Project 1: Line following using the e-puck’s camera

Lucie Dross, Chloé Udressy
26 May 2020

Professor: Alcherio Martinoli
Supervisors: Chiara Ercolani, Anwar Quraishi
Overview

• Introduction
  • E-puck
  • Objectives

• Methods
  • Algorithm implementation
  • Trajectory using odometry and supervisor data

• Experiments and discussion

• Conclusion
E-puck

Sensors and actuators used
• camera RGB (52x39 resolution)
• 8 IR proximity sensors
• motion sensors
• 2 wheel motors

In Webots
• non-linearities and noise of sensors and actuators approximately reproduced
Objectives

• following 4 line shapes

• reacting to line ends

• avoiding and getting around obstacles

• comparing ground-truth and odometry positions
Structure of our algorithm

Initial stage: “passive” line searching

0. 360° rotation

First stage: active line searching and obstacle avoidance

1.1 searching and going to line
1.2 obstacle avoidance

Second stage: line following and obstacle circumvention

2.1 simple line following
2.2 obstacle circumvention
2.3 sharp turn
2.4 line ending, T shaped
2.5 line ending, | shaped
2.6 line perpendicular to the trajectory of the robot
Line searching and following

• proportional controller

\[ k = \frac{m_i - m_l}{m_i} \]

\( m_i = \) middle of the image
\( m_l = \) middle of the line

• line following: bottom row (38th)

\[ v_{\text{left}} = \text{offset} - 35k_{br} - 30k_{br}^3 - 25k_{br}^5 - 15k_{br}^7 \]
\[ v_{\text{right}} = \text{offset} + 35k_{br} + 30k_{br}^3 + 25k_{br}^5 + 15k_{br}^7 \]

• line searching: 18th and 32nd rows
Obstacle avoidance

Stage 1 (line searching) \(\rightarrow\) basic obstacle avoidance
- Braitenberg avoidance

Stage 2 (line following) \(\rightarrow\) obstacle circumvention

Alternation between:
- Braitenberg avoidance
- Braitenberg attraction
- straight motion
- rotations
Supervisor

- Position in the global coordinate system
- Ground-truth trajectory effectuated by the robot
- `wb_supervisor_node_get_position` (EPUCK)

Odometry

- Data from motion sensors and initial position
- Conversion from the robot coordinate system to the global one
- Translation of wheel encoder readings into linear motion
Results

Supervisor vs Odometry data

- Supervisor Data
- Odometry Data
- * Starting point supervisor
- + Starting point odometry

Cumulative error through time

- Error [m]
- Time step [-]
Experiments

- Three experiments on the 4 line shapes in order to evaluate
  - Obstacle circumvention n°1
  - Line reaching n°2
  - Obstacle avoidance n°3

<table>
<thead>
<tr>
<th>Experiment</th>
<th>n°1</th>
<th>n°2</th>
<th>n°3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success [%]</td>
<td>63</td>
<td>78</td>
<td>73</td>
</tr>
<tr>
<td>Failure [%]</td>
<td>37</td>
<td>22</td>
<td>27</td>
</tr>
<tr>
<td>Tries</td>
<td>57</td>
<td>60</td>
<td>60</td>
</tr>
</tbody>
</table>
Weaknesses and improvements

• **Constraints linked to the robot**
  • Noise on distance sensors
  • Delay between the camera image and the robot position
  \[\text{low-pass filter}\]

• **Obstacle circumvention**
  • Alternance of attraction and avoidance behaviours
  \[\text{single set of Braitenberg coefficients mixing attraction and avoidance depending on the distance sensor}\]

• **Line acquiring**
  • Shadows
  \[\text{enhancing the line_or_shadow function}\]
From Webots to reality

• Perception-to-action loop: delays!
• Memory much more limited
• Odometry errors more important (wheel diameters, misalignement…)
Conclusion
Thank you for your attention!