SIS project

Esther Barberis and Manuel Walser
28.05.2020
Project goal

Implement a C program for the E-puck to exit the maze by reading road signs

- Horizontal/Left turn
- Vertical/Right turn
- Black sign/ turn around
horizontal stripes
- high variance of the amplitude for the columns
- only one peak for the rows.
- Ratio col/rows = 2.222222e+00

vertical stripes
- high variance of the amplitude for the rows.
- only one peak for the columns.
- Ratio col/rows = 0.5154005
In the case of the noisy image, we notice on the graphs more ambiguity between the amplitudes for the rows and the columns.

Vertical: Ratio columns/sums = 0.98

Horizontal: Ratio columns/sums = 1.017
Amplitude row

• For every row of the image we applied:

\[ \text{mean of rows} = \sum_{m=0}^{\text{camera width}} \frac{\text{fft of image}(m)}{\text{camera width}} \]

• Finally, we got the Amplitude of the mean of rows and calculating the over all mean

\[ \text{Amplitude row} = \frac{\text{real}((\text{mean of rows})^2 + \text{imaginary}((\text{mean of rows})^2)}{\text{camera height}} \]
1. Get mean of each row (see image)
2. Get amplitude for each mean calculated before
3. Get mean amplitudes calculated in 2
4. If Amplitude of row < Amplitude of columns → Horizontal, turn left
Optimizations

- Use of a filter, to get mean over neighbour cells to smooth out noise.

```c
for (j=0; j<camera_height; j=j+1) {
    for (i=0; i<camera_width; i=i+1) {
        image[j][i] = signal_data[j][i];
    }
}

for (j=2; j<camera_height-2; j=j+1) {
    for (i=2; i<camera_width-2; i=i+1) {
        som = 0;
        for (y=-2; y<=2; y=y+1) {
            for (x=-2; x<=2; x=x+1) {
                som = som + image[j+y][i+x];
            }
        }
        signal_data[j][i] = (int) som/25.0;
    }
}
```
Criteria for the ratio rows/column:

- if \( \frac{\text{mag col}}{\text{mag rows}} < 0.6 \rightarrow \text{vertical, turn right} \)
- if \( \frac{\text{mag col}}{\text{mag rows}} > 0.6 \rightarrow \text{horizontal, turn left} \)
- if \( \frac{\text{mag col} + \text{mag rows}}{2} < \text{BlackThreshold} \), turn around
Different Scenarios tested

- Start 1 and 2
- Clear and noisy images
- Camera noise
- fft processing with filter
Noisy images/With black sign into account

Noisy

Success Failure
Noisy images/Without black sign into account

Noisy

- Success
- Failure
# Time [s] for start 1

<table>
<thead>
<tr>
<th>Failures</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear images /No filter</td>
<td>54.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clear images/With filter</td>
<td>54.54</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Noisy/No filter</td>
<td>63.4</td>
<td>93.55</td>
<td>126.3</td>
<td></td>
</tr>
<tr>
<td>Noisy/ filter</td>
<td>63.26</td>
<td>90</td>
<td>124.25</td>
<td></td>
</tr>
</tbody>
</table>
Odometry

Implementation in webots:

- $\varphi = 2 \pi \left( \frac{N_{\text{turns per wheel}}}{N_{\text{total of turns}}} \right) = \text{wb\_position\_sensor\_get\_value()}$
- Distance covered by the left wheel $d_l = \frac{\varphi(\text{left})}{\Delta t} \cdot \text{wheel radius}$;
- Distance covered by the right wheel $d_r = \frac{\varphi(\text{right})}{\Delta t} \cdot \text{wheel radius}$;
- The mean distance of both tyres $d_{\text{is}} = \frac{d_l + d_r}{2}$;
- Delta orientation = $\theta' = \frac{d\theta}{\text{(axle length)}}$;
- $x(t) = x(t - 1) + \int (d_{\text{is}} \cdot \cos(\theta)) \cdot dt$
- $y(t) = y(t - 1) + \int (d_{\text{is}} \cdot \sin(\theta)) \cdot dt$

\[ \dot{\xi}_I = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{r \dot{\phi}_1}{2} + \frac{r \dot{\phi}_2}{2} \\ \frac{-r \dot{\phi}_1}{2l} + \frac{-r \dot{\phi}_2}{2l} \end{bmatrix} \]
Supervisor/Odometry results – start 2
What went wrong/What could have been done better?

- Implement a high pass filter on the 2D array
- Using the fft with 1D array would have been perhaps simpler in terms of filtering
- Take the ratio of the maximum amplitude instead of the mean of magnitudes for the conditions
- Problem with the first sign in webots in start 1, with z variable.