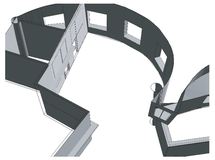
Wide-Area Egomotion Estimation from Known 3D Structure

Olivier Koch, Seth Teller

Massachusetts Institute of Technology, Computer Science and Artificial Intelligence Laboratory

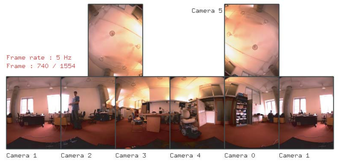
Robotics, Vision and Sensor Networks Lab

3D localization from omnidirectional video



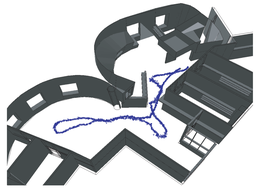
coarse 3D model

+



omnidirectional video sequence

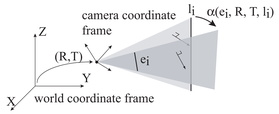
=



accurate 3D localization

- 6-DOF camera localization (translation, rotation)
- Scales to large environments (several buildings)
- Robust to clutter, lighting variations and modest changes in scene

Matching 2D image edges with 3D model line segments



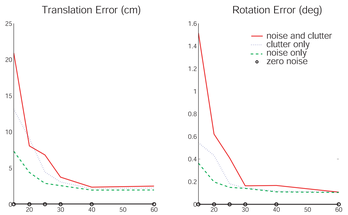
R: camera rotation
T: camera translation
e_i: image line
l_j: model line segment
α: error angle

Minimize:

$$\xi(R,T) = \frac{1}{n} \cdot \sum_{i=1}^{n} \alpha(e_i, R, T, l_i)^2$$

Correspondence = 1 model line + 1 image edge

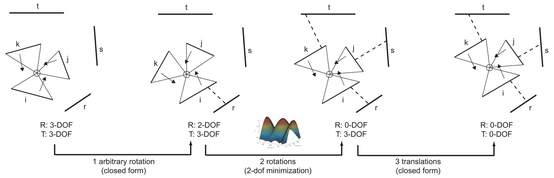
Number of correspondences needed varies with conditions.



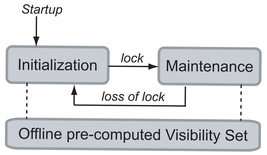
Localization error vs # correspondences

The three-line alignment algorithm

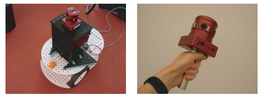
Aligning three 2D image edges (i,j,k) onto three 3D model line segments (r,s,t)



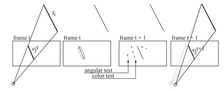
System Overview



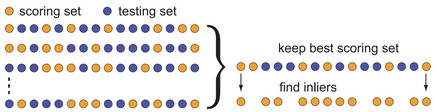
PointGrey Research Ladybug camera
75% of full sphere
6 x 1024 x 768 8-bit JPG images @ 15Hz



Maintenance: robust edge-line matching



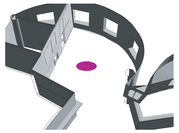
Correspondence update
Each correspondence is updated using a hue-based and an angle-based matching function.

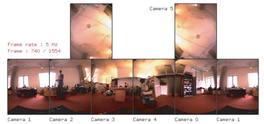


Random sample consensus
We draw a number of random correspondence sets, score them and keep the best one.
Inliers are promoted in the state machine. Outliers are demoted.

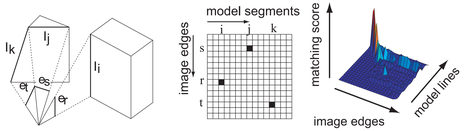
Correspondence state machine
Each correspondence is assigned a state that changes over time.

Initialization Algorithm





Input:
- estimate of the camera position
- a single omnidirectional image

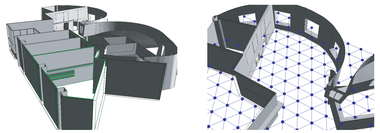


For each pair of image edges (r,s) and 3D model lines (i,j), score by comparing angles. Process all pairs and update scoring table.

Example: candidate matches on the image (right) for a given 3D model line segment (left).

Extract set of correspondences using random sample consensus.

Visibility computation using OpenGL

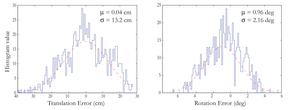


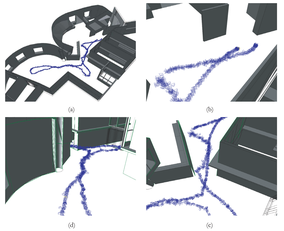
Hexagonal model subdivision (step = 1.5m)

Face visibility computation using OpenGL. Each face in the model is rendered using a unique color.

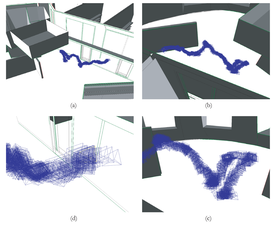
Real Datasets

	lab	corridor	handheld
Number of frames	1,500	7,800	1,900
Frame rate (fps)	5	5	15
Excursion duration (min)	5	26	2
Excursion length (m)	120	936	33
Total # of 3D segments	3,000	7,400	3,000
Total surface area (m^2)	450	7,000	450

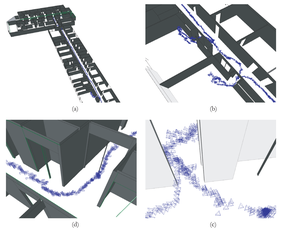




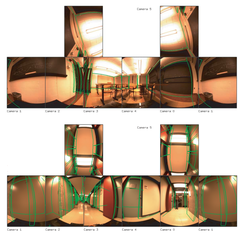
lab dataset



hand-held dataset



corridor dataset



reprojected view

Future Work

- Signature-based initialization
- Better performance (> 1Hz)
- Better accuracy (sensor fusion)
- Online update of the 3D model

Acknowledgments

We gratefully acknowledge the support of Draper Laboratory's University Independent Research and Development Program.

References

A. Ansar and K. Daniilidis. Linear pose estimation from points or lines. In Proc. ECCV, volume 4, pages 282–296, New York, May 2002.

A. Bartoli and P. Sturm. Structure from motion using lines: Representation, triangulation and bundle adjustment. CVIU, 100(3):416–441, Dec. 2005.

S. Coorg and S. Teller. Matching and pose refinement with camera pose estimates. Technical Report MIT/LCS/TM-561, MIT, 1996.

T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. IEEE Trans. PAMI, 24(7):932–946, 2002.

E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. In IEEE ICCV, volume 2, pages 1508–1515, Beijing, China, 2005.