Project FoSIBLE
Fostering Social Interactions for a Better Life of the Elderly

Deliverable

WP3/Task 3.6 Early Hardware Prototypes, Integration of a vision based sensor technology into a furniture object

Responsible

AIT

Participants

AIT, Mauser Care, FHG

Abstract

The integration of the 3D sensing devices used in the project FoSIBLE into the furniture object and in InHaus smart home Lab environment is described. Three UCOS sensors that will be positioned in the living room and kitchen will perform people counting and a simple gesture control of the iTV widget of the FoSIBLE social media platform.

Table of Content

1 Introduction

1.1 Purpose of the Document

The purpose of this document is to document the integration of the 3D sensing devices used in the project FoSIBLE into the InHaus smart home Lab environment. Three UCOS sensors that will be positioned in the living room, kitchen will perform people counting and a simple gesture control of the iTV widget of the FoSIBLE social media platform.

1.2 Definitions, Acronyms and Abbreviations

Acronym	Description
3D Sensor	Devices that delivers spatial (three dimensional 3D) information of a given scene , typical x,y and depth-z
AE(R)	Address Event (Representation)
HW	Hardware
IF	Interface
IP	Internet Protocol
IR	Infrared
MS	Microsoft ®
OS	Operating System
PC	Personal Computer
PCB	Printed circuit board
PoE	Power over Ethernet
R/C	Remote control
SW	Software
TAE	Timed address event data
UDP	User Datagram Protocol
USB	Universal serial bus

2 Overview

2.1 Goal of the sensor integration

The sensors that will be used for the gesture interaction with the iTV application and for the "social state monitoring" is supposed to integrate seamlessly into the furniture object. The impression of a camera that is watching the user must be prevented.

Based on the results of WP1 and WP2, the objective of this task is to develop and build prototypes of the interfacing devices that can be used to prove the new interaction concepts. These prototypes will be a combination of sensors and actuators with furniture. As a result, the goal of this work package is to develop a set of "smart" furniture that accompanies the TV-based applications by natural and intuitive user input and output options. Major topics that are addressed by these concepts are awareness (being aware of presence/state/well-being of other persons), gathering context information (collecting information about the current state or intention of the habitant), and control (controlling devices/applications). The innovative hardware solutions will be connected to the platform of the TV-based social media center in order to ease interaction of elderly people with the applications.

2.1.1 Gesture control of the iTV set

To control the iTV-set the user has the possibilities to turn the widget on and off by using simple hand gestures. This is realized by sending a remote control command sequence ("macro") to the TV set.

2.1.2 Awareness on user position in the room and number of persons present

To support the awareness on "social state" there will be a function for registering how many people are present in the room or apartment. Furthermore, to support the smart home functionality, there will be a function to locate the persons in front of the TV set, Smart PC or in the kitchen.

2.2 Requirements

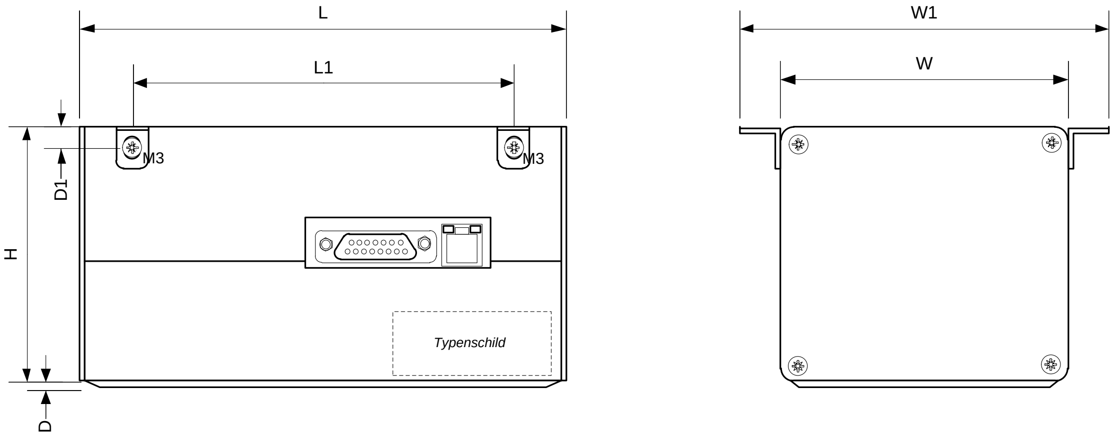
1. Gesture sensors must recognize four simple gestures performed in front of TV set.
2. Gesture sensors must transfer recognized gestures to the Mauser runtime hardware.
3. People counting sensors must recognize the number of people in certain areas of the service apartment.
4. People counting sensors must transfer the people count to the Mauser runtime hardware.
5. Mauser runtime hardware must initiate actions in the smart home environment such as the turning-on of devices/TV/light based on data of (2) and (4).
6. Gesture sensor must be integrated seamlessly into the furniture of the service apartment (the "camera" impression must be avoided).
7. All sensors must be integrated into the service apartment infrastructure (power and data connection).
8. Mauser runtime hardware must pass information on to the "data broker" as needed

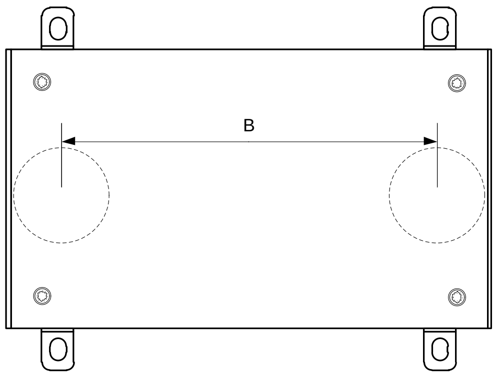
3 Sensors specifications with respect to integration

3.1 UCOS2XL Sensor

The UCOS2XL sensor exhibits an uncoated aluminium housing with four M3 mount holes at its two longer sides. There are two round openings positioned in front of the device, which are needed for the 3D sensing function. For the 3D sensing capability the two openings have to be kept free from occlusion. A covering of these openings with glass is permitted as long as the reduction of the light intensity is kept below 5%.

The device is connected to a LAN infrastructure via an Ethernet cable with a RJ45 connector. The RJ45 socket is located at the side of the device. The device is supplied with external DC power via an SUBD connector located next to the Ethernet socket. The device needs not to be cooled actively which enables a more easy integration into the furniture object. A drawing of the device with dimensions is given in Figure 1.





	UCOS2	UCOS-XL
L	175 mm	306 mm
L1	137 mm	268 mm
B	130 mm	250 mm
W	104,5 mm	
H	88 mm	
D	3 mm	
D1	10 mm	
W1	235 mm	

Nicht maßstabsgetreu!

Figure 1. UCOS2XL technical drawing with dimensions in mm.

3.2 Mounting specifications

The field of view of the device is limited to an angle of 60° horizontally and 60° vertically. This has to be considered when the device is placed within the furniture and when the furniture itself is placed with respect to the main seating-accommodation in the room.

Detection zone, also called monitoring area, is a term for the visible angle corresponding to the sensor's detection field of view at shoulder height.

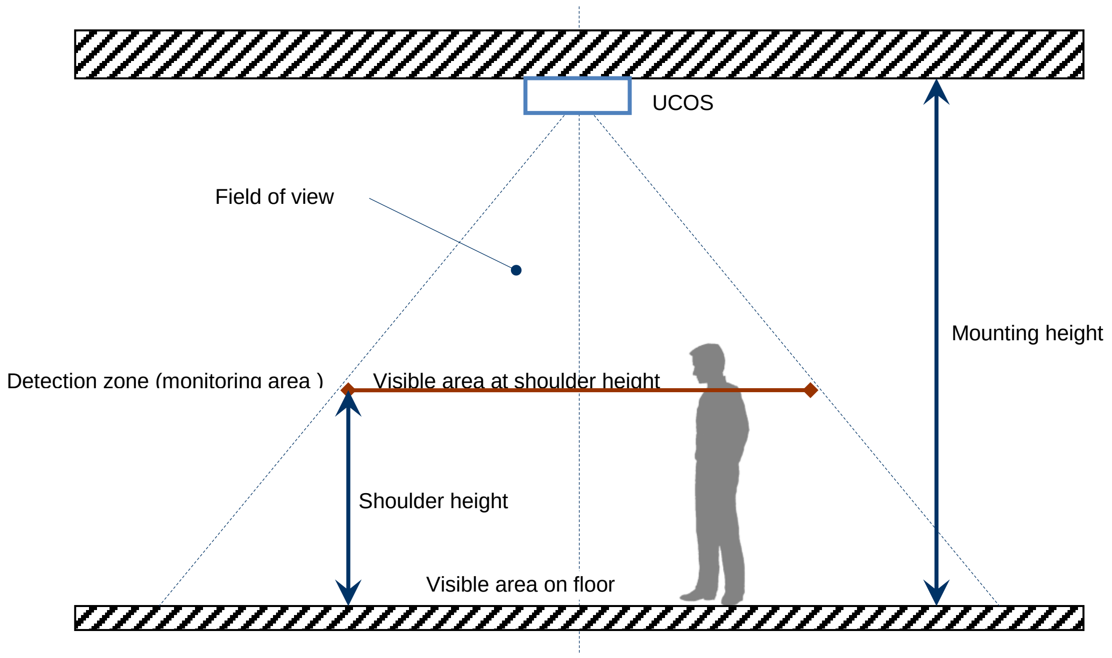


Figure 2. Visualisation of the dimensions in the sensor settings.

The detection zone depends on the height of mounting and the lenses used. The system can be adapted at the same height to cover different monitoring areas through a change of the lenses.

Mounting height in m	UCOS XL Lens f=4,0
2,4	not applicable
2,6	
2,8	
3,0	
3,2	
3,4	
3,6	244
3,8	269
4,0	295
4,2	321
4,4	346
4,6	372
4,8	397
5,0	423

Figure 3. Relation between the height of the UCOS sensor and the detection zone.

4 Integration

4.1 Fixed Sensor Mounting

In this case the end-user has a defined space from where to interact with the system, which is integrated in the furniture in a fixed position.

Restricting the end-users freedom of choice to some degree, this configuration is less expensive and allows more options of integrating the system into furniture with a less technical looking design. Another advantage is the option to cover the optical sensor when it is not in use, making a clear separation between on and off situations as well as avoiding the „being watched effect".

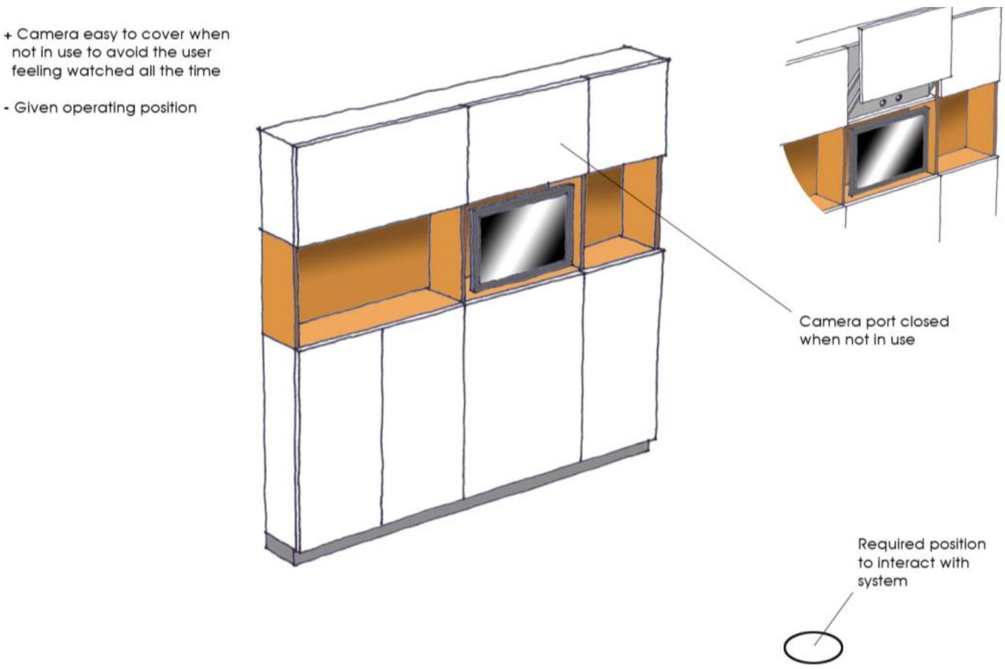


Figure 4. Fixed Sensor mounting.

4.2 Sensor mount on the iTV set

For the sake of simplicity and compatibility with the selected iTV hardware, a simpler mount adapter had to be realized in the project to connect the sensor with the TV set. This adapter connects to the VESA mount grid on the backside of the TV set. It allows a firm mounting of the sensor(s) independently of the selected furniture object and also for demonstration without furniture.

A mounting bracket for the UCOS-XL device has been designed and manufactured by Mauser Care and is available at AIT, but it needs some extra space above the TV set (see VESA UCOS adapter in Figure 5

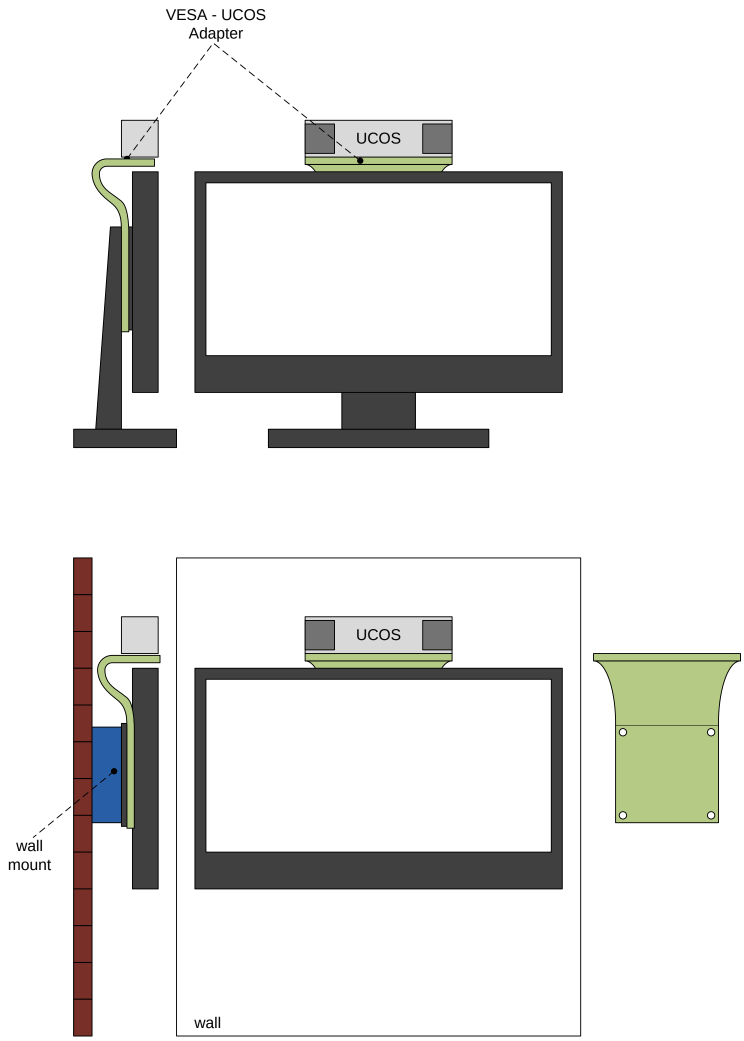


Figure 5. VESA Sensor mounting adapter selected for use in the project.

5 InHaus Installation

5.1 Gesture control of iTV application (widget)

5.1.1 Overview

The following components were rolled out in the inHaus:

- One UCOS XL was integrated into the iTV set with a miniature PC for gesture recognition.
- Four gestures defined to the signals "left"/"right"/”up”/down” of TV widget via TCP/UDP to Mauser runtime HW which in turn controls via the R/C IR interface the TV set.
- Auto start of the widget is assumed, the corresponding IR codes of the remote control for 'bring widget to foreground' and 'send to background' will be sent.

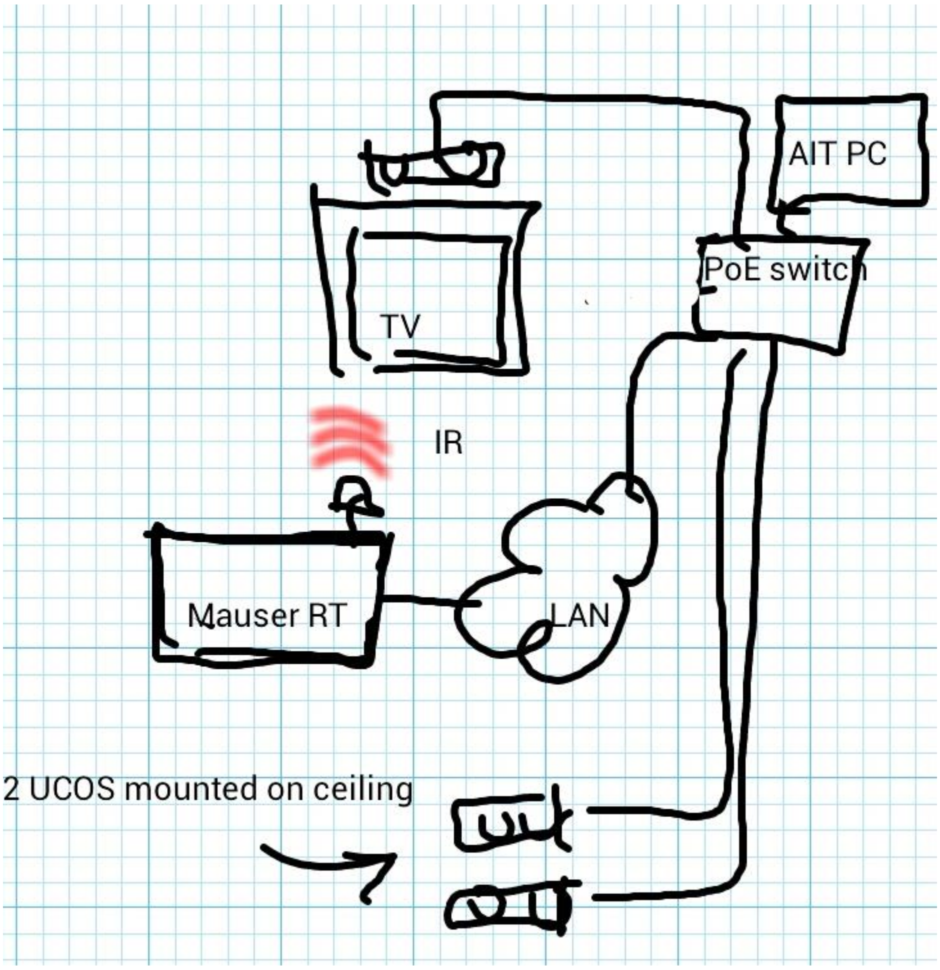


Figure 6. Block diagram of UCOS sensor and other hardware connectivity.

After the evaluation result of D7.2 the need for an improvement of the gesture recognition interface became evident. The gesture recognition algorithms have therefore been ported from the Matlab programming environment to the C programming langue. This has been achieved by partly compiling the Matlab code and partly by newly implementing the core algorithm in C. The gesture recognition software package itself that was developed in the FOSIBLE project now runs as an executable binary on a miniature PC. This allows an easy integration of the device into the furniture. The next section reports on tasks and results performed in that respect.

5.1.2 Implementation

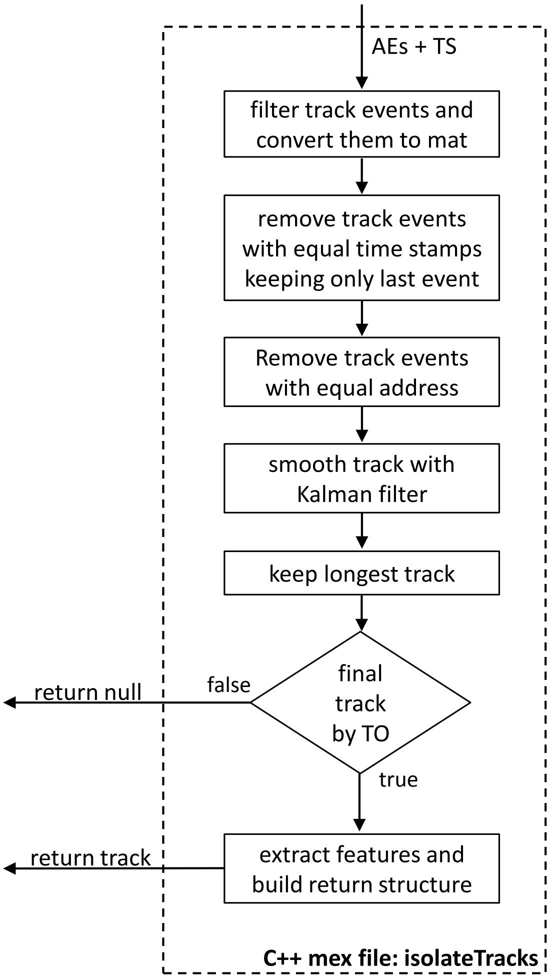
<table>
<tr><th>Tasks</th><th>Result</th></tr>
<tr><td>Transfer code to command line tool</td><td>Tool can be compiled with MALTAB Compiler and started from the command line.

Parameters:
<table>
<tr><td>mode</td><td>There are two different modes: in the test mode the tracks are displayed for debug purpose (slows the down the application); in the app mode this output is disabled. The mode can be switched over the GUI.</td></tr>
<tr><td>ipu</td><td>IP address of UCOS</td></tr>
<tr><td>ips</td><td>IP address of remote device (send to address)</td></tr>
<tr><td>port</td><td>UDP port of remote device</td></tr>
</table>
PROTOTYPE: fosibleDemoTool.exe mode ipu [ips port]</td></tr>
<tr><td>UDP handler communication</td><td>UDP message format implemented :

<SVSGESTURE>
<SENSOR>name</SENSOR>
<GESTURE>up</GESTURE>
<SN>0x00000001</SN>
</SVSGESTURE></td></tr>
<tr><td>Implementation of UDP interface</td><td>Interface implemented and tested</td></tr>
<tr><td>Demo test</td><td>Demo works for the four gestures: left, right, up and down. If it is not possible to distinguish a gesture it is invalid.</td></tr>
<tr><td>Performance optimization</td><td>Performance has been improved by porting the MATLAB code to C++. For a detailed description of the cmex file see section below.</td></tr>
<tr><td>Sensor</td><td>- camera focus
- shiftx and shifty parameter of stereo algorithm
- Peoplecount application:
  o 2 scan lines → set from the middle of the sensor pane to its edges. Tracks are interrupted if they cross one of the count lines.
  o grid size → changed from 16 to 32 (smoother tracks)</td></tr>
<tr><td>Test on a EEE box pc</td><td>Compiled MATLAB file runs on EEE box pc without additional latency</td></tr>
<tr><td>Visualization GUI</td><td>Indicates direction of track by highlighting start (red) and end point (blue) of the track.</td></tr>
<tr><td>Test tool</td><td>Specification: Server application (JAVA) that receives the UDP messages sent by the demo tool and visualizes the results of the algorithm.</td></tr>
</table>

```
while(programIsRunning)
      bin = fetchAEs();
      A = isolateTracks(bin);
      if(~isempty(A))
            recogniceGesture(A);
      end
end
```

Fields of track structure	
x	x-coordinates of events
y	y-coordinates of events
d	disaprity values of events
tt	time stamps
box	bounding box [x,y,w,h]
angle	orientation [-90 … +90][1]
ecc	Eccentricity [0…1][2]
v	velocity [vx,vy]



1) Orientation is derived by calculating the covariance matrix of the track. The eigenvalues of this matrix correspond to the major and minor axes of the track, so the orientation can thus be extracted from the angle of the eigenvector associated with the largest eigenvalue.

2) The relative difference in magnitude of the eigenvalues is an indication of the eccentricity of the track, or how elongated it is. If the eccentricity is one the track resembles a line.

Figure 7. Algorithm flow diagram and data types.

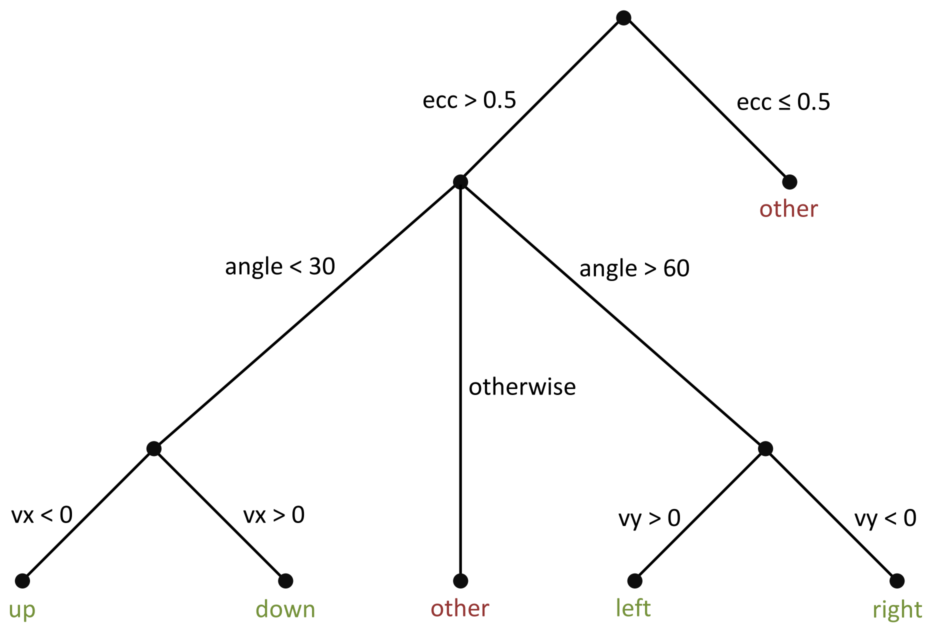


Figure 8. Implemented gesture recognition decision tree.

5.2 Monitoring of “social state”

Two UCOS for person detection mounted on ceilings in living room and in kitchen: each device monitors an area of approx. 3 x 3 meters on the floor and detects the number of persons in the room. These data are sent from UCOS via TCP/UDP to the Mauser runtime hardware which in turn sends it to the data broker.

The two devices are installed in the living room to monitor the area in front of the TV and in the kitchen to monitor the area in from of the smart PC.

Mounting structures will be provided by the inHaus to mount the devices in a correct height of 3m to 4m above the floor level. The inHaus also provides and installs CAT5E ethernet cabling from a central point (furniture behind TV set) to the two mount positions. AIT provides three Power over Ethernet (PoE) converters and a PoE switch to supply the UCOS devices.

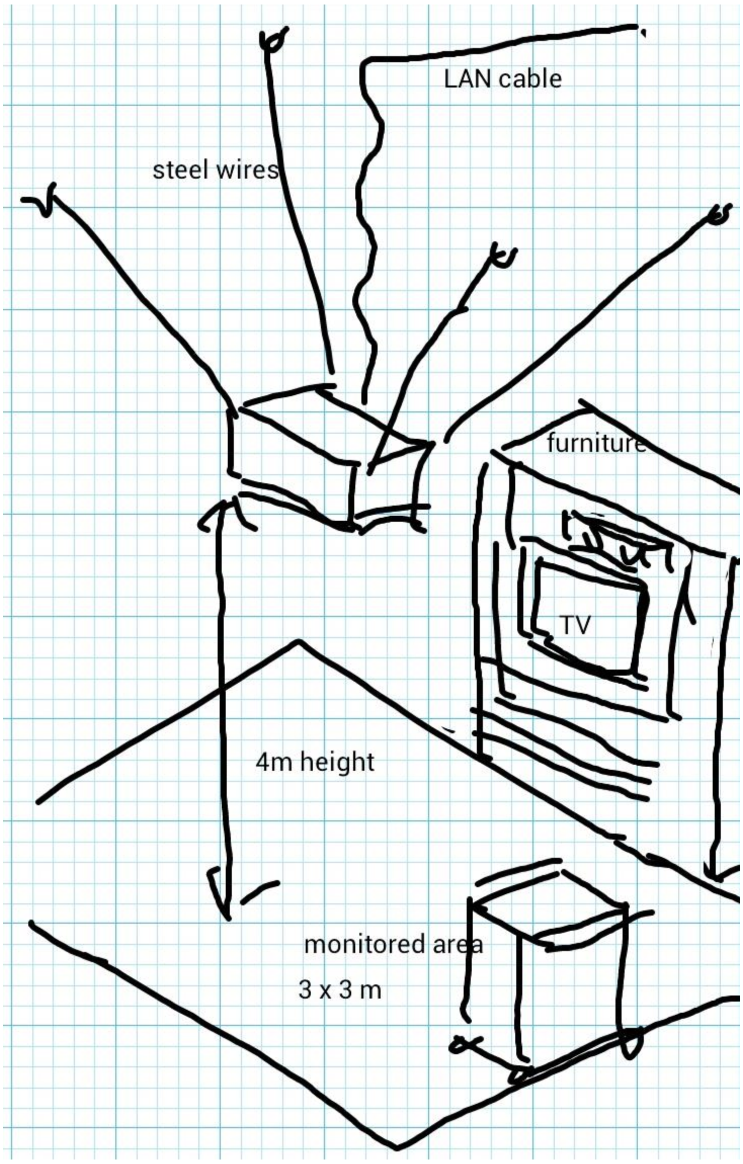


Figure 9. Sensor arrangement for function people counting in room.

The figure below shows the positions of sensor mounting and other components in the inHaus service apartment. A PoE router and a miniature PC will be integrated in the furniture that holds the TV set.

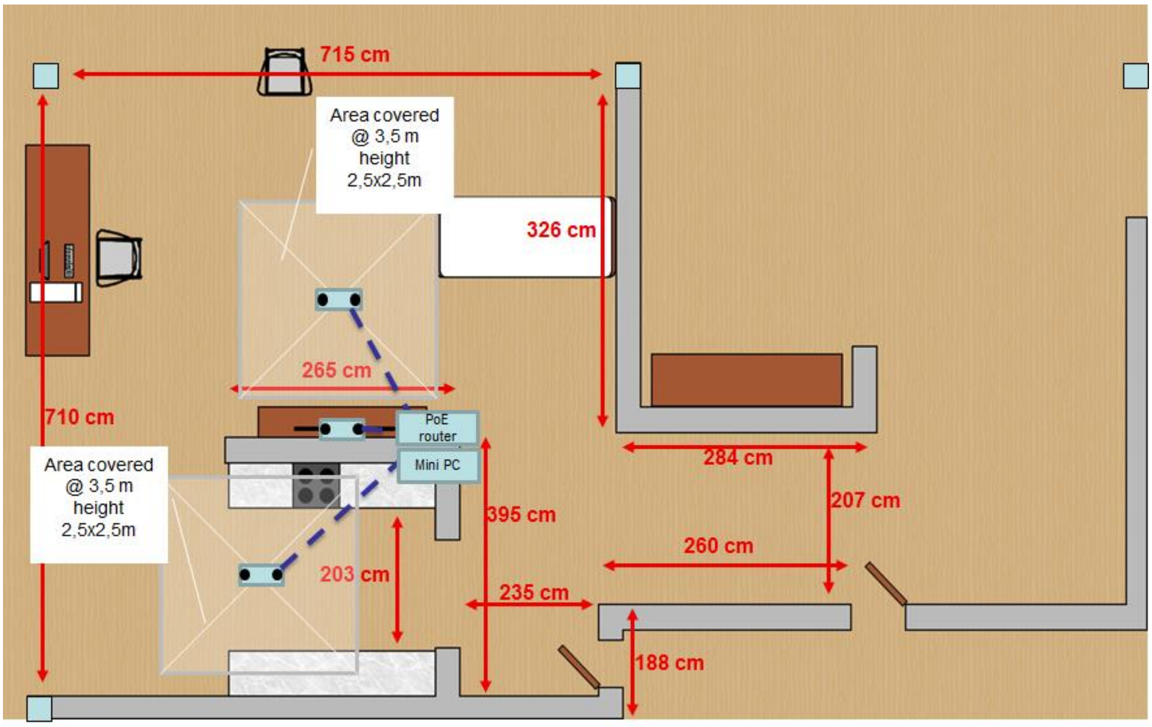


Figure 10. Sensor mounting for people counting in the inHaus service apartment.

5.3 Additional sensors

The UCOS sensors are part of a sensor setting that make up the overall functionality of the inHaus service apartment. These sensors are used for the applications in D3.5. The devices are the described UCOS sensors.

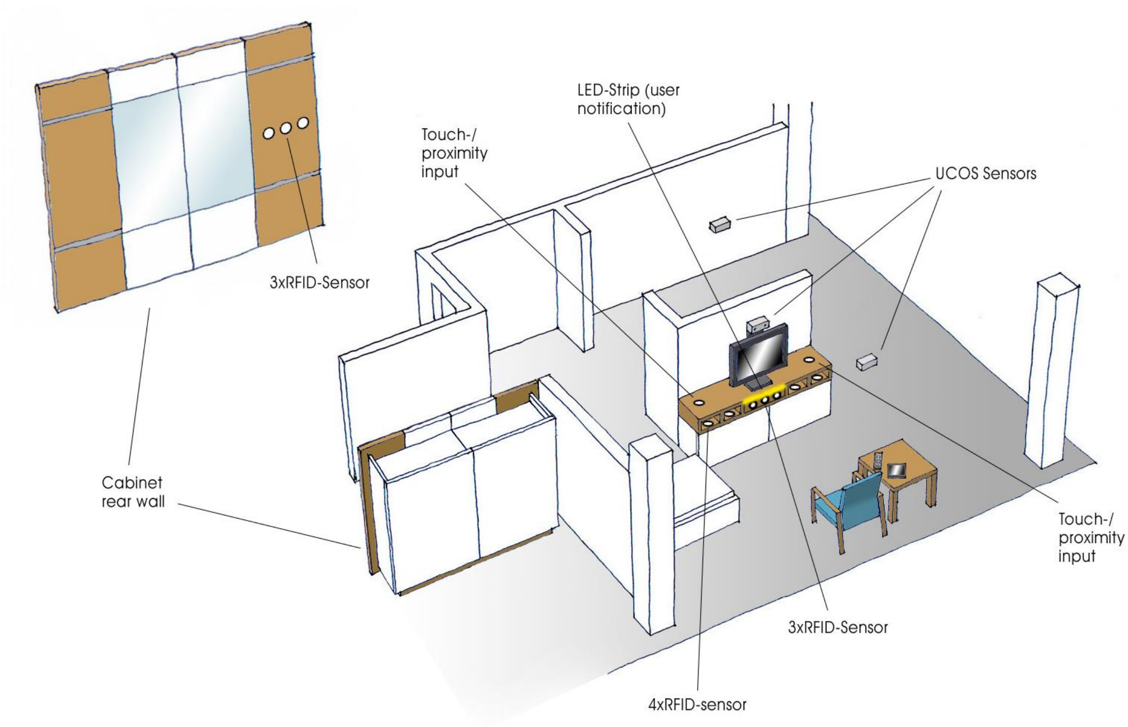


Figure 11: Sensors used in inHaus Service Apartment

The combination of these various sensors makes it possible that different sensor scenarios can be implemented.

6 Sensor Scenarios

The UCOS sensors are presented in deliverable 3.5 with different applications. In that deliverable, these applications are described in more detail. However, there are some scenarios that are important for the deliverable 3.6. One of them is the use of gesture recognition to trigger shortcuts. One useful example is using these gestural shortcuts to open the FoSIBLE widget, to hide it (e.g. for watching TV without closing the widget) and to make it visible again. Another example is to close pop-up windows through gestural shortcuts. Doing so, the UCOS sensor is used to perform a quick interaction with the dialogue on the iTV, so that requests from the sensor environment can be done without using the remote control.

Moreover, the people counting technology is used for the correct calculation of the activity index and also for presenting an social status in the widget that contains information about the availability instead of just displaying if the person is online or not.

Figure **12** shows different contexts in which the UCOS sensors are used for different interaction tasks.

Sensor /Actor	TV	Widget	Community	Ambient	Person Counting	Gesture	Action
Gesture	-	off	incoming message, TV recommendation	Ambient Light on	-	up-down	open widget, Ambient light off
Gesture	Stand-by	off	TV-Reminder	Ambient Light on	-	up-down	TV on, Channel, Ambient light off
Gesture	on	off	TV-Reminder	Ambient Light on	-	up-down	Channel, Ambient light off
Gesture	-	off	-	Ambient Light on	-	right-left	Ambient light off
Gesture	on	on, visible, no popup	-	-	-	right-left	Widget invisible
Gesture	on	on,	-	-	-	up-down	Widget

		not visible					visible
Gesture	on	on, visible, no popup	-	-	-	right-left	close popup
Person Counting	-	-	-	-	>1	-	Activity Index off / consider no. of person
Person Counting	on	on	-	not at chair	=0		state "not available"
Person Counting	on	on	-	at chair	=1		state "available"
Person Counting	on	on	-	-	>1		state "in company"

Figure 12: Relationships of the sensors

7 Data communication and protocols

7.1 UDP communication

All communication of the hardware components use TCP/UDP connections that can simply be realized in any software component used in the project. The data will be transferred via wired standard Ethernet cabling. The sensors will send their observations gestures or people counts to the Mauser runtime hardware, which in turn controls smart living functions or the TV set via a R/C emulator.

7.2 Gesture recognition datagrams

The gesture datagrams are plain text XML data that codes the sensor that was sending the data as well as the gesture. A packet serial number is sent to be able to detect the unlikely event of data packet loss.

```
<SVSGESTURE>
<SENSOR>name</SENSOR>
<GESTURE>up</GESTURE>
<SN>0x00000001</SN>
</SVSGESTURE>
```

The meaning of the XML fields is as follows:

- <GESTURE> {up, down, left, right}
- <SN> Packet serial number in HEX format

7.3 People count datagrams

The example below shows a XML datagram that is sent from the UCOS sensor to the Mauser runtime hardware.


```
<SVSAREACOUNT>
<SENSOR> Name</SENSOR>
<ORT> Location </ORT>
<COUNT>0.5</COUNT>
<BOARD_TIME>20000101 01:00:17.000</BOARD_TIME>
</SVSAREACOUNT>
```


It is important to know that the Count value is a weighted average of people counts over time. A value 1 means that in average there was one person present over the observation period. A value >1 means in average there was more than one person present over the observation period. Therefore for persons being alone in the apartment a value <1 is expected whereas presence of more than one person is signalled by a value >1.