SECREDAS
Product Security for Cross Domain Reliable Dependable Automated Systems

DELIVERABLE REPORT
D4.1: “Conception of sensor, components, and sensor data processing”

Document Type: Deliverable
Document Number: D4.1
Primary Author(s): Nadja Marko
Document Date: 04.09.2019
Document Version / Status: v3/Final
Distribution Level: Public
Reference DoA: 30 April 2018

Project Coordinator: Patrick Pype, NXP Semiconductors, patrick.pype@nxp.com
Project Website: www.secredas.eu (in progress)
JU Grant Agreement Number: 783119

SECREDAS has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement nr.783119. This Joint Undertaking receives support from the European Union’s Horizon 2020 research and innovation programme and Austria, Belgium, Czech Republic, Germany, Finland, Hungary, Italy, The Netherlands, Poland, Romania, Sweden and Tunis.
CONTRIBUTORS

<table>
<thead>
<tr>
<th>Name</th>
<th>Organization</th>
<th>Name</th>
<th>Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nadja Marko</td>
<td>VIF</td>
<td>Ulrico Celentano</td>
<td>UOULU</td>
</tr>
<tr>
<td>Victor Exposito</td>
<td>VIF</td>
<td>Andras Varadi</td>
<td>CMS (Commsignia)</td>
</tr>
<tr>
<td>Leos Mikulka</td>
<td>AVL-SF</td>
<td>Rodolfo Simões Ferreira</td>
<td>PDMFC</td>
</tr>
<tr>
<td>Ying Liu</td>
<td>AVL-SF</td>
<td>Arturo Medela</td>
<td>TST</td>
</tr>
<tr>
<td>Filipe Carvalho</td>
<td>AVL-SF</td>
<td>Mark Pfeiffer</td>
<td>MERANTIX</td>
</tr>
<tr>
<td>Jana Viehbeck</td>
<td>senetics</td>
<td>Eric Nassor</td>
<td>CRF</td>
</tr>
<tr>
<td>Michael Wiehl</td>
<td>senetics</td>
<td>Stelios Karagiannis</td>
<td>BeyondVision</td>
</tr>
</tbody>
</table>

FORMAL REVIEWERS

<table>
<thead>
<tr>
<th>Name</th>
<th>Organization</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roy Pennings</td>
<td>NXP</td>
<td>14-08-2019</td>
</tr>
<tr>
<td>Project Steering Board</td>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

DOCUMENT HISTORY

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Author / Organization</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>V1</td>
<td>20.05.2019</td>
<td>All authors</td>
<td>Integration of inputs from all partners</td>
</tr>
<tr>
<td>V2</td>
<td>22.07.2019</td>
<td>All authors</td>
<td>Rework of inputs based on comments from internal review, add executive summary, introduction and conclusion</td>
</tr>
<tr>
<td>V3</td>
<td>21.08.2019</td>
<td>All authors</td>
<td>Rework of inputs based on review comments from Roy Pennings (NX).</td>
</tr>
</tbody>
</table>
Executive summary

According to SAE’s definition, the highest level of automated driving is level 5 fully autonomous driving. In the past several years, level 5 autonomous driving was made as a target for many technical players, such as OEMs, Tier 1s, and so on. However, in recent months, quite a good amount of companies steers to making good functions for level 2 and level 3 ADAS functionalities, such as automated emergency brake, lane keep/change assist, highway pilot. Why the technical trend changes its direction? There are many reasons behind, such as the demand of practical use cases and regulations. One reason from the technical perspective is that it is extremely challenging to obtain a robust environment understanding for all traffic scenarios.

As explained above, environment sensing plays a key role in realizing every level of autonomous driving. Without the ability of environment sensing is like humans without sensing competence. As we all know that a single sensor will not enable safe and robust environment sensing, therefore, it is common to use multi-type and multiple sensors to cover the environment of the automated vehicle. Consequently, the corresponding sensor data processing and fusion become a challenging topic.

Analogy to our human sensing system, popular sensors in autonomous driving area include RGB and infrared cameras, radar, LIDAR, ultrasonic sensor, etc. Each sensor type has its pros and cons in covering various scenarios. For example, a vision camera and computer vision or deep learning approaches can make a good scene understanding, and a radar or LIDAR sensor is good at delivering distance information without the need of much further data processing. Hence, it is vital to combine the sensing results from each sensor. Moreover, functional safety is of great importance to achieve automated driving, and functional safety requires the fusion of different types of sensors and hardware.

In this deliverable, various sensing components used in the SECREDAS project for advancing software algorithms and improve autonomous driving functions are described. The sensing components comprise radar, LIDAR, camera, and some other sensors that have been selected and developed by partners of the project. Sensor fusion concepts and also other sensor data analysis methods are elaborated and will be used and implemented in the SECREDAS project to provide secure and reliable functions for automated vehicles. Further, used computing platforms are shortly described.
# Table of Contents

Executive summary .................................................................................................................. 3  
Table of Contents ....................................................................................................................... 4  
Acronyms/abbreviations ........................................................................................................ 5  
1 Introduction .......................................................................................................................... 6  
  1.1 Role of deliverable ........................................................................................................... 6  
  1.2 Related SECREDAS documents ................................................................................... 6  
  1.3 Document structure ....................................................................................................... 7  
2 Sensing components ............................................................................................................. 8  
  2.1 Radar ................................................................................................................................ 8  
  2.2 LIDAR ............................................................................................................................. 9  
  2.3 Camera .......................................................................................................................... 12  
  2.4 Other ............................................................................................................................ 15  
3 Computing platforms ........................................................................................................... 22  
  3.1 GCP & portable PC (MERANTIX) .................................................................................. 22  
  3.2 ARTPEC 6 & portable PC (CRF) .................................................................................... 22  
  3.3 Arduino IDE (senetics) .................................................................................................. 23  
  3.4 NVIDIA Jetson Platforms .............................................................................................. 23  
4 Data processing ..................................................................................................................... 25  
  4.1 Sensor fusion .................................................................................................................. 25  
  4.2 Sensor data analysis ........................................................................................................ 29  
  4.3 Collective Perception ..................................................................................................... 33  
5 Conclusions ............................................................................................................................ 36  
Figures ..................................................................................................................................... 37  
Tables ....................................................................................................................................... 37  
References ................................................................................................................................. 37
<table>
<thead>
<tr>
<th>Acronyms/abbreviations</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADAS</td>
<td>Advanced Driver Assistance System</td>
</tr>
<tr>
<td>AES</td>
<td>Advanced Encryption Standard</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>APP</td>
<td>Application</td>
</tr>
<tr>
<td>ASIL</td>
<td>Automotive Safety Integrity Level</td>
</tr>
<tr>
<td>CAN</td>
<td>Controller Area Network</td>
</tr>
<tr>
<td>C-ITS</td>
<td>Cooperative Intelligent Transport Systems</td>
</tr>
<tr>
<td>COTS</td>
<td>Commercial off the shelf</td>
</tr>
<tr>
<td>CPS</td>
<td>CyberPhysical Systems</td>
</tr>
<tr>
<td>CSE</td>
<td>Cryptographic Services Engine</td>
</tr>
<tr>
<td>DAC</td>
<td>Discretionary Access Control</td>
</tr>
<tr>
<td>DRM</td>
<td>Digital Rights Management</td>
</tr>
<tr>
<td>ECU</td>
<td>Embedded Control Unit</td>
</tr>
<tr>
<td>EKMS</td>
<td>Electronic Key Management System</td>
</tr>
<tr>
<td>eSE</td>
<td>embedded Secure Element</td>
</tr>
<tr>
<td>eNVM</td>
<td>embedded Non-volatile Memory</td>
</tr>
<tr>
<td>ETSI</td>
<td>European Telecommunications Standards Institute</td>
</tr>
<tr>
<td>EV</td>
<td>Electrical Vehicle</td>
</tr>
<tr>
<td>FMEA</td>
<td>Failure Mode Effective Analysis</td>
</tr>
<tr>
<td>FOTA</td>
<td>Firmware Over The Air</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field Programmable Gate Array</td>
</tr>
<tr>
<td>Gateway</td>
<td>A VCU that connects multiple networks within a car</td>
</tr>
<tr>
<td>GDPR</td>
<td>General Data Protection Regulation</td>
</tr>
<tr>
<td>GNSS</td>
<td>Global Navigation Satellite System</td>
</tr>
<tr>
<td>HLC</td>
<td>High-Level Controller</td>
</tr>
<tr>
<td>HMI</td>
<td>Human Machine Interface</td>
</tr>
<tr>
<td>HSM</td>
<td>Hardware Security Module</td>
</tr>
<tr>
<td>HW</td>
<td>Hardware</td>
</tr>
<tr>
<td>IC</td>
<td>Integrated Circuit</td>
</tr>
<tr>
<td>IDS</td>
<td>Intrusion Detection Systems</td>
</tr>
<tr>
<td>IoT</td>
<td>Internet-of-Things</td>
</tr>
<tr>
<td>IP</td>
<td>Intellectual Property</td>
</tr>
<tr>
<td>IPv6</td>
<td>Internet Protocol Version 6</td>
</tr>
<tr>
<td>ISE</td>
<td>Integrated SE</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standardization Organization</td>
</tr>
<tr>
<td>IT</td>
<td>Information Technology</td>
</tr>
<tr>
<td>ITS</td>
<td>Intelligent Transport Systems</td>
</tr>
<tr>
<td>IVS</td>
<td>In-Vehicle System</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>LIN</td>
<td>Local Interconnect Network</td>
</tr>
<tr>
<td>LLC</td>
<td>Low-Level Controller</td>
</tr>
<tr>
<td>LTE</td>
<td>Long Term Evolution (4th generation Mobile Internet)</td>
</tr>
<tr>
<td>MAC</td>
<td>Media Access Control</td>
</tr>
<tr>
<td>MCU</td>
<td>Micro Controller Unit</td>
</tr>
<tr>
<td>M2M</td>
<td>Machine-to-Machine</td>
</tr>
<tr>
<td>OBU</td>
<td>On-Board Unit</td>
</tr>
<tr>
<td>OEM</td>
<td>Original Equipment Manufacturer</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>OTA</td>
<td>Over-The-Air</td>
</tr>
<tr>
<td>PDU</td>
<td>Protocol Data Unit</td>
</tr>
<tr>
<td>PHY</td>
<td>PHYSical Layer</td>
</tr>
<tr>
<td>PIA</td>
<td>Privacy Impact Assessment</td>
</tr>
<tr>
<td>PKI</td>
<td>Public Key Infrastructure</td>
</tr>
<tr>
<td>POC</td>
<td>Proof Of Concept</td>
</tr>
<tr>
<td>RBAC</td>
<td>Role-Based Access Control</td>
</tr>
<tr>
<td>ROS</td>
<td>Robot Operating System</td>
</tr>
<tr>
<td>RSA</td>
<td>Asymmetric Encryption Algorithm developed by Rivest, Adi Shamir &amp; Len Adleman</td>
</tr>
<tr>
<td>RSS</td>
<td>Received Signal Strength Indication</td>
</tr>
<tr>
<td>RSU</td>
<td>Road Side Unit</td>
</tr>
<tr>
<td>RTLS</td>
<td>Real Time Location System</td>
</tr>
<tr>
<td>RTOS</td>
<td>Real Time Operating System</td>
</tr>
<tr>
<td>SE</td>
<td>Secure Element</td>
</tr>
<tr>
<td>SSL</td>
<td>Secure Sockets Layer</td>
</tr>
<tr>
<td>SW</td>
<td>Software</td>
</tr>
<tr>
<td>TCB</td>
<td>Trusted Computing Base</td>
</tr>
<tr>
<td>TEE</td>
<td>Trusted Execution Environment</td>
</tr>
<tr>
<td>TLS</td>
<td>Transport Layer Security</td>
</tr>
<tr>
<td>TRL</td>
<td>Technology Readiness Level</td>
</tr>
<tr>
<td>UWB</td>
<td>UltraWide Band</td>
</tr>
<tr>
<td>UC</td>
<td>Use Case</td>
</tr>
<tr>
<td>UWB</td>
<td>Ultra Wideband</td>
</tr>
<tr>
<td>VCU</td>
<td>Vehicle Control Unit</td>
</tr>
<tr>
<td>VDS</td>
<td>Vehicle and Driver Status</td>
</tr>
<tr>
<td>VPN</td>
<td>Virtual Private Network</td>
</tr>
<tr>
<td>V2X</td>
<td>Vehicle-to-X, where X stands for either Vehicle or Infrastructure</td>
</tr>
<tr>
<td>WAVE</td>
<td>Wireless Access in Vehicular Environments</td>
</tr>
<tr>
<td>WP</td>
<td>Work Package</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Role of deliverable

This deliverable describes sensing components, computing platforms and sensor data processing algorithms that have been selected by the Work Package 4 (WP4) participants and are improved in Task 4.1. These components and software algorithms are applied and integrated in the other tasks of WP4.

1.2 Related SECREDAS documents

Related documents are all WP4 produced deliverables:

- D4.2 Test concept for sensor, components, and sensor data processing.
- D4.3 Final specification for a common demonstrator and proof-of-concept (POC) implementation of an interference free LIDAR measurement principle.
- D4.9 Modular Security Testing System.
- D4.4 Final release of a proof-of-concept (POC) implementation (components and system) of an interference free LIDAR measurement principle.
- D4.5 Development of data processing components.
- D4.6 Robustified image segmentation.
- D4.7 Sensor System Integration and Validation.
- D4.8 Integrated Sensor System Prototype.

The WP3 deliverable D3.4 Initial design patterns for common technology elements describing an initial set of design patterns was not yet available at the time this deliverable was written, as this deliverable is due to end of October. However, a first list of design patterns is already available from an intermediate version of D3.4 (see Table Error! No text of specified style in document.-1) and will be referenced in the individual sections. Most components/software algorithms are related to data fusion and sensor data analytics design patterns.

Table Error! No text of specified style in document.-1: Design Patterns

<table>
<thead>
<tr>
<th>Design Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Data Governance &amp; Multi-party Access control (TUE)</td>
</tr>
<tr>
<td>(2) Data Usage Control (Fh-IESE)</td>
</tr>
<tr>
<td>(3) Conditional Safety Certificates (Fh-IESE)</td>
</tr>
<tr>
<td>(4) Data Fusion Approach (AVL-SF)</td>
</tr>
<tr>
<td>(5) Authentication and Authorization module (IDEMIA (OTM))</td>
</tr>
<tr>
<td>(6) End-2-End Security (IDEMIA (OTM))</td>
</tr>
<tr>
<td>(7) Drowsiness Detection System (IDEMIA (OTM))</td>
</tr>
</tbody>
</table>
1.3 Document structure

In chapter 0 the sensing components are described, structured to Radar, LIDAR, Camera and Other sensing components. Chapter 0 depicts the used computing platforms. Finally, chapter 0 includes the software algorithms for processing the sensor data.
2 Sensing components

Environment sensing plays an essential role in achieving every level of autonomous driving. A single sensor will not enable safe and robust environment sensing; therefore, it is common to use multi-type and multiple sensors to cover the external and internal environment of the automated vehicle. Radar sensors, Lidar sensors, cameras, etc. are combined and used in multiple devices/places in the vehicle to be able to sense the internal and external environment. This chapter describes the selected sensing components that are applied in the SECREDAS Use Cases (UCs) and that provide data for the data processing software.

2.1 Radar

2.1.1 Object detection (BeyondVision)

2.1.1.1 Overview
Radar can be used when LIDAR sensors cannot provide reliable response due to weather conditions (e.g. rain, mist) or other environmental conditions, responsible for decreasing the performance and quality of the LIDAR sensors. Moreover, radar is used for cases that require longer operating distances. Radar has many disadvantages compared to LIDAR, however, radar could be used as a backup functionality when LIDAR fails to meet/enhance the effectiveness of distance measuring and obstacle recognition. Currently, BeyondVision will not gather any RADAR data, but will responsible for processing data from other partners or open repositories in order to conduct further analysis and to achieve the best results in terms of accuracy. Finally, cases which might include compromised or manipulated data through malicious actions will be considered as well, for analysis and detection of possible malicious incidents regarding the sensor data.

2.1.1.2 Integration
Radar is used in the following UCs: UC 1, UC2, UC3 and UC6. A partnership will be established with PDMFC in matters of data processing. Related Designed Patterns are:

- (4) Data Fusion Approach;
- (14) Sensor data analytics.

Part of the integration plan will include a definition of the benefits of radar for image recognition.
2.2 LIDAR

2.2.1 LIDARs and deep learning object classification (BeyondVision)

2.2.1.1 Overview
Beyond-Vision develops automated detection and deep learning classification for capturing real environments through automated LIDAR data processing. Data is collected using various sensors, specifically on drones. Although currently mostly used for internal support of the drone’s functionality, the developed technologies can be directly applied to the automotive industry and its large dataset of applications. Through clustering algorithms, the main purpose is to combine the collected data in order to enhance responsiveness and accuracy recognition. Some of the applications have already been developed (such as navigation, home automation control, personalized weather stations and innovative sport and fitness applications). Environmental sensors used for recognition related to environmental conditions, will enhance or calibrate the LIDAR sensors in case of any detected harsh conditions.

2.2.1.2 Integration
The LIDAR sensors will be used in:

- UC1 for the integrity of speed measurements, measuring vehicle-to-vehicle distance, support image and 3d representation, real-time obstacle detection;
- UC2 for simulating environmental cases such as weather conditions which might apply in this UC as well as detecting driver safety;
- UC6 for information gathering to enhance incident investigations.

There is a close cooperation with PDMFC in deep learning algorithms. Related design patterns are:

- (4) Data Fusion Approach;
- (14) Sensor data analytics.

2.2.2 LIDARs for collision avoidance (VIF)

2.2.2.1 Overview - LIDARs for collision avoidance function
The SPIDER is a mobile robot capable of omni-directional driving. Thus, a 360° environment perception with the same accuracy and range is required. The SPIDER is intended to be operated in a closed environment like a proving ground, where the access of humans is prohibited. However, in order to ensure maximum safety, the SPIDER should detect humans (or objects) approaching from any arbitrary angle and reduce speed or initiate an emergency brake if they come too close. In addition, the SPIDER is always monitored during operation by a human who can initiate an
emergency shutdown via a radio emergency button. Four LIDARS are installed in the SPIDER to meet the requirements. Each sensor is located at each corner of the SPIDER, as shown in Figure Error! No text of specified style in document.-1.

![Diagram of LIDAR sensors on SPIDER](https://www.ouster.io/product-os1)

Figure Error! No text of specified style in document.-1: Location of LIDAR sensors on SPIDER

The risk of collision is reflected by the safety zones shown in Figure Error! No text of specified style in document.-2. The SPIDER distinguishes between the following safety zones:

- **Danger zone**: The danger zone surrounds the SPIDER itself and the likelihood of a collision is very high. An emergency brake procedure will be initiated immediately if an external object is detected in this zone. The detection of such an object is signalled by the SPIDER via a light signal and an acoustic warning.

- **Movement zone**: The movement zone reflects the area with a high likelihood of collision if the current speed and movement direction of the SPIDER are maintained. Due to the omni-directional movement capabilities of the SPIDER, the movement zone rotates around the SPIDER, corresponding to its performed rotation. In case an environment object is detected within the movement zone, an emergency stop will be initiated.

- **Safety zone**: The detection of an object within the safety zone will be signalled by the SPIDER via a light signal and an acoustic warning. Furthermore, the maximum speed is limited to a pre-defined value as a precaution.

- **Detection zone**: The detection zone defined by the sensors describes the area in which an environment object can be detected. In case of detection overlaps in the area observed by the different sensors, the detection zone is the area is covered by all sensors.

- **Unobservable zone**: The environment objects within the unobservable zone are out of range of the sensors and cannot be detected.

---

1 [https://www.ouster.io/product-os1](https://www.ouster.io/product-os1) Last accessed 03/06/2019
In Figure Error! No text of specified style in document.-3 an architecture concept is shown, where the squares in blue represent hardware devices and the squares in purple are software functions executed on the High-Level Controller (HLC), running the Robot Operating System (ROS) on top of Linux. The HLC produces steering commands for the low-level controller (LLC), which is responsible for communicating with the control units actuating the steering and driving motors, summarized as drivetrain.

Figure Error! No text of specified style in document.-3: SPIDER function architecture concept.
2.2.2.2 Integration
The LIDAR sensors will be used for the collision avoidance function which will be the System Under Test (SUT) in task 4.4. For the testing ViF will collaborate with AVL and MERANTIX. To test the functionality within the security testing framework, ViF will cooperate with AVL, AIT, IoTr, TNO and SecInto. Related design patterns are:

- (4) Data Fusion Approach;
- (14) Sensor data analytics.

2.2.3 Sensor error detection (PDMFC)
2.2.3.1 Overview
PDMFC will work with LIDARs from third parties and process the acquired data. If/where possible, PDMFC will perform autonomous tests on the sensors so that possible occurrences of errors in the sensors from environmental changes or from other vehicles are mitigated and reported from the partners who collected and shared the data.

2.2.3.2 Integration
The LIDAR sensors will be used in UC1, where a road intersection requires cars to be able to perceive the environment around them. It can later be employed in the WP9 DEMO I. Related design patterns are:

- (4) Data Fusion Approach;
- (14) Sensor data analytics.

2.3 Camera
2.3.1 Increase robustness of perception (MERANTIX)
2.3.1.1 Overview
MERANTIX focuses on the development of intelligent algorithms for image processing with the goal of increasing the robustness of image-based perception solutions. As the focus lies on researching new approaches and the software development during deployment, well known open source datasets for autonomous driving will be used as well as new camera recordings. In order to record new data, MEANTIX will rely on third party camera and computing hardware. A major part of the required software will be developed within the SECREDAS project and is therefore not yet on the market.

2.3.1.2 Integration
Most of the software developed by MERANTIX affects model training and evaluation. In order to train and test the camera perception models, the actual software does not need to be integrated into the vehicle, but into cloud processing pipelines. Beyond that, MERANTIX will integrate the resulting models (which were trained offline) into WP9 DEMO I to demonstrate vehicle behavior in UC sub-scenario 1.5.
2.3.2 Road-side surveillance cameras (CRF)

2.3.2.1 Overview
CRF will provide roadside surveillance cameras to SECREDAS. These cameras (current brand model: Axis P1367-E; see Figure 2-4) will be used in WP4 to provide sensor data for sensor fusion algorithms (being developed by CRF) toward the WP9 DEMO I demonstrator. CRF is developing and commercializing multiple types of network cameras, several of which are particularly well adapted to the road surveillance market.

Figure Error! No text of specified style in document.-4: Canon Axis P1367-E camera

The Axis P1367-E camera has a high video quality with a high level of detail: 5 Mega Pixel resolution (3072x1728) at 30 frame per second with a CMOS sensor of 1/2.9” with progressive scan. The default lens provides a focal of 2.8-8.5 mm at F 1.2 and a field of view of 72°-36° horizontally and 40°-20° vertically, adapted to global traffic analysis. If a more specific task is required, the lens (CS mount) can be changed to achieve a better zoom. It is also possible to change the image orientation to the corridor format (vertical images) for a better adaptation to the scene.

In order to be positioned at the roadside, the camera is protected by an enclosure against wind, rain, snow (IK10-, IP66- and NEMA 4X-rating). The enclosure also allows efficient camera operation in the range -40 to +50°C. It also detects and sends out a signal in case of tampering attempts.

Challenging light conditions video surveillance area are neutralized through Axis Lightfinder technology (which guarantees true colour rendering even under poor light conditions) and WDR – Forensic Capture technologies (reduces the noise level and increases the image gain to enable exceptional detail, removes distortions caused by different light levels in various image zones) with fast automatic switching between these two modes.

The video is encoded in H.264; moreover, Axis zip-stream technology is used to optimize video compression. Multiple video streams may be sent simultaneously on the network, encoding either specific regions of interest (up to 8 cropped view areas) or sending video to multiple recipients (for example to a traffic manager and in parallel to a video analytic server).

The camera provides IPv4/v6 connectivity (RJ45 10BASE-T/100 BASE TX PoE) with several types of secure network protocols HTTPS, SSL/TLS, SSH to control the camera or send video streams. The camera may include software modules for automatic incident detection. The Axis traffic cameras can automatically send an alert in case of congestion, a
stopped vehicle or even when someone driving in the wrong direction. The camera can also read license plates. The camera generates metadata which can be send in parallel to the video streams. The camera also provides an API, which allows 3rd party software integration. Software modules can thus be directly integrated inside the camera, either from internal developments at CRF or from the SECREDAS project. The camera provides 1GB RAM and an ARTPEC v6 processor.

2.3.2.2 Integration
The roadside surveillance cameras will be used in WP4 to provide sensor data for sensor fusion algorithms developed in the scope of the project and in the WP9 DEMO I. DEMO I will target UC1 and sub-scenario 1.1, 1.2, 1.3 and 1.4. This integration work is done in collaboration with TNO and Commsignia. The camera is a commercially available component which will be used with no modifications. Related design patterns are:

- (4) Data Fusion Approach;
- (14) Sensor data analytics.

2.3.3 Methods related to Data fusion and AI for Entity Identification (BeyondVision)

2.3.3.1 Overview
BeyondVision is working in software for image recognition and will work with CRF. Deep learning is used for image recognition in order to detect objects or obstacles. Algorithms “trained” on millions of pictures using a Predictive Model will make it possible to conduct face recognition, pattern gradient matching and scene identification. Most of the inputs are derived from the following cameras: Multispectral, 360 degrees and RGB Cameras. Collection, sampling and data processes are currently being applied. The datasets will be tested and train the proposed machine learning models.

2.3.3.2 Integration
The developed software will be used in UC1, UC2 and UC4. Authentication processes will be integrated to support authentication processes and ID management. The partnership with CRF will focus mostly on concepts related to data fusion and on using AI to identify objects, people and other entities. The provided inputs and data will be used to extend the current behavior on drones. The related Scenarios are the following: UC1, 2, 4 and 6.

Partnership: BeyondVision will work with PDMFC, who will provide hardware parts related to cameras. The partnership will be extended to CRF who will provide datasets from their camera along with Data fusion methods and Artificial Intelligence processes related to entity identification of objects, obstacles and persons. Related design patterns are:

- (4) Data Fusion Approach;
- (5) Authentication and Authorization module;
- (7) Drowsiness Detection System;
• (14) Sensor data analytics;

Integration plans include the following:

2.3.4 Drones equipped with cameras (PDMFC)

2.3.4.1 Overview
CRF and PDMFC will collaborate to explore the CRF cameras and to develop new software while also working closely with BeyondVision. PDMFC will also use its expertise with drones in preparation for WP9 DEMO I. DEMO I will emulate a crossroad; it is, however, extremely difficult to prepare the infrastructure required for such a real-world environment. PDMFC expects to be able to short-cut this difficult task by using CRF’s cameras in its drones. A drone flying in the same position with a camera can easily mimic a post with a camera, therefore a full surveillance system can be achieved using only drones and cameras. Drones can also be used to copy other subjects' functions in DEMO I, such as pedestrians. BeyondVision will be involved, mostly at the software level.

2.3.4.2 Integration
Drones will be used in DEMO I of WP9, which covers UC1, UC3 and UC6, and will enhance the testing of cameras and V2X communication. Several tests will be performed, including some latency tests. Related design patterns are:

• (4) Data Fusion Approach;
• (5) Authentication and Authorization module;
• (13) Secure communication to mobile robots (VIF).

2.4 Other

2.4.1 Integration of different sensors – Sensor box (VIF)

2.4.1.1 Overview
Figure Error! No text of specified style in document.-5 shows a sensor box on which different sensors are mounted. Its size is 0.5m x 0.5m x 0.5m and it is controlled by a NVIDIA TX1 development Kit2 which is placed inside the box. This board runs a L4T based on Ubuntu 16.04 as operating system where the Robot Operating System (ROS) carries out the message communication of the sensors and provides tools for the integration, such as visualization or simulation. Figure Error! No text of specified style in document.-6 provides an example of this visualization in which real-time data from LIDAR, stereo camera and long-range radar is shown. In this current development, the box includes:

• LIDAR;

---

- GPS;
- Stereo camera;
- Long range radar;
- Short range radar.

In further developments, more sensors will be implemented in order to carry out a depth sensor comparison and data analysis.
2.4.1.2 Radar
The sensor box includes two different radars, a long and a short range one. The communication of both radars is through a CAN in which a CAN-USB adapter is used to be able to get the data from the SensorBox development board.

**Long Range Radar**
A model Continental ARS408\(^3\) radar was installed in the front-left side of the sensor box. The radar works on a frequency of 77 GHz and is able to recognize an object at a maximum distance of 260 meters. The field of view in this case is only +/- 4°. In shorter distances, the field of view is wider, +/-90° until 70 meters.

**Short Range Radar**
A Continental SRR208\(^4\) radar is installed in the front-right side of the SensorBox and used to recognize objects at shorter distances. The maximum operational distance of this radar is 40 meters with an angle of +/-20°. The maximum operational angle is reached up to +/-75° for approximately 20 meters distance.

2.4.1.3 LIDAR
A LIDAR was also installed on the top of the SensorBox. The chosen model is an OusterOS1\(^5\) in the 16 lines version. It means that this LIDAR uses 16 horizontal lines to map the area with a maximum range of 100 meters. Three resolutions are available: 512, 1024 and 2048 pixels and two frequency rates, 10Hz and 20Hz.

2.4.1.4 Stereo Camera
The ZED Stereo Camera\(^6\) is installed on the front of the SensorBox and used for two purposes: the first purpose is to record a video which will be used as a reference when the data is analyzed and visualized. The second purpose is to calculate the depth of the objects that are in front of the camera until approximately 7 meters distance. This camera supports different resolutions (up to 2K) and frames per second (up to 60 fps).

2.4.1.5 GNSS Module
A GNSS module is also installed. In this case, a ublox Evaluation Kit C94-M8P\(^7\) was used. Beyond the localization of the box in outdoor scenarios, this GPS has been also used for time synchronization of the box. To improve the localization obtained by the module, the GNSS was configured to received correction data from the UHF frequency antenna (433 Mhz) which is installed on the right side of the box. This antenna received the RTCM messages from a base station localized within area of the SensorBox.

---

5 https://www.ouster.io/product-os1 Last accessed 16/05/2019  
6 https://www.stereolabs.com/zed/ Last accessed 16/05/2019  
7 https://www.u-blox.com/de/product/c94-m8p Last accessed 16/05/2019
2.4.1.6 Integration
The sensor box will be the system under test in Task 4.4. ViF will collaborate with AVL and AIT for system integration and validation. Related design patterns are:

- (4) Data Fusion Approach;
- (14) Sensor data analytics.

2.4.2 Sensors to measure vital signs (Senetics)

2.4.2.1 Overview
Senetics focuses on the development of sensor modules on the steering wheel which can measure various vital signs and environmental parameters in the car. The goal is to monitor for possible driver fatigue. Unlike other concepts, the focus is not on measuring the steering movements or checking eye activity via cameras. The sensor data will be used in sensor fusion algorithms to be able to detect levels of fatigue. Via software it is possible to monitor the parameters as real-time data.

In order to detect vital signs and environmental parameters, Senetics is using different sensors. Some sensors measure via the skin and other sensors measure via the air. The sensors are in one of three modules at different positions on the steering wheel according to their functionality. For example, the gas sensor to detect the air quality is located at the top of the steering wheel. The pulse sensors are on the right and the left side of the steering wheel, because the hands of the driver are mostly located at the 10 o’clock and 2 o’clock position. For the module at the 12 o’clock position, the following sensors are used: a gas sensor for air quality (TVOC and eCO₂) and an ultrasonic sensor for the breathing frequency.

The same sensors are used at the 3 o’clock and the 9 o’clock positions: a skin conductance sensor for skin conductance with electrodes, a force sensor for the contact pressure on the steering wheel, a temperature sensor for the body temperature, an ECG sensor with electrodes for an ECG, a photoplethysmography sensor for pulse and oxygen saturation and a temperature and humidity sensor for the environmental temperature and humidity.

All these parameters provide information about fatigue. Senetics will develop an algorithm to detect the level of fatigue and the human health status via vital signs monitoring. This fatigue detection algorithm is necessary because individual parameters cannot provide sufficient information about fatigue. The fusion of the sensor data will provide a reliable indication and is the reason for using an algorithm. Every parameter has its own weighting in accordance with its significance. The algorithm is explained in more detail in chapter 4.1. The structure and individual parameters are shown in more detail in Figure Error! No text of specified style in document.-7.
2.4.1.3 Integration

With regard to the integration of this solution, the system will be part of the WP7 laboratory demonstrator. UC2.2 will be used to detect drowsiness and the health status of the driver. No specific collaboration on this component is foreseen, because only Senetics is working on fatigue detection and health status via vital signs monitoring. In WP7, Senetics is working with other partners who want to detect fatigue and drowsiness using other methods. All demonstrator systems will feature in one laboratory demonstrator in year three.

The steering wheel sensor clips are mainly used for data acquisition of vital signs and environmental parameters. The next steps will be data analysis and data fusion to develop an algorithm to detect driver health status and drowsiness.

2.4.3 Real Time Location System (TST)

2.4.3.1 Overview

The proposed solution involves a Real Time Location System (RTLS) using Ultra Wideband (UWB) technology, which will help in the stationary vehicle detection in some WP 9 demonstrators. RTLS can accommodate different UCs. Since the SECREDAS WP9 demonstrators will run in scenarios close to real-life conditions, it seems logical to try out the “traditional” RTLS scheme. The location of tagged objects (in this case: cars) is established using several fixed anchors in known locations around the area in which the tagged objects are located. These anchors can be separate units or can be incorporated into Wireless Access Points. A number of implementation schemes are possible; some of them based on radio signal strength (commonly referred to as Received Signal Strength Indication or RSSI schemes), and
some others based on the measurement of Time, where the time it takes the radio signal to travel between transmitter and receiver is measured using one or more of a variety of different techniques and then, knowing the value of the speed of light and applying simple calculations, the distance can be calculated with high accuracy. Figure Error! No text of specified style in document.-8 shows a depiction of the traditional RTLS scheme based on a fixed infrastructure.

Figure Error! No text of specified style in document.-8: RTLS fixed infrastructure

A different approach (and recommendation) in a first-responders situation would be to employ the relative location among a group of nodes. In this situation, there is no fixed infrastructure so nodes must establish their location relative to other nodes in the network. Thus, this is ideal for situations where emergency services arrive at a building that is not equipped with an RTLS infrastructure but yet they need to track the progress of personnel as these enter the building (only the first-responders are tagged). In order to have this information at command centre level (which in this particular case and demonstration could be a vehicle), one or more of the first responders need to be linked wirelessly to the command vehicle outside the building. If absolute location is required, then at least two nodes that are in known locations relative to the target are required. These would need to be "dropped" by the first responders or their support crew on arrival at the building and their locations noted. This approach could possibly be applied to certain contexts related to the automotive environment, and therefore TST would like to test its usability and validity.

Relative location on its own is useful in that it tells team members where they are relative to other team members, however, for maximum usefulness absolute location is also required. Networks created using this technology have a maximum limitation of up to 16 location sensors and 9,000 tags, which are the mobile devices attached to assets or carried by people. Overlaps may happen in a scenario that heavily populated. Nevertheless, the network should be capable of sending warnings concerning to collisions. A depiction of the relative location solution is shown in Figure Error! No text of specified style in document.-9.
For the case of the “stationary vehicle” the idea is to use the RTLS system based on UWB, relying on a collection of mobile tags, which in this case would go in the vehicles, and beacons, which delimit an area of action. In the proposed scenario we could perform a couple of trials looking for the optimal solution. Therefore, we may use both this version of the system, including tags and beacons, as well as another, simpler version, in which we would only deploy the tags in the vehicles and establish a communication directly among them, this time not relying on the beacons. The result would be that the real position (with high precision) of each vehicle can be determined, as well as the relative distances between one vehicle and another. Apart from this, the use of UWB localization would also contribute to a significant increase in the precision of vehicle-to-roadside (eg. an intersection) location. This will become essential for urban intersections when automated vehicles need to operate with high efficiency within a limited space.

As of today, UWB positioning is an emerging additional source of good relative position information for ITS and the automotive sector, as referenced in ETSI EN 302 890-2 (ITS Position and Time) [REF]. Having said this, it is worth mentioning that UWB relative location information in a vehicle will not be used directly but rather as a source for its location / dead reckoning engine.

2.4.3.2 Integration
With regard to the integration of this solution, the UWB+RTLS system will be part of the tests conducted in WP9 Demo II UC2.2, to help protect connected vehicles from external threats. The component itself, which is related to the V2X communication CTE, will be fully developed by TST. Nevertheless, its further integration will be addressed by the main partners involved in the WP9 DEMO II demonstrator.
3 Computing platforms

There are various performance critical requirements that have to be fulfilled by the computing platforms that run the complex software algorithms for processing sensor data. This chapter summarizes the used computing platforms.

3.1 GCP & portable PC (MERANTIX)

MERANTIX relies on two kinds of computing platforms for the SECREDAS project:

1. For large scale model training, cloud computation devices will be used. For this purpose, MERANTIX uses the Google Cloud Platform (GCP). Given the required training data, no communication with other on-board modules and machines of the autonomous vehicle is required. Therefore, the training process can be off-boarded to cloud devices which significantly reduces training time and therefore accelerates the development process.

2. In addition to cloud computation, MERANTIX will also rely on local machines for prototyping and for on-board deployment. When perception models are deployed on a vehicle, it is crucial to limit the transmission of safety critical information to a minimum. Therefore, on-board processing of image data is required which allows for a smaller attack surface. The on-board computation devices (i.e. for UC1.5) will directly rely on the recorded camera information and publish the results to other software modules within the autonomous vehicle software pipeline. For on-board deployment MERANTIX will use a portable notebook with a built in Nvidia GPU (GTX 1060).

The training software will be integrated into a cloud computation platform independent of any demonstrator. For UC1.5, on-board computation is required in order to run the robust perception models in the vehicle. Therefore, the models will be integrated into a mobile computing platform which can communicate with the rest of the vehicle through middleware messaging.

3.2 ARTPEC 6 & portable PC (CRF)

In addition to the roadside surveillance camera, CRF will provide a computing platform for advanced object detection and fusion inside WP4 and for WP9 DEMO I (sub-scenario 1.1, 1.2, 1.3 and 1.4). The network cameras already include a programmable computing platform which allows inclusion of specialized processing modules. The selected camera (Canon Axis P1367-E) includes an ARTPEC 6 processor and 1GB of RAM. This platform accommodates video analytics on the edge to detect people and objects. Specialized software modules developed by CRF or other Canon group companies can be embedded into the Canon camera computing platform.
In order to carry out experiments with more advanced analytics that are not yet commercially available, CRF will also provide a separate computing platform dedicated to video analytics and communication. The platform tested by CRF is currently composed of a portable PC (Intel core i7) with an external graphical card Nvidia GeForce GTX 1080 contained in an eGPU enclosure Razer Core X. This platform allows the execution of object detection and tracking software relying on deep neural network algorithms. This video analytic platform can receive encoded video and metadata from Canon cameras, but it could potentially also receive video streams from other network cameras available at the demonstration site. Depending on the selected architecture, a sensor fusion algorithm developed in Task 4.3 could also be embedded into this computing platform or into another platform such as the RSU provided by CMS. The integration work in the WP9 DEMO I demonstrator will be carried out in collaboration with TNO and Commsignia. Related design patterns are:

- (4) Data Fusion Approach;
- (14) Sensor data analytics.

### 3.3 Arduino IDE (senetics)

Senetics uses the following software to program the microcontrollers and the sensors:

1. “Arduino IDE” is used to control the microcontrollers and to develop the program for all sensors. In “Arduino IDE” it’s not possible to have an output of all data of different microcontrollers at the same time.

2. To mitigate the previous issue, Senetics uses “Qt Creator” software to develop a GUI which shows all sensor data from different microcontrollers in real time. It’s also possible to develop plots in the GUI for various parameters.

### 3.4 NVIDIA Jetson Platforms

#### 3.4.1 BeyondVision

For the initial development, BeyondVision will use a server to train Neural Networks written in Python. After the software development, two Jetson Nano with Ubuntu installed, will run the software and receive all data from the LIDARs and cameras. A partnership is already established with PDMFC for integration activities. Related Scenarios are: UC 1, 2, 3, 4 and 6. Related Designed Patterns are:

- (1) Data Governance & Multi-party Access control;
- (3) Conditional Safety Certificates;
- (4) Data Fusion Approach;
- (5) Authentication and Authorization module;
- (6) End 2 End Security;
- (11) Anomaly-based Network Monitoring;
• (12) Secured Embedded Network;
• (14) Sensor data analytics.

3.4.2 PDMFC
To implement the software developed by PDMFC and by third parties, the Linux operating system will be used on an NVIDIA Jetson. Containers, as Dockers or Kubernetes, can also be tested due to the security environment. The following UCs will be used: UC1, UC3 and UC6. The following design patterns will be covered:
• (4) Data Fusion Approach;
• (13) Secure communication to mobile robots;
• (14) Sensor data analytics.

3.4.3 VIF
The sensor box is controlled by a NVIDIA TX1 development Kit\(^8\), which is placed inside the box. This board runs a L4T based on Ubuntu 16.04 as operating system where ROS carries out the message communication of the sensors and provides tools for the integration, such as visualization or simulation (see section 0).

4 Data processing

The analysis of sensor data including sensor fusion algorithms is important to enable automated driving. Automated driving functions rely on the sensing system of the vehicle and therefore, sensor data, sensor fusion and object detection algorithms are crucial. In this chapter, several concepts are described that advance sensor fusion and detect anomalies in sensor data to increase safety and reliability of perception.

4.1 Sensor fusion

4.1.1 Environment perception (AVL-SF)

4.1.1.1 Overview
Analogous to our human sensing system, popular sensors in autonomous driving area include RGB and infrared cameras, radar, LIDAR, ultrasonic sensor, etc. Each sensor type has its pros and cons in covering various scenarios. For example, a vision camera and computer vision or deep learning approaches can generate a good understanding of the scene/situation, whereas a radar or LIDAR sensor is good at providing distance information without the need of much further data processing. Hence, it is vital to combine sensing results from each sensor. Moreover, functional safety is of great importance to achieve automated driving and functional safety requires the fusion of different types of sensors and hardware.

AVL-SF’s aim is to first investigate and implement fusion methods in order to achieve robust environment sensing performance by combining the advantages of different sensors. This approach is used to mitigate sensor errors. Activities are based on AVL SF’s previous and current work in core functionality on sensing, perception (object recognition) and sensor fusion. Sensor fusion refers to the merging of data from various sensors to gain more information compared to using a single sensor. It remains open whether low level sensors outperform high level sensors and if so, in which scenarios and applications. The Kalman filter, which was developed in 1960s, and variations thereof will be used.
AVL-SF will research low level and high-level sensor fusion in order to determine the proper application scenarios for each fusion type. Based on the results, AVL-SF will be able to choose the correct fusion method at different data processing stages. Some fusion methods will be implemented and benchmarked. This will create an in-depth understanding of the pros and cons of each method help to identify the corresponding function flaws. Functional safety is an optional topic in sensor fusion. AVL-SF also plans to further utilize and apply the selected fusion methods to provide outputs for other SECREDAS activities related to WP4, such as a development of a Local and Global Dynamic Map ("Shared World Model") to increase overall safety of all traffic participants.

4.1.1.2 Integration
The sensor fusion approach will be applied in UC1 (WP9, Demo I) and UC2 (WP 9, Demo II). Primary focus is to use the aforementioned methods to improve safety and robustness of automated driving functionalities. Developed methods can also be demonstrated as separate smaller demos by using MiL/SiL or potentially ViL tests as part of WP4.

No concrete cooperation with other partners has yet been established, but certain synergies have already been identified. Examples are: BeyondVision (deep learning algorithms) and CEA (LiDAR, radar fusion). Also, the in-vehicle detection by UOULU and Senetics can be seen as complementary to the work of AVL-SF to create a system that is capable of handling both out- and in-vehicle events. The applied Design Pattern is: (4) Data Fusion Approach.
4.1.2 Vital sign measurement (senetics)

4.1.2.1 Overview
The aim here is to make the driver aware of his fatigue while driving. For this purpose, data must be evaluated in real time in order to calculate the data in an algorithm. The algorithm assigns different weightings to the parameters and calculates these weighting factors accordingly. Depending on the amount and type of parameters detected, the algorithm then displays different levels of fatigue. The level of fatigue is shown on a display in the car in form of a traffic light. Each level of fatigue is assigned a different traffic light color: green shows light fatigue, yellow means moderate fatigue and red indicates heavy fatigue. This way, the driver receives timely information on when it is time to take a break. The procedure from the sensor data to the output of the information of the driver’s fatigue level on the display is shown in Figure Error! No text of specified style in document.-11.

Figure Error! No text of specified style in document.-11: Rough system architecture of the steering wheel sensor clips with decision tree included.

4.1.2.2 Integration
The algorithm will be part of the WP7 laboratory demonstrator will be used mainly for data analysis and data fusion to detect driver fatigue, drowsiness and health status. UC2.2 will used. There is no cooperation with other partners in WP4 concerning this algorithm, because only Senetics is working on the detection of fatigue and the health status via vital signs monitoring. In WP7 Senetics is actively cooperating with other partners who want to detect drowsiness by other methods. The sensor clips, with the algorithm included, will be collected in the WP7 laboratory demonstrator in year three.
4.1.3 Driver and vehicle status (UOULU)

4.1.3.1 Overview

The system investigated by UOULU makes inference and predictions about the current status of the vehicle and the current well-being and health status of the driver, in order to assess their capability to safely perform his/her tasks. A number of sensors, both unobtrusive wearables and in-vehicle, are used as data source. Commercial off the shelf (COTS) sensors are used while dedicated methods and algorithms are developed and adapted within SECREDAS. Remote monitoring is provided to enable services (e.g. maintenance). Optionally and in addition to that, the decision-making system could provide its output metrics to the actuators envisaged in other use case scenarios.

Sleep quality affects readiness during the following work period, while environment quality (temperature, oxygen/carbon dioxide, etc.) affect both driver’s readiness and passenger comfort. Car resource use affects the need of maintenance and more generally logistics. To assess alertness and physical ability of the driver, hence their ability to engage in the driving task, the Sensor Gateway collects health and wellbeing data from Wearable Sensors (on-body sensors) and presence and status of the driver from Dashboard Sensors (on-board sensors). Sensing data are represented by unobtrusive wearables that measure physiological metrics (hearth rate, sleep quality, etc.) collected both before (hours, days) and during the driving task, and in-vehicle sensors observing driver position (e.g. distance from the dashboard), plus use-case-specific information about the car(s), such as temperature, carbon dioxide, use of on-board facilities (e.g. occupancy of seats, utilization of toilets, etc.). Moreover, to ensure that selected vehicle components satisfy security and safety requirements, also data from those components can be collected by the Sensor Gateway.

![Diagram](image)

Figure Error! No text of specified style in document.-12: Data processing for driver and vehicle status.

Data processing is hierarchically split between on-board and in-the-cloud (see Figure Error! No text of specified style in document.-12). To improve safety and to cope with possible temporarily loss of connection, a certain amount of information is kept and maintained on board. Data analysis carried out on-board uses less information than available in the cloud, therefore it provides a simpler but faster response at the vehicle.
The Sensor Gateway delivers data to the Data Analytics through the Cloud Gateway. The outcomes of the Data Analytics are further made available to remote stakeholders (thus enabling maintenance, fleet management, logistics, service adaptation, optimization). Outcomes are also sent back to the Sensor Gateway. Capability of driver and vehicle are continuously evaluated and decision metrics are generated and properly routed to generate alerts and warnings. In particular, the Sensor Gateway may provide decision metrics to External Actuators for prompt reaction (optional).

Sensor data is collected and analyzed to produce indicators for alerting/warning the driver or the safety of the vehicle, but also to provide information about the conditions of the vehicle for service and maintenance planning. Signals from the Sensor Gateway, either generated locally or coming from the Cloud Gateway, are also forwarded to the driver as applicable, e.g. as visual or auditive stimuli, or to trigger driver-related actions, such as activating (audio/video) communication between the driver and the control center.

4.1.3.2 Integration
The data processing component described here is related to the Sensor Data Analytics design pattern. The above system is used in the health related UC2.3 (see SECREDAS D1.2), which targets the rail domain but is applicable to the road case as well, including coach and truck services. The activity feeds into Demo II in WP9. UOULU cooperates with Nokia, Solita and Haltian. Moreover, related activities are carried out at Roche, Senetics and AVL among others. In relation to those, collaboration perspectives target both alternative realizations/implementations and complementary solutions.

4.2 Sensor data analysis
4.2.1 Increase robustness of perception (MERANTIX)
4.2.1.1 Overview
Similar to the computation platforms, also the data processing pipeline differs for model training and on-board deployment of the algorithms:

1. For model training, offline computations can be conducted. Thus, data needs to be recorded and can be transferred to the location of processing, which as described above can be a local computer or a cloud device. The processing is not real-time critical.

2. During deployment on a vehicle, the algorithms should not rely on communication to a cloud device for safety reasons and therefore on-board computation is required. Therefore, it has to be assured that the algorithms can run in real-time on an on-board computer, which will process the data originating from an on-board camera.
4.2.1.2 Integration
As the goal is to run the robust perception models on-board the vehicle in WP9 DEMO I, the data pipeline and processing needs run in real-time to avoid any delays. The processing pipeline needs to be integrated with the rest of the system of DEMO I, as the input to the models are on-board camera images. Therefore, the on-board communication needs to be set up using the middleware, such that all relevant modules can communicate with each other and process the data efficiently.

4.2.2 Error and attack detection (PDMFC)

4.2.2.1 Overview
As mentioned in the section on sensing components, PDMFC will work with cameras and develop software for image recognition. LIDARs will also be used in order to exploit them and analyze the potential failures that can be present in its usage.

All autonomous vehicles are very dependent on the sensors they use to obtain information about the outside environment. If any of these sensors has a critical failure or sends incorrect information, the result can be catastrophic. With this in mind, it is necessary to refine all error detection mechanisms and attack vectors on these sensors. In order to predict in real-time whether the data from the sensors is accurate, PDMFC intends to build a neural network capable of analysing data from different sensors (sensor fusion); if the data is invalid, the network should then notify other systems to provide mitigation methods.

There are several types of attacks that can be used on sensors. Some of the most common attacks are: Blinding, Jamming, Replay Signal etc. Our research will focus on detecting and mitigating attacks on devices like LIDAR, RADAR and CAMERA.

4.2.2.2 Integration
PDMFC has created Deep Learning algorithms capable of detecting some types of CAN bus (WP6) attack with 96% accuracy; the intention however is to extend this solution to other environments (as represented in WP4, WP5 and WP6) and if possible, detect an attack before it actually happens. For this, PDMFC is currently approaching different partners across the mentioned WPs who could provide real data on the different sensor types. The applied Design Pattern is: (4) Data Fusion Approach.

4.2.3 Real-time sensor data analysis (BeyondVision)

4.2.3.1 Overview
Image recognition software is being developed in Python by using some of its most useful libraries, such as TensorFlow, CUDA, and OpenCV. This software makes use of Neural Networks to process large data easier and faster. Data processing from BeyondVision will be focused on:

1. Data processing deriving from LIDAR, RADAR and CAMERA;
2. Data processing related to Design patterns such as (11) Anomaly-based network monitoring.

**LIDAR, RADAR and CAMERA:** Deep learning algorithms will be tested to evaluate the effectiveness of data collected and processed from LIDAR, RADAR and CAMERA. This will provide a combined real-time data analysis which will result in advanced obstacle recognition and other conditions derived from UC1, UC2 and UC4. For model training, a large dataset will be used which will be collected with drones. It is therefore important to maintain a well-designed deep learning model in order to process all the collected data and continuously evaluate the results. Drones will enhance this process, because with them large amounts of data can be collected. Drones can also easily be adapted to UCs for further investigation. Moreover, data processing includes the control and handling of LIDAR vs RADAR functionality, their combined usage and process collaboration for providing the best and most accurate results.

**Anomaly-based network monitoring:** Advanced algorithms and deep learning methods will be used in order to detect anomalies inside the network. The process of detecting anomalies inside the network is based mostly on the TCP/IP model, whether anomalies based on the data which the sensors (LIDAR, RADAR and CAMERA) are giving is something different. However, by applying the acquired experience from using Intrusion Detection Systems (IDS), machine learning and deep learning methods, BeyondVision will develop a dedicated detection system appropriate for using it on automotive and its applications.

### 4.2.3.2 Integration

**Related Scenarios are:** WP9, Demo 1 – Scenario 1, 2 and 3. WP9 Demo 2 – Scenario 1 and 2.

**Integration plans:** In the later stages of the project, software for sensor fusion will be developed, as mixing data from LIDAR and CAMERA will supply a clearer image of the car’s outside environment. This is groundbreaking and the starting point for software that will be able to predict car trajectories.

### 4.2.4 Video analytics (CRF)

#### 4.2.4.1 Overview

CRF network cameras are intelligent cameras and provide not only captured images but also a real time analysis of the video content. In SECREDAS, the video analytic tools provided by CRF may be either commercially available modules from the Canon group or prototypes of advanced analytic modules developed by CRF. The video analytic modules allow detection and recognition of objects in the videos. Using advanced camera calibration and multi-camera object tracking, the video analytic modules are also able to compute the geographical position and velocity for each detected object. Table **Error! No text of specified style in document.**-2 lists the main parameters provided for each detected object by the video analytic module.
| Id<sup>i</sup> | Identifier of the detected object i |
| Conf<sup>i</sup> | Confidence level of the detection of the object i |
| x<sup>i</sup> | Position of the detected object i |
| v<sup>i</sup> | Velocity of the detected object i |
| Age<sup>i</sup> | Age of the detected object i, i.e. for how long the object has been detected |
| Class<sup>i</sup> | Class of the detected object i. For example: pedestrian, cyclist, vehicle |

Table Error! No text of specified style in document.-2: Detected object parameters from video analytics

CRF may also provide video analytic modules from the Canon Group (specifically: Axis Citilog) targeted to traffic analysis. These modules use advanced calibration to describe the road and the traffic structure, for example the lane type definition (e.g. sidewalk, emergency lane) or the traffic direction. By using this information, it becomes possible to automatically detect different types of traffic incidents: a stopped vehicle (in incorrect lane, in fluid traffic or in congested traffic), congestion, a pedestrian on a road, a wrong-way vehicle, a slow vehicle, loss of visibility or debris on the road (see Figure Error! No text of specified style in document.-13).

![Figure](image13.png)

Figure Error! No text of specified style in document.-13: Examples of traffic incident detection: stopped vehicle on emergency lane (left), slow vehicle in fluid traffic (right)

Moreover, it is also possible to report with high probability that there is a stopped vehicle, a moving vehicle or an object on some predefined parts of a road (see Figure Error! No text of specified style in document.-14).
4.2.4.2 Integration

The video analytic module will be used in WP4 and for WP9 DEMO I. DEMO I will target scenario 1 and sub-scenarios 1.1, 1.2, 1.3 and 1.4. Specific API are being developed to allow integration with partner modules, in particular those of with TNO and CMS. Several CTEs defined in WP3 are related to these UCs and may impact on the development of the API: Authentication/ Authorization, C-ITS services, V2X Communication, Identity Management, Certificate Management. Related design patterns are: (14) Sensor data analytics.

4.3 Collective Perception

4.3.1 Secure and Collective Perception as a service (Commsignia & CRF)

4.3.1.1 Overview

Within SECREDAS, Commsignia and several partners are developing means to exchange sensor “output” in real-time between traffic participants (between vehicles and between the traffic and the infrastructure). The exchange takes place through V2X communication supported by WP5.2. Message services currently being defined (drafted) within ETSI ETSI TR 103 562 (Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS)) and ETSI TS 103 324 (Specification of the Collective Perception Service) can be used for common, interoperable and standardized data exchange.

Within WP4, a flexible and secure interface towards this service will be developed by Commsignia to allow different vendors to supply or harvest this service. The Collective Perception Message (CPM) allows the transfer of three types of main information blocks:

- Sensor (or fused sensor installation) Field of view;
- Detected objects;
- Free space (confirmed empty).
As each type of sensor and processing algorithm is unique and has different characteristics, standardized abstraction methods and software modules need to be implemented between a proprietary sensor and the CPM service. This interface needs to be able to transfer as much information and trust between the sensor and the service as possible (e.g. a non-standard quality indicator should be translated into a standard value with the least loss of information as possible).

The sensor provides a vendor-specific API (could also be conformant to interoperable specification e.g. can feature a common automotive sensor interface like Autosar). The data needs to be converted (value conversion, geocoordinate system conversion, etc.) into generic data types. The second step is to fill the CPM input structure with all mandatory information, which can include the use of presets or calculation of certain elements (see Figure 46).

4.3.1.2 Integration
Collective perception is used for Demo 1 scenarios planned to be shown at the Helmond test site. CPM may be generated by either roadside units where perceived objects are recognized by the roadside infrastructure, e.g. by CRF roadside sensors (see 4.2.4) or by vehicles providing objects detected by their onboard sensors. CPM is an enabler to various higher-level mechanisms like collision avoidance or misbehavior detection which will be further elaborated in the next deliverable.

CPM functionality (implemented by e.g. Commsignia) exists on top of any detection method (sensor) capable to identify objects that are relevant for traffic safety (vehicles, vulnerable road users, animals, dangerous objects) or occupiable road space (e.g. free space between two moving vehicles). The CPM interface described above uses any useful partial information to share such measurement result in a broadcast mode of communication. Thus, a CPM service consists of a sensor system and a V2X transmitter. Since CPM is standardized, any sensor can share information in an abstract manner. This also results that any receivers may build on the information received. Users of the CPM service are applications residing in vehicles or other traffic participants or roadside systems usually including their own sensor systems (internal sensors), but uses CPMs to collect information from external sources to build a better perception of...
the surrounding by fusing internally measured environmental perception with received information (e.g. a vehicle may have a very good vision of its surrounding, with V2X it can also exchange this information with vehicles 2 blocks away to detect objects / e.g. a cyclist / without line of site).
5 Conclusions

In this deliverable, various sensors for different applications such as collision avoidance, measurement of vital signs and a Real Time Location System have been described. The used sensors are mainly third-party sensors that are used and combined to enable advanced data processing algorithms. The algorithms include concepts for integrating different sensors (sensor fusion), sensor data analytics to improve perception and hence, safety and reliability of sensor data. Over the coming months, the concepts will be further elaborated and integrated in the other tasks of WP4, in UCs and in several demonstrators.
SECREDAS has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement nr. 783119. This Joint Undertaking receives support from the European Union’s Horizon 2020 research and innovation programme and Austria, Belgium, Czech Republic, Germany, Finland, Hungary, Italy, The Netherlands, Poland, Romania, Sweden and Tunis.