Fast 3D Recognition and Pose. Viewpoint Feature Histogram

Radu Bogdan RUSU

November 2, 2010
1. Motivation
2. F/PFH
3. Priors
4. VFH
5. Examples/Results
6. Conclusion
Motivation (1/2)

Got Robots. Now what?
Motivation (2/2)

Wanted: Recognition and 6D Pose

Something along the lines of:

- **Input:**

- **Output:**

  - Object 1: ID=XX, Pose=YY
  - Object 2: ID=XX, Pose=YY
Motivation

2. F/PFH

3. Priors

4. VFH

5. Examples/Results

6. Conclusion
For every point pair \( \langle (p_s, n_s); (p_t, n_t) \rangle \), let
\[
\begin{align*}
u &= n_s, \\
v &= (p_t - p_s) \times u, \\
w &= u \times v
\end{align*}
\]

\[
f_0 &= \langle v, n_j \rangle \\
f_1 &= \langle u, p_j - p_i \rangle / \| p_j - p_i \| \\
f_2 &= \| p_j - p_i \| \\
f_3 &= \text{atan}(\langle w, n_j \rangle, \langle u, n_j \rangle)
\]

\[
i_{\text{hist}} = \sum_{x=0}^{x=3} \left[ \frac{f_x \cdot d}{f_{x_{\text{max}}} - f_{x_{\text{min}}}} \right] \cdot d^x
\]
Basic Concepts :: PCL/Features/(F)PFHEstimation

\[ f_0 = \langle v, n_j \rangle \]
\[ f_1 = \langle u, p_j - p_i \rangle / \| p_j - p_i \| \]
\[ f_2 = \| p_j - p_i \| \]
\[ f_3 = \arctan(\langle w, n_j \rangle, \langle u, n_j \rangle) \]

\[ \text{hist} = \sum_{x=0}^{x=3} \left[ \frac{f_x \cdot d}{f_{x_{\text{max}}} - f_{x_{\text{min}}}} \right] \cdot d^x \]
For every point pair \( \langle (p_s, n_s); (p_t, n_t) \rangle \), let
\[ u = n_s, \quad v = (p_t - p_s) \times u, \quad w = u \times v \]
Motivation

Point Feature Histograms (PFH) (1-4/4)

Points lying on different geometric primitives

VFH :: Fast 3D Recognition and Pose

Radu Bogdan RUSU
Fast Point Feature Histograms (FPFH) (1/2)

Basic Concepts

Re-formulate: $\text{FPFH}(p) = \text{SPF}(p) + \frac{1}{k} \sum_{i=1}^{k} \frac{1}{\omega_k} \cdot \text{SPF}(p_k)$
Theoretical formulation
Classification results using FPFH and CRF: 97.36%
(F)PFH Usage (2/3)

Classification results using **FPFH** and **CRF**: 98.27%
Motivation

(F/PFH)

Priors

VFH

Examples/Results

Conclusion

(F)PFH Usage (3/3)

Multiple Scans :: PCL/Registration/IterativeClosestPoint

points on similar surfaces

Radu Bogdan RUSU
Outline

1. Motivation
2. F/PFH
3. Priors
4. VFH
5. Examples/Results
6. Conclusion

Radu Bogdan RUSU
Priors

Assumptions and Constraints

- Dense depth + decent surface normal estimates
- Some form of prior segmentation, attention filter, etc (light clutter segmentation - not the purpose of this paper)
- the acquisition viewpoint

- VFH is meta-local and operates on point clusters (!)
1. Motivation

2. F/PFH

3. Priors

4. VFH

5. Examples/Results

6. Conclusion
VFH

Theoretical formulation: first component

▶ 1 fixed point \( c = \overline{p} = \frac{1}{k} \cdot \sum_{i=1}^{k} p_i \)

▶ \( \alpha = v \cdot n_j, \phi = u \cdot \frac{(p_j - p_i)}{d}, \theta = \arctan(w \cdot n_j, u \cdot n_j) \)
Theoretical formulation: second component

- add angles between viewpoint $V_p$ and centroid normal:
  \[ \beta = n \cdot V_p \]
VFH :: Fast 3D Recognition and Pose

Theoretical formulation

Viewpoint Feature Histogram

Viewpoint component

extended FPFH component
Outline

1. Motivation
2. F/PFH
3. Priors
4. VFH
5. Examples/Results
6. Conclusion
How to acquire data:

- Something like a pan-tilt unit works well...
How to acquire data:

- Simulated camera views for a similar camera model + identical stereo image processing code
Usage Examples/Results (3/9)

“Training”
Usage Examples/Results (4/9)

Test on Train

VFH :: Fast 3D Recognition and Pose

Radu Bogdan RUSU
Usage Examples/Results (5/9)

How to use and interpret the results
Usage Examples/Results (6/9)

How to use and interpret the results
Usage Examples/Results (7/9)

How to use and interpret the results
Usage Examples/Results (8/9)

How to use and interpret the results

VFH

Spin

VFH :: Fast 3D Recognition and Pose
Usage Examples/Results (9/9)

Performance Indicators

Numbers are not relevant - use them as indicators:

- extremely fast (SSE optimized implementation in PCL http://pcl.ros.org): \( \approx 0.3 \text{ms} / \text{point cluster} \)
- scales very well (kudos to FLANN): tried it on \( \approx 55000 \) scenes
- recognition performance (%50 + 1, first 10 NN, capped on \( d_{th} = 50 \)):

<table>
<thead>
<tr>
<th>Method</th>
<th>Object Recognition</th>
<th>Pose Estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td>VFH</td>
<td>98.52%</td>
<td>98.52%</td>
</tr>
<tr>
<td>Spin</td>
<td>75.3%</td>
<td>61.2%</td>
</tr>
</tbody>
</table>

- good behavior for synthetic data simulation (advantage if CAD models already exist)

Radu Bogdan RUSU
Outline

1. Motivation
2. F/PFH
3. Priors
4. VFH
5. Examples/Results
6. Conclusion
Questions?

Open Source! Go try it out and let us if it’s useful or not!

Point Cloud Library

http://pcl.ros.org

Radu Bogdan RUSU

VFH :: Fast 3D Recognition and Pose