S) + E^{(2)}_{smooth}(S) + E_{Laplace}(S)$$

$$E_{match}(D,S) = \sum_{data\,pts} trunc_{\delta}(dist(S,\mathbf{d}_i)^2)$$

$$E^{(1)}_{smooth}(S) = \sum_{surfels}\sum_{neighbors}\left(n_i - n_{i_j}\right)^2,\ \|n_i\| = 1$$

$$E^{(2)}_{smooth}(S) = \sum_{surfels}\sum_{neighbors}\left\langle \mathbf{s}_i - \mathbf{s}_{i_j}, \mathbf{n}(\mathbf{s}_i)\right\rangle^2, \|n_i\| = 1$$

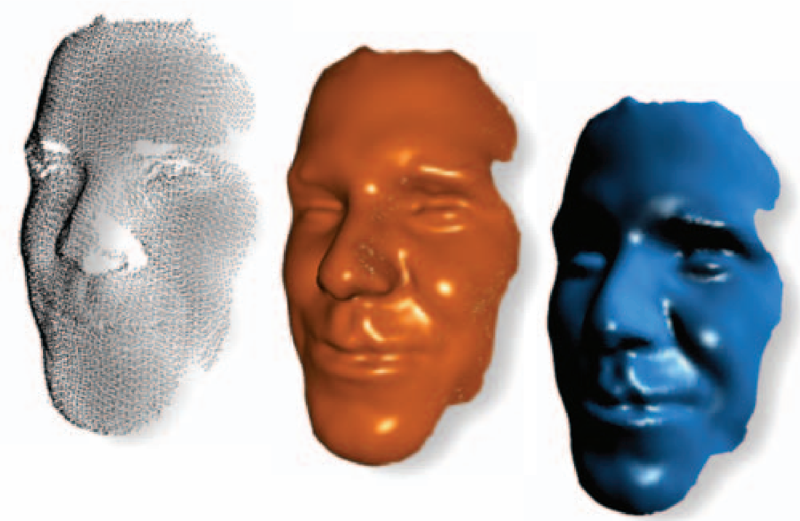
$$E_{Laplace}(S) = \sum_{surfels}\sum_{neighbors}\left(\mathbf{s}_i - average\right)^2$$



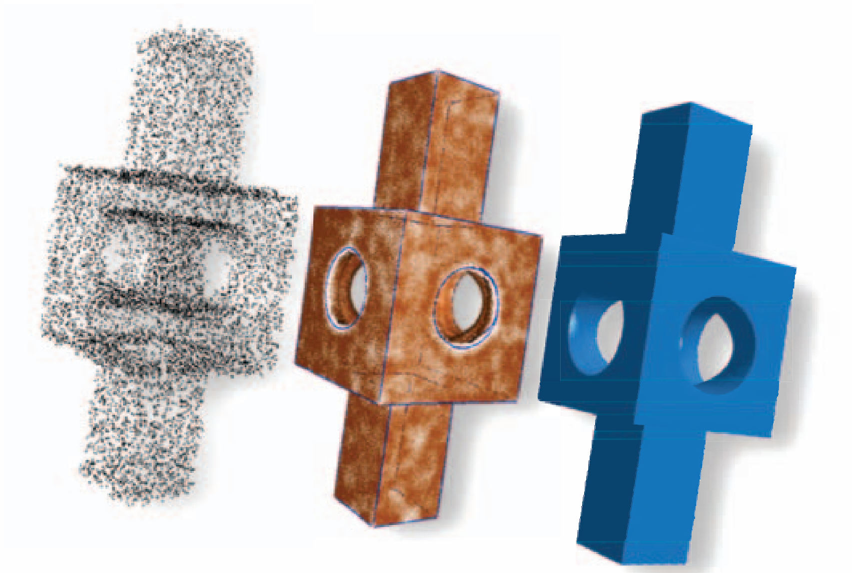
move points and normals

3D Examples

3D reconstruction results:



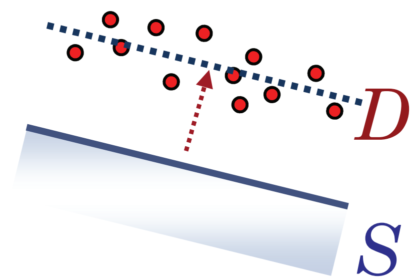
With discontinuity lines:



3D Reconstruction Summary

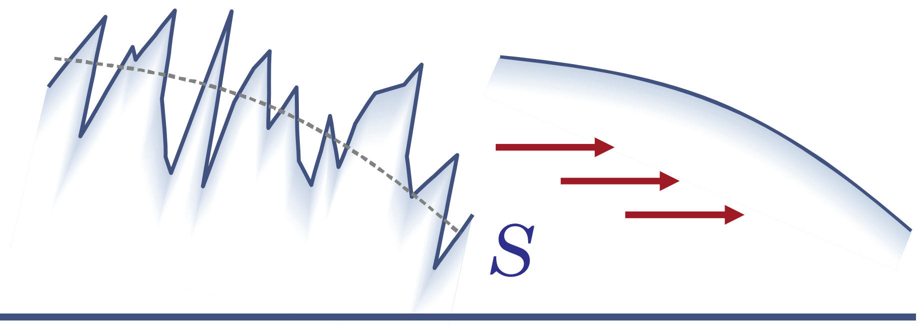
Data fitting:

$$E(D \mid S) \sim \Sigma_i \, \mathrm{dist}(S, d_i)^2$$



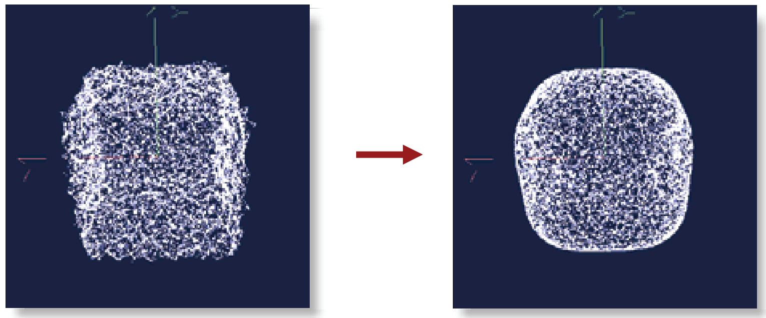
Prior: Smoothness

$$E_s(S) \sim \int_S \mathrm{curv}(S)^2$$



Optimization:

Yields 3D Reconstruction



Surface Reconstruction References

Surface Reconstruction References

- **Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D. & Silva, C. T. (2003).** Computing and Rendering Point Set Surfaces, *IEEE Transactions on Visualization and Computer Graphics 9(1), 3–15*.
- **Alliez, P., Cohen-Steiner, D., Tong, Y., Desbrun, M. (2007):** Voronoi-based Variational Reconstruction of Unoriented Point Sets. *Symposium on Geometry Processing 07, pp. 39–48*.
- **Amenta, N., Bern, M. & Kamvysselis, M. (1998).** A new Voronoi-based surface reconstruction algorithm, *SIGGRAPH 98, pp. 415–421.*
- **Amenta, N., Choi, S. & Kolluri, R. K. (2001).** The power crust, SMA 01, pp. 249–266.
- **Carr, J. C., Beatson, R. K., Cherrie, J. B., Mitchell, T. J., Fright, W. R., McCallum, B. C., Evans, T. R. (2001):** Reconstruction and representation of 3D objects with radial basis functions, *SIGGRAPH 2001, pp. 67–76*.
- **Curless, B. (1997).** New Methods for Surface Reconstruction from Range Images, *PhD thesis, Stanford University*.
- **Desbrun, M., Meyer, M., Schroeder, P., Barr, A. H. (1999).** Implicit fairing of irregular meshes using diffusion and curvature flow. *SIGGRAPH 99, pp. 317–324*.
- **Desbrun, M., Meyer, M., Schroeder, P., Barr, A. H. (2000).** Anisotropic Feature-Preserving Denoising of Height Fields and Images. *Graphics Interface 2000, pp. 145–152*.

Surface Reconstruction References

Surface Reconstruction References

- **Diebel, J. R., Thrun, S. & Brünig, M. (2006).** A Bayesian method for probable surface reconstruction and decimation, *ACM Transactions on Graphics 25(1), 39–59*.
- **Gal, R., Shamir, A., Hassner, T., Pauly, M., Cohen-Or, D. (2007).** Surface Reconstruction using Local Shape Priors. *Symposium on Geometry Processing 07*.
- **Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W. (1992).** Surface reconstruction from unorganized points, *SIGGRAPH 92, pp. 71–78*.
- **Kolluri, R. Shewchuk, J.R., O'Brien, J.F. (2004).** Spectral Surface Reconstruction from Noisy Point Clouds, *Symposium on Geometry Processing 04, pp. 11-21*.
- **Kolluri R. (2005).** Provably good moving least squares. *Proc. Symp. Discrete Algorithms 05*.
- **Lang, J. & Pai, D. K. (1999).** Bayesian Estimation of Distance and Surface Normal with a Time-of-Flight Laser Rangefinder. *3DIM 1999, pp. 109–117*.
- **Levin, D. (2003).** Mesh-independent Surface Interpolation, *Advances in Computational Mathematics*.
- **Mederos, B. Amenta, N. Velho, L. & de Figueiredo, L. H. (2005).** Surface Reconstruction for Noisy Point Clouds. *Symposium on Geometry Processing 2005, pp. 53–62*.

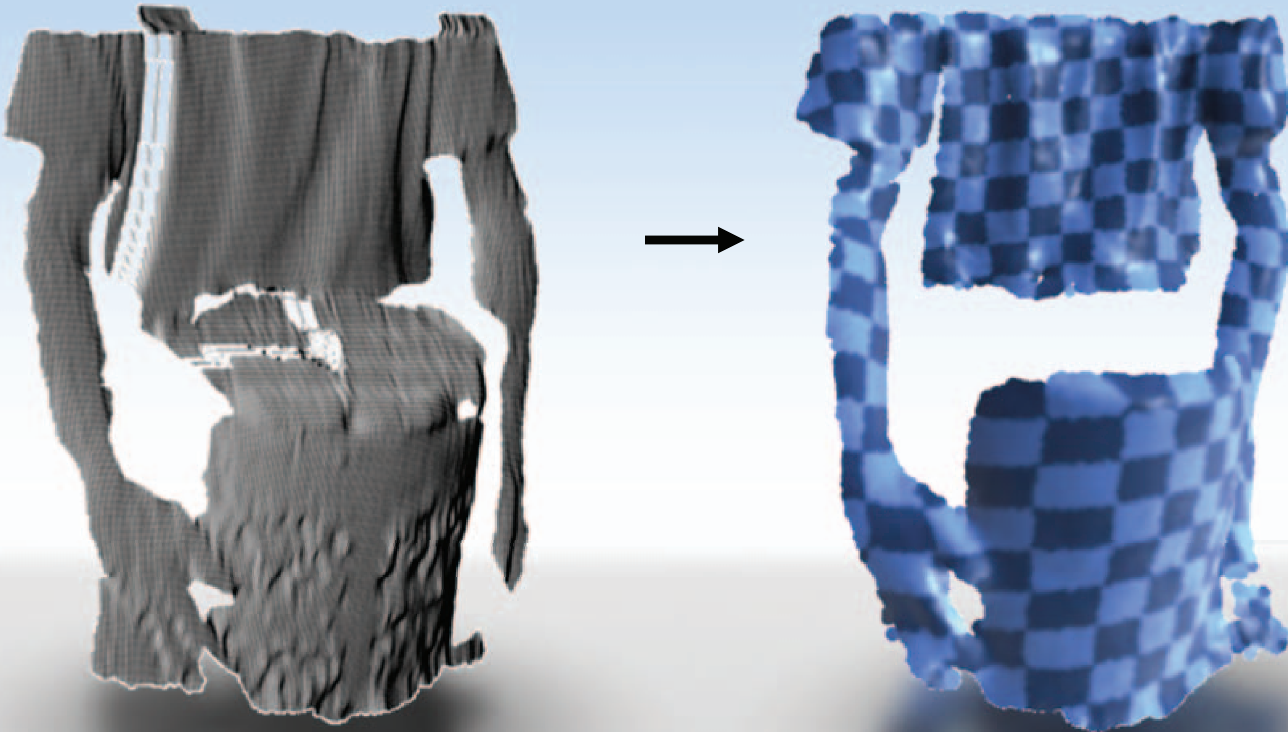
Surface Reconstruction References

Surface Reconstruction References

- **Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., Seidel, H.-P. (2003).** Multi-level partition of unity implicits, *ACM Transactions on Graphics 22(3), 463–470*.
- **Pauly, M., Mitra, N. J. & Guibas, L. J. (2004).** Uncertainty and Variability in Point Cloud Surface Data, *Symp. on Point Based Graphics 04*.
- **Steinke, F., Schölkopf, B. & Blanz, V. (2005).** Support Vector Machines for 3D Shape Processing, *Eurographics 05*.
- **Taubin, G. (1995).** A signal processing approach to fair surface design. *SIGGRAPH 95, pp. 351–358*.
- **Turk, G. & O’Brien, J. F. (1999).** Shape transformation using variational implicit functions, *SIGGRAPH 99, pp. 335–342*.

Presented in detail in this tutorial:

- **Jenke, P., Wand, M., Bokeloh, M., Schilling, A., Strasser, W. (2006).** Bayesian Point Cloud Reconstruction, *Eurographics 2006*.
- **Huang, Q.-X., Adams, B., Wand, M. (2007).** Bayesian Surface Reconstruction via Iterative Scan Alignment to an Optimized Prototype. *Symp. Geometry Processing 07*.



Animation Reconstruction

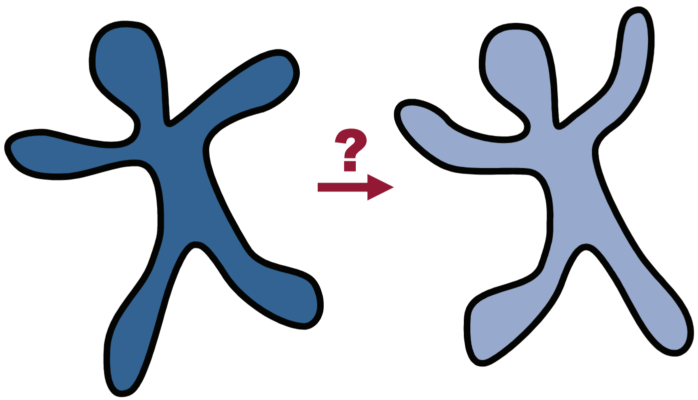
[data set: P. Phong, Stanford University]

Extension to Animations

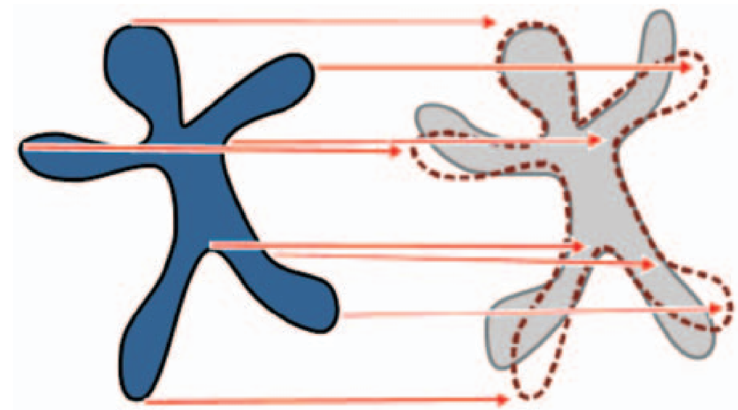
Animation Reconstruction

- Not just a 4D version
  - Moving geometry,
    not just some hypersurface
- Key component: correspondences
  - Latent variables (no direct measurement)
  - Inferred by *motion priors*
- Intuition for “good correspondences”:
  - Match target shape
  - Little deformation

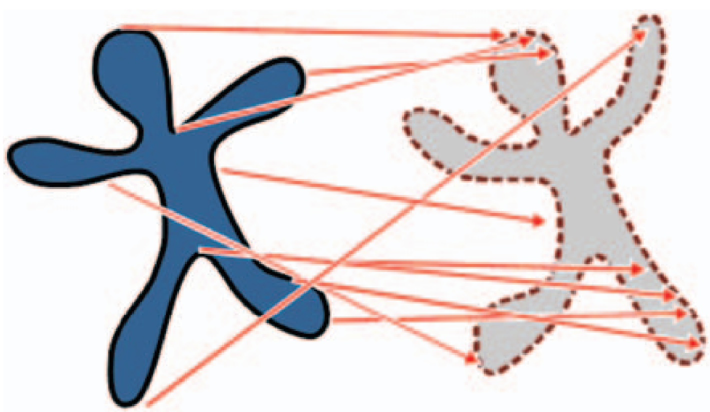
Example



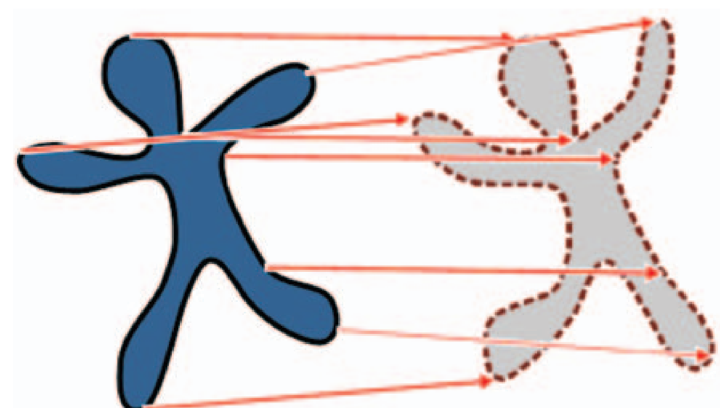
Correspondences?



✗ no shape match



✗ too much deformation

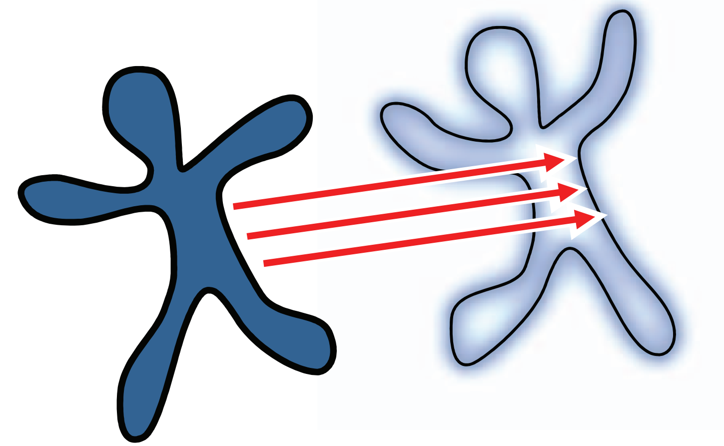


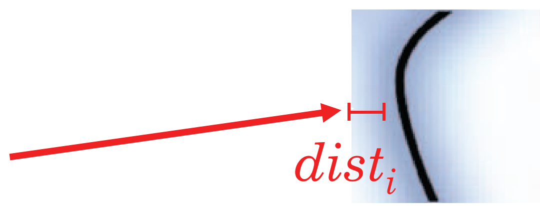
✓ optimum

Statistical Model

Model:

$$-\log p = \sum_{i=1}^{n} \left[dist_i^{\,2} + rigid_{N(i)}^{\;2} \right]$$





Distance:

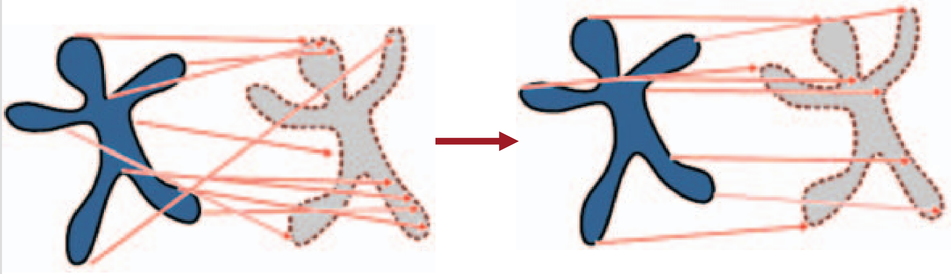
Deformation / rigidity:

Animation Reconstruction

Two additional priors:

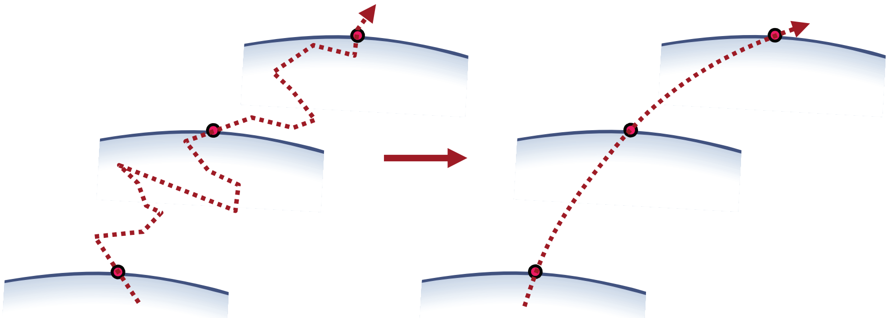
Deformation

$$E_d(S) \sim \int_S \mathrm{deform}(S_t\,,\, S_{t+1})^2$$

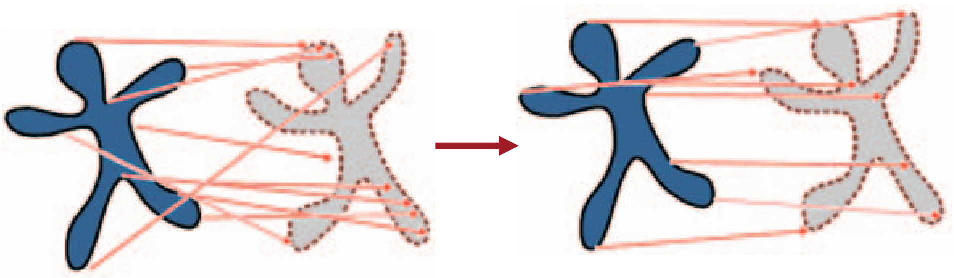


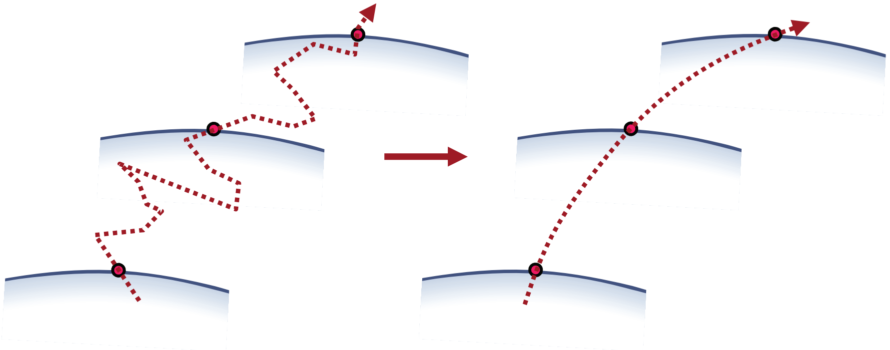
Acceleration

$$E_a(S) \sim \int_{S,t} \ddot{s}(x,\, t)^2$$



Animation Reconstruction





Not just smooth 4D reconstruction!

- Minimize
  - Deformation
  - Acceleration
- This is quite different from smoothness of a 4D hypersurface.

Animations

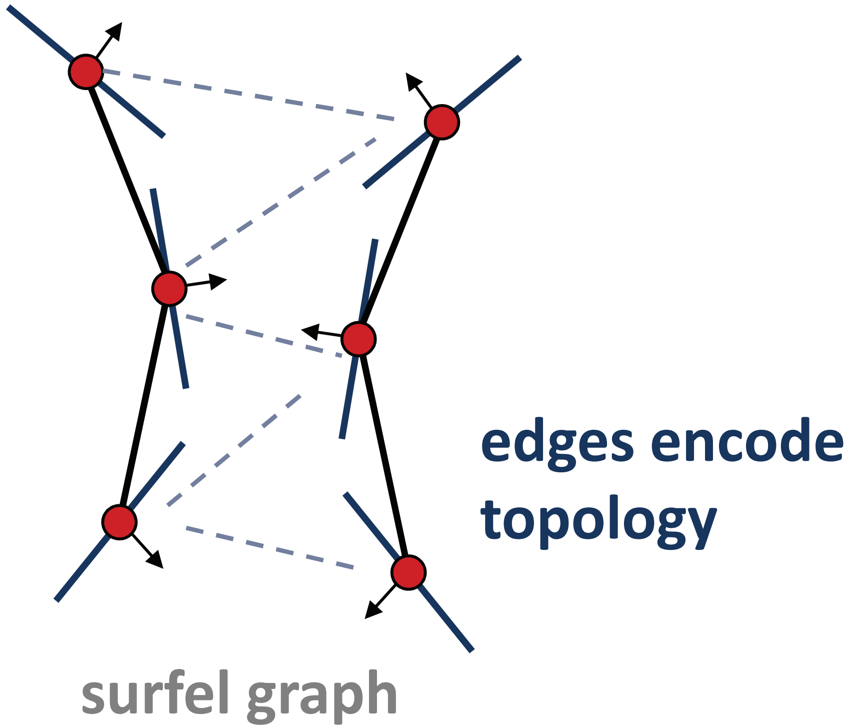
Refined parametrization of reconstruction *S*

- Surfel graph (3D)
- Trajectory graph (4D)

Discretization

Refined parametrization of reconstruction *S*

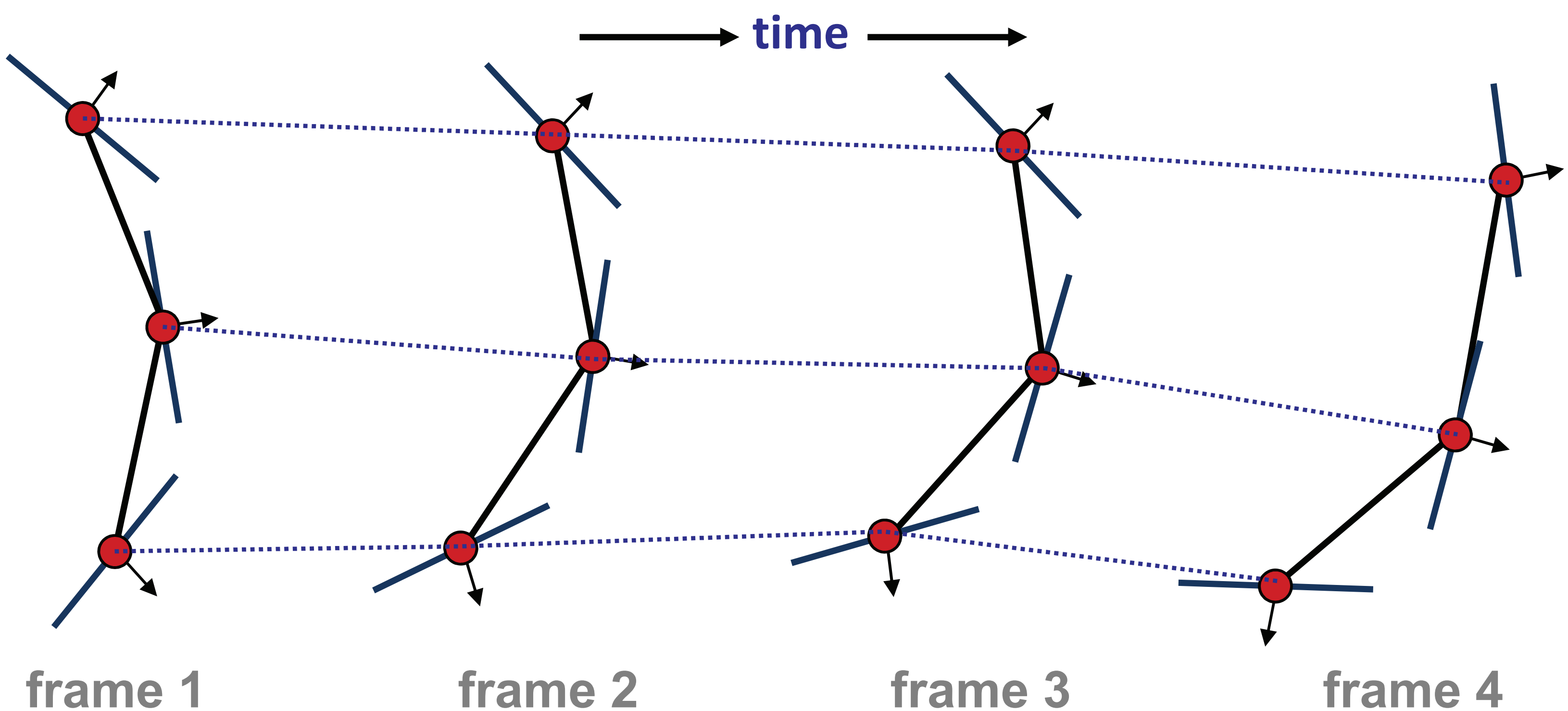
- Surfel graph (3D)
- Trajectory graph (4D)



Discretization

Refined parametrization of reconstruction *S*

- Surfel graph (3D)
- Trajectory graph (4D)



Missing Details...

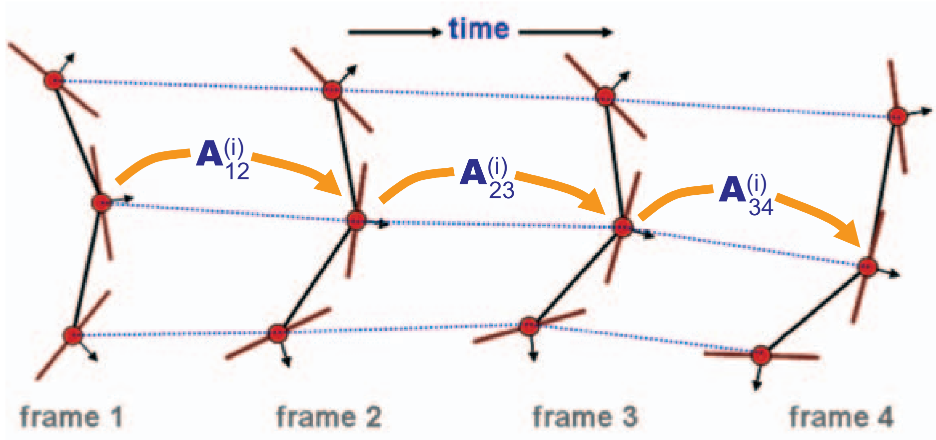
How to implement...

- The deformation priors?
  - How to actually quantify deformation?
  - This is somewhat difficult!
- Acceleration priors?
  - This is rather simple...

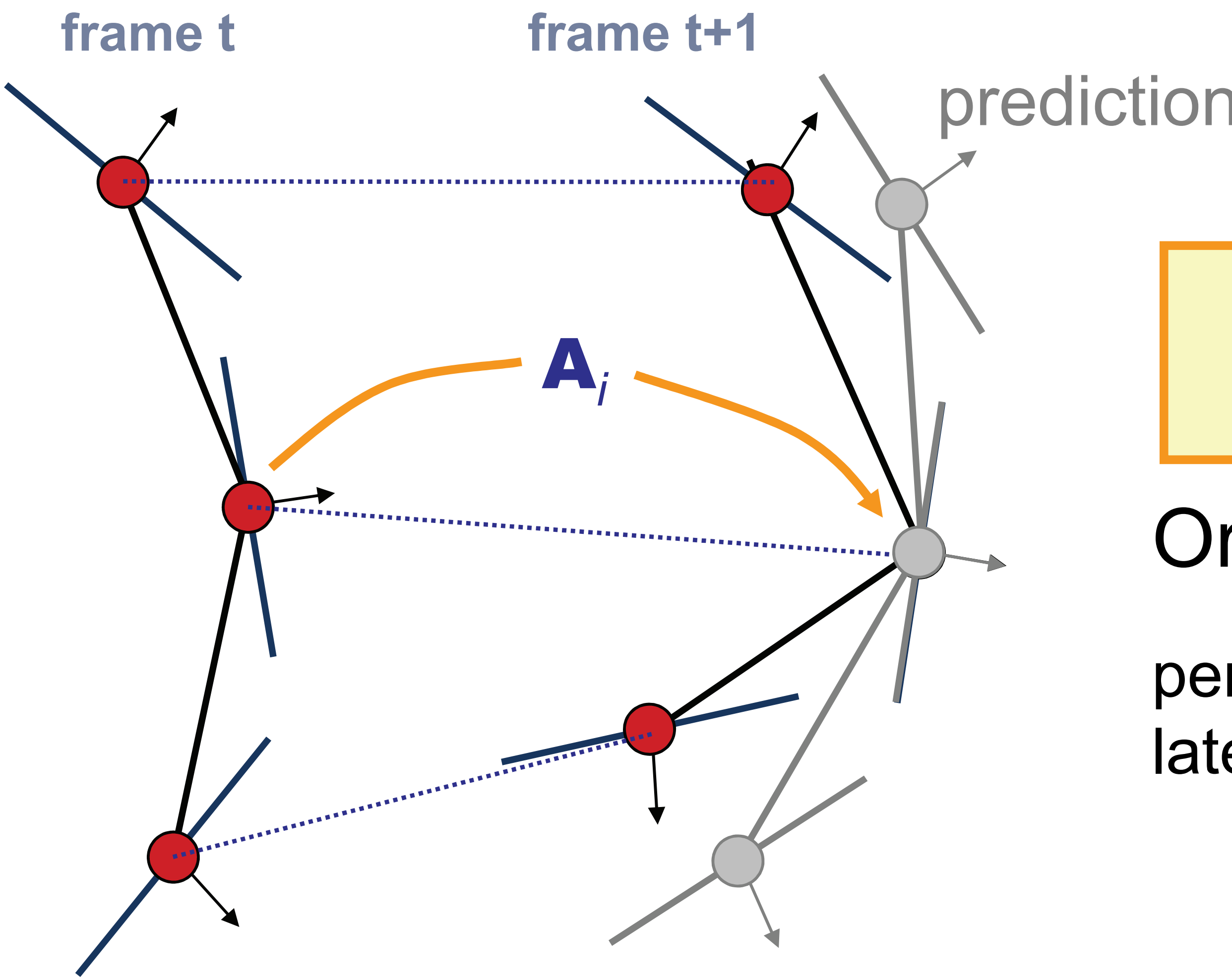
Implementation

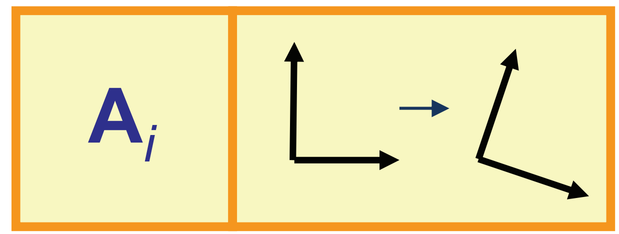
Deformation model

- Latent transformation $\mathbf{A}^{(i)}$ per surfel
- Transforms *neighborhood* of s_i
- Minimize error (both surfels and $\mathbf{A}^{(i)}$)



Deformation

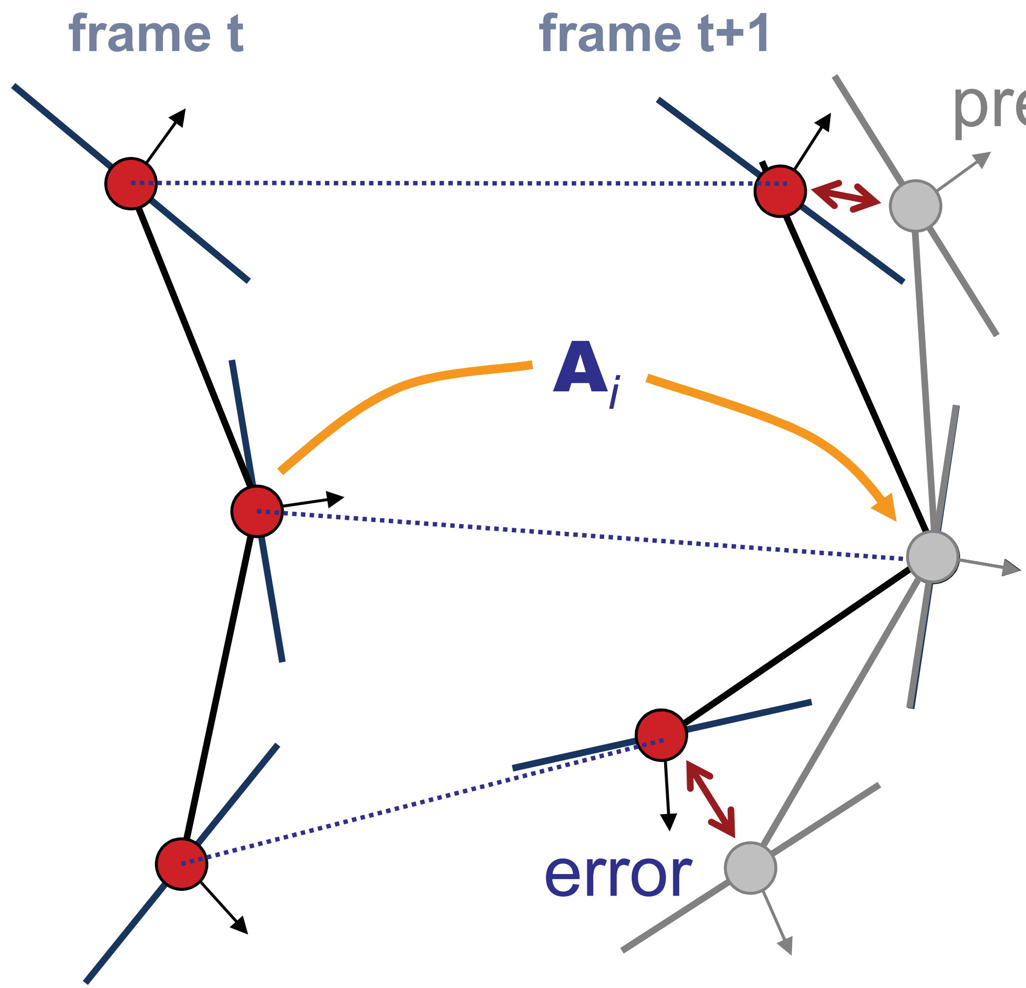


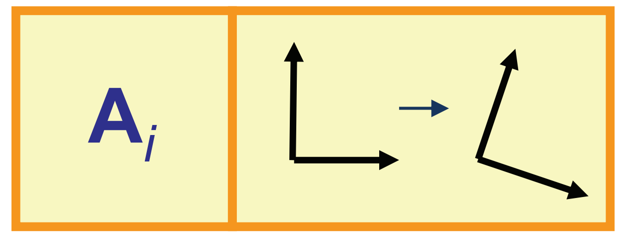


Orthonormal Matrix $\mathbf{A}_i$

per surfel (neighborhood), latent variable

Deformation





Orthonormal Matrix $\mathbf{A}_i$

per surfel (neighborhood), latent variable

$$E_{deform}(S) = \sum_{surfels} \sum_{neighbors} \left[\mathbf{A}_i^t \left(\mathbf{s}_i^{(t)} - \mathbf{s}_{i_j}^{(t)} \right) - \left(\mathbf{s}_i^{(t+1)} - \mathbf{s}_{i_j}^{(t+1)} \right) \right]^2$$

Unconstrained Optimization

Orthonormal matrices

- Local, 1st order, non-degenerate parametrization:

$$\mathbf{C}_{\times i}^{(t)} = \begin{pmatrix} 0 & \alpha & \beta \\ -\alpha & 0 & \gamma \\ -\beta & -\gamma & 0 \end{pmatrix} \qquad \begin{aligned} \mathbf{A}_i &= \mathbf{A}_0 \exp(\mathbf{C}_{\times i}) \\ &\doteq \mathbf{A}_0 (I + \mathbf{C}_{\times i}^{(t)}) \end{aligned}$$

- Optimize parameters α, β, γ, then recompute $\mathbf{A}_0$
- Compute initial estimate using [*Horn 87*]

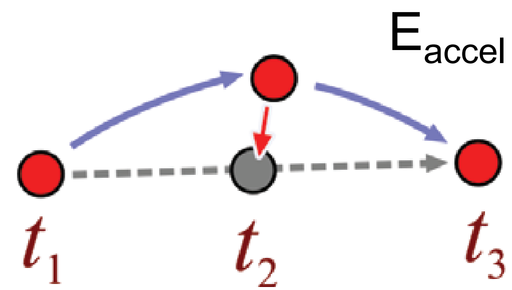
[Hoffer et al. 04]

Acceleration

Acceleration priors

- Penalize non-smooth trajectories

$$E_{accel}(A) = \left[\mathbf{s}_i^{t-1} - 2\mathbf{s}_i^{t} + \mathbf{s}_i^{t+1}\right]^2$$



- Filters out temporal noise

Optimization

For optimization, we need to know:

- The surfel graph
- A (rough) initialization close to correct solution

Optimization:

- Non-linear *continuous optimization* problem
- Gauss-Newton solver (fast & stable)

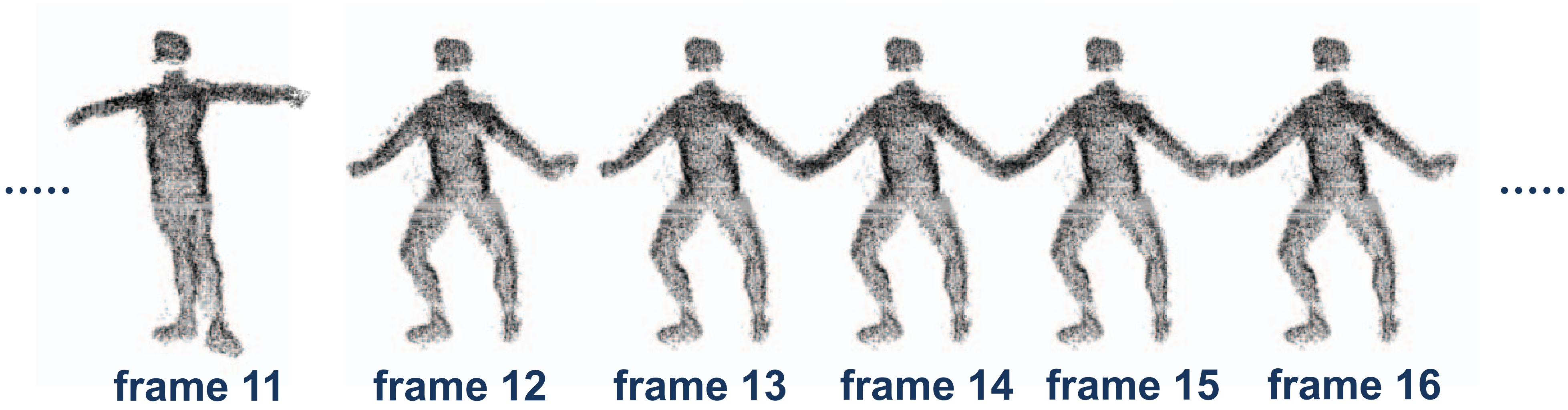
How do we get the initialization?

- *Iterative assembly* heuristic to build & init graph

Global Assembly

Assumption: Adjacent frames are similar

- Every frame is a good initialization for the next one
- Solve for frame pairs

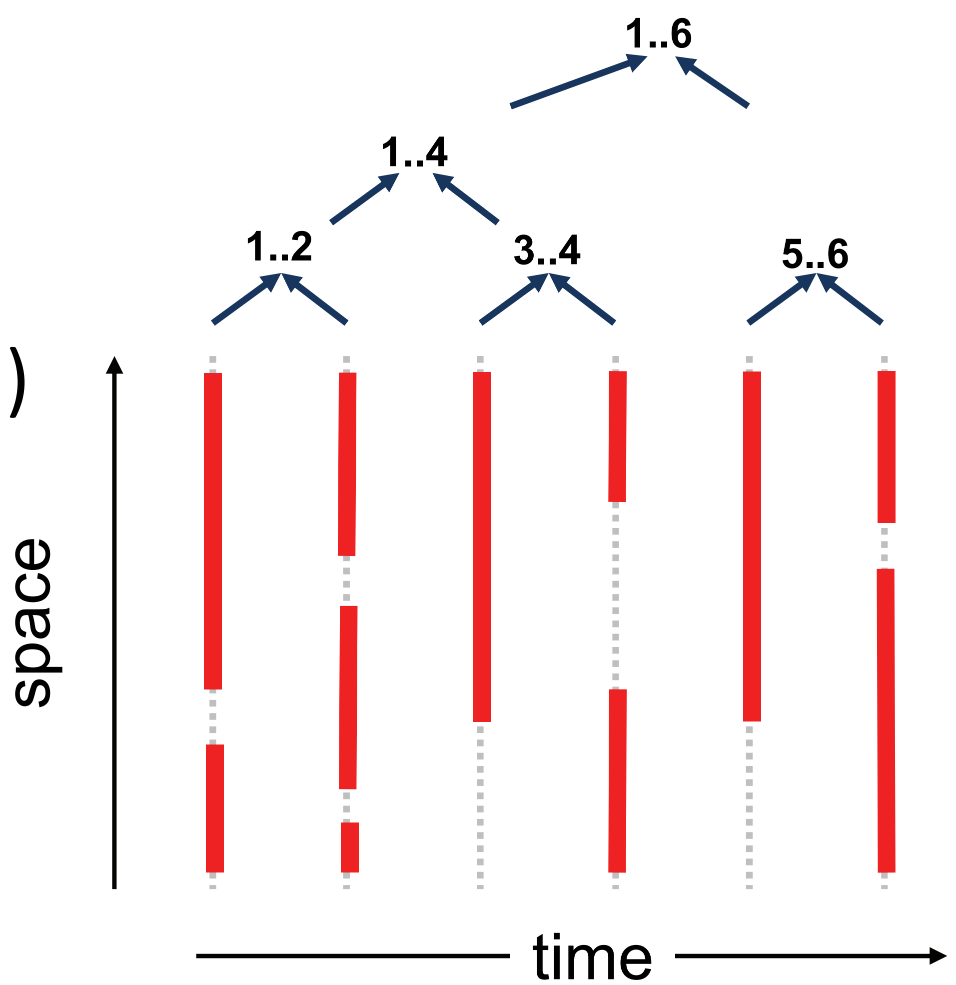


[data set courtesy of C. Theobald, MPI-Inf]

Iterative Assembly

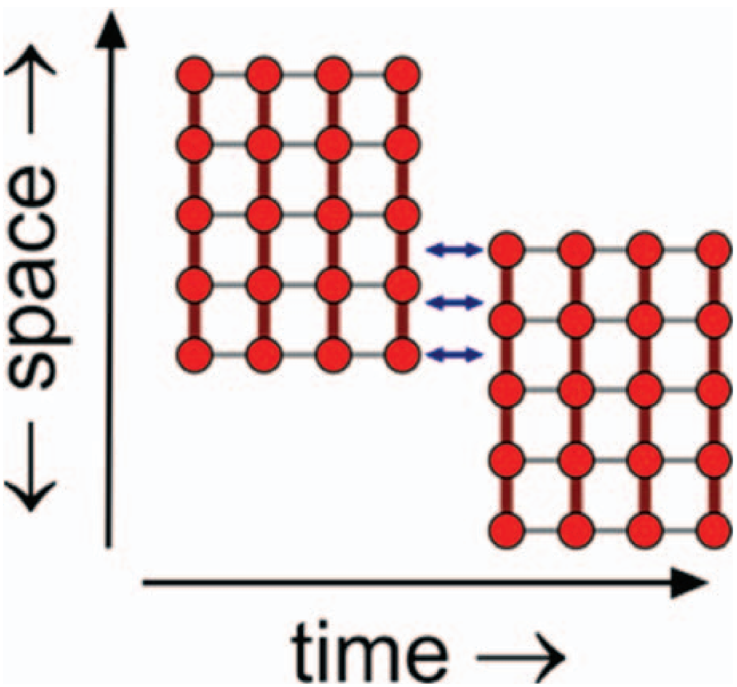
Iterative assembly

- Merge adjacent frames
- Propagate hierarchically
- Global optimization (avoid error propagation)

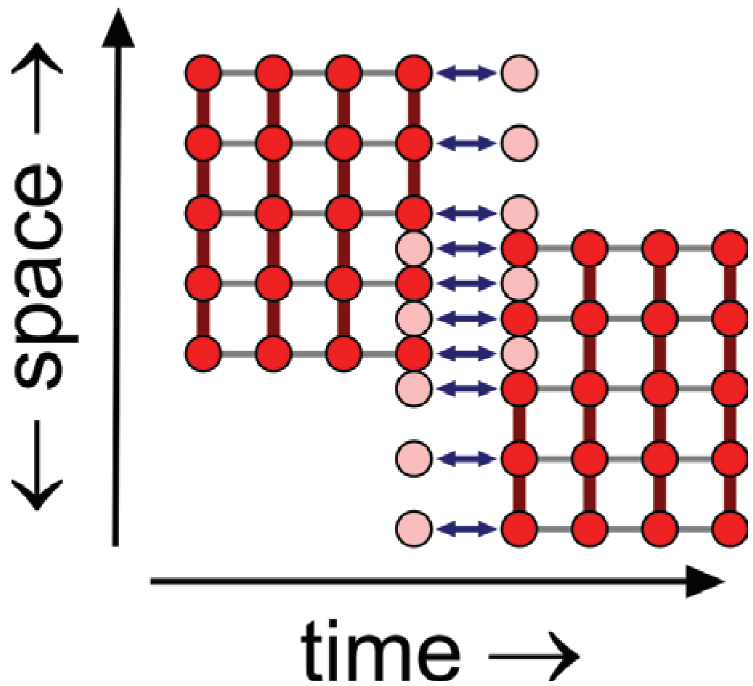


Iterative Assembly

Pairwise alignment



adjacent
trajectory sets

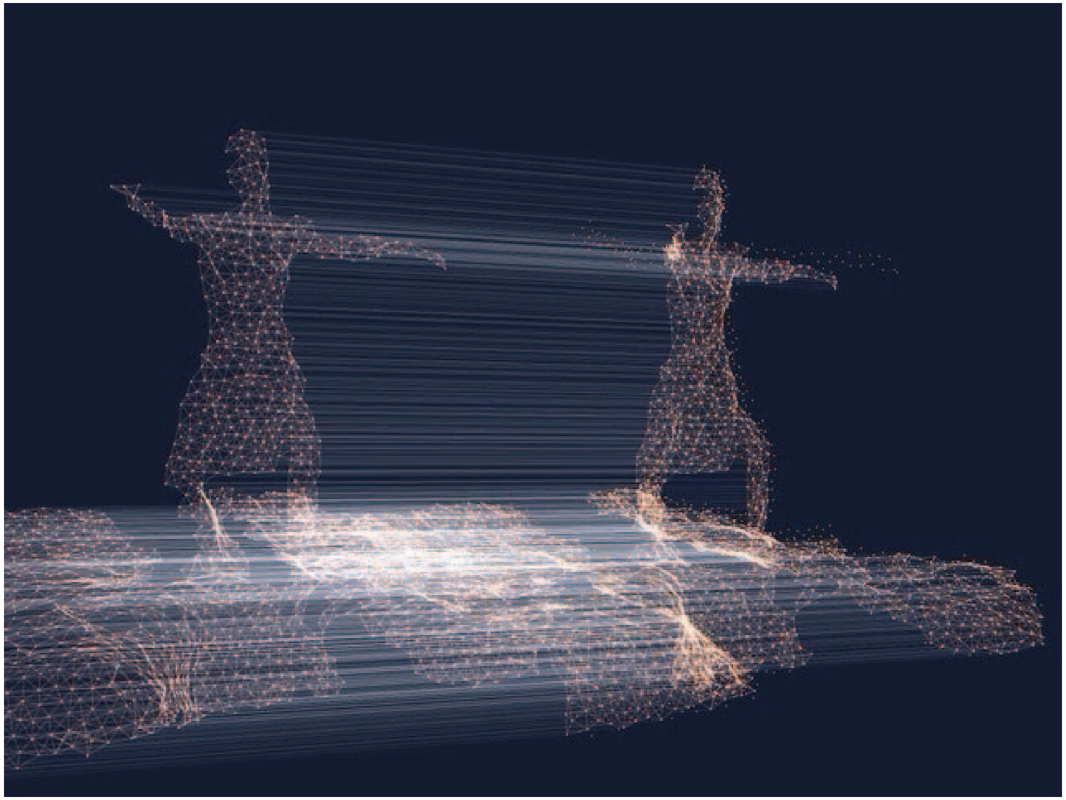


aligned
frames

Alignment

Alignment:

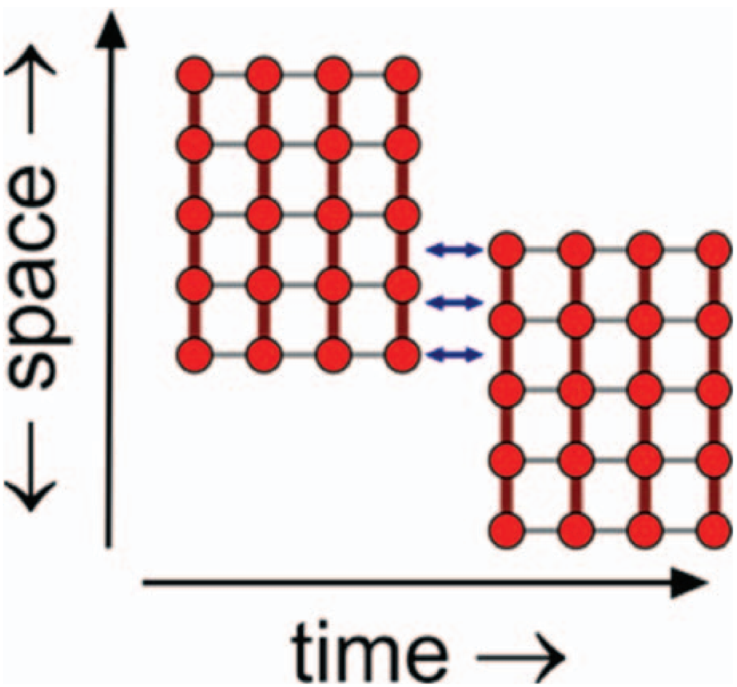
- Two frames
- Use one frame as initialization
- Second frame as “data points”
- Optimize



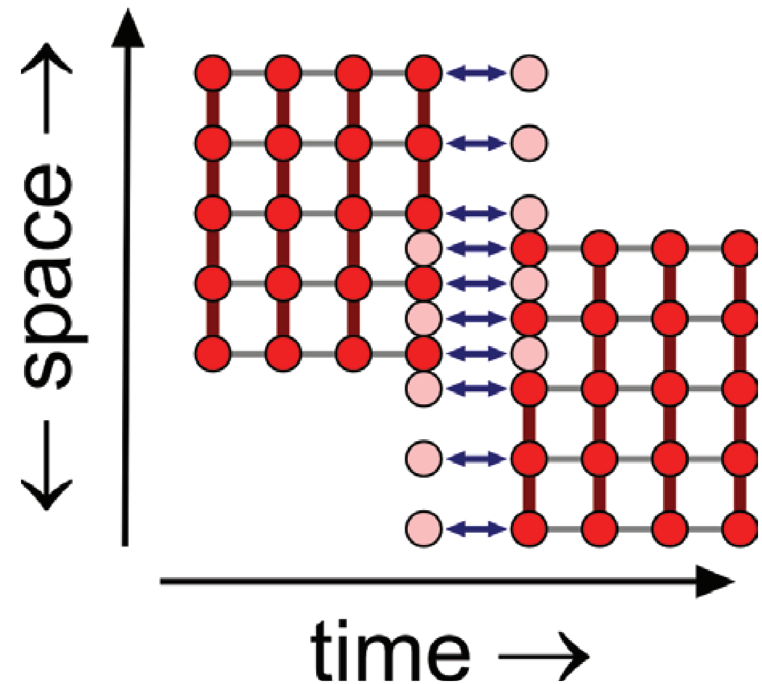
[data set: Zitnick et al., Microsoft Research]

Iterative Assembly

Pairwise alignment



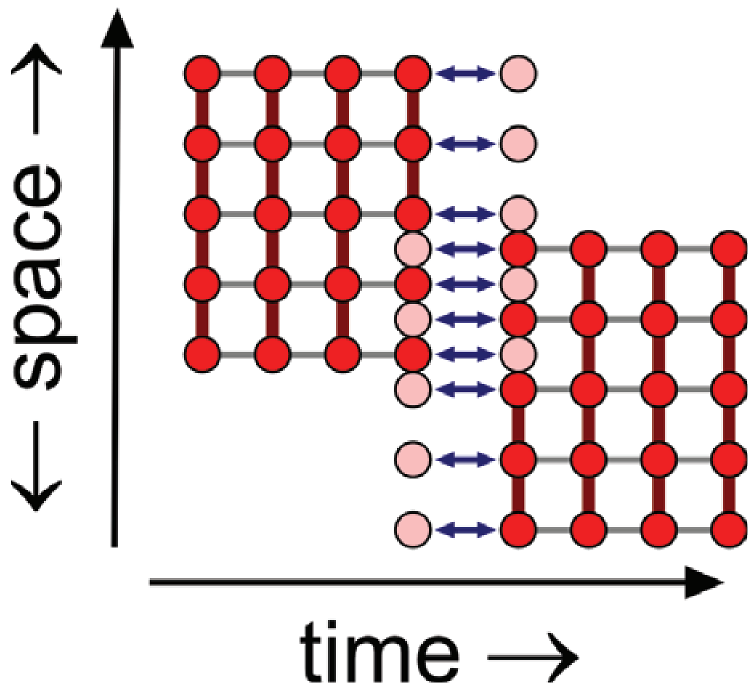
adjacent
trajectory sets



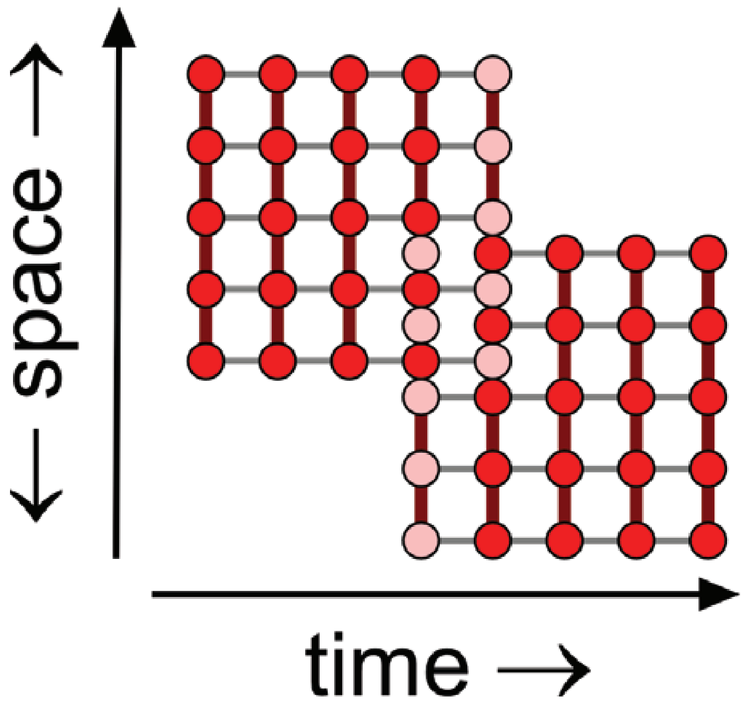
aligned
frames

Iterative Assembly

Topology stitching



aligned frames

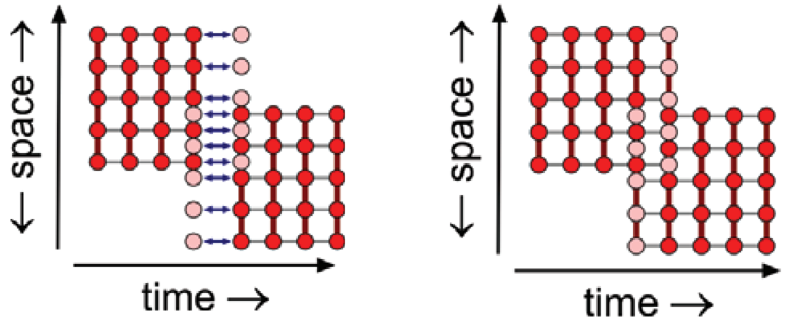


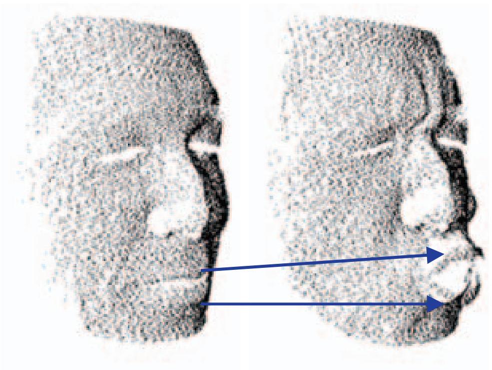
merged topology

Topology Stitching

Recompute Topology

- Recompute kNN/ε-graph
- Topology is global





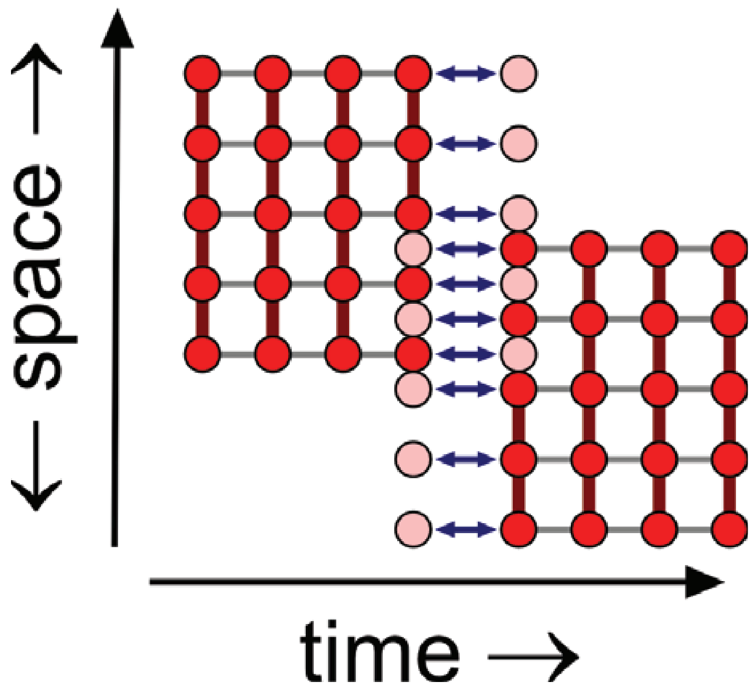
Sanity Check:

- No connection if distance changes

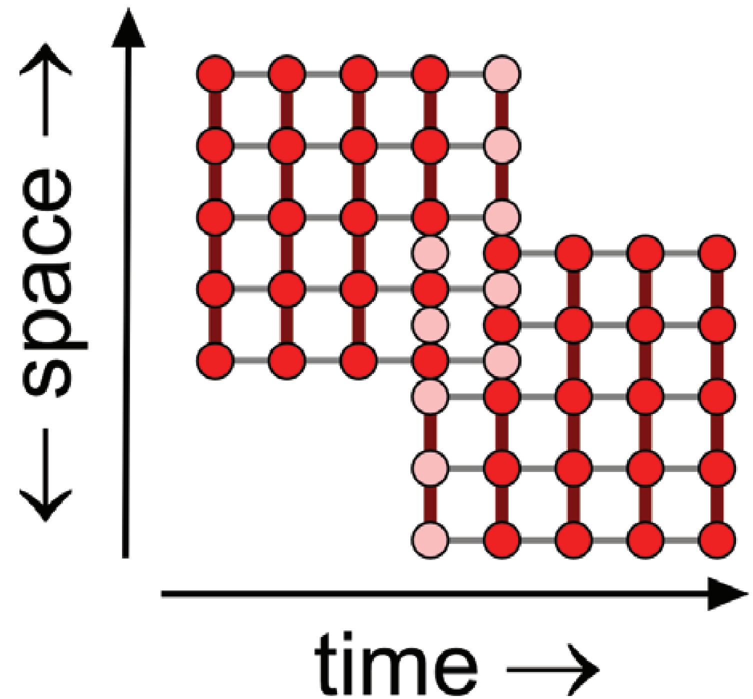
[data set courtesy of S. König, S. Gumhold, TU Dresden]

Iterative Assembly

Topology stitching



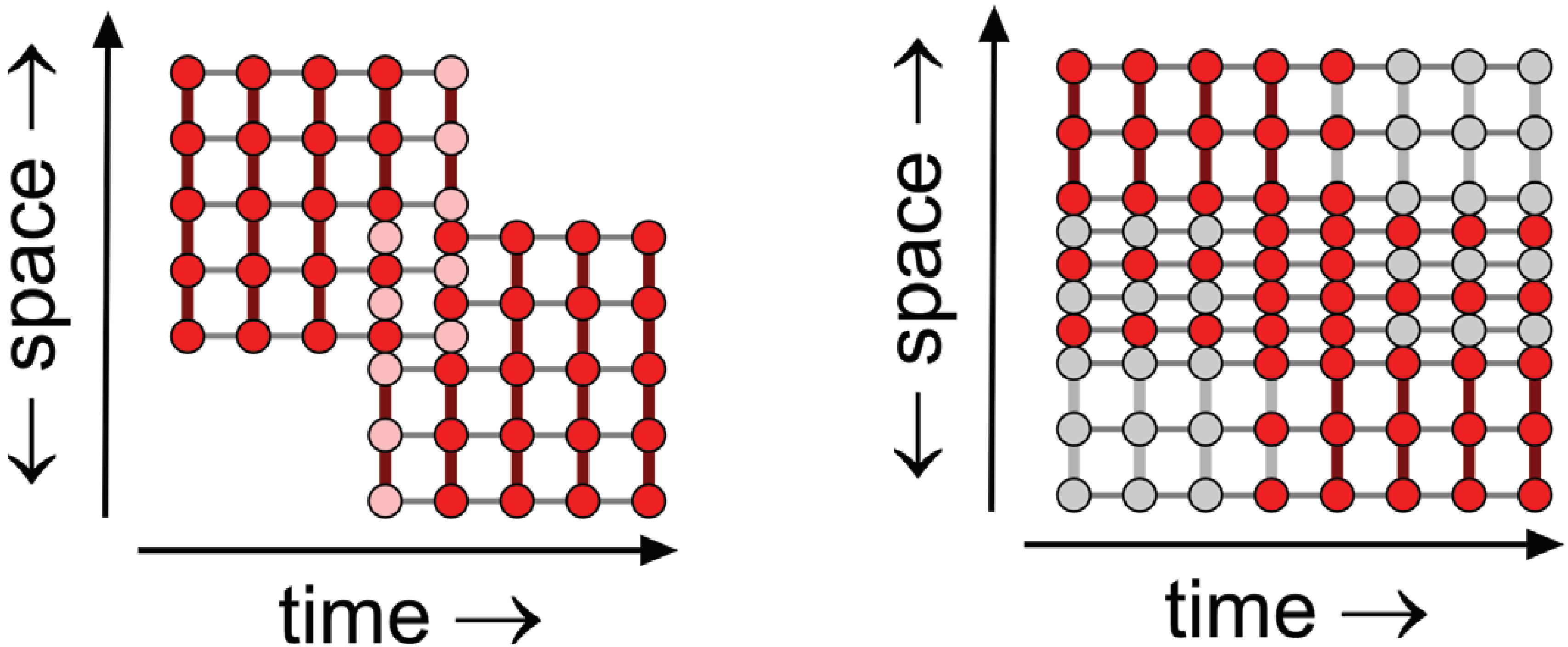
aligned frames



merged topology

Iterative Assembly

Problem: incomplete trajectories

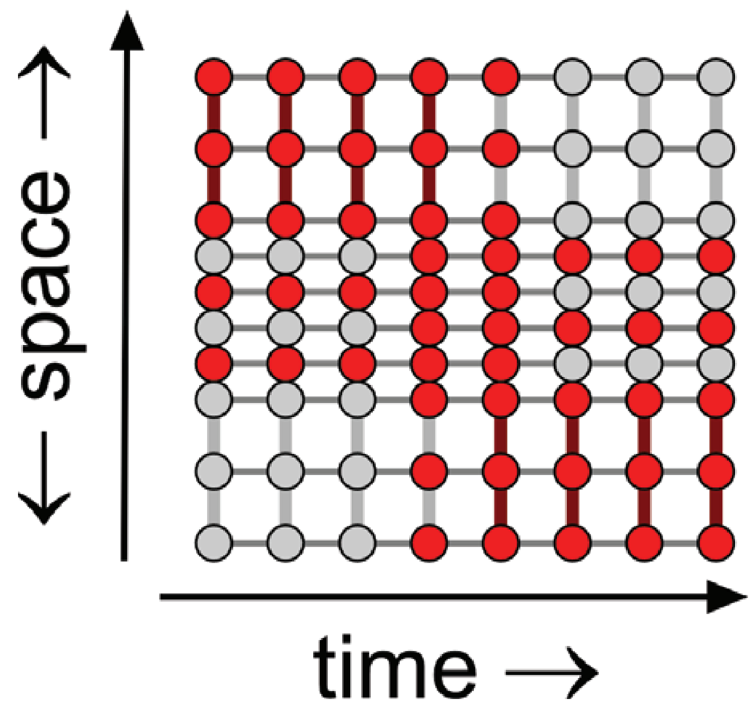


merged topology

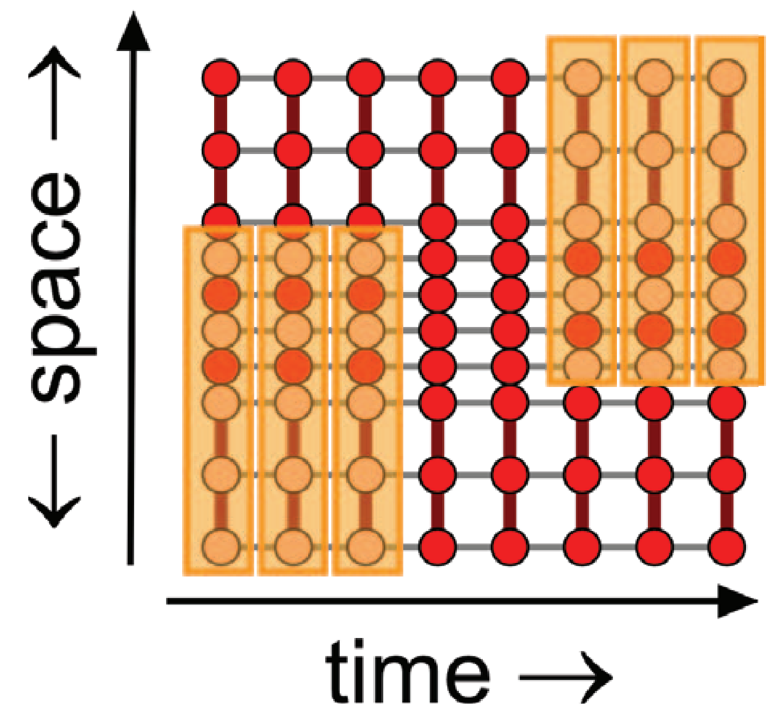
uninitialized surfels

Iterative Assembly

Hole filling



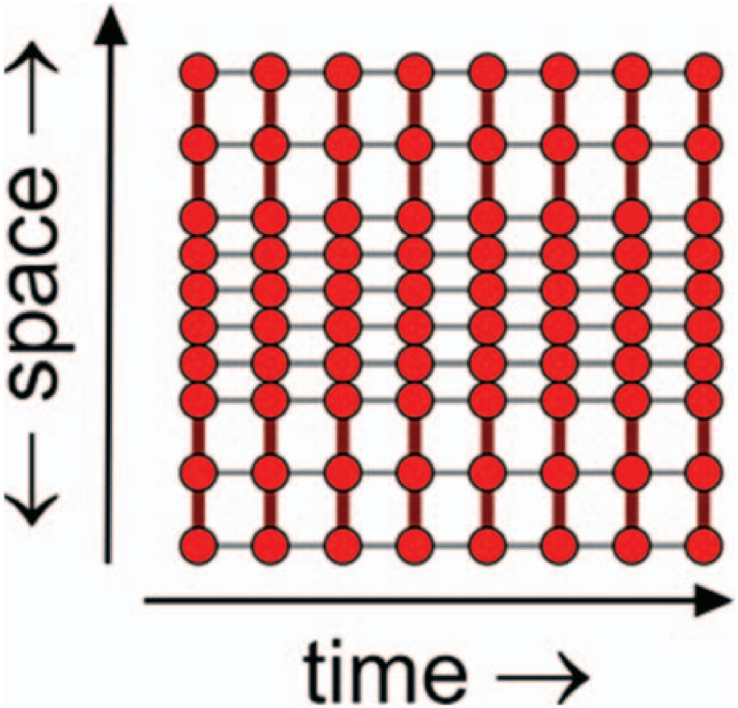
uninitialized surfels



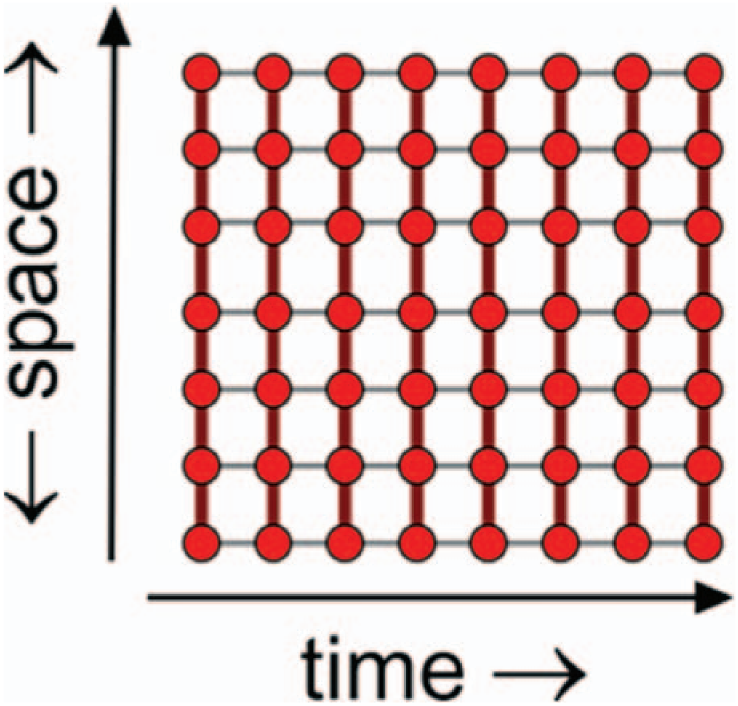
copy from neighbors, optimize

Iterative Assembly

Resampling



hole filled
result



remove dense surfels
(constant complexity)

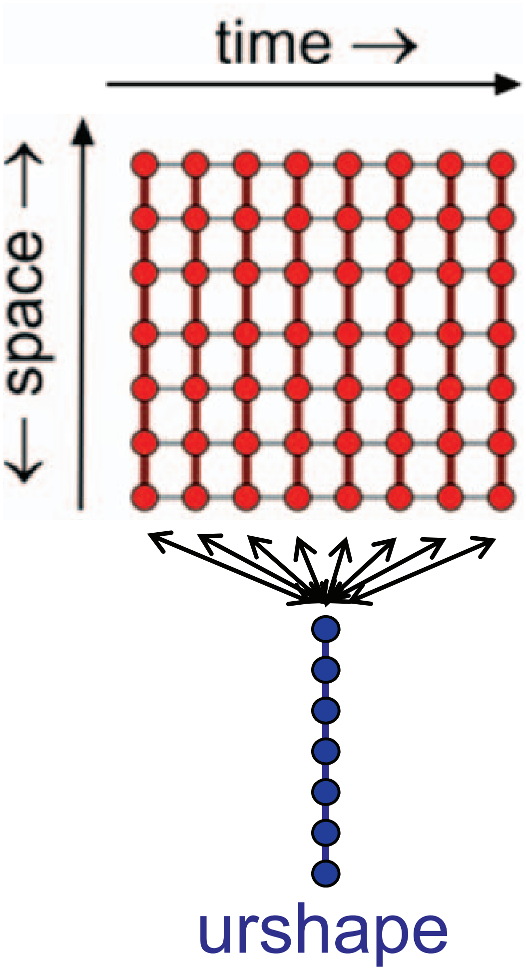
Global Optimization

Last step:

- Global optimization
- Optimize over all frames simultaneously

Improve stability: Urshapes

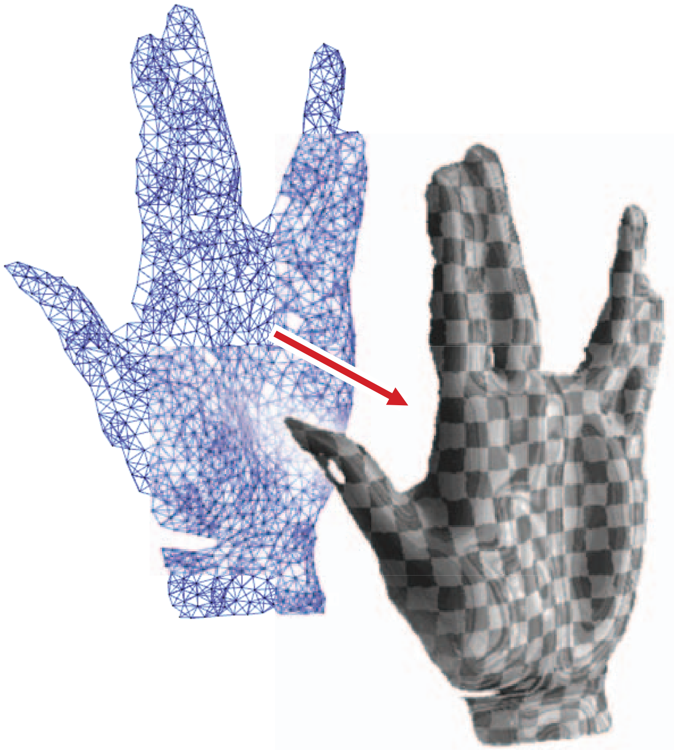
- Connect hidden “latent” frame to all other frames (deformation prior only)
- Initialize with one of the frames



Meshing

Last step: create mesh

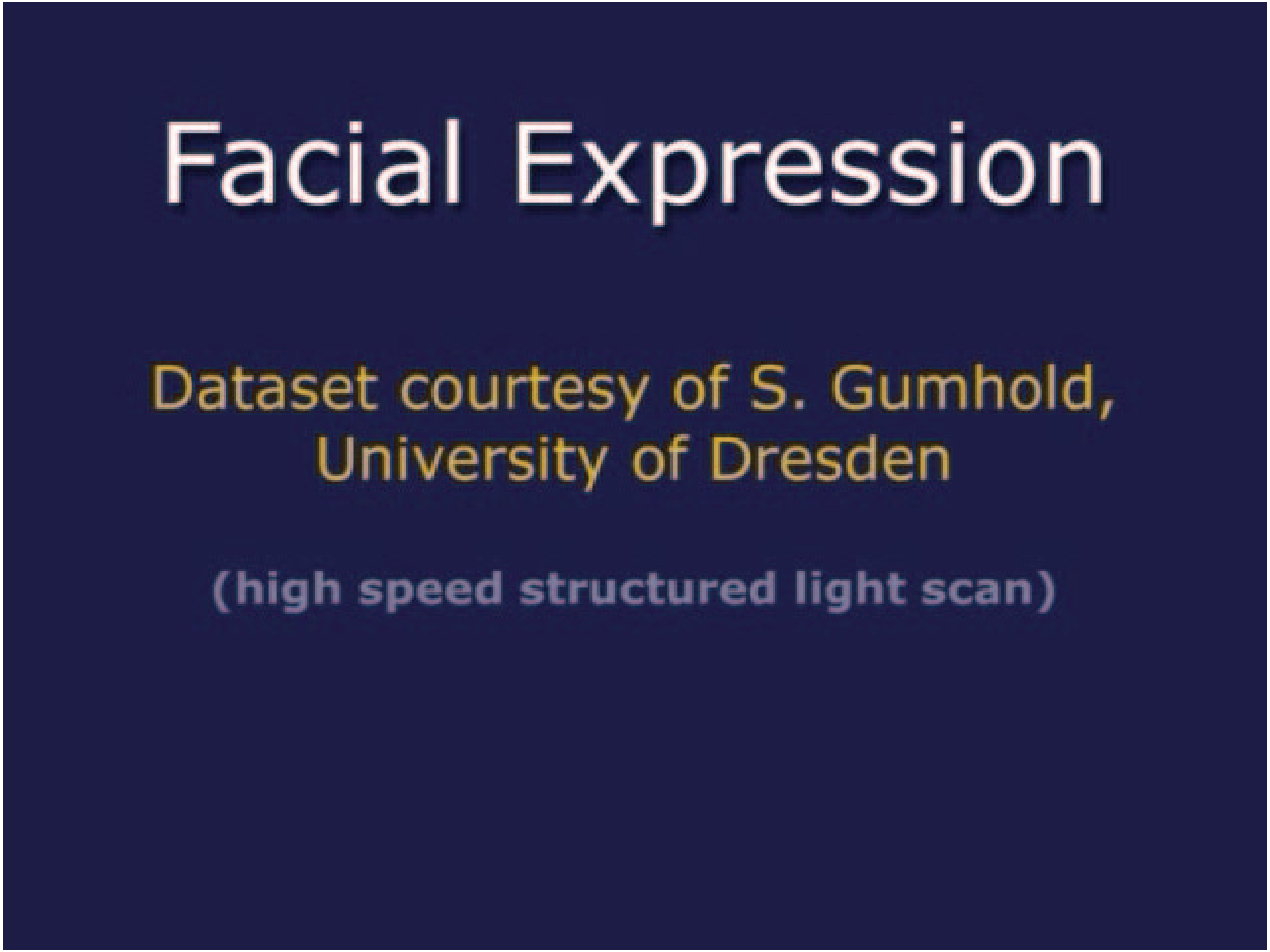
- After complete surfel graph is reconstructed
- Pick one frame (or urshape)
- "Marching cubes" meshing [*Hoppe et al. 92*, *Shen et al. 04*]
- Morph according to trajectories (local weighted sum)



[data set courtesy of O. Schall, MPI Informatik Saarbrücken]

Results

frames	surfels	data pts	preprocessing	reconstruction	
20	49,500	963,671	6 min 52 sec	4 h 25 min	[Pentium-4, 3.4GHz]



frames	surfels	data pts	preprocessing	reconstruction	
20	32,740	400,000	6 min 59 sec(*)	7 h 31 min	[Pentium-4, 3.4GHz / (*)3.0GHz]

Improved Version:

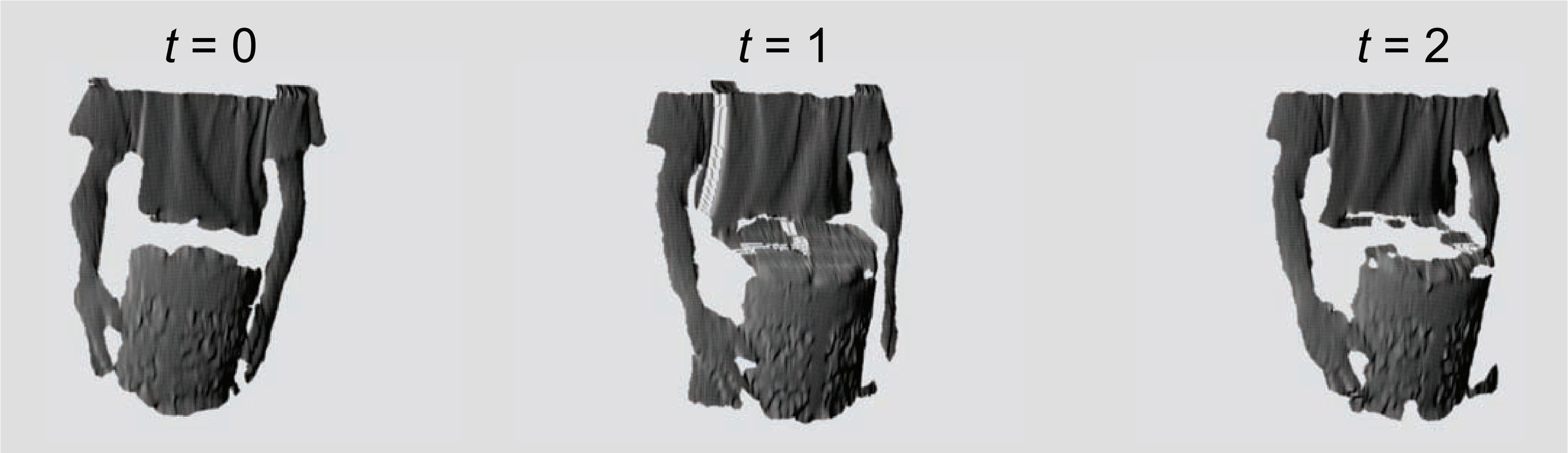
Factorization Model

Improved Version

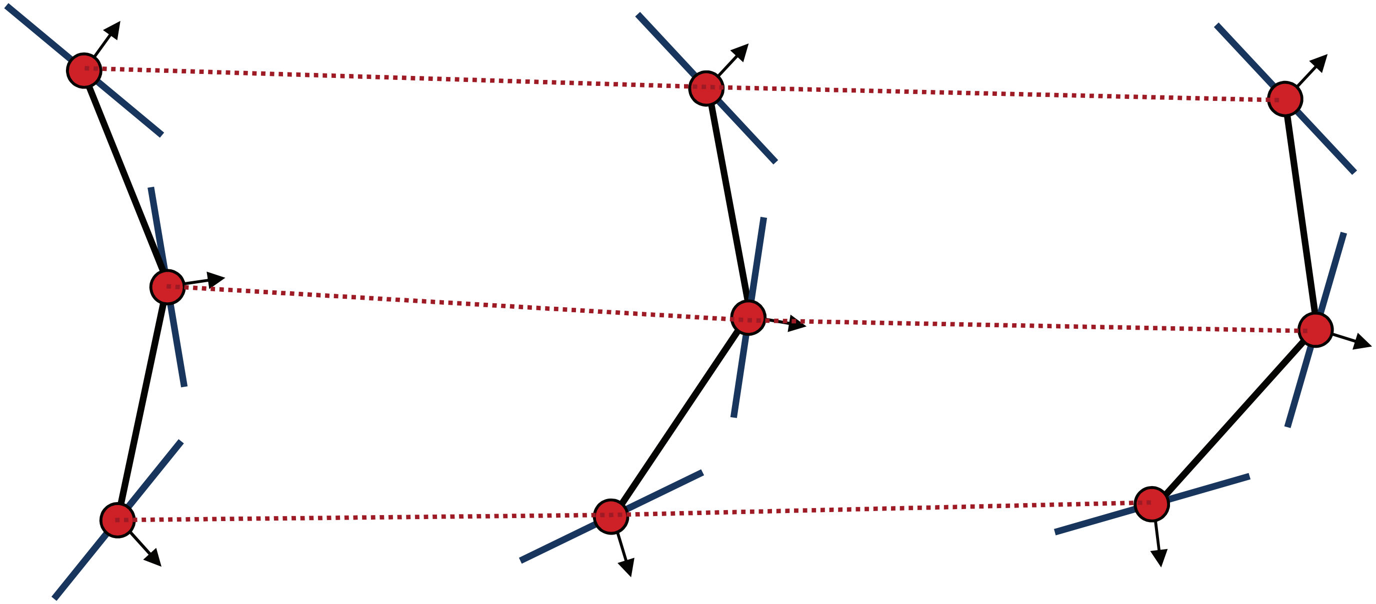
Factorization Model:

- Solving for the geometry in every frame wastes resources
- Store one urshape and a deformation field
  - High resolution geometry
  - Low resolution deformation (adaptive)
- Less memory, faster, and much more stable
- Streaming computation (constant working set)

We have so far...

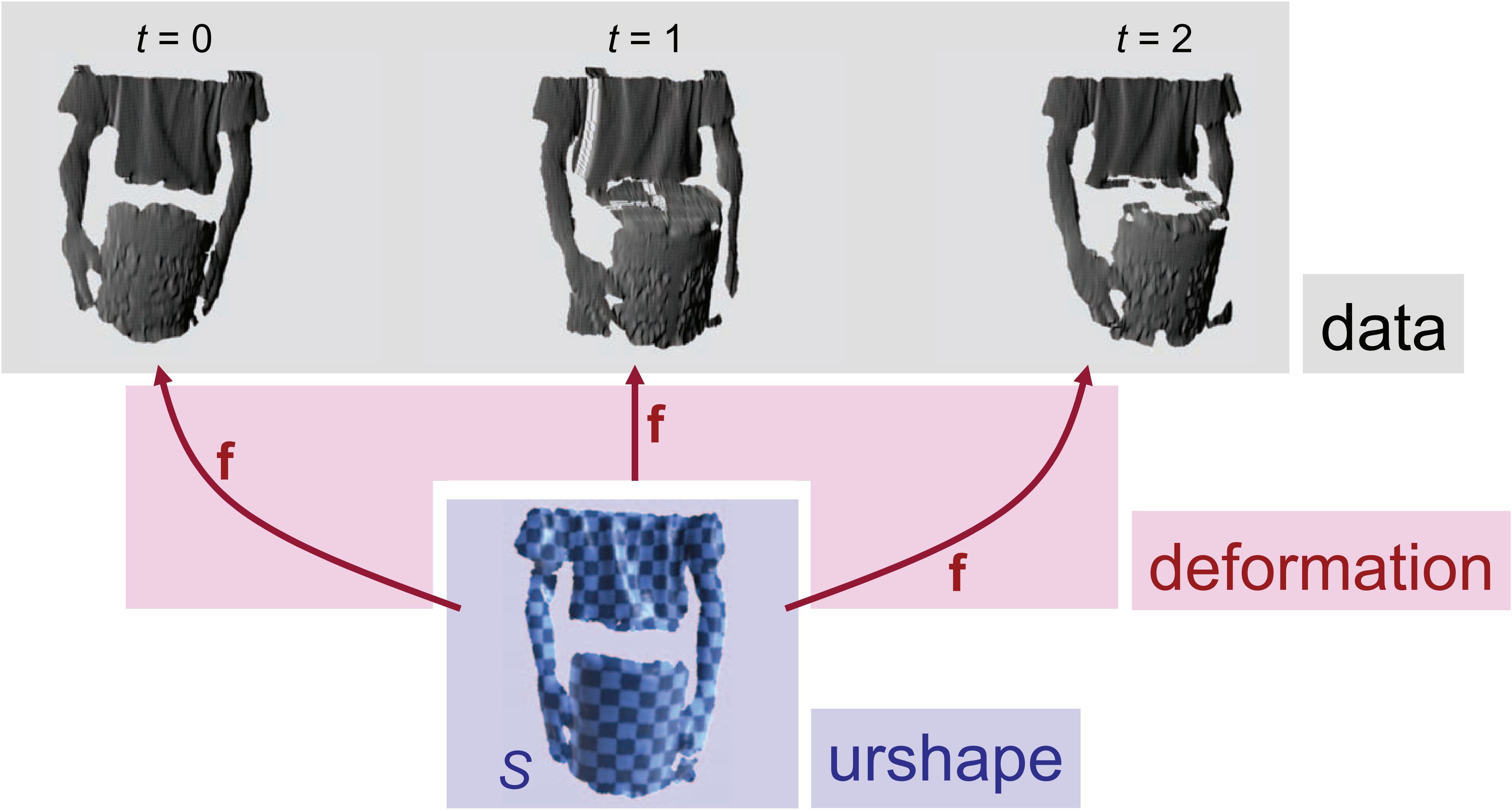


data



trajectories

New: Factorization



Components

Variational Model

- Given an initial estimate,
  improve *urshape* and *deformation*

Numerical Discretization

- *Shape*
- *Deformation*

Domain Assembly

- Getting an initial estimate
- *Urshape* assembly

Components

Variational Model

- Given an initial estimate, improve *urshape* and *deformation*

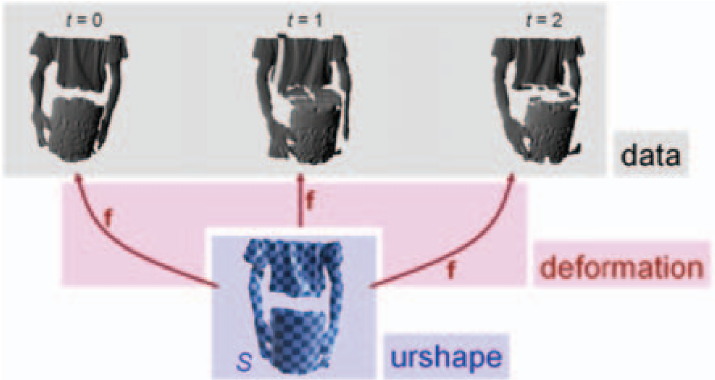
Numerical Discretization

- *Shape*
- *Deformation*

Domain Assembly

- Getting an initial estimate
- *Urshape* assembly

Energy Minimization



Energy Function

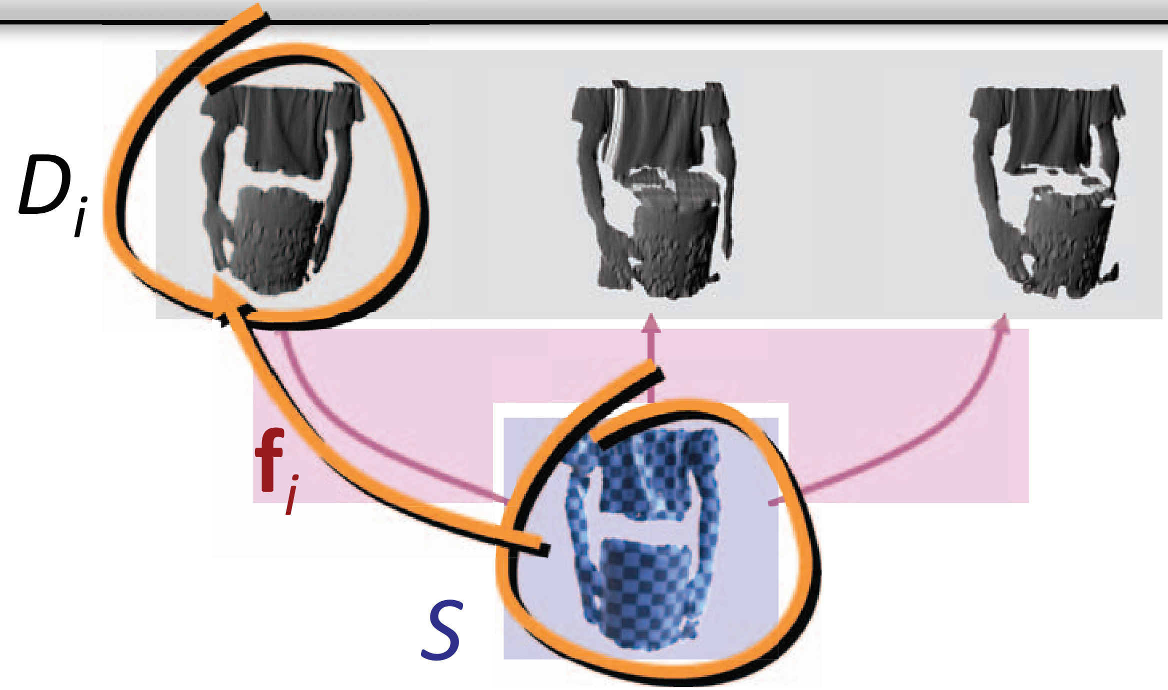
$E(\mathbf{f}, S) = E_{data} + E_{deform} + E_{smooth}$

Components

- $E_{data}(\mathbf{f}, S)$ – data fitting
- $E_{deform}(\mathbf{f})$ – elastic deformation, smooth trajectory
- $E_{smooth}(S)$ – smooth surface

Optimize S, $\mathbf{f}$ alternatingly

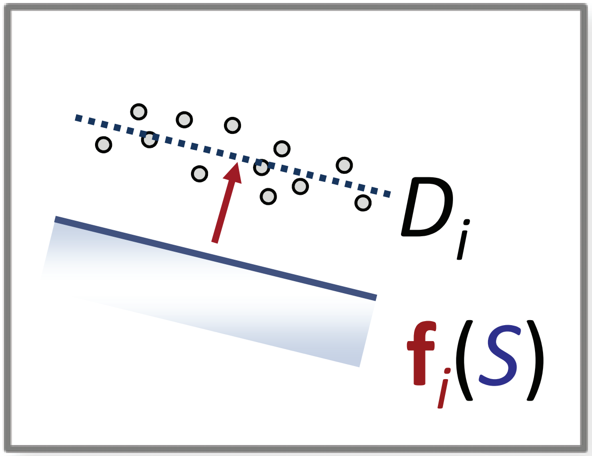
Data Fitting



$E_{data}(\mathbf{f}, S)$

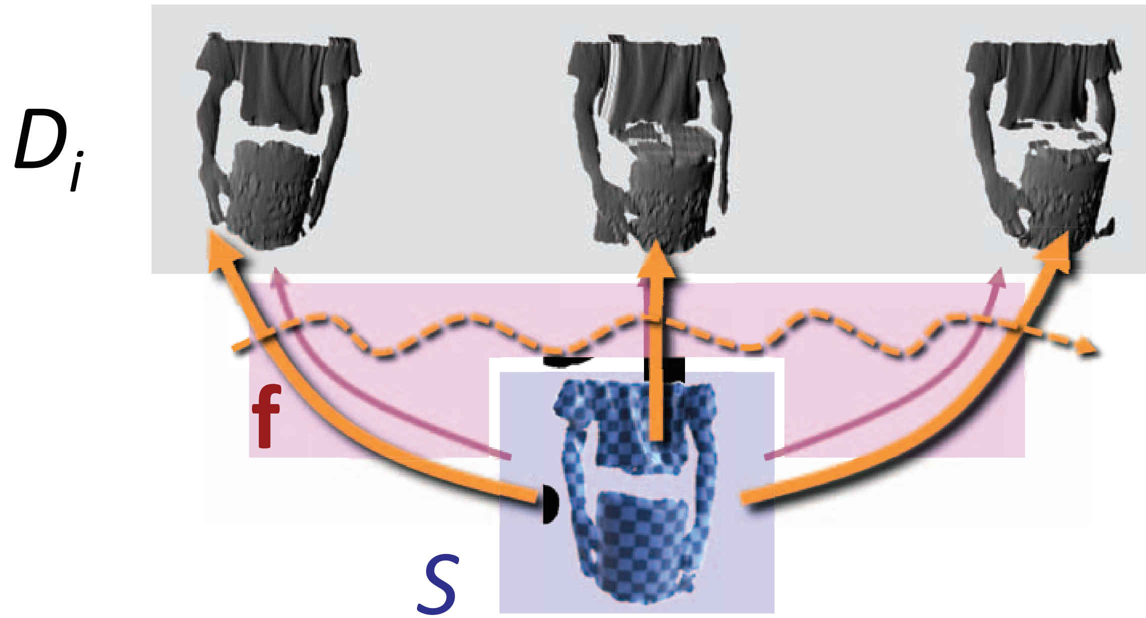
Data fitting

- Necessary: $\mathbf{f}_i(S) \approx D_i$
- Truncated squared distance function (point-to-plane)



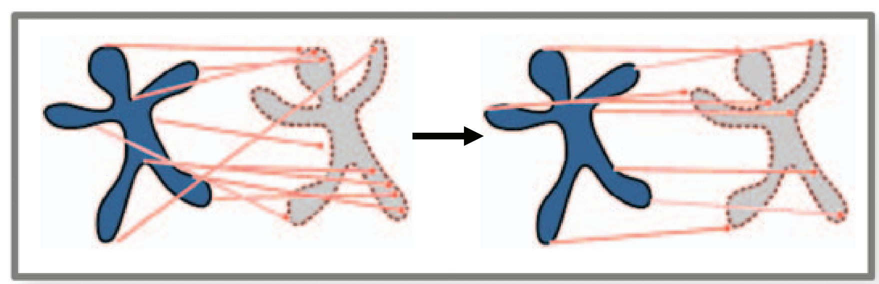
Elastic Deformation Energy

$E_{deform}(\mathbf{f})$

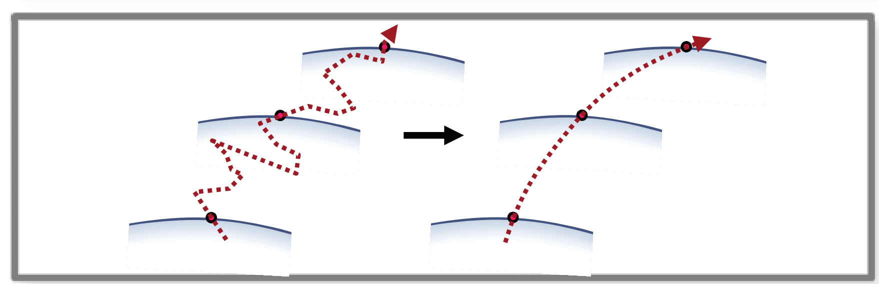


Regularization

- Elastic energy

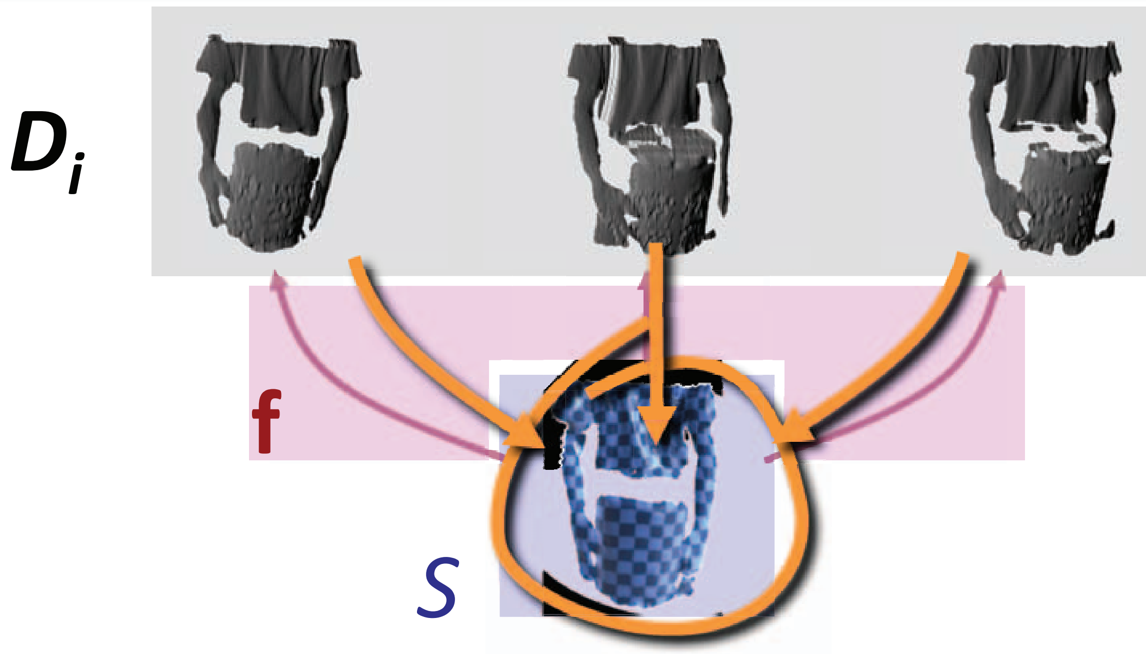


- Smooth trajectories



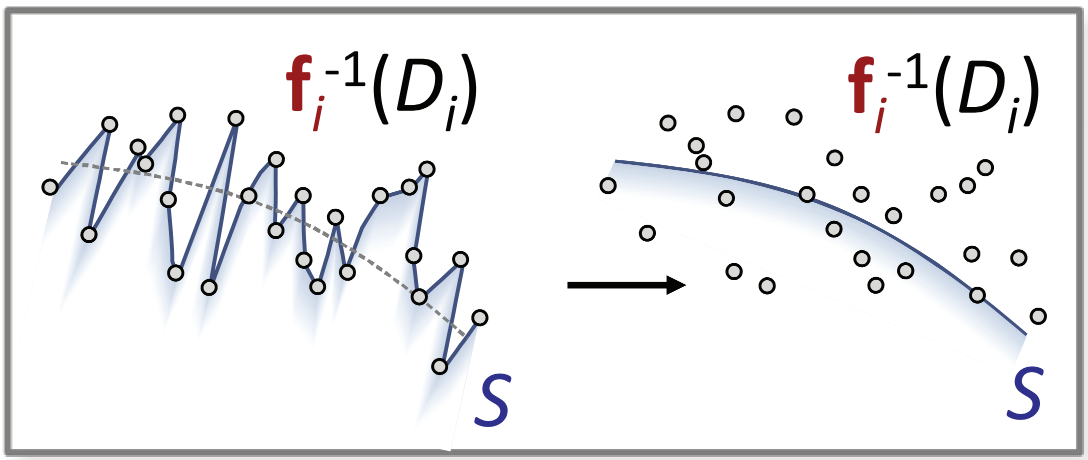
Surface Reconstruction

$E_{smooth}(S)$

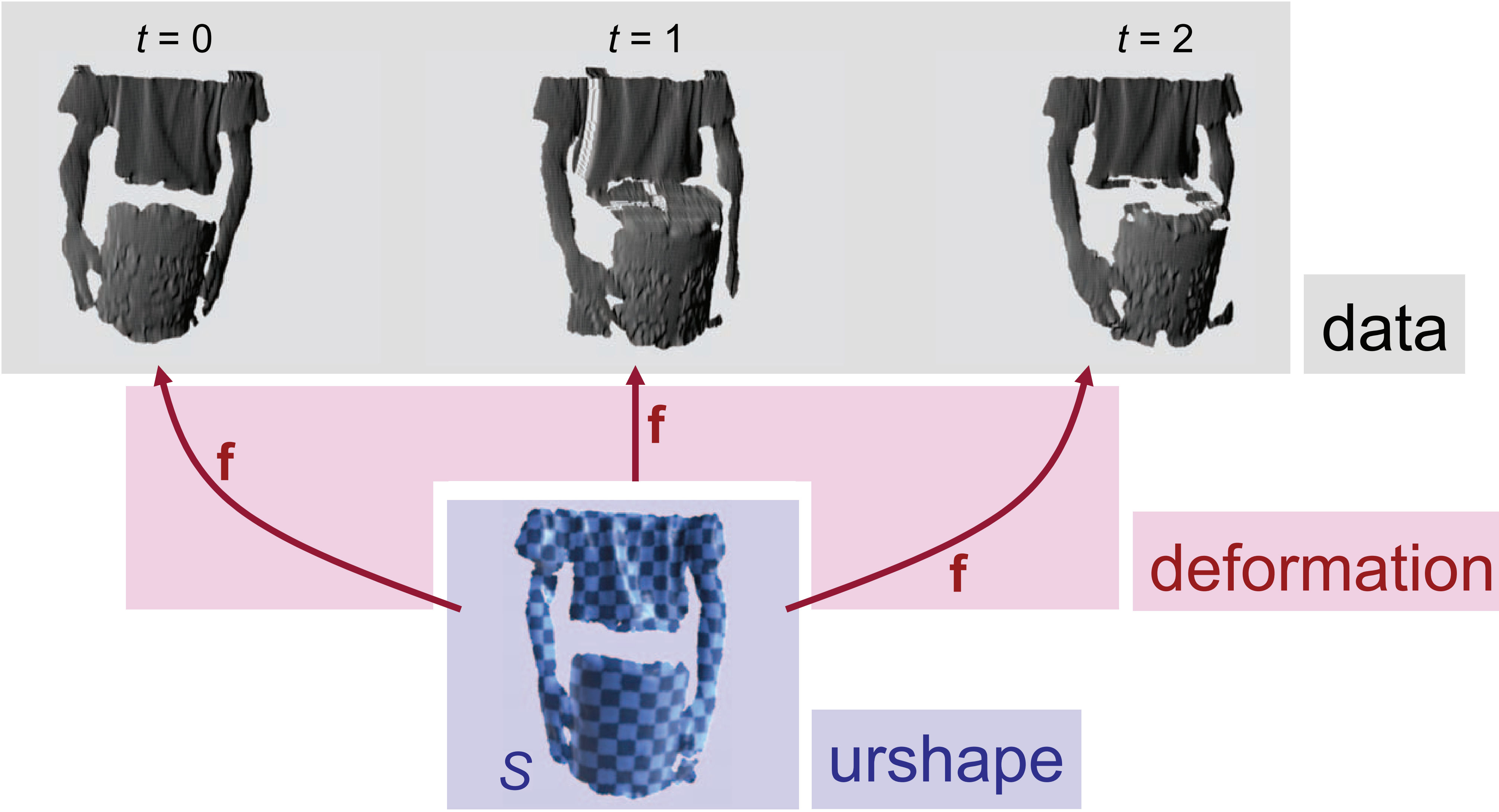


Data fitting

- Smooth surface
- Fitting to noisy data



Factorization



Components

Variational Model

- Given an initial estimate,
  improve *urshape* and *deformation*

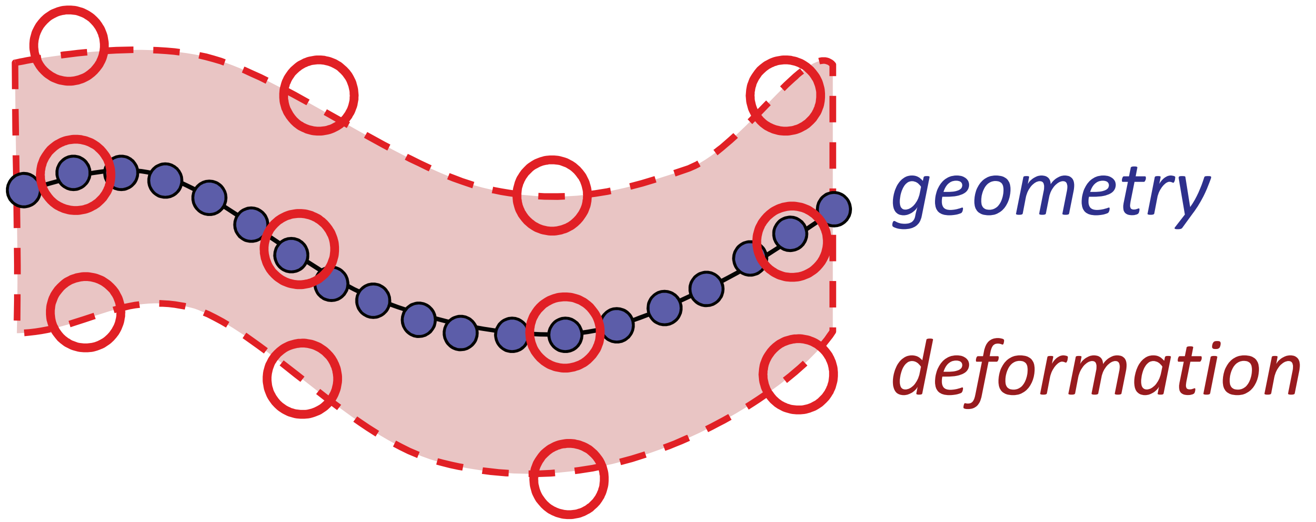
Numerical Discretization

- *Shape*
- *Deformation*

Domain Assembly

- Getting an initial estimate
- *Urshape* assembly

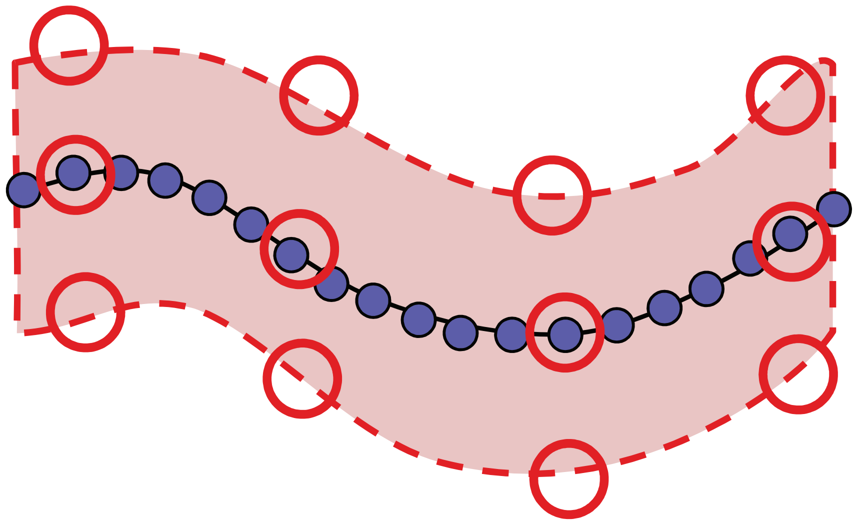
Discretization



Sampling:

- Full resolution *geometry*
- Subsample *deformation*

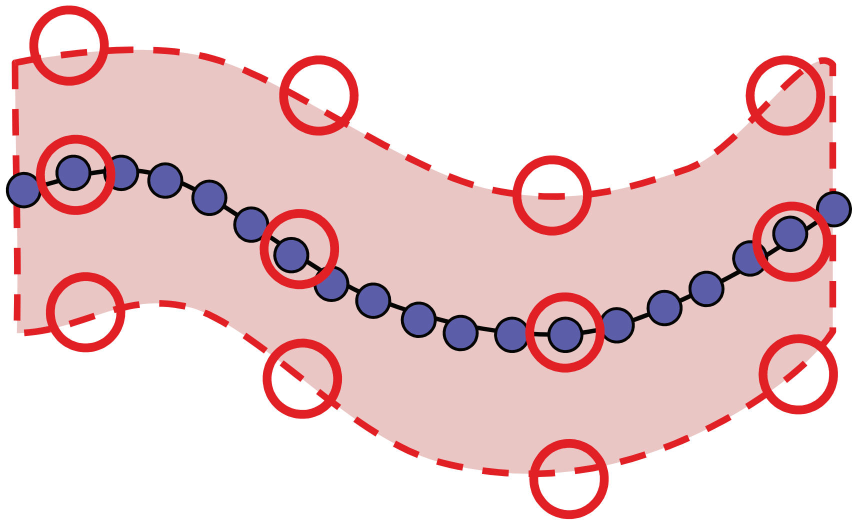
Discretization



Sampling:

- Full resolution *geometry*
  - High frequency
- Subsample *deformation*
  - Low frequency

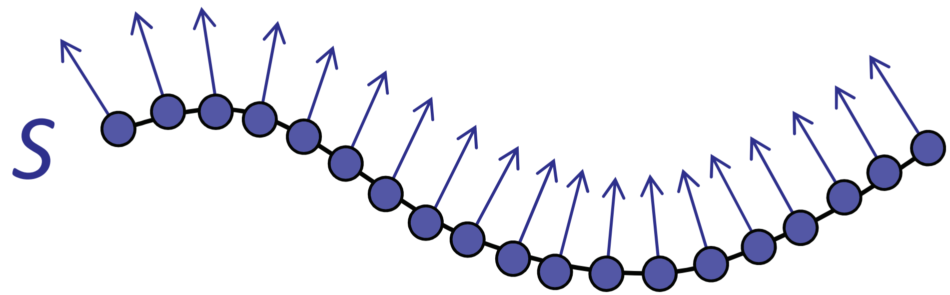
Discretization



Sampling:

- Full resolution *geometry*
  - High frequency, stored once
- Subsample *deformation*
  - Low frequency, all frames ⇒ more costly

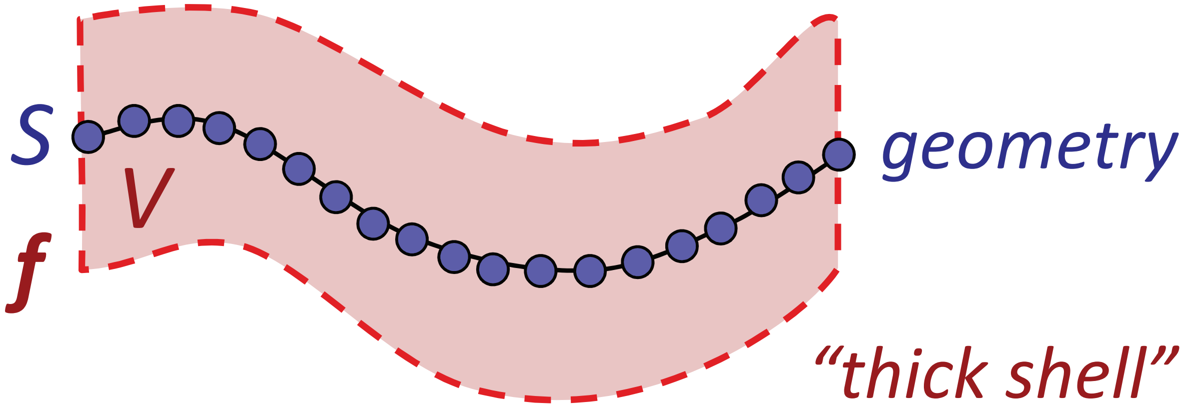
Shape Representation



Shape Representation:

- Graph of *surfels* (point + normal + local connectivity)
- E_{smooth} – neighboring planes should be similar
- Same as before...

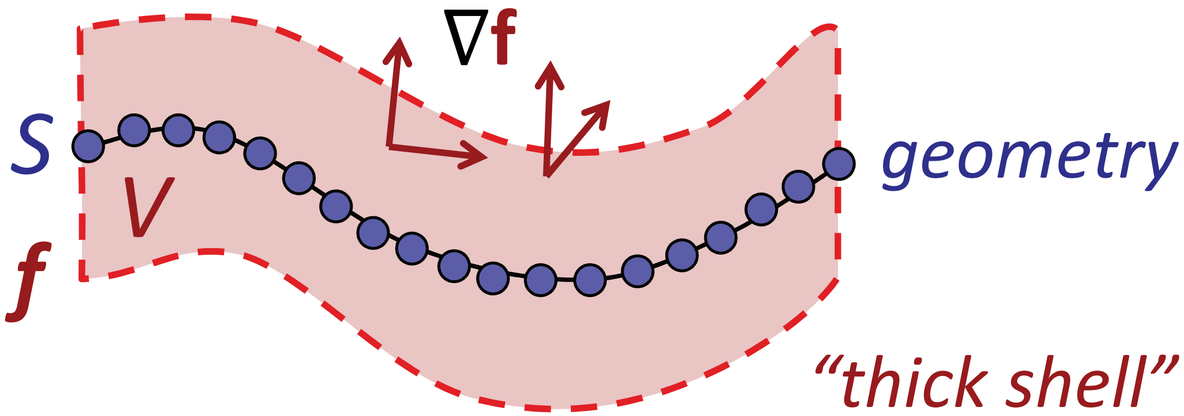
Deformation



Volumetric Deformation Model

- Surfaces embedded in "stiff" volumes
- Easier to handle than "thin-shell models"
- General – works for non-manifold data

Deformation



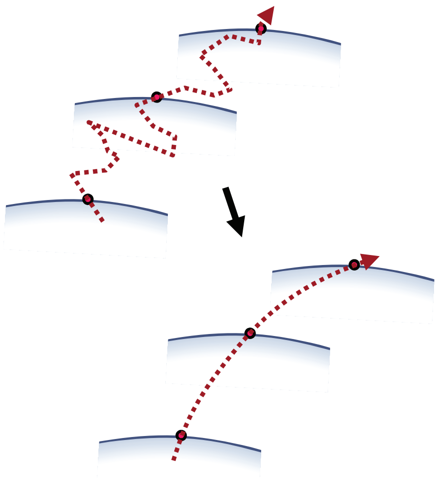
Deformation Energy

- Keep deformation gradients $\nabla \mathbf{f}$ as-rigid-as-possible
- This means: $\nabla \mathbf{f}^{\mathsf{T}} \nabla \mathbf{f} = \mathbf{I}$
- Minimize: $E_{deform} = \int_T \int_V \| \nabla \mathbf{f}(\mathbf{x},t)^{\mathsf{T}} \nabla \mathbf{f}(\mathbf{x},t) - \mathbf{I} \|^2 \, d\mathbf{x} \, dt$

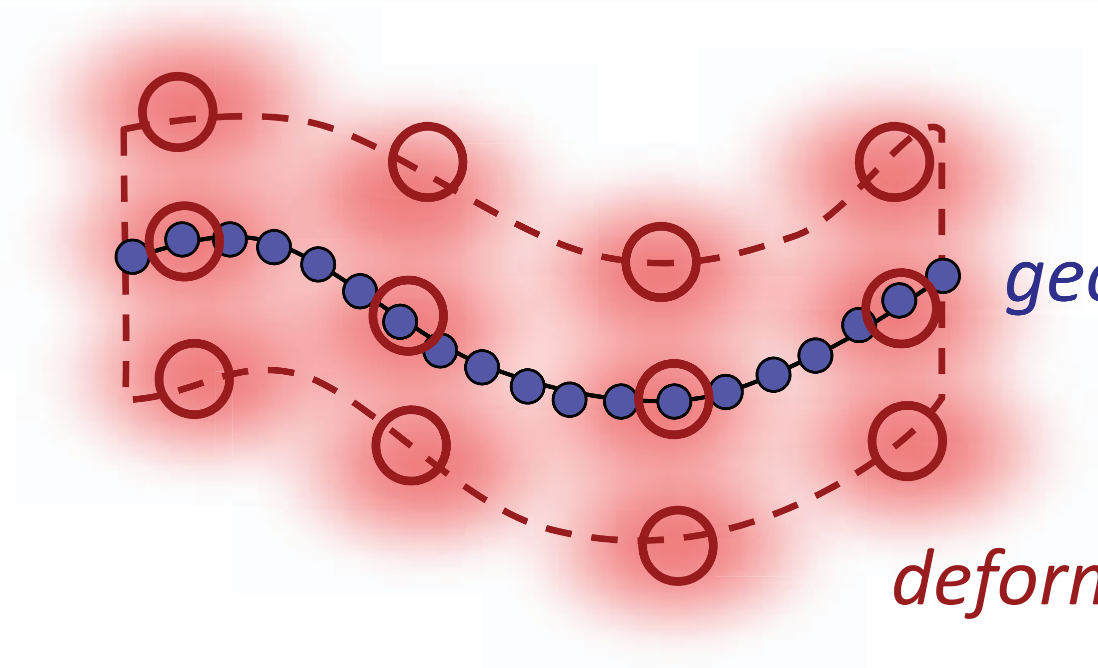
Additional Terms

More Regularization

- Volume preservation: $E_{vol} = \int_T \int_V ||\det(\nabla \mathbf{f}) - 1||^2$
  - Stability
- Acceleration: $E_{acc} = \int_T \int_V ||\partial_t^2\ \mathbf{f}||^2$
  - Smooth trajectories
- Velocity (weak): $E_{vel} = \int_T \int_V ||\partial_t\ \mathbf{f}||^2$
  - Damping



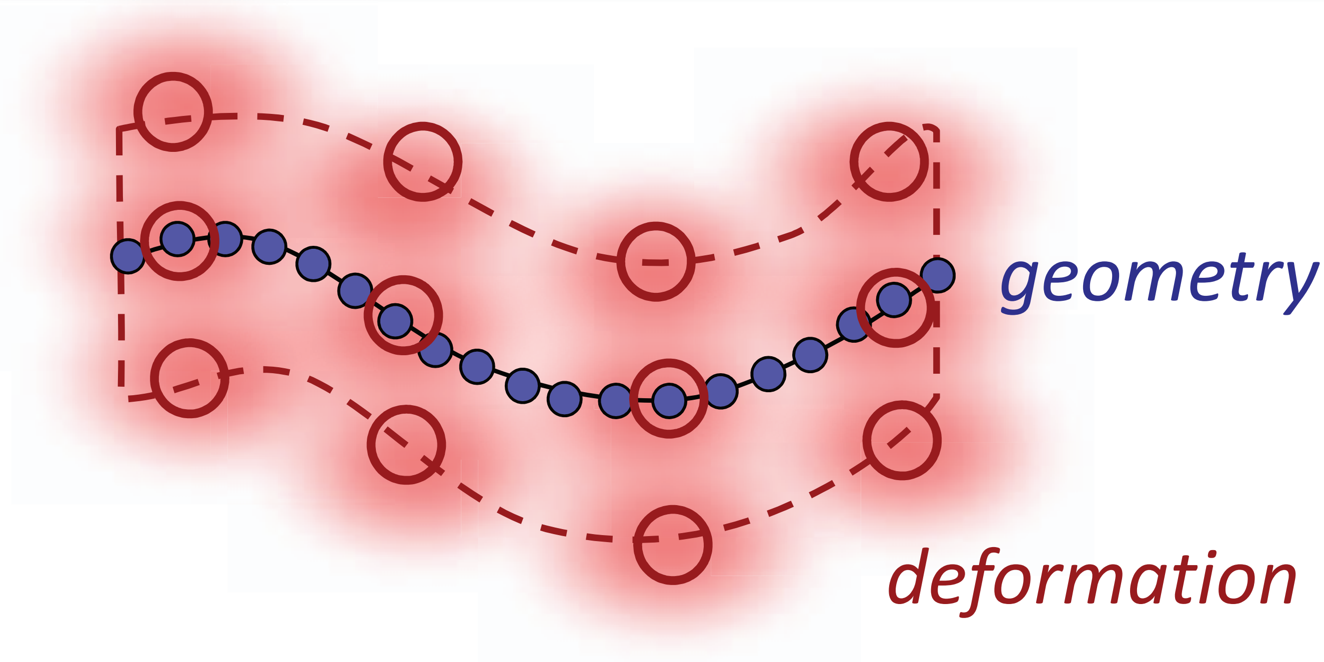
Discretization



How to represent the deformation?

- Goal: efficiency
- Finite basis:
  As few basis functions as possible

Discretization



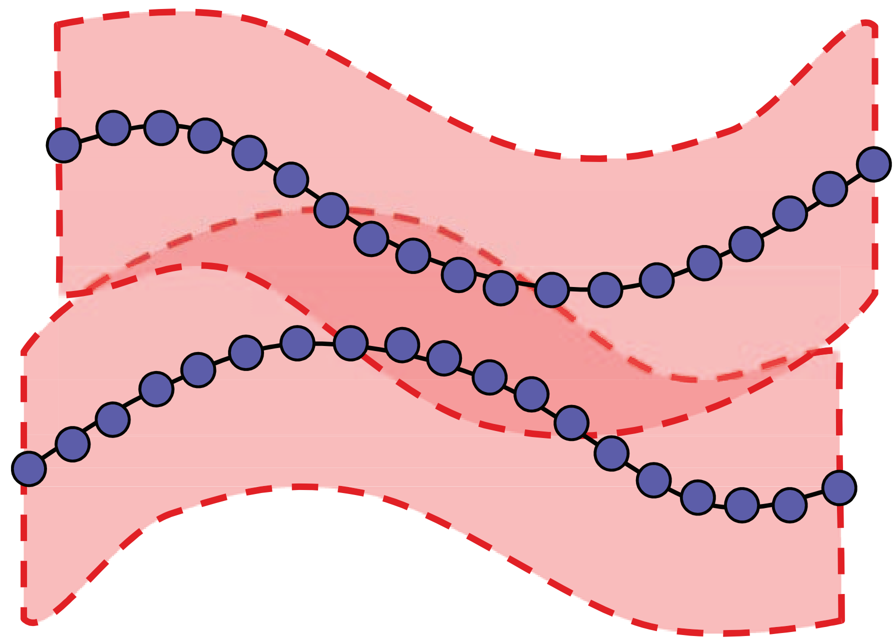
Meshless finite elements

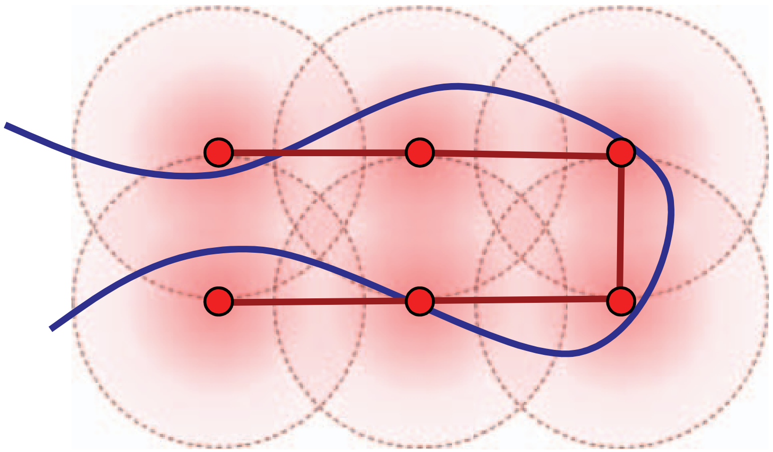
- Partition of unity, smoothness
- Linear precision
- Adaptive sampling is easy

Meshless Finite Elements

Topology:

- Separate deformation nodes for disconnected pieces
- Need to ensure
  - Consistency
  - Continuity
- Euclidean / intrinsic distance-based coupling rule
  - See references for details

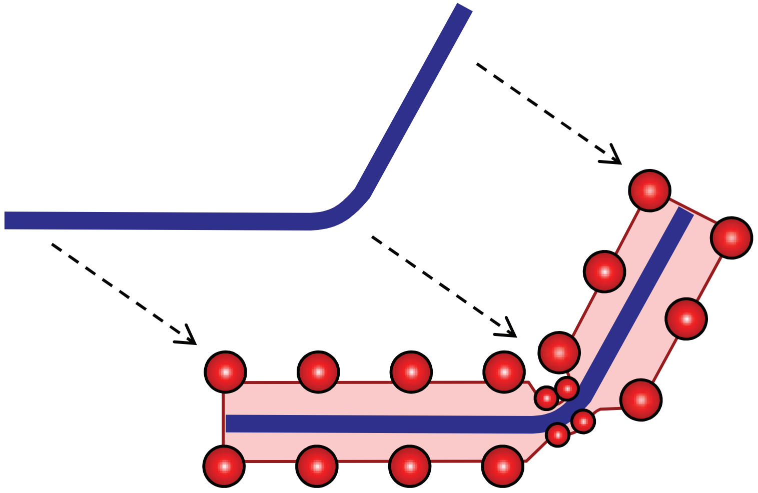




Adaptive Sampling

Adaptive Sampling

- Bending areas
  - Decrease rigidity
  - Decrease thickness
  - Increase sampling density
- Detecting bending areas: residuals over many frames



Components

Variational Model

- Given an initial estimate,
  improve *urshape* and *deformation*

Numerical Discretization

- *Deformation*
- *Shape*

Domain Assembly

- Getting an initial estimate
- *Urshape* assembly

Urshape Assembly

Adjacent frames are similar

- Solve for frame pairs first
- Assemble urshape step-by-step

[data set courtesy of C. Theobald, MPC-VCC]

Hierarchical Merging

data

f(*S*)

f

S

Hierarchical Merging

data

f(S)

f

S

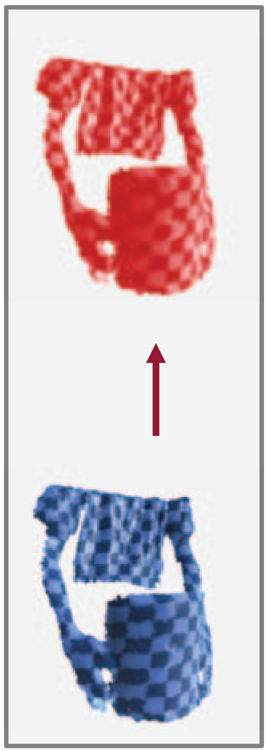
Initial Urshapes

data

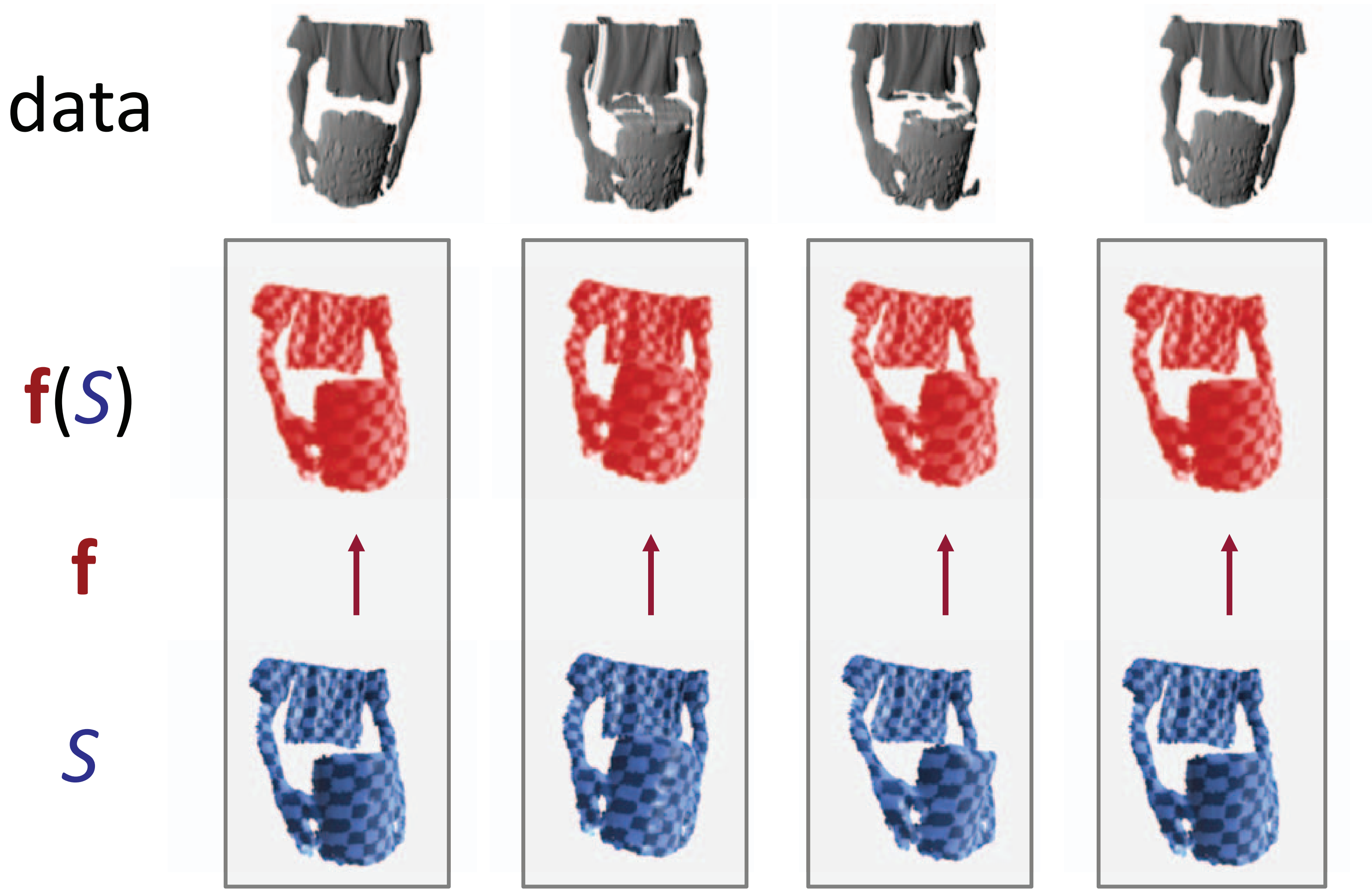
f(S)

f

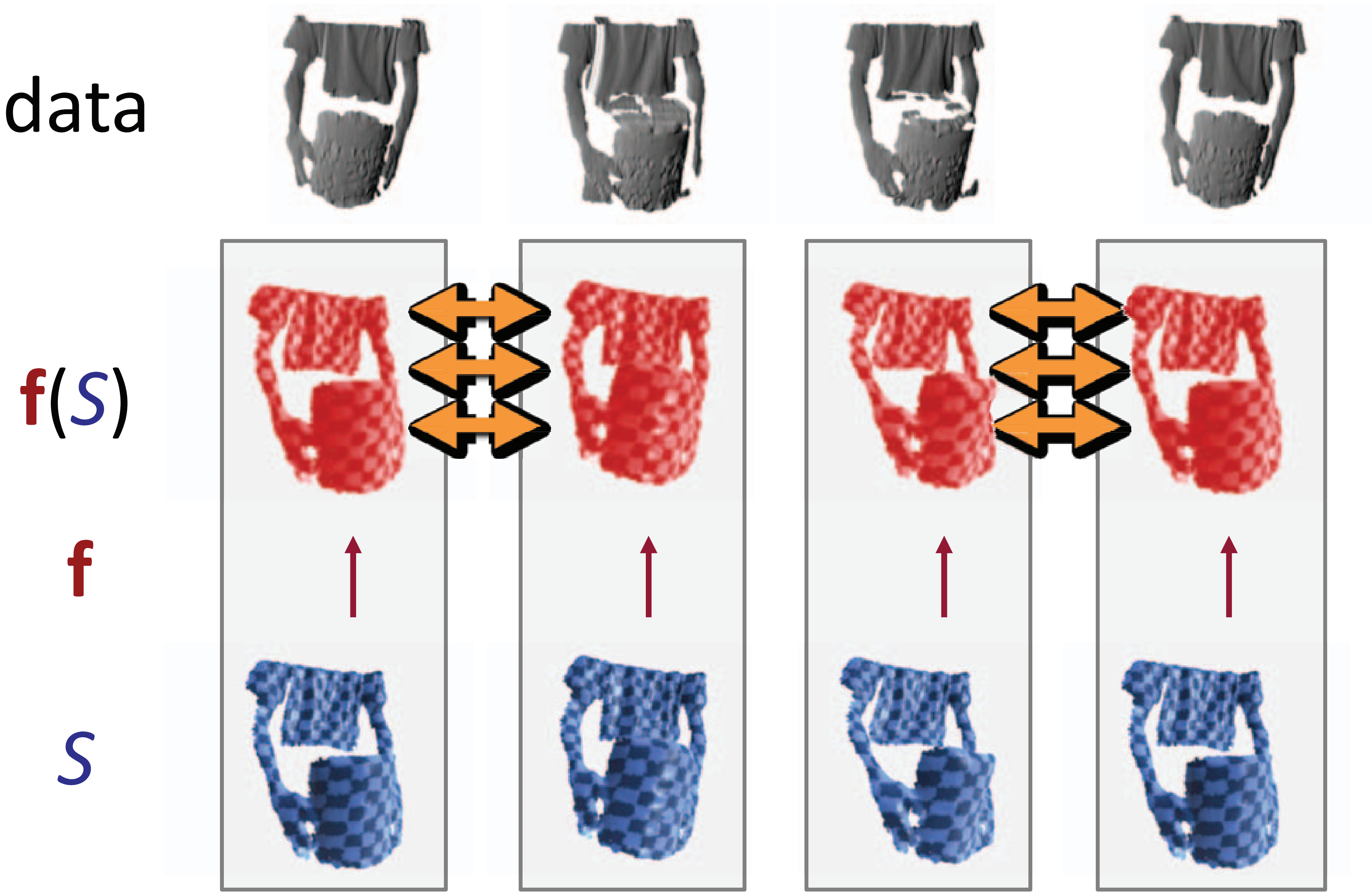
S



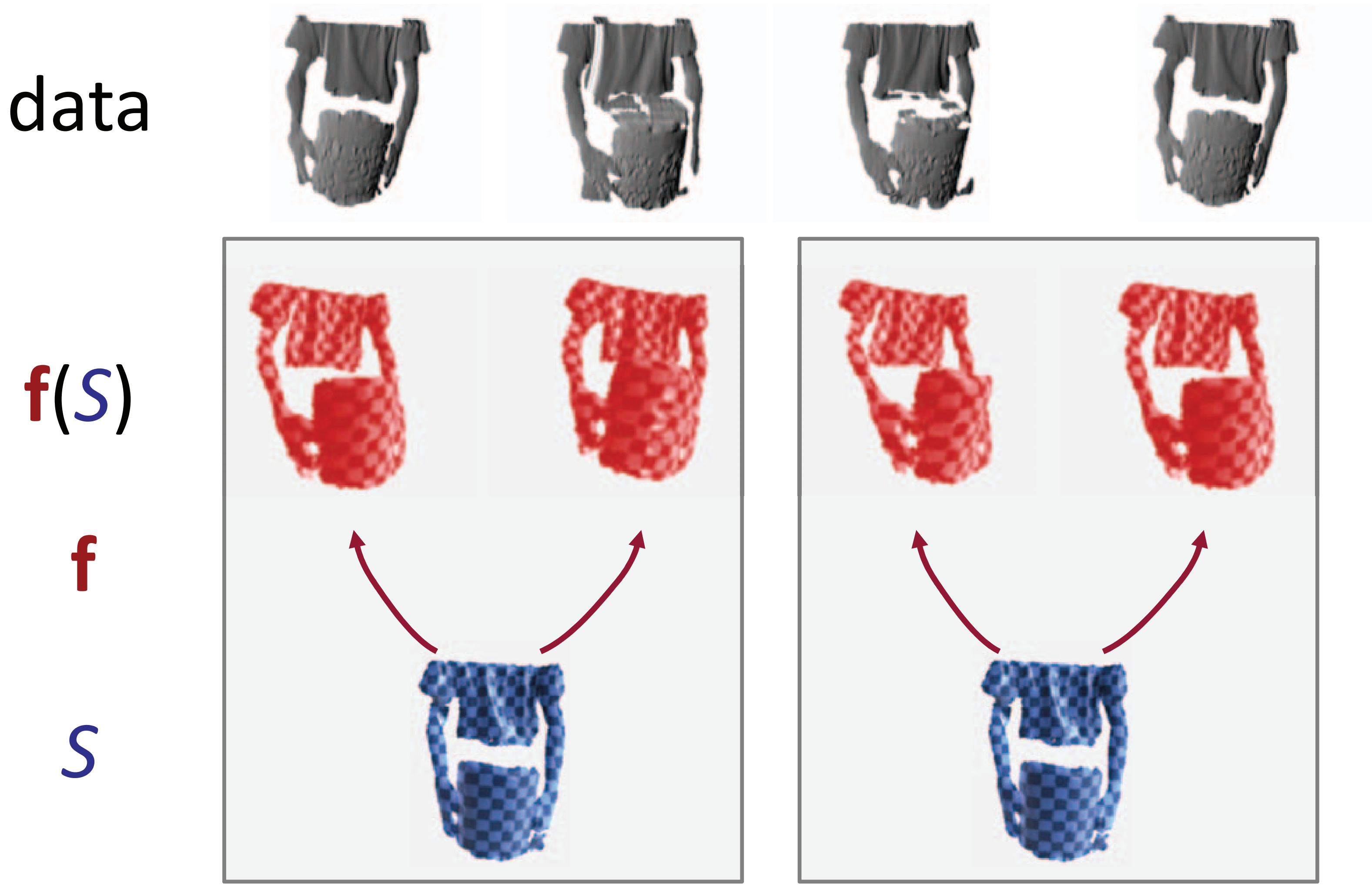
Initial Urshapes



Alignment



Align & Optimize



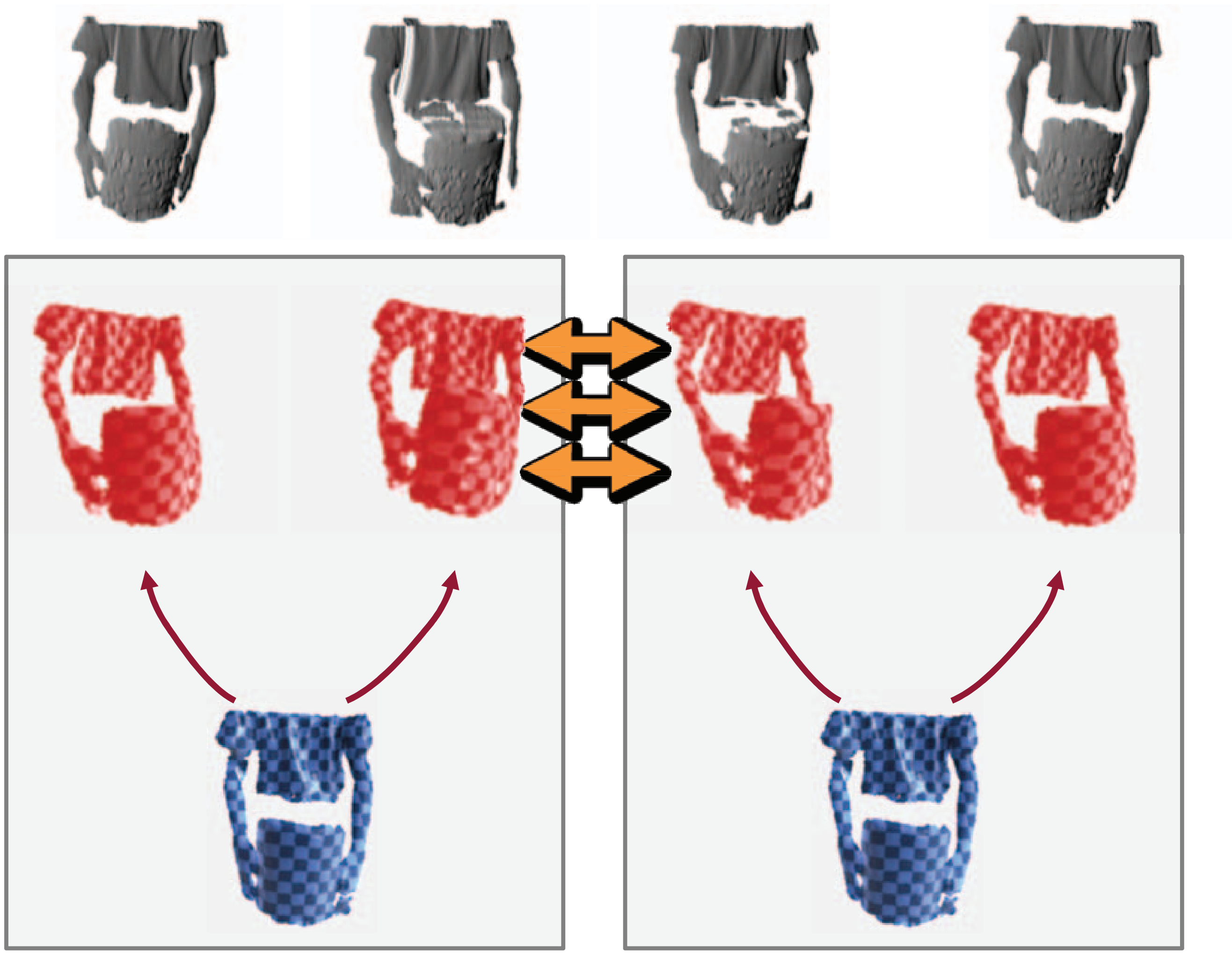
Hierarchical Alignment

data

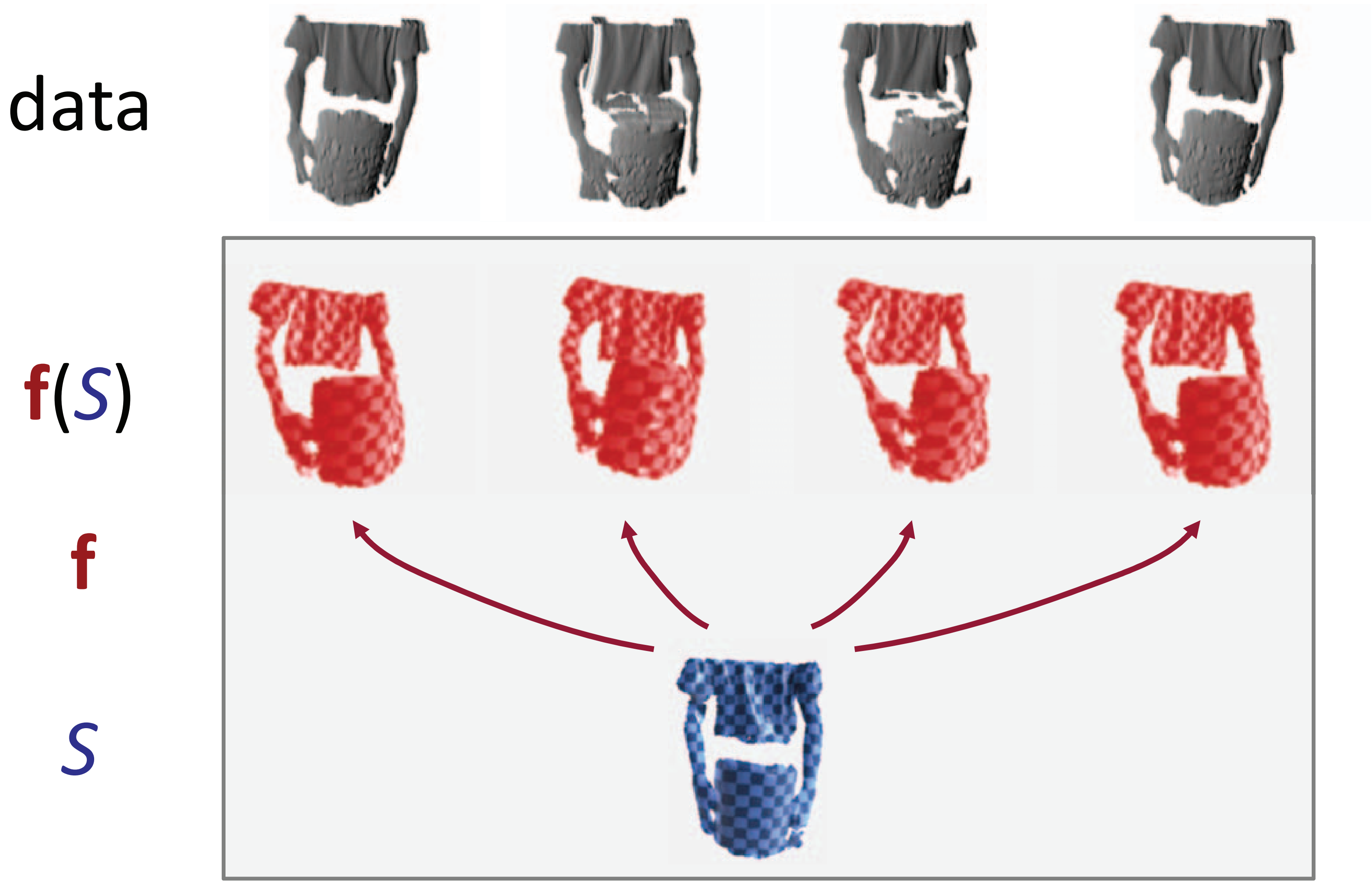
f(*S*)

f

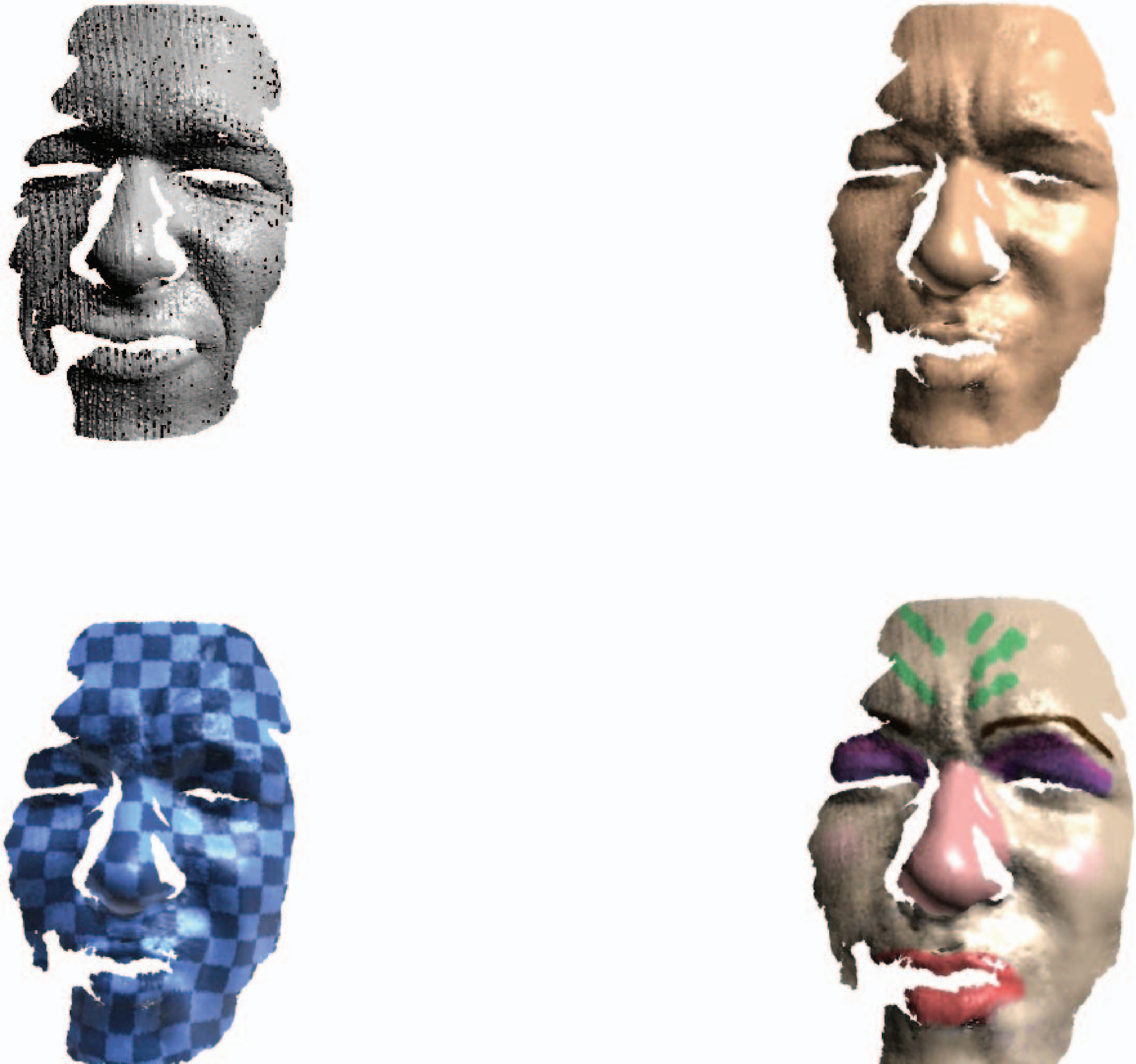
S



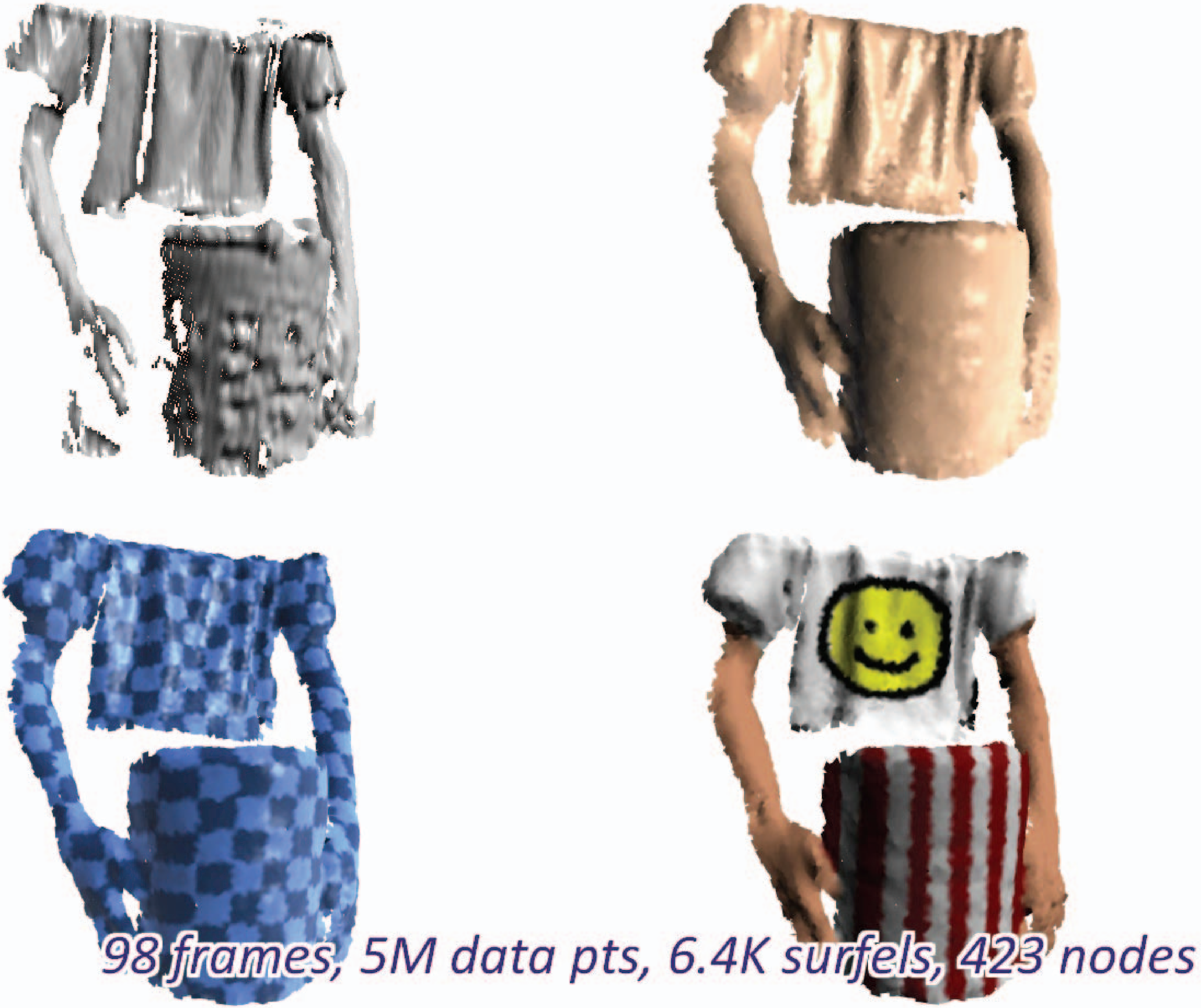
Hierarchical Alignment



Results



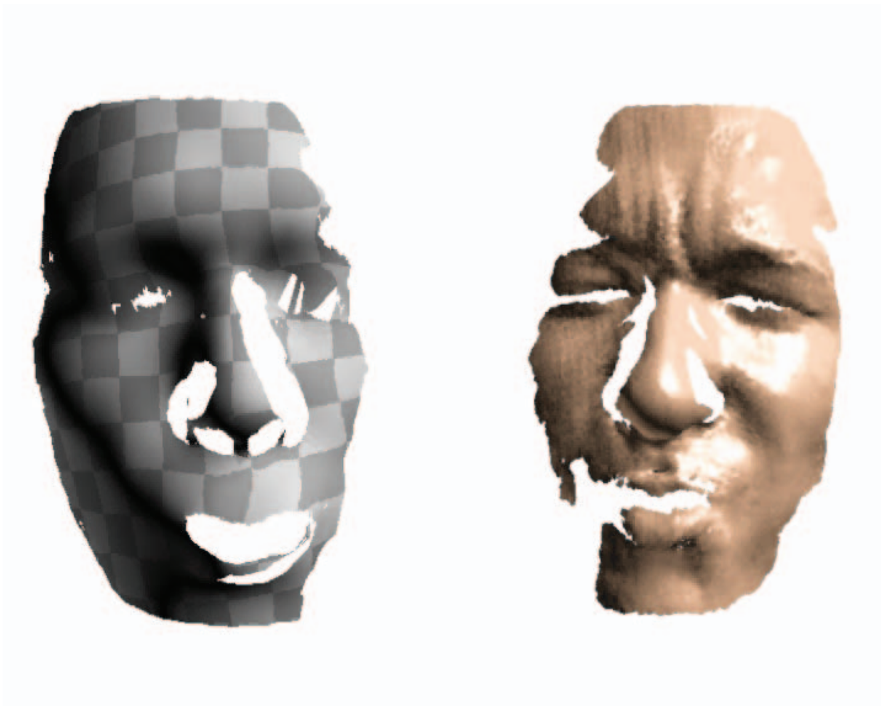
79 frames, 24M data pts, 21K surfels, 315 nodes

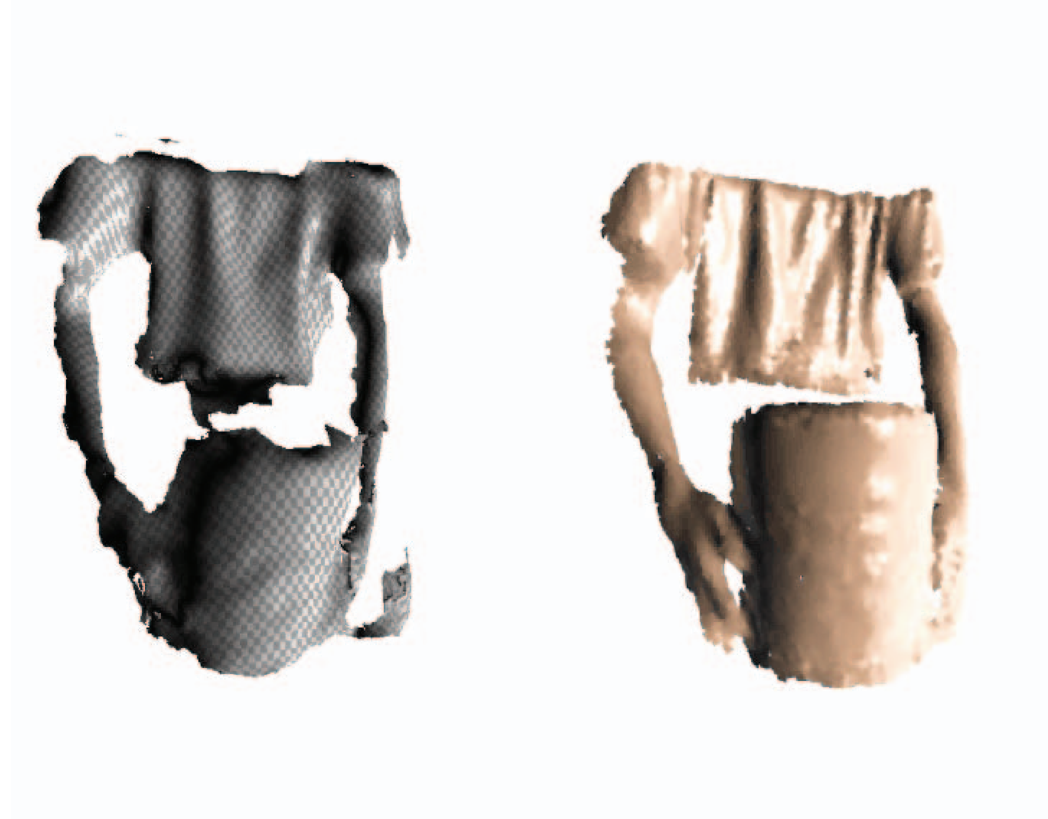

98 frames, 5M data pts, 6.4K surfels, 423 nodes

120 frames,
30M data pts,
17K surfels,
1,939 nodes

34 frames,
4M data pts,
23K surfels,
414 nodes

Quality Improvement





old version | **new result** | old version | **new result**

Animation Reconstruction References

Animation Reconstruction References

- **Adams, B., Ovsjanikov, M., Wand, M., Seidel, H.-P., Guibas, L. J. (2008).** Meshless Modeling of Deformable Shapes and their Motion. In: *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*.
- **de Aguiar, E., Stoll, C., Theobalt, C., Ahmed, N., Seidel, H.-P., Thrun, S. (2008).** Performance capture from sparse multi-view video. In: *ACM Trans. Graph. 27(3), 1—10*.
- **Alexa, M., Cohen-Or, D., Levin, D. (2000).** As-rigid-as-possible shape interpolation. In: *Proc. SIGGRAPH 2000, pp. 157—164*.
- **Allen, B., Curless, B., Popović, Z. (2003).** The space of human body shapes: reconstruction and parameterization from range scans. In: *Proc. SIGGRAPH 03, pp. 587—594*.
- **Allen, B., Curless, B., Popović, Z. (2002).** Articulated body deformation from range scan data. In: *ACM Trans. Graph. 21(3), 612—619*.
- **Anuar, N., Guskov, I. (2004).** Extracting Animated Meshes with Adaptive Motion Estimation. In: *Proc. Vision Modeling and Visualization (VMV) 2004, pp. 63-71*.
- **Besl, P. J., Mckay, N. (1992).** A Method for Registration of 3-D Shapes. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence 14(2), 239–256*.

Animation Reconstruction References

Animation Reconstruction References

- **Bickel, B., Botsch, M., Angst, R., Matusik, W., Otaduy, M., Pfister, H., Gross, M. (2007).** Multi-scale capture of facial geometry and motion. In: *ACM Trans. Graph. 26(3)*.
- **Botsch, M., Pauly, M., Gross, M., Kobbelt, L. (2006).** PriMo: coupled prisms for intuitive surface modeling. In: *Proc Symposium on Geometry Processing, pp. 11–20*.
- **Bronstein, A. M., Bronstein, M. M., Kimmel, R. (2006).** Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. In: *Proc. National Academy of Sciences (PNAS) 103(5), 1168-1172*.
- **Brown, B., Rusinkiewicz, S. (2007).** Global Non-Rigid Alignment of 3-D Scans. In: *ACM Transactions on Graphics (Proc. SIGGRAPH) 26(3)*.
- **Carceroni, R. L., Kutulakos, K. N. (2002).** Multi-View Scene Capture by Surfel Sampling: From Video Streams to Non-Rigid 3D Motion, Shape and Reflectance. In: *Int. J. Comput. Vision 49(2-3), 175–214*.
- **Carranza, J., Theobalt, C., Magnor, M. A., Seidel, H.-P. (2003).** Free-viewpoint video of human actors. In: *Proc. ACM SIGGRAPH 2003, pp. 569–577*.
- **Chen, Y., Medioni, G. (1992).** Object modelling by registration of multiple range images. In: *Image Vision Comput. 10(3), 145–155*.

Animation Reconstruction References

Animation Reconstruction References

- **Fries, T.-P., Matthies, H. G. (2003).** Classification and Overview of Meshfree Methods. *Technical report, TU Brunswick, Germany Nr. 2003-03*.
- **Gatzke, T., Grimm, C., Garland, M., Zelinka, S. (2005).** Curvature Maps for Local Shape Comparison. *In: Proc. Shape Modeling International, pp. 244–256*.
- **Haehnel, D., Thrun, S., Burgard, W. (2003).** An Extension of the ICP Algorithm for Modeling Nonrigid Objects with Mobile Robots. In: *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*.
- **Li, H., Sumner, R. W., Pauly, M. (2008).** Global Correspondence Optimization for Non-Rigid Registration of Depth Scans. In: *Computer Graphics Forum (Proc. SGP 2008) 27(5)*.
- **Mitra, N. J., Flory, S., Ovsjanikov, M., Gelfand, N., Guibas, L., Pottmann, H. (2007).** Dynamic Geometry Registration. In: *Symposium on Geometry Processing, pp. 173–182*.
- **Mitra, N. J., Gelfand, N., Pottmann, H., Guibas, L. (2004).** Registration of Point Cloud Data from a Geometric Optimization Perspective. In: *Symposium on Geometry Processing, pp. 23—31*.
- **Park, S. I., Hodgins, J. K. (2006).** Capturing and animating skin deformation in human motion. In: *ACM Trans. Graph. 25(3), 881—889*.
- **Pekelny, Y., Gotsman, C. (2008).** Articulated Object Reconstruction and Markerless Motion Capture from Depth Video. In: *Proc. Eurographics 2008*.

Animation Reconstruction References

Animation Reconstruction References

- **Süssmuth, J., Winter, M., Greiner, G. (2008).** Reconstructing Animated Meshes from Time-Varying Point Clouds. In: *Computer Graphics Forum (Proceedings of SGP 2008) 27(5), 1469—1476*.
- **Sand, P., McMillan, L., Popović, J. (2003).** Continuous capture of skin deformation. In: *ACM Trans. Graph. 22(3), 578–586*.
- **Sharf, A., Alcantara, D. A., Lewiner, T., Greif, C., Sheffer, A., Amenta, N., Cohen-Or, D. (2008).** Space-time Surface Reconstruction using Incompressible Flow. In: *ACM Trans. Graph. (Proc. Siggraph Asia 2008)*.
- **Shinya, M. (2004).** Unifying Measured Point Sequences of Deforming Objects. In: *Proc. 3D Data Processing, Visualization, and Transmission, pp. 904—911*.
- **Starck, J., Hilton, A. (2005).** Spherical Matching for Temporal Correspondence of Non-Rigid Surfaces. In: *Proc. Conf. on Computer Vision, pp. 1387—1394*.
- **Terzopoulos, D., Platt, J., Barr, A., Fleischer, K. (1987).** Elastically deformable models. In: *Proc. SIGGRAPH 87, pp. 205—214*.
- **Varanasi, K., Zaharescu, A., Boyer, E., Horaud, R. P. (2008).** Temporal Surface Tracking Using Mesh Evolution. In: *Proc. European Conf. on Computer Vision, Springer-Verlag, Marseille, France, pp. 30—43*.

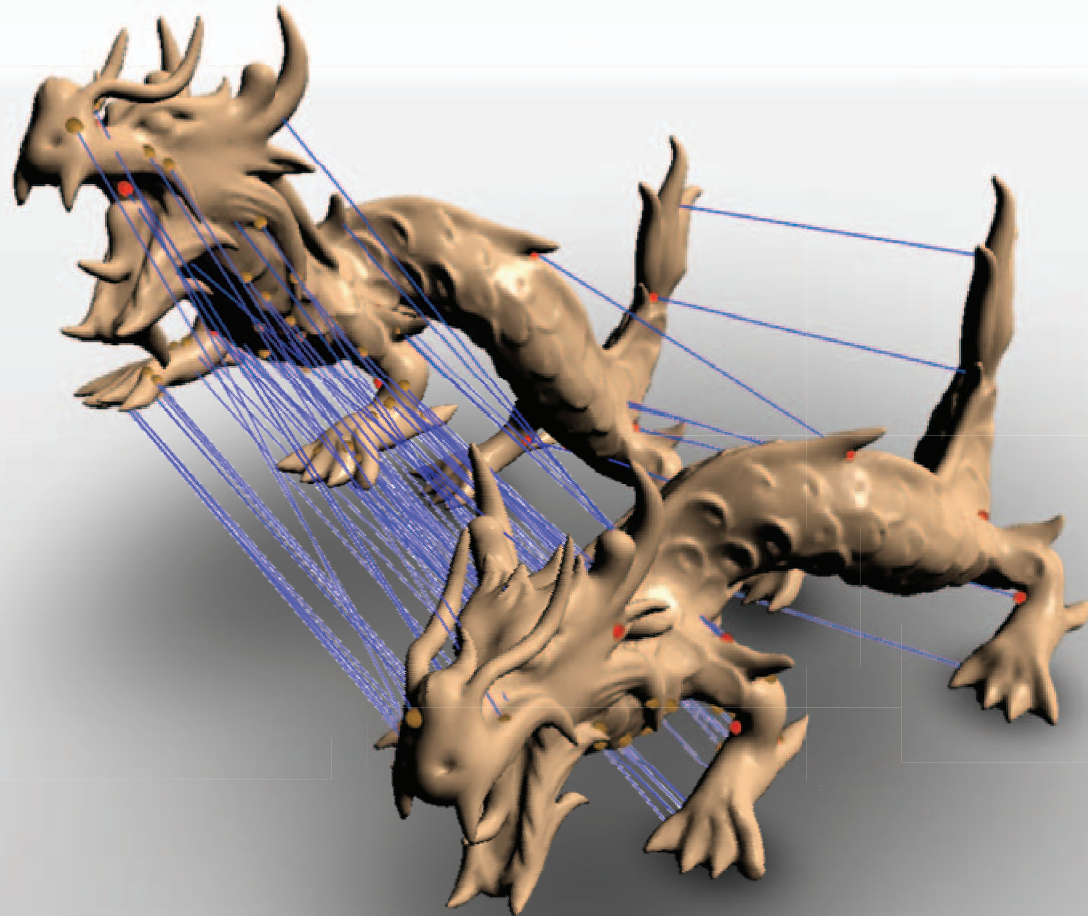
Animation Reconstruction References

Animation Reconstruction References

- **Zhang, L., Snavely, N., Curless, B., Seitz, S. M. (2004).** Spacetime Faces: High-Resolution Capture for Modeling and Animation. In: *Proc. Siggraph 2004, pp. 548-558*.
- **Zitnick, C. L., Kang, S. B., Uyttendaele, M., Winder, S., Szeliski, R. (2004).** High-quality video view interpolation using a layered representation. In: *Proc. Siggraph 2004, pp. 600—608*.

Presented in detail in this tutorial:

- **Wand, M., Jenke, P., Huang, Q., Bokeloh, M., Guibas, L., Schilling, A. (2007).** Reconstruction of deforming geometry from time-varying point clouds. In: *Proc. Symposium on Geometry Processing, pp. 49—58*.
- **Wand, M., Adams, B., Ovsjanikov, M., Berner, A., Bokeloh, M., Jenke, P., Guibas, L., Seidel, H.-P., Schilling, A.:** Efficient Reconstruction of Non-rigid Shape and Motion from Real-Time 3D Scanner Data. In: *ACM Transactions on Graphics 28(2), April 2009*.



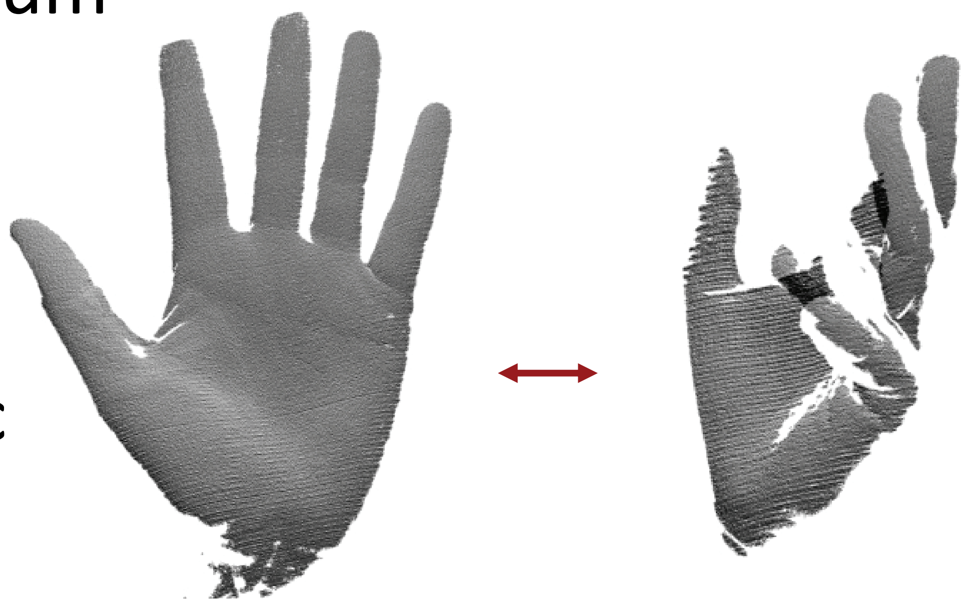
Global Deformable Matching

[data set: Stanford 3D Scanning Repository]

Problem Statement

Deformable Matching

- Two shapes: original, deformed
- How to establish correspondences?
- Looking for global optimum
  - Arbitrary pose



Assumption

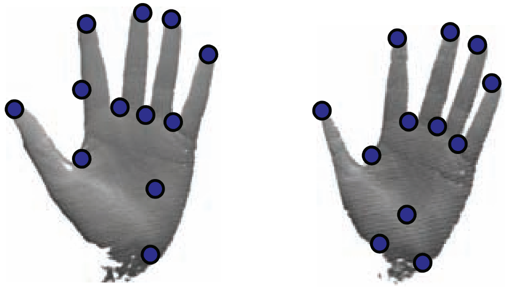
- Approximately isometric deformation

[data set: S. König, TU Dresden]

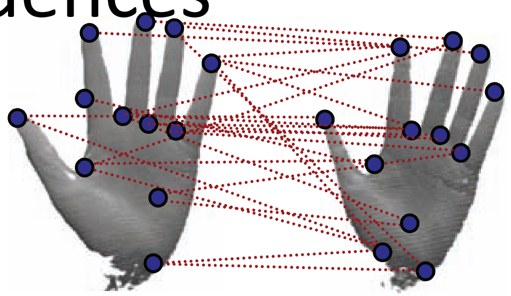
Algorithm

Feature-Matching

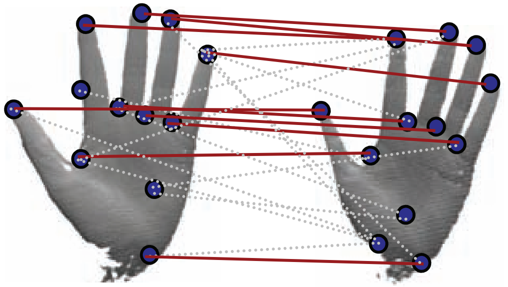
- Detect feature points



- Local matching: potential correspondences



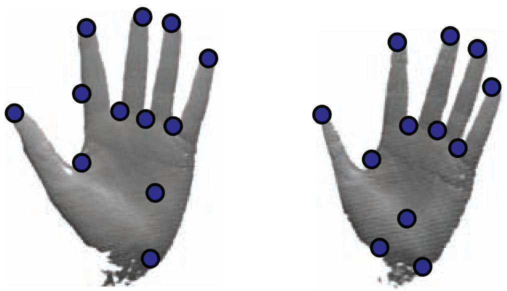
- Global filtering: correct subset



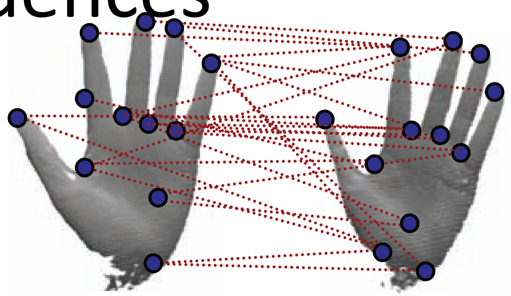
Algorithm

Feature-Matching

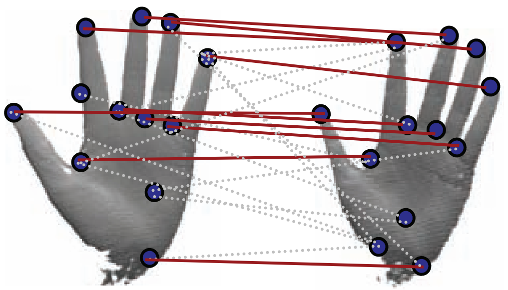
- Detect feature points
  - Geometric MLS-SIFT Features
  - Slippage Features (more sensitive)



- Local matching: potential correspondences



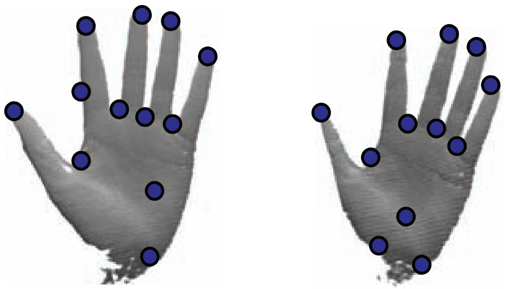
- Global filtering: correct subset



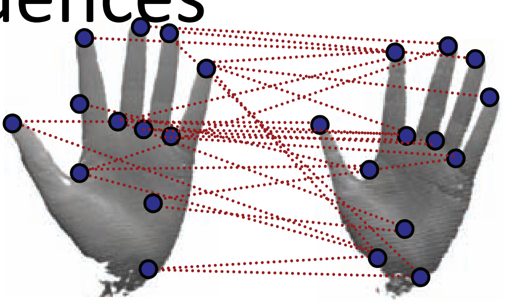
Algorithm

Feature-Matching

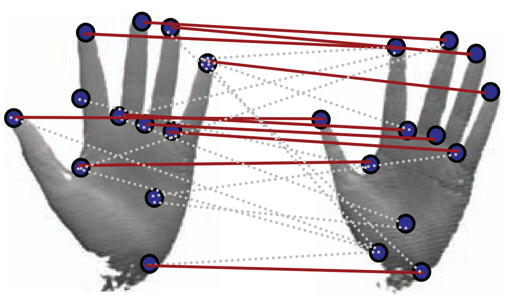
- Detect feature points
  - Geometric MLS-SIFT Features
  - Slippage Features (more sensitive)



- Local matching: potential correspondences
  - Descriptors (curvature, histograms)
  - Local ICP



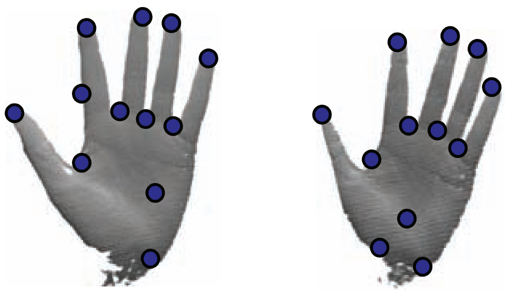
- Global filtering: correct subset



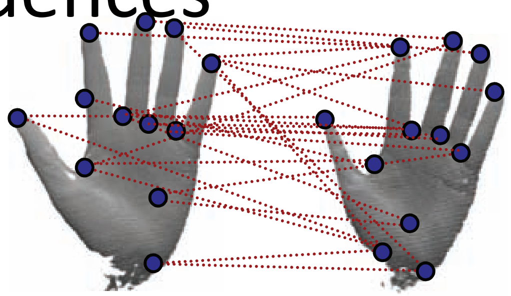
Algorithm

Feature-Matching

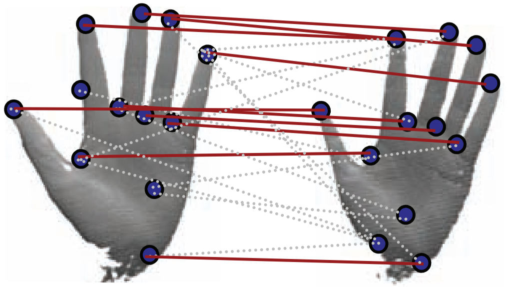
- Detect feature points
  - Geometric MLS-SIFT Features
  - Slippage Features (more sensitive)



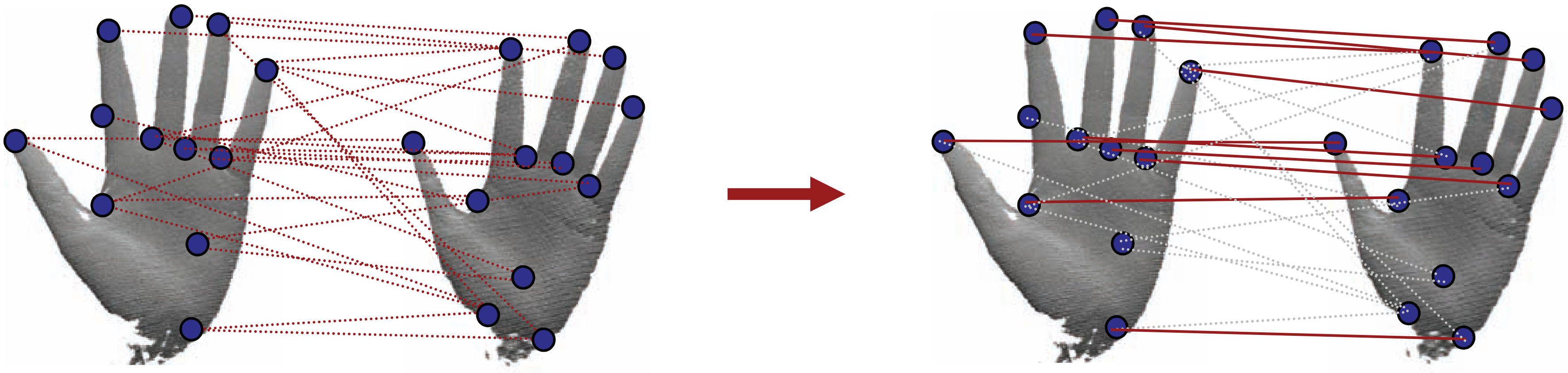
- Local matching: potential correspondences
  - Descriptors (curvature, histograms)
  - Local ICP



- Global filtering: correct subset
  - Quadratic assignment
  - Spectral relaxation [Leordeanu et al. 05]
  - RANSAC



Quadratic Assignment



Most difficult part: Global filtering

- Find a consistent subset
- Pairwise consistency:
  - Correspondence pair must preserve intrinsic distance
- Maximize number of pairwise consistent pairs
  - Quadratic assignment (in general: NP-hard)

Spectral Matching

Simple & Effective Approximation:

- Spectral matching [Leordeanu & Hebert 05]
- Form compatibility matrix:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{21} & a_{31} & \\ a_{12} & a_{22} & & \\ a_{13} & & \ddots & \\ & & & \ddots \end{pmatrix}$$

Diagonal:
Descriptor match

Off-Diagonal:
Pairwise compatibility

All entries within [0..1]
= [no match...perfect match]

Spectral Matching

Approximate largest clique:

- Compute eigenvector with largest eigenvalue
- Maximizes Rayleigh quotient:

$$\arg\max \frac{\mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x}}{\|\mathbf{x}\|^2}$$

- "Best yield" for bounded norm
  - The more consistent pairs (rows of 1s), the better
  - Approximates largest clique
- Implementation
  - For example: power iteration

Spectral Matching

Postprocessing

- Greedy quantization
  - Select largest remaining entry, set it to 1
  - Set all entries to 0 that are not pairwise consistent with current set
  - Iterate until all entries are quantized

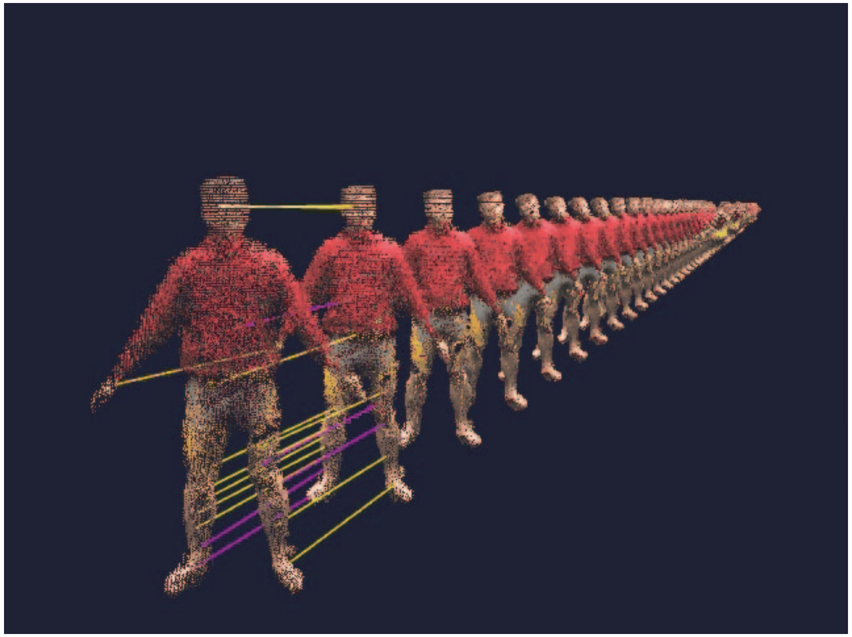
In practice...

- This algorithm turns out to work quite well...
- ...but there are other, more flexible alternatives.

Spectral Matching Example

Application to Animations

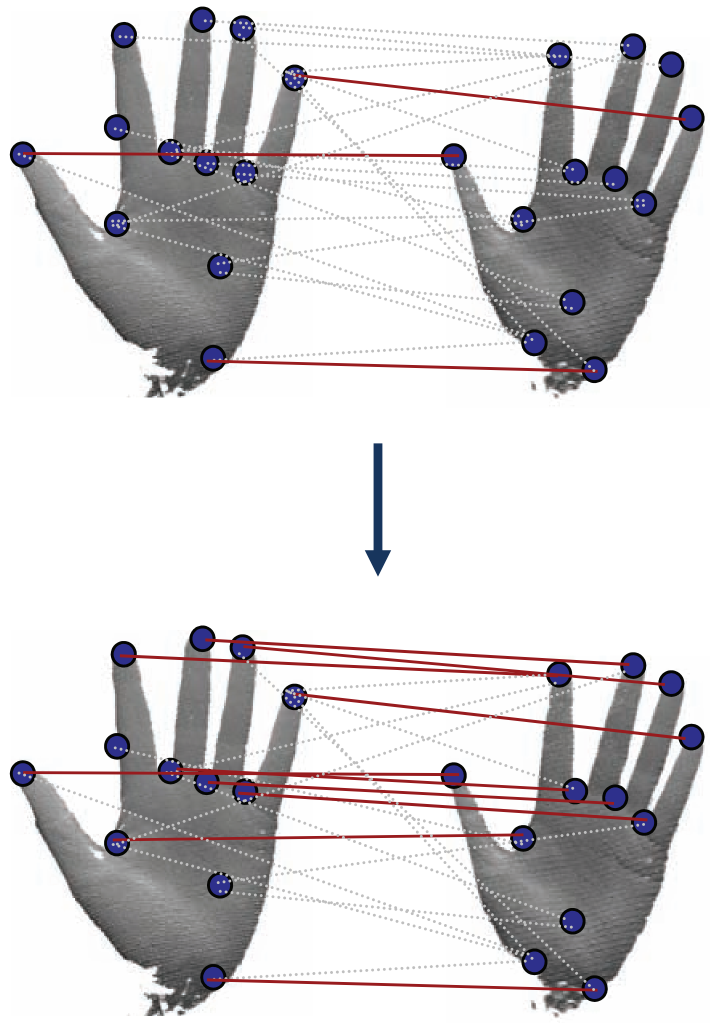
- **Feature points:**
  Geometric MLS-SIFT features [Li et al. 2005]
- **Descriptors:**
  Curvature & color ring histograms
- **Global Filtering:**
  Spectral matching



RANSAC Algorithm

RANSAC Idea

- Starting correspondence
- Add more that are consistent
  - Preserve intrinsic distances
- Importance sampling algorithm



Advantages

- Efficient (small initial set)
- General (arbitrary criteria)

Ransac Details

Algorithm: Simple Idea

- Select correspondences with probability proportional to their plausibility
- First correspondence: Descriptors
- Second: Preserve distance (distribution peaks)
- Third: Preserve distance (even fewer choices)

  ...

- Rapidly becomes deterministic
- Repeat multiple times (typ.: 100x)
  - Choose the largest solution (larges #correspondences)

Ransac Details

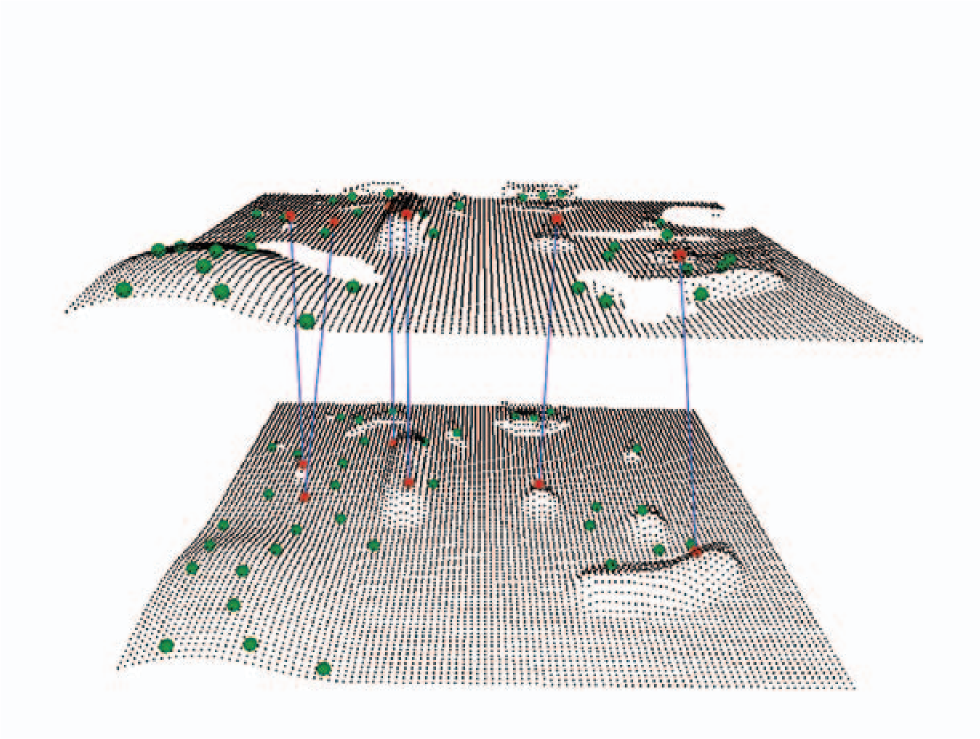
Provably Efficient:

- Optimal solution in expected $O(n^3 \log n)$ for n candidate correspondences and sphere topology
- Much faster in practice (using descriptors)

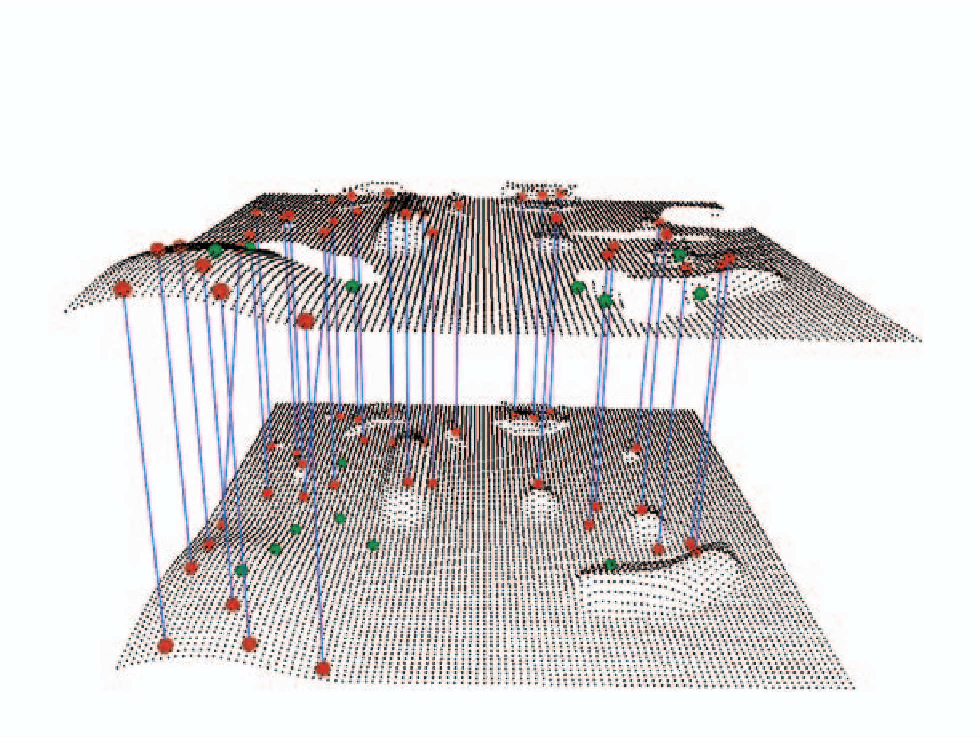
Flexible:

- In later iterations (> 3 correspondences), allow for outlier geodesics
- Can handle topological noise

Results: Topological Noise

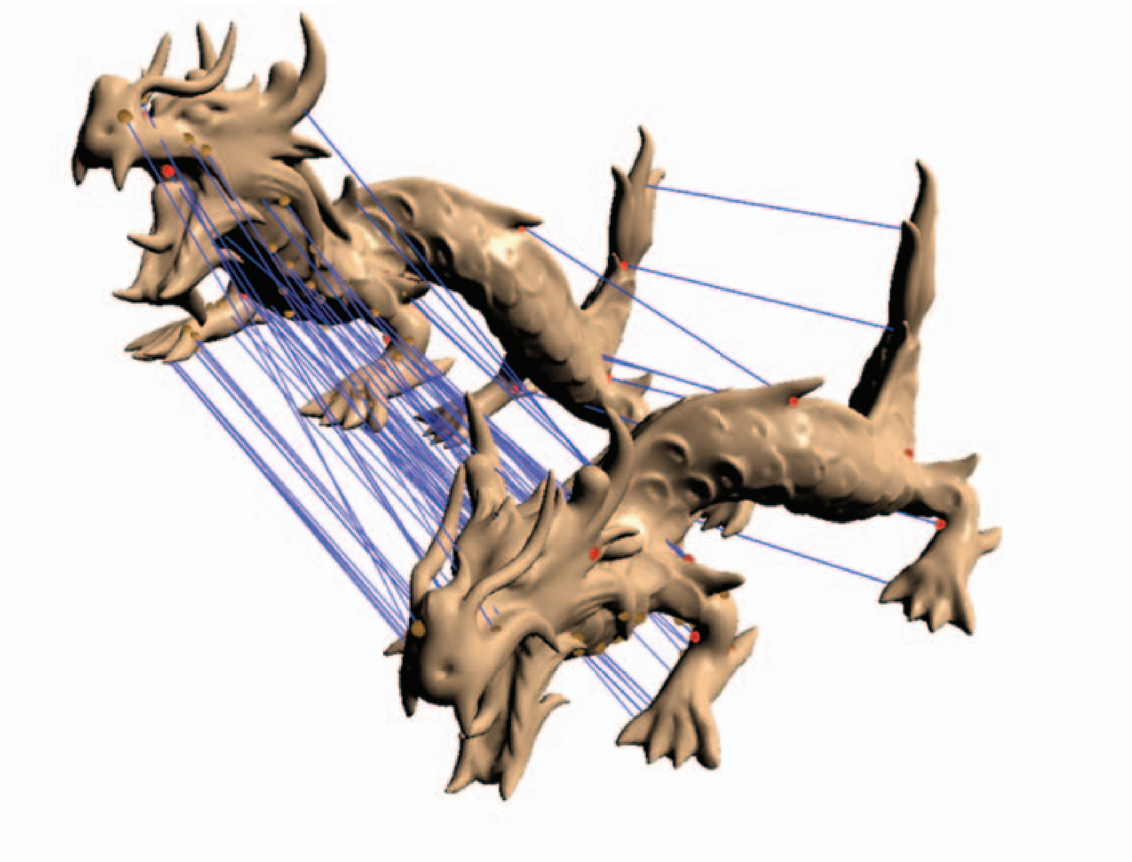


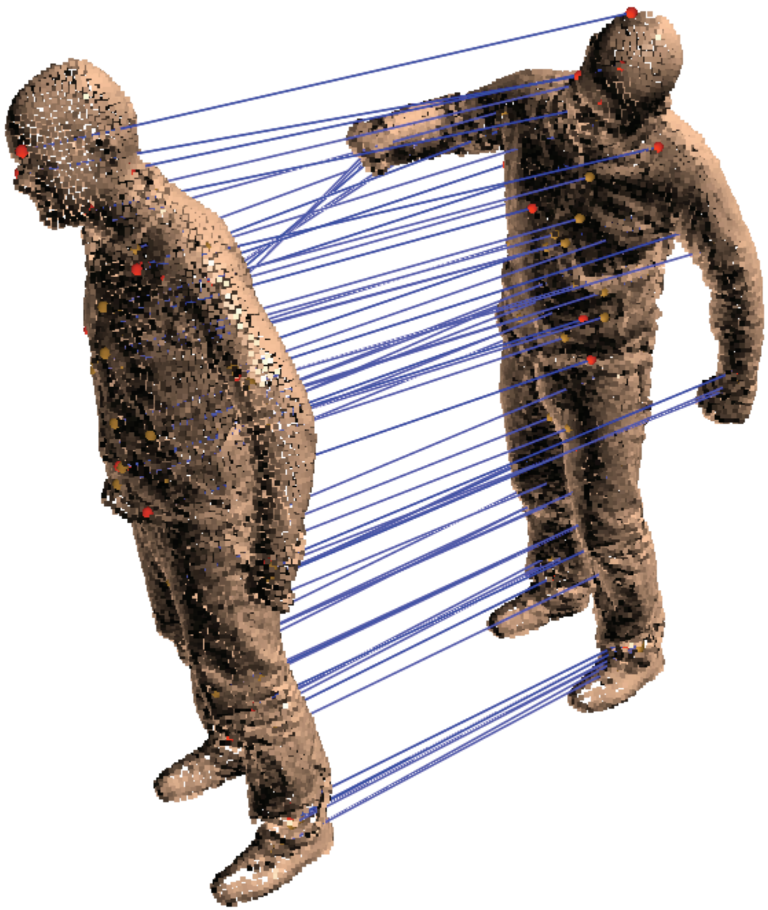
Spectral Quadratic Assignment
[Leordeanu et al. 05]



Ransac Algorithm
[Tevs et al. 09]

Results





[data sets: Stanford 3D Scanning Repository / Carsten Stoll]

Global Matching References

Global Matching References

- **Ahmed, N., Theobalt, C., Roessl, C., Thrun, S., Seidel, H.-P. (2008).** Dense Correspondence Finding for Parametrization-free Animation Reconstruction from Video. In: *Proc. Conf. Computer Vision and Pattern Recognition (CVPR)*.
- **Anguelov, D., Koller, D., Srinivasan, P., Thrun, S., Pang, H.-C., Davis, J. (2004).** The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces. In: *Advances in Neural Information Processing Systems (NIPS 2004)*.
- **Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J. (2005).** SCAPE: Shape Completion and Animation of People. In: *ACM Transactions on Graphics, Proc SIGGRAPH 2005*.
- **Bendels, G. H., Degener, P., Wahl, R., Körtgen, M., Klein, R. (2004).** Image-Based Registration of 3D-Range Data Using Feature Surface Elements. In: *VAST 2004, pp. 115-124*.
- **Bokeloh, M., Berner, A., Wand, M., Schilling, A., Seidel, H.-P. (2008).** Slippage Features. *Technical report (WSI-2008-03), University of Tubingen, WSI/GRIS*.
- **Chang, W., Zwicker, M. (2008).** Automatic Registration for Articulated Shapes. In: *Computer Graphics Forum (Proceedings of SGP 2008) 27(5)*.
- **Gatzke, T., Grimm, C., Garland, M., Zelinka, S. (2005).** Curvature Maps for Local Shape Comparison. *In: Proc. Shape Modeling International, pp. 244–256*.

Global Matching References

Global Matching References

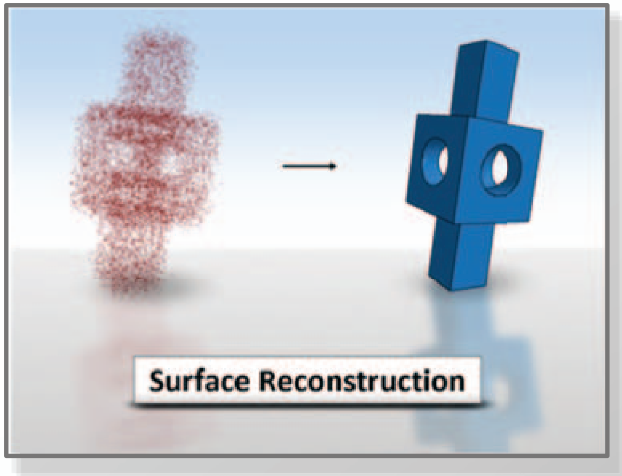
- **Gelfand, N., Mitra, N. J., Guibas, L. J., Pottmann, H. (2005).** Robust Global Registration. In: *Symposium on Geometry Processing, pp. 197–206*.
- **Huang, Q.-X., Adams, B., Wicke, M., Guibas, L. J. (2008).** Non-Rigid Registration Under Isometric Deformations. In: *Computer Graphics Forum (Proc. SGP 2008) 27(5)*.
- **Huang, Q.-X., Flцry, S., Gelfand, N., Hofer, M., Pottmann, H. (2006).** Reassembling Fractured Objects by Geometric Matching. In: *ACM Trans. Graphics 25(3), 569-578*.
- **Huber, D., Hebert, M. (2001).** Fully Automatic Registration of Multiple 3D Data Sets. In: *IEEE Computer Society Workshop on Computer Vision Beyond the Visible Spectrum (CVBVS 2001)*.
- **Li, X., Guskov, I. (2005).** Multiscale Features for Approximate Alignment of Point-based Surfaces.. In: *Symposium on Geometry Processing, pp. 217—226*.
- **Lowe, D. (2003).** Distinctive image features from scale-invariant keypoints. In: *International Journal of Computer Vision, pp. 91—110*.
- **Starck, J., Hilton, A. (2007).** Correspondence labelling for wide-timeframe free-form surface matching. In: *Proc. Int. Conf. Computer Vision 2007*.
- **Zhang, H., Sheffer, A., Cohen-Or, D., Zhou, Q., van Kaick, O., Tagliasacchi, A. (2008).** Deformation-Drive Shape Correspondence. In: *Computer Graphics Forum (Proc. Symposium on Geometry Processing 2008) 27(5), 1431-1439*.

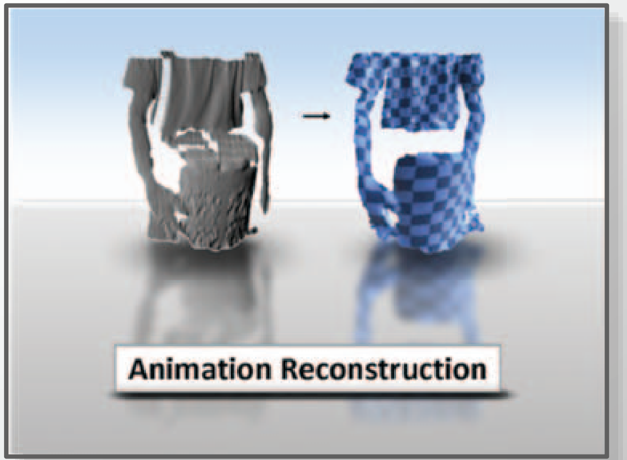
Global Matching References

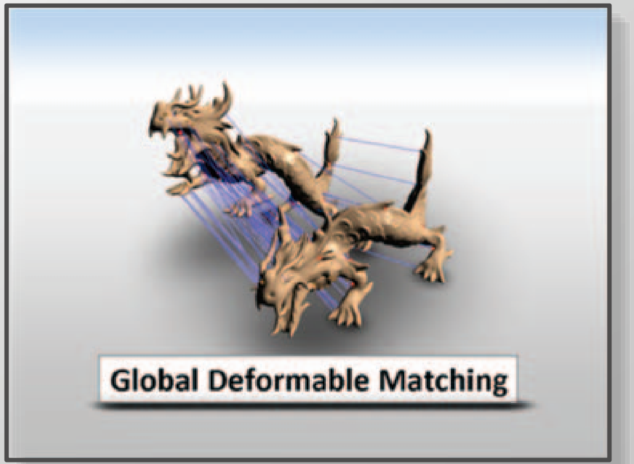
Presented in detail in this tutorial:

- **Leordeanu, M., Hebert, M. (2005).** A Spectral Technique for Correspondence Problems using Pairwise Constraints. In: *International Conference of Computer Vision (ICCV), pp. 1482—1489*.
- **Tevs, A., Bokeloh, M., Wand, M., Schilling, A., Seidel, H.-P. (2009).** Isometric Registration of Ambiguous and Partial Data. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09), 2009*.

Conclusions







Conclusions

We have discussed:

- Statistically motivated shape reconstruction
  - “Bayesian smoothing”
- Extension to animated data
  - “Animation reconstruction”
- Assembling a template from data only
  - “Urshape assembly”
- Attempts to make this globally convergent
  - “Feature tracking”, “Deformable matching”

Open Problems

Open Problems

- Global convergence
  - How to assemble an urshape from deformed pieces
  - No coherence assumptions
  - Topological and geometric noise
  - The building blocks seem to be in place already...
  - ...but a complete, general system is missing.
- Computer vision
  - Can we reconstruct from (multi-view-) video sequences?
  - Without template geometry?
  - (with template $\rightarrow$ next session)