Using Computer Vision to Process Vehicle Dashboard Displays in Transportation Safety Research

Kristin Jiating Chen, Alexander Cates, Rick Huey, Marcelo Simas, James Jenness, Gonzalo Rivero
Outline

• Introduction
• Nature of data
  ▪ Videos
  ▪ Icons
• Methodology – Machine learning pipeline with OpenCV
• Results
  ▪ Model performance
  ▪ Future research
Introduction – Background of the Overall Study

• Understand driver behavior in the context of driver-assist systems in Toyota Safety Sense system (TSS)
  ▪ Driver-assist systems
    *Adaptive Cruise Control*  *Lane Departure Alert*  *Pre-Collision System*
  • when the systems are triggered, related driver-assist icons appear on the vehicle dashboard displays

When driver-assist icons were presented  When driver-assist systems were used  How drivers interpret and respond to the presence of the icons  How drivers understand systems over time
Introduction – Research Problem

• Problem
  ▪ Identify the presence of icons on the vehicle dashboard displays

• Pilot study
  ▪ Collect data by recording the central dashboard displays while driving instrumented Toyota vehicles
Nature of Data - Videos

• Video recordings of central dashboard display
  ▪ Pilot study data
    • 200+ 1-min videos
  ▪ Study data (estimated)
    • 10-20 instrumented Toyota vehicles
    • 12 weeks participation per vehicle
      – 500+ 1-min videos per week
Nature of Data - Icons

- 10 icons

Adaptive Cruise Control  Lane Departure Alert

Leading Car  Headway Bar Indication  Lane Line Indication  Brake
Methodology

• Technique
  ▪ OpenCV: open source computer vision library
• Machine learning Pipeline utilized OpenCV Python API and R

Frame extraction from videos ➔ Feature matching between icons and extracted frames (image preprocessing) ➔ Visualization
Methodology – Frame Extraction

- Extract frames from videos per half-second
Methodology – Feature Matching

- Feature
- Good feature – corner feature

Corner features (E, F)  Flat features (A, B)

Edge features (C, D)
Methodology – Feature Matching

1. Detect features in icons
2. Match same features between two images
Methodology – Feature Matching

• Matching Criteria
  ▪ 3 matched features between two images
Methodology – Feature Matching

- Scale invariant
- Rotation invariant
Methodology – Feature Matching

• Mismatching
  ▪ Local features detected in the frames are noise

• Strategy
  ▪ Lower noise and narrow target areas for feature matching
Methodology – Image Preprocessing

• Crop the dashboards from the frames - focus on region-of-interest
• Deskew the dashboards
• Denoise
Methodology – Image Preprocessing

• Crop into 3 parts
  ▪ Narrow down the target area

Adaptive Cruise Control

Lane Departure Alert
Methodology – Feature Matching
Methodology – Visualization (R Shiny)

- Interactive time series plots
Results – Modeling Performance

• Sample data: 10 icons, 29 selected frames, 290 combinations

• Confusion Matrix

<table>
<thead>
<tr>
<th>n = 290</th>
<th>Predicted Negative</th>
<th>Predicted Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual Negative</td>
<td>TN = 213</td>
<td>FP = 4</td>
</tr>
<tr>
<td>Actual Positive</td>
<td>FN = 25</td>
<td>TP = 48</td>
</tr>
<tr>
<td></td>
<td>238</td>
<td>52</td>
</tr>
</tbody>
</table>

TPR: 0.66  TNR: 0.98  FPR: 0.018  FNR: 0.342

• Runtime (4 threads multiprocessing)
  ▪ Week 1 subject 1 real study data: 10 icons, 57,000 frames, 570,000 combinations (500+ 1-min videos)
  ▪ Computation time

Frame extraction: 10 mins  Image preprocessing: 2 hours+  Feature matching: 2 hours+
• Conclusion: Works well!
  - Leads to 66% true positive rate and 98% true negative rate
  - Meets computation requirement when data scales up
  - Processes with reasonable computation time

• Current research
  - Customizes matched features criteria for each icon
    - Balance FP/FN error

• Future research
  - Train customized model
Thank you for listening!

- **Contact information**
  
  Kristen Chen [KristinChen@Westat.com](mailto:KristinChen@Westat.com)
 
  Alexander Cates [AlexanderCates@Westat.com](mailto:AlexanderCates@Westat.com)

- **Citation**


  [https://www.toyota.com/safety-sense/featurevideo/DtoqSOUSR0A](https://www.toyota.com/safety-sense/featurevideo/DtoqSOUSR0A)

  [https://docs.opencv.org/3.4/db/d27/tutorial_py_table_of_contents_feature2d.html](https://docs.opencv.org/3.4/db/d27/tutorial_py_table_of_contents_feature2d.html)