Rapid Classification of Vehicle Heading Direction with Two-Axis Magnetometer

Niklas Wahlström¹
Roland Hostettler²
Fredrik Gustafsson¹
Wolfgang Birk²

¹Division of Automatic Control
Linköping University
Linköping, Sweden

²Division of Systems and Interaction
Luleå University of Technology
Luleå, Sweden
Background

Traffic monitoring in wireless sensor network

- Sensor nodes equipped with a magnetometer.

Limitations:

- Energy budget
- Computational resources.

Information you can extract

- Number of vehicles
- Type of vehicle
- Heading direction
Problem formulation

- 2-axis magnetometer has been deployed on the roadside

- Magnetometer measures a distortion of the magnetic field.

**We want to classify the heading direction of the vehicle!**
The vehicle can be modeled as a magnetic dipole:

\[
\mathbf{h}(t) = \frac{3(\mathbf{r}(t) \cdot \mathbf{m})\mathbf{r}(t) - \|\mathbf{r}(t)\|^2 \mathbf{m}}{\|\mathbf{r}(t)\|^5}
\]

We measure two components of the magnetic field.

\[
\mathbf{y}(kT) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{h}(kT) + \mathbf{e}(kT), \quad k = 1, \cdots, N
\]
Simulated examples

Three different vehicle are heading in positive $x$-direction

Niklas Wahlström, Roland Hostettler, Fredrik Gustafsson, Wolfgang Birk
Rapid Classification of Vehicle Heading Direction with Two-Axis Magnetometer
Simulated examples

Three different vehicles are heading in positive $x$-direction.

All measurement trajectories are turning clockwise!
Idea:

Classify heading direction by the turn of the measurement trajectory!
Real world data
Real world data

Classify driving direction by the turn of the measurement trajectory!

Niklas Wahlström, Roland Hostettler, Fredrik Gustafsson, Wolfgang Birk
Rapid Classification of Vehicle Heading Direction with Two-Axis Magnetometer
The theorem

Assume the magnetic dipole model

$$h = \frac{3(r \cdot m)r - \|r\|^2m}{\|r\|^5}$$

then

$$\begin{vmatrix} r^x & dr^x \\ r^y & dr^y \end{vmatrix} > 0 \iff \begin{vmatrix} h^x & dh^x \\ h^y & dh^y \end{vmatrix} > 0$$

Observe: Independent of $m$!
The classifier

Integrate over all infinitesimal area segments

\[ f = \int \begin{vmatrix} h_x^x & dh^x \\ h_y^y & dh^y \end{vmatrix} = \int \begin{vmatrix} h_x^x(t) & dh^x(t)/dt \\ h_y^y(t) & dh^y(t)/dt \end{vmatrix} dt. \]

The time discrete version will then be

\[ f = \sum_{k=1}^{N-1} \begin{vmatrix} h_x^x_k & (h_x^x_{k+1} - h_x^x_k)/T \\ h_y^y_k & (h_y^y_{k+1} - h_y^y_k)/T \end{vmatrix} T \]

\[ = \sum_{k=1}^{N-1} (h_x^x_k h_y^y_{k+1} - h_x^x_k h_y^y_k) \]

\[ = (h_x^x_{1:(N-1)})^T h_y^y_{(1+1):N} - (h_y^y_{1:(N-1)})^T h_x^x_{(1+1):N} \]
The classifier

- Sum over all triangles
- The enclosed area can be computed as two inner products!

\[ \hat{f} = \left(y^x_{1:(N-1)}\right)^T y^y_{(1+1):N} - \left(y^y_{1:(N-1)}\right)^T y^x_{(1+1):N} \]

- The sign of \( \hat{f} \) determines the heading direction.

Note: \( h \) has been replaced with the measurement \( y \) which contains noise.
The improved classifier

The variance of $\hat{f}$ can be reduced by trading for some bias.

- Idea: Average over larger triangles!

$$
\hat{f}_p = (y^{x}_{1:(N-p)})^T y^{y}_{(1+p):N} - (y^{y}_{1:(N-p)})^T y^{x}_{(1+p):N}
$$

Observe: The feature still only consists of two inner products!
### Experimental results

- **2 sensor nodes**
- **45 min**
- **88 vehicles travelling south-north**
- **99 vehicles travelling north-south**

Correct classification by the two sensors

<table>
<thead>
<tr>
<th></th>
<th>South-North (Sensor 1)</th>
<th>North-South (Sensor 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor 1</td>
<td>87/88</td>
<td>91/99</td>
</tr>
<tr>
<td>Sensor 2</td>
<td>82/88</td>
<td>99/99</td>
</tr>
</tbody>
</table>
Summary

Vehicle heading direction classification using a 2-axis magnetometer.

- A two-fold classification problem.
- One strong feature has been derived and extracted from data.
- It is fast (difference of two inner products)
- Theoretical justification is provided.
- It works on real world data with good results
Vehicle heading direction classification using a 2-axis magnetometer.

- A two-fold classification problem.
- One strong feature has been derived and extracted from data.
- It is fast (difference of two inner products)
- Theoretical justification is provided.
- It works on real world data with good results

⇒ Fast and accurate classifier for this application!