International Journal of Advanced Robotic Systems

OPEN ACCESS
ARTICLE

Point-and-Command Paradigm for Interaction with Assistive Robots

Regular Paper

Razvan Gabriel Boboc[1]*, Adrian Iulian Dumitru[1] and Csaba Antonya[1]

1 Transilvania University of Brasov, Brasov, Romania
*Corresponding author(s) E-mail: razvan_13_13@yahoo.com

Received 14 December 2014; Accepted 02 April 2015

DOI: 10.5772/60582

Abstract

This paper presents a multi-modal interface for the interaction between a human user and an assistive humanoid robot. The interaction is performed through gestural commands and a dialogue mechanism to provide a 'natural' means for the user to command an assistive robot to perform several given tasks. A "Point-and-Command" interaction concept was proposed with the aim of providing the user with an easy-to-use and intuitive interface and to improve the efficiency of human-robot collaboration. Likewise, a decision support system (DSS) was implemented based on fuzzy logic, in order to warn the user when an active robot is about to become unusable and should be replaced. Thus, the interaction time, which depends on the robot's battery life, can be increased by replacing the assistant with another more fully charged one, making the interaction seem more natural.

Keywords pointing gesture, multimodal interface, speech recognition, DSS, assistant robot, HRI

1. Introduction

The primary focus of robotics has been limited to industrial and manufacturing applications. Recently, however, many more kinds of robot have been developed and they are becoming a necessary part of our life. Nowadays, robots are becoming a common presence in various environments, covering many applications, including tour guiding, elder care, rehabilitation, search and rescue, surveillance, education, and general assistance in everyday situations. They work together with humans in factories, offices and in their homes. This idea of turning robots into a ubiquitous technology took shape gradually, alongside human progress and the evolution of technology, as a response to the human desire to create mechanized agents in order to help with everyday needs.

In this way, a new broad field of robotics emerged, generically named "Assistive Robotics" (AR). Initially developed for assisting people with disabilities or elder care [1], AR now includes companion robots, educational robots and, in more general terms, all robots that provide aid or support to human users [2]. AR is aided in its endeavours by the large number of service robots (SRs) that have emerged over recent years. An SR operates at home or in office environments, and performs certain tasks.

Thus, with the advent and development of assistive mobile robots, the need to find new and more intuitive ways for people to interact with them has become increasingly evident [3], so that any user, whether experienced or not, can use them without requiring too many preliminary instructions. In this sense, human-robot interaction (HRI) requires simple and easy-to-use interfaces. The general term adopted by most researchers for such an interaction is 'natural', which means that the user should be able to

interact with robots or other technologies as they would interact with other people or with the real world in everyday life. This type of interaction involves the need for new and interdisciplinary components to become part of the robotic research, which was already envisaged some time ago. For instance, in [4], the following sentence appears: "Robots of the future will interact with humans in a natural way. They will understand spoken and gestural commands and will articulate themselves by speech and gesture."

This phrase is emblematic of the further evolution of robotics. As we can see, spoken words and gestural commands are two of the most frequently used forms of interaction, and they are able to become advanced interaction components due to the latest developments in vision and audio technologies.

Although an ideal interface for HRI applications should contain a single interaction modality, multimodal interfaces (MI) can offer a number of advantages over traditional interfaces (e.g., a user-friendly experience, providing redundancy for a better understanding of commands). MIs allow a person to communicate with a robot by simultaneously using speech, gesture, gaze, facial expressions or, more recently, brain signals [5]. Using a pointing gesture to indicate a location or to select an object, combined with speech in order to express the desired intent, would be an example of a typical case of MI; this is also the topic of our work.

In this paper, we present a specific approach to multimodal interaction with a humanoid mobile robot that assists people in domestic or office environments. The aim of our research is to develop an assistive robot that can help humans in their daily activities and naturally interact with them, providing 'communicative' capabilities. The implemented interface allows the user to express their intentions using both verbal and non-verbal expressions. In this way, the interaction is more natural and more human-centred. We propose an interaction paradigm that consists of a combination of both ways of sending commands to the mobile robot. Thus, the robot can move in an indicated direction and perform a required task. Moreover, the unpleasant situations caused by power loss (when the robot's battery is fully discharged) can be avoided by using a decision support system that monitors the operating state of the robot.

The rest of this paper is organized as follows: Section 2 provides a literature review; Section 3 presents an overview of the system proposed, while Section 4 describes the proposed interaction paradigm; Section 5 discusses the decision-making aspects; Section 6 deals with the results obtained; and Section 7 offers conclusions and directions for future work.

2. Related Work

In this section, we intend to briefly review related work that covers a number of areas: multimodal interfaces in HRI, the pointing gesture and decision making.

2.1 Multimodal interfaces in HRI

Multimodality is a characteristic of human communication. People often use numerous information channels in their everyday interactions and it is desirable that this characteristic be transferred to HRI, allowing humans to communicate in a natural manner with robots. One of the first attempts to address this idea was the "Put-That-There" system presented in [6], which is considered the forerunner of multimodal interfaces. This system combines speech and gesture recognition in order to allow users to command events on a large-format raster-scan graphics display. Following that, a variety of new systems emerged based on multimodal interaction with the user, and most of these are mentioned in [7] and [8]. Despite the large number of systems, with various combinations of inputs, recognition methodologies and domains of application, there are still many shortcomings that need to be considered.

In [9], a multimodal scheme for interaction with service robots is proposed, focusing on motion detection, sound localization, people tracking, user (or other person) localization, and the fusion of these modalities. An interesting approach is presented in [10], in which a multimodal conversational interaction system is developed which enhances the naturalness and expressivity of interactions. A brief overview of some key aspects, issues and challenges of multimodal interaction is given in [11].

There is a variety of ways to interact with a mobile robot, but we think that gesture and speech channels are the most suitable, a conclusion that it supported by other authors too (for instance, [12] and [13]).

2.2 Pointing gesture

The use of the pointing gesture is one of the first ways in which people communicate with the world [14]. It is a foundational building block of human communication [15] because it is used during the early phases of language development, in combination with speech, in order to name objects, indicating a developmentally early correspondence between word and gesture [16]. Pointing is a deictic gesture [17], used to orient the attention of an observer on a location or object, or to indicate a direction or event. The deictic gesture class includes a larger set of gestures that are used to draw attention to an object [18], but in this work we are just interested in pointing, which is the most important way of communicating during infancy and involves extending the hand towards a specific location.

Gestures cannot be separated from speech. They are "elements or instruments of language", as the philosopher Wittgenstein called them [19]. Moreover, gesture and speech work together for adults, to convey a single message integrated in both time and meaning [20], while pointing gestures support the learning of language for infants [21]. Several interpretations of pointing gestures have been identified by developmental psychologists [15], but two of the most obvious are imperative pointing and declarative

pointing [22]. These are used instead of spoken deictics, or to supplement them. It is well known that pointing at a reference object is a much faster and more convenient method than describing it verbally.

When dealing with assistive robots, the user should be able to inform them about an object's location. This interaction is crucial in a home environment, because most of the tasks performed by the robot require the manipulation of objects. In this regard, the pointing gesture is recognized as an essential way of interacting with robots. The deictic communication should be bidirectional. The operator uses pointing with verbal cues, and the robot asks for supplementary details if something is not clear.

Many research works have dealt with the interaction with robots using deictic communication. In some of these, a robot is used as a testbed to investigate specific issues relating to the social development of humans [23] [22]. Other works try to enable a robot to generate its own pointing gestures [18] [24], or investigate the use of pointing gestures in interactions between a human and a robot [25] or between two robots [26], but most papers focus on the ability of a robot to respond to pointing gestures performed by human operators. Some papers focus on the recognition methodology and the estimation of pointing direction [27] [28]; others focus on the control component of the system [29] or on the process of achieving a natural deictic communication between robots and humans [30].

We propose a simple solution for deictic communication that fuses vision and speech recognition using a single device; we call this paradigm "Point-and-Command" (PaC).

2.3 Decision making

Another issue that should be considered in HRI is the decision-making process. When a robot and a human share the same space, the same objects or a common goal, they need to collaborate. The robot must have a perception of its environment and may ask the human to help when needed. On the other hand, while the human should also have an awareness of their environment, they must bear in mind the robot's internal state in addition. Besides this, both human and robot must be able to coordinate their actions. The trust of the user in the robot's autonomous decisions depends largely on the feedback received.

In this paper, we are interested in both human and robot decision making.

In the section relating to the robot's decision making, fuzzy logic (FL) was used when information was vague, inconsistent or incomplete (decisions under uncertainties), e.g., for person detection, and tracking in cluttered and unstructured environments [31]. Other stochastic models, like Markov decision processes (MDP) [32], partially observable Markov decision processes (POMDP) [33], Bayesian networks (BN) [34], deep dynamic Bayesian networks (DDBN) [35], and hidden Markov models (HMM) [36], have been used in designing user-assistive systems for intention recognition or speech recognition. An interesting approach to collaborative human-robot decision making is presented in [37], making use of probabilistic robotics representations.

Decision support systems (DSS) represent a class of computer-based information systems, which include knowledge-based systems, that support decision-making activities. The main aim of a DSS is to help decision makers to make the best decision when dealing with complex situations and information [38].

There are some studies that make use of DSSs for robotics applications. In [39], the operation of a fleet of robots is controlled using local decision makers, based on MDPs embedded in robots, in combination with a DSS for the operator to help decide whether is necessary to teleoperate robots when they are in degraded states. A DSS was also implemented in [40], in the form of a leader selection mechanism, to help a single pilot control a group of UAVs (unmanned aerial vehicles). Some papers deal with the use of DSSs in robot selection [41] [42] or for selecting solutions for autonomous robotic systems [43]. In [44], a DSS based on fuzzy logic was presented for improving hand-eye coordination in children, through a haptic robotic interface. However, to our knowledge, DSSs have been put to little or no use in HRI applications.

In this paper, we implement a DSS to select between two robots, allowing them to act as assistants in a home environment. The DSS is based on fuzzy logic and informs the user when a robot will become unusable, in order that it might be replaced by another one; however, this information is provided in a timely manner, allowing the first robot to reach the place where it can recharge its battery.

3. System Architecture

In this section, we give an overview of the system architecture, highlighting some of its features in order to set the context for illustrating the Point-and-Command (PaC) paradigm proposed in this paper.

The system is designed for the interaction with personal mobile assistive robots for tedious, repetitive tasks, operating at home or in office environments. In Fig. 1, the general schema of the system is depicted, with all the 'blocks' that are involved in the interaction.

The human operator and the environment are the 'end-points' of interaction, in the sense that an operator interacts with the environment by means of his/her 'assistant': a mobile robot. He/she sends gestural and voice commands to the robot, which are captured using a Kinect device. Speech and gesture recognition are two separate processes, but they are combined in the speech/gesture fusion block in order to provide more complex commands. A robot selection DSS is responsible for choosing which of the two

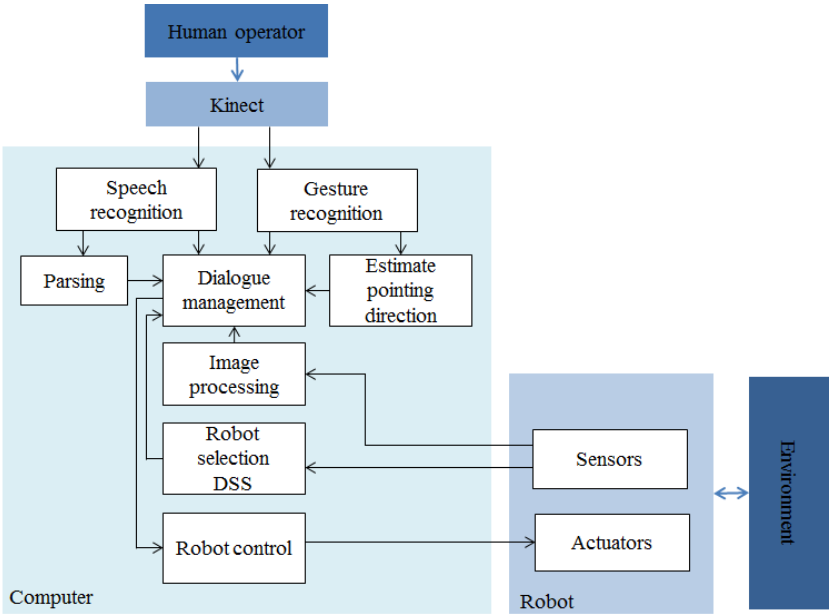

Figure 1. System overview

robots should be the 'active' one, prepared for interaction. A control unit sends the commands to the robot, which are generated in the dialogue management block.

All computations are performed on a computer (desktop PC). The software allows data from the robot's sensors and motors to be accessed. The robot used in this research has two embedded cameras: one is located on the robot's forehead and the second is at mouth level. They can only be used one at a time for capturing images; they cannot be used simultaneously. The images taken are sent via Wi-Fi to the computer, where they are processed. For communication with the robot's actuators, the Device Communication Manager (DCM) module of the robot's API was used.

In the first phase, the system recognizes the human's pointing gestures and verbal cues. These are combined and then transformed into commands that are transmitted to the robot via wireless connection. The pointing direction is recognized using data captured by the Kinect device and the estimated direction is then used to inform the robot about the location of interest. As mentioned previously, the system is designed to help humans in their everyday activities and therefore the 'workspace' should be seen as a slightly more intelligent living environment (with computer, vision sensor and markers).

In this work, we make use of the concept of patterns. Keeping in mind the idea developed in [45] and extended in [46], we propose some general patterns that are frequently encountered in relation to assistive tasks. Taking inspiration from the design patterns which exploit the reusability aspect, as well as from the work presented in [47], certain patterns that were observed within the interaction activity were recreated.

Fig. 2 shows two such 'patterns', with their components. In the picture it can be seen that the "Move to Target" (MtT) pattern contains the following 'sub-tasks': Marker Detection (MD), Target Identification (TI) and Navigation (N). Thus, when the robot receives the command to move to a target, it must firstly detect a marker placed in the environment, in order to determine its position. Then, it has to identify the target according to the direction indicated by the human's pointing gesture; and finally it has to navigate to that target. Target identification and navigation may overlap; in the event that the target is too far away, the robot should move in its general direction before reattempting the procedure of target identification. In this way, the sub-procedures are activated when a command is given. Some sub-tasks have their own sub-routines; for instance, an "Execution" sub-task may comprise reading sensor information and then controlling actuators according to data obtained from the sensors.

The robot can perform the following tasks: navigation, grabbing an object, pushing an object, fetching and carrying, and teleoperation tasks.

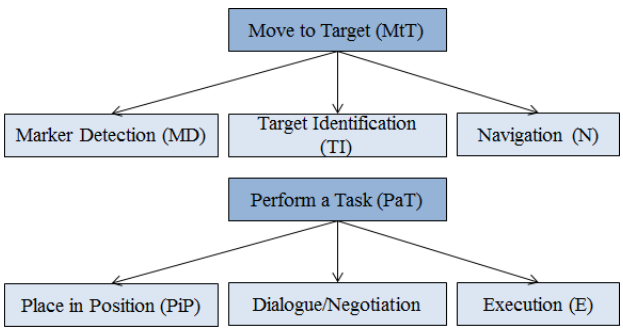

Figure 2. Structure of two patterns: "Move to Target" and "Perform a Task"

The "Grab an Object" task consists of Detect Object, Grasp Object and Lift Object sub-tasks. All data are processed on the host computer. The real-time video stream captured

from the robot's camera is processed on the computer and the robot only receives control commands for its actuators.

Teleoperation functions are realized using the implementation described in [48].

4. Point-and-Command Paradigm

There is an increasing trend towards developing robots to assist humans in more and more application domains. This is partly due to the fact that personal robots have practically invaded the research and education marketplace and they are supposed to begin invading our lives very shortly. A large number of robots already work in human living spaces, as well as hospitals, in exhibition centres, universities or museums. However, in order to be a good assistant, a robot should possess social and cognitive skills. It is therefore important to make sure it is able to interact with people in a human-like manner.

People use pointing gestures in communication as a mechanism to indicate a reference object or to inform a listener about its location. In other words, pointing is used to establish the identity or spatial location of an object within the context of the application domain [49]. Pointing gestures can also be used by humans to provide directional information to robots [50]. They are especially useful for users who need to interact with mobile robots, but who do not have a priori knowledge of HRI.

In this paper, we exploit the idea of combining human pointing gestures with speech in order to communicate intentions to a robot, in a user-centred approach called "Point-and-Command" (PaC). In simple terms, the idea of the interaction process is the following: a person points at an object (or simply to a location) and says something such as "grab that object", and then the robot will navigate in the indicated direction and will perform the task (identify the object and grab it). In [30], five processes are outlined that are involved in natural deictic communication between robots and humans: context focus, attention synchronization, object recognition, believability establishment and object indication. These processes will also occur in our case, but in other forms.

In our case, the scenario is conceived for laypeople, who require an easy-to-use interface. A user sitting at his/her desk, or a mobility-impaired person, can use his/her voice and upper body parts to ask a personal mobile robot to bring them an object from the room (or workspace). To achieve this, they should indicate the object under consideration or specify the Point of Interest (PoI) [50]. This indication is not explicit; it only involves a gestural and a voice cue, and this mode of interaction is therefore considered to involve high-level commands, while a certain level of autonomy is assumed on behalf of the robot [51].

The pointing direction is determined by computing the angle between the operator's arm and his/her body. The Kinect API gives the coordinates of 20 joints, but in our case, only seven are relevant, which are depicted in Fig. 3. The coordinates of the shoulder and wrist are used to create a line directed towards the target. The focus is not on the totally accurate calculation of the pointing angle, since this information is just to give an initial direction to the robot; it will then automatically (using vision) or semi-automatically (by asking for new details) detect the target. This is why we choose not to use more precise methods of tracking (e.g., fingertip detection combined with face detection, like in [27]), but this aspect will be considered in future work.

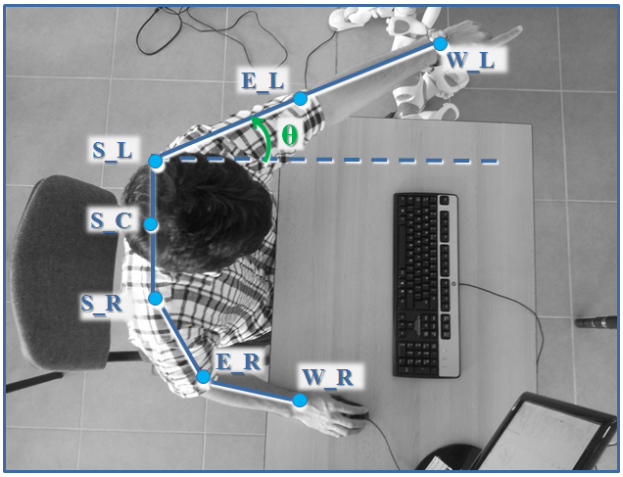

Figure 3. The joints of the human body obtained from the Kinect sensor, and the estimated angle (in green): S – shoulder, E – elbow, W – wrist; C – centre, R – right, L – left

Thus, the estimated pointing angle is obtained from the coordination of the center of the shoulders, shoulder and wrist of the operator, forming two vectors and being projected onto the horizontal plane. In other words, the vector from the shoulder to the wrist indicates the direction in which the operator is pointing. The user shoulders are approximately perpendicular to the Kinect sensor and thus, the angle formed by the line of the shoulders with the line of the arm gives the direction in which the object is located. There are two possible situations for each arm: when the angle is larger than 90°, the object is located on the same side as the pointing arm, relative to the human body, and the angle = 90° - θ (Fig. 3); alternatively, when the angle is less than 90°, the object is located on the opposite side of the body relative to pointing arm, and the angle is β, as in Fig. 4.

The robot has a default fixed position, where the power supply is located. It initially has the same orientation as the human operator. Thus, it will start moving in the direction indicated by the human arm, but from its original position. Since it knows its initial orientation towards the operator, it can rectify its orientation by rotating its head and looking for objects by colour.

A flow diagram of the Point-and-Command process, which illustrates the explanations given above, is presented in Fig. 5.

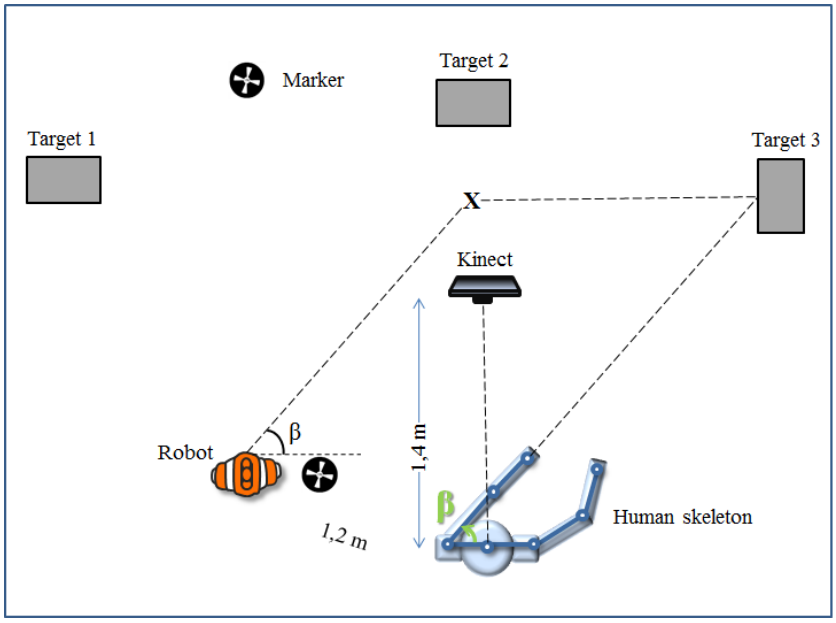

Figure 4. The experimental setup

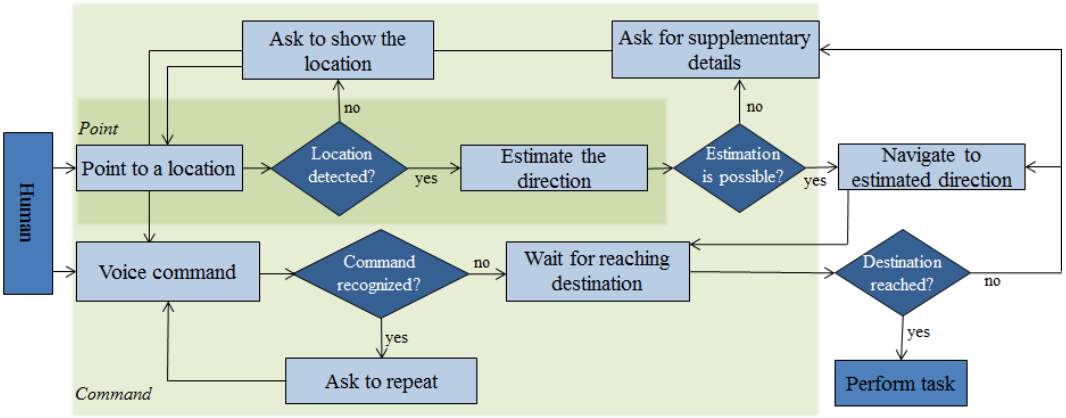

Figure 5. Workflow diagram

4.1 Gesture detection and recognition

Gesture recognition is an important aspect of robotics because it provides an intuitive way to naturally communicate with robots. There is a variety of techniques and technologies available for gesture detection, but the most modern approaches are vision-based. They involve one or more video cameras that record the movement of the human's body parts.

In this work, a Kinect sensor was used to track the human arms. The Kinect software API provides the positions of the body joints of the user in real time, at the frame rate of 30 frames per second (fps).

The frames obtained from the Kinect sensor are converted into feature vectors that contain the positions of different joints of the body. The motion frame is expressed as an 18-element feature vector, containing the x,y,z positions for each joint of each arm (wrist, elbow and shoulder) expressed in metres by the coordinate system of the Kinect camera. Each gesture to be included in the gesture vocabulary will have been previously recorded and saved as a feature vector in a database, in order to form a set of templates. In this way, a gesture is represented by a sequence of arm postures over time and the recognition process involves the comparison of the current sequence with a range of previously recorded sequences. For this process of gesture recognition, the dynamic time warping (DTW) algorithm was chosen, because it is both robust and accurate enough for our needs. Through this technique, the similarities between two feature vector sequences are computed, in order to find the optimal alignment. The DTW algorithm, and its application for gesture recognition with the Kinect device, is presented in [52]. In contrast to the implementation in the aforementioned work, we used the FastDTW method presented in [53]. The accuracy obtained by applying this technique was 93% for a set of gestures performed by the users during tests.

Each gesture in the database is represented in 33 frames. The duration of the gesture performed by the user does not have a critical role because DTW can compare sequences of different lengths.

The algorithm performs the time alignment and normalization by computing a temporal transformation allowing the two signals to be matched [54]. This means that it is not sensitive to the duration of the gesture.

In a pre-processing step, the coordinates are normalized with the distance that connects the upper body joints to reduce the variations caused by the different sizes of the users.

This algorithm was used because we want to extend the gesture vocabulary in order to provide a more complex gesture interaction.

4.2 Speech recognition

For capturing the human voice, the same device (the Kinect) was used. Kinect for Windows SDK allows the use of the Microsoft.Speech recognition API. The streaming audio is captured by the Kinect microphone array using a speech recognition engine provided by the SDK. Microsoft.Speech was used to create grammars that can recognize a single word or a short phrase. Depending on the interaction context, different grammars are implemented for each situation. Each grammar has several rules that define a pattern or a sequence of words. An example from the grammar of the Point-and-Command initial command is as follows:

<Robot name> <please (opt) > <verb1> <spatial deixis> <**and**> <verb2> <article (opt)> <property> <name>

<Robot name> is the name of the robot. This could be NAO 1 or NAO 2, or something else if the operator wishes to call them by specific names, and it is used to indicate that the operator is 'speaking' to the robot. If the utterances do not contain this 'keyword', they will be ignored by the system. <Verb1> contains a set of verbs like "go" and "look at", and <spatial deixis> contains words like "there" and "that". The <**and**> conjunction is written in bold to emphasize the Point-and-Command idea. In <verb2> there is another set of actions: "bring", "grab", "push", etc.; the subsequent <article> is optional (opt), but could be "that", "an", "a", etc. <Property> refers to a property of an object—e.g., colour ("red", "blue", "green") and shape ("round", "square")—and <name> could be the name of a known object or a noun ("sphere", "cube").

The speech recognizer also provides a confidence level for each recognized word. If a word is detected, but it has a very low confidence level, the robot will notify the human, asking him/her to repeat the command.

In the dialogue management block depicted in Fig. 1, there are more sub-functions. Besides managing the dialogue between human and robot, this block is also responsible for combining spoken words with pointing gestures, and deciphering the meaning of these complex commands. This is achieved with the help of an inference engine.

The inference engine contains rules that help to identify the task that the robot should perform. The inputs for the inference engine are the speech/gesture commands and the output is represented by the required task. Speech recognition and gesture recognition are separate processes, but they run simultaneously. After a gesture has been recognized, its name is stored as a variable which will become the input for the inference engine. The same thing happens with a recognized word. Thus, for these two variables, the inference engine returns the task required, according to the rules set. There are three ways of giving a command: by gesture, by voice, or by a combination of the two. Likewise, there are simple tasks (e.g., navigation, grabbing) and more complex tasks (e.g., fetching). Simple tasks can be requested using just one of the two modalities of interaction, but complex tasks require the fusion of both of them, as in the Point-and-Command paradigm presented above. After a preliminary command ("NAO!"), the system is made to 'listen' continuously. If the user gives a voice command, the system waits four seconds for another command, and if it does not come, it will only consider the voice command given. When the command consists of a speech/gesture combination that is considered incompatible, the complex command is ignored. In this way, the system knows at each moment what task is to be performed by the robot.

5. Decision Making

5.1 Object detection

People use objects in most of their everyday activities. When they talk about an object within an environment, they will often point at it while using specific terms (e.g., "that") or even describing the object (i.e., "the red ball"), in order to draw someone's attention towards it. This is a common behaviour used all over the world. We saw how this behaviour can be used in HRI applications, and how the robot can 'understand' what a human operator wants, but now we must discuss the second part of this operation: the identification of an object in the environment.

After the system has estimated the direction of movement, the image-processing unit 'comes into play'. The robot moves forward in that direction, searching for potential objects. During this process, data from the robot's sensors (i.e., video camera and sonar sensor) are processed for the detection of objects or obstacles. EmguCV was used for image processing, which is a. NET wrapper for the OpenCV library[1]. The library allows object detection by colour and by shape. An OpenCV implementation of the Canny

1 http://www.emgu.com/wiki/index.php/Main_Page [Accessed on 9 Nov 2014]

algorithm was used for edge detection and the Hough transform technique was then used to detect lines and circles in the image [55]. To detect the colour of the objects and algorithm, HSV colour space was used[2].

The objects used for the experiments were Styrofoam balls and wood blocks, in different colours (Fig. 6). The balls have a diameter of 5 cm and the cubes have sides of 4 cm length to allow the robot to grasp them.

Figure 6. The experimental setup

No	Distanc e [m]	Angle [°]	Averag e time [s]	Distanc e [m]	Angle [°]	Averag e time [s]	Distanc e [m]	Angle [°]	Averag e time [s]
	Experiment 1			Experiment 2			Experiment 3		
1	3.02	26.6	3.5	3.15	5.1	4.04	2.88	51.4	3.23
2	3.81	45.7	4.87	3.51	50.2	4.6	4.15	12.6	5,9
3	3.71	14	5.41	5.49	55	7.12	4.78	41.2	6,41

Table 1. Location of pointing targets relative to robot position and average time required for performing a task

5.2 Human-robot collaboration

The interaction between human and robot follows a specific course. The initiative to start an interaction belongs to the user. He/she must pronounce the robot's name or perform a 'wave' gesture in order to draw the robot's attention. With this, the connection is established and the robot is 'ready' to receive further commands.

5.3 DSS

One of the main limitations of mobile robots concerns their energetic autonomy, meaning their battery life. In the case of the mobile platform used in this research, it cannot surpass 40 minutes of autonomy in the operational mode. To avoid these limitations, a docking station was proposed in [56] which would not require human assistance. This seems to be a good solution, but we have tried an approach from a different perspective. Since there are two identical robots in our laboratory, we chose to use both of them. When the battery of the first one begins to go flat, the second 'swings into action'. However, the decision of selecting the other robot belongs to the human operator. For this purpose, a decision support system (DSS) was developed, in order to help the user to decide when to stop the interaction with one robot and call the other.

The DSS, based on fuzzy logic (FL), is designed to deal with uncertainty and aid the operator in making decisions regarding which robot to interact with, considering their battery levels and distance from the location of the power supply.

The decision support system was therefore designed as a fuzzy logic system, with the following main components: fuzzification, an inference engine, a rule base, and defuzzification. The system has four input variables and one output variable. The input variables are:

1. the percentage of power remaining;
2. the distance between the robot and the power supply;
3. a variable that shows the internal state of the robot;
4. operating time.

The percentage of power remaining is obtained from the robot's internal sensor and fuzzified into the following fuzzy sets: "very low", "low", "medium", "good" and "fully charged". The distance between the robot and the location of the power supply is determined by periodically locating the Naomark placed in that location and extracting the distance. The fuzzy sets for this variable are "very close", "close", "not so far", "far" and "very far". The internal state of the robot is determined by its temperature sensors and fuzzified into the following: "normal", "hot" and "very hot". The operating time variable records the time elapsed since the NAO began to operate, and it

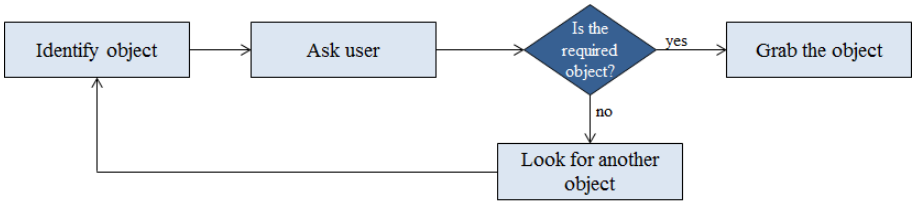

Figure 7. Flow diagram for the operation of identifying a target object

2 http://opencv-srf.blogspot.ro/2010/09/object-detection-using-color-seperation.html [Accessed on 20 Oct 2014]

includes "very short", "short", "medium", "long" and "very long". The membership functions belonging to these fuzzy variables are trapezoidal.

The output of the fuzzy system is a value from 1 to 4, representing the operating state of the robot: "good", "normal", "low" and "very low". The rule base of the system contains a total of 375 rules that determine the robot's state, which are then used to inform the user of when the robot is fully operable and when it is not. For instance, one rule from the rule base is as follows:

If (*remaining_power* is *very_low*) and (*distance* is *very_close*) and (*internal_state* is *normal*) and (*operating_time* is *very_short*) then (*robot_state* is *bad*)

The robot's operating state is permanently displayed on the graphical user interface (GUI) and the robot also informs the user periodically about its state using vocal messages, according to the operating state variable. The DSS is continuously active and, in this way, the user will know when it is time to ask the robot to return to its charging location, in order to give it time to get there on foot before full battery discharge. Hence, the interaction remains natural while situations where the robot might suddenly collapse due to a low battery are avoided.

6. Fetching Robot Application

As we mentioned previously, the platform selected for this research is the humanoid robot NAO (Fig. 9), created by the French company Aldebaran Robotics to be a true daily companion[3]. NAO is a medium-sized robot used in many universities for educational and research purposes. The multitude of sensors and actuators, the complete programming environment and the human-like appearance make it a favourite tool for developers of HRI applications. The robot's is equipped with two cameras: one in its forehead and one below. They support four resolutions and the frame rate for each resolution is 30 frames per second.

A simple testing scenario was proposed in order to examine the functionality of the entire system. The experiment was performed in an indoor environment: the Industrial Virtual Informatics and Robotics Laboratory at the Transilvania University of Brașov. Three boxes with objects of various colours and shapes were placed in the environment in different positions, as illustrated in Fig. 10. Two NAO robots were sat on the left-hand side of the user at a certain distance from him/her, next to the power supply.

The tests were performed by three different people, each of them three times. During the first round of tests, the users asked NAO to bring them the following objects: a red ball from the left box, a green ball from the box placed in the middle, and a yellow cube from the right box, relative to the user's own position. The users were initially informed of how the system worked and they were able to try it out a few times to get familiar with the interaction. Table 1 shows the distances between the robot and the objects, but also the average time elapsed (for each of the three users) before the robot had successfully performed the task. Some snapshots from one of the experiments are given in Fig. 11.

The tests lasted about two hours. During this period, the first robot selected for interaction operated for 35 min; after this, the DSS informed the user that it was time to call the second robot. The second robot operated for seven more minutes, until the end of the first round of tests. In the second and the third round, the proportion of operation time was similar, as shown in Fig. 8.

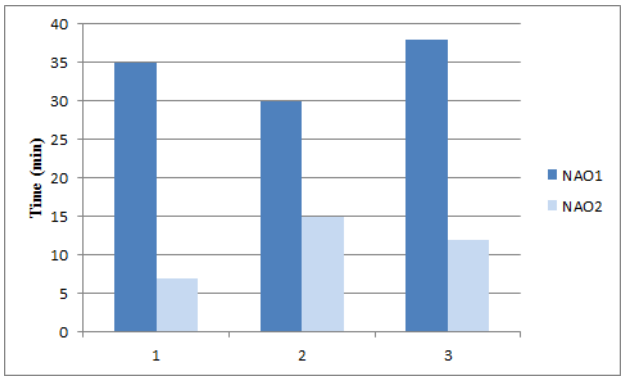

Figure 8. Operating time for each robot during the experiment

The accuracy of pointing gesture recognition was 91%, and for speech recognition, 95%. Table 2 gives the average errors of estimating the pointing angle.

No.	Average error [°]		
	Experiment 1	Experiment 2	Experiment 3
1	4.56	3.7	5.18
2	3.29	3.12	3.9
3	4.3	4.65	5.32

Table 2. Average error of estimated pointing angle

Regarding the vision-processing algorithms, the accuracy of object detection and recognition was also acceptable. Each experiment required the recognition of nine objects: three (for each task) x three (for each user). So, in total, 27 objects had to be recognized throughout all three experiments. From these experiments, the robot only misidentified an object twice, but in both cases, after the user gave it another instruction, the robot found the correct object. The robot always asked the user before performing a grasping action. The workflow diagram of this process is depicted in Fig. 7.

We also considered the case when two objects of the same colour are next to each other. When the robot faces such a situation, it may check which one the user is asking for. The user should answer using voice commands (e.g., "left" or "right"). If objects of the same colour are very close to each other, the robot will not be able to identify them separately.

3 http://www.aldebaran.com/en/humanoid-robot/nao-robot [Accessed on 10 Feb 2014]

Figure 9. Two NAO robots awaiting commands

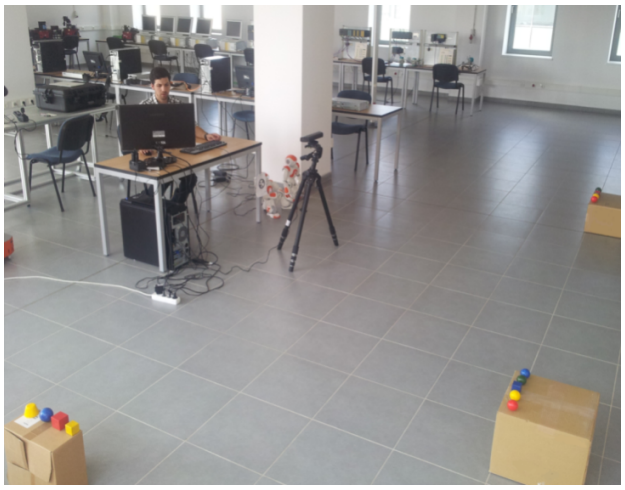

Figure 10. The operator sits in front of the Kinect sensor. He points at one of the three boxes and asks the robot to bring him a coloured object.

7. Conclusion

7.1 Summary

Along with the development of more and more intelligent biped robots, the urgent need for personal robots in homes, in offices, and in educational or cultural institutions has grown rapidly. In this context, the researchers should consider the development of new and intuitive ways of interacting with robots, in order to allow their owners (who are usually laypeople) to express their intensions in a simple manner and to feel comfortable in their presence. Verbal and non-verbal communication plays an important role in these types of application.

This paper presents the first step in an attempt to develop an interface for assisting people in their everyday tasks, using two humanoid robots that can navigate through the human space, and are able to detect and fetch objects. The human user is able to decide when to end an interaction with one robot, whose battery may be running out, and to select the other robot to continue the task that the first robot has begun. This is possible thanks to a DSS, which collects vital pieces of information concerning the robot's internal state and its environment, informing the user in time to allow the robot to return to its home position on foot (before the end of its battery life). To achieve this, the user simply sends the 'back home' command using the speech function.

The human-robot interaction is based on pointing gestures combined with vocal commands. A "Point-and-Command" interaction concept was proposed with the aim of providing the user with an easy-to-use and intuitive interface and to improve the efficiency of human-robot collaboration. The system doesn't hold all the knowledge required to fulfil any give task, but it does present the possibility of retrieving information from the user.

We tested the functionality of the system in an indoor environment and the results proved that all users could interact with the humanoid robots without requiring previous knowledge about them. The main contribution of the work is the natural interaction achieved with our interface.

7.2 Future work

The system presented in this paper has several limitations which we intend to overcome in future research projects.

Figure 11. (a) NAO walks in the indicated direction, identifies the red ball, grabs it and returns to the user; b) the robot's line of sight viewed from different positions

Some of these are not directly connected with the subject of the paper, like the problem that an object could fall from the robot's hand while it is moving.

In our work, the main processing unit is a computer located on the user side, and this unit communicates with the robot via wireless commands. We plan to transfer a section of this 'knowledge' to the robot side, to make it more intelligent and to reduce the amount of information transferred.

We plan to continue similar tests with more users and to extract more test result data to develop a background against which comparisons can be made with similar systems. After this, real-life situations should be considered, in order to satisfy the main goal of the system: assisting people with their everyday tasks.

Further research should include improvements in the following areas: human-robot communication, pointing angle estimation, robot localization and obstacle avoidance. Additionally, the robot's learning ability should be investigated. It must be able to learn and 'keep in mind' the identified objects.

8. Acknowledgements

This paper was supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), ID134378, and financed by the European Social Fund and the Romanian Government.

This article is a revised and expanded version of a paper entitled "The Point-and-Command Paradigm in Human-Robot Interaction", presented at the 6th Győr Symposium, the 3rd Hungarian-Polish and the 1st Hungarian-Romanian Joint Conference on Computational Intelligence.

We would like to express our eternal appreciation for the guidance and support we received from our teacher, Mr Doru Talabă, up until his unexpected and sad death, following a terrible car accident.

9. References

[1] Haigh KZ, Yanco HA. Automation as caregiver: a survey of issues and technologies. In: Proceedings of AAAI 02 Workshop "Automation as Caregiver"; Edmonton. Alberta. Canada; 2002. p. 39-53.

[2] Feil-Seifer D, Matarić MJ. Human-Robot Interaction. In: Meyers RA, editor. Encyclopedia of Complexity and Systems Science. New York: Springer; 2009, p. 4643-4659.

[3] Butnariu S, Gîrbacia F. The Command of a Virtual Industrial Robot Using a Dedicated Haptic Interface. Advanced Materials Research. 2013; 837: 543-548.

[4] Triesch J, Von Der Malsburg C. A gesture interface for human-robot-interaction. In: Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition (FG '98); 14-16 April 1998 ; Nara. Japan; 1998. p. 546-551.

[5] Postelnicu CC, Talabă D. P300-Based Brain-Neuronal Computer Interaction for Spelling Applications. IEEE Transactions on Biomedical Engineering. 2013; 60: 534-543.

[6] Bolt RA. Put-that-there: Voice and gesture at the graphics interface. In: Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '80); New York. USA; 1980. p. 262-270.

[7] Jaimes A, Sebe N. Multimodal human–computer interaction: A survey. Computer Vision and Image Understanding. 2007; 108: 116-134.

[8] Karray F, Alemzadeh M, Saleh JA, Arab MN. Human-Computer Interaction: Overview on State of the Art. International Journal on Smart Sensing and Intelligent Systems. 2008;1:137-159.

[9] Böhme HJ, Wilhelm T, Key J, Schauer C, Schröter C, Groß HM, Hempel T. An approach to multi-modal human–machine interaction for intelligent service robots. Robotics and Autonomous Systems. 2003;44:83-96.

[10] Csapo A, Gilmartin E, Grizou J, Han J, Meena R, Anastasiou D, Jokinen K, Wilcock G. Multimodal conversational interaction with a humanoid robot. In: Proceedings of the 3rd IEEE International Conference on Cognitive Infocommunications (CogInfoCom); 02 - 05 December; Budapest. Hungary; 2012. p. 667-672.

[11] Turk M. Multimodal interaction: A review. Pattern Recognition Letters. 20144; 36:189-195.

[12] Nguyen-Duc-Thanh N, Lee S, Kim D. Two-stage Hidden Markov Model in Gesture Recognition for Human Robot Interaction. International Journal of Advanced Robotic Systems. 2012; 9.

[13] Morales SOC, Enrĩquez GB, Romero FT. Speech-Based Human and Service Robot Interaction: An Application for Mexican Dysarthric People. International Journal of Advanced Robotic Systems. 2013; 10.

[14] Colonnesi C, Stams GJJM, Koster I, Noomb MJ. The relation between pointing and language development: A meta-analysis. Developmental Review. 2010;30:352–366.

[15] Kita S. Pointing: A foundational building block in human communication. In: Kita S, editor. Pointing: Where Language, Culture, and Cognition Meet. Mahwah, New Jersey: Erlbaum; 2003, p. 1-8.

[16] Roth WM. Gestures: Their Role in Teaching and Learning. Review of Educational Research. 2001; 71:365-392.

[17] McNeill D. Hand and Mind: What Gestures Reveal About Thought. Chicago: University of Chicago Press; 1992.

[18] Sauppé A, Mutlu B. Robot deictics: how gesture and context shape referential communication. In:

Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction; Bielefeld. Germany; 2014. p. 342-349.

[19] Wittgenstein L. The Blue and Brown Books (Preliminary Studies for the Philosophical Investigations). New York: Harper Torchbooks; 1965.

[20] Butcher C, Goldin-Meadow S. Gesture and the transition from one- to two-word speech: when hand and mouth come together. In: McNeill D, editor. Language and Gesture. Cambridge, United Kingdom: Cambridge University Press; 2000, p. 235-258.

[21] Goldin-Meadow S. Pointing Sets the Stage for Learning Language—and Creating Language. Child Development. 2007;78:741-745.

[22] Scassellati B. Investigating models of social development using a humanoid robot. In: Proceedings of the 2003 International Joint Conference on Neural Networks; Portland. Oregon. USA; 20-24 July 2003; 2003;4:2704-2709.

[23] Nagai Y. Learning to Comprehend Deictic Gestures in Robots and Human Infants. In: Proceedings of 2005 IEEE International Workshop on Robot and Human Interactive Communication; Nashville. USA; 2005. p. 217-222.

[24] Chao F, Wang Z, Shang C, Meng Q, Jiang M, Zhou C, Shen Q. A developmental approach to robotic pointing via human–robot interaction. Information Sciences. 2014;283:288-303.

[25] Beuter N, Spexard T, Lutkebohle I, Peltason J, Kummert F. Where is this? - gesture based multimodal interaction with an anthropomorphic robot. In: Proceedings of the 8th IEEE-RAS International Conference on in Humanoid Robots; 1-3 December 2008; Daejeon. South Korea; 2008. p. 585-591.

[26] Hafner V, Kaplan F. Learning to Interpret Pointing Gestures: Experiments with Four-Legged Autonomous Robots. Biomimetic Neural Learning for Intelligent Robots. 2005; 3575: 225-234.

[27] Breuer T, Ploeger PG, Kraetzschmar GK. Precise pointing target recognition for human-robot interaction. Presented at the Workshop on Domestic Service Robots in the Real World, SIMPAR; Darmstadt. Germany; 2010.

[28] Droeschel D, Stuckler J, Behnke S. Learning to interpret pointing gestures with a time-of-flight camera. In: proceedings of the 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI); 6-9 March 2011; Lausanne. Switzerland; 2011. p. 481-488.

[29] Schiffer S, Ferrein A, Lakemeyer G. Caesar: an intelligent domestic service robot. Intelligent Service Robotics. 2012;5:259-273.

[30] Sugiyama O, Kanda T, Imai M, Ishiguro H, Hagita N. Natural deictic communication with humanoid robots. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007); 29 October – 2 November 2007; San Diego. California. USA; 2007. p. 1441-1448.

[31] Granata C, Bidaud P, Salini J, Ady R. Human activity analysis: a personal robot integrating a framework for robust person detection and tracking and physical based motion analysis. Paladyn Journal of Behavioral Robotics. 2013;4:131-146.

[32] McGhan CLR, Nasir A, Atkins EM. Human Intent Prediction Using Markov Decision Processes. Presented at the Infotech@Aerospace Garden Grove, California; 2012.

[33] Broz F, Nourbakhsh I, Simmons R. Planning for Human-Robot Interaction in Socially Situated Tasks: The Impact of Representing Time and Intention. International Journal of Social Robotics. 2013;5:193-214.

[34] Patel M, Valls Miro J, Dissanayake G. Dynamic Bayesian Networks for Learning Interactions between Assistive Robotic Walker and Human Users. Advances in Artificial Intelligence. 2010; 6359: 333-340.

[35] Zhao Y, Wang H, Ji Q. Audio-Visual Tibetan Speech Recognition Based on a Deep Dynamic Bayesian Network for Natural Human Robot Interaction. International Journal of Advanced Robotic Systems. 2012; 9.

[36] Rossiter J. Multimodal Intent Recognition for Natural Human-Robotic Interaction [thesis]. School of Electronic, Electrical and Computer Engineering; University of Birmingham; 2011.

[37] Kaupp T, Makarenko A, Durrant-Whyte H. Human–robot communication for collaborative decision making—A probabilistic approach. Robotics and Autonomous Systems. 2010;58:444-456.

[38] Cebi S, KahramanC. Developing a group decision support system based on fuzzy information axiom. Knowledge-Based Systems. 2010;23:3-16.

[39] Mouaddib AI. Controlling and Sharing Authority in a Multi-Robot System. In: Proceedings of 1st Conference on Humans Operating Unmanned Systems (HUMOUS '08); Brest. France; 2008.

[40] Ding XC, Powers M, Egerstedt M, Shih-yih Y, Balch T. Executive decision support. IEEE Robotics & Automation Magazine. 2009;16:73-81.

[41] Wang MJJ. A decision support system for robot selection. Decision Support Systems. 1991;7:273-283.

[42] Tansel İç I, Yurdakul M, Dengiz B. Development of a decision support system for robot selection. Robotics and Computer-Integrated Manufacturing. 2013;29:142-157.

[43] Heikkilä T, Dalgaard L, Koskinen J. Designing Autonomous Robot Systems—Evaluation of the R3-

COP Decision Support System Approach. Presented at ERCIM/EWICS Workshop on Dependable Embedded and Cyber-physical Systems (SAFECOMP 2013 - Workshop DECS); Toulouse. France; 2013.

[44] Pernalete N, Gottipati R, Grantner J, Edwards S, Janiak D, Haskin J., Dubey RV. Integration of an Intelligent Decision Support System and a Robotic Haptic Device for Eye-Hand Coordination Therapy. In: Proceedings of the IEEE 10th International Conference on Rehabilitation Robotics (ICORR 2007); 13-15 June 2007; Noordwijk. Netherlands; 2007. p. 283-291.

[45] Kahn PH, Freier NG, Kanda T, Ishiguro H, Ruckert JH, Severson RL, Kane SK. Design patterns for sociality in human-robot interaction. In: Proceedings of the 3rd ACM/IEEE International Conference on Human-Robot Interaction; 12-15 March 2008; Amsterdam. The Netherlands; 2008. p. 97-104.

[46] Sauppé A, Mutlu B. Design patterns for exploring and prototyping human-robot interactions. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems; 26 April – 1 May 2014; Toronto. Ontario. Canada; 2014. p. 1439-1448.

[47] Taylor G, Wray RE. Behavior Design Patterns: Engineering Human Behavior Models. Presented at the Behavioral Representation in Modeling and Simulation Conference; Arlington, VA. 2004.

[48] Boboc RG, Toma MI, Panfir AN, Talabă D. Learning new skills by a humanoid robot through imitation. In: Proceedings of IEEE 14th International Symposium on Computational Intelligence and Informatics (CINTI); 19 - 21 November 2013; Budapest.Hungary; 2013. p. 515-519.

[49] Karam M, Schraefel MC. A taxonomy of Gestures in Human Computer Interaction [technical report]. University of Southampton; 2005.

[50] Bauer A, Wollherr D, Buss M. Information retrieval system for human-robot communication—Asking for directions. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA '09); 12-17 May 2009; Kobe. Japan; 2009. p. 4150-4155.

[51] Jones G, Berthouze N, Bielski R, Julier S. Towards a situated, multimodal interface for multiple UAV control. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA); 3-8 May 2010; Anchorage. Alaska; 2010. p. 1739-1744.

[52] Hernández-Vela A, Bautista M Á, Perez-Sala X, Ponce-López V, Escalera S, Baró X, Pujol O, Angulo C. Probability-based Dynamic Time Warping and Bag-of-Visual-and-Depth-Words for Human Gesture Recognition in RGB-D. Pattern Recognition Letters. 2014;50:112-121.

[53] Salvador S, Chan P. Toward accurate dynamic time warping in linear time and space. Intelligent Data Analysis. 2007;11:561-580.

[54] Corradini A. Dynamic time warping for off-line recognition of a small gesture vocabulary. In Proceedings of the IEEE/ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems; Vancouver. Canada ; 2001. p. 82-89.

[55] Shin S. Emgu CV Essentials. Birmingham: Packt Publishing; 2013.

[56] Navarro-Guerrero N, Weber C, Schroeter P, Wermter S. Real-world reinforcement learning for autonomous humanoid robot docking. Robotics and Autonomous Systems. 2012; 60:1400-1407.