Towards a Mobile-based ADAS Simulation Framework

João Gonçalves et. al.

May 15+16, 2014 - Berlin-Adlershof, Germany
Agenda

1. Background
   1.1. ADAS
   1.2. Mobile-based ADAS
   1.3. Distributed Mobile-based ADAS

2. Proposal
   2.1. GeoStream
   2.2. Driving Simulators
   2.3. SUMO
   2.4. Mobile-based ADAS

3. Preliminary verification

4. Conclusions & Future Work
1.1 Advanced Driver Assistance Systems (ADAS)

- Navigation Systems (GPS)
- Adaptive cruise control
- Blind spot detection
- Traffic sign recognition
- Intelligent speed adaptation
- Automatic parking
- Lane departure warning system
- Collision avoidance system
- Driver drowsiness detection
1.2 Mobile-based ADAS

- Huge number of mobile devices (increasing)
- A lot of unexplored helpful applications
- Easy and cheap setup
- Higher penetration

Fig. 1: Mobile-based ADAS

1.2 Testing Mobile-based ADAS

- How to test them safely and in a low-cost environment?
- Most simulation systems are complex or expensive!
  (Driving simulators)
1.3 Distributed Mobile-based ADAS

- Seen as a single ADAS by the user
- Send feedback to the network (requires connectivity)
- Improve the overall reliability of the ADAS

Fig. 2: Distributed ADAS (Waze)
1.3 Testing Distributed Mobile-based ADAS

How to test them?

(ADAS problems)^n :(
2. Proposal

- SUMO, IC-DEEP ext., High Fidelity Simulators
- MAS
- Human Factors Analysis
- Open/processable data

Fig. 3: Proposal’s architecture
2.1 GeoStream (OSM Import)

- Create environments that resemble reality
- Seamless import from OSM to Driving Simulators
- However SUMO network import is more complex (JOSM? Proprietary-Open GIS?)

Fig. 4: Data import to Unity3D engine
2.2 Driving Simulators

- DRIS High-Fidelity Simulator
- IC-DEEP low cost simulator (Unity3D)
- Share the simulation state
2.3 SUMO Coupling (Work in Progress)

Requirements
- Synchronize simulation state
- Coherent simulation representation
- Human-in-the-loop simulation
- Include ADAS testing capabilities

Challenges
- Allow latitudinal movement (lane “freedom”)
- Possible communication bottleneck?
2.4 Mobile-based ADAS (GPS Mocking)

- Bound service receives socket communications
- Changes the device status
- Noticeable by all running applications (even Google Navigation)

Fig. 7: Mobile ADAS architecture
3.1 Preliminary verification (GeoStream & IC-DEEP)

- Real GPS logging driving at Porto’s downtown.
- Cross-validate results in our simulator with Google Earth
- Reproduce the circuit in the simulator

Fig. 8: GPS logs analysis
3.2 Preliminary verification (ADAS testing)

- Driving statistics meet those of the driving simulator (speed and distance)
- Successful coupling and usage of other system apps (Google Navigation)

Fig. 9: Developed test ADAS
4.1 Testing Mobile-based ADAS

- How to test them safely and in a low-cost environment?
- Most simulation systems are complex or expensive!
  (Driving simulators)
4.1 Testing Mobile-based ADAS

- How to test them safely and in a low-cost environment?
- Most simulation systems are complex or expensive!
  (Driving simulators)

Extend IC-DEEP with ADAS testing capabilities
4.2 Testing Distributed Mobile-based ADAS

How to test them?

$(\text{ADAS problems})^n :( $
4.2 Testing Distributed Mobile-based ADAS

How to test them?

\[(\text{ADAS problems})^n : (\text{ADAS problems})^n\]

Mobile-based ADAS Simulation Framework

(SUMO + IC-DEEP extension + MAS)
4.3 Conclusions

- Successfully tested Mobile-based ADAS
- Testing Distributed Mobile-based ADAS is a challenge
  - Requires more integration & synchronization
  - Communication bottleneck with micro-simulators
- Coupling different simulators is desirable...
- … to allow multifaceted simulations
4.4 Future Work

- SUMO coupling with IC-DEEP
- DRIS (High-Fidelity) Simulator integration
- Include behaviour elicitation through peer-designed agents
- Use the latter to implement a MAS and model cultural/geographical idiosyncrasies
Towards a Mobile-based ADAS Simulation Framework

MSc Student
João S. V. Gonçalves

Universidade do Porto
Faculdade de Engenharia
FEUP

LIACC
artificial intelligence and computer science laboratory
ADAS Interaction

Diagram:

- **Android Device**
  - START SERVICE
  - CONNECT()
  - NOTIFY()
  - NOTIFY SYSTEM()

- **IC-DEEP**
  - CONFIGURE
  - START GAME
  - UPDATE COORDS
  - UPDATE STATUS

**GAME LOOP**
1.2 Serious Games

“A mental contest, **played with a computer in accordance with specific rules, that uses entertainment to further government or corporate training, education, health, public policy, and strategic communication objectives.”

- Michael Zyda
1.2 Serious Games - why?

- Conducting Human Factor Analysis
- Simulate Artificial Societies with behaviour elicitation through peer-designed agents

Fig. 10: IC-DEEP @ LIACC
Interesting questions...

- How does SUMO connect to multiple mobile devices?
- How much data preparation is needed for SUMO?
- Why Distributed ADAS pose a bottleneck in the simulation?