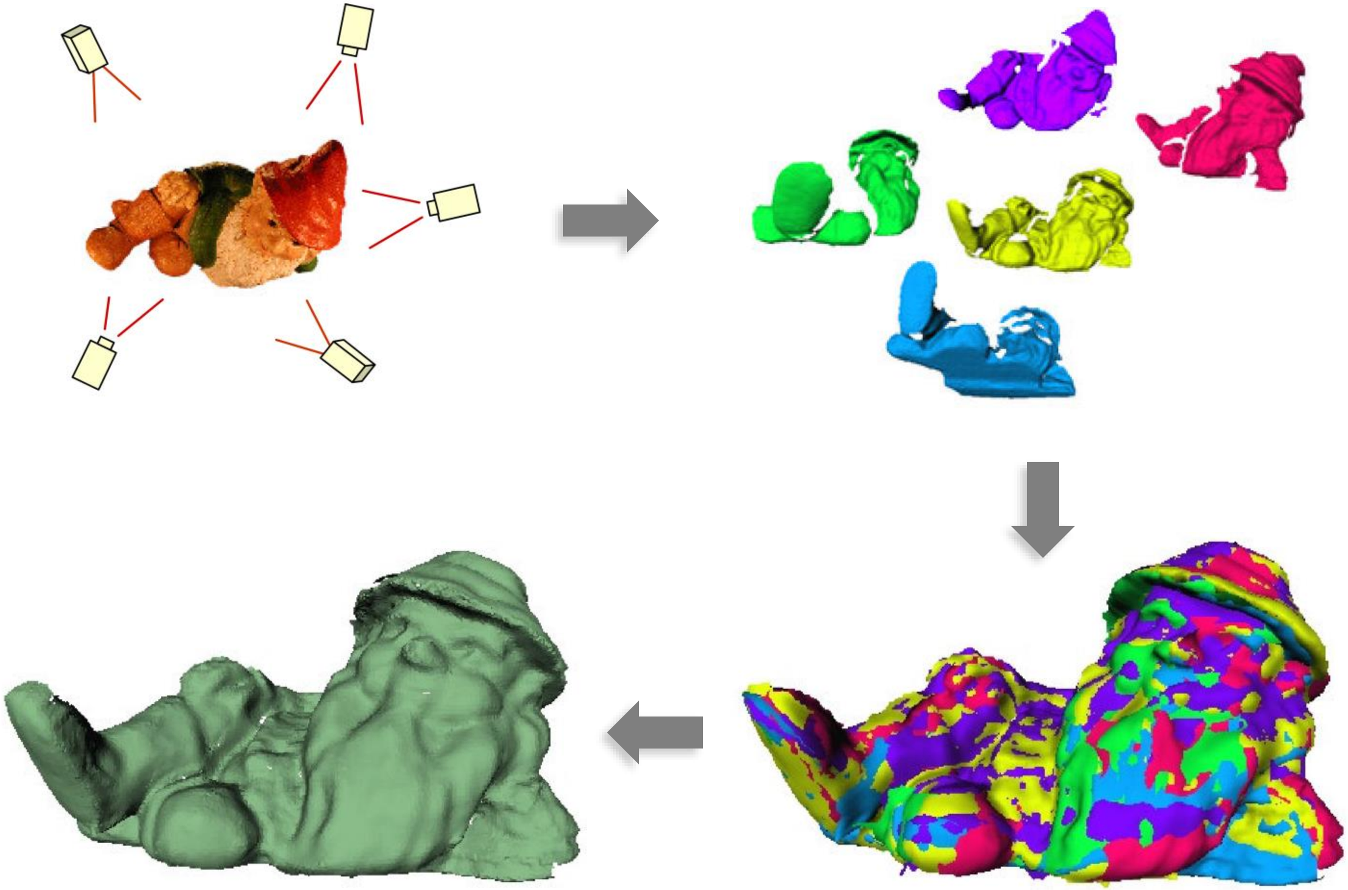
Shape Matching

Qixing Huang

Feb. 13th 2017

THE UNIVERSITY OF TEXAS AT AUSTIN

Geometry Reconstruction Pipeline



Binary Encoded Light Stripes

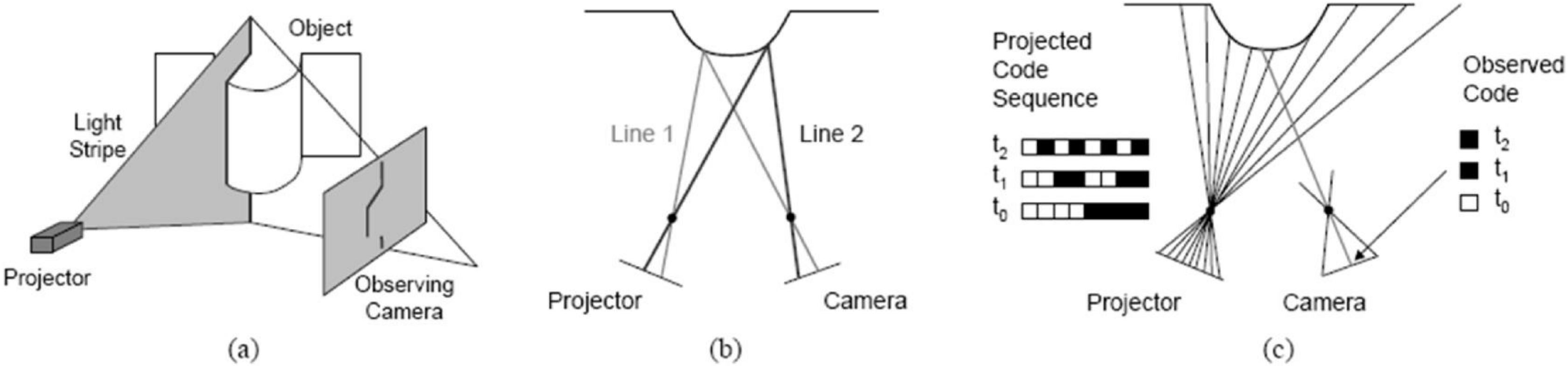
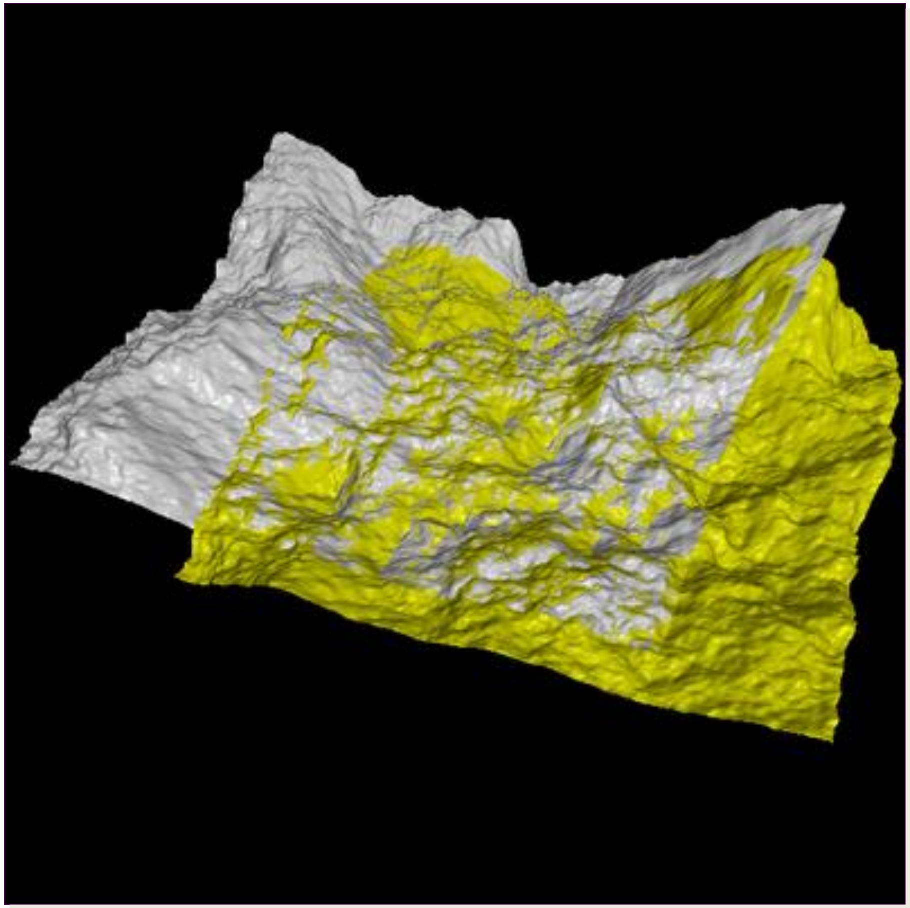


Figure 2. (a) Standard light sectioning with one light stripe. (b) Top view of light sectioning using more than one stripe. (c) Light sectioning using structured light.

- Set of light planes are projected into the scene
- Individual light planes are indexed by an encoding scheme for the light patterns
  - Obtained images are used to uniquely address the light plane corresponding to every image point

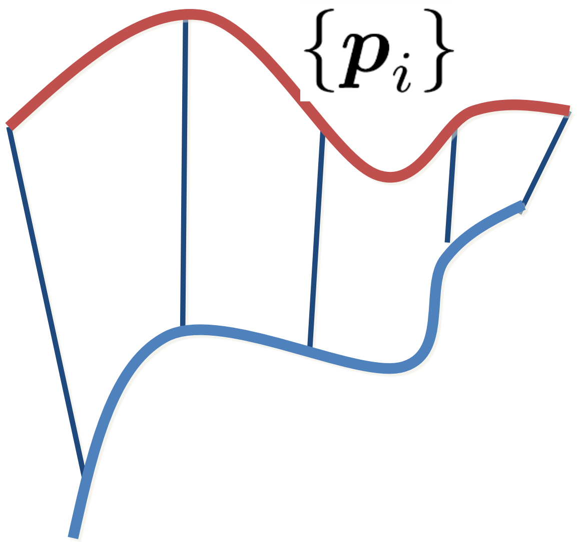
ICP for Pairwise Alignment

Pairwise Alignment



ICP
[Besel and Mckay' 92]

ICP Formulation



$$\underset{R,\boldsymbol{t}}{\text{minimize}} \quad \sum_{i=1}^{n} d^2(R(\boldsymbol{p}_i)+\boldsymbol{t}, T)$$

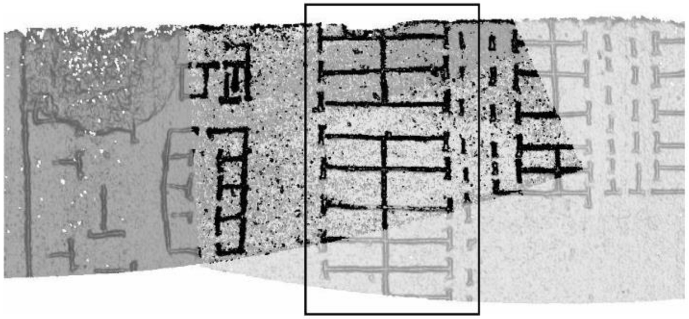
$$\underset{R,\boldsymbol{t}}{\text{minimize}} \quad \sum_{i=1}^{n} \min_{\boldsymbol{q}\in T} \ \|R(\boldsymbol{p}_i)+\boldsymbol{t}-\boldsymbol{q}\|^2$$

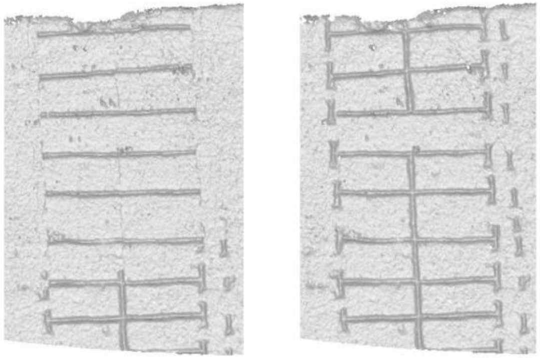
ICP Variants

- Point-plane distance [Chen and Medioni' 91]

$$\underset{R,\boldsymbol{t}}{\text{minimize}} \quad \sum_{i=1}^{n} \left((R\boldsymbol{p}_i + \boldsymbol{t} - \boldsymbol{q}_i)^T \boldsymbol{n}_i \right)^2$$

- Stable sampling [Gelfand et al. 03]



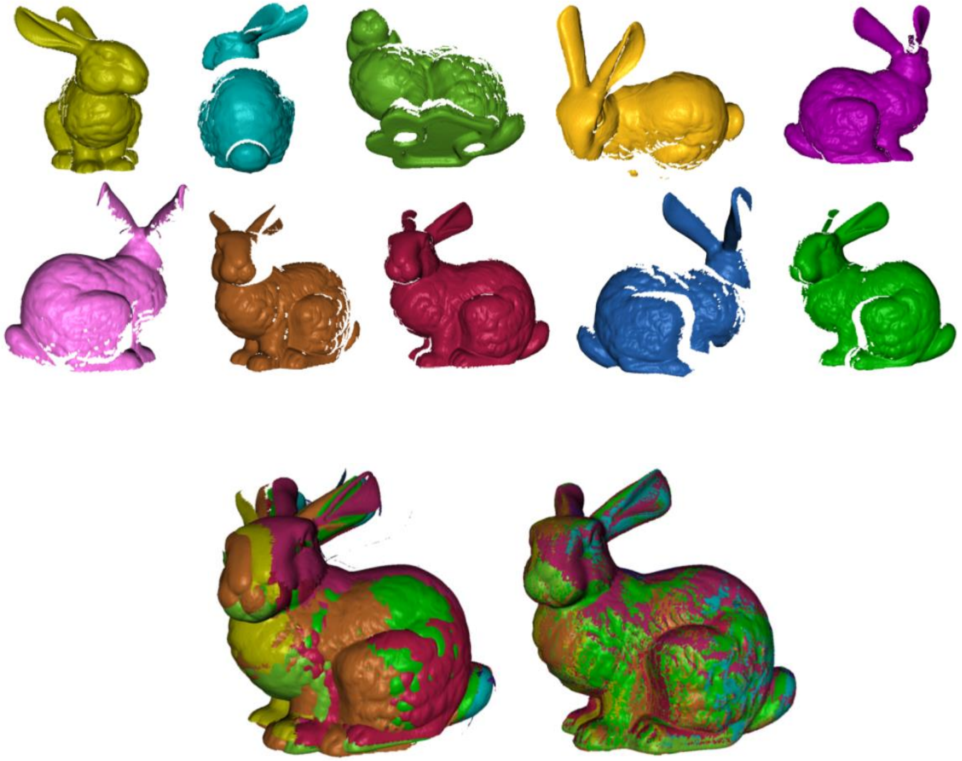


- Robust norm

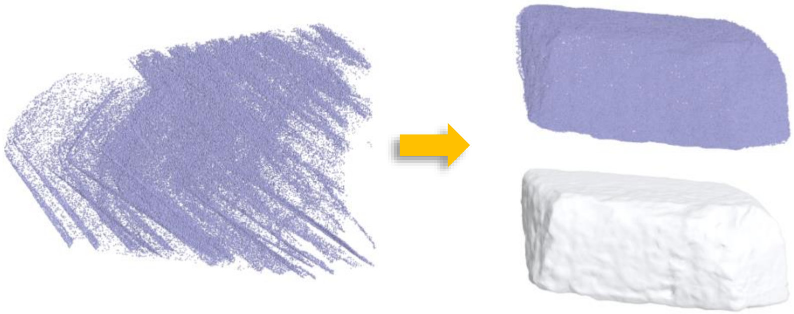
$$\underset{R,\boldsymbol{t}}{\text{minimize}} \quad \sum_{i=1}^{n} \min_{\boldsymbol{q} \in T} \| R(\boldsymbol{p}_i) + \boldsymbol{t} - \boldsymbol{q} \|$$

Today’s lecture

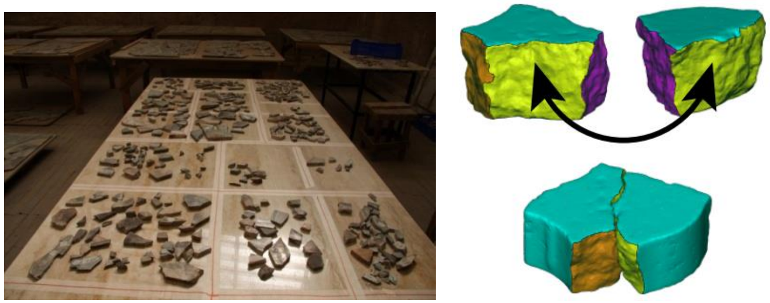
- Rigid matching --- how to generate the initial guess



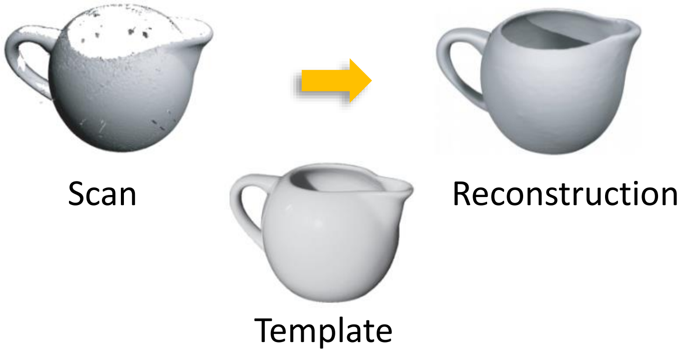
Applications



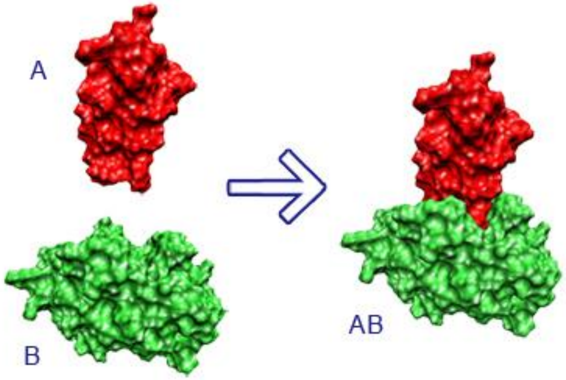
Surface reconstruction



Fragment assembly

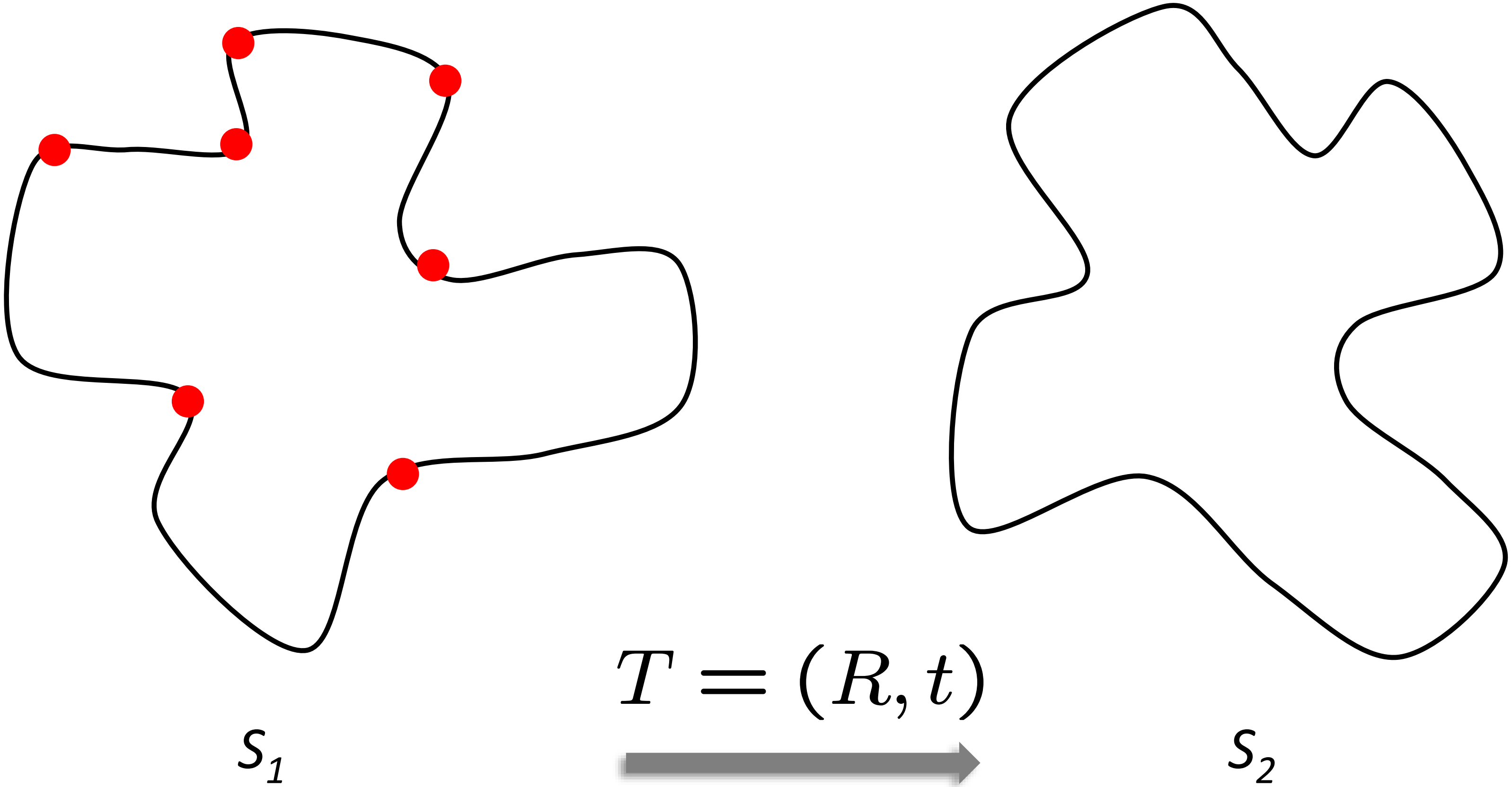


Object completion



Protein docking

Rigid Matching



$$T = (R, t)$$

S_1

S_2

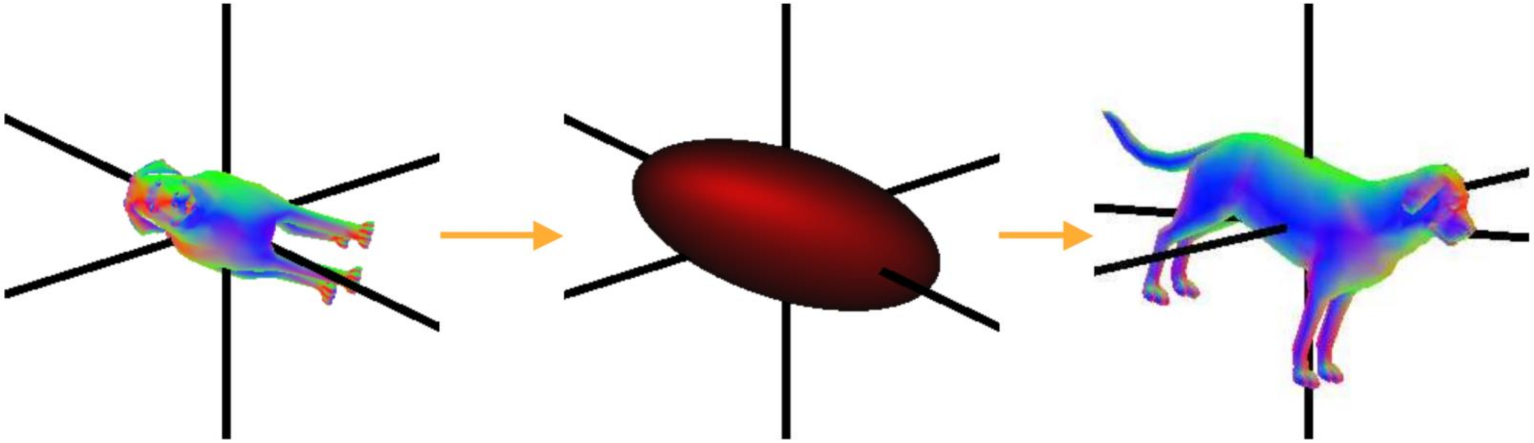
Rotation and translation

Approach --- PCA

- Use PCA to place models into a canonical coordinate frame

Covariance matrix computation

Principal Axis alignment



Principal axis computation

- Given a collection of points $\{p_i\}$, form the co-variance matrix:

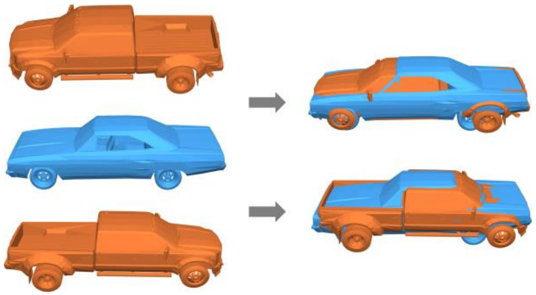
$$\mathbf{c} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{p}_i$$

$$C = \frac{1}{N}\sum_{i=1}^{N} \mathbf{p}_i \mathbf{p}_i^T - \mathbf{c}\mathbf{c}^T$$

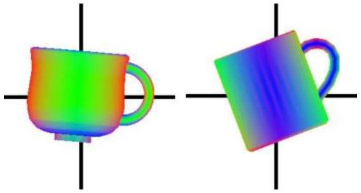
- Compute eigenvectors of matrix C

Issues with PCA

- Principal axes are not oriented

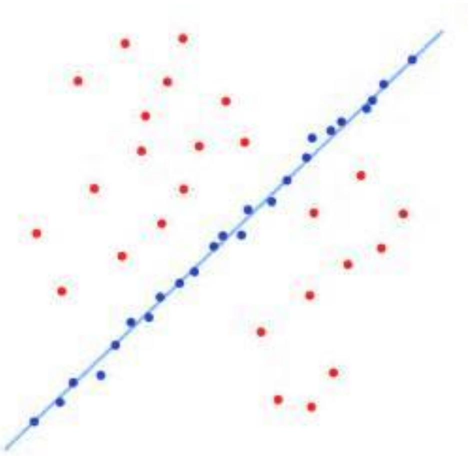


- Axes are unstable when principal values are similar

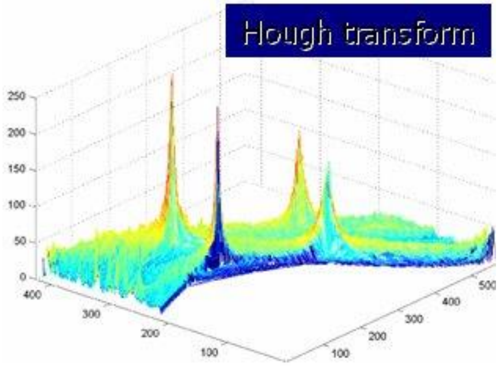


- Partial similarity

Approaches --- correspondence-based



RANSAC



Voting

	1	3	5	2	4
1	*1*	*1*	*1*	*0*	*0*
3	*1*	*1*	*1*	*0*	*0*
5	*1*	*1*	*1*	*0*	*0*
2	*0*	*0*	*0*	*1*	*1*
4	*0*	*0*	*0*	*1*	*1*

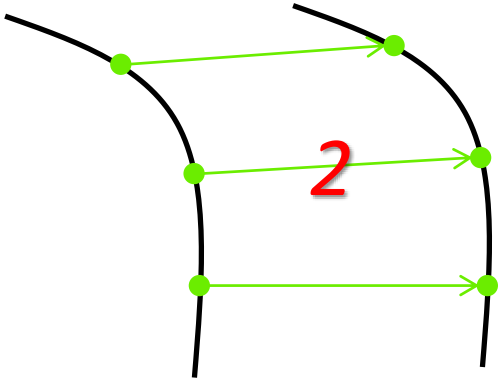
Spectral matching

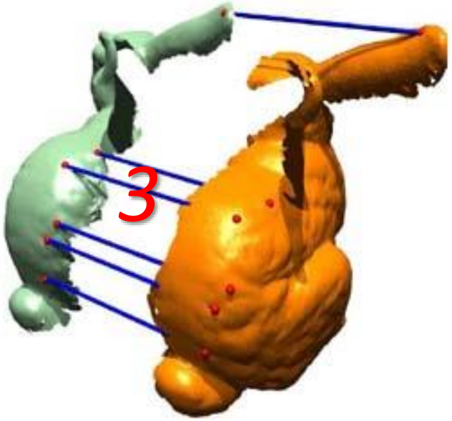
Partial similarity

Stable

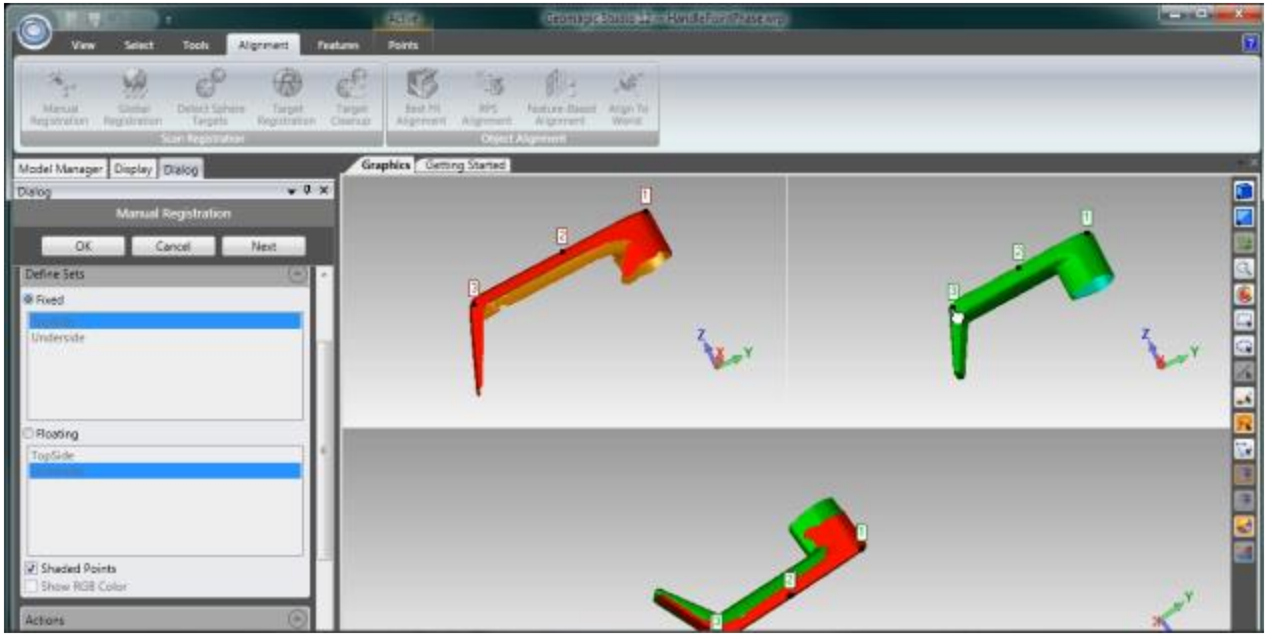
RANSAC

- How many point-pairs specify a rigid transform?
  - In R^2?
  - In R^3?
- Additional constraints?
  - Distance preserving
  - Stability?





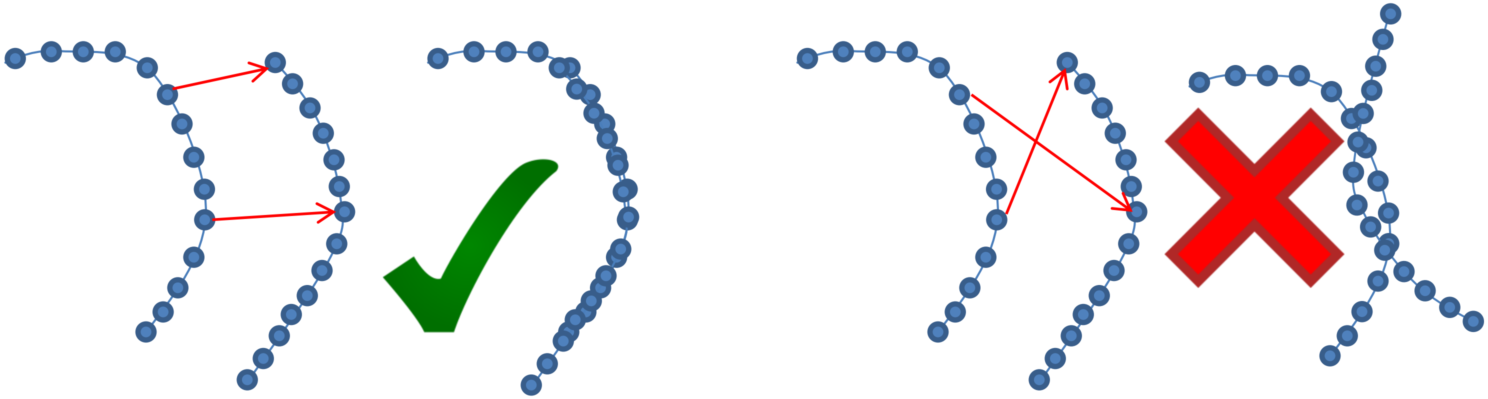
Software



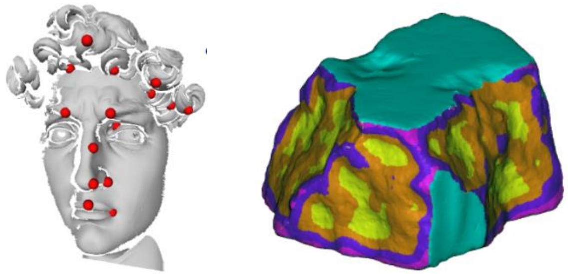
Geomagic

RANSAC

- Preprocessing: sample each object
- Recursion:
  - Step I: Sample three (two) pairs, check distance constraints
  - Step II: Fit a rigid transform
  - Step III: Check how many point pairs agree. If above threshold, terminates; otherwise goes to Step I



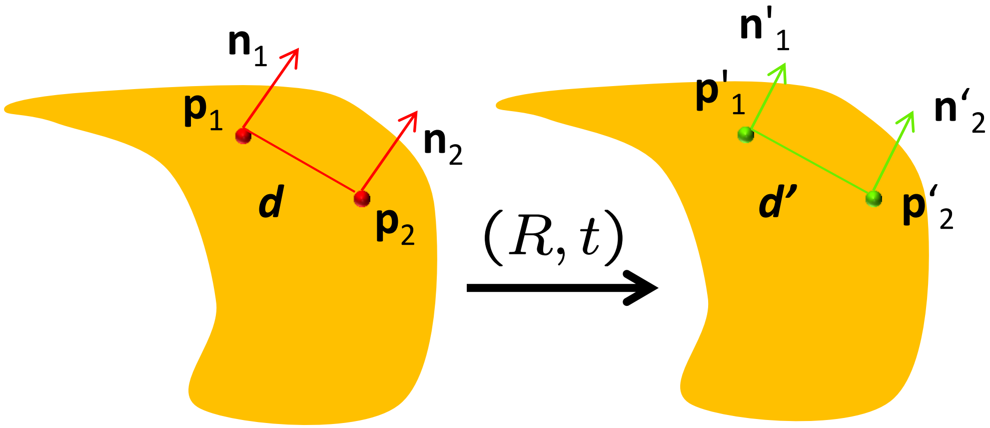
RANSAC --- facts



- Sampling
  - Feature point detection
    [Gelfand et al. 05, Huang et al. 06]
- Correspondences
  - Use feature descriptors $m << O(n^2)$
  - The candidate correspondences
  - Denote the success rate $p \approx \frac{n}{m}$
- *Basic* analysis
  - The probability of having a valid triplet p^3
  - The probability of having a valid triplet in N trials is $1-(1-p^3)^N$

RANSAC+

- How many surfel (position + normal) correspondences specify a rigid transform?



Constraints:

1. $\|\mathbf{p}_1 - \mathbf{p}_2\| \approx \|\mathbf{p}'_1 - \mathbf{p}'_2\|$
2. $\angle(\mathbf{n}_1, \mathbf{d}) = \angle(\mathbf{n}'_1, \mathbf{d}')$
3. $\angle(\mathbf{n}_2, \mathbf{d}) = \angle(\mathbf{n}'_2, \mathbf{d}')$
4. $\angle(\mathbf{n}_1, \mathbf{n}_2) = \angle(\mathbf{n}'_1, \mathbf{n}'_2)$

$$\mathbf{t} = \frac{\mathbf{p}'_1 + \mathbf{p}'_2}{2} - \frac{\mathbf{p}_1 + \mathbf{p}_2}{2}$$

$$[\mathbf{n}_1, \mathbf{n}_2, d] \xrightarrow{R} [\mathbf{n}'_1, \mathbf{n}'_2, \mathbf{d}']$$

Reduce the number of trials from O(m^3) to O(m^2)

Success rate:

$$1 - (1 - p^2)^N$$

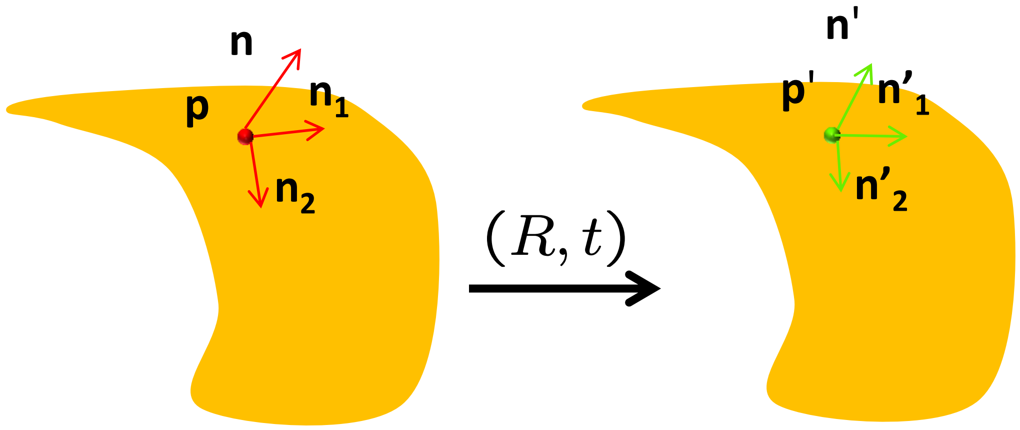
RANSAC++

- How many frame correspondences specify a rigid transform?
  - Principal curvatures
  - Local PCA

Further reduce the number of trials from O(m^2) to O(m)

Success rate:

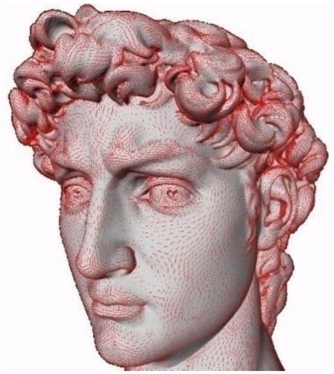
$$1-(1-p)^N$$



$$\mathbf{t} = \mathbf{p}' - \mathbf{p}$$

$$R(\mathbf{n}, \mathbf{n}_1, \mathbf{n}_2) \approx (\mathbf{n}', \mathbf{n}'_1, \mathbf{n}'_2)$$

Principal directions are unreliable



Implementation details

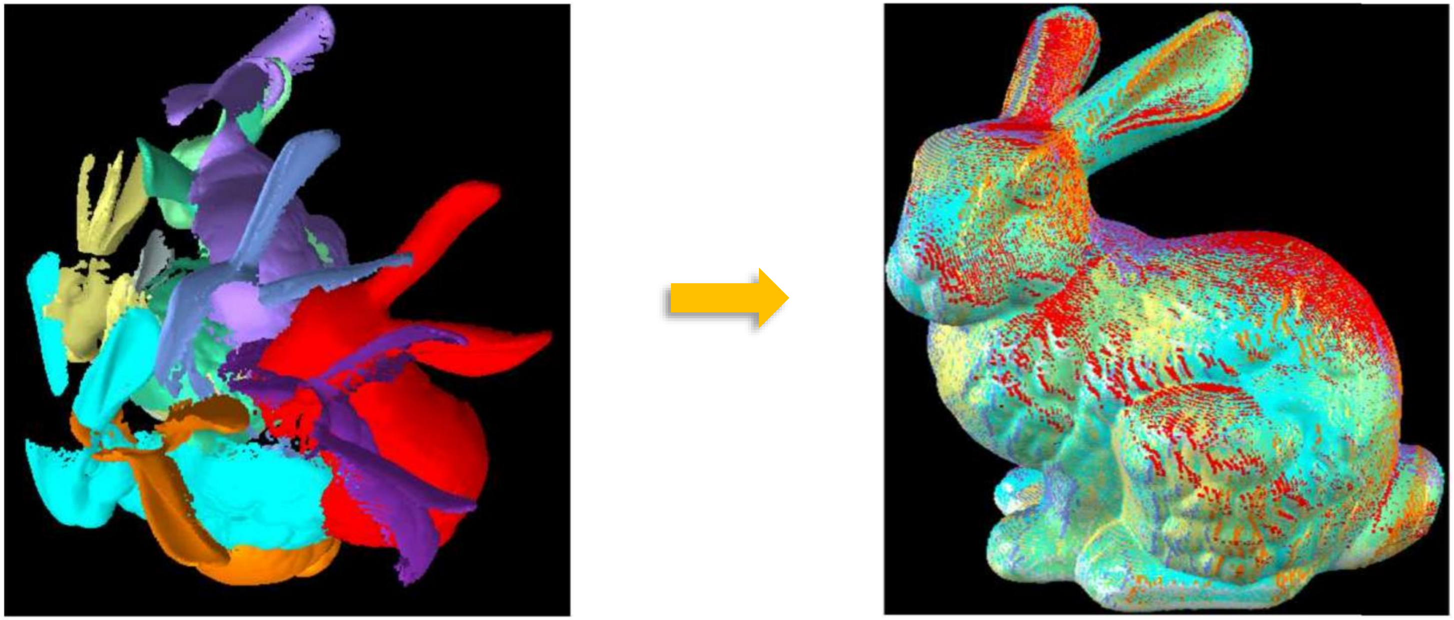
- Use feature points

3D SIFT features

Patch features

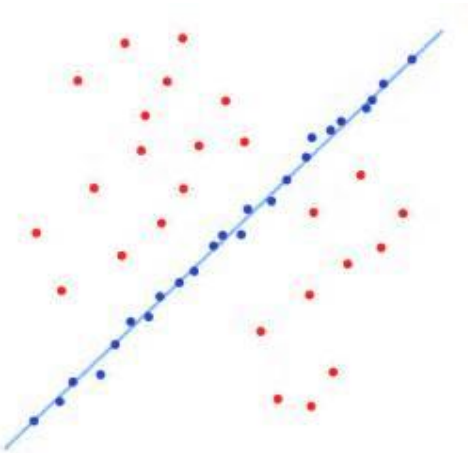
Implementation details

- Parameters?

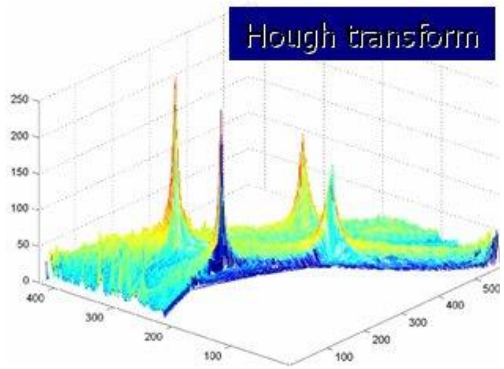


Learn parameters from registered scans

Approaches --- correspondence-based



RANSAC



Voting

	1	3	5	2	4
1	*1*	*1*	*1*	*0*	*0*
3	*1*	*1*	*1*	*0*	*0*
5	*1*	*1*	*1*	*0*	*0*
2	*0*	*0*	*0*	*1*	*1*
4	*0*	*0*	*0*	*1*	*1*

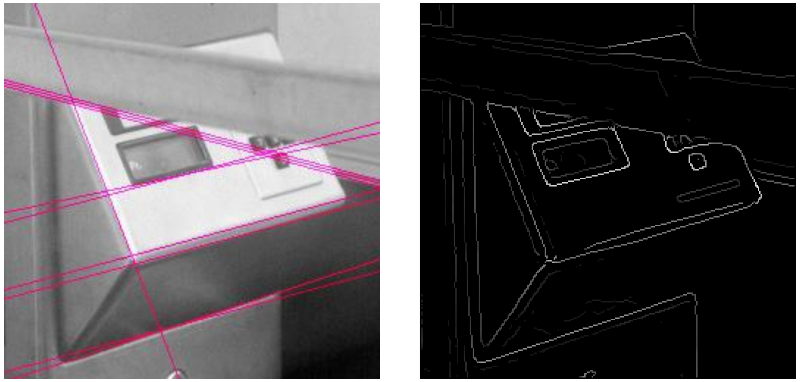
Spectral matching

Partial similarity

Stable

Hough transform for line fitting

- Line detection in an image
  - what is the line?
  - How many lines?
  - Point-line associations?



- **Hough Transform** is a voting technique that can be used to answer all of these questions
  - Record vote for each possible line on which each edge point lies
  - Look for lines that get many votes.

Voting

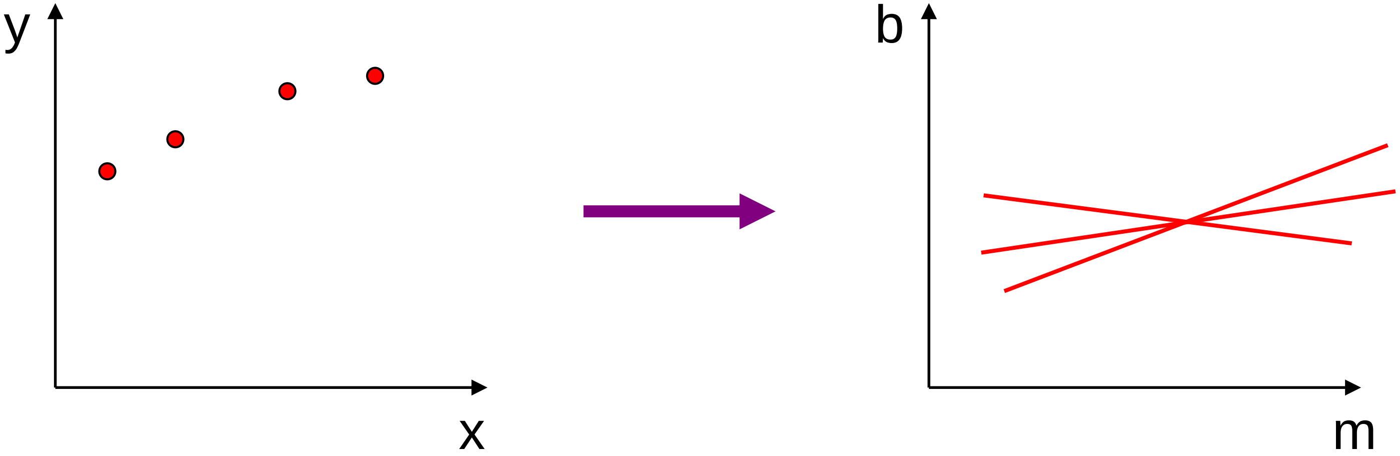
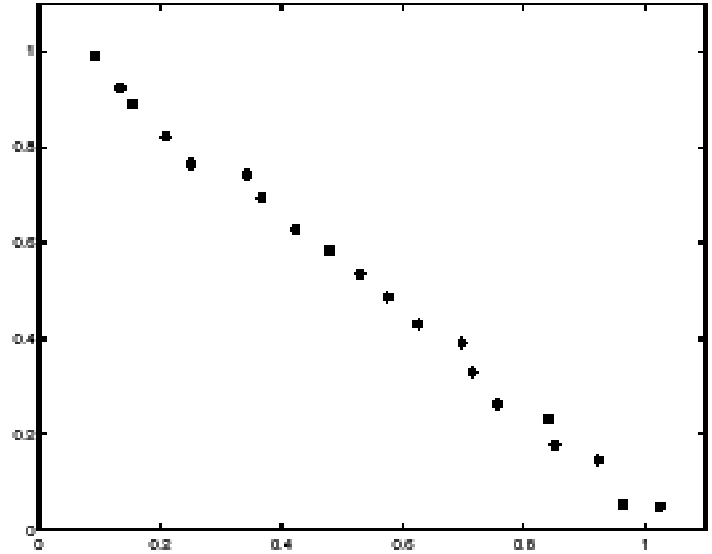
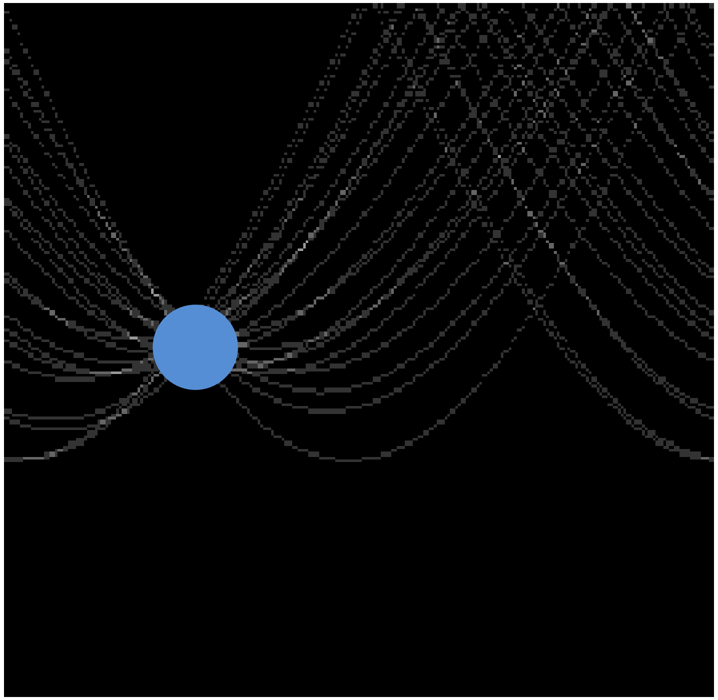


image space

Hough (parameter) space

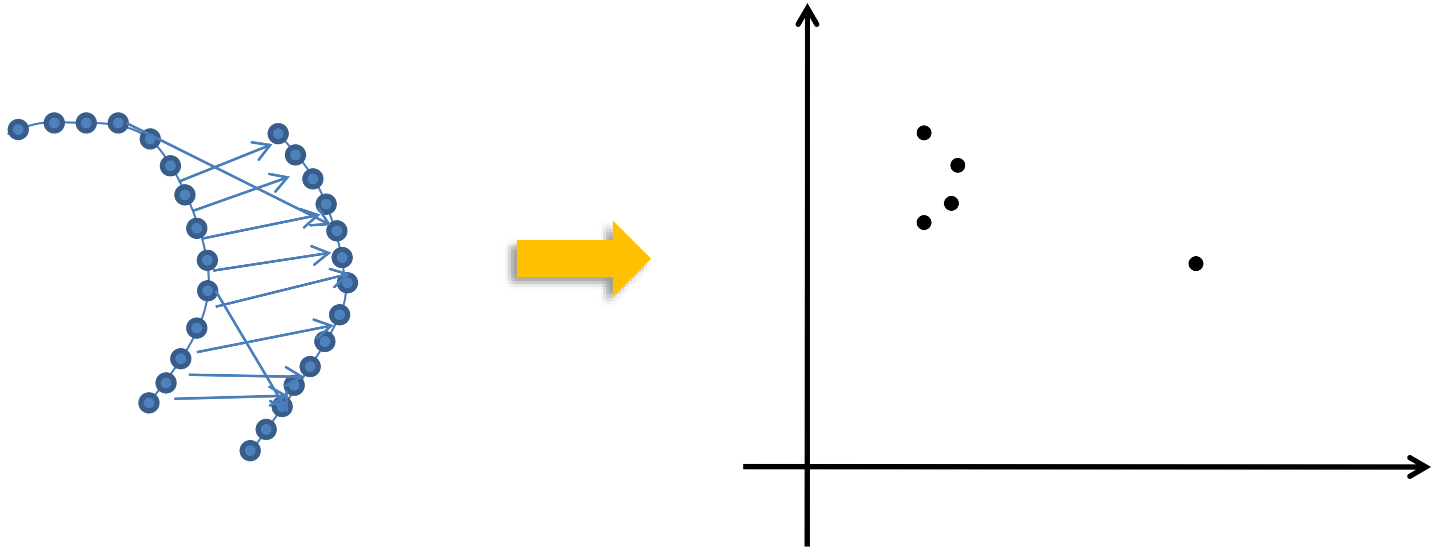
Clustering



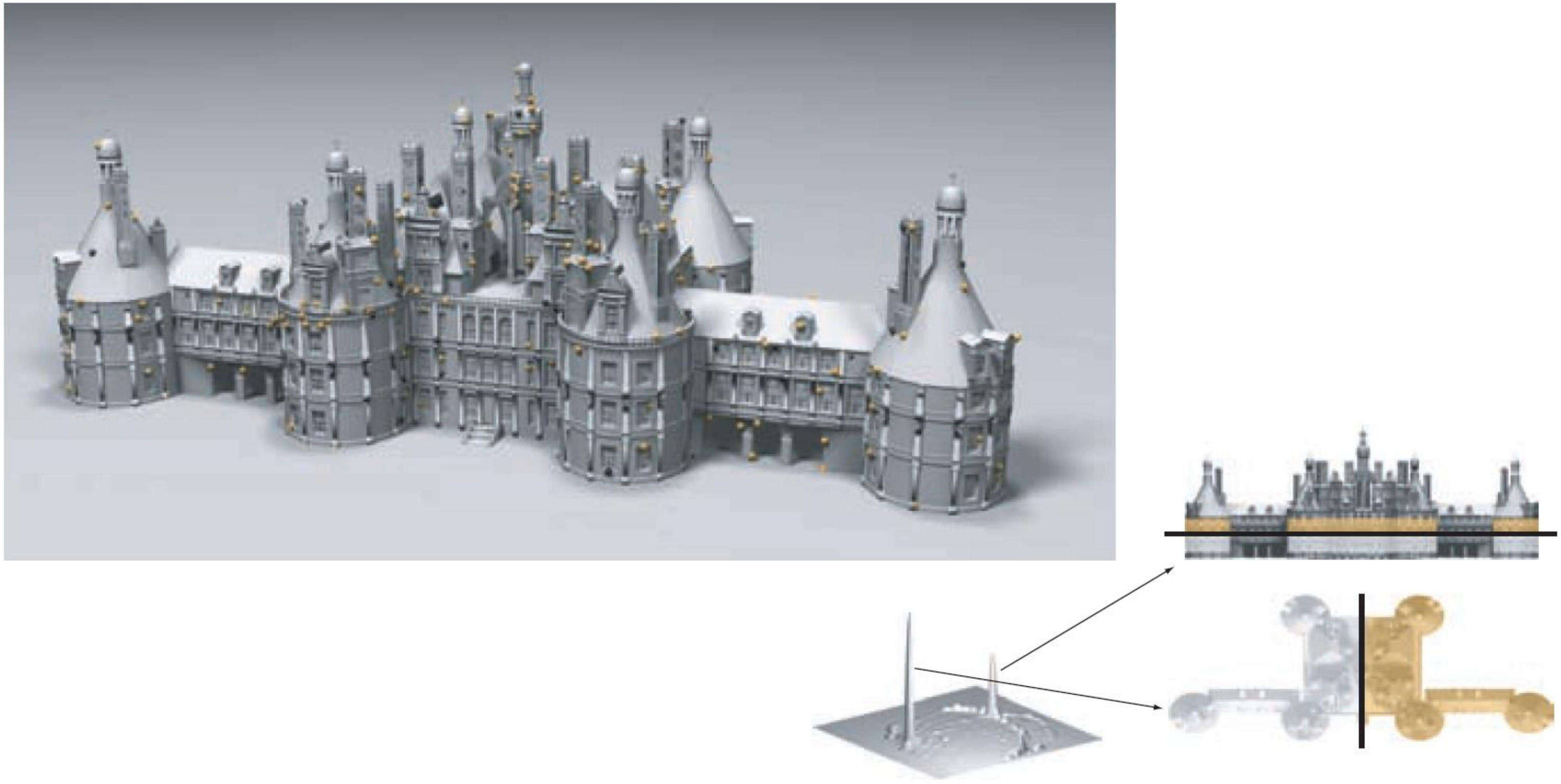


Rigid matching

- Rigid transform detection from feature correspondences

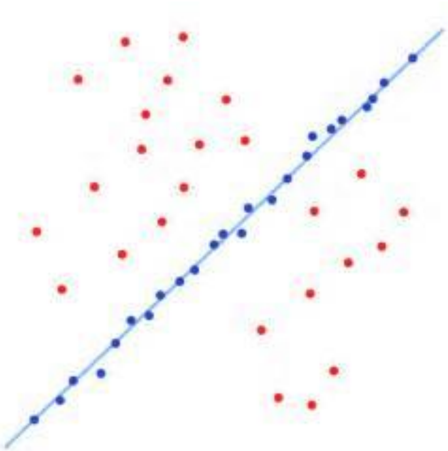


Symmetry detection

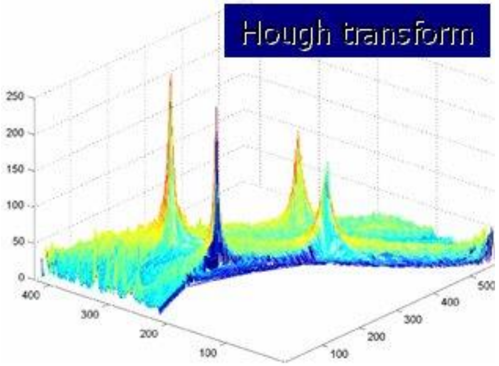


Partial and Approximate Symmetry Detection for 3D Geometry, N. Mitra, L. Guibas, and M. Pauly, SIGGRAPH’ 06

Approaches --- correspondence-based



RANSAC



Voting

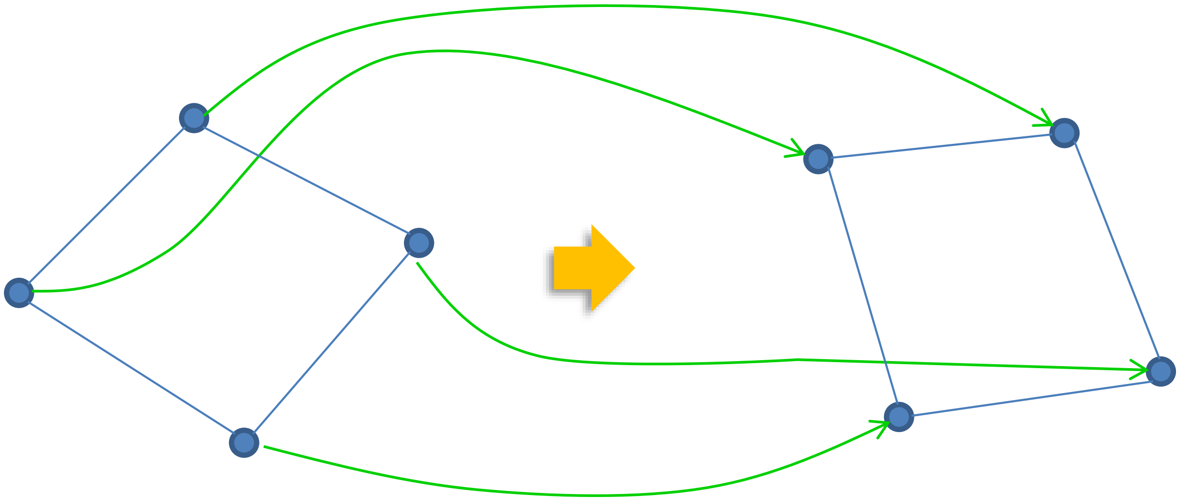
	1	3	5	2	4
1	*1*	*1*	*1*	*0*	*0*
3	*1*	*1*	*1*	*0*	*0*
5	*1*	*1*	*1*	*0*	*0*
2	*0*	*0*	*0*	*1*	*1*
4	*0*	*0*	*0*	*1*	*1*

Spectral matching

Partial similarity

Stable

Distance preservation ⇔ Rigidity?

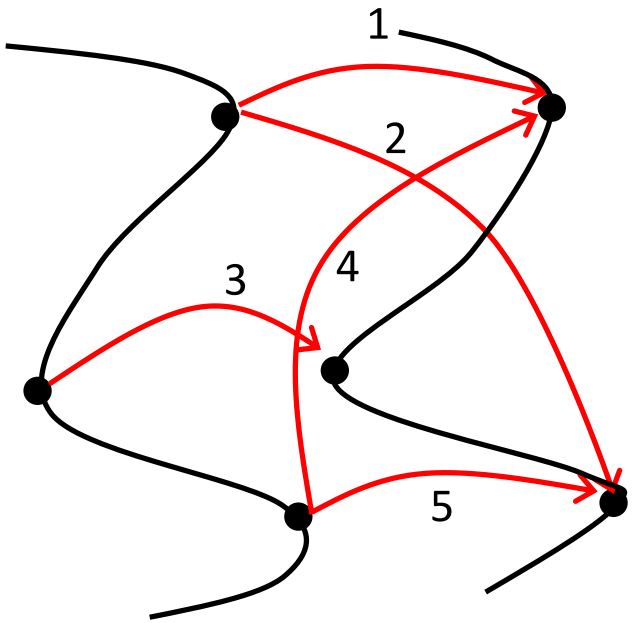


$$\|\mathbf{p}_i - \mathbf{p}_j\| = \|\phi(\mathbf{p}_i) - \phi(\mathbf{p}_j)\| \Longleftrightarrow \phi(\mathbf{p}_i) = R \cdot \mathbf{p}_i + \mathbf{t}$$

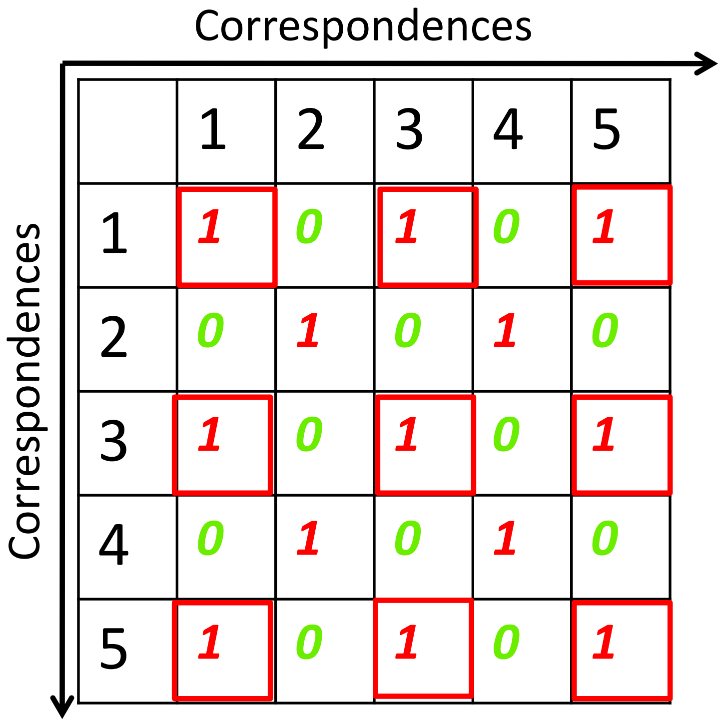
$$\det(R) = -1$$

Spectral approach

0: Inconsistent, 1: Consistent



Correspondences



	1	2	3	4	5
1	1	0	1	0	1
2	0	1	0	1	0
3	1	0	1	0	1
4	0	1	0	1	0
5	1	0	1	0	1

Consistency matrix

A Spectral Technique for Correspondence Problems using Pairwise Constraints, M. Leordeanu and M. Hebert, ICCV 2005

Clique extraction

	1	2	3	4	5
1	*1*	*0*	*1*	*0*	*1*
2	*0*	*1*	*0*	*1*	*0*
3	*1*	*0*	*1*	*0*	*1*
4	*0*	*1*	*0*	*1*	*0*
5	*1*	*0*	*1*	*0*	*1*

Consistency matrix

permute

	1	3	5	2	4
1	*1*	*1*	*1*	*0*	*0*
3	*1*	*1*	*1*	*0*	*0*
5	*1*	*1*	*1*	*0*	*0*
2	*0*	*0*	*0*	*1*	*1*
4	*0*	*0*	*0*	*1*	*1*

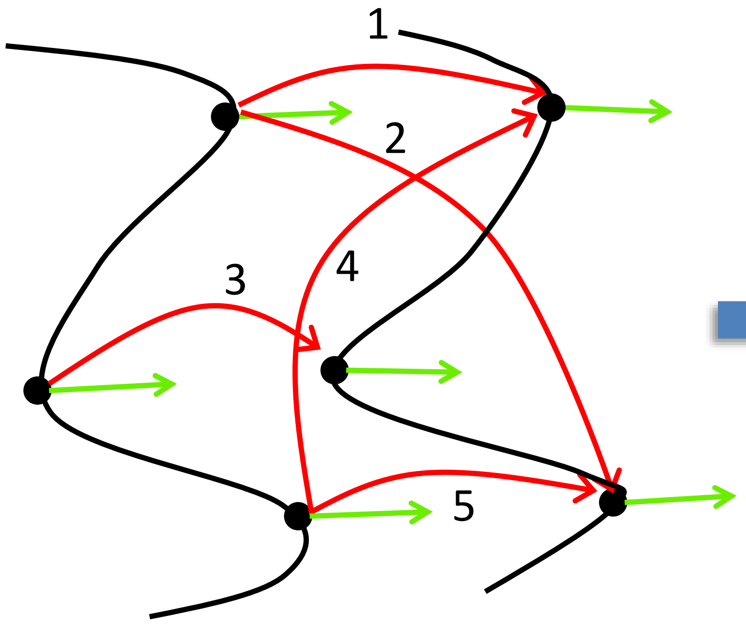
Consistency matrix

Algorithm

- Step 1: Compute the maximum eigenvector ***v*** of ***C***
- Step 2: Sort the vertices based on magnitude of v and initialize the cluster
- Step 3: Incrementally insert vertices while checking the clique constraint
- Step 4: Stop if the size of the cluster is small, otherwise accept the cluster and go to Step 1

Incorporate normals/frames

Clustering becomes easier



Correspondences

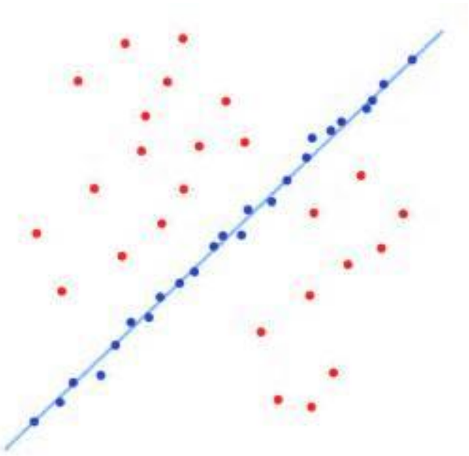
0: Inconsistent, 1: Consistent

Correspondences

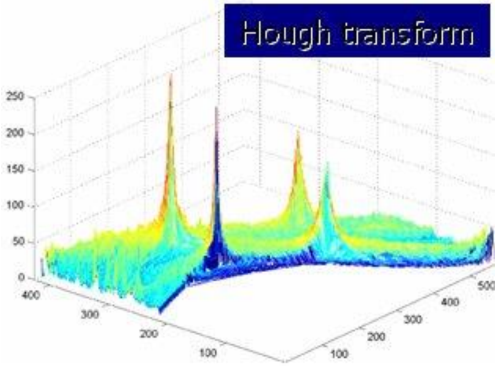
	1	3	5	2	4
1	*1*	*1*	*1*	*0*	*0*
3	*1*	*1*	*1*	*0*	*0*
5	*1*	*1*	*1*	*0*	*0*
2	*0*	*0*	*0*	*1*	*0*
4	*0*	*0*	*0*	*0*	*1*

Consistency matrix

Approaches --- correspondence-based



RANSAC



Voting

	1	3	5	2	4
1	*1*	*1*	*1*	*0*	*0*
3	*1*	*1*	*1*	*0*	*0*
5	*1*	*1*	*1*	*0*	*0*
2	*0*	*0*	*0*	*1*	*1*
4	*0*	*0*	*0*	*1*	*1*

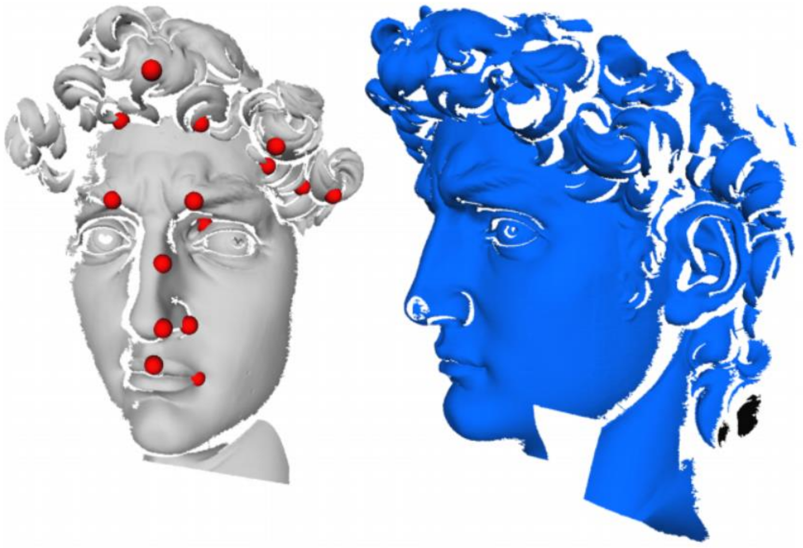
Spectral matching

Partial similarity

Stable

Post-processing

- Refine the match via ICP



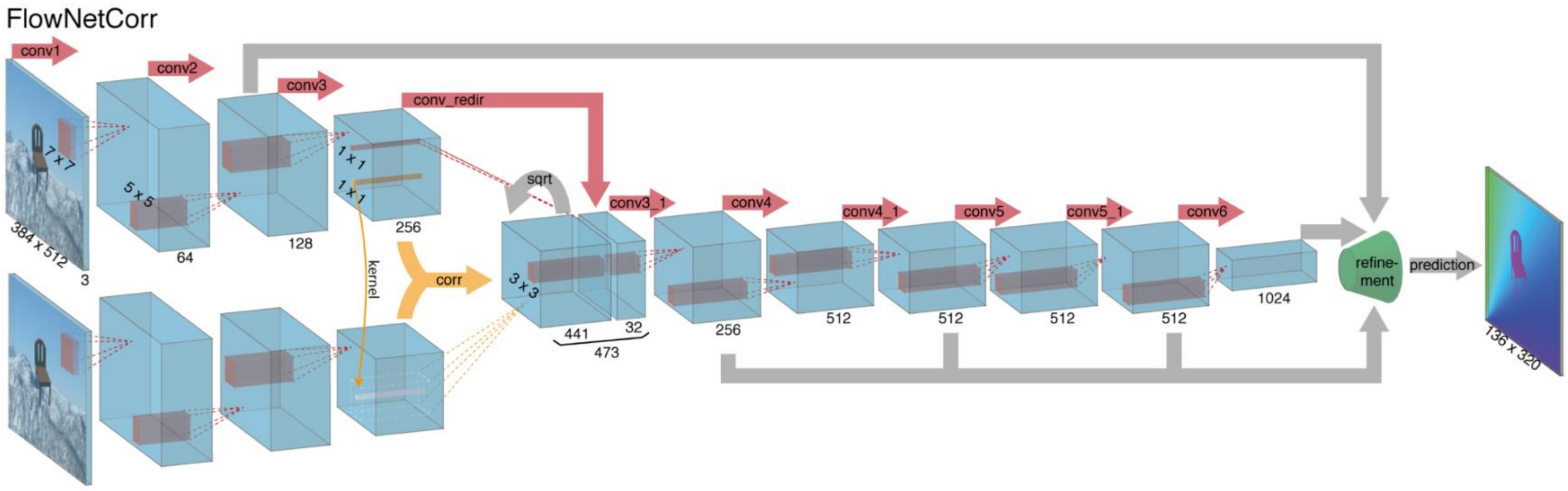
Input

After matching

After registration

Pair-wise matching in the deep learning era

[Fischer et al. 15]



Feature extraction

Matching

FlowNet: Learning Optical Flow with Convolutional Networks. ICCV' 15