

TASK 3: “AUTOMOTIVE LIGHTING WITH EYE GUIDANCE”

V-Cycle

Requirements:

- SW00: The system shall detect the pupil of the conductor's eye.
- SW01: The system shall detect the direction of the movement of the pupil of the conductor's eye.
- SW02: The video stream shall calibrate the camera to make the pupil visible.
- SW03: The system shall follow the movement of the pupil as long as it is moving.
- SW04: The system shall vary the lighting intensity according to the environment's lighting and the direction of the pupils.
- SW05: The system shall interface with the car's system via CAN Bus.
- SW06: The system shall respect the AUTOSAR architecture.
- SW07: The system shall be implemented inside the Raspberry Pi.
- SW08: The system shall send the acquired data to the IHM interface.

System requirements:

- SW00: The system shall be able to detect a demarcated pupil in dark and bright environments.
- SW01: The system shall detect four directions according to the movement of the pupil : right, left, up and down.
- SW02: The camera shall be implemented in a rotative base in order to follow the driver's gaze.
- SW03: The system shall capture the driver's gaze as long as headlights are one.
- SW04: The system shall be able to detect exterior lighting intensity (sun, moon, street lights) and adjust the headlights intensity according to it.
- SW05: The communication shall be assured between all 4 nodes of the system (Lights control, Somnolence, Parking and the main controller) via CAN Bus.
- SW06: The system shall follow the AUTOSAR implementation (Basic Software, Run Time Environment, Application)
- SW07: The system shall be optimized for Raspberry Pi.
- SW08: All data meant to be viewed by the user shall be displayed on the IHM interface.

Unit tests:

- SW00:
 - Inputs : Img_Face_Acquired (Boolean)
 - Output : Pupil_Detected (Boolean), Face_In_Range(Boolean), Face_Left(Boolean), Face_Right(Boolean), Face_Left(Boolean), Face_Up(Boolean)
- SW01:
 - Inputs : Pupil_Detected(Boolean)
 - Outputs : Eye_Is_right(boolean) Eye_Is_left(boolean) Eye_Is_up(boolean) Eye_Is_down(boolean)
- SW02:

- Inputs : Face_Left(Boolean), Face_Right(Boolean), Face_Left(Boolean), Face_Up(Boolean)
- Outputs : Camera_Left(Boolean), Camera_Right(Boolean), Camera_Left(Boolean), Camera_Up(Boolean)
- SW03:
 - Inputs : Headlights_State(Boolean)
 - Outputs : System_State(Boolean)
- SW04:
 - Inputs : Light_intensity_ext(Float)
 - Outputs : Light_intensity_headlights(Float)
- SW05:
 - Inputs : Data sent via CAN Bus
 - Outputs: CRC check results
- SW06:
 - Input: User's pupil of the eye from computer's camera.
 - Output: Eye tracking data following the user's eyes' direction.
- SW07:
 - Inputs : Latency between eye movement and eye movement detection Latency between eye movement detection and actuators response
 - Outputs : High_latency_detected(Boolean)
- SW08:
 - Input: User's pupil of the eye from Pi camera.
 - Output: Eye tracking data following the user's eyes' direction.

Integration tests:

- Function 1:
 - Input: Data from Pi camera.
 - Output: Fluent real time acquisition of clear frames.
- Function 2:
 - Input: Real time video stream.
 - Output: Pupil movement tracking.
- Function 3:
 - Input: Pupil position.
 - Output: Actuators' commands and control.

System tests:

Inputs : Real time video stream, captured in multiple lighting conditions and different eye movements..

Outputs : Headlights control according to pupil movements.

High Level Design:

Functions:

- Function 1: video_capture(). Reads video from Pi camera and returns it.
- Function 2: video_treatment(). Reads video from video_capture() and returns eye tracking data.

- Function 3: headlight_direction(). Reads eye tracking data from video-treatment() and activates actuators accordingly.

Low Level Design:

Function 1:

```
While true
    Camera open
Return Camera
```

Function 2:

```
While true
    Read input
    HEADLIGHT_STRAIGHT=TRUE
    If FACE_DETECTED
        While EYES_OPEN
            If EYES_LEFT
                HEADLIGHT_LEFT = TRUE
            If EYES_RIGHT
                HEADLIGHT_RIGHT = TRUE
            If EYES_UP
                HEADLIGHT_UP = TRUE
            If EYES_DOWN
                HEADLIGHT_DOWN = TRUE
```

Function 3:

```
Read input data
If HEADLIGHT_LEFT turn actuator left
If HEADLIGHT_RIGHT turn actuator RIGHT
If HEADLIGHT_UP turn actuator UP
If HEADLIGHT_DOWN turn actuator DOWN
```