

HSM3D: Feature-Less Global 6DOF Scan-Matching in the Hough/Radon Domain

Andrea Censi

Caltech



Stefano Carpin

UCMERCED

3D data alignment how-to

- The data is 3D, but the *sensor motion* is 2D?
(e.g., 3D scanner on a car)
 - ◆ \Rightarrow Then adapt known 2D methods to work on 2D slices of data.
 - ◆ see Olson, “Real-Time Correlative Scan Matching” SaD10.2

- Do you have a good initial guess?
 - ◆ \Rightarrow Then use an iterative registration procedure.
 - ◆ see next talk, “... a comparison of ICP and NDT”

- Are there consistent, regular, features? (e.g., planar surfaces)
 - ◆ \Rightarrow Then use some feature matching scheme (RANSAC, etc.)

- No features, unstructured data, no initial guess, general 3D motion?
 - ◆ keep listening...

“Local” and “global” algorithms

■ Challenges of **local** algorithms:

- ◆ How to establish correspondences?
- ◆ How to speed up convergence?
- ◆ How to avoid local minima? (re-weighting, outliers, etc.)

■ Challenges of **global** 6DOF algorithms:

- ◆ Must deal with multiple solutions.
- ◆ Must reduce the 6DOF problem \Rightarrow use of invariants
- ◆ If using voting procedures, trade-off with cell size:
 - bigger: more robust, faster
 - smaller: more precise
- ◆ “Fancy” math is sometimes necessary (no Euler angles...)

Related work

- Makadia et al, “Fully automatic registration of 3D point clouds” Computer Vision and Pattern Recognition, 2006.
 - ◆ Creates a *translation-invariant* statistics
 - histogram of surfaces having the same orientation
 - ◆ This decouples rotation and translation
 - same general idea of HSM3D
 - ◆ **Cons:** needs smooth surfaces, fails in simple cases (spheres).
- Reyes, Medioni, Bayro, “Registration of 3D points using geometric algebra and tensor voting,” Int. J. Computer Vision, 2007.
 - ◆ **Cons:** *really* hard to understand
 - ◆ **Cons:** complexity quadratic in the number of points
- Huge literature of (tangentially) related problems (registration of volumes, etc.).

HSM3D

3D version of the algorithm presented in: *Censi, Grisetti, Iocchi, "Scan matching in the Hough domain", ICRA 2005.*

- **Design goal:** fast global estimate(s) with limited precision.
- **Input:** two sets of (oriented) 3D points.
- **Output:** ordered set of candidate roto-translations.
- **Overview:**
 1. Compute the 3D Hough (Radon) transform.
 - ◆ Uses whole Hough transform, not just the peaks (no planes necessary!)
 2. Compute the two Hough Spectra.
 - ◆ This will be our translation-invariant.
 3. Compute rotations hypotheses by matching the two spectra.
 4. Given the rotation, find the translation
 5. Rank the hypotheses.

HSM3D

3D version of the algorithm presented in: *Censi, Grisetti, Iocchi, "Scan matching in the Hough domain", ICRA 2005.*

- **Design goal:** fast global estimate(s) with limited precision.
- **Input:** two sets of (oriented) 3D points.
- **Output:** ordered set of candidate roto-translations.
- **Overview:**
 1. Compute the 3D Hough (Radon) transform.
 - ◆ Uses whole Hough transform, not just the peaks (no planes necessary!)
 2. Compute the two Hough Spectra.
 - ◆ This will be our translation-invariant.
 3. Compute rotations hypotheses by matching the two spectra.
 4. Given the rotation, find the translation (**the easy part**)
 5. Rank the hypotheses.

HSM3D

3D version of the algorithm presented in: *Censi, Grisetti, Iocchi, "Scan matching in the Hough domain", ICRA 2005.*

- **Design goal:** fast global estimate(s) with limited precision.
- **Input:** two sets of (oriented) 3D points.
- **Output:** ordered set of candidate roto-translations.
- **Overview:**
 1. Compute the 3D Hough (Radon) transform.
 - ◆ Uses whole Hough transform, not just the peaks (no planes necessary!)
 2. Compute the two Hough Spectra.
 - ◆ This will be our translation-invariant.
 3. Compute rotations hypotheses by matching the two spectra.
 4. Given the rotation, find the translation (**the easy part**)
 5. Rank the hypotheses. (**scenario dependent**)

HSM3D

3D version of the algorithm presented in: *Censi, Grisetti, Iocchi, "Scan matching in the Hough domain", ICRA 2005.*

- **Design goal:** fast global estimate(s) with limited precision.
- **Input:** two sets of (oriented) 3D points.
- **Output:** ordered set of candidate roto-translations.
- **Overview:**
 1. **Compute the 3D Hough (Radon) transform.**
 - ◆ Uses whole Hough transform, not just the peaks (no planes necessary!)
 2. **Compute the two Hough Spectra.**
 - ◆ This will be our translation-invariant.
 3. **Compute rotations hypotheses by matching the two spectra.**
 4. Given the rotation, find the translation (**the easy part**)
 5. Rank the hypotheses. (**scenario dependent**)

Hough/Radon transform

Definition. The Hough Transform maps a 3D image into a function defined on $\mathbb{S}^2 \times \mathbb{R}$.

$$\text{HT} : (\mathbb{R}^3 \rightarrow \mathbb{R}^+) \rightarrow (\mathbb{S}^2 \times \mathbb{R} \rightarrow \mathbb{R}^+)$$

Given a kernel $k : \mathbb{R} \rightarrow \mathbb{R}^+$, the value of the HT of i at point $(s, \rho) \in \mathbb{S}^2 \times \mathbb{R}$ is defined as:

$$\text{HT}[i](s, \rho) = \int_{\mathbb{R}^3} i(v) k(\langle s, v \rangle - \rho) dv \quad (1)$$

- $\mathbb{R}^3 \rightarrow \mathbb{R}^+$ is, generalizing, the point cloud.
- $\mathbb{S}^2 \times \mathbb{R}$ is the sets of planes (direction $s \in \mathbb{S}^2$, distance from origin $\rho \in \mathbb{R}$)
- It is not necessary for the points to contain planes for the HT to be meaningful.
- We also define an *oriented* HT transform if points come with a normal.

Hough Spectrum

The Hough Spectrum (HS) maps a 3D “image” to a function defined on the sphere:

$$\text{HS} : (\mathbb{R}^3 \rightarrow \mathbb{R}^+) \rightarrow (\mathbb{S}^2 \rightarrow \mathbb{R}^+)$$

$$\text{HS}[i](s) = \|\text{HT}[i](s, \cdot)\|_2$$

Properties:

- the HS is **invariant to translations** of the input:

$$\text{HS}[i \circ t] = \text{HS}[i]$$

- the HS is **rotated by a rotation** of the input:

$$\text{HS}[i \circ r] = \text{HS}[i] \circ r$$

Hough Spectrum

The Hough Spectrum (HS) maps a 3D “image” to a function defined on the sphere:

$$\text{HS} : (\mathbb{R}^3 \rightarrow \mathbb{R}^+) \rightarrow (\mathbb{S}^2 \rightarrow \mathbb{R}^+)$$

$$\text{HS}[i](s) = \|\text{HT}[i](s, \cdot)\|_2$$

Properties:

- the HS is **invariant to translations** of the input:

$$\text{HS}[i \circ t] = \text{HS}[i]$$

\Rightarrow decouples rotation and translations

- the HS is **rotated by a rotation** of the input:

$$\text{HS}[i \circ r] = \text{HS}[i] \circ r$$

Hough Spectrum

The Hough Spectrum (HS) maps a 3D “image” to a function defined on the sphere:

$$\text{HS} : (\mathbb{R}^3 \rightarrow \mathbb{R}^+) \rightarrow (\mathbb{S}^2 \rightarrow \mathbb{R}^+)$$

$$\text{HS}[i](s) = \|\text{HT}[i](s, \cdot)\|_2$$

Properties:

- the HS is **invariant to translations** of the input:

$$\text{HS}[i \circ t] = \text{HS}[i]$$

⇒ decouples rotation and translations

- the HS is **rotated by a rotation** of the input:

$$\text{HS}[i \circ r] = \text{HS}[i] \circ r$$

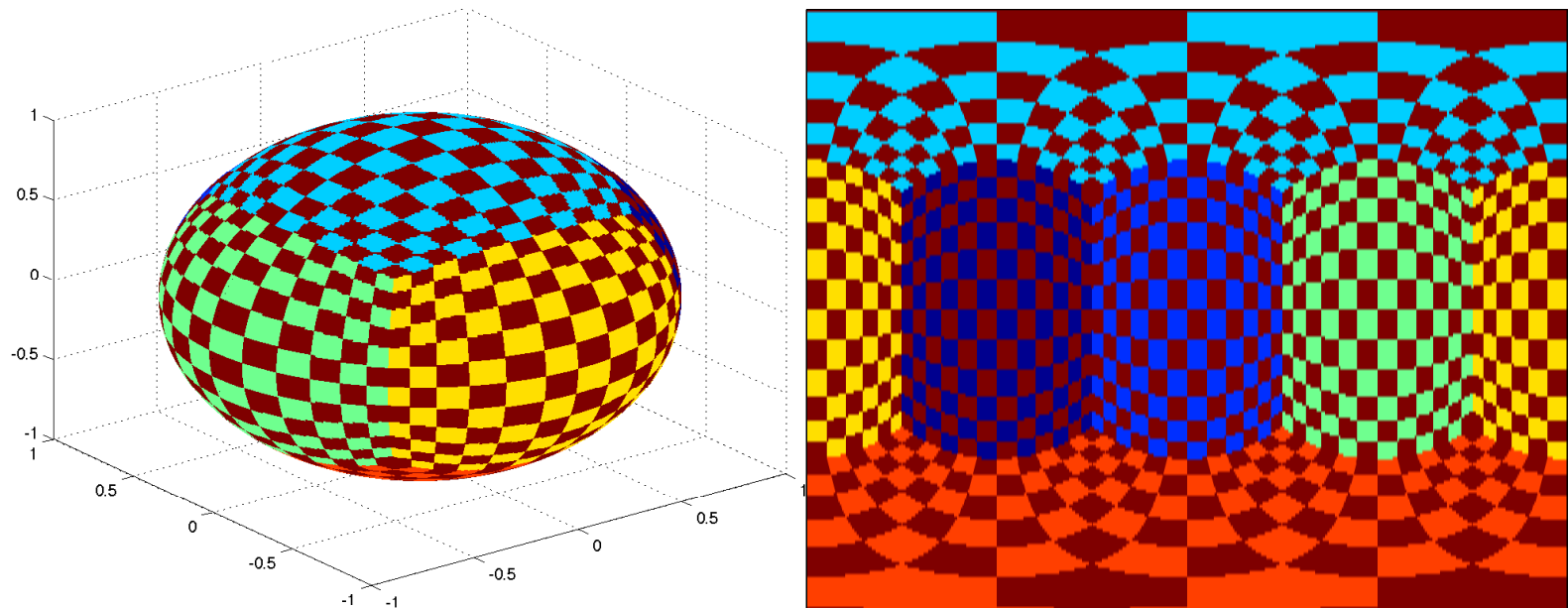
⇒ allows to find r given the two spectra

Representing functions on S^2

- The Hough Spectrum is a function defined on the sphere:

$$\text{HS} : S^2 \rightarrow \mathbb{R}^+$$

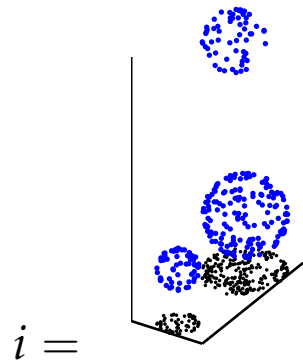
- The trickiest part of the code is handling this representation.



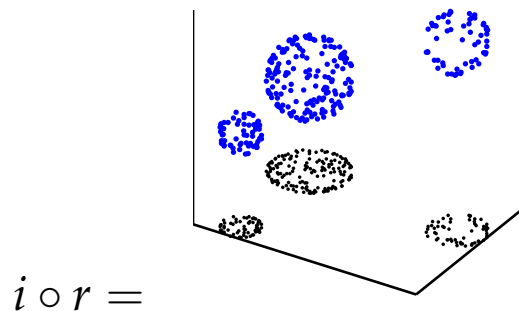
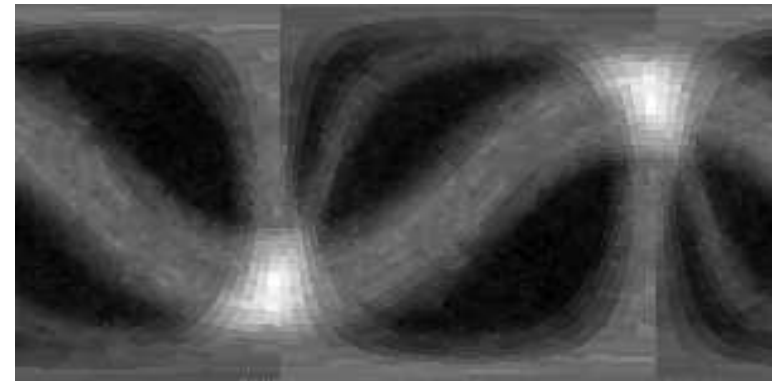
- In the next slides, Hough Spectra are shown projected to a cylinder.

Examples of spectra

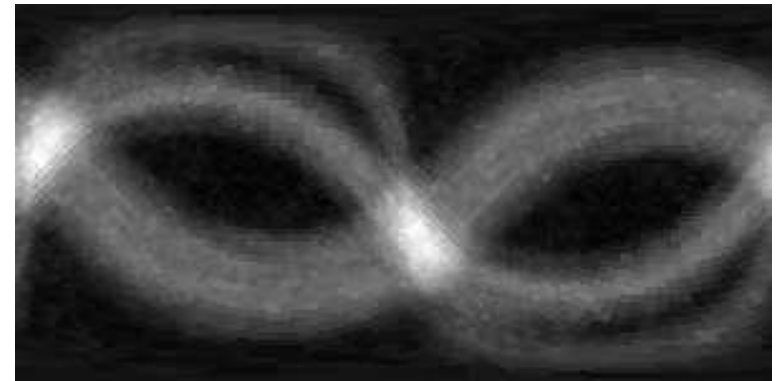
- Hough Spectra of points clouds sampled from 3 spheres, two different positions.
- Hough Spectrum is the same, just rotated.



HS(i) =



HS($i \circ r$) =



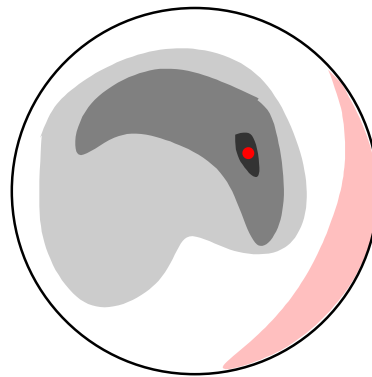
- Alignment is possible even though the surface histogram is flat
 - ◆ (Makadia's algorithm would fail)

Rotational matching problems

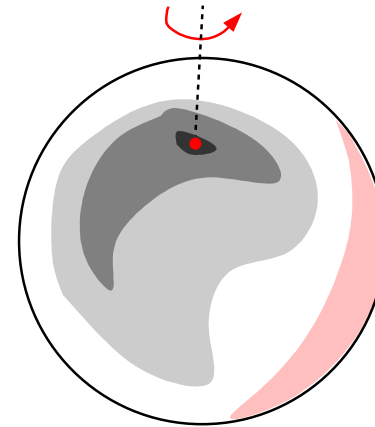
- **Problem:** let f_1, f_2 functions on the sphere, and $f_1 = f_2 \circ r$. Find r .
- The “elegant” way, Fourier Transform on $SO(3)$ [Chirikjian], has complexity $O(nR^2) + O(R^3 \log^2 R)$.
- HSM3D uses a simple method to align two spherical images:
 - ◆ Align two candidate points (maxima)
 - ◆ Look for remaining rotation by cross correlation of the spheres.



1. first input



2. second input



... partially aligned

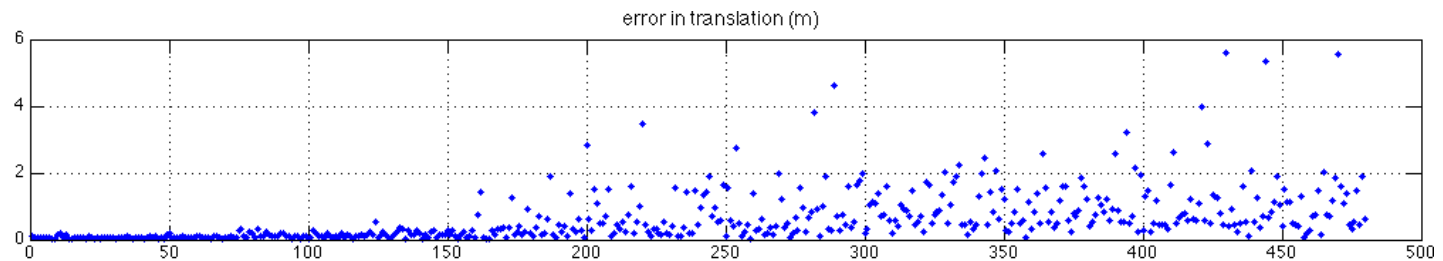
- Complexity is $O(nR^2) + O(R^2 \log R)$, where n number of points, R angular resolution.

Results

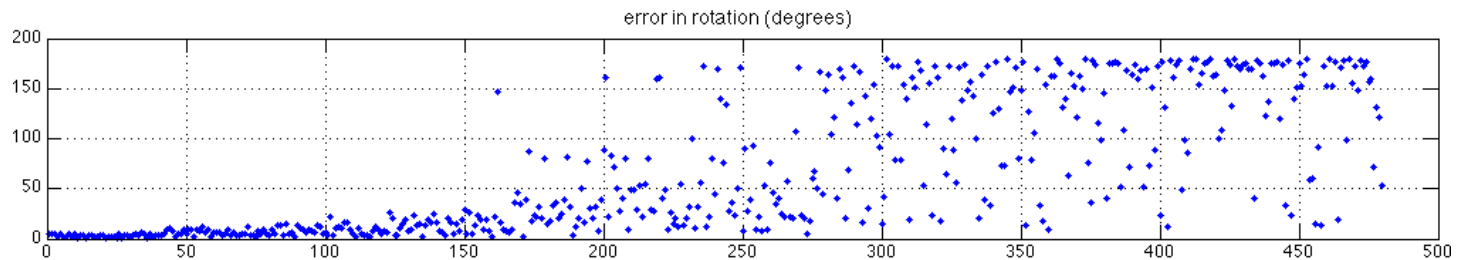
Results for realignment after random rotation (data by Andreas Nuechter).

- ICP breaks down for rotation of >60 deg
- HSM3D has constant performance
 - ◆ cpu time: 3 s for 21,000 points, 3 deg resolution

translation
errors



rotation
errors



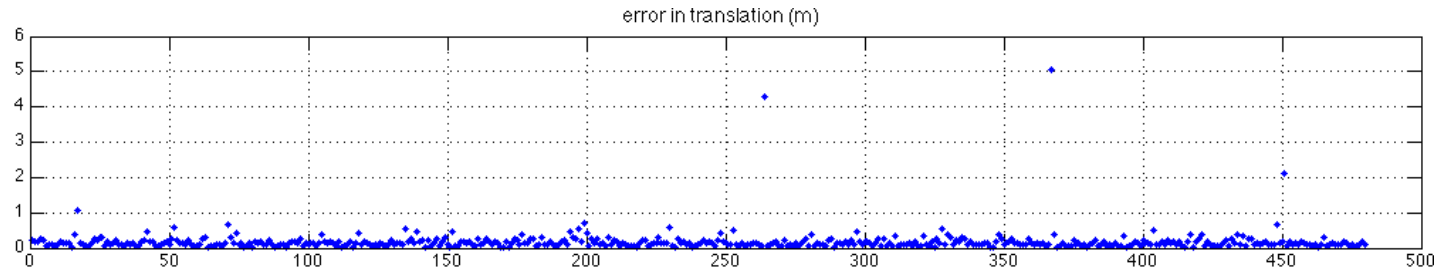
Initial angular error \rightarrow

Results

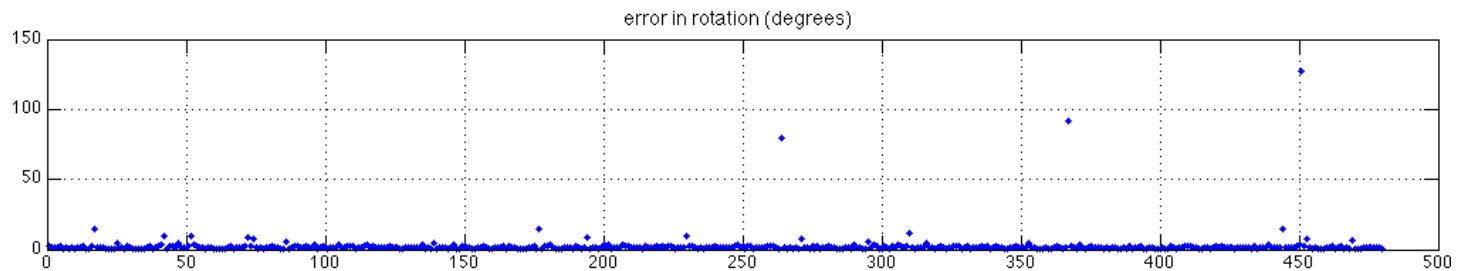
Results for realignment after random rotation (data by Andreas Nuechter).

- ICP breaks down for rotation of >60 deg
- HSM3D has constant performance
- ◆ cpu time: 3 s for 21,000 points, 3 deg resolution

translation
errors



rotation
errors



Initial angular error \rightarrow

HSM3D Summary

Summary:

- HSM3D finds general 6DOF motions.
- HSM3D does not need planes/other features.
- HSM3D is global, and detects multiple hypotheses.
- HSM3D is resolution-complete.
- Matlab and C++ source code available (see URLs in the paper)

TODO:

- better engineering of the algorithm, tuning to range-finder data, faster implementations, ...
- ask Stefano Carpin for preliminary results on stereo vision data.