An Educational Framework for complex robotics projects

Simon Untergasser[1], Manfred Hild[1] and Benjamin Panreck[1]

Beuth University of Applied Sciences Berlin, Berlin, Germany,
simon.untergasser@beuth-hochschule.de,
home page: http://neurorobotik.de

Abstract. In robotics, several subject areas, such as electronics, mechanics and programming, come together. It can be a challenge to train new generations of robotics engineers in all the necessary areas. Humanoid robots in particular are very complex systems that require solid knowledge. At the same time, the goal of good teaching is to provide equal opportunities to all learners and to foster their individual abilities. In this paper, we present an educational framework that makes it possible to realize robotics projects of different levels of complexity in a short time. By partitioning into these levels, there are meaningful tasks for each level of difficulty in which learners can realize themselves. The framework is presented with a project course for humanoid robotics as example.

Keywords: humanoid robotics, project-based, university course

1 Introduction

Humanoid Robotics is receiving a lot of attention for its potential usage in different fields of society. Research is being done in care applications, socially assistive situations and education ([Gross et al., 2019], [Belpaeme et al., 2018]), among other areas. To ensure future research in humanoid robotics, new generations of researchers must be trained. However, their education can only be provided in a high quality if the students have access to appropriate hardware. On the one hand, there are commercially available products, such as Nao from Aldebaran Robotics ([Gouaillier et al., 2009]). On the other hand, there are open source projects like iCub ([Metta et al., 2008]) and robots built by research groups themselves like Myon ([Hild et al., 2012]). All these robots are highly complex systems that are fragile, expensive and not easy to maintain. As a result, there are usually few of these systems, if any, available for education. This can be a barrier to high-quality teaching. Another, already mentioned difficulty is the high complexity of these systems. Even for experienced researchers, it often takes a lot of time and effort to become familiar with such a system. Therefore, it is understandable that one cannot complete a large project on a humanoid robot within a short period of time, unless a lot of simplifications are made (like using preexisting solutions). Additionally, the different learning needs and performance levels of the students must be taken into account. This applies to any kind of complex problem that is to be solved with trainees. As long as the trainees have no experience in solving such tasks, a lot can go wrong. This includes not only the availability of possibly required hardware, as in the case of robotics. A group of trainees is always diverse in their level of knowledge of individual content. The level of comprehension capacities also varies naturally. How can we help weaker learners? Frustration should be avoided to guarantee satisfying results (see [Meindl et al., 2019] for possible consequences of frustration). And how can we ensure that the stronger learners are not bored when the content is taught more slowly? What happens if certain teaching content is not understood by individuals? Everyone has strengths and weaknesses and there should be no systematic disadvantage. Last but not least, hardware, as mentioned above, is often a bottleneck. Not only must availability be ensured, but also functionality. How to deal with technical problems, whether or not they are the fault of the learners? We address these questions with a didactic concept that makes it possible to implement complex projects even in a large course with many students. It provides equal opportunity for students with different performance levels while still delivering measurable results. With smooth transitions between different levels of complexity, frustration does not arise for weaker learners, nor does motivation disappear for the better ones. Furthermore, our curriculum can handle technical problems of complex hardware. We show how this concept is realized in a project course in the program *Humanoid Robotics* at Beuth University of Applied Sciences in Berlin. For this university course, we introduce three levels of increasing complexity. For each level, different hardware requirements are given and different contents of the accompanying lectures are relevant. Nevertheless, there are enough meaningful tasks to satisfy all needs and performance levels. In addition, learners have the opportunity to participate in a research project with their work and have direct contact with future clients.

The paper is structured as follows. In section two, the didactic concept is described in detail based on the robotics course example. Part three describes the integration of the curriculum into the research project *RoSen*. The paper concludes with part four.

Related Work

Many studies on robots in education use these machines to deliver teaching content or to motivate students. For example, the authors of [Donnermann et al., 2020] use the robot Pepper to provide individual tutoring as exam preparation. A similar situation is established in [Rosenberg-Kima et al., 2020]. Here, the robot Nao serves as a moderator of a group assignment. Another field of application is the use of robots as a mechatronic example for engineering subjects (see for example [de Gabriel et al., 2015]). Close to the approach of this work is [Danahy et al., 2014]. Here, LEGO robots were used over a period of 15 years to implement various teaching programs at universities. Different engineering topics such as mechanics, electronics and programming were transported in the form of robotics projects. Another interesting platform is presented in

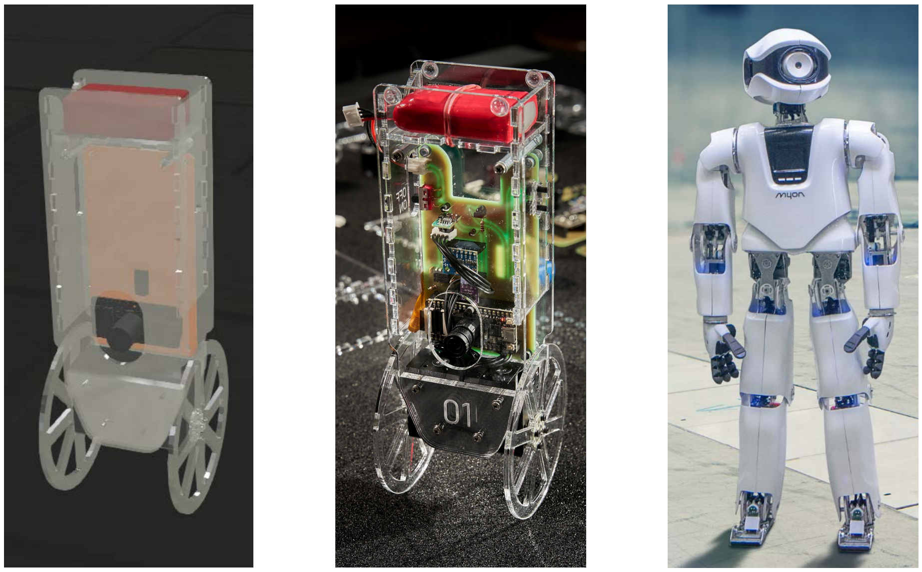

Fig. 1. Students start in the simulation, shown on the left side. When successful, they get a real MiniBot shown in the middle. At the end of the semester, they get the chance to test their code on a big humanoid robot like Myon shown on the right (source: Messe Berlin).

[Čehovin Zajc et al., 2015]. The authors describe a self build mobile robot based on a iRobot Roomba. They use their TurtleBot inspired robot to teach a university course for the development of intelligent systems. However, all these approaches do not do justice to the wide spectrum of knowledge students of humanoid robotics need to know.

2 Concept

This section first describes the problem in more detail. Then it outlines the proposed concept in the robotics course.

In any learning situation with multiple learners, you face the challenge of having to serve different levels of proficiency. Some of the learners understand the content faster than others. Typically, the goal is to maintain an average speed and level of complexity so that most of the group can follow along. This can mean that the material is nevertheless too difficult or fast for some learners, so that frustration can spread among this part of the group. These learners then have to spend more of their free time on the subject in question. As a result, they may miss out on other courses. At the same time, the better learners are bored because the material is not challenging enough for them or is taught too slowly. Often it is particularly important to keep these learners motivated, as the entire course can benefit from their level of performance. This is especially true in project courses, where students work together to achieve a larger goal. To address this issue, a multi-level didactic approach is presented below. It allows the teaching content to be adapted to different personalities and still get valuable results from all participants. At the same time, it provides measurable results in order to make fair assessments. The project course "Humanoid Robotics", in which complex robotics projects are realized, serves as an example. The key concepts are:

- multiple levels of complexity
- flexible, permeable boundaries between levels (you can level up anytime)
- meaningful tasks in each level
- measurable, fair outcomes at all levels
- as much independence from technical dependencies as possible

It is possible that there are teaching contents which can not divided into several levels. However, with some effort and abstraction, it is possible in many cases. For example, one goal of the research project *RoSen* (described in part 3) is using a complex humanoid robot like Myon (see figure 1 on the right). This can only be achieved if all intermediate steps are understood. These include, for example, controlling the actuators, visual processing of camera data, or handling the up to 32 degrees of freedom. However, concentrating on simpler subtasks of the bigger goal allows for a much broader range of projects. For example, localizing the robot in complex environments and building a map of the environment are essential parts of the RoSen project. To solve this task, however, the robot Myon is not necessary in the first step. All that is needed is a small analogue, called *MiniBot* (shown in the center of Figure 1), which has the necessary capabilities and degrees of freedom so that relevant algorithms can be tested on it before being applied to Myon. This MiniBot provides a huge reduction in complexity. It also circumvents potential hardware bottlenecks, since the MiniBot is relatively inexpensive to produce and therefore each student can have their own robot. However, the lowest level of complexity is formed by a simulation of the two robots (large and small) in the simulation environment WeBots ([Michel, 2004]). This virtual environment is used for the development and initial testing of algorithms and project ideas and serves as a playground for the first level. Figure 1 shows the simulated MiniBot on the left. In the transition to the second level the code developed by the students can be transferred to the MiniBot and adapted to the real world. The transition to the third level is achieved when everything works on the MiniBot. Then the students get the opportunity to test their project developments on the large robot. The three-level curriculum of the example presented here is shown in Figure 2. The total 15-week course is divided into three blocks of equal length. The lower part of the blocks list the main course contents. In the following, the two courses forming the formal framework of the robotic projects are first outlined in terms of content. Building on this, the concepts and contents of the three levels are described in detail.

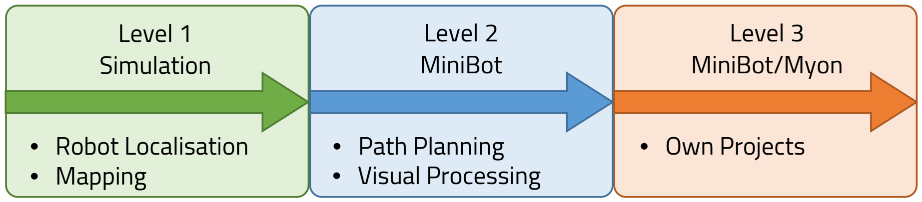

Fig. 2. The timeline of the project course is divided in three blocks. Students start in the simulation, continue with the MiniBot and can end up programming Myon. Contents of the student tasks are listed in the lower part of the blocks.

2.1 University Courses

The *Humanoid Robotics* project module forms the formal framework for the students' project work. The details for the WeBots simulation environment (see next section) and the simulated worlds and robots in it are presented and discussed here. Additionally, the hardware and programming details of the robots are introduced. Students can carry out projects alone or in small groups of up to three people. The examination consists of three videos, which are due at the end of the respective blocks. In these videos, the project group members describe their current progress, any problems and their solutions. The expected content relates to the content of the block, but of course also to the intended projects. In the first five weeks only the simulation is used. Here the students are given tasks that build on each other in order to get to know the simulation environment and the behavior of the robots better. This already includes the first algorithms for robot localization and map creation ([Durrant-Whyte and Bailey, 2006]), which are implemented and tested as part of the tasks. The students also get to know extensions and improvements of the algorithms, which they can later implement and test as possible projects. In the following five weeks, the students who have successfully submitted their first video will each receive a MiniBot. The first tasks are to let the real MiniBot balance and to port the code that has already been written for orientation. Here it is of particular didactic relevance to overcome the difference between the simulated world and the real world, where, for example, sensor values are always noisy. In the course of the second block, the students are encouraged to come together in groups and to develop project ideas. In the second video, at the end of the second block, the MiniBot is expected to balance, move when switched on and form a map. A sketch and first developments regarding the project ideas are also expected here. Individual performance can vary greatly, which is why there are only a few predetermined expectations. The third block now consists of the implementation of the project ideas on the MiniBot and, in the case of successful projects, the adaptation of the code for Myon. The lecture *Machine Learning* deals with the required algorithms. The contents are adapted to the timeline in figure 2 to enable successful projects. Here, too, there is a division into three subject blocks. The first block deals with the mathematical basics of the course (probability theory, graph theory, parti-

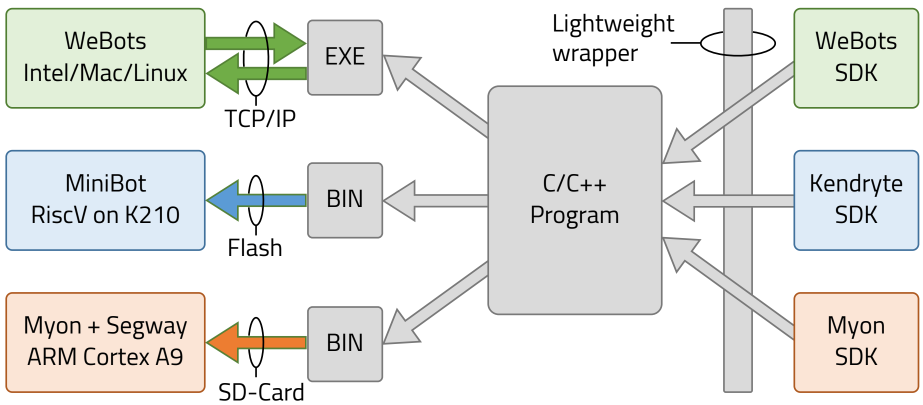

Fig. 3. The C-Program in the middle is written by the students in the course of their projects. There are different SDKs for all platforms available, since they all have different hardware and therefore different implementations. These differences are hidden below the SDKs.

cle filter, etc.). In addition, the robot localization and map formation (SLAM) is dealt with in detail. The characteristics of the different robotic platforms, for example, the specific features of the sensor values, are taken into account here. The second part deals with neural networks in general and the visual processing with the help of these networks in particular. Architectures that are suitable for mobile robot applications are examined in more detail. These are so-called light weight networks such as MobileNet ([Howard et al., 2017]) and YOLO ([Redmon et al., 2016]). By the end of the second block, the students have got to know all the algorithms for their projects. In the third part of the lecture, further algorithms from machine learning are presented, but they have no direct relation to the projects (e.g. regression, decision trees, boosting, etc.). Platform-specific SDKs (Software Development Kits) have been developed to make the transitions between simulation, MiniBot and Myon as easy as possible. The same functionalities, independent of the robotic platform, are available to the students. These are implemented differently in the background on the different platforms. In the course of the lectures, however, it will be discussed exactly where the differences are and what to look out for (e.g. different number of sensors, different processing capabilities). Figure 3 shows the merging of the different platforms on a conceptual level. The common basis is always the C code, which is created by the students as part of their projects. Depending on which platform they want their code to run on, a different SDK is integrated. This enables easy porting of the code between all platforms once the SDKs are available. The details of the platforms are described in more detail below.

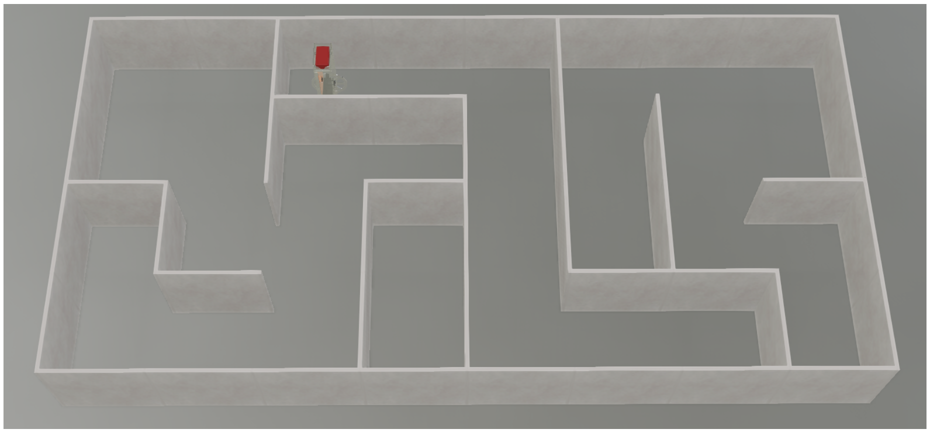

Fig. 4. The initial simulated world for the MiniBot experiments. The walls within a two square meters rectangle can be added or removed according to a raster.

2.2 Simulation

To maximize the efficiency of learning, the concept described above was developed in close collaboration with students. To this end, selected students were given the opportunity to work on specific sub-aspects. One of these sub-aspects was the development and construction of the MiniBot (see section 2.3). Another part was to create a simulation of the MiniBot (see figure 1 on the left) and a corresponding world in the simulation environment WeBots. As can be seen in figure 4, this world consists of a set of walls placed on a table. Within the rectangle of two square meters, the walls can be added or removed as desired, creating different room geometries. The simulation corresponds exactly to a model in the university's lab, where the real MiniBot can drive around. The real model also consists of movable walls of the same size as in the simulation. The algorithms tested in the simulation can therefore be tested in the exact same environment in reality. Accordingly, the simulated MiniBot is also an exact replica of the real robot. This includes shape, weight and components, but also the driving behavior and sensor qualities. As can be seen in figure 3, the students' C program communicates with the simulation environment via a TCP/IP connection. In order for the communication to work, a C program is running in in the WeBots Simulation, which executes a TCP/IP server and handles communication (this includes reading sensor values and writing motor commands). The TCP/IP server sends the sensor values (including camera image) to the central C program and receives the motor commands from it.

2.3 MiniBot

When the students are ready to transition from the simulation to the real world, they receive a MiniBot as shown in the middle of figure 1. The central computing unit of the MiniBot is the Kendryte K210 with a Risc-V dual core processor and the so-called Kernel Processing Unit (KPU), which accelerates the computation of Convolutional Neural Networks. The K210 is mounted on the Maix BiT board from Sipeed. The set from Sipeed includes a microphone, camera, SD card slot and a LCD display. In addition to that, a gyro sensor, a time of flight sensor, some switches and an infrared receiver for a remote control were built onto the MiniBot (see figure 5 on the left for all parts of one robot). The battery was placed at the very top for better weight distribution. The drive is provided by two Dynamixel XH430 from Robotis. The wheels, as well as the body of the MiniBot are made from acryl (PMMA) with a laser cutter. Figure 5 also shows the charger and power supply for the battery. To run the same central C code from figure 3 on the MiniBot, it is compiled into a BIN file using the MiniBot's SDK. This BIN file is then flashed onto the K210. It is also possible to place neural networks in the SD card's memory and then run them using the KPU. However, the implementation details are hidden in the background in the respective SDKs.

2.4 Myon

Finally, if the students handle the MiniBot successfully as well, they can test their code on the large robot Myon (see figure 1 on the right). For this purpose, a student developed a drivable lower body, the prototype of which can be seen in figure 5 on the right side. This development is part of the student's final thesis. Also, part of the work is to bring Myon into WeBots, including the mobile lower body. So far the head with its functionalities is already realized in the simulation. Due to the mobile lower body, the possible behaviors of the large robot correspond to those of the MiniBot. The algorithms developed by the students can therefore be brought to the large robot without having to develop everything from scratch. Of course, minor adjustments are necessary. For example, Myon has eight time of flight sensors, whereas the MiniBot has only one. Also, the size of the wheels in relation to the body size is completely different. However, the adaptation of the parameters is part of the didactic unit of the last block and is an important lesson to learn.

3 Integration of Research and Education

In addition to realizing their own project ideas, the students of the current semester were involved in the research project RoSen (Robots in Senior Living Facilities). In the following, the project process and its goals are described first. Then follows a description of the students' role in the project and the advantages and challenges it poses to them.

Project RoSen

The RoSen project is a cooperative project between two universities and a housing cooperative that operates senior housing facilities. With the help of conversa-

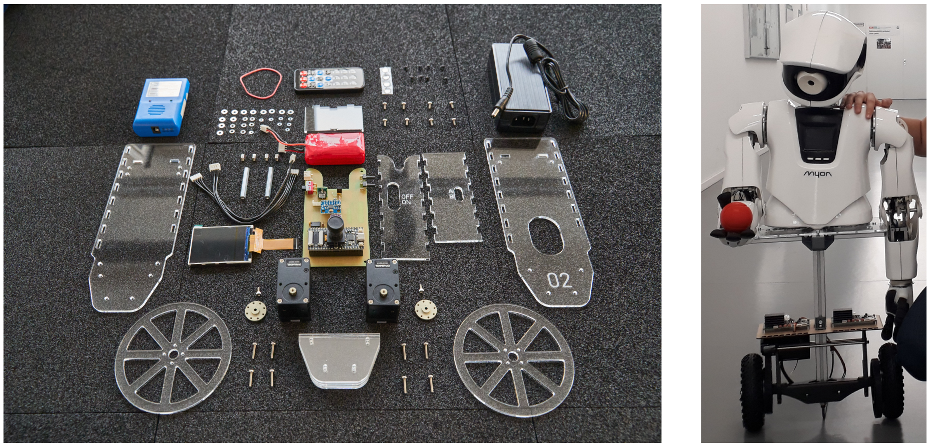

Fig. 5. Left: Parts of a MiniBot, which will be assembled from students in collaboration with teachers. Right: Myon with the prototype of its drivable lower body.

tions between students and residents of the senior housing facilities, the project aims to discuss wishes and needs of the participants. These are then tested for their technical feasibility. Together with the residents and technical experts, the students develop possible application scenarios to create a win-win situation. After prioritization and further elaboration of the ideas, the agreed on applications are implemented and the robots are then brought into interaction with the seniors for defined time periods. These interactions are observed and evaluated. In addition, there are surveys of all participants on the fulfillment of the expected effort and benefits. The primary goal, then, is to gather and analyze the needs and expectations of potential users, as well as the young junior technologists.

Integration into the project module

In the third block of the project course, one of the students' tasks is to contact residents of the senior citizens' residence. Due to the health situation, the planned interviews will not be conducted in person, but by letter or telephone. For this purpose, the students write a welcome letter to selected seniors, in which they introduce themselves, formulate their concerns and make suggestions on how to proceed. Optionally, the same is done by telephone. In the course of several conversations, the above-mentioned project goals are worked out together with the seniors. In the module, data protection, appropriate forms of approaching and the methods for finding the needs of potential users are discussed. This gives the students the opportunity to align their robotics projects with the RoSen project and the wishes and needs of the seniors. They can test and evaluate their ideas directly in interaction with users. They can try out the algorithms they have learned and implemented (such as robot localization and recognition of visual impressions) in a realistic environment. Direct involvement in active research also provides a unique insight. Interested students can also participate in the project after the end of the semester in the context of final theses. Nevertheless, it is also a challenge for those involved. It is not just a technical event where everyone can program for themselves. The social component through direct contact with potential users opens up completely new perspectives.

3.1 Results

After the first semester of the presented university courses we can report first results. All participating students (17 people) have completed the course and all of them have realized a project. We observed a typical distribution of performance levels, some easier projects, most of them with an intermediate level of difficulty and some very challenging ones. Some results of the projects will now be incorporated into the hardware and software, so that the base system for the next generation of students already starts at a higher level. One third of the students this semester are interested in continuing the projects in their bachelor thesis.

4 Conclusion

When young recruits learn about complex machines, it usually takes years before they are good enough to carry out larger projects. It's the same with humanoid robots. Understanding and being able to enable these very complex machines to do something is a tedious task. This work presented how such complex projects can nevertheless be realized in a short time. The described educational framework meets all the needs of the learners. By dividing the study contents into several difficulty levels, projects with different demands can be realized. Thus, every learner has a chance to successfully complete a project. For the specific example of humanoid robotics projects, three of these levels were introduced. Students were first taught to program the virtual version of the small robot MiniBot in the simulation environment WeBots. Each student who was able to successfully implement the algorithms from the accompanying machine learning lecture here will receive a real MiniBot. In terms of functionality, this MiniBot is a small copy of the large robot Myon. The transfer from simulation to the real world has many obstacles. Those who successfully master these obstacles and enable the MiniBot to find its way around are allowed to test their own developments on the large robot. To enable these seamless transitions between platforms, the MiniBot was developed along the lines of Myon. A virtual version of the MiniBot was then built in WeBots. A virtual version of the Myon is also nearly complete. In addition, SDKs for all platforms have been developed, allowing all platforms to be programmed with the same code. Through the RoSen research project, students can directly try out what they have learned in an application-oriented environment and practice initial customer contact. In each level of the curriculum there are enough topics that can be worked on by the learners so that they can contribute to the complete project in a meaningful way.

References

[Belpaeme et al., 2018] Belpaeme, T., Kennedy, J., Ramachandran, A., Scassellati, B., and Tanaka, F. (2018). Social robots for education: A review. *Science robotics*, 3(21).

[Danahy et al., 2014] Danahy, E., Wang, E., Brockman, J., Carberry, A., Shapiro, B., and Rogers, C. B. (2014). Lego-based robotics in higher education: 15 years of student creativity. *International Journal of Advanced Robotic Systems*, 11(2):27.

[de Gabriel et al., 2015] de Gabriel, J. M. G., Mandow, A., Fernández-Lozano, J., , and García-Cerezo, A. (2015). Mobile robot lab project to introduce engineering students to fault diagnosis in mechatronic systems. *IEEE Transactions on Education*, 58(3):187–193.

[Donnermann et al., 2020] Donnermann, M., Schaper, P., and Lugrin, B. (2020). Integrating a social robot in higher education–a field study. In *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 573–579. IEEE.

[Durrant-Whyte and Bailey, 2006] Durrant-Whyte, H. and Bailey, T. (2006). Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110.

[Čehovin Zajc et al., 2015] Čehovin Zajc, L., Rezelj, A., and Skočaj, D. (2015). Teaching intelligent robotics with a low-cost mobile robot platform.

[Gouaillier et al., 2009] 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. IEEE.

[Gross et al., 2019] Gross, H.-M., Scheidig, A., Müller, S., Schütz, B., Fricke, C., and Meyer, S. (2019). Living with a mobile companion robot in your own apartment-final implementation and results of a 20-weeks field study with 20 seniors. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 2253–2259. IEEE.

[Hild et al., 2012] Hild, M., Siedel, T., Benckendorff, C., Thiele, C., and Spranger, M. (2012). Myon, a new humanoid. In *Language grounding in robots*, pages 25–44. Springer.

[Howard et al., 2017] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

[Meindl et al., 2019] Meindl, P., Yu, A., Galla, B. M., Quirk, A., Haeck, C., Goyer, J. P., Lejuez, C. W., D'Mello, S. K., and Duckworth, A. L. (2019). A brief behavioral measure of frustration tolerance predicts academic achievement immediately and two years later. *Emotion*, 19(6):1081.

[Metta et al., 2008] Metta, G., Sandini, G., Vernon, D., Natale, L., and Nori, F. (2008). The icub humanoid robot: an open platform for research in embodied cognition. In *Proceedings of the 8th workshop on performance metrics for intelligent systems*, pages 50–56.

[Michel, 2004] Michel, O. (2004). Cyberbotics ltd. webots™: professional mobile robot simulation. *International Journal of Advanced Robotic Systems*, 1(1):5.

[Redmon et al., 2016] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

[Rosenberg-Kima et al., 2020] Rosenberg-Kima, R. B., Koren, Y., and Gordon, G. (2020). Robot-supported collaborative learning (rscl): Social robots as teaching assistants for higher education small group facilitation. *Frontiers in Robotics and AI*, 6:148.