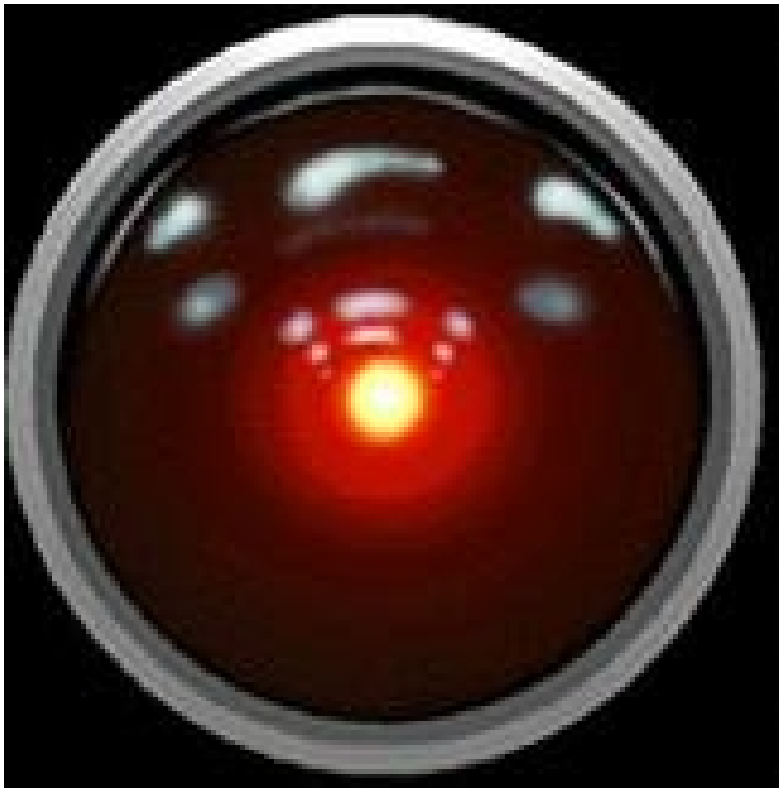
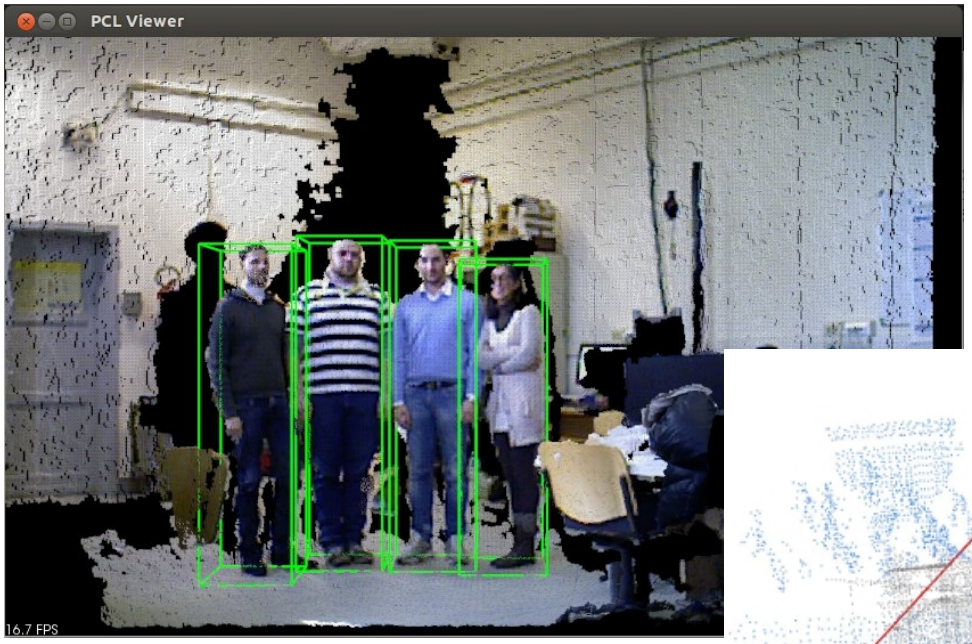


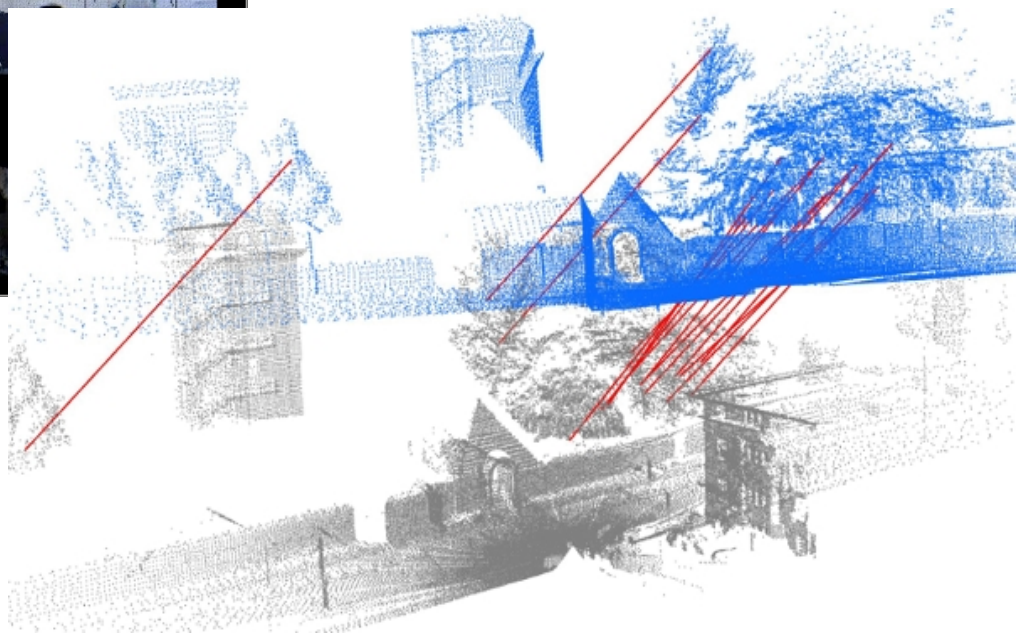
CS 378: Autonomous Intelligent Robotics

Instructor: Jivko Sinapov

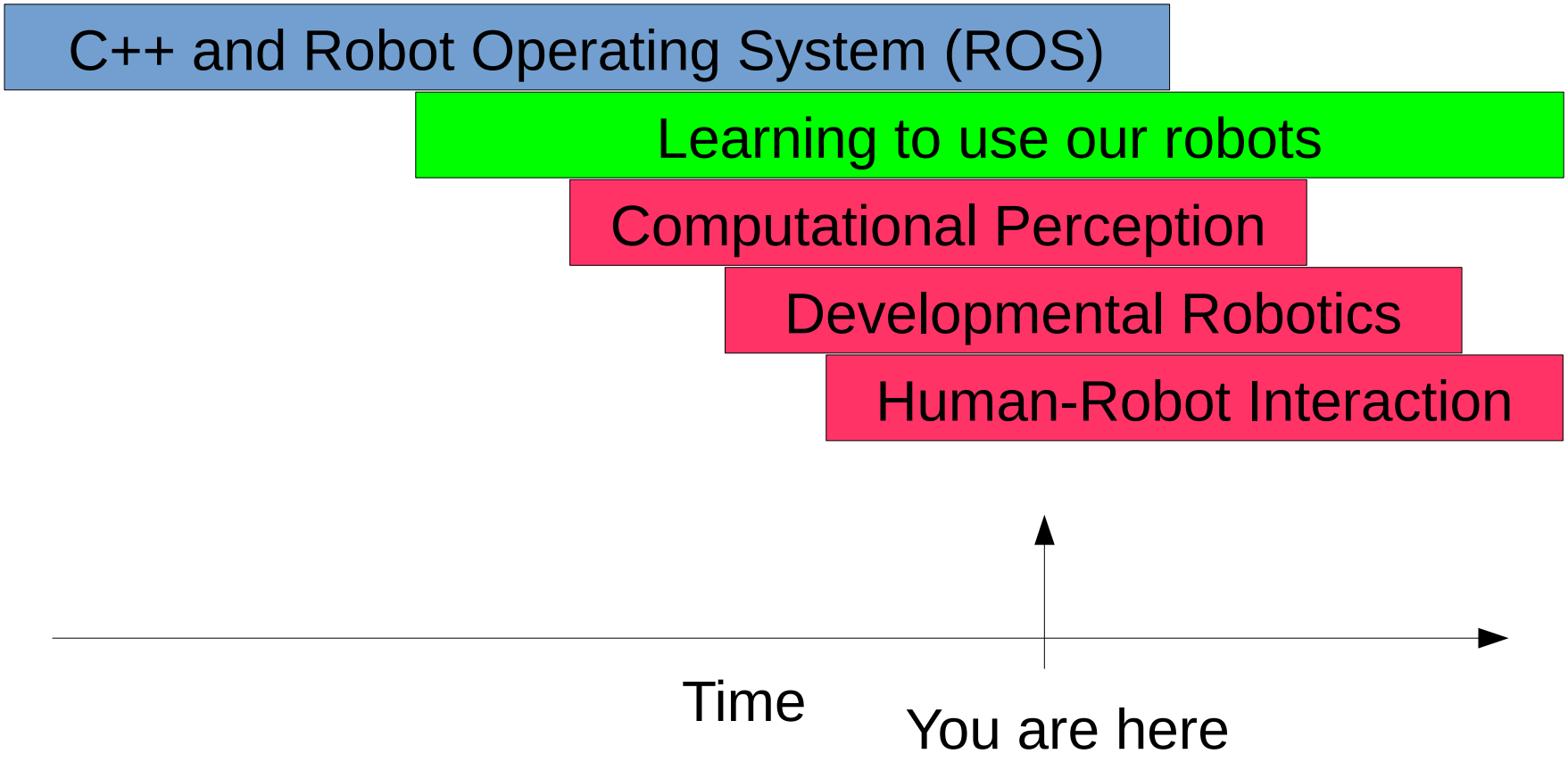
http://www.cs.utexas.edu/~jsinapov/teaching/cs378/

People Detection and Model Registration in PCL





Semester Schedule



Announcements

Homework 6 is out, due ~~4/5~~ 4/7

Announcements

1-2 Volunteers needed for...tomorrow, 10 am – 11 am

Help is needed to setup the robot arm robot in the union for a demo

The demo is at an event used to raise money for FRI

Installing our code base

- Github page:
  - https://github.com/utexas-bwi/bwi

Getting your project code up on github

GitHub guide:

https://guides.github.com/activities/hello-world/

Readings for this week

1) Alex Smola and S.V.N. Vishwanathan, Introduction to Machine Learning, Chapter 1, Cambridge University Press, 2008

Readings for this week

In addition, this week, you get to pick a published, peer-reviewed conference or journal article.

Your reading response should be about your pick.

Robotics and AI Conferences

- IEEE International Conference on Robotics and Automation (ICRA)
- IEEE International Conference on Intelligent Robots (IROS)
- IEEE International Conference on Development and Learning (ICDL)
- Robotics Science and Systems (RSS)

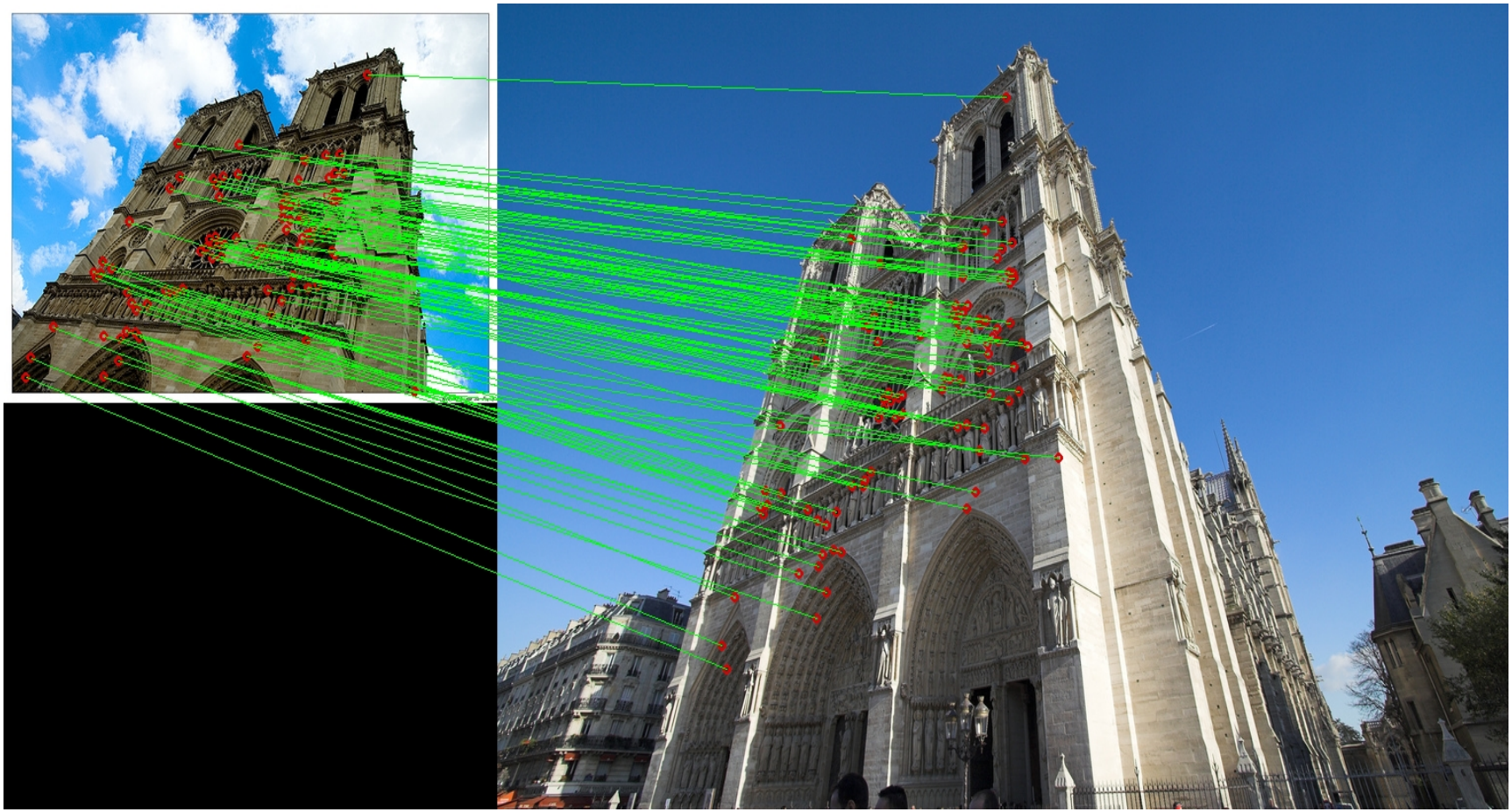
Robotics and AI Conferences (con't)

- ACM / IEEE International Conference on Human-Robot Interaction (HRI)
- International Conference on Social Robotics (ICSR)
- AAAI Conference on Artificial Intelligence (AAAI)
- International Joint Conference on Artificial Intelligence (IJCAI)

Robotics Journals

- IEEE Transactions on Robotics (TRO)
- IEEE Transactions on Autonomous Mental Development (TAMD)
- International Journal of Robotics Research (IJRR)
- Robotics and Autonomous System (RAS)

Last Time...



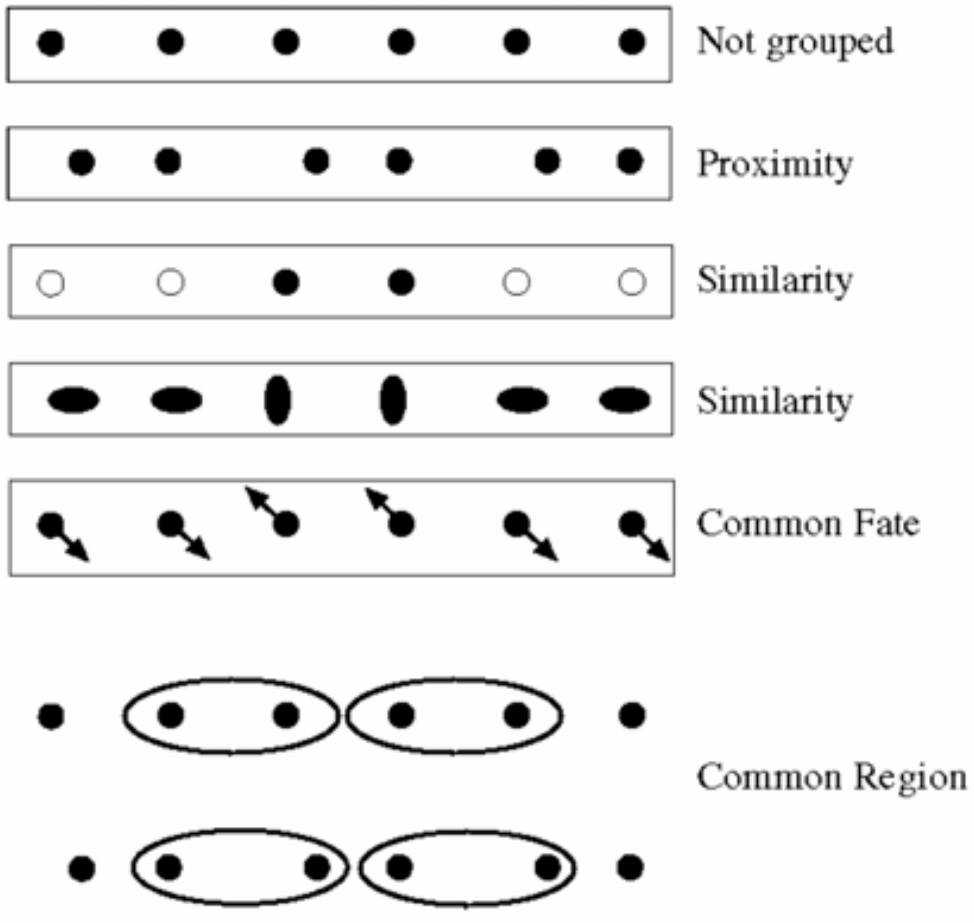
To summarize...

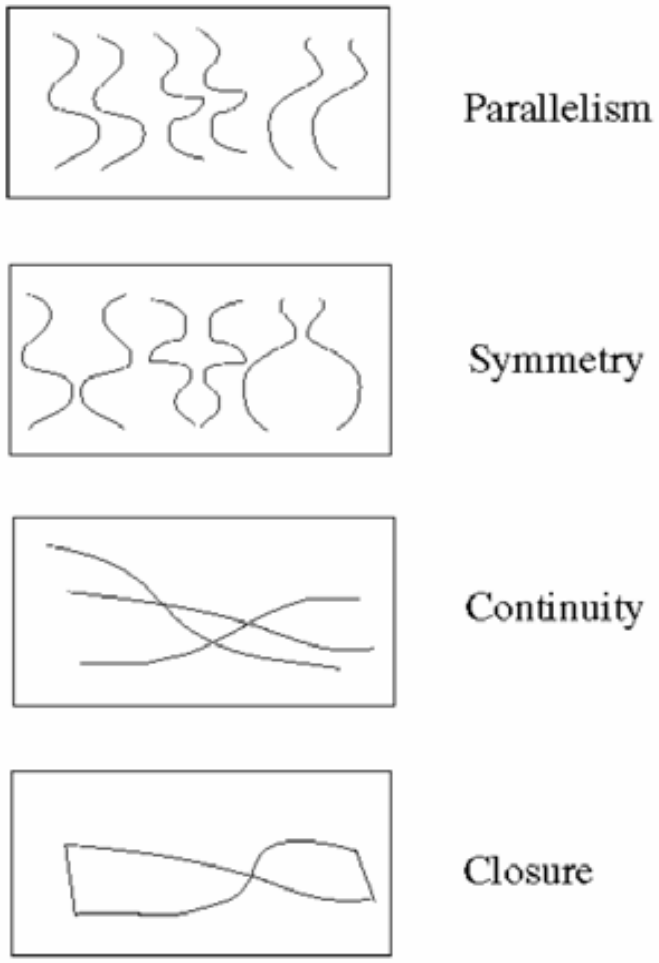
- Feature detectors:
  - Find interest points in image (e.g., using difference of Gaussians, Harris corner detection, etc.)
- Feature descriptors
  - Each detected feature can be represented by a numerical descriptor encoding orientation, scale, etc.

Optical Flow

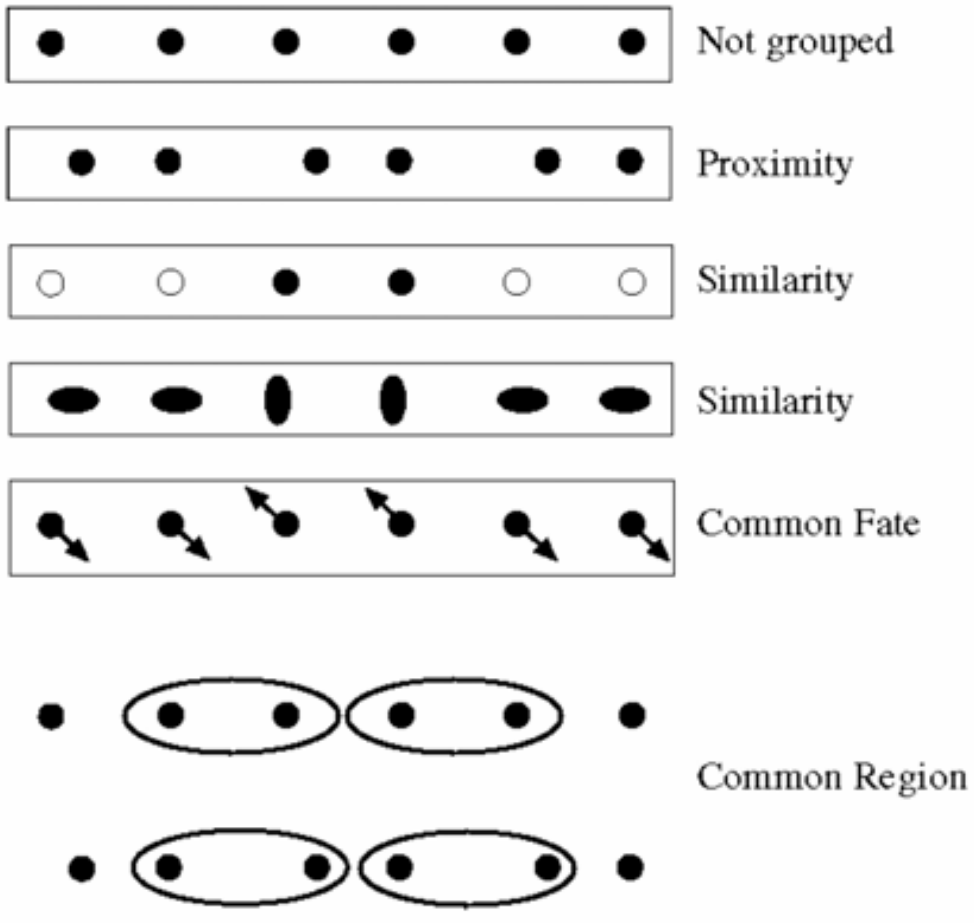
- Interest key points and feature descriptors are great but suffer from one limitation:
  - They ignore time

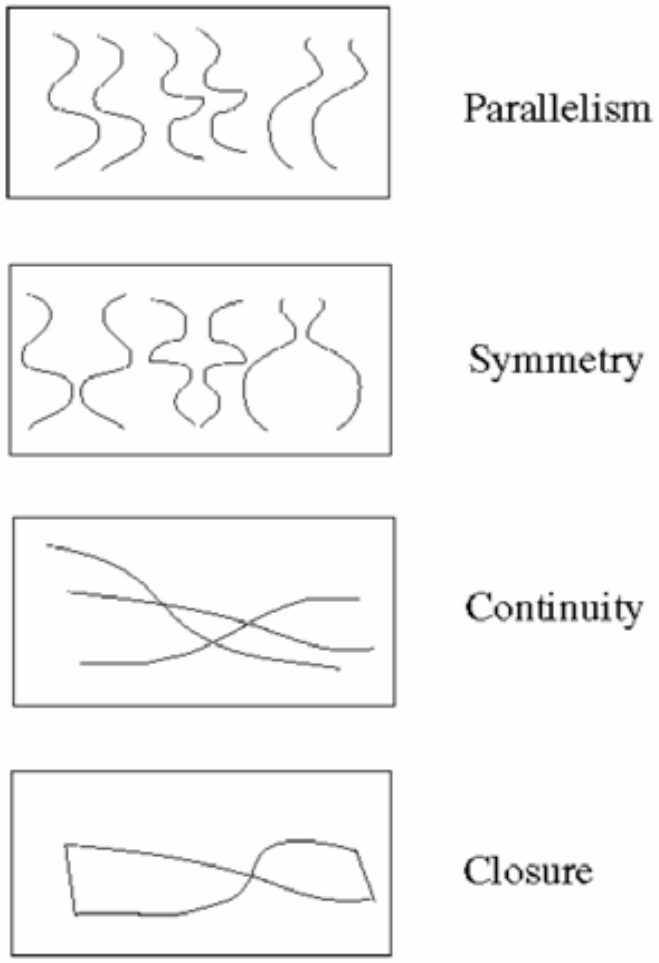
Why Optical Flow?



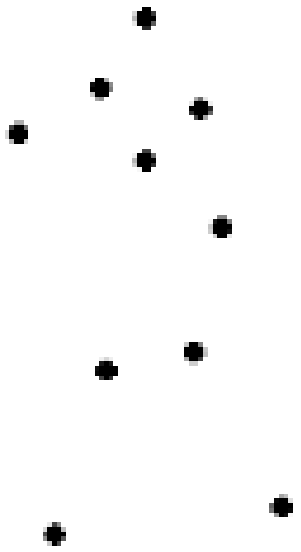


Why Optical Flow?

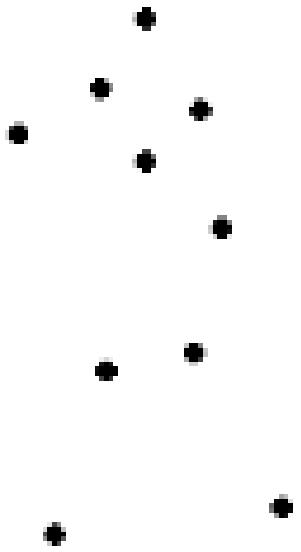




Why Optical Flow?



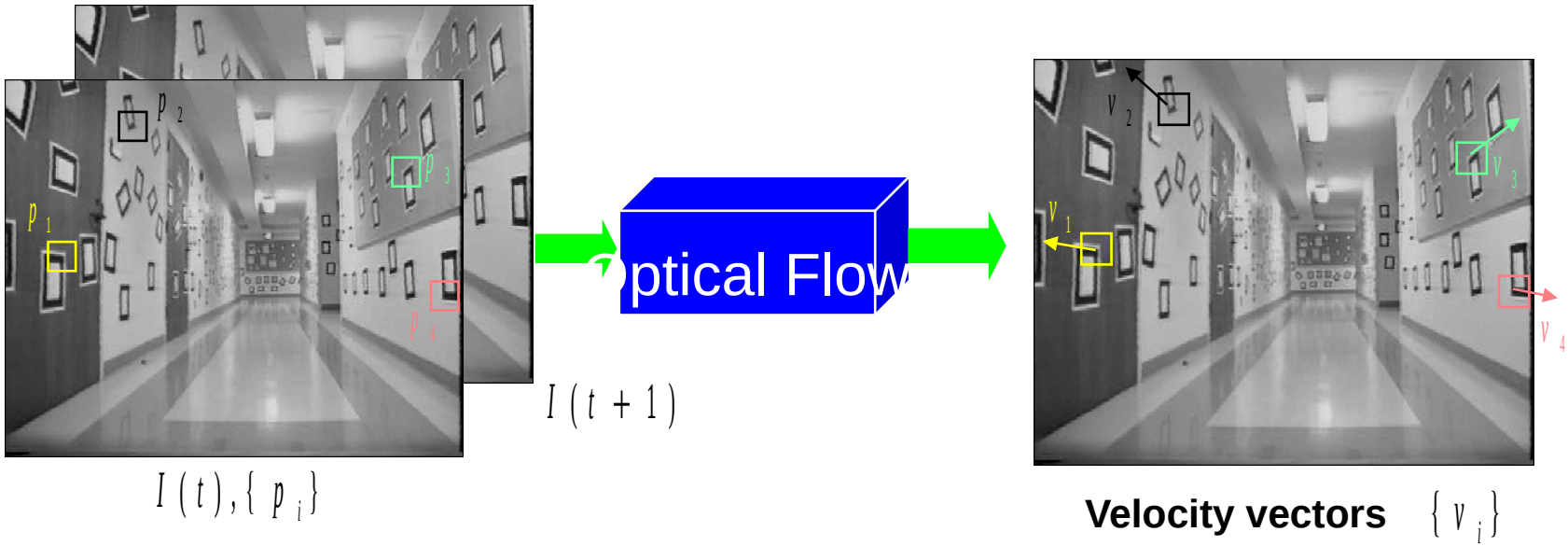
Why Optical Flow?



Optical Flow Video

- http://www.youtube.com/watch?v=o8NOabnZPIY

What is Optical Flow?

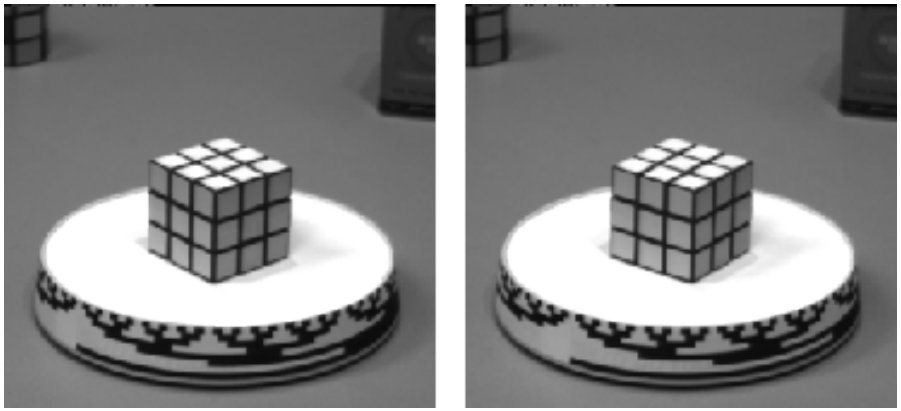


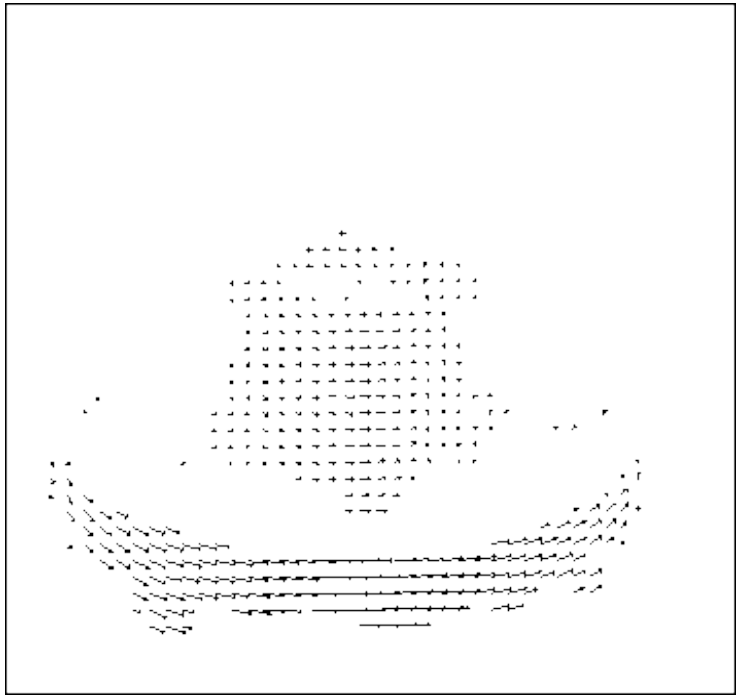
"Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image"

- Horn and Schunk, 1981

Motion Fields

- The motion field is the projection of the 3D scene motion into the image





Motion Fields and Camera Movement

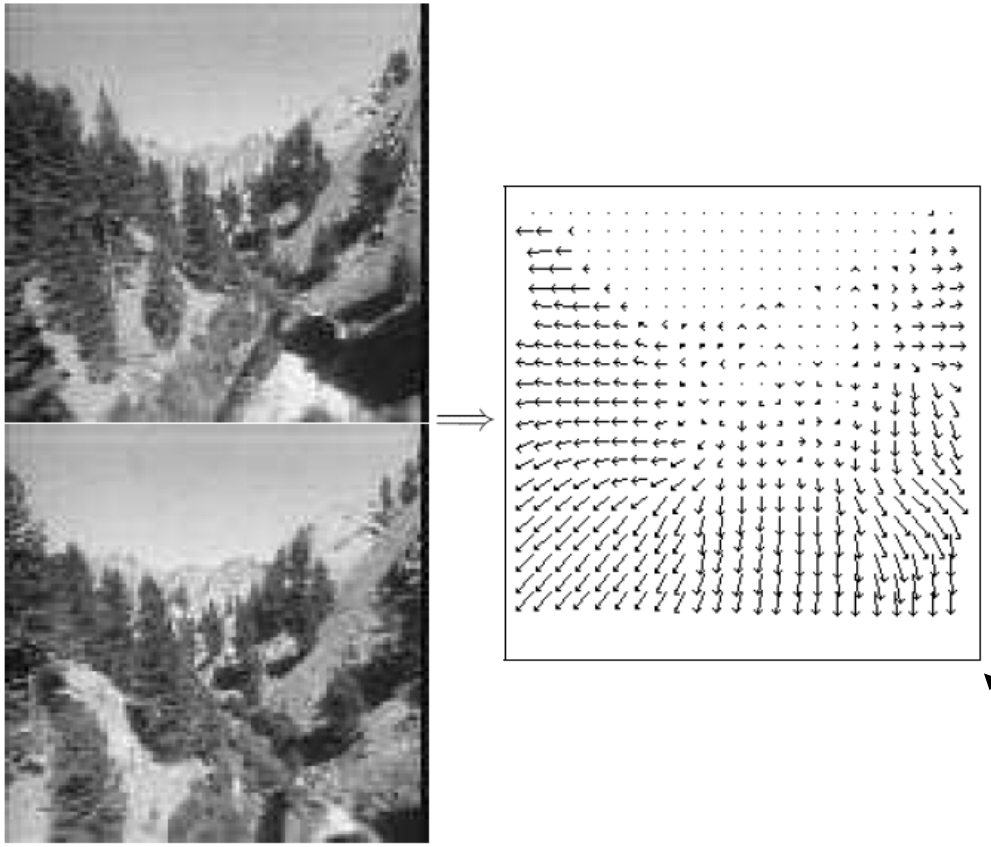


Figure 1.2: Two images taken from a helicopter flying through a canyon and the computed optical flow field.

Length of flow vectors inversely proportional to depth Z of 3D point

points closer to the camera move more quickly across the image plane

Figure from Michael Black, Ph.D. Thesis

Why do we want optical flow?

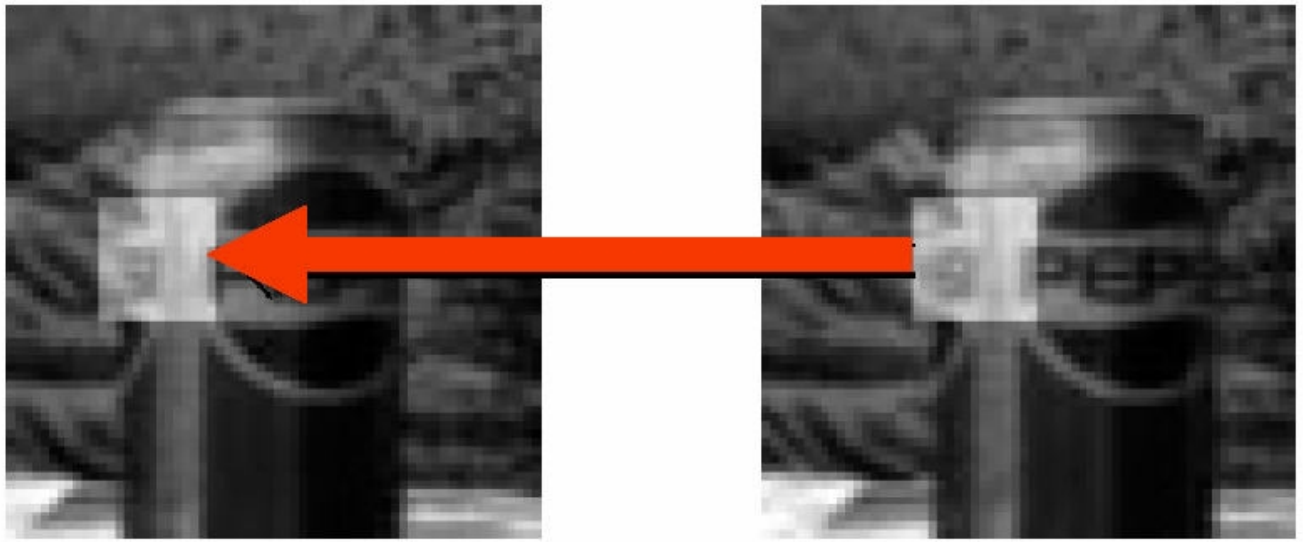
- For example, autonomous helicopters:
  - https://www.youtube.com/watch?v=V4r2HXGA8jw

Computing Optical Flow

- Given a set of points in an image, find those same points in another image
- Or, given point $[u_x, u_y]^T$ in image I_1 find the point $[u_x + \delta_x, u_y + \delta_y]^T$ in image I_2 that minimizes ε:

$$\varepsilon(\delta_x, \delta_y) = \sum_{x=u_x-w_x}^{u_x+w_x} \sum_{y=u_y-w_y}^{u_y+w_y} \left(I_1(x,y) - I_2(x+\delta_x, y+\delta_y) \right)$$

Optical Flow Assumptions



Assumption

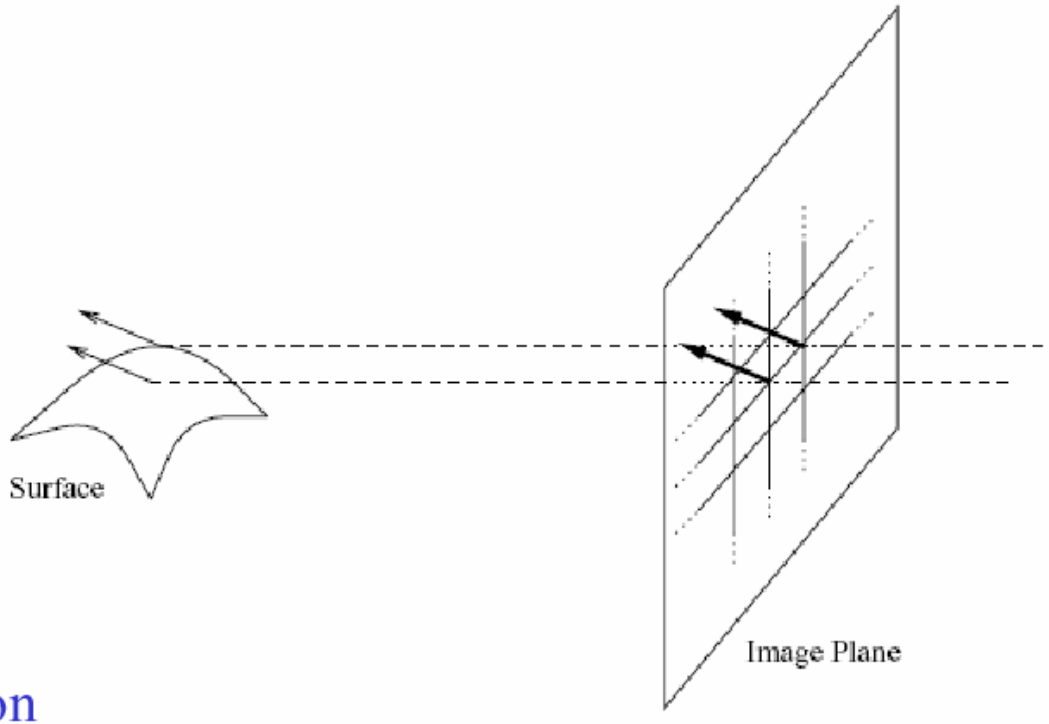
Image measurements (e.g. brightness) in a small region remain the same although their location may change.

$$I(x+u, y+v, t+1) = I(x, y, t)$$

(assumption)

* Slide from Michael Black, CS143 2003

Spatial Coherence

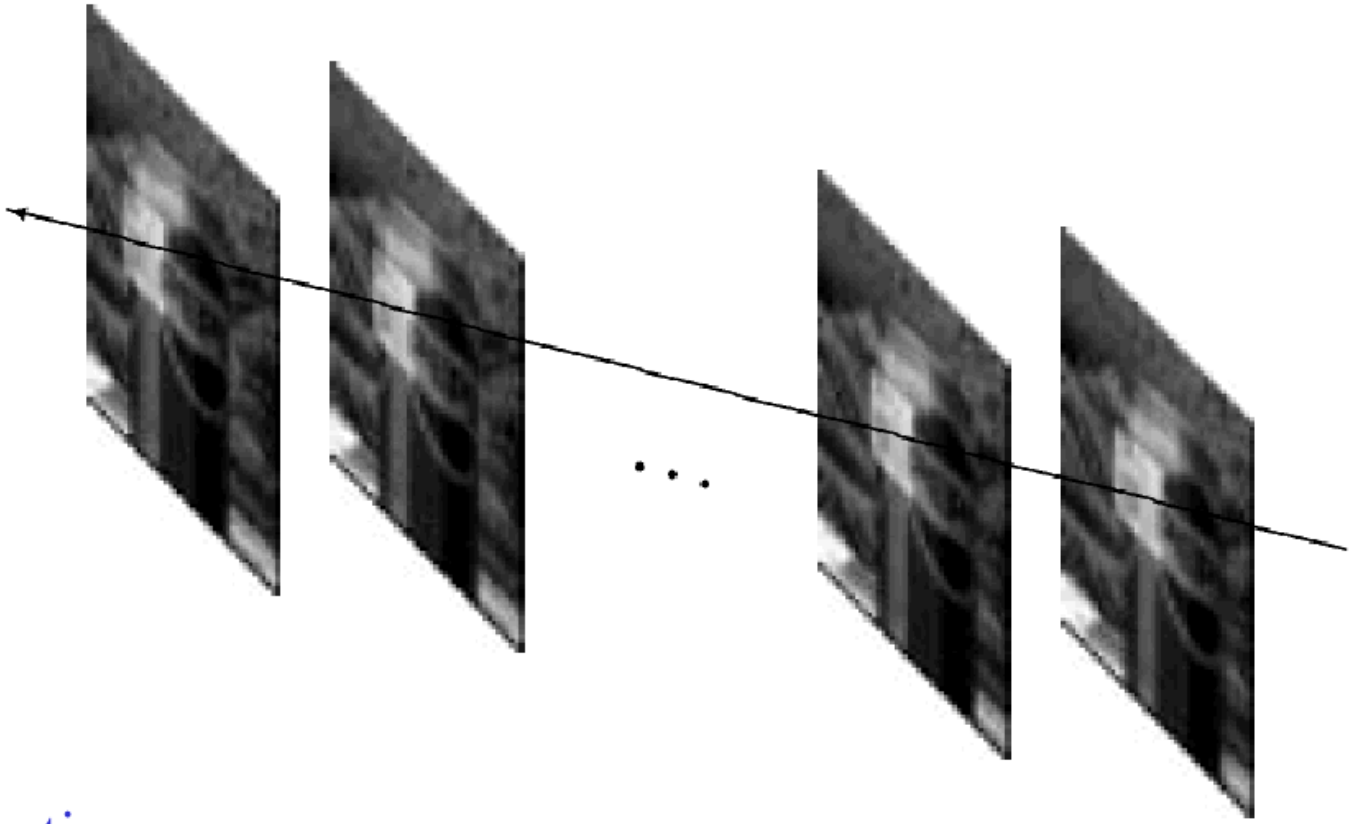


Assumption

* Neighboring points in the scene typically belong to the same surface and hence typically have similar motions.
* Since they also project to nearby points in the image, we expect spatial coherence in image flow.

* Slide from Michael Black, CS143 2003

Temporal Persistence

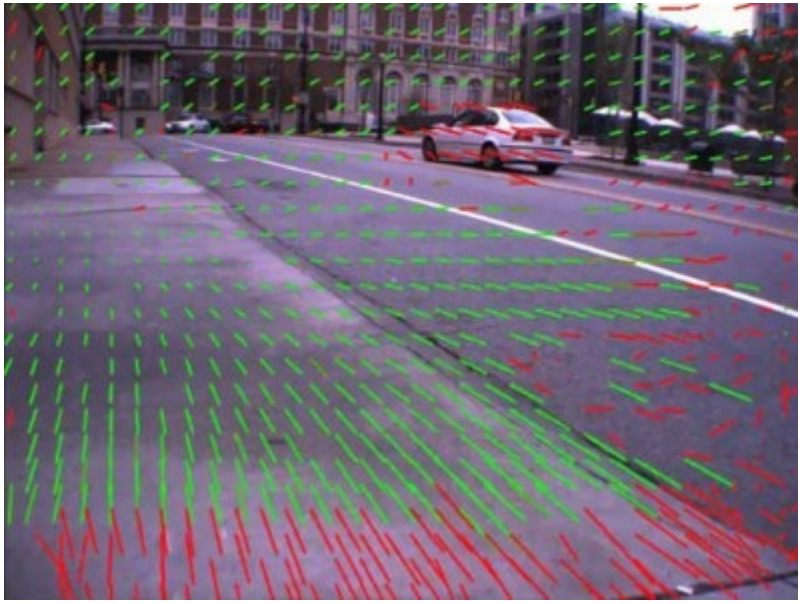


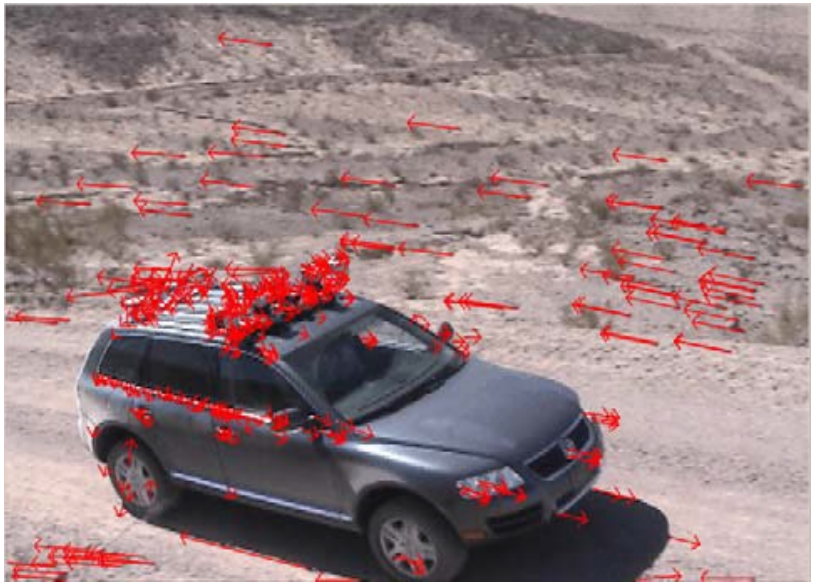
Assumption:

The image motion of a surface patch changes gradually over time.

* Slide from Michael Black, CS143 2003

Dense vs. Sparse Optical Flow

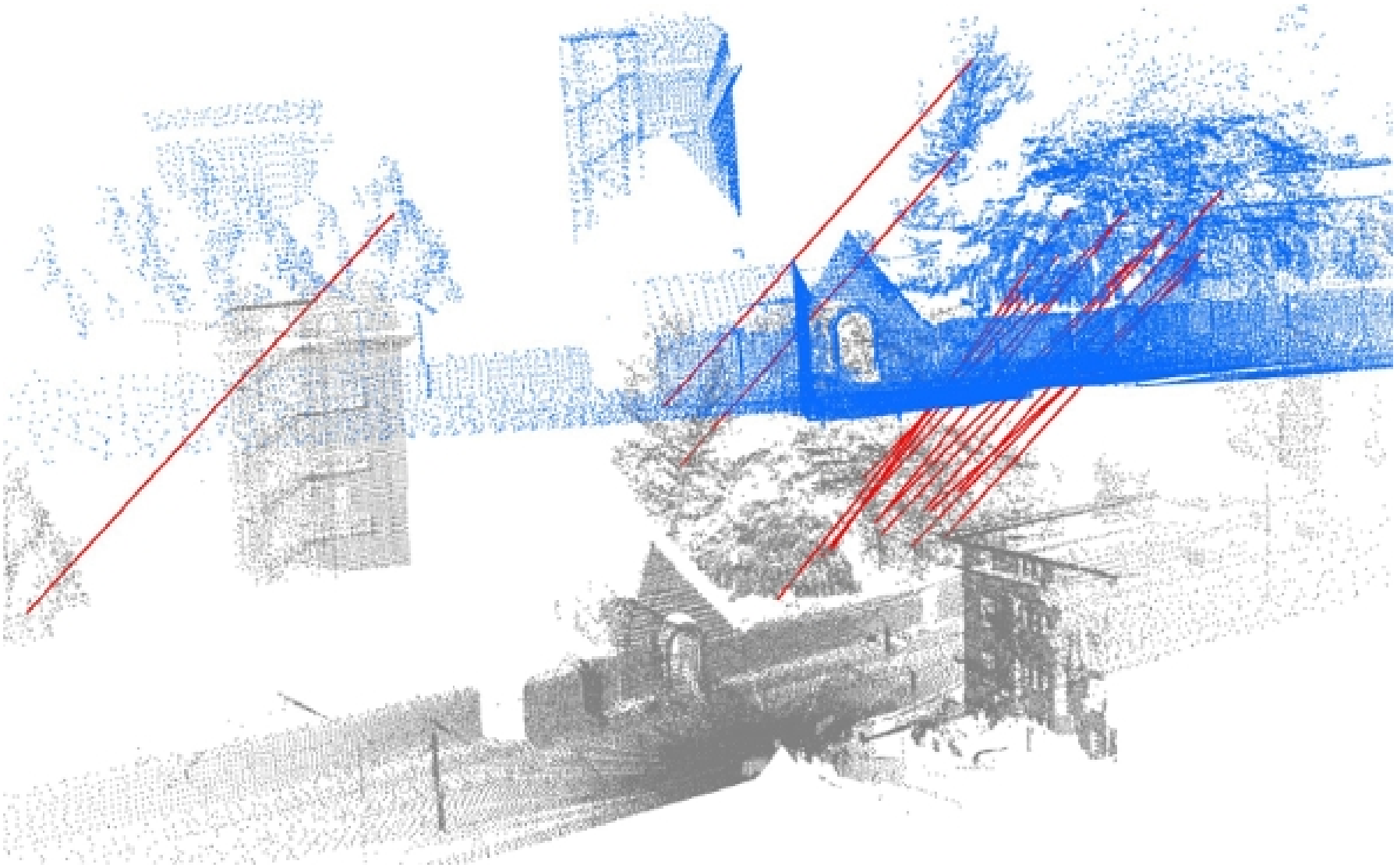




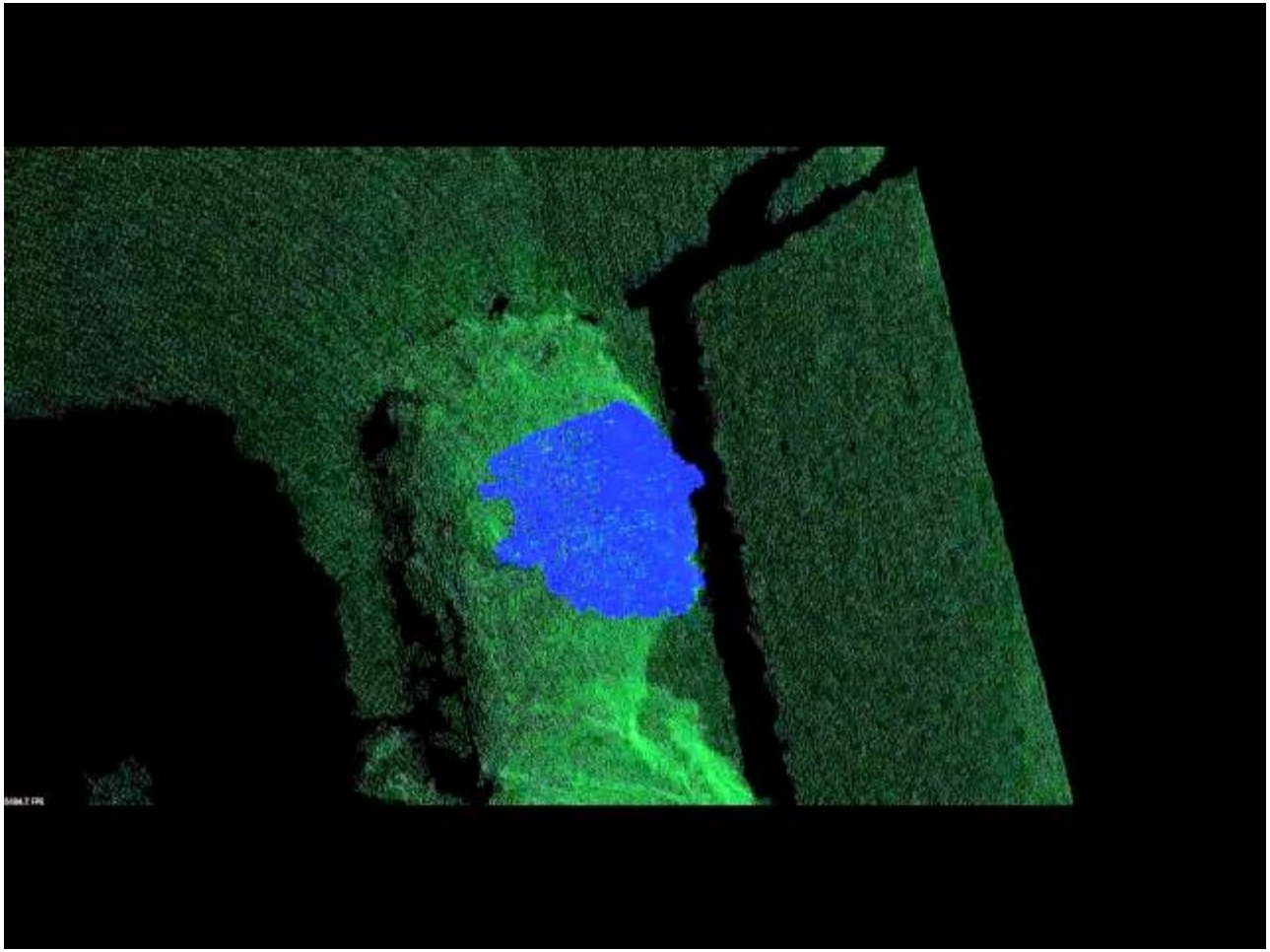
Computing Optical Flow in OpenCV

http://docs.opencv.org/3.1.0/d7/d8b/tutorial_py_lucas_kanade.html#gsc.tab=0

Registration in 3D



Overview of PCL Tutorial on Alignment



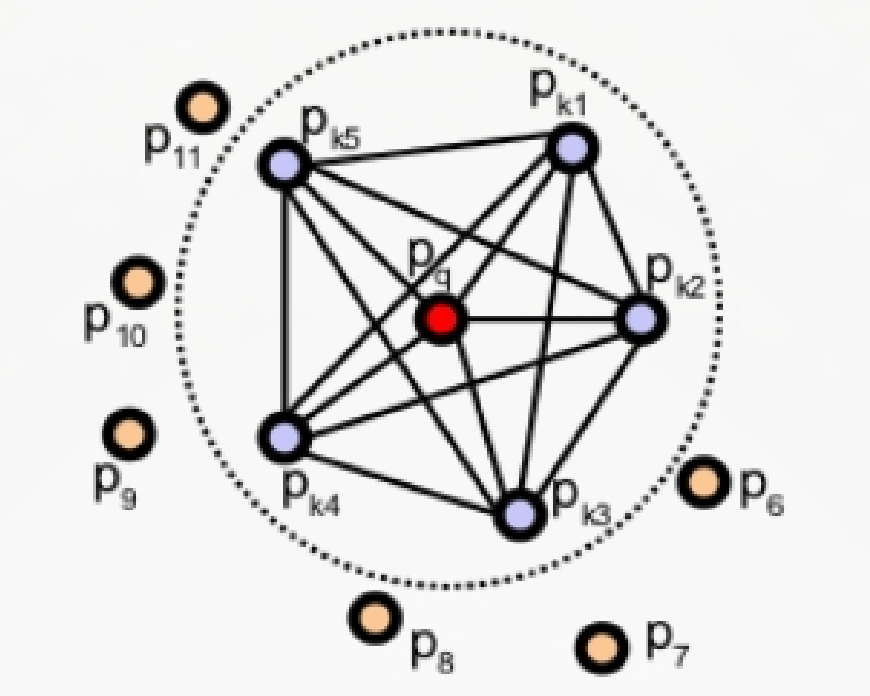
http://pointclouds.org/documentation/tutorials/template_alignment.php

What's in the 'Feature Cloud'?

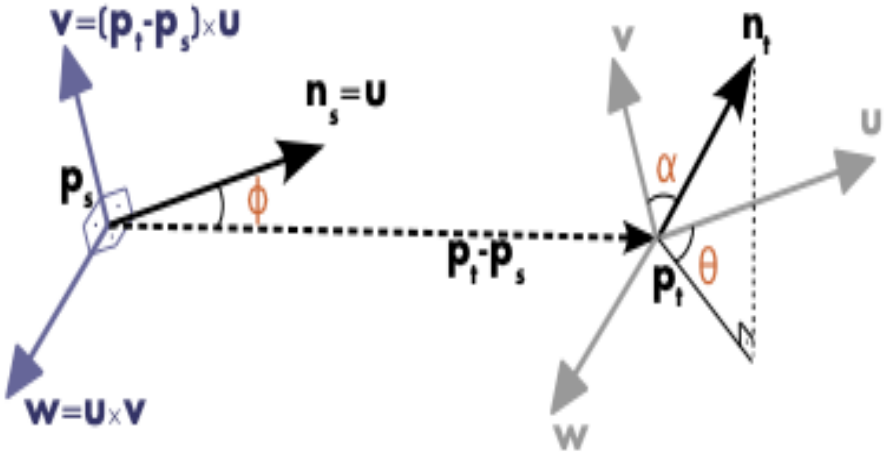
Computing Point Feature Histograms

- Main idea: compute a feature vecctor that describes the relationships between the normals of different points observed near a location in the cloud

Computing Point Feature Histograms



Computing Point Feature Histograms



$$\mathsf{u} = \boldsymbol{n}_s$$

$$\mathsf{v} = \mathsf{u} \times \frac{(\boldsymbol{p}_t - \boldsymbol{p}_s)}{\|\boldsymbol{p}_t - \boldsymbol{p}_s\|_2}$$

$$\mathsf{w} = \mathsf{u} \times \mathsf{v}$$

Resulting Features
for
each edge in graph:

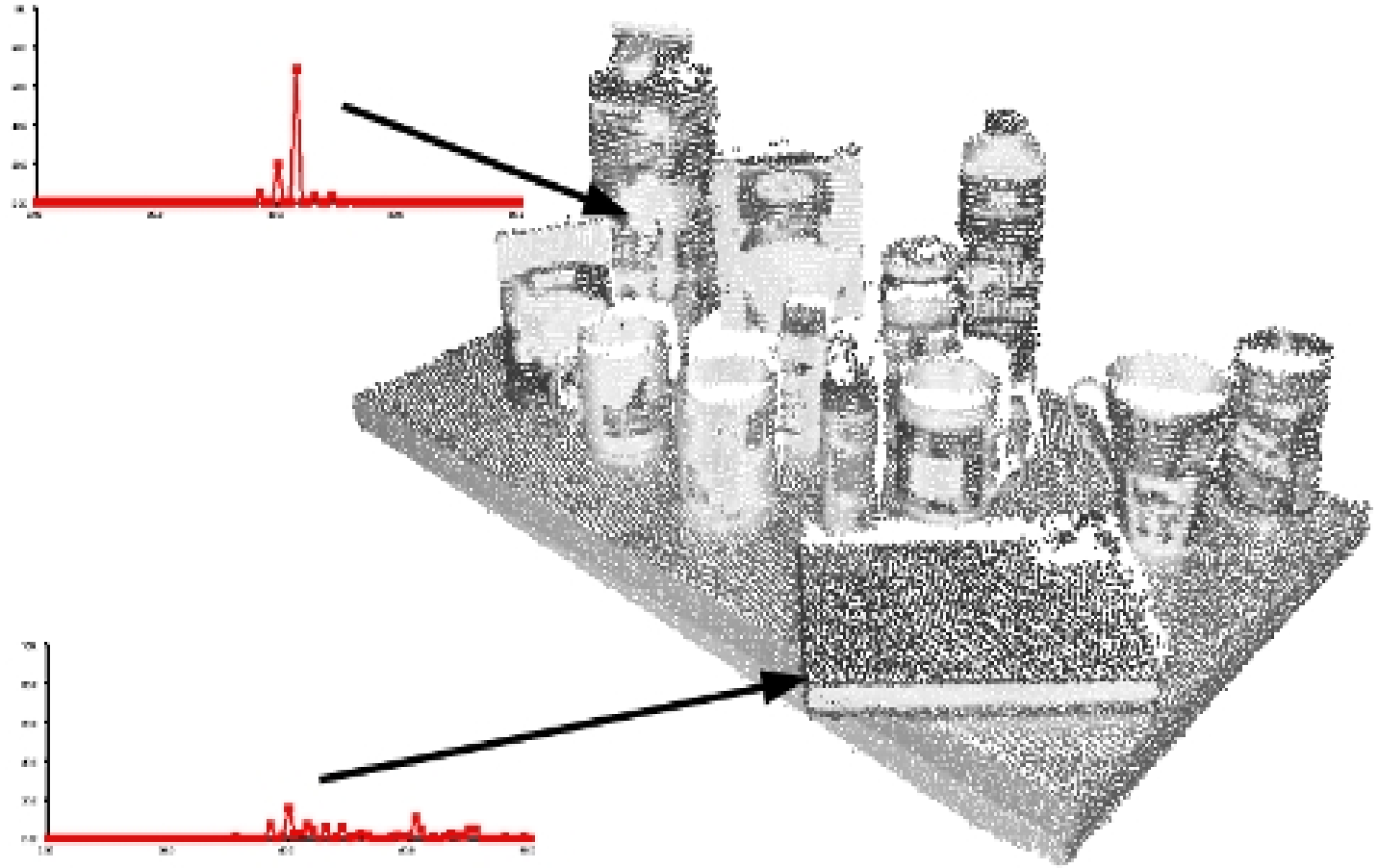
$$\alpha = \mathsf{v} \cdot \boldsymbol{n}_t$$

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

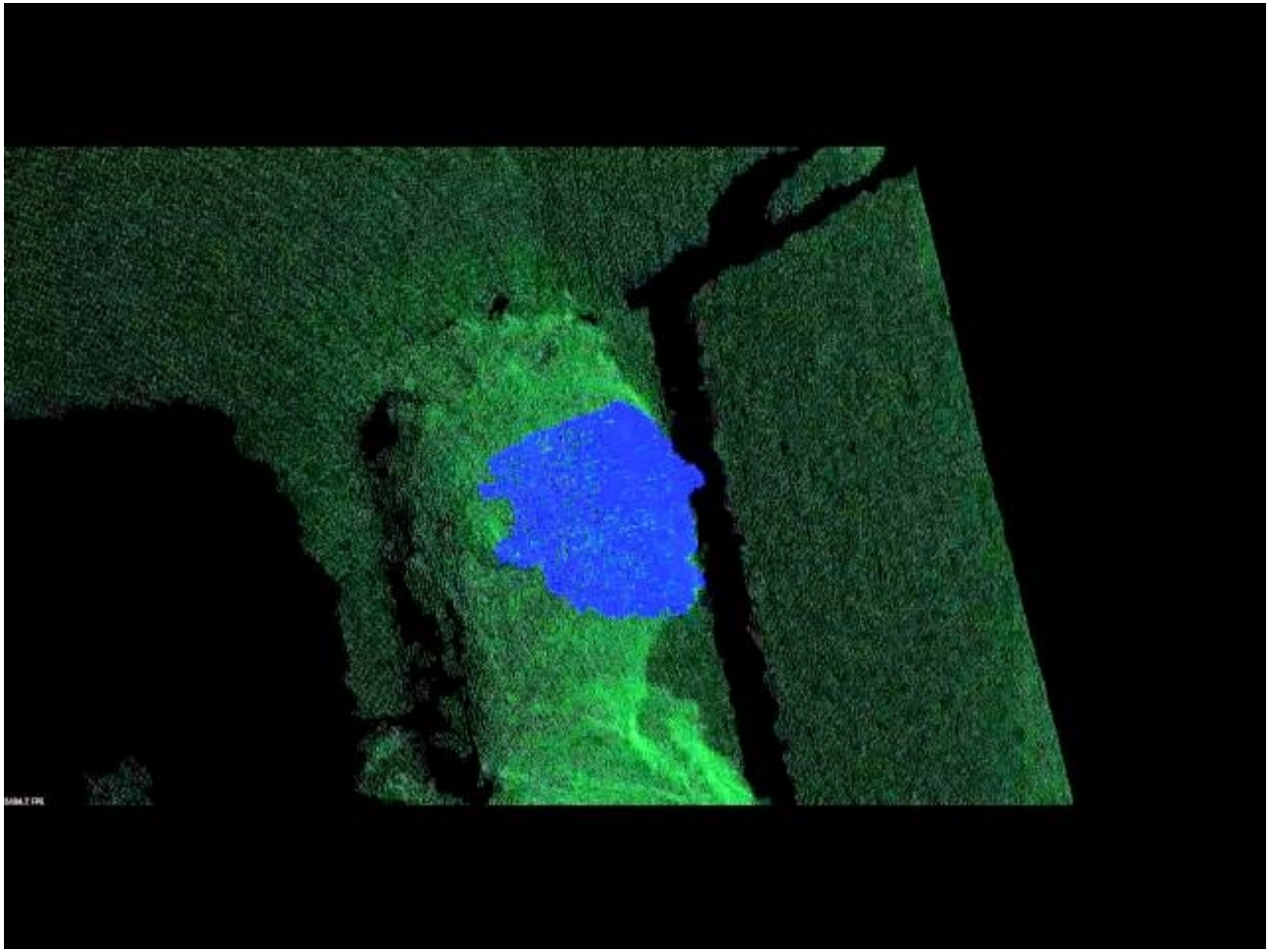
$$\theta = \arctan(\mathsf{w} \cdot \boldsymbol{n}_t, \mathsf{u} \cdot \boldsymbol{n}_t)$$

$$d = \|\boldsymbol{p}_t - \boldsymbol{p}_s\|_2$$

Computing Point Feature Histograms



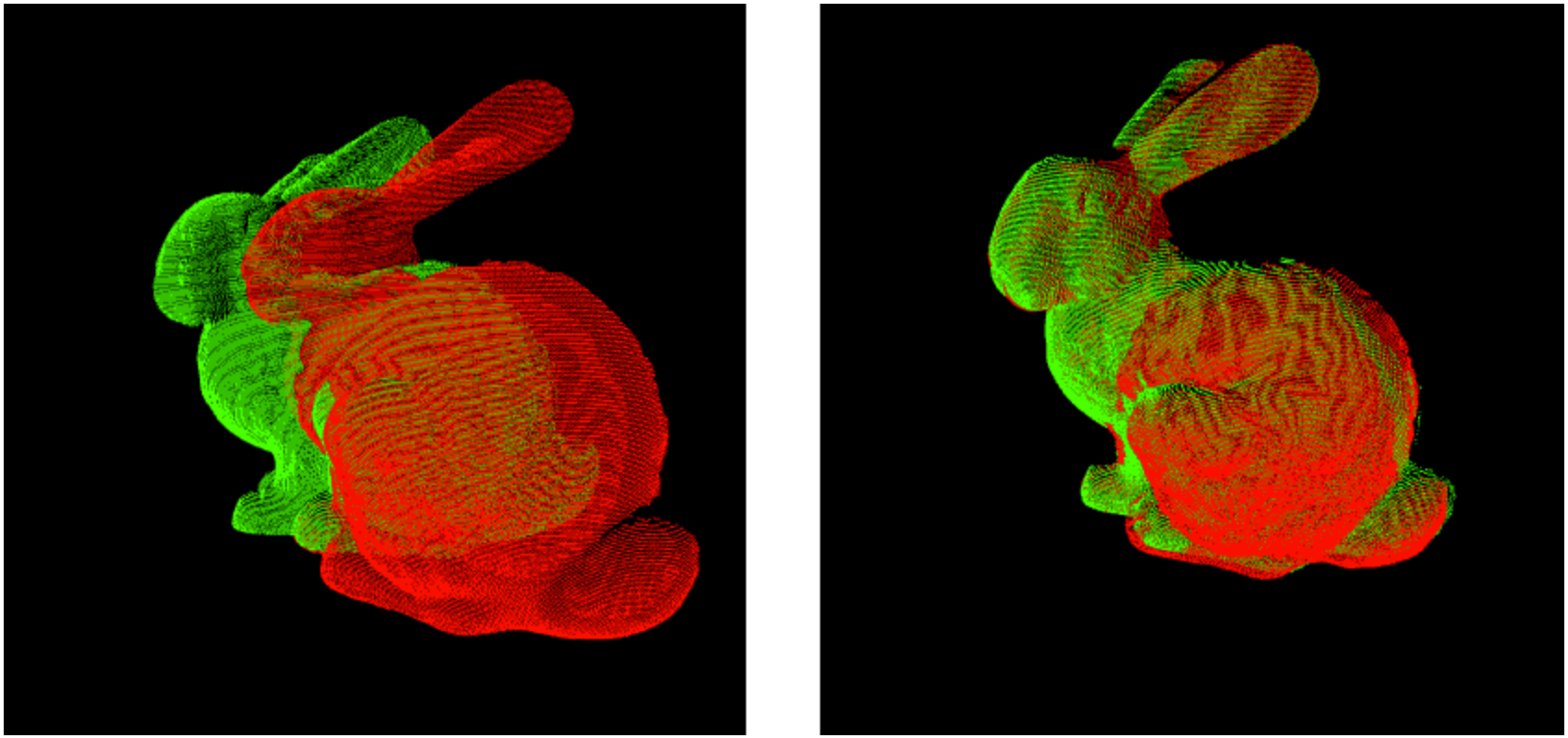
But does it really work that well??



http://pointclouds.org/documentation/tutorials/template_alignment.php

Another Example with Robot Data

Refining the Alignment using Iterative Closest Point

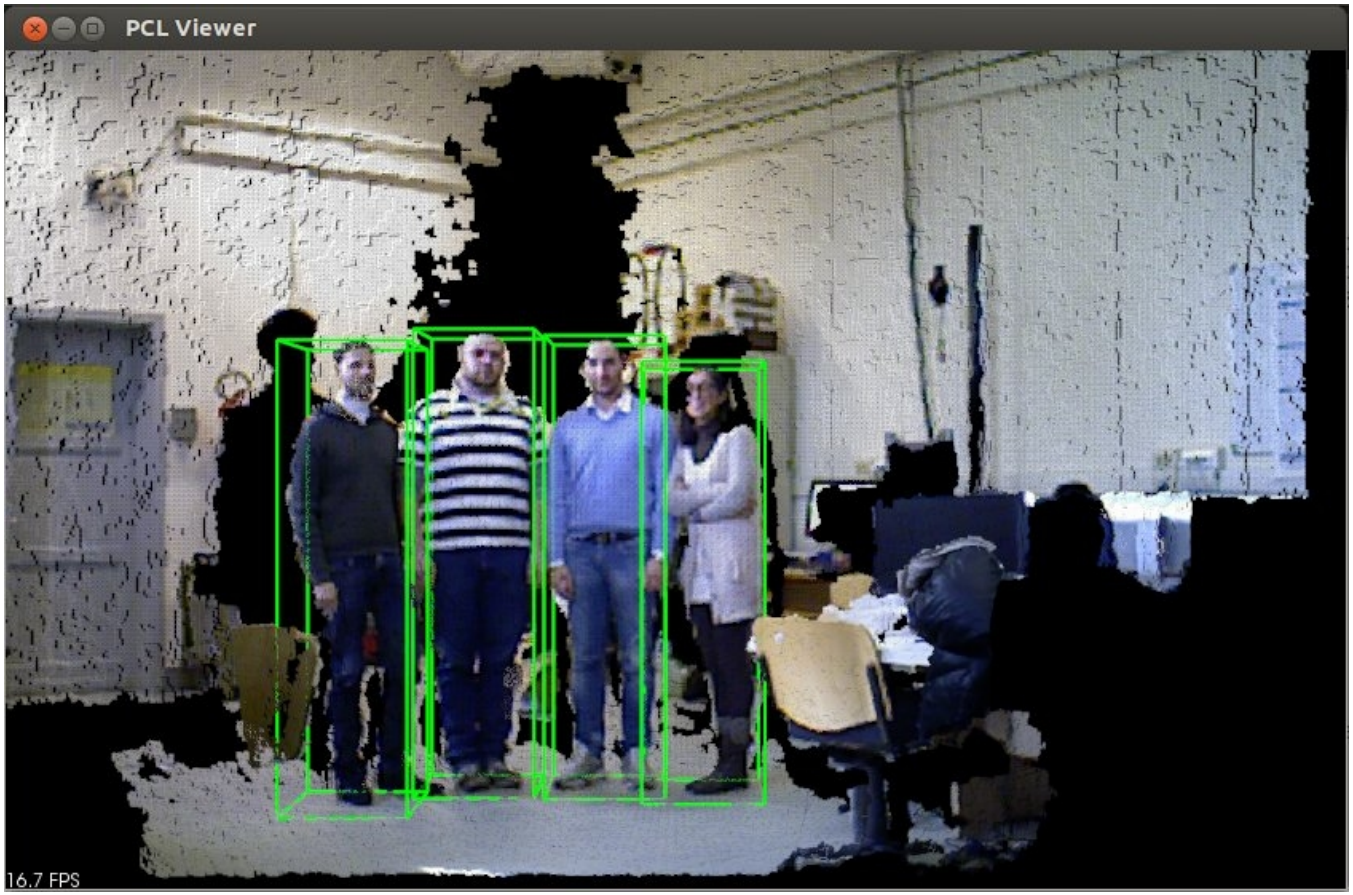


[http://vihari.github.io/personal_website/images/3dregistration.png]

Refining the Alignment using Iterative Closest Point

https://www.youtube.com/watch?v=uzOCS_gdZuM

Person Detection in PCL



http://pointclouds.org/documentation/tutorials/ground_based_rgbd_people_detection.php#ground-based-rgbd-people-detection

THE END