ROBOTS IN ASSISTED
LIVING ENVIRONMENTS

UNOBTRUSIVE, EFFICIENT, RELIABLE AND MODULAR SOLUTIONS FOR INDEPENDENT AGEING

Research Innovation Action

Project Number: 643892 Start Date of Project: 01/04/2015 Duration: 36 months

DELIVERABLE 3.10

Integrated data analysis system II

Dissemination Level	**Public**
Due Date of Deliverable	Project Month 24, March 2017
Actual Submission Date	12 April 2017
Work Package	WP3, *Modular conceptual home architecture design and ICT method development for efficient, robust and flexible elder monitoring and caring*
Task	T3.5: *Privacy and resource-sensitive integrated data analysis*
Lead Beneficiary	TWG
Contributing beneficiaries	NCSR-D
Type	DEM
Status	Submitted
Version	Final

Project funded by the European Union’s Horizon 2020 Research and Innovation Actions

Abstract

This deliverable consists of the integrated data analysis prototype that implements the ADL recognition methods developed in this WP.

History and Contributors

Ver	Date	Description	Contributors
01	1 Jun 2016	First draft, establishing document structure.	NCSR-D
02	7 April 2017	Created prototype with components in D3.5. Prepared virtual machine demonstration.	NCSR-D and TWG
03	11 April 2017	Final version given for review	TWG and NCSR-D
04	12 April 2017	Internal review comments and corrections	ROBOTNIK
Fin	12 April 2017	Final document preparation and submission	NCSR-D

CONTENTS

List of Figures

1 Introduction

1.1 Purpose and Scope

This deliverable is an integrated ROS stack for ADL recognition. This stack comprises the prototype implementations of the ADL recognition methods developed in T3.2. The technical description and the explanation of the methodological advances that these prototypes demonstrate is documented in D3.4 and D3.5, the two reports prepared by T3.2.

Within the scope of this deliverable is the ROS-compatible versions of the method prototypes and any auxiliary software needed for integration (Section 2), and the demonstration of the operation of the integrated stack (Section 3).

The robot prototype and other software needed for the operation of the robot (not directly related to ADL recognition) is out of scope and will be developed in Task 4.3 *Robotic platform design and integration* and demonstrated in D4.6 *Integrated Robotic Platform I*. The overall RADIO Home and software needed for fusing the outputs ADL recognition methods, including fusing with home automation sensor data, is also out of scope and will be developed in Task 4.4 *Smart home design and integration* and demonstrated in D4.8 *Integrated smart home with robotic platform extensions*.

1.2 Approach

This deliverable is the result of integrating the prototypes of the sensor data analysis methods into a software system using the ROS middleware as a basis. The design of the pipelines and interconnections between the components aims at providing the modularity needed in order to:

- Have explicit and minimal dependencies, so that the system can be partially installed to satisfy the different clinical and obtrusiveness requirements that each deployment needs to meet;
- Have the modularity that allows executing different modules either near the sensors (e.g., on-board the robot) or at more central computational infrastructures (e.g., a home computer), allowing different configurations to address different needs on battery autonomy, on-board computational power, network bandwidth, and the transmission of sensitive raw content.

This integrated prototype is a software prototype, not addressing its embedding in the RADIO hardware architecture which is the focus of WP4.

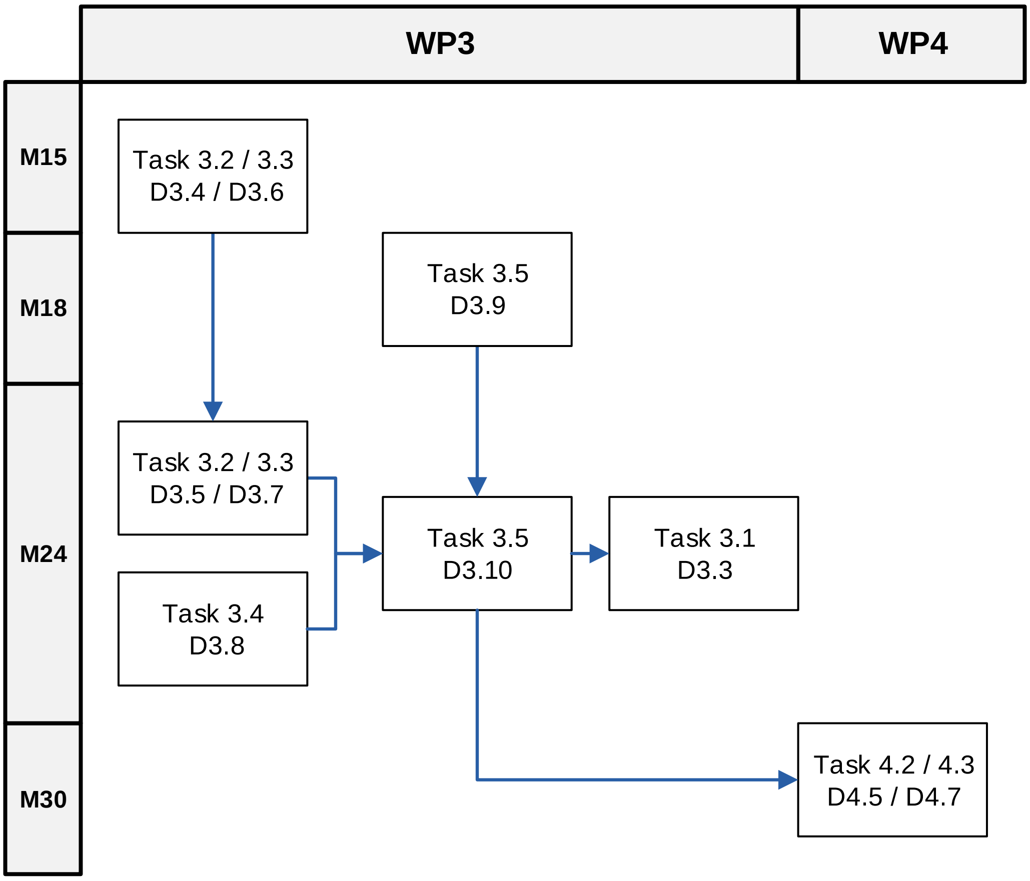

Figure 1: Relation to other Work Packages and Deliverables

1.3 Relation to Other Work Packages and Deliverables

This document is part of a cluster of closely related deliverables. The initial conceptual architecture (D3.1) planned development in Task 3.2 until M18, including a first overview of the state of the art upon which RADIO development was based. In this version, all methodological information has been moved to D3.4 *ADL and mood recognition methods* and this document only retains the architectural design and component interconnections needed to satisfy the user requirements (WP2) and the findings of D3.6 *Network robustness and efficiency methods*.

In this manner, work around ADL and Mood recognition is organized as follows:

- D3.2 *Conceptual Architecture II* (R): design and interconnections of components
- D3.4 *ADL and mood recognition methods I* (R): methods developed in order to implement the components specified in D3.2
- D3.9 *Integrated data analysis system I* (DEM): prototypes of the methods from D3.4

From M19 until M24 the architecture is maintained as a *living document* updated to record adjustments necessitated as development in WP3 progresses. The final version (D3.3, M24) documents the architecture and interfacing of the final data analysis (D3.5, D3.8) and networking (D3.7) methods as prototyped in D3.10.

This deliverable was originally planned to be integrated into the final robot prototype (D4.7), but has been included in the first prototype (D4.6).

2 PROTOTYPE

The following ADL recognition suite is the result of integrating in ROS the method prototypes from D3.5 and adding them to D3.9, the previous iteration of this deliverable:

Description	Source code repository and release used for this deliverable	
HumanPatterRecognition: Recognizes human walking patterns in laser scans and tracks walking [D3.4, Section 2].	https://github.com/radio-project-eu/HumanPatternRecognition	v2.0
ROSVisual: Tracks moving objects in the RGB/depth modality [D3.4, Section 3] and classifies motion as bed or chair transfer [D3.5, Section 2].	https://github.com/radio-project-eu/ros_visual	v1.1
Motion Analysis: Recognizes motion and classifies it as “bed transfer” and “pill intake” events [D3.4, Section 4].	https://github.com/radio-project-eu/motion_analysis	v1.2
Simple Motion Analysis/FPGA: Re-implementation of the motion analysis above without using the OpenCV library. This version is better suited to FPGA optimization.	https://github.com/radio-project-eu/simple-visual-events	master branch
AUROS: Recognizes talking, watching TV, listening to music, and doing housework by acoustic analysis of the audio modality [D3.5, Section 3].	https://github.com/radio-project-eu/AUROS	v1.0
BLE Localization: A system of applications for BLE beacons, BLE gateway, and mobile device (including the robot) for localizing mobile devices using the beacon signal [D3.5, Section 5].	https://github.com/radio-project-eu/relative_multihop_localization https://github.com/radio-project-eu/robot_ble_localization https://github.com/radio-project-eu/indoor_relative_localization https://github.com/radio-project-eu/ble_gateway	

Auxiliary software was developed for D.10 to facilitate the integration of the outputs of the components above, or using the components above to provide further services. Together with the *HPR Wrapper* and the *Motion Analysis Wrapper* that were developed for D3.9, the final integrated system has:

Description	Source code repository and release used for this deliverable	
HPR Wrapper: Uses HPR output to recognize and time “walked 4m” events.	https://github.com/radio-project-eu/hpr_wrapper	v1.0
Motion Analysis Wrapper: Uses the output from motion analysis to time the bed transfer event.	https://github.com/radio-project-eu/motion_analysis_wrapper	v1.0
ROSVisual Wrapper: Uses the output from ROSVisual to time chair and bed transfer events and to recognize and time “walked 4m” events.	https://github.com/radio-project-eu/ros_visual_wrapper	v1.0
Anti-theft alarm: Application running on the BLE Gateway. Identifies theft attempts, by observing gyroscope and accelerometers values from sensors installed on the robot, and sent notifications messages to the RADIO backend system.	https://github.com/radio-project-eu/anti_theft_alarm	
Map convergence: Application running on the robot. Consumes robot pose messages from the localization package and converts them to the corresponding coordinates in the RADIO Home model.	https://github.com/radio-project-eu/map_convergence	

The following auxiliary software is not part of the ADL recognition stack, but is part of the solution and is needed for deployment and configuration:

Description	Source code repository and release used for this deliverable	
Motion Analysis: Technicians’ interface for motion analysis configuration. This is implemented as a separate ROS node that publishes configuration packages to the main motion analysis node that implements the method.	https://github.com/radio-project-eu/motion_analysis	v1.2

3 DEMONSTRATION

The ADL recognition stack in this deliverable is demonstrated as a virtual machine in OVA format and can be downloaded at http://hdl.handle.net/21.15101/13

The downloaded OVA needs to be imported to VirtualBox [https://www.virtualbox.org] or similar virtualization software. The OVA appliance includes a full Ubuntu and ROS/Indigo installation, the prototypes, and sample recordings from the RADIO Dataset (cf. D3.5, Section 2.2), so that the demonstration does not rely on downloading the complete dataset.

Once the virtual machine boots, it logs in without a password. The demonstration starts by opening a terminal and executing the script called “run”

The demonstration opens a new window to show the video recording, while the recognized events are printed in the terminal where “run” executes.