



HAL
open science

Digitalization of Training Tasks and Specification of the Behaviour of a Social Humanoid Robot as Coach

Peter Forbrig, Alexandru Bunea, Ann Pedersen, Thomas Platz

► **To cite this version:**

Peter Forbrig, Alexandru Bunea, Ann Pedersen, Thomas Platz. Digitalization of Training Tasks and Specification of the Behaviour of a Social Humanoid Robot as Coach. 8th International Conference on Human-Centred Software Engineering (HCSE), Nov 2020, Eindhoven, Netherlands. pp.45-57, 10.1007/978-3-030-64266-2_3. hal-03250488

HAL Id: hal-03250488

<https://inria.hal.science/hal-03250488v1>

Submitted on 4 Jun 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Digitalization of Training Tasks and Specification of the Behaviour of a Social Humanoid Robot as Coach

Peter Forbrig¹[0000-0003-3427-0909] and Alexandru Bunea¹[0000-0001-8315-3405] and Ann Petersen³[0000-0002-5613-2104] and Thomas Platz^{2,3}[0000-0003-2629-9744]

¹ University of Rostock, Department of Computer Science, Albert-Einstein-Str. 22,
18055 Rostock, Germany

{Peter.Forbrig | Alexandru-Nicolae.Bunea}@uni-rostock.de

²Institut für Neurorehabilitation und Evidenzbasierung, An-Institut der Universität
Greifswald, BDH-Klinik Greifswald, Karl-Liebknecht-Ring 26a, 17491 Greifswald

³Neurorehabilitation Research Group, Universitätsmedizin Greifswald, Fleischmannstraße
44, 17491 Greifswald

Thomas.Platz@uni-greifswald.de

Abstract. The number of physiotherapists and occupational therapists is not sufficient to cope with the demands of the increasing number of stroke survivors worldwide. These patients need specific training to promote recovery and prevent stroke-related disability. The paper discusses aspects how a social humanoid robots might serve as therapeutic assistant. It is not intended to replace human therapists, but to provide them with therapeutic assistance once therapeutic decisions are taken and the therapy has been introduced to the patient by the human therapists and the day to day practice needs to be supervised. The paper provides a case study of an HCD approach and exemplifies the strategy of digitalizing training tasks, the identification of exceptions, and ways of modeling the dynamic behaviour of the humanoid robot. This is demonstrated by task models. Some extensions of task models are suggested.

Keywords: Social Humanoid Robot · Human Factors · Interaction Design · Robot Assistance · Arm Rehabilitation for Stroke Patients · Task Models.

1 Introduction

Within the project E-BRAiN (Evidence-based Robot-Assistance in Neurorehabilitation; www.ebrain-science.de) we want to develop software that allows a humanoid robot to observe carefully selected training exercises. The project has the goal to support therapists by a new companion in the training of patients after a stroke. It was decided to follow a Human-Centered Design (HCD) approach that was adapted from Harte et.al. [9]. To study the tasks and contexts is one of the first things one has to do.

A stroke is a leading cause of disability [8]. Brain lesions caused by stroke affect various body functions and lead to activity limitations and participations restrictions. Frequently, intensive specific training schedules are necessary to promote the recovery of body functions and reduce disability thereby. Platz and Lotze report in [16]

about the clinical effectiveness of specific exercises in cases when stroke survivors suffer from mild to moderate arm paresis causing focal disability, i.e. limitations of dexterity. A specific group of such exercises is called Arm-Ability Training (AAT)

First, we will provide an overview of some of these exercises for post-stroke patients. Afterwards, the social humanoid robot Pepper is shortly introduced. This is followed by a discussion of digitalizing three of the training tasks with paper and paper. Models for specifying the collaboration between patients and therapists are discussed in the fourth section. In the fifth section we focus on modelling the collaboration of patients and therapists resp. humanoid robots. Some lessons learned from methodological and technological aspect are discussed in the six section before a summary with outlook follows.

2 Arm-Ability Training (AAT)

Arm-Ability Training has been especially designed to promote manual dexterity recovery for stroke patients who have mild to moderate arm paresis. Platz and Lotze report in [16] about its design, clinical effectiveness, and the neurobiology of the actions. The idea of AAT goes back to the identification of sensor motoric deficits of stroke survivors in [17]. **Fig. 1** provides an overview of the suggested training activities that are a result of previous research in [14] and [15].

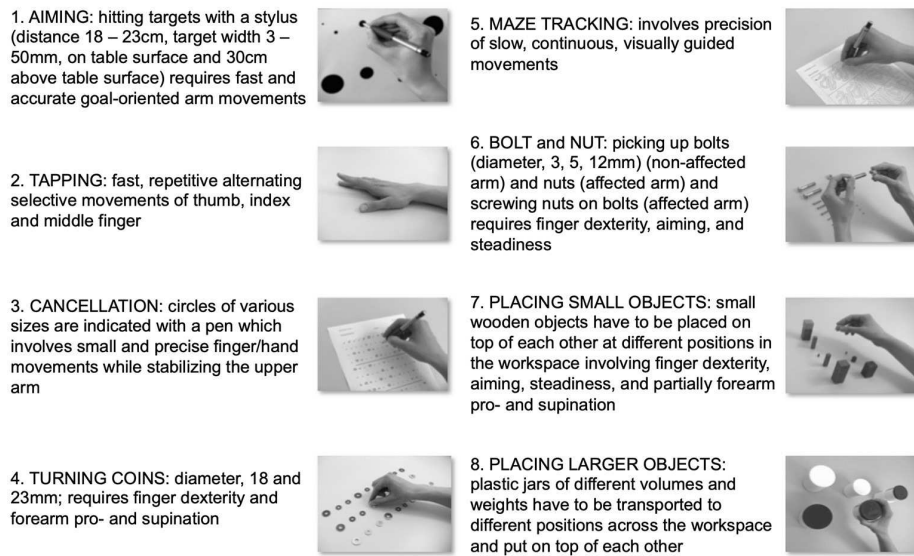


Fig. 1. Training tasks of AAT (from [16]).

The clinical effectiveness of the arm ability training was discussed in [16] on the basis of two single-blind randomized controlled studies that included 74 patients.

Platz et al. (2009) were able to show is superior clinical effect on focal disability compared to therapeutic time-equivalent “best conventional” therapy.

However, the training is resource intensive because one therapist is necessary to observe and support the exercises of a patient. The idea arose to use a robot for assisting the patients. The robot is intended to provide instructions, provides feedback, be a motivator and suggest alternatives if something does not work well. It was analysed which existing kind of robot fits to the requirements of the arm ability training. As a result, the humanoid robot Pepper was selected for first experiments. A more detailed discussion related to humanoid robots can be found in Forbrig et. al. [5] and [6].

3 The Humanoid Robot Pepper

SoftBank Robotics [20] offers an interesting social humanoid robot and call it Pepper. It is already in use for different domains. Pepper welcomes customers at banks or shopping centers. It gives advices in railway stations or airports. Sometimes it is also used to educate children or to entertain and support elderly people like in [10].

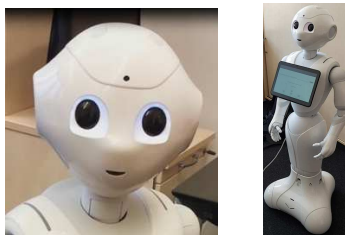


Fig. 2. Humanoid robot Pepper from SoftBanks Robotics [20].

Pepper looks a little bit like child and has a very nice appearance. Most people like Pepper and smile while they are looking at the robot. Pepper can talk, listen, move the head and focus on people that he interacts with. His eye blinking simulates the blink of an eye of humans in a nice way. The robot can also blink with shoulders and ears. Additionally, Pepper can move its arms and fingers and is gesturing while talking. Even that he does not have real legs the robot can move around.

It is our assumption that patients can establish a good relationship with the robot. We assume that this is very helpful for the training tasks that are intended to support the functional recovery of patients. We want to use Pepper mainly to supervise highly standardized training sessions as instructor and motivator and like in [10] or [13] and try to find the appropriate feedback. Therefore, within our E-BRAiN project we are cooperating with psychologists and Professionals from health care management. They will analyse whether a humanoid robot as therapy assistant can acceptable for patients. It is, however, well possible that the approach is only helpful for specific sub-groups of stroke survivors and specific types of functional training. It is one goal of our project to identify such groups and forms of training.

4 Robot-Supported Training Tasks

The goal of the project E-BRAiN is a combined therapy of human experts und humanoid robots. Background for any robot-supported training tasks are the assessment, goal-determination, selection of appropriate therapy, information and consent about any findings and plans with the patient and equally the introduction of the training tasks to the patients by a human therapist.

Accordingly, (human) therapists need to assess patients, explore the individual treatment goals, select, suggest and agree on the appropriate training, and introduce the training exercises to patients. They also need to supervise the first cycles of training performance. Once, this level of therapeutic care is achieved, there might be a role for a social robot to support additional largely standardized training sessions.

However, before this can be done, the therapist has to specify personal data and medical training specifications in detail. Additionally, personal goals of the patients have to be identified. Such a goal could be: “I want to play chess with my grandchildren again”. Later in the therapy, these goals are used to motivate patients. Pepper can say: “Today your performance was very good. Soon you will be able to move pieces on the chessboard”.

We learned from psychologist that individualized therapeutic goals are best expressed in a positive way, e.g. “I want to open a door by myself” instead of: “I do not want to ask for help if I to open a door.” It is also known that training achieves the best results, when people are motivated. Psychological experiments showed for musicians who all trained the same exercise that training-related differences in changes of brain activation could be observed depending on whether the musicians were motivated or only forced to perform the exercises. Therefore, motivation seems to be one important aspect for training-related learning and hence presumably also for functional recovery post stroke. This is especially important, because success of the training tasks is only very slowly apparent after a large number of repetitions over a series of training sessions. Hence, there might be role for robot-supported training tasks once the level of routine day-to-day practice is achieved during therapy and the humanoid robot can help to assist training while supporting a patient’s motivation at the same time.

Pepper has a lot of sensors. Nevertheless, it is sometimes very difficult to analyse manual tasks when performed with paper and pencil. Therefore, a digitalization of training tasks might help a lot. It was our idea to replace paper by tablets. Applications had to be implemented to allow the exercises to be done on those tablets. We will shortly present this approach for the three training tasks AIMING, CANCELLATION and MAZE TRACKING

4.1 Digitalization of training task AIMING

A patient has to hit circles of different size from left to right and afterwards from right to left (see left part of **Fig. 3**) that are placed perpendicularly. Next, circles located horizontally on the table have to be hit (see right part of **Fig. 3**. Circle at the

button). This can also be seen from the digitalized version with two tablets represented in **Fig. 4**.

After ten perpendicularly presented circles are hit, connection lines between the circle at the bottom of the left part of Fig. 3 to all other horizontally presented circles have to be drawn. Once this is done a patient has to start hitting the perpendicular circles again.

A patient has one minute for this exercise. When a circle is missed the patient has to try to hit it again. The task AIMING has to be repeated four times (e.g. for times repeated practice over approximately one minute).

The goal of becoming faster while always meeting the precision demand of hitting both the smaller (more difficult to hit) and larger (easier to hit) targets affords continuously the adjustment of the motor behaviour, will only gradually be reached, and implies an underlying improvement of performance, i.e. motor learning.



Fig. 3. Manual training task AIMING.

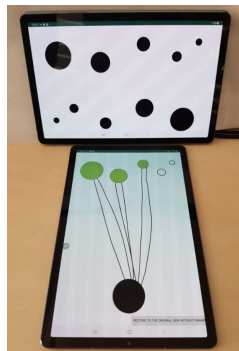


Fig. 4. Digitalized training task AIMING.

To implement the conventional AAT therapy manual procedure straight away would have triggered a verbal comment from Pepper that a circle was not hit in a correct way, if such an error occurred. However, we felt that it would be better since more direct, if the application on the tablet itself provides an appropriate feedback. In

this way the robot is not considered as a kind of “opponent” because it has to provide corrective feedback. Negative aspects are related to the tablet.

We considered the following types of feedback for the task “hit the circle” on the tablet:

- The circle that has to be hit is blinking
- Properly targeted circle becomes green
- A specific acoustic sound will be provided for correct and incorrect hits

Some of the suggested feedback are redundant. It will be evaluated in the future which kind of patients would like to have which feedback.

We are also yet not able to answer the question: “When should the robot take corrective actions?” Experiments with different versions of our application will show, which solution is appreciated by patients and which solution does not assist, but puts more pressure on the patients. Psychologists will help us to find the right solutions.

4.2 Digitalization of the Training Task CANCELLATION

Patients have to cross out symbols of O’s. They are not allowed to put the arm on the table. The arm has to be moved freely in the air. **Fig. 5** presents the tablet implementation for CROSSING. One has to start from left to right followed by the following line from right to left. Afterwards, crossing out has to be done from left to right again. At the top of the tablet one can see the remaining time (10 seconds) and the reached points (80). If the crossing out was done correctly the symbol gets a green background. If something was wrong, the background becomes red. In this case the attempt has to be repeated.

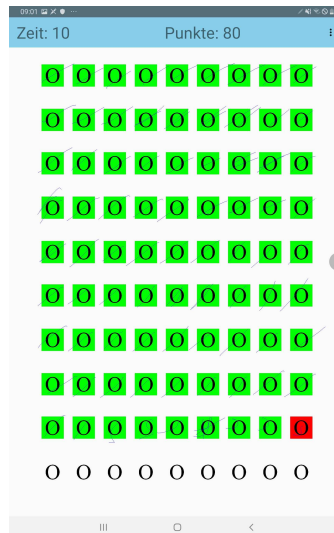


Fig. 5. Screenshot from the tablet application.

Again, the robot itself does not comment on errors as a first instance. It only comments on exceptions like unexpected stop of the activities. Additionally, Pepper informs about the result and relates it to previous exercises. This is combined with some motivational sentences.

4.3 Digitalization of the Training Task MACE TRACKING

For the MACE TRACKING exercise patients have to follow the mace as far as possible in one minute without hitting its boundaries. The result is captured as the percentage of the correctly followed mace. 100 percent are the complete mace.

In case there is time left, a patient can start to follow a second labyrinth, but this time s/he starts with the opposite direction. In this case more than 100 percent can be reached.

Each mistake, touching or crossing the border, results in one second penalty.

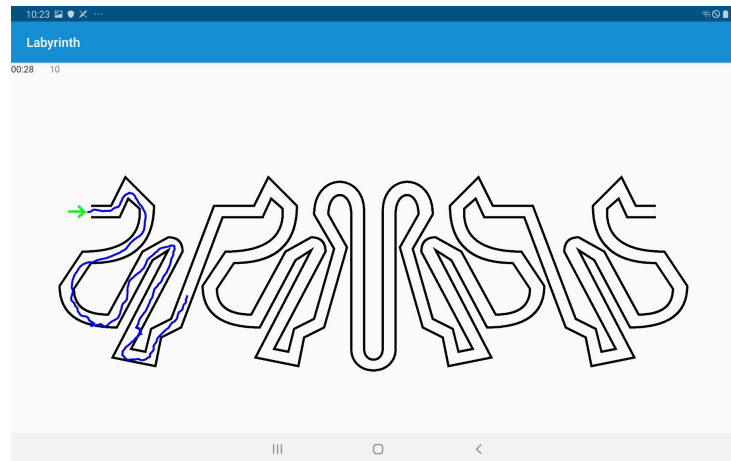


Fig. 6. Example of a training task of MACE TRACKING on a tablet.

We discussed the preconditions for the observation of three specific training tasks. The support for further training tasks is under development. Some videos can be found on the website of [3].

5 Modelling Collaboration

In this paragraph we want to model the task model for Pepper, the patient in DSL-CoTaL [4], that is based on features of Buchholz et. Al. [2]. The DSL supports the specification of a team model that reflects the collaborative activities of the participants.


```

team coop {
  root training = greeting >> train >> finish
  task greeting = pepper.greet |=| patient.greet
  task train = pepper.introduce >>
    patient.performs_exercises{*} [> patient.finishes_exercises
  task finish = pepper.finish_exercise ||| patient.bye
}

```

Fig. 7. Specification of a simplified Training collaborative model.

The specification of **Fig. 7** expresses that a training consists of a greeting part that is followed by the training (>>-enabling). Afterwards some ending activities have to be performed. In the greeting part pepper and the patient have to perform the task greet in any order (|=|-order independence). The role model of pepper and patient must have a task greet each. For the training pepper provides an introduction that is followed by the performance of the patient, which is an iteration that is finished by finishing the exercises. At the end pepper finishes exercises and in parallel (|||) the patient says bye.

```

role pepper {
  root root =
    greet >> training {*} [> finish_exercise
  task training =
    introduce >> starting >> train
  task introduce =
    tell_introd ||| show_pict1 ||| show_pict2
  task introduce pre patient.allInstances.greet
  task starting = say_can_we_tart
  task train =
    show_video ||| say_please_start |||
    comment_performance{*}[>end_commenting >>
    comment_results
}

```

Fig. 8. Role model of Pepper.

After greeting the training pepper has to wait until all patients (most of the time there is only one) greeted. First a short introduction is provided. In parallel to the talk two pictures are shown on the tablet of the robot that demonstrate important elements of the training task. Afterwards the patient is asked, whether the training can start. If ready, the robot says: "Please start" and later comments regarding the performance are provided. Finally, the robot comments on the results of the training.

Fig. 9 presents the captured situation at the end of a task while Pepper says the number of reached points and presents a graph with recently reached results.

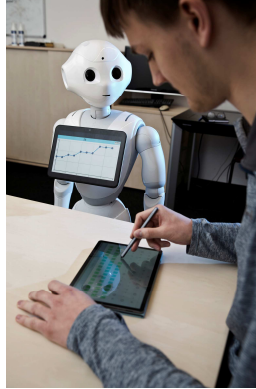


Fig. 9. Training Task CANCELLATION with presenting previous results by Pepper.

Currently, from our language DSL-CoTaL code can be generated to CoTaSE [2], to CTTE[12] and to HAMSTERS [11].

Fig. 10 shows the generated CTTE model of the specification of **Fig. 7**.

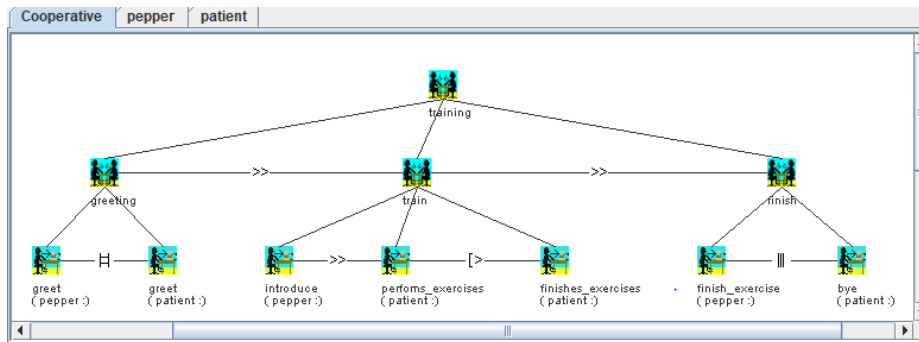


Fig. 10. Generated cooperative model for CTTE.

It would also be good to generate code directly for Pepper. To be able to do that some more specific language constructs like **say(text)** would be necessary. In [7] we provide a little bit more extended discussion about this aspect.

Additionally, we recognised a further problem: in the specification of the role model of Pepper. Sometimes Pepper has to ask: “Can we proceed?”. This has to be repeated until a positive answer is given. Even for the case of maximal three repetitions, the task model becomes quite complex.

It would be nice to have iterations that can end with a certain condition like a repeat until statement in programming languages. Our language DSL-CoTaL will be extended in this way.

Fig. 11 gives an impression how the language could look like. The task asking is repeated until the task yes is performed.

```

object answer { attribute: String value }
component askingForStart [text, seconds]{
  root roottask = asking{until answer.value == "Y"}
  task asking = say(text) >> yes [] no [] wait(seconds)
  task yes = input(yes) action answer.setValue("Y")
  task no = input(no) action answer.setValue("N")
}

```

Fig. 11. Object specification and component with iteration.

A modified version of the component (without the new language constructs) was used to generate the CTTE model for an instance of the generic component askingForStart with the two instance parameters “Can we” and “180”. The instance is created by *askingForStart[Can we, 180]*.

The result is presented by **Fig. 12**

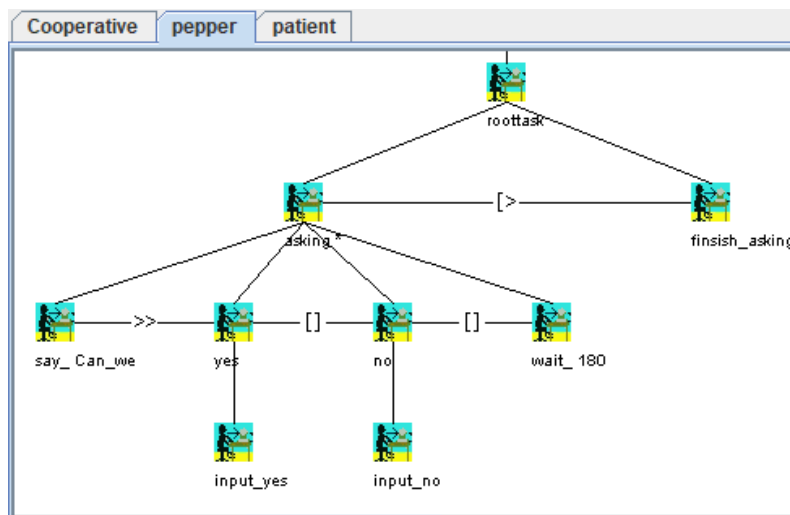


Fig. 12. CTTE model for instance *askingForStart[Can we, 180]*.

6 Lessons Learned

6.1 Methodology

Following the HCD approach worked quite well. Therapists were interviewed, the artefacts they work with were analysed and videos of their work with patients were taped. Textual transcriptions were produced and evaluated. Specific events were extracted and highlighted. In this way, exceptions for our applications were identified. As example we can mention the exception that patients did stop the crossing out at the

end of a line even when the instructions said that they should continue in the following line and all these lines were visible. We did not expect such a behaviour.

The therapist said in the case: “There is still time left. You should continue in the following line.” We do the same now with our application. The tablet app throws an exception when no activities can be identified at the end of a line within three seconds and informs the robot. The robot says the sentences of the therapist.

Modelling the behaviour of the therapist, patient and the robot was useful to understand the domain. It also supported the communication with experts. Especially, the animation of specifications helped a lot to establish a common ground between all participants.

As we know already from requirements engineering, we recognized that communication between psychologists, sociologists, expert from medicine and people from computer science in natural language is very problematic. Requirements are sometimes very vague like: “The robot has always to say some motivational words”. Neither is clear what the robot should say nor when. Modelling helped us a lot to be more precise and to understand the ideas of other people. Some ideas for further extending task model features were identified. Interaction will be the challenge for success [1].

6.2 Technology

We started to implement applets in java on the robot and on the tablets. Connection was realized directly via wireless local area network. Connection was often lost and applications had to start again.

Therefore, we used the Message Queuing Telemetry Transport (MQTT) technology [18]. It was designed as an extremely lightweight publish/subscribe messaging transport opportunity for the “Internet of Things”. MQTT was designed for low-bandwidth, high-latency and unreliable networks. Messages are sent via a server and it can be checked whether a message was received by recipient. If something went wrong, a message is sent again.

Using MQTT made our applications much more stable. We really recommend it for similar applications.

Having a stable message exchange established in our applications, we discussed the idea of implementing the behaviour of the robot externally. In this way, the robot has only an interpreter of messages that delivers the execution of commands. As a consequence, available technologies like knowledge management, machine learning or behaviour specification can be used without limitations.

The robot becomes something like a thin client. We have the impression that this software architecture facilitates digital implementation. However, we will investigate these aspects in more detail in the future. Alternative designs will be evaluated.

7 Summary and Outlook

The paper provides a case study of digitalizing Arm-Ability Training (AAT) for patients with mild to moderate arm paresis after a stroke causing focal disability. It was discussed how a social humanoid robot like Pepper can assist patients during AAT. Such a robot will play the role of a supervisor providing instructions and feedback, a motivator and supporter. Its application is restricted to highly standardized training schedules and excludes clinical decision making.

AAT training tasks need a digitalization to allow a smooth observation by humanoid robot. For three training tasks it was shown how training can be supported based on tablets and corresponding applications. It was shortly demonstrated how collaboration of patients and robot can be modelled by task models. Using a textual domain-specific language allows the generation of code for other tools that visualize the task models and allow their animation. A comprehensive domain-specific language (DSL) for robot actions will be a challenge for the future.

Currently, a data base is under development that contains user models of the patients. Beside general personal data and any individual details regarding the prescribed training, the data base has to store goals of the patients they want to reach. Such goals can be: “I want to be able to hold playing cards in my hands to be able to play with my grandchildren”. The robot will refer to these goals in order to motivate a patient. In addition, medical data will be stored that characterize the ability of a person like data characterizing the degree of (lack of) dexterity as assessed with the Box-and-Block-Test or the Nine-Hole-Peg-Test (Platz et al., 2005). From the point of view of a computer scientist, they are a kind of metrics [19]. These data allow very individualized hints. We hope to be able to exploit these resources in the future.

We will evaluate in the E-BRAiN project which kind of patients accepts the collaboration with Pepper using our apps.

References

1. Ahn, H. S., JongSuk Choi, Hyungpil Moon, and Yoonseob Lim. (2018): Social Human-Robot Interaction of Human-care Service Robots. In Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction (HRI '18). ACM, New York, NY, USA, 385-386. DOI: <https://doi.org/10.1145/3173386.3173565>
2. Buchholz, G., Forbrig, P. (2017), Extended Features of Task Models for Specifying Cooperative Activities. PACMHCI 1(EICS): 7:1-7:21
3. E-BRAiN: <https://wwwswt.informatik.uni-rostock.de/webebrain/>, last visited September 30, 2020
4. Forbrig, P., Dittmar, A., Kühn, M. (2018), A Textual Domain Specific Language for Task Models: Generating Code for CoTaL, CTTE, and HAMSTERS. EICS 2018 conferences, Paris, France, p. 5:1-5:6
5. Forbrig, P. and Bunde, A. (2020), Using the Social Humanoid Robot Pepper for Training Tasks, 1st Workshop on Adapted Interaction with Social Robots (cAESAR), IUI 2020, 17 March 2020, Cagliari, Italy

6. Forbrig, P. and Platz, T.(2020). Supporting the Arm-Ability Training of Stroke Patients by a Social-Humanoid Robot, accepted for IHET-AI 2020 conference, April 24 - 26, Lausanne, Switzerland
7. Forbrig, P and Bunde, A. (2020). Modelling the Collaboration of a Patient and an Assisting Humanoid Robot during Training Tasks. accepted for HCI 2020, Copenhagen, Denmark, 19-24 July
8. GBD 2015 Disease and Injury Incidence and Prevalence Collaborators. Global, regional, and national incidence, prevalence, and years lived with disability for 310 diseases and injuries, 1990–2015: a systematic analysis for the Global Burden of Disease Study 2015. *Lancet*. 2016; 388:1545–602.
9. Harte R, Glynn L, Rodríguez-Molinero A, Baker PM, Scharf T, Quinlan LR, et al. A human-centered design methodology to enhance the usability, human factors, and user experience of connected health systems: a three-phase methodology. *JMIR Hum Factors* 2017 Mar 16;4(1):e8
10. Manca, M., Paternò, F.; Santoro, C., Zedda, E., Braschi, C., Franco, R., Sale, A. (2020); The Impact of Serious Games with Humanoid Robots on Mild Cognitive Impairment Older Adults, *International Journal of Human-Computer Studies*, 2020, <https://doi.org/10.1016/j.ijhcs.2020.102509>
11. Palanque, P. and Martinie, C. (2016). Designing and Assessing Interactive Systems Using Task Models. In *Proceedings of the CHI EA '16*. ACM, New York, NY, USA, 976–979. <https://doi.org/10.1145/2851581.2856686>
12. Paternò, F. (1999). ConcurTaskTrees: An Engineered Approach to Model-based Design of Interactive Systems. *The Handbook of Analysis for Human Computer Interaction*, pp. 1–18
13. Petersen, S., Houston, S., Qin, H., Tague, C., & Studley, J. (2017). The Utilization of Robotic Pets in Dementia Care. *Journal of Alzheimer's disease: JAD*, 55(2), 569–574. doi:10.3233/JAD-160703
14. Platz, T., Pinkowski, C., van Wijck, F., Kim, I.H., di Bella, P., Johnson, G. (2005). Reliability and validity of arm function assessment with standardized guidelines for the Fugl-Meyer Test, Action Research Arm Test and Box and Block Test: a multicentre study. *Clin Rehabil.*, 19(4):404-11. doi: 10.1191/0269215505cr832oa.
15. Platz T, van Kaick S, Mehrholz J, Leidner O, Eickhof C, Pohl M (2009) Best conventional therapy versus modular Impairment-oriented training (IOT) for arm paresis after stroke: a single blind, multi-centre randomized controlled trial. *Neurorehabilitation and Neural Repair* 23:706-16
16. Platz, T., & Lotze, M. (2018). Arm Ability Training (AAT) Promotes Dexterity Recovery After a Stroke—a Review of Its Design, Clinical Effectiveness, and the Neurobiology of the Actions. *Frontiers in neurology*, 9, 1082. doi:10.3389/fneur.2018.01082
17. Platz T., Roschka S, Doppl K, Roth C, Lotze M, Sack AT, et al. Prolonged motor skill learning - a combined behavioural training and theta burst TMS study. *Restor Neurol Neurosci*. 2012, 30:213–24. doi: 10.3233/RNN-2012-110205
18. Pulver, T. (2019). *Hands-On Internet of Things with MQTT: Build connected IoT devices with Arduino and MQ Telemetry Transport (MQTT)*, Packt Publishing Ltd., Birmingham
19. Singer, B. and Garcia-Vega, J. (2017). The Fugl-Meyer Upper Extremity Scale. *Journal of physiotherapy* 63, 1, 53. <https://doi.org/10.1016/j.jphys.2016.08.010>.
20. SoftBank Robotics: <https://www.softbankrobotics.com/corp/robots/>, last visited November 11th, 2019