Fast 3D Recognition and Pose Using the Viewpoint Feature Histogram

Solange Ramos
Felipe Moreno

Main goal

Propose a method to identify 3D objects and its pose to develop capabilities in the area of perception for mobile manipulation.

Proposed method

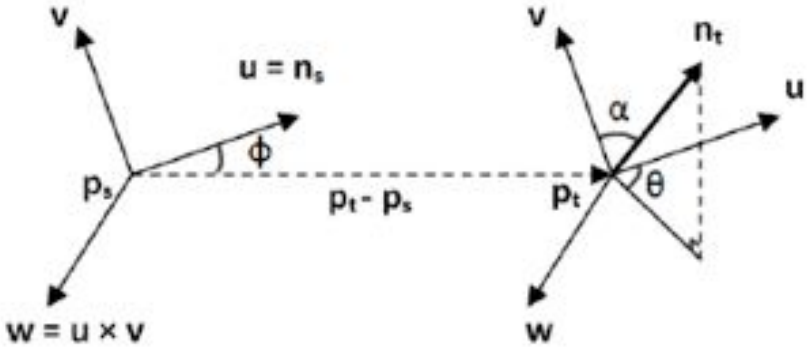
III. Architecture

Our system architecture employs the following processing steps:

- Synchronized, calibrated and epipolar aligned left and right images of the scene are acquired.
- A dense depth map is computed from the stereo pair.
- Surface normals in the scene are calculated.
- Planes are identified and segmented out and the remaining point clouds from non-planar objects are clustered in Euclidean space.
- The Viewpoint Feature Histogram (VFH) is calculated over large enough objects (here, objects having at least 100 points).
  - If there are multiple objects in a scene, they are processed front to back relative to the camera.
  - Occluded point clouds with less than 75% of the number of points of the frontal objects are noted but not identified.
- Fast approximate K-NN is used to classify the object and its view.

Point Feature Histogram

The PFH is a histogram that collects the pairwise pan, tilt and yaw angles between every pair of normals on a surface patch. O(n ∗ n).



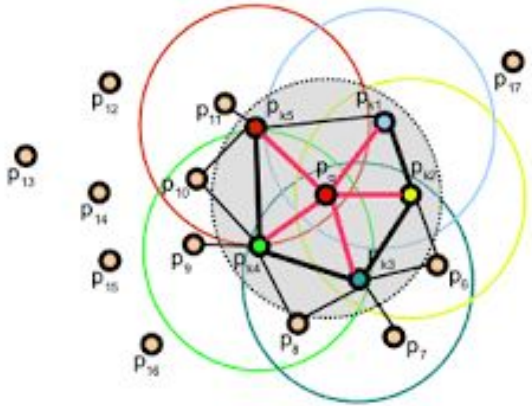
$$\alpha = \mathrm{v} \cdot \boldsymbol{n}_j$$

$$\phi = \mathrm{u} \cdot \frac{(\boldsymbol{p}_j - \boldsymbol{p}_i)}{d}$$

$$\theta = \arctan(\mathrm{w} \cdot \boldsymbol{n}_j, \mathrm{u} \cdot \boldsymbol{n}_j)$$

Fast Point Feature Histogram

The FPFH measures the same angular features as PFH, but estimates the sets of values only between every point and its k nearest neighbors, thus reducing the computational complexity to O(k ∗ n).



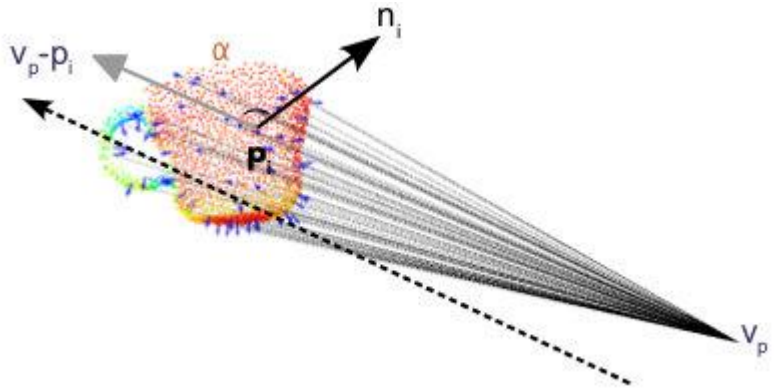
$$\alpha = \mathsf{v} \cdot \boldsymbol{n}_j$$

$$\phi = \mathsf{u} \cdot \frac{(\boldsymbol{p}_j - \boldsymbol{p}_i)}{d}$$

$$\theta = \arctan(\mathsf{w} \cdot \boldsymbol{n}_j, \mathsf{u} \cdot \boldsymbol{n}_j)$$

Viewpoint Feature Histogram

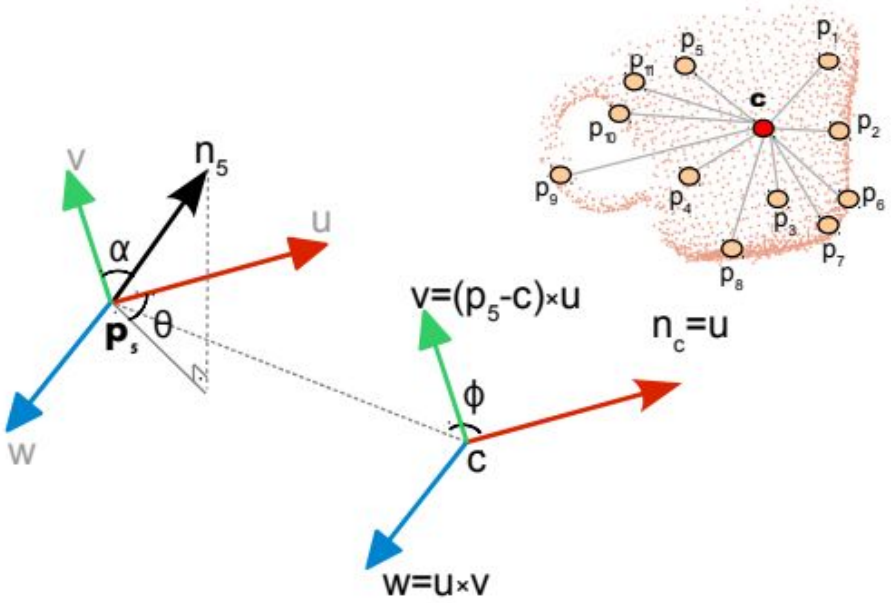
The FPFH measures the same angular features as PFH, but estimates the sets of values only between every point and its k nearest neighbors, thus reducing the computational complexity to O(k ∗ n).

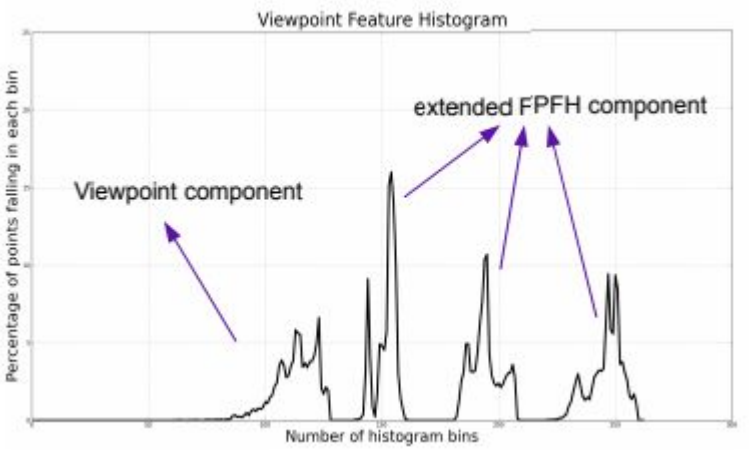


$$\alpha = \mathrm{v} \cdot \boldsymbol{n}_j$$

$$\phi = \mathrm{u} \cdot \frac{(\boldsymbol{p}_j - \boldsymbol{p}_i)}{d}$$

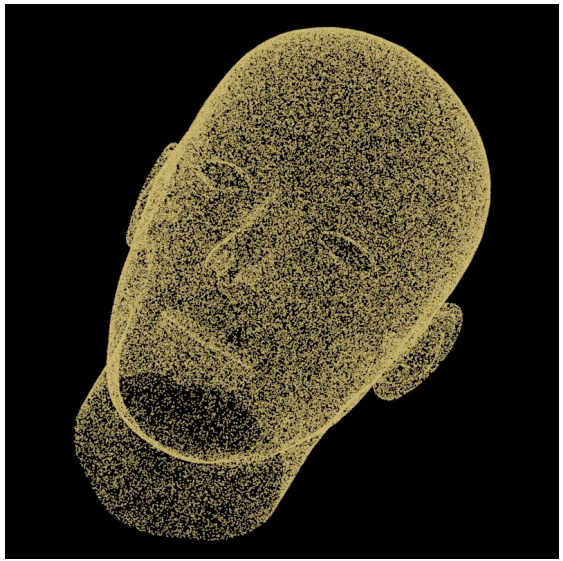
$$\theta = \arctan(\mathrm{w} \cdot \boldsymbol{n}_j, \mathrm{u} \cdot \boldsymbol{n}_j)$$

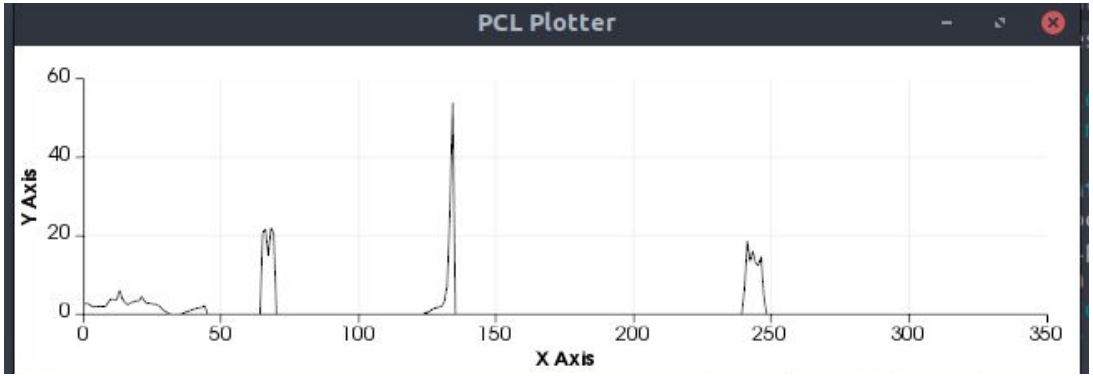

v
n5
u
α
θ
ps
w
p5
p1
p11
c
p10
p2
p9
p4
p3
p6
p7
p8
v=(p5-c)×u
nc=u
ϕ
c
w=u×v

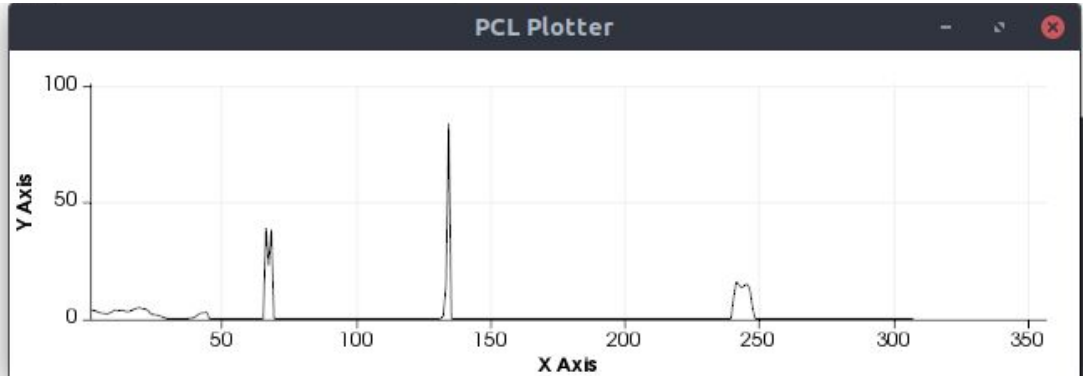

Viewpoint Feature Histogram
extended FPFH component
Viewpoint component
Percentage of points falling in each bin
Number of histogram bins

Experimental results

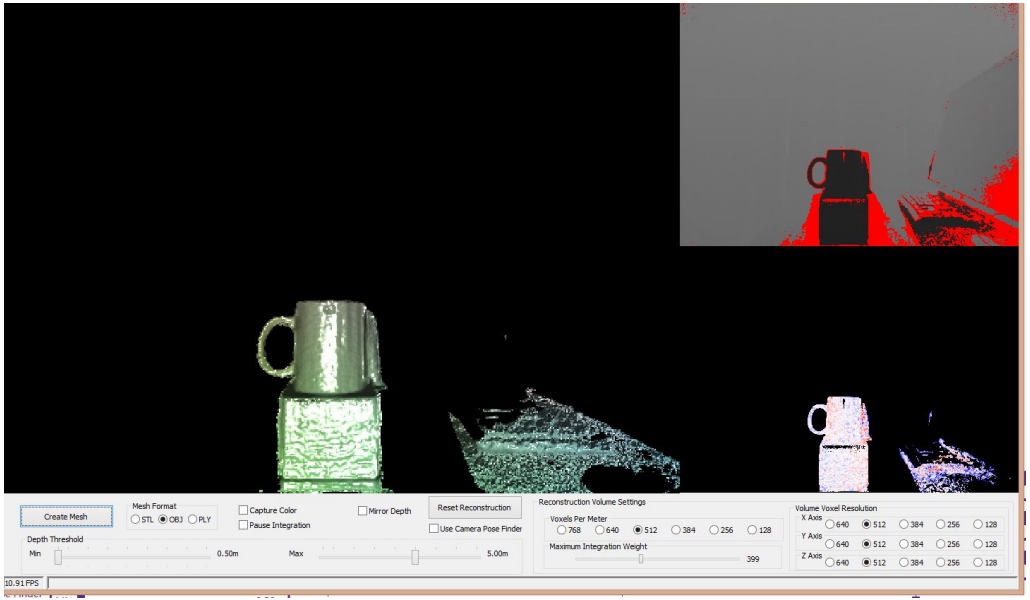
Test on face model

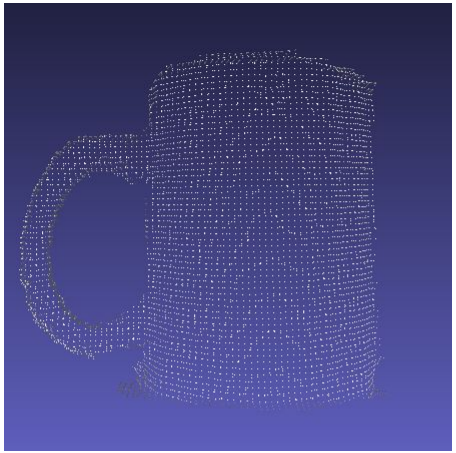


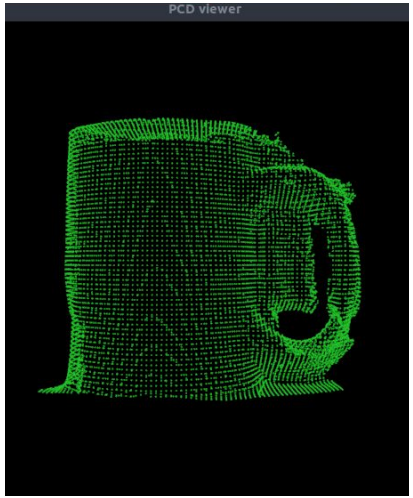


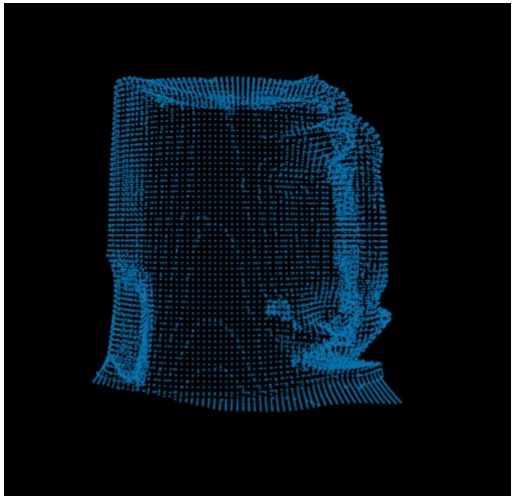


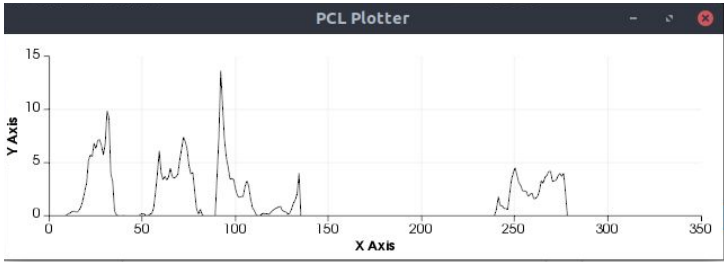
Test on cup model

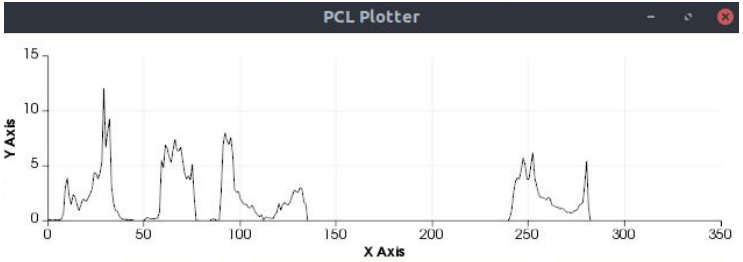





PCD viewer




PCL Plotter
Y Axis
X Axis
0
5
10
15
0
50
100
150
200
250
300
350


PCL Plotter
Y Axis
X Axis
0
5
10
15
0
50
100
150
200
250
300
350

Conclusiones

1. Los PCD y su respectivo VFH se pueden calcular en tiempo real.
2. EL VFH brinda información para ambas tareas, identificación de objetos y reconocimiento de pose y distancia aproximados desde el punto de vista.