Classifying the Capabilities of Robotic Systems

What is a robot?

Manal Linjawi

Supervisor: Prof. Roger Moore

Advisor: Dr. Amanda Sharkey

Department of Computer Science

University of Sheffield

This thesis submitted for the degree of

Doctor of Philosophy

July 2020

I would like to dedicate this thesis to my loving daughters …

Declaration

I declare that this report is composed by myself and the work contained herein is my own except where explicitly stated otherwise in the text. This work has not been submitted for any other degree or any other university. I have designed the framework and collected the dataset described in chapter 3 and used in the application in chapter 6.4. Parts of the work presented in chapter 4, 5, 6, and 7 have been published in conference proceedings as given below:

1. Linjawi, M., and Moore, R. K. (2017). A proposed structure to capture the operational and technical capabilities of different robots. In *UK-RAS (Robotics and Autonomous Systems),* pages 127–129.

2. Linjawi, M., and Moore, R. K. (2018). Towards a comprehensive taxonomy for characterizing robots. In *Annual Conference Towards Autonomous Robotic Systems,* pages 381–392. Springer.

3. Linjawi, M., and Moore, R. K. (2019). Evaluating ToRCH Structure for Characterizing Robots. In *Annual Conference Towards Autonomous Robotic Systems,* pages 319–330. Springer.

Manal Linjawi
July 2020

Acknowledgements

For this presented thesis,

I would like to express my deep appreciation to Prof. Roger Moore for all the support throughout my research. Without him, I would never have received so much guidance at every stage of my PhD. His motivation was very crucial and priceless.

Also, I would like to acknowledge that chapter 3 and chapter 4 were developed based on consultation with some very helpful professional roboticists, such as David Bisset from EU robotics and many others as they have added to and corrected me with their specific suggestions.

A special thanks to my friends in Sheffield Robotics for all of the ideas, discussions and good moments that enthused me throughout this journey.

I would like to thank my sponsor, the government of Saudi Arabia, for giving me the greatest opportunity to collaborate in the development of the robotic domain.

Finally, I would like to express my deepest thanks to my daughters, mother and the whole family who have supported and encouraged me with their immense understanding.

Abstract

There are various types of robots, yet there are no defined characteristics that relate them to each other. In order to compare robots, a detailed cross-domain classification system is required. The classification needs to be simple enough to be applicable to all robotic fields, yet comprehensive enough to capture robots accurately. The aim of the research reported in this thesis is to develop a novel classification scheme, subsequently named '**ToRCH**' (**To**ward **R**obot **CH**aracterization), that categorizes robots according to their characteristics via a hierarchical structure. The layers of the hierarchy capture robot capabilities, sub-categorizes them and provides appropriate measurement levels. Some capabilities were adopted from the **M**ulti-**A**nnual **R**oad map (**MAR**), that was developed to shape the European research development and innovation program, and the research reported in this thesis first extends MAR in a number of important dimensions. Then the study utilizes the extensive capability layers in ToRCH to characterize a robot's performance in a form defined as the '**R**obot **C**apability **P**rofile' (**RCP**). The RCP helps in designing, developing, deploying and testing a robot for specific applications. It also facilitates the assessment of the best application that matches the specification of any particular robot. Finally, several aspects of ToRCH are evaluated including its structure, its usability and its generated RCPs. The results confirm that ToRCH is able to capture the capabilities of different robots in a way that could answer the question ***'what is a robot?'***.

Table of contents

Chapter 1

Introduction

1.1 What is a robot?

The word “robot” derives from the Slavic word *robota* which means "a slave". This implies that the basic concept in developing any robot is that it should support human activities and ease human tasks (Cangelosi and Schlesinger, 2014). Robots, in reality, are programmed machines that are linked to rapid developments in computer science. The availability of cheap computers, the internet, robot clubs and societies, robot kits and the advancement of computer science, has established a robot bubble. Many individuals, companies, educational sectors and governments have started involving robots in their domains. Therefore, there has been a great diversity in robot developments around the world. These developments intersect with several concepts that define the major characteristics of robots. Firstly, the robot must be a machine. Secondly, the robot must detect its surroundings. Thirdly, the robot must be set with goals. Finally, it must act according to the intentions for which it was developed (Arrick and Stevenson, 2003).

The Robot Institute of America has defined the robot as a reprogrammable machine with a multi-functional manipulator designed by multi-programming motions to perform different types of tasks (Bartneck and Forlizzi, 2004). The International Standards Organization ISO 8373 defines a robot as an automatically controlled, multi-purpose program with a re-programmable manipulator in three axes that can be either fixed to a specific place or movable to perform autonomous industrial applications (Bartneck and Forlizzi, 2004; Devlin, 2018; International Federation of Robotics, 2019). The Oxford English Dictionary defines robots as machines that perform complex sequences to achieve autonomous behavior. This definition refers to the characteristics listed above, but also states that one of the characteristics of a robot is to execute complex programs to perform autonomous or semi-autonomous actions (Cangelosi and Schlesinger, 2014). The Merriam-Webster's Dictionary defines a robot as any machine that looks like a man or any mechanical device that operates autonomously or through remote control to perform superficially in a human way (Duffy, 2003). The International Federation of Robotics (IFR) has defined a service robot as any robot that autonomously or semi-autonomously performs functions that serve humans (Bartneck and Forlizzi, 2004; International Federation of Robotics, 2019). Another definition of a robot is that it is a machine component that looks like a human and performs human activities, and which illustrates a similar component to humans (e.g., controller, body, power, sensors, tools). Robots have also been defined as machines that perform complicated and sequential tasks in a mechanical way that depends on the robot's control system (Dobra, 2014).

The idea of what a robot should perform within its domain influences the perception of what a robot is as well as its intention and actions. This results in no clear definition of what a robot is to everyone. Some definitions state that robots are any programmed machines. This creates two spectra of definitions. The first spectrum is robots with limited receptive ability and less artificial intelligence, such as washing machines and vacuum robots compared to the second spectrum which includes robots with compound kinematics, anthropomorphic locomotion, complex artificial intelligence, and complicated receptive ability such as humanoid robots (Schaefer et al., 2012). The variations in such criteria highlight the importance of characterizing the capabilities of autonomous systems so that robots can be classified.

1.2 Defining, characterizing and classifying robots

General classifications are performed by recognizing similarities and defining patterns, as defined by Sokal (Sokal, 1974). The classification process goes back to the oldest human civilizations, which began organizing and grouping entities (Sokal, 1974). The definition has two important interconnected concepts that lie at the heart of classification: the first concerns how we classify and is related to the sense of perception as understood by psychology and philosophy. It considers the similarities between entities, the criteria for grouping, and the relations between groups. Applying this concept to robot classification emphasizes how robots are similarly defined, the criteria used to group them, and the relationship between the robot groups, including the names given to them. The second concept is a taxonomy, which is the science of classification. It relates to the principles and procedures of classifying entities. In this concept, the taxonomy is concerned with the way in which the classification should be undertaken, according to the subject of the classification, by defining its principles and procedures. This includes considering whether the defined principles, available in the robotic field, are best for its classification or if there are other procedures that can be used and adopted or even need to be developed (Sokal, 1974). Applying the second concept to the robot domain emphasizes the necessity of finding the common principles and dimensions that can be best used for robot definition and classification. This concept also highlights whether there are any procedures that use these principles and dimensions in classifying robots; if there are, then it is necessary to know how to apply them, and if there are not, then it is necessary to know how to develop them.

Different domains use different methods to classify their entities and subjects. Multi-disciplinary approaches are performed among different domains to generalize a standard classification procedure and to develop basic classification theories. Some classification systems specify certain theories and methodologies to describe the classification procedures (Sokal, 1974). Developing the procedure of classification and apply the principle will define, characterize and classify robots. Today, most classifications use computer algorithms to classify and solve analytical and logical problems. Other classifications are based on logic or certain philosophies to generate a tailored methodology to support the principles of the field (Sokal, 1974). This research studies different robot classifications, according to how robots are considered in relation to their characteristics and capabilities. Therefore, the proposed classifications concentrate on characterizing robot properties and capabilities in order both to accurately define them and to characterize them.)

1.3 Classification and its demands

The definition of classification is to categorize objects into groups according to known specific characteristics. These characteristics define the similarities between the objects being categorized. The more characteristics used to define and categorize the objects, the more levels are needed in the classification. Classification requires three key related concepts. The first concept is the structure of groups / classes, organized according to specific principles. The second concept is the attributes needed to define each group or class. The third is the procedure of assigning objects to the groups and classes. There are additionally six systematic properties for any classification: process, boundaries, membership, criteria for assignment, typicality and structure (Jacob, 2004).

One of the main reasons for any classification is to organize objects, allocate them and retrieve them. Libraries, for instance, must organize their contents in such a way to ensure easy retrieval of information/journals/documents/books etc. Therefore, much work has gone into how to classify library materials for clear storage and retrieval. Several classification approaches exist, some of which depended on specific catalogs and principles that reflected directly on the type of information or document, an example of such a classification is the bibliographic classifications. Other classifications were based on specific logic, studies and research, purpose or human protocol and agreements. For example, the Dewey Decimal classification used in many libraries, organizes according to the field and division of knowledge. Each field is presented through a number. In contrast, the National Library of Medicine classification (NLM), is a specialist classification for medical aspects, which presents them in indexes according to the type of knowledge. Some librarians might prefer one over the other, but also the content of the library influences the choice of categorization, such as if the library contained general information or specific.

Usually, any classification involves several indicators. Some indicators represent the structure and the hierarchy of classes and sub-classes. Another indicator reflects the exclusiveness and comprehensiveness of the classified objects. Other classification indicators specify the narrow or broad perspective of the classification, where the narrow perspective covers the classification of the subject itself and the broad perspective depends on the relation between concepts of the classified objects. Additionally, one of the most important indicators is the clarity of a classification for grouping objects and their aggregations. Also, it depends on the simplicity and the ease of applying its procedures and functions. These indicators implicitly specify to some extent the classification popularity (Da Graça Simões et al., 2018).

One of the most popular scientific classification system was the classification of animals and living organisms developed in the nineteenth century by Linnaeus (1758). This classification divides and subdivides all living organisms from the general into detailed, specific groups. The Linnaeus system is a hierarchical classification system developed for living organisms. It has seven levels, as each level is called a taxon (plural taxa), which is from the ancient Greek (taxis), which means 'arrangement'. As each level is part of the level set above it, it becomes more specific with more characteristics to describe and differentiate its organisms. Therefore, organisms are gathered into taxa and given a taxonomic rank within this hierarchy. The system as it is used today begins with three domains Archaea, Bacteria, and Eukarya. Each domain is divided further into subdivisions until it becomes specific species. The levels are Kingdom, Phylum, Class, Order, Family, Genus, and Species (Polaszek, 2010).

The plant classification is similarly to the animal kingdom classification. It follows the same classification structure "System Natura" (Linnaeus, 1758). The fact that some regions have called a plant a different name from other regions leads to multi-naming and could cause confusion, hence, the importance of classifying a plant clearly. Nevertheless, some plants are difficult to define and classify, such as the cryptogram (Esser, 1982). The plants classification goes from Kingdom (the overall umbrella), then division, class, order, to the variety (as the lowest level with a lot of specifications). Cryptograms generate spores for reproduction, thus belong to the kingdom of fungi, but have special characteristics that would fit within the kingdom of plants. Additionally, they do not have stems, roots, leaves, flowers or even seeds but they have reproductive hidden organs and are sometimes described as organisms, sometimes as plants. Other times they were included in the fungi kingdom, but most of the time they are studied under the plant kingdom with specific methods for classifying them (Awasthi, 2005).

Therefore, any classification must define and form the foundation of all related characteristics of the objects that are being classified. The related aspects used for organizing the objects and ultimately in grouping them, provide some standardization and a consistent approach in classifying the objects. Statistical classification is an example of such a process, as described by Hancock (2013). It lists characteristics or attributes to be defined by the classifier. These characteristics distinguish the categories from each other. Some of the characteristics that could be common among different fields are listed as follow:

1. Classification components: Each classification needs to define the following components: (a) classification title, (b) classification version, (c) classification unit, (d) classification level.

2. The classification keys: Any classification has a predefined set of questions to determine the group of the classified objects, known as classification keys. Classification keys list all the questions to ensure that the object is classified correctly, accurately and consistently. The classification key should be able to include more questions as necessary in case of updating the classification to ensure consistency within the added groups.

3. Custodian (developer) of the classification: Responsible for maintaining the classification and its development, and presenting the classification to other countries, users and agencies. The custodian should support the classification by giving advice on the classification, listing the professional procedures and its best practice, reviewing the classification for adding new concepts and inclusion of classification reviews.

4. Classification conceptual basis: The assumption that the classification should be clear, on agreed concepts and with broad acceptance. The concept should cover all the main and detailed aspects. The main concepts should be explained and described, including why it was undertaken. These explanatory notes represent to the user the conceptual principle of the classification that would help in illustrating what the classification performs, what it categorizes, and why.

5. Classification structure:

- The flat, or linear, classification, contains all the different categories in one layer. The listing does not need to be aggregated into specific smaller classes or clusters; therefore, the groups are listed in one level. Characteristics of a single group cannot overlap with another group, which makes them mutually exclusive. This type of structure is usually exhaustive to include every element.
- The hierarchical classification contains more than one layer as some groups need to be aggregated into smaller clusters, each with some detailed specifications. This type of structure contains all the general aspects within the top levels and become more detailed moving down through the lower levels. The lower levels should contain enough characteristics to differentiate between other objects in the classification (Jacob, 2004).

5. Classification type:

- The reference classification, also known as international classification, depends on developing a framework to collect and group objects for future retrieval. Its development depends on the direct adaptation of a currently known series of actions that generate a common procedure and facilitate it. This type of classification depends on comprehensive consultation, being accepted broadly and agreed on officially. Most commonly it is general and could be legally changed for specific circumstances.
- The derived classification is created depending on the reference classification. It can be generated either by applying the concept of the reference classification in a different way, such as taking the reference classification as a main categorization and developing some other aspect, or by rearranging the reference classification to create a new and different classification.

6. Classification characteristics: Classification needs to be exhaustive to capture any possible object. The classification also needs to be mutually exclusive where each objects should only be part of one group and not multiple groups.

In general, to classify any robot, a classification procedure specifically for them is required. Such a classification could adopt and contain all the classification specifications mentioned earlier, and that need to be mandatorily clarified. Any developed classification raises several challenges. One of the main challenges is whether the categorization and the groupings make sense and provide the same classes. Also defining why, the classification is categorized in this way and to clarify to the user why it is structured accordingly. Thirdly, to verify whether the illustrated categories, types, levels or scores/units are correct or not. Finally, this classification should be able to be updated whilst maintaining the integrity of the classification, especially when it is applied to the robotic domain, which is constantly in development.

1.4 Challenges and limitations

Robot classification is based on defining the labels that can be given to a robot. Constructing these labels requires defining what needs to be labeled in a robot. These labels are known as characteristics of robot capabilities. They are the dimensions that are used in performing the definition of a robot. The field of robotics is comprehensive, multidisciplinary and complicated. Every day, a new robot is developed with advanced capabilities to be used in different fields and unexpected applications. Therefore, in order to define robots, the classifying dimensions and robot characteristics should be applicable to all the different fields, types and sub-types in all domains, which is considered one of the main challenges of this research. This is especially the case given that robot characteristics vary according to the field and this wide variation is hard to capture in a single list or procedure.

Defining and classifying robots according to their characteristics requires each of the characteristics to be available in the taxonomy dimensions. These robot characteristic dimensions are not clearly established because they are increasing as the field of robotics develops.

On the other hand, classifying robots according to a specific feature would lead to overlap when it includes other sets of characteristics. For example, classifying a manufacturing robot with manipulation characteristics that contain some social aspects would mean that it is considered as manufacturing robot interacting with humans. Therefore, it is considered as a social robot, even though it does not perform any social task capabilities nor does it perform social behavior. This ambiguity of robot definitions and robot classifications from different fields is considered one of the main challenges of this research.

Another significant challenge in defining a robot is that each robot has to fit within a specific section of the classification according to the criteria being applied, but there are some robots that could be assigned to different sections of the classification at the same time, which complicates the implementation of the classification.

Therefore, clear characteristics and dimensions need to be set in order to define robots accurately and consequently, to classify them correctly. In addition, developing new dimensions is essential for the robot domain as they should not overlap while describing various types of robots from different fields.

1.5 Problem statement

Current classifications are both ambiguous and partial as they are specific to their field. Different fields cover various dimensions and measurements. For example, social robots are defined as such if they perform their tasks alongside a human, even if they do not perform any social interaction. But social robot classifications are categorized according to the complexity of the interaction or to how well the robot can support social skills. Therefore, social robots performing around humans without any interaction cannot use these classifications. In contrast, as another example, industrial robots are classified more clearly, as their characteristics are widely used within industry and research, yet the defined dimensions are not applicable when applied to other robotics fields, so there is a lack of compatibility. Therefore, there is no universal taxonomy for defining and classifying robots across domains. Additionally, even if a usable classification exists in some robotics fields, classifications become incomplete and outdated as roboticists constantly develop new characteristics. Defining robots is problematic, especially when hardware or software is changed or upgraded. A new classification with clearly defined dimensions is necessary to evaluate a robot's capabilities and it should be applicable to all robots. The creation of these dimensions requires analysis to identify the differences and similarities between robots, which can then be placed within a usable framework.

1.6 Aim and objectives

The aim of this research is to define the dimensions required for characterizing and describing a robot, no matter what domain the robot belongs to. Characterizing the capabilities of autonomous systems in such a general way requires defining dimensions. These dimensions need to be in a single framework. The framework should be able to capture all robot characteristics and to present them in a specific format that enables detailed robot description, characterization, classification and comparability. The framework should also allocate the developed dimensions within its parts according to their relation to each other. Hence, in order to achieve a framework for such a classification, a detailed capability description of robots should be captured to characterize and define robot performance in any field.

This research is intended to answer the questions: Can an overarching framework for identifying, classifying and comparing robots across all possible fields and domains be developed? If so, what are the dimensions needed within such a framework? Can robots be defined through the dimensions of this framework? What steps are required to define a robot through this framework?

In order to answer these questions the research objectives were as follows:

- to understand the characteristics of existing robots by capturing an inventory of representative robots.
- to identify dimensions that are available in different domains and fields and that are applicable to all robot types.
- to design a framework that captures all the dimensions.
- to adopt measures when they exist and develop ones where they do not exist.
- to describe the utilization of the framework.
- to explore the framework applications.
- to evaluate the framework.

1.7 Contributions

This robot classification research, as described in this thesis, makes several important contributions:

- A framework, defined as ‘ToRCH’, [1] that covers most of the robot characteristics which comprise robot features and robot capabilities. Each capability is presented with various levels.

- A tool that presents robot capabilities in a profile. The robot capabilities profile, abbreviated as (RCP) would help roboticists to present their robot capabilities in the form of a simple notation, which is also being used to compare robots.

- A list of capabilities in defining robots and identifying their levels of capability.

[1] This is different from Torch, the open-source library for machine learning and scientific computing

- An application that practically implements ToRCH, as 'proof of concept'.
- A demonstration of the ToRCH framework (including its levels, dimensions and measures) to map between the robotic domain and the application domain.
- A classification that collates various robot capabilities and presents them with several levels within different layers of a framework. The classification defines which capabilities are required for specific robots such as manufacturing robots, social robots, simulated robots and reconfigurable robots.

1.8 Publications

- Linjawi, M., and Moore, R. K. (2017). A proposed structure to capture the operational and technical capabilities of different robots. In *UK-RAS (Robotics and Autonomous Systems)* pages 127–129.
- Linjawi, M., and Moore, R. K. (2018). Towards a comprehensive taxonomy for characterizing robots. In *Annual Conference Towards Autonomous Robotic Systems,* pages 381–392. Springer.
- Linjawi, M., and Moore, R. K. (2019). Evaluating ToRCH Structure for Characterizing Robots. In *Annual Conference Towards Autonomous Robotic Systems,* pages 319–330. Springer.

1.9 Thesis overview

Chapter 2: Background and related work

This chapter begins by listing the most popular types of robots according to their application area and related field. The chapter also lists robots according to their builtin technology. Then it lists the best-known robotic classification criteria and system abilities according to requirements. The chapter pinpoints some of the main issues and gaps in related works, and it highlights possible new classifications.

Chapter 3: Developing the conceptual framework

This chapter illustrates how the system was formed and evolved. In order to understand the robot system, we must present it through a defined framework. This chapter presents the methods that were applied to the structure from its formation to its final configuration. A key part of the structure formation involved known methodologies, such as artificial intelligence, performance, capabilities and interactions (such as proximate and ultimate). This chapter demonstrates the concepts, methodologies and the developmental phases of the conceptual framework's hierarchy.

Chapter 4: Beyond MAR

This chapter introduces the capabilities that are not available in MAR but which are required in the conceptual framework. The capabilities are highlighted, described and developed according to different stages. The second section of the chapter introduces the different measures that were used in each section of the conceptual framework.

Chapter 5: ToRCH

This chapter describes ToRCH; its structure, layers, capabilities relations and levels. It presents ToRCH as a straight foreword yet comprehensive framework. The chapter includes the quality criteria of the structure and defines how to add a new capability to ToRCH. The chapter also describes an automated version of ToRCH as proof of concept. Some content of this chapter was based on work undertaken by the author, which has been published elsewhere (Linjawi and Moore, 2017, 2018).

Chapter 6: ToRCH Applications

This chapter adopts the ToRCH structure as a tool to generate a robot capability profile (RCP). It demonstrates some robot capability profiles as examples of its usage. Some content of this chapter was based on work undertaken by the author, which has been published elsewhere (Linjawi and Moore, 2018).

Chapter 7: Evaluation of ToRCH

This chapter evaluates the ToRCH framework from different perspectives. The first section evaluates the structure of ToRCH. The second section assesses ToRCH as a tool. The third section evaluates RCP as the outcome of ToRCH. Some content of this chapter was based on work undertaken by the author, which has been published elsewhere (Linjawi and Moore, 2019a)

Chapter 8: Conclusion

This chapter includes a summary of the ToRCH framework, its implications and possible future work.

Chapter 2

Types of Robots

2.1 Introduction

This chapter presents alternative classifications, one based on the application domain, another on the type of technology used in robotics, then it discusses robot social characteristics, next it reviews some of the previous attempts to classify robots and finally, it revises systems abilities and their requirements.

2.2 Robot classification according to the application areas

The most used and presented robot classification depends on the robot application which is designed for a specific field or service. The most common robots used for applications are industrial robots, healthcare robots, assistive robots, educational robots, domestic robots, search and rescue robots, underwater robots, space robots, robotics in agriculture, robotics in construction, robotics in hazards environments, and so on. For each application, there are certain types of robots used with specific characteristics that are required by the field or because of the service it provides (Siciliano and Khatib, 2008). The following section illustrates some of the robotics application areas, according to the fields :

2.2.1 Industrial robots

Industrial robots are very important in manufacturing. They are the most widespread robots and the oldest in use. All basic robot controls and structures have been developed within industrial robots. The successful implementation of a full robot system has returned numerous advantages to the manufacturers that have adopted them. They provide the manufacturers with robust productivity, constancy in production lines, accuracy in each product and reduction of manufacturing costs. They also enable automation, which can be seen, for example, in car assembly, power train assembly and engine assembly. All industrial robots, such as the Kuka robot, consist of several fundamental elements: sensors, controller and tools (which might be known as end factors or peripherals). From the industrial perspective, these robots are developed according to the manufacturer specifications and the industrial parameters, which are described as follows (Siciliano and Khatib, 2008):

- Kinematics: each robot has a mechanical structure that determines its mechanism. The attachment of the parts of the robot through the joint defines its possible motion. There are several basic mechanical structure types used in industrial robots, where each industrial robot must belong to one of them. Each type has its own set of kinematic properties that determines the joint motion of the robot with specific computational methods. These mechanical structures, known as kinematics types, have several dimensions that indicate their movement within the area of its workspace. Some of these kinematics types are cartesian robots, articulated robots, cylindrical robots, parallel robots, SCARA robots, spherical/polar robots, etc.
- The shape of the working envelope: also known as the robot reach or the form of workspace. It is the maximum distance that the robot can extend its arm and reach a certain point. The robot type has a big impact on its working envelope. Some of the known working envelope shapes are the revolute envelope, the cube-shaped envelope (used for cartesian robots), the cylindrical envelope (used for cylindrical robots), and the spherical envelope (used for a robot that needs a full or half-spherical area, such as polar and articulated robots).
- Drive system actuator: where actuators are powered either by electronic motors, hydraulic motors or pneumatic motors.
- Number of axes: which is also known as the degree of freedom (DOF). Most industrial arm robots contain six axes of motion and freedom. The DOF defines the number and type of rotary joints. The robot user can specify the range of values the axes can cover by defining the X-axis, Y-axis and Z-axis and how much the robot moves along and around these axes.
- Load capacity: also known as the payload. This can be defined by the minimal load and maximal load object weight value that the robot can manipulate.
- Repeatability: defined as the time it takes the robot to repeat the task.
- Motion range: as the maximum degree that can be reached by the robot wrist.

- Wrist speed: to measures the number of wrists moves per second.
- Acceleration and speed: to show how fast the robot can reach its required position. The acceleration indicates how fast the robot axes can accelerate.

There are other important parameters that are used in industry to specify their robots and distinguish them from each other, such as dimensions; including height, weight and size; autonomy, batteries and internal/external sensors (Robotnik, 2018).

2.2.2 Rehabilitation and healthcare robots

The rehabilitation and healthcare rehabilitation robot is designed to assist people either by providing essential activities relating to their disability or functionality that improves a person's cognitive or physical ability. The rehabilitation robot can be categorized into several parts. These categorizations are illustrated in Figure 2.1 (Siciliano and Khatib, 2008).

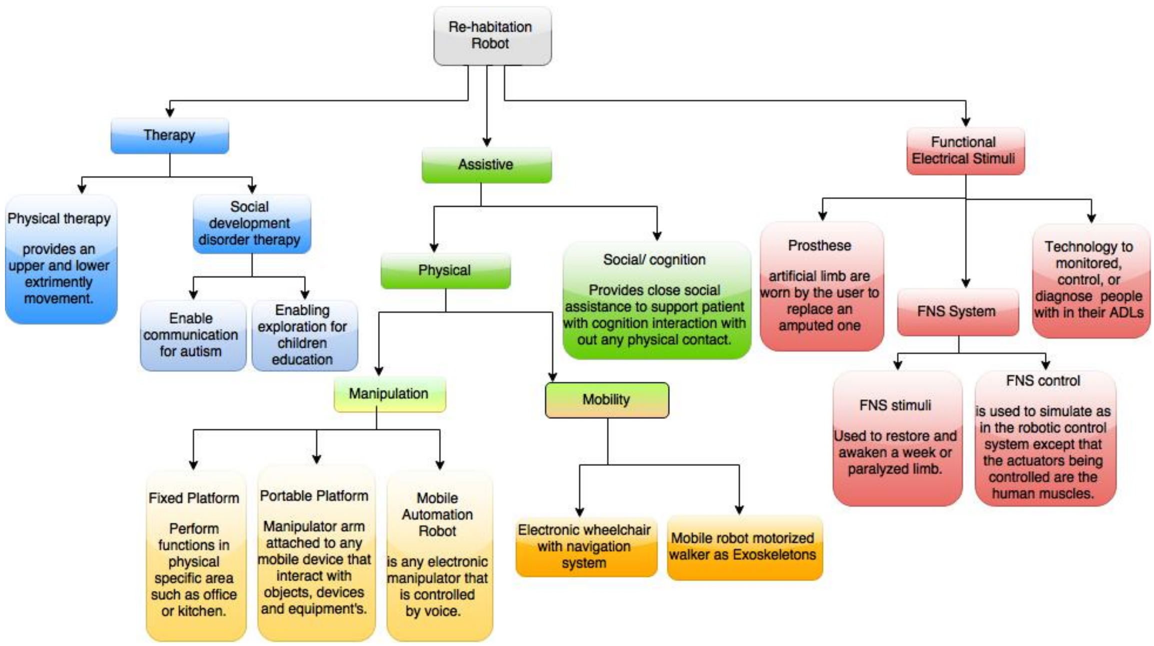

Figure 2.1: Rehabilitation robot classification (Siciliano and Khatib, 2008), as described on page 1224 of the Handbook of robotics.

The first category in the rehabilitation robot is the robot that is used in therapy. Any therapist robot has two users: the patient and the human therapists. Once the rehabilitation robot is installed correctly, it can performs the therapy for a long periods of time without any exhaustion. It also measures the treatment better than human therapists since it contains sophisticated scales to measure the treatments and present the effects significantly. The second category of the assistive robotics in the rehabilitation robot is the assistive robotics system that provides physical, social or cognitive support. The third category of the classification concerns the provision of an electronic stimulation, either through prostheses or using specific techniques, that enable the patient to perform everyday activities (Siciliano and Khatib, 2008).

There is another classification that depends on the data quality presented by the assistive system. This describes the data quality of ten dimensions used by the assistive system. The dimensions are body function, activities, participation, data type, functionality, source of data, the destination of data, interoperability, data distribution and type of response (Beevi et al., 2015). These dimensions help in making comparisons between the assistive systems.

2.2.3 Medical robots and computer integrated surgery

The healthcare medical robot is part of the rapidly growing application-driven research field. The medical robot system can be defined as tools that support surgery with an information infrastructure controlled by a human and via computer. Therefore, such robots are known as surgical assistance robots. Its main goals are not to replace the surgeon, but to provide him or her with additional support, such as safety, efficiency, technical capabilities, consistency and reduced patient morbidity. The robotic surgical assistance is divided into two categories (Siciliano and Khatib, 2008):

- Surgeon extender robots, such as the DaVinci system (Bodnera et al., 2005), are developed to overcome some of the surgeon's limitations. They provide the surgeon with additional perception and supplementary dexterous manipulation during surgery. They also eliminate hand tremor by substituting it with high precision in the surgical instruments. They assist the surgeon to perform operations remotely. Finally, they reduce the time of operations.
- Auxiliary surgical support robots, such as the Automated Endoscopic System for Optimal Positioning (AESOP), are developed to work with the surgeon and perform routine tasks that are usually performed by the surgeon's assistant nurse. Among the benefits of these robots are the reduction of the number of people in the operating theater, improved task performance, provision of extra safety and enabling the surgeon to have more control over the surgery by providing instructions that can be followed precisely during the operation.

2.2.4 Robots for education

Advances in robot technology have delivered successful robots designed for education. The main purpose of these robots is to support learning and inspire learners. Robot programmers aim to provide state-of-the-art educational supporting tools (Siciliano and Khatib, 2008). The educational robot provides several learning methods that can support different teaching roles in the educational sector:

- To enable collaboration in learning by, for example, providing consistent and durable learning material. The social robot is the best robot for this role since it can inspire learning and provide defined expectations.
- To provide inexpensive hands-on robotic materials and methods for the new generation of robotics programmers, where the robot in this role is presented as a programmable project.
- The robot is in the center of learning in the robotics field. This role is expensive and the researcher is considered the main individual that programs the robot with new capabilities.
- To support active learning, such as in museums. This role involves engaging with significant educational concepts and improving educational standards for students.

The success of educational robots has been demonstrated by several evaluation methods that have measured their effectiveness, from different aspects, in the education sector (Siciliano and Khatib, 2008). To support educational robots, the educational sector, IEEE, and AAAI have established competitions for pupils at middle and high school level. These robotics tournaments often take place at conferences and in universities. They have several goals, such as to attract young students to the robotics field, to introduce individuals to a fascinating activity, to improve their robotics knowledge and skills, and inspire them about robotics. These competitions also support learning activities, such as the best way to develop a method for specific robot performance and the best way to program a capability within a robot, which assists robotics educators in presenting, developing and evaluating the efficiency and the effectiveness of a robot (Siciliano and Khatib, 2008).

2.2.5 Domestic robots and smart homes

Etymologically, the word robot means to help people in their everyday activities, especially in performing unpleasant tasks. Domestic cleaning robots can vacuum, mow the lawn and do the ironing. Robotics has also advanced in the direction of improving the concept of smart homes. There have been new dimensions for characterizing the domestic cleaning robot according to the task (Siciliano and Khatib, 2008): control of the workplace, complexity, scale, size, dimension (2D or 3D) and type of surface on which the robot works (even, uneven, horizontal, or vertical). There are also some technical factors that have challenged the growth of domestic robots. Some examples of these factors are: finding the exact absolute positioning; distinguishing between processed and the unprocessed areas; availability of obstacle avoidance sensors; coordination between multi-robots; and ensuring that relevant safety measures are in place (Siciliano and Khatib, 2008). There are two other basic divisions of domestic robots: domestic appliances and smart homes.

- Domestic appliances refer to robotic appliance systems that contain sensors, actuators and smart controls, which are very important elements in domestic robotics. Some of the domestic appliance robotic systems are the ironing robot (which irons any type of trousers by blowing very hot air on the silky lining attached to the trousers, thereby removing its crease), intelligent refrigerators (which contain an Internet connection that enables some communication functionality) and smart wardrobes (which can select clothing appropriate to the occasion) (Siciliano and Khatib, 2008).
- The smart homes presents various technological developments in the home. Smart homes are equipped with technology that anticipates and responds to their residents. The aims are to achieve the comfort of the inhabitants to deliver additional security and to provide them with the entertainment they desire. In order to perform these features, the system should be provided with special robotic equipment; such as sensors, video observing, motion indicators, actuators, pressure pads, numerous health monitoring systems and good communication technology. The main goal of these homes exceeds the concept of home automation, since it is intended to improve the quality of life for the elderly/sick people and to help them live independently and with dignity by providing essential services. Some of these services are:
  - Using a key-less system and fingerprint sensors.
  - Performing basic functions, such as turning on a smart display, lights and air-conditioning. The ability to open curtains, doors or to water the plants and feed pets upon resident request via mobile phone.
  - The ability to perform location and motion tracking either by smart floor padding or camera detection.
  - Smart microwaves (adjusting measures for specific foods/amounts) and smart refrigerators (monitoring of food availability, consumption and expiration, with the creation of and can create a shopping list accordingly).

- The ability to track activities within the house and among its residents either by social interaction technology or by an application-centered shared context concept.
- Indicating health situations through health sensors for the elderly and patients; and updating their current status with their physicians and relatives.
- Some smart homes might contain robotic rooms. The rooms react to the behavior of the patient for their interactions. They recognize, measure and express the patient's behavior and act accordingly (Siciliano and Khatib, 2008).

2.2.6 Search and rescue robotics

Remote robotic handling systems have been developed for hazardous applications. This development began within nuclear remote management operations, which used engineered systems to keep humans away from danger. Hazardous environments require humans to jeopardize their lives to perform risky missions. The availability of robots with special capabilities to accomplish specific tasks saves humans from danger, especially when the magnitude of the hazards reaches a point that threatens a human physically or causes a long-term medical reaction. The remote handling system allows humans to work successfully in a protected location away from exposure to the hazard (Siciliano and Khatib, 2008).

There have been many different natural hazard areas that use these types of robots. Very dangerous environments (such as radioactive explosions, toxicity threats, bomb explosions or nuclear radiation) need modern robotic technologies. Also, rescue operations (such as in firefighting, removal of nuclear or chemical contamination, reactor breakdown or collapsed mines) require the use of rescue robots.

Some rescue operations require a real-time video and special sensory system to present the situation correctly, which have been designed mostly for these rescue robots. The rescue robot performs rescue missions, especially when natural disasters have occurred. The goal is to locate survivors as quickly as possible and move towards them carefully without adding any injury. This can involve either interacting directly with the survivor or victim, the structure around the survivor, or perform some survivor-supporting activity. Such robots contain remote handling systems that enable many subsystems to perform their tasks. These subsystems involve different aspects of capabilities within the robots. There are two main ways that the remote handling system can operate in hazardous environments through telepresence or through tele-operated technology. Using either of them provides human accessibility to the environment and allows an effective rescue operation. Designing this robotic system should consider the nature of the environment and should be designed in a way to avoid any mission failure. There are several factors for the robot to fail such missions, one of the most likely is the equipment failure within the system itself. The system should be capable, at least, to perform maintenance on its own parts or to be able to perform the required maintenance on the system remotely. Such aspects require extensive training to overcome any unconsidered obstacles. There have been several competitions around the world to support this area and discover new technology for it.

There are three categories for classifying this type of robot: modality, size, and task. The first type defines the modality of the robot performance environment according to whether it is aerial, ground, underwater or water surface, and so on. The second type defines the size of the robot as man-backpack if one person could carry it, man-portable robot if two people can carry it, and maxi-rescue robot for very large robots that probably need a trailer to be transported (Siciliano and Khatib, 2008).

The robotic domain is expanding so quickly, that there are many other field robots that have not been covered in this literature: intelligent vehicles, mining robots, robots in construction, robots in agriculture, space robots and systems, aerial robots, underwater robots and many others. Each of these field robots has specific capabilities required in its field of applications. Therefore, these robots have different characteristics and specifications according to the field they belong to.

2.3 Robot classification according to technology

Robots are also classified according to the technology type, which are used in their implementation. Different types of robot technology have different characteristics and features. Some of these robot types are described in the following sections.

2.3.1 Software robot

Software robot are designed to imitate human actions with applications to automate the human process. Some of these robots are:

Web robots

Web robots automatically and continuously run tasks over the Internet; they are also known as spiders, wanderers and crawlers. They automatically gather sophisticated data and analyze specific performance on websites and servers. They are classified according to their functionality (which presents as their crawling behavior and the tasks performed by their programs) or according to the types of requests and information they collect (such as web, text, image, audio, document). Some other web robots, known as bots, can interact with Internet users through messaging or chat, but they perform their required tasks purely on digital data (Doran and Gokhale, 2011, 2012; Heylighen, 1999).

Robotic Process Automation (RPA)

Robotic Process Automation designed with artificial intelligence to automates business and workflow, such as IBM-RPA or blueprism. These robots are developed to increase the automation of work and to mechanize production. They are applied to automate a transactional (buying or selling) rule-based tasks, where the tasks need to have structured data and clear parameters. The RPA act as tools to apply artificial intelligence to the task. The RPA can monitor clients as they perform actions on business application websites and save their actions as a list of repeated desired tasks to be performed in the future for the same setting in specific circumstances, such as invoice request or reorder products. Such routine activities are performed by RPA to reduces time and effort. They connect to the back-end system to execute previously save actions with predefined rules(Vander Aalst et al., 2018).

The following example illustrates some key steps in creating an invoice by the RPA, although these steps are different from one RPA to another. All relative data need to be saved in customer management software (which manages and tracks customer and business interactions). The invoice request needs to be received and the invoice file needs to be created, filled and emailed. The RPA software robot first opens the customer management software and login into checks for any invoice request has been received via email. The robot opens a new email to the invoice requestor to attach the invoice into it. Then it open invoices data presented in an Excel template and scans the data. Then it performs different checks to determine the quality of the data scanned. Then it enters the invoice data from the scanned Excel template into the customer management software that generates the invoice document. Then the created invoice document is filled in the customer management software. Finally, the email is sent with a confirmation and all opened applications are closed. The more advanced the RPA, the more clients it can monitor and handle at the same time.

2.3.2 Multi-agents and swarm robots

Swarm robotics refers to the collection of robots that perform self-organized behavior to mimic animals (colony). Through simple rules and local interactions, they are scalable, robust and have flexible coordination on a large number of robots. their main characteristics are: autonomous, the ability to act to change the environment in which they are situated, communication and sensing capabilities, they are not centrally controlled, and cooperation while performing the required tasks (Brambilla et al., 2013). There are several dimensions that are used to classify robot swarm: collective size, communication aspects such as (communication range, communication topology, communication bandwidth), collective re-configurability, computational resources of miniature robots (Trenkwalder, 2019), processing ability, swarm behavior classification according to learning (Berger et al., 2016), and collective compositions (heterogeneous and homogeneous) behavior (Dudek et al., 1996; Goldberg, 1996). Different classifications are also presented for distributed systems that could be applied to swarm robots, such as classification of the application type (search and rescue, surveillance, transportation, etc.), domain classification (ground, terrain, air, space, underwater, etc.), and classification of the type of interaction seen in the system: collective, cooperative, collaborative (Nickerson, 1997) or coordinative (Farinelli et al., 2004; Parker, 2008).

2.3.3 Modular robots

Modular robots are not built for specific tasks or applications. They are designed to perform a variety of tasks as they are able to reconfigure their shape and properties, allowing them to have a different morphology and locomotion method. Most modular robots are mechanically simple, easy to manufacture, reliable, simple to maintain, and relatively cheap. They are usually classified according to the geometry of the system, such as chains, lattices and hybrid (Chennareddy et al., 2017; Gilpin and Rus, 2010). They are composed of several parts, known as building blocks, that are designed to be rearranged with different connections. The building blocks are made of a structured unit with docking interfaces for connection, communication and sometimes power-sharing. They can also include modules for manipulation (grasping or holding), sensing (camera or otherwise) or locomotion, such as wheels or legs (Yim et al., 2007).

Modular robots are either reconfigurable or self-reconfigurable. The reconfigurable requires the user to alter its morphology, where the self-reconfigurable has the capabilities to alter its morphology autonomously. In order to be fully autonomous, systems need to have self-assembling, self-reconfiguring, self-disassembling, self-repairing and self-organizing capabilities (Gilpin and Rus, 2010).

2.3.4 Soft robots

Most robots are made of hard materials that provide rigidity to support repeatability. In contrast, 'soft' robots are constructed to perform various behaviors in unknown environments, and may need to touch or manipulate un-known objects without damaging them, or touch and interact with humans safely (Pfeifer et al., 2013). In general, there is a common understanding 'soft' robotics could refer to the soft external body that contains the subsystems of a traditional robot (such as actuators, driving electronics and a power source) or it could refer to the compliance structure of the combined geometries of several solid materials or even the inherited compliance of the materials (Wang and Iida, 2015). Most soft robots have bodies intended for specific performance, they are soft to mimic biological systems, support various degree of motion freedom and absorb the energy that comes from a collision. These characteristics require some modifications to the subsystem to provide for the desired behavior (Rus and Tolley, 2015). Some of these subsystems are (1) soft materials, which require special designs for super-elasticity, appropriate toughness and suitable resilience (Polygerinos et al., 2017). (2) Flexible and stretchable sensors and electronics (Rus and Tolley, 2015). (3) Modular and efficient actuators. (4) Self-organization and distributed control. (5) Power source and fabrications (Rus and Tolley, 2015).

2.3.5 Bio and biohybrid robots

Bio-robotics is a field that covers the intersection between biology and robots (Webb, 2001). It also covers cybernetics, bionics and genetic engineering (Feinberg, 2015). It refers to the study of how to develop robots that simulate living biological organisms mechanically or chemically. The term is also used, in reverse concept, as a definition for using biological organisms within robots or as part of robots. The implementation of the biological organisms with advanced technology known as a cybernetic organism. Living organisms generate biological functionalities that could not be developed technologically, are used in the creation of robots. These robots are defined as biohybrid robots. The main concept is to combine the biological materials with the artificial materials to provide biological actuation, sensation and control, etc., especially if the functionality cannot be created completely from scratch (Appiah et al., 2019; Morimoto et al., 2018; Wilson et al., 2013, 2015). For example, in developing the bio-hybrid finger, tissue-engineered living muscles are applied to and connected to a

3D printing of a finger-like structure. Applying electric stimulation to the living muscles contracts the motion of the finger and causes it to move. Such systems utilize living muscles as part of the robotic system to provide energy-efficient motion (Morimoto et al., 2018).

2.4 Social robots

There are many definitions for a robot, as discussed in chapter one. Each definition covers specific robot settings. Engelhardt and Edwards (1992) defined robot interactions as smart systems with programmable tools to perform sensing and some thinking, and which act to benefit humans and improve productivity (Bartneck and Forlizzi, 2004). This definition highlights productivity and does not mention any social capabilities or interactions. Yet social aspects and social interactions are developed in all of the social robots in any field. Social interactions are also being developed in most of the robotics labs and researches. Bartneck and Forlizzi (2004) provided a special definition of the social robot, stating it is either an autonomous or semi-autonomous machine that is capable of interaction and communication with humans. To perform successful communication between the robot and humans, the social robot should follow the behavioral norms expected by the people with which it interacts (Bartneck and Forlizzi, 2004; Cangelosi and Schlesinger, 2014). Therefore, the social robot is complex, since it requires a multidisciplinary approach for it to be successfully surrounded by humans (Fong et al., 2003).

Likewise, social robots are defined by Duffy (2003) as "a physical entity embodied in a complex, dynamic and social environment sufficiently empowered to behave in a manner conducive to its own goals and those of its community". According to this definition, in order for a robot to be considered as social, it has to maintain two-directional interactions. This includes any interaction performed either through interaction with others directly or indirect interaction with no explicit communication. The indirect interaction involves mutual recognition of objects through interfaces (Fong et al., 2003). The performance of these interactions depends on several factors within the robot, such as its ability to understand the social structure and its level of embodiment.

Most social robots are designed either to interact with their developers or to interact with specific user groups. For example, some robots are developed for the elderly, whereby they help support some mental and emotional needs; an example is Paro, a social robot that has been used with dementia patients (Kidd et al., 2006). Other social robots have been developed to provide therapy for autistic children. Each of these robot types has its own characteristics, goals, roles, benefits, functionalities, specific hardware and software components (Cabibihan et al., 2013). Therefore, within each user group, the robot needs to have specific capabilities and social behavior. Robots in general are designed either as interaction-oriented robots, task-oriented robots (Kanda and Ishiguro, 2004), behavior base-oriented robot (Duffy et al., 2005) or biologically inspired robots (Fong et al., 2003). Interaction-oriented robots, behavior base-oriented robots and biologically inspired robots must have social interaction within them. Task-based robots in manufacturing are likely to have general social interaction, including the ability of the robot to learn from and to collaborate with humans. This requires social capabilities that are adaptable to different environments, capable of performing social interactions in different areas, and not restricted to a specific domain or to a special user group. Social capabilities should thus enable the robot to perform effectively in the social environment for which it has been developed, whether that is specific or general, where the participants depend on the robot's built-in properties and capabilities.

Socially interactive robots are embodied in many forms and sizes. They can be embodied as human-like to capture human skills, or as life-like (Schmitz, 2011) to present any biologically inspired skills in the robot (Siciliano and Khatib, 2008). They can range from simple embodied entities with elementary functionality, to very advanced embodied agents that perform numerous complex human interactions (Fong et al., 2003). Therefore, all social robots have to some extent to be embodied to enable their performance of any capabilities (McGrenere and Ho, 2000) and interactions. These interactions can be performed either individually, where the robot should act as an active social entity, or as a member of a group, where the robot acts within a social group of heterogeneous members. A robot that has been developed to perform individually was defined as an individual social robot (Winfield, 2012); such robots should recognize other entities, especially those they are communicating with, perform the communication process with them, accumulate memorized actions that have happened, and perform some active learning (Fong et al., 2003). This type of robot performs individual social robot interaction, which is known as human–robot interaction, and they are built to be companions, assistants or servants. These robots have to follow societal patterns and restrict communication to some human conventions. Some of these human conventions can be presented as social learning, imitation, emotion, gesture, language communication, individual acknowledgment and recognition. However, if the robot has been developed to perform as a member of a group, it is defined as a group-oriented social robot (Fong et al., 2003). The group members organize themselves to achieve complex tasks, perform collective behavior patterns or incorporate their physical figures to build a massive construction. This type of robot performs a group social robot interaction, which is known as cooperative interaction (Bartneck and Forlizzi, 2004). Such robots are not designed to present as individuals; instead, they are designed to present social behavior that is obtainable from a collective group of robots. These types of social robots would not adhere to any societal human norms while communicating between each other nor they would have any human communication conventions (Fong et al., 2003).

Socially interactive robots are utilized in industrial, educational and domestic sectors, and many others besides. They are planned and developed to provide services and benefit humans. Some of these sectors require social robots with interaction skills while performing or solving tasks (Salichs et al., 2006). The robot needs to be socially interactive and to perform social skills. These skills are essential to generate a communication model that is as natural as possible for the human channels where the communication needs to be performed spontaneously. Social interactive skills are required whenever the robot is interacting closely with a human, for example when providing entertainment, assisting the elderly, or even changing human performance and behavior (Severinson-Eklundh et al., 2003) such as in autism therapy (Cabibihan et al., 2013). Moreover, the robot needs social interactive skills when remotely communicating with a human to convey information correctly. In general, the range of social interaction degree is captured through social skills requirements that scale from no social skills to an essential degree of social skills. The requirement of social skills depends on the robot application domain, as illustrated in Figure 2.2 (Dautenhahn, 2007).

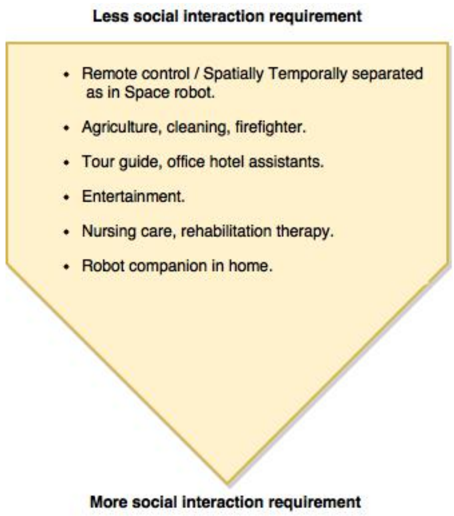

Figure 2.2: Social robot application domains and their levels of interaction (Dautenhahn, 2007), based on figure 4, social skills requirements are increased according to the robot application domains.

Social robots are seen as a bridge between humans and technology. The ease of interaction usability, via the social robot, breaks down the barrier of digitally complicated technology for individuals. This encourages applying social skills in any type of robot designed to interact with humans (Duffy, 2003). Social robots should include human features, known as human–robot interaction features, to indicate how humans will interact with the robot. These features are based on human skills, behavior and cognition to specify how humans interact with each other. These features within robots provide some identification of the interaction level between robots and humans or groups of humans. Some of these features are listed below (Fong et al., 2003):

- Communication through natural language and with high-level dialogue, such as in peer-to-peer communication.
- Perception of social aspects, such as human sensation, activities, behavior, and recognition and tracking of people or their aspects.
- Ability to establish, maintain and terminate social relations.
- Ability to express and perceive emotions.
- Ability to learn, imitate, develop, and conclude social competition.
- Use of natural non-verbal communication indicators, such as body language, gestures, facial expressions, postures, sign language and/or haptic language such as tapping.
- Use of paralinguistic communication, such as tone of voice, accent, inflection, and/or pronunciation (Cowan et al., 2019; Fong et al., 2003).
- Presentation of a distinctive personality and characteristics.

Socially, robots are used in different sectors, such as in research, in games for entertainment, in education, and in advanced therapies. They are as it is becoming more prevalent in most of the robot development and researches (Anzalone et al., 2015). Consequently, social robot users come from various backgrounds and cultures, are of both genders, and cover a large age range. Social robots need, therefore, to be sophisticated with diverse skills to accommodate all these types of users. Most of these social skills are defined, developed and functionally applied within social robots. The following section presents the socially interactive robot characteristics, and it is followed by some social robot design perspectives (Fong et al., 2003).

Emotion

Emotions are very important in order to perform comprehensive interactions and effective communication. People generally interact with a robot in the same way they interact with each other. Therefore, a desirable robot social interaction requires some kind of emotion. Emotions are very complex, since they depend on the type of embodiment, psychological circumstances, and social context. There are three emotional theories defined for humans that can be adopted in robotics (Wundt, 1874).

- The first theory categorizes emotions as discrete categories and defines all the basic emotion types or modes, such as sad, happy, nervous, frightened, and surprised.
- The second theory categorizes emotions on a continuous scale, such as arousal.
- The third theory indicates the importance of both categories and defines their dimensions.

Emotions are presented within social robots by the development of one of the following methodologies (Fong et al., 2003). The first is artificial emotion, which facilitates the interaction between the robot and the human with some kind of reality by providing the robot with an emotional social response to the human. It presents the robot's internal state, such as happy or sad. It also presents feedback on its goals and intentions. These emotions in robots are categorized into models that present moods and these types of emotions in the movement of the robot. The robot is required to have a large degree of freedom to present a moods model with complex consecutive emotional movements (Fong et al., 2003). The second method uses emotion to control preferences among several behavior modes, with a control mechanism, where each mode has a set of planning, learning, adaptation, and actions. Within this methodology, robots are designed so that interaction changes the robot's emotions through computational models of emotions that mimic and modify the robot's internal state and result in it amending its kinematics and actions (Fong et al., 2003). Social emotions are presented in many ways (e.g., emotions can be presented in objects, such as balls with different colors to be used by children, or in textiles with specific notations, which are known as real-time wearable emotions (Lee et al., 2016; Lenzi et al., 2011)). Additionally, social emotion is presented to facilitate a behavior for human–robot interaction, to provide feedback, or as a control mechanism between different behavior modes.

Communication

There are numerous ways to present communication within robots. Some communication depends on the robot type. However, robots that communicate only with other robots are not considered social, as their type of interaction is known as cooperative interaction (Bartneck and Forlizzi, 2004), since they perform data transfer and data communication (e.g., the interaction between modules in reconfigurable robots). Some swarms are referred to as a simple agent, while the rest are considered to present as an insect society (Andina, 2007; Beni, 2004). Therefore, the social insect and its society are referred to as social, and hence they perform social interactions. But, social robots interact with humans need to include social norms, and their form of interaction is known as individual social robot interaction. These interactions are performed either verbally or non-verbally. Verbal communication can include speech, and non-verbal communication can be through body language (such as gestures, facial expressions (Gockley et al., 2006), or sign language). Some of these types were presented in social robot emotions and dialogues by Fong et al. (2003). Figure 2.3 lists the human social communication types that can be adapted to robot communication from human communication (DeVito, 2011).

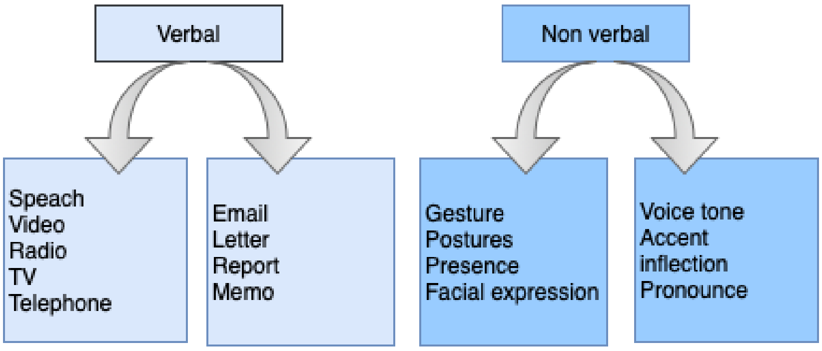

Figure 2.3: Human communication types need to be adopted to robotics domain to present robot communication types, based on DeVito in human communication type (DeVito, 2011; Hewett, 1998)

There are two perspectives that describe the communication channels: the first perspective was presented by Goodrich and Schultz (2007) and illustrates the human-robot communication channels that fall within the following two types (Yanco and Drury, 2002): (1) Remote interaction: the robot and the human are separated and not co-located within the same area; (2) Proximate interaction: the human and the robot are co-located within the same area. The communication type usually depends on the application condition, which requires robot mobility, a robot's physical manipulation or a robot's social interaction. These interactions have been referred to by different terms, as illustrated in Figure 2.4. The remote interaction is denoted as tele-operation when the application requires mobility. It is denoted as tele-manipulation when the application requires physical manipulation (Goodrich and Schultz, 2007), and remote interaction with social interaction applications is denoted as tele-presence (Shinozawa et al., 2003). On the other hand, the proximity interaction is denoted as robot assistance for both the mobility application and the physical manipulation application. It is considered a social robot for the social requirement application that requires proximity interaction (Goodrich and Schultz, 2007; Shinozawa et al., 2003).

The second perspective is to present the communication channels as the robot sensing system, examples of general sensors: light sensor, sound sensor, temperature sensor, contact sensor; proximity sensor: infrared, ultrasound, photoresister; distance sensor: laser and encoder; pressure sensor; navigation/positioning sensors. And some examples of common

Table 2.1: Three types of social interaction as mobility, manipulation and social interaction related to remote and proximity interaction present specific interaction type. The summary table list different interaction related to these two main interaction aspects.

Interaction Application requirement	**Mobility**	**Manipulation**	**Social interaction**
Remote interaction	supervisory control / Tele-operation	Tele-manipulation	Tele-presence
Proximate interaction	robot assistant in wheel chair	robot assistant in integrated surgery	social robot for education and therapy

robotic sensors are: camera, whiskers (Mitchinson et al., 2007), proprioceptive sensing (Salter et al., 2007), active touch (Prescott et al., 2011), active vision (Breazeal et al., 2001). Through the communication channels, the robot can sense its environment, as presented in Figure 2.4. These sensing channels are adopted from human sense concepts, and the robot can contain any of the following sense channels: haptic channel for touch communication, echoic channel for voice/sound communication, and iconic channel for visual communication.

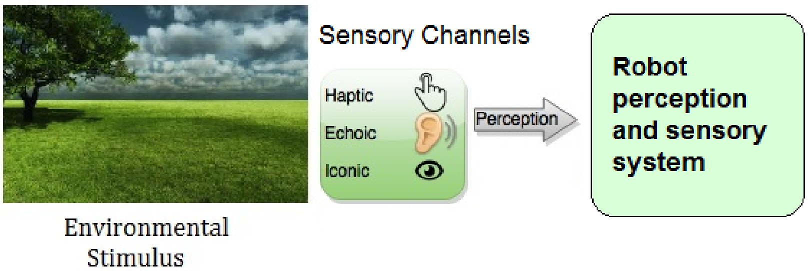

Figure 2.4: Types of communication sense channels.

Dialogue is also used to communicate and share information through common concepts between the robot and the human. The dialogue is presented in three types: The first type is low-level dialogue, according to which the robot is considered autonomous and capable of learning a language. The second type is the ability to learn non-verbal social conventions, such as figure movement, gestures and body status. Moreover, this dialogue concept requires the robot to perform them. Examples of these performances are to shake hands while greeting or to wave a hand while leaving. Presenting some of these non-verbal dialogues allows for social interaction. Through non-verbal conventions, a robot can present itself with social norms, such as how to stand in line and how to walk along a pathway. The third type is natural language dialogue, which allows the robot a range of interactions from its physical capabilities to its socially required features (Fong et al., 2003).

Interaction

There are different interaction classifications presented in the literature. Robots depend on the following interaction aspects:

- **Interaction paradigm**, such as a tool, avatar, cyborg extension, sociable partner (Breazeal, 2003, 2004).
- **Interaction role** such as machine, tools, or assistant companion partner (Dautenhahn, 2007).
- **Time-space taxonomy**, known as CSCW taxonomy, such as synchronous, asynchronous, collocated, or non-collocated (Penichet et al., 2007).
- **Robot interaction presentation/performance** type: hologram, mixed reality, virtual reality, simulated/animated or a real interaction (Kidd and Breazeal, 2004).
- **Human–robot ratio (H:R)** and level of shared interaction among teams. This also includes multi-robot interaction settings as compositions of robot teams, and dimensions for defining the multi-robot systems (Yanco and Drury, 2002).
- **Type of robot interaction with the environment**: coupling, coordination, or communication (Moore, 2016).
- **Type of engagement** in the interactions between the human and the robot: hands, head, body, eyes, facial expression (Cameron et al., 2018), or gestures (Sidner et al., 2005). Interaction can also be verbal or non-verbal, and it can be on the inter-personal, intra-personal, social, or public levels.
- **Interaction contact type** : as direct contact: interacting face to face with the robot (Breazeal, 2003); or as mediated contact: seeing the robot on videos, media, etc (Goodrich et al., 2008).
- Human–robot interaction **focuses on specific social aspects**, such as empathy (Duff et al., 2014), with-me-ness (Lemaignan et al., 2016), expressiveness and attention (Bruce et al., 2002) and many others.

In order for a robot to have good human interaction, it has to be perceived as human and to interact in a human-like way. There are many aspects that need to be built within the robot to increase the human-oriented concept. These aspects will present the robot like a human and optimize the interaction level with a human. This implies that the robot should detect and recognize human social actions (Fong et al., 2003): First, the robot has to implement different human senses, such as tracking human faces and bodies. Second, the robot has to interpret human communication channels with all their varieties, including verbal and non-verbal communication. Third, the robot has to mimic human perception in order to ensure both active and passive perception. The robot needs to guarantee people-tracking perceptions and to include speech, facial gestures, and body recognition in its perception.

Additionally, there are **human–robot interaction views**, presented by Dautenhahn (2007), that describe the three different dimensions related to socially intelligent robots: the human-centered view covers the observer's perspective and elicits certain human perspectives, the robot cognitive center view covers the robot's cognitive and social task capabilities, and the robot-centered view covers the robot's social intentions and behaviors.

Personality

The more the robot has a compelling personality, the more people will love to interact with it. There are five personality types used in robots (Fong et al., 2003):

- The tool-like personality is given to robots that perform functional and command tasks.
- The pet or creature-type personality is usually applied to toys to present animal-like characteristics.
- The cartoon personality is used to exaggerate or to highlight a specific behavior.
- The artificial personality usually contains an artificial mechanism and characteristics, and these types of robots are popular in science fiction movies.
- The human-like personality is used to exhibit human characteristics.

Relationship periods

Some robots are required to develop their interaction skills over time to maintain their relationship with their human partners. Therefore, there are two types of interaction periods between humans and robots. The first type is considered as a short-term relationship, where the interaction is performed over a short period of time, such as the relationship between a robot and visitors to a museum (Fong et al., 2003; Schulte et al., 1999). The second type is considered as a long-term relationship, where the robot has to maintain the relationship between itself and the human partner to achieve their goals (Breazeal, 2004; Dautenhahn, 2004; Falcone et al., 2003; Gockley et al., 2005).

User modeling

Social robots need to interpret social behaviors by applying user models. User models classify users according to subgroups and then perform either quantitative or qualitative analysis on specific parameters. By performing user model concepts, human interaction feedback is evaluated according to the specific user type saved in one of the robot models. This process leads the social robots to incorporate all the required skills in its behavior. Updating the model confirms that each specific user type model is updated with its proper behaviors and skills (Fong et al., 2003).

Socially situated learning

The capability of social robots to learn skills is important, since it increases the chances of successful social interaction and leads to better communication. The learning skill provides the robot with the ability to perform tasks in a better way, since it updates old skills with new ones. There are several human–robot interaction learning techniques (Fong et al., 2003): Generating a sequence of behavior and matching it with available human model behaviors. Tracking the movement sequence of available known behaviors, such as tracking a bird's nest-building movements and adding it to the list of bird behaviors in the memory. Direct copying or imitation. Imitation requires a variety of sense, perception, interpretation, task performance, physical features, and motor capabilities. Even with the right set of imitation capabilities built within it, the robot has to know when and what to imitate. It also has to know how to map imitated actions to behaviors. Finally, it has to know what the goal of the imitation was and whether it was achieved or not.

Intentionality

All social robots must have intentions in order to interact socially. Intentions within robots direct their performance of tasks and behaviors, and they establish the robot's goals. They also indicate the ability to present attention when needed. There are three methods to understand intentions and present them through behaviors and actions (Fong et al., 2003): by the physical stance, where expectations of human intentions are established in the physical features of the robot; by the design stance, where expectations of human intentions are established in the robot design and performance of its actions; and by the intentional stance, which depends on the human's beliefs and wishes to predict robot performance.

Engagement

Engagement is the process of establishing, maintaining and ending physiological connections between two participants. If there had been a high level of engagement in an interaction, the interaction most likely will happen again. The level of engagement in interaction for social robots varies depending on the social structure that is programmed in the robot and how it evolves. There is a wide range of interaction structures available for any communication. Each structure has a specific set of processes that define the interaction capabilities for the robot (Anzalone et al., 2015). Each process within the interaction has to be defined through engagement sub-processes and the amount of its involvement. Examples of engagement processes within any interaction are: directed gaze, mutual facial gaze, emotional engagement, action synchronization, gesture, and tone recognition (Rich et al., 2010; Sidner et al., 2005).

Social skills

Social skills are an important requirement for social robots. Through social skills, most of the HRIs are performed. Social skills are important in some robotics domains more than others. The spectrum of the social skills requirements increases for robots that are around humans and decreases for remote-controlled robots. Therefore, the range of skills is wide for any social robot, but the skills are essential for their performance (Dautenhahn, 2007). Social skills, when applied to an individual social robot, is known as HRI, where is when it is applied to swarm robotics, it is referred to as Human Swarm Interaction, HSI (Bashyal and Venayagamoorthy, 2008).

Relation between the human and the social robot

The relation describes the functionality of the robot and defines their goals and purpose. The relation defines some important social robot characteristic regarding the interaction duration and design specifications (Shibata, 2004).

Social robot classification

Individual social robots can demonstrate a wide range of social behaviors. These behaviors are performed to express and communicate with robots and social agents. There have been several classifications for social robots, but most have the same concepts. The most important classification concepts depend on a robot's ability to support the social model into which it fits. The classification depends on the robot's ability to perform its role and to support the complexity of the interaction within the role. This ability depends on its built-in design features and it's human–robot interaction scenarios. There are two classifications that complement each other, but which also overlap in some concepts (Tzafestas, 2015). In general, these classifications are defined as follows (Breazeal, 2003; Fong et al., 2003; Tzafestas, 2015):

- **Socially evocative robots:** robots that depend on human emotions to 'buy into' their morphology and receive the required emotional response when humans interact with them. This is done by involving the human in creating, developing and maintaining their robot's social structure and activities (Breazeal, 2003; Fong et al., 2003).

- **Social interface** (Fong et al., 2003), also known as **socially communicative** (Breazeal, 2003) robots: robots whose behavior is concerned with human social cues and communication modalities in order to interact with humans, providing a natural way to communicate and interface with humans. These robots communicate with no cognition concepts and act passively in all their interactions. Therefore, their social skills are modeled at the interaction level. There are two types of social interface robot. The first is known as the interface model type, which expresses the interaction through verbal interaction and a GUI interface. The second is the performance model, where the robot is presented as an avatar. The designer of these robots decides which behavior should be applied and when. Therefore, the robot's social behavior depends on the designer's perspective (Breazeal, 2003; Fong et al., 2003).

- **Socially receptive** (Fong et al., 2003) or **responsive** (Breazeal, 2003) robots: this robot acts like the social interface or communicative robot mentioned above, except that it benefits from and builds more social behaviors while interacting with people. It learns and increases its social actions through interaction with others by learning behaviors from humans, enculturating itself in its environment, and performing a successful imitation of its surroundings. This type of robot requires collaborative cognitive modeling, where the interaction needs to modify new actions, such as gestures, that require the internal state of the robot and its motor system to be affected (Breazeal, 2003; Fong et al., 2003).

- **Sociable:** this type of social robot is considered as an agent with internal social aims and goals. Its behavior is a product of computational psychological social behavior. It actively interacts not only to respond to human action, but also to satisfy its own goals, such as to learn, to promote its perspectives, or to improve its performance. The robot evaluates the human interaction at two interaction levels.

- The first interaction level depends on the interface and the interaction that is being performed between the human and the robot.
- The second interaction level depends on the psychological computational programs built within the robot. It requires the robot to have standard human social models within it, to which it then tries to map any human behavior. This mapped model guides the robot on how to interact with the human (Breazeal, 2003; Fong et al., 2003).

- **Socially situated:** this robot socially interacts with its environment on a situational base. The environment contains objects that the robot perceives and reacts to. These robots have to distinguish between social agents, humans, and objects. The robot should also act differently toward each one of them (Dautenhahn, 2007; Fong et al., 2003).

- **Socially embedded:** this robot is structurally coupled with its environment. The social structure is a key element for this type of robot, but the robot has fewer socially interactive skills. It can interact with humans and social agents (Dautenhahn, 2007; Fong et al., 2003).

- **Socially intelligent:** these robots are human mimics and present a lot of human social interaction with different structures. Such robots are considered as intelligent, and their intelligence level depends on cognitive agents. These agents create a human model for the robot's cognition, interaction and competence (Dautenhahn, 2007; Fong et al., 2003).

- **Socially interactive:** this robot has its own diversity of interaction skills controlled by specific cognition architecture. It is designed to engage in interaction and to perform any social skills. Social interaction known in HRI is the main perspective in these robots, such as emotion, dialogue, recognition, natural cues, language, personality, and social competence ability (Dautenhahn, 2007).

2.5 Previous attempts to classify robots

The word robot has been used very frequently in various domains with different applications. There are so many types of robots that it is clearly important to understand how and why they are different. There are several robot classifications available in the robotics field. The following section lists and describes the most general, known and commonly used classifications to differentiate between robots.

- Classifying the robot according to one of its components, such as sensors and actuates, to cover what the robot consists of. This classification defines the hardware and software features within the robot.
- Classifying the robot performance environment according to whether it is on the ground, in the air or in water. Aslo known as the modality of the robot performance (Siciliano and Khatib, 2008).
- Classifying the robot according to the domain or field criteria to which the robot application belongs to field (Siciliano and Khatib, 2008).
- Capability classification; such as perception, mobility (Campion et al., 1993), manipulation (Yousef et al., 2011), or robot intelligence (Bhatnagar et al., 2017; Winfield, 2017); where each has its own set of classification attributes and taxonomy.

Some recent surveys by the United Nation have categorized all types of robots in three categories (Thrun, 2004): The first category is the industrial robot, which is considered the most developed and fastest growing category in robotics. The second category is the professional service robot, which is considered less used but growing very rapidly. The third category is the personal social robot, which is expected to develop very rapidly and to be the most used in the near future (Bartneck and Forlizzi, 2004).

The following section illustrates cross-domain robot characteristics that can be applied to any type of robot:

Robot's position: the robot's position can be either fixed or mobile. In the case of a robot having a fixed position, its position is defined with a fixed frame. But if the robot performs mobility, then defining its movements needs to be calculated relative to several fixed predefined frames. These robots also need to have end-factor sensing that calculates the movement by comparing the new position with the predefined fixed ones. Most industrial robots contain some mathematical calculation relative to the base of their position. These working frame calculations are one of the most important characteristics in the industrial-type mobile robot. Industrial robots are categorized as manual manipulator, fixed sequence, variable sequence, playback or numerical control robots (Dobra, 2014).

Robot application performance domains: the robot can be designed either as an industrial or as a non-industrial application. For industrial applications, the robot can perform any of the industrial capabilities. But the non-industrial robot should be able to perform capabilities that belong to the domain it executes in. This could, for example, be producing material goods for building, mining or agriculture, or it could be providing services such as medicine, transport, trade or tourism (Dobra, 2014). Other classification links the purpose of the robot with the application area: healthcare, education, entertainment, industry, search and rescue, home and workspace, public service, and social science (Baraka et al., 2019). The classification is also presented with different perspective on application type: robot for companionship, robot for personal empowerment, robot for transportation, robot in space, robot for technology research (Baraka et al., 2019).

Robot locomotion: locomotion defines the robot's movement ability from one point to another to perform various tasks. Any robot has to be defined as one of the following types (Dobra, 2014):

- Stationary robots: the workspace and the geometry for the stationary robot manipulator are defined, and the volume of space which the end of the effectors can reach is illustrated. Some examples of these robots are the cartesian robot, where the robot arm is made of three mutually perpendicular prismatic joints; the cylindrical robot, where the first or second joint of a cartesian joint is replaced by a revolution joint; or the spherical robot, where either the first or second joint of a cartesian robot is replaced by a revolution joint (Dobra, 2014).
- Wheeled robots can have one or more wheels according to the application (Dobra, 2014).
- Legged robots can have two or more legs (Dobra, 2014). They are also known as anthropomorphic (bipedal locomotion) or zoomorphic locomotion that is multi-legged (Tzafestas, 2015): quadruped (four legs), hexapod (six legs), or octopod (eight legs).
- Climbing robots are able to climb walls, house roofs, ceilings and geometric structures (Chu et al., 2010).
- Robots can perform other locomotion types, such as swimming, flying, climbing and hopping (Dobra, 2014).

Robot performance environment: can be on ground, aerial, underwater, or space robot (Dobra, 2014).

Robot software and hardware controlling architecture: can be deliberative Sense-Plan-Act (SPA), reactive (Sense-Act), behavior-based such as SPA, SPA...SPA, or hybrid SPA with any sequence (Dobra, 2014; Gat et al., 1998). With the advancement in robotics the capabilities controlling methods became more sophisticated especially in areas such as feedback, open loop and feed foreword (Dobra, 2014; EU-Robotics, 2016; Oswald, 1980)

Control system of the robot: one of the most well-known classifications categorizes the robot into three types: non-servo with open-loop system, servo with closed-loop system, and servo-controlled with closed-loop system with a continuously controlled path (Dobra, 2014). Another control system classification is presented through an architecture that describes the layers of the control system developed in the robot (Brooks, 1986) and its subsumption (Prescott et al., 1999). There is also an additional system performance classification that defines some quantitative measures for mini robots and systems (Steinfeld et al., 2006).

Source of power: the robot can be provided with any of the following power supplies: pneumatic, hydraulic, electrical, energy storage, organic waste, nuclear fusion, radioactive, or even animal/human feces (Dobra, 2014).

Robot's relationship with the human during operation: this covers different types of human-robot relations: automated, biotechnic or interactive (Dobra, 2014).

Robot generations: this characteristic analyzes the robot's evolution and its adaptability levels throughout different generations. This classification provides first, second, third and fourth generations for robot development (Dobra, 2014).

Robot size: this can be either macro-robot, micro-robot, and nano-robot, or small-sized robots of less than 6 kg, medium-sized robots of around 100 kg, and large-sized robots of more than 100 kg. For example, the humanoid robot is considered a medium-sized robot (Dobra, 2014).

Degrees of freedom: of the robot performance: some robots manipulate with six degrees of freedom and others perform with less or more (Richard, 1981).

Robot kinematic structure: according to which the robot is capable of performing an open loop or closed loop either in any of its performance, manipulation or mobility (EU-Robotics, 2016; Richard, 1981).

Drive technology: electric, hydraulic, pneumatic, and so on (Dobra, 2014).

Robot morphology: morphology describes the physical frame and structure of the robot. It provides the robot with physical and social functionalities, attractiveness and the ability to demonstrate emotions (Fong et al., 2003). The robot performs its tasks depending on its internal and external features, form and structure. Therefore, there must be a balance between the robot morphology and its appearance (Pfeifer, 2000). The types of morphology applied to the robot are likely to provide the human with the boundaries of interactions (Falcone et al., 2003; Fong et al., 2003). Designing a robot requires careful consideration of its morphology since the designer needs to decide the robot's functionality according to selected morphology. Therefore, robot morphology includes the following concepts:

- All robot sensors, structure, material, peripherals and control systems (Delcomyn, 2007; Pfeifer, 2000)
- All the internal structural features of the robot, such as:
  - The geometry of the robot's mechanical structure, known as kinematics.
  - The force that acts on the kinematic skeleton, known as kinetics.
- All the external structural features of the robot and its appearance: size, color, shape of the outer body frame (functional robot, artificial-shaped robot, bio-inspired robot, etc,.(Baraka et al., 2019), structure and pattern. There are appearance classification that describes the robot external look, where robots are grouped into three main categories: bio-inspired, artifact-shaped robots, functional robots (Baraka et al., 2019; Shibata, 2004).

Robot embodiment: through which the robot connects to its environment. Embodiment is measured by the degree of perturbation the robot performs on the environment via its perturbatory channels. The more perturbatory channels embedded in the robot morphology, the more it is embodied with its environment. The higher the degree of embodiment applied to the robot, and the more morphological intelligence that can be implemented within the robot, the more the robot is able to perform capabilities (McGrenere and Ho, 2000) and the more it will be accepted by its users (Duffy et al., 1999). There are several types of embodiment that are used within social robots. Each embodiment type requires a specific implementation that supports the robot with specific behaviors (Fong et al., 2003). These embodiment types are: an anthropomorphic embodiment as it employs human physical features such as in NAO (Gouaillier et al., 2009), zoomorphic embodiment to employs animal physical features such as in MiRO (Mitchinson and Prescott, 2016), a caricature robot to exaggerate specific features such as in Jibo robot (Breazeal, 2017b), functional robot acts as a tool such as the industrial Kuka robot (Bischoff et al., 2010), or screen character embodiment such as in the receptionist robot (Lee et al., 2010).

There is another embodiment perspective developed for human-like robots that presents a degree of anthropomorphic embodiment. The embodiment perspectives are presented in a framework containing the three extreme social robot embodiment designs. The first corner presents the robot in totally human design. The second corner presents the robot with an iconic design as a totally comic figure. The third corner presents the robot in an abstract design with only a machine frame. This framework, illustrated in Figure 2.5, presents a comparison of additional physical modalities needed in the social robot (Duffy, 2003).

Designing a social robot can require a human or animal-like design, which requires the appropriate level of embodiment. The higher the degree of embodiment applied to the

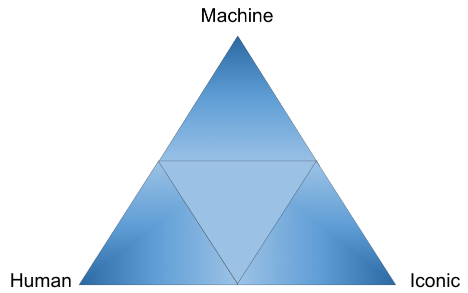

Figure 2.5: A framework to present the degree of robot embodiment, as defined by Duffy (2003).

robot, the more morphological social and cognitive aspects can be implemented within the robot, which would increase its acceptance by its users (Duffy et al., 1999). But when this embodiment reaches the highest degrees of familiarity to an almost realistic point, the sensitive imperfection becomes repulsive; consequently, the robot will be emotionally rejected by the user, which is known as the uncanny valley. Therefore, the uncanny valley should be kept in the designer's mind while designing the social robot (Duffy, 2003).

Degree of autonomy: the robot is called autonomous if it can perceive its surroundings, make judgments or choices, and is able to coordinate its actions to perform tasks without human intervention. There are also some types of robots that act autonomously depending on their intentions (Breazeal, 2003). These levels range from direct control for tele-operated robot to dynamic autonomy for peer-to-peer collaboration (Goodrich and Schultz, 2007). In general, any robot with complete remote control is not considered a robot because it is not making any decisions (Bartneck and Forlizzi, 2004). There are different levels of autonomy, and each level can be carried out by a robot to perform an autonomous task (Beer et al., 2014; Veres, 2014). Some social robots can be designed to perform different levels of autonomous control where each task falls into one of the autonomy levels. The robot performance should consequently activate the appropriate level of autonomy for the task it performs (Kahn Jr et al., 2007; Steinfeld et al., 2006) regardless of its interaction level, especially applied for social robots. Therefore, autonomy was defined by Baraka et al. (2019) as how the robot performs its tasks without human intervention, not interaction.

There is another classification that has been defined for the development of industrial robot. This classification illustrates the direction of the development for the main capabilities (Wood, 2015). The dimensions are (1) the robot's cognition ability: to understand and process capabilities. It also includes the robot's aims and plans to achieve its goals. The robot's manipulation ability: the robot's ability to move within its surroundings. The robot's interaction ability: the robot's ability to interact with humans. This is illustrated in Figure 2.6. The figure also highlights that with the development of these dimensions within the industrial robots, the autonomy would increase and therefore, all of the industrial robots will belong to service robot type.

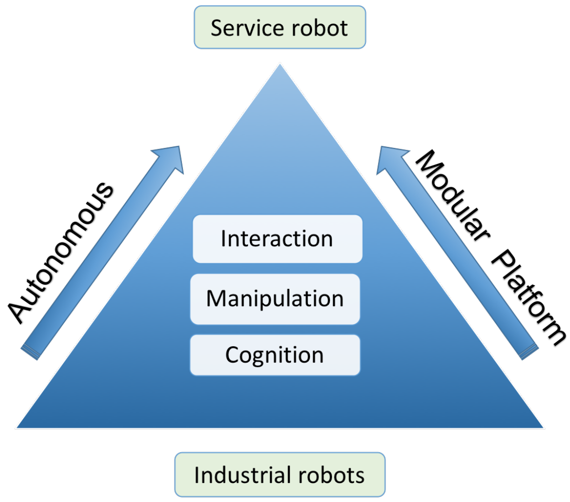

Figure 2.6: Based on the technological capabilities aspect in industrial robots as presented by Wood (2015).

Goals: that the robot is designed for or implemented to fulfill. Many types of robots are categorized according to designer intention and what performance and capabilities are intended to be present in the robot. The most common intentions are: contest, service robot, personal enrichment, manufacturing, entertainment (Hobby, 2016), human expression and discovery (Falcone et al., 2003), or a biologically inspired robot such as social/non-social robot or anthropomorphic/zoomorphic robot (Duffy, 2003; Schmitz, 2011). Any of these goals can be broken down into smaller sub-goals, each of the which further breaks down into robot capabilities (Brooks, 1986).

Robot role: the human role and the robot role are important factors in defining the robot as it affects the performance and its fluidity. The human role, defined in HRI applications as supervisor, operator, mechanic to control the task, peer and by-stander (Scholtz, 2003) where the robot role defines the scenarios of the interaction (Goodrich and Schultz, 2007) and draws the behavior of the robot. Therefore, the robot roles and the human roles are important as each role has gathers different information, provides awareness to the performed capabilities and behaviors (Steinfeld et al., 2006). Defining a role for either the robot or the human determines the responsibilities and authorities of performance for each of them. Moreover, the role determines the necessary skills within each partner, which helps in evaluating the performances of each much more easily (Goodrich and Schultz, 2007).

The robot roles are the same as the human role with an additional of two roles: the mentor and the informer role (Goodrich et al., 2008). Additionally, the robot role was presented through four interaction paradigm for HRI: a tool, cyborg extension, avatar, and social partner (Breazeal, 2004). By assigning a role to the robot in the intended interaction, the level of functionality can be determined (FeilSeifer et al., 2007).

Moreover, another role classification was defined for the human and the robot together, considering the role of a robot towards the human: robot for you, robot as you, robot with you, robot as if you, robot around you, and robot as part of you (Baraka et al., 2019). The role is also defined either seldomly such as a machine role or intensively such as the role of an assistant (Dautenhahn, 2007). There is also Goodrich and Schultz (2007), who provided a taxonomy in proximity robots within the application area which presents the human role accordingly. This taxonomy is summarized in Table 2.2.

These several classifications on roles emphasize how people recognize robot and distinguish the types of skills, capabilities and interaction performed by them.

Table 2.2: Different robot applications with their human roles for both remote and proximity interaction.

	Remote		**Proximity**	
Application area	**Role**	Examples	**Role**	Examples
Search/rescue	supervisor	search operation	peer	for unstable structure
Assistive	supervisor	integrated surgery	peer	elderly therapy
Military	supervisor	reconnaissance	peer/mentor	patriot support
Education	supervisor	Online teacher	peer/mentor	classroom teacher
Space	supervisor	exploration	peer/mentor	astronaut assistance
Industry	supervisor	construction	peer/mentor	building companion

Proximity: the type of interaction required between the robot and the human defines the proximity. There are three main proximity dimensions: remote, collocated, and physical (Baraka et al., 2019).

Temporal profile: defines the time-related aspect between the robot and the human. It covers time span (Short time, medium time, long time, and life long) and duration of the interaction and its frequency (Baraka et al., 2019).

Robot intelligence: define the intelligence aspects within the robot and indicate how smartly the robot can interact with its surroundings. Robot intelligence presents the overall performance of the robot and its final goals (Bostrom, 2012) or describes the external perception of the robot actions (Schaefer et al., 2012). It illustrates the intention of the developer by implementing robot performance through specific techniques (He et al., 2017) or through a specific behavior and by performing an adoptive interaction (Kihlstrom and Cantor, 2000). Intelligence requires the robot to accomplish basic abilities (such as understanding, selecting, learning, performing and behaving) carried out with sophisticated skills (Dautenhahn and Billard, 1999; Pfeifer and Scheier, 2001). Also, intelligence needs to support the robot's capability of generating new skills in relation to abstract new concepts (Tzafestas, 2015).

There are several intelligence concepts that are provided in the literature to describe general intelligence (Legg et al., 2007), including Thorndike's types of intelligence for human resource management, which categorizes intelligence into abstract, mechanical, social (ManagementMania, 2016), and Gardener's theory of multiple intelligence (Helding, 2009). Also, there are many artificial intelligence definitions, but most of them can be categorized into (1) a system that thinks like a human, (2) acts like a human, (3) thinks rationally and (4) acts rationally (Kok et al., 2009; Martınez-Plumed et al., 2018).

Several attempts have been made to classify robot intelligence. One of the robot intelligence classifications aims to determine how intelligent a robot is Winfield (2017). This intelligence depends heavily on embodied intelligence that emerges from interaction. It is a combination of four distinct kinds of intelligence: morphological, social, individual, and swarm. Figure 2.7 presents the accumulation of the four types of intelligence and shows them in one simple relation. The study also suggests a quantifiable intelligence vector that attempts to determine the intelligence as a numeric value.

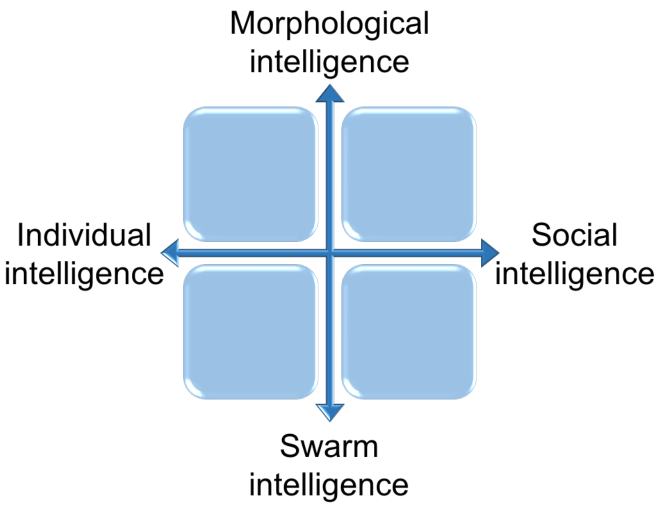

Figure 2.7: Based on intelligence categories defined by Winfield (2017).

- Social intelligence is the social-intellectual ability of the robot. It has been defined as the capability of the robot to learn either by imitating or by following instructions

(Winfield, 2017). One of the basic concepts of social robots is to imitate human intelligence. This is considered a fundamental concept for AI. Therefore, social skills are not an add-on for an attractive purpose any more; rather they are considered to be part of the cognition and intellect that demonstrates the level of intelligence in robots. Roboticists would develop an intelligent robot by building its cognition and implanting it with planning, reasoning, a sensory system, navigation, and manipulation, finally adding social skills as part of social cognition. However, researchers have discovered that social intelligence has to be developed as a fundamental concept for any intellectual robot. Therefore, developing a socially intelligent robot is a key concept for developing an intelligent robot that contains credible human–robot interaction (Dautenhahn, 1995, 2007).

- Individual intelligence covers the individual agent within the robot in performing a task according to the physical situation or social situation (Edmonds and Dautenhahn, 1999). It also covers the response as actions, learning and adaptation to specific behavior (Winfield, 2017).
- Swarm intelligence is presented as a result of actions that come from a group of swarms, flock, and herd (Winfield, 2017). It is also known as a group intelligence of a cluster of robots (Edmonds and Dautenhahn, 1999). Each robot in the group is autonomous, situated in the environment, and has sensory and communication capabilities. They do not have access to centralized control; rather, their AI cooperates to tackle a task (Brambilla et al., 2013). Their performance depends on computational, communication and capabilities aspects, which could be notated as collective intelligence (Beni, 2004; Dudek et al., 1996; Heylighen, 1999; Parker, 2008).
- Morphological intelligence captures the movement ability within the robot. This intelligence uses the body to calculate and decide the type of interactions to perform and therefore define its behavior. It also covers the skills that the body could support physically. The robot body will also have some inherent movements according to its morphology, known as inherent intelligence or morphological computation (Müller and Hoffmann, 2017; Winfield, 2017).

The intelligence demonstrated by machines is being constantly advanced in many directions. Therefore, an atlas for robot intelligence has been developed which covers the motivation, application, and kinds of systems and devices. The atlas defines several criteria for describing the aspects of any intelligent system (Bhatnagar et al., 2017), and it also presents several intelligence maps to be used in the robotics domain that would help in defining the intelligence aspect of the robot.

There is another aspect of artificial intelligence classification. It presents the artificial intelligence facets and all of its related boundaries for analyzing the evolution of AI. The classification tracks the evolution of AI throughout the years within nine facets. The analysis illustrates the systematic growth that shapes the boundaries of AI (Brooks, 1991; Martınez-Plumed et al., 2018). Moreover, there are several AI evaluations procedures that were developed, including the most well-known assessment, the Turing test, which requires natural language processing, knowledge representation, automated reasoning and machine learning (Kok et al., 2009).

Moreover, there has been much research in AI and in robotics. Some of these experiments should be replicable and reproducible. Obviously, repeating the experiment will not lead to the same exact output, but if the experiment is described in full detail, listed with a complete data set and downloadable code, and presented with a description of the hardware, then the result should fall within the expected outcome, which is within the margin from the outlined criteria. These reproducibilities, known as the r-articles among lists of research, are an important feature of any new scientific methodologies especially for intelligence in robotics (Bonsignorio, 2017; Stoelen et al., 2015).

Another intelligence perspective is available to covers AI methods that are developed within machines. These methods support the machines or systems with techniques to imitate a special bio-inspired intelligent behavior (Tzafestas, 2015). This aim to understand how to make the robot intelligent and which technique should be applied to get the required intelligent behavior. Other AI methods are presented through software technology that supports the intelligence mechanisms within devices (Poole et al., 1998). Some example of these software technologies are machine learning or deep learning. And as AI encompasses these machine learning techniques, either deep learning or supervised/unsupervised learning, could be applied to any capability within the robot. Therefore, AI is considered an outer umbrella that holds any capabilities, such as speech recognition and visual perception, that are performed through algorithms to mimic intelligent agents or any agent behavior that emerge from the simulation and captured in its performance (Duffy et al., 2000).

A collection of attributes for robot characterization for specific domains: are presented, such as (1) modeling human–robot interaction for intelligent mobile robotics (Rogers et al., 2005), (2) an atlas of physical human–robot interaction that covers different robot sensors, camera classification, systems, design, planning, and so on (Adams et al., 2012), (3) design-centered frameworks for social human–robot interaction (Bartneck and Forlizzi, 2004), (4) the dimensions of human–robot interaction in socially intelligent robots (Dautenhahn, 2007), (5) ontology for robotics and automation (Prestes et al., 2013). (6) Layered approach to develop and build robot control system was also described (Brooks, 1987). These collections of attributes help in defining a robot from the same field or of the same type.

There is also a **collection of evaluations attributes and benchmarks** such as fostering progress in performance evaluation and benchmarking of robotic and automation systems (Bonarini et al., 2006; Bonsignorio et al., 2014), multivariate evaluation of interactive robot systems (Huang and Mutlu, 2014), and robot function classification (Winfield, 2012) and many others.

2.6 System abilities according to robot requirement, presented by MAR

The Multi-Annual Road-map (MAR) (EU-Robotics, 2016), with the Strategic Research Agenda (SRA), provides a strategy for identifying the current ability levels for specific capabilities in several (albeit, still limited) robotics sectors. MAR presents the required market target ability levels for each sector, by providing a clear set of marketing goals that are relevant to the robotics fields. It is a significant and large document that covers over 300 pages. It is well structured and organized according to the robotics domains, and it is updated annually by experts at EU Robotics AISBL, for Association Internationale Sans But Lucratif, which is a Brussels based non-profit international association for all European robotics stakeholders. EU Robotics AISBL prioritizes the technology and the strategic development that is shaping European research development and innovation. MAR contains a detailed explanation of robot characteristics, hence it is considered a technical guide with detailed progress for specific robot capabilities. It presents several robot abilities: configurability, interaction ability, dependability, motion ability, manipulation ability, perception ability, decisional autonomy, and cognitive abilities. Each ability is presented with sub-types, with some levels annually added for each of the sub-types (EU-Robotics, 2016). However, using MAR to characterize robots is a very complex process, which is why it is not popular or well used. In addition, MAR does not include all robot capabilities (e.g., emotion, social capabilities, cognitive interactions).

MAR was developed in order that public investment would be directed toward research that would deliver the required capabilities for specific robotics domains. This required developing an intermediate scoring system to map between the application domain and robot domain, as presented in Figure 2.8. Both domains require some scoring for the capabilities. This scoring needs the capabilities to be described in levels. The required capabilities for an

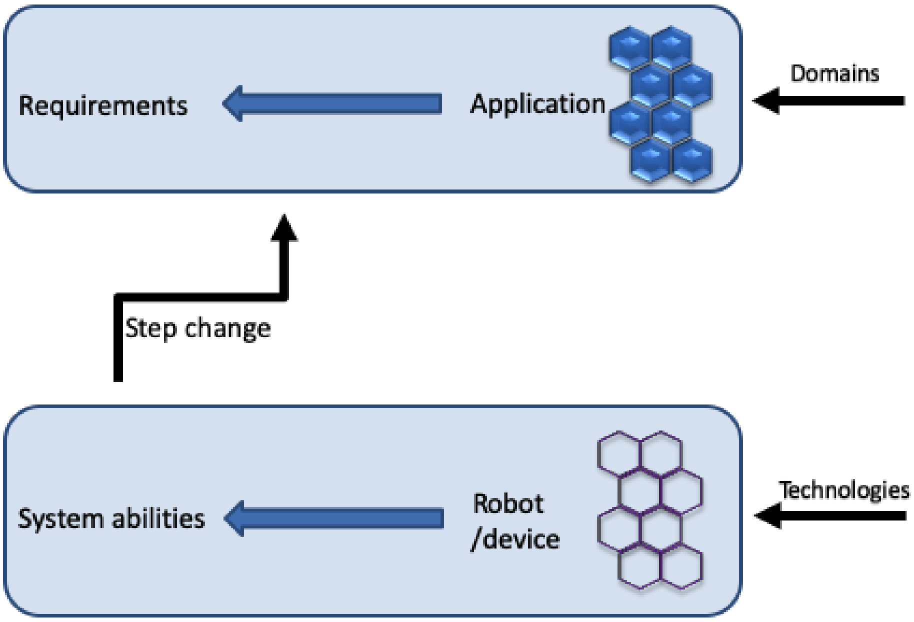

Figure 2.8: Capturing the relation between the requirements and system abilities as presented in MAR, based on the figure in page 4 in MAR, EU-Robotics (2016).

application need to be described through capability execution levels, and the performance of robot capabilities should be described through a range of listed maturity levels. The selected level describes what the robot is capable of performing, so the requirements of the application can be specified to match robot capabilities. These required levels and the robot capability levels can be mapped and linked (Bisset, 2017).

The MAR capabilities were categorized according to capabilities that are visible (e.g., cognition, perception, autonomy), capabilities that are visible through performance (e.g., interaction, mobility, manipulation), and capabilities that are not visible through performance (e.g., adaptability, configurability, dependability) (Bisset, 2017). These classifications are not clearly distinguished by roboticists in developing and designing capabilities. Additionally, they are not used in the robot application domain, so they cannot be used to map between the two domains. A further difficulty with MAR is that non-specialists will struggle to use the capabilities levels, as the capabilities are not categorized and their levels are extremely technical. Therefore, capabilities need to be illustrated with a higher level of categorization and abstraction to be understandable by non-specialists working with the application domains. These categories should be as uncomplicated as possible, so using general aspects such as ‘social’, ‘physical’ and ‘cognitive’ for categories will aid in mapping between the applications and requirements of a robot.

2.7 Outcome of the review

There are several robot classifications, each of which depends on several aspects of the robot. For example, classification according to the robot functions depends on the robot tasks (Winfield, 2012), field classification depends on the robot's physical features (Siciliano and Khatib, 2008), domain classification depends on specific requirements (EU-Robotics, 2016), size classification depends on the robot's size (Dobra, 2014), and robot intelligence depends on the knowledge acquired, skills or behavior performed by the robot (Winfield, 2017). Each of the robot classifications has its own list of characteristics and features related to the classification and does not cover every single aspect of the robot. Therefore, they cannot be used to define robots and classify them in general. Among all of the mentioned classifications criteria, there is no single classification that emphasizes common robot characteristics that could be applied to all robots in different domains, types and fields. These common robot characteristics, if they exist, could be used to identify robots and compare between them throughout the robotics field. In another context, the mentioned classification criteria have been generalized but not been defined as to how to use them together as a single process. Hence, they neither identify robots nor compare between them, and, implicitly they do not classify them. Additionally, robots are developed with different characteristics and capabilities to satisfy specific application requirements. Robot requirements specify the expected set of capabilities presented within the robot. Robot capabilities are intrinsic to the robotics domain and are applicable to any robot. Therefore, developing a capabilities classification to assess which capabilities a robot is developed to perform is important for defining robots and distinguishing them from each other.

MAR (EU-Robotics, 2016) presents a list of detailed capabilities, but there is no clear methodology for capturing these capabilities. In addition, MAR's list of capabilities is limited to specific areas and does not include all the capabilities available in the robotics field. For example, it does not include advanced social capabilities, such as the presentation of emotion or social skills, which are becoming more prevalent in recent robotics developments (Anzalone et al., 2015). Also, MAR represents several social and physical interactions but does not cover other types of interaction, such as wearable technology (Lee et al., 2016; Lenzi et al., 2011; Stead et al., 2004), virtual interaction, augmented interaction, and embedded interaction, which are used by smart devices in the Internet of Things (IoT) (Rastic-Dulborough, 2014). On the other hand, as robotics is an expanding field, new capabilities are developed every day. MAR (EU-Robotics, 2016) offers levels for robot capabilities, but it was not designed to define or embrace any capability. Rather, it was developed to shape the European research development and innovation program (EU-Robotics, 2016). Yet MAR was updated annually, from 2014-2020, to cover the capabilities that the program was considering priorities in each of its development years (Robotics, 2014). For example, the 2017 edition of MAR noted that object recognition levels 13 and 14 would be reviewed for the next edition, but without an explanation of how. Consequently, it does not include all newly developed capabilities or even all the available capabilities. Thus, it is not sufficient to represent all possible robot capabilities. However, it is important to develop a capabilities classification that can map between applicant requirement and robot technology, and therefore one that can capture any capabilities including the newly developed ones. Such a classification could find the best matched robot for any given need. In order to do this, there must be some kind of filtering and grouping of robot capabilities that will lead to a better definition of the robot capabilities, which in turn will generate a classification and a taxonomy.

The challenge to this required taxonomy is the rapid development of robot features, capabilities and requirements, which raises the question of how to define and measure the robot's capabilities effectively and efficiently. This has to take into consideration the potential of changing or upgrading some capabilities aspects of the robot. For example, capabilities that depend on the software can be more easily altered or upgraded than capabilities dependent on the hardware of the robot, as the latter are more fixed and not easily alterable (although they could still be changeable). This leads to some unsolved categorization questions, such as how to set the boundaries to identify robot capabilities and classify them accordingly, how to define scientific concepts in robot capabilities classification and what these concepts will be, and how well a particular robot will perform a capability from an operational perspective. Above all, it raises the question: How can the robot best suited to any application requirement be found according to the capabilities and their measure?

Consequently, a framework of greater scope is required both to capture all present known capabilities and to allow capturing any extension to them. The required framework should serve as an umbrella to hold different types of capabilities in its specific subsections. It should also be flexible enough to include any new developing capabilities within its wider sections. This flexibility is important in presenting robot capabilities available now, as well as those that will be developed in the future. The desired framework should capture all characteristics that affect the capabilities and demonstrate all of the different degrees of the capabilities. The presentation of the capabilities and their maturities should be illustrated in a clear and comprehensive format. The format should be able to demonstrate the differences between the capabilities as they are being scored. The framework should define the quality and degree of capabilities within the robot. The capabilities should be presented with their different degrees presented in a list, and the evaluator needs to match the robot capabilities against the checklist available in the framework. If the capability is not listed, the framework should be flexible enough in a way that the capability can be added within the appropriate and most suitable section, according to the capability context. This would give the framework the advantage of presenting and adding any capabilities to its hierarchy, which would make it the most suitable hierarchy for capturing capabilities for any robot. The next chapter demonstrates the development of the proposed framework according to the main dimensions of the robotics field. The following chapter demonstrates how such a framework was established and assembled.

✳ Chapter 2: Summary

Classification is one of the most important subjects in science. All of the mentioned robot classifications have been formed independently, either according to the field, domain, type, size or according to specific robot characteristics. These classifications do not cover all aspects of the robot's capabilities; rather, they include only some of the ever-growing range of possible robot characteristics that can be used to identify robots and to compare them. A multidisciplinary classification is required to describe robots more accurately and to understand what they can and cannot perform regardless of their operating domains and fields. Additionally, none of the robot classification systems covers all characteristics available within the robots; consequently, they are not useful as a comparison tool for assessing the differences between robots. Therefore, there is a need to develop a set of dimensions that covers all robot characteristics and capabilities. These dimensions would allow direct robot comparisons for robots, whether from the same or different fields. Describing these dimensions in one framework would illustrate the relation between the dimensions. It would also provide an easier method to describe a robot and to document its characteristics in a way that avoids any misrepresentation. This research aims to develop new classification dimensions that capture all robot characteristics regardless of the field and domain. It is intended to be based on logic and philosophies of the robotics domain. The classification requires a framework that contains all of the main robotic principles and links them to each other. The development of the framework will illustrate the classification procedure. The classification procedures should be tailored to support the principles of the robotics field.

Chapter 3

Developing a classification framework

3.1 Introduction

The aim of this research is to define a robot according to their characteristics in order to categorize and classify it. The robot classification needs to be applicable to all types of robots across the robotics field. The classification procedure will be applied as a tool to describe and compare robots across domains. Developing such a tool requires capturing robot characteristics and categorizing them in a way that can present an adequate description for any given robot. To develop such a tool, several steps must be undertaken. The first step is to develop descriptive dimensions for the classification. This is performed by a review of the literature on robot characteristics, including robot features, capabilities and other taxonomies. The second step in developing this classification is to find out how to group the chosen list of characteristics, according to common similarities, which, in return, identifies the main and sub dimensions for describing and classifying robots. Through these dimensions, robots can be identified, described and compared even if they belong to different domains or fields. This chapter, therefore, illustrates the development phases for creating the dimensions of this classification framework.

3.2 Methodology

Several methodologies have been used in this research. The objective was to develop a hierarchical framework covering all the necessary dimensions needed to describe robots and to designate their specific outstanding characteristics. Therefore, the research followed several methodological steps, the inductive approach and the exploratory approach through an iterative process, such methodologies were used in libraries and archives classification (Da Graça Simões et al., 2018). The study starts with a literature review, covering, understanding and discussing theoretical concepts of the robotic field. Next empirical evaluation for some robotic data sets to identify the main robotic principles. Then, according to these principles, the robot characteristics were defined, characterized and layered in conceptual structure. According to different robotic related disciplines, the concept of the structure was reviewed and altered. After that, the structure was completed by appending several unlisted aspects. Finally, the structure was evaluated. These aspects are illustrated in more details in the following phases:

1. The first phase looked at specific robot types to generate a hierarchical structure through:
   - Literature review: a review on robot characteristics was undertaken. Specifications, standards, measures and classifications were collected from the literature. This covered information available through existing descriptions, websites, journals and documents.
   - Categorization and generalization: using grounded theory (or systematic method), robot characteristics were categorized under general dimensions and sub-categorized according to similarities. This grouping was essential to define the major dimensions for the structure (e.g., perception, sensors, activities, size), allowing for a generalization of similar capabilities to define a specific group of robots.
   - Paradigm and theory: defining relations between characteristics enabled the creation of an outline structure. Linking between categories identified the relations between characteristics to present the structure in a simple form.
2. The second phase adopted general robotic concepts and added them to the developed framework as needed. Using this methodology defines and explains the relationship between the adopted concepts in the robotics' domain. This methodology was applied through the following steps:
   - Different robot classifications and performance concepts were applied to the framework.
   - The developed framework presented robots in the robot domain. This was then linked to the application domain (how the characteristics will be applied as requirements) by adapting a framework of spoken language systems and its underlying technology (Moore, 2000). The adopted framework was mapped

between the application and robot domains, identifying new relational dimensions between them.

- The MAR capabilities, sub-capabilities and their levels were adopted for the conceptual framework to cover the robot sub-capabilities quantitatively. This was done through a thematic analysis of the robot capabilities.
- The conceptual framework was confirmed with roboticists to refine the structure and consolidate the categorizations. This was done through personal meetings reviewing a structured questionnaire that describes the suggested hierarchy of robot domains.
- The conceptual framework was analyzed to pinpoint where capabilities or sub-capabilities were missing. Adapting the MAR style, new capabilities and sub-capabilities were developed with their levels.
- Robot ontology classifications were studied and applied in order to create a top-level categorization for robotic concepts, properties and their relations. The categorization helps in identifying the main dimensions in the robot domain. The classification also encompassed all robots, their main aspects and their relations to each other presented in one comprehensive structure. The structure was used to describe robots and present them through outlined profiles that could be generated automatically. The automated version of the developed structure was presented as a proof of concept working tool (Olszewska et al., 2017; Prestes et al., 2013).

3. The third phase was to evaluate the framework using the following methods:
   - The first evaluation methodology was to assess the ToRCH framework by personally meeting several roboticists, using a guide of structured questions, as mentioned earlier.
   - The second evaluation process was to assess the framework as a tool. Structured questionnaires were used to evaluate: The accuracy of the framework in capturing robot capabilities. It was given to NAO robot programmers who were asked to describe the capabilities of NAO as presented in textual scenarios using the developed conceptual structure. This evaluation also demonstrates the richness of the framework for capturing different capabilities and its preciseness in scoring capabilities of the same robot, by both of its developers. Finally, the effectiveness of the framework in capturing the capabilities of different research projects using the same robot was assessed.
   - The third method was to assess the outcome of the framework by using a structured questionnaire in generating robot capabilities profiles (RCPs). The RCP was utilized to present robot capabilities before and after the development of a

robot within a research project. It was also to presented capabilities of a robot used for two different projects and compare individual robot capabilities against a connected external system or device.

3.3 Development of the structured framework

The hierarchy of robot characteristics was achieved through several phases as follows.

3.3.1 Collecting and organizing robot characteristics

In order to include wide variety of robot characteristics, a large data set of robots with their specifications was required, listing robots and their sensors, actuates, software programming aspects and their performance. This was presented in an Excel spreadsheet, with each row for a different robot and the columns listing the robot's characteristics. When a new robot was added to the sheet, a new row was created with new columns as needed to cover all the hardware, software and capabilities within the robot. While initially, the intention was to add robots until all characteristics were included, the advancement of robot technology means that this would be impossible. Including all the current robot characteristics in one data set created an unwieldy and unusable data sheet. Nevertheless, this process of listing the hardware, software and capabilities of robots was important to outline the major characteristics and to format the dimensions needed in classifying robots more generally. Setting the main domains was an important step in developing a structured framework, through listing the characteristics, selecting the major ones as dimensions, categorizing and grouping them. This was the research plan for the classification framework. This approach faced several problems:

First, there is no agreed set of robot characteristics among the robot domains for classification because hardware, software and capabilities improve, increase or change across domains, especially with the rapid evolution of research and development.

Second, the features of every robot could not be clearly defined, organized and grouped for several reasons. Thus, some categorizations within the literature are applicable for some domains but not suitable for all of the robotic domains. For example: social skills and behavior are important characteristics for defining social robots, but with the advancement of hardware and software used for interaction and interface, these elements could be used in a robot not intended for a social role such as manufacturing robots. Conversely, most house robots performing simple chores around humans are not considered as social, as they lack such characteristics such as vacuum robots. The exoskeleton was sometimes classified as social because it interacted with humans, although it did not need to present social skills and behavior. Another example is when software models were upgraded for each robot the feature of the old programmed method is no longer applicable. Also looking at the robot hardware, the perception and sensors in an assistive exoskeleton (to capture signals from the human body) would be different from the perception and sensors required in a “social” robot (to capture the necessary input for social skills and interactions). Likewise, attempting to categorize robots according to the dimension of size was not possible. A nano-bot could be, for example, medical or industrial.

Third, as the Excel sheet was filled with new robots and dimensions, it looked increasingly random and difficult to understand. It became a matrix with no recognizable pattern, despite attempts to structure and organize it, as the amount of information increased. Unsuccessful pattern recognition led to the unsuccessful categorization of the robot features and characteristics. Another problem occurred in the grouping process of the listed characteristics. Defining robot characteristics into groups, according to which specific features clearly belonged to a specific field and thus the robot to a specific group, was not a straightforward process, as these groupings changed when new features were added. For example, the capability of identifying a human was classified as a social skill, but a robot that uses sensors for human perception in the industrial field is not considered a social robot and is often not defined as industrial but rather as collaborative. Therefore, defining and organizing “human perception”, for example, differ across fields and applications. Fourth, presenting the relations between robot definitions in the literature and their hardware, software and capabilities were impossible. Some characteristics can be used to define the robot type, others the robot application; for example, human interaction could be presented differently according to whether it was a characteristic of a social, home-help or collaborative robot.

Therefore, defining robot applications and domains according to the robot characteristics was not successful, especially when the classification is needed for the successful deployment of a robot. Hence, another approach was needed to define robots accurately for classification and deployment.

Outcome from the data analysis: presenting robot characteristics and capabilities across domains is a challenge as they were not standardized. Hence, applying specific characteristics cannot be used to classify robots across all domains. This study needed to research and define a set of characteristics that could be used for all the different robotic domains. These dimensions need to be recognized in the robotics domains, or any other linked domains, such as for health and safety or training requirements. Such dimensions might be an important characteristic in defining a robot, even if it was not directly used or linked in describing the robots. Therefore, defining new dimensions for identifying robots, which could be applied to some robotics fields, is an essential step in creating this framework. These dimensions need to act like an umbrella for robot characteristics. Even if a specific characteristic is currently unknown, there should be the flexibility to add it under the umbrella and to organize characteristics to create a structured taxonomy. Once the workable dimensions are identified and listed, the database can be used to create a program as proof of concept, covering these dimensions and characteristics. Presenting the most important robot characteristics and capabilities for all applications allows for a framework that is of use across robotics domains. Where a robot characteristic is not available it should set with no and where a capability is not present, it should be assigned a value of zero. Thus, whereas all characteristics and capabilities should be available in the structure only available characteristics and capabilities within the robot will be defined with yes and assigned a value. This creates a filter for defining a robot and searching for the best-fit robot for any application or requirement.

3.3.2 Developing new dimensions

Robot technology is constantly developing, so a framework must be developed to capture all characteristics. Developing such a model requires accurate definitions and inclusive dimensions that can ultimately help in predicting which robots are most suitable for specific applications. This requires collating information about robot capabilities and performance so that the needed description specified in the dimensions of the conceptual structure is fulfilled. In order for this structure to be easily understood and used, it needs clear organization according to a hierarchy of layers. It would also include the robot hardware, software and interactions necessary in the hierarchical structure. Such a format is necessary for robot identification, presentation and to help in the design of new devices from existing technology. Thus, the structure should list which capabilities are presented in the robot and what available features allow the performance of a specified capability. This conceptual framework also needs to clarify the relationship between different capabilities and performances of the robot and be in a format that can capture both currently available and any potential characteristics. Including all these concepts in one structure will create a rich and comprehensive model; however, the complexity of such a model would mean that it is hard to validate and define its faults. One step towards validation would be by supporting the framework through concepts derived from the literature review, as suggested by Dautenhahn (Dautenhahn, 2017). The following part of this chapter presents the adopted concepts that have been used to construct the conceptual framework.

Defining and adapting intelligence

Most robots are defined with specific types of artificial intelligence (AI). Some link their AI with bio-inspired decisional capabilities (such as in swarm robotics) while others link their AI with advanced meta mathematical algorithms (such as Deep Blue). At the data collection phase, similar robots were listed with different types of AI, which highlights it as one of the key aspects that are needed in describing and comparing robots. Therefore, a review of the literature on intelligence and AI was undertaken. Intelligence has several definitions and various aspects. In general, it is an ability that is composed of several functionality (Legg et al., 2007).

The review aimed to understand how intelligence, in general, is assessed, used and described, especially by considering several taxonomies for intelligence, such as an early taxonomy defined in 1920 by Thorndike (ManagementMania, 2016) that divided human intelligence into three parts:

- Mechanical intelligence, which describes the ability to understand concrete objects, their manoeuvres and the physical capacity to manage objects.
- Abstract intelligence, which refers to the ability to understand and manage ideas.
- Social intelligence, which describes the ability to cooperate with other social members and act sensibly towards them in sociable situations (Kihlstrom and Cantor, 2000).

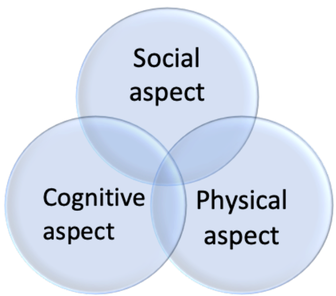

Figure 3.1: Adapting Thorndike's intelligence model in the robotic domain (ManagementMania, 2016). Presenting the mechanical intelligence to capture the physical aspect of the robot, the abstract intelligence to represent the cognitive aspect of the robot and the social intelligence to show the social aspect of the robot.

These three intelligences are considered independent of each other. Thorndike's classification has been widely used in human resource management for recruiting, staffing and job development, which could be ideal to be used in deploying robots to their best applications. Thus they were adopted into the conceptual framework.

A second taxonomy is the multiple intelligence theory created by Gardner (Helding, 2009; Pal et al., 2004), an American psychologist. It defines seven mutually independent intelligences, although he later added two more. The original seven (Gardner, 2011) are (1) Linguistic intelligence: the ability to understand and express verbal expressions. (2) Logical-mathematical intelligence: the ability to solve mathematical equations and logical problems. (3) Spatial intelligence: the ability to think visually and solve visual problems. (4) Musical intelligence: the ability to understand and perform music. (5) Bodily-kinaesthetic intelligence: the ability to control the body and perform manoeuvres. (6) Inter-personal intelligence: the ability to be social with others. This could extend to cover the ability to be a member of a society or the ability to communicate publicly. (7) Intra-personal intelligence: the ability to perform individual self-understanding and develop it.

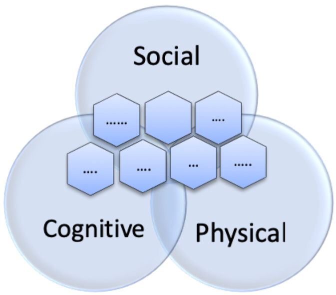

Figure 3.2: Distributing Gardner's taxonomy for intelligence into Thorndike's intelligence model as applied to intelligence in robots (ManagementMania, 2016; Pal et al., 2004). Gardner's taxonomy for intelligence describes skills obtained by the robot to support Thorndike's intelligence. Each skill is presented as a cell in a honeycomb. Linguistic intelligence, musical intelligence, inter-personal intelligence and intra-personal intelligence are all mapped to the show the social aspect of the robot. The logical-mathematical intelligence is mapped to present the cognitive aspect of the robot. Where the spatial intelligence and the bodily-kinaesthetic intelligence are mapped to capture the physical aspects of the robot.

For many developers, an approximation of human intelligence is the goal for AI development (Adams et al., 2012). Skills and task capabilities together define whether the robot is suitable for a specific application (Morrow and Khosla, 1997). Hence, intelligence was adopted in the framework in relation to assessing whether a robot is suitable for a specific application. Listing robots for specific applications requires defining its task capabilities and skills that enable it to be successfully deployed to perform a task (Bisset, 2017). Therefore, a description of the robot's intelligence, skills and task capabilities are important for the robot deployment. It defines the ability of the robot, whether it is capable to perform the tasks that are required for specific applications by its listed skills (Bisset, 2017). Hence, the dimensions for intelligence need to be linked to "capabilities", because capabilities indicate what skills a robot can perform. Adapting Thorndike's intelligences allows for the description of intelligence in three relevant dimensions, as presented in Figure 3.1. Gardner's seven intelligences were easily allocated within the three main dimensions of Thorndike's taxonomy as sub-dimensions of intelligence that are defined in the framework as skills/behavior. The three intelligence dimensions with their skills are presented in Figure 3.2.

Using these categories for intelligence captures important characteristics and activities that enrich the robot description. However, it also raises the following questions:

- What are the differences between a robot's performance of task capabilities and its intelligence?
- What is the relationship between a robot's capabilities, interactions and intelligence?
- What is the relationship between the hardware and software of the robot and its interactions?
- What would define robot intelligence?
- What is the relationship between robot intelligence and artificial intelligence?

These questions help in presenting, describing and classifying robots according to the perspective of robot capabilities and intelligence. Capability is the power and the ability to perform something. Robot performance is a sequence of actions, so the system has to be designed and performed according to the required specifications (Robeyns, 2003). All robots are designed with capabilities to perform certain tasks, which implies a specific physical, social and logical ability (Robeyns, 2003). Since capabilities are linked to present the quality of the robot intelligence capacity (Legg et al., 2007), therefore, both are presented in the structure as two different sections within the same layer "the capabilities layer", as presented in Figure 3.3 and Figure 3.4 in the first phase. The lower section presents the robot capabilities, including physical (also known as morphological to cover mobility and manipulation), cognitive (also defined as individual or abstract to cover logical aspects such as learning and reasoning) and social (such as emotion, social skills and behavior). The higher section maps the three robot intelligence categories (physical, cognitive and social) to a specific set of skills that are needed to perform the capabilities, that are presented in the lower section. The different sections of intelligence and capabilities are presented in Figure 3.4 (first phase).

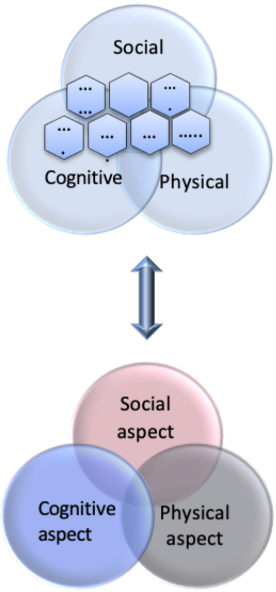

Figure 3.3: Adapting human intelligence to a robotics taxonomy using two sections, which allow for extra dimensions to describe and explain some of the main robot characteristics. The upper section describes the intelligence dimensions, presented with all and their supported skills and the lower section describes the robot capabilities types. To differentiate between the two sections the intelligence section is presented in light blue and the capabilities section is presented in colors.

Intelligence is an elusive concept. It is the ability to perform tasks intellectually by accomplishing them with different skills, behavior and talents (He et al., 2017). Robot intelligence is difficult to define because every robot is considered an intelligent agent (Poole et al., 1998). This is further complicated because AI the computational power in software to develop learning abilities or perform behavior, skills or performance more intelligently. Thus, Poole, Mackworth and Goebel, in *Computational Intelligence: A Logical Approach* (Poole et al., 1998), describe AI as "computational intelligence" which includes the programming nature in the AI. Thus, how can intelligence, a nebulous category at best, be illustrated within a robot? (Poole et al., 1998; Russell and Norvig, 2016) Therefore, intelligence needs to be categorized systematically within the conceptual framework, which was done by presenting Thorndike's intelligence as the main categories and adding Gardner's intelligence as the seven subcategories of the main three. These adopted intelligence categories and subcategories are presented in Figure 3.3 and Figure 3.4 (second phase). Consequently, the skills available in the robot are linked to intelligence (Saffiotti, 2017). Each intelligence present in the robot requires skill to support it. Hence, the intelligence categories were sub-categorized according to specific skills and behaviors adapted from Gardner's theory of multiple intelligence and incorporated in Thorndike's model (Pal et al., 2004; Winfield, 2017).

1. Morphological or physical intelligence (the performance of physical tasks such as avoiding, passing, following, approaching, touching).
2. Individual, abstract or cognitive intelligence (the performance of artificial cognition, exchanging information and planning).
3. Social intelligence (any social interaction, for example, group or interpersonal).

Thorndike's and Gardner's ideas about human intelligence do not include an important intelligence concept developed in robotics, namely collective intelligence (Heylighen, 1999), organization intelligence or swarm intelligence. This type of intelligence captures the behavioral intelligence of a group. It treats the individuals within a group or colony as decentralized systems with self-organized collective behavior. Therefore, each individual might have one skill, while the colony as a whole includes different skills. Likewise, collective intelligence is the ability of several individuals to perform a specific skill together. This type of intelligence is also presented in manufacturing, where a set of robots work together. Therefore, this intelligence dimension needs to be included in the conceptual structure of the framework. Thus, swarm intelligent and collective intelligence are set together and presented as the fourth dimension in the intelligence section, as in Figure 3.4 (third phase).

Although there is a difference between swarm intelligence and collective intelligence, "collective intelligence" is used for both categories to indicate that each individual in that group or colony is not considered individually intelligent but rather that the group as a whole is achieving a cooperative task known as collective intelligence (Dautenhahn, 2017; Kube and Zhang, 1993). Thus, this intelligence dimension was defined as (4) Collective intelligence: any performance or behavior presented by the group as a whole (Chrysostomou, 2017; Trianni, 2017).

Intelligence is presented in the structure as four main dimensions (physical, cognitive, social and collective) with several other sub-dimensions. The intelligence sub-dimensions were adopted from Gardner's taxonomy and mapped into the main four Thorndike dimensions. These sub-dimensions allow for the presentation of robot skills and behaviors through exhibited intelligence. However, as Thorndike and Gardner categorize human intelligence, their dimensions do not encompass collective intelligence. Therefore, they would not cover any non-human skills and behaviors such as that presented in swarms. Consequently, any skill that is not presentable through Thorndike's and Gardner's categorization could be added by the developer to the intelligence section of the hierarchy, especially where the robot displays non-human intelligence. Skills were presented in the structure via the honeycomb

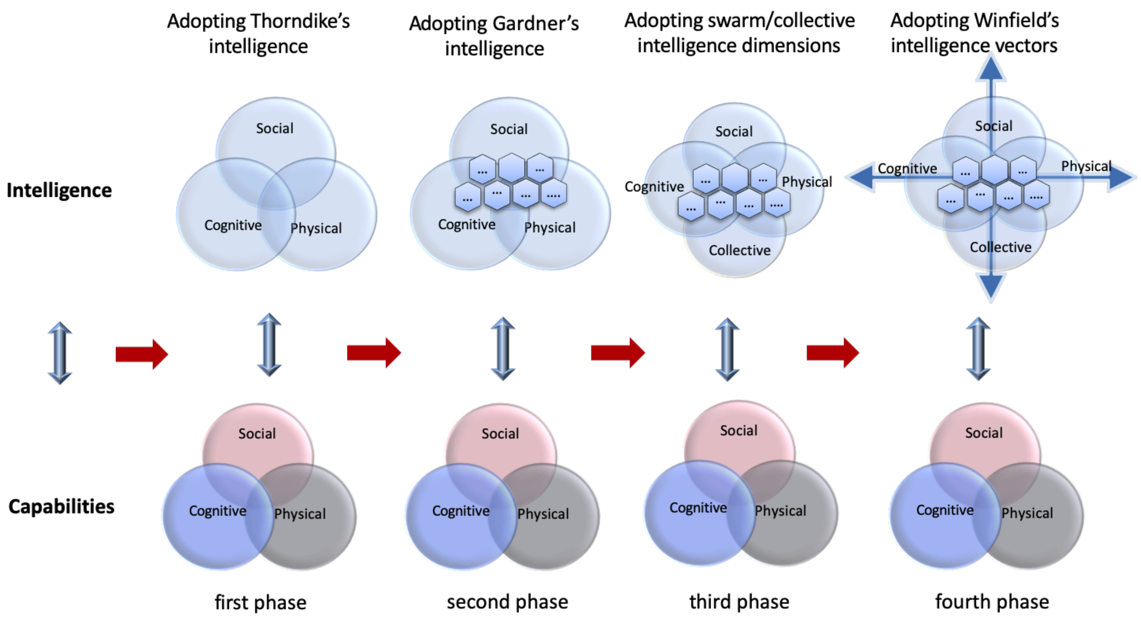

Figure 3.4: The Figure illustrates the phases in developing the main structure of the framework. It starts by adapting human intelligence in robotics in two sections. The upper section presents the robot intelligence and the lower section presents the robot capabilities. The first phase illustrates the different types of intelligence and capabilities, which allow extra descriptive dimensions to explain some of the main robot characteristics. The second phase illustrates the adaption of Gardner's intelligence as part of Thorndike's intelligence in the intelligence section. The third phase illustrates the adaption of swarm intelligence into the intelligence section. The fourth phase illustrates the adaption of Winfield's intelligence vectors into the intelligence section.

format within the intelligence section. Each skill was presented through a cell within the honeycomb. Each present skill available in the robot high-light's a cell in the honeycomb. The intelligence section has no limit to the number of skills and no boundaries for skills to be applied. Any skill manifest within the robot could be captured and presented. The skill is usually composed of several behaviors performed by the robot. The accumulation of behaviors expresses a skill and the collection of skills exhibits a specific intelligence. As the robot deploys a particular or a combined intelligence, the framework could express them through the sections and, accordingly, the framework is able to present the behaviors, skills and intelligence presented within the robot.

Winfield described robot intelligence according to the embodied intelligence that emerges from interaction and integration (Winfield, 2017). He categorized it according to four distinct types: social, morphology, individual and swarm intelligence. These categories supported the intelligence section of the framework and strengthened it through the literature. They also provide vectors for the intelligence dimensions, which maybe be measured, and this is presented in Figure 3.4 (fourth phase).

Adapting all four intelligence dimensions provides the framework with an extra section to demonstrate the obtained behavior of the robot (1) morphological / physical intelligence, (2) individual / abstract / intelligence (artificial cognition, exchanging information and planning), (3) social intelligence and (4) collective / swarm intelligence (any performance or behavior presented by the group as a whole). This intelligence section, along with sub-intelligence skills, are linked to the capabilities section, as presented in Figure 3.5. Robot intelligence thus describes the purpose of a robot's performance and the external perception of its actions (Schaefer et al., 2012). It also clarifies what sort of intelligence the designer aims to present in the robot by expressing the ultimate cause of the robot's actions/behavior as captured through specific skills (Fong et al., 2003; Kihlstrom and Cantor, 2000).

Linking intelligence and skill in this way is essential for describing the robot, developing a taxonomy and allowing for comparison between robots. The four intelligence dimensions cover the intelligences that might be acquired by the robot (Fong et al., 2003; Management-Mania, 2016; Pal et al., 2004; Winfield, 2017). The conceptual framework presents the dimensions necessary to describe robots and their characteristics though the capability and intelligence sections. The increasing development of robot capabilities over time requires a categorization structure that is flexible enough to present available robot characteristics and capture any new features and abilities, as noted by Dautenhahn (Dautenhahn, 2017). Therefore, the capability section, serves as an umbrella and allows for the inclusion of new capabilities as they arise. However, this requires the categorization to be presented in both detailed and abstract representation in order to handle any and all capabilities.

Adapting the causes of behaviors and actions

The structure was refined through conversations with robotics experts, in order, for example, to adopt the proximate and ultimate behaviors causation concepts (Marshall, 2017). This has been adopted from philosophy and biology into robotics. It considers the causation of behaviors and teleology (Dewsbury, 1999). Thus, the proximate cause of a bicycle moving is the pedaling of the cyclist but the ultimate cause may be more ambiguous: it could be for enjoyment or speed. In biology, a proximate cause of a bird's flight could be that it has wings, but an ultimate cause would be to escape predators. Using these concepts in the structure enhances its descriptive capabilities (Vessey and Drickamer, 2010). The

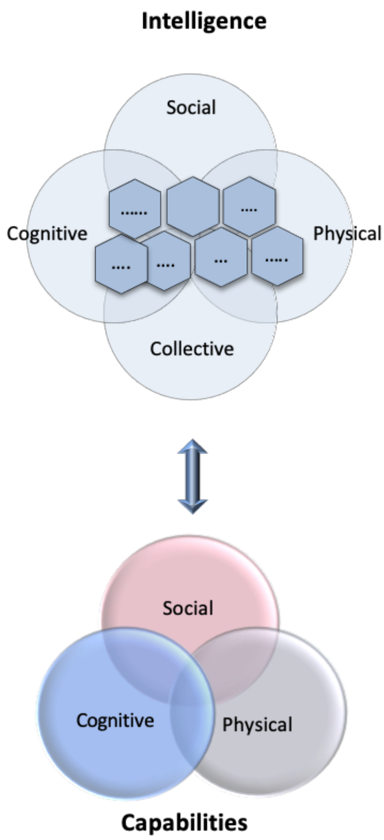

Figure 3.5: Presenting the robot capabilities with the three dimensions of physical, social and cognitive capabilities and the robot intelligence with the four main dimensions of physical/morphological, individual/cognitive, social and collective/swarm intelligence.

capability section is “proximate”, including the installed/developed systems in the robot, and the intelligence section is “ultimate” as it includes the goals for which the robot’s behavior and performance are intended. This differentiates between the installed programs and the reasons for the robot’s performance, as shown in Figure 3.6 (Burkhardt, 2005; Dewsbury, 1999; Scott-Phillips et al., 2011; Vessey and Drickamer, 2010). So using the first example mentioned earlier, if applied on robot, the proximate cause of the bicycle movement requires the robot to perform physical interaction by moving the bicycle pedals, however, the ultimate cause requires the robot to show speed or enjoyment as its goal.

The ultimate causation, “why the robot exists”, presents the goals of the robot developers in creating the robot. It also shows the external presentation of the robot by listing its skills and the robot user’s/designer’s perception of its goals. In contrast, the proximate section of the conceptual framework explains “how the robot performs required tasks” and hence what

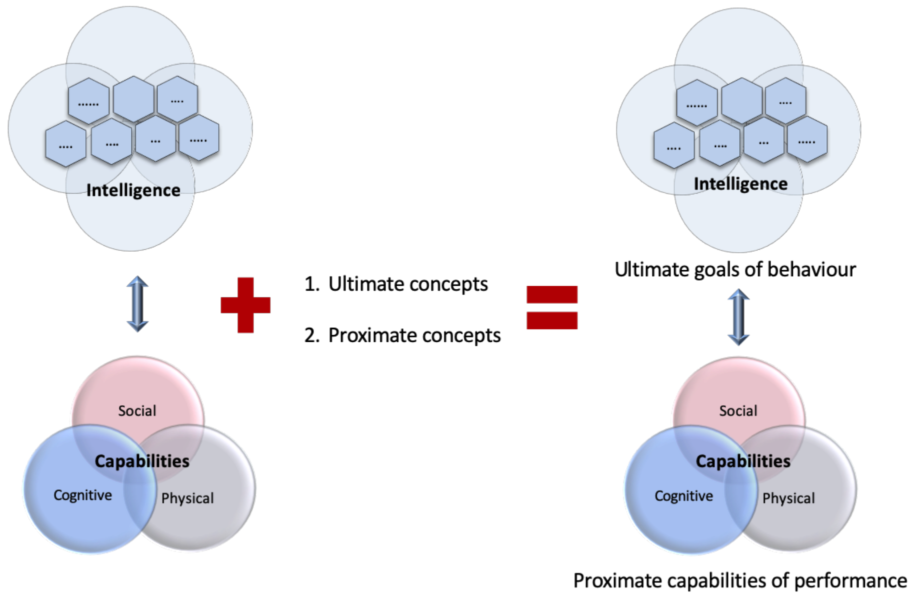

Figure 3.6: Adding the ultimate and proximate behavior into robotics allows for expressing the robot's actions through two different perspectives. The proximate behavior identifies the immediate actions performed by the robot and lists their mechanisms. It answers the "what is performed?" question. Where the ultimate behavior describes how the robot actions reflect on its desired obtained behavior, and answers "why the robot is performing its action?". Thus, adapting ultimate behavior and proximate behavior into robotics presents the proximate capabilities for the robot's performance and the ultimate reasons for its behavior.

the robot is performing. The proximate section lists the robot's potentiality in executing tasks through its capabilities. These aspects are shown as sub-systems deployed within the robot.

The capabilities and intelligence/skills performance of a robot depend on its features (Dautenhahn, 2007). Therefore, the robot features were added as an additional layer. This layer was situated below the capabilities and intelligence/skills as it is the foundation for these. This layer for robot features includes the structure, software and hardware. In turn, robot interactions limit performance according to the robot's capabilities, which set the boundaries of its execution. Therefore, interaction was allocated between the capabilities and intelligence section, as illustrated in Figure 3.7.

The main structure contains two main layers the features layer and the capabilities layer. The first layer covers all the robot HW and SW aspect. The second layer of the framework presents what the robot is designed to perform (robot intelligence and skills) and includes the capabilities deployed to achieve its performance goals (defined in the first section of the structure as a list of robot capabilities). This, as suggested by Dautenhahn (Dautenhahn,

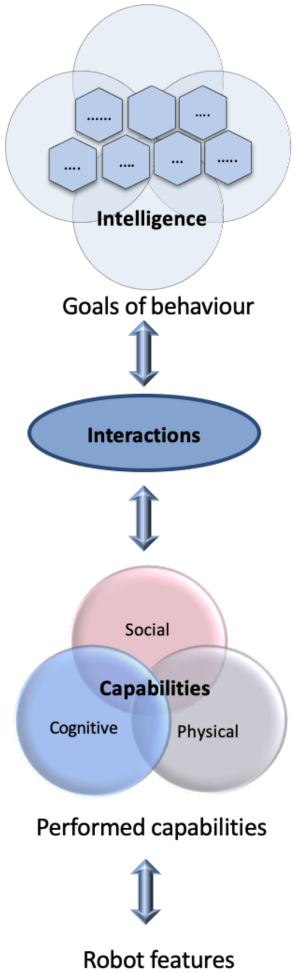

Figure 3.7: Interaction depends on capabilities and a robot's physical characteristics. Interaction presents behavior that affects robot skills and intelligence. Therefore, interactions were allocated between intelligence and capabilities. Robot features affect robot behavior therefore robot feature was allocated under the robot capabilities.

2017), links the two sections through the cause and effect of an action, that is, through human robot interaction (HRI). The interaction capabilities map between the two sections to describe the robot. However, even where HRI with a social context is not explicit, such as when there may no social interaction, there will be physical or cognitive interaction (Dautenhahn, 2017). In refining the conceptual structure, therefore, social, physical and cognitive interactions were added between the two sections, the capabilities section and the intelligence section, as each section presenting different properties.

Performing any of these three interactions is shown through the execution of the capabilities developed within the robot. The interaction reveals and demonstrates the skills, behavior, and intelligence of the robot (Rogers et al., 2005). Therefore, the interactions link the robot capabilities and the robot intelligence/skills section and thus it is allocated between them, this is shown in Figure 3.8. Additionally, these three robot interactions are also considered as capabilities themselves as referenced in a lot of the robotic literature (Mavridis, 2015), which reinforces their location within the capabilities section. Therefore, robot capabilities consist of several sections: The first section illustrates capabilities performed by the robot. The second section illustrates interaction capabilities accompanied by the robot performance and the third section is the intelligence section that illustrates the robot capabilities by obtaining some goals in building skills and behavior. The intelligence, skills, and behavior demonstrate the reasons for the robot achievement. All three sections are part of the technical capability layer, as each section describes a specific aspect of the robot capabilities.

These three types of interaction capture the consequences of robot actions. Whenever the robot executes any action, it must be determined by reciprocal exchangeable action. These inverse inter-activities could be either for communication between social agents, direct involvement with objects or even for interchanging signals, data or information between the automated systems. These interactions present the distinct form the robot is practicing in affecting its surroundings. Human interaction in robotics is always considered as HRI and it is always refereed with social context. However, not all robots perform human robot interactions as some robots perform only physical interaction where other exchange data signals, defined in this research as cognitive interaction. Nevertheless the framework needs to capture all three types of interaction: social, physical and cognitive interactions (Dautenhahn, 2017).

Therefore, the interaction section includes these three types: social, physical and cognitive (Yanco and Drury, 2002). Social interaction is defined in MAR (EU-Robotics, 2016) as any interaction performed by robots towards social agents, including human (Fong et al., 2003). Physical interaction includes any physical movements presented during interactions, such as manipulating an object (Cherubini et al., 2016). Cognitive interaction includes any digital interaction performed by the robot and any digital device, either through sending or receiving digital information, signals or data (Rastic-Dulborough, 2014).

Defining the different reasons for robot interactions depends on the robot capabilities, which in turn depend on the robot features (Fischer, 2017). Therefore, interactions were located over the capability section and feature layer, and as the capability depends on the robot feature, the capability section is located over the feature layer. Moreover, the interaction of robot capabilities illustrates some interactivity presented as interaction cycle. This cycle is also defined by Winfield (2012)) as the ‘robot world feedback loop’ that captures the

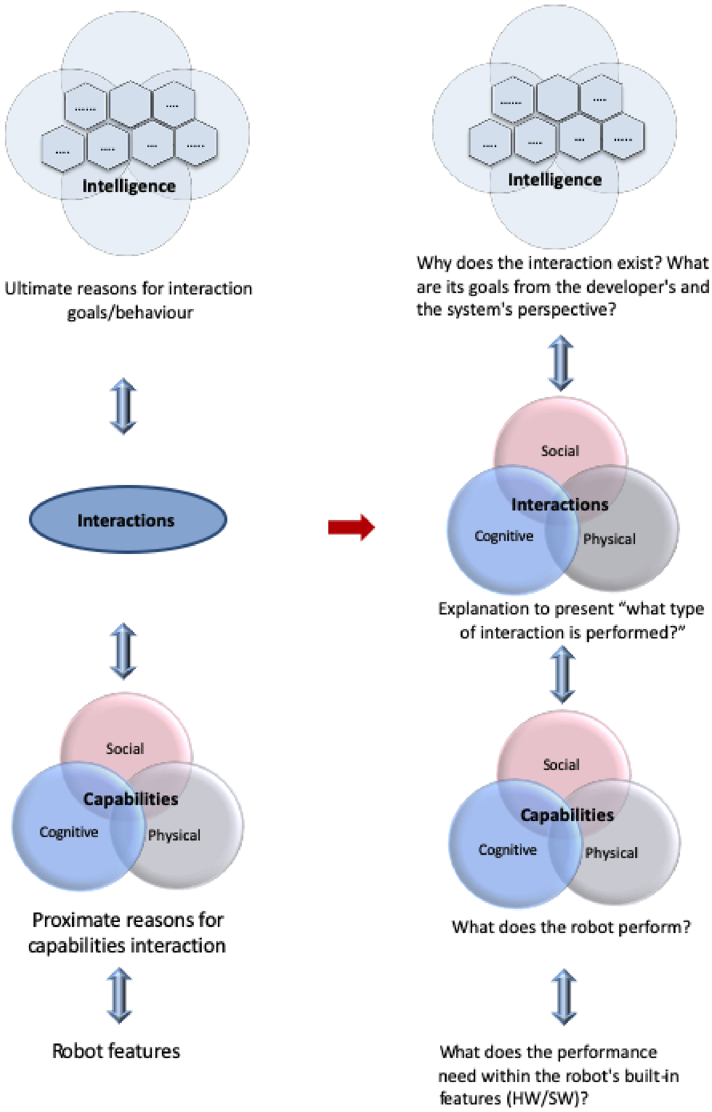

Figure 3.8: Adapting and allocating robot interaction to the structure as part of the technical capability section but between the capability section and the intelligence section. The interaction section contains three types of interactions: physical, cognitive and social; which are the outcome of the robot capability performance. Therefore, it was allocated above the robot capability section. These interactions also are designed to present the robot with skills, goals and behaviors, as they shape the robot intelligence, hence they were allocated below robot intelligent section.

dynamic rotations of actions. These cycles demonstrate specific behaviors for certain skills, or vice versa, that demonstrate robot intelligence, and which is listed in the intelligence section. Therefore, the interaction affects robot behaviors, which in turn imply specific skills/dimensions of intelligence (Saffiotti, 2017), which is, therefore, the interaction is located between the capabilities section and the intelligence section. Relations between the dimensions are presented as described in a specific order in the framework shown in Figure 3.9.

To summarize, two main layers were developed: The first layer is "the robot features layer", that captures robot HW/SW, and supports the various robot capabilities. The second layer "the capabilities layer", captures different types of capabilities, and consists of three main sections: general performed capabilities, interaction capabilities and intelligence/skills capabilities. The first section is the performed capabilities: this captures any capabilities developed in the robot. The second section is the interaction capabilities: this captures any type of interaction performed by the robot. And the third section is the intelligence/skills capabilities: this captures the skills, behavior and robot goals. The interaction capabilities section is located between the performed capabilities of the robot and the intelligence/skills capabilities since it links between both sections. Interaction depends on robot capabilities and supports the robot with the skills to present special intelligence, as illustrated in Figure 3.9. By using this hierarchy as a tool in defining robot characteristics, a method to define robot capabilities can be established, enabling capabilities scoring for the robot (Bisset, 2017) and generating the robot capabilities profile. Some applications require specific types of robot characteristics, features and goals of performance, hence the framework groups these and presents them clearly in layers and sections to be checked for each robot. Therefore, it bridges the gap between the robotics fields and application domains by a filtering system that matches robot characteristics and requirements. Such a mapping framework is outlined briefly in the next section, as it provide the main suggestions for the future work of this research.

Adapting relational model between application domains and technology

The mapping between robot applications and domains is considered one of the challenges in robotics. The mapping process should cautiously link requirements and technology based on mixing/matching between robot characteristics and application requirements. It also leads in discovering or assigning new dimensions while bridging the two sides. This requires characterizing each side individually, to find out a corresponding connection between the two dimensions. A similar mapping has been presented in an abstract framework for spoken language applications and their related technology (Moore, 2000). Adapting this framework to map between robot applications and robot domains, as shown in Figure 3.10, provides several advantages. It enables a user to select the application requirements for a task and it identifies which robot would be appropriate for such requirements. Also, it identifies

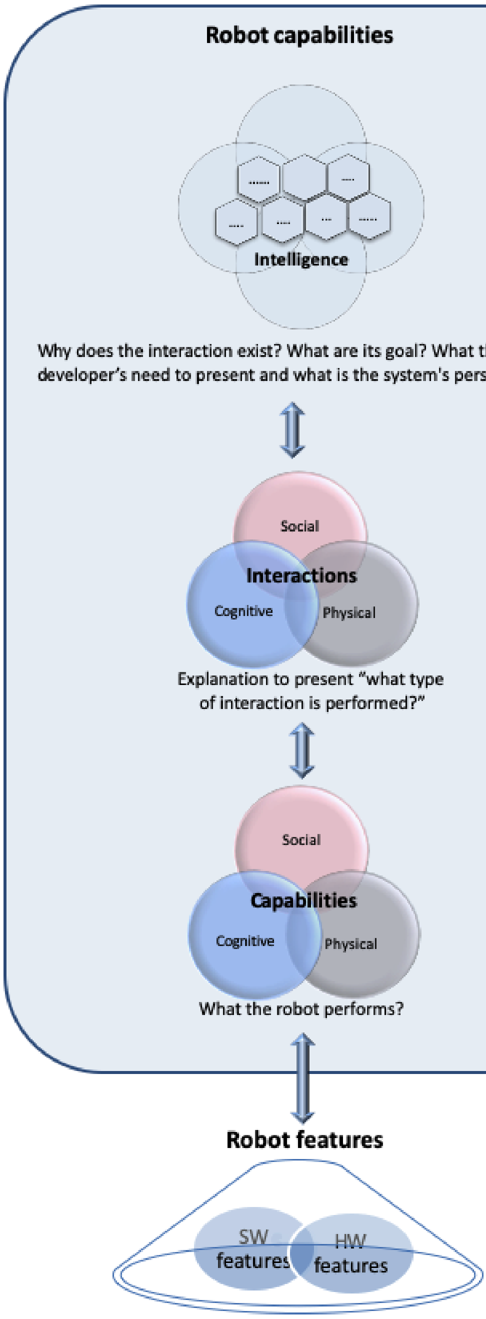

Figure 3.9: Introducing the robot features into the structure. The robot features contains the robot HW and the robot SW. The HW and the SW are allocated together in a funnel that leads to capabilities layer. As any capability section in the capabilities layer would depends on the HW and the SW of the robot. Therefore, all of the three different sections of capabilities are grouped together in the capabilities layer.

which potential application is best for a particular robot. However, difficulties occur in this mapping process through the update of information or lack of knowledge, either of the existing technology or in relation to listing user requirements.

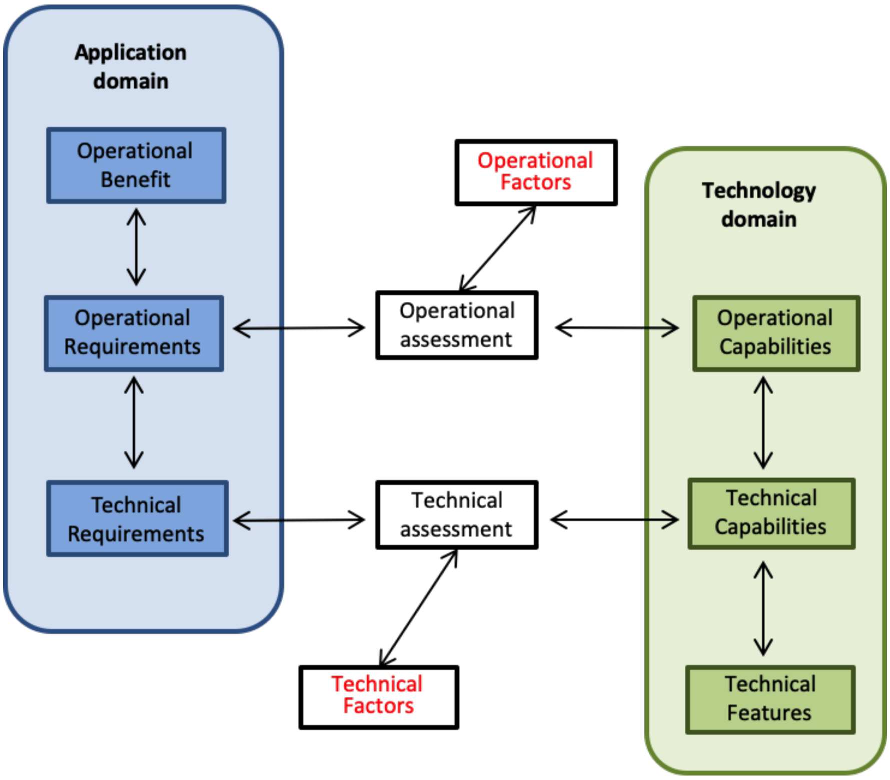

Figure 3.10: A model of the relationship between applications and the underlying technology. The operational requirement and the technical requirement are also known as functional and nonfunctional requirements.

Nevertheless, as long as the application requirements and robot specifications are updated and outlined, and the bridging between the domains is clearly defined, the mapping process should be straightforward. Therefore, it is important to identify the dimensions and characteristics in the mapping process between robot applications and their technological development. The adaption of the “relationship” framework for this process clarifies that the three layers enable bridging between the two domains and define the mapping process between applications and their robots. Also, adapting the three layers into the conceptual framework provide it with: (1) technical features layer, (2) technical capabilities layer and (3) operational capabilities layer. Grouping the technical capabilities in the framework in one layer allows them to be mapped to the technical requirements of the application. Grouping the operational robot characteristics within an operational layer enable them to be mapped by linking the operational requirements of the application (Breazeal, 2017a). Thus, adapting Moore's "relationship" model allows for the mapping, linking and presentation of new specifications, as presented in Figure 3.11.

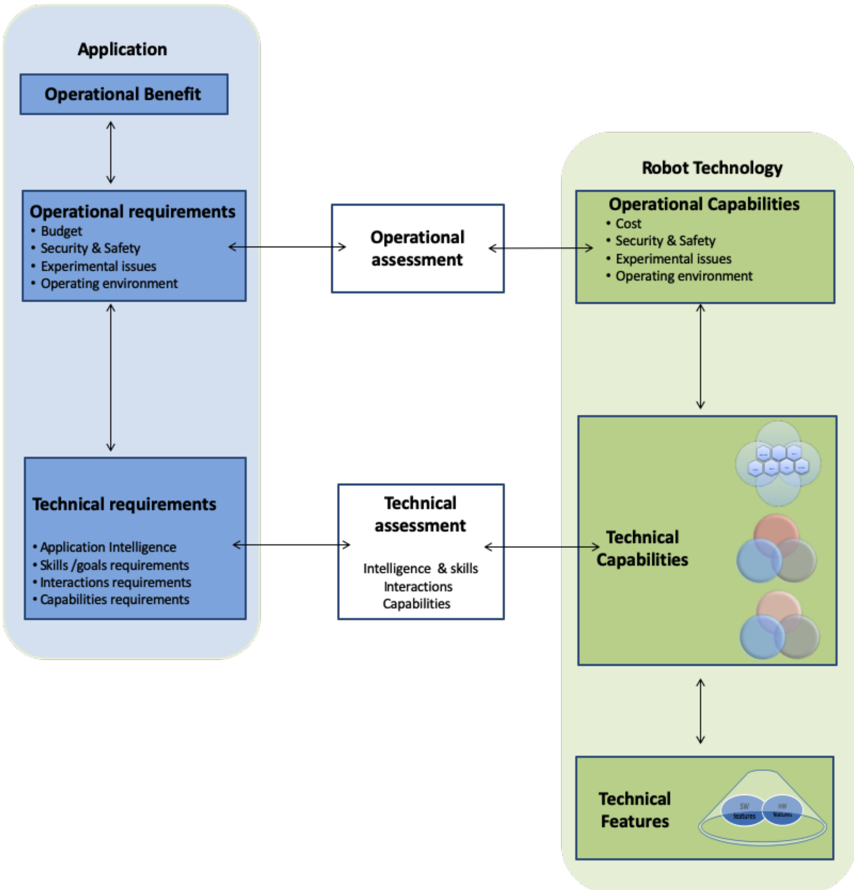

Figure 3.11: Adapting model from speech and technology into robotics, to include the three layers of different mapping dimensions between the application domain and the robotics domain.

Adapting the relationship model into the conceptual framework supports the categorization of robot characteristics into three main groups. Each group is presented in a layer: a robot features layer to include the hardware and software; a technical capabilities layer to include the different robot capabilities, including interactions and intelligence; and an operational capabilities layer to capture all of the operational aspects of the robot. A detailed description of the robot characteristics categorization is illustrated in Figure 3.12.

The categorization captures most robot characteristics and features and includes them in the framework. Examples of these robot characteristics are perception and performing action; as common capabilities were added to the framework in the capability sections. These capabilities are gathered according to a perception-action concept and allocated in

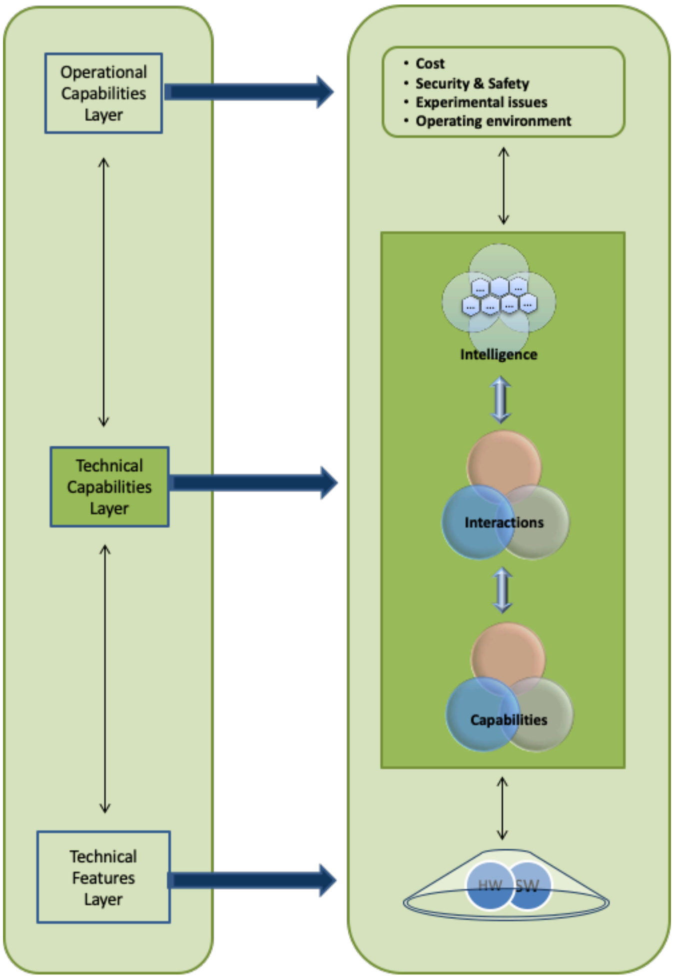

Figure 3.12: Applying Moore's relationship model into the hierarchy of robot characteristics categorizes the characteristics into three main groups: (1) Characteristics to specify the robot hardware and software that are mapped into the robot features layer in Moore's model and in the first layer of the structure. (2) Characteristics to specify differences in robot capabilities are mapped into the technical capabilities layer in Moore's model and in the second layer of the structure. But these characteristics are subdivided into three types in the structure. The first characteristics type is the performance capability of the robot, which is covered in the first section of the second layer "capabilities" of the structure. The second type of capabilities is the robot interaction, which is covered in the second section of the second layer in the structure, the "interaction" section. The third type of characteristics is the robot ability to present some skills, behavior and intelligence, which is covered in the third section of the second layer in the structure, the intelligence" section. (3) Finally characteristics that are part of the operational capabilities are mapped to the third layer in Moore's model and in the "the operational capabilities layer" of the structure.

the capabilities section of the framework. Together, perception and action present a cycle of interaction (Bodnera et al., 2011), supporting the allocation of interaction capabilities at a level above the different robot capabilities and performances. Allocating these capabilities within this setting does not present any actual performance of any robot but it allows for simply visualizing the capabilities and differentiating between them. Most of these robot performance characteristics are included and presented in Figure 3.13. Furthermore, to match the conceptual framework, each of the capabilities is categorized according to the three major types: physical, social and cognitive capabilities.

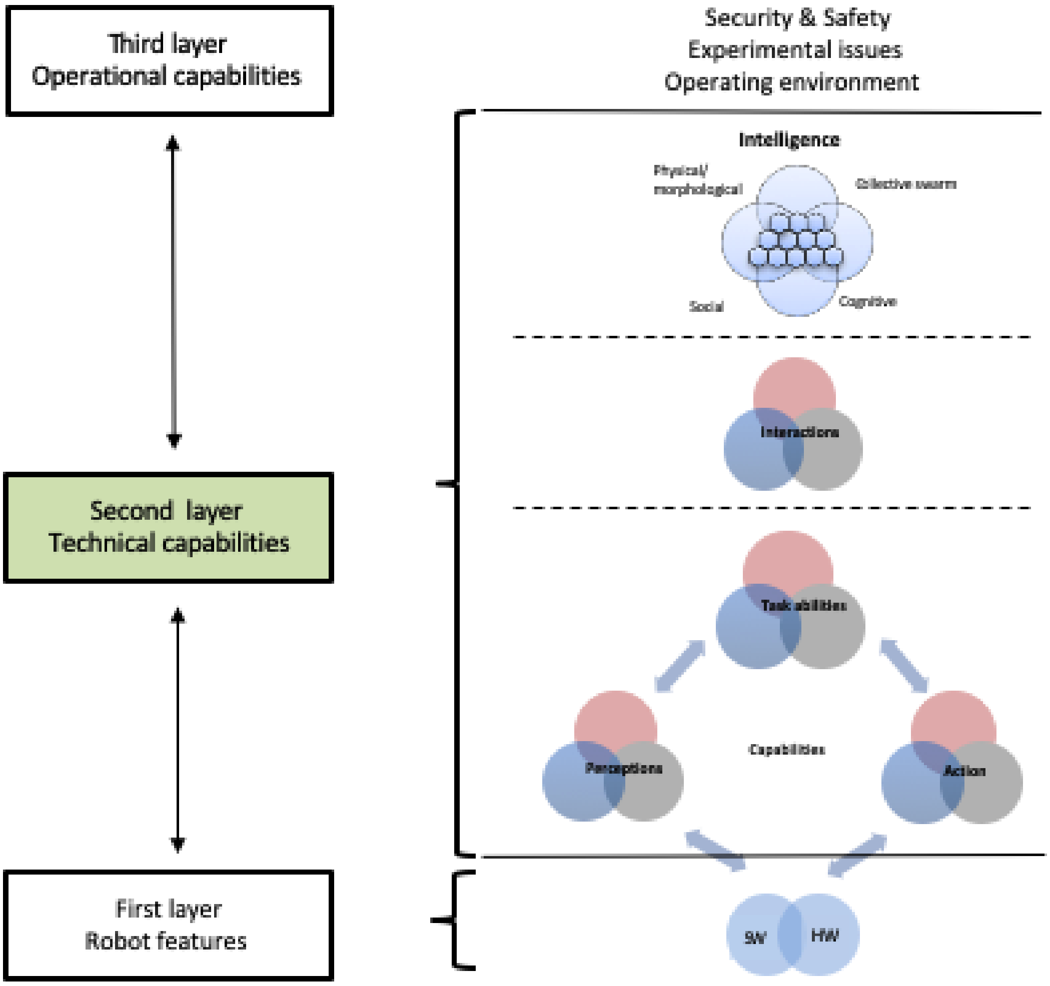

Figure 3.13: Robot capabilities to include robot input and output abilities such as perception task abilities and future actions. It also include interaction capabilities and its goals, skills and intelligence. The setting of the structure simplifies visualizing the capabilities that it encompasses.

3.3.3 Adapting MAR

MAR covers a limited list of robot abilities (EU-Robotics, 2016). Each ability contains a list of sub-capabilities and each of the sub-capabilities is provided with maturity levels. The maturity levels present the progress, achievement or development stages of the sub-capabilities. The level concept was adapted from the Technology Readiness Levels (TRL) and each of the sub-capabilities is evaluated according to its related specific parameters. Thus, each level captures an additional degree of performance. All sub-capabilities start at level 0, which indicates no capability presented and as the capability improves the indicator increases and moves up a level to mirror the effectiveness of the capability. Having the MAR list of sub-capabilities and their levels within the conceptual framework in the capabilities layer is a good starting point to measure the capabilities. It can evaluate robot capabilities with specific and defined measures, as suggested by Walker (Walker, 2017). If a sub-capability does not exist within the robot, it will be assigned level 0. If a sub-capability exists within the robot, it is assigned to the level matching the robot's performance. These levels allow the capabilities of any given robot to be defined through specific numeric values which represent the technological maturity of the capability. Therefore, the adaption of MAR levels provides the conceptual framework with an established quantifiable evaluation system. Such an evaluation system enables roboticists to present and manage capability levels within their robots. It also supports the design and development of robot capabilities (Deutsch et al., 2010). On the other hand, adding these levels to the application domain will help to describe application requirements using the same scheme that was used to describe robot capabilities. Using the scheme between the two domains aids in the mapping process. The application domain and its application requirements are not covered in this thesis but they are proposed as a main concept that needs to be researched for future work.

Allocating the MAR sub-capabilities There are nine capabilities presented in MAR: configuration ability, adaptability, interaction ability, dependability, motion ability, manipulation ability, perception ability, decisional autonomy and cognitive ability. Each of these capabilities is presented with sub-capabilities. The listed capabilities in MAR were allocated in the sub-capabilities sections of the conceptual framework. The thematic analysis indicated where MAR did not define capabilities or sub-capabilities, and which are missing in the conceptual framework, as they need to be created and developed, as presented in Table 3.1. Therefore, a general capabilities section was added to the conceptual framework as part of the performance capability section, above what could be termed "independent capabilities". Thus, task adaptability, dependability, and decisional autonomy are mapped to the "general capability" section. MAR presents three types of interaction capabilities: social, physical and cognitive interactions. The available MAR interaction capabilities are allocated to the conceptual framework's in the second section, "the interaction section".

Table 3.1: Allocation of MAR sub-capabilities in the framework's sections and layers.

MAR sub-capabilities	**ToRCH Framework**
Mechatronic configuration	Feature, 1st layer
Parameter adaptability	Feature, 1st layer
Component adaptability	Feature, 1st layer
Object perception	Perception capability, independent section, 2ed layer
General perception	Perception capability, independent section, 2ed layer
Tracking perception	Perception capability, independent section, 2ed layer
Self-location perception	Perception capability, independent section, 2ed layer
Environment scene perception	Perception capability, independent section, 2ed layer
Interpretive	Perception capability, independent section, 2ed layer
Un/constrained motion	Task capability, independent section, 2ed layer
Grasp, hold, hand ability	Task capability, independent section, 2ed layer
Learning and reasoning ability	Task capability, independent section, 2ed layer
Action and envisioning ability	Task capability, independent section, 2ed layer
Task adaptability	General section, 2ed layer
Dependability	General section, 2ed layer
Decisional Autonomy	General section, 2ed layer
HR interaction feedback	Interaction capability section, 2ed layer
Robot-to-robot interaction	Interaction capability section, 2ed layer
Human interaction levels of extent	Interaction capability section, 2ed layer
Interaction complexity	Interaction capability section, 2ed layer
Human interaction modality	Interaction capability section, 2ed layer
Social interaction learning	Interaction capability section, 2ed layer
HRI methods	Interaction capability section, 2ed layer
Object interaction	Interaction capability section, 2ed layer
Human interaction cognitive ability	Interaction capability section, 2ed layer
HR interaction safety	Operational, 3ed layer

The next chapter presents such capabilities not available within MAR but which are essential and common to characterize robots in the robotics domain.

❋ Chapter 3: Summary

The proposed framework covers the most important characteristics used to describe robots. It presents them hierarchically in different layers and levels according to their relationship with each other. The framework contained three main layers (robot feature, technical layer and the operational layer). All of the listed MAR capabilities and interactions were adapted and mapped into the conceptual framework as a quantitative basis for measuring capabilities. The MAR capabilities are intended for the European robotics community, and it has been heavily dependent on specific industries. Therefore, the MAR capabilities did not provide a comprehensive list of capabilities, and the adopted capabilities were not enough for a comprehensive framework. As a result, the conceptual framework required further development, especially for the areas that had not adopted any of the MAR capabilities.

Chapter 4

Beyond MAR

4.1 Introduction

The proposed conceptual framework needs to include all the primary robot characteristics and capabilities within its hierarchical dimensions. As noted in Chapter 3, the adopted MAR list of capabilities is not comprehensive and reflects the interests of the European robotics community, as capabilities are developed within robotics in relation to research and grant interests and the needs of industry. Thus, MAR does not currently include, for example, emotional capabilities, certain behavioral capabilities or even social agent perception capabilities. These missing capabilities need to be identified and allocated in the conceptual framework at suitable levels. This chapter will describe the process of capturing and adding new robot characteristics and developing more capabilities with their levels.

4.2 Defining capabilities and filling the gaps

The limited list of capabilities within MAR was adopted within the conceptual framework providing some sections with sub-capabilities and measures. However, other sections were left with no defined capabilities. The conceptual framework sections are listed with their corresponding adopted capabilities in Table 4.1. The interaction capabilities presented in the framework are listed as follows, where each section was allocated with some of the MAR interactions: The physical interaction - any interaction between the robot and any visible/tangible object in its surroundings. The section covers the MAR list of interaction capabilities such as object interaction. Social interaction to cover interactions between the robot and any social agents in the robot's environment. This section includes any interaction capability types with social agents that are presented in MAR, such as human-robot interaction feedback, human-robot interaction modality, and human interaction level of extent. Cognitive-signal informatics interaction covers the interaction of signals, data or simple text that are presented in MAR, such as robot-to-robot interaction. Social/cognitive interaction that cover any social interaction with learning and cognitive ability, such as human-robot interaction complexity, social cognitive interaction with learning, and human interaction ability with cognitive concepts. Physical/cognitive interaction that covers learning while physically touching. No capabilities were listed under this section, which indicates it should be developed. Social/physical interaction covers physical contact for social purpose. No capabilities were listed under this section in MAR, which indicates it should be developed. Physical/social/cognitive interactions, such as interaction control methods.

Table 4.1: Presenting the sections of the conceptual framework, some with the adopted capabilities and others for which new capabilities needed to be developed

Framework layer	Framework Sections	Source
Features	Mechatronic configuration Parameter adaptability Component adaptability	MAR
Technical capabilities	Physical perception Cognitive perception Social perception	MAR Novel development needed Novel development needed
Technical capabilities	Physical interpretive Cognitive interpretive Social interpretive	MAR Novel development needed Novel development needed
Technical capabilities	Physical task Cognitive task Social task	MAR MAR Novel development needed
Technical capabilities	Physical purposefulness Cognitive purposefulness Social purposefulness	MAR Novel development needed Novel development needed
Technical capabilities	Physical envisioning Cognitive envisioning Social envisioning	MAR Novel development needed Novel development needed
Technical capabilities	General	MAR
Technical capabilities	Physical interaction Cognitive interaction Social interaction	MAR & development needed MAR & development needed MAR
Operational capabilities	Not included in this research	

Allocating all of the possible robot capabilities presented in the conceptual framework would help in defining and deploying the robot in the robot application domain. This section reinforces the conceptual framework by adding capabilities within the different layers of the framework. Each capability section is defined with various sub-capabilities to capture the performance of main capabilities. The sub-capabilities include different levels, where each level indicates a specific degree of technological maturity.

4.3 The first layer: robot features

The first layer of the conceptual framework, “the features layer”, describes the structure of the robot’s hardware (HW) and software (SW). The hardware includes sensors, actuators, kinetics, kinematics, embodiment, morphology, locomotion and so on. The software includes the operating systems (e.g., ROS or YARP), architectures and any programmable modules installed within the robot (e.g., visual or tactile processing) (EU-Robotics, 2016). This layer also covers networking and memory concepts such as Wi-Fi and i-cloud. The following sections present the HW and SW characteristics that belong to the features layer: the first section presents some hardware/software capabilities and the second section presents some important measures of the HW and SW components available within this layer.

4.3.1 Novel capabilities for the first layer

MAR uses capability levels to describe the technological maturity of HW and SW, such as mechatronic configurability, component reliability and parameter adaptability. These HW/SW low-level capabilities have been adopted into the features layer of the conceptual framework. This allows for the presentation of the advancement of HW and SW components. In this research, no novel capabilities were developed to this layer.

4.3.2 Measurable parameters for the first layer

Although no new capabilities were added, the features layer draws on the wider literature to include some measurable HW/SW robotic aspects. These robot HW/SW measures describe the robot components specifically and the robot capabilities in general. These parameters are listed as follows:

1. Robot presentation/performance types: real, hologram, simulation, mixed reality (Kanda and Ishiguro, 2004; Wainer et al., 2006).
2. Communication dimensions: communication medium, communication media, and the form of communication (Goodrich and Schultz, 2007).
3. Robot interface as robot input/output:
   - Robot input sensors, such as camera, microphone, tactile sensors and command from device.
   - Robot output sensors, such as identification, sound, movements, light within the robot.
4. Robot design approach for robot development, such as biological design, functional design and design theory.
5. Environment and context: the framework initially did not include any context for the robot capabilities. This section acknowledges the circumstances that form the setting in which the robot operates. This oversight was highlighted by Dautenhahn (Dautenhahn, 2017), so the environmental settings were added to the first layer along with parameter configurability, size and weight as these are linked to the HW and SW of the robot.

The hardware and software determine the robot's capabilities and the types of interactions it can perform (Breazeal, 2004; McGrenere and Ho, 2000). Therefore, the kinetics and kinematics of the robot define its possible actions as well as its constraints, and determine the suitability of the actions (Ziemke, 2003). This implies that robots with different physical components have different capabilities. Defining the main HW and SW will predict all the possible capabilities within the robot. Therefore, the list of capabilities is presented in the next layer, which is defined as "the robot capabilities layer".

4.4 The second layer: technical capabilities

The second layer of the conceptual framework captures the technical capabilities. Any capability performed by the robot is presented in this layer. The following sections cover some characteristics of the technical capabilities layer: the first section captures novel robot capabilities (developed for this research) and the second section presents measurable parameters that affect the robot's technical capabilities.

4.4.1 Novel capabilities for the second layer

Capabilities development procedures

Different robots have different capabilities and it is impossible to list all available robot capabilities, but it is important to demonstrate how to present a capability in a measurable notation. Any capabilities can be developed according to the generic capability development procedure.

The procedure used to create novel capabilities is as follows:

- The first level is always set at zero, indicating no performance capability.
- The second level indicates some performance of the capability according to a predefined variable.
- The third level indicates a predefined set of variables that increase the maturity of the capability performance.
- The fourth level defines flexible variables that affect the dynamic performance of the capability.
- The fifth level defines several environmental variables affecting the performance of the capability.

Importantly, the leveling above zero may encompass more than one level. For example, recognition includes recognition of a single sample or one of many samples. Both capture the recognition capability, but each additional level indicates an improvement of the performed capabilities.

Capabilities development

This section presents the novel capabilities and sub-capabilities that have been added to the conceptual framework. The capabilities were identified and the sub-capabilities were developed according to the generic capabilities development procedure, presented above in section 4.4.1. The capabilities and the novel sub-capabilities, with their maturity levels that needed to be developed for this framework, are presented in Table 4.1.

Perception capabilities

Perception capabilities capture the performance of perception towards different things. Physical perception was adopted from MAR, but social and cognitive perception were developed using the generic procedure in section 4.4.1, as presented in Figure 4.1. The following list covers the most important perception types, where each type was developed with levels:

1. Developing perception to capture any digital information gathered by signals or digital data as part of the cognitive perception, presented in Figure 4.1 as perception of digital information. This perception

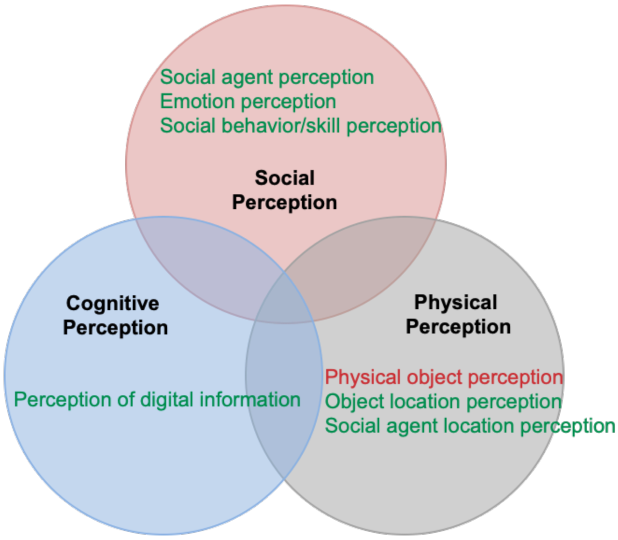

Figure 4.1: Distribution of the robot perception capabilities. Capabilities presented in red text were adopted from MAR and capabilities presented in green text were developed for the framework.

of digital information is presented in five levels using the generic development capability procedure presented in section 4.4.1 above and using the language of MAR object perception levels as a structure (EU-Robotics, 2016).

- Level 0 - No perception of signals or digital information.
- Level 1 - Data/signals (feature) detection: sensing data/signals (such as bits, letters, an acoustic signal, graphics) gathered from a specific location and mapped to a meaningful model.
- Level 2 - Data/signals detection: the system makes sense of data to recognize a specific topic based on feature extraction. The multiple persistent features can be grouped to build models of distinct types of information, which allows information of different types to be differentiated from each other.
- Level 3 - Information recognition - single instance: the information models created from sense data can be matched to specific known instances of information with a reliability that is appropriate to the task.
- Level 4 - Information recognition - one of many: the information models created from sense data can be matched to one of a number of specific instances of known information with a reliability that is appropriate to the task.

As can be seen, levels 1 and 2 cover the detection; where levels 3 and 4 cover recognition.

2. Developing perception to identify humans with different degrees of maturity, presented in Figure 4.1 as social agents perception. This type of perception is allocated within the social perception type. This identification of social agent perception is presented in nine levels. Again, the generic procedure in section 4.4.1 was used, along with the language of MAR object perception levels (EU-Robotics, 2016).

    - Level 0 - No human social agent recognition.

- Level 1 - Feature detection to sense an aspect of a human social agent: sense data is gathered from the environment, and can be mapped onto human profile model. The richness of the sense data is such that it is possible to apply a feature-detection process to create a set or sets of features of a person.
- Level 2 - Human social agent detection of a whole social agent: multiple persistent human features can be grouped to build a distinct human profile that allows humans to be differentiated from each other and from the environment.
- Level 3 - Person recognition - single instance: a human social agent model created from sense data can be matched against a specific human social agent model in the database with a reliability that is appropriate to the task.
- Level 4 - Person recognition - one of many: matching a person to one of several defined possible human social agent identities. The human models created from sense data can be matched against several known identities with a reliability that is appropriate to the task.
- Level 5 - parameterized person recognition: recognizing a generic human social agent by matching the person feature with known parameters. The models created from sense data can be matched to a number of known human social agent parameters. The settings for the parameters (e.g. size ratio, gender specification) can be deduced from the sensed human social agent model.
- Level 6 - Context-based recognition: improved human social agent recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognize human social agents by reducing ambiguities through expectations based on context or location.
- Level 7 - Person-variable recognition: the system is able to recognize a human social agent where there are variations in his/her profile, such as haircut and skin color, Thus, a person is recognized according a change in some variables according to a specific scale.
- Level 8 - Novelty recognition: the ability to recognize novelty in a human social agent. The system is able to recognize a specific feature in a person that may have changed, such as by identifying a specific scar on a person.

3. Developing perception to identify emotion, mode or attitudes (Kirby et al., 2010), presented in Figure 4.1 as emotion perception. This type of perception is allocated within the social perception type. This capability identifies and recognizes any social agent emotion, mode or attitudes and it is presented over in six levels. The levels were developed using the generic procedure in section 4.4.1, with the language of MAR object perception. However, the first level, “emotion features” was not referenced as it was not applicable (EU-Robotics, 2016).
    - Level 0 - No emotion perception.
    - Level 1 - Emotion detection: the robot is able to sense an aspect of emotion. Sense data is gathered from the social agent and can be mapped. The richness of the sense data is applied to detect some features of social agent emotions.
    - Level 2- Emotion recognition - single instance: an emotional model created from sense data can be matched against a specific model in the database.
    - Level 3 - Emotion recognition - one of many: matching an emotional model to one of several defined possible models. The emotional models created from sense data can be matched against several known identified models with a reliability that is appropriate to the task.
    - Level 4 - Parameterized emotional recognition: recognizing a generic emotion by matching the emotional feature with known parameters. The emotional models created from sense data can

be matched to a number of known emotional parameters. The settings for the parameters can be deduced from the sensed model.

- Level 5 - Context-based emotion recognition: improved emotion recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognize emotion by reducing ambiguities through expectations based on context or location.

4. Developing perception to identify social behavior/skills, presented in Figure 4.1 as social behavior/skill perception. This type of perception is presented in six levels. Again, the levels were developed with the generic procedure in section 4.4.1 and using the language of the first six levels of MAR for object detection (EU-Robotics, 2016).
    - Level 0 - No behavior/skills perception.
    - Level 1 - Social behavior/skills feature detection: the robot is able to sense an aspect of social behavior. Sense data is gathered from the social agent and can be mapped. The richness of the sense data is applied to detect some features of social behavior.
    - Level 2 - Social behavior/skills recognition - single instance: a social behavior model created from sense data can be matched against a specific model in the database.
    - Level 3 - Social behavior/skills recognition - one of many: matching a social behavior model to one of several defined possible models. The social behavior models created from sense data can be matched against several known identified models with a reliability that is appropriate to task.
    - Level 4 - Parameterized social behavior/skills recognition: recognizing social behavior by matching the behavior with known parameters. The social behavior models created from sense data can be matched to a number of known behavior parameters. The settings for the parameters can be deduced from the sensed model.
    - Level 5 - Context based social behavior/skills recognition: improved social behavior recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognize social behavior by reducing ambiguities through expectations based on context or location.

5. Developing perception to identify the location of an object, presented in Figure 4.1 as object location perception. This capability defines the maturity levels in defining object location and is presented in eight levels. The generic procedure in section 4.4.1 is used, with the language of MAR self-location perception as a framework (EU-Robotics, 2016).
    - Level 0 - No perception of object location either in terms of its position relative to the environment or with its own location.
    - Level 1 - Object position: the robot identifies the object location as a result of object perception.
    - Level 2 - External beacons: the robot identifies the object location as a result of information derived from the inspection of external beacons. Beacons may be active or passive and include global beacons.
    - Level 3 - Relative location: the system is able to calculate the object location relative to a specific defined location with a degree of accuracy that is sufficient for the task.
    - Level 4 - Feature-based location: the system calculates the object position within an environment based on the motion of fixed features in the environment, such as by using SLAM to build and maintain a local map.
    - Level 5 - Mapped location: the robot is able to relate the object location position to a map that it has been given or that it has

- Level 6 – Spatial occupancy: the system calculates the position of the object structures based on indirectly gathered sense data to provide a spatial notion of occupancy.
- Level 7 - Object-coupled location: the system is able to calculate the position of an object in conjunction with other objects it is connected to, such as to an object that is being gripped by the robot.

6. Developing perception to identify the location of a social agent (either human or other social agent), presented in Figure 4.1 as social agent location perception. It is presented in eight levels using the generic procedure in section 4.4.1 alongside the language of MAR self-location perception (EU-Robotics, 2016):
    - Level 0 - No perception of social agent location either in terms of its position relative to the environment or with its own location.
    - Level 1 - Social agent position: the robot identifies the social agent location as a result of perception.
    - Level 2 - External beacons: the robot identifies the social agent location as a result of information derived from the inspection of external beacons. Beacons may be active or passive and include global beacons.
    - Level 3 - Relative location: the system is able to calculate the social agent location relative to a specific defined location, with a degree of accuracy that is sufficient for the task.
    - Level 4 - Feature-based location: the system calculates the social agent position within an environment based on the motion of fixed features in the environment, such as by using SLAM to build and maintain a local map.
    - Level 5 - Mapped location: the robot is able to relate the social agent location position to a map that it has been given or that it has acquired. This may be a location within a task-relevant space.
    - Level 6 – Spatial occupancy: the system calculates the position of the social agent structures based on indirectly gathered sense data to provide a spatial notion of occupancy.
    - Level 7 - Social agent coupled location: the system is able to calculate the position of a social agent in conjunction with other objects or agent it is connected to.
7. Development of the modes of perception to define the robot's collection methods to perform perceptive capabilities. These methods could be, for example, visual, auditory, olfactory, physical (mechanical, magnetic, chemical) or a combination of these.

Interpretation capabilities

Interpretation capability is the ability of the robot to combine the different models of sensed data into a high-level description and to create knowledge used for actions. Within interpretation capabilities, the physical object interpretation was adopted from MAR; however, social aspect interpretation and cognitive digital data or signals interpretation were developed for the conceptual framework, using the generic procedure in 4.4.1, as presented in Figure 4.2.

1. Developing the cognitive interpretive capability to capture digital data or signal interpretation, presented in Figure 4.2 as interpretation of digital information. The description of the digital signal or data is

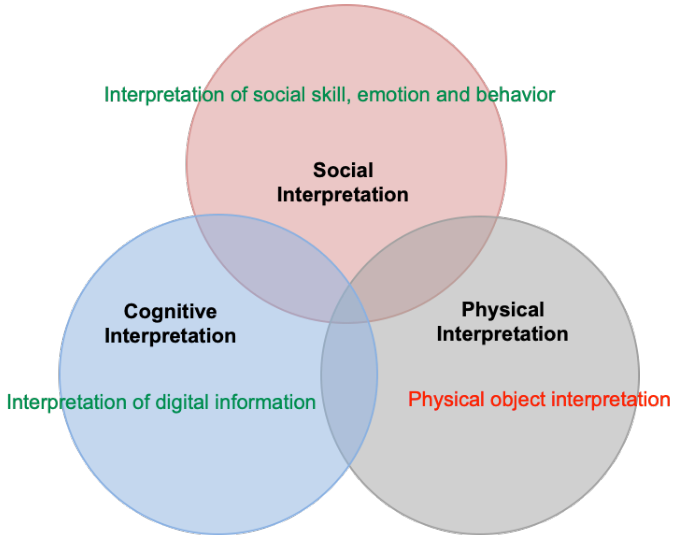

Figure 4.2: Robot interpretation capabilities adapted from MAR are presented in red text, where those developed for the framework are presented in green text.

presented in seven levels, using the language of MAR's object interpretation capability (EU-Robotics, 2016):

- Level 0 - No digital data or signal interpretation available.
- Level 1 - Fixed digital data or signal interpretation: the robot has a fixed interpretation of the perceived digital data or signal because they are pre-categorized.
- Level 2 - Basic digital data or signal interpretation: the system performs interpretation of digital data or signals into fixed categorized notions. Digital data or signals are categorized according to features and properties predefined as types of notations.
- Level 3 - Data delineation: the system is able to interpret and disambiguate some perceived digital data or signals in order to understand the information type. The disambiguation is based on built-in predefined notations.
- Level 4 - Digital data or signal category interpretation: the system is able to interpret digital data or signals based on predefined types that are relevant. This is performed by any of the lowest levels of the data-mining techniques.
- Level 5 - Digital data or signal structural interpretation: the system is able to interpret digital data or signal perception and to extract structural concepts for it. It is able to identify structural relationships and create a base knowledge of it.
- Level 6 - Digital data or signal basic semantic interpretation: the system is able to apply semantic tags to digital data or signals allowing it to plan actions based on strategies and objectives of the information type and its semantics.

2. Developing social interpretive capability to capture the interpretation of social skills or behavior, presented in Figure 4.2 as interpretation of social skill, emotion and behavior. The descriptions of these social aspects are presented in six levels, developed with the language of object interpretive capability

in MAR. However, levels 1 and level 2 did not match the concept of object interpretation and were therefore not included (EU-Robotics, 2016).

- Level 0 - No social interpretive ability: the robot does not need to interpret any social aspects of the identified agent.
- Level 1 - Fixed social interpretation: the robot uses sense data to perform fixed interpretations of social aspects of the recognized social agent since the social aspects are pre-categorized.
- Level 2 - Social delineation: the system is able to disambiguate specific aspects in the sense data, in order to understand the social aspect presented by the social agent. The disambiguation is based on built-in notation of social knowledge.
- Level 3 - Social category interpretation: the system is able to interpret social actions based on categories that are related to the event.
- Level 4 - Social structural interpretation: the system is able to interpret perceived social actions and extract the social concepts. It is able to identify social structures between people and their relationships.
- Level 5 - Basic semantic interpretation: the system is able to apply semantic social tags to people, allowing it to plan actions based on their personal objectives.

The perception and interpretation capabilities are allocated in the same section (technical capabilities) of the conceptual framework. Usually, perception is performed first and then interpretation follows. Together perception and interpretation define the possible actions the robot can carry out. Therefore, allocating any of the robot capabilities in the framework requires the capabilities to be listed, organized depending on the most common practices used in the robotic domain, presenting the capabilities in phases as a part of the main cycle, notated as "The capabilities cycle". The capabilities cycle has several phases containing all the possible actions that the robot can perform. It does not represent the sequence of robot actions, rather it lists the capabilities, both to simplify visualizing them and to enable the allocation of new capabilities within the capabilities cycle.

Capabilities cycle where a robot has several capabilities, it executes them in sequential order. For instance, most commonly, perception is performed to gather data about the robot's surroundings, which are then interpreted and translated. By executing perception and interpretation, the robot defines the next actions to execute. All its executable actions are listed as task abilities, therefore, distributing these capabilities in the ToRCH framework requires they be in an understandable order. The perception capability is one of the first actions that most robots perform and is therefore located in the first phase of the capabilities. As soon as perception data are obtained the robot performs interpretation on the data, hence the interpretation capability is allocated in the second phase of the capabilities. According to the interpretation of the data, the robot executes one of its possible programmed capabilities as a response, thus the task abilities are placed in the third phase of the capabilities. The task abilities section contains all the possible actions for which the robot is designed and programmed, so all the performed actions should be in the list of task abilities.

After performing the action, the robot can describe whether that action has a designated purpose or not. Therefore, the purposefulness ability is allocated after the task ability section. Similarly, the envisioning capabilities depend on the ability of the robot to act purposefully, which requires evaluating the purpose of actions which in turn can help the envisioning possibilities for future actions. Therefore, envisioning was allocated after acting purposefully in the capabilities list. If performing any action requires the robot to look for a response from its surroundings, this will redirect the robot to perform its perception capabilities again. Thus, in ToRCH the capabilities are in a cycle, which is located in the independent capabilities sub-section of the capabilities performance section in the conceptual framework, as described in Figure 4.3.

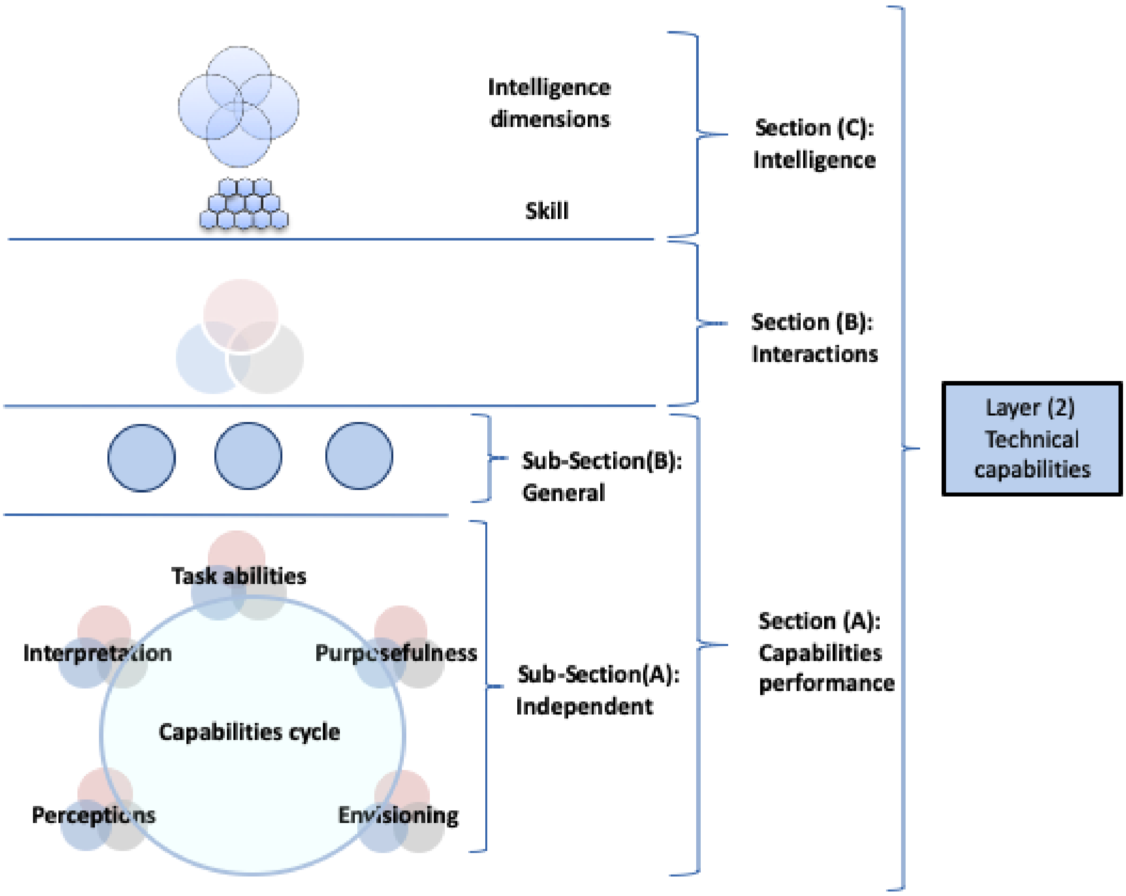

Figure 4.3: Illustration of the robot capabilities cycle. As perception comes first then interpretation for the perceived data. Then the performance of the task and then defining the purpose of the action, if available and then finally, the envisioning capabilities.

Robot task capabilities

A robot can perform three types of task capabilities: physical, social and cognitive task capabilities, as shown in Figure 4.4. The physical and cognitive tasks were adopted from MAR, but the social task capabilities were developed using the generic procedure in section 4.4.1 above to capture social tasks performed by the robot. The following list of social task capabilities was developed to be part of the social capabilities:

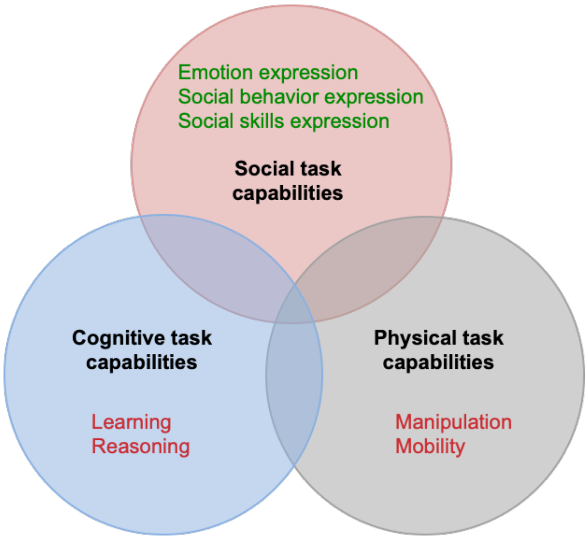

Figure 4.4: The adopted task capabilities are presented in red text where the developed task capabilities are presented in green text.

1. Developing emotion expression capability to capture the robot expressed emotion, presented in Figure 4.4 as emotions expression. They are presented in six levels, developed using the generic procedure in section 4.4.1 above and the language of reasoning capability and action ability levels from MAR (EU-Robotics, 2016):
   - Level 0 - No emotion expression ability: the robot does not need to express emotions in performing its tasks.
   - Level 1 - Expressing a predefined emotion: the robot is able to execute a predefined emotion. The emotion is programmed to be executed to present the robot status.
   - Level 2 - Expressing a set of predefined emotions: the robot is able to execute predefined emotions. Emotions are programmed to be executed to present the robot status.
   - Level 3 - Decision-based expression of emotion: the robot is able to alter its own emotions according to the social perception of another agent. It presents different emotions as a social feedback action.
   - Level 4 - Parameterized expression of emotion: expressing a generic emotion by matching the emotion expression with known parameters. The emotion models created from sense data can

be matched to a number of known emotion parameters. The settings for the parameters can be deduced from the sensed model.
  - Level 5 - Sense-driven expression of emotion: the robot is able to modulate its own emotions in proportion to perception-derived parameters. Emotion levels are expressed in different ways.

2. Developing the degree of social behavior expressions of the robot performance, presented in Figure 4.4 as social behavior expression. It is presented in six levels as they are developed using the generic procedure in section 4.4.1 using the same language as in the emotion expression capability:
   - Level 0 - No social behavior ability: the robot/system does not need to express social behavior in performing its tasks.
   - Level 1 - Expressing a predefined social behavior: the robot/system is able to execute a predefined social behavior. The social behavior is programmed to be executed to present the robot/system status.
   - Level 2 - Expressing a set of predefined social behaviors: the robot/system is able to execute predefined social behaviors. Social behavior is programmed to be executed to present the robot status.
   - Level 3 - Decision-based expression of social behavior: the robot/system is able to alter its own social behavior according to the social perception of another agent. It presents different social behavior as a social feedback action.
   - Level 4 - Parameterized expression of social behavior: expressing social behavior by matching the social behavior expression with known parameters. The social behavior models created from sense data can be matched to a number of known social behavior parameters. The settings for the parameters can be deduced from the sensed model.
   - Level 5 - Sense-driven expression of social behavior: the robot/system is able to modulate its own social behavior in proportion to perception-derived parameters. Social behavior levels are expressed in different ways.

3. Developing the degree of expressing human social skills (for example, through speech, dialogue and body language), presented in Figure 4.4 as social skill expression. That is presented in six levels and uses the same language as the emotion expression:
   - Level 0 - No social skills ability: the robot does not need to express social skills in performing its tasks.
   - Level 1 - Expressing a predefined human social skill: the robot is able to execute a predefined human social skill. The human social skill is programmed to be executed to present the status of the robot.
   - Level 2 - Expressing a set of predefined human social skills: the robot is able to execute predefined human social skills. Human social skills are programmed to be executed to present the robot status.
   - Level 3 - Decision-based expression of human social skills: the robot is able to alter its own human social skills according to the human social perception of another agent. It presents different human social skills as a social feedback action.
   - Level 4 - Parameterized expression of human social skills: expressing human social skills by matching the social skill expression with known parameters. The social skills models created from sense data can be matched to a number of known social skills parameters. The settings for the parameters can be deduced from the sensed model.

- Level 5 - Sense-driven expression of human social skills: the robot is able to modulate its own social skills in proportion to perception-derived parameters. Human social skill levels are expressed in different ways.

The task capabilities depend on perception and interpretation capabilities, so they are allocated to the second phase of the robot capabilities cycle and are presented in the independent capabilities sub-section of the conceptual framework, as described in Figure 4.3.

Purposefulness capabilities

The action capabilities are the ability of the robot to perform an action purposely; that is, the robot knows the reason for performing it. While the actions are covered by robot action abilities, the reason or purpose for robot actions is presented as a separate ability. Performance of purposeful actions are categorized into three types: physical purposeful actions (i.e., the ability of the robot to act purposely towards objects); social purposeful actions(i.e., the ability of the robot to act purposely towards social agents) and cognitive purposeful actions(i.e., the ability of the robot to act purposely towards information, signals or digital data).

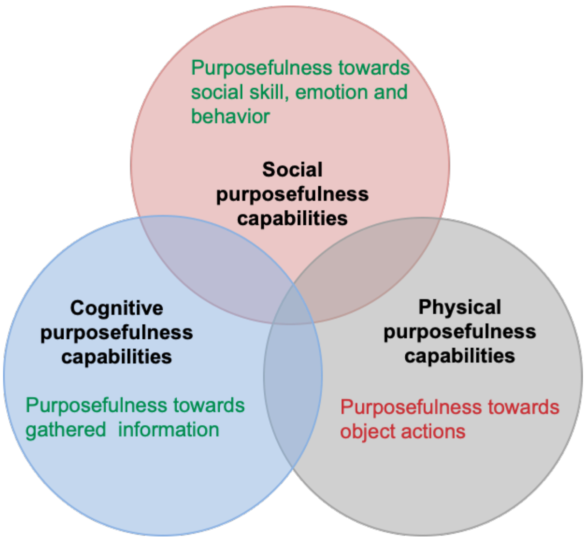

Figure 4.5: The purposefulness capabilities are presented in red text where the developed social and cognitive purposefulness capabilities are presented in green text.

1. Developing social action capabilities- that is to define the purposefulness of actions performed towards people, shown in Figure 4.5 as purposefulness towards social skills, emotions and behavior. They are presented in ten levels. The levels were developed with the same language as the purpose of action towards objects (EU-Robotics, 2016).

- Level 0 - No social action ability: robots are defined by having some level of action towards people.
- Level 1 - Defined social action: the robot executes fully predefined social actions as a sequence of social sub-actions. This social sequence actions can repeat until stopped by an operator or other system events.
- Level 2 - Decision-based social action: the robot alters its course of social action based on perceptions or social events. It is able to select between a set of predefined social actions based on its decisional autonomy ability.
- Level 3 - Sense-driven social action: the robot is able to modulate its level of social action in proportion to social parameters derived from its perceptions. The perceptions are used to drive the selection of predefined levels of social action.
- Level 4 - Optimized social action: the robot is able to alter the social sub-task sequence it applies to the execution of a social action in response to perceptions or a need to optimize a defined social action parameter.
- Level 5 - Knowledge-driven social action: the robot is able to use knowledge from social perception to determine the social action or sequence of actions to be performed. Knowledge is gained either by accumulation over time or through the embedding of knowledge from external sources, including user input that associates properties with perceptions.
- Level 6 - Plan-driven social actions: the system is able to use accumulated information about social actions to inform its plans for action.
- Level 7 - Dynamic social planning: the system is able to monitor its social actions and alter its plans for social actions in response to its assessment of success.
- Level 8 - Task social action suggestions: the system is able to suggest social actions that contribute to the goals of the specific mission.
- Level 9 - Mission proposals: the system is able to propose social missions that align with high-level social objectives.

2. Developing cognitive action capabilities to capture the purposefulness of actions performed towards information digital data and signals, presented in Figure 4.5 as purposefulness towards gathered information. They are presented in three levels with same language as the purpose towards objects (EU-Robotics, 2016).
   - Level 0 - No informatics action ability: robots are defined by having some level of action towards information.
   - Level 1 - Defined action: the robot executes fully predefined informatics actions as a sequence of sub-actions. This sequence of actions towards information can repeat until stopped by an operator or other system events.
   - Level 2 - Decision-based action: the robot is able to alter its actions on information based on perceptions or events.

Envisioning capabilities

Envisioning capabilities are the ability of the robot to picture future events logically. They are divided into three categories: physical action envisioning to predict subsequent physical actions, social action envisioning to predict subsequent social actions, and cognitive action envisioning to predict subsequent informatics availability and concomitant tasks.

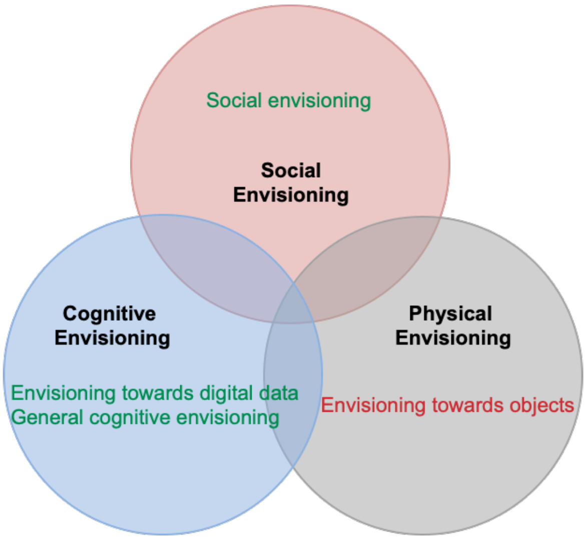

Figure 4.6: The adopted envisioning capabilities are presented in red text and the developed social and cognitive envisioning capabilities are presented in green text.

1. Developing social envisioning capabilities are presented in six levels, shown in Figure 4.6 as social envisioning. The levels were developed with the same style that was presented in the first six levels of the object envisioning capabilities adopted from MAR (EU-Robotics, 2016).
   - Level 0 - No social envisioning ability: the robot is not able to predict subsequent states of social actions.
   - Level 1 - Social prediction: the robot is able to project the effect of its social action to predict short-term local interactions with detection of social members in the environment. The robot only has the ability to predict its social actions with respect to other available social agents.
   - Level 2 - Dynamic social prediction: predicts its social action with respect to static objects available in the environment and also with respect to the existence of social agents.
   - Level 3 - Rigid social interaction prediction: the system is able to envision the effect of its planned social actions on identified social agents. For example, it is able to predict how an agent will behave when socially acting in a particular way.
   - Level 4 - Basic environment social envisioning: the system is able to observe social events in the environment that relate to its performance and envision the impact of the social event on the actions of itself and other agents.
   - Level 5 - Envisioning social user responses: the system is able to envision the actions of a social action by an agent responding to events. In other words: it has the ability to predict other agents' social actions on specific occasions.
2. Digital data or signal envisioning capabilities are presented in the cognitive envisioning capabilities section of the conceptual framework, shown in Figure 4.6 as envisioning towards digital data. Developing

this type of envisioning from gathered information, digital signals or data is presented in five levels using the same language as the object envisioning capabilities adopted from MAR (EU-Robotics, 2016).

- Level 0 - No envisioning ability: the robot has no prediction of subsequent defined information.
- Level 1 - Information prediction: the robot is able to predict consequences of defined information in short term.
- Level 2 - Long-term information prediction: the robot is able to predict consequences of its defined information over the long term and to present how would it impact the users.
- Level 3 - Information projection: the robot can predict the effect of information on specific events.
- Level 4 - Rigid information prediction: the robot is able to predict the effect of defined information on identified agents.

3. Developing general cognitive envisioning capabilities in six levels, shown in Figure 4.6 as general cognitive envisioning. It was developed using the same language as the object envisioning capabilities but adopted according to general cognitive aspects from MAR (EU-Robotics, 2016).
   - Level 0 - No cognitive envisioning ability: the robot has no prediction of subsequent cognitive actions such as learning and reasoning.
   - Level 1 - Cognitive prediction: the robot is able to predict the consequences of its cognitive (learning and reasoning) actions in the short term.
   - Level 2 - Long-term cognitive prediction: the robot is able to predict the consequences of its cognitive (learning and reasoning) actions in the long term and to present how would it impact the users.
   - Level 3 - Cognitive function projection: the robot can predict the effect of its cognitive (learning and reasoning) activities on specific events.
   - Level 4 - Rigid cognitive interaction prediction: the robot is able to predict the effect of its planned cognitive actions on identified agents.
   - Level 5 - Flexible cognitive interaction: the robot can predict the effect of its planned reasoning and learning on agents.

The ability to act purposefully and the envisioning capabilities depend on each other, so they were allocated to the same section of the conceptual framework. Knowing the purpose of an action usually helps in envisioning future possibilities. Envisioning capabilities evaluate the purpose of current actions; therefore, the capabilities to act purposefully and the envisioning capabilities were presented together in the fourth and fifth phase of the robot capabilities cycle in the independent capabilities sub-section of the capabilities performance section in the conceptual framework, as described in 4.3.

General and independent capabilities

The lists of capabilities and sub-capabilities defined above are performed without the need of other capabilities; therefore, they are defined as ‘independent capabilities’. The independent capabilities are specialized and may be executed sequentially one after the other depending on the robot’s design and performance, but they are nonetheless independent of each other. On the other hand, there are other criteria for capabilities that are reliant on one or more of the independent capabilities to be executed. These capabilities have been adopted from MAR (EU-Robotics, 2016) and they are defined as general capabilities and listed as task dependability, decisional autonomy and task adaptability.

The general capabilities are responsible for and manage the independent capabilities. They require the existence of the independent capabilities, so they have been positioned above the independent capabilities, creating a hierarchy of two sub-sections: the independent sub-sections (A) and the general capabilities sub-sections (B), as presented in Figure 4.3. The independent and general capabilities together make up the capabilities performance section (A). Any newly developed capability with overall executive functions and responsibilities for managing independent capabilities should be allocated to this general capability section. However, one novel general capability has been developed as follows:

Developing the general actions performed by the robot presented through three levels, using the generic procedure in section 4.4.1. Levels 1 and 2 use one pre-defined variable, therefore this covers the first and second levels of the generic set:

- Level 0 - No general action ability: the robot does not perform a general action in their tasks.
- Level 1 – Expressing a pre-defined general action: the robot is able to perform a pre-defined general action.
- Level 2 – Expressing a set of pre-defined general actions: the robot is able to execute pre-defined general actions.

Robot interaction capabilities

A robot can interact with objects and its surroundings physically, with signals, digital data, devices and systems, cognitively, and with social agents socially (EU-Robotics, 2016; Walker, 2017). Interaction depends on the robot's performance; hence the interaction section is positioned above the capabilities performance section (A), and it is defined as the interaction section (B) in the conceptual framework, as presented in Figure 4.3.

1. Developing physical motion interaction capabilities, shown in Figure 4.7 as physical motion interaction, created with five levels. These levels were developed using the generic procedure presented in section 4.4.1 above and using the same language as the interaction capabilities.
   - Level 0 - No physical interaction: many robot systems will be able to operate successfully without physical interaction, such as simulated robots like Amazon's Alexa.
   - Level 1 - Self-utilization: interaction emerges from special settings of electronic circuits.
   - Level 2 - Static self-peripheral utilization with a single appendage: the system is able to use peripherals to perform physical interaction within its environment. This relates to its manipulation capabilities, but the robot remains static.
   - Level 3 - Static self-peripheral utilization interaction within its environment. This relates to its manipulation capabilities, but the robot remains static.

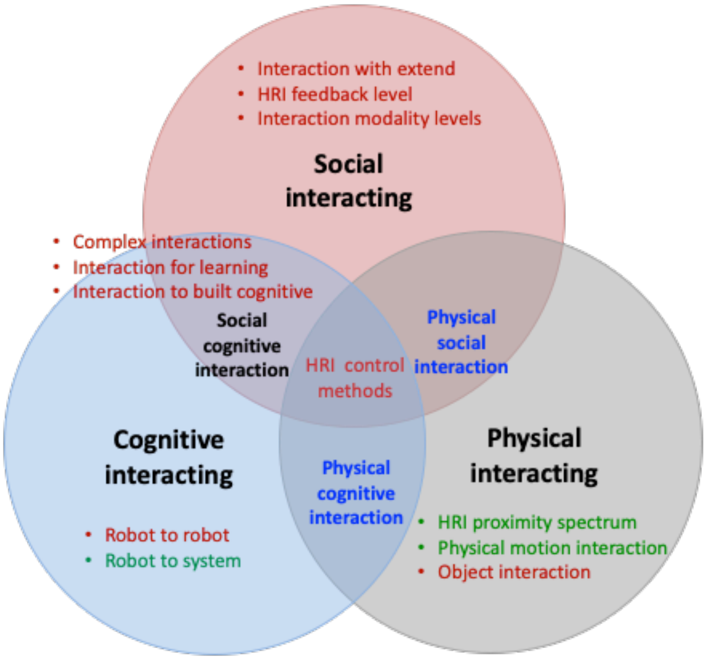

Figure 4.7: The adopted MAR interaction capabilities are presented in red text, and the developed social and cognitive interaction capabilities are presented in green text. Some other intersection areas that have no capabilities are presented in blue text.

- Level 4 - Dynamic self-peripheral utilization: the system is able to use peripherals to perform physical interaction within its environment. This relates to its manipulation capabilities, but the robot has dynamic movement.

2. Developing human-robot interaction proximity levels, adopted from the Reeler project (Hasse and Bulgheroni, 2017) in seven levels and shown in Figure 4.7 as HRI proximity spectrum.
   - Level 0 - None: no physical contact with a human.
   - Level 1 - Minimal: independent, unlikely to have contact with a human.
   - Level 2 - Parallel: works independently alongside a human.
   - Level 3 - Operative: operated or commanded by a human.
   - Level 4 - Cooperative: cooperates physically with a human.
   - Level 5 - Social: interacts socially with a human.
   - Level 6 - Integrated: worn or implanted in a human.
3. Robot to system/device interaction is part of the cognitive interaction capability, shown in Figure 4.7 as interaction from robot to system. It captures digital data or signal interaction between the robot and any other object/system/device. It was developed in six levels. The levels were developed using the generic procedure presented in section 4.4.1, using the same language as the MARS robot-to-robot interaction (EU-Robotics, 2016) and the conceptual ideas of the Internet of Things (IoT).

- Level 0 - No robot-system interaction: the robot is unable to communicate with other external systems.
- Level 1 - External system knowledge: the robot is aware of external systems but does not communicate with them.
- Level 2 - Communication of status: the robot is able to communicate its own status and task status to external systems.
- Level 3 - Reception of information: the robot is able to request information from external systems.
- Level 4 - Action on information: the robot is able to dynamically respond to information provided by external systems.
- Level 5 - Selection capability: the robot is able to differentiate between different types of external systems and choose which to interact with to accomplish a given task or mission.

4.4.2 Measurable parameters for the second layer

The capabilities listed in the second layer include levels to measure the capability. Each capability could be adjusted by several parameters. Therefore, the following lists present the main parameters that could affect the mentioned capabilities.

General capabilities parameters

While a robot might perform a capability with a high level of technological maturity, this is not the only parameter that requires measuring. For example, the length of time a robot requires to perform a capability could improve, indicating that the 'time parameter' is improving. This section presents parameters that could be linked to some of the capability sections within the conceptual framework. Some parameters are related to most capabilities, such as user expectation and time scales. These parameters do not cover the level of capabilities; rather, they measure the quality of performance of a specific capability level. Thus, for example, if two robots perform the same capability at the same level but one executes the capability with better measures, e.g. more quickly, then the capability of both robots are captured at the same level but with a different performance quality, measured by the specific parameter, 'the time parameter'. Therefore, the capability enhancement depends on the improvement of that specific parameter. MAR contains several parameters for some of the main capabilities. These parameters can be adopted by the conceptual framework, as presented in Figure 4.8. Some examples are task density (the number of tasks a system encounters at the same time), task complexity, time scale, communication band width and communication latency. This exploratory research pinpoints the need to define these measures for each of the capabilities.

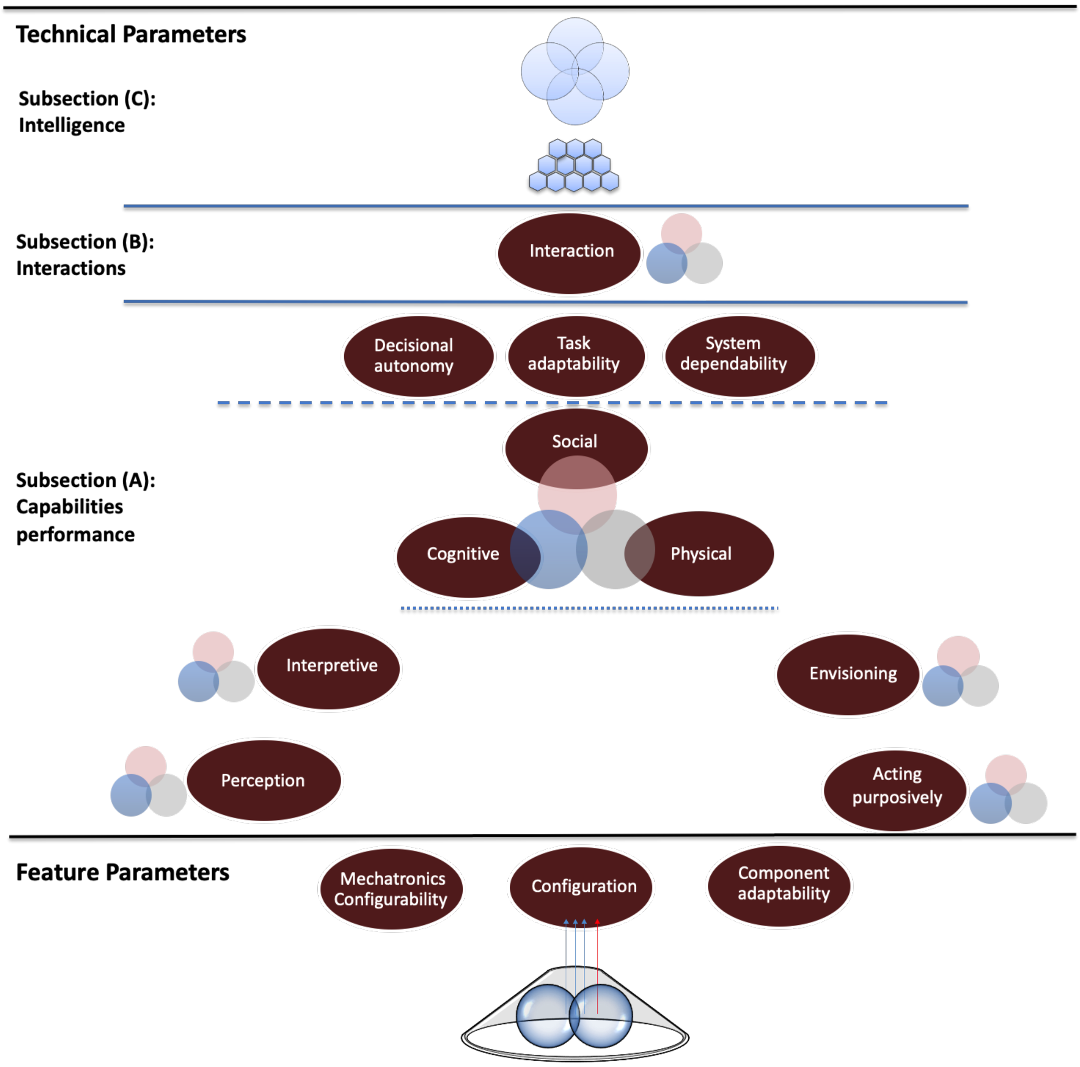

Figure 4.8: Allocating capabilities parameters and measures into ToRCH.

Interaction parameters

Some interaction classifications that are presented as an interaction parameters for robots are:

- Robot personality: whether the robot is a tool, pet or creature, cartoon, artificial being, or human-like, which affects the robot interaction (Fong et al., 2003).
- Interaction paradigm, as developed Breazeal: to classifying robots as a tool, avatar, cyborg extension or sociable partner (Breazeal, 2004; Dragone et al., 2005).
- Interaction role, as developed by Dautenhahn: to classifying robots as a supervisor, operator, team-mate, machine or bystander (Cabibihan et al., 2013).

- Time-space taxonomy, developed by Tzafestas and known as CSCW taxonomy (synchronous, asynchronous, collocated, non-collocated) (Tzafestas, 2015).
- Human-robot ratio (H:R) developed to indicate the level of shared interaction among teams; H:R, H:RR, HH:R, (HH):(RR),(HH):RR, HH:(RR) (Yanco and Drury, 2002).
- Robot interaction with the environment, developed by Moore, to indicate the type of interaction performed with the environment: coupling, coordination or communication (Moore, 2016).
- Multi-robot interaction to indicating the composition of robot teams, it lists dimensions to specifying the multi-robot systems and their taxonomy (Yanco and Drury, 2002).
- Interaction media: whether the interaction is performed via visual, audio or physical means or a combination of media. These interactions depend on the input and output sensors, as mentioned previously in the HW and SW layer.
- Social degree of contact: direct contact (interacting face to face with the robot); mediated contact(seeing the robot on videos, media, etc); extended contact (knowing someone who has met the robot); imagined contact (imagining yourself interacting with the robot).

Some interaction parameters that are presented for social robots are:

- Defining a social robot with identity, character, stereotypes, and roles (Duffy et al., 2005).
- Social models of interactions (Breazeal, 2003; Dautenhahn, 2007).
- Social interaction sub-type: any social interaction needs to be sub-categorized: group interaction, interpersonal, inter-species or intra-space interaction.
- Social interaction communication: verbal and non-verbal interaction.

Intelligence dimensions and parameters

A robot's intelligence is demonstrated by its performance (Saksida et al., 1997; Schaefer et al., 2012) and this is presented within the intelligence section (C) of ToRCH, as shown in Figure 4.3. Artificial intelligence (AI) in robotics is opaque and confusing, especially because intelligence is not well defined and has no clear measurements. In this research, the (AI) of the robot performance is described by four main dimensions: physical/ morphological, social, cognitive and collective intelligence (Pal et al., 2004). Each of these dimensions is supported by several skills. Any robot can present its intelligence according to a combination of these dimensions by illustrating some skills and behavior.

Each skill is expressed with additional parameters to emphasize the skill the robot would like to express (Besio et al., 2010). These parameters are shown in the set of interaction cycles, which are caused by the outcome of the robot performance. The interaction cycles are developed to illustrate a specific demonstration of a skill that would support the robot with particular intelligence. The more interaction cycles developed within the robot performance to present a skill, the more intelligence is exhibited in the robot (Winfield, 2012). Moreover, the more skills available in the robot to express an intelligence, the greater that intelligence is expressed and defined as an external ability. Therefore, the skills and intelligence dimensions are captured within the robot to list the goals of its behavior, interaction and capabilities; and to define which skill was developed with the robot to illustrate this goal. Any parameter that could affect the interactions, behavior, skills and intelligence of a robot should be captured in this layer where this exploratory research identifies the need to define these parameters.

4.5 The third layer: operational capabilities

Operational capabilities are presented within the third layer of the conceptual framework in line with suggestions (Breazeal, 2017a; Dautenhahn, 2017). This layer is not covered in this thesis.

- Cost (Yan et al., 2015)
- Safety and security, e.g. safety measures types and levels of IOS safety capabilities (De Santis et al., 2008)
- Operational interactions parameters
- Operational and interaction environment
- Practicality and usability, e.g. user evaluation of the system.
- Testing and training aspects, e.g. human training to operate the robot, robot intuitiveness level in supporting training

The next chapter describes and presents ToRCH as one complete structure, as well as its qualities, guidelines and usage.

❋ Chapter 4: Summary
The conceptual framework is required to be developed to distinguish robots by their capabilities across various domains. This chapter presented the enhancement of the framework by adding a comprehensive set of capabilities with different maturity levels. It also allocates the novel capabilities in the framework in a way that has logical relationships. This framework was designed to characterize robots and robot capabilities, so it has been entitled “Towards Robot Characterization Hierarchy” or ToRCH. It presents the major capabilities within the robot, including its interactions, behaviors and goals of intelligence. Thus, the framework adds these dimensions and measures to the robotics field. This implicitly adds to the robotics literature some comprehensive lists of capabilities presented in a logical hierarchy. This list could be used to identify robots and classify them. This, in turn, will help in describing robots and deploying them.

Chapter 5

ToRCH

5.1 Introduction

This chapter presents the Towards Robot Characterization Hierarchy (ToRCH) framework from different perspectives. It will describe its conceptual dimensions, capabilities, the relation between the capabilities and how they are ranked. The chapter also describes the quality criteria of ToRCH. Then it will cover the guidelines for its use and demonstrate its ability to add undefined capabilities. Finally, the chapter will show how ToRCH can be used.

5.2 What is ToRCH?

Although several classifications have been developed, they are not comprehensive and do not include the ever-growing range of possible robot characteristics. This continuous expansion in robot technology requires a framework for a classification system that encompasses what a robot can and cannot do. It should also cover future developments, regardless of the domain in which the robot operates. Therefore, such a classification system should be general enough and have sufficiently broad dimensions to capture all possible robot capabilities. This classification should be a useful instrument that can identify any robot. It should also be an essential tool for making comparisons between robots across different domains.

The ToRCH classification system is suitable for identifying robots by clarifying their features, capabilities and goals. ToRCH is the first framework that can capture robot capabilities and specify them in a quantitative presentation. ToRCH allocates robot capabilities in a hierarchical framework in order to illustrate the logical relations between the capabilities and also to identify the difference between similar capabilities. The sequence of the capabilities in the hierarchy is intended to simplify the capabilities scoring process rather than for presenting the order of the performance and execution of capabilities. Therefore, the framework functions as a tool to represent in a numerical format the performance of robots used in any field. ToRCH thus fulfills a need in robotics by clearly defining the capabilities of any robot and enabling roboticists to optimize a robot's capabilities against application requirements. ToRCH allows for a straightforward procedure to determine robot capabilities across various domains by creating a Robot Capabilities Profile (RCP). The RCP allows application developers to select the most appropriate available capability or to suggest new capabilities. ToRCH is currently the only existing accessible tool for roboticists to illustrate their robots' capabilities and to present, either visually and quantitatively, the development of the robot capability within the research progress. ToRCH also supports the demonstration of upcoming capabilities in the future design of a robot.

5.3 TORCH description

Applying the framework for spoken language applications and their related technology (Moore, 2000) within ToRCH, as illustrated in section 3.3.2, ToRCH divides robot characteristics into three layers, shown in Figure 5.1.

The first layer: robot features Figure 5.1 presents layer (1). The robot's hardware (HW) and software (SW) are covered in this layer. The HW covers all sensors, actuators, kinetics, kinematics, embodiment, morphology, locomotion, etc., while the SW covers the operating systems (e.g., ROS or YARP), architectures and any programmable modules installed within the robot (e.g., visual or tactile processing) (EU-Robotics, 2016), as presented in Table 5.1, 5.2. This layer also covers networking and memory concepts such as Wi-Fi and iCloud. The HW/SW determine and set the limits of the robot's capabilities and their type of interactions (Breazeal, 2004; McGrenere and Ho, 2000). Capabilities presented in this layer capture the capabilities of the HW/SW within the robot. This layer is not covered in this research.

The second layer: technical capabilities Figure 5.1 presents layer (2). This layer illustrates the capacity of the robot to perform its task. All of the listed capabilities were either adopted from MAR (EU-Robotics, 2016) or developed for ToRCH. This layer is divided into

Table 5.1: Capturing some of the robot hardware and allocate them into the first layer.

Hardware	**Characteristics**
Sensors	external and internal sensors
Actuators	external actuators and internal actuators
Internal structure	• kinematics: – geometry of the mechanical structure, such as Cartesian, articulated, cylindrical, parallel, spherical/polar, swing arm, etc. – variables of the manipulator's joints and links • kinetics: – force that acts on the kinematic skeleton – joint motors and degree of force of that motor
External structure	• physical measures of the robot, including weight and dimensions, presented in width x, length y, height z • robot power source • color, shape, body frame, outer body texture and pattern
Embodiment	anthropomorphic, zoomorphic, caricature, functional, screen character, etc.
Locomotion	fixed place, bipedal, wheeled, quadruped, hexapod, octopod, climbing, etc.
Design	design approach and design structure, etc.
Electronics	computational platforms
Mechanics	mechanical system
Materials	internal and external materials

three sections: (performance capabilities (A), interaction (B), intelligence (C)) described as following:

The first part of the technical capabilities layer presents the performance capabilities, shown in Figure 5.1, layer (2), section (A). It categorizes the capabilities into two subsections: the independent capabilities subsection (A1) and the general capabilities subsection (A2). The independent capabilities are categorized into the following five sub-sub-sections, which are:

(1) perception and (2) interpretation capabilities categorize the robot's perception capabilities into social, cognitive, and physical perception. These categories also include the robot's modes of perception, known as the interaction modes, the data collection method, (i.e., visual,

Table 5.2: Capturing some of the robot software and allocate them into the first layer.

Software	**Characteristics**
Operating system	presented O.S.(e.g., ROS, or YARP)
Memory	memory size, sensory memory (instinct data/ imitating data)
Networking and communication	• network connections such as Wi-Fi or cables, I-cloud and connectors • network purposes (e.g., Internet of things(IOT) and Internet of skills (IOS)
System engineering and architecture	programmed modules, managing complex system, system life cycle, systems architecture and design
System design and development	system design, system theories, system integration and system of systems

auditory, and physical [mechanical, magnetic, chemical, signal, haptic]). This captured data is translated through any of the interpreted capability categories. Perception and interpretation capabilities are presented in Figure 5.1, layer (2), section (A), subsection (A1), perception capabilities (1) and interpretive capabilities (2). (3) Task abilities categorize the tasks the robot can perform into physical (e.g., mobility and manipulation), social (e.g., emotions, social structure, relationships and behaviors, and cognitive tasks (learning, reasoning, skill gathering and methods in problem-solving (Martınez-Plumed et al., 2018))subsection). A list of action purposes task abilities are presented in Figure 5.1, layer (2), section (A), subsection (A1), task abilities (3). (4) Actions and (5) envisioning capabilities categorize the actions performed by the robot as physical, cognitive or social. The robot's action is considered a listed task ability, as mentioned above under task abilities (3). Envisioning is also categorized as physical, cognitive, or social. A list of purpose of actions and envisioning abilities are presented in Figure 5.1, layer (2), section (A), subsection (A1), purpose of actions capabilities (4) and envisioning capabilities (5).

The technical layer also includes the robot's general abilities, as presented in Figure 5.1, performance capabilities in section (A), general capabilities in sub-section (A2). These capabilities are not part of any of the categorized capabilities mentioned above but they are considered to be a separate type of capabilities that perform some control on other capabilities. These capabilities control, change, modify and determine the sequence of the capabilities mentioned earlier in the independent capabilities subsection (A1). Some of these general capabilities are task complexity, adaptability, dependability and decisional autonomy. Since

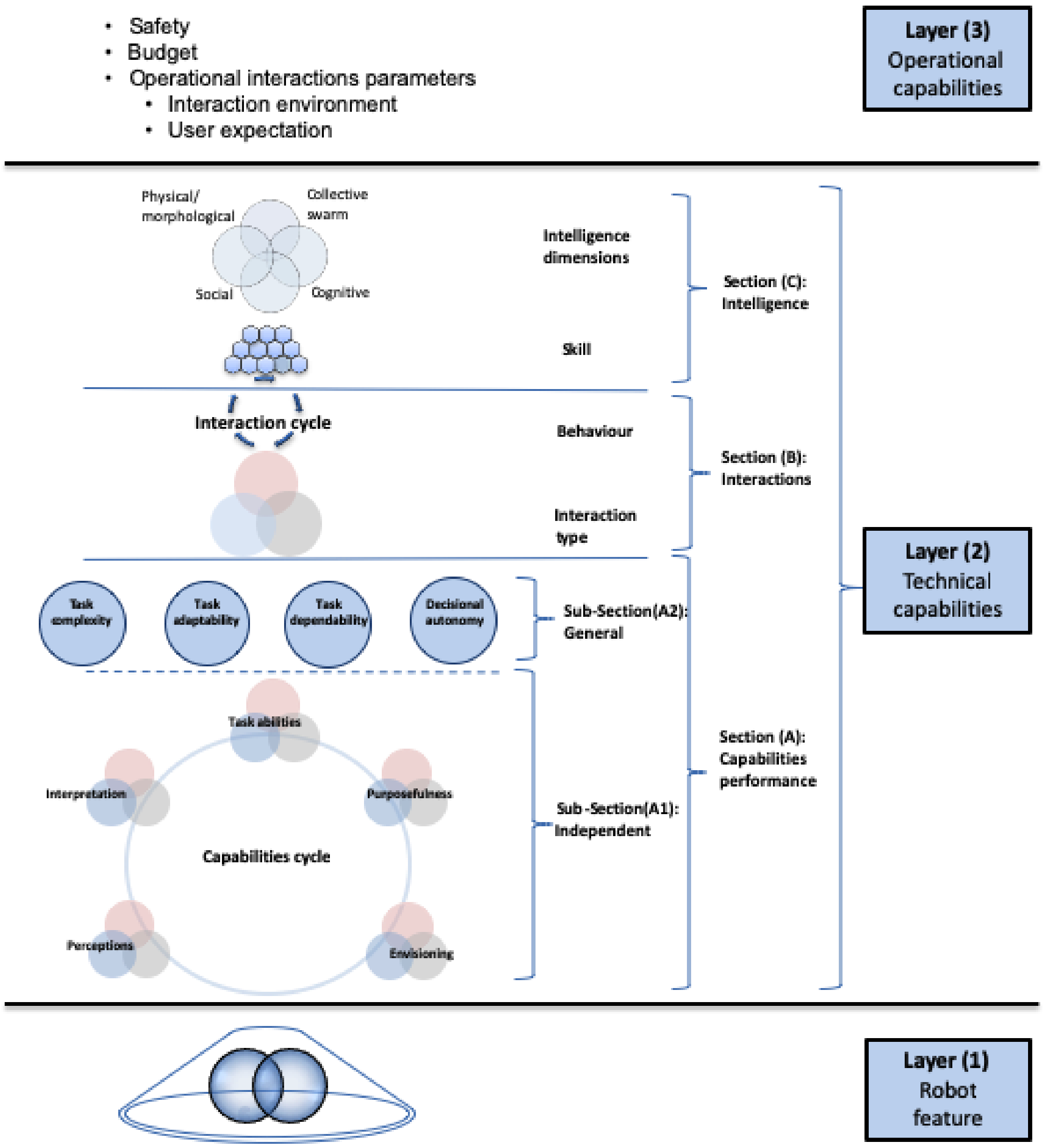

Figure 5.1: The diagram illustrates a conceptual framework for characterizing robots. It contains three main layers. The first layer is the robot feature (HW and SW). The second layer contains all the robot technical capabilities, categorized into three main sections: (A) performance capabilities, (B) interaction capabilities and (C) intelligence capabilities. Each of these sections is classified further according to their content.

they are in control of other capabilities they are presented in the hierarchy at a level higher than the others.

The second part of the technical capabilities layer covers the interaction capabilities of the robot, as presented in Figure 5.1, layer (2), section (B). This layer captures the type of interaction performed by the robot either as social (toward a social agent), cognitive (through digital data towards other devices or robots), or physical (towards objects in the environment) (Yanco and Drury, 2002). Robot interaction is performed according to the capabilities and implementation of the interaction cycle, which is also known as 'the robot-world feedback loop' (Winfield, 2012). The interaction capabilities and the interaction cycle are categorized as cognitive, social or physical.

The third part of the technical capabilities layer clarifies the 'intelligence' of the robot, presented in Figure 5.1, layer (2), section (C). This third section outlines the purpose of the robot's performance, including the external perception of its actions (Schaefer et al., 2012). This intelligence is obtained through skills that are, in turn, defined through specific behaviors or sets of actions (Kihlstrom and Cantor, 2000). Therefore, the robot acquires intelligence though the skills developed within it. In this research, these skills are adopted from Gardner's intelligences and presented as skills (Pal et al., 2004). These skills are not limited only to his categorization, instead, they could be added as any skill that could be obtained by the robot (Legg et al., 2007). This intelligence can also arise from behavior that mimics biological intelligence, such as swarms that behave in the same way as bees (Russell and Norvig, 2016). These intelligences are captured as part of the behavioral aspect of the robot. There are several possible types of intelligence (ManagementMania, 2016; Pal et al., 2004; Winfield, 2017): (1) physical-morphological intelligence, such as bodily-kinaesthetic skills or visual-spatial skills; (2) cognitive intelligence, such as logical-mathematical skills, or exchanging information; (3) social intelligence, such as musical or verbal-linguistic skills; (4) collective intelligence, which captures emerging interaction skills in either heterogeneous (non-identical) or homogeneous (identical) robots. It also includes collaborative or cooperative skills.

The third layer: operational capabilities Figure 5.1 presents layer (3). This layer defines the robot's operational profile in terms of cost, duration, safety, testing, training, acceptance, and usability. This also covers the operational environment capabilities of the robot; i.e., ground, aerial or underwater (Siciliano and Khatib, 2008). This research does not cover the operational capabilities layer.

5.4 ToRCH list of capabilities

While ToRCH is made up of three layers (robot features, technical capabilities and operational capabilities), this research concentrates on the technical capabilities layer, which is presented in Figure 5.2, layer (2). These technical capabilities are related to a robot's performance, interactions and goals. The capabilities are placed within the ToRCH hierarchy and ranked into layers, sections, subsections, categories, capabilities, sub-capabilities, sub-sub-capabilities and levels; as discussed further in this chapter.

5.5 ToRCH hierarchy

Within ToRCH, capabilities are connected within its various layers that create its hierarchy, as seen in Figure 5.3. Each capability is connected according to the context of that capability and its relations to other capabilities. For example, the list of relations, in layer (2), section (A), subsection (A1) are:

The first relation: all perception capabilities are performed to gather data for interpretation. Therefore, perception and interpretation are presented in one location, with perception as the first phase and interpretation as the second. The second relation: most robot actions, defined as task abilities, depend heavily on perception and interpreted data, so task action is allocated above and after perception and interpretation, as task action capabilities are dependent on these perception and interpretation capabilities. The third relation: if the robot is programmed to execute any of its task abilities, then the task abilities are categorized into cognitive, social or physical, where physical capabilities demonstrate physical performance towards the robot's environment, cognitive capabilities demonstrate cognitive performance towards gathered data, and social capabilities demonstrate social performance towards social agents. If any of these executed actions are performed without any machine learning or reasoning, then the robot has no explanation for its performed tasks. Hence, a simple set of executions do not support the robot with any predictions, knowledge of purposefulness or envisioning capabilities. However, if the robot executes AI concepts (such as machine learning and/or deep learning) as part of its cognitive capabilities (where it continually captures and interprets sensor data to perform algorithms, model insight and recognize patterns to make predictions), then the robot would be supported to demonstrate some purposefulness of action and envisioning capabilities. Therefore, purposefulness and envisioning capabilities are presented after performing the task abilities and after understanding them. The fourth relation: if the robot could define the purpose of its action then it would predict some of the future actions.

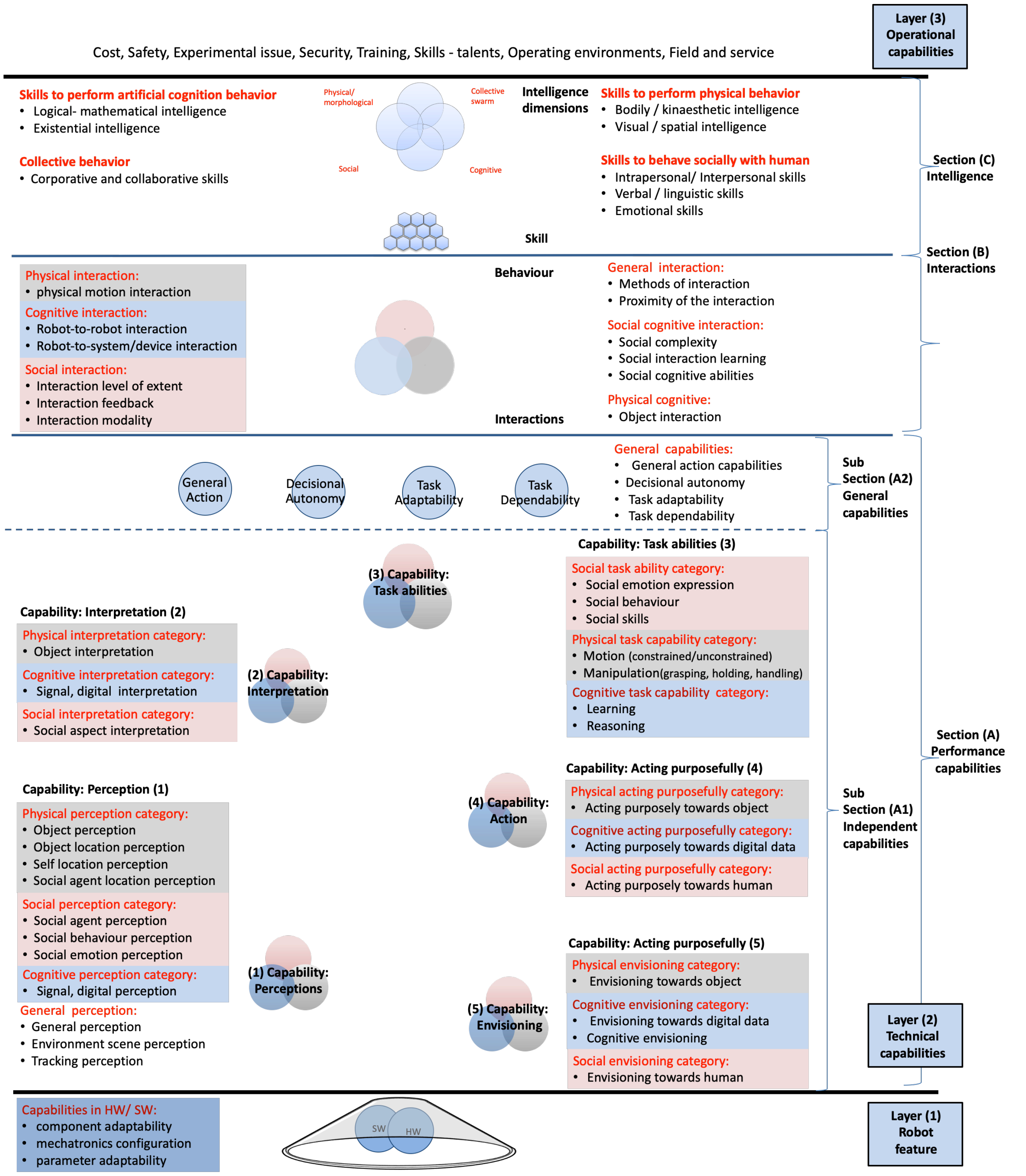

Figure 5.2: ToRCH framework showing the allocation of capabilities and sub-capabilities to their relevant layers (features layer (1), technical layer (2) and operational layer (3)), sections (performance capabilities (A), interactions (B), intelligence, skills (C)) and subsections (independent capabilities (A1) and general capabilities (A2)). Capabilities listed with a black triangle were developed by ToRCH, but the rest of the capabilities were adopted from MAR.

Therefore, envisioning capabilities are presented after purposefulness. These capabilities are presented in a capability cycle within the hierarchy. The robot does not need to follow the order of execution of these capabilities, they are presented in the hierarchy to capture and list as many capabilities as possible.

The ability to perform several actions requires the robot to decide which action is executed first. This type of controlling capability is part of a general capability and it is therefore allocated above independent capabilities, as shown in section (A), subsection (A2).

Performance of perception, interpretation, task capabilities, purposefulness and envisioning capabilities create the ‘interaction cycle’ between a robot and its surroundings; therefore, the interaction section is allocated above the general capabilities, as seen in section (B). Any interaction between the robot and its environment demonstrates some skills and behaviors within the robot, so skills are presented through interactions, as shown in section (C). For example, if the interaction is social, the robot demonstrates social skills or behavior.

Specific skills support intelligence within the robot, and where the robot demonstrates a higher level of similar skills, they demonstrate a higher level of that intelligence, be it physical, cognitive or social. Any skills or behavior would support specific intelligence within the robot. Wherever the robot demonstrates a higher level of similar skills, a higher level of that intelligence is demonstrated, either physical/morphological, swarm/collective, social, individual or combined.

These sections of the hierarchy are demonstrated as one section over the other. The list of capabilities in the lower section contributes in generating in its upper section. And each of the upper sections in the hierarchy is either made of, controls or manages the capabilities available in the lower sections. Therefore, these sections of the hierarchy are presented within funnels to present the accumulation of specific capabilities to generate the upper capabilities sections, as presented in Figure 5.3.

5.6 Classifying capabilities in ToRCH

As presented above, in the ToRCH framework, robot capabilities are classified into groups within the hierarchy depending on their capability types. The capabilities are allocated to specific tiers within the ToRCH hierarchy, creating the layers, sections, subsections, categories, capabilities, sub-capabilities, sub-sub-capabilities and levels, as presented in Figure 5.4. This classification of capabilities can be compared to the Linnaean classification for living creatures of kingdoms, classes, orders, genera, species, sub-species. Likewise,

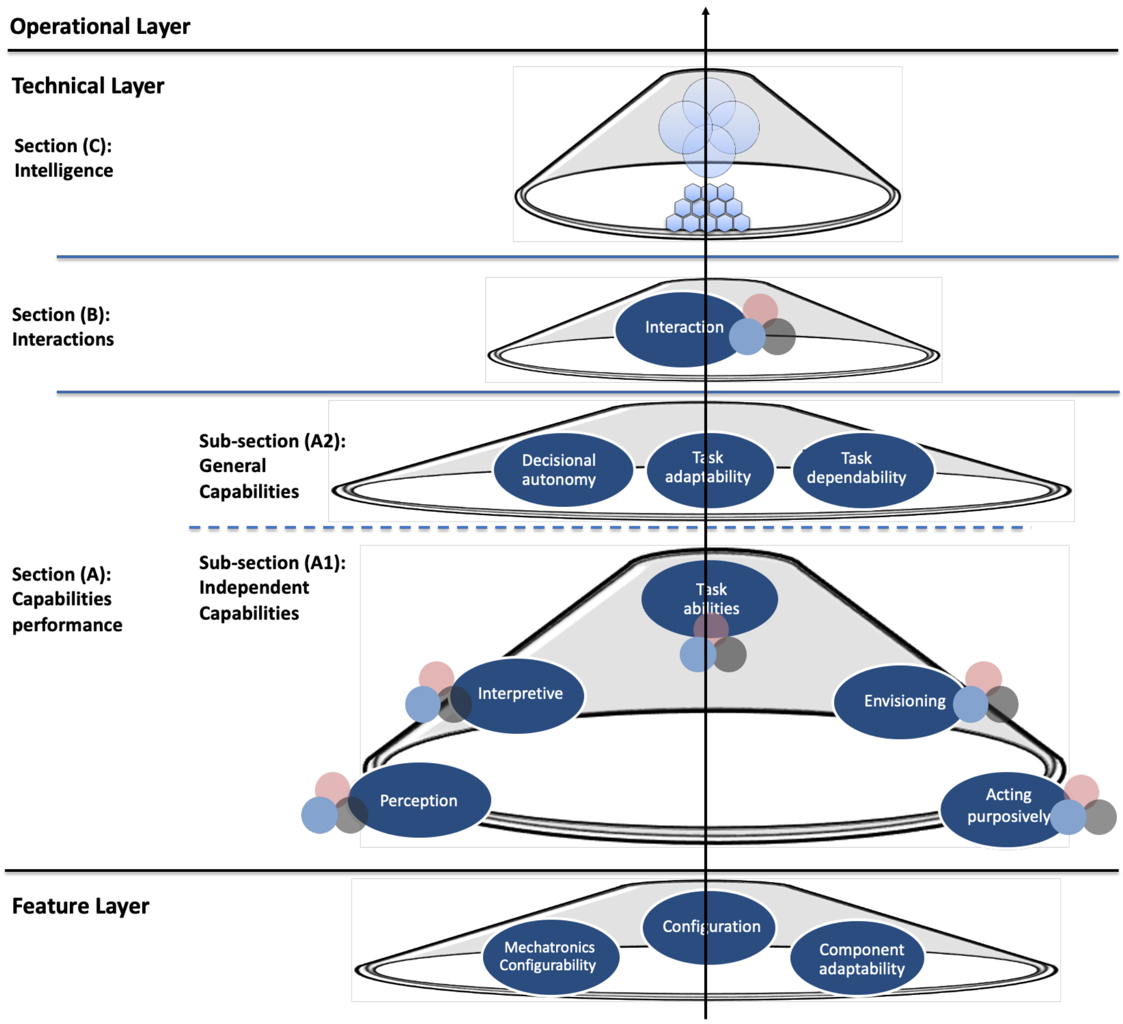

Figure 5.3: ToRCH capabilities and their relationship to each other are demonstrated. The capabilities are shown in a nested demonstration. Each section or subsection of the hierarchy consists of, controls and manages the layer beneath it. Each lower level contributes to developing the capabilities in the upper levels. This demonstration is only for capability presentations, and robot execution does not have to follow the arrangement of the structure.

the ToRCH system classifies capabilities from the most general down to detailed specific abilities. It provides for each rank a name and a tier number. The name is related to the capability concept and the tier number indicates the number of vertical links from the very first tier (i.e., the most general capabilities). There are eight tiers used to define the ToRCH ranking classification.

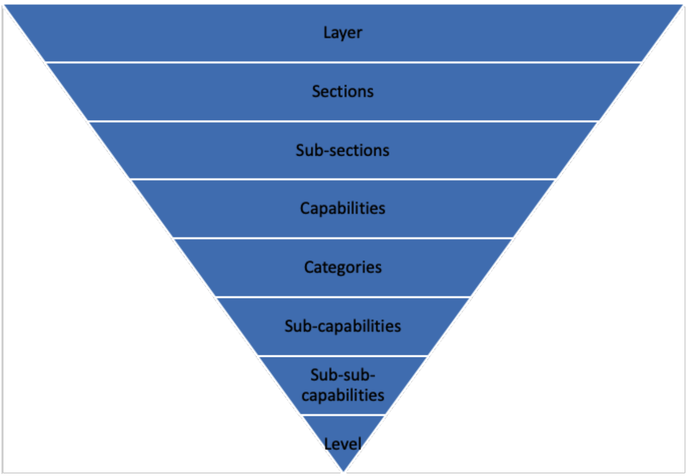

Figure 5.4: The tiers by which capabilities are classified and ranked in ToRCH, from the general (up) to more specific (down).

5.6.1 ToRCH capabilities ranked according to the hierarchy

Capabilities are allocated according to their context, with some capabilities assigned within others, which creates a ranking. Each rank contains a collection of similar capabilities and subsumes under it another more specific set, as presented in Figure 5.5. All general capabilities are located on the first four tiers of the ranking and the specific capabilities are allocated within the lower tiers, tiers 5, 6 and 7 (to capture the categories, sub-capabilities and sub-sub-capabilities).

This ranking creates groups of capabilities within tiers. Each group presents its capabilities with their implicit shared properties and allocates capabilities in the same location. For example, tier 4 includes perception, which divides into the categories physical, social and cognitive categories of tier 5. Selecting social perception from tier 5 leads to the social agent perception sub-capabilities on tier 6. In this case, social agent perception is a sub-capability of social perception and it does not contain any sub-sub-capability (tier 7), but there are levels for each of the social agent perception sub-capabilities. These social perception capabilities share the same social properties and are allocated to the same section of ToRCH. However, not all capabilities follow the full ranking process of eight tiers; some might skip a tier or more. For example, the interaction capability is ranked according to layers, sections, categories and levels, thus, it skips the subsections and capabilities tiers (3 and 4). The more advanced the technology of the capability and its development, the further down the ranking the capability will be, as it can be created and subdivided.

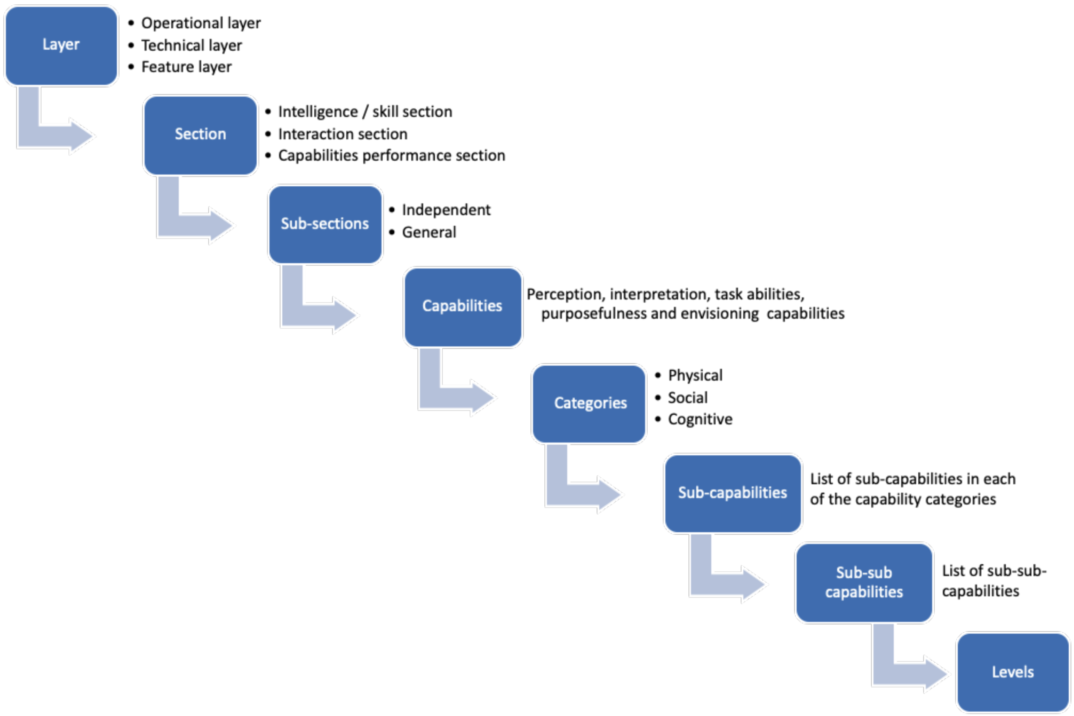

Figure 5.5: Hierarchy of the ToRCH schema

Another example of the ranking process is ‘unconstrained motion capability’, which is a sub-sub-capability (tier 7) and is measured through 8 levels of technological maturity (tier 8). Moving up through the tiers, unconstrained motion is a sub-sub-capability of ‘motion capability’ (tier 6), as a type of physical capability (tier 5) under ‘task ability’ (tier 4), which is an independent capability (tier 3) that falls under the technical capabilities section (tier 2) in the technical layer of ToRCH (tier 1).

5.6.2 Formal grammar used in ToRCH ranking

ToRCH can also be presented through formal grammar. The formal grammar presents a set of production rules to generate the structure of the framework in the form of a string. The formal grammar is applied for theoretical computer science and mathematical logic. The grammar does not describe any meaning for the string but lists some rules. Each rule has two sides, one in the left and the other in the right. The left-hand side is the non-terminal side and the right-hand side can consist of either an empty string, a single terminal or a single terminal symbol followed by a non-terminal symbol with nothing else. The rules begin with the Starts symbol as follows:

Start —–> Layer | Layer . Start

Layer —–> <layer-name> (Layer- Exp)

Layer-Exp —–> Section | Section . , . Layer-Exp

Section —–> <section-name> | (Section-Exp)

Section-Exp —–> SubSection | Capability | Section-Exp . , . Section-Exp

SubSection —–> <subsection-name> | (SubSection-Exp)

SubSection-Exp —–> Capability | Category | SubSection-Exp . , . SubSection-Exp

Capability —–> <capability-name> | (Capabilities-Exp)

Capability-Exp —–> Category | SubCapability | SubSubCapability | Level | Capabilities-Exp .,. Capabilities-Exp

Category —–> <category-name> | (Category-Exp)

Category-Exp —–> SubCapability | SubSubCapability | Level | Category-Exp .,. Category-Exp

SubCapability —–> <subcapability-name> | (SubCapability-Exp)

SubCapability-Exp —–> SubSubCapability | Level | SubCapability-Exp .,. SubCapability-Exp

SubSubCapability —–> <sub-subcapability-name> | (SubSubCapability-Exp)

SubSubCapability-Exp —–> Level | SubSubCapability-Exp .,. SubSubCapability-Exp

Level —–> <level-name> (<level-value>)

5.6.3 Tree diagram of ToRCH

The ToRCH capabilities hierarchy is illustrated in a tree diagram, as presented in Figure 5.6, to show the whole classification. The diagram also demonstrates the relationship between the capabilities through their links to the base the second layer ‘Technical capabilities layer’. The diagram clearly illustrates the links and ranks of the capabilities drawn for this layer. In summary, ToRCH is divided into layers: technical features, technical capabilities and the operational capabilities. The technical layer is divided into sections: intelligence/skill, interaction and capability performance. The capability performance section is divided into sub-sections: general and independent. The independent subsection is comprises three overlapping categories: physical, social and cognitive capabilities. Finally, each of these contains a set of sub-capabilities and potentially sub-sub-capabilities and these lower tiers are presented with their technological maturity levels.

5.7 ToRCH quality criteria

The ToRCH classification scheme has the following quality criteria:

- The ToRCH hierarchy captures robot capabilities through its classification system. This implies that different robot capabilities are grouped into sections, categories and capabilities according to the criteria defined by ToRCH.
- The criteria for the ToRCH hierarchy (layers, sections and capabilities) are well defined and clearly specified. The criteria of each ranking are clearly outlined.
- The hierarchy also indicates the capabilities within a specific subsection, in tier n, which must follow the specifications, qualities and requirement of the superset, in tier n-1, under which it falls. Hence, all of the specifications, qualities and requirements of the superset, in tier n, are applicable to the subset, in tier n+1.
- Some parts of the ToRCH framework overlap, such as the categories describing the capabilities (tier 4), which allows ToRCH to present the capability with more than one set of specifications. This indicates that some capabilities must follow the specifications of two or three supersets. Such capabilities will, therefore, contain a combined capability.
- The relations between the ToRCH capabilities within the hierarchy are clear and simple. However, the relations as presented are intended to simplify the use of ToRCH rather than to indicate the robot's actual activities and their order of execution.
- The type of relation between an upper and lower tier in ToRCH is a 'sub-super relation' and is clearly known. This relationship is distinguished from other types as it demonstrates a simple inheritance relation, but it does not include the composition of relation to form new relations.
- The levels of technological maturity presented within each of the capabilities are clear and simple, as each additional level improves the capability.

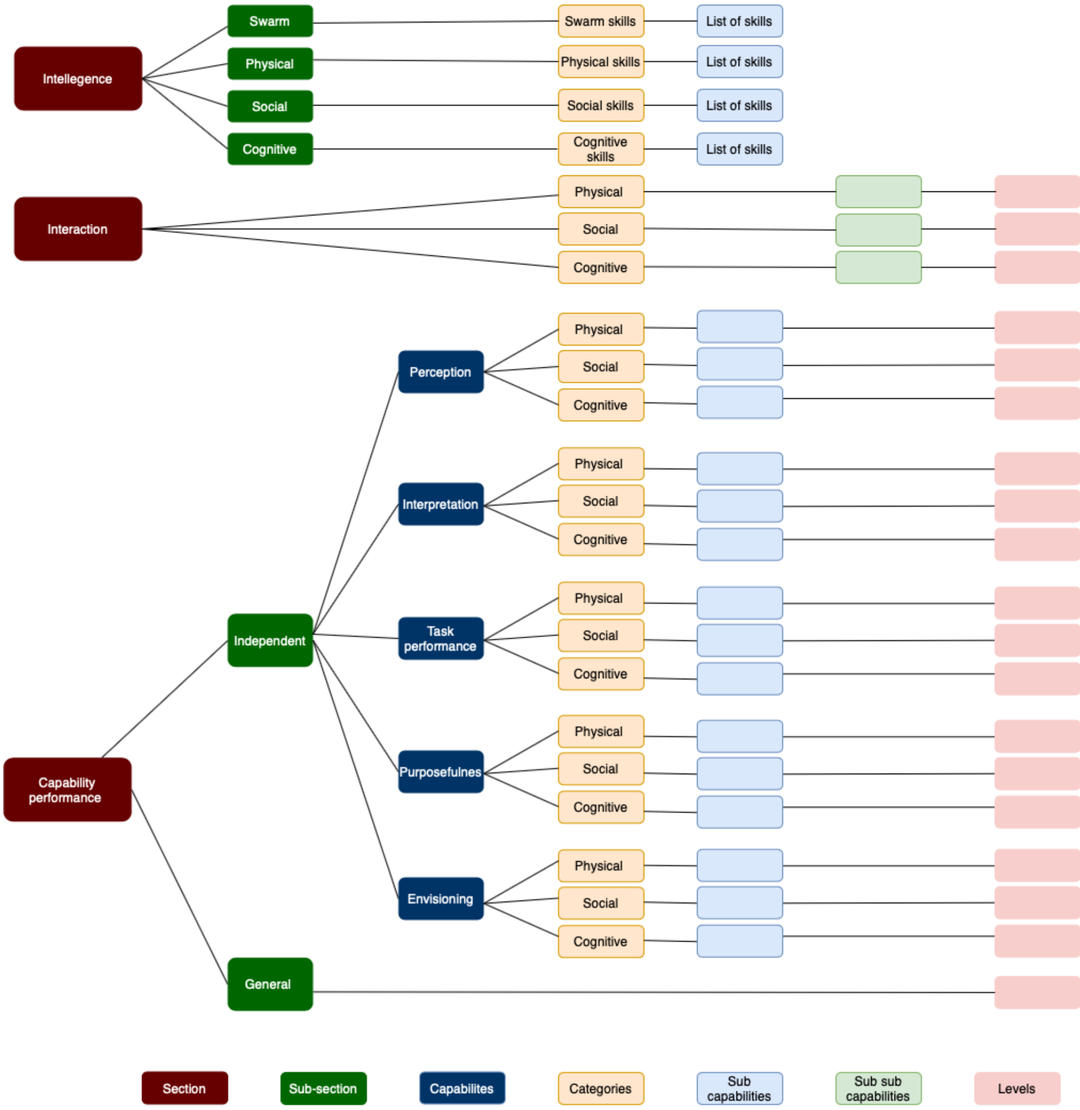

Figure 5.6: Tree diagram presenting the ToRCH hierarchy for the second layer 'Technical capabilities layer'. The tree diagram shows how technical capabilities are branched and allocated. The number of divisions and types of each capability are illustrated. The diagram also demonstrates the relationships between the capabilities. In the diagram every tier within the hierarchy is shown by a different color.

5.8 Guidelines

5.8.1 Outlines in defining robot capability

The ToRCH classification hierarchy captures all robot capabilities. Following the hierarchy within the technical layer ensures that all capabilities are defined. Scoring each capability indicates whether it is present in the robot or not, and, if it is present, at what level of technical maturity. As capabilities are listed in a spectrum from the general to the specific, a score of 0 in the general capability means all sub-ranked tiers will be 0. Defining the levels is the most important step in this process, this process is outlined by answering the questions listed in Appendix A. The quantitative data that results from this process creates the robot capability profile (RCP), the key aspect in defining the robot's performance. The following steps show how this is achieved:

- Answering the questions listed in Appendix A, scores the capabilities in Tables 5.3 and 5.4. The tables list the capabilities from the bottom layer of ToRCH (robot features layer) to the top (interaction capabilities in the technical layer). Reading from left to right, the columns list capabilities from the general with broad spectrum to the specific according to the tiers in Figure 5.5. The capability is scored with either a 0, if it does not exist, or a numerical score in the far right column. Hence, for example, if 'perception" (column 1) does not exist in the robot, all sub-ranking tiers (columns 2 and 3) will rank 0 in the scores column. If 'perception' is present, the user must select the categories of perception (physical, cognitive, social) and fill in the scores accordingly in the far right column. The final section in Table 5.4 contains the skills, behaviors and intelligence of the robot. This section is filled with text and does not require scoring.

- The levels of capabilities are listed in Appendix A. Following the questionnaire presented in that appendix enables the user to assess the scores (i.e., the technical maturity level) for each capability. This is then added to the scores column.

- Finally, the scores create the RCP, which can be used to describe the robot.

Some robots may be developed with novel capabilities that have not yet been defined in ToRCH and assigned a place in the schema's hierarchy. In such cases, each new capability has to be defined and allocated. The process of allocating new capabilities within the ToRCH framework is presented in the next subsection 5.8.2.

5.8.2 Adding novel capabilities to the ToRCH framework framework

As robots become more advanced, the ToRCH framework is broad enough to capture new developments. To add a new capability requires defining its level of technical maturity to give it a score. The capability then needs to be allocated within the hierarchy. Careful consideration is required to assess the capability levels. A new capability can be added using the generic capability procedure, outlined in section 4.4.1. As with all levels, a score of 0 indicates no capability. At level (1), the robot can execute the capability with one main attribute; at level (2), the robot can execute the capability with several attributes; and so forth. If the capability cannot be defined with distinctive levels, as the technology does not yet have maturity levels and therefore, can be only added as a subsection of another capability. But if the capability can be defined with at least two levels, then it can be allocated to the ToRCH framework under its super-capabilities with two levels. The capability needs to be defined with a name that clearly describes that capability. Allocating the capability within ToRCH requires the following steps:

- The capability needs to be defined in each tier of the ToRCH framework. The capability needs to be defined in each column in Table 5.3 and Table 5.4.
- The added capabilities cannot be defined in and allocated to more than one row in the table.
- As capabilities that share properties are nested, the novel capability must fit within the hierarchy. For example, 'object perception' is a physical perception, so it will be within the physical category and the perception capability. Thus, the new capability should be added within the relevant tier.
- Where a capability is not technologically mature, it may be listed as a general capability. Therefore, not all capabilities will be defined down through the tiers to the sub-sub-capabilities. Hence, some capabilities are not defined in the lower tiers but only in the general tiers.

Within the ToRCH hierarchy, the higher tiers (1 to 3) are mandatory. As the capabilities become more advanced, they are more likely to be present down through the lower tiers. The following grammar presents the mandatory and optional tiers for adding a novel capability. The elements presented with an asterisk are the mandatory general capabilities. The items presented in square brackets are specific and optional, and, while it is advisable to include them where possible, yet it is not required to do so. However, the more specific the description of the capability, the better the definition.

$$Layers * \left(Sections * \left([Subsections] \left([Capabilities] \left([Categories] \right.\right.\right.\right.$$
$$\left.\left.\left.\left. ([Subcapabilities]([Subsubcapabilities][Level])) \right) \right) \right) \right)$$

5.8.3 Scoring robot capabilities

The score represents the level of technological maturity of the capability. Hence, a score of 0 for a capability would indicate the capability does not exist within the robot. A level 3 for a capability would indicate the third maturity level for the capability and would score 3 for the capability profile. At level 1, the robot would have basic performance of the capability. At the highest level, the robot would execute several capabilities with an advanced performance that indicates highest maturity. If any capability is not filled, or listed in the Table 5.3 and 5.4, it will be scored as 0 and counted as not existing within the robot. Thus, filling the table correctly with the right score for each of the listed capabilities for the robot is a key aspect in defining its capabilities. In ToRCH, as shown in Table 5.3 and Table 5.4, the scores are grouped according to the capability ranking of sections, subsections, capabilities, categories, sub-capabilities, sub-sub-capabilities and levels. These can be listed or presented for an RCP in separate mini-tables or graphs.

5.8.4 Generating the Robot Capabilities Profile (RCP)

Presenting the scores of the robot capabilities and grouping them according to the tables representing the ToRCH hierarchy offers a specific format for illustrating the scores. This format provides several capability sections that give a comprehensive Robot Capabilities Profile (RCP). This illustrates the capabilities scores in quantitative values rather than a textual description. It captures the essence of robot performance and its purpose in ontological un-ambiguous description (Prestes et al., 2013). The advantages of the robot capabilities profile are:

- It specifies the availability of each capability and a description of that capability's technological maturity.
- The profile illustrates the different capabilities through similar property descriptions.
- Presenting the robot capabilities in this way creates a unified presentation.
- The scores are grouped and nested to include capabilities with similar properties.

Table 5.3: The capabilities list in the layers and sections of ToRCH

Dimensions	Category	Sub-capability, source	Range	Score
Robot feature layer (1)		HW&SW section (A) & (B)		
		mechatronic configuration	0-5	...
	general	parameter adaptability	0-5	...
		component adaptability	0-5	...
Technical capability layer (2)				
Performance capability (A)		Independent capabilities subsection (A1)		
Perception	general	general perception, MAR	0-8	...
	general	environment scene perception,MAR	0-5	...
	general	tracking perception, MAR	0-5	...
	physical	object perception, MAR	0-12	...
	physical	object location, ToRCH	0-7	...
	physical	self location perception, MAR	0-6	...
	physical	social agent location, ToRCH	0-7	...
	cognitive	signals/digital perception,ToRCH	0-4	...
	social	social agent perception, ToRCH	0-8	...
	social	emotion perception, ToRCH	0-5	...
	social	behavior perception, ToRCH	0-5	...
Interpretation	physical	object interpretation, MAR	0-8	...
	cognitive	signals/digital ",ToRCH	0-6	...
	social	human agent interpretation,ToRCH	0-5	...
Task abilities	physical	motion: unconstrained, MAR	0-7	...
	physical	motion: constrained, MAR	0-5	...
	physical	manipulation: grasping, MAR	0-8	...
	physical	manipulation: holding, MAR	0-5	...
	physical	manipulation: handling, MAR	0-9	...
	cognitive	signal/data/info learning, MAR	0-15	...
	cognitive	signals/data/info reasoning, MAR	0-8	...
	social	emotion expression, ToRCH	0-8	...
	social	social behavior, ToRCH	0-4	...
	social	social skills, ToRCH	0-4	...
Acting purposely	physical	acting purposely to objects, MAR	0-8	...
	cognitive	acting purposely to digital, ToRCH	0-2	...
	social	acting purposely to human, ToRCH	0-7	...
Envisioning	physical	objects envisioning, MAR	0-5	...
	cognitive	digital data envisioning, ToRCH	0-5	...
	social	human action envisioning, ToRCH	0-5	...
	cognitive	cognitive envisioning, ToRCH	0-6	...

- It highlights capabilities that are not listed in ToRCH but which can be added.
- It can distinguish between robots based on their capabilities.

Table 5.4: Continuation of the capabilities list in the layers and sections of ToRCH

Dimensions	Category	Sub-capability, source	Range	Score
Technical capability layer (2)				
Performance capability (A)				
sub-section (A2)		general action capabilities, ToRCH	0-2	...
General capability		decisional autonomy, MAR	0-11	...
		task adaptability, MAR	0-4	...
		system dependability, MAR	0-7	...
Technical capability layer (2)				
Interactions capability section (B)				
	general	methods of interaction, MAR	0-8	...
	general	interaction proximity level, ToRCH	0-6	...
	physical	physical motion interaction, ToRCH	0-4	...
	physical-cognitive	object interaction	0-9	...
	cognitive	robot-to-robot interaction, MAR	0-9	...
	cognitive	robot-to-sys/device interaction, MAR	0-9	...
	social	interaction levels of extent, MAR	0-7	...
	social	HRI feedback, MAR	0-8	...
	social	human interaction modality, MAR	0-5	...
	social-cognitive	cognitive social complexity, MAR	0-4	...
	social-cognitive	social interaction learning, MAR	0-3	...
	social-cognitive	social cognitive abilities, MAR	0-7	...
Technical capability layer (2)				
Intelligence section (C)				
	Skills			...
	Behavior			...
	Dimensions			...

- Using the RCP allows robot capabilities to be compared against application requirements.
- It also helps application developers to select the most appropriate robot for their specific requirements.
- It enables designers to describe any future performance in their robot. The RCP helps them to assess which areas will be developed.

5.9 Outcome of ToRCH

ToRCH provides advantages both to robot developers and to users. It covers both the robot and the application domains. In particular, it:

- Offers a method of capturing capabilities that is applicable to all robots.
- Identifies robots through a detailed description captured in the RCP.
- Can be used to classify robots according to type through their capabilities. For example, a social robot must include some social capabilities.
- Can be used as a tool by roboticists to demonstrate their robots.
- Can be used to illustrate the potential capabilities of robots.
- Describes robot capabilities using numeric values.
- Defines robots according to their performance of capabilities.
- Allows for a comparison of robot performance according to their programmable capabilities.
- Differentiates between similar robotics research by clearly indicating which capabilities are included in a robot.
- Identifies the development of capabilities within a robot by allowing for comparison of a robot's performance before and after development.
- Can be used by roboticists to identify the level of capabilities that require development.
- Helps to develop robots by using existing capabilities of robots in the same field or from different fields (Sita et al., 2017).
- Clarifies the strength and limitations of the robot, allowing better development and deployment.
- Can be used to set the configuration of a robot by selecting the levels of its capabilities to fulfill a specific application requirement.
- Can simply and clearly map the robot capabilities and application requirements.
- Robots could be filtered through the ToRCH dimensions, that would classify any type of robot.
- The ToRCH sections are categorized from the broader aspects to the most confine, to capture present robot characteristics and any future development.

The next chapter will adopt the ToRCH framework as a tool to obtain the capabilities of any given robot for different applications.

❋ Chapter 5: Summary

Within ToRCH, robot abilities are allocated to layers, sections, sub-sections, categories, capabilities, sub-capabilities, sub-sub-capabilities and levels as groupings that implicitly assign some abilities as part of others according to their properties. By following the ToRCH hierarchy, capabilities are grouped, ranked and scored leading to the creation of the Robot Capabilities Profile (RCP). New capabilities can be added as the field of robotics develops. This chapter has described ToRCH and the quality criteria of its structure. It has also provided guidelines for its use.

Chapter 6

ToRCH Applications

6.1 Introduction

The previous chapters have presented the Towards Robot Characterization Hierarchy (ToRCH); this chapter will describe how to use the extracted capability scores to present Robot Capability Profiles (RCPs). The most common applications for the RCPs are: to describe a robot capability in a simple and clear presentation; to classify robot types according to their capabilities; and to use the profile in an executable program to map between robot capabilities and application requirements.

6.2 Robot capability profiles

To investigate the extracted robot capabilities via ToRCH, different RCPs were created from the data collected through an online questionnaire, listed in appendix A. This comprised 57 questions that enabled participants to provide scores for capabilities and sub-capabilities, all of which are listed in the ToRCH technical capability layer. The survey was distributed to roboticists from different robotics labs and companies, who were invited to describe the robot they worked with. There were 23 responses for 16 different robots. All the data sets scored robot capabilities, creating for each robot an RCP. The RCPs were used to demonstrate the different applications of ToRCH, as discussed in the following sections of this chapter.

The capabilities questions were listed according to the allocation of capabilities within the ToRCH framework, as presented in the Figure 6.1. The capabilities presented in the ToRCH lower levels were presented first, then those from the higher levels, as section(A)

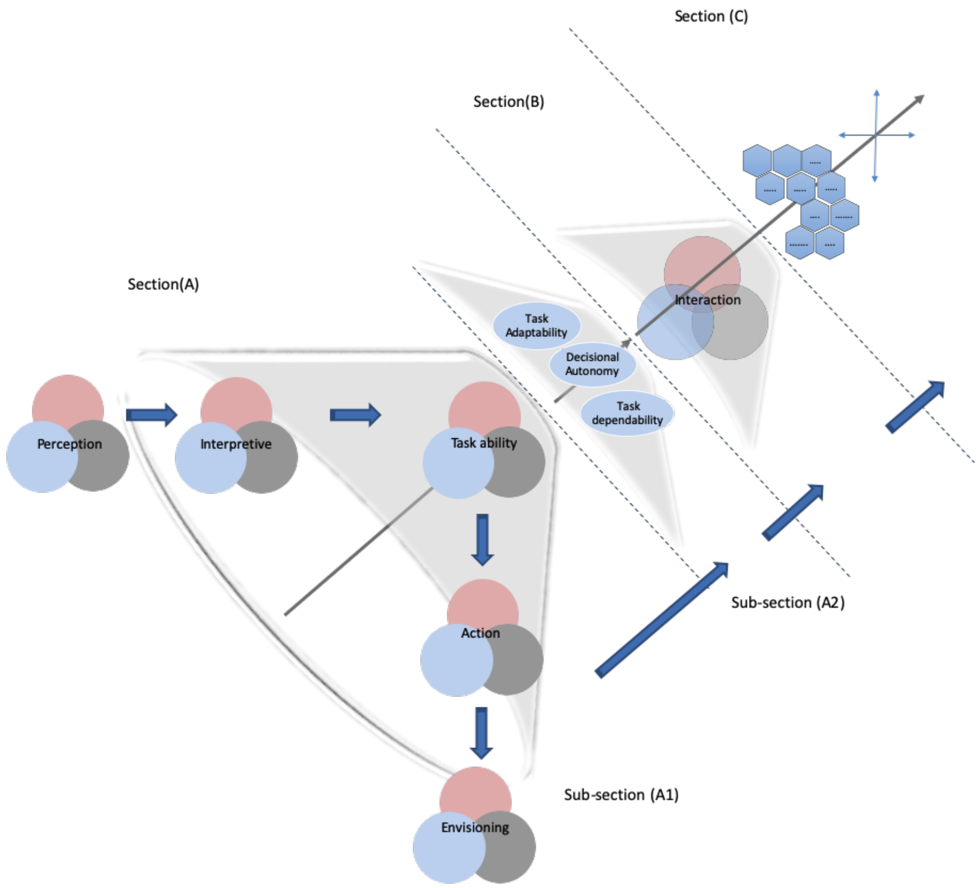

Figure 6.1: The figure presents the order of the capabilities needed to be scored. Scoring starts form the section(A) then section (B) towards section (C); capturing the two subsections (A1) then (A2). For each capability, scores of the physical, social and cognitive types are also recorded. Therefore, the RCP proceeds with the same categorization as the hierarchy. The capabilities questions are demonstrated according to the allocation of the capabilities in the hierarchy, which consequently synchronizes the robot scores accordingly.

first, subsection (A1) followed by subsection (A2), then section(B) and finally section(C). Each capability list includes the physical, social and cognitive types of questions. Along with each question, the technological maturity range of each capability was presented as participants scored each capability. Thus, where a robot did not execute the listed capability, it would be given a score of 0. For the simplest robots, such as the kilobot, the questionnaire took 10 minutes to complete. Where a robot was more sophisticated, such as Pepper, it could take up to 40 minutes. The more capabilities available within the robot, the more time was needed to score the capabilities.

Since the RCP process can identify a robot and classify its capabilities, regardless of the application field or robot type, it can bridge the gap between the robot domain and the application domain. The RCP follows the ToRCH framework, as presented in Figure 5.1, as it scores the first layer (robot features) capabilities to describe the components of the robot. Then the scores of the second layer (robot technical capabilities) to capture the robot performance. Finally, the scores of the third layer (robot operation capabilities) capture application general information and operational focused values. Therefore, the RCP should illustrate these three sections for the different types of capabilities. However, for this research, the RCP concentrates on the second layer of technical capabilities and captures the scores of its capabilities. Hence, this study did not capture any capabilities presented in the first and third layers of ToRCH, nor did this study cover the parameters that would affect the quality of capabilities and influence the performance of capabilities by a robot.

6.3 Utilizing the RCP

6.3.1 Using the RCP to identify robot capabilities

The RCP can identify robots and describe their capabilities in a precise, numeric description. It is generated by grouping scores of similar sub-capabilities, from the same location in ToRCH, and listing them together under the main capability. Hence, the scores relating to perception capabilities were listed one after another, all labeled with the ‘perception’ tag, similarly the scores related to interpretation capabilities were listed one after the other and labeled with the ‘interpretation’ tag, and so forth. If the robot contains a score in any of the sub-capabilities, then it does demonstrate the capability, and if the robot scores zero’s for all of the sub-capabilities, then it does not demonstrate the capability. These scores can be used to create a graph or a table of scores, either of which can be used to describe a robot and to compare them with each other.

Presenting the data

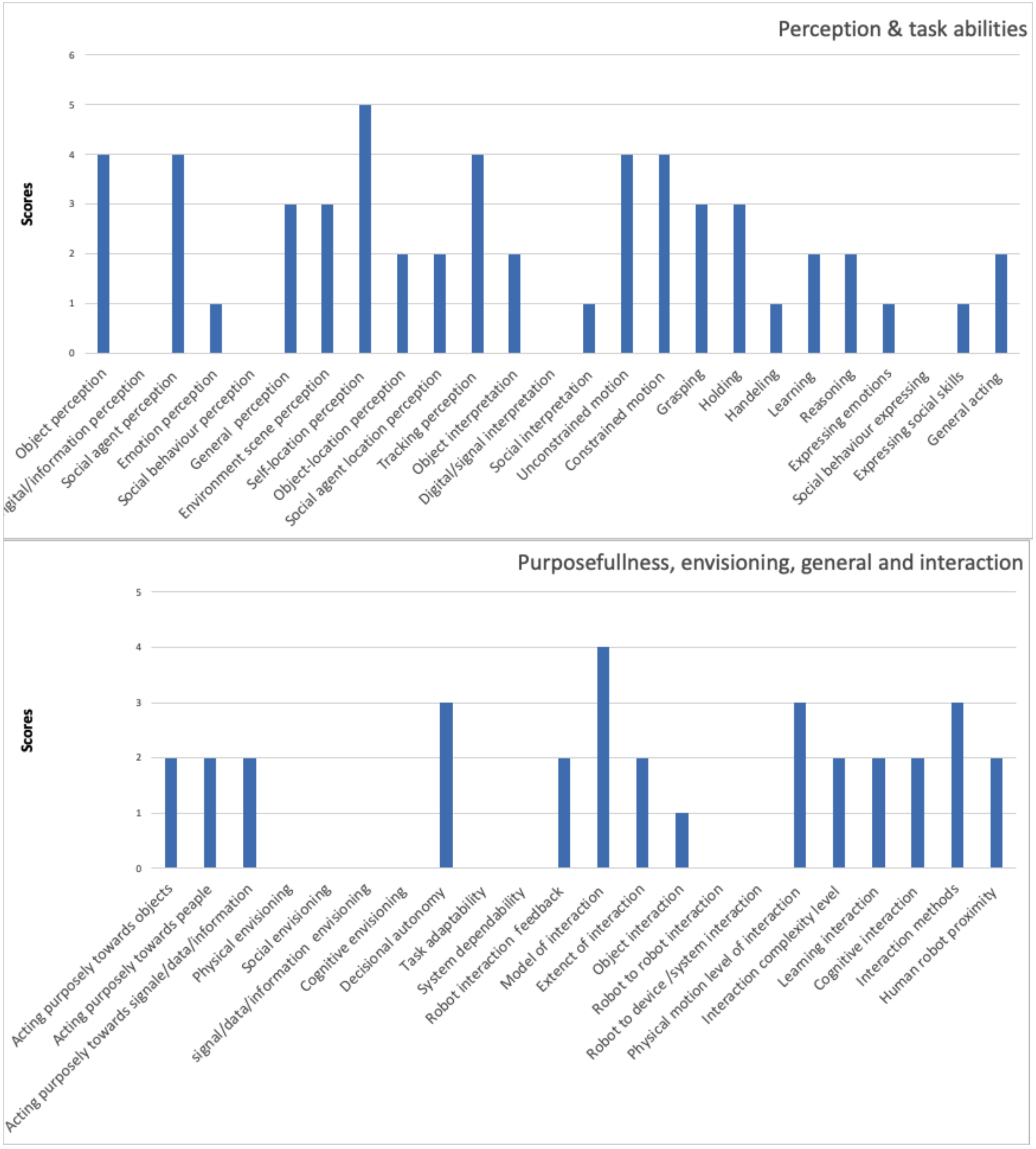

Figure 6.2: This graph profile presenting the capabilities for the social Wakamaro robot

Analyzing the data

The collected dataset from the survey was analyzed. The scores were shown in a bar graph to present the capabilities of each robot individually. To demonstrate the application of RCPs

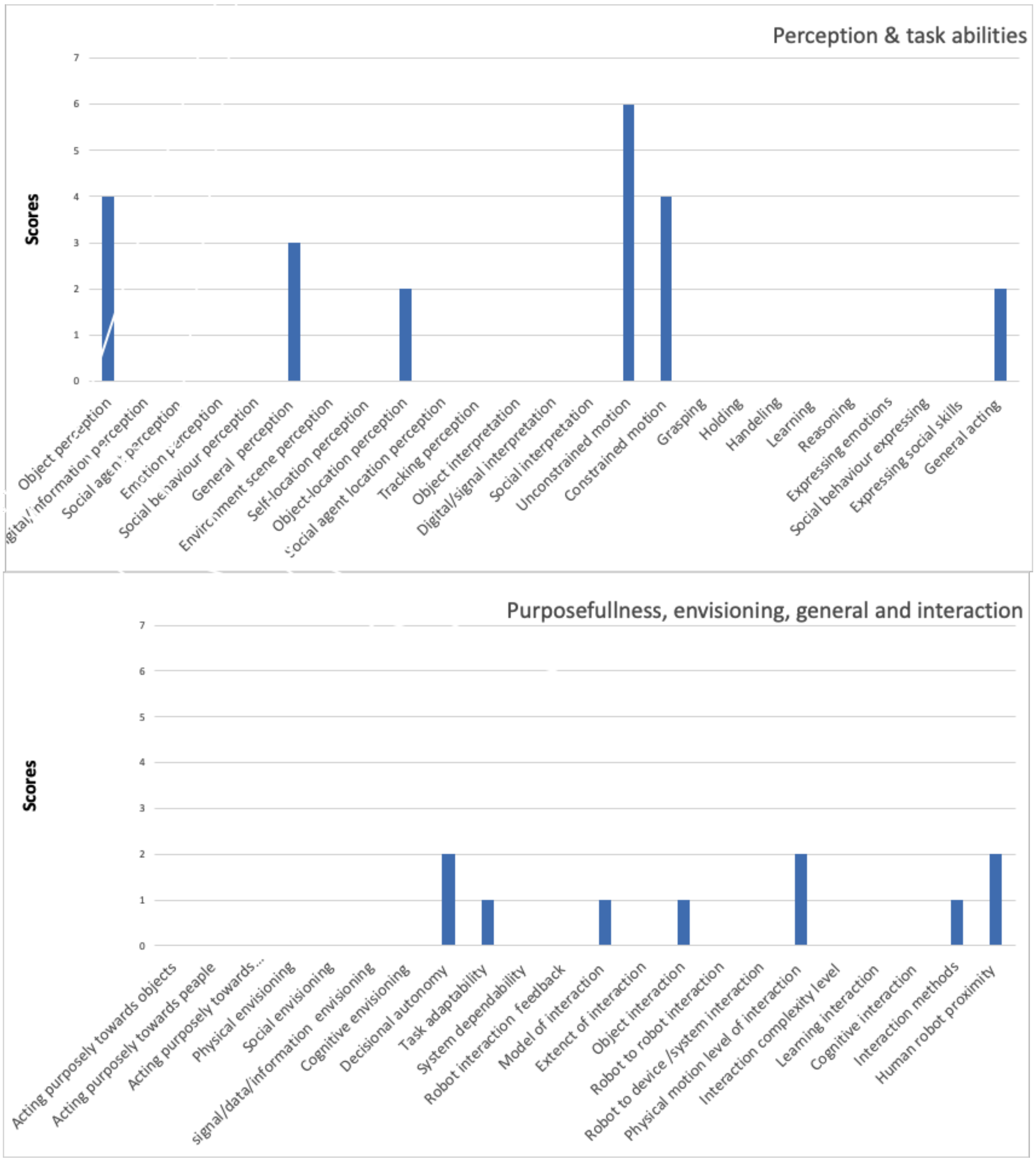

Figure 6.3: This graph profile presenting the capabilities for the industrial Kuka robot

for robot description, this section shows some examples of RCPs that have been extracted and analyzed. The first RCP is for Wakamaro, a social robot (see Figure 6.2); the second is for Kuka, an industrial robot (see Figure 6.3) and the third is for Jibo, the personal assistance robot (see Figure 6.4). Each robot had its capabilities presented in RCP graphs:

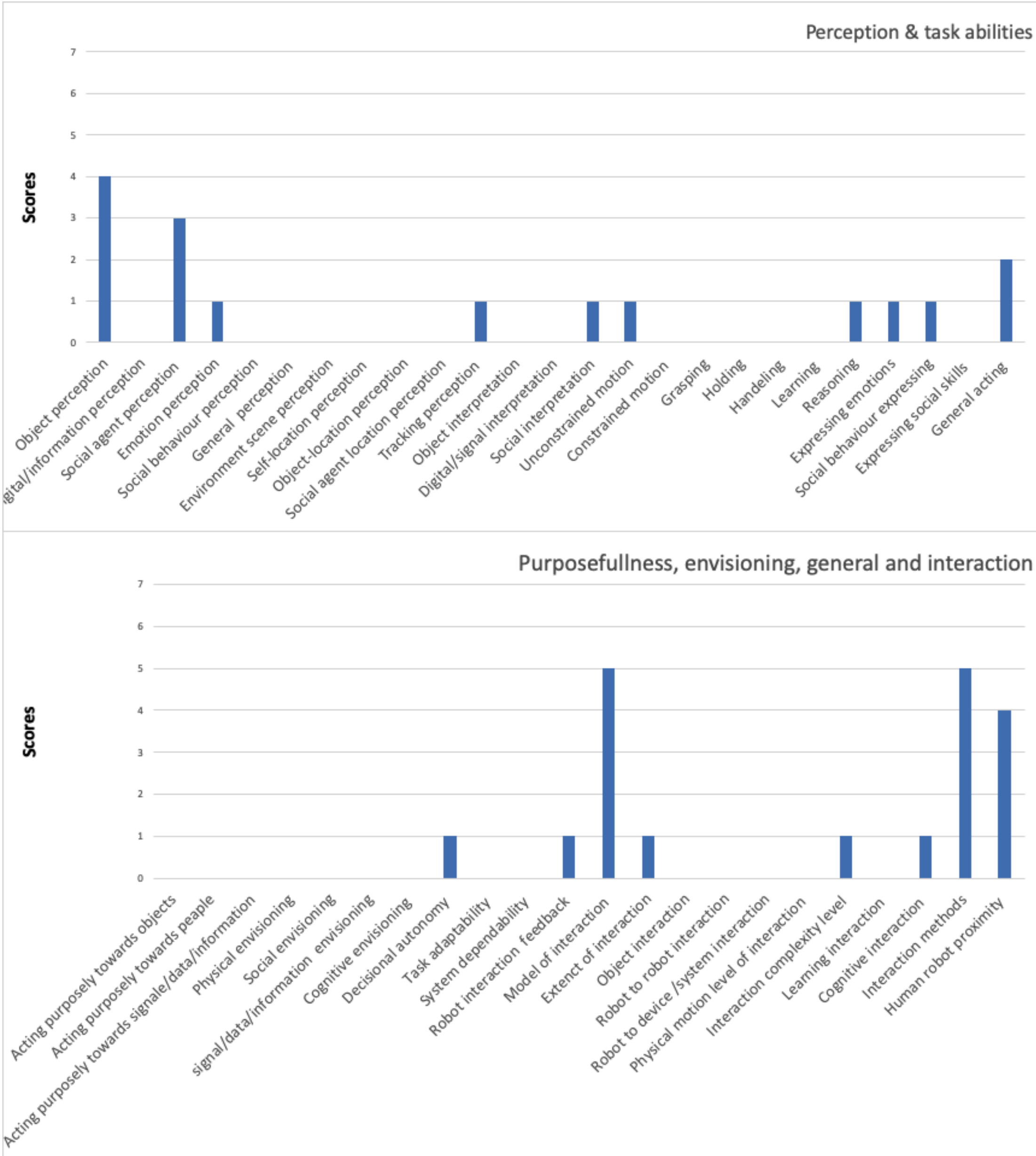

Figure 6.4: This graph profile presenting the capabilities within the personal assistance robot, Jibo

6.3.2 Using the RCP to specify the capabilities of different types of robots

The ability to specify the capabilities of specific robot types is an important aspect of classification. Identifying robot capabilities for similar type of robots indicates the requirements of capabilities for these specific robot types. Robots belonging to the same types should contain the same capabilities since they perform the same behaviors. These similar capabilities can be considered necessary to specific robot types. The capabilities might exist with different technological maturity levels, but their existence is mandatory for that specific robot type. This section presents the most common capabilities for the most known type of robot. The mandatory capabilities for each robot type, are highlighted in the ToRCH hierarchy. The ToRCH framework is presented for each type of robot, as described in the following sections.

Industrial robots

Industrial robots vary greatly in their capabilities and what they are required to accomplish. A simple industrial robot, for example, can grasp and place an item (such as moving a box), and needs to have physical capabilities (grasping, holding and handling capabilities at levels more than zero). These types of industrial task would not require any cognitive and social capabilities. However, some of the leading industrial robots may have learning or human interaction skills or are designed to perform in an open environment where they are surrounded by other robots or humans. Thus, the simplest version of these robots do not perform any cognitive capabilities, such as learning and reasoning, or social capabilities; rather, they perceive humans as 'objects' within their environment and would demonstrate 'object perception' rather than social agent perception.

But if such robots perceive human features and physically move away from them, then they might have signal or tracking perception and a low level of social agent perception. Such a robot usually does not perform cognitive interaction to interact with other devices or robots, but it may perform some. In most cases, it performs physical interaction to interact with objects. The interactions are developed within the robot purely to support it with dexterity capabilities and skills related to moving objects. These skills support the robot with morphological/physical intelligence. Therefore, according to the skills that were developed, the morphological/physical intelligence dimension is demonstrated and expressed further than any other intelligence dimensions. The RCP for this type of industrial robot can be compared to the RCP for similar robot but has, for example, an additional appendage to detect whether it is gripping an object or not. These two robots would have similar low-level social and cognitive capabilities, as the additional detecting feature would not increase the need for cognitive or social capabilities. Instead, it would increase the level of the physical manipulation and handling capability. The interaction skills and intelligence description of both robots would be the same. Most industrial robots need to have some of the capabilities listed in the physical sections of ToRCH, as presented in Figure 6.5.

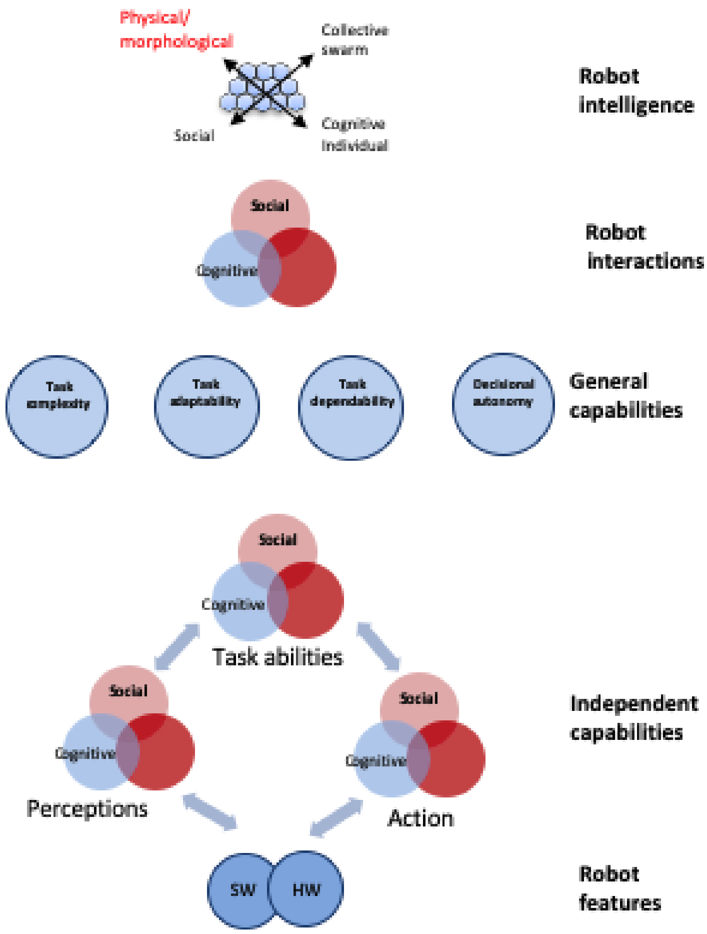

Figure 6.5: Industrial robot capabilities specifications presented in ToRCH. The robot must have physical capabilities specifications in any of the physical sections of the ToRCH framework, illustrated in red in the diagram.

Social robots

The social aspect of the robot can be situated in any of the social categories within ToRCH. Social robots are engaged in diverse social scenarios and thus demonstrate great variety. The first example is an RCP for Zeno (Hanson et al., 2009). Zeno has both hardware and software in the first layer of ToRCH. In the second layer, if Zeno is programmed to identify people, it would demonstrate social perception and social recognition capabilities, accordingly it would be situated in the social perception category. If Zeno is programmed to name the identified person and greet him or her, it would be situated in the social task abilities category. If Zeno is programmed to act purposefully with social intent, then it would be situated in the social action category. The interaction categories are defined by Zeno's social interaction cycles.

For example, if Zeno is programmed to perform a greeting, such as waving and shaking hands, it would have two skills to support social intelligence. If Zeno is programmed to say “hello”, presenting some verbal/linguistic skills, it would demonstrate social intelligence. In the third layer, the operational capabilities of Zeno (include operational aspects such as cost, duration of performance, safety issues, testing and training, acceptance and usability rate and other operational environment capabilities) are proposed as future-work as they are not covered in this thesis.

A second example is the NAO robot (Gouaillier et al., 2009). For this example, a NAO robot is programmed to perform a sequence of activities. In the first and third layers of the ToRCH framework, the NAO would cover the same concepts as described above for Zeno. In the second layer, the programmed activities would describe NAO’s capabilities as presented in sections (A), (B) and (C) in Figure 6.1. In this example, NAO’s first activity is to recognize a predefined red ball on the floor among other unknown objects. Performing this activity requires the robot to have object recognition and interpretation capabilities. The second activity is to recognize some predefined faces of people in the room and move towards them, which requires the robot to have social perception, self-location perception, social agent location perception and physical mobility capabilities. The third activity is to have a social conversation and perform a dance. Performing this activity requires the NAO to have social interaction and motion capabilities. The NAO robot also needs to have some basic levels of decisional autonomy. However, it is not performing any task adaptability nor any task dependability, as covered in subsection (A2). In section (C), the activities performed by the NAO robot present the robot with searching and social skills. It is these social skills that are required for a robot to be classified as social; hence, social robots need to have some capabilities listed in the social sections of ToRCH, as presented in Figure 6.6.

Simulated robots

When considering a simulated robot, most are presented in the first layer (robot features) with software and no hardware, as they are simulating an actual robot but with no physical machine. These robots are presented on a screen, sometimes simulated with three dimensions for display, manipulation and analysis, as presented in Figure 6.7.

BabyX (Lawler-Dormer, 2013) is an example of an interactive simulated robot. If BabyX is programmed to identify and track an object, it would be situated as having physical perception. If BabyX identifies a person, it would have social perception. If BabyX is performing learning, it would be situated as having cognitive task ability. As

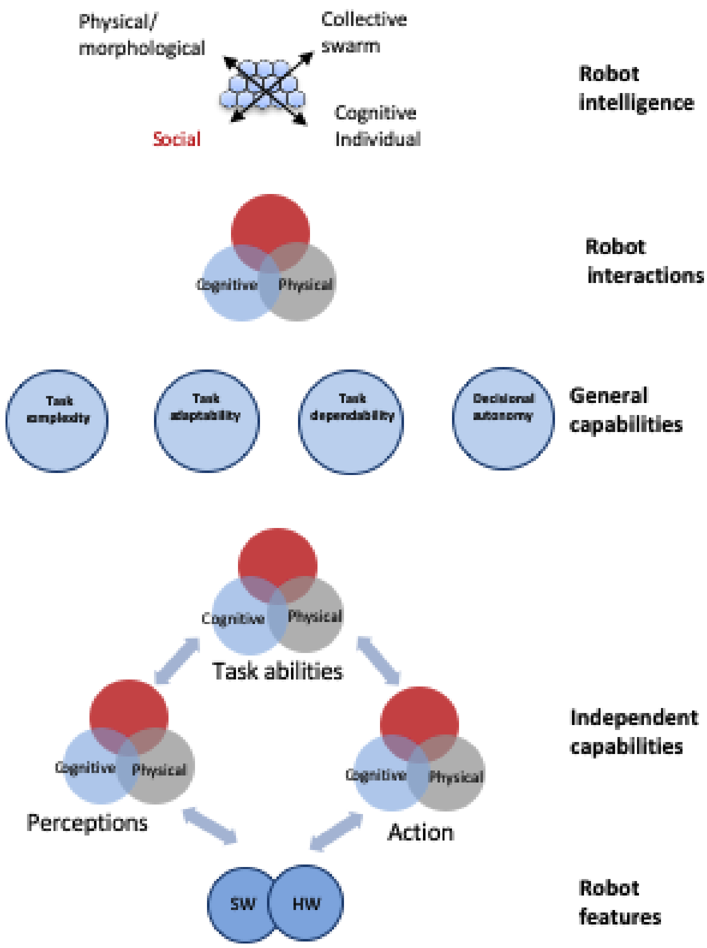

Figure 6.6: Social robot specification can be presented in any of the social sections of the ToRCH framework, illustrated in red in the diagram

BabyX is simulated, it cannot perform any actual physical capabilities. BabyX's interaction cycles will define the presentation of a specific skill. If BabyX's interactions present some verbal/linguistic skills, then it will demonstrate social intelligence. If it is programmed to perform interaction for the purpose of presenting learning skills, then it will have cognitive intelligence. For the third layer, the operational capabilities within BabyX (such as cost, safety, training, acceptance and usability) proposed to be defined in the future as it is not covered in this thesis.

Some simulated robots have been developed by researchers to demonstrate the performance of robots. The robot activities can be physical, cognitive or social. All of these activities can be presented in ToRCH as simulated tasks. In this case, ToRCH is capturing

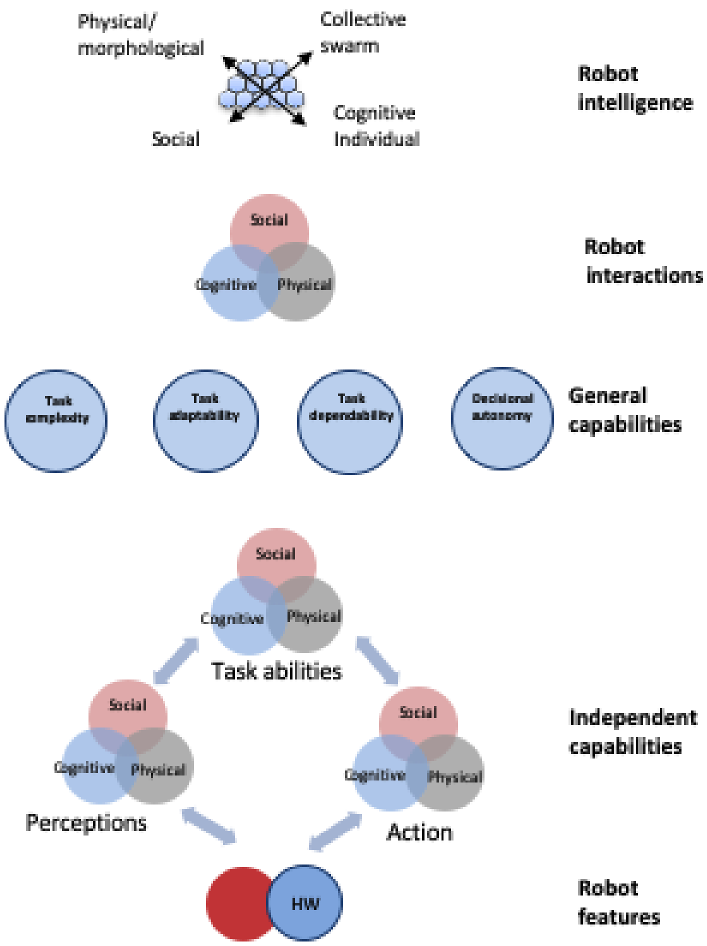

Figure 6.7: Simulated robot capabilities are presented in the ToRCH framework. Their capabilities can be available in any of the ToRCH sections. Simulated robots present as software, without any physical presentation, therefore they do not contain any mechanical hardware and they are presented as a software in the features layer, highlighted in red.

simulated activities for physical, social and cognitive tasks, rather than the actual performance.

Swarm and swarm-anoid robots

Applying ToRCH to create an RCP for a swarm robot requires presenting the individual robot capability in one profile (using the same concepts applied for the social robot) as well as the capability of the swarm as a whole in another profile. This then captures the collective behavior that emerges from the collective interactions of swarm robots with their surroundings. If the swarm contains robots with different capabilities, then all the robots should be illustrated with separate RCPs; thus, there needs to be an RCP for each individual type and an RCP for the whole group (Trianni, 2017). Since swarm robots have been developed with different capabilities, there are no mandatory designated areas, but most of the robots in swarm robotics interact with each other, so they have a cognitive interaction dimension known as digital interaction capabilities, which is required for robot-to-robot or robot-to-system communication.

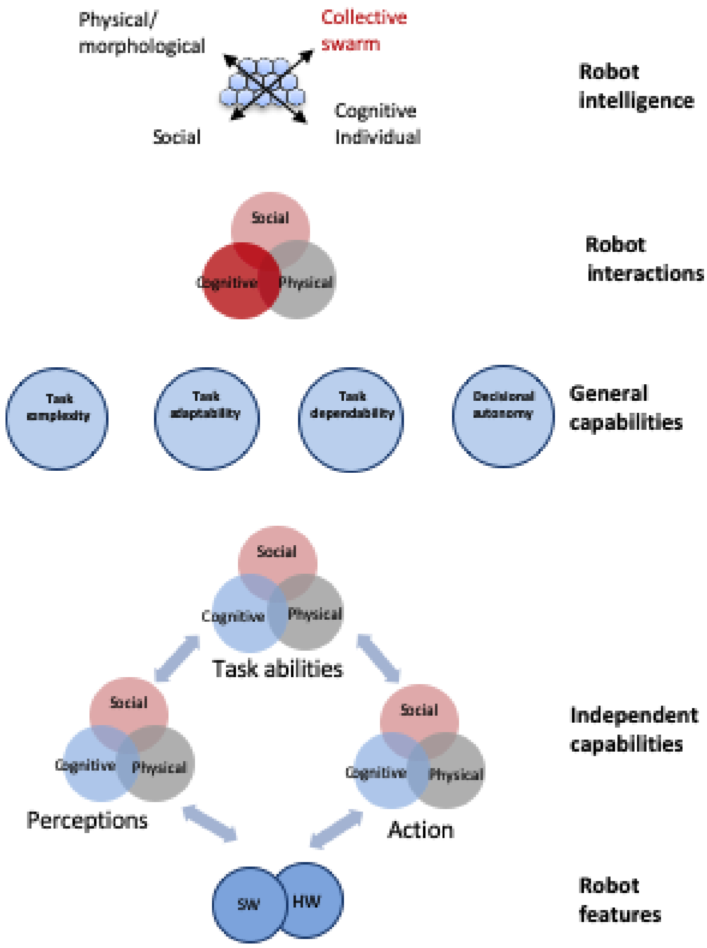

Figure 6.8: Most swarms perform cognitive interaction and present collective swarms intelligence, therefore these two sections are highlighted in red in the ToRCH framework.

Regarding the system profile, the robot-to-robot communication is considered within the individual robot, so it is not captured for the whole swarm profile, unless the swarm as a whole is communicating with another system. Both individual and whole swarm profiles require some homogeneous/heterogeneous characteristics that use signal and/or cue-based interactions (Ferrante, 2013) to support the collective or swarm intelligence, as presented in the third section of ToRCH. These skills map the capabilities into specific swarm requirements, as presented in Figure 6.8.

Telepresence robot or teleoperated technology

These two types of remote-handling robot need specific communication settings, which are listed in the first layer of ToRCH, the feature layer. The robot settings will affect how such robots perform any capabilities according to their design. Therefore, the capabilities for these robots can be in any of the ToRCH sections depending on their design aim.

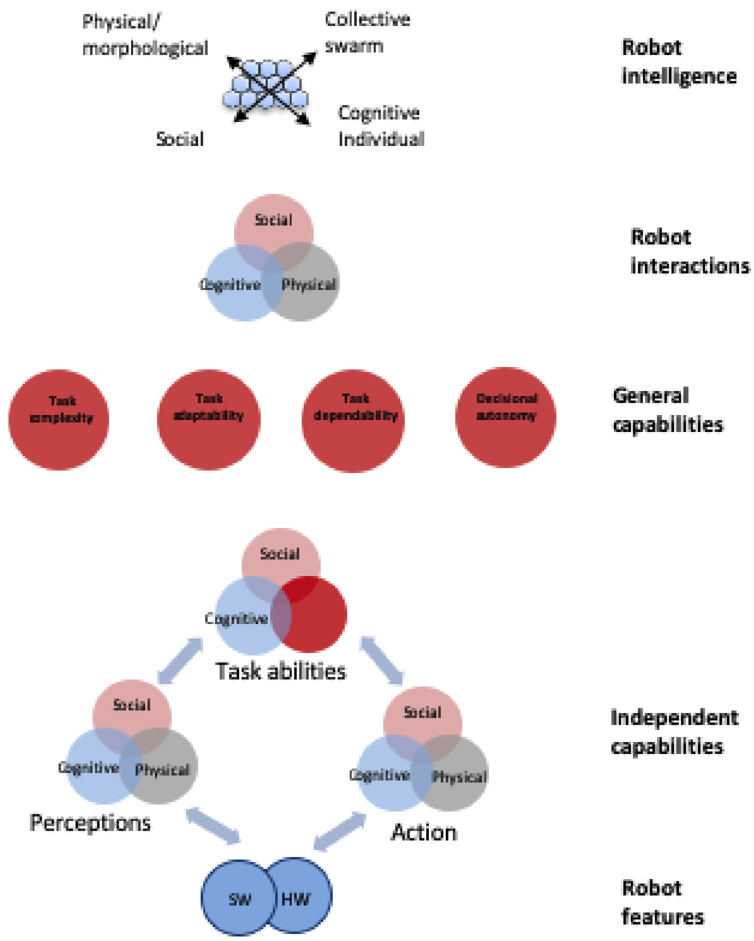

Figure 6.9: Teleoperated technology is presented with any capabilities listed in the ToRCH framework. The capabilities levels presented in the red section, such as physical task performance are required within the robot with higher scores.

The teleoperated robots can have capabilities listed in any part of ToRCH, as long as the capability is remotely controlled. For example, the da Vinci surgery robot, that facilitates complex surgery by enabling a minimally invasive approach, is teleoperated and controlled by a surgeon from a console. The capabilities of such robots require task adjustment capabilities within the environment it operates on, where these adjustment capabilities would help the person to make decisions on tele-operating the robot. These adjustment task capabilities are presented in the general capabilities section of ToRCH and are highlighted in red in Figure 6.9. These capabilities allow the robot to be teleoperated without the existence of human. The robot also requires a teleoperated communication method that is captured in the interaction section of ToRCH.

The telepresence robot performs physical mobility. These robots usually do not perform any physical tasks but they could be enhanced with cognitive capabilities, for example analyzing signals, data or information in the event. Also, the telepresence robots do not perform social capabilities as the social skills and interaction are performed via the person in the telepresence.

Modular robots and re-configurable robots

Modular robots and re-configurable robots are designed to perform various tasks that require different morphological structures. They can switch between shapes to perform their tasks and are capable of re-configuring themselves. Therefore, their hardware and software, presented in the first layer of ToRCH, require the modules to disconnect and reconnect according to the required task.

These physical capabilities to disconnect and reconnect are captured in the hardware/software and are presented in the first layer of ToRCH, the feature layer. The wider capabilities of these types of robots are not limited to a specific section of ToRCH as they can be developed to perform a wide range of tasks. As most of the modular and re-configurable robots perform different levels of task adaptability, which is presented in the general capabilities section of ToRCH. This general capability section covers these robots' necessary capabilities, as presented in Figure 6.10. To capture the capabilities of these robots, each individual module should be presented in an individual RCP, with any combined set of perform-able modules in another RCP, as illustrated with swarm robots.

6.3.3 Using the RCP to map between application and robots

ToRCH captures various capabilities where it can map between robot capabilities and application requirements. The mapping process is one of the most important elements to improve

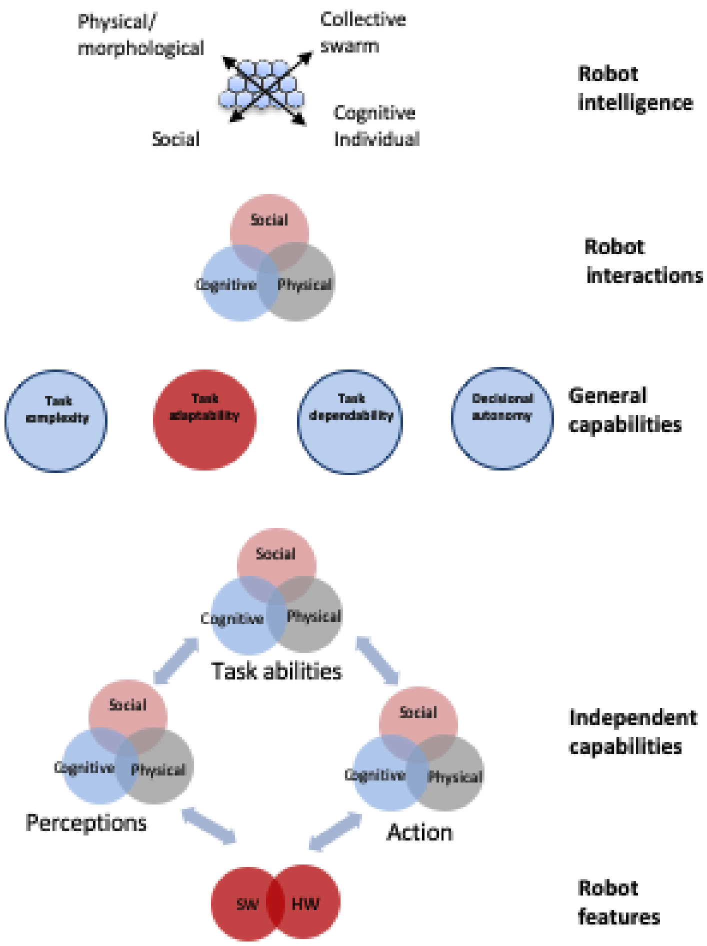

Figure 6.10: Modular robots and re-configurable capabilities are presented in the ToRCH framework. Their capabilities are heavily concentrated in the red sections of the hierarchy

robot development and increase their deployment. Including Moore's Framework within ToRCH, as was discussed in Chapter 3 provides the ToRCH dimensions which are mirrored in the application domain, creating the application requirement profile (ARP). The ARP and the RCP are illustrated in Figure 6.11. This mirroring maps between applications and robots. These links help to map and evaluate robots with their applications and vice versa. Applying ToRCH, for example, on the technical link (the lower link in the model) supports the technical assessment with further subdivided evaluations. The sub-evaluations and assessments follow ToRCH to present the relationships between the application requirements and the currently available technology in the robot. To discover the detailed links between the robot domain and the application domain requires applying the ToRCH dimensions to shape the mapping process.

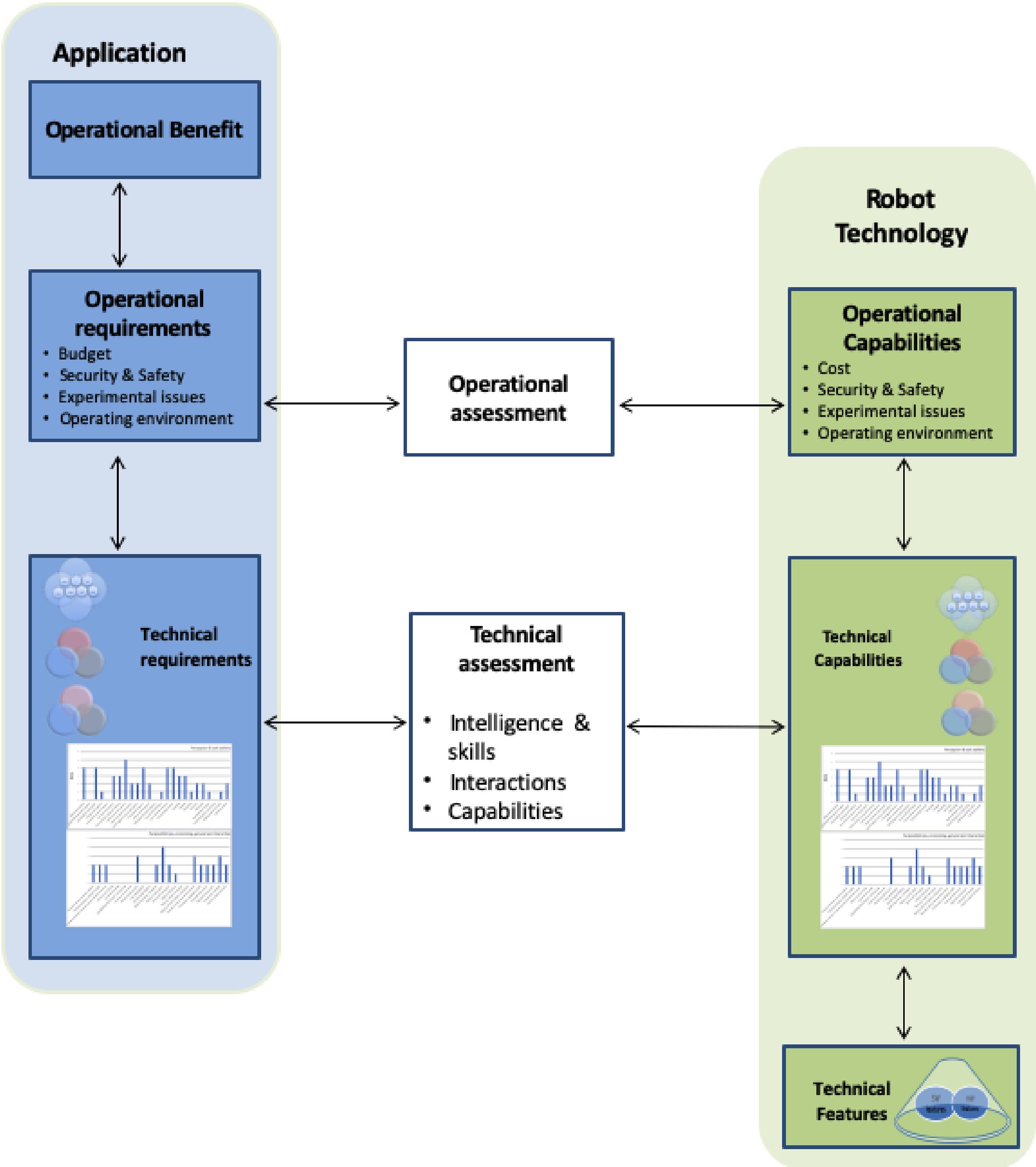

Figure 6.11: Applying ToRCH technical dimensions in the robot domain and the application domain helps in assessing the technical layer. Operational requirements are also known as non-functional requirements, and the technical requirements are known as the functional requirements.

The bridging process relies on extensive assessments between domains, one being the application with its requirements and the second covering the robot technology with its features and capabilities. The assessments between the domains are applied on the operational and technical layers. Applying the operational dimensions to map between operational requirements and operational capabilities requires identifying operational aspects, such as budget, role and operational environment, also known as non-functional requirements. These operational dimensions are captured in ToRCH but are not covered in this study; instead, this study focuses on applying the technical dimensions listed in ToRCH, and on the relations between the technical capabilities of the robot and the technical requirements of the application, as presented in Figure 6.12.

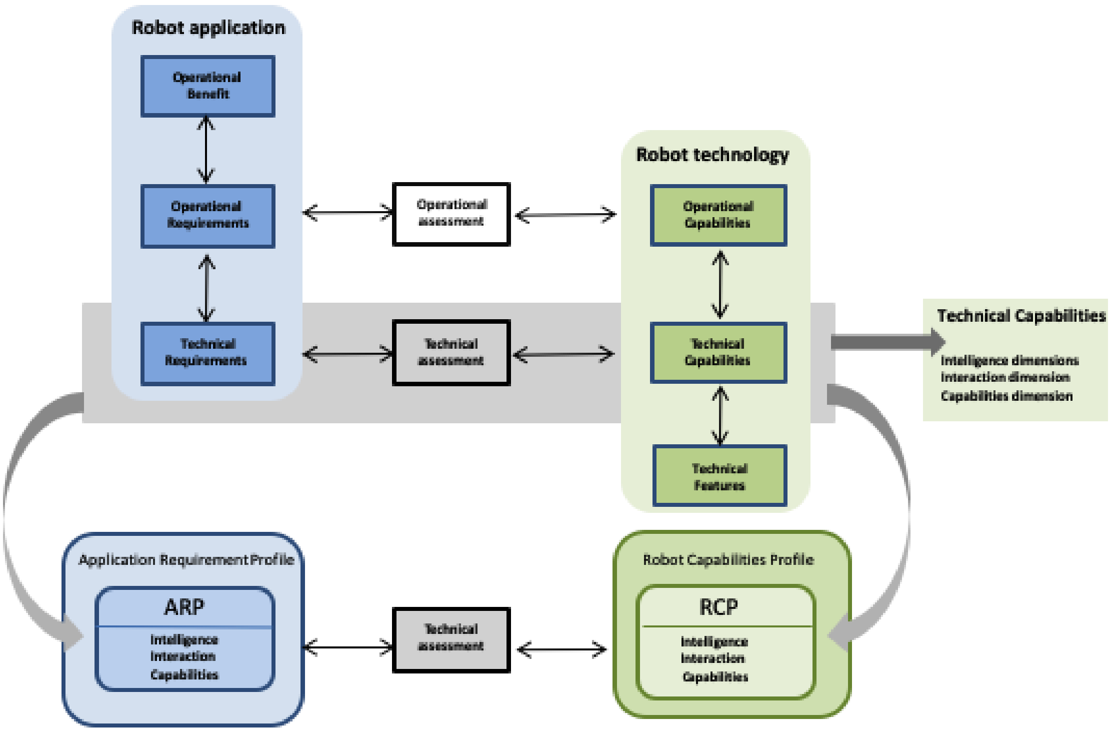

Figure 6.12: Applying ToRCH technical dimensions in the robot domain and the application domain guides the assessing process between the Robot Capabilities Profile (RCP) and the Application Requirement Profile (ARP).

Applying the dimensions on both capability and application domains helps the technical assessment process. Each robot application must define its technical requirements, and each technical requirement maps to the appropriate robot capabilities. These capabilities are the main aspect of this research, which focuses on how to identify capabilities in the robot and accurately map them to the required robot applications. These capabilities include interaction and intelligence aspects, such as the interaction cycles performed by the robot and the goals of the robot performance. These mapping considerations are presented in Figure 6.13.

ToRCH, therefore, allows for the creation of an RCP and, through the application domain, an ARP. Bridging the gap between the two profiles involves filtering and matching among

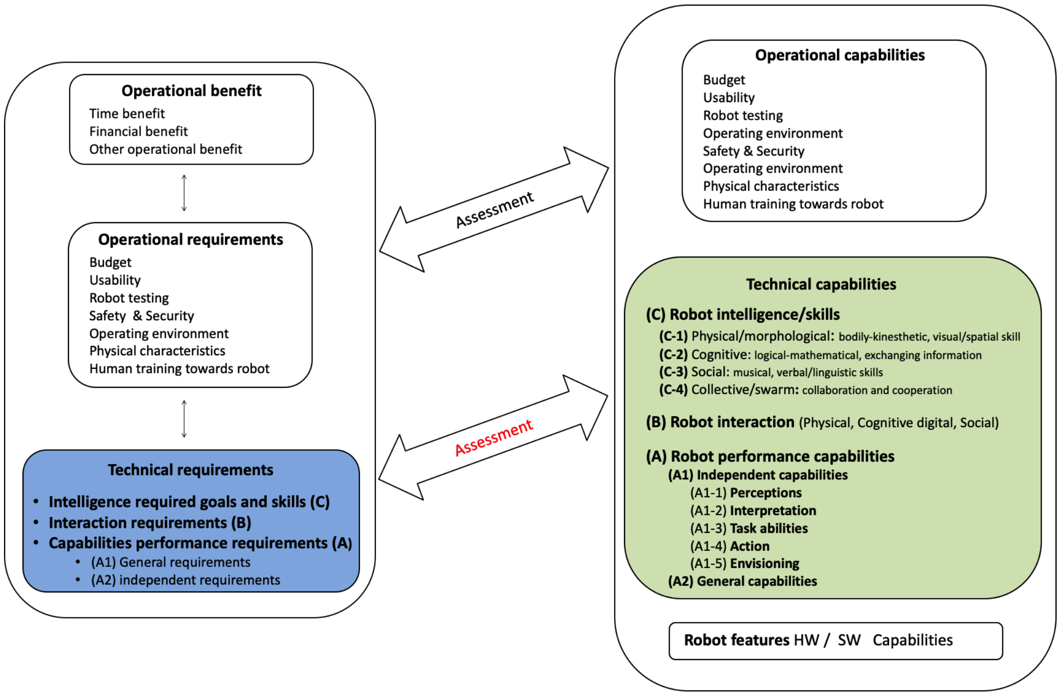

Figure 6.13: The robot domain is presented through the three layers of ToRCH: the robot features layer, the robot technical capabilities layer and the robot operational capabilities layer, illustrated on the right side of this figure. Any given robot should be described by scoring the capabilities listed in ToRCH according to these three layers. The application domain, illustrated on the left side of the figure, is also presented in three layers: the technical requirement layer, the operational requirement layer and the operational benefit layer. This study concentrated on describing the technical capability layer, the second layer in the robot domain, which is proposed to be mapped to the technical requirement layer available in the application domain.

different technical capability dimensions provided in the profiles through the support of ToRCH layers, sections, and subsections, as presented in Figure 6.14. Each profile is presented on one side of the framework.

The RCP helps identify the capabilities within a robot, including missing capabilities, making it easier to deploy the robot according to its specifications, even if the robots are from a different field. Additionally, using the RCP and applying the Moore Model clarifies the abilities and limitations of the robot, and helps in comparing the RCP and ARP allowing better robot deployment decisions. The mapping process between the RCP and ARP helps in defining the dimensions of assessment, as presented Figure 6.14.

The RCP defines what capabilities need to be developed in the robot and helps define which capabilities need to be improved to satisfy specific application requirements. This will

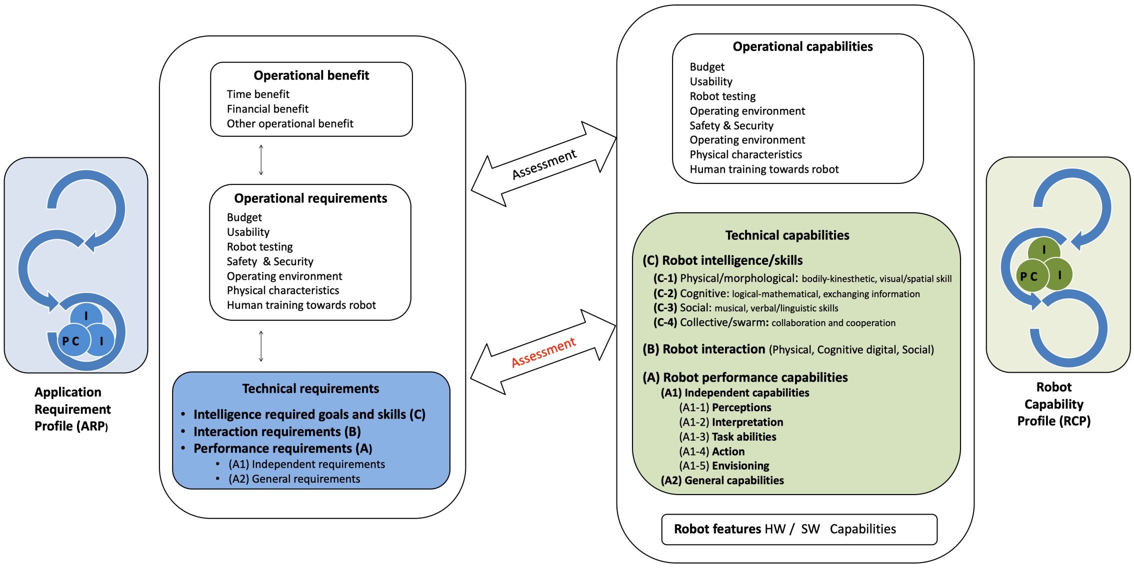

Figure 6.14: The assessment of RCP with the ARP, through the three sections of the ToRCH, maps between the two domains. This study focuses on the technical assessment, the middle layer in the RCP and the bottom layer of the ARP. The application requirements are assessed against the robot capabilities. Also, the interaction requirements are assessed against robot interactions. And finally, the intelligence requirements, regarding goals and skills, are assessed against the robot intelligence goals and skills.

help business sectors, as well as researchers, make decisions about how to deploy a robot more effectively. Additionally, the RCP and ARP help to develop robots by using available capabilities in specific fields and adapting them to other fields. Also, the RCP and the ARP would help in defining the common language between the two domains, as each domain has some of its own terminologies.

The ToRCH framework demonstrates what needs to be evaluated between requirements and robot capabilities. ToRCH's layers, sections and subsections function as assessment dimensions for evaluating and determining robot applications and the most appropriate robot, indicating what changes are needed within the robot to fulfill a specific application requirement. The assessment dimensions will also highlight a robot for an application by clarifying the most advanced features essential for a particular application. In general, ToRCH highlights the most suitable capabilities through its layers, sections and subsections, as well as those that are most incompatible with an application. ToRCH also defines important details between the domains by presenting them in a structured hierarchy. These details can be estimated for each robot and application requirements, and the ToRCH framework outlines dimensions that support linking between the robot and application requirements. ToRCH will, therefore, allow the identification of any robot application and appropriate robot. It will also identify the changes needed for a robot to fulfill the specifications of an application requirement, and, for any given robot, it will find most suitable application because it clarifies and defines the capabilities and features essential for a particular application. By using the robot-application mapping framework, ToRCH supports both the robot and application domains by assessing the essential layers and sections for developing robot capabilities in laboratories, and by identifying application requirements that are developing within the application domain. Therefore, the ToRCH presents a new method for outlining robot descriptions, specifications, evaluations, and validations and help in mapping them to the application domain (Moore, 1993).

6.4 Automating ToRCH

6.4.1 ToRCH application

The ToRCH classification dimensions and structure can capture any new features and capabilities within a robot. The ToRCH dimensions create a checklist that helps define robot capabilities and applications requirements, and it can thereby present a border for defining robot characteristics and define the mapping process more accurately. As ToRCH shows the common dimensions in both robots and application dimensions, it facilitates mapping between them and supports the electronic assessments. Although this study did not cover the application requirements nor it define how to score the requirements, it demonstrates how to apply the scoring scheme to map between the robot capabilities and the application requirements. Therefore, this automated application version presents a demo of ToRCH and how it could serves the robotic domain as well as the application domain. This application shows ToRCH as an electronic tool to bridge between the domains with an executable application presented as a proof-of-concept. It was developed to illustrate the relation between Robot Capabilities Profile (RCP) and Application Requirement Profile (ARP).

The application is a working computational model for categorizing robot capabilities and application requirements. To provide a mapping between the RCP and ARP, the ToRCH flexible layers and sections are used to develop the automated mapping via a Java application. The application is designed to query robots according to the user requirements. The user specifies the capabilities and required levels for each capability and the program can identify a robot according to the user's specifications, as the application queries the database and returns the robots with the required capabilities score.

The application was developed to exhibit the relation between the robot and application domains in an executable version. The application also demonstrates the comprehensive ToRCH framework in a single electronic form. It illustrates the possibility of ToRCH in defining robots through an RCP, as well as the possibility of capturing user requirements as presented in an ARP. The application also demonstrates the usability of the ToRCH dimensions in assessing between the application request and the robot capabilities, as presented in Figure 6.15.

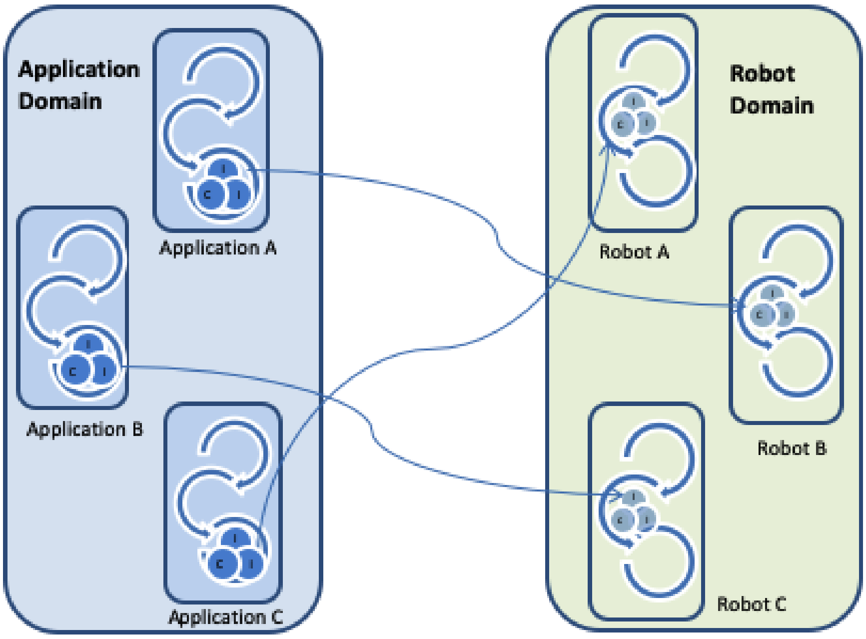

Figure 6.15: Mapping between Application Requirement Profiles(ARP) and Robot Capabilities Profiles(RCP). The robot capabilities scores are captured via ToRCH for robots the A, B and C. In the figure, each of these robots are presented with RCP. Each profile contains three main layers, according to the ToRCH hierarchy, the feature layer, the technical capabilities layer and the operational capabilities layer; which could also be matched to one of the ARPs profile available in the application domain, if the match exists. The application domain is presented through the ARPs, where each application has its own ARP that contain three layers description. These layers could be mapped with the RCPs layers in the robot domain to find a match between the application requirement and a robot.

6.4.2 Database tables and data collection

The scoring of robot capabilities was collected in this research through an online survey. The scores were distributed in an Excel sheet and stored in the database as robot profiles. The entire captured datasets were provided by the researchers and robot developers who participated by scoring the robot they worked with. The participants were from a variety of labs and included Sheffield Robotics. Therefore, the datasets consist of different types of robots, and the robot scores were presented as RCPs and were saved in the MYSQL database to be used for this application as a demonstration of ToRCH. The java application was connected to the MySQL database 'sample', with other two tables that have been extracted. The first table, 'ROB', contains all the robot capabilities presented in the survey, as listed in the technical layer of ToRCH, including skills and intelligence. The second table is 'SKILL' which contains all the skills to be requested in the robot search or the skills needed to be added, if not already listed. The tables were also created with automatic IDs so that new records can automatically be added.

6.4.3 Object-oriented aspects of ToRCH

The application adds robots' capabilities scores to the database; the robots in the database are searchable depending on the ToRCH dimensions. The application tool reads the records from the saved database and allows the user to select the capabilities scores of the required robot. The scores are entered through a GUI form that is also designed and structured according to the ToRCH framework. The form saves the requested capabilities values and queries the database accordingly. If any object matches, it is displayed to the user as a robot that fulfils the user's requirements. The application is developed according to the Model-View-Controller (MVC) architecture to separate the application into three major components. The MVC components are as follows:

The first package is the Controller containing the 'DataBaseConn.java' class and the 'Query.java' class. The 'DataBaseConn.java' connects the application with the database and the 'Query.java' class has a 'getRobosObjectList()', which also takes the SQL statement as an argument and returns the result from the database as an array list of robot objects. This class also contain an 'insertRecord()' method which inserts a new robot record into the database. The robots are scored according to their capabilities following the ToRCH categorizations and scoring system. The user of the application must have the capability scores of the robot that he is searching for. And the application then searches all the robots in its connected database to find the matching robot for the specified requested score. The requested scores should contain the technical requirements, which are also known as the functional requirements. These technical requirements are mapped to the robot capabilities which are captured and used in detailed in this study. The requested scores should also contain the operational requirements, known as the non-functional requirements, that are mapped to the operational capabilities of the robot, however neither are covered in this study, as presented in Figure 6.14.

The second package is the Model. In this package, there is one class called ‘Robots.java’, which presents a blueprint model for the robot object. Each RCP saved in the database is illustrated through this robot class by creating a robot object (presented in Figure 6.16 as objects) and saved in a linked list for later use. The class declares 14 private variables with their own set and get methods. It also contains two constructors with different arguments of different types, ‘Display()’ method and ‘ToString()’ method.

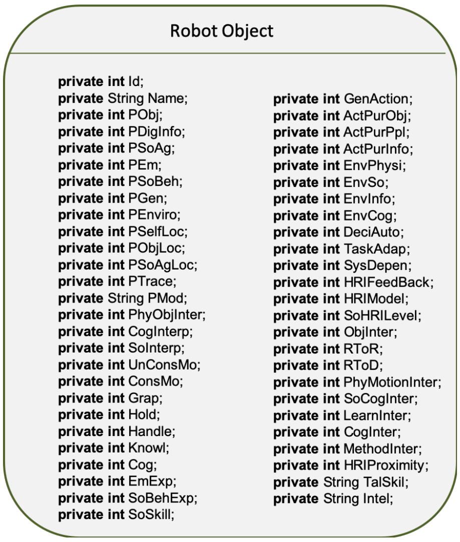

Figure 6.16: The variables of the robot object, created by the blueprint class ‘Robots.java’.

The third package is the View. This package is the default package and has one class called ‘RobotCapabilities.java’. This class contains the main method as the starting point for executing the program. The main method calls a splash screen with the ToRCH logo, after which it calls the initialization method that displays a graphical user interface (GUI) form.

The GUI form is presented with different sections, each containing specific GUI components. The GUI components present the main capabilities of ToRCH in sections and are listed along with their possible values to be requested and chosen. Each main capability is presented in a separate division in the form. The capabilities are listed and grouped according to the ToRCH framework.

The GUI forms contain two main parts. The first is for the user to set the score of the requested capabilities and the second displays the robot(s) returned from the search. The input part of the form includes several sections, each presenting one main capability: perception, interpretation, task performance, action and envisioning, interaction, skills and intelligence. Each capability then contains a list of sub-capabilities with a drop-down list covering all possible values for the capabilities. The GUI form includes more than 40 drop-down lists, each capturing one of the robot capabilities values as a user request. The form distributes the drop-down lists according to the main layers of the ToRCH hierarchy. The form includes all of the ToRCH capabilities in drop-down lists, check-boxes and J-lists to enable users to request every capability of the technical layer of ToRCH.

The user selects a value from each drop-down list for the capabilities to create a request object. The requested object is then matched against each robot profile in the linked list captured from the database. This matching process is performed as a search on the linked list. The search filters robots according to the specified capabilities levels requested, and only robots with matching capability scores are returned as results to the user. The outcome of the query is displayed in the output part of the form and presented as robots that fulfill the user's request. The GUI is presented in Figure 6.17. The full program is available online in Linjawi and Moore (2019b).

6.4.4 Outcome of the ToRCH development

The application provides all of the ToRCH listed capabilities in one form. It illustrates the potential diversity of capabilities possible and it provides a practical outline of the robot in response to a user's request. The application mirrors the ToRCH capabilities and their scoring in the datasets object presentation and the GUI. The application enables the ToRCH dimensions to be searched, allowing the user to choose the capabilities values that he or she requires; in response to a user's choice of capabilities values, the application queries the database for a robot with matching capability scores. The application also provides a feature that enables the user to add new capabilities to the database. The application demonstrates the ToRCH dimensions as a valid framework for bridging between the capability and application domains.

The next chapter will evaluate ToRCH and the generated RCPs.

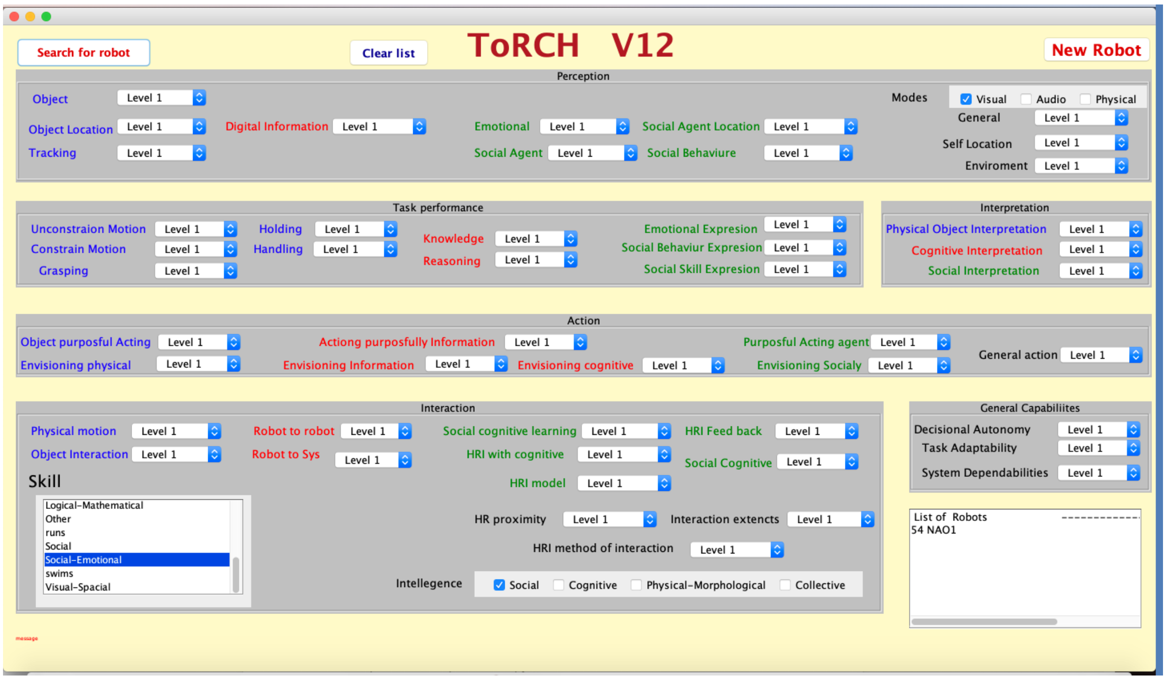

Figure 6.17: The GUI form in the application contains the technical layer of the ToRCH framework. It presents the different capabilities within several sections, as presented in the ToRCH hierarchy. Each section contains lists of related capabilities with drop-down boxes with relevant scores. The user chooses the capabilities scores that are needed within the robot and the application would display the most match robot. The form contains a display area to presents the corresponding robot for the requested scores. This figure gives a run-time execution example of the application.

✻ Chapter 6: Summary

In this chapter, we have presented the ToRCH framework as a tool that can capture different capabilities of any robot and present them in an RCP. The RCP consists of the scores of the capabilities, which it groups into sections and subsections. It also demonstrates the capabilities of different types of robots in a hierarchical classification that can be used to describe the robots more specifically, and thus, of use to users and developers. The RCP and ARP offer an accurate description that helps to optimize robot capabilities against application requirements moreover, it can help robot users and application developers select the most appropriate capability levels that are needed in their robots and that match their applications. This relation was illustrated in the proof-of-concept, automated version of ToRCH.

Chapter 7

Evaluation of ToRCH

7.1 Introduction

The ToRCH framework was evaluated in several phases. Each phase assessed a specific aspect. The first phase evaluated the structure of ToRCH by meeting with roboticists to review its concepts. The second phase evaluated ToRCH as a tool and was performed by conducting a series of questionnaires to test specific features. The third phase evaluated the outcome of ToRCH the Robot Classification Profile [RCP].

7.2 First evaluation phase: Assessing the ToRCH structure

The first evaluation was to assess the structure in general and to check the quality of the ToRCH hierarchy as a whole through an iterative feedback process and consultation. It evaluated the allocation of ToRCH layers, sections and capabilities. This evaluation was intended to check whether the proposed conceptual framework as a whole met its classification specification and could fulfill its intended purpose. This phase checked where capabilities were positioned within the framework, and whether they were correctly allocated as sub- or super-capabilities, that is, whether they were nested under capabilities or as containing capabilities (i.e. correct place in the hierarchy).

The assessment was performed by meetings with roboticists from different fields. There were 29 roboticists, from 13 different labs, drawn from both academia and industry. A structured questionnaire was used for each interview. Each meeting lasted between 30 minutes and an hour and covered: what dimensions were needed within the structure, how they were presented as layers, robot characteristics required for the classifications, groupings within each layer and their location, and relations between layers and their concepts.

These personal meetings were part of an iterative process where every amendment was then assessed in the next meeting. The methodology used was:

- Heuristic evaluation: principles and guidelines were evaluated by experts. Roboticists evaluated the ToRCH conceptual framework with their expertise.
- Cognitive walk-through: robotics experts performed a cognitive walk-through of the ToRCH framework by analyzing the different robot scenarios within the questionnaire, to identify missing layers, sections capabilities/sub-capabilities and characteristics. Also, a capabilities technical rehearsal was performed to check whether the capabilities were located within the correct areas of the framework. Walking through the process confirmed whether each capability grouping and breakdown was connected correctly and fitted with the framework so that the capabilities were presented as cohesive. For example, it checked whether perceptions and actions in the second layer could be assumed from the features levels, or that they could lead to interaction and intelligence capabilities. However, linking between the capabilities from one layer to another was not covered in the study.

The outcome of these consultations formed the major structure of the ToRCH framework. Although the first and third layers of the framework are important in describing the robot, for reasons of time and resources, the research concentrated on the second layer. Therefore, this evaluation focused on the second layer as it captures the capabilities, which are the core of this research. A copy of the questionnaire is in Appendix B.

7.3 Second evaluation phase: Assessing ToRCH as a tool

Evaluating ToRCH as a measurement required defining its accuracy, richness, precision and effectiveness as a tool. These qualities were measured using of a series of structured questionnaires. The evaluation focused on using the ToRCH framework to capture the capabilities of robots and distinguish them from each other. This evaluation phase is performed in several subsections, as each subsection assesses a particular property that is needed within the ToRCH description. Therefore, these properties, (such as accuracy, richness, precision and effectiveness), are evaluated if they could be identified as ToRCH features or some of its qualities. This is done through defining accuracy, richness, precision and effectiveness to be available with in ToRCH and therefore to be extended as one of the ToRCH specifications.

- To evaluate the accuracy of the framework, participants defined the capabilities of a robot and inter rater liability evaluation was performed, between the participants' scores, to check the degree of agreement between their answer and the model answer, as presented in section 7.3.1.
- To evaluate the richness, an evaluation checked whether the capabilities captured by ToRCH described a robot performance clearly, and whether ToRCH contained an adequate list of capabilities. This was done by comparing between capabilities captured by ToRCH and those captured by MAR, to define the capabilities of a robot, as presented in 7.3.2.
- To evaluate the precision, participants scored the same capabilities for a specific robot, as presented in section 7.3.3. The participants were the developers of the robot and the capabilities scores were analyzed for an inter-rater reliability to define the degree of agreement between them.
- To evaluate the effectiveness, participants scored the capabilities of two different projects but developed in the same robot. Each project contained a set of capabilities that had been developed for its research requirements, as presented in section 7.3.4. This was intended to demonstrate ToRCH's ability to extract different robot performances developed within the same robot.

7.3.1 Evaluating the accuracy of ToRCH

Abstract

To evaluate the accuracy of the ToRCH framework, participants defined the capabilities of a robot by scoring them. The evaluation checked if all participants have defined the same capability for the robot scenarios by having slimier scores. Using the ToRCH, lists of the capabilities scores are recorded and assessed to define the accuracy of the framework. Therefore, the experiment performs an inter-rater liability evaluation to check the degree of agreement between the participant's answers and the model answer.

Introduction

The second phase in evaluating ToRCH was performed by different roboticists to check the framework's accuracy in capturing capabilities. The NAO robot, a commonly used commercial and social robot, was selected as it can be developed to present a wide range of capabilities. NAO robots (Gouaillier et al., 2009) can be developed and programmed to demonstrate different capabilities for specific application requirements. They can be delivered in different forms, such as just a torso or with a complete body, and are flexible and easily programmable. Hence, the NAO clearly demonstrates different application requirements. To assess NAO, ToRCH needs to be comprehensive so that it can capture the robot's full range of capabilities. Determining which capabilities a NAO has been developed to perform is important for distinguishing any one robot from another and for identifying the robot according to its execution ability. Therefore, this evaluation focused on distinguishing between two sets of performed capabilities that had been developed in the NAO robot. This evaluation presented two different demonstrations of the NAO robot and requested participants to identify and differentiate between the capabilities within the two performances. A questionnaire was conducted to obtain capabilities presented in two scenarios involving the robot. The questionnaire determined the level of agreement between roboticists in using ToRCH to describe a robot's capabilities, through the generated RCP.

Method

The ToRCH method scores robot capabilities and generates the RCP. The RCP can be used to compare the capabilities of different NAO robots. However, the ToRCH structure could be used for any robot since it creates for each robot a personalized profile. Therefore, assessing which capabilities a NAO robot has been programmed to perform is important so that it can be distinguished from other NAOs. The extent to which ToRCH allowed users to do this was evaluated through a questionnaire. It was assumed that the RCP would distinguish different capabilities within two robots' performances. Therefore, the questionnaire was conducted to evaluate the accuracy of the RCP in defining capabilities and differentiating between the NAO robots. An online questionnaire was provided to participants, and this investigated how they would use ToRCH to score robot capabilities. The questionnaire covered two different scenarios involving NAO robots, with the extracted RCP listing the scores in a structured numeric format, which would be available for further analysis. The participants rated the ToRCH capabilities as these related to the robot and scenario. The questionnaire described the two scenarios textually, indicating different capabilities for each, and participants were asked to select the levels of the capabilities indicated. A copy of the questionnaire is in Appendix C. The collected data was analyzed and calculated to compare the agreement rate between participant responses and the model scores. Both scenarios are presented in the following section, and the RCPs, shown as the model answers for scenario 1 (S1) and scenario 2 (S2), are presented in Table 7.1. The table presents the categories of capabilities, the specified capabilities, and the model scores.

Scenario 1 (S1) The NAO robot is programmed to recognize a predefined ball on the floor among other unknown objects. The robot moves towards the ball and says to the person in the room, "I found the missing ball." If the person smiles, the robot maps the facial expression to a predefined model of 'happy' and says, "I found your ball." If the person does not smile, the robot says, "I will dance for you."

Scenario 2 (S2) The NAO robot is programmed to recognize a predefined ball on the floor among other unknown objects. The robot recognizes an individual among the people in the room because it has a picture of that person in its database. It moves towards that individual, laughs and says, "I found you and I found your ball."

To generate the RCP, participants analyzed the textual description of the scenario, by checking each capability against the described robot performance. However, not all capabilities from ToRCH were included in the questionnaire, nor did the scenarios cover all capabilities of the NAO. The scenarios were designed to demonstrate some differences in capability. Scenario 1 did not include 'object interpretation', 'agent location perception' and 'emotion expression' capabilities, as they were not relevant. Likewise, scenario 2 did not include 'object interpretation', 'emotion perception' and 'object location perception' for the same reason. The scenario demonstrated the levels of the capability as it mentions which capability maturity is performed in the scenario.

For example, in emotion perception,

- Level 0 - indicates no emotion perception.
- Level 1 - indicates emotion detection as the robot scenes an aspect of emotion.
- Level 2 - indicates a single instance of emotion recognition as an emotion model created from scene data can be matched against a specific model saved in the robot.
- Level 3 - indicates multiple instances of emotion recognition the emotion model created from scene data can be matched against several models saved in the robot.

For scenario 1, the robot can define happiness in the person but not other emotions, and without matching it with a predefined model, so it was assigned as level 1 in the model answer. For scenario 2, the robot is not checking the personal emotions in the person, so it was assigned as level 0 in the model answer. The levels demonstrated in the scenarios were also primarily designed to indicate whether the capability was evident or not, and so they were demonstrated at low levels. However, the scenarios and the model scores were designed and calculated by the author.

Each capability was scored to determine the technological maturity level exhibited in the scenario. The scoring gave a numeric score that maps to the capability level; for example, a robot with a zero score for a capability would achieve a score of zero, level one, would score one, level three would score three, and so forth. The final step was to sum up the scores for all the capabilities in each scenario to obtain a robot capability profile (RCP). Hence, the RCP gives a numerical display of the robot's performance. For the RCPs created from this questionnaire, only 11 capabilities relevant to the scenarios were examined. These capabilities are presented in Table 7.2. Capabilities not demonstrated in the scenarios were not included in the questionnaire due to time constraints and to ensure ease of use. The scenarios each sampled 11 capabilities, so together they covered 22 capability scores. A total of six questions covered capabilities included in both the MAR and ToRCH, and a further five questions related to capabilities only available in ToRCH. To avoiding leading the participants, the language used to describe the scenarios did not directly map to the capabilities or levels being measured. The capabilities assessed are presented in Table 7.1. Six capabilities are covered in both scenarios, and the other five capabilities were covered in one scenario but not both.

Participants Twelve participants researchers graduate teaching assistants for robotic labs in Sheffield University were recruited through the university email. The online recruitment ensured that all participants had programmed the NAO robot previously and had some experience using it. Participants were provided with an online link which contained a description of two scenarios outlining robot capabilities, and several questions. The participants were asked to fill in the the questionnaire according to the scenario described. All participants were self-defined as NAO robot programmers, thus according to their personal knowledge of the NAO robot and their interpretation of the scenarios, they selected the capability levels that best matched the described performance in the written scenarios. The studies were approved by the University of Sheffield Research Ethics Committee and informed consent was given by all participants. No personal or demographic information was recorded. For ease of collecting data the questionnaires were online. Leading questions were avoided and

Table 7.1: ToRCH categories, capabilities and levels according to the model answers (presented numerically) for the two scenarios. The scores represent the capability maturity levels demonstrated in the NAO performance. Any capability demonstrated within the scenario scores more than zero. The scoring depends on the capabilities that need to be programmed within the NAO robot to perform the scenario.

Category	Capabilities	S1 score	S2 score
Physical perception	Object perception	3	3
Physical perception	Self-location perception	1	1
Social perception	Agent perception	2	4
Social perception	Emotion perception	1	0
Physical perception	Object location perception	1	0
Physical perception	Agent location perception	0	1
Physical interpretation	Object interpretation	0	0
Physical task	Unconstrained motion	1	1
Social task	Emotion expression	0	1
Social interaction	Human interaction modality	2	2
Social interaction	Interaction level of extent	1	1

Table 7.2: The table contains the questions covering capabilities in both scenarios, whether the capability is available in MAR, ToRCH or both frameworks. It also shows where capabilities were used and in which scenario. The final column presents the range of capability levels, and each was distinguished according to the parameters or variables related to the maturity of the capability.

Capability questions	**Availability**	**Usage in the scenario**	**Levels**
Object perception	MAR/ToRCH	recognize the ball in S1 & S2	0-12
Self location perception	MAR/ToRCH	moving towards something S1& S2	0-7
Object interpretation	MAR/ToRCH	not available	0-9
Unconstrained motion	MAR/ToRCH	movement capabilities in S1 & S2	0-7
Interaction modality	MAR/ToRCH	interaction type in S1 & S2	0-6
Interaction levels of extent	MAR/ToRCH	repeated interaction in S1 & S2	0-7
Agent perception	ToRCH	recognize a person in S1 & S2	0-8
Emotion perception	ToRCH	recognize a smile/happiness in S1	0-4
Object location perception	ToRCH	moving towards objects in S1	0-7
Agent location perception	ToRCH	moving towards agent in S2	0-7
Emotion expression	ToRCH	expressing happy emotion in S2	0-3

no scores were given to the participants to avoid influencing their answers. No data was shared outside of the research team.

Procedure As the twelve participants assessed this evaluation, from their personal knowledge of the NAO robot and their interpretation of the scenarios, they selected a level for the capabilities that best matched the described performance. The list of capabilities and their available levels are presented in Table 7.2. Participants had to score a level for each of the capabilities and could leave a comment explaining their choice of level. The data from the questionnaires were downloaded and extracted for analysis. Their scores were compared with the model scores for the scenarios to assess the accuracy of ToRCH in capturing a range of capabilities, as shown in Table 7.1. An inter-rater reliability analysis was performed on the data (Grayson and Rust, 2001), calculated using Cohen's kappa measure. This indicates the extent to which the selection of levels by participants matched with the model scores. This is shown in Table 7.3.

In order to calculate Cohen's kappa measure, the capability model scores were defined, and the agreement between the model score and the rater was calculated for both scenarios. A calculated mean score for each scenario represented the extent of agreement between the participants and the model scores, as shown in Table 7.3. A guideline for mean kappa suggests that a value of 0 indicates no agreement between the model and participants, values below 0.40 are poor agreement, values between 0.40 and 0.75 indicate a fair to good level of agreement, and values over 0.75 indicate excellent agreement between participants and model score (Landis and Koch, 1977).

Results

Each of the 12 participants performed 22 assessments, making a total of 264 capability assessments. The correct assessment of a capability occurred 231 times and incorrect assessments 33 times. Therefore, the accuracy of these assessments was 88%. The majority of the incorrect assessments were of interpretive capabilities, which shows that interpretive capabilities require a more detailed description in the scenarios to be correctly assessed. The survey, therefore, demonstrates the accuracy of ToRCH for assessing capabilities, indicating that it is a practical tool for classifying capabilities in robots.

There were some differences between the levels assigned to capabilities by participants and the model answer, which are shown in Figure 7.1. Cohen's kappa scores were obtained, and for scenario 1 the mean kappa was 0.46, which indicates a moderate agreement. For

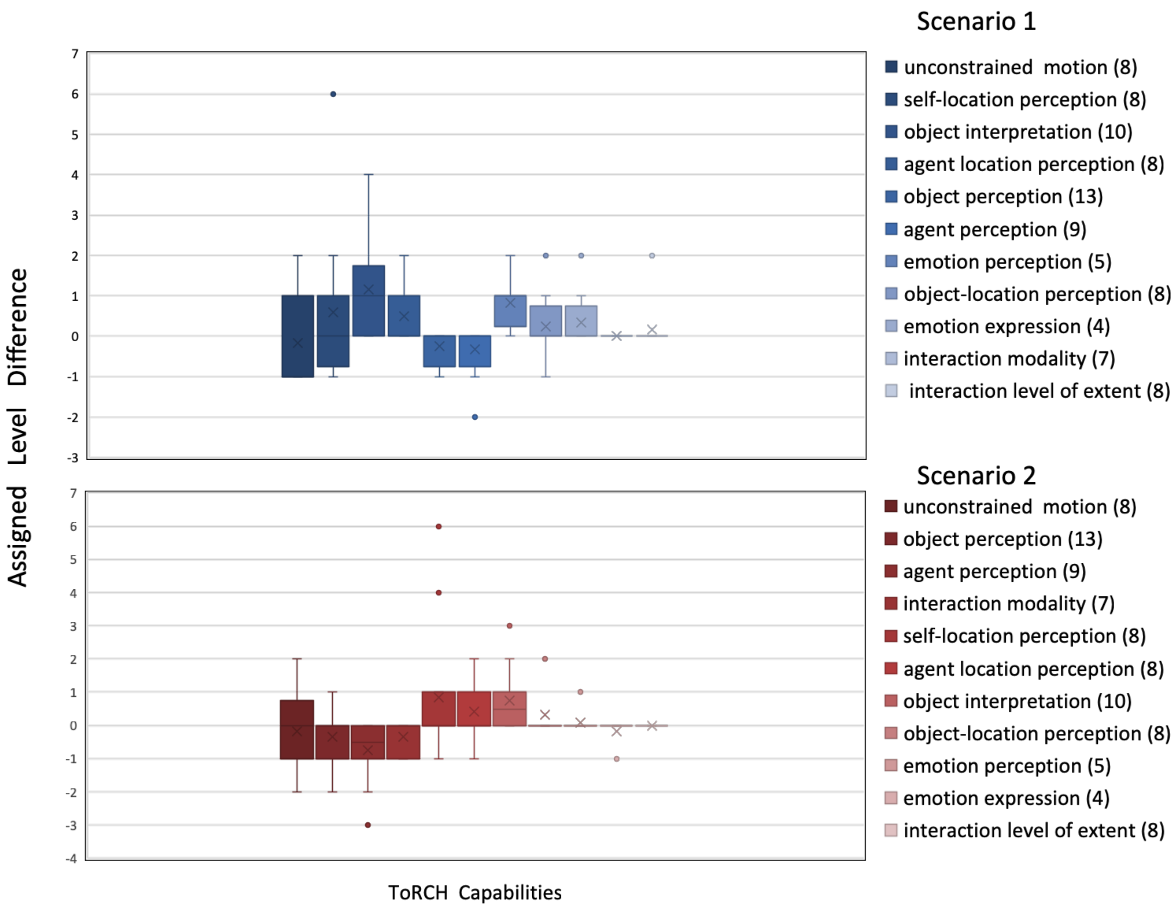

Figure 7.1: This figure demonstrates the difference in level scoring from the model answer for each capability for scenarios 1 and 2. Zero difference indicates the participant gave the same score as the model answer. The scoring difference from the model answers is presented for each capability in the inter-quartile range with a median. The number in the brackets after the capability name indicates the total number of levels available for the capability listed.

scenario 2, the mean kappa was 0.55, again showing moderate agreement. The overall mean kappa was 0.5, indicating moderate agreement for users of ToRCH according to the Cohen's kappa classifications. Table 7.3 and the Figure 7.2 show the Cohen's kappa results.

Discussion

It should be noted that the Cohen's kappa test indicates only whether a participant agrees or disagrees with the model score, not the degree of agreement. Thus, for example, in scenario 2, when assessing 'interaction modality', which has seven levels, all 12 raters agreed the capability existed in the robot, but only eight agreed with the model score. However, the four

Table 7.3: Cohen's kappa scores for the survey participants, for each scenario, and the mean for both scenarios.

Participant	**CK scores S1**	**CK scores S2**	**mean S1 & S2**
1	0.22	0.63	0.43
2	0.19	0.12	0.16
3	0.61	0.52	0.57
4	0.48	0.50	0.49
5	0.31	0.65	0.48
6	0.27	0.14	0.20
7	0.06	0.28	0.17
8	0.29	0.33	0.31
9	0.75	0.75	0.75
10	0.61	0.88	0.74
11	0.87	0.87	0.87
12	0.87	0.87	0.87
mean	0.46	0.55	0.5

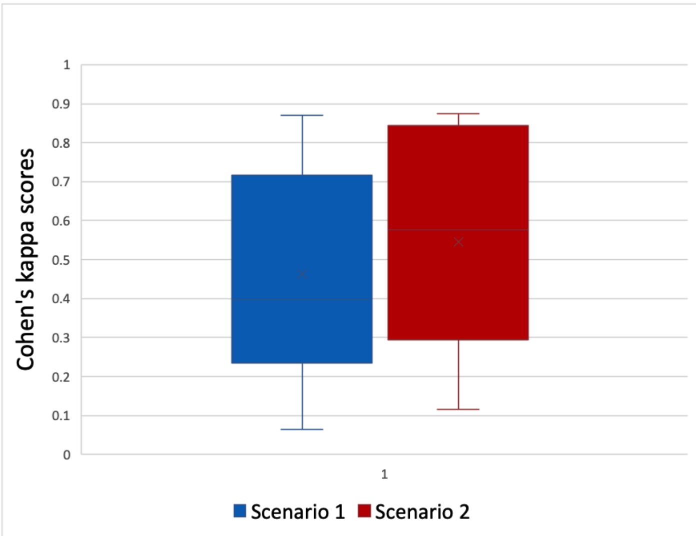

Figure 7.2: This figure shows the distribution of the Cohen's kappa scores for both scenarios.

who did not match the model score were only one level out. This can be seen in Figure 7.1; where participants gave the same score as the model, this is indicated by 0. Any number above 0 shows where participants have ranked the capabilities higher than the model answer, and a number below 0 indicates an underscoring and underestimating of capabilities. The figure also presents outlying scores, that is, those outside the average range. Such outliers were then excluded from further analysis.

Analyzing the mean kappa agreement scores for both scenarios indicates a slight increase in agreement in scenario 2. This could imply that as participants became more comfortable in assessing capabilities and their levels, they were able to increase their accuracy due to practice. Thus, it may be that training or instruction in the ToRCH method is needed for initial use.

The analysis of the data revealed that 'self location perception' and 'unconstrained motion' demonstrated the greatest disparity from the model answer. Raters possibly used their knowledge of a specific version of the NAO robot, rather than the version used when rating the scenarios and generating the model answer. For example, an older version of the NAO robot could have a lower level of unconstrained motion, (that is, how the robot moves within the environment), such as predefined open loop (level 1), rather than open path motion (level 3). This may have created the difference in the scoring of this capability.

In contrast, other capabilities more closely matched the model answer. For 'interaction level of extent', all 12 participants matched the model answer for both scenarios, agreeing on level 1 (temporally restricted interaction). This level indicates the time required for an interaction between the robot and the user, which is not necessarily dependent on the robot. Hence, assessing the levels of some capabilities depends on the context in which that capability is presented.

Further analysis was performed on the data in Figure 7.1. This analysis generated the number of under- and overestimated capability levels for both scenarios, as shown in Table 7.4. Zero (0) indicates that the participant score matched the model answer. It can be seen that 'object interpretation' was overestimated by raters, implying that this capability and its levels were not clear to participants.

The capabilities 'object perception', 'agent perception', 'self-location perception' and 'unconstrained motion' had the greatest discrepancy in scores. These capabilities depend on robot features more than the scenario described. The features can differ even between robots of the same manufacturer, depending on how they have been programmed and developed. The most recent versions are more sophisticated, which could potentially lead to scoring higher levels of the capabilities.

The data were analyzed further to determine the level of agreement between the raters over the two scenarios. Figure 7.3 presents the distribution of participant scores for the capabilities in both scenarios. The scores demonstrate that the participants identified differences between the capabilities over both scenarios. The participants selected higher scores for 'social agent perception' and perceived no capabilities for the 'interaction level of extent' and 'self location

Table 7.4: Each capability is presented with the number of its available levels, and the number of participants who underestimated or overestimated capability levels for scenario 1 and scenario 2.

		Scenario1		Scenario2	
Capability	**Levels**	**Under**	**Over**	**Under**	**Over**
Object perception	13	1	0	2	1
Self-location perception	8	1	2	1	1
Object interpretation	10	0	4	0	2
Unconstrained motion	8	1	2	2	2
Interaction modality	7	0	0	1	0
Interaction level of extent	8	0	0	0	0
Agent perception	9	1	0	2	0
Emotion perception	5	0	2	0	0
Object location perception	8	1	1	0	0
Agent location perception	8	0	2	1	2
Emotion expression	4	0	1	0	0

perception'. The scores show that participant ratings were different enough to indicate a significant difference between the capabilities demonstrated in the two scenarios.

Limitations

In the course of this study, several limitations are acknowledged and presented: The limited number of participants (12) available for the survey. Participants were required to be expert in programming a NAO robot. The limited number of participants can offer only a preliminary evaluation of the ToRCH framework. The presentation of the scenarios in a textual format created some uncertainty in defining the levels of capabilities. The textual format was chosen to facilitate the wide distribution of an online survey. However, where capabilities such as dexterity or autonomy are assessed, observation of actions of the robot, especially its programmable method within the scenario would help in the evaluation of levels. The capabilities of the NAO robot demonstrated in this questionnaire may have differed from the model the participants had knowledge of, which again could influence the scoring of capabilities. The use of two similar scenarios presented in the same order for all participants could have affected their scoring choices, which is suggested by the greater match of model scores in the second scenario. The terminology used to present capability levels had some ambiguity that may have resulted in discrepancy in scoring levels. The experience and knowledge of participants varied, which would influence their scores, potentially adding to the disparity of results. In answering an online survey, participants were required to

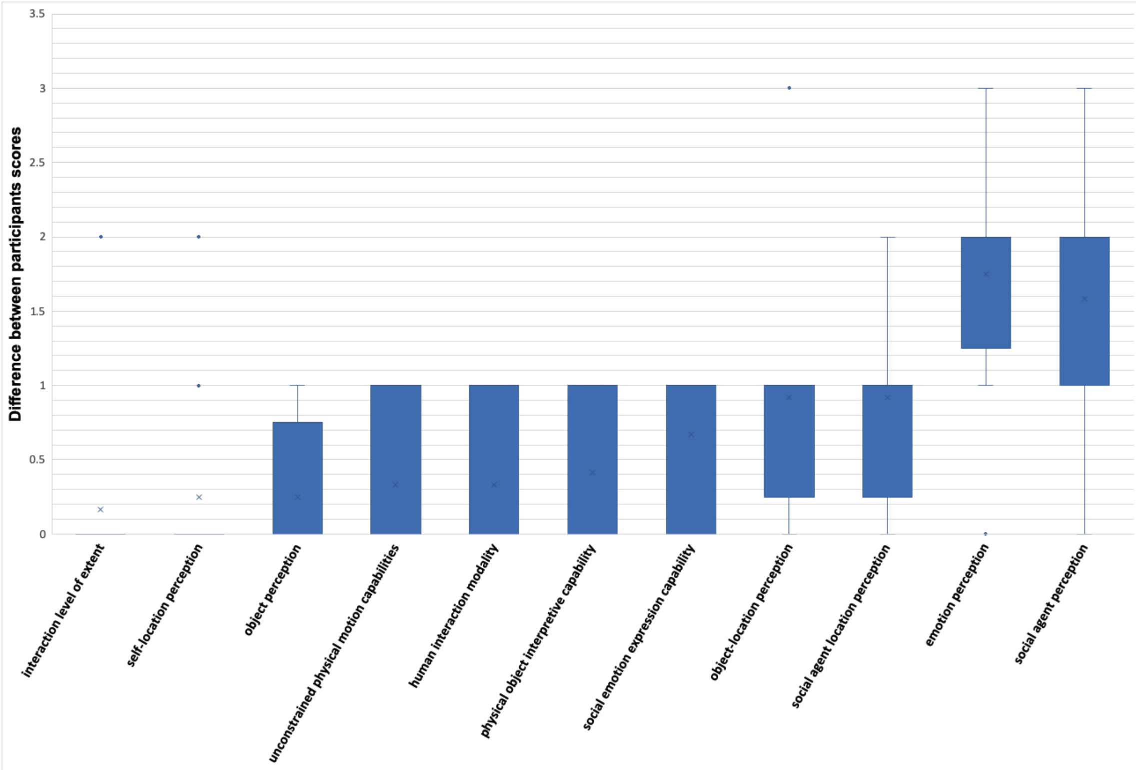

Figure 7.3: This figure shows the difference between scoring the capabilities for both scenarios. This implies that participants could identify the differences in the NAO robot capabilities in both scenarios.

scroll up and down to move between the scenario and subsequent questions, which added to the difficulty of clearly scoring the capabilities. Some capabilities were felt to need more levels to describe their technological maturity. To overcome this, some participants under- or over-scored a capability.

7.3.2 Evaluating the richness of ToRCH

Abstract

To evaluate the richness of the ToRCH framework, an evaluation was performed to check if the capabilities captured by ToRCH describe robot performance clearly. The evaluation also assessed whether the ToRCH contained an adequate list of capabilities. This was done by comparing the capabilities captured by ToRCH and distinguish them with the capabilities captured by MAR. The assessment checked which of the two systems would define the capabilities of a robot comprehensibly but clearly.

Introduction

This aspect of the evaluation process was intended to assess the comprehensiveness of the capabilities and their levels as extracted via ToRCH. This was done by comparing the capabilities captured by MAR and ToRCH. The hypothesis was that when presenting the model answer capability profiles of the two scenarios, the capabilities included in MAR would not change levels, but those in ToRCH would demonstrate a difference in levels as the ToRCH framework captures more capabilities.

Method

This evaluation required two types of profiles, one produced by ToRCH and the other one produced by MAR. These profiles were compared to define which type of profile is most suitable in identifying robot capabilities. This experiment was designed to discover the profile with the most plentiful quantities of capabilities to describe robot performance. The more capabilities descriptions a profile has, the better the profile is at gathering, expressing and identifying the capabilities of the robots. This evaluation used the previous experimental settings as it exploited the capabilities model answer of the two scenarios and investigated their profiles. The scenarios were designed to allow comparison between the capability profiles generated by MAR and ToRCH, and to demonstrate which type of profile is more suitable and accurate at capturing robot capabilities in the scenarios. The scenarios were intended to demonstrate the most common capabilities presented in any social robot, such as the NAO robot, and the questionnaire included only the list of capabilities available in the scenarios for both MAR and ToRCH.

Participants There were no direct participants engaged in this evaluation. Direct comparison between the scores of the ToRCH list of capabilities and the MAR list of capabilities was performed. Both scores were used to compare and differentiate between the capabilities of the scenarios and define which list would show a difference in the capabilities scores from both scenarios.

Materials and procedures This evaluation generated robot capability profiles (RCP), for scenario 1 and scenario 2, by using: (1) a MAR type of profile and scoring the capabilities listed in MAR and generating the profiles according to the model answers of scenario 1 and scenario 2. The evaluation also generated a robot capability profile by using (2) a ToRCH type of profile and scoring the capabilities listed in ToRCH and generating the profiles according to the model answers of scenario 1 and scenario 2. The two sets of profile were compared to define the similarities or dissimilarities between them.

Result

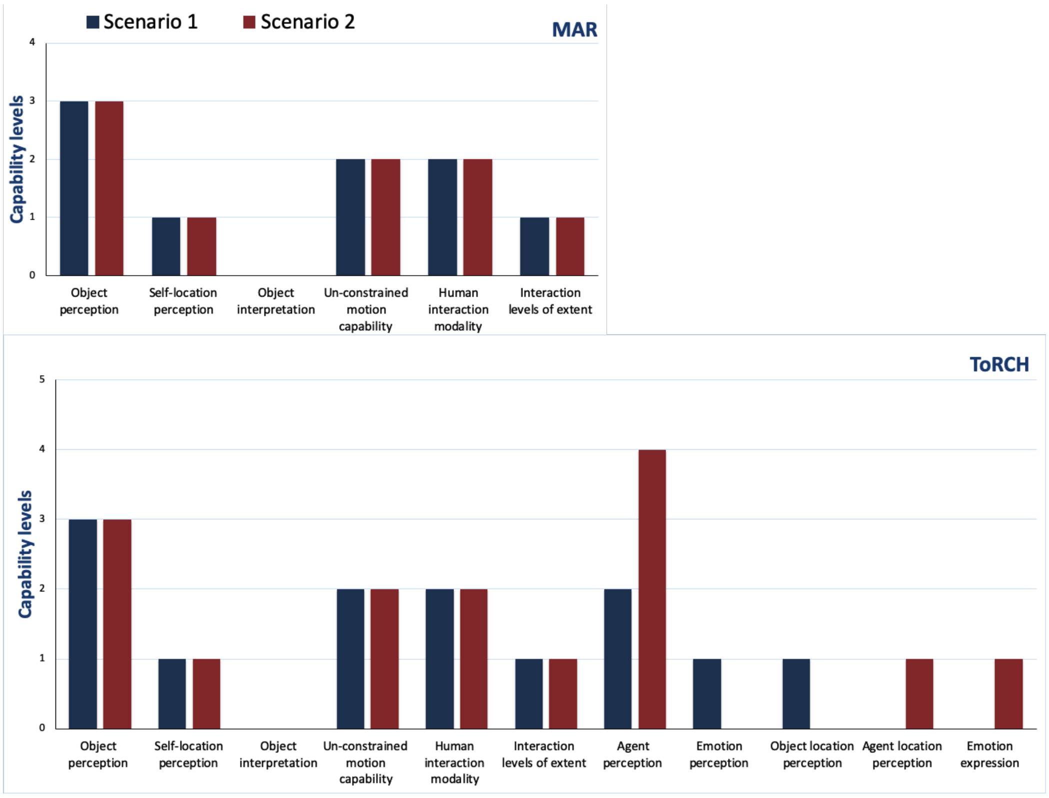

Figure 7.4: The upper chart shows no difference between scenarios 1 and 2 using the MAR list of capabilities. The lower chart presents the difference in the NAO capabilities for both scenarios using ToRCH. The presented values are the model scores for the scenarios.

The model scores shown in Table 7.1 are presented in bar graphs. The MAR and ToRCH RCPs are presented in separate graphs. Figure 7.4 demonstrates the model scores of both scenarios for the RCP of MAR (the upper graph) and the RCP of ToRCH (the lower graph). Each type of RCP includes its specified set of capabilities extracted from the questionnaire.

A comparison between the two shows that ToRCH (the lower graph) can capture the differences in capabilities demonstrated in the NAO scenarios, whereas MAR (the upper graph) does not. This can be seen in the last four bars in the ToRCH chart (the lower graph). For example, ToRCH captures 'emotion perception' and 'object location perception' capabilities for scenario 1 but not for scenario 2. Also 'agent location perception' and 'emotion expression' capabilities were captured for scenario 2 but not scenario 1. Comparing the MAR and ToRCH RCPs demonstrates that the RCP generated by ToRCH is more comprehensive, as it includes more capabilities.

Also, ToRCH enables the capture of a greater number of capabilities for both scenarios due to the comprehensive levels of details in its framework. For example, it demonstrates the difference in 'agent perception' capabilities between the two scenarios as each scenario presents the capability with different levels.

Conclusion

This evaluation presented the RCPs extracted by MAR and ToRCH, demonstrating that the RCP using MAR capabilities does not show any difference between the two scenarios whereas the RCP generated by ToRCH does. This assessment also shows that there is a significant correlation between the number of scored capabilities in the RCP and the clarity of the capabilities that the robot is executing, and consequently, it illustrates how well the RCP can describe a robot. The more capabilities that were scored, the richer the RCP and the more comprehensive the description is captured. This analysis confirms that an inclusive list of capabilities presented in ToRCH can capture the robot performance in much greater detail than MAR. This outcome allows us to propose ToRCH as a comprehensive and practical tool for describing a robot's capabilities.

7.3.3 Evaluating the precision of ToRCH

Abstract

To evaluate the precision of the ToRCH framework, participants as developers of a telepresent robot, scored the capabilities that they had developed in their robot. The capabilities scores were analyzed for inter-rater reliability to define the degree of agreement between the scores.

Introduction

This evaluation focused on whether participants would score the capabilities of a single robot at the same level. It evaluated the precision of ToRCH by examining the repeatability of participant scores. For this, two raters participated by scoring the capabilities of a single robot. Both participants had collaborated in the development of the robot and its capabilities. It can be speculated that a comparison between their scores, for each of the ToRCH capabilities, should reveal the scores to be identical. There is an expectation that there will also be some relation between the scores of some capabilities and specific robot features, for example between mobility and physical wheels in a robot.

Method and procedures

To evaluate the precision of ToRCH in capturing capabilities, two participants scored the same capabilities of a robot and their scores were evaluated and compared. The participants needed to be experienced with the robot to identify its capabilities. The robot Pepper was selected as its two developers from the Sheffield Robotics Institute volunteered to participate in the assessment. Both contributed to developing the telepresence performance of the robot for a specific project, referred as [Proj1]. After giving informed consent, the participants filled in a hard-copy questionnaire that captured the levels for each of the capabilities in Pepper. Based on their knowledge of Pepper and capabilities developed for telepresence, the participants selected the capabilities and levels that best matched the robot. To assess the framework, the questionnaire included all capabilities available in ToRCH. The questions are listed in appendix A.

Participants Researchers from Sheffield Robotics Lab were asked to volunteer in this study. They were asked as they were developing the capability of a single robot. Two participants were provided with a hard copy questionnaire to fill in. These studies were approved by the University of Sheffield Research Ethics Committee and informed consent was given to the participants. No personal or demographic information was recorded. Leading questions were avoided and no scores were given to the participants to avoid influencing their answers. The questionnaires were carried out face-to-face as a recording was made but no identification or personal information was recorded. The collected data was kept in accordance with University of Sheffield best practice, on the password-protected university servers. Hard copy data and the audio recordings were stored and will be destroyed in accordance with the University of Sheffield guidelines. No data was shared outside of the research team.

Result

The scores of both participants are presented in Figure 7.5 and in several tables. Each table contains a list of capabilities belonging to a specific section of the ToRCH structure, the range of levels for those capabilities and both participants' scores. The first table (Table 7.5) includes the perception and interpretation capabilities. The second table (Table 7.6) presents the task performance capabilities. The third table (Table 7.7) contains the purposefulness and envisioning capabilities. Finally, the fourth table (Table 7.8) presents the interaction capabilities.

Table 7.5: Perception and interpretation capabilities with their types, range of levels and participant scores for [P1] and [P2], where P1 refers to first participant and P2 refers to the second participant.

Description	**Perception/interpretation capabilities**	**Range**	**P1**	**P2**
general	general perception	0-9	1	1
general	environment scene perception	0-6	0	0
general	tracking perception	0-6	0	0
physical	object perception	0-13	1	1
physical	object location	0-8	0	0
physical	self location perception	0-7	1	1
physical	social agent location	0-8	0	0
cognitive	signals and digital perception	0-5	1	1
social	social agent perception	0-9	0	0
social	emotion perception	0-6	0	0
social	social behavior perception	0-6	0	0
physical	object interpretation	0-9	0	0
cognitive	signals/digital interpretation	0-6	2	2
social	human agent interpretation	0-6	0	0

Table 7.6: Task capabilities, their types, range of levels and participant scores for participant [P1] and participant [P2] are listed. The manipulation capabilities were scored differently by the participants. The first participant [P1] captured the default capabilities setting of the Pepper robot and scored them accordingly. But the second participant scored zero for these capabilities as they were not performed and executed for the tele-present project [Proj1].

Description	**Task capabilities**	**Range**	**P1**	**P2**
physical	motion: unconstrained	0-8	2	2
physical	motion: constrained	0-6	0	0
physical	manipulation: grasping	0-9	1	0
physical	manipulation: holding	0-6	2	0
physical	manipulation: handling	0-10	1	0
cognitive	signal/data/information: learning	0-16	0	0
cognitive	signal/data/information: reasoning	0-9	0	0
social	emotion expression	0-9	0	0
social	social behavior	0-5	0	0
social	social skills	0-5	0	0

Table 7.7: Acting purposely and envisioning capabilities with their types, range of levels and participant scores for participant [P1] and participant [P2]. The capability of acting purposefully towards objects was scored differently by participant [P1] than participant [P2]. As participant [P1] scored the default existence of the capability within the robot and participant [P2] scored the capability to zero indicating that the capability was not used in the project [Proj1].

Description	Purposefulness/envisioning capabilities	Range	P1	P2
general	general acting	0-3	2	2
physical	acting purposely towards objects	0-9	1	0
cognitive	acting purposely towards digital data	0-3	0	0
social	acting purposely towards human	0-8	0	0
physical	objects envisioning	0-6	0	0
cognitive	signal/digital/info envisioning	0-6	0	0
cognitive	cognitive envisioning	0-6	0	0
social	social/human envisioning	0-6	0	0
general	decisional autonomy	0-12	0	0
general	task adaptability	0-5	0	0
general	system dependability	0-8	2	2

Table 7.8: Interaction capabilities with their types, range of levels and participant scores for P1 and P2.

Description	Interaction capabilities	Range	P1	P2
general	methods of interaction	0-9	1	1
physical	physical motion interaction	0-5	4	4
physical/cognitive	object interaction	0-10	0	0
cognitive	robot-to-robot interaction	0-10	0	0
cognitive	robot-to-system/device interaction	0-10	0	0
social	interaction levels of extent	0-8	0	0
social	human-robot interaction feedback	0-9	7	7
social	human interaction modality	0-6	0	0
social/cognitive	cognitive social complexity	0-5	0	0
social/cognitive	social interaction learning	0-4	0	0
social/cognitive	social cognitive abilities	0-8	0	0

Analyzing Cohen's Kappa Calculating the Cohen's kappa agreement scores for both participants for all capabilities resulted in a value of 0.823 similarity between their answers. According to the Cohen's kappa guideline, as the value resulting from this test is above 0.75, it indicates an excellent agreement between the raters and is a step towards evaluating ToRCH as an accurate and precise framework for capturing and assessing capabilities.

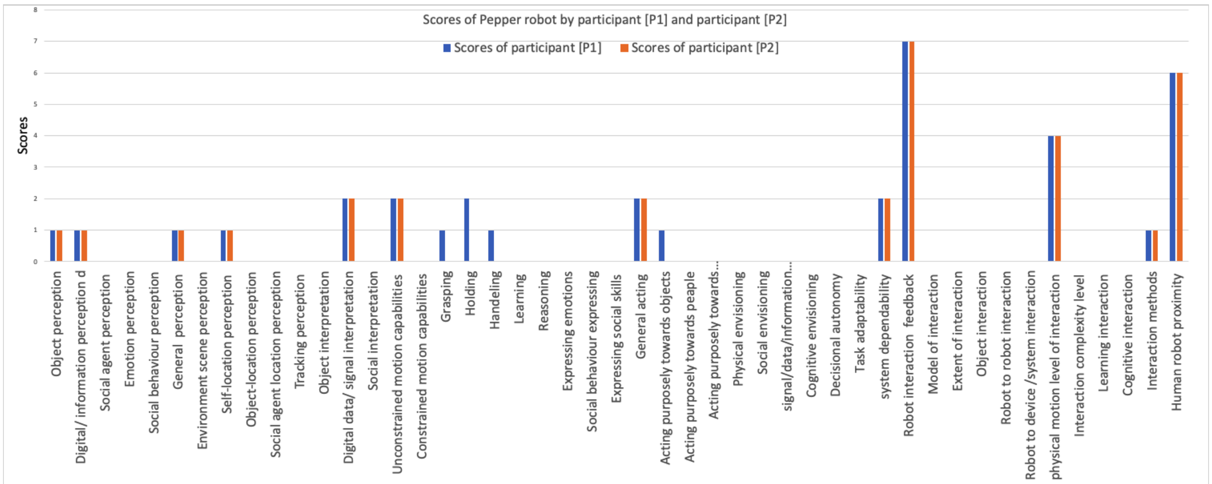

Figure 7.5: Pepper robot capabilities as scored by two participants. Most capabilities are scored the same, except three. One participant captured the capability of the robot with the manufacturing settings but the other participants scored the capability that was presented after development for a telepresence project.

Discussion and limitations

This evaluation scored capabilities and compared scores. The scores were analyzed further for inter-rater reliability statistics. A mean kappa was calculated and showed an excellent agreement between the two raters. The experiment demonstrated the accuracy of ToRCH for capturing capabilities. Nevertheless, it also demonstrated the importance of the user to know exactly what needs to be scored, such as the default robot capability or capability demonstrated within a project.

Thus, a user of ToRCH can use the framework to score an aspect of a robot, its default manufacturing setting, or what a project requires in the way of capabilities. It also highlights the need for a user to decide how they score the capability where an existing capability within a robot is not used in a project, that is, whether they score a robot's default capabilities levels or only the capabilities used in a project. Either method can be of use depending on what the user needs to capture.

7.3.4 Evaluating the effectiveness of ToRCH

Abstract

To evaluate the effectiveness of the ToRCH framework, participants scored the capabilities of their different projects that had been developed within the same robot. Each project contains

its own research specifications and was developed accordingly. Therefore, each project contains a set of capabilities that had been developed specifically for it. This evaluation intended to demonstrate ToRCH's ability to extract different robot capabilities performances even if they were developed in the same robot.

Introduction

Many robots are developed and programmed with capabilities to perform different activities and achieve a range of tasks. The effectiveness of ToRCH in distinguishing capabilities for different projects, tele-presence capabilities referred as [Proj1] and social robot capabilities referred as [Proj2], developed in the same robot, was evaluated. Each project had its own specifications and development requirements for the robot and therefore the robot contained two different sets of capability performances. Each performance was implemented by its own developer.

Methods

The capability scores were described and analyzed in several different ways for this experiment. First, the social capabilities presented in both projects were compared. Second, all ToRCH capabilities with similar scores were analyzed. Finally, the third assessment considered why capabilities were scored differently and explore the reasons behind the dissimilarity between the scoring of both participants.

Participants and procedures The participants in this experiment were developing a Pepper robot. Each participant developed different social requirements for different projects, [Proj1] and [Proj2]. In a one-to-one session, the participants filled in a hard-copy questionnaire, structured according to ToRCH, to score the capabilities of Pepper according to their project specification, which was developed in the robot. Where capabilities were not available in Pepper, such as robot-to-robot interaction and the envisioning capabilities, it was expected they would be scored at 0. Both participants of the research projects concentrated on social capabilities, but the first participant [P1] of the project [Proj1] was to develop telepresence capabilities and the second participant [P3] was to develop social interaction capabilities for project [Proj2]. Project tele-presence [Proj1] data was previously collected in evaluating the preciseness of ToRCH section 7.3.3, and this data was used again in this evaluation, for 'the effectiveness of ToRCH'. The participant [P3] developing the social robot capabilities project [Proj2] scored the capability of the project. Both lists of the scores were analyzed for this evaluation, to explore the effectiveness of ToRCH. Both participants scored all of the capabilities listed in ToRCH. The range of levels for each capability is presented in Tables 7.10 and 7.9. At the end of the questionnaire, participants were invited to evaluate the ToRCH framework and leave comments. The evaluation took around 60 minutes and a copy of the questions in presented in Appendix A.

Results and Discussion

First: Social capabilities scored for both projects Social capabilities scored are listed in Table 7.9.

Table 7.9: The social capability for the telepresence project [Proj1] scored by participant [P1] and social interaction project [Proj2] scored by participant [P3] are listed.

Social capabilities scores	**Range**	**P1:**tele-presence[Proj1]	**P3:**social inter-action[Proj2]
social agent perception	0-9	0	2
social agent location	0-8	0	1
social emotion perception	0-6	0	0
social behavior perception	0-6	0	1
social human agent interpretation	0-6	0	1
emotion expression	0-9	0	2
social behavior	0-5	0	2
social skills	0-5	0	2
social acting purposely towards human	0-8	0	2
social/human envisioning	0-6	0	0
social interaction levels of extent	0-8	0	1
social human-robot interaction feedback	0-9	7	1
social human interaction modality	0-6	0	3
cognitive social complexity	0-5	0	1
social/cognitive interaction learning	0-4	0	0
social cognitive abilities	0-8	0	2

The social capabilities for the telepresence project [Proj1] evaluated by [P1] scored 0, as for this project the social capabilities were presented through the virtual person located elsewhere but displayed on the robot through telepresence communication. In contrast, the social capabilities for the social interaction project [Proj2] evaluated by [P3] scored higher than 0 for almost all of the social capabilities, apart from ‘social emotion perception’, ‘social envisioning performance’ and ‘social interaction learning’ because the robot was not designed to perform these capabilities.

Analyzing the emotion perception scores indicates that the robots were not developed with any emotion perception in either of the research projects. However, for the social interaction project [Proj2] the ‘social skills’, ‘social behavior’ and ‘emotion expression’ capabilities scored level 2, as the robot performed social interaction, as presented in Table 7.9. By contrast, these capabilities scored level 0 in the telepresence project [Proj2], as these social aspects were presented through the communication of the telepresence person. This illustrates that dissimilarity is due to the requirements of the project.

Second: Capabilities scored similarly by both projects A direct comparison between the scores for both projects was performed. The capabilities that were scored similarly by both participants are presented in Table 7.10. Analyzing the data of similar capabilities scored for both participants shows that either the capabilities were required for both projects or the similar score capabilities are heavily related to the robot features and therefore both participants scored the same level. This can be discussed further:

- The manipulation capabilities were scored at level 2 and level 1, for each of the sub-capability in both projects [Proj1] and [Proj2]. As both Pepper projects required no manipulation with objects, it is likely that these scores were assigned to the default setting of the robot by both developers. This is also reflected in ‘object interaction capability’ which scored level 0 for both projects.
- Both participants also scored level 2 for the robots’ general action capabilities. This level indicates that the robot Pepper performed several actions as part of its in-built capabilities. Similarly, the ‘physical motion interaction’ capability was scored level 4 by both participants. The level illustrates that the system can use peripherals to perform physical interaction within its environment. This capability level defines the capability feature within the robot rather than capturing the requirement of a project or presenting a maturity level for its capability.
- The robot was also not developed with envisioning capabilities, as these were not required by the projects. These are capabilities for advanced development.
- Assessing the constrained motion scores at level 0 for both projects indicates that the robot did not respond or react to any external force, which is a default setting of the Pepper robot.
- Robot-to-system/device interaction was assigned 0 by both projects [Proj1] and [Proj2]. The score of this capability set was due to the manufacturing default of the Pepper

robot. These capabilities were not changed in the development of the project. Such capabilities are usually used for swarm robotics or robots with IoT.

Table 7.10: List of scores that were scored similarly by both participants [P1] and [P3].

Description	**Capabilities with similar scores**	**Range**	**P1** tele-presence [Proj1]	**P3** social [Proj2]
social perception	emotion perception	0-6	0	0
physical task	constrained motion	0-6	0	0
physical	manipulation: grasping	0-9	1	1
physical	manipulation: holding	0-6	2	2
physical	manipulation: handling	0-10	1	1
cognitive	signal/data learning	0-16	0	0
general task	general acting	0-3	2	2
physical	objects envisioning	0-6	0	0
cognitive	signal/digital/info envisioning	0-6	0	0
cognitive	cognitive envisioning	0-6	0	0
social	social/human envisioning	0-6	0	0
physical	physical motion interaction	0-5	4	4
physical/cognitive	object interaction	0-10	0	0
cognitive	robot to system interaction	0-10	0	0
social/cognitive	social interaction learning	0-4	0	0

Third: different capabilities scores for each of the research projects To analyze the different capabilities scores for both projects, a detailed list of the capabilities that were scored differently is presented in the Figure 7.6. The differences can be analyzed as follows:

- Social agent perception capability scored level 2 for the social research project [Proj2] and scored level 0 for the telepresence research project [Proj1], as presented in Table 7.9. This capability level detected a person in the project [Proj2] as part of its social performance aspect. The score was higher than the score selected for the telepresence research project [Proj1], which was level 0 as there was no requirement for the robot to perceive an agent in this project [Proj1].
- The tracking perception capability scored level 0 for the telepresence research project [Proj1], and it scored level 3 for the social research project [Proj2], presented in the Figure 7.6, as it is was a requirement of [Proj2].

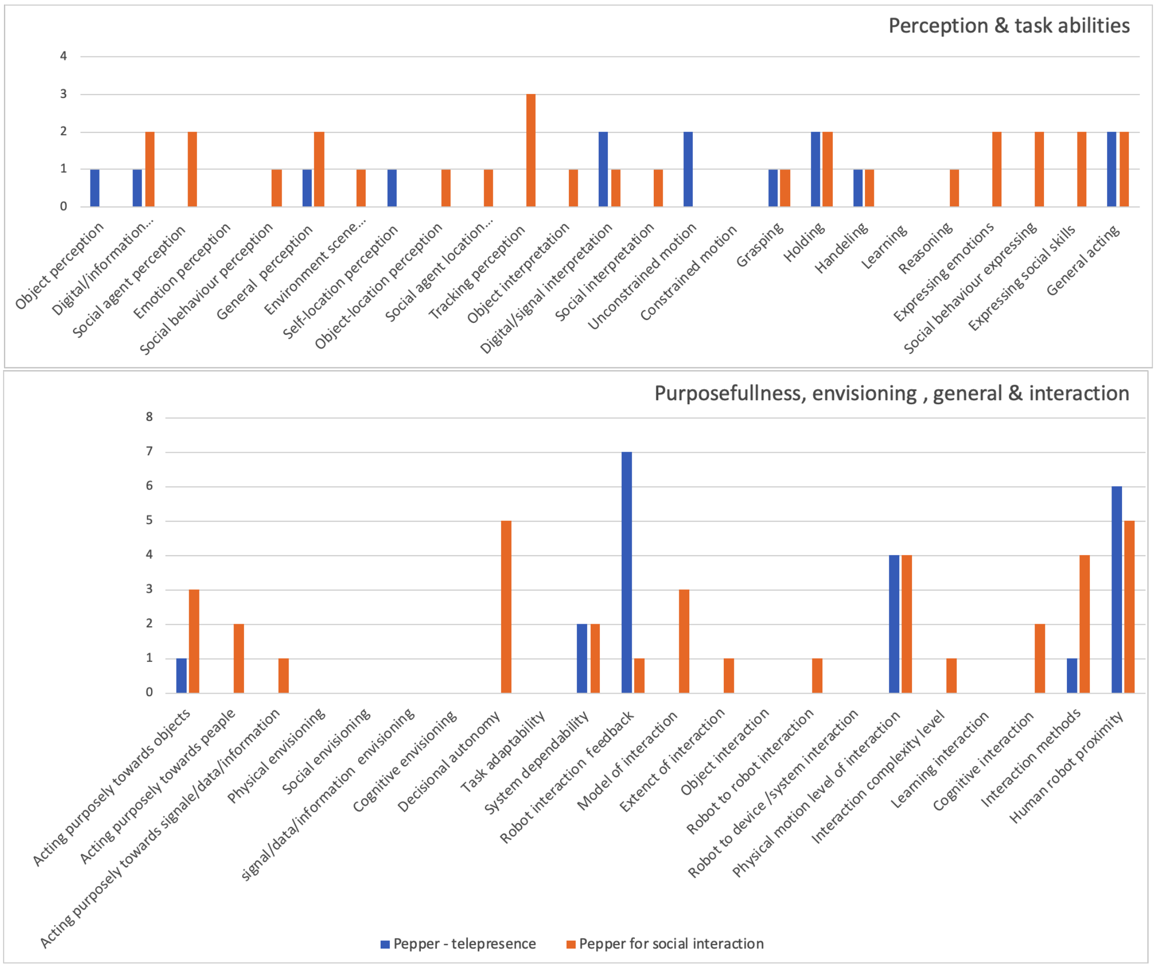

Figure 7.6: The chart shows a detailed list of the scored capabilities by participants [P1] and [P3] for Pepper robot for both projects telepresent research project [Proj1] and social research project [Proj3], respectively.

- The unconstrained motion capabilities scored level 2 for the telepresence research project [P3], and it scored level 0 for the social research project [P4], presented in the Figure 7.6, as it was a requirement of P4 that the robot did not perform any mobility.
- Decisional autonomy scored level 5 for the social research project [Proj2], but it scored level 0 for the telepresence research project [Proj1], as [Proj1] did not require any decisional capability in its performance as it was tele-operated.
- The feedback interaction capability scored level 7 for the telepresence research project [Proj1]. Level 7 specifies the telepresence communication type. The same capability was scored level 1 for the social research project [Proj2], as the robot was not designed

to provide any type of feedback to the user, where the user is assessing the robot through direct observation.

Additional analysis was performed on the extracted data to define the disagreement between the scoring of the raters. The assessment was designed to define discrepancy in scoring the capabilities of the two projects [Proj1] and [Proj2] as the Pepper robot was programmed differently in each project. In this evaluation, we speculated that there would be inconsistency in scoring the capabilities of the two projects with different scores for each research project. Using the inter-rater reliability analysis Cohen's kappa (Grayson and Rust, 2001), a very low agreement value was expected, which would indicate little agreement between the raters, as they were scoring two different projects. The Cohen's kappa measure to define the disagreement between both participants was calculated as 0.11, which defined no agreement between the scores and indicates that the raters scored different projects.

Conclusion

Most of the robots had been developed by several roboticists. Each developer had added a specific capability to the robot to increase its performance. Some of these capabilities were linked together, while others had been developed separately. The evaluations explained the relations between specific capabilities scores and robot project type. The assessments also showed that some capabilities were scored according to the project requirements, which clarifies that some specific capabilities scored at higher levels than others. The evaluations showed the interdependence between the scores of different capabilities. It defined a correlation between certain types of capabilities and the type of robot performance. For example, social capability scored higher in the social interaction research project [Proj2] and lower for the telepresence project [Proj1]. Additionally, the telepresence research project [Proj1] contained less decisional capabilities, whereas the social research project [Proj2] scored these capabilities higher.

One of the experiments also demonstrated that capabilities reliant on the robot features scored the same for both projects regardless of the capabilities that were developed as software in the robots. The experiment also showed that ToRCH was able to extract two different robot research project capabilities, as robot performances were different, even though the capabilities were developed in the same robot. This feature could help developers and designers to use an existing capability in a specific robot and adapt it; either by activating the module from one project to another in the same robot or by copying the module from a robot to another similar robot.

In general, this evaluation phase defines some important criteria for ToRCH, it demonstrated ToRCH as an accurate, rich, precise and effective tool in capturing robot capabilities and in expressing the different performances of any given robot. Therefore, this phase enhances the usage of ToRCH.

7.4 Third evaluation phase: Assessing the outcome of ToRCH (RCP)

Abstract

This third evaluation phase contains several assessments that focuses on the outcome of the ToRCH (RCP) and demonstrates its usages.

Introduction and general method

The evaluation involved assessing the robot capabilities profile (RCP) generated via ToRCH. The profile lists the capability scores and presents them in a specific order. This phase analyzed different aspects of the RCP. It illustrates three main uses of the RCP. The first utilization is to describe the research progress by capturing the RCP of robots in research before and after development. The second utilization is to illustrate the performance of the capabilities of a robot for different projects. The third utilization is to illustrate the capabilities of a connected robotic system.

Participants Researchers from Sheffield Robotics Lab were asked to volunteer in this study. They were recruited through the university email and had developed some capabilities within a robot for their research. They were asked to score the capability of their developed robot. Each volunteer filled the online questionnaire twice, one time to score the capabilities of the robot before development and the second time is to score the capabilities of the robot after development. There were three participants that scored the capabilities of their robots via an online questionnaire. The studies were approved by the University of Sheffield Research Ethics Committee and informed consent was given by all participants. No personal or demographic information was recorded. The questionnaires were online, as it was the simplest method to collect the data. Leading questions were avoided and no scores were given to the participants to avoid influencing their answers. No data was shared outside of the research team. The collected data was kept in accordance with University of Sheffield best practice, on the password-protected university servers.

7.4.1 First assessment: RCP to demonstrate robot development within research projects

Introduction

Using ToRCH to describe the progress of the development of a robot provides a full insight into how and what was developed within the robot. RCP was captured once before development and once again after development. The RCP illustrates the capabilities that were developed within the robot during the research period.

Procedures

This usage of RCP is illustrated by three robot research projects: a drone project, a Miro social robot and a Pepper social robot. Each of the research projects had two RCP scores. The scores were plotted as a graph with two bars, one before and one after development. A comparison between two successive bars highlights the achieved progress in the research project.

Development of a drone Within this research, project [Proj3], the developer increased the capability of the drone robot dramatically. The RCP illustrates the increased levels of scores in almost all the listed capabilities, as presented in Figure 7.7. Two capabilities decreased their levels. Analysis of the decreased capabilities indicates the following:

- The first decreased capability was 'acting purposefully towards signals'. The robot was assigned to level 1 due to its default manufacturing setting, however, after the development of the robot, the researcher shifted 'action purposefully towards the signals' to level 0. This was done due to the requirement of the project. The requirement is reflected as a decrease in the capability and consequently a reduction of its score. Hence, this decrease in the levels of the capabilities was demonstrated as part of the development of the robot.
- The second capability that decreased was 'human-robot proximity'. Human-robot proximity was scored level 3 and then decreased to level 2. The human-robot proximity capability levels are defined as follows

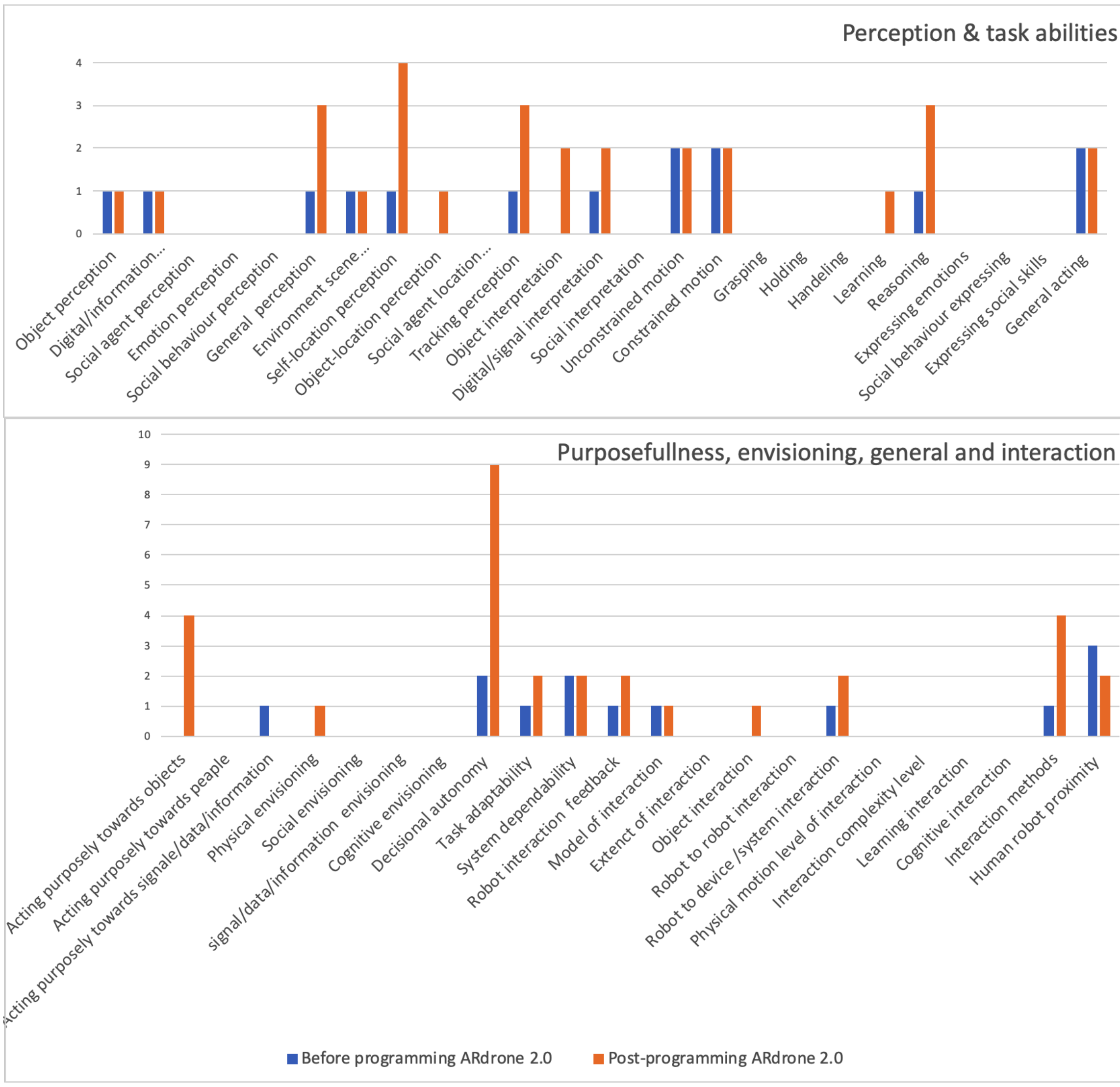

Figure 7.7: Drone capability scores presented as before development and after development

- level 3: describes a robot that is operated and commanded by a human.
- level 2: describes a robot working independently alongside a human.

This decrease in capability was again due to the project requirement.

Development of a Miro social robot The research project, project [Proj4], on the Miro robot increased the levels of all the listed capabilities, apart from one, as presented in Figure 7.8. The exception was in the interaction methods, which the developer of the robot scored level 5 before the development and level 3 after the development. The interaction method capability levels are defined as follows:

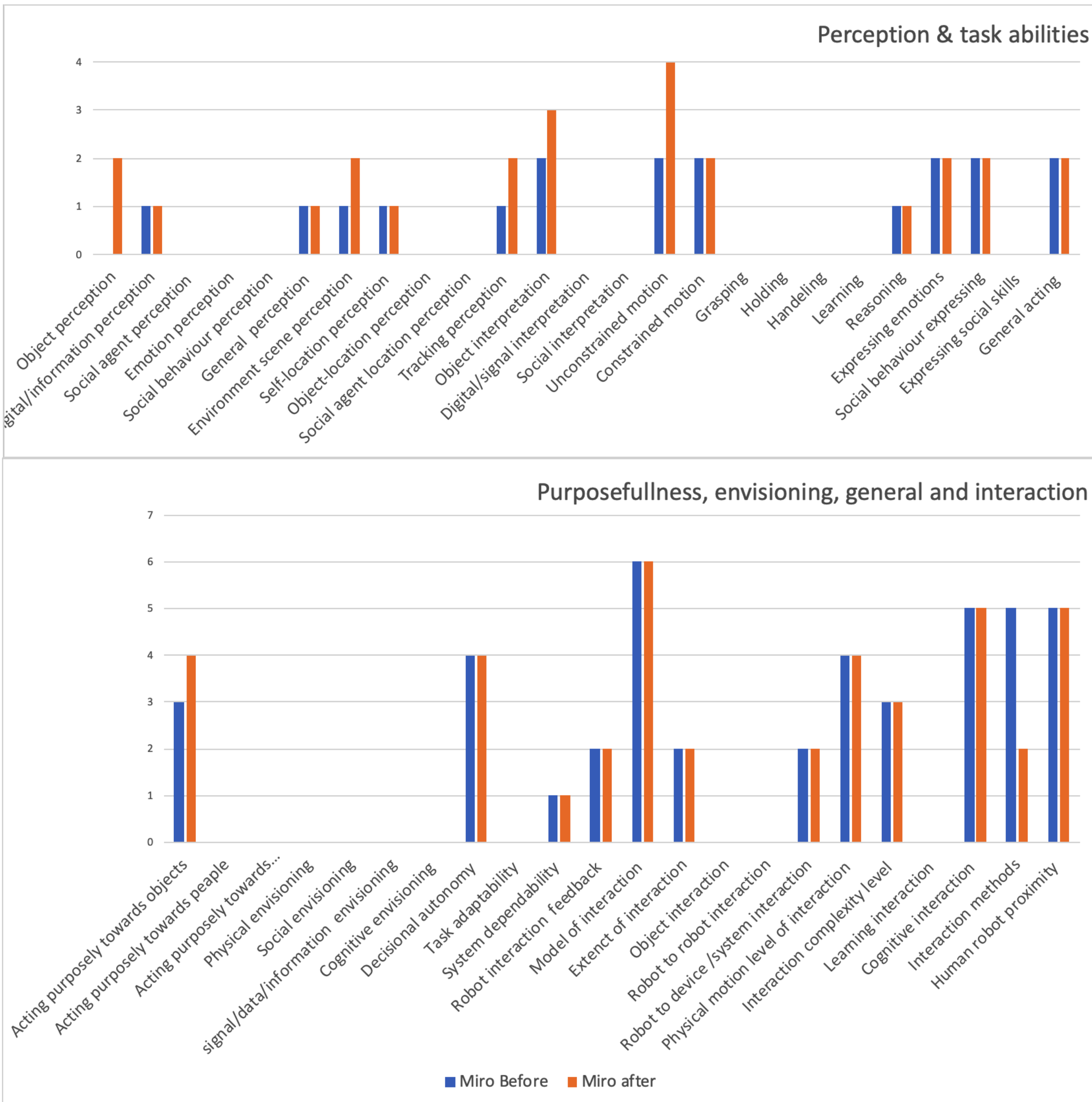

Figure 7.8: MiRo capability scores presented as before development and after development.

- level 3: indicates that the user has to predefine the next task.
- level 5: like task sequence controls, indicates that the system was able to execute sub-tasks autonomously.

The reduction in the score of this capability was due to the research requirement of the new system.

Development of a Pepper social robot The before and after scores of the RCP were captured for the research development of the social robot Pepper, project [Proj5],. The

RCPs of the robot before and after the development were identical, demonstrating that the development was not captured via ToRCH hierarchy. This could be due to the development focusing on improving the performance of existing capability levels and the efficiency at which they are performed, rather than increasing a level of a capability. As the current version of ToRCH does not include any attributes or parameters to capture the degree of performance of each capability level, the research development did not affect the RCPs. This indicates that future updates of ToRCH should include a way to capture different performance within the same level.

Results

The RCP can display the development of a robotic research project by presenting the capabilities of the robot before and after development. This evaluation illustrates the progress of three different robots; the drone, Miro and Pepper; which brings to light the importance of the RCP as the main outcome of ToRCH.

7.4.2 Second assessment: Using RCP to demonstrate the different capabilities of two different projects

Introduction

RCP describes robot capabilities. Some robots are developed with several sets of capabilities. Usually, this is seen when a robot is used for more than one project. This suggests that the generated RCPs for each project would be different form each other.

Procedures

For this RCP utilization and assessment, the Pepper, Kuka and Kilobot robots were used because they were developed for two different research projects. Each project executes different capabilities from the other. Therefore, the RCP was generated for each of these research projects. The robot's capabilities for each of these three robots were scored twice, once for each project, and presented graphically in Fig 7.9, Fig 7.10 and Fig 7.11. Comparing the scores within each graph indicates which robot project performed more capabilities than the others. It also illustrates which research project specified capabilities at a higher level.

The Pepper robot The Pepper robot was used as a development platform for two different research projects. The capabilities scores of the Pepper robot for the two research projects are presented in Fig 7.9. The data for both projects were collected online for the previous experiments and used for this demonstration as [Proj1] and [Proj2]. The graph indicates higher scores in almost all social capabilities within the social interaction research project [Proj2], whereas the RCPs illustrate fewer capabilities being developed for the Pepper robot for the telepresence project [Proj1].

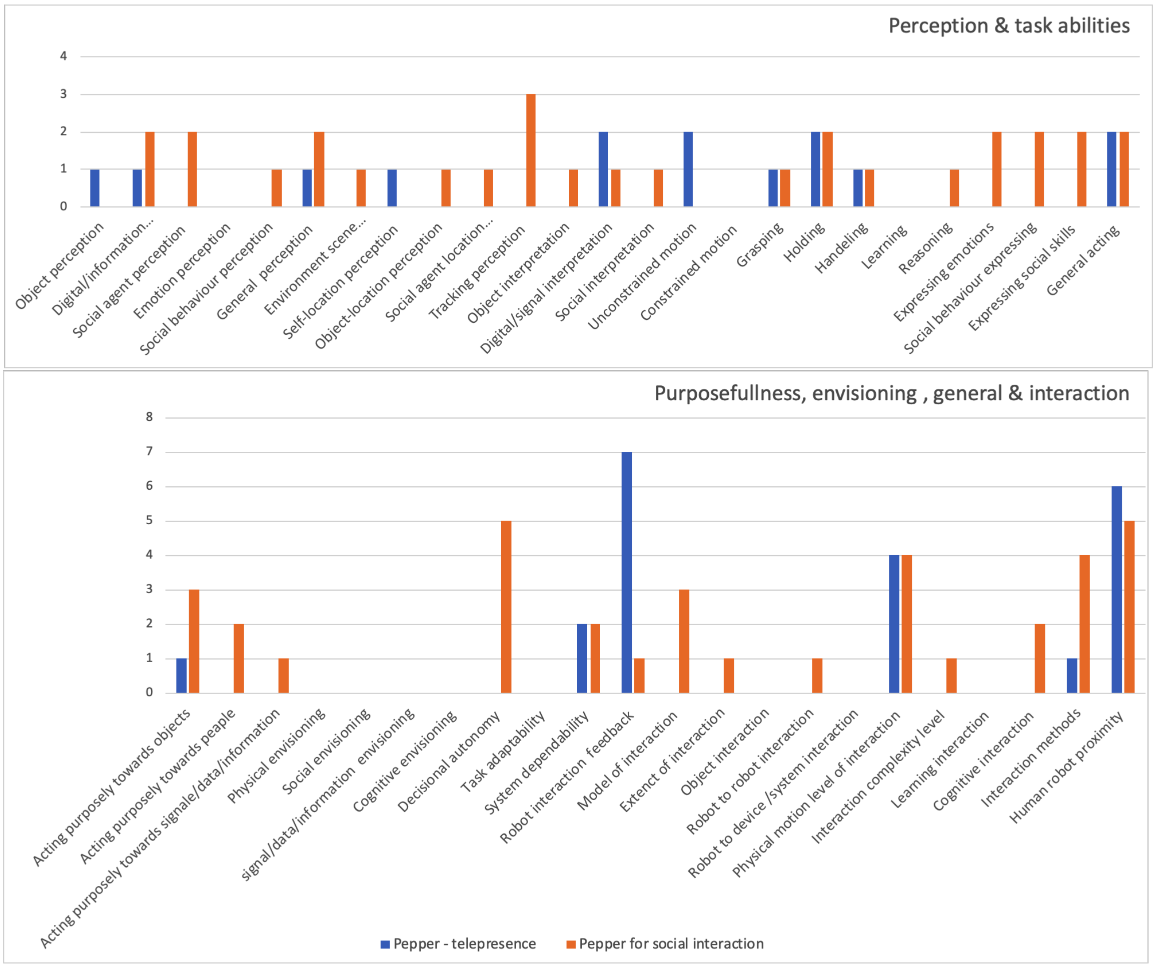

Figure 7.9: Pepper capability scores for social interaction research project [Proj2] and telepresence project [Proj1] indicate the attainment of high levels of social capabilities for [Proj2] and lower levels for [Proj1].

The Kuka robot The Kuka robot was used as a development platform for two different research projects. The data for both projects were collected online and used for this demonstration as [Proj7] and [Proj8]. The capabilities scores of the Kuka robot for the two research projects are presented in Fig 7.10. The figure indicates more capability performance for the Kuka robot for project [Proj7] than for project [Proj8]. The figures also indicate that the Kuka robot for the project [Proj7] performs most of its activities on physical objects in contrast to the robot in project [Proj8], which demonstrated mostly social and cognitive capabilities.

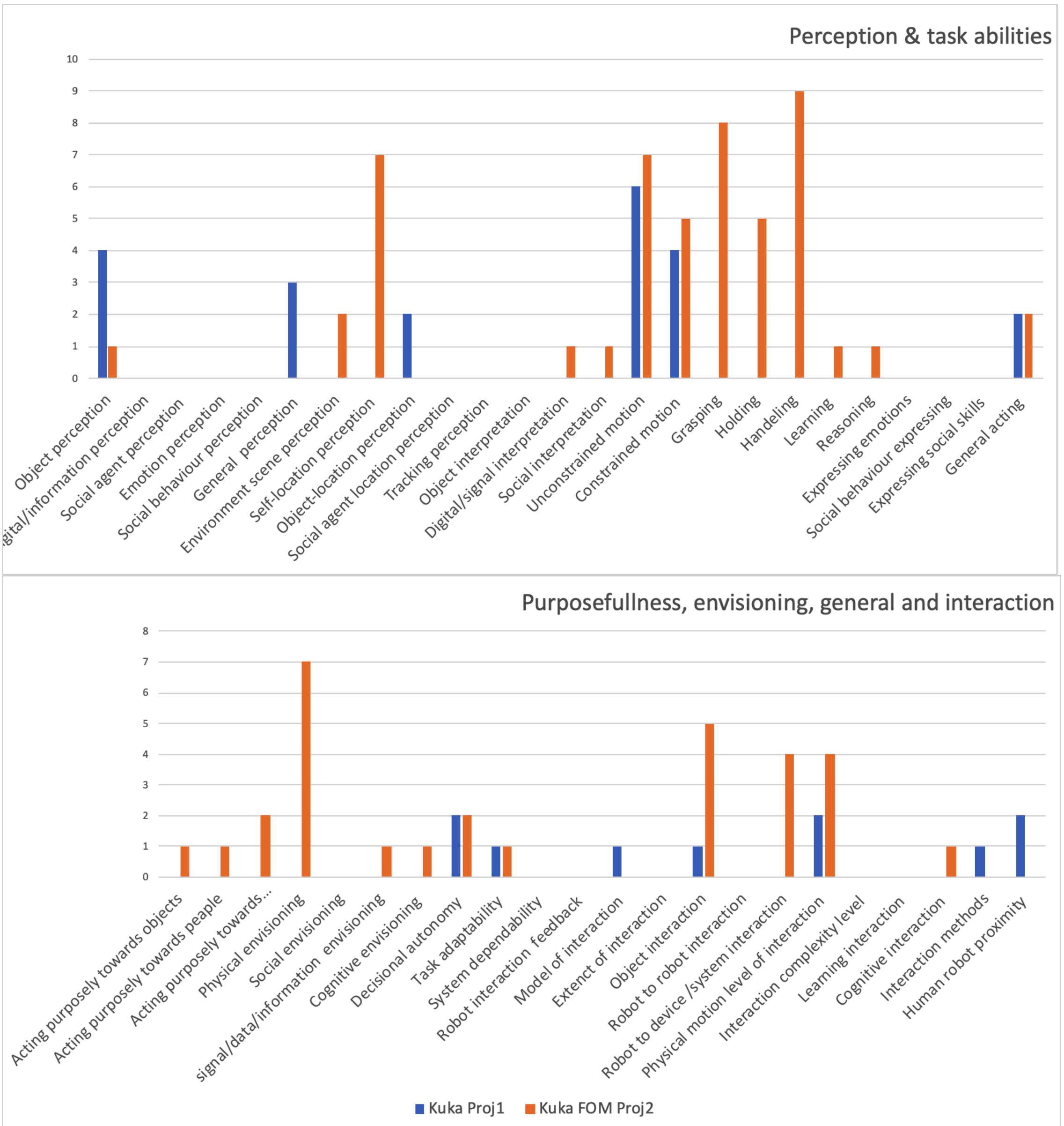

Figure 7.10: Kuka capability scores for project [Proj7] and project [Proj8] indicate more capability performed for KuKa in [Proj8] than [Proj7].

The Kilobot robot The Kilobot robot was used as a development platform for two different research projects. The data for both projects were collected online and used for this demonstration as [Proj9] and [Proj10]. The RCPs containing the capabilities scores for the two research projects [Proj9] and [Proj10] are presented in Fig 7.11. The graph indicates a major similarity of capabilities between the projects. There were differences in two of the capabilities: the first was the ‘general perception’ capability, which was available in project [Proj9], the second was ‘decisional autonomy’, which had a higher maturity level for project [Proj9] than project [Proj10]. Also, the graph indicates that neither social nor cognitive abilities were developed for either of the research projects.

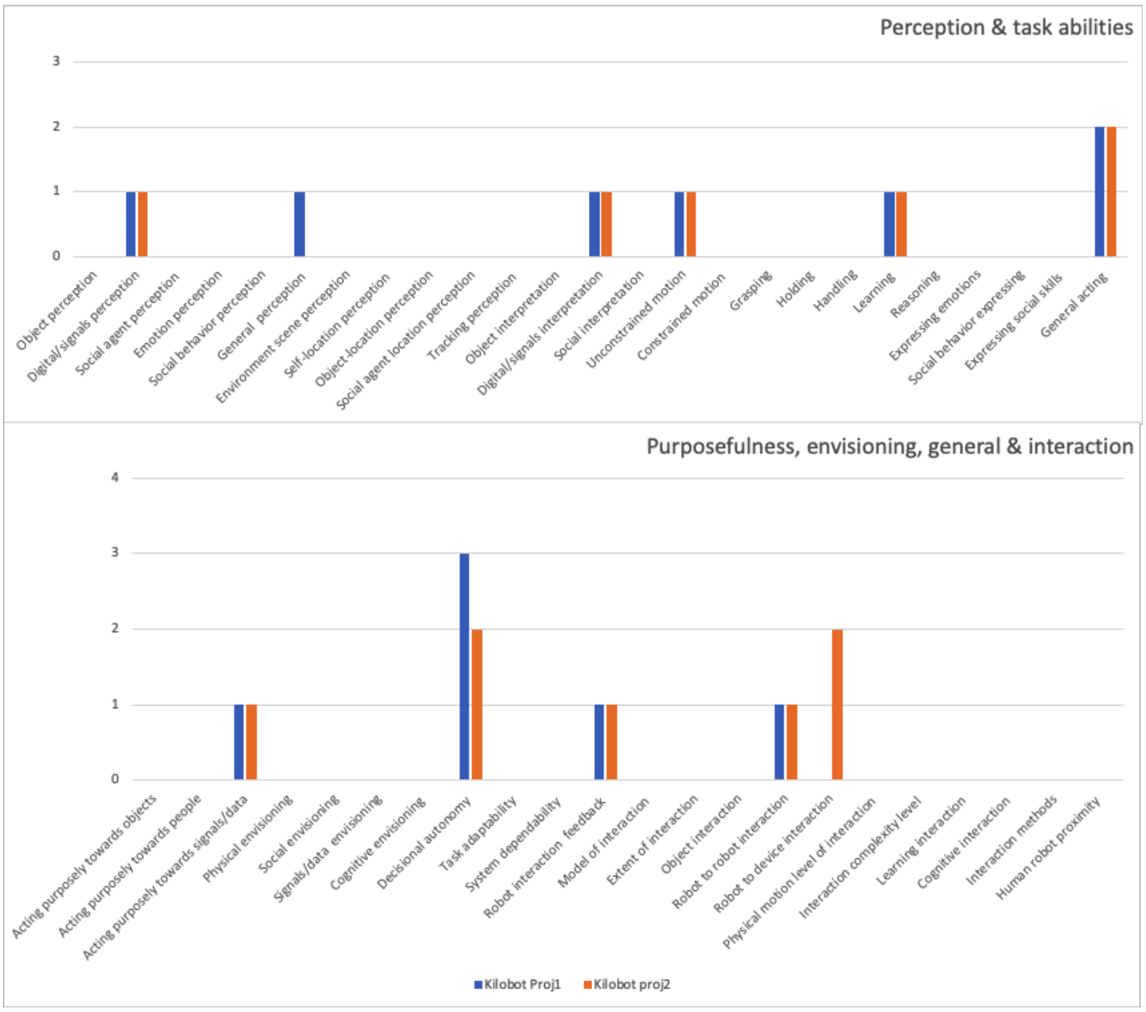

Figure 7.11: Kilobit capability scores for [Proj9] and [Proj10] had a significant similarity between the two robots. The graphs also show that there were no social and or cognitive aspects within either robot.

Results

The utilization and the assessment of the RCP illustrate how the RCP can differentiate between two different research projects developed using the same robot. The RCPs for the development of these research projects are clear and comprehensive, which makes it easier for roboticists to utilize the RCP to present the development of their research and illustrate the development of their robots.

7.4.3 Third assessment: Assessing RCP of robots with other systems

Introduction

Robots can perform a series of actions. They perform their tasks either individually or jointly with external systems. The external system can act as peer, guidance or a controller. Some of the connected systems can contain hardware and software similar to those that the robot has. In some cases, the external system can have more technical features than those available in the robot. Regardless where the hardware or software are installed in the connected system, the capabilities performed by each part of the system are illustrated through the RCPs. The data for both projects, the Drone project [Proj11] and the Kilobot project [Proj11], were collected online and followed the University of Sheffield Research Ethics Committee guidelines. No personal or demographic information was saved.

Procedures

The RCP shows which capabilities are performed via the single robot, the connected system/device or both. This utilization of the RCP and its assessment are presented in the following two research projects.

Drone connected to external system A drone was developed to connect to an external system, project [Proj11]. The drone and the external system were each developed to perform different tasks. The RCPs were scored for both parts to illustrate the performed capabilities of the individual drone and the capabilities of the external system. The RCPs are presented in Fig 7.12.

The drone contains a camera that acts like a real-time interface. It transfers all the captured images to an external system, and the external system controls the drone and analyzes the images received. Therefore, the drone was developed solely with the capabilities of flying and

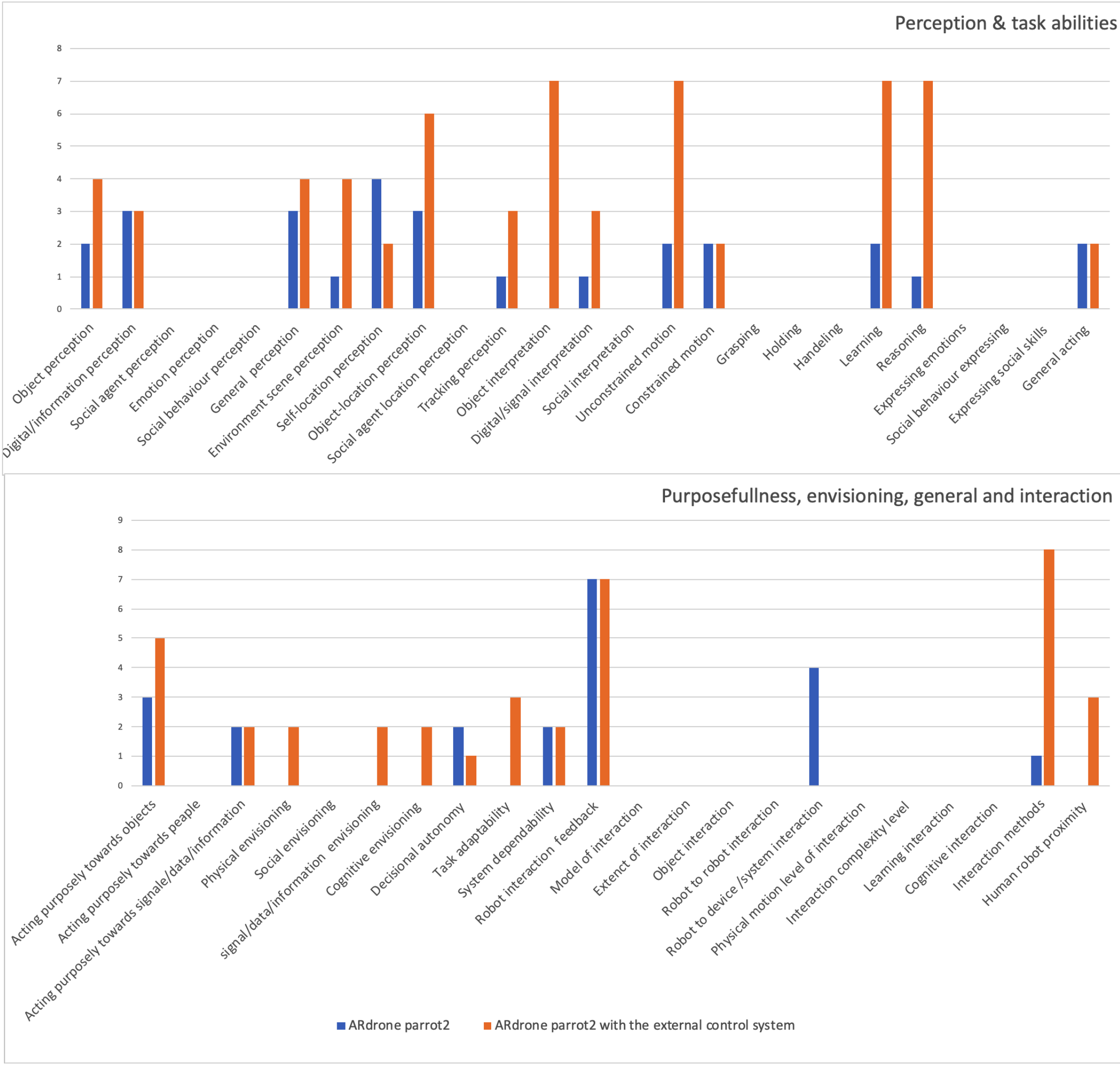

Figure 7.12: The drone capabilities scores [Proj11] are illustrated along side the capabilities of the drone connected with the external systems.

capturing images of objects in the environment. Some of its manufacturer's default settings were disabled due to the requirements of the research, resulting in reduced capabilities. In contrast, the external connected system was developed with several capabilities. The aim of the research project was to study the feasibility of a novel control method developed to connect the robot drone and the external system. The RCP of the drone was compared to the RCP that included the drone and its connected external system, as presented in Fig 7.12. The RCPs demonstrate the difference in achievement of the connected system compared to the individual drone.

Kilobot connected to other Kilobots A swarm of Kilobots was developed so that the Kilobots could communicate with each other, project [Proj12]. The RCP of the Kilobot as a single robot was captured and the RCP of the swarm as a whole system was scored in a separate profile. The individual Kilobot robot capabilities scores are shown alongside the capabilities performed by the whole swarm system, as presented in Fig 7.13. The graph demonstrates which part of the connected system performs which capability, in a way that identifies the obtainer of the capability and its originator or provider.

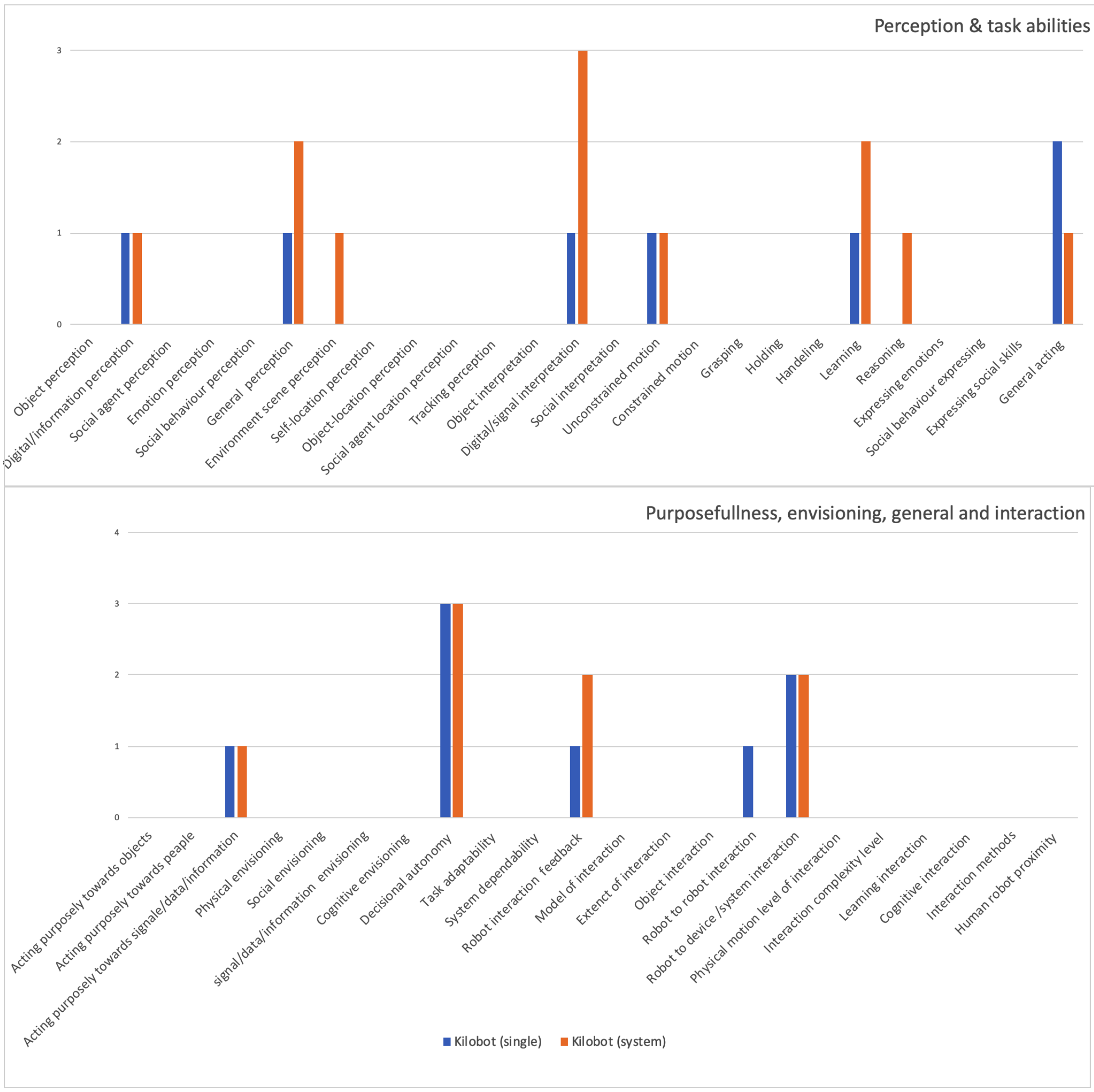

Figure 7.13: The capabilities scores are presented for the Kilobot [Proj12] as an individual single robot and also for the Kilobot swarm system as a whole system.

Results

The experiment highlighted that some of the capabilities levels were described by joint phrases, such as “performing capability A **and** performing capability B” or “performing capability C **or** performing capability D”. Such joint phrases resulted in some ambiguity in the scoring of the levels. This caused the rater to pick either of the levels as long as the capability was mentioned, without considering the joint clause. Such issues might cause some incorrect scoring of the capability levels and their distribution, yet it would not affect the existence of the capability in the robot.

These two listed studies illustrate the capacity of the RCP to capture the performance of different parts of the connected system. The RCP determines which capabilities are performed by the robot and which capabilities are performed by the connected system. It clarifies the source of the capability in joint systems.

This evaluation proposes the utilization of the RCP to identifying the source and the region of the capabilities that are exhibited by a robot. The evaluation is applied to robots that are designed to connect with another system. The evaluation highlights that the RCP’s can identify which capabilities originate from which part of the robot.

7.5 Discussion and outcome of all evaluation

The first evaluation phase assessed the structure of ToRCH by holding personal meetings with renowned roboticists to correct the ToRCH hierarchy. The outcome of this evaluation is presented in Table 7.11.

Table 7.11: This table shows how ToRCH structure was evaluated ***(Phase 1)***

Assessment	Method	Outcome of the assessment	Implications of the assessment
Evaluating the ToRCH framework	Personal meetings (over 20)	1. Identifying classification dimensions and settings its main layers and sections 2. Allocating characteristics within layers and developing the subsections 3. Amending the relations between the layers, sections and subsections	Development of the framework

The second evaluation phase assessed ToRCH as a tool. The outcome of the following sets of evaluation are presented in Table 7.12, to evaluate the accuracy of the tool, and an experiment was performed to check whether participants would define the capabilities of a robot similarly, as presented in section 7.3.1. The evaluation showed 88% accuracy of ToRCH in capturing robot capabilities as the experiment showed a moderate agreement between raters in defining the scores of the capabilities. To evaluate the richness of the tool, an experiment was performed to check whether the capabilities captured by ToRCH are sufficient to describe the performance of the robot. The experiment checked whether ToRCH contains an adequate list of capabilities by comparing the list with another capability list, such as MAR. The evaluation demonstrated that the capabilities captured by ToRCH are much more descriptive and detailed than the capabilities captured by MAR, as presented in section 7.3.2. This positions ToRCH as a tool to define robot capabilities. To evaluate the precision of the tool, an experiment was conducted to check if two participants developing the same robot scored the same capabilities levels for their specific robot, as presented in section 7.3.3. The capabilities scores were analyzed for inter-rater reliability revealing a high level of agreement between both participants. To evaluate the effectiveness of ToRCH in capturing different capabilities, an experiment was performed to score the capabilities of different research projects that were developed in a robot, as presented in section 7.3.4. This evaluation demonstrated ToRCH's capability to extract different performances for different projects that are developed using the same robot. The experimental analyses included the following: justification of the social capability scores, explanation for the similarity between the capability scores, reasons for the difference between the capability scores and degree of disagreement among the raters.

The third evaluation phase has assessed the outcome of ToRCH, the RCP. The conclusions of these assessments and their implications are presented in Table 7.14, and they were three: The first assessment was to demonstrate the development of the three research project robots: the drone, the Miro social robot, and the Pepper social robot. This assessment was performed by illustrating the development of robots in different projects (drone, Miro, Pepper). The outcomes of this study illustrate the advantages and disadvantages of the RCP and its scoring scheme. The second assessment was to demonstrate the different capabilities of robots that were developed differently. The assessment used the Pepper robot, the Kuka robot and the Kilobot robot to identify their development in separate projects. The RCP for each robot in each project was captured separately as each of the robot projects was distinct. Finally, the third assessment was to evaluate the capacity of the RCP in capturing the source of the capabilities in a joint system. It also illustrated which capabilities were performed by

*Table 7.12: This table shows how ToRCH was evaluated as a tool **(Phase 2)***

Assessment	Method	Outcome	Implications
Accuracy	Study	88% accuracy in defining capabilities	Capabilities are captured
assessing the accuracy in the capabilities scoring	Survey with inter-rater reliability analysis 'Cohen's Kappa analysis'	Moderate agreement between raters	Required training, understanding terminologies, specifying versions, engagement with the development, understanding the default settings of the robot, understanding how capabilities are spread between levels
Richness illustrate the amount of capabilities that are scored	Comparison between ToRCH and MAR in describing robot capabilities performance	Describing robot execution through ToRCH is descriptive, by illustrating a lot of details to identify the capabilities of the robot	ToRCH is adequate for defining robot capabilities. None existing capabilities in ToRCH should be added
Precision checking the repeatability in scoring the capabilities	Survey and inter-rater reliability analysis'Cohen's Kappa analysis'	Excellent agreement between two developers for their RCP scoring	Describing the capabilities via ToRCH requires identifying which capabilities need to be scored, e.g., as executed capabilities, or as capabilities present but not executed
Effectiveness detecting the capabilities of different projects	Survey and inter-rater reliability analysis 'Cohen's Kappa analysis'	No agreement between raters, as capabilities scores were different due to different projects and requirements, even though the capabilities were developed within the same robot	Capabilities scores depending on the development of the robot SW are always different, whereas capabilities depending on the robot physical features score are always the same. Significant correlation between: (1) scores of related capabilities. (2) capabilities scores and related features. (3) capabilities scores and the project type

which part of the system. This assessment was conducted via two studies: the drone and the Kilobot.

Capabilities scores

The scoring system used in ToRCH follows the one that was created and used in MAR. The scoring scheme contains several levels, listed from zero, which indicates no availability, to the highest score that capability could present, indicating the most advanced capabilities. This levelling schema has several scenarios that might causes some inconsistency in the scoring arrangement and leads to misinterpretation of the capabilities and its levels. These misinterpretations could be due to one of the issues that are listed in the Table 7.13

Table 7.13: This table shows how the scores used by ToRCH might cause some misunderstanding and mis-scoring

The error	**Illustration of the error**	**Description of the error**	**Implications of the error**
combined capabilities levels	1, 2 & 3 ,4	capability 2 and capability 3 are listed in one level	two capabilities performance will score the same level
combined capabilities	1, 2, 3, 4, 5	the listed levels contain a level from a another establish capability, example level 4. Illustrated as (1, 2, 3, 1, 5)	The capability (a) will score either 1,2,3 or 5 and the capability (b) will score level 4, if it exists. Two levels form this list will be scored causing issue in the profile
capability level is misplaced	1, 2, 4, 5	missing a capability level 3	caused the scores to be underscore or over scored
capabilities levels are out of order	1, 3, 2	the capability are not listed with improvement but the levels are increasing	scores are not reflecting the improvement of the capabilities. Improvement of the capabilities are presented as a deterioration in performance.
several capability level are applicable	1, 2, 3, 4	the robot perform several levels from the listed capability	the rater need to score several levels which causes some issue in presenting the profile
none of the level are applicable	1, 2, 3, 4, ?	the robot perform the capability but it does not use any of the method mentioned	the rater need to score a level but it is not mentioned in the list. Therefore, this new level has to be added to the list

Table 7.14: This table lists the assessment of different illustrations of RCP and the outcomes of the evaluation ***(Phase 3)***

Assessment	**Illustration**	**Implications**
3.1 Describing the capabilities of a research project through the RCP	a. Development of a drone	Some capabilities were listed by descending concept but were scored in an ascending order
	b. Development of a Miro robot	Decreases in capabilities scores were due to project requirements
	c. Development of a Pepper robot	Robot development did not reflect the capabilities levels. This could be due to the development of a specific parameter that did not increase a capability level but did increase the efficiency of its performance within that specific level. Additionally, some robots can score two capability levels together, especially when capabilities are developed with switch on/off aspects
3.2 Presenting the differences in developing capabilities for the same robot	Illustrating the differences between two research project RCPs applied on three different robots: Pepper, Kuka, and Killobat.	Significant differences between the RCPs of the robots due to the differences in the requirements of their projects
3.3 Identifying the origin of the capabilities	Comparing RCP for individual single robot capabilities against the capability of the robot while connected to an external system	Some capabilities levels include joint phrases such as or/and, which cause some confusion to the rater. Also, adding new swarm capability was suggested, such as completing a task with various degrees of human intervention/control. Moreover, generating the RCP can help developers adopt capabilities from one robot to another

❋ Chapter 7: Summary

The previous chapter established a novel procedure to capture robot execution via the ToRCH framework. This framework is referred to as a tool as it captures and presents complex robot capabilities in a numerical presentation defined as the Robot Capability Profile RCP. In this chapter, the ToRCH and its RCP were evaluated in several phases. Each phase evaluated and assessed a specific aspect of ToRCH and its RCP. In general, the evaluations of ToRCH and its generated RCPs illustrate a significant description of the robot throughout the capabilities scores. The RCP contains a huge nested list of scores that help to indicate the capabilities levels of robot performance. It also facilitates the reporting of the capabilities in a numerical and efficient manner. In order to detect any deficiency of any score in the RCP, that might have a small substantial outcome or could consequently affect the description of the robots, a large sample needed to be captured. The sample requires all of its participants to be a developer of the same capabilities within the same robot. Currently, it is not feasible to perform this type of study due to the lack of multi-robot developers and the lack of developers who are able to program different capabilities within a robot. Once the robotic field is able to accumulate such a large number of participants, a group study can be performed, where each person would be capable of developing several capabilities and then scoring them in an RCP. This would then be compared with other RCPs scored by other participants, making it significantly clear which capabilities levels had been combined, misplaced or mis-ordered in the ToRCH scoring system.

Chapter 8

Conclusion

8.1 Reviewing the scope of the thesis

The aim of this research has been to develop a framework capable of capturing the capabilities of all robots. The framework uses dimensions (cognitive, physical and social) and a clear numeric scoring system to identify robots capabilities and implicitly characterize them. Applying the dimensions of the user requirements helped in assessing the research and its outcomes. The study also evaluated the Towards Robot Characterization Hierarchy (ToRCH) framework, its dimensions and its outcome the Robot Capability Profile (RCP). The research question for this thesis was 'can an overarching framework for identifying, classifying and comparing robots across all fields and domains be developed?', if so, then:

- What are the dimensions needed within such a framework?
- Can robots be defined through the dimensions of this framework?
- What steps are required to define a robot through this framework?

In answering these questions, the research performed the following, providing comprehensive answers of these question:

- Understanding the characteristics of existing robots by capturing an inventory of representative robots by building a database.
- Identifying which dimensions are available in the domains and fields that are applicable to all robot types.
- Designing a framework (ToRCH) capable of capturing all the dimensions and capabilities.

- Adopting measures when they exist (MAR) and developing them where they do not exist, in a systematic manner.
- Describing the utilization of the framework.
- Exploring the framework applications.
- Evaluating the framework and its outcomes.

In order to achieve the inclusive framework, there have been several research phases:

To identify the capabilities and characteristics of existing robots, a literature review was conducted (Chapter 2: Types of robots).

To collate the major capabilities, the Towards Robot Characterization Hierarchy (ToRCH) was developed with several dimensions. The dimensions were presented with different layers, each layer containing lists of capabilities. The capabilities were categorized by type and then further sub-categorized into sub-capabilities. Finally, each sub-capability was supported with a measuring system to indicate the technological maturity of the capability, adopted from MAR (Chapter 3: Developing a classification framework).

To create a comprehensive framework, new capabilities were required with their measures. Therefore, once capabilities available in MAR were adopted and allocated to some sections of the framework, other sections were found to be lacking capabilities. These sections were identified and highlighted to be developed with capabilities and measures that followed MAR. These developed capabilities were allocated to the most suitable section of the framework (Chapter 4: Beyond MAR).

To support the use of ToRCH, the framework was clearly described, including its hierarchical structure and a list of capabilities. The quality criteria and guidelines were written. Instructions for using ToRCH, including how to add new capabilities to the framework and how to identify robots by scoring their capabilities and generating their profiles (RCPs) (Chapter 5: ToRCH).

To explore the framework, its usability and the RCP generated by ToRCH, were described. An illustration of how the RCP can be used to identify robots was included. A description of how ToRCH classifies robot types according to their common capabilities was presented. There was also an illustration of how to map between user requirements and robots, through the use of RCP and ARP. The mapping was supported by an automated version of ToRCH through the development of a Java application (Chapter 6: ToRCH Applications).

ToRCH evaluations were performed through three main assessments, listed in Table 8.1 (Chapter 7: Evaluation of ToRCH).

Table 8.1: Evaluation of ToRCH

Evaluation	Approach	Description and outcome
1 Evaluating ToRCH		
1.1 Amending the ToRCH dimensions	Iterative process of discussion with several roboticists, using a structured questionnaire	This supported the ToRCH hierarchy of technical and operational layers with clear definitions of their sub-layers
1.2 Assessing the defined dimensions		Clear definitions of sections and sub-sections
1.3 Allocating capabilities and grouping them		Clear definitions of capabilities, sub-capabilities and sub-sub-capabilities
2 Evaluating the ToRCH as a tool		
2.1 Accuracy of the tool	Structured questionnaire	Testing the accuracy of capturing robot capabilities
2.2 Richness of the tool	Empirical study	Validating what ToRCH can measure
2.3 Preciseness of the tool	Structured questionnaire	Checking consistency in scores
2.4 Effectiveness of the ToRCH		Ability to detect capabilities of different projects
3 Evaluating the outcome of ToRCH-RCP		
3.1 Describing robot capabilities within a research study		Assessing an RCP before and after robot development
3.2 Capturing RCPs for specific robot applications over two different projects	Structured questionnaire	Assigning robot capabilities for a specific project
3.3 Comparing individual robot capabilities against a connected external system		Assessing individual robot capabilities against the capabilities of their external systems

Robot capabilities need to be expressed in terms that match the application requirements. Therefore, ToRCH was specifically designed to encompass any developments in robotic capabilities according to the requirements. By performing these phases and developing the ToRCH framework, the RCP was used to define a robot more precisely. The RCP describes robot capabilities, which moves toward an understanding of what comprises a robot, and partially answers the main question of the thesis “What is a robot?”.

8.2 Original contributions

This research has resulted in:

- The creation of a novel framework (ToRCH), for enumerating and characterizing robot capabilities.
- The ToRCH framework has the flexibility to add new capabilities as they are developed in the robotic domain.
- The ToRCH framework is referred to as a tool to capture and present complex robot capabilities in a numerical presentation defined as the Robot Capability Profile (RCP).
- The main outcome of ToRCH is the RCP, which describes robot capabilities according to the framework settings of its layers and sections.
- As the RCP mirrors the ToRCH hierarchy, it arranges the list of capabilities and their scoring in a specific format to ease the comparison process between robots.
- The ToRCH framework identifies the robot performance by scoring all its capabilities, even the general capabilities or the sub-capabilities.
- The outcome of the ToRCH framework, the RCP, was evaluated by several perspectives with several phases.

8.3 Limitations

The creation and evaluation of ToRCH involved many phases. A potential limitation is the small sample of robotics programmers and developers involved. In evaluating ToRCH, the sample of participants and number of robots was small. Nevertheless, as an initial evaluation of the framework, this demonstrated the usefulness of the RCP and the specifications of ToRCH. Ideally, to evaluate ToRCH, a wider sample is needed, possibly involving a workshop where a single robot is developed by a group of researchers. Currently, it is not feasible to perform this type of study due to the lack of single robot developers, especially in PhD research. Once it is possible to accumulate a large number of participants able to rate their developed capabilities in a single robot (as collaborated work), it will specify what the weaknesses are in the scoring system.

8.4 Implications of ToRCH

8.4.1 Distinguishing between a robot and other classes of machines

ToRCH can help in distinguishing between a robot and a machine as it captures the characteristics of a system through its capabilities, which clarifies their identification. Capturing these specific characteristics though the RCP will clarify what the system is can perform and through scoring the technological maturity levels of its capabilities. This will consequently help in identifying systems and comparing them. For example, if the system is performing interactions either socially or with objects, then it is likely that it can be defined as a robot, but if it performs interactions only with signals and data, then it most likely will be defined as a machine.

8.4.2 Benchmark and KPI

A benchmark is a method that uses measures to compare performance through indicators. The indicator describes the states of a particular action and assesses a specific success level. Usually, indicators are set to the highest achieved performance presented in the industry which it is involves in determining. Several known industrial dimensions have their own set of indicators, such as quality, time and cost. Each indicator is used to measure specific criterion accurately and they are mainly used in the benchmarking. Commonly, benchmarking and indicators are used by managers to evaluate different aspects of a specific item, operation or procedure. The benchmarking process helps in identifying performances developed in a particular lab and provides some guidelines for the comparing processes. Additionally, it directs in performing method inspections, product testing and the evaluating procedures.

The ToRCH framework lists robot capabilities and defines their performance levels. The assigned levels could be set as indicators for benchmarking especially if these levels could be expanded into a full specification of standardized tests to determine if the level has been achieved or not. Technical or product benchmarking can be used to ensure robots match user needs.

These standardized tests then could, for example, indicate the capability of a sophisticated robot by setting its capability levels to the highest available in the industry. The indicators could be set as benchmarks to describe an outstanding and exceptional performance, which can be used to clearly differentiate between performances and to set some capabilities goals for future development. Moreover, these capabilities indicators elucidate which capability performance levels are considered the excellent and which capability performance levels are considered borderline. Therefore, the ToRCH levels could be used as indicators and benchmarks to compare robot capabilities, to estimate robot development and to track its achievement.

The proposed benchmarking scheme contains the comprehensive ToRCH huge list of capabilities and their levels of maturities, known as capabilities scores, which are captured through series of capabilities questions. The proposed scheme also contains other measurements that are related to the context of each capability, known as parameters, but are not covered in full details in this thesis. All of these three elements are proposed to be illustrated in a numeric format to clarify the complexity of the robot in a simple clear representation. And as these scores are captured for each section of ToRCH, according to the ToRCH structure, they could be combined together to generate a matrix with several sections and layers. This matrix contains an extraordinarily comprehensive collection of capabilities scores that could obtain further complex capability calculations but are presented in a simple single dashboard. Furthermore, this dashboard could be arranged differently by shifting the sections or the layers of the ToRCH structure, or by presenting the required sections and dropping others to simplify managing and analyzing robot capability. Therefore, the ToRCH can be used as a basic tool to benchmark robot capability through its established framework to provide guidelines for creating a benchmarking dashboard.

8.4.3 Displaying robot capabilities through scaled circles

Robot capability profiles can be presented as scaled circles. Robot capabilities are divided into three types: physical, cognitive and social. Each of these types has its own set of sub-capabilities and is represented by its own circle. When the robot achieves the highest score in the levels of these sub-capabilities, it is shown as an un-scaled circle, a cycle with a full parameter. To calculate the scaling of each circle according to the level scored, the total size of the circle was first divided by the number of sub-capabilities. Next, the score achieved by the robot is divided by the maximum achievable score in each sub-capability. This value is then multiplied by the proportion allocated to that sub-capability. Each capability type circle is then scaled by this value.

The following example illustrates this process with the social robot Pepper. This robot is used for social abilities, as demonstrated by its capability profile in Figure 8.1 and presented in the scaled diagram, Figure 8.2.

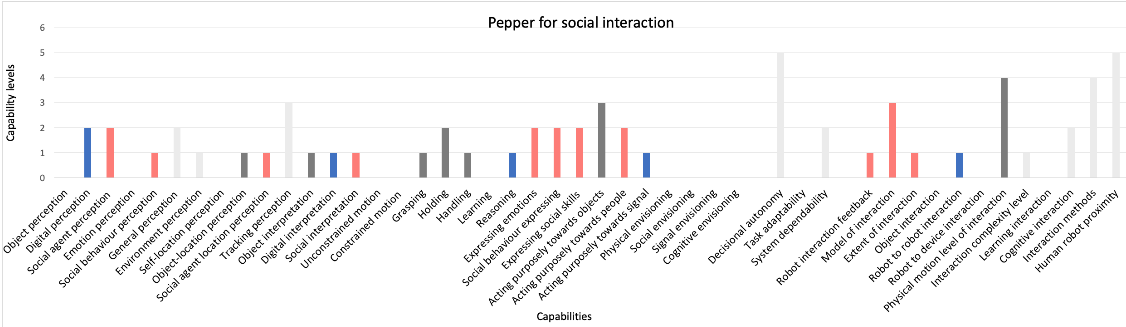

Figure 8.1: The diagram captures the capabilities of the social robot Pepper. It presents the capabilities levels in different colors. The blue present cognitive capabilities, gray present physical capabilities and red present social capabilities.

8.4.4 Determining artificial intelligence (AI) in robots

AI mathematical techniques are present in machine learning, deep learning, big data and natural language programming (NLP). The definition of AI can cause confusion as it does not refer to the computational hypothesis of the field, and therefore could be called 'computational intelligence' (Poole et al., 1998). Thus, while machine learning, deep learning, big data, and NLP are part of AI, not all AI is computational intelligence, as shown in the use of AI in robotics. On the contrary, computational intelligence also includes neural, fuzzy and evolutionary methods. In general, AI in robots can refer to techniques used in the programming of the robot capabilities, presenting the interactions of robots as expressed in human or animal terms or presenting the behaviors and skills expressed by the robot to capture its external abilities. Therefore, referring to AI purely in terms of mathematical and computational techniques creates confusion around AI aspects of robotics. The ability of ToRCH to illustrate robotics AI helps in describing and clarifying which form of AI is being referred to. To help in understanding AI in robots, the different forms of AI are illustrated in several locations within the ToRCH framework, as presented in Figure 8.3.

The most common form of AI, referred to in this research as computational intelligence, can be captured as a capability such as reasoning, learning, acting purposefully and envisioning. Some of these AI areas are in speech recognition (speech to text and text to speech), image recognition (such as face recognition), handwriting recognition, computer vision and other areas that depend on sensing data. These AI aspects can be presented as methods for perception and interpretation capabilities, and therefore allocated under perception and interpretation capabilities in the ToRCH framework. Some advanced interpretations are even modeled with realistic simulations.

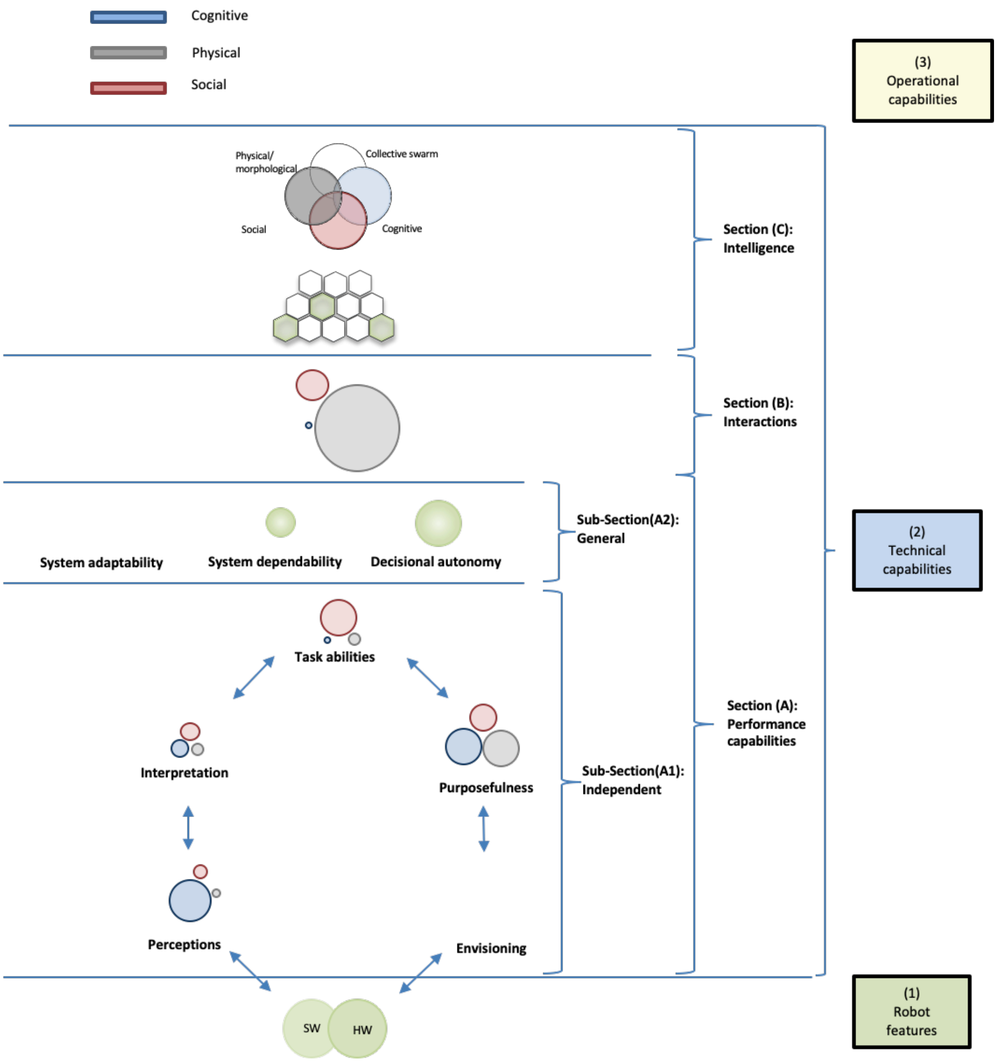

Figure 8.2: The diagram shows the Pepper capabilities as scaled circles. In the independent capability sub-section (A1): the cognitive perception is developed more than the social and physical, but the types of interpretation are almost the same, with a medium amount. The social task ability is far more developed than the physical and cognitive task abilities. The types of purposefulness tend to have high margins where envisioning capabilities do not exist in the robot. The figure also shows the amount of development for the general capabilities in sub-section (A2), interaction capabilities in section (B) as well as the number of skills the robot is developed to exhibit and its directions of intelligence which are captured in section (C).

Other AI aspects emphasize how natural intelligence is processed, which is known as natural language processing or knowledge representation, such as text generation, question answering, context extraction and machine translation. These AI simulations capture the

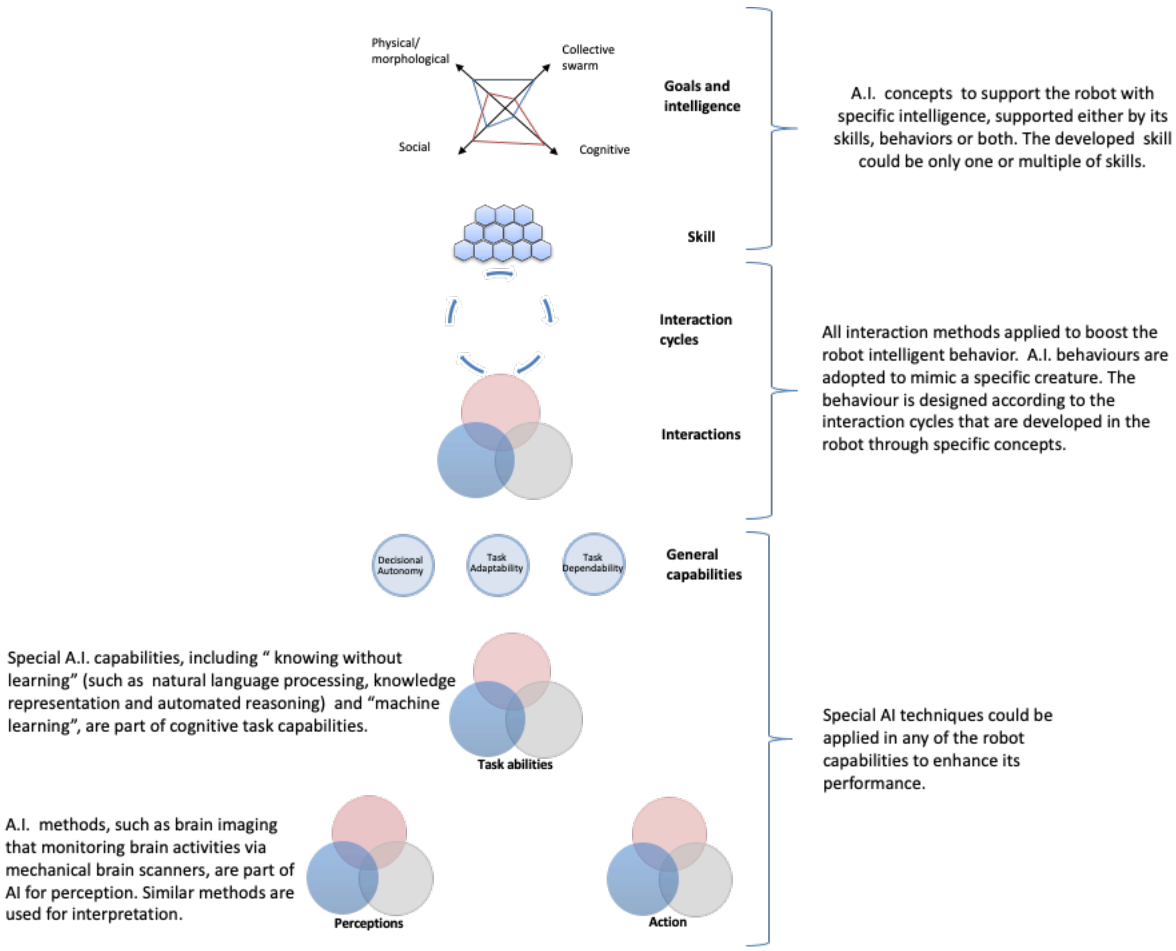

Figure 8.3: The AI aspect is allocated in several locations within the ToRCH framework, to capture mathematical techniques or methods that were developed in the robot capabilities, skills and behavior. The AI aspect also could be as in the learning AI aspects or in its AI expressions.

learning aspects of the robot by using machine learning and deep learning. Therefore, these AI aspects are allocated to the cognitive task ability section in the ToRCH framework.

Another type of AI is captured through robot behavior (Russell and Norvig, 2016). The robot illustrates a specific skill or behavior though its interaction cycles, known as behavior-based intelligence (Goodrich and Schultz, 2007). This form of AI is allocated to the interaction/skill subsection of the intelligence capability. Such a robot is designed to demonstrate behavior that mimics a specific agent, which could be human or animal. This behavior is expressed through the three task abilities via robot interaction cycles and can be expressed through interaction with a social agent or defined through the Turing test; therefore, it can also be presented in the interaction layer of ToRCH.

There are also different actions or skills, such as specific bee or fish manoeuvres, that can be developed in the robot to support it with specific intelligence. Installing this AI aspect would present the robot with some intelligence, therefore, it can be presented in the intelligence layer of ToRCH.

Consequently, attempting to allocating AI within ToRCH involves describing the robot with more accurate explanations that are suitable for an artificial agent and follows the AI prospective. Listing these AI aspects on ToRCH clarifies the various subfields of AI that are applicable on robot, which will enhance the literature with accurate descriptions for roboticists when referring to the various AI aspects in their robots.

8.4.5 Determining autonomy

The capabilities levels that have been scored reflects the extent to which the capability is performed without human intervention. Capabilities scored at lower levels require more human intervention, where capabilities with high levels of scoring are performed with less human intervention. The higher the level of sophistication of a capability, such as a robot's ability to deal with flexible variables rather than one predefined variable, the more likely it is that the robot requires less human intervention in performing the capabilities. Therefore, high scoring of the technical maturity of the capability, indicates less human intervention and if the high scoring occurs on most of the ToRCH list of capabilities, it indicates that the robot has less human intervention which is a greater degree of autonomy. Also, less human intervention indicates a higher technical maturity of the capability, so if most of the ToRCH list of capabilities scores were low, it would indicate that the robot required more human intervention, thus had less autonomy. Alternatively, the degree of autonomy of the robot does not indicate the technical maturity levels of the capabilities. Some robots may have a high level of autonomy by performing the capability with low technical maturity level but performing the capabilities sequentially, which implies a high degree of autonomy. Therefore, the ToRCH framework illustrates the difference between the autonomy of each capability within the robot and the autonomy of the complete robotic system as a whole, as presented in Figure 8.4. Thus, the conceptual framework can help in defining the correlated relationship between the autonomy of a specific capability and the autonomy of the whole system. The relation would help in defining a matrix or a mathematical procedure to measure and calculate the autonomy of the robot according to the autonomy levels of its individual capabilities.

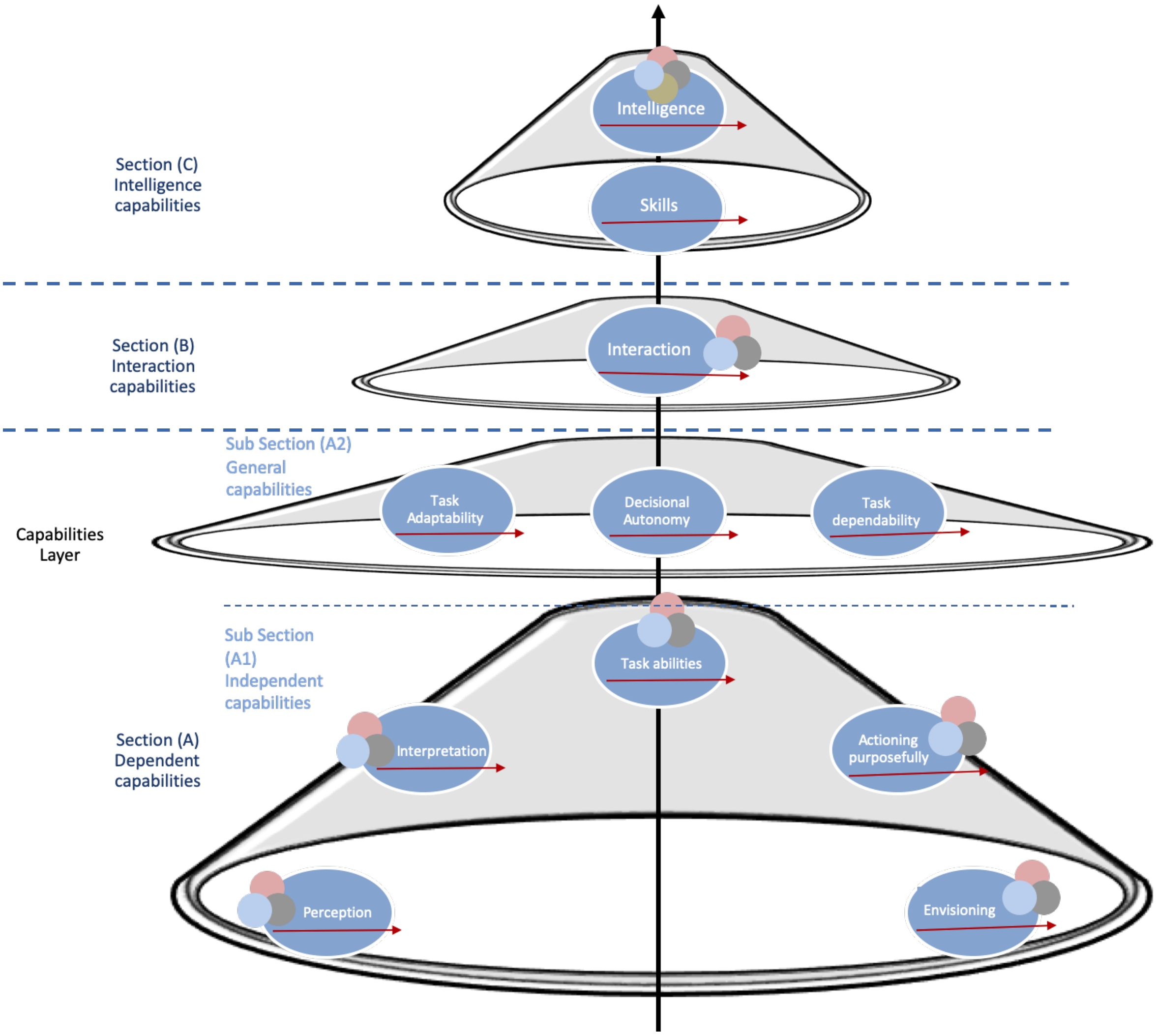

Figure 8.4: Autonomy levels for each of the robot capabilities, presented as horizontal red arrows, could be calculated for the whole robot. The figure also demonstrates the autonomy level of the robot as a complete single entity, shown as the black vertical arrow. /colorred Autonomy levels for each of the robot capabilities depend on their scores. In the figure they are presented as horizontal red arrows. They could be calculated for the whole robot and estimate its general autonomy, demonstrated as the black single vertical arrow. Mapping the autonomy levels of each capability to the existing known scheme of autonomy level of the robot is an important aspect that needs to be addressed in future work. This could be determined via a matrix that depends on the scores of the robot capabilities that is aligned with the autonomy level of the whole robot.

8.5 Future work

8.5.1 Determining the relations between hardware/software and robot capabilities

Some studies have considered the relation between the built-in features of the robot and its capability performance (including the technical capabilities and the operational capabilities). Defining these relations would help in choosing the robot's hardware and software to perform specific capabilities requirements to fulfill specific application demands (Hochgeschwender et al., 2013; Trenkwalder, 2019).

8.5.2 Determining more specific capabilities

There are more capabilities that could be developed in the robot, especially capabilities that belong to the operational layer. These capabilities can be allocated in ToRCH according to the types of the capability and the categories of ToRCH; physical, social and cognitive. The capability also needs to be categorized in levels, according to their specifications, to be allocated in the framework.

8.5.3 Determining more specific interactions

Interaction capabilities in ToRCH are divided into three categories: social interaction (interaction with a social agent), cognitive interaction (interaction with other robots or systems through data or signals), and physical interaction (interaction with physical objects in the environment) (Yanco and Drury, 2002). These three intersecting interactions include four sections between the categories: cognitive-social, physical-social, cognitive-physical, and social-cognitive-social. These sections are listed with some of the interaction classifications that can support the framework with any possible interaction type. Additionally, as the field of robotics develops and becomes more sophisticated, interactions would be more specific and detailed. These interaction classifications would be capable of encompassing them. The interaction sections are listed in Figure 8.5.

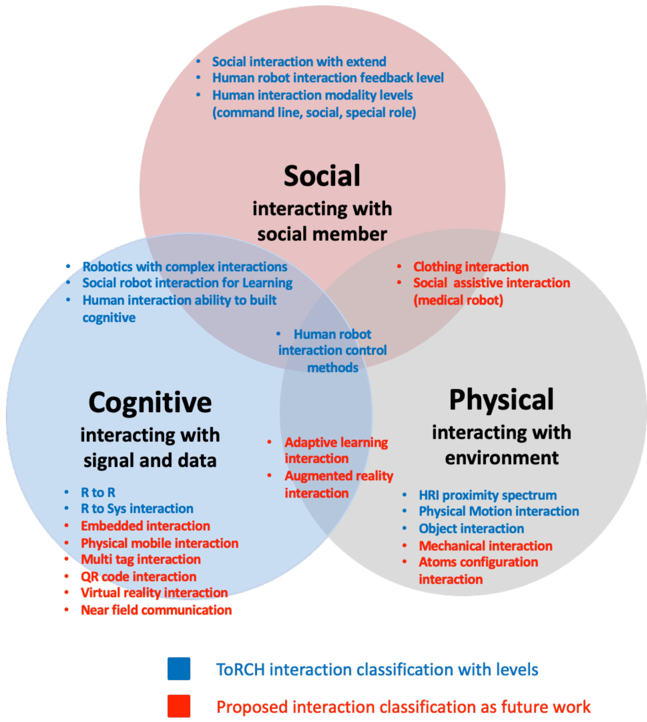

Figure 8.5: Distribution of the various robot interaction classifications in different sections of the interaction layer of ToRCH. The list of interactions contains some interactions that have already been developed for ToRCH research and other interactions types that need to be identified.

8.5.4 Determining parameters for interaction cycles

Capabilities are performed through interaction cycles to establish behavior in the robot and to demonstrate a special skill. Both skills and behaviors are illustrated in the ToRCH intelligence dimensions. Interactions parameters would specify interaction and describe how to modulate it. Parameters, such as the number of interaction partners, the complexity of the interaction or predictability of the environment (EU-Robotics, 2016), need to be described for each interaction cycle, as they capture more aspects that help in describing the interaction cycle.

An interaction cycle, known as ‘the robot-world feedback loop’ (Winfield, 2012), is the outcome of the performance of specific capabilities or sub-capabilities performed along with interactions for a specific task. In turn, the task can be complicated and include more than one cycle. Therefore, the cycle can be categorized as cognitive, social or physical; or it can be a mix of them, according to the performed interaction. The interaction cycle also can be either static or dynamic, as it illustrates the robot's behavior or action (Werger and Mataric, 2000) for a specific skill within a defined intelligence. For future work, these interaction capabilities parameters can be adapted to ToRCH to support the process of describing interactions and therefore identifying robots.

8.5.5 Determining more specific parameters for each of the capability levels

Some robotic research does not intend to increase the capabilities levels in the robot but they specify to improve several aspects that affect the performance of the capabilities levels. These aspects were presented earlier as measurable parameters, presented in section 4.4.2, as they contribute in defining and adjusting a specific level in a capability. They measure the efficiency of the level and what affects its performance. Determining and defining the parameters for each capability level, with scale and range, should identify, modulate and describe the level more clearly.

8.5.6 A statistical approach applied on ToRCH to describe robots capabilities

Applying a statistical procedure, such as multidimensional scaling, hierarchical cluster analysis and factor analysis, to ToRCH can provide quantitative descriptions to enhance visualization. It could define the differences and connections between robot groups according to their scored capabilities and also it also could establish new relations.

ToRCH generates multiple scores belonging to different dimensions. These scores could be lowered to improve interpretation. Lowering the scores or eliminating some dimensions are practical if the dissimilarity between the capability are preserved. The dissimilarities between scores have reasonable meanings, with the lower levels indicating the capability is simple with less performance and the scores with high levels indicates that the capability is executed with a high complicated performance. Protecting these interpretations while decreasing the ToRCH capabilities dimensions and it's capabilities levels requires specialized extraction method, such as factor analysis and multidimensional scaling, as they help in generating an understandable visualization.

Some multidimensional scaling requires that each capability should have the same number of levels, which is not the current case for ToRCH. In such cases, the scores could be normalized to a standardized range (e.g. 0-1). However, ToRCH capabilities are continually developing, and as they reach the same amount of levels for each capability, the scaling process could simply visualize the similarities and the differences between robots.

Another statistical visualization enhancer is the factor analysis. It could be applied to explore patterns in the ToRCH scores. For instance, it could define the consistent correlation among groups of scores and then replace the correlated scores by the underlying factors to reduce the numbers of the scores. The SPSS could receive the ToRCH scores as input and generate a patterned matrix with correlated groups after performing the Exploratory Factual Analysis (EFA). The underlying capabilities and their levels, also known as underlying factors, are the constructor of the generated matrix. Identifying and labeling the underlying factors of the capabilities scores should define the measuring theory for which specific capabilities move together, as some capabilities scores unitedly increase together or decrease together in unison. This collaboration depends on specific leading factors. Replacing these collaborated scores with the score of the shared factor reduces the list of scores of ToRCH into a smaller set of scores that can be visualized more easily. However, this analysis relies on collecting a large number of capability scores from different participants for various robots and could be future work.

8.5.7 Determining requirements

Defining the requirements aspects that are frequently used in robot applications and mapping them to the list of capabilities, is important for robot deployment. All of the robot requirements need to be listed and categorized accurately, in order to be matched and mapped to ToRCH robot capabilities. The main aspect of mapping between the two domains should answer the main common questions; “why, what, how, and which". These overview perspectives need to be specified for both domains, as illustrated in the Figure 8.6.

8.6 Final remarks

Developments in robotics occur on a daily basis, where each robot performs specific action and contains certain capabilities. The ToRCH framework allows a clear numeric description of robot capabilities. Attempting to combine different capabilities into one description without categorization and nesting will most likely lead to an inaccurate representation of the robot capabilities, an invalid or misleading evaluation of the robot performance, create confusion, especially if the sub-capability scores have higher values than the main

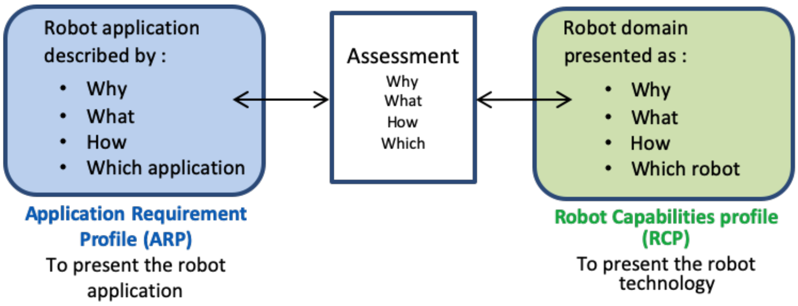

Figure 8.6: The mapping between the robot domain and the application domain through general aspects; such as "where", "why", "what", "how", and "which"; would smooth the assessment and help in using the mapping procedure. The "where " captures aspects in the operational layer, the "why" to captures the intelligence layer, the "what" to capture what capabilities needed to be performed, "how" to captures how the robot is performing interactions, and "which" identifies which application or robot " is used.

.

capabilities, or if the given score is for unavailable capabilities or sub-capabilities. Therefore, it is important for robot identification to capture all of its performed capabilities in a clear scoring scheme, as illustrated by the Robot Capabilities Profile (RCP), with a simple nested categorization presented via the "ToRCH" framework.

The ToRCH framework and its RCP helps in designing, developing, deploying and testing a robot for specific applications. They also facilitate dimensions to assess between the application domain and the robot domain. The framework was evaluated to confirm its ability to capture and score robot capabilities in detailed representation. By demonstrating this, the framework initiates a procedure to extract the capabilities of a robot and list their levels. Recognizing the capabilities levels of the robot outlines its performance and allows for defining it. It aids in determining the robot's capabilities, which to some extent helps answering the main question of this thesis "what is a robot?".

References

Adams, S., Arel, I., Bach, J., Coop, R., Furlan, R., Goertzel, B., Hall, J. S., Samsonovich, A., Scheutz, M., Schlesinger, M., et al. (2012). Mapping the landscape of human-level artificial general intelligence. *AI magazine*, 33(1):25–42.

Andina, D. (2007). Swarm intelligence and its applications in swarm robotics. *Conference on Computational Intelligence, Man-Machine Systems and Cybernetics*.

Anzalone, S. M., Boucenna, S., Ivaldi, S., and Chetouani, M. (2015). Evaluating the engagement with social robots. *International Journal of Social Robotics*, 7(4):465–478.

Appiah, C., Arndt, C., Siemsen, K., Heitmann, A., Staubitz, A., and Selhuber-Unkel, C. (2019). Living materials herald a new era in soft robotics. *Advanced Materials*, 31(36):1807747.

Arrick, R. and Stevenson, N. (2003). *N. Wiley Robot Building For Dummies*. John Wiley & Sons.

Awasthi, D. (2005). *Cryptogams: algae, bryophyta and pterldophyta*. Krishna Prakashan Media.

Baraka, K., Alves-Oliveira, P., and Ribeiro, T. (2019). An extended framework for characterizing social robots. *arXiv preprint arXiv:1907.09873*.

Bartneck, C. and Forlizzi, J. (2004). A design-centred framework for social human-robot interaction. *RO-MAN 2004. 13th IEEE International Workshop on Robot and Human Interactive Communication (IEEE Catalog No.04TH8759)*, 1(1):591–594.

Bashyal, S. and Venayagamoorthy, G. K. (2008). Human swarm interaction for radiation source search and localization. In *2008 IEEE Swarm Intelligence Symposium*, pages 1–8. IEEE.

Beer, J. M., Fisk, A. D., and Rogers, W. A. (2014). Toward a framework for levels of robot autonomy in human-robot interaction. *Journal of human-robot interaction*, 3(2):74–99.

Beevi, A., Wagner, S., Hallerstede, S., and Pedersen, C. F. (2015). Data quality oriented taxonomy of ambient assisted living systems. In *2015 IET International Conference on Technologies for Active and Assisted Living (TechAAL)*.

Beni, G. (2004). From swarm intelligence to swarm robotics. In *International Workshop on Swarm Robotics*, pages 1–9. Springer.

Berger, M., Seversky, L. M., and Brown, D. S. (2016). Classifying swarm behavior via compressive subspace learning. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5328–5335. IEEE.

Besio, S., Caprino, F., and Laudanna, E. (2010). Guidelines for using robots in education and play therapy sessions for children with disabilities. *Uniservice, Trento Italy*.

Bhatnagar, S., Alexandrova, A., Avin, S., Cave, S., Cheke, L., Crosby, M., Feyereisl, J., Halina, M., Loe, B. S., Martínez-Plumed, F., et al. (2017). Mapping intelligence: Requirements and possibilities. In *3rd Conference on Philosophy and Theory of Artificial Intelligence*, pages 117–135. Springer.

Bischoff, R., Kurth, J., Schreiber, G., Koeppe, R., Albu-Schäffer, A., Beyer, A., Eiberger, O., Haddadin, S., Stemmer, A., Grunwald, G., et al. (2010). The kuka-dlr lightweight robot arm-a new reference platform for robotics research and manufacturing. In *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*, pages 1–8. VDE.

Bisset, D. D. (2017). Personal communication.

Bodnera, J., Augustina, F., Wykypiela, H., Fisha, J., Muehlmanna, G., Wetscherb, G., and Schmida, T. (2005). The da Vinci robotic system for general surgical applications a critical interim appraisal. *Swiss Medical Weekly*.

Bodnera, J., Augustina, F., Wykypiela, H., Fisha, J., Muehlmanna, G., Wetscherb, G., and Schmida, T. (2011). *Perception-action cycle: models, architectures, and hardware*. Springer Science & Business Media.

Bonarini, A., Burgard, W., Fontana, G., Matteucci, M., Sorrenti, D. G., and Tardos, J. D. (2006). Rawseeds: Robotics advancement through web-publishing of sensorial and elaborated extensive data sets. In *In proceedings of IROS*, volume 6, page 93.

Bonsignorio, F. (2017). A new kind of article for reproducible research in intelligent robotics [from the field]. *IEEE Robotics & Automation Magazine*, 24(3):178–182.

Bonsignorio, F., Del Pobil, A. P., and Messina, E. (2014). Fostering progress in performance evaluation and benchmarking of robotic and automation systems [tc spotlight]. *IEEE Robotics & Automation Magazine*, 21(1):22–25.

Bostrom, N. (2012). The superintelligent will: Motivation and instrumental rationality in advanced artificial agents. *Minds and Machines*, 22(2):71–85.

Brambilla, M., Ferrante, E., Birattari, M., and Dorigo, M. (2013). Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41.

Breazeal (2017a). Personal communication.

Breazeal, C. (2003). Toward sociable robots. *Robotics and autonomous systems*, 42(3):167–175.

Breazeal, C. (2004). Social interactions in HRI: the robot view. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 34(2):181–186.

Breazeal, C. (2017b). Social robots: From research to commercialization. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '17, pages 1–1, New York, NY, USA. ACM.

Breazeal, C., Edsinger, A., Fitzpatrick, P., and Scassellati, B. (2001). Active vision for sociable robots. *IEEE Transactions on systems, man, and cybernetics-part A: Systems and Humans*, 31(5):443–453.

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*, 2(1):14–23.

Brooks, R. (1987). A hardware retargetable distributed layered architecture for mobile robot control. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, volume 4, pages 106–110. IEEE.

Brooks, R. A. (1991). Intelligence without representation. *Artificial Intelligence*, 47(1-3):139–159.

Bruce, A., Nourbakhsh, I., and Simmons, R. (2002). The role of expressiveness and attention in human-robot interaction. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, volume 4, pages 4138–4142. IEEE.

Burkhardt, R. W. (2005). *Patterns of behavior: Konrad Lorenz, Niko Tinbergen, and the founding of ethology*. University of Chicago Press.

Cabibihan, J.-J., Javed, H., Ang, M., and Aljunied, S. M. (2013). Why robots? a survey on the roles and benefits of social robots in the therapy of children with autism. *International Journal of Social Robotics*, 5(4):593–618.

Cameron, D., Millings, A., Fernando, S., Collins, E. C., Moore, R., Sharkey, A., Evers, V., and Prescott, T. (2018). The effects of robot facial emotional expressions and gender on child–robot interaction in a field study. *Connection Science*, 30(4):343–361.

Campion, G., Bastin, G., and D'Andrea-Novel, B. (1993). Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 462–469. IEEE.

Cangelosi, A. and Schlesinger, M. (2014). *Developmental Robotics: from babies to robots*. The MIT Press.

Chennareddy, S., Agrawal, A., and Karuppiah, A. (2017). Modular self-reconfigurable robotic systems: a survey on hardware architectures. *Journal of Robotics*, 2017.

Cherubini, A., Passama, R., Crosnier, A., Lasnier, A., and Fraisse, P. (2016). Collaborative manufacturing with physical human–robot interaction. *Robotics and Computer-Integrated Manufacturing*, 40:1–13.

Chrysostomou, d. (2017). Personal communication.

Chu, B., Jung, K., Han, C. S., and Hong, D. (2010). A survey of climbing robots: Locomotion and adhesion. *International Journal of Precision Engineering and Manufacturing*, 11(4):633–647.

Cowan, B. R., Doyle, P., Edwards, J., Garaialde, D., Hayes-Brady, A., Branigan, H. P., Cabral, J., and Clark, L. (2019). What's in an accent? the impact of accented synthetic speech on lexical choice in human-machine dialogue. *arXiv preprint arXiv:1907.11146*.

Da Graça Simões, M., de Freitas, M. C. V., and Rodríguez-Bravo, B. (2018). Theory of classification and classification in libraries and archives: Convergences and divergences. *KO KNOWLEDGE ORGANIZATION*, 43(7):530–538.

Dautenhahn, K. (1995). Getting to know each other—artificial social intelligence for autonomous robots. *Robotics and Autonomous Systems*, 16(2-4):333–356.

Dautenhahn, K. (2004). Robots we like to live with?! - a developmental perspective on a personalized, life-long robot companion. *RO-MAN 2004. 13th IEEE International Workshop on Robot and Human Interactive Communication (IEEE Catalog No.04TH8759)*, 1(1):17–22.

Dautenhahn, K. (2007). Socially intelligent robots: dimensions of human–robot interaction. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 362(1480):679–704.

Dautenhahn, K. (2017). Personal communication.

Dautenhahn, K. and Billard, A. (1999). Bringing up robots or—the psychology of socially intelligent robots: From theory to implementation. In *Proceedings of the Third Annual Conference on Autonomous Agents*, AGENTS '99, pages 366–367, New York, NY, USA. ACM.

De Santis, A., Siciliano, B., De Luca, A., and Bicchi, A. (2008). An atlas of physical human–robot interaction. *Mechanism and Machine Theory*, 43(3):253–270.

Delcomyn, F. (2007). Biologically inspired robots. In *Bioinspiration and Robotics Walking and Climbing Robots*. InTech.

Deutsch, C., Meneghini, C., Mermut, O., and Lefort, M. (2010). Measuring technology readiness to improve innovation management. In *Proc. of The XXI ISPIM Conference*.

DeVito, J. A. (2011). *Human communication: The basic course*. Pearson Higher Ed.

Devlin, K. (2018). *Turned On: Science, Sex and Robots*. bloomsbury Sigma.

Dewsbury, D. A. (1999). The proximate and the ultimate: past, present, and future. *Behavioural Processes*, 46(3):189–199.

Dobra, A. (2014). General classification of robots. Size criteria. In *Proc. of Robotics in Alpe-Adria-Danube Region (RAAD)*, pages 1–6. IEEE.

Doran, D. and Gokhale, S. S. (2011). Web robot detection techniques: overview and limitations. *Data Mining and Knowledge Discovery*, 22(1-2):183–210.

Doran, D. and Gokhale, S. S. (2012). A classification framework for web robots. *Journal of the American Society for Information Science and Technology*, 63(12):2549–2554.

Dragone, M., Duffy, B., and O'Hare, G. (2005). Social interaction between robots, avatars & amp; humans. *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005.*, pages 24–29.

Dudek, G., Jenkin, M. R., Milios, E., and Wilkes, D. (1996). A taxonomy for multi-agent robotics. *Autonomous Robots*, 3(4):375–397.

Duff, A., Lepora, N. F., Mura, A., Prescott, T. J., and Verschure, P. F. (2014). *Biomimetic and Biohybrid Systems: Third International Conference, Living Machines 2014, Milan, Italy, July 30–August 1, 2014, Proceedings*, volume 8608. Springer.

Duffy, B. R. (2003). Anthropomorphism and the social robot. *Robotics and Autonomous Systems*, 42(3-4):177–190.

Duffy, B. R., Dragone, M., and O'Hare, G. M. (2005). Social robot architecture: A framework for explicit social interaction. In *Android Science Towards Social Mechanisms, CogSci 2005 Workshop, Stresa, Italy*.

Duffy, B. R., Rooney, C., O'Hare, G. M., and O'donoghue, R. (2000). *The Social Robot*. PhD thesis, University College Dublin.

Duffy, B. R., Rooney, C. F. B., Hare, G. M. P. O., and Donoghue, R. P. S. O. (1999). What Is a Social Robot? *The 10th Irish Conference on Artificial Intelligence & Cognitive Science, University College Cork*, 1(1):1–3.

Edmonds, B. and Dautenhahn, K. (1999). Social intelligence. *Computational & Mathematical Organization Theory*, 5(3):199–201.

Engelhardt, K. and Edwards, R. (1992). Humanrobot integration for service robotics. In *Human-robot interaction*, pages 315–346. Taylor & Francis Ltd., London, UK.

Esser, K. (1982). *Cryptogams: cyanobacteria, algae, fungi, lichens*. CUP Archive.

EU-Robotics, A. (2016). *Robotics Multi-Annual Roadmap for Robotics in Europe, Horizon 2020*. SPARC Robotics, Brussels, Belgium.

Falcone, E., Gockley, R., Porter, E., and Nourbakhsh, I. (2003). The personal rover project:: The comprehensive design of a domestic personal robot. *Robotics and Autonomous Systems*, 42(3-4):245–258.

Farinelli, A., Iocchi, L., and Nardi, D. (2004). Multirobot systems: a classification focused on coordination. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(5):2015–2028.

FeilSeifer, D., Skinner, K., and Matarifa, M. (2007). Benchmarks for evaluating socially assistive robotics. *Interaction Studies: Social Behaviour and Communication in Biological and Artificial Systems*, 8(3):423–439.

Feinberg, A. W. (2015). Biological soft robotics. *Annual review of biomedical engineering*, 17:243–265.

Ferrante, E. (2013). *Information transfer in a flocking robot swarm*. PhD thesis, Université Libre de Bruxelles.

Fischer, K. (2017). Personal communication.

Fong, T., Nourbakhsh, I., and Dautenhahn, K. (2003). A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42(3-4):143–166.

Gardner, H. (2011). *Frames of mind: The Theory of Multiple Intelligences*. Hachette UK.

Gat, E., Bonnasso, R. P., Murphy, R., et al. (1998). On three-layer architectures. *Artificial intelligence and mobile robots*, 195:210.

Gilpin, K. and Rus, D. (2010). Modular robot systems. *IEEE Robotics & Automation magazine*, 17(3):38–55.

Gockley, R., Bruce, A., Forlizzi, J., Michalowski, M., Mundell, A., Rosenthal, S., Sellner, B., Simmons, R., Snipes, K., Schultz, A. C., et al. (2005). Designing robots for long-term social interaction. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1338–1343. IEEE.

Gockley, R., Forlizzi, J., and Simmons, R. (2006). Interactions with a moody robot. In *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pages 186–193. ACM.

Goldberg, D. (1996). Heterogeneous and homogeneous robot group behavior. In *Proceedings of The National Conference of Artificial Intelligence*, pages 1390–1390.

Goodrich, M. a. and Schultz, A. C. (2007). Human-robot interaction: A survey. *Foundations and Trends in Human-Computer Interaction*, 1(3):203–275.

Goodrich, M. A., Schultz, A. C., et al. (2008). Human robot interaction: a survey. *Foundations and Trends® in Human–Computer Interaction*, 1(3):203–275.

Gouaillier, D., Hugel, V., Blazevic, P., Kilner, C., Monceaux, J., Lafourcade, P., Marnier, B., Serre, J., and Maisonnier, B. (2009). Mechatronic design of NAO humanoid. In *2009 IEEE International Conference on Robotics and Automation*, pages 769–774.

Grayson, K. and Rust, R. (2001). Interrater reliability, iv. *Journal of Consumer Psychology*, 10(1&2):71–73.

Hancock, A. (2013). Best practice guidelines for developing international statistical classifications. expert group meeting on international statistical classifications.

Hanson, D., Baurmann, S., Riccio, T., Margolin, R., Dockins, T., Tavares, M., and Carpenter, K. (2009). Zeno: A cognitive character. In *AI Magazine, and Special Proc. of AAAI National Conference, Chicago*.

Hasse, C. and Bulgheroni, M. (2017). Reeler- responsible ethical learning with robotics, els ethics - should society be afraid of robots? Workshops in European Robotics Forum (ERF) in group Ethical, Legal, & Societal (ELS) No.1.

He, W., Li, Z., and Chen, C. P. (2017). A survey of human-centered intelligent robots: issues and challenges. *IEEE/CAA Journal of Automatica Sinica*, 4(4):602–609.

Helding, L. (2009). Howard gardner's theory of multiple intelligences. *Journal of Singing*, 66(2):193.

Hewett, D. (1998). The nature of human communication. *Human Communication research*, 1(1):3–8.

Heylighen, F. (1999). Collective intelligence and its implementation on the web: algorithms to develop a collective mental map. *Computational & Mathematical Organization Theory*, 5(3):253–280.

Hobby (2016). Types of robots. Retrieved from http://www.hobby.net.au/flex/types-of-robots/1135/1.

Hochgeschwender, N., Gherardi, L., Shakhirmardanov, A., Kraetzschmar, G. K., Brugali, D., and Bruyninckx, H. (2013). A model-based approach to software deployment in robotics. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3907–3914. IEEE.

Huang, C.-M. and Mutlu, B. (2014). Multivariate evaluation of interactive robot systems. *Autonomous Robots*, 37(4):335–349.

International Federation of Robotics, I. (2019). Standardization, robot definitions at iso. Retrieved from https://ifr.org/standardisation.

Jacob, E. K. (2004). Classification and categorization: a difference that makes a difference. *Library Trends*, 52(3):515–540.

Kahn Jr, P. H., Ishiguro, H., Friedman, B., Kanda, T., Freier, N. G., Severson, R. L., and Miller, J. (2007). What is a human?: Toward psychological benchmarks in the field of human–robot interaction. *Interaction Studies*, 8(3):363–390.

Kanda, T. and Ishiguro, H. (2004). A development approach for socially interactive humanoid robot. In *Conference on Development and Learning*, page 38. Citeseer.

Kidd, C. D. and Breazeal, C. (2004). Effect of a robot on user perceptions. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 4, pages 3559–3564. IEEE.

Kidd, C. D., Taggart, W., and Turkle, S. (2006). A sociable robot to encourage social interaction among the elderly. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 3972–3976. IEEE.

Kihlstrom, J. F. and Cantor, N. (2000). Social intelligence. In Sternberg, R. J., editor, *Handbook of Intelligence*, pages 359–379. Cambridge University Press.

Kirby, R., Forlizzi, J., and Simmons, R. (2010). Affective social robots. *Robotics and Autonomous Systems*, 58(3):322–332.

Kok, J. N., Boers, E. J., Kosters, W. A., Van der Putten, P., and Poel, M. (2009). Artificial intelligence: definition, trends, techniques, and cases. *Artificial intelligence*, 1.

Kube, C. R. and Zhang, H. (1993). Collective robotics: From social insects to robots. *Adaptive Behavior*, 2(2):189–218.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174.

Lawler-Dormer, D. (2013). Baby X: digital artificial intelligence, computational neuroscience and empathetic interaction. In *Proceedings of the 19th International Symposium on Electronic Art*. ISEA International.

Lee, J., Kim, D., Ryoo, H.-Y., and Shin, B.-S. (2016). Sustainable wearables: wearable technology for enhancing the quality of human life. *Sustainability*, 8(5):466.

Lee, M. K., Kiesler, S., and Forlizzi, J. (2010). Receptionist or information kiosk: how do people talk with a robot? In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, pages 31–40. ACM.

Legg, S., Hutter, M., et al. (2007). A collection of definitions of intelligence. *Frontiers in Artificial Intelligence and Applications*, 157:17.

Lemaignan, S., Garcia, F., Jacq, A., and Dillenbourg, P. (2016). From real-time attention assessment to with-me-ness in human-robot interaction. In *The Eleventh ACM/IEEE International Conference on Human Robot Interaction*, pages 157–164. IEEE Press.

Lenzi, T., Vitiello, N., De Rossi, S. M. M., Persichetti, A., Giovacchini, F., Roccella, S., Vecchi, F., and Carrozza, M. C. (2011). Measuring human–robot interaction on wearable robots: A distributed approach. *Mechatronics*, 21(6):1123–1131.

Linjawi, M. and Moore, R. K. (2017). A proposed structure to capture theoperational and technical capabilities ofdifferent robots. In *UK-RAS (Robotics and Autonomous Systems)*, pages 127–129. UK-RAS.

Linjawi, M. and Moore, R. K. (2018). Towards a comprehensive taxonomy for characterizing robots. In *Annual Conference Towards Autonomous Robotic Systems*, pages 381–392. Springer.

Linjawi, M. and Moore, R. K. (2019a). Evaluating ToRCH structure for characterizing robots. In *Annual Conference Towards Autonomous Robotic Systems*, pages 319–330. Springer.

Linjawi, M. and Moore, R. K. (2019b). ToRCH app. https://github.com/MLinjawi/ToRCH.

Linnaeus, C. (1758). *Systema naturae*, volume 1. Stockholm Laurentii Salvii.

ManagementMania (2016). Thorndikes intelligence theory. Retrieved from https://managementmania.com/en/thorndikes-intelligence-theory.

Marshall, J. (2017). Personal communication.

Martınez-Plumed, F., Loe, B. S., Flach, P., O hEigeartaigh, S., Vold, K., and Hernández-Orallo, J. (2018). The facets of artificial intelligence: a framework to track the evolution of AI. https://www.ijcai.org/proceedings/2018/718.

Mavridis, N. (2015). A review of verbal and non-verbal human–robot interactive communication. *Robotics and Autonomous Systems*, 63:22–35.

McGrenere, J. and Ho, W. (2000). Affordances: Clarifying and evolving a concept. In *Graphics interface*, pages 179–186.

Mitchinson, B., Martin, C. J., Grant, R. A., and Prescott, T. J. (2007). Feedback control in active sensing: rat exploratory whisking is modulated by environmental contact. *Proceedings of the Royal Society B: Biological Sciences*, 274(1613):1035–1041.

Mitchinson, B. and Prescott, T. J. (2016). Miro: A robot "mammal" with a biomimetic brain-based control system. In *Conference on Biomimetic and Biohybrid Systems*, pages 179–191. Springer.

Moore, R. K. (1993). Section 1: Users guide. *EAGLES Handbook of Standards and Resources for Spoken Language Systems. Speech Research Unit, DRA Malvern, Worcs., U.K.*, 1(1):1–28.

Moore, R. K. (2000). *The future of speech-based services: bringing in the benefit*. COST249 Workshop on Voice Operated Telecom Services (VOTS2000), Gent-Belgium.

Moore, R. K. (2016). Introducing a pictographic language for envisioning a rich variety of enactive systems with different degrees of complexity. *International Journal of Advanced Robotic Systems*, 13(2):74.

Morimoto, Y., Onoe, H., and Takeuchi, S. (2018). Biohybrid robot powered by an antagonistic pair of skeletal muscle tissues. *Science Robotics*, 3(18):eaat4440.

Morrow, J. D. and Khosla, P. K. (1997). Manipulation task primitives for composing robot skills. In *Proceedings of International Conference on Robotics and Automation*, volume 4, pages 3354–3359 vol.4.

Müller, V. C. and Hoffmann, M. (2017). What is morphological computation? on how the body contributes to cognition and control. *Artificial life*, 23(1):1–24.

Nickerson, R. C. (1997). A taxonomy of collaborative applications. In *Proceedings of the AIS 1997 Americas Conference on Information Systems*, volume 52, pages 560–562.

Olszewska, J. I., Barreto, M., Bermejo-Alonso, J., Carbonera, J., Chibani, A., Fiorini, S., Goncalves, P., Habib, M., Khamis, A., Olivares, A., et al. (2017). Ontology for autonomous robotics. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 189–194. IEEE.

Oswald, R. K. (1980). Positioning system employing feedforward and feedback control. US Patent 4,200,827.

Pal, H. R., Pal, A., and Tourani, P. (2004). Theories of intelligence. *Everyman's Science*, 39(3):181–192.

Parker, L. E. (2008). Distributed intelligence: Overview of the field and its application in multi-robot systems. *Journal of Physical Agents*, 2(1):5–14.

Penichet, V. M. R., Marin, I., Gallud, J. A., Lozano, M. D., and Tesoriero, R. (2007). A classification method for cscw systems. *Electronic Notes in Theoretical Computer Science*, 168:237–247.

Pfeifer, R. (2000). On the role of morphology and materials in adaptive behavior. In *From Animals to Animats 6: Proceedings of the 6th International Conference on Simulation of Adaptive Behavior*, volume 1, pages 24–32.

Pfeifer, R., Marques, H. G., and Iida, F. (2013). Soft robotics: the next generation of intelligent machines. In *Twenty-Third International Joint Conference on Artificial Intelligence*.

Pfeifer, R. and Scheier, C. (2001). *Understanding intelligence*. MIT press.

Polaszek, A. (2010). *Systema Naturae 250-The Linnaean Ark*. CRC Press.

Polygerinos, P., Correll, N., Morin, S. A., Mosadegh, B., Onal, C. D., Petersen, K., Cianchetti, M., Tolley, M. T., and Shepherd, R. F. (2017). Soft robotics: Review of fluid-driven intrinsically soft devices; manufacturing, sensing, control, and applications in human-robot interaction. *Advanced Engineering Materials*, 19(12):1700016.

Poole, D. L., Mackworth, A. K., and Goebel, R. (1998). *Computational Intelligence: A Logical Approach*, volume 1. Oxford University Press New York.

Prescott, T. J., Diamond, M. E., and Wing, A. M. (2011). Active touch sensing.

Prescott, T. J., Redgrave, P., and Gurney, K. (1999). Layered control architectures in robots and vertebrates. *Adaptive Behavior*, 7(1):99–127.

Prestes, E., Carbonera, J. L., Fiorini, S. R., Jorge, V. A., Abel, M., Madhavan, R., Locoro, A., Goncalves, P., Barreto, M. E., Habib, M., et al. (2013). Towards a core ontology for robotics and automation. *Robotics and Autonomous Systems*, 61(11):1193–1204.

Rastic-Dulborough, O. (2014). Internet of things (IoT) and human computer interaction. Technical report, University of Southampton.

Rich, C., Ponsler, B., Holroyd, A., and Sidner, C. (2010). Recognizing engagement in human robot interaction. *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, 1(1):375–382.

Richard, P. (1981). *Robot Manipulators: mathematics, programming, and control: the computer control of robot manipulators*. MIT Press.

Robeyns, I. (2003). The capability approach: an interdisciplinary introduction. In *Training course preceding the Third International Conference on the Capability Approach, Pavia, Italy*.

Robotics, E. (2014). Strategic research agenda for robotics in europe 2014–2020. *Available at: eu-robotics. net/cms/upload/topic groups/SRA2020 SPARC. pdf (accessed 21.4. 2018)*.

Robotnik (2018). Robotnik automation s.l.l. corporation. Retrieved from https://www.robotnik.eu.

Rogers, T. E., Peng, J., and Zein-Sabatto, S. (2005). Modeling human-robot interaction for intelligent mobile robotics. In *Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on*, pages 36–41. IEEE.

Rus, D. and Tolley, M. T. (2015). Design, fabrication and control of soft robots. *Nature*, 521(7553):467–475.

Russell, S. J. and Norvig, P. (2016). *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,.

Saffiotti, A. (2017). Personal communication.

Saksida, L. M., Raymond, S. M., and Touretzky, D. S. (1997). Shaping robot behavior using principles from instrumental conditioning. *Robotics and Autonomous Systems*, 22(3-4):231–249.

Salichs, M. A., Barber, R., Khamis, A. M., Malfaz, M., Gorostiza, J. F., Pacheco, R., Rivas, R., Corrales, A., Delgado, E., and Garcia, D. (2006). Maggie: A Robotic Platform for Human-Robot Social Interaction. *2006 IEEE Conference on Robotics Automation and Mechatronics*, pages 1–7.

Salter, T., Michaud, F., Létourneau, D., Lee, D., and Werry, I. P. (2007). Using proprioceptive sensors for categorizing human-robot interactions. In *2007 2nd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 105–112. IEEE.

Schaefer, K. E., Billings, D. R., and Hancock, P. A. (2012). Robots vs. machines: Identifying user perceptions and classifications. In *Proc. of Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*, pages 138–141. IEEE.

Schmitz, M. (2011). Concepts for life-like interactive objects. *Proceedings of the Fifth International Conference on Tangible, embedded, and embodied interaction*, 1(1):157.

Scholtz, J. (2003). Theory and evaluation of human robot interactions. In *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*, pages 10–pp. IEEE.

Schulte, J., Rosenberg, C., and Thrun, S. (1999). Spontaneous, short-term interaction with mobile robots. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, volume 1, pages 658–663. IEEE.

Scott-Phillips, T. C., Dickins, T. E., and West, S. A. (2011). Evolutionary theory and the ultimate–proximate distinction in the human behavioral sciences. *Perspectives on Psychological Science*, 6(1):38–47. PMID: 26162114.

Severinson-Eklundh, K., Green, A., and Hüttenrauch, H. (2003). Social and collaborative aspects of interaction with a service robot. *Robotics and Autonomous Systems*, 42(3-4):223–234.

Shibata, T. (2004). An overview of human interactive robots for psychological enrichment. *Proceedings of the IEEE*, 92(11):1749–1758.

Shinozawa, K., Reeves, B., Wise, K., Lim, S., Maldonado, H., and Naya, F. (2003). Robots as new media: A cross-cultural examination of social and cognitive responses to robotic and on-screen agents. *Proceedings of of Annual Conference of Internation Communication Association*, 1(1):998–1002.

Siciliano, B. and Khatib, O. (2008). *Springer handbook of robotics*. Springer.

Sidner, C. L., Lee, C., Kidd, C. D., Lesh, N., and Rich, C. (2005). Explorations in engagement for humans and robots. *Artificial Intelligence*, 166:140–164.

Sita, E., Studley, M., Dailami, F., Pipe, A., and Thomessen, T. (2017). Towards multimodal interactions: Robot jogging in mixed reality. In *Proc. of the 23rd ACM Symposium on Virtual Reality Software and Technology*, page 78. ACM.

Sokal, R. R. (1974). Classification: purposes, principles, progress, prospects. *Science*, 185(4157):1115–1123.

Stead, L., Goulev, P., Evans, C., and Mamdani, E. (2004). The emotional wardrobe. *Personal and Ubiquitous Computing*, 8(3-4):282–290.

Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A., and Goodrich, M. (2006). Common metrics for human-robot interaction. In *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pages 33–40. ACM.

Stoelen, M. F., de Tejada, V. F., Huete, A. J., Balaguer, C., and Bonsignorio, F. P. (2015). Distributed and adaptive shared control systems: Methodology for the replication of experiments. *IEEE Robotics & Automation Magazine*, 22(4):137–146.

Thrun, S. (2004). Toward a framework for human-robot interaction. *Human-Computer Interaction*, 19(1):9–24.

Trenkwalder, S. M. (2019). Computational resources of miniature robots: Classification and implications. *IEEE Robotics and Automation Letters*, 4(3):2722–2729.

Trianni, V. (2017). Personal communication.

Tzafestas, S. (2015). *Sociorobot World. A Guided Tour for all.* Springer.

Vander Aalst, W. M., Bichler, M., and Heinzl, A. (2018). *Robotic process automation*. Springer.

Veres, S. M. (2014). Autonomous robotic systems. *Lecture Notes in Control and Information Sciences, Department of Automatic Control and Systems Engineering, The University of Sheffield*.

Vessey, S. and Drickamer, L. (2010). Integration of proximate and ultimate causes. In *Encyclopedia of Animal Behavior*, pages 180 – 185. Academic Press.

Wainer, J., Feil-Seifer, D. J., Shell, D. A., and Mataric, M. J. (2006). The role of physical embodiment in human-robot interaction. In *ROMAN 2006-The 15th IEEE International Symposium on Robot and Human Interactive Communication*, pages 117–122. IEEE.

Walker, R. (2017). Personal communication.

Wang, L. and Iida, F. (2015). Deformation in soft-matter robotics: A categorization and quantitative characterization. *IEEE Robotics & Automation Magazine*, 22(3):125–139.

Webb, B. (2001). Can robots make good models of biological behaviour? *Behavioral and brain sciences*, 24(6):1033–1050.

Werger, B. B. and Mataric, M. J. (2000). Broadcast of local eligibility: behavior-based control for strongly cooperative robot teams. In *Proceedings of the Fourth International Conference on Autonomous Agents*, pages 21–22. ACM.

Wilson, E. D., Assaf, T., Pearson, M. J., Rossiter, J. M., Anderson, S. R., and Porrill, J. (2013). Bioinspired adaptive control for artificial muscles. In *Conference on Biomimetic and Biohybrid Systems*, pages 311–322. Springer.

Wilson, E. D., Assaf, T., Pearson, M. J., Rossiter, J. M., Dean, P., Anderson, S. R., and Porrill, J. (2015). Biohybrid control of general linear systems using the adaptive filter model of cerebellum. *Frontiers in neurorobotics*, 9:5.

Winfield, A. (2012). *Robotics: A Very Short Introduction*. Oxford University Press.

Winfield, A. F. T. (2017). How intelligent is your intelligent robot? arXiv preprint arXiv:1712.08878.

Wood, L. (2015). Service robots: The next big productivity platform. https://robohub.org/service-robots-the-next-big-productivity-platform.

Wundt, W. M. (1874). *Grundzüge der physiologischen Psychologie*, volume 1. W. Engelman.

Yan, Z., Fabresse, L., Laval, J., and Bouraqadi, N. (2015). Metrics for performance benchmarking of multi-robot exploration. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3407–3414. IEEE.

Yanco, H. A. and Drury, J. L. (2002). A taxonomy for human-robot interaction. In *Proc. of AAAI Fall Symposium on Human-Robot Interaction*, pages 111–119.

Yim, M., Shen, W.-M., Salemi, B., Rus, D., Moll, M., Lipson, H., Klavins, E., and Chirikjian, G. S. (2007). Modular self-reconfigurable robot systems [grand challenges of robotics]. *IEEE Robotics & Automation Magazine*, 14(1):43–52.

Yousef, H., Boukallel, M., and Althoefer, K. (2011). Tactile sensing for dexterous in-hand manipulation in robotics—A review. *Sensors and Actuators A: physical*, 167(2):171–187.

Ziemke, T. (2003). What's that thing called embodiment? In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 25.

Appendix A

ToRCH profile questions

1. Please select the level of object perception in your robot? Adopted from MAR.
   - Level 0 - No object recognition.
   - Level 1 - Feature detection: sense data is gathered from a region of the environment such that the data has a spatial component and can be mapped to a model of that region. The richness of the sense data is such that it is possible to apply a feature detection process to create a set or sets of features that persist.
   - Level 2 - Object detection: multiple persistent features can be grouped to build models of distinct objects allowing objects to be differentiated from each other and from the environment.
   - Level 3 - Object recognition - single instance: object models created from sense data can be matched to specific known instances of an object with a reliability that is appropriate to the task.
   - Level 4 - Object recognition - one of many: object models created from sense data can be matched to one of a number of specific instances of known objects with a reliability that is appropriate to the task.
   - Level 5 - Parameterised object recognition: object models created from sense data can be matched to a number of known, parametrised object types. The settings for the parameters (e.g. size ratio, curvature, joint position etc.) can be deduced from the sensed object model. Note that in conjunction with single instance recognition ability this implies the ability to recognise a known (possibly learned) instance of a generic object, for example a particular brand of canned drink based on the generic recognition of a drinks can shape.
   - Level 6 - Context based recognition: the system is able to use its knowledge of context or location to improve its ability to recognise objects by reducing ambiguities through expectations based on location or context.
   - Level 7 - Object variable recognition: the system is able to recognise objects where there is a degree of variability in the object that approaches the scale of the object. For example, many generic objects such as coffee mugs vary in shape size and colour.
   - Level 8 - Novelty recognition: the system is able to recognise novelty in a known object, or parametrised object type. For example, a known mug where the handle is missing or broken.

- Level 9 - Unknown object categorisation (rigid): the system is able to assess an unknown rigid object based on sense data and deduce properties that are relevant to the task.
- Level 10 - Object property detection: it is possible to use sense data and the derived object model to deduce the properties of an object. For example analysis of the sense data may provide surface texture information, knowledge about de-formability, or the content of an object.
- Level 11 - Flexible object detection: the system is able to detect the shape and form of objects that are deformable and generate parametrised models of flexible objects. This includes articulated objects and objects with flexible and rigid components.
- Level 12 - Flexible object classification: the system is able to classify flexible objects by their properties and parameters. It is able to recognise specific known objects relevant to the task with an appropriate level of reliability.

2. Please select the (cognitive) perception level to capture digital data, signal or information? excluding object, social agent, emotion, social behaviour, self location, environment scene, object location and tracking perception. Developed for ToRCH and adapted from MAR.

    - Level 0 - No perception for digital data, signal or information information.
    - Level 1 - digital data, signal or information (feature) detection: sensing data/signal/information (such as bits, letters, an acoustic signal, graphics, etc.), gathered from a specific location and mapped to a meaningful model.
    - Level 2 - digital data, signal or information detection: the system make sense of data to recognise a specific topic based on feature extraction. The multiple persistent features can be grouped to build models of distinct types of information allowing information of different types to be differentiated from each other.
    - Level 3 - Information recognition - single instance: the information models created from sense data can be matched to specific known instances of information with a reliability that is appropriate to the task.
    - Level 4 - Information recognition - one of many: the information models created from sense data can be matched to one of a number of specific instances of known information with a reliability that is appropriate to the task.

3. Please select the level of social agent perception? Developed for ToRCH and adapted from MAR.

    - Level 0 - No human social agent recognition.
    - Level 1 - Feature detection to sense an aspect of a human social agent: sense data is gathered from the environment, and can be mapped onto to a human profile model. The richness of the sense data is such that it is possible to apply a feature detection process to create a set or sets of features of a person.
    - Level 2 - Human social agent detection of a whole social agent: multiple persistent human features can be grouped to build distinct human profile allowing humans to be differentiated from each other and from the environment.

- Level 3 - Person recognition - single instance: a human social agent model created from sense data can be matched against a specific human social agent model in the database with a reliability that is appropriate to the task.
- Level 4 - Person recognition - one of many: matching a person to one of several defined possible human social agent identities. The human models created from sense data can be matched against several known identities with a reliability that is appropriate to the task.
- Level 5 - Parameterised person recognition: recognising a generic human social agent by matching the person feature with known parameters. The models created from sense data can be matched to a number of known human social agent parameters. The settings for the parameters (e.g. size ratio, gender specification etc.) can be deduced from the sensed human social agent model.
- Level 6 - Context based recognition: improved human social agent recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognise human social agents by reducing ambiguities through expectations based on context or location.
- Level 7 - Person variable recognition: the system is able to recognise a human social agent where there are variations in his/her profile such as haircut and skin colour, for example, recognising a person with a change in some variables according to a specific scale.
- Level 8 - Novelty recognition, the ability to recognise novelty in a human social agent. The system is able to recognise a specific feature in a person that may have changed. For example, identify a specific scar on a person.

4. Please select the level of emotion perception? Developed for ToRCH and adapted from MAR: level 1 was not included as emotions has no features.
    - Level 0 - No emotion perception.
    - Level 1 - Emotion detection: the robot is able to sense an aspect of emotion. Sense data is gathered from the social agent and can be mapped. The richness of the sense data is applied to detect some features of social agent emotions.
    - Level 2- Emotion recognition - single instance: an emotional model created from sense data can be matched against a specific model in the database.
    - Level 3 - Emotion recognition - one of many: matching an emotional model to one of several defined possible models. The emotional models created from sense data can be matched against several known identified models with a reliability that is appropriate to the task.
    - Level 4 - Parameterised emotional recognition: recognising a generic emotion by matching the emotional feature with known parameters. The emotional models created from sense data can be matched to a number of known emotional parameters. The settings for the parameters can be deduced from the sensed model.
    - Level 5 - Context based emotion recognition: improved emotion recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognise emotion by reducing ambiguities through expectations based on context or location.
5. Please select the level of social behaviour perception? Developed for ToRCH and adapted from MAR.
    - Level 0 - No behaviour perception.

- Level 1 - Social behaviour feature detection: the robot is able to sense an aspect of social behaviour. Sense data is gathered from the social agent and can be mapped. The richness of the sense data is applied to detect some features of social behaviour.
- Level 2 - Social behaviour recognition - single instance: a social behaviour model created from sense data can be matched against a specific model in the database.
- Level 3 - Social behaviour recognition - one of many: matching a social behaviour model to one of several defined possible models. The social behaviour models created from sense data can be matched against several known identified models with a reliability that is appropriate to task.
- Level 4 - Parameterised social behaviour recognition: recognising social behaviour by matching the behaviour with known parameters. The social behaviour models created from sense data can be matched to a number of known behaviour parameters. The settings for the parameters can be deduced from the sensed model.
- Level 5 - Context based social behaviour recognition: improved social behaviour recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognise social behaviour by reducing ambiguities through expectations based on context or location.

6. Please select the level of general perception capabilities of the robot? Adopted from MAR.

   - Level 0 - No external perception: some robots do not sense their environment but simply carry out sets of pre-programmed moves triggered by a starting event.
   - Level 1 - Direct single or multiple parameter output sensing: a robot uses sensors that provide a single, or multiple parameter outputs directly. For example, a distance sensor, or a contact sensor. The robot utilises these outputs to directly alter behaviour within an operating cycle.
   - Level 2 - Low-level processing parameter sensing: a robot system may use fixed and known markers in the environment to indicate objects or way-points (e.g. Bar-codes, reflective strips etc.). The detection of these markers provides triggers to alter or switch between behaviours or sequences of behaviours
   - Level 3 - Multi-parameter perception: a robot uses multiple single parameter sensors to create a unified model of the environment. Sensed data can be collected from multiple types of sensor as well as multiple sensors of the same type. Each sensor contributes information to the model. The model is used to alter the behaviour of the system.
   - Level 4 - Feature-based perception: sense data is gathered from a region of the environment such that the sense data has a spatial mapping. The richness of the sense data information content is such that it is possible to apply feature extraction to the sense data and thereby interpret the content of the sense data as a set or sets of features. The system performs a data reduction with an assumption about the expected features. The presence of features is used to alter behaviour.
   - Level 5 - Grouped feature detection: the sense data gathered from the environment can be processed such that features can be aggregated to capture linkages between features. A group of features may relate to the same real object in the environment, but where the object has not been identified. The characteristics of the feature group can be used to alter the behaviour of the system. For example, a set of features of the same colour that move in the same way may relate to a pink ball.

- Level 6 - Object identification: the system can identify objects or coherent entities that it has detected in the scene through sets of grouped features and can use this identification to alter the system behaviour. The importance of this level is that a data source or a priori object model is required.
- Level 7 - Property identification: the system is able to deduce the properties of objects in the scene or scene itself and utilise those properties within system behaviour.
- Level 8 - Hidden state identification: the system is able to infer properties of an object, person or scene that are not directly observable. The scene and objects are not fully available in data sources ahead of time and scene interpretation and classification is required.

7. Please select the level of environment scene perception? Adopted from MAR.
    - Level 0 - No scene perception: the robot does not need to be able to interpret the environment in order to carry out its task.
    - Level 1 - Basic feature detection: the robot is able to detect features in the environment that relate to static structures in that environment.
    - Level 2 - Static structures: the robot is able to identify static structures in the environment in a way that is appropriate to the task.
    - Level 3 - Combined structures: the system is able to provide a consistent interpretation of the static structures in the environment over time. For example, it is able to identify the floor, walls and ceiling of a room and apply these as physical constraints to a model.
    - Level 4 - Multiple object detection: the system is able to delineate multiple objects from the static environment where there may be partially occluded with respect to the sense data gathered. For example, it is able to delineate objects on the floor of a room.
    - Level 5 - Object arrangement detection: the system is able to detect arrangements of objects, for example, objects in a stack or mixed in a receptacle and identify the relationships between objects with a success appropriate to the task. For example, a chair with books on it and a wine glass on top of the books.
    - Level 6 - Dynamic object detection: the system is able to detect an object that is moving within a static environment.

8. Please select the level of self-location perception (self-localization)? Adopted from MAR.
    - Level 0 - No perception of its self-location neither in terms of its position relative to the environment nor with respect to the relative position of its own structure.
    - Level 1 - Actuator position: the robot knows where its own mechanical structures are because of an assessment of the position of each of its actuators. For example, a platform can assess its own position based on the amount its wheels have turned.
    - Level 2 - External beacons: the robot knows its own location as a result of information derived from the inspection of external beacons. Beacons may be active or passive and include global beacons.
    - Level 3 - Relative location: the system is able to calculate its own location relative to its previous or future location with a degree of accuracy that is sufficient for the task.

- Level 4 - Feature-based location: the system calculates its position within an environment based on the motion of fixed features in the environment. For example by using SLAM to build and maintain a local map.
- Level 5 - Mapped location: the robot is able to relate its own position to a map that it has been given or that it has acquired. This may be a location within a task-relevant space.
- Level 6 - Spatial Occupancy: the system calculates the position of its own mechanical structures based on indirectly gathered sense data (i.e. Sense data gathered other than from the motion control system). This provides a spatial notion of occupancy.
- Level 7 - Object coupled location: the system is able to calculate the position of its own mechanical structures in conjunction with objects it is connected to. For example an object that is being gripped by the robot, or the position of the user in an assistive task.

9. Please select the level of object-location perception? Developed for ToRCH and adapted from MAR self location perception but did not include level 1 the actuator position.

   - Level 0 - No perception of object-location neither in terms of its position relative to the environment nor with its own location.
   - Level 1 - Object position: the robot identifies the object location as a result of object perception.
   - Level 2 - External beacons: the robot identifies the object location as a result of information derived from the inspection of external beacons. Beacons may be active or passive and include global beacons.
   - Level 3 - Relative location: the system is able to calculate the object location relative to a specific defined location with a degree of accuracy that is sufficient for the task.
   - Level 4 - Feature-based location: the system calculates the object position within an environment based on the motion of fixed features in the environment. For example by using SLAM to build and maintain a local map.
   - Level 5 - Mapped location: the robot is able to relate the object location position to a map that it has been given or that it has acquired. This may be a location within a task-relevant space.
   - Level 6 – Spatial occupancy: the system calculates the position of the object structures based on indirectly gathered sense data to provide a spatial notion of occupancy.
   - Level 7 - Object coupled location: the system is able to calculate the position of an object in conjunction with other objects it is connected to. For example, an object that is being gripped by the robot.

10. Please select the level of social agent location perception? Developed for ToRCH and adapted from MAR from self location perception but did not include level 1 the actuator position.

    - Level 0 - No perception of social agent-location nether in terms of its position relative to the environment nor with its own location.
    - Level 1 - Social agent position: the robot identifies the social agent location as a result of perception.

- Level 2 - External beacons: the robot identifies the social agent location as a result of information derived from the inspection of external beacons. Beacons may be active or passive and include global beacons.
- Level 3 - Relative location: the system is able to calculate the social agent location relative to a specific defined location, with a degree of accuracy that is sufficient for the task.
- Level 4 - Feature-based location: the system calculates the social agent position within an environment based on the motion of fixed features in the environment. For example by using SLAM to build and maintain a local map.
- Level 5 - Mapped location: the robot is able to relate the social agent location position to a map that it has been given or that it has acquired. This may be a location within a task-relevant space.
- Level 6 – Spatial Occupancy: the system calculates the position of the social agent structures based on indirectly gathered sense data to provide a spatial notion of occupancy.
- Level 7 - Object coupled location: the system is able to calculate the position of social agent in conjunction with other objects it is connected to.

11. Select the levels of tracking perception ability? Adopted from MAR.

    - Level 0 - No tracking: ability to carry out tasks without any tracking ability.
    - Level 1 - Tracked feature perception: features detected in the sense data are tracked over time. The tracking of features is used to build internal models of the environment. The tracking of markers in the environment is equivalent to tracking derived features.
    - Level 2 - Static object tracking: it is possible to track a detected object. The detected location of the object can be maintained with a reliability and accuracy that is compatible with the task.
    - Level 3 - Dynamic object tracking: is possible to identify an object and track it using sense data. As the object moves the system is able to disambiguate the motion of the robot from the motion of the object.
    - Level 4 - Tracking object shape: it is possible to track an object as it changes shape during the execution of a task. This may represent changes due to processes being applied to the object, or because it can be articulated.
    - Level 5 - Flexible object tracking: it is possible to identify a flexible or deformable object and track it.
    - Level 6 - Animate object tracking: it is possible to identify and track an animate object and extract the pose of the object.

12. Please select the robot's modes of perception to define the data collection methods used by the robot to perform the perceptive capabilities? Developed for ToRCH and adapted from MAR. Developed for ToRCH.

    - Visual.
    - Auditory
    - Olfactory

- Physical (Mechanical, Magnetic, Chemical, Others.)
- I do not know in which level the robot would be located

13. Please select the level of the physical object interpretive capability (object interpretation)? Adopted from MAR.

    - Level 0 - No interpretive ability: it does not need to interpret objects in the environment.
    - Level 1 - Fixed sensory interpretation: the robot has a fixed interpretation of the perceptions that occur because they are pre- categorised. For example, all sensed objects are applied to an occupancy grid and assumed to represent actual objects in the environment.
    - Level 2 - Basic environment interpretation: the robot uses sense data to interpret the environment into fixed notions of environmental space that are pre-categorised. For example, it will search for floor and wall segments in the sense data as these are relevant to its task even if the environment it is sensing has neither.
    - Level 3 - Object delineation: the robot is able to disambiguate objects from an interpretation of its static environment. The disambiguation of objects is based on built-in notions of object and environment. These notions may only be valid within a narrow operating context.
    - Level 4 - Object category interpretation: the robot is able to interpret the shapes and forms of objects based on categories of objects that are task relevant. It is able to interpret sense data to identify coherent instances of an object over a time scale appropriate to the task. Note that this ability level is particularly affected by the cognition ability parameters.
    - Level 5 - Structural interpretation: the robot is able to interpret perceptions so as to extract structural information from the environment. It is able to identify the structural relationships between objects in the environment.
    - Level 6 - Basic semantic interpretation: the robot is able to apply semantic tags to locations and objects allowing it to plan actions based on functional objectives that depend on the semantics of objects and locations.
    - Level 7 - Property interpretation: the robot is able to interpret perceptions to determine the properties of objects or locations in the environment.
    - Level 8 - Novelty interpretation: the robot is able to interpret perceptions to identify novelty in objects or locations.
    - Level 9 - Environmental affordances: the robot is able to interpret the environment in terms of what it affords. For example, it is able to interpret the ground conditions in a muddy field as being too unstable for the load it is carrying.

14. Select the robot's (cognitive) digital data, signal or information interpretive capability level (digital data, signal or information interpretation)? Developed for ToRCH and adapted from MAR.

    - Level 0 - No digital data, signal or information interpretation available
    - Level 1 - Fixed digital data, signal or information interpretation: the robot has a fixed interpretation of the perceived digital data, signal or information because they are pre-categorised

- Level 2 - Basic digital data, signal or information interpretation: the system performs interpretation of digital data, signal or information into fixed categorized notions. Digital data, signal or information are categorized according to features and properties pre-defined as types of notations
- Level 3 - Data delineation: the system is able to disambiguate some perceived digital data, signal or information from interpretation in order to understand the information type. The disambiguation is based on built-in predefined notations.
- Level 4 - digital data, signal or information category interpretation: the system is able to interpret digital data, signal or information based on pre-defined types that are relevant. This is performed by any of the lowest levels of the data mining techniques.
- Level 5 - Digital data or signal structural interpretation: the system is able to interpret digital data, signal or information perception and to extract structural concepts for it. It is able to identify structural relationships and create a base knowledge of it.
- Level 6 - Digital data, signal or information basic semantic interpretation: the system is able to apply semantic tags to digital data, signal or information allowing it to plan actions based on strategies and objectives of the information type and its semantics

15. Select the robot's social interpretive capability level? Developed for ToRCH and adapted from MAR.

    - Level 0 - No social interpretive ability: the robot does not need to interpret any social aspects of the identified agent.
    - Level 1 - Fixed social interpretation: the robot uses sense data to perform fixed interpretations of social aspects of the recognised social agent since the social aspects are pre-categorised.
    - Level 2 - Social delineation: the system is able to disambiguate specific aspects in the sense data, in order to understand the social aspect presented by the social agent. The disambiguation is based on built-in notation of social knowledge.
    - Level 3 - Social category interpretation: the system is able to interpret social actions based on categories that are related to the event.
    - Level 4 - Social structural interpretation: the system is able to interpret perceived social actions and extract the social concepts. It is able to identify social structures between people and their relationships.
    - Level 5 - Basic semantic interpretation: the system is able to apply semantic social tags to people allowing it to plan actions based on their personal objectives.

16. Please select the level of unconstrained motion capabilities in the robot, from location (A) to location (B)? Adopted from MAR to include motion either liner or rotatory motion.

    - Level 0 - No motion: All robots move in their environments, except fixed robots.
    - Level 1 - Pre-defined open loop motion: the robot carries out predefined moves in sequence. The motion is independent of the environment and events in the environment. The robot may not be able to maintain a position if subject to external forces, may be able to statically rest at a given position.

- Level 2 - Predefined closed-loop motion: the robot carries out predefined moves in the sequence where each motion is controlled to ensure position and/or speed goals are satisfied within some error bound. So, for example, a robot can move to and maintain a position (within some error margin) against forces less than the resultant motive force at the point of contact. A platform will similarly be able to execute fixed motions where the accuracy of these motions in the environment will depend on other abilities such as its perception ability.
- Level 3 - Open path motion: the robot can execute a motion that follows a path with a given path accuracy. This path is described by a specific point on the robot. The robot is able to return to any given point on the path with an accuracy that is appropriate to the task.
- Level 4 - Position constrained path motion: the robot can execute a path motion where the path is constrained by physical objects or by defined zones that must be avoided. For example, a robot arm that can operate through a physically constrained region such as a hole in a wall, or a platform that can move to avoid a known area of the environment such as a step-down. The robot is able to execute a path to an unvisited location obeying constraints.
- Level 5 - Force constrained path motion: the robot can execute a path motion while applying a specified force in a given direction related to the motion. For example, moving over the surface of an object while applying a force perpendicular to the surface as might be required when polishing a surface.
- Level 6 - Parametrised motion: the robot can execute a path move that optimises for a parameter. For example, a path that reduces energy consumption, covers an area or constrains the angle range of a joint, or the torque or force in a joint or linkage.
- Level 7 - Position constrained parametrised motion: the robot can operate through a physically constrained region while at the same time optimising a parameter or set of parameters that constrain the motions of the robot. For example, a robot arm may be able to reach a high shelf while maintaining a centre of gravity, or a platform robot operates in a room away from a charging station while optimising power usage.

17. Please select the level of constrained motion capabilities in the robot (motion with reaction to external forces)? Adopted from MAR to include motion either liner or rotatory

    - Level 0 - Un-reactive: the robot does not respond to external forces acting on it.
    - Level 1 - Compliant motion: the robot can execute motions that change in response to external forces applied to the robot such that the force exerted on the external body is controlled. The robot is able to maintain position and path in the absence of any external force. The force is working on the robot only at the intended tool tip, and the environment is static and rigid.
    - Level 2 - Reactive motion: the robot is able to react to externally applied forces contacting any part of the robot, not just at the intended tool-tip. The reaction may result in stiffening to resist the force or in lowering stiffness to reduce impact effect. The system is able to apply a force in a given direction and maintains that force against a rigid or semi-rigid body.
    - Level 3 - Soft medium motion: the robot is able to move into and within a soft medium, with passive dynamics. It is able to maintain a position and path within this medium while optimising motion and force parameters as demanded by the task.

- Level 4 - Multiple soft medium motions: the robot can move through multiple soft but passive environments, e.g., water and mud, during the same motion.
- Level 5 - Dynamic motion: the robot is able to alter its own dynamics of motion in response to multiple active external dynamic forces in order to optimise motion parameters; the robot can identify the interaction dynamics of the external forces.

18. Please select the level of manipulation-grasping capabilities in the robot? Adopted from MAR.

    - Level 0 - No grasping ability: many robots will not require the ability to grasp objects.
    - Level 1 - Simple pick and place: the robot is able to grasp an object at a known pre-defined location using a single predefined grasp action. The robot is then able to move or orient the object and finally un-grasp it. The robot may also use its motion ability to move the object in a particular pattern or to a particular location. Grasping uses open-loop control.
    - Level 2 - Known object pick and place: the robot is able to grasp a known object at a known pre-defined location using a predefined grasp action. The robot is then able to move or orient the object and finally ungrasp it. At the same time, the robot should ensure grasp stability, i.e., not accidentally lose the object even when moving. The robot may also use its motion ability to move the object in a particular pattern or to a particular location. Grasping uses open loop control.
    - Level 3 - Tolerant grasp: the robot is able to grasp a known object that is not located at an exact location, may have some orientation variation and is in the general location within the span of the gripper from some known location. Tolerance in the grasp action is able to absorb the difference in location or orientation. The operation is able to compensate for the differences in the picking location without affecting the required placement accuracy.
    - Level 4 - Tolerant grasp with sensors: the robot is able to grasp a known object that is not located at an exact location, may have some orientation variation and is in the general location within the span of the gripper from some known location. Tolerance in the grasp action is able to absorb differences in location or orientation. The operation is able to compensate for the differences in the picking location without affecting the required placement accuracy. The grasping uses sensors to control the grasping operation.
    - Level 5 - Location unknown pick: the robot is able to pick up a known object where the location and orientation of the object are not pre-defined. The robot may use perception ability to locate the object and decisional autonomy to plan and execute the grasp action in the context of the task.
    - Level 6 - Generic pick: the robot is able to pick up an object belonging to a certain parametrised type where the dimensions, location and orientation are unknown. The robot may use perception ability to locate the object and decisional autonomy to plan and execute the grasp action in the context of the task.
    - Level 7 - Complex object grasping: the robot is able to pick up an object belonging to a certain parameterised type where the object can be articulated, or consists of multiple separate parts.
    - Level 8 - Pick up unknown object: the robot is able to grasp a geometrically unknown object - an unknown object at a known pre-defined location selecting a grasp action on-line The robot is then able to move or orient the object and finally un-grasp it. During the whole operation, grasp stability must be guaranteed. The robot may also use its motion ability to move the object in a particular pattern or to a particular location.

19. Please select the manipulation-holding capabilities level of the robot? Adopted from MAR.
    - Level 0 - No holding ability: many robots will not require the ability to hold objects.
    - Level 1 - Simple holding of known object: the robot retains the object as long as no external perturbation of the object occurs.
    - Level 2 - Dynamic holding of known objectives: the robot can retain a grasp on a known object under some defined maximum level of external perturbation of the object.
    - Level 3 - Simple holding of modelled object: the robot can retain a grasp on an unknown but modelled object as long as there is no external perturbation of the object.
    - Level 4 - Dynamic holding of modelled object: the robot can retain a grasp on an unknown but modelled object under some defined maximum level of external perturbation of the object.
    - Level 5 - Holding unknown objects: the robot can dynamically adapt to the characteristics of the object and retain a grasp up to defined maximum levels of perturbation.
20. Please select the level of manipulation-handling capabilities in the robot? Adopted from MAR.
    - Level 0 - No handling ability: many robots will not require the ability to handle objects.
    - Level 1 - Simple release: the robot is able to release an object at a known pre-defined location, but the resulting orientation of the object is unknown. The object should not be prematurely released.
    - Level 2 - Moving to orientation: the object can be placed at a predefined place with a fixed orientation.
    - Level 3 - Variable placement: the robot is able to alter its placement action to accommodate small changes in the location of the destination for a picked object. For example, it is able to join two parts where the positional tolerance of the mating part is greater than the accuracy needed to place the part correctly. The placement variation is derived from sensor data online during the handling process. The robot may use decisional autonomy during placement.
    - Level 4 - Compliant placement: the robot is able to use compliance in the placement process to fit a picked part into a statically held part. For example the insertion of one part into another where the insertion forces vary during insertion as a result of friction. The robot may use perception ability and decisional autonomy during placement.
    - Level 5 - Positioning for placement: the robot is able to orient and align a known object and then place it within the context of a task
    - Level 6 - Generic positioning for placement: the robot is able to orient and align a known parameterised object where the dimensions, location and orientation and surface properties are unknown and place it appropriately in the context of the task.
    - Level 7 - Complex part placement: the robot is able to manipulate an object belonging to a certain parametrised type where the object can be deformable, fragile, articulated, or consists of multiple separate parts. The robot is able to exercise the articulations of the object or disassemble it within the context of a task.
    - Level 8 - Unknown object handling: the robot is able to determine the generic grasping properties of an unknown object. It is able to use those properties to determine how to handle and place the object. The robot may use perception ability and decisional autonomy during placement.

- Level 9 - Understanding object through handling: the robot is able to deduce properties of an object through handling of it. For example, if it contains a liquid, if it can be articulated, to determine its centre of gravity, or estimate its dimensions.

21. Please select the level of acquisition of knowledge levels (cognitive learning abilities) in the robot? Adopted from MAR.

    - Level 0 - No acquired knowledge: the robot does not acquire knowledge during its operation.
    - Level 1 - Sense data knowledge: the system is able to acquire knowledge about its environment based on sense data gathered moment to moment.
    - Level 2 - Persistent sense data knowledge: the system is able to accumulate knowledge about its environment based on sense data that persists during the execution of the current task.
    - Level 3 - Property knowledge: the system is able to acquire knowledge about the properties of objects in the environment by observation.
    - Level 4 - Deliberate acquisition: the system is able to acquire knowledge about the composition of its operating environment by executing actions that are deliberately designed to increase knowledge through exploration. For example to determine if a cup is full of liquid.
    - Level 5 - Place knowledge: the system is able to accumulate knowledge about the location and types of objects and environmental features in terms of matching objects to pre-defined and known types.
    - Level 6 - Knowledge scaffolding: the system has the ability to integrate embedded knowledge of objects and places with related knowledge gained from the environment.
    - Level 7 - Requested knowledge: the system is able to recognise that it has insufficient knowledge about an object or place relevant to the task and can formulate a question to gain that knowledge either from a person, or an external data source such as the internet or another robot.
    - Level 8 - Distributed knowledge: the system is able to communicate its gained knowledge to other robots or systems and can receive and integrate knowledge from other robots or systems.
    - Level 9 - Interaction acquisition: the system is able to acquire knowledge about its environment and objects within it through planned interactions with the environment and objects. For example, the robot deliberately selects an object of interest and picks it up to examine it more closely, putting it back where it picked it from.
    - Level 10 - Object function: the system is able to acquire knowledge about the function of objects in the environment. This knowledge may be acquired directly or indirectly through observation.
    - Level 11 - User knowledge: the system is able to acquire knowledge about the user by observation.
    - Level 12 - Critical feedback: the system is able to acquire knowledge about its actions by analysis of critical feedback that follows completion of the action.
    - Level 13 - Long-term observation: the system is able to distinguish between long-term and short-term changes in the environment and the objects within it.
    - Level 14 - Patterns of behaviour: the system is able to acquire knowledge about the patterns of behaviour of the user that relate to the task. For example, learning how to carry out an assembly process by observation.

- Level 15 - Observation learning: the system is able to acquire knowledge indirectly from observing other robots or people carrying out tasks.

22. Please select the reasoning cognitive abilities of the robot? Adopted from MAR.

    - Level 0 - No reasoning ability: can simply execute a pre-determined pattern of activity.
    - Level 1 - Reasoning from sense data: the robot is able to make basic judgements of sense data sufficient to allow actions to be controlled.
    - Level 2 - Pre-defined reasoning: the robot is able to use basic predefined knowledge about the environment to guide action and interaction.
    - Level 3 - Basic environment reasoning: the robot is able to use knowledge of the environment gained from perception in conjunction with stored knowledge to reason about the environment. For example, it can build a map of the environment and plot a path to a goal.
    - Level 4 - Reasoning with conflicts: the system is able to reason about the environment and objects when there is conflicting or incomplete information. For example missing sections of a map, or competing classifications for an object.
    - Level 5 - Dynamic reasoning: the system is able to reason about the perceived dynamics in the environment.
    - Level 6 - Safety reasoning: the system is able to reason about safety in the environment.
    - Level 7 - Task reasoning: the system is able to reason about the appropriate courses of action to achieve a task where there are alternative actions that can be undertaken. Typically the system will be able to identify the course of action which matches the desired task parameters, typically these involve time to completion, resource usage, or a desired performance level.
    - Level 8 - Task hypothesis: the system is able to reason about the priorities of different tasks within a mission and propose priorities based on its knowledge of the mission and the tasks. The system will be able to fix on a task that must be achieved but makes decisions about how to sequence tasks to achieve mission objectives.

23. Please select the level of emotion expression capability performed by the robot? Developed for ToRCH.

    - Level 0 - No emotion expression ability: the robot does not need to express emotions in performing their tasks.
    - Level 1 - Expressing a pre-defined emotion: the robot is able to execute a pre-defined emotion. The emotion is programmed to be executed to present the robot status.
    - Level 2 - Expressing a set of pre-defined emotions: the robot is able to execute pre-defined emotions. Emotions are programmed to be executed to present the robot status.
    - Level 3 - Decision based expression of emotion: the robot is able to alter its own emotions according to the social perception of another agent. Presenting different emotions as a social feedback action.
    - Level 4 - Parameterised expression of emotion: expressing a generic emotion by matching the emotion expression with known parameters. The emotion models created from sense data can be matched to a number of known emotion parameters. The settings for the parameters can be deduced from the sensed model.

- Level 5 - Sense driven expression of emotion: the robot is able to modulate its own emotions in proportion to perception-derived parameters. Emotion levels are expressed in different ways.

24. Please select the level of social behaviour expression performed by the robot? Developed for ToRCH.

    - Level 0 - No social behaviour ability: the robot does not need to express social behaviour in performing their tasks.
    - Level 1 - Expressing a pre-defined social behaviour: the robot is able to execute a pre-defined social behaviour. The social behaviour is programmed to be executed to present the robot status.
    - Level 2 - Expressing a set of pre-defined social behaviours: the robot is able to execute pre-defined social behaviours. Social behaviour is programmed to be executed to present the robot status.
    - Level 3 - Decision based expression of social behaviour: the robot is able to alter its own social behaviour according to the social perception of another agent. Presenting different social behaviour as a social feedback action.
    - Level 4 - Parameterised expression of social behaviour: expressing social behaviour by matching the social behaviour expression with known parameters. The social behaviour models created from sense data can be matched to a number of known social behaviour parameters. The settings for the parameters can be deduced from the sensed model.
    - Level 5 - Sense driven expression of social behaviour: the robot is able to modulate its own social behaviour in proportion to perception-derived parameters. Social behaviour levels are expressed in different ways.

25. Please select the level of expressing social skills capability performed by the robot (speech, dialogue, body language)? Developed for ToRCH.

    - Level 0 - No social skills ability: the robot does not need to express social skills in performing their tasks.
    - Level 1 - Expressing a pre-defined social skill: the robot is able to execute a pre-defined social skill. The social skill is programmed to be executed to present the status of the robot.
    - Level 2 - Expressing a set of pre-defined social skills: the robot is able to execute pre-defined social skills. Social skills are programmed to be executed to present the robot status.
    - Level 3 - Decision based expression of social skills: the robot is able to alter its own social skills according to the social perception of another agent. Presenting different social skills as a social feedback action.
    - Level 4 - Parameterised expression of social skills: expressing social skills by matching the social skill expression with known parameters. The social skills models created from sense data can be matched to a number of known social skills parameters. The settings for the parameters can be deduced from the sensed model.
    - Level 5 - Sense driven expression of social skills: the robot is able to modulate its own social skills in proportion to perception-derived parameters. Social skill levels are expressed in different ways.

26. Please select the level of the ability of the robot to act purposely towards objects? Adopted from MAR.

- Level 0 - No action ability: robots are defined by having some level of action on the environment
- Level 1 - Defined action: the robot executes fully pre-defined actions as a sequence of sub-actions. This sequence can repeat until stopped by an operator or other system events.
- Level 2 - Decision based action: the robot is alter its course of action based on perceptions or system events. It is able to select between a set of pre-defined actions based on decisional autonomy ability.
- Level 3 - Sense driven action: the robot is able to modulate its action in proportion to parameters derived from its perceptions. The perceptions are used to drive the selection of pre-defined actions or the parameters of pre-defined actions.
- Level 4 - Optimised action: the robot is able to alter the sub-task sequence it applies to the execution of a task in response to perceptions or a need to optimise a defined task parameter.
- Level 5 - Knowledge-driven action: the system is able to utilise knowledge gained, from perceptions of the environment including objects within it, to inform actions or sequences of action. Knowledge is gained either by accumulation over time or through the embedding of knowledge from external sources, including user input that associate properties with perceptions.
- Level 6 - Plan driven actions: the system is able to use accumulated information about tasks to inform its plans for action.
- Level 7 - Dynamic planning: the system is able to monitor its actions and alter its plans in response to its assessment of success.
- Level 8 - Task action suggestions: the system is able to suggest tasks that contribute to the goals of specific mission.
- Level 9 - Mission proposals: the system is able to propose specific missions that align with high-level objectives.

27. Please select the level of the ability of the robot to act purposely towards people or social agent? Developed for ToRCH.

    - Level 0 - No action ability: robots are defined by having some level of action towards social agent.
    - Level 1 - Defined social action: the robot executes fully pre-defined social actions as a sequence of social sub-actions. This social sequence actions can repeat until stopped by an operator or other system events.
    - Level 2 - Decision based social action: the robot is alter its course of social action based on perceptions or social events. It is able to select between a set of pre-defined social actions based on its decisional autonomy ability.
    - Level 3 - Sense driven social action: the robot is able to modulate its level of social action in proportion to social parameters derived from its perceptions. The perceptions are used to drive the selection of pre-defined levels of social action.
    - Level 4 - Optimised social action: the robot is able to alter the social sub-task sequence it applies to the execution of a social actions in response to perceptions or a need to optimise a defined social action parameter.

- Level 5 - Knowledge-driven social action: the robot is able to use knowledge from social perception to determine the social action or sequence of actions to be performed. Knowledge is gained either by accumulation over time or through the embedding of knowledge from external sources, including user input that associate properties with perceptions.
- Level 6 - Plan driven social actions: the system is able to use accumulated information about social actions to inform its plans for action.
- Level 7 - Dynamic social planning: the system is able to monitor its social actions and alter its plans for social actions in response to its assessment of success.
- Level 8 - Task social action suggestions: the system is able to suggest social actions that contribute to the goals of specific mission.
- Level 9 - Mission proposals: the system is able to propose social missions that align with high-level social objectives.

28. Please select the level of the ability of the robot to act purposely towards information, digital data and signal? Excluding objects and social agents. Developed for ToRCH.

    - Level 0 - No action ability: robots are defined by having some level of action towards digital data, signal or information.
    - Level 1 - Defined digital data, signal or information action: the robot executes fully pre-defined actions as a sequence of sub-actions. This sequence of actions towards digital data, signal or information can repeat until stopped by an operator or other system events.
    - Level 2 - Decision-based action: the robot is able to alter its actions on digital data, signal or information based on perceptions or events.
    - Level 3 - Sense driven digital data, signal or information action: the robot is able to modulate its level of action in proportion to parameters derived from its digital data, signal or information perceptions. The perceptions are used to drive the selection of pre-defined levels of digital data, signal or information action.
    - Level 4 - Optimised digital data, signal or information action: the robot is able to alter the sub-task sequence it applies to the execution of a digital data, signal or information actions in response to perceptions or a need to optimise a defined digital data, signal or information action parameter.
    - Level 5 - Knowledge-driven digital data, signal or information action: the robot is able to use knowledge from digital data, signal or information perception to determine the digital data, signal or information action or sequence of actions to be performed. Knowledge is gained either by accumulation over time or through the embedding of knowledge from external sources, including user input that associate properties with perceptions.
    - Level 6 - Plan driven digital data, signal or information actions: the system is able to use accumulated information about digital data, signal or information actions to inform its plans for action.
    - Level 7 - Dynamic digital data, signal or information planning: the system is able to monitor its digital data, signal or information actions and alter its plans of action in response to its assessment of success.

- Level 8 - Task digital data, signal or information action suggestions: the system is able to suggest digital data, signal or information actions that contribute to the goals of specific mission.
- Level 9 - Mission proposals: the system is able to propose digital data, signal or information missions that align with high-level objectives.

29. Please select the level of the physical envisioning ability of the robot? Adopted from MAR.

    - Level 0 - No envisioning ability: the robot is not able to predict subsequent states of physical actions.
    - Level 1 - Motion prediction: the robot is able to project the effect of its motion to predict short term local interactions with detected objects in the environment. The robot only has the ability to predict its motion with respect to static objects.
    - Level 2 - Dynamic motion prediction: the robot is able to project the effect of its motion to predict short term interactions with both static and dynamic objects in the environment that the system can detect.
    - Level 3 - Function projection: the system is able to project the effect of its function onto the local environment in order to be able to assess its effectiveness. For example, a robot may assess the coverage of a room it has cleaned in order to identify areas it has missed.
    - Level 4 - Rigid interaction prediction: the system is able to envision the effect of its planned actions on rigid objects and structures that it has identified. For example, it is able to predict how an object will behave when grasped in a particular way.
    - Level 5 - Flexible object interaction: the system is able to envision the effect its planned actions will have on flexible objects that it has parametrized.
    - Level 6 - Basic environment envisioning: the system is able to observe events in the environment that relate to the task and envision their impact on the actions of the robot.
    - Level 7 - Envisioning safety: the system is able to assess the safety implications on users of observed events occurring in the working environment.
    - Level 8 - Envisioning user responses: the system is able to envision the actions of a user responding to events in the environment.

30. Please select the social action envisioning ability level of the robot? Developed for ToRCH and adapted from MAR.

    - Level 0 - No social envisioning ability: the robot is not able to predict subsequent states of social actions.
    - Level 1 - Social prediction: the robot is able to project the effect of its social action to predict short-term local interactions with detection of social members in the environment. The robot only has the ability to predict its social actions with respect to other available social agents.
    - Level 2 - Dynamic social prediction: predict its social action with respect to static object available in the environment and also with the respect of the existence of social agent.
    - Level 3 - Rigid social interaction prediction: the system is able to envision the effect of its planned social actions on identified social agents. For example, it is able to predict how an agent will behave when socially acting in a particular way.

- Level 4 - Basic environment social envisioning: the system is able to observe social events in the environment that relate to its performance and envision the impact of the social event on the actions of itself and other agents.
- Level 5 - Envisioning social user responses: the system is able to envision the actions of a social action by an agent responding to events. In other words: the ability to predict other agents' social actions on specific occasions.

31. Please select the robot's level of envisioning ability towards digital data, signal or information? Excluding objects and social agents. Developed for ToRCH and adapted from MAR.
    - Level 0 - No envisioning ability: the robot has no prediction of subsequent defined digital data, signal or information.
    - Level 1 - digital data, signal or information prediction: the robot is able to predict consequences of defined digital data, signal or information in short term.
    - Level 2 - Long time digital data, signal or information prediction: the robot is able to predict consequences of its defined digital data, signal or information on the long-term and present how would it impact the users.
    - Level 3 - digital data, signal or information projection: predict the effect of digital data, signal or information on specific events.
    - Level 4 - Rigid digital data, signal or information prediction: predict the effect of defined digital data, signal or information on identified agents.
32. Please select the cognitive envisioning ability level of the robot? Developed for ToRCH and adapted from MAR.
    - Level 0 - No cognitive envisioning ability: the robot has no prediction of subsequent of cognitive actions such as learning and reasoning.
    - Level 1 - Cognitive prediction: the robot is able to predict consequences of its cognitive (learning and reasoning) actions in short term.
    - Level 2 - Long term cognitive prediction: the robot is able to predict consequences of its cognitive (learning and reasoning) actions in the long-term and present, how would it impact the users.
    - Level 3 - Cognitive function projection: predict the effect of its cognitive (learning and reasoning) activities on specific events.
    - Level 4 - Rigid cognitive interaction prediction: predict the effect of its planned cognitive actions on identified agents.
    - Level 5 - Flexible cognitive interaction: predict the effect of its planned reasoning and learning towards agents.
33. Please select the level of general action capability performed by the robot? Developed for ToRCH.
    - Level 0 - No general action ability: the robot does not perform general action in their tasks.
    - Level 1 – Expressing a pre-defined general action: the robot is able to perform a pre-defined general action

- Level 2 – Expressing a set of pre-defined general actions: the robot is able to execute pre-defined general actions.

34. Select the level of decisional autonomy of the robot? Adopted from MAR.

    - Level 0 - No autonomy: any robots exhibit a degree of autonomy. For each capability it performs there are a degree of carrying the task without a user intervention.
    - Level 1 - Basic action: a robot that executes a sequence of actions that are unaffected by the environment and makes decisions based on the locations of actuators to proceed to the next action step
    - Level 2 - Basic decisional autonomy: the robot makes decisions based on basic perceptions and user input and chooses its behaviour from predefined alternatives.
    - Level 3 - Continuous basic decisional autonomy: the system alters the parameters of a behaviour in response to continuous input from perceptions, or based on input control from a user interacting continuously with the system. The system may be able to override or ignore user input when certain criteria are encountered.
    - Level 4 - Simple autonomy without environment model: the system uses perception to make moment to moment decisions about the environment and so controls interaction with the environment in order to achieve a predefined task.
    - Level 5 - Simple autonomy with environment model: the system uses perception to make moment to moment decisions about the environment and so controls interaction with the environment in order to achieve a predefined task. The decisions made take into account an internal model of the environment.
    - Level 6 - Task autonomy: the system utilises its perception of the environment to sequence different sub-tasks to achieve a higher level task. For example, cleaning a room based on a self-constructed room map where it returns to areas that have been missed and to a recharging station when the battery runs low. The events that cause behavioural changes are external and often unpredictable.
    - Level 7 - Constrained task autonomy: the system adapts its behaviour to accommodate task constraints. These might be negative impacts in terms of failed sensors, or the need to optimise power utilisation or other physical resources the process depends on, (water, chemical agents, etc.). Alternatively these might be constraints imposed by sensing ability, the environment or the user.
    - Level 8 - Multiple task autonomy: the system chooses between multiple high-level tasks and can alter its strategy as it gathers new knowledge about the environment. Will also take into account resource limitations and attempt to overcome them.
    - Level 9 - Dynamic autonomy: the system is able to alter its decisions about actions (sub-tasks) within the time frame of dynamic events that occur in the environment so that the execution of the task remains optimal to some degree.
    - Level 10 - Mission oriented autonomy: the system is able to dynamically alter its tasking both within and between several high level tasks in response to dynamic real time events in the environment.

- Level 11 - Distributed autonomy: the source for task and mission decisions can originate from outside of the system. The system is able to balance requests for action with its own tasking and mission priorities and can similarly communicate requests for action.

35. Select the level of task adaptability of the robot? Adopted from MAR.

    - Level 0 - No adaptation: the system does not alter its task or operating behaviour in response to experience gained over time.
    - Level 1 - Recognition of the need for adaptation: the system recognises that the performance of a particular task could be optimised according to some metric, but no adaptation is performed.
    - Level 2 - Single task adaptation: a single task performed during the process cycle is adapted over time to optimise a particular metric. This adaptation is achieved by strategic overview of the performance of the system while carrying out the task. Adaptation is the result of accumulated experience.
    - Level 3 - Multiple task adaptation: A set of tasks performed during the process cycle is adapted over time to optimise a particular metric. This adaptation could include the reordering of tasks or adaptation of individual tasks. This optimisation is achieved by strategic overview of the performance of the system while carrying out the set of tasks. Adaptation is the result of accumulated experience.
    - Level 4 - Communicated task adaptation: the process of adaptation is carried out between multiple independent agents. The adaptation is communicated between agents and applied individually within in each agent. Agents can be both real and simulated and of different types including non-robotic agents.

36. Select the robot's level of system dependability (performing tasks without errors)? Adopted from MAR.

    - Level 0 - No dependability: All useful robots are dependable to some degree, even laboratory prototypes. This level exists for completeness.
    - Level 1 - Mean failure dependability: the dependability of the robot is based on the mean time to failure of its components. The dependability is based on the design of the robot. The robot is not itself able to increase its dependability. For Failure Dependability this relates to the failure of all component parts of the robot including software components. For Functional dependability this relates to the frequency of failure of the system functions with respect to the task being undertaken, and for environmental dependability it relates to the failure of the robot to correctly interpret the environment, for example falling down a step, or failing to detect a hazard. For Interaction dependability it relates to the failure of the robot to interact with a human or another robot in a functional or intuitive manner that is appropriate to the task.
    - Level 2 - Fails safe: the robot design is such that there are fail-safe mechanisms built into the system that will halt the operation of the robot and place it into a safe mode when failures are detected. This includes any failures caused by in-field updates. Dependability is reduced to the ability to fail safely in a proportion of failure modes. Fail-safe dependability relies on being able to detect a failure.
    - Level 3 - Failure recovery: the robot is able to recover from a proportion of failures by restarting or resuming its operation

- Level 4 - Graceful degradation: the robot is able to recognise the impact of a proportion of failures on its function and operation and is able to compensate for the effect of the failure to maintain dependable operation. Function effectiveness or the ability to achieve optimal working may be impacted.
- Level 5 - Task dependability: the robot system is able to recognise the impact of a failure on the overall task it is undertaking and re-task activities in order to minimise the impact of the failure on the task. This may also include self-repair as an alternative task.
- Level 6 - Mission dependability: the robot is able to recognise the impact of a failure on the overall objectives of a mission and communicate the nature of the failure to other systems and robots to minimise the impact on the mission objectives. In turn the robot is able to receive and interpret mission failures from other robots and systems and re-task its actions to compensate.
- Level 7 - Predictive dependability: the robot system is able to predict that a planned future action may result in a loss of dependability, or that the effect of the partial failure of a component can be mitigated by altering future actions. Thus the robot is able to extend its dependability by taking action in advance of failure in order to reduce the effect on dependability

37. Please select the (social) human-robot interaction feedback level presented by the robot? Adopted from MAR.

    - Level 0 - No feedback: the robot system does not provide any feedback to the user.
    - Level 1 - Visual feedback: the user is able to assess the state of the robot by direct observation. The robot system does not provide any means of feeding back information to the user.
    - Level 2 - Vision data feedback: the system feedbacks visual information about the state of the operating environment around the robot based on data captured locally at the robot. The user must interpret this visual imagery to assess the state of the robot or its environment.
    - Level 3 - Simple haptic feedback: the robot system is able to feedback a physical force that represents the forces at the end effector of the robot. The force feedback is delivered to the user via a single point of contact, for example, a joystick.
    - Level 4 - Augmented haptic feedback: the system is able to feedback to the operator signals and forces that augment the force information from the end effector such that the augmentation enhances the interaction between the user and the robot.
    - Level 5 - Multiple point feedback: the robot system is able to feedback to the operator signals and physical forces that represent multiple forces at the end effector of the robot. The force feedback is delivered to the user via multiple points of contact, for example to each finger of the operator's hand.
    - Level 6 - Augmented multiple point feedback: the robot system is able to augment with additional information the feedback of a set of physical forces that represent the forces at the end effector of the robot. The force feedback is delivered to the user via multiple points of contact, for example to each finger of the operator's hand. This augmentation enhances the interaction between the user and the robot with additional information which may be derived from additional sensing or additional interpretation.

- Level 7 – Tele-presence: the system is able to provide multi-modal feedback to the operator such that they experience tele-presence. Typically this requires close synchronisation between different feedback channels.
- Level 8 - Augmented tele-presence: the system is able to augment the experience of tele-presence with additional information that enhances the interaction between the user and the robot.

38. Please select the level of (social) human interaction modality? Adopted from MAR.
    - Level 0 - No Human interaction: the robot does not interact with humans in a social context.
    - Level 1 - Commanded tasks: the robot carries out tasks on command given by a human user. The robot does not seek interaction with the user until it reaches pre-defined points in the task.
    - Level 2 - Direct interaction: the robot carries out tasks through direct interaction with a human. The interactions may modulate the task or the sequence of tasks. The robot uses pre-defined interaction patterns
    - Level 3 - Socially acceptable commanded interaction: the robot operates as an assistant to a person by carrying out useful tasks. These tasks are commanded but in a way that is socially acceptable to the person. The robot needs to take into consideration a variety of verbal and non-verbal modalities of interaction, and will utilise knowledge of the context and scope of its application domain to interpret and execute the command.
    - Level 4 - Interaction as an assistant: the robot operates as an assistant to a person observing what tasks need to be carried out without necessarily being commanded. The robot is able to observe the context of the person and the social environment and within a limited range of tasks it is able to decide how and when to execute tasks.
    - Level 5 - Interaction as a personal companion: the robot operates as a personal companion to a person by carrying out useful tasks for them. The robot adapts to and learns from its changing roles over the course of long-term interactions with the person. The robot is able to change its tasks depending on changes in the person's needs, abilities, interests and preferences.
    - Level 6 - Socially situated companion robots: the robot is able to operate as a personal companion to a group of people who interact socially. The robot understands the dynamics between the group and is able to interact with each member of the group in a socially appropriate way.
39. Please select the (social) human interaction levels of extent presented in the robot? Adopted from MAR.
    - Level 0 - No social interaction with humans: the robot system does not utilise any form of social interaction with humans, other forms of interaction may take place.
    - Level 1 - Temporally restricted interactions: the robot interacts with people only a small number of times while carrying out a task, and interactions are brief. The robot's social behaviour follows a pre-defined script. The robot does not use information exchanged in the interaction to change its social behaviour towards that person. The robot's social behaviour follows rules and conventions expected for specific pre-defined interactions.
    - Level 2 - Temporally extended interactions: the robot interacts with a person over an extended time period, interactions are brief but may be repeated. The robot is able to utilise information given by the person during the interaction to change its interaction with the person in order to personalize its behaviour within a range of a priori defined options.

- Level 3 - Behaviour modulated interactions: the robot interacts with a person over an extended time period, interactions are brief but repeated. The robot is able to recognise a set of pre-defined human behaviours during the interaction. The robot uses this information, together with information supplied by the person to change its interaction with that person in order to personalize its behaviour within a range of a priori defined options.
- Level 4 - Long-term interactions: the robot interacts with the person repeatedly over extended periods of time. The robot can identify the person from the person's behaviour and/or appearance over repeated interactions. The robot forms a model of the person that it can use to reason about the person. The robot uses this knowledge to personalize its behaviour and to adapt its behaviour over time from the history of interaction with the person.
- Level 5 - Task modulation through interaction: the robot interacts with the person repeatedly over extended periods of time. The robot can individually identify the person from the person's behaviour and/or appearance. In addition, the robot is able to infer short-term goals, basic intentions and desires from its observations during the interaction. The robot uses this knowledge to alter its interaction and the tasks it carries out.
- Level 6 - Accumulated personal knowledge through interaction: the robot interacts with the person repeatedly over extended periods of time. The robot can individually identify the person from the person's behaviour and/or appearance. In addition, the robot is able to accumulate knowledge about long-term characteristics of the person from its observations during the interaction, such as preferences, habits, goals, and beliefs. The robot uses this knowledge to alter its interaction and the tasks it carries out.
- Level 7 - Multi-party long-term interactions: the robot is able to extend its interactions to multi-party situations. This requires awareness of the dynamics and interactions among groups of people, and a model of relationships among the humans and towards the robot.

40. Please select the (physical/cognitive) object interaction level of the robot? Adopted from MAR.

    - Level 0 - No cognition based interaction with objects: many robot systems will be able to operate successfully without cognitive interaction with the objects.
    - Level 1 - Environmental context utilisation: the system is able to use context information about the environment to guide interaction with a specific object. This relates to the transfer of knowledge from the environment to the manipulation of a specific object. For example knowledge about a surface onto which an object is to be placed altering the placement strategy, or knowledge about the relationship between objects
    - Level 2 - Property identification: the robot is able to pick up an object that belongs to one of a number of known object types and determine properties of the object from its holding and manipulation of it. It is able to use these determined properties to control how the object is manipulated and placed. For example, a robot may pick up a cup and determine that it is full of liquid
    - Level 3 - Object placement: the system is able to manipulate and place an object in a way that is compatible with its state and context. For example property knowledge is used when orienting and object.

- Level 4 - Composite object manipulation: The system is able to identify that an object is composed of multiple different objects that are connected but which may be separable. Within the context of the task the system may be able to separate the parts, or exploit the union between them.
- Level 5 - Generalised object manipulation: the system is able to interact with an unknown object and as a result of the interaction categorise the object in terms of its categorical relationship to other known or discovered objects. This includes generic categorisations such as "it is a container for liquid".
- Level 6 - Novel object manipulation: Based on contextual and historical knowledge the system is able to establish that an identified object is novel as district from being unknown. Novelty may result from the object being broken or incomplete. For example a known mug is missing a handle, or a bottle its cap. The system is then able to manipulate the object taking into account its altered state.
- Level 7 - Use of affordances: The system is able to deduce that an object affords an action. The robot is able to grasp an object that has desired affordances within the context of the task or mission and manipulate the object in order to gain use of the afforded action.

41. Please select the (cognitive) robot-to-robot interaction level? Adopted from MAR.

    - Level 0 - No interaction: robot operates on its own without communication with another robot.
    - Level 1 - Communication of own status: two or more robots communicate basic status information and task specific status. Status information is pre-defined for the task. The information communicated only relates to the state of the robot within the task.
    - Level 2 - Communication of task status: two or more robots are able to communicate information about the task they are performing in terms of task completion, time to completion, and information about task barriers, resources etc. This information is at a high level and will impact on the planning of a common task, or tasks in a common space.
    - Level 3 - Communication of environment information: two or more robots share information about their local environments, or share wider scale information that they have acquired or been given. The robots are able to assimilate the information and extract task relevant knowledge from it.
    - Level 4 - Team communication: two or more robots are able to communicate task level information during execution of the task such that it is possible to implement dynamic planning between the robots in the team. Each robot carries out its own tasks with awareness of the other robots in the team.
    - Level 5 - Team coordination: two or more robots are able to collaborate to achieve a task outcome that could not be achieved by either robot alone, or by each robot operating independently.
    - Level 6 - Capability Communication robots: robots are able to communicate their own task capabilities and utilise cooperative working between teams of heterogeneous robots where there is no prior knowledge of the composition of the team.

42. Please select the (cognitive) robot-to-system or device interaction level in the robot? Developed for ToRCH.

- Level 0 - No robot-system interaction: robot is unable to communicate with other external systems.
- Level 1 - External system knowledge: the robot is aware of external systems but does not communicate with them.
- Level 2 - Communication of status: the robot is able to communicate its own status and task status to external systems.
- Level 3 - Reception of information: the robot is able to request information from external systems
- Level 4 - Action on information: the robot is able to dynamically respond to information provided by external systems.
- Level 5 - Selection capability: the robot is able to differentiate between different types of external systems and choose which to interact with to accomplish a given task or mission.

43. Please select the physical motion interaction level of the robot? Developed for ToRCH.
    - Level 0 - No physical interaction. Many robot systems will be able to operate successfully without physical interaction for example simulated robots, Alexa.
    - Level 1 - Self-utilization - interaction emerges from special settings of the electronic circuits of the robot.
    - Level 2 - Static self-peripheral utilization with a single appendage – the system is able to use peripherals to perform physical interaction within its environment. This relates to its manipulation capabilities, but robot remains static.
    - Level 3 - Static self-peripheral utilization with multiple appendages – the system is able to use peripherals to perform physical interaction within its environment. This relates to its manipulation capabilities, but robot remains static.
    - Level 4 - Level 4 - Dynamic self-peripheral utilization – the system is able to use peripherals to perform physical interaction within its environment. This relates to its manipulation capabilities, but robot has dynamic movement.

44. Please select the level of (social/ cognitive) interaction with complexity of the robot? Adopted from MAR.
    - Level 0 - No interaction: the robot operates on its own without social interaction.
    - Level 1 - Single task interaction: the robot carries out a clearly defined task, or a small number of tasks. These require the robot to socially interact with people in an acceptable way but the range of its interaction abilities are limited to the task and do not depend on knowledge of the human interaction partner(s) outside of the context of the task.
    - Level 2 - Multiple defined task interaction: the robot carries out a range of tasks, and needs to be able to engage with humans in a number of these tasks in a socially appropriate manner. These interactions need to take into consideration knowledge about the individual human it is interacting with and the different context of each task. The robot acts according to its Human Interaction level.

- Level 3 - Undefined parametrised interactions: the robot interacts with a human partner according to its Human Interaction Level. During and through interaction the human partner can modify the robot's behaviour, skills and knowledge about the domain in order to allow it to deal with new or unexpected situations.
- Level 4 - Unconstrained, open-ended: the robot cannot a priori predict precisely which interactions the human will engage in at any given point in time. The robot needs to be able to use extensive knowledge, knowledge about interaction histories with the user and extensive social signal processing abilities in order to determine the context of the interaction (e.g. to determine and appropriately respond to dialogues on various topics), and to respond with appropriate and socially acceptable behaviour. The robot needs to be able to monitor, adapt to and learn from changing user's preferences and needs.

45. Please select the level of (social/ cognitive) social interaction with cognitive learning ability of the robot? Adopted from MAR.

    - Level 0 - No interaction learning: the robot does not learn from its interactions with people. It may learn in other ways.
    - Level 1 - Learning social sequences: the robot directly learns from specific sequences of social interaction and socially intelligent behaviour that take place during interactions with a user.
    - Level 2 - Social learning by observation: the robot learns over an extended period by observing the social interactions it has with its own interaction partners.
    - Level 3 - Third party learning: the robot learns by observing third-party interactions between people or between people and other robots. The robot is able to extract social interaction knowledge from these third-party interactions and utilise this knowledge within the context of its own function.

46. Please select the level of (social/ cognitive) human interaction with a robot that have a cognitive ability? Adopted from MAR.

    - Level 0 - No Cognitive Human Interaction: Many robot systems will be able to operate successfully without cognitive interaction with the user.
    - Level 1 - Fixed interaction: Interaction between the user and the robot follows a fixed pattern. Typically this takes place via a user interface with well defined inputs and outputs. Typical of this type of interaction are domestic vacuum cleaning robots which offer simple button interfaces and display a minimum amount of status information. Fixed interaction also includes interaction via a computer based user interface where interactions directly control the robot according to pre-defined sets of commands with specific meaning. The connection between the user and the robot may involve a wireless link. Any interpretation of commands is fixed and embedded.
    - Level 2 - Task context interaction: The system is able to interpret commands from the user that utilise task context semantics within a domain specific communication framework appropriate to the range of the task. The system is able to relay task status to the user using task context semantics suitable for the task.
    - Level 3 - Object and location interaction: The system is able to interpret user interactions that refer to objects, locations or actions in as is appropriate to the task. This includes the ability to interpret

user interactions that identify objects locations and actions as well as processing commands that reference locations, objects and actions relevant to the task. Dialogues are initiated by the user.

- Level 4 - Robot triggered interaction: The system is able to start a dialogue with the user in a socially appropriate manner relevant to its task or mission. The robot has a basic understanding of the social interaction appropriate to the task/mission domain. Interaction may continue throughout the operating cycle for each task as is appropriate to the task/mission.
- Level 5 - Social interaction: The system is able to maintain dialogues that cover more than one type of social interaction, or domain task. The robot is able to manage the interaction provided it remains within the defined context of the task or mission.
- Level 6 - Complex social interaction: Dialogues cover multiple social interactions and tasks, where the robot is able to instruct the user to carry out tasks, or enter into a negotiation about how a task is specified. The interaction is typified by a bi-directional exchange of commands.
- Level 7 - Intuitive Interaction: The robot is able to intuit the needs of a user with or without explicit command or dialogue. The user may communicate to the robot without issuing explicit commands. The robot will intuit from the current context and historical information the implied command.

47. Please select the (general) human-robot methods of interaction independent of the cognitive context? Adopted from MAR.

    - Level 0 - No interaction: the robot has no operational interaction with a user.
    - Level 1 - Direct control: the user constantly provides input signals to control the robot. The user provides control of the robot moment to moment. The system can translate, alter, or block these controls within parameters set by the user or system. The user controls are in the form of parameters that alter the control of the robot. These parameters may be continuous quantities, for example, a steering direction, or binary controls.
    - Level 2 - Direct physical interaction: the user controls the robot by physically interacting with it. The robot reacts to the user interaction by feeding back physical information to the user via the contact point. For example, the user teaches a motion sequence to the robot, or feels the surface of an object the robot is in contact with.
    - Level 3 - Position selection: the user controls the robot by issuing pre-defined actions sequentially. The system is able to execute these pre-defined actions autonomously. The user selects the subsequent action at the completion of each action. For example, a robot is able to move between defined points in its environment, or carry out a fixed action such as releasing an object, as commanded by the user.
    - Level 4 - Traded autonomy: the system is able to operate autonomously during some parts of a task or in some tasks. Once this task or sub-task is complete the user will either select the subsequent task or intervene to control the system by direct interaction to carry out a task. This results in alternating sequences of autonomous and direct control of the system by the user.
    - Level 5 - Task sequence control: the user controls the robot by issuing pre-defined sub-tasks or tasks sequentially. The system is able to execute sub-tasks autonomously, these sub-tasks will involve a higher level of decisional autonomy than the pre-defined tasks in Level 3. On

completion of the sub-task user interaction is required to select the next sub-task resulting in a sequence of actions that make up a completed task.

- Level 6 - Supervised autonomy: the robot system is able to execute a task autonomously under most operating conditions. The system is able to recognise when it is unable to proceed or when it requires user input to select alternative strategies or courses of action. These alternatives may involve periods of direct.
- Level 7 - Task selection: the system is able to autonomously execute tasks but requires the user to select between strategic task alternatives in order to execute a mission.
- Level 8 - Mission goal setting: the system is able to execute tasks to achieve a mission. The user is able to interact with the system to direct the overall objectives of the mission.

48. Please select the (general) human-robot proximity level? Developed for ToRCH.
    - Level 0 - None: no physical contact with human
    - Level 1 - Minimal: independent, unlikely to contact a human
    - Level 2 - Parallel: works independently alongside a human
    - Level 3 - Operative: operated or commanded by a human
    - Level 4 - Cooperative: cooperates physically with a human
    - Level 5 - Social: interacts socially with a human
    - Level 6 - integrated: Worn or implanted on a human
49. Please select the talent and skills the robot is programmed and designed to perform? Developed for ToRCH.
    (a) Visual-spatial skills
    (b) Bodily-kinaesthetic skills
    (c) Learning skills
    (d) Logical-mathematical skills
    (e) Emotional skills
    (f) Musical skills
    (g) Heterogeneous robotic skills
    (h) Homogeneous robotic skills
50. Please select the dimension of intelligence presented in the robot? Developed for ToRCH.
    (a) Physical-morphological intelligence
    (b) Cognitive intelligence
    (c) Social intelligence
    (d) Collective intelligence

Appendix B

Questionnaire to validating the concepts of the robot framework

In this research there will be an explanation of the proposed framework followed by a questionnaire. The survey outlines the framework levels and sub-sections. The main purpose is to validate the framework, with the potential to add to or adjust the framework based on the information gathered.

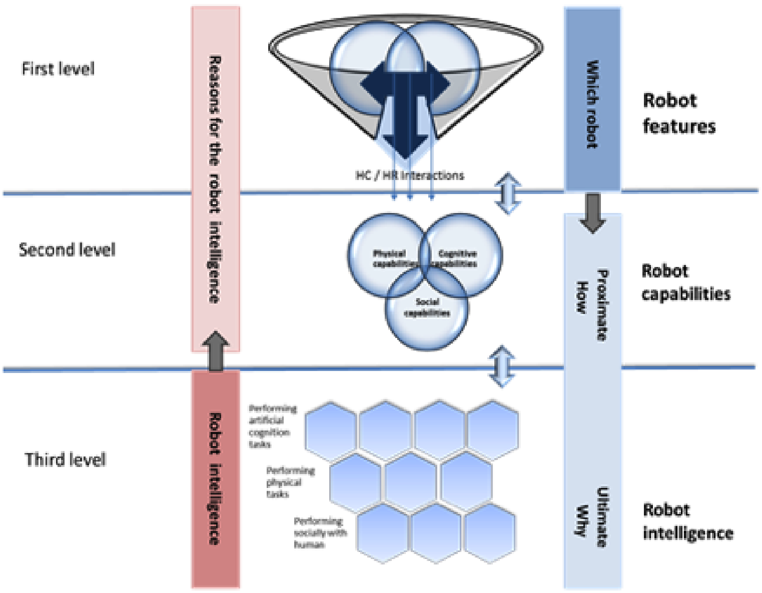

Figure B.1: The figure presents the main layers of the framework and their relationship to each other.

Answer the following question according the Figure B.1:

1. Do you agree with the relations between the three levels to describe any robot or computational device?
2. Are there any fundamental robotic descriptive concepts missing in the framework?
3. Do you think there is a missing concept in the first level to describe the robot features correctly?
4. Do you agree that the second level concepts describe the robot capabilities correctly within the framework?
5. Do you agree that the third level concepts describe the robot intelligence dimensions correctly within the framework?

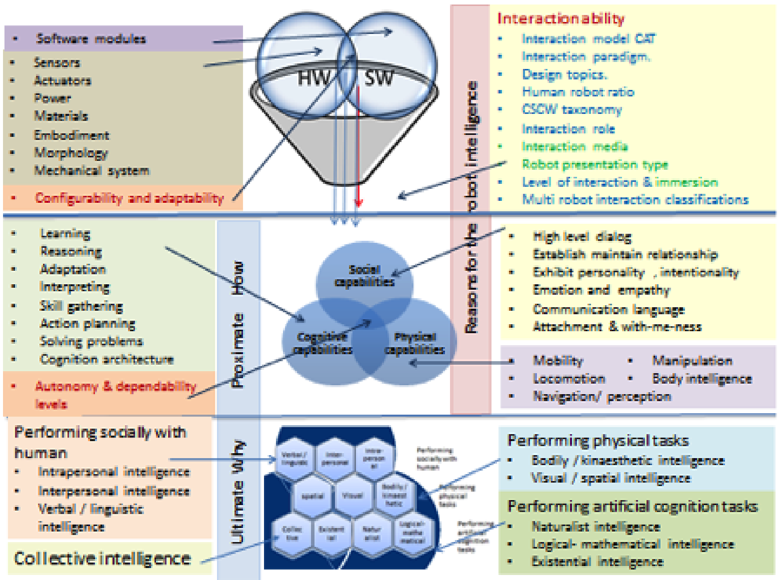

Figure B.2: The figure presents the ToRCH capabilities and their relationship to each other. It illustrates the capabilities in a nested perspective.

Answer the following question according the Figure B.2:

6. . Would you agree or disagree with the separate elements in each level of the framework. Where possible please give the reasons.
   (a) HW in the first level to present "robot feature"
      i. Agree
      ii. Disagree (if you don't agree please state why)
   (b) SW in the first level to present "robot feature"
      i. Agree
      ii. Disagree (if you don't agree please state why)
   (c) Interaction in the first level to present "robot feature"
      i. Agree

   ii. Disagree (if you don’t agree please state why)

(d) Physical capabilities in the second level to present "robot capabilities"

   i. Agree
   ii. Disagree (if you don’t agree please state why)

(e) Cognitive capabilities in the second level to present "robot capabilities"

   i. Agree
   ii. Disagree (if you don’t agree please state why)

(f) Social capabilities in the second level to present "robot capabilities"

   i. Agree
   ii. Disagree (if you don’t agree please state why)

(g) The bodily/kinaesthetic, visual/spatial, naturalist, musical, logical/mathematical, existential, verbal/linguistic, interpersonal, intrapersonal, and collective intelligence in the third level to capture the robot intelligence.

   i. Agree
   ii. Disagree (if you don’t agree please state why)

7. Where would you place “social robot” within this framework?

   (a) First level
   (b) Second level
   (c) Third level

8. Where would you place “lawnmower robot” within this framework?

   (a) First level
   (b) Second level
   (c) Third level

9. Where would you place “exoskeleton robots in health care” within this framework?

   (a) First level
   (b) Second level
   (c) Third level

10. Where would you place "internal medical nanobots or nanomites used in health care" within this framework?

   (a) First level
   (b) Second level
   (c) Third level

11. Where would you place any "simulated robots" within this framework?

  (a) First level
  (b) Second level
  (c) Third level

12. Are you aware of any taxonomy which relates to this framework, if so what is it called, and how is it used?
13. How would you use this framework?
14. Where could you use this framework?
15. Why would you use this framework?
16. In your opinion who could use this framework?
17. Where would you rate the acceptance level of the framework? rank from 0 to 10.
18. Name
19. Email
20. Area of your field: manufacturing domain, healthcare, agriculture domain, civil domain, commercial domain, logistics and transport consumer robots. If other, please specify.

Appendix C

Questionnaire to validating the accuracy of ToRCH

NAO Scenario 1:

The NAO robot is programmed to recognize a predefined ball on the floor among other unknown objects. The robot moves towards the ball and says to the person in the room, “I found the missing ball”. If the person smiles, the robot maps the facial expression to a predefined model of ’happy’ and says, "I found your ball." If the person doesn’t smile the robot says, "I will dance for you".

NAO Scenario 2:

The NAO robot is programmed to recognize a predefined ball on the floor among other unknown objects. The robot recognises an individual among the people in the room because their pictures are in its database. It moves toward that individual, laughs and says, "I found you and I found your ball".

Select one of the levels for each of the listed question, for both Scenario 1 and Scenario 2

1. Please select the level of object perception in your robot?

    - Level 0 - No object recognition.
    - Level 1 - Feature detection: sense data is gathered from a region of the environment such that the data has a spatial component and can be mapped to a model of that region. The richness of the sense data is such that it is possible to apply a feature detection process to create a set or sets of features that persist.

- Level 2 - Object detection: multiple persistent features can be grouped to build models of distinct objects allowing objects to be differentiated from each other and from the environment.
- Level 3 - Object recognition - single instance: object models created from sense data can be matched to specific known instances of an object with a reliability that is appropriate to the task.
- Level 4 - Object recognition - one of many: object models created from sense data can be matched to one of a number of specific instances of known objects with a reliability that is appropriate to the task.
- Level 5 - Parameterised object recognition: object models created from sense data can be matched to a number of known, parametrised object types. The settings for the parameters (e.g. size ratio, curvature, joint position etc.) can be deduced from the sensed object model. Note that in conjunction with single instance recognition ability this implies the ability to recognise a known (possibly learned) instance of a generic object, for example a particular brand of canned drink based on the generic recognition of a drinks can shape.
- Level 6 - Context based recognition: the system is able to use its knowledge of context or location to improve its ability to recognise objects by reducing ambiguities through expectations based on location or context.
- Level 7 - Object variable recognition: the system is able to recognise objects where there is a degree of variability in the object that approaches the scale of the object. For example, many generic objects such as coffee mugs vary in shape size and colour.
- Level 8 - Novelty recognition: the system is able to recognise novelty in a known object, or parametrised object type. For example, a known mug where the handle is missing or broken.
- Level 9 - Unknown object categorisation (rigid): the system is able to assess an unknown rigid object based on sense data and deduce properties that are relevant to the task.
- Level 10 - Object property detection: it is possible to use sense data and the derived object model to deduce the properties of an object. For example analysis of the sense data may provide surface texture information, knowledge about de-formability, or the content of an object.
- Level 11 - Flexible object detection: the system is able to detect the shape and form of objects that are deformable and generate parametrised models of flexible objects. This includes articulated objects and objects with flexible and rigid components.
- Level 12 - Flexible object classification: the system is able to classify flexible objects by their properties and parameters. It is able to recognise specific known objects relevant to the task with an appropriate level of reliability.

Please note any comments regarding object perception (e.g., levels, clarity, usage of the levels)?

2. Recognizing a person - Please select the level of social agent perception? Developed Copy MAR

- Level 0 - No human social agent recognition.
- Level 1 - Feature detection to sense an aspect of a human social agent: sense data is gathered from the environment, and can be mapped onto to a human profile model. The richness of the sense data is such that it is possible to apply a feature detection process to create a set or sets of features of a person.

- Level 2 - Human social agent detection of a whole social agent: multiple persistent human features can be grouped to build distinct human profile allowing humans to be differentiated from each other and from the environment.
- Level 3 - Person recognition - single instance: a human social agent model created from sense data can be matched against a specific human social agent model in the database with a reliability that is appropriate to the task.
- Level 4 - Person recognition - one of many: matching a person to one of several defined possible human social agent identities. The human models created from sense data can be matched against several known identities with a reliability that is appropriate to the task.
- Level 5 - Parameterised person recognition: recognising a generic human social agent by matching the person feature with known parameters. The models created from sense data can be matched to a number of known human social agent parameters. The settings for the parameters (e.g. size ratio, gender specification etc.) can be deduced from the sensed human social agent model.
- Level 6 - Context based recognition: improved human social agent recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognise human social agents by reducing ambiguities through expectations based on context or location.
- Level 7 - Person variable recognition: the system is able to recognise a human social agent where there are variations in his/her profile such as haircut and skin colour, for example, recognising a person with a change in some variables according to a specific scale.
- Level 8 - Novelty recognition, the ability to recognise novelty in a human social agent. The system is able to recognise a specific feature in a person that may have changed. For example, identify a specific scar on a person.

Please note any comments regarding social agent perception (e.g., levels, clarity, usage of the levels)?

3. Recognising emotion - Please select the level of emotion perception?

   - Level 0 - No emotion perception.
   - Level 1 - Emotion detection: the robot is able to sense an aspect of emotion. Sense data is gathered from the social agent and can be mapped. The richness of the sense data is applied to detect some features of social agent emotions.
   - Level 2- Emotion recognition - single instance: an emotional model created from sense data can be matched against a specific model in the database.
   - Level 3 - Emotion recognition - one of many: matching an emotional model to one of several defined possible models. The emotional models created from sense data can be matched against several known identified models with a reliability that is appropriate to the task.
   - Level 4 - Parameterised emotional recognition: recognising a generic emotion by matching the emotional feature with known parameters. The emotional models created from sense data can be matched to a number of known emotional parameters. The settings for the parameters can be deduced from the sensed model.
   - Level 5 - Context based emotion recognition: improved emotion recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognise emotion by reducing ambiguities through expectations based on context or location.

Please note any comments regarding emotion perception (e.g., levels, clarity, usage of the levels)?

4. Robot movement - Please select the level of self-location perception?

    - Level 0 - No emotion perception.
    - Level 1 - Emotion detection: the robot is able to sense an aspect of emotion. Sense data is gathered from the social agent and can be mapped. The richness of the sense data is applied to detect some features of social agent emotions.
    - Level 2- Emotion recognition - single instance: an emotional model created from sense data can be matched against a specific model in the database.
    - Level 3 - Emotion recognition - one of many: matching an emotional model to one of several defined possible models. The emotional models created from sense data can be matched against several known identified models with a reliability that is appropriate to the task.
    - Level 4 - Parameterised emotional recognition: recognising a generic emotion by matching the emotional feature with known parameters. The emotional models created from sense data can be matched to a number of known emotional parameters. The settings for the parameters can be deduced from the sensed model.
    - Level 5 - Context based emotion recognition: improved emotion recognition using contextual knowledge. Using knowledge of context or location to improve its ability to recognise emotion by reducing ambiguities through expectations based on context or location.

    Please note any comments regarding self-location perception (e.g., levels, clarity, usage of the levels)?

5. Robot movement - Please select the level of object-location perception?

    - Level 0 - No perception of object-location neither in terms of its position relative to the environment nor with its own location.
    - Level 1 - Object position: the robot identifies the object location as a result of object perception.
    - Level 2 - External beacons: the robot identifies the object location as a result of information derived from the inspection of external beacons. Beacons may be active or passive and include global beacons.
    - Level 3 - Relative location: the system is able to calculate the object location relative to a specific defined location with a degree of accuracy that is sufficient for the task.
    - Level 4 - Feature-based location: the system calculates the object position within an environment based on the motion of fixed features in the environment. For example by using SLAM to build and maintain a local map.
    - Level 5 - Mapped location: the robot is able to relate the object location position to a map that it has been given or that it has acquired. This may be a location within a task-relevant space.
    - Level 6 – Spatial occupancy: the system calculates the position of the object structures based on indirectly gathered sense data to provide a spatial notion of occupancy.
    - Level 7 - Object coupled location: the system is able to calculate the position of an object in conjunction with other objects it is connected to. For example, an object that is being gripped by the robot.

Please note any comments regarding object-location perception (e.g., levels, clarity, usage of the levels)?

6. Robot movement - Please select the level of social agent location perception?

    - Level 0 - No perception of social agent-location nether in terms of its position relative to the environment nor with its own location.
    - Level 1 - Social agent position: the robot identifies the social agent location as a result of perception.
    - Level 2 - External beacons: the robot identifies the social agent location as a result of information derived from the inspection of external beacons. Beacons may be active or passive and include global beacons.
    - Level 3 - Relative location: the system is able to calculate the social agent location relative to a specific defined location, with a degree of accuracy that is sufficient for the task.
    - Level 4 - Feature-based location: the system calculates the social agent position within an environment based on the motion of fixed features in the environment. For example by using SLAM to build and maintain a local map.
    - Level 5 - Mapped location: the robot is able to relate the social agent location position to a map that it has been given or that it has acquired. This may be a location within a task-relevant space.
    - Level 6 – Spatial Occupancy: the system calculates the position of the social agent structures based on indirectly gathered sense data to provide a spatial notion of occupancy.
    - Level 7 - Object coupled location: the system is able to calculate the position of social agent in conjunction with other objects it is connected to.

    Comments regarding the social agent-location perception (levels, clarity, usage of the levels)?

7. Selecting the ball - Please select the level of the physical object interpretive capability?

    - Level 0 - No interpretive ability: it does not need to interpret objects in the environment.
    - Level 1 - Fixed sensory interpretation: the robot has a fixed interpretation of the perceptions that occur because they are pre- categorised. For example, all sensed objects are applied to an occupancy grid and assumed to represent actual objects in the environment.
    - Level 2 - Basic environment interpretation: the robot uses sense data to interpret the environment into fixed notions of environmental space that are pre-categorised. For example, it will search for floor and wall segments in the sense data as these are relevant to its task even if the environment it is sensing has neither.
    - Level 3 - Object delineation: the robot is able to disambiguate objects from an interpretation of its static environment. The disambiguation of objects is based on built-in notions of object and environment. These notions may only be valid within a narrow operating context.
    - Level 4 - Object category interpretation: the robot is able to interpret the shapes and forms of objects based on categories of objects that are task relevant. It is able to interpret sense data to identify coherent instances of an object over a time scale appropriate to the task. Note that this ability level is particularly affected by the cognition ability parameters.

- Level 5 - Structural interpretation: the robot is able to interpret perceptions so as to extract structural information from the environment. It is able to identify the structural relationships between objects in the environment.
- Level 6 - Basic semantic interpretation: the robot is able to apply semantic tags to locations and objects allowing it to plan actions based on functional objectives that depend on the semantics of objects and locations.
- Level 7 - Property interpretation: the robot is able to interpret perceptions to determine the properties of objects or locations in the environment.
- Level 8 - Novelty interpretation: the robot is able to interpret perceptions to identify novelty in objects or locations.
- Level 9 - Environmental affordances: the robot is able to interpret the environment in terms of what it affords. For example, it is able to interpret the ground conditions in a muddy field as being too unstable for the load it is carrying.

Please note any comments regarding physical object interpretive capability (e.g., levels, clarity, usage of the levels)?

8. Robot movement - Please select the level of unconstrained physical motion capabilities?

- Level 0 - No motion: All robots move in their environments, except fixed robots.
- Level 1 - Pre-defined open loop motion: the robot carries out predefined moves in sequence. The motion is independent of the environment and events in the environment. The robot may not be able to maintain a position if subject to external forces, may be able to statically rest at a given position.
- Level 2 - Predefined closed-loop motion: the robot carries out predefined moves in the sequence where each motion is controlled to ensure position and/or speed goals are satisfied within some error bound. So, for example, a robot can move to and maintain a position (within some error margin) against forces less than the resultant motive force at the point of contact. A platform will similarly be able to execute fixed motions where the accuracy of these motions in the environment will depend on other abilities such as its perception ability.
- Level 3 - Open path motion: the robot can execute a motion that follows a path with a given path accuracy. This path is described by a specific point on the robot. The robot is able to return to any given point on the path with an accuracy that is appropriate to the task.
- Level 4 - Position constrained path motion: the robot can execute a path motion where the path is constrained by physical objects or by defined zones that must be avoided. For example, a robot arm that can operate through a physically constrained region such as a hole in a wall, or a platform that can move to avoid a known area of the environment such as a step-down. The robot is able to execute a path to an unvisited location obeying constraints.
- Level 5 - Force constrained path motion: the robot can execute a path motion while applying a specified force in a given direction related to the motion. For example, moving over the surface of an object while applying a force perpendicular to the surface as might be required when polishing a surface.

- Level 6 - Parametrised motion: the robot can execute a path move that optimises for a parameter. For example, a path that reduces energy consumption, covers an area or constrains the angle range of a joint, or the torque or force in a joint or linkage.
- Level 7 - Position constrained parametrised motion: the robot can operate through a physically constrained region while at the same time optimising a parameter or set of parameters that constrain the motions of the robot. For example, a robot arm may be able to reach a high shelf while maintaining a centre of gravity, or a platform robot operates in a room away from a charging station while optimising power usage.

Please note any comments regarding unconstrained physical motion capabilities (e.g., levels, clarity, usage of the levels)?

9. Robot expression of emotion - Please select the level of social emotion expression capability?

- Level 0 - No emotion expression ability: the robot does not need to express emotions in performing their tasks.
- Level 1 - Expressing a pre-defined emotion: the robot is able to execute a pre-defined emotion. The emotion is programmed to be executed to present the robot status.
- Level 2 - Expressing a set of pre-defined emotions: the robot is able to execute pre-defined emotions. Emotions are programmed to be executed to present the robot status.
- Level 3 - Decision based expression of emotion: the robot is able to alter its own emotions according to the social perception of another agent. Presenting different emotions as a social feedback action.
- Level 4 - Parameterised expression of emotion: expressing a generic emotion by matching the emotion expression with known parameters. The emotion models created from sense data can be matched to a number of known emotion parameters. The settings for the parameters can be deduced from the sensed model.
- Level 5 - Sense driven expression of emotion: the robot is able to modulate its own emotions in proportion to perception-derived parameters. Emotion levels are expressed in different ways.

Please note any comments regarding social emotion expression capability (e.g., levels, clarity, usage of the levels)?

10. Human-robot interaction - Please select the level of human interaction modality?

- Level 0 - No Human interaction: the robot does not interact with humans in a social context.
- Level 1 - Commanded tasks: the robot carries out tasks on command given by a human user. The robot does not seek interaction with the user until it reaches pre-defined points in the task.
- Level 2 - Direct interaction: the robot carries out tasks through direct interaction with a human. The interactions may modulate the task or the sequence of tasks. The robot uses pre-defined interaction patterns
- Level 3 - Socially acceptable commanded interaction: the robot operates as an assistant to a person by carrying out useful tasks. These tasks are commanded but in a way that is socially acceptable to the person. The robot needs to take into consideration a variety of verbal and

non-verbal modalities of interaction, and will utilise knowledge of the context and scope of its application domain to interpret and execute the command.

- Level 4 - Interaction as an assistant: the robot operates as an assistant to a person observing what tasks need to be carried out without necessarily being commanded. The robot is able to observe the context of the person and the social environment and within a limited range of tasks it is able to decide how and when to execute tasks.
- Level 5 - Interaction as a personal companion: the robot operates as a personal companion to a person by carrying out useful tasks for them. The robot adapts to and learns from its changing roles over the course of long-term interactions with the person. The robot is able to change its tasks depending on changes in the person's needs, abilities, interests and preferences.
- Level 6 - Socially situated companion robots: the robot is able to operate as a personal companion to a group of people who interact socially. The robot understands the dynamics between the group and is able to interact with each member of the group in a socially appropriate way.

Please note any comments regarding human interaction modality (e.g., levels, clarity, usage of the levels)?

11. Time/number of interaction[s] - Please select the social human interaction level of extent?

- Level 0 - No social interaction with humans: the robot system does not utilise any form of social interaction with humans, other forms of interaction may take place.
- Level 1 - Temporally restricted interactions: the robot interacts with people only a small number of times while carrying out a task, and interactions are brief. The robot's social behaviour follows a pre-defined script. The robot does not use information exchanged in the interaction to change its social behaviour towards that person. The robot's social behaviour follows rules and conventions expected for specific pre-defined interactions.
- Level 2 - Temporally extended interactions: the robot interacts with a person over an extended time period, interactions are brief but may be repeated. The robot is able to utilise information given by the person during the interaction to change its interaction with the person in order to personalize its behaviour within a range of a priori defined options.
- Level 3 - Behaviour modulated interactions: the robot interacts with a person over an extended time period, interactions are brief but repeated. The robot is able to recognise a set of pre-defined human behaviours during the interaction. The robot uses this information, together with information supplied by the person to change its interaction with that person in order to personalize its behaviour within a range of a priori defined options.
- Level 4 - Long-term interactions: the robot interacts with the person repeatedly over extended periods of time. The robot can identify the person from the person's behaviour and/or appearance over repeated interactions. The robot forms a model of the person that it can use to reason about the person. The robot uses this knowledge to personalize its behaviour and to adapt its behaviour over time from the history of interaction with the person.
- Level 5 - Task modulation through interaction: the robot interacts with the person repeatedly over extended periods of time. The robot can individually identify the person from the person's behaviour and/or appearance. In addition, the robot is able to infer short-term goals, basic

intentions and desires from its observations during the interaction. The robot uses this knowledge to alter its interaction and the tasks it carries out.

- Level 6 - Accumulated personal knowledge through interaction: the robot interacts with the person repeatedly over extended periods of time. The robot can individually identify the person from the person's behaviour and/or appearance. In addition, the robot is able to accumulate knowledge about long-term characteristics of the person from its observations during the interaction, such as preferences, habits, goals, and beliefs. The robot uses this knowledge to alter its interaction and the tasks it carries out.
- Level 7 - Multi-party long-term interactions: the robot is able to extend its interactions to multi-party situations. This requires awareness of the dynamics and interactions among groups of people, and a model of relationships among the humans and towards the robot.

12. Please note any comments regarding the social human interaction of extent (e.g., levels, clarity, usage of the levels)?