Athens Journal of Education - Volume 11, Issue 3, August 2024 – Pages 213-226

Supporting Distributed Learning through Immersive Learning Environments

By Carsten Lecon [*]

In this paper, we describe a teaching scenario using a virtual environment (known also in the context of the 'metaverse'). This is motivated by the challenges that arise during the pandemic. More and more teaching scenarios are transferred to online learning settings, which allow learning at any time and at any time. One of the possibilities are virtual 3D environment. These allow more intensive immersion than for example video conferences. Furthermore, they offer new didactic concepts, for example, for group activities. The benefit of using virtual 3D environments we demonstrate by a concrete learning scenario: the simulation of robot programming. A further advantage when using virtual 3D environments are personal assistants (conversational/ pedagogical agents), for example, to the ease the work load borne by teachers; meanwhile, this works well also with natural language due to advantage stage of artificial intelligence implementations.

Keywords: e-learning, e-teaching, immersion, virtual 3D rooms, gamification, cyber-physical systems

Introduction

Since the pandemic epidemic, it was common for teaching to switch from the classroom to online places like, for example, video conference systems. But even now we still have challenges, in addition to the aftermath of the corona: For example, the current crisis is leading to much higher energy costs, which concerns the arrival to the training location and in particular external training (ride costs). Working and learning from home has also become more and more established. To create a learning environment that is as authentic as possible, virtual 3D environments are an option. We will go into the special properties of virtual 3D rooms and show their suitability for training, especially in university teaching and adult education.

Since there was not enough time for the creation of e-learning self-learning materials (self-directed, asynchronous learning) and synchronous accompaniment of the learning process (at schools, universities, and also in companies) was still expected, appropriate on-line tools were used, especially video conferencing systems. However, to achieve a better sense of presence, virtual 3D rooms are available in which participants can act as avatars ("alter ego") in an environment conducive to learning. In the following, we will explain the properties of such virtual worlds and use specific scenarios to illustrate their effectiveness.

First, we will discuss the general properties of 3D virtual spaces (Section 2). We use this environment to implement a learning scenario for computer science

[*]Professor, Aalen University, Germany.

https://doi.org/10.30958/aje.11-3-3 doi=10.30958/aje.11-3-3

students in higher semesters (Section 3). The virtual environment designed for distributed learning should be integrated into the existing e-learning infrastructure (Section 3.1).

The task to be completed is carried out in group work (collaboration). What needs to be considered is explained in Section 3.2. Especially in the virtual 3D environment, in which the learners are connected in a distributed manner, pedagogical agents or conversational agents (Section 3.3) can be used as additional participants to provide support, including as companions when entering the virtual environment for the first time, for example, or to assist in the fulfilment of learning objectives, whereby the agents can be targeted to test and support different levels of competence (section 3.4).

The first results are depicted in Section 5. The paper ends with a summary and an outlook on the next steps.

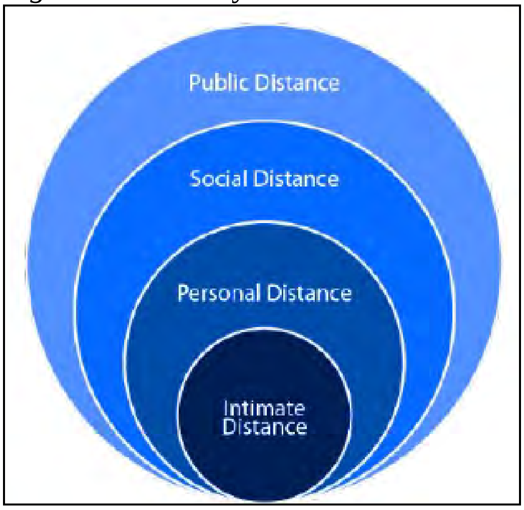
In this paper we only describe scenarios, which are currently being implemented. The evaluation of the scenario is currently in progress.

Virtual 3D Rooms

Immersion plays an important role in a virtual learning environment, that is, ‘immersing’ or ‘sinking’ in an artificially generated environment. Studies have shown that immersion in virtual environments has a major impact on learning efficiency. This is achieved, e.g., by allowing multiple perspectives, situated learning, and transfer (Dede 2009). Meanwhile, proven immersion environments are virtual 3D environments. [Churchill, EF, Snowdown, D] argue that there is a ‘paradigm shift’, because such virtual environments “provide a space that contains or encompasses data representations and users. Virtual 3D room support supports that the key aspect of many pedagogical methods and theories involves how students learn through observation, imitation, and interactions with other students and the instructor (Beck, 2019).

Regarding the virtual user experience, there are two aspects (Girvan, 2018): Sense of presence (“being in a virtual room”) and ‘Shaped by its inhabitants’ (the user - ‘inhabitant’ can become a content consumer and a content producer.). The user experience is also influenced by technical features (Girvan (2018): Avatars, multiple concurrent users, content creation tools, persistence, and space representation. The interaction of an avatar is significantly influenced by the so-called "embodiment", e.g., by appearance, gesture options, facial expression, etc. (Churchill & Snowdown, 1998). However, social proximity [39], that is, proximity to someone opposite, can also play a role in a virtual environment. "Walking through" a person is not usually possible, but if one avatar encounters another on their way (for example, if the user in question is inexperienced in controlling his/her avatar) and then automatically stands in front of you, this can lead to a feeling of distress. Since the virtual 3D world is basically just a piece of software, countermeasures can be taken if necessary, for example, by placing a "ban mile" around the avatar in order to maintain the social distance (Borkovic, Skovira, & Kohun, 2021, p. 39; Figure 1); this is 4 to 7 feet.

Figure 1. Proximity



Virtual 3D environments are advantageous in many ways in contrast to – for example – often used 2D video conferences. Simple conversational situations are perceived as more natural than in 2D environments (Garau et al., 2023). The virtual environment can be comfortable and friendly, so that this environment can trigger relaxation, silence, and pleasure in the user. Induced positive emotions can be a decisive factor for collaborative work and self-regulated learning, which is also reinforced by the intensive flow and presence experience (Boekaerts, 1996). In general, for collaborative learning and working, spatial and social immersion are central characteristics of these environments, as well as appropriate usability. Spatial immersion means feeling as being in another place as in reality (Boekaerts, 1996; Zinn, Guo, & Saga, 2016). In Lai (2011) it is postulated that the effect of collaborative learning depends on the quality of the interaction. With distributed learning, virtual 3D environments offer better possibilities than, for example, video conferences or asynchronous media such as Wikis or chat groups.

In principle, virtual worlds promote the activity of participants, since the avatar reduces risk perception, especially for shy and insecure students (Lueckemeyer, 2015).

It makes sense to offer the possibility to place (interactive) 3D objects in such a 3D environment, which can also be interacted with in a protected, safe environment (Mantovani, & Castelnuovo 2003). In this way, digital twins can also be integrated into the 3D environment (Qi et al., 2021; Dembski, Wassner, & Letzgus, 2019). The size of objects can also be experienced particularly intensively in a virtual 3D environment.

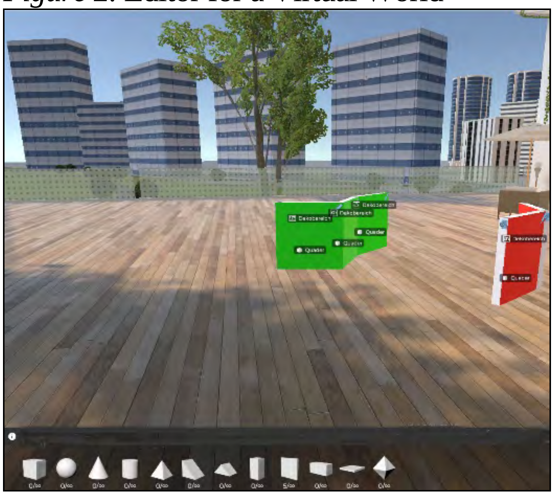
"Second Life" can be seen as a quasi-pioneer of virtual 3D worlds; this environment is still used in some cases, for example (Dede, 2009) in the context of ‘self-efficacy’. In the meantime, there are now many commercial and open source providers of virtual 3D environments.

Conceptual Framework

We will implement a kind of cyber-physical application in the context of a computer science course: A (virtual) robot has to navigate through a maze. To do this, a software template must be completed with the appropriate source code. Beforehand, learners have to build a parkour using an editor. This learning scenario can be expanded to include a gamification concept, in which learners have to go through the course as quickly as possible. In this case, participants (represented as avatars) not only 'see' the moving robot in the virtual environment, but also each other. Since the processing of this task takes place as a group work, we address several areas of competence. The use of conversational agents is also being considered; for example, we try to find out what kind of programming error it is and then offer appropriate help.

We use the virtual 3D learning environment for programming training in universities, among other things. While in a subproject this environment is used to learn a programming language (in the first semesters) (preliminary work see Lueckemeyer (2015)), in the project presented here (inspired by Lecon, and Herkersdorf (2014)) the students are supposed to steer a virtual robot through a maze and collect items, i.e. the corresponding algorithms should be developed and programmed. Beforehand, the students have to create a robot with a 3D modelling tool and provide appropriate animations. The robot movement is programmed in group work. The parkour is setup using an editor (see Figure 2). In addition to finding the way, the challenges here include dealing with collisions and, among other things, considering the physical properties (for example, how many items can be picked up at the same time?). The running time is measured and can be compared between the student groups; this is a kind of gamification.

Figure 2. Editor for a Virtual World

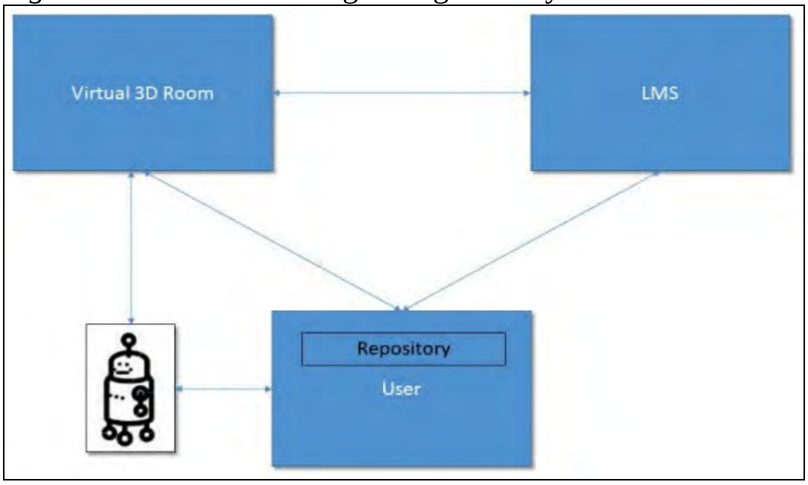


We use virtual 3D rooms as part of blended learning, which offers 'apparent effectiveness in offering flexible, timely and continuous learning" (Bizami, Tasir, & Na, 2022). Particularly immersive media, such as virtual 3D learning environments, are ideal for their implementation, among other reasons, because they are applicable and relevant to contemporary life and transferable to 'real world' situations (Bdidara, & Rusman, 2016).

Integration into an e-Learning Infrastructure

We determine appropriate interfaces for the seamless integration of virtual 3D rooms into existing teaching/learning infrastructures (Figure 3).

Figure 3. Interfaces: Learning Management System – 3D Environment - Repository



The user enters the virtual world through a Learning Management System (LMS), which most educational institutions have. The current programme code can be accessed via a repository. The code can be analysed and used as a basis for support by conversational agents (Section 3.3). In addition, the virtual robot's log data, which provide information about the success or failure of the task, can be sent to the LMS.

In the next expansion stage, it is planned that a real robot will transfer its data to its digital twin in virtual space. The virtual environment is generated beforehand using the sensor data from the real robot by driving through a specific area in the real world.

Collaboration

For our scenario, distributed learning takes place as collaboration. In Lai (2011) one definition of collaboration is 'a situation in which two or more people learn or attempt to learn something together and, more especially as joint problem solving'. In contrast to "cooperation", where each member makes an independent contribution, whereas in a "collaboration" everyone works at the same time to solve a problem or task. The advantages are specified in the meta-study (Lai, 2011), among other things, as follows: "[…] knowledge is constructed through interactions among collaborators. This approach emphasises that the whole group behaviour is more than the sum of the individual parts. 'An interesting question about how the group is made up, including what the different skills and previous knowledge are. The following statement can be found in Lai (2011): '[...] collaboration tends to benefit students with lower ability, while there appears to be no carryover effect for students with higher ability.' In general, heterogeneous groups seem to be the most ideal, especially for large groups, as stated and proved in Oliver, and Marwell (1988): "When groups are heterogonous, a larger interest group can have a smaller critical mass" (Macy, 1999). In this way, one ideally avoids the "social trap" (Oliver, & Marwell (1988):

- *Free-rider-problem*: A group member can benefit from the result of the group work without contributing anything themselves.
- *Problem of efficacy*: A group member does not benefit despite constructive contributions.

We were able to observe the effectiveness and creativity of the group work with a heterogeneous composition. In a course on virtual reality and animation techniques, interdisciplinary groups consisting of computer science students and user experience students formed. The respective core competencies (advanced programming knowledge on the one hand and design and 3D modelling expertise on the other) led to very mature VR projects in terms of functionality and human-computer interaction.

The efficiency of group work is also influenced by the so-called *social facilitation* (Harkins, 1987). This means that people who have prior experience working on the task act better when they are observed by other participants. This is easier to implement in a virtual 3D environment than, for example, in video conferences. On the other hand, people who are inexperienced and therefore insecure tend to be less productive when they feel like they are being watched. But because these people are partially anonymised by the visual representation as an avatar, they don't feel pressured. If they look out of the window in the real world or do research (in books or on the Internet), for example, the other participants in the virtual world do not notice. The "penetrating" look that is not visible in the virtual environment when the other person is waiting for an answer quickly reduces stress.

Conversational Agents

In general, the support of teachers or tutors is helpful or even necessary in learning settings. In a normal classroom, this is usually the case when the teaching staff is physically present. In the virtual 3D space, this can be done by the teaching staff represented as avatars. However, this only applies in synchronous learning settings. If learners want to use the virtual learning environment independently of the curriculum, their own virtual 'intelligent' virtual avatars to communicate with: conversation agents (Allouch, Azaria, & Azoulay, 2021) that act as 'pedagogical agents' (Terzidou, & Tsiatsos, 2014). Since the individual agents each act independently, any number of such "helpers" is possible, in principle.

In general, there are a few things to keep in mind when using conversational agents:

- Dialogue form: Should communication take place via text, speech bubbles, or natural language? In the latter case, artificial intelligence methods can be integrated.
- Type of appearance: Should the agent always be there (at what distance? Typically, in social distance; see above). Or should the agent appear on "call"? Or should you explicitly go to a certain point (e.g., a "counter")? Should the agent be visible and audible to everyone or only to a specific person? If this is the case, the other participants should at least be informed of its existence; otherwise, it can happen that it is "walked through", and communication with an "invisible spirit" would also seem strange.
- Appearance: Do you want the agent to dress formally or informally? It will probably be something in between: more casual clothing. Gender also plays a role. A neutral, gender-neutral visualisation could take place through a comic-like representation, but the agent should also radiate seriousness (trust). The agent should also be clearly recognisable as an agent. If you can customise your avatar, in extreme cases, the agent could look the same.

In order for an educational agent to be able to provide support, two implementation options are conceivable in our scenario: either the user addresses the agent directly with a question, or a need for intervention is automatically recognised; this can be the case if the user gets stuck or the solution (the software programme) is wrong or not optimal.

In the first case, natural language communication is required, as known from chatbots (see Adamopoulou, and Moussiades (2020) for an overview). In the other case, there must be a way to observe the actions of users. This in turn can be synchronous or asynchronous. In the case of synchronous observation, a "guardian" must be integrated into the software development system; in the asynchronous case, for example, the current status of the software stored in a version control system (VCS: Version Control System (Spinellis, 2004)) - Such a system should be used anyway and is accessed via the repository mentioned above.

Furthermore, we will use conversational agents to help learners navigate a virtual environment quickly by providing guidance and support; especially if they are entering the virtual environment for the first time (and perhaps an introductory tutorial either doesn't exist or hasn't been completed). In this way, the teachers are not bound by the time-consuming assistance, but can take care of the actual content.

In our context, we use graphically embodied agents (Allouch, Azaria, & Azoulay, 2021) which are visualised by a virtual body (avatar); they have, as well as voice-understanding and speech generation abilities, the interaction intensity can be increased by gestures and – to some extent – facial expressions. Another step towards the most effective human-computer interface is when the agent can both recognise human speech and is able to form and speak complete sentences (as, e.g., in Zargham et al., (2021)). The so-called "Virtual Agents" offer increased immersion and interactivity, a personalised experience, and the ability to convey complex concepts and processes in an engaging and interactive way.

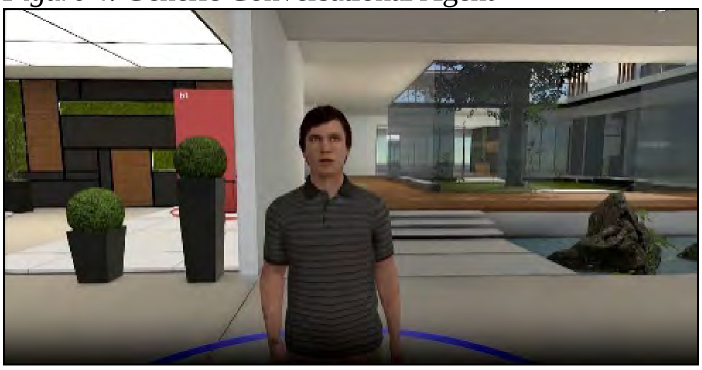
The actions of an agent can either be rule-based or be controlled by artificial intelligence. A rule-based behaviour is, for example, the implementation of an introductory tutorial at the beginning of a session in virtual space or the detection of problems in software programming. In this case, the agent can react to the following situations, among others.

- The user has not made an entry for a long time – this can indicate that he/she does not know what to do. Based on the analysis of the source code created so far, the agent can suggest based on a comparison with the model solution.
- It will use wrong or useless software libraries. The agent can point out this and, if necessary, provide information on where to look for suitable libraries.
- The task will not be solved after a certain time. Here, the agent can refer to relevant learning materials or tutorials; in extreme cases, he can also disclose parts of the sample solution.

When simulating a real conversation situation, the counterpart (the agent) can be equipped with artificial intelligence for speech recognition and the formulation of answers. It is usually advisable to use existing systems, such as current *ChatGPT* (see, for example, Thorp (2023)). However, in such cases, licence fees apply, which must be considered when planning conversational agent-supported virtual learning environments.

A generic virtual assistant (Figure 4) was implemented as a kind of "proof of concept", which responds to natural language and responds using a speech processor, the interaction logic was implemented with ChatGPT.

Figure 4. Generic Conversational Agent[1]



To describe learning goals (among others of virtual agents) and to verify compliance, we use common taxonomies.

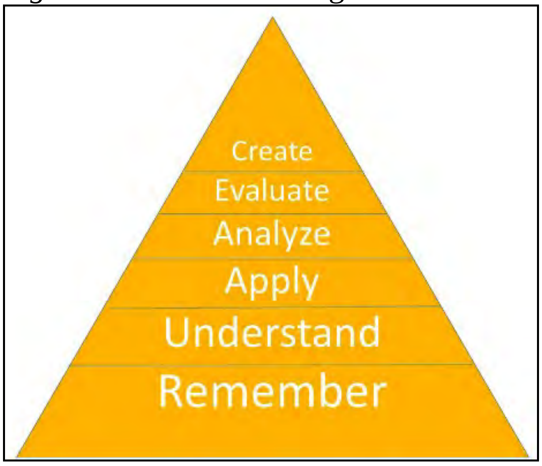
[1] https://tricat.net/neuigkeiten/generative-ai-meets-corporate-metaverse/.

Learning Goals: Taxonomy

Learning goals are specified by the competencies to be achieved.
One of the first acquisitions of skills is defined according to Bloom's taxonomy (Bloom et al., 1956). It is a taxonomy of learning goals in the cognitive domain consisting of knowledge, comprehension, application, analysis, synthesis, and evaluation. This taxonomy was designed for K-12 teachers and college instructors.

This was adapted by Anderson and Krathwohl with the aim of better applicability for curriculum design (Anderson, & Krathwohl, 2001): The Anderson-Krathwohl Taxonomy (AKT). These propose a two-dimensional matrix with four types of knowledge (factual, conceptual, procedural, metacognitive) and six dimensions of cognitive processes (remember, understand, apply, analyse, evaluate, create). The taxonomy hierarchy is shown as a pyramid in Figure 5.

Figure 5. Bloom s Learning Goals Taxonomy



The achievement of these competencies can best be formulated using appropriate verb forms for the levels.

Dreyfus describes a simplified approach using a five-stage model for adult skill acquisition (Dreyfus, 2004): Novice, advanced beginner, competence, proficiency, and expertise.

Furthermore, to keep the description of learning objectives simpler, these levels of competence are sometimes broken down into "professional competence", "interdisciplinary competence" and "particular methodological competence".

These terms are usually not precisely defined and must be adapted for the respective domain, which can also lead to misunderstandings. For example, in the context of programming education, the term "apply" can be interpreted for the execution or implementation of a procedure in a given situation, but also for the application of programming language constructs; the latter would be a misinterpretation (Kiesler, 2002).

Pedagogical principles, especially with regard to immersive blended learning, are proposed in Bizami, Tasir, and Na (2022): human agency, capability, self-reflection, double loop learning, nonlinear learning.

In our context (teaching and applying programming), we use an adoption of the AKT, described by considering the competences in Kiesler (2002): Factual Knowledge, Conceptual Knowledge, Procedural Knowledge, Meta-cognitive Knowledge and the above-mentioned process dimensions. These are combined in the domain of ‘programming competencies’.

In the scenario described, the students should reach the highest level: climbing to the top of the pyramid (Figure 4). For each process dimension, we will give an example (according to the sentences in Kiesler, (2002)):

- Remember: Knowledge of computer science-related topics should have been imparted and checked in the first semesters of the course.
- Understand: Understanding the syntax and semantics is a prerequisite for the development of software, and is taught in the lectures, which participants are required to attend. This also includes dealing with software development environments.
- Apply: The complete software consists of individual – partially standardised - procedures; certain software quality requirements have to be met; in the context of cooperative software development, uniform programming conventions should be adhered to.
- Analyse: Information relevant to the implementation must be extracted from the task description. In addition, it is important to find the most suitable ones from the abundance of existing software frameworks and libraries, as well as to look for known algorithms (for example, the A* algorithm for efficient way-finding (Hart, Nilsson, & Raphael, 1968).
- Evaluate: There are different algorithms to find the way and collecting the items. Some of these were covered in the lectures. It is now a matter of finding the right one and adapting it if necessary, also with regard to the time complexity.
- Create: An executable application (robot in action) should result based on the preliminary considerations of the previous competence levels.

In order for a conversational agent to be able to provide support, two implementation options are conceivable in our scenario: either the user addresses the agent directly with a question, or a need for intervention is automatically recognised; this can be the case if the user gets stuck or the solution (the software programme) is wrong or not optimal.

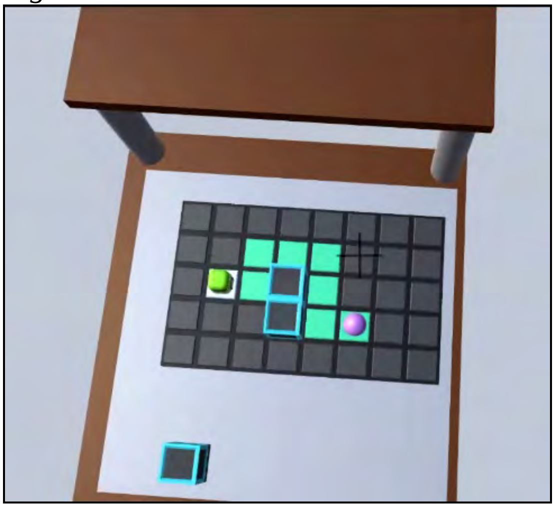
In the first case, natural language communication is required, as known from chatbots (see Adamopoulou, and Moussiades (2020) for an overview). In the other case, there must be a way to observe the actions of users. This in turn can be synchronous or asynchronous. In the case of synchronous observation, a "guardian" must be integrated into the software development system; in the asynchronous case, for example, the current status of the software stored in a version control system (VCS: The version control system (Spinellis, 2004)) can be accessed; such a system should be used anyway (see also Figure 3, ‘Repository’).

Results

Regarding the concepts mentioned above, we have implemented a robot simulation using the ROS interface (Koubâa, 2017). The course can be set manually; there is a corresponding editor for this. The programming itself should be done by the students. This is the next step, together with an evaluation. Before, to illustrate the optimal solution, we use the A* algorithm for path finding (later, the students have to find out this approach). The robot s commands are transferred via the client/server architecture.

The described scenario is a proof-of-concept (working application, see Figure 6).

Figure 6. Robot Simulation Environment



You can see on the picture a small maze, the robot from above (green), the target point (violet), and the proposed path (turquoise), calculated by the A* algorithm.

There is an effective integration of ROS with the game engine *Unity* in a virtual 3D environment, which creates an intensive learning experience for robot programming. The results of the programming task are visible to the students: the robot essentially drives through the participant's feet in the virtual environment. Real-time communication between the ROS and virtual environment is possible (Lecon, Soeser, & Stegmaier, 2024).

The immersive setting leads to an improvement in student engagement and understanding of complex programming concepts. The scalability of this project enables one to address a heterogeneous group of students by using different use cases with, regarding different (competence) levels of the students.

Parallel (in another subproject) conversational agents are implemented, which can also be used in our scenario for individual support of the teacher.

The next steps are as follows:

- Fully integration in the virtual 3D environment: up to now, we simulate the virtual environment; but the appropriate interfaces (publish-subscribe model) are existing.
- Evaluation by students of a face-to-face university (synchronous group work) and a distance learning university (asynchronous group work).

Conclusion

We have presented a robot programming learning scenario using a virtual 3D environment. Such an environment is meaningful useful, for example, if a teaching/ learning is not possible, for example during pandemic-related restrictions –, or when most teaching takes place online anyway: for distant learning courses.

The use of virtual worlds can overcome the methodological-didactic challenges faced by some courses that require face-to-face interaction and the development of practical skills, such as programming and laboratory seminars. The advantage of such environments is, for example, the more intensive social and spatial presence (immersion), so a more natural communication is possible in the three-dimensional environment than in a two-dimensional one.

We have presented a programming scenario as an example of an application. An algorithm is to be developed and implemented in order to steer a robot through a course previously created by the trainer or the learners themselves. The result can be 'seen' in the form of a 3D model within the 3D environment: the robot practically drives in front of your eyes.

The participants are embodied by avatars, which means social immersion in virtual space. Facial expressions can be simulated to some extent, but such an abstract representation can also contribute to more objective communication. In addition to the participants themselves, computer-generated agents can be integrated, for example, to relieve teachers by acting as a 'contact person' who can always be reached.

In our case, such conversational agents act for support in using the virtual environment itself and to advise on technical matters. The design of such agents is diverse; we have highlighted individual aspects.

Actually, we have a prototype as a proof-of-concept. Next, we will evaluate this setting as well in a presence university as an addition to traditional classroom teaching, as well as part of the online learning material at a distant learning university.

Acknowledgments

Part of this work is funded through the German Federal Ministry of Education and Research (Project 16INB2005C – AVILAB2).

References

Adamopoulou, E., & Moussiades, L. (2020). An Overview of Chatbot Technology. In I. Maglogiannis, L. Iliadis, & E. Pimenidis (eds.), *Artifical Intelligence Applications and Innovations*, AIAI 2022, *IFIP Advances in Information and Communication Technology*, 584, 373-383. Springer.

Allouch, M., Azaria, A., & Azoulay, R. (2021). Conversational Agents: Goals, Techniques; Vision, and Challenges. *Sensors, 21*(24), 8448.

Anderson, L. W., & Krathwohl, D. R. (2001). *A Taxonomy for Learning, Teaching, and Assing. A Revision of Bloom s Taxonomy for Educational Objectives*. New York: Addison Wesley Longma.

Bdidara, J., & Rusman, E. (2016). Towards a Pedagogical Model for Science Education: Bringing Educational Contexts Through a Blended Learning Approach. *Open Learning: The Journal of Open, Distance, and e-Learning, 32*(1), 6-20.

Beck, D. (2019). Special Issue: Augmented and Virtual Reality in Education: Immersive learning research. *Journal of Educational Research, 57*(7), 619-1625.

Bizami, N. A., Tasir, Z., & Na, K. S. (2022). Innovative Padagogical Principles and Technological Tools Capabiliies for Immersive Blended Learning: A Systematic Literature Review.*Education and Information Technologies*, 1-53.

Boekaerts, M. (1996). Self-Regulated Learning at the Junction of Cognition and Motivation. *European Psychologist, 1*(2), 100-112.

Bloom, B. et al. (1956). Taxonomy of Educational Objectives *1*, 20-24.

Boekaerts, M. (1996). Self-Regulated Learning at the Junction of Cognition and Motivation. *European Pychologist*, *1*(2), 100-112.

Borkovic, D. J., Skovira, R. J., & Kohun, F. (2021). Virtual Social Distancing: A Digital Ethnography of Online Learning. *Issue in Information Systems, 22*(4), 244-257.

Churchill, E. F., & Snowdown, D. (1998). Collaborative Virtual Environments: An Introducy Review of Issues and Systems. *Virtual Reality, 3*(1), 3-15.

Dede, C. (2009). Immersive Interfaces for Engagement in Learning, *Science, 323*(66), 66-69.

Dembski, F., Wassner, U., & Letzgus, M. (2019). The Digital Twin – Tacking Urban Challenges with Models, Spatial Analysis and Numerical Simulations in Immersive Virtual Environments. In *eCAADe + SIGraDI Conference - Architecture in the Age of the 4th Industrial.*

Dreyfus, S. E. (2004). The Five-Stage Model of Adult Skill Acquisation. *Bulletin of Science Technology & Society, 24*(3),77-181.

Garau, M., Slater, M., Vinayagamoorthy, V., Brogni, A., Steed, A., & Sasse, M. A. (2023). The Impact of Avatar Realism and Eye Gaze Control on Perceptual Quality of Communication in Shared Immersive Virtual Environments. In *Conference on Human Factors in Computing Systems – Proeeding Nr. 5,* 529-536.

Girvan, C. (2018). What is a Virtual World? Definition and Classification. *Education Tech Research Dev. 66*, 1087-1100.

Harkins, S. G. (1987). Social Loafing and Social Faciliation. *Journal of Experimental Social Pschology*, (3), 1-18.

Hart, P. E., Nilsson, N. S., & Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum-Cost Paths. *IEEE Transaction of Systems Science and Cypernetics*, *4*(2).

Kiesler, N. (2002). On Programming Competences and its Classification. In *Koli Calling '20: Proceedings of the 20th Koli Calling International Conference on Cmputing Educatoin Research*, November 19-22, 2002, Koli, Finland, ACM, New York, USA.

Koubâa, A. (Ed.) (2017). Robot Operation System (ROS) – The Complete Reference (Volume 1). *Studies in Computational Intelligence*, *625*.

Lai, E. R. (2011*). Collaboration: A Literature Review*. Research Report. Pearson.

Lecon, C., Herkersdorf, M. (2014). Virtual Blended Learning – Virtual Worlds and Their Integration in Teaching Scenarios. In *9th International Conference on Computer Science and Education (IEEE ICCSE)*, 22-24.08.2014.

Lecon, C., Soeser, S., & Stegmaier, A. (2024). Programming Teaching in a Virtual Environment Using the ROS Interface. In *18th Annual International, Education and Development Conference*, Valencia, Spain.

Lueckemeyer, G. (2015). Virtual Blended Learning Enriched by Gamification and Social Aspects in Programming Education. In *10th International Conference on Computer Science and Education (IEEE ICCSE 2015)*, 438-444. Cambridge, UK.

Macy, M. W. (1999). Learning Theory and the Logic of Critcal Mass. *American Sociological Review*, *55*(6), 809-826.

Mantovani, F., & Castelnuovo, G. (2003). The Sense of Presence in Virtual Training Skills: Aquisation and Transfer of Knowledge through Learning Experience in Virtual Environments. In G. Riva, F. Davide, & W. A. Ijsselsteijn (eds.), *Being There: Concepts, effects, and measurement of user presence in synthetic environments*. Amsterdam: Ios Press.

Oliver, P. E., & Marwell, G. (1988). The Paradox of Group Size in Collective Action: A Theory of the Critical Mass. *American Sociological Review, 58*(1), 1-8.

Qi, Q. F. T., Hu, T., Anwer, N., Liu, A., Wei, Y., Wang, L., et al. (2021). Enabling Technologies and Tools for Digital Twin. J*ournal of Manufactoring Systems,* 58(B), 3-21.

Spinellis, D. (2004). Version Control Systems. *IEEE Software, 22*(5), 108-109.

Terzidou, T., & Tsiatsos, T. (2014). The Impact of Pedagogical Agents in 3D Collaborative Serious Games. In *2014 IEEE Global Engineering Education Conference (EDUCON)*, 1175-1182. Military Museum and Culture Center, Harbiye, Istanbul, Turkey, 3-5 April 2014.

Thorp, H. H. (2023). ChatGPT is Fun, but Not an Author. *Science, 379*(6630): 313.

Zargham, N., Bonfert, M., Porzel, R., Doring, T., & Malaka, R. (2021). Multi-Agent Voice Assistents: An Investigation of User Experience. In *NUM'21: Proceedings of the 20th International Conference on Mobile and Ubiquitous Multimedia*, 98-107. December 2021.

Zinn, B., Guo, Q., & Saga, D. (2016). Entwickung von und Evaluierung der virtuellen Lern- und Arbeitsumgebung VILA. (Development and evaluation of the virtual learning and working environment VILA). *Journal of Technical Education, 4*(1), 89-117.