Exploring Vision as a Basic Navigational Sensor

Key Ideas
The growing capacity of embedded processors and rapidly improving performance/cost ratio of digital cameras makes mobile computer vision systems more attractive than ever before.

Here we fit off-the-shelf vision components (webcams and Intel’s OpenCV computer vision library) to a simple robot to explore the viability of vision for local navigation.

Initial results have been promising: We achieve robust 0.5% linear accuracy and 3% angular accuracy with weakly calibrated, commodity hardware and well-known CV techniques.

Comparing Sensor Costs (ignoring cost of computation)

<table>
<thead>
<tr>
<th>Dimensionality</th>
<th>Type</th>
<th>Unit Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-D</td>
<td>Infrared Ranger</td>
<td>$10-20</td>
</tr>
<tr>
<td>0-D</td>
<td>Sonar Ranger</td>
<td>$40-80</td>
</tr>
<tr>
<td>1-D</td>
<td>Scanning Laser Ranger</td>
<td>$5000+</td>
</tr>
<tr>
<td>2-D</td>
<td>640x480 Camera</td>
<td>$20-80</td>
</tr>
</tbody>
</table>

Direction of Travel
Distant features (~ 10' away) serve as stable angular reference points during long movements.

Precipice Detection
Floor-level features should all appear to trace paths of the same length*. An abrupt decrease in the length of observed paths can be used to detect and avoid driving off a precipice, (stairs, table edge, etc.)

Precipice: apparent path of a feature at floor level
Camera view: a feature below floor level traces a shorter path (or none) as seen by the camera

(* the same length after correcting for camera perspective)

Speed / Distance
Knowledge of camera angle and height, plus an assumption that the floor is flat enable us to calculate distance traveled via simple geometry.

To make the calculation more robust in the presence of feature-tracking errors we prefilter feature paths based upon direction of apparent movement, then take the median speed indicated by the remaining paths.

Global Location and Orientation
Distant feature constellations provide stable environment-relative orientation references. Together with a map of the surrounding terrain, triangulation can then be used to determine robot/camera pose.

Swappable Feature-Tracking Subsystems
Our work so far uses the image-pyramid-based Lucas-Kanade feature tracking implementation included in OpenCV. Other algorithms can be easily substituted (SIFT, PCA, ...) to take advantage of new developments in computer vision techniques.

Future Work
• Cooperate with static camera systems (Irisnet nodes, security cameras)
  • Extend the range and resolution of a static camera system with mobile cameras
  • Use static cameras to assist navigation in dynamic environments
  • Automatically calibrate static camera systems with help of mobile platforms as reference points
• Actively participate in dynamic environments (safely interacting with people and machines)
• Share multiple cameras between navigation and other tasks
• Automatically map features usable for global navigation

Research at Intel
Copyright © 2003 Intel Corporation. All rights reserved. Intel and the Intel logo are registered trademarks of Intel Corporation and its subsidiaries in the United States and other countries.