*Goncalves, Luis, et al. "A visual front-end for simultaneous localization and mapping." *IEEE International Conference on Robotics and Automation*. Vol. 1. IEEE; 1999, 2005.

[Paper review]

*A Visual Front-end for Simultaneous Localization and Mapping

Lab Seminar

201260290 김진형

Contents

SLAM!?

- **Simultaneous Localization And Mapping**
  - Building a map for **UNKNOWN AREA** while estimating the pose of the robot relative to the map
  - Chicken and egg problem:
    - A map is needed to localize the robot
    - A pose estimate is needed to build a map

Sensors

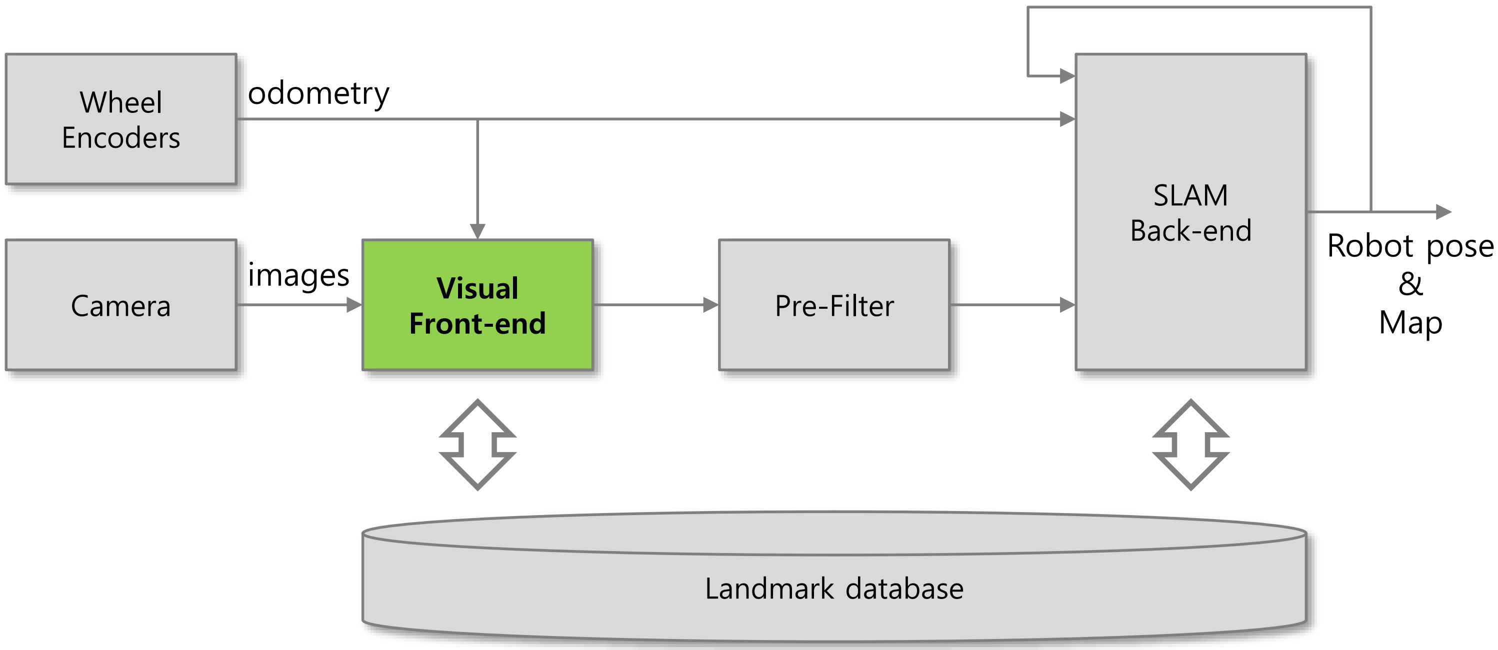
Traditional

- Sonar, Laser range finder
- Data association problem
  - Tracking multiple hypothesis over time
  - Use man-made landmarks(Beacon)

Vision

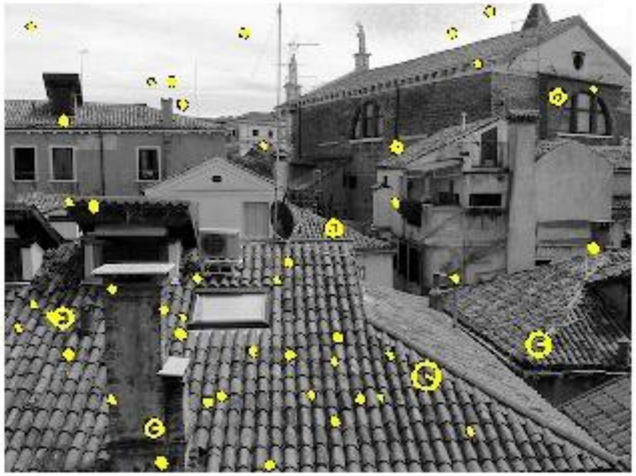
- Information-rich sensor
- Relatively cheap
- Generating unique landmarks
- Has been used sub-optimally

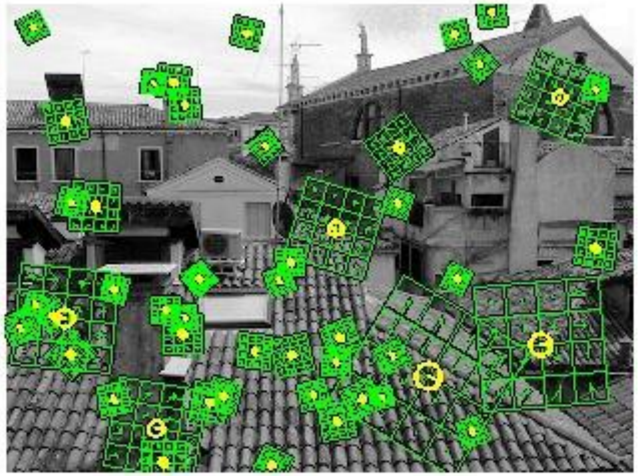
vSLAM


Wheel
Encoders
odometry
Camera
images
Visual
Front-end
Pre-Filter
SLAM
Back-end
Robot pose
&
Map
Landmark database

Object recognition core algorithm

- *SIFT(Scale Invariant Feature Transform)
  - Generates image features, "Keypoints"
    - ➢ Invariant to image scaling and rotation
    - ➢ Partially invariant to change in illumination and 3D camera viewpoint
    - ➢ Many can be extracted from typical images
    - ➢ Highly distinctive





*Lowe, David G. "Distinctive image features from scale-invariant keypoints." *International journal of computer vision* 60.2 (2004): 91-110.

Why SIFT for vSLAM?

Reliably detect landmarks upon subsequent visits to a location

Provide a dense set of reliable feature correspondences between the new view of a location and previously stored views

Good feature correspondences without feature tracking

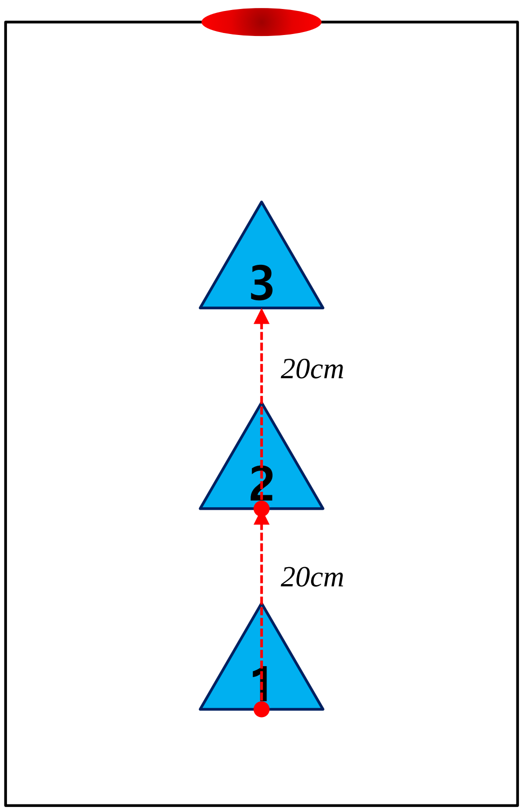
↓

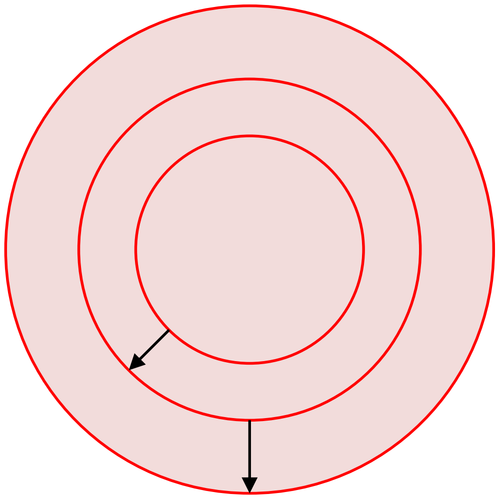
Overcome "kidnapping problem"

Building landmarks

1. Acquire 3 images while traveling apprx. 20cm btw. images
2. Use the SIFT-based object recognition algorithm to find feature correspondences
3. Solve for pose and structure of the three views
4. Store the 3D structure and the associated images and features

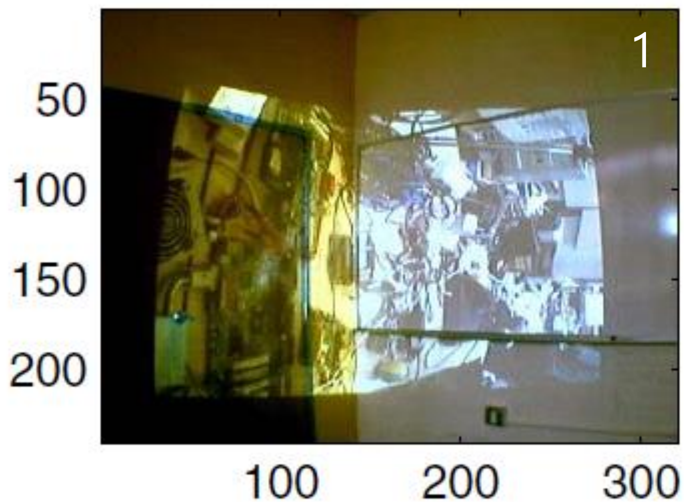
Acquiring images



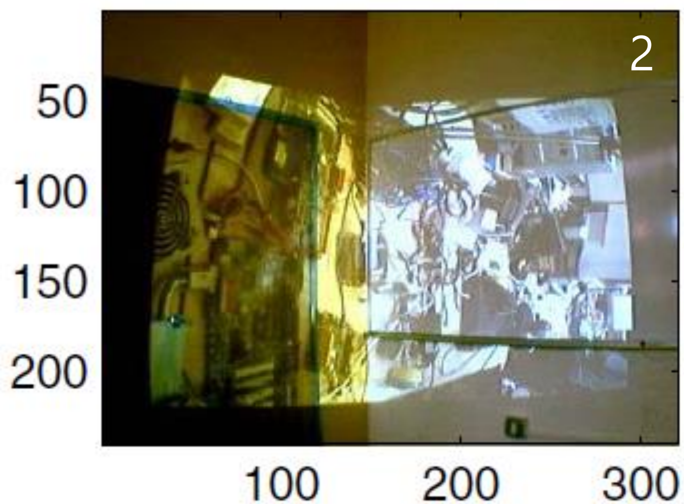


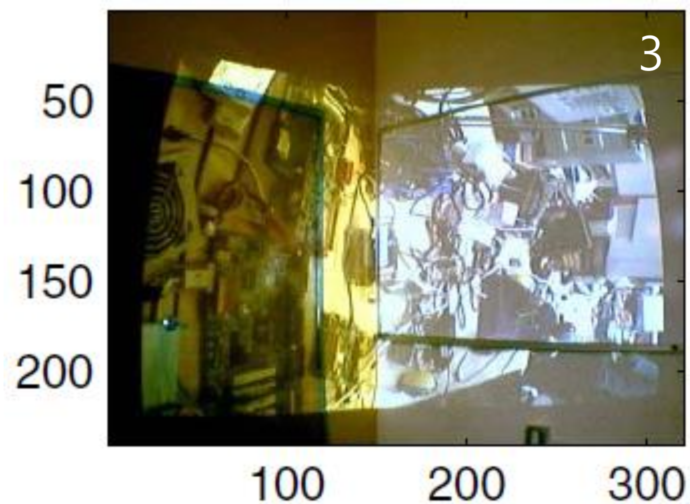
Very little disparity btw. images

Acquiring images



Projector used to texturize walls

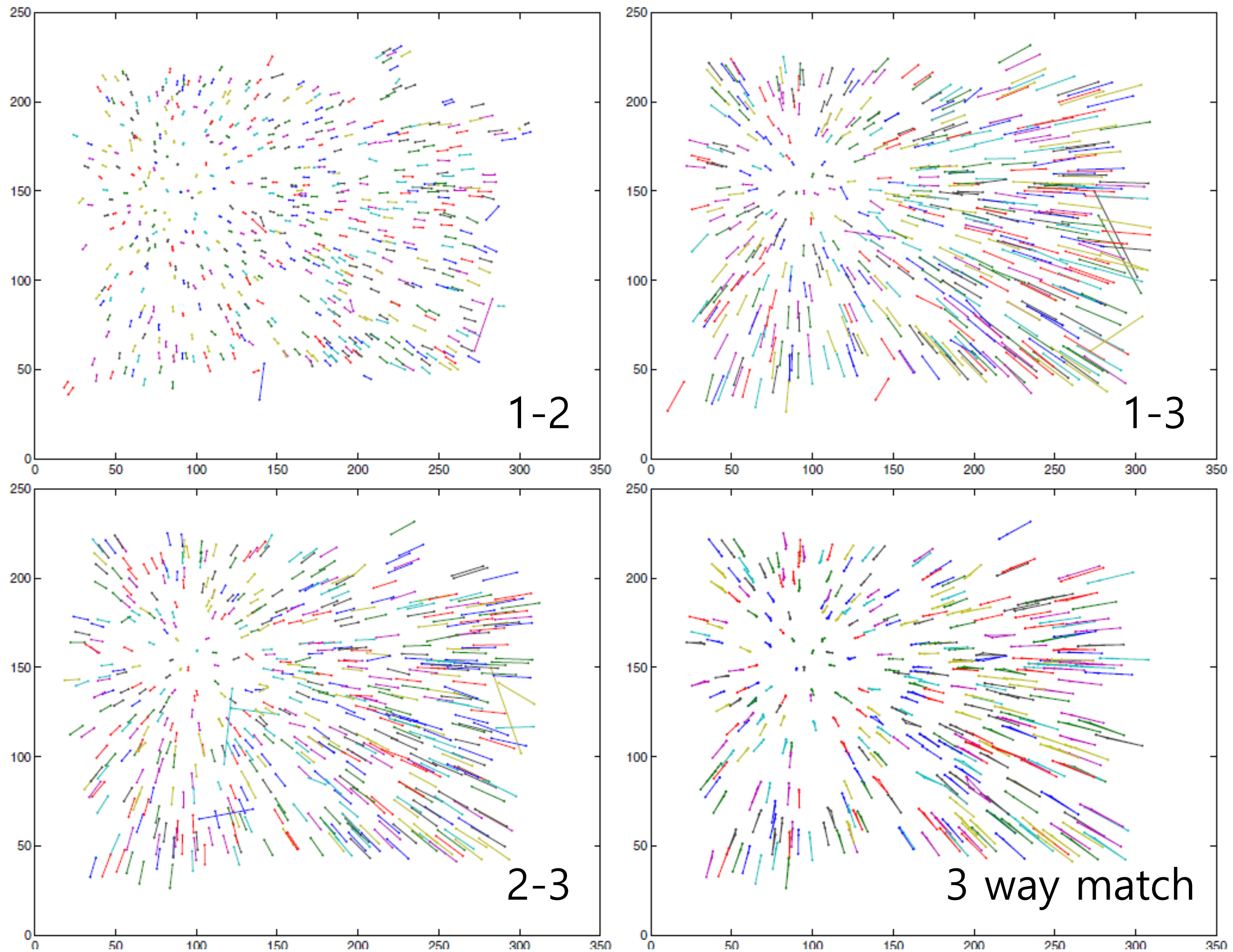




Finding image feature correspondence

- **Find the correspondences btw views 1-2, 1-3 by training the object recognition algorithm on view 1 and finding all the matching features in view 2 and 3 respectively**
- **Find the correspondences btw views 2-3 by training the recognition algorithm on view 2 and finding all the matching features in view 3**
- **Find the set of consistent three-way matches by, for each matching pair btw views 1-2**
  - Checking if the feature from view 1 exists in the list of correspondences of views 1-3
  - Checking if the feature from view 2 exists in the list of correspondences of views 2-3
  - Checking that the corresponding feature in view 3 is the same if the above two steps

Three-way match



Solving for pose and structure of three views

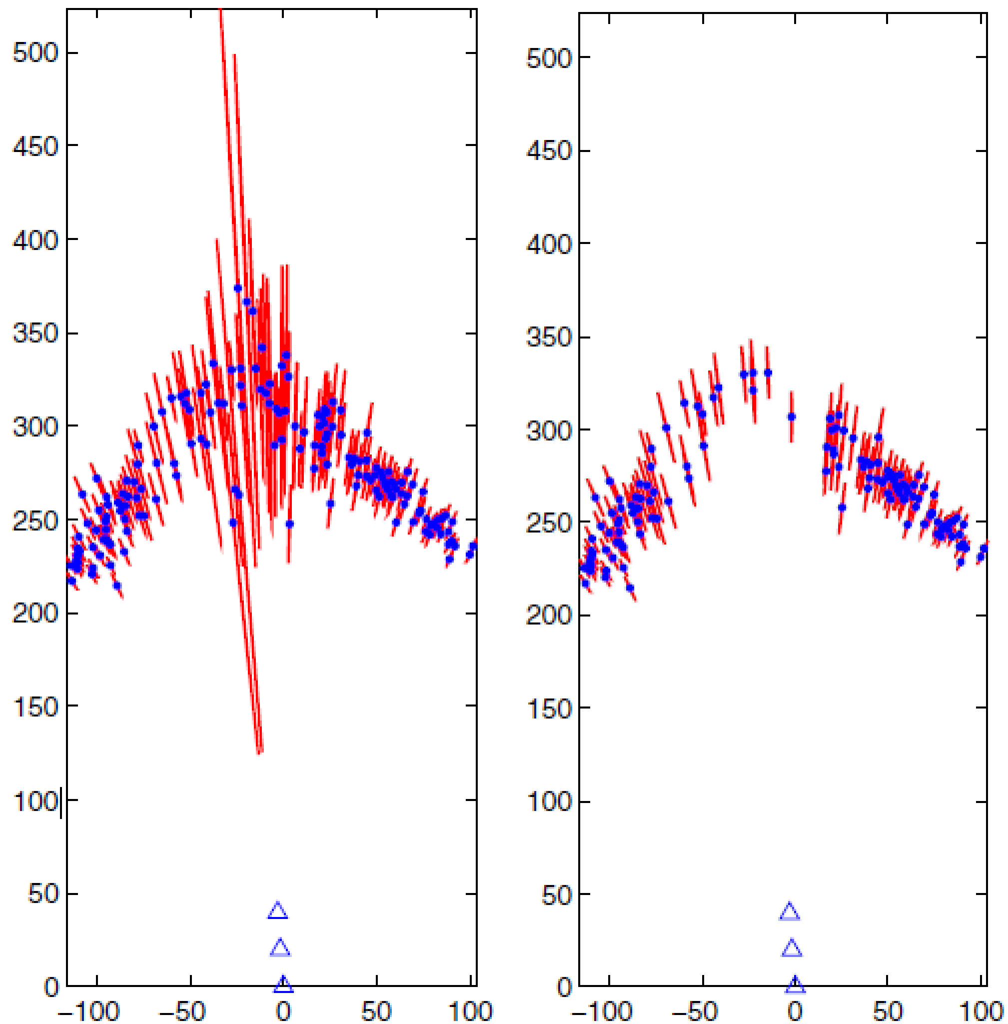
- **Compute unwarped homogeneous coordinates for the features**
  - Use camera calibration & lens calibration
- **Compute an initial estimate of the poses of the views and the 3D structure using two-view algorithm**
  - *Longuet-Higgins algorithm
- **Refine the pose and structure estimates**
  - *Nonliniear minimization algorithm : converges 5 to 10 iter.

    Optimizes the motion and the 3D structure so as to minimize the reprojection error of the 3D structure onto the three views.

*Longuet-Higgins, H. Christopher. "A computer algorithm for reconstructing a scene from two projections." *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms, MA Fischler and O. Firschein, eds* (1987): 61-62.

*Hartley, Richard, and Andrew Zisserman. Multiple view geometry in computer vision. Cambridge university press, 2003.

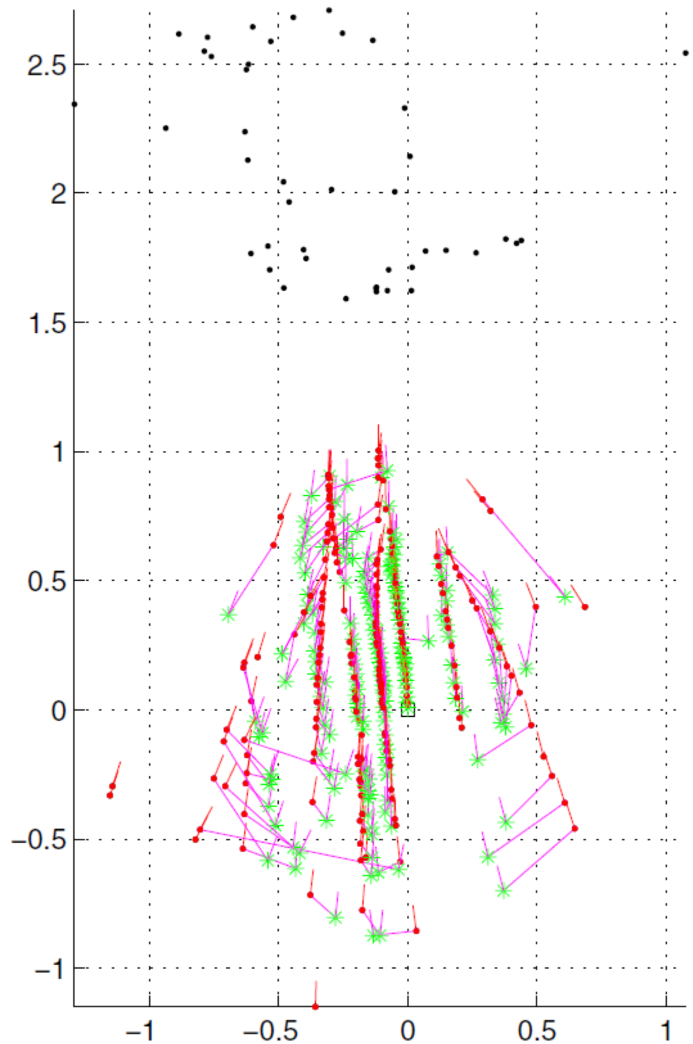
Refining pose & structure



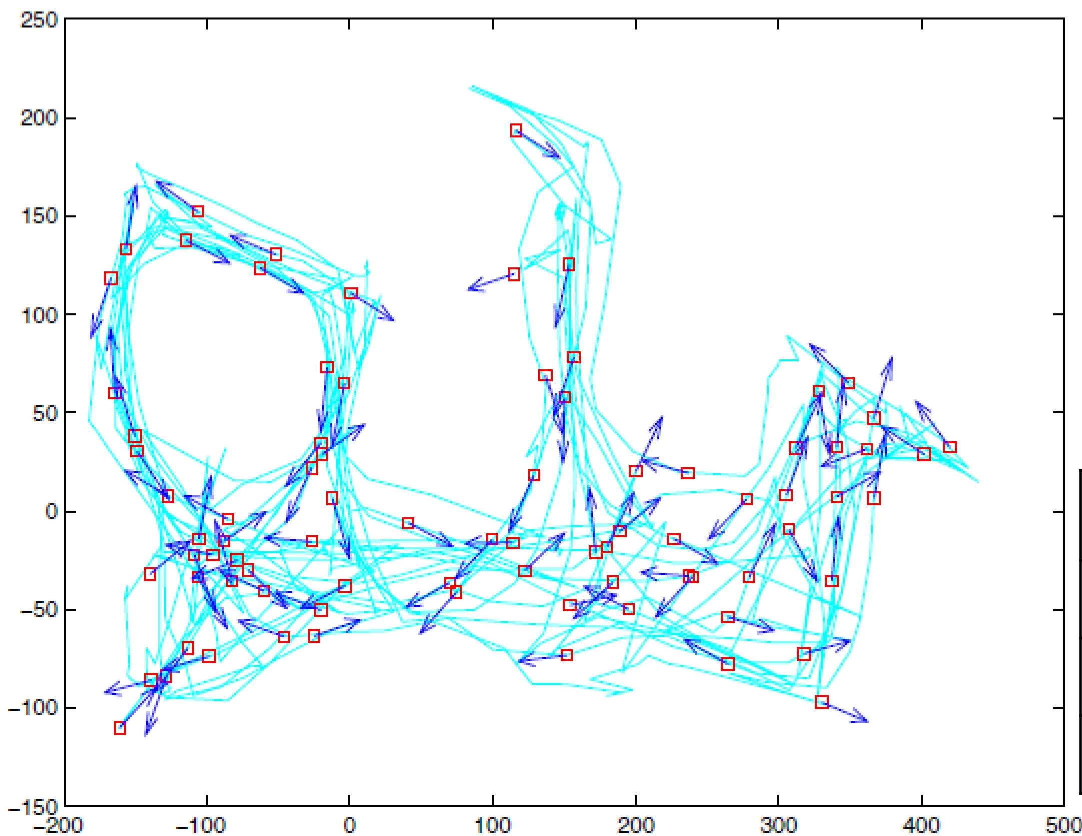
Storing 3D structure and associated features

- **The items that need to be stored in the database are:**
  - The 3D structure. This will be used to estimate the robot pose from new viewpoints
  - One of the original three views of the landmark. This will be used to identify landmarks from new viewpoints
  - The list of features in the original view of the landmark. Along with a link to the corresponding 3D structure point of the landmark

Pose estimation example 1



Pose estimation example 2



Error	Distance (cm)	Orientation (deg)
Mean	6.68	1.26
Median	10.19	1.66
RMS	13.87	2.31

Conclusion

- **Described a system to generate highly unique and recognizeable visual landmarks**
- **Estimates relative pose to a landmark with an accuracy of 10cm and 2degrees**
- **The system has implemented to run in real-time on an 400MHz embeded processor**
  - And is well suited for consumer robotics applications