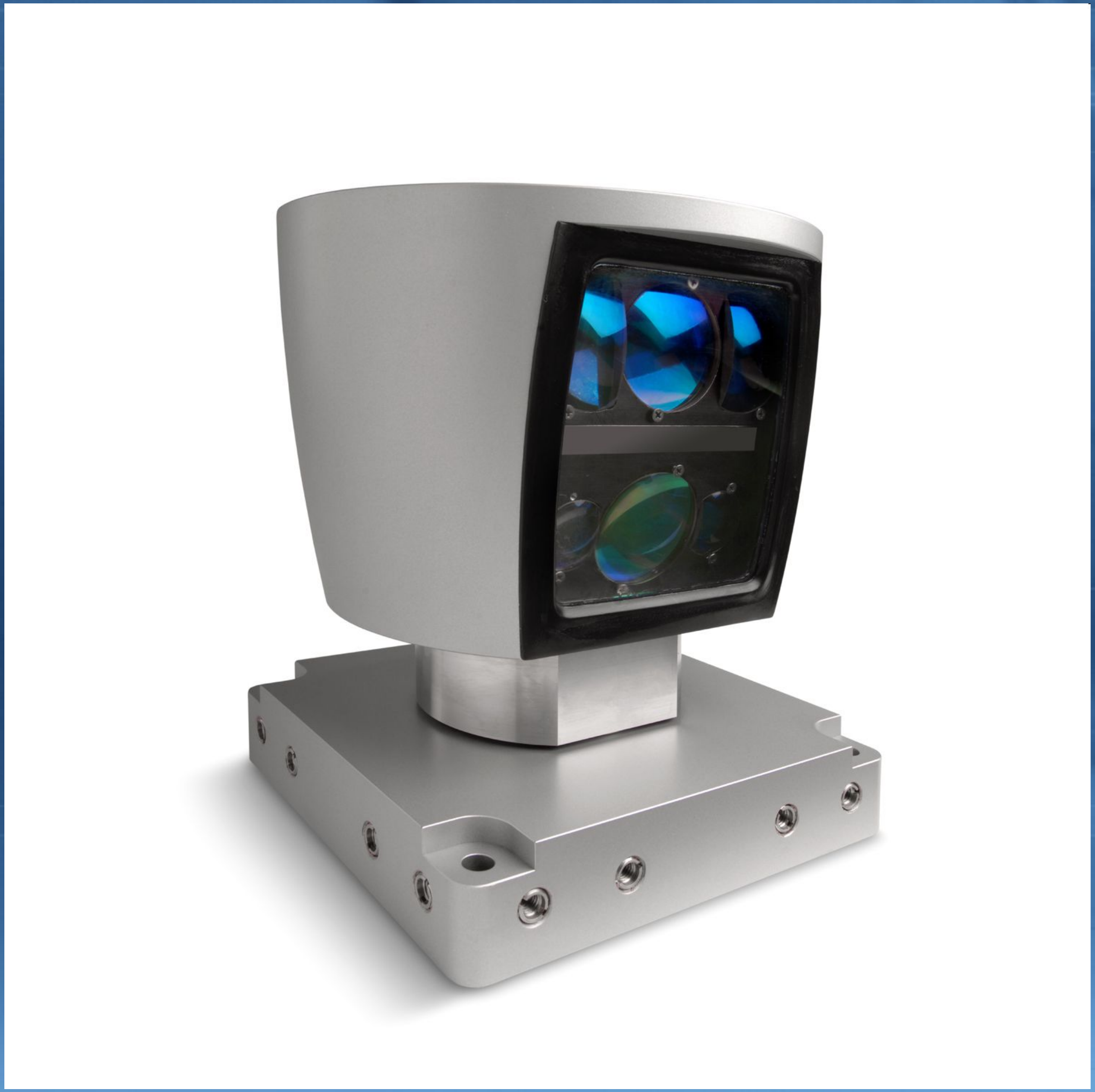
SPAWAR
Systems Center
San Diego

Plane Detection in a 3D environment using a Velodyne Lidar

Jacoby Larson
UCSD ECE 172

SPAWAR
Systems Center
San Diego

Velodyne Lidar Sensor



SPAWAR
Systems Center
San Diego

Velodyne

Used by CMU and Stanford in DARPA Urban Challenge races

Velodyne Technical Specifications

SPAWAR Systems Center San Diego

- **Sensor**
  - 64 lasers
  - 360 degree field of view (azimuth)
  - 0.09 degree angular resolution (azimuth)
  - 26.8 degree vertical field of view (elevation) -+ 2° up to -24.8° down with 64 equally spaced angular subdivisions (approximately 0.4°)
  - <2 cm distance accuracy
  - 5-15 Hz field of view update (user selectable)
  - 50 meter range for pavement (~0.10 reflectivity)
  - 120 meter range for cars and foliage (~0.80 reflectivity)
  - >1.333M points per second
  - <0.05 milliseconds latency
- **Laser**
  - Class 1 - eye safe
  - 4 X 16 laser block assemblies
  - 905 nm wavelenth
  - 5 nanosecond pulse
  - Adaptive power system for minimizing saturations and blinding
- **Mechanical**
  - 12V input (16V max) @ 4 amps
  - <29 lbs.
  - 10" tall cylinder of 8" OD diameter
  - 300 RPM - 900 RPM spin rate (user selectable)
- **Output**
  - 100 Mbps UDP Ethernet packets

Problem Statement & Motivation

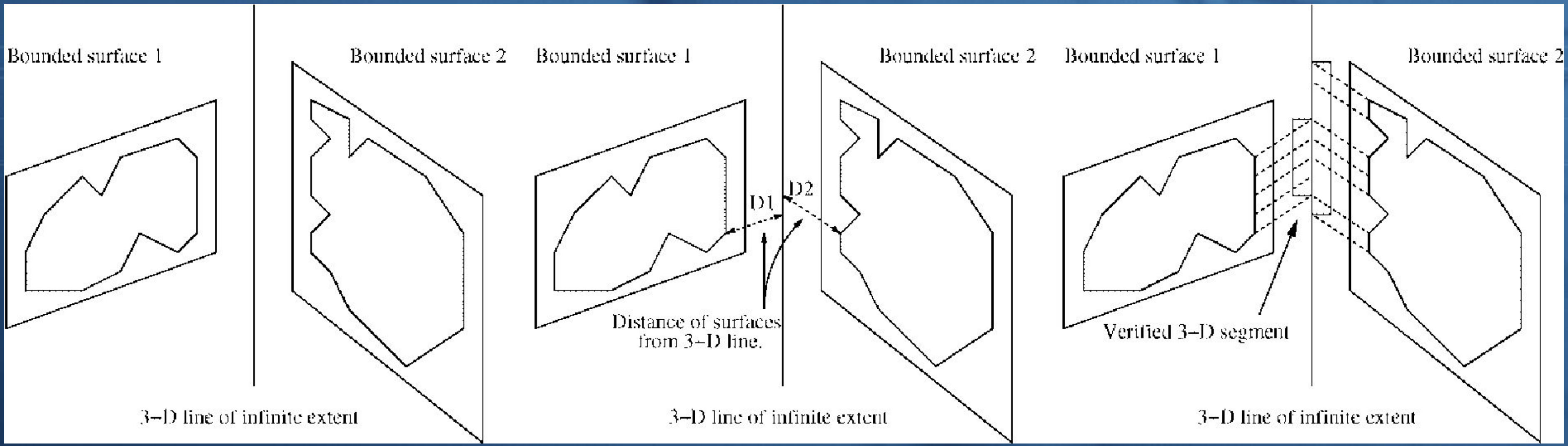
- Computer vision has a tough time determining range in real time and gathering data in 360 degrees at high resolution
- There is a need to classify objects in the real world as more than just obstacles, but as roads, driving lanes, curbs, trees, buildings, cars, IEDs, etc.
- 3D laser range finding sensors such as the Velodyne provide 360 degree ranging data that can be used to classify objects in real time

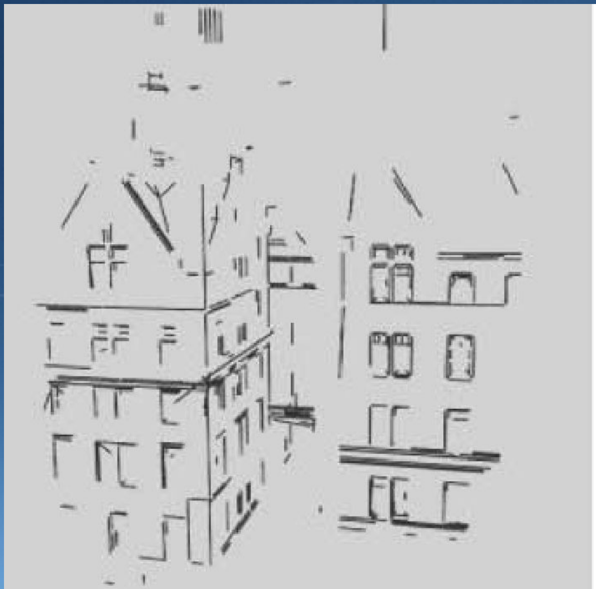
SPAWAR
Systems Center
San Diego

Related Research & Basic Approach

- Stamos, Allen, “Geometry and texture recovery of scenes of large scale”, Computer Vision and Image Understanding, Volume 88, Issue 2, pgs 94-118, Nov. 2002
  - Determine surface planes on roads, buildings, etc.
  - Find the intersections of neighboring planes to produce set of edges
  - Compare and match up these edges with those of a 2D photo image

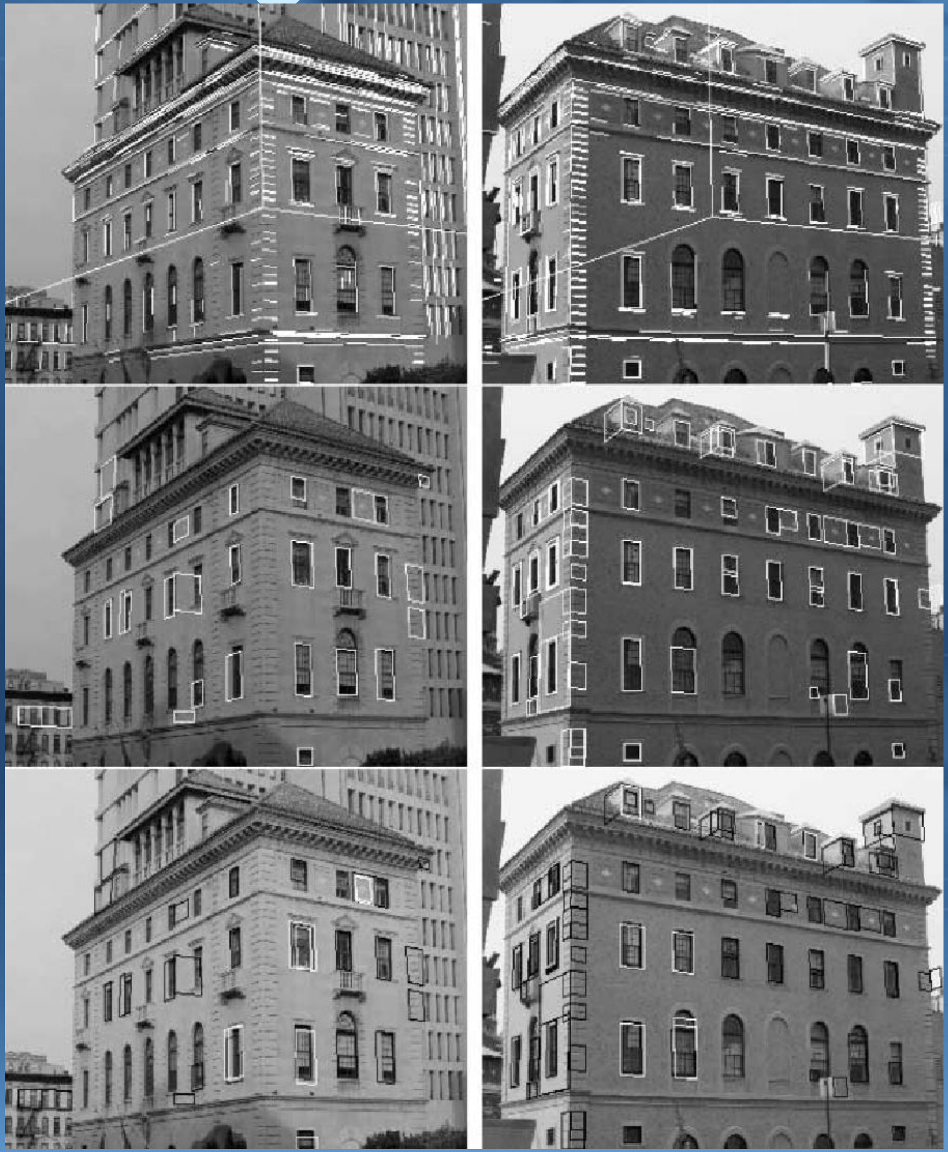
Intersection of Planes





SPAWAR
Systems Center
San Diego

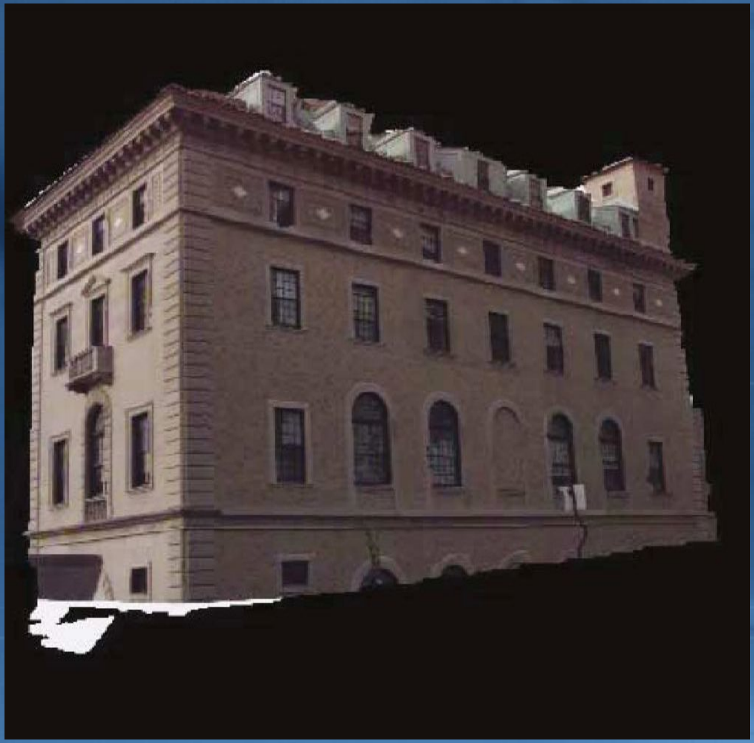
Edges of Photos



Combine Intersections and Edges

SPAWAR
Systems Center
San Diego

Final Result

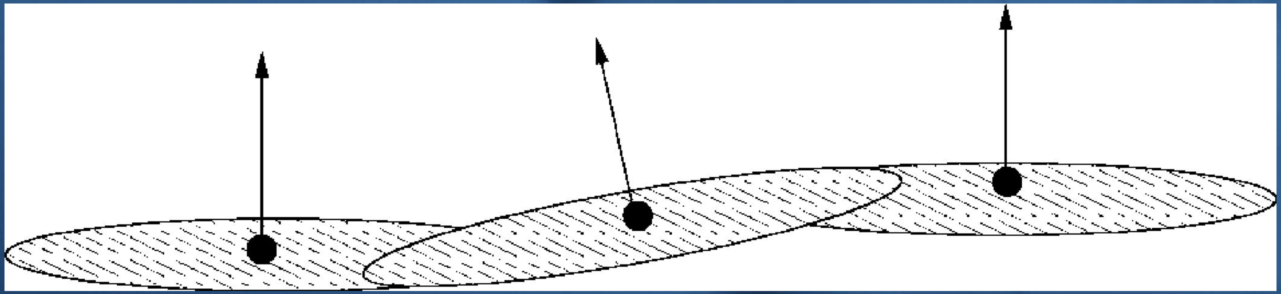


SPAWAR Systems Center San Diego

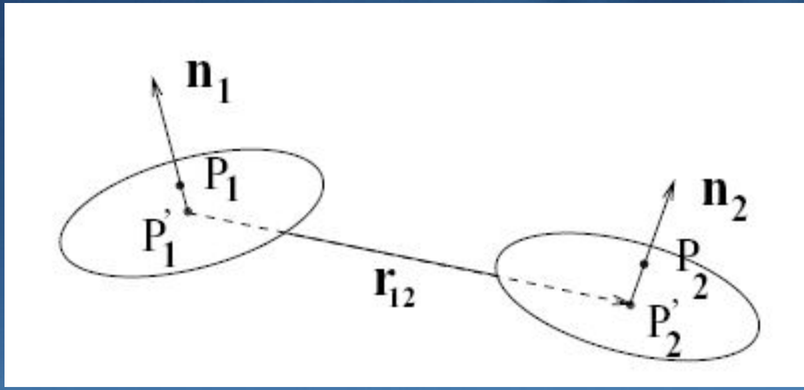
My Approach

- Select points randomly from lidar (1 million/second)
  - This should allow real-time processing whereas their approach was done offline because they looked at all data points
- Compare neighbors of random point to determine if the surface is planar and come up with a surface normal
- Combine those points with similar surface normals
- Select the group who’s surface normal matches the expected road normal
- Create a polygon from those points (Convex Hull vs. Alpha Shapes)
- Draw them on the screen

My Approach



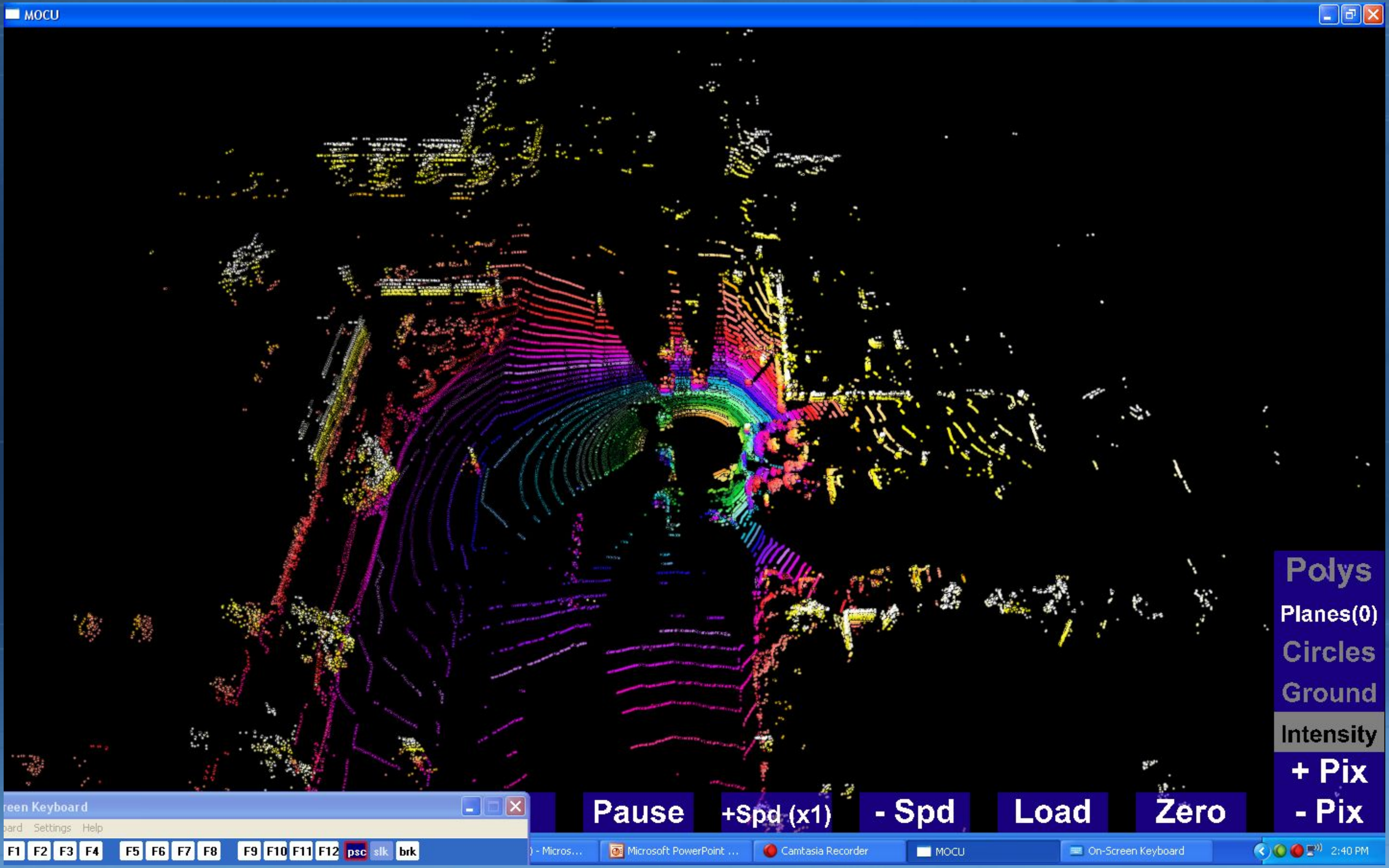
Random points and their respective planes and normals



Compare surface normals and planes to group like planes

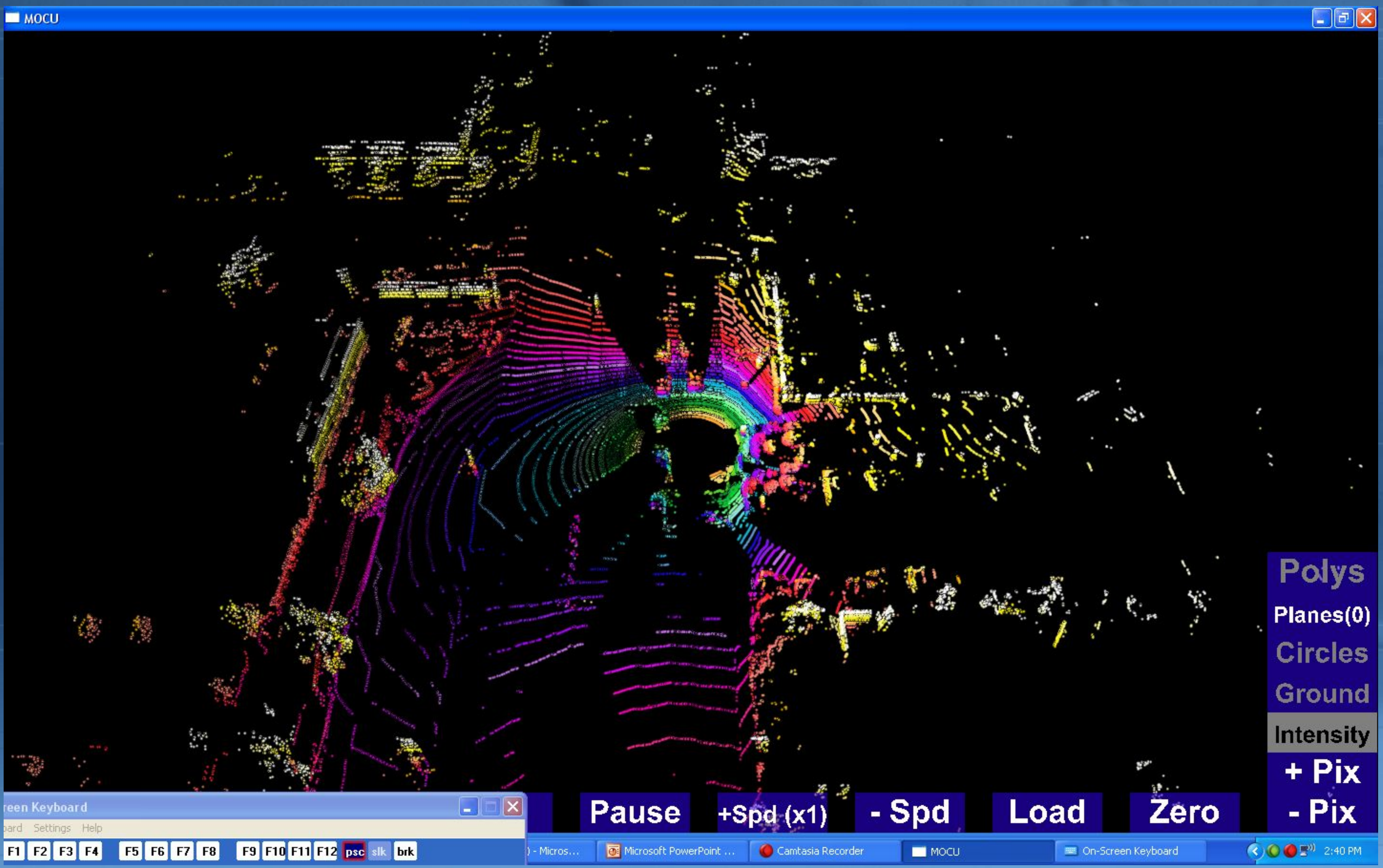
SPAWAR
Systems Center
San Diego

Demonstration



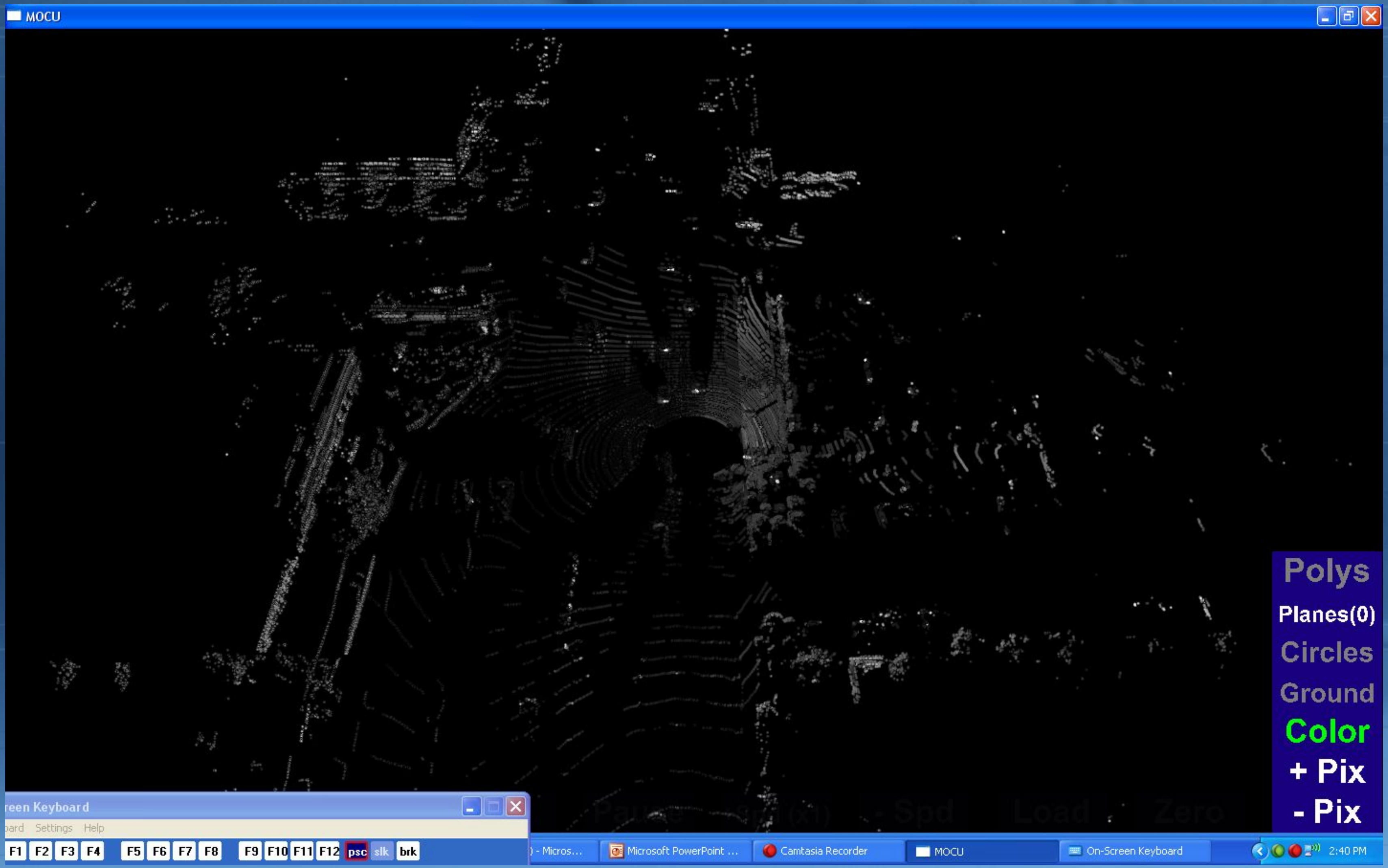
SPAWAR
Systems Center
San Diego

Screenshots



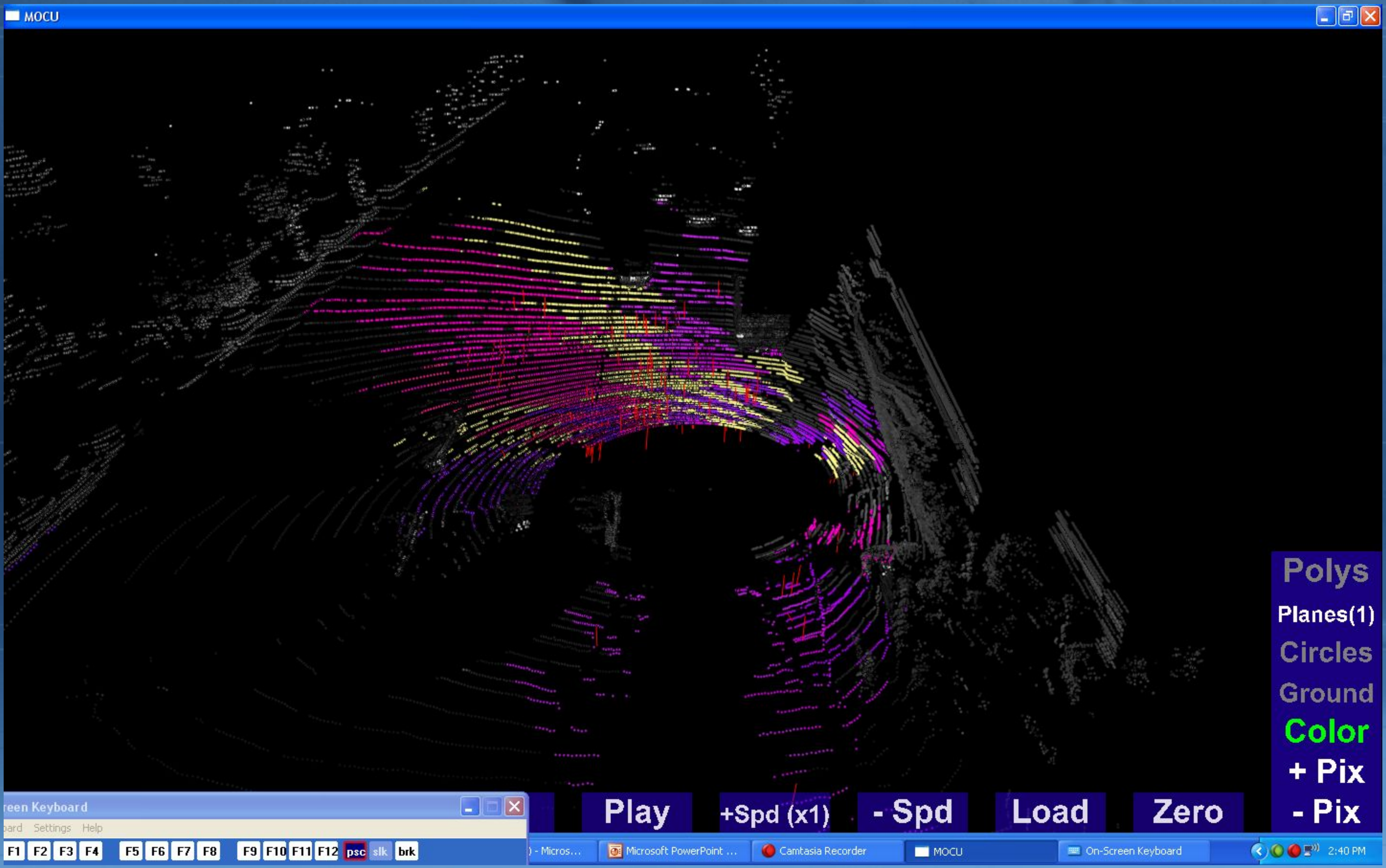
SPAWAR
Systems Center
San Diego

Screenshots



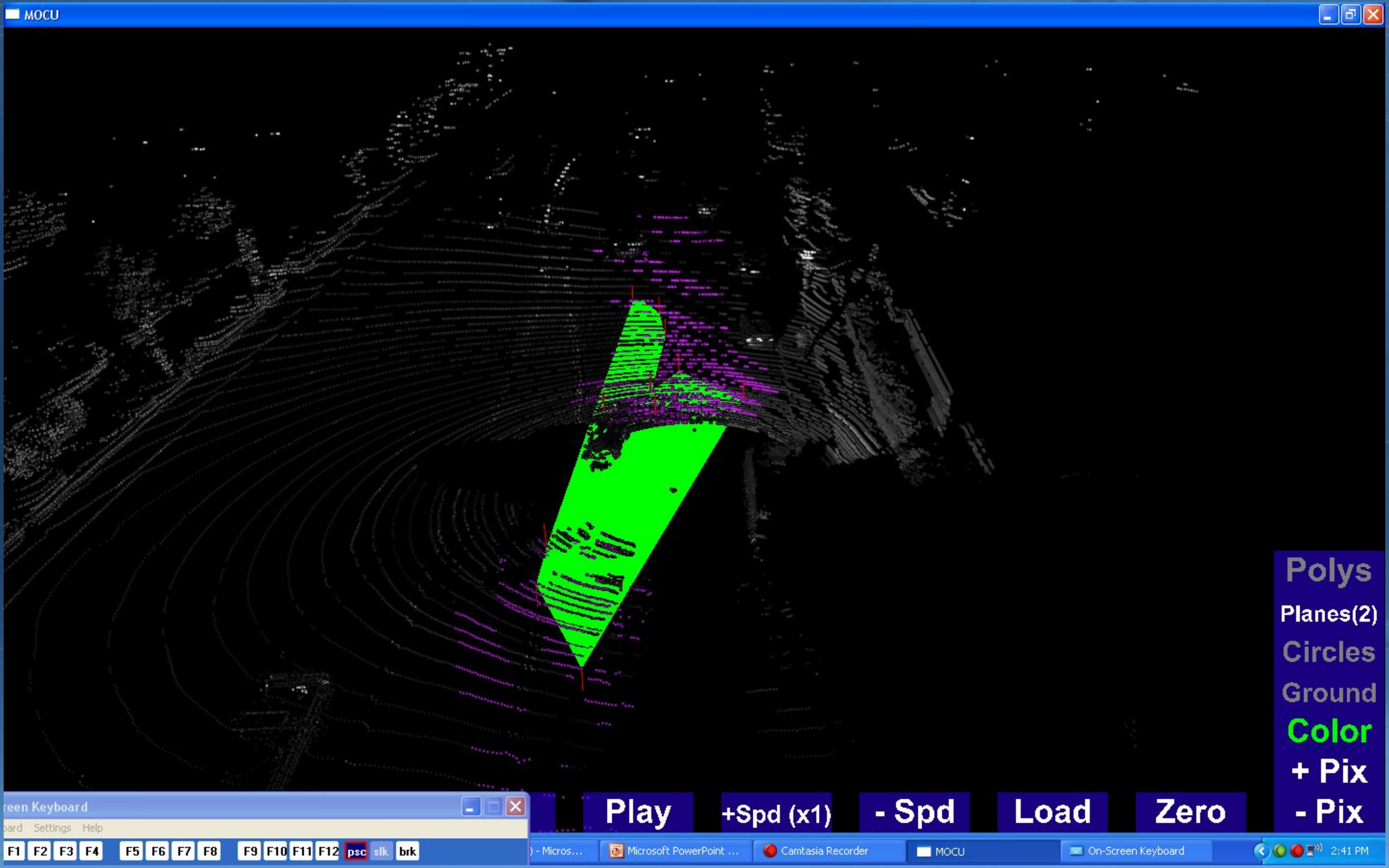
SPAWAR
Systems Center
San Diego

Screenshots



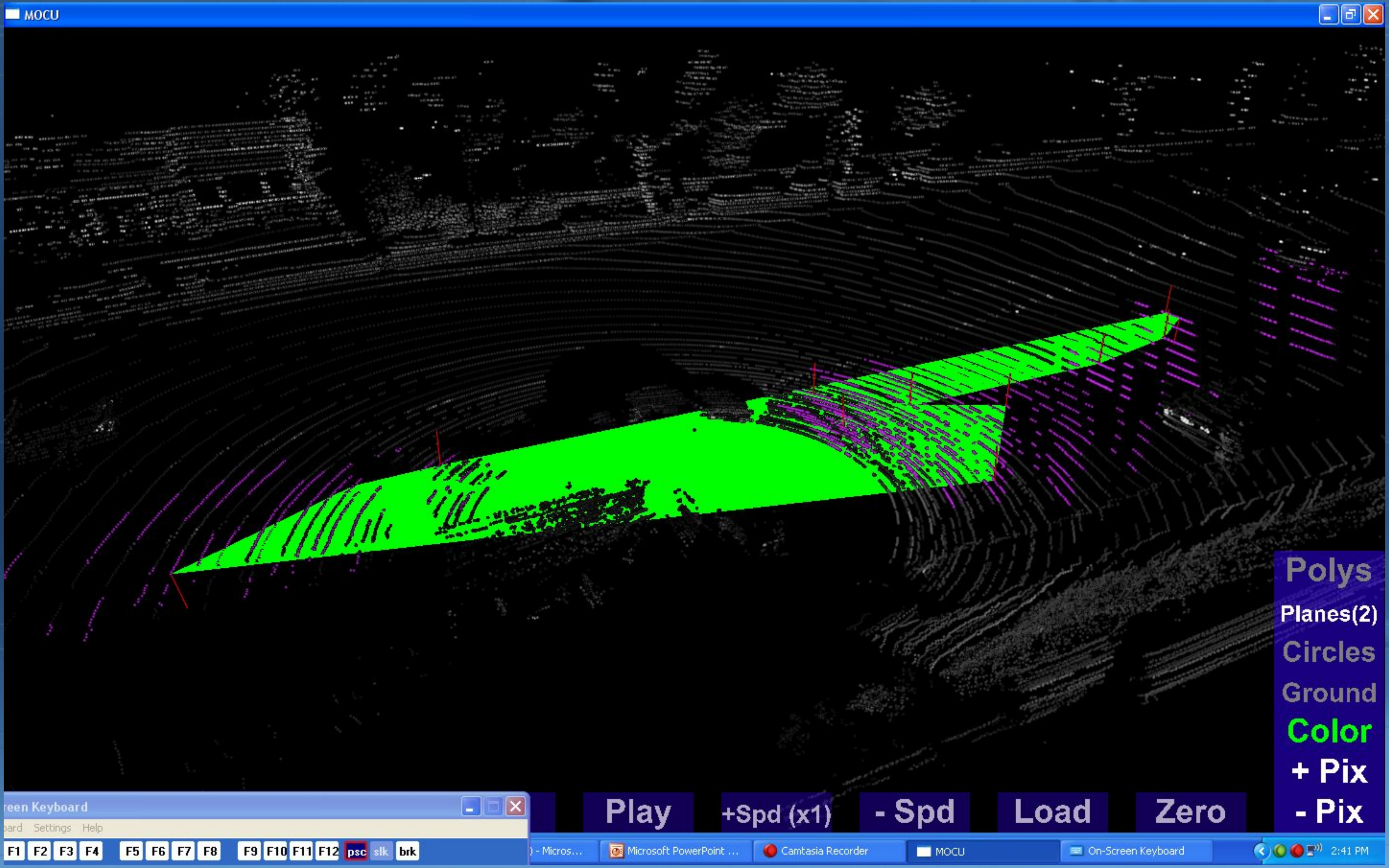
SPAWAR
Systems Center
San Diego

Screenshots



SPAWAR
Systems Center
San Diego

Screenshots



SPAWAR
Systems Center
San Diego

Results

- Good
  - Able to produce a polygon of the road surface
  - When classifying a set of data points as planar, the data was more trustworthy when searching lots of neighbors
  - Finds buildings and roads very easily
  - Real-time processing
- Bad
  - Polygon algorithm I used wasn’t too robust and doesn’t handle holes (could use alpha shapes algorithm)
  - Velodyne laser firings aren’t sequencial so looking at many neighbors can include too much area and reduce number of true planar surfaces
  - Didn’t have enough time to find planar intersections and compare with 2D photos

SPAWAR Systems Center San Diego

Future Work

- Once full width of the road has been detected, it should be fairly simple to do lane detection and curb detection
- Building detection can be done by searching for orthogonal normals
- Detection and classification of cars (using data from the road)
- Detection and classification of boats
- Detection and classification of road signs
- Still would like to merge 2D photos with 3D lidar data for more complete 3D modeling
- Create an automatic photo-lidar registration module to reduce set up time
- Contact Google to create 3D model of the world for their Google Maps.

Questions?

SPAWAR
Systems Center
San Diego