Lucas Berghäuser
Paul Idstein
Fabian Knöller
19. Dezember 2015

ADAM - Advanced Driving Assistance on a Mobile

Project Status Report - 12/2015

K R D I T

The project „Advanced Driving Assistance on a Mobile“ consists of 5 consecutive project phases:

1. Kickoff
2. Requirements
3. Design
4. Implementation
5. Test

The following report gives an overview about the status of every project phase.

1. Kickoff

The project phase Kickoff has already been completed. During this phase the followg tasks have been conducted:

- acquire group members
- understand project idea

- finalise project vision
- form a team and assign roles to each member

The idea of ADAM is to enable any car driver to use a commodity mobile device with front facing camera, to use said device as a dash cam and give warnings on dangerous situations, that could occur while driving. The assistant should therefore act as a third eye that is always on the road.

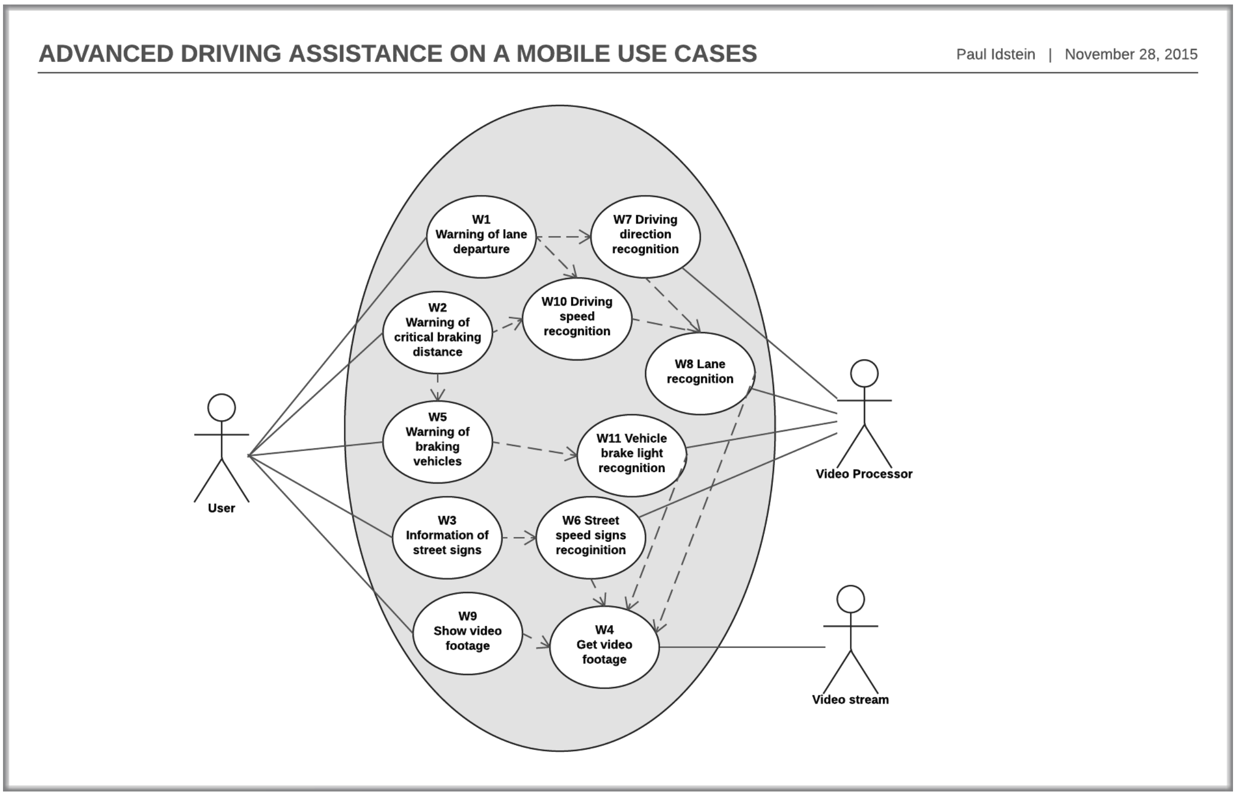
The team to realize the project consists of three members:

- Lukas Berghäuser: Student in Computer Science, has worked on several programming tasks, no knowledge in Android and vision engineering so far, experienced team worker
- Paul Idstein: IT consultant for mobile middleware, author, developer and graduated CS student, has experience in Android ecosystem, has participated in large IT development projects as software engineer and IT architect
- Fabian Knöller: Requirements Engineer and Student in Computer Science, has low experience in Android programming, has participated in large IT projects as requirements engineer

2. Requirements

R

The project phase Requirements has already been completed.



ADVANCED DRIVING ASSISTANCE ON A MOBILE USE CASES

Paul Idstein | November 28, 2015

W1 Warning of lane departure
W7 Driving direction recognition
W2 Warning of critical braking distance
W10 Driving speed recognition
W8 Lane recognition
W5 Warning of braking vehicles
W11 Vehicle brake light recognition
W3 Information of street signs
W6 Street speed signs recognition
W9 Show video footage
W4 Get video footage
User
Video Processor
Video stream

The functional and non-functional requirements of the project have been determined. To reach this goal, the team had to go beyond the starting requirements and get to a more specific, detailed level. The results of this phase is a System Requirements Specification (SRS) document, which has already been submitted and can be found in the Git repository.

https://github.com/idstein/ADAM/blob/master/SRS.md

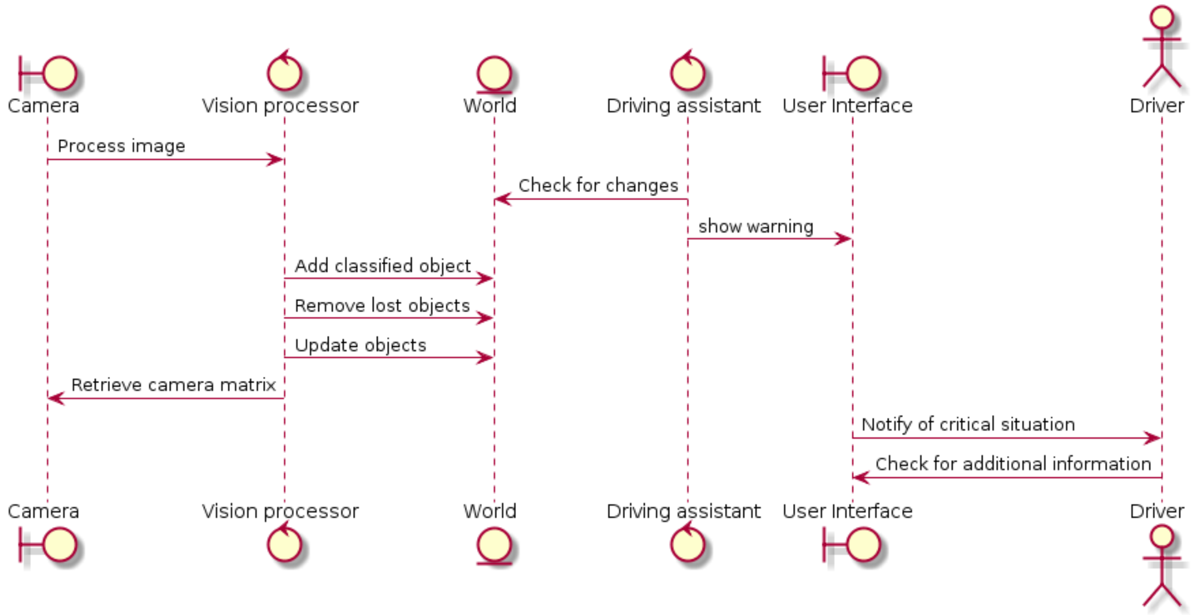
3. Design

The project phase Design is currently in progress.

To operate across a broad set of systems, the assistant will utilise common libraries such as OpenCV. The camera and user interface will be addressed in the operating system SDKs. That way we can separate image processing and computer vision from the user interaction. Thereby it is possible to develop and evaluate the OpenCV related recognition and object classification independent of mobile device specific components.

As first step the requirements have been transformed into Use Cases which determine the main goals and steps of the application. Those Use Cases are part of the SRS document specified above.

The next steps will include an architecture model and setting up a process for implementation.

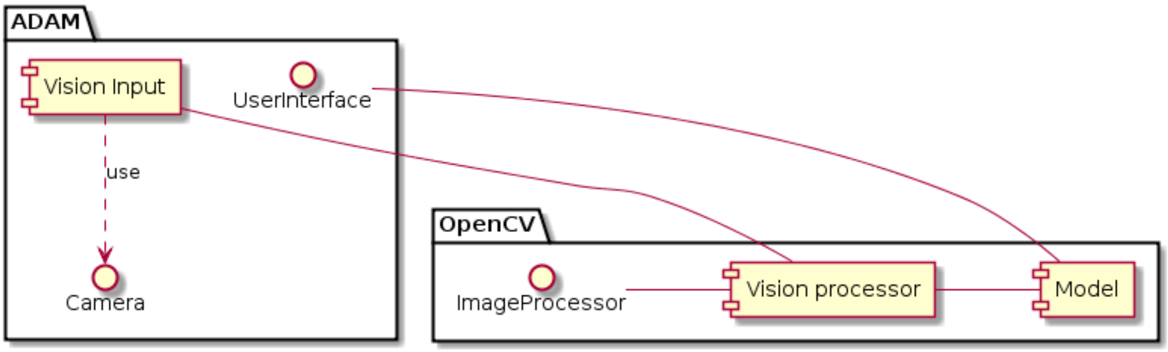


PlantUML: Interaction between components

http://plantuml.com/plantuml/png/XP71JiCm38RlUGehTrw00n8Q5ou88GtENkAsXZQUR6-ZRq-w3ar0KO-s_i-Vu-paK3Bg21iSI34aAQxIGtnbOocmLqba9Xe35MSV8Rw9zZbUam_AHouje1cRO_4OTjzJuzP8uR4uQOE9Oi3aeZ0Xf65smitTWWvkuNakW0TiAInupkIyJerlEaeVqDJngSFIalsB_5ggijR92KwefQR3mZ8_HFSvG-gdcGrJ1jczK_9LweK6EHBqOdv1R9NPxZCwhSGlLvmChapLaUREWBN-37-UMvdv0ohp9CxD2D90KdPEs8En7z2h9rnIooUxdXjpvYbVMIwrCPpfBm00

```
boundary Camera
control "Vision processor"
entity "World"
control "Driving assistant"
boundary "User Interface"
actor Driver
Camera -> "Vision processor" : Process image
"Driving assistant" -> "World" : Check for changes
```

```
"Driving assistant" -> "User Interface" : show warning
"Vision processor" -> "World" : Add classified object
"Vision processor" -> "World" : Remove lost objects
"Vision processor" -> "World" : Update objects
"Vision processor" -> Camera : Retrieve camera matrix
"User Interface" -> Driver : Notify of critical situation
Driver -> "User Interface" : Check for additional information
```



PlantUML: Component overview

http://plantuml.com/plantuml/png/AqXCpavCJrLGyYz8pNCEKrAevb9GyCm52WKKvIUd5XVd5odeAaI7PHPdvkSf5C34Oe6gC0L1AdtpKr9pOhbgkRWAO4OxkZZwGWo6QV7CAoWjYILaC5ILdfwTWdDYRcfHeeALGcbnAb15Q75gaMTUIMfHMc9oAhAj0000

```
package "OpenCV" {
  ImageProcessor - [Vision processor]
  [Vision processor] - [Model]
}

package "ADAM" {
  [Vision Input] - [Vision processor]
  [Vision Input] ..> Camera : use
  UserInterface - [Model]
}
```

4. Implementation

The project phase Design is not started yet.

Android related:
Camera access
Dashcam (camera passthrough)
Settings and user preferences
Warning notifications

OpenCV related:
Camera calibration (intrinsic, extrinsic parameters)

Traffic sign recognition and classification (right-of-way, speed, warning, regulatory, supplemental)
Lane detection
Driving vehicle relative speed to lane and speed of other traffic participants
Distance detection

Nevertheless a Git repository has been created. It can be reached at https://github.com/idstein/ADAM, but only on invitation because it is a private repository. (Contact Paul Idstein for further information)

T

5. Test

The project phase Test is not started yet.
The test cases should be specified based on the already available Use Cases.