DELIVERABLE No.2.2

Low-fidelity prototypes and mock-ups for perceptual crossing

Author(s):	**Patrizia Marti, Jelle Stienstra, Ernesto Di Iorio, Michele Tittarelli, Iolanda Iacono**
Project no:	**287624**
Project acronym:	**ACCOMPANY**
Project title:	**Acceptable robotiCs COMPanions for AgeiNg Years**

Doc. Status: Final Draft
Version: 0.2
Actual date of delivery: 22 November 2013
Contractual date of delivery: Month 24
Project start date: 01/10/2011
Project duration: 36 months
Peer reviewer: IPA

Project Acronym: ACCOMPANY
Project Title: **Acceptable robotiCs COMPanions for AgeiNg Years**

EUROPEAN COMMISSION, FP7-ICT-2011-07, 7th FRAMEWORK PROGRAMME
ICT Call 7 - Objective 5.4 for Ageing & Wellbeing

Grant Agreement Number: 287624

DOCUMENT HISTORY

Version	Date	Status	Changes	Author(s)
0.1 0.2	2013-10-10 2013-11-15	Draft Final	Initial Draft	Patrizia Marti Jelle Stienstra Ernesto Di Iorio Michele Tittarelli Iolanda Iacono

AUTHORS & CONTRIBUTERS

Partner Acronym	Partner Full Name	Person
UNISI	University of Siena	Patrizia Marti
Contributors: UNISI	University of Siena	Jelle Stienstra Patrizia Marti Ernesto Di Iorio Michele Tittarelli Iolanda Iacono

Short description

This document describes the work carried out in WP2, Task 2.2: Perceptual Crossing for interaction design.

It is articulated in two main parts. Part A presents the theoretical background of perceptual crossing, how the framework guided the research in ACCOMPANY, our objectives in pushing the state of art, and four scenarios that bridge the theory and find application within the project.

Part B contains the actual implementation of the four scenarios and a preliminary evaluation carried out to fine tune the scenario implementation.

Table of Contents

1. PART A.

Exploring Perceptual-Crossing to Empower Empathy

1.1 Introduction

T2.2 explores the possibility of achieving a shared perception with the robot in order to empower empathic relations and to enrich the experience of use as an emergent and dynamic outcome of the interaction. In exploring shared perception with the robot, the concept of "perceptual crossing" is taken as a main source of inspiration for design. Perceptual crossing is the recognition of an object of interaction which involves the perception of how the behaviour of the object and its perception relate to our own. Examples of perceptual crossing in human-human interaction occur when two people catch each others' eye, in case of mutual touch, kinaesthetic or acoustic interactions (proto-conversation with babies, dialogue, choral singing etc.), resonance of movements.

This document explores perceptual crossing from different viewpoints. A theoretical background is provided in the initial part of the document to frame the concept of perceptual crossing and to investigate the possibilities of exploring it in the field of human-robot interaction. Four scenarios of perceptual crossing have been designed in order to experiment with the concept, feed the theory and provide input for the integration in Care-O-bot.

The second part of the deliverable contains a description of the scenario implementation in form of low-fidelity prototypes and multimedia materials (video probes). The prototypes were realised using the Magabot platform since Care-O-bot is not available at the University of Siena, and the simulation environment cannot be used to test shared perception. In fact, in order to test perceptual crossing, two physical agents are necessary, in particular to synchronise their movements.

Even if the Magabot platform is very limited and equipped with few sensors, the scenarios were successfully implemented and tried out. The value of this implementation does not rely on the software developed, but on the insights we got to feed the theory and provide input about the scenarios that can be implemented, at a far richer level, on Care-O-bot.

A second reason for using Magabot is related to the fact that the same platform is being used by University of Twente to run user evaluation on the acceptability of the robot. Having the same platform will help to support this evaluation including perceptual crossing scenarios.

1.2 Theoretical Background

Social intelligence and interactivity is crucial for a robotic companion to interact with elderly people repeatedly and long-term because they help to create a positive relationship between the person and robot.

Perceptual-crossing as coined by Lenay and Auvray (Auvray, Lenay, & Stewart, 2009) (Lenay, 2010) and further explored by Marti (Marti, 2010) and Deckers (Deckers et al. 2011; 2012; 2012; 2013) in the field of interaction design boils down to the phenomenon in which two people look at each other and have their eyes meeting, a moment where I see you seeing me. This crossing can occur on the visual as well as other modalities such as touching. On itself this can be intensely experienced yet it does not convey functionality as such. Therefore perceptual-crossing is mainly applied to initiate ‘conversation’ as it does provide grounds for a shared context. Perceptual-crossing is often extended with the phenomenon in which a person sees where the other person is looking at. For example, I see in your eyes that things occur behind me. Here, the spatiality expands the context though does not intentionalize yet.

Based on the Interaction Frogger Framework by Wensveen, Stienstra et al. (Stienstra, Bruns Alonso, Wensveen & Kuenen, 2012) proposed interactive materiality as a behaviour-changing paradigm that applies direct action-perception loops between person and artefact to engage in interaction. Artefact’s as such can perceive as well as act, this can be in the visual field or in the tangible. Through providing the artefact with behaviour in the interaction, behaviours of the person can be persuaded. This relates to perceptual-crossing in a way that the person’s actions are mapped to the actions of the artefact in a natural expressive way, providing ground for an engaging relationship.

The research on perceptual crossing in the psychological field of enactive approach to social cognition (Iizuka et al., 2009; Iizuka & Di Paolo, 2007; Di Paolo et al., 2008; De Jaegher, 2009; Froese & Di Paolo 2010) focuses on one and two dimensional simulation models in which uncertainties are avoided. In these simulation models, the moving behaviour of a single actor is simulated. Users are validating whether perceptual-crossing with these behaviours can be achieved over different dimensions such as speed, direction, geometry of movement and modality. The simulations are focused on defining the limitations of non-holistic systems.
Auvray et al. (2008) carried out experiments where subjects were able to distinguish animate objects from inanimate ones with the same appearance and movement only by perceiving very simple tactile stimuli. Empirical evidence has been found in their

experiments to sustain the central role of dynamic mutuality and shared intentionality in forming several aspects of an ongoing interaction.
A fundamental insight we can draw from these experiments for the design of empathic interaction is that an important clue in interaction is its interwoven nature which has to be shared between the subjects.

In the field of design, perceptual crossing is explored within the research-through-design approach that intends to feed research through the creation of physical hypotheses in actual context, not organized lab settings.

Paramount in achieving this perceptual-crossing is continuous interaction and that occurs in time preferably without latency between the 'action and reaction', this in order to have the parties engage in the same context. We thus aim for a mergence between action and reaction as in interactive materiality (Stienstra et al. 2012). Contrary to static symbolism, interactive materiality concerns active and perceptive 'material' capable of interaction that is designed to persuade for behaviour change. Deckers et al. (2012) distinguish several design notions that allow designers (in the broadest sense of the word) to achieve the crossing. The first one "Focus the Senses", is a requirement for the sensing system: it should be able to distinguish the person and able to focus. The sensing system should have a certain "Subtleness of Movement", it should be able to express and further "Show Explorative Behaviour" and "Recognize Explorative Behaviour Subject". This will make the system appear is subjective being. In the course of being, the system should be sensitive to "React to External Event" even towards "Reflect Contextual Noise". "Remember and Anticipate Perception over Time" provides ground for building a common history and anticipate on the behaviour of the person.
These notions, applied in design cases by students of Eindhoven University of Technology (Deckers et al. 2011) explore the nuances and modalities in which behaviour and the feeling of perceptual-crossing can be achieved (visual, tactile and even blowing). Yet the design work and principles by Deckers et al. are mainly focused on person artefact interaction and lack contextual integration.

Perceptual-crossing dominantly explores subject – object relations (transforming the object to have subjective qualities), yet third party (the context of action, the intentions of both subjects towards the environment) plays little role in here. We see an opportunity in pushing the *perceptual-crossing paradigm* to a *contextual resonance* paradigm in which two entities (person and artefact) go beyond an awareness of sharing context and arrive at the sharing of intentions (understanding ánd acting upon) within. Our work extends the research on perceptual-crossing with functionality (in our case the increase of empathy through togetherness), the integration of contextuality and multiplicity of intentions. In other words, we contextualize the paradigm and allow the subject and object to assess and act upon the intentions that are of functional value in the context.

The paradigms of perceptual crossing seem to have similarities with synchronization, yet we would like to stress that perceptual-crossing is not about this. A following behaviour will create a 'mirror', but we could say that the figure in the mirror is not someone else, not another subjectivity to engage with. We aim to achieve that the elderly person becomes immersed in an engagement with another subjective being: the Care-O-bot. Not to be a mirror, but an entity to empathise with.

The work within this project applies principles of perceptual crossing to achieve a mutual understanding, for both entities (robot and person) to immerse in a shared context and to 'grope' the intentions of the other. The 'groping' and the acting upon in our paradigm go hand in hand. (Happen at same time in the active-perception-loop). This work explores the reciprocal character and seeks further inspiration in Schutz's (Schutz, 1967) phenomenology of the social world. His work prominently hints at understanding the others actions and intentions within interaction.

2 Design Cases

Four interaction designs were developed to explore several layers in achieving perceptual-crossing. All of them intend to empower an empathic relationship between the person and robot but differ in their contextual resonation with the person.
The scenarios are described from the person's perspective. This description is followed by a script depicting how each scenario is achieved technically from the robot's perspective.
Since the research has been carried out in the context of research-through-design, the scenarios are elaborated back to the theory introduced in the background, to generate new knowledge and designs.

2.1 Interaction Designs and scenarios

As anticipated in the introduction, the following section presents four scenarios of perceptual crossing between the older person and Care-O-bot.

I-see-you-seeing-me.

The first interaction design establishes a bound between person and robot. Pivoting around a single point the Care-O-bot will follow (through looking and pivoting on its location) the elderly person when the latter is facing the Care-O-bot. When the elderly person does not face the robot, no attention (thus following) will be given.

I just woke, time for a coffee. Let me move from the bedroom to the kitchen. Ow it is still dark in the living room. I will open the curtains first. Did I just pass the companion robot? Hmm maybe he still sleeps, ahh well let me first walk to the curtain and give myself some sunlight... Yes the Care-O-bot is awake; as I turned around to make a move the kitchen he swiftly moved his face in my direction. On my way to the kitchen he keeps looking at me. He looks in my eyes. Ahh he stopped moving now I turned my back. (the robot stops moving and therefore does not make sounds)

Walk-with-me.

In the second interaction design the robot does not pivot around a single point. The Care-O-Bot *accompanies* the person from A to B.

My coffee can use some sugar and perhaps some milk. Care-O-bot won't be able to grab the little things but might help me carrying it to the coffee table. Ahh the tablet already shows that we can add these. Lets proceed. Ok I will have to go to the kitchen myself but the Care-O-Bot will accompany me there and carry the ingredients back to the living. Let me stand up and go to kitchen. Nice, Care-O-bot walks with me. He did not already race to the kitchen; he joins me in the walk. Keeping my pace. I am not alone.

Lets-move.

In this interaction design the Care-O-bot will anticipate or provoke action from the person. The Care-O-bot that somehow blocks the way for the person but when the person initiates to get out the robot will make way. This initiation of movement might also lie with the robot.

I need to get up; I just want to get some air. No I do not need your help robot, let me just do my things. Get out of here, you keep blocking my way. (Moving out of chair) Ahh you move aside thanks. It is as if you knew I wanted to step out.

Lets-cross-paths.

Moving together from A to B might not always be exactly what is needed, the fourth interaction design mediates in the interaction how the person and robot switch sides along the route from A to B.

Thanks for walking with me but hey I really have to get to the fridge and if you keep riding this way I won't get there! (Slows down the pace) Thanks for going a bit faster, now I can step behind you. ahh you already knew we were to switch sides?

2.2 Interaction Design implementation

As said before, a direct action-perception loop is paramount to achieve a perceptual-crossing. The interaction design shall not have the Care-O-bot moving in a pre-programmed way but attune its pace and movement to the pace and movement of the person. Following a direct mapping between the action of the person and the action of the robot, expressive qualities in the movement are consolidated. If the person moves fast, the robot does so accordingly (yet, not synchronized but with its own character).

Assuming that the sensing and actuating capabilities of the elderly person are understood, in the following we describe the sensing and actuating capabilities of the robot (enriched by the environment sensors) required for the four scenarios.
Furthermore, for each scenario we elaborate the behavioural / mappings.

The robot is to move in forward and backward direction and turn around. Two wheels on the side of the robot that can be controlled independently achieve this. For the Care-O-bot it is important to see the position of the older person.
For the first scenario, a gaze detection of the elderly person would be optimal but the shoulder direction could be a solid representation. Gaze detection can be easily achieved with Care-O-bot using the face detection module developed in WP4.
For the second and fourth scenario the moving location of the elderly person is required. These can be measured with body posture detection available in the Care-O-bots environment.
The third scenario builds upon the detection whether the person stands up or not. The Care-O-bot thus utilizes the internal senses (camera in the robot itself) and external senses (ceiling camera) to comply the fused knowledge of where the person is and what he/she is doing. As mentioned before, latency should be avoided. This is still an issue in the sensing.
As said in the introduction, the scenarios are experimented at the University of Siena, using the Magabot platform. This platform mounts a shell similar to Care-O-bot (Fig. 1).
The Magabot platform and the shell are equipped with five distance sensors on the front and three distance sensors on each side to detect the person walking in front or next to the robot.

Figure 1. Magabot equipped with a shell similar to Care-O-bot

In the I-see-you-seeing-me-scenario (1), the directional movement of body and torso of the Care-O-bot are mapped to the directional movement of the person. When the person faces the robot, the robot will direct to the moving person. When the person moves without being directed to the robot, the robot will be less sensitive in moving.

In the walk-with-me (2) scenario, the Care-O-bot navigates on a straight line from A to B. From A to B, the person next to him is detected and the speed of moving along the line is adjusted. Here reciprocal movement is applied in first instance as a different behaviour persuades the elderly person to cross sides as intended in the 4th scenario.

The let's-move (3) scenario again is a mere pivot around a point, yet the movement of the robot is backward and turning. The track is predefined, but the expressive movement is mapped to the movement of the person getting up from the chair. The challenge here is to avoid pre-programming of discrete action-reaction (standing up = move), and let the movement emergence in interaction from the actions of the person (i.e. standing up or showing intention to do so, leads to slight movement of the robot). The last scenario has two lines on which the Care-O-bot can move A to B and A' to B' which are parallel. The speed of moving is adjusted in the interaction and when appropriate the robot can move from one to the other track switching sides with the elderly person.

3 Back to theory and reframing

The first scenario (1) consists of a basic Perceptual-Crossing intended to achieve a crossing and initial engagement. This interaction makes the person feel her being seen by something that has subjectivity as well.

The walk-with-me (2) builds upon that in such a way that intentions of moving is shared and that the intentions are done together. The person and robot both want to get from A to B, but they do this together. We aim at togetherness, to achieve an empathic relation.

In 'lets-move' (3), we explore the boundaries of proactive behaviour in which the robot insists that the person drinks by standing close, interacting physically with the cup and person. In order to change/give the person an intention. Both robot and person can provoke and anticipate on the other. The situation focuses on understanding and acting upon behaviour of one another IN the interaction.

The crossing sides (4) push Perceptual-Crossing towards Contextual Resonance merging two intentions (both entities going from A to B and switching sides). We call this contextual because the objective to reach B or B' is contextual (the robot needs to get to the fridge or the person). The explorations further involve obstacles (context) on the route from A to B forcing person and robot to walk behind each other while the normal intention is to move forward side by side in a togetherness described in the walk-with-me. This comprises a situation in which the person OR the robot should let the other pass. In the interaction, in the movement by person and robot, a shared context shall emerge as robot and person target to 'grope' the intentions of the other (in the action-perception loop). The 'groping' and the acting upon in our paradigm go hand in hand building upon the reciprocal character.

4. PART B
Scenario implementation

4.1 Implementation and testing

In the following, the scenarios are explained on a more technical level: the target, idea followed by interim video's that show one of the iteration implementations with some remarks that guide the update towards the final design.

For each scenario, we added also a paragraph titled: Expert evaluation. This means that the working prototype implemented on Magabot, was evaluated by a person from the University of Siena team who did not take part to the implementation of the scenarios. This person commented the interim videos as well as the final one, and provided recommendation from improvements. Other two experts from the UNISI team actually tried out the prototype and commented the interaction.

The final implementation of the scenarios can be found at:

https://vimeo.com/79409786

Scenario I - I see you

TARGET: * Basic Interaction Mode: I-see-you-seeing-me.
Directional movement of body and torso of the Care-O-bot mapped to the directional movement of the person (When the person faces the robot, the robot will direct to the moving person. When the person is moving without being directed to the robot, the robot will be reactive in moving, on a continuous scale, obviously).

IDEA: We used 5 front sonars of Magabot to trace the presence of a user in the range [0 - 100 cm]. The robot rotates in the direction in which the distance (detected by sonar i-th) between the robot and user is the smallest. In other words where the user is close to the robot, the robot will turn to. Of course, this implementation only gives the "impression" of shared gaze, since the Magabot is not equipped with webcams to detect the person. If the scenario will be implemented in Care-O-bot, it would be ideal to use the face detection module developed in WP4.

INTERIM VIDEO REPORT: http://youtu.be/DAIfW4NQqJ8

EXPERT EVALUATION: The movement from one side to the other does not use a pre-set speed: the speed is applied on a continuous scale. However, when the person faces the robot, it should turn more, to give the impression of a perceptual crossing. Unfortunately, a very subtle behaviour is not feasible with the sensors available on Magabot. The ideal implementation would require for example the system developed by UVA to measure the direction of the person.

RE-DESIGN: In order to mitigate the problem described above, a new implementation was developed to obtain a dynamic rotation speed

TECHNICAL REPORT:

Base protocol: Magabot.h, HighLevelProtocol.ino, LowLevelProtocol.ino [native libraries Magabot].

Control method: the robot is speed driven by setting/adjusting the high levels of the electrical voltages (eg. v1, v2). The control is independently applied to each of Magabot's motors.

Mapping Command input (the input can also run by serial monitor of Arduino IDE):

- 'a': anti-clockwise rotation (-v1 = v2);
- 'd': clockwise rotation (v1 = -v2);
- 'p': stop (v1 = v2 = 0).

Active sensors:

- the five sonar sensors placed on Magabot's front;
- the bumpers to stop Magabot if encounters an obstacle.

Parameters:

- pingMaxRange = 100 [cm], maximum distance between the robot and the user to be detected;
- velocity = 15, constant numeric voltage value to apply to the motor for the rotation.

The code flow:

Once it is activated, Arduino extracts from the system the data coming from the 5 sonars at the maximum clock allowed (input '1 'from the serial monitor of Arduino IDE). The system calculates the user's movement according to sensor that is registering minimum value, if the user is located within the scanning radius (pingMaxRange).

Magabot turns anti-clockwise if the minimum value is scanned from the two sonars on the left side. It turns in clockwise rotation if the minimum value comes from the two sonars on the right side. The robot stops if the minimum value refers to the sensor placed on the centre.

The algorithm stops with the 'p' command launched by the serial monitor of the Arduino IDE.

The speed of rotation is dynamic: faster if the user is detected by the external sensors (1 and 5), slower if the user is detected by the internal sensors (2 and 4).

It has to be noted that, the Magabot platform only mounts two Arduino and a very limited number of sensors. In order to implement the gaze sharing, a completely different architecture and platform would be necessary. However, the objective of task 2.2 was to develop a low-fidelity prototype to proof the perceptual crossing concept and experiment with a continuous interaction and a dynamic robot behaviour adaptation.
As explained above, integrating this scenario in Care-O-bot would require to use the face detection module developed in WP4, so to have a realistic sharing of the gaze between the person and the robot.

Scenario II (a) - Walk with me (robot chaser)

TARGET: * Moving Interaction Mode: Walk-with-me.
The Care-O-bot will accompany the person from A to B. It shall not be moving in a pre-programmed way but attune its pace and movement to the pace and movement of the person.

IDEA: the robot is programmed to stay at a distance equal to or less than 1 meter behind the user. If the difference between the distance detected from sonars at time t+1 and the distance detected at time t is greater than the minimum distance set, the robot performs forward movements.

INTERIM VIDEO REPORT (robot chaser):
https://www.youtube.com/watch?v=zvamTwiRjTY

EXPERT EVALUATION: The movement of the robot smoothly appears when the person starts moving in front of the robot. The speed seems to progress on a continuous scale. However it should be avoided that the person looks like holding his speed and tries not to walk too far ahead. In order to avoid the impression that the person adjusts his movements to the robot's ones, the robot should actually speed up considerably.

RE-DESIGN: The system has been redesigned to be more sensitive to the user's movements, by changing interaction range and clock.

TECHNICAL REPORT:
Base protocol: Magabot.h, HighLevelProtocol.ino, LowLevelProtocol.ino [native libraries Magabot].
Control method: the robot is speed driven by setting/adjusting the high levels of the electrical voltages (eg. v1, v2). The control is independently applied to each of Magabot's motors.

Mapping Command input (the input can also run by serial monitor of Arduino IDE):

- 'w': forward translation (v1 = v2);

- 's': backward translation (-v1 = -v2);
- 'p': stop (v1 = v2 = 0).

Active sensors:
- central sonar placed on Magabot's front;
- the bumpers to stop Magabot if encounters an obstacle

Parameters:
- systemClock = 500 [milliseconds], clock time to repeat the reading from the sonar;
- thresholdMovingToStop = 280 [cm], maximum distance from which the user is detected and followed by Magabot;
- thresholdStopToMoving = 80 [cm], minimum distance from which the user is detected and followed by Magabot;
- thresholdBackwarToStop = 30 [cm], distance from the user under which Magabot moves back.

The code flow:
Once is activated, Arduino extracts from the system, data from 5 sonars at the maximum clock allowed (input '1 'from the serial monitor of Arduino IDE). The robot moves at a proportional speed according to the distance between robot and user, if user is within the range [thresholdStopToMoving, thresholdMovingToStop]. The robot moves back at a constant speed if the user is detected within the range [0, thresholdBackwarToStop]. The robot stops if the user is detected within the range [thresholdBackwarToStop, thresholdStopToMoving]. The robot also stops if the user is not detected within the maximum range (>= thresholdMovingToStop). The algorithm stops with the 'p' command launched by the serial monitor of the Arduino IDE.

Scenario II (b) - Walk with me (robot predecessor)

TARGET: * Moving Interaction Mode: Walk-with-me.
The Care-O-bot will accompany the person from A to B. It shall not be moving in a pre-programmed way but attune its pace and movement to the pace and movement of the person.

IDEA: the robot is programmed to stay at a distance equal to or less than 1 meter ahead the user, as it is shown in the interim video. If the difference between the distance of sonars at time t+1 and the distance at time t is greater than the minimum distance set, the robot performs backward movements. The interim video below shows the interaction. This behaviour is not reported in the final video.

INTERIM VIDEO REPORT (robot predecessor):
https://www.youtube.com/watch?v=a0qpUGha4Gw

EXPERT EVALUATION: This is a difficult interaction that turns out pretty nice. Sometimes the user gets too close, and the robot should go a bit faster at times. In the end of the movie, the robot stops far from the user, this gives the idea that it wants to be far. However, this conflicts with what happened before, where the robot looks like to have the intention to be close to the person during the movement. So, in the scenario the robots seems to have the intention to be near though ahead of the person, but in the end it looks like to have the intention to be ahead and far.

There is a moment in which the robot stops the movement. This should be avoided unless there is no activity of the user. For this and the other scenarios, it would be desirable if the robot keeps moving. This will help the robot explore its space (see whether to respond to the user or not) and secondly, it will give the impression to be more ‘life’ and active.

RE-DESIGN: The system has been implemented to be more sensitive to user’s movements by changing interaction range and clock. Speed movement has been modified to be dynamic on a continuous scale.

TECHNICAL REPORT:

Base protocol: Magabot.h, HighLevelProtocol.ino, LowLevelProtocol.ino [native libraries Magabot].

Control method: the robot is speed driven by setting/adjusting the high levels of the electrical voltages (eg. v1, v2). The control is independently applied to each of Magabot’s motors.

Mapping Command input (the input can also run by serial monitor of Arduino IDE):

- ‘s’: backward translation (-v1 = -v2);
- ‘p’: stop (v1 = v2 = 0).

Active sensors:

- central sonar placed on Magabot’s front;
- the bumpers to stop Magabot if encounters an obstacle

Parameters:

- systemClock = 500 [milliseconds], clock time to repeat the reading of sonar;
- thresholdBackwarToStop = 180 [cm], maximum distance from which the user is detected and preceded by Magabot.

The code flow:

Once is activated, Arduino extracts from the system, data from 5 sonars at the maximum clock allowed (input '1 'from the serial monitor of Arduino IDE).

The robot moves backward at a constant speed proportional to the distance detected, if the user is detected within the range of measurement [0, thresholdBackwarToStop].

The robot stops if the user is not detected within the maximum range of measurement (>= thresholdBackwarToStop)

The algorithm stops with 'p' command launched by the serial monitor of the Arduino IDE.

Scenario II (c) - Walk with me (robot next to me)

TARGET: * Moving Interaction Mode: Walk-with-me.
The Care-O-bot will accompany the person from A to B. It shall not be moving in a pre-programmed way but attune its pace and movement to the pace and movement of the person.

IDEA: the robot uses the proximity sensors located on the shell, so it can walk next to the user. The robot stops when the user stops walking.

INTERIM VIDEO REPORT (robot next to me): http://youtu.be/xrs7Ujn2yf8

EXPERT EVALUATION: The video shows two or maybe three speeds of the robot according to keeping up with the user. The robot is supposed to change its speed, when the user does the same. That is not clear in the video. The robot brakes abruptly, as mentioned in the previous (Scenario IIb). A continuous movement should be applied even if it is very slow.

The robot should go faster, to avoid that the person attunes his speed to the robot. The robot should try to be a little bit more to the side of the person, to give the impression of ‘togetherness’.

RE-DESIGN: The speed of forward movement has been changed to be dynamic and incremental.

TECHNICAL REPORT:
Base Protocol #1 (on board): Magabot.h, HighLevelProtocol.ino, LowLevelProtocol.ino [native libraries Magabot].
Base Protocol #2 (of the Magabot Shell): UltrasoonBoard.ino [native code provided by Magabot]

Control method: the robot is speed driven by setting/adjusting the high levels of the electrical voltages (eg. v1, v2). The control is independently applied to each of Magabot’s motors.
The decisional logic is located on the Arduino # 2 (Magabot Shell), while the Arduino # 1 (on-board) receives high-level commands through the digital input pin 4 and 5.

Mapping Command input (the input can also run by serial monitor of Arduino IDE):

- ‘w’: forward translation (v1 = v2);
- ‘s’: backward translation (-v1 = -v2);
- ‘a’: anti-clockwise rotation (-v1 = v2);

- ‘d’: clockwise rotation (v1 = -v2);
- ‘p’: stop (v1 = v2 = 0).

As a quick note, the list of movement patterns and available sensors is repeated for all scenarios since each scenario uses a sub-set of them. This scenario contains the full set.

Active sensors:
- 3 proximity sensor placed on the Magabot Shell;
- the bumpers to stop the Magabot if it encounters an obstacle.

Parameters:
- USER_ACTIVATION_DISTANCE = 100 [cm], [Arduino #2], maximum valid distance between user and sensors;
- USER_MINIMUM_VALID_DISTANCE = 10 [cm], [Arduino #2], minimum valid distance between user and sensors;
- PING_INTERVAL = 66 [millisec], [Arduino #2], clock time between two acquisitions by proximity sensors of Magabot Shell.
- systemClock = 500 [millisec], [Arduino #1], time clock to repeat the reading of the digital pins 4 and 5.

The code flow:
The logic to acquire data from sensors and the decision-maker of motion is contained on the Arduino #2 (Magabot Shell). Arduino#1 (on board) receives the commands from Arduino # 2 and interprets them to control the Magabot’s motors. The communication between the two Arduino is allowed by wiring two digital output pins of the Arduino # 2 (pin 12 and pin 13) with two digital input pins of Arduino # 1 (pin 4 and pin 5).

Unfortunately, it is not possible to show the electrical circuit of the platform since the firm who provided the Magabot did not provide it. The following pictures are inserted to help understanding the communication between the two Arduino, using the digital pins available.

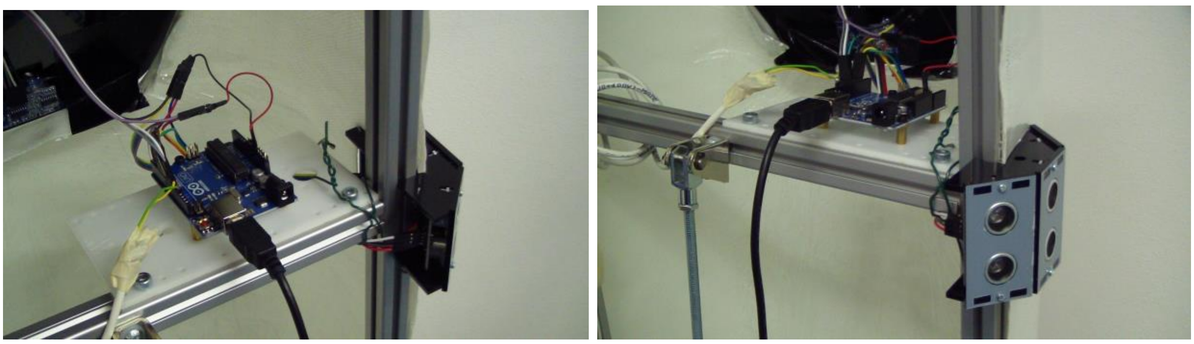

Figure 2. Arduino and sensors mounted on the shell

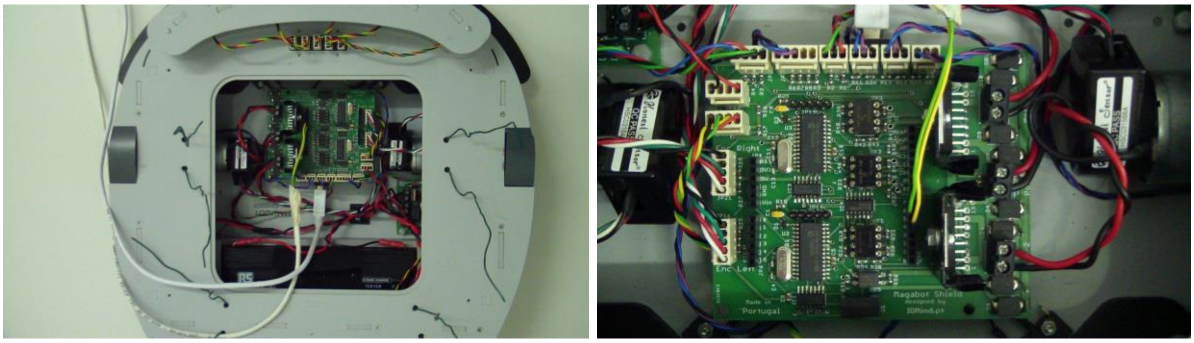

Figure 3. Magabot platform

The forward movement's speed is incremental: robot starts moving slowly, then it increases the speed until it reaches the user.
Command mapping (between Arduino #2 and Arduino #1) structure:

- (0, 0) -> 'p', stops the robot;
- (0, 1) -> 'w', robot forward movement at incremental speed;
- (1, 0) -> 's', robot backward movement.

Each digital pin available on the platform can take values 0 or 1. This is a tuple of digital signals that are sent from the two digital pins on Arduino mounted on the shell, to the two digital pins in input to the Arduino placed inside Magabot. The meaning of the tuples is explained above.

Scenario III - Lets move

TARGET: * Anticipating/Provoking Interaction Mode: Lets-move. (Understand and act upon behaviour IN the interaction).
In the scenario this can be applied in several places, including the situations described below as an example.
Situation A: in the come to and get part, the robot makes space by moving back so that the person has space to get up and start walking.
This behaviour should not be pre-programmed but it should emerge from the actions of the person (i.e. standing up or showing intention to do so).
Situation B: proactive attitude in which the robot insists that the person drinks by standing close and interacting physically with the cup and person, in order to change/give the person an intention. This triangle is an interesting example of robot mediation.
However, according to the theoretical framework we adopted in the project, it is preferable to focus on the situation A, as it is IN the interaction. The triangle can be a fifth as it does explore a shared contextually with a third artefact (while the other shares an intention).

IDEA: Based on Scenario II (c), the robot start actions when it detects the user standing up.

Before the user gets up, the robot causes the user's action by simulating "impatience" (forward and backward movements). When robot detects the user standing, it moves back to give way to the user and then it moves next to the user for walking together ("Walk with me" like).

TECHNICAL REPORT:

Base Protocol #1 (on board): Magabot.h, HighLevelProtocol.ino, LowLevelProtocol.ino [native libraries Magabot].
Base Protocol #2 (of the Magabot Shell): UltrasoonBoard_letsMove.ino [native code provided by Magabot]

Control method: the robot is speed driven by setting/adjusting the high levels of the electrical voltages (eg. v1, v2). The control is independently applied to each of Magabot's motors.
The decisional logic is located on the Arduino # 2 (Magabot Shell), while the Arduino # 1 (on-board) receives high-level commands through the digital input pin 4 and 5.

Mapping Command input (the input can also run by serial monitor of Arduino IDE):

- 'w': forward translation (v1 = v2);
- 's': backward translation (-v1 = -v2);
- 'p': stop (v1 = v2 = 0).

Active sensors:

- 3 proximity sensor placed on the Magabot Shell;
- 1 proximity sensor placed on Magabot Shell front;
- the bumpers to stop the Magabot if it encounters an obstacle.

Parameters:

- USER_ACTIVATION_DISTANCE = 100 [cm], [Arduino #2], maximum valid distance between user and sensors;
- USER_MINIMUM_VALID_DISTANCE = 10 [cm], [Arduino #2], minimum valid distance between user and sensors;
- PING_INTERVAL = 66 [millisec], [Arduino #2], clock time between two acquisitions by proximity sensors of Magabot Shell;
- systemClock = 500 [millisec], [Arduino #1], time clock to repeat the reading of the digital pins 4 and 5;
- DEFAULT_VELOCITY = 6, [Arduino #1], start velocity when robot move forward;
- MAX_VELOCITY = 16, [Arduino #1], maximum velocity when robot move forward.

The units of the velocities cannot be given. In fact the speed of Magabot depends on external factors like friction of the wheels, weight and battery level. Since the speed varies according to external factors, and therefore it cannot be anticipated, numerical values are used and transformed in volts by the system.

The code flow:
The logic to acquire data from sensors and the decision-maker of motion is contained on the Arduino #2 (Magabot Shell). Arduino#1 (on board) receives the commands from the Arduino # 2 and interprets them to control the Magabot's motors. The communication between the two Arduino is allowed by wiring two digital output pins of the Arduino # 2 (pin 12 and pin 13) with two digital input pins of Arduino # 1 (pin 4 and pin 5).
Before moving next to the user, the robot provokes an action when the user sits down (by continuous forward and backward movement).
When the robot walks next to the user, the forward movement's speed is incremental: the robot starts moving slowly, then it increases the speed until it reaches the user.
Command mapping (between Arduino #2 and Arduino #1) structure:

- (0, 0) -> 'p',stops the robot;
- (0, 1) -> 'w', robot forward movement at incremental speed;
- (1, 0) -> 's', robot backward movement;
- (1, 1) -> ibrid command, fired when user is in front of Magabot Shell central censor.

3.1 How prototyping feeds theory

Within the design process of the robots behaviour (the mapping between sensing and actuating), several iterations have been done to fine tune the prototype behaviour. Through this, each iteration guided by an expert evaluation fed the design process. Besides the sensitizing in the design process informing the interaction nuances; the scenario's themselves fed the theory.
Of course, we do not claim that our explorations and designs inform and transform the phenomenological backdrop, however our work does provide insights in how to design from the theory and make it practical.
The following points were apparent from the iterative prototyping process and should be taken in consideration:

(1) The few sensors and actuators chosen do not fulfil the expectations that the moving object (the robot) creates even though the robot is obviously stripped from its action possibilities.
(2) Even with a very limited platform, the prototype shows some intelligent behaviour that emerges in interaction through enabling the elderly person to grasp and act upon intentions of the robot and vice versa. Empathy through perceptual crossing and resonance in context does occur in our limited intention-crossing.

The early experiments described in this document can contribute to a later implementation in Care-O-bot, that is holistic and makes the robot actions meet the elderly person's expectations.
Further experiments of the prototype will be executed in year 3 of the project with the objective to select the scenarios that will be integrated in Care-O-bot. Standing the complex equipment and software modules that have been developed in the main

robot platform so far, we expect a richer interaction between the robot and the person where perceptual crossing plays a central role.

As a final remark, it is important to state that a main finding of the experiment is that the nuance in movement is paramount. In order to have a direct feedback loop and for the user to have a sense of direct experience (needed to make valuable judgements on the intentions of the robot) the interaction should occur in real-time on both sense/actuate as well as system architecture level. The pre-programmed computing will not allow for the interactions we achieved.

References

Auvray, M., Lenay, C., & Stewart, J. (2009). Perceptual interactions in a minimalist virtual environment. *New Ideas in Psychology*, 27(1), 32-47.

De Jaegher, H. (2009). Social understanding through direct perception? Yes, by interacting. *Conscious. Cogn*. 18, 535–542.

Deckers, E. J. L., Wensveen, S. A. G., Ahn, R. M. C., & Overbeeke, C. J. (2011). Designing for perceptual crossing to improve user involvement. In Proc. of *CHI 2011* (pp. 1929-1938). New York, NY: ACM Press.

Deckers, E. J. L., & Levy, P. D. (2012). Designing for perceptive qualities: 7 showcases. In Proc. of *DIS2012* (pp. 496-505). New York, NY: ACM Press.

Deckers, E.J.L., Levy, P.D., Wensveen, S.A.G. & Overbeeke, C.J. (2012). Designing for Perceptual Crossing: Applying and Evaluating Design Notions. *International Journal of Design*, 6(3), 41-55.

Deckers, E.J.L. (2013). *Perceptive Qualities in Systems of Interactive Products*. Doctoral Dissertation, Eindhoven University of Technology.

Di Paolo, E., Rohde, M., and Iizuka, H. (2008). Sensitivity to social contingency or stability of interaction? Modelling the dynamics of perceptual crossing. *New Ideas Psychol*. 26, 278–294.

Iizuka, H., and Di Paolo, E. (2007). “Minimal agency detection of embodied agents,” in Proc. of *ECAL2007*, 485–494.

Iizuka, H., Ando, H., and Maeda, T. (2009). “The anticipation of human behavior using ‘Parasitic Humanoid’,” in *Human-Computer Interaction. Ambient, Ubiquitous and Intelligent Interaction*, 284–293.

Froese, T., and Di Paolo, E. (2010). Modeling social interaction as perceptual crossing: an investigation into the dynamics of the interaction process. *Connect. Sci*. 22, 43–68.

Lenay, C. (2010). “It’s so touching”: Emotional value in distal contact. *International Journal of Design*, 4(2), 15-25.

Marti, P. (2010). Perceiving While Being Perceived. *International Journal of Design*, 4(2), 27-38.

Marti, P. & Stienstra, J.T. (2013). Exploring empathy in interaction: scenarios for respectful robotics. *GeroPsych: The Journal of Gerontopsychology and Geriatric Psychiatry*, *26*(2), 101-112..

Schutz, A. (1967). *The phenomenology of the social world*. Evanston, IL: Northwestern University Press.

Stienstra, J.T., Bruns Alonso, M., Wensveen, S.A.G., & Kuenen, C.D. (2012). How to Design for Transformation of Behavior through Interactive Materiality. In Proc. of *NordiCHI 2012* (pp.21-30). New York, NY: ACM Press