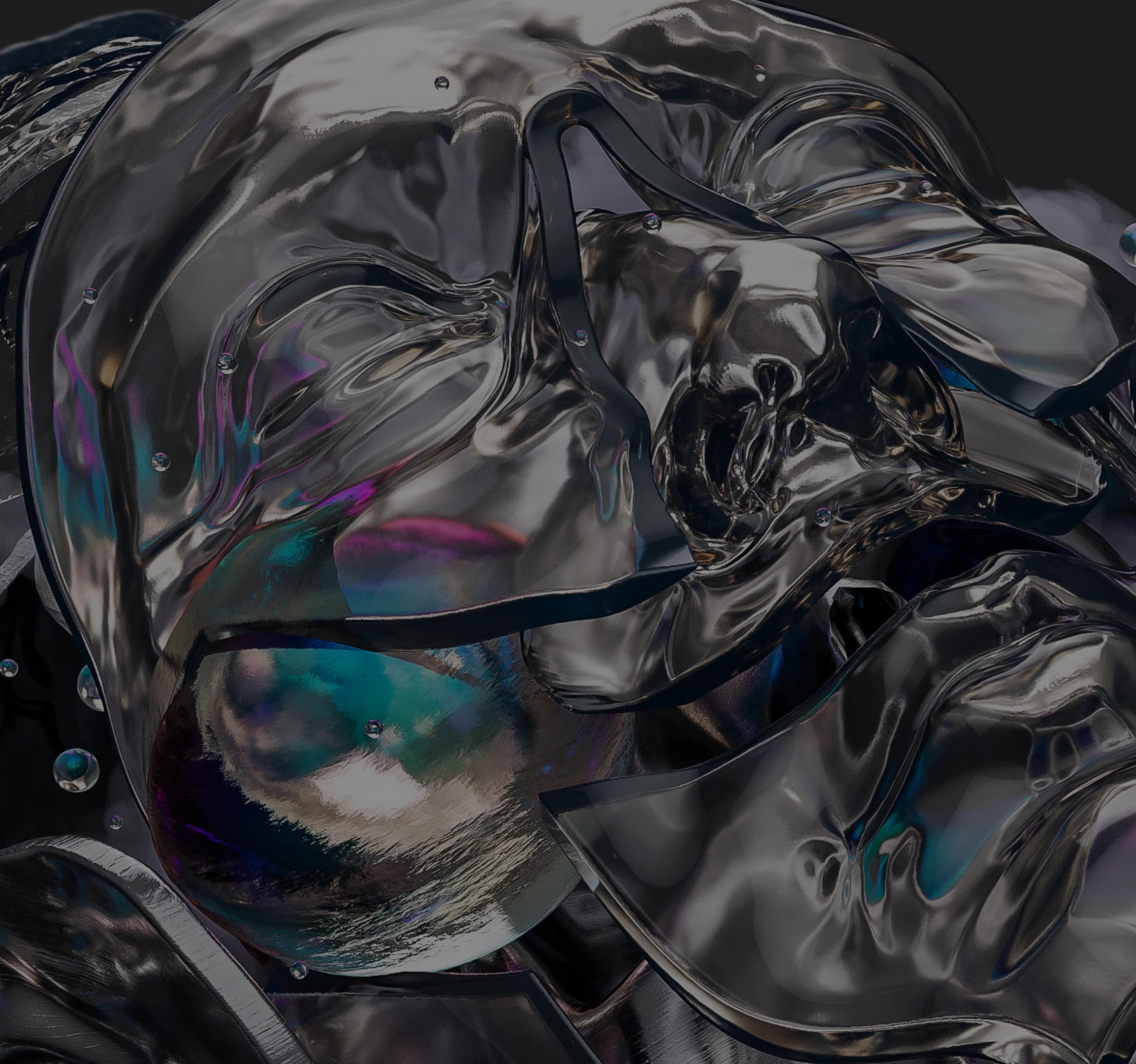


PROCEEDINGS OF THE
1st GAME TECHNOLOGY SUMMIT

part of

CLASH OF REALITIES

11th International Conference on the Technology and Theory of Digital Games

This document is made available by OPUS Würzburg, a publishing service of the University Library of the Julius-Maximilians University, Würzburg, Germany.

This document is licensed under the Creative Commons Attribution 4.0 International License (CC BY 4.0):
http://creativecommons.org/licenses/by/4.0
This CC license does not apply to third party material (attributed to another source) in this publication.

Year of publication: 2021

ISBN: 978-3-945459-36-2
DOI: 10.25972/OPUS-24577

Preface

High Impact Trends in Games – AI in Natural User Interfaces.
Proceedings of The 1^{st} Games Technology Summit as part of the Clash of Realities - 11^{th} International Conference on Art, Technology and Theory of Digital Games

Summit Topics

AAA companies and indie devs alike harness the power of Artificial Intelligence (AI), Machine Learning (ML) and novel means of user interaction to facilitate game design & development and to provide for more intricate, authentic, entertaining, immersive and fun experiences. The available tools, established approaches, methods and gadgets in development are as broad as their application. With respect to AI perspectives develop from classical search and planning algorithms as well as generative algorithms towards utilising machine learning methods, simulation and optimisation. Even self-adapting, real-time capable systems are proactively utilised in the games industry. With respect to novel approaches of user interaction, especially the integration of multiple channels of communication (multi-modal approaches), for instance considering gestures and voice at the same time, have paved the road for natural user interfaces that diminish the cognitive load of designers, developers and, particularly, the players of next gen computer and video games.

The 1^{st} Games Technology Summit was organised in two tracks: A scientific track and an industry track. We discussed how the successes in AI in Natural User Interfaces have been impacting the games industry (industry track) and which scientific, state-of-the-art ideas and approaches are currently pursued (scientific track).

Scientific Track

For the scientific track of this half-day event, we have accepted five submissions covering different aspects of how technological advancements impact the development of digital games and how advanced game technologies also foster use cases outside pure entertainment. The accepted submissions cover a broad range from dialogue generation for games to self-assembling agent ensembles as representations of the emerging field of AI in games. Natural user interfaces are represented by contributions about impossible spaces in virtual reality, immersive game technologies for innovative education and multimodal environments for psychomotor skill training, of which the latter two also represent contributions of applying games technologies beyond entertainment.

Accepted articles of the scientific track are:

- Rafael Epplée and Eike Langbehn. "Overlapping Architecture: Implementation of Impossible Spaces in Virtual Reality Games".

- Richard Davies, Nathan Dewell and Carlo Harvey. “A framework for interactive, autonomous and semantic dialogue generation in games”.
- Khaleel Asyraaf Mat Sanusi and Roland Klemke. “Immersive Multimodal Environments for Psychomotor Skills Training”.
- Samuel Truman and Sebastian von Mammen. “Interactive Self-Assembling Agent Ensembles”.
- Roland Klemke, Khaleel Asyraaf Mat Sanusi and Melina Rose. “Immersive Game Technologies for Innovative Education - A Method for Experimental Interdisciplinary Technology Transfer”.

Industry Track

The industry track complemented the scientific track with state-of-the-art presentations from innovative industrial practice. Two presenters from the AI research at Electronics Arts exemplified technological trends in the AAA industry: Han Liu (AI Scientist, Electronic Arts) and Sebastian Starke (AI Scientist, Electronic Arts).

Acknowledgments

The organizers wish to thank all those who contributed to the 1^{st} Games Technology Summit: The researchers who contributed papers, the reviewers who generously offered their time and expertise, the industry representatives offering insightful talks, the organisers of the Clash of Realities conference for their support and the participants of the summit. All of you contributed to the successful start of this series of events.

Cologne and Würzburg, September 30, 2021
Roland Klemke, Martin Lorber, Sebastian von Mammen

Program Committee Members

Sebastian von Mammen
Julius-Maximilians University, Würzburg

Wolfgang Höhl
Technical University of Munich

Roland Klemke
Open University of the Netherlands

Benedikt Hensen
RWTH Aachen University

Roberto Dillon
James Cook University, Singapore

Per Backlund
University of Skövde

Ian Williams
Birmingham City University

Heide Lukosch
University of Canterbury

Mikhail Fominykh
Norwegian University of Science and Technology

Jean-Luc Lugrin
Julius-Maximilians University, Würzburg

Florian Niebling
Julius-Maximilians University, Würzburg

Daniel Roth
Technical University of Munich

Ralf Klamma
RWTH Aachen University

George Papagiannakis
University of Crete

Pejman Sajjadi
Pennsylvania State University

Fotis Liarokapis

CYENS - Centre of Excellence

Sebastian Oberdörfer
Julius-Maximilians University, Würzburg

Michael Kickmeier-Rust
Graz University of Technology

Riccardo Berta
University of Genoa

Contents

Immersive Game Technologies for Innovative Education - A Method for Experimental Interdisciplinary Technology Transfer

Roland Klemke[1,2][0000−0002−9268−3229], Khaleel Asyraaf Mat Sanusi[1][0000−0001−6766−4416], and Melina Rose[1]

[1] Cologne Game Lab, TH Köln, Cologne, Germany rk@colognegamelab.de
http://www.colognegamelab.de/
[2] Open University of the Netherlands, Heerlen, The Netherlands http://www.ou.nl/

Abstract. Immersive, sensor-enabled technologies such as augmented and virtual reality expand the way human beings interact with computers significantly. While these technologies are widely explored in entertainment games, they also offer possibilities for educational use. However, their uptake in education is so far very limited. Within the ImTech4Ed project, we aim at systematically exploring the power of interdisciplinary, international hackathons as a novel method to create immersive educational game prototypes and as a means to transfer these innovative technical prototypes into educational use. To achieve this, we bring together game design and development, where immersive and interactive solutions are designed and developed; computer science, where the technological foundations for immersive technologies and for scalable architectures for these are created; and teacher education, where future teachers are educated. This article reports on the concept and design of these hackathons.

Keywords: Immersive Technologies · Educational Games · Hackathons.

1 Introduction

According to a recent EU report 43% of European Citizens are lacking critical digital competencies while 90% of the professions in near future will require digital competencies. Therefore, an urgent need for the 71 millions of EU students and 5.7 millions of EU Teachers to increase their digital skills via better education, arises [5]. However, innovation processes for technology enhanced learning are not yet well-prepared for disruptive innovation and require novel approaches of development and technology-transfer [9].

A huge obstacle in the uptake of advanced technological solutions (such as serious games, augmented and virtual reality) in schools on primary and secondary level, is that generally, teacher education does not contain the concepts of conceptualising, designing, and applying such solutions to the extent required (though there are exceptions such as [4]).

At the same time, the increasingly available study programmes for game designers and game developers only begin to take educational games and advanced educational technologies into account [14]. Currently, those programmes lack the theoretical underpinning of pedagogic and didactic theories to contribute to the development of educational technologies more substantially. Likewise, computer science education mostly focuses on the technical aspects of solutions with too little attention to design aspects and pedagogical underpinnings. However, subdomains in computer science education such as Human-Computer Interaction or Digital Learning may focus more on these aspects to ensure a better learning outcome.

We believe, that technology-innovation procedures and interdisciplinary ways of designing and developing new systems as explored in novel fields such as game design and games-technology research can be beneficial to the development of innovative, immersive educational technologies. Consequently, here we report on the progress achieved in the ImTech4Ed project (Immersive Technologies for Education), where we aim at exploring such novel ways of technology innovation for education based on experiences in utilising interdisciplinary methods of game design and development.

2 Related Work

Research suggests that teachers have difficulties developing comfort with technology [3] while many teachers are pessimistic and skeptical with immersive technologies [17]. Even future teachers (e.g. pre-service teachers) recognize that immersive technologies are highly motivating, engaging, and able to support students better, still have concerns about their use and implementation. Many of them feel inexperienced in using such technologies [6] or find the available tools difficult, especially at the beginning [7]. In addition, teachers may also need to design, model and program immersive activities, however, there is little support in creating mixed reality education spaces [8].

Immersive Education is a method that utilizes a simulated environment in which the learners can be completely immersed in the learning process/task and moving beyond the mere use of virtual worlds to become more embedded into the physical world around us. Literature on the convergence of the technical, pedagogical and cognitive components and interactions within immersive environments is scarce, and past attempts fall short of fully describing the affordances of augmented and mixed reality, though [1] provides a first approach of mapping learning activities to specific game mechanics, [19] provides a more general approach of using game mechanics for encoding and presenting knowledge in a serious game, and [21] discusses the advantages of learning in an embodied immersive virtual environment. ImTech4Ed builds on these and provides a Pedagogical Framework for building Immersive Environments filling in the current blank. Finally, the importance of augmented reality (AR) and virtual reality (VR) games is high. AR and VR games elicit players' cognitive and affective responses in different ways, where technological, pedagogical and evidence-based findings for AR and VR are discussed in [2, 10, 24]. Collaborative games increase cognitive involvement in solving problems, which leads to enjoyment, pleasure and confidence among users [23].

Recent approaches in the relatively new interdisciplinary game design educational programmes for bachelor and master students successfully demonstrate the value of interdisciplinary education and problem-based learning [14] for cross disciplinary collaboration of programmers, designers, and artists. ImTech4Ed builds on such insights and aims at taking this approach one step further by connecting to educational science and computer science and extending to international and cross institutional scope. At the core of this project is the interdisciplinary collaboration of students from connected disciplines aiming at lowering barriers for interdisciplinary thinking and understanding. In line with our approach, [25] explore the technologies that can be used in curricula to make education "smarter" and more adaptive in order to better meet the needs of today's learners. The main emphasis is placed on the theory and best practices of incorporating emerging technologies into curricula so as to educate learners in the 21st century. Adding to these findings, ImTech4Ed aims at examining, which pedagogical aspects are most suitable for integrating emerging technologies. Consequently, it will potentially help educators and stakeholders design and implement curricula that effectively prepare learners for the challenges of tomorrow.

3 Methodological Approach

Our approach builds upon interdisciplinary and international cooperation among different connected higher education institutions to create quick, student-driven, creative hackathons as initiator for immersive educational technologies to be prototyped and further developed towards educational assets. It is connected to the principles of co-creation [11], which brings together various stakeholder groups in innovation processes as well as participatory design [22], which allows for an iterative, prototyping-based development process involving designers and stakeholders. Our approach goes beyond these by not only connecting designers/developers to stakeholders but at the same time connecting multiple educational disciplines and adding the explicit goal of educating future stakeholders by preparing them for innovation.

Four pillars represent the fundamental of our approach, which aim at defining a stable method towards the reproducible, creative, and innovative development of immersive prototypes.

1. *Methodological Guidelines* provide the technological and didactical context for the development of prototypes.
2. The *Authoring Technology Framework* aim at bridging the gap between non-technical didactic staff and technical developers.
3. *Experimental Hackathons* are the core interdisciplinary, experimental and explorative development activity towards the fast creation of innovative immersive prototypes.

4. *Application and Evaluation* provide a quality approval and selection mechanism to ensure applicability and educational potential of immersive prototypes.

With these pillars, we aim at addressing the identified gaps: teachers need to be better informed about the use of immersive technology (methodological guidelines); technologies need to be suited to the technological competencies of teachers (authoring technology framework); and interdisciplinarity and practical application need to be explored in education (experimental hackathons). With the application and evaluation step, evidence will be generated about the feasibility of this approach. Short project cycles aligned to human-centered design processes ensure the possibility to quickly react to feedback; the interdisciplinary collaboration ensures that relevant perspectives are considered.

3.1 Methodological Guidelines

The methodological guidelines are developed based on best practice experiences and a literature study. These guidelines aim at providing practical support for the use of immersive technologies for learning. They constitute an inquiry-based pedagogical approach for university students and secondary school in-service teachers, which takes technological and didactic constraints as well as diversity and accessibility issues into account. The guidelines provide a framework for developers and educators, in order to create immersive game prototypes making appropriate use of augmented and virtual reality and other immersive technologies.

In particular, the guidelines combine traditional instructional activities with AR/VR experiences within digital games. The methodological guidelines outline the pedagogical and didactic approach to support game-based, immersive education.

3.2 Authoring Technology Framework

Innovative immersive technologies require advanced authoring and development environments to produce high quality outputs. However, most of these environments and tools target specialized professional developers and users, limiting their applicability in educational non-technical contexts. At the same time, tools applicable by non-technical users often miss the flexibility and complexity required for innovative technological solutions.

The authoring technology framework bridges this gap by mapping authoring tools and development environments to process steps and requirements. The authoring technology framework thus connects the methodological guidelines, which represent the didactic view on immersive solutions to the technical tools available to implement these. It is therefore based on a technology survey of available tools and platforms and maps these to the methodological and didactic aspects of implementing immersive solutions.

3.3 Experimental Hackathons

"A hackathon is a highly engaging, continuous event in which people in small groups produce working software prototypes in a limited amount of time" [15]. Hackathons are established in many interdisciplinary domains as a means to quickly develop experimental, innovative prototypes [12]. Recently, hackathons have also been explored for their educational potential [16, 18].

Within the ImTech4Ed project, we aim at combining both aspects of the hackathons: their potential for technological innovation as well as their interdisciplinary educational potential. To do so, we organise the hackathons as multi-location, interdisciplinary events, which bring together educators and students from game design and development, computer science, and educational sciences. Interdisciplinary teams compiled of students from the different disciplines collaborate within these hackathons intensively over a period of four days.

Immersive prototypes are the result of fast paced short cycled development actions with small teams working on very focused small-scale prototypes solving a concrete problem:

- Fast-paced development cycles lead to quick solutions and fast feedback cycles allowing us to quickly react to feedback without risking long development periods to be wasted.
- Developing small-scale immersive prototypes allow to experimentally explore a range of different educational and technological aspects, rather than focusing on very few. This gives also the desired flexibility to choose from a selection of prototypes those that seem promising for further development into educational practice.
- While prototypes developed in this fast-paced way cannot be expected to be at production level quality immediately, the approach to work towards several of these prototypes in parallel fosters creativity, interdisciplinary collaboration, and an open-minded development approach.

3.4 Application and Evaluation

The application and evaluation aspect of the outcomes of the hackathons is done for two purposes: (1) to evaluate prototypes for their potential contribution to education, and (2) to inform members of the target group (i.e. teachers and pre-service teachers), who get involved into the evaluation procedure about the innovation that can be achieved with these prototypes and to educate them on possible use cases.

Consequently, the evaluation represents an important step to ensure applicability, usability, potential, and quality of the immersive prototypes and concepts. The prototypes will be tested and evaluated by various means, comprising their conceptual and technical evaluation by technological experts, their domain-related assessment by educational experts as well as their application in educational contexts by educators and learners.

Prototypes with a positive evaluation are selected for further development in subsequent project phases to increase functional completeness, stability and scalability. Furthermore, since (pre-service) teachers are involved into the evaluation procedure, it also supports the transfer of knowledge, experience, and technology from the development teams into the educational domain by involving educators and learners in the evaluation process.

4 Expected Results, Limitations, and Conclusion

4.1 Expected Results

ImTech4Ed aims at delivering its methodological guidelines together with a set of immersive educational prototypes evaluated in educational practice. The direct impact of ImTech4Ed is expected to be on participating students, pupils, teachers, educators, and researchers in broadening their view and understanding of interdisciplinary approaches and collaborative international work towards the creation of immersive educational technologies. However, ImTech4Ed aims at impacting innovation processes in the development and creation of innovative educational technologies with this multidisciplinary methodology.

With the integration of educational processes into the design and development methodology of ImTech4Ed we expect to be able to address issues of technology acceptance early on in the education of future educators. Likewise, interdisciplinary thinking increases media competence of future educators by being involved in conceptual and technological development of educational prototypes. As stated in the introduction, the lack of technological aspects in teacher education is among the obstacles to the uptake of these innovative technologies in practical education. We expect, that interdisciplinary approaches such as envisioned by ImTech4Ed help to lower these barriers.

4.2 Limitations

The ImTech4Ed project is still in an early development phase. Currently, the conceptual framework as outlined throughout section 3 is designed and planned. The first ImTech4Ed hackathon is planned and will be performed in late Spring 2021. However, due to the COVID-19 situation, the project consortium had to deviate from original plans especially with respect to the organisation of hackathons, which caused delays and changed constellations. We expect that hackathons performed in an online-only manner might negatively impact the intensiveness of team collaborations.

4.3 Conclusions

While still in early phases, the ImTech4Ed project already provides an innovative methodology towards the creative and innovative creation of immersive educational prototypes and solutions that will potentially help teachers, to increase the use of immersive technologies in education. With this methodology, ImTech4Ed aims at bridging the gap between advanced innovations in immersive game technologies and their slow uptake in educational practice.

The interdisciplinary, iterative approach based on methodological guidelines, authoring technology framework, experimental hackathons, and application and evaluation furthermore aims at being reproducible and reusable across various educational contexts. The long-term benefit is expected to go beyond the direct outputs: interdisciplinary thinking towards the envisioning, design, and creation of immersive educational technologies aims at improving the way these technologies are created and brought into educational practice in a sustainable way.

Acknowledgements The ImTech4Ed project is funded under grant agreement number 2020-1-DE01-KA203-005679 as a strategic partnership for higher education.

References

1. Arnab, S., Lim, T., Carvalho, M. B., Bellotti, F., de Freitas, S., Louchart, S., Suttie, N., Berta, R., and De Gloria, A. Mapping learning and game mechanics for serious games analysis. British Journal of Educational Technology, 46(2):391–411, 2015. doi: 10.1111/bjet.12113.
2. Bacca Acosta, J. L., Baldiris Navarro, S. M., Fabregat Gesa, R., Graf, S., & Kinshuk (2014). Augmented reality trends in education: a systematic review of research and applications. Journal of Educational Technology and Society, 2014, vol. 17, núm. 4, p. 133-149.
3. Blankenship, R., & Kim, D. (2012). Revealing authentic teacher professional development using situated learning in virtual environments as a teaching tool. Int'l Forum of Teaching Studies, 8(1), 36-53.
4. Bucher, K. and Grafe, S. (2018). Designing augmented and virtual reality applications with pre-service teachers. In Proceedings of the 10th International Conference on Virtual Worlds and Games for Serious Applications (VS Games '18) (Würzburg, Germany)
5. Chetty, K., Qigui, L., Gcora, N., Josie, J., Wenwei, L., & Fang, C. (2018). Bridging the digital divide: measuring digital literacy. Economics: The Open-Access, Open-Assessment E-Journal, 12(2018-23), 1-20.
6. G. Cooper, H. Park, Z. Nasr, L. P. Thong & R. Johnson (2019) Using virtual reality in the classroom: preservice teachers' perceptions of its use as a teaching and learning tool, Educational Media Int'l, 56:1, 1-13, DOI: 10.1080/09523987.2019.1583461
7. Delello, J.A. Insights from pre-service teachers using science-based augmented reality. J. Comput. Educ. 1, 295–311 (2014). https://doi.org/10.1007/s40692-014-0021-y
8. Elliott, J. B., Gardner, M., & Alrashidi, M. (2012). Towards a framework for the design of mixed reality immersive education spaces. Proceedings of the 2nd Eur. immersive initiative summit, Paris, France, 63-76.
9. Flavin, M., & Quintero, V. (2020). An international study of technology enhanced learning-related strategies from the perspective of disruptive innovation. Interactive Technology and Smart Education.
10. Freina, L., & Ott, M. (2015, April). A literature review on immersive virtual reality in education: state of the art and perspectives. In The international scientific conference elearning and software for education (Vol. 1, No. 133, pp. 10-1007).

11. Gemser, G., & Perks, H. (2015). Co-creation with customers: An evolving innovation research field. Journal of Product Innovation Management, 32(5), 660-665.
12. Juraschek, M., Büth, L., Martin, N., Pulst, S., Thiede, S., & Herrmann, C. (2020). Event-based education and innovation in Learning Factories–concept and evaluation from Hackathon to GameJam. Procedia Manufacturing, 45, 43-48.
13. Kelle, S., Klemke, R., Gruber, M., & Specht, M. (2011, June). Standardization of game-based learning design. In International conference on computational science and its applications (pp. 518-532). Springer, Berlin, Heidelberg.
14. Klemke, R. & Hettlich, M. (4 Mar 2019). Programmieren, was andere spielen sollen: Game Informatics in Forschung, Theorie und Praxis (Translated title of the contribution: Programming, what others play: Game Informatics in Research, Theory and Application) In: Games studieren – was, wie, wo?: Staatliche Studienangebote im Bereich digitaler Spiele. Bartholdy, B., Breitlauch, L., Czauderna, A. & Freyermuth, G. S. (eds.). Bielefeld: Transcript Verlag, p. 319-324 6 p.
15. Komssi, M., Pichlis, D., Raatikainen, M., Kindström, K., & Järvinen, J. (2014). What are hackathons for?. IEEE Software, 32(5), 60-67.
16. Lara, M., & Lockwood, K. (2016). Hackathons as community-based learning: a case study. TechTrends, 60(5), 486-495.
17. McNair, C. L., & Green, M. (2016). Preservice teachers' perceptions of augmented reality. Literacy Summit Yearbook, 74-81.
18. Nandi, A., & Mandernach, M. (2016, February). Hackathons as an informal learning platform. In Proceedings of the 47th ACM Technical Symposium on Computing Science Education (pp. 346-351).
19. Oberdörfer, S. and Latoschik, M. E. (2019). Predicting learning effects of computer games using the gamified knowledge encoding model. Entertainment Computing 32. doi:10.1016/j.entcom.2019.100315
20. Rosell, B., Kumar, S., & Shepherd, J. (2014, May). Unleashing innovation through internal hackathons. In 2014 IEEE Innovations in Technology Conference (pp. 1-8). IEEE.
21. Slater, M. (2017). Implicit learning through embodiment in immersive virtual reality. In Virtual, Augmented, and Mixed Realities in Education, eds. D. Liu, C. J. Dede, R. Huang, and J. Richards (Singapoore: Springer), Smart Computing and Intelligence
22. Spinuzzi, C. (2005). The methodology of participatory design. Technical communication, 52(2), 163-174.
23. Suh, A., & Prophet, J. (2018). The state of immersive technology research: A literature analysis. Comp. in Human Behavior, 86, 77-90.
24. Wu, B., Yu, X., & Gu, X. (2020). Effectiveness of immersive virtual reality using head-mounted displays on learning performance: A meta-analysis. British Journal of Educational Technology, 51(6), 1991-2005.
25. Yu S. , Ally M. and Tsinakos A. (2020). Emerging Technologies and Pedagogies in the Curriculum Yu et al (Eds) Book, ISBN 978-981-15-0617-8, Springer, 2020 Retrieve from https://www.springer.com/gp/book/9789811506178

Immersive Multimodal Environments for Psychomotor Skills Training

Khaleel Asyraaf Mat Sanusi[1] [0000−0001−6766−4416] and Roland Klemke[1,2] [0000−0002−9268−3229]

[1] Cologne Game Lab, TH Köln, Cologne, Germany
[2] Open University of the Netherlands, Heerlen, The Netherlands

Abstract. Modern immersive multimodal technologies enable the learners to completely get immersed in various learning situations in a way that feels like experiencing an authentic learning environment. These environments also allow the collection of multimodal data, which can be used with artificial intelligence to further improve the immersion and learning outcomes. The use of artificial intelligence has been widely explored for the interpretation of multimodal data collected from multiple sensors, thus giving insights to support learners' performance by providing personalised feedback. In this paper, we present a conceptual approach for creating immersive learning environments, integrated with multi-sensor setup to help learners improve their psychomotor skills in a remote setting.

Keywords: immersive learning technologies · multimodal learning · sensor devices · artificial intelligence · psychomotor training

1 Introduction

The global pandemic event of Covid-19 has shown that many learning and teaching activities can be performed in a remote manner, however, this is rarely the case for psychomotor skills development as they require hands-on practice. Psychomotor skills need to be physically executed, in most cases, repetitively to the extent that the muscle memory is trained, which automates them [6]. The learning model for training psychomotor training needs to be well structured, consisting of well-defined instructions and feedback that can be provided during the training. This ensures the development of such skills in a correct manner while ensuring consistent progress during the iterative practice sessions. As such, constant feedback from the teacher is essential for the learner to avoid developing improper techniques during training and ensuring the desired goal can be achieved in a shorter duration. However, a remote setting for psychomotor training makes the learning process tedious, ineffective, and inefficient, obstructing the beginner's progress. Therefore, the remote setting for psychomotor skill training affects both the learners and the teachers alike.

Nonetheless, educational researchers in the field of technology enhanced learning and artificial intelligence (AI) are continuously working to enhance psychomotor training by embedding multiple sensor technologies and machine learning models to facilitate multimodal learning experiences [2, 5, 14]. Also, technologies such as virtual reality (VR) and augmented reality (AR) are increasingly used to provide learners a more immersive feeling in an authentic setting [3, 10, 13]. The use of these combined technologies introduces new technological affordances that can be leveraged in the field of psychomotor learning, especially in a remote manner. However, experts or teachers may still be skeptical of these technologies, especially AI, since they are relatively new and not widely adopted.

Therefore, in this paper, we present a conceptual approach for creating immersive and interactive multimodal training environments using sensor and AI technologies to deliver instructions and feedback to learners in an authentic setting. The development of this prototype will be an early and significant step towards combining such technologies in an authentic setting for collecting multimodal data and giving immediate feedback for learners to improve their psychomotor skills.

2 Related Work

2.1 Sensors in Psychomotor Learning

Sensors are emerging technologies that are increasingly becoming more portable and smarter, enabling efficient methods for the acquisition of performance data, allowing effective monitoring and intervention. Sensors have been explored in the learning domain for their affordances to provide support to the learner. Schneider et al. [16] analysed 82 prototypes found in literature studies based on Bloom's taxonomy of learning domains. They explored how the prototypes can be used to provide formative assessment, especially as a feedback tool. The analysis revealed that sensor-based learning applications can assist learning scenarios in several areas, including the psychomotor domain. This research recommends researchers and educators to consider sensor-based platforms as dependable learning tools to reduce the workload of human teachers or mentors and, therefore, contribute to the solution of many current educational challenges.

In the psychomotor domain, motion sensors such as accelerometers and gyroscopes are predominantly used for the acquisition of motion data to recognize human activities. Pereira dos Santos et al. [15] emphasize the importance of combining motion sensors to obtain higher accuracy in detecting not only simple but complex activities as well. Combining these sensors enables the collection of multimodal data and, therefore, a more accurate representation of the learning process [2]. The collection of multimodal data can be done by using various types of sensors such as wearable sensors, depth cameras, Internet of Things devices, computer logs, etc. Moreover, these sensors are typically synchronized with human activities because one can analyse historical evidence of learning activities and the description of the learning process [11]. Limbu et al. [12] designed a system combining Pen sensors in Microsoft Surface and EMG sensors in Myo armband for learners to practice their calligraphy skills deliberately. It allows the calligraphy teacher to create an expert model, in which later the learners can use to practice while receiving guidance and feedback. Similarly, Schneider et al. [17] also designed a system to support the development of complex non-verbal communication skills for public speaking. The system uses the Microsoft Kinect v2 depth camera sensor that is placed in front of the learner to track the skeletal joints of the learner's body, along with a microphone, while presenting and provide real-time feedback based on common public speaking mistakes such as facial expressions, body posture, voice volume, gestures, and pauses.

To better understand learners' psychomotor performance, AI, more precisely machine learning approaches, has been explored. Educational researchers are increasingly using machine learning models to classify the activities based on the collected multimodal data. For instance, Di Mitri et al. [5] investigated to what extent multimodal data and Neural Networks can be used for learning Cardiopulmonary Resuscitation skills by utilising a multi-sensor system consisting of a Kinect for tracking body position and a Myo armband for collecting electromyogram information. Another example, Mat Sanusi [14] applied the same framework as the previous author but in a table tennis domain consisting of more dynamic movements and complex techniques. The author experimented with smartphone sensors (accelerometer and gyroscope) for motion data and a Kinect to detect forehand strokes during training. Both study results show a high classification rate of the activities when combining the sensors, emphasizing the importance of the multimodal approach in classifying complex psychomotor activities.

2.2 Immersive Learning Technologies

Immersive learning technologies, such as Augmented reality (AR) and Virtual Reality (VR), are poised to be the next educational technology medium for effective education, even in remote scenarios. Immersive learning technologies place individuals into an interactive training or learning environment, either virtually or physically, to replicate/enrich the authentic learning context of a specific skill. This provides a ubiquitous opportunity for experiential learning at any time and place. These immersive technologies can further accelerate learning experiences by providing just-in-time instructions and feedback. For instance, Chan et al. [3] developed a dance training system based on VR integrated with motion capture technology. The system projects a virtual teacher on the wall screen and demonstrates specific movements, allowing the learners to imitate. When mistakes are detected by the motion capture system, the virtual teacher provides visual immediate feedback by suggesting improvements on particular limbs at particular moves. In another example, Kyan et al. [10] implemented a VR ballet dance training system that tracks the learner's skeletal joints using the Kinect depth camera sensor. This system applies a similar concept to the "magic mirror" approach in which the virtual character projected on the wall screen moves accordingly to the learners, enabling them to reflect on their performance in real-time.

However, the use of visual feedback is commonly insufficient to not only the dance use case but in most psychomotor activities due to their complexity.

Learning psychomotor skills typically involves more complicated techniques, and learners may not be able to focus on the virtual teacher all the time. Thus, an ideal solution would be the combination of multiple modalities (e.g., visual, audio, haptic), as advocated by the multimodal approach within the immersive technologies. This can be seen in the work of Jadhav et al. [9] in which they designed a wearable soft robotic glove that provides haptic feedback in virtual environments. In their work, learning piano was used as an application case. As a result, the haptic feedback compliments the visual feedback provided from the virtual reality headset and audio feedback generated by the headphone, emphasizing the need for the multimodal approach in psychomotor learning.

While multimodal technologies provide affordances for immersion, designing a learning experience that is immersive mostly depends on the design of the learning activity. For example, Herrington et al. [8] stressed that the learning environment and task create the conditions for “True” immersion. Their study found that the task itself is the key element of immersion and engagement in cognitive (synonym to psychomotor) learning, which plays an important role in motor skills development. Similarly, it can be argued that problems presented to the student for learning should be real or that simulation should have ultra-realistic physical similarity to an actual context. Therefore, it can also be argued that the instructions and feedback provided by the learning environment should be realistic for the beginners to perform the tasks in a correct manner; for example, actual human teachers like/personalised feedback to create immersive learning experiences. Such instructions and feedback that are able to match a real human expert can be provided by using AI. In our work, we aim to create a “truly” immersive and interactive training environment with the help of immersive multimodal learning technologies to train psychomotor skills by providing human-teacher-like instructions and feedback.

3 Research Aim

Our main objective for this research is to create an immersive and interactive training environment using immersive multimodal learning technologies. This learning environment will integrate the multi-sensor setup, enabling the collection of multimodal data and, thereof, machine learning analysis to help learners improve their psychomotor skills. Therefore, the following research questions have been constructed:

RQ1 **How can we create an immersive and information rich (remote/self-learning) training environment for psychomotor skills that delivers effective instructions and meaningful feedback to the learner?**

RQ2 **Can the new immersive training environment improve existing learning methods in remote or self-directed education for psychomotor skills?**

RQ3 **How can human teachers and experts contribute to and benefit from the training environment?**

4 Research Methodologies

While immersive technologies are becoming more prominent and progressively used in the context of psychomotor training, exploiting them to create a virtual training environment has no assurance in supporting learners effectively. The intended outcome of the project is a prototypical solution for psychomotor learning that is immersive, along with a systematic process of designing that solution. Therefore, it is important for this research to follow a methodological approach for designing, developing, and testing such a prototype. Hence, we conduct our research in the Design-based Research (DBR) approach, which is a common iterative methodological approach for prototypical solutions [1] (see Fig. 1).

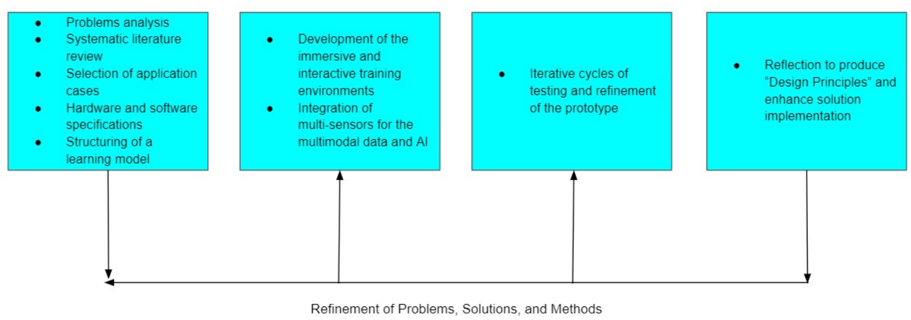


Fig. 1. The DBR approach in the context of this paper [1].

The DBR methodological approach is divided into four phases. For the first phase, it is ideal to start by analysing the problems in the domain of immersive multimodal environments that need addressing. A systematic literature review will then be followed to identify and synthesize related work publications. The next iteration will be the selection of application cases, in which each case includes several psychomotor tasks. Further, we specify the hardware and software components that are appropriate for the selected application cases. In the last iteration of this phase, a learning model, which comprises instructions and feedback, should be constructed.

The next phase should be the implementation of the training environment. A structured learning model from the previous phase should be included to create immersion for learners. Next, to further improve the immersion and learning outcomes, the training environment will be integrated with multi-sensor setup for the collection of multimodal data and AI for the machine learning analysis to provide feedback to learners based on their performance.

Subsequently, in the next phase, a user test involving the teachers/experts should be conducted to reveal essential aspects of how the prototype can be improved. Additionally, questionnaires and surveys are helpful to provide a general idea of how users perceive the interaction between the prototype. The refinement of the prototype should then be followed to ensure that the prototype is ready to be tested with the learners.

Lastly, it is imperative to keep detailed records during the design research process concerning how the design outcomes (e.g. principles) have worked or have not worked, how the innovation has been improved, and what are the changes have been made. Through this documentation, it can be helpful for other researchers and designers who are interested in those findings, can examine them in relation to their context and needs.

5 Conclusion

In this paper, we presented a conceptual approach for creating immersive and interactive multimodal training environments using sensors and AI to deliver meaningful instructions and effective feedback to learners in an authentic setting. The development of this prototype will be an essential step towards combining such technologies in an authentic setting for collecting multimodal data and providing immediate feedback for learners to improve their psychomotor skills remotely, thereby further enhancing the immersive learning experiences.

References

1. Anderson, T., Shattuck, J. (2012). Design-based research: A decade of progress in education research?. Educational researcher, 41(1), 16-25. https://doi.org/https://doi.org/10.3102/0013189X11428813
2. Blikstein, P., Worsley, M. (2016). Multimodal learning analytics and education data mining: Using computational technologies to measure complex learning tasks. Journal of Learning Analytics, 3(2), 220-238. https://doi.org/https://doi.org/10.18608/jla.2016.32.11
3. Chan, J. C., Leung, H., Tang, J. K., Komura, T. (2010). A virtual reality dance training system using motion capture technology. IEEE transactions on learning technologies, 4(2), 187-195. https://doi.org/10.1109/TLT.2010.27
4. Cratty, B. J. and Noble, . Clyde Everett (2016, April 1). Psychomotor learning. Encyclopedia Britannica. https://www.britannica.com/science/psychomotor-learning. Last accessed 10 March 2021
5. Di Mitri, D., Schneider, J., Trebing, K., Sopka, S., Specht, M., Drachsler, H. (2020, July). Real-Time Multimodal Feedback with the CPR Tutor. In International Conference on Artificial Intelligence in Education (pp. 141-152). Springer, Cham. https://doi.org/https://doi.org/10.1007/978-3-030-52237-$7_1$2
6. Feder, K. P., Majnemer, A. (2007). Handwriting development, competency, and intervention. Developmental Medicine Child Neurology, 49(4), 312-317. https://doi.org/https://doi.org/10.1111/j.1469-8749.2007.00312.x

7. Hattie, J., Timperley, H. (2007). The power of feedback. Review of educational research, 77(1), 81-112. https://doi.org/https://doi.org/10.3102/003465430298487
8. Herrington, J., Reeves, T. C., Oliver, R. (2007). Immersive learning technologies: Realism and online authentic learning. Journal of Computing in Higher Education, 19(1), 80-99.
9. Jadhav, S., Kannanda, V., Kang, B., Tolley, M. T., Schulze, J. P. (2017). Soft robotic glove for kinesthetic haptic feedback in virtual reality environments. Electronic Imaging, 2017(3), 19-24. https://doi.org/https://doi.org/10.2352/ISSN.2470-1173.2017.3.ERVR-102
10. Kyan, M., Sun, G., Li, H., Zhong, L., Muneesawang, P., Dong, N., Elder, B., Guan, L. (2015). An approach to ballet dance training through ms kinect and visualization in a cave virtual reality environment. ACM Transactions on Intelligent Systems and Technology (TIST), 6(2), 1-37. https://doi.org/https://doi.org/10.1145/2735951
11. Lahat, D., Adali, T., Jutten, C. (2015). Multimodal data fusion: an overview of methods, challenges, and prospects. Proceedings of the IEEE, 103(9), 1449-1477. https://doi.org/10.1109/JPROC.2015.2460697
12. Limbu, B. H., Jarodzka, H., Klemke, R., Specht, M. (2019). Can You Ink While You Blink? Assessing Mental Effort in a Sensor-Based Calligraphy Trainer. Sensors, 19(14), 3244. https://doi.org/https://doi.org/10.3390/s19143244
13. Limbu, B., Vovk, A., Jarodzka, H., Klemke, R., Wild, F., Specht, M. (2019, September). WEKIT. one: a sensor-based augmented reality system for experience capture and re-enactment. In European Conference on Technology Enhanced Learning (pp. 158-171). Springer, Cham. https://doi.org/https://doi.org/10.1007/978-3-030-29736-$7_1$2
14. Mat Sanusi, K. A. (2020). Stroke in Pong like Ma Long: Forehand Strokes Detection for Automatic Feedback Generation in Table Tennis Performance based on Multimodal Data and Neural Networks. Master Thesis. University of Saarland, Germany.
15. Dias Pereira dos Santos, A., Yacef, K., Martinez-Maldonado, R. (2017, July). Let's dance: how to build a user model for dance students using wearable technology. In Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization (pp. 183-191). https://doi.org/https://doi.org/10.1145/3079628.3079673
16. Schneider, J., Börner, D., Van Rosmalen, P. and Specht, M., (2015). Augmenting the senses: a review on sensor-based learning support. Sensors, 15(2), pp.4097-4133. https://doi.org/https://doi.org/10.3390/s150204097
17. Schneider, J., Börner, D., Van Rosmalen, P., Specht, M. (2016). Can you help me with my pitch? Studying a tool for real-time automated feedback. IEEE Transactions on Learning Technologies, 9(4), 318-327. https://doi.org/10.1109/TLT.2016.2627043

A framework for interactive, autonomous and semantic dialogue generation in games

Richard Davies[1] [0000−0003−0656−9129], Nathan Dewell[1] [0000−0002−6844−7939], and Carlo Harvey[1] [0000−0002−4809−1592]

[1]DMTLab, Birmingham City University, Millennium Point, Curzon Street, Birmingham, United Kingdom
richard.davies@bcu.ac.uk
https://www.bcu.ac.uk/computing/research/digital-media-technology

Abstract. Immersive virtual environments provide users with the opportunity to escape from the real world, but scripted dialogues can disrupt the presence within the world the user is trying to escape within. Both Non-Playable Character (NPC) to Player and NPC to NPC dialogue can be non-natural and the reliance on responding with pre-defined dialogue does not always meet the players emotional expectations or provide responses appropriate to the given context or world states. This paper investigates the application of Artificial Intelligence (AI) and Natural Language Processing to generate dynamic human-like responses within a themed virtual world. Each thematic has been analysed against human-generated responses for the same seed and demonstrates invariance of rating across a range of model sizes, but shows an effect of theme and the size of the corpus used for fine-tuning the context for the game world.

Keywords: Natural Language Processing · Interactive Authoring System · Semantic Understanding · Artificial Intelligence.

1 Introduction

Explicit and rich stories in virtual environments (VEs) are a product of large volumes of authoring. Traditional authoring methods introduce a large burden to narrative generators and story conveyors to ensure they are maintaining a world state that is both contextual to player interactions and bears semantic association to the virtual world. Many interactions that require some associated response to the player from the virtual space, yield none [8]. Additionally, NPC dialogue is commonly perceived as being predictable or scripted [16]. Whilst it is possible for personality and emotional state to be perceived in games, this is typically done through careful authoring and tracking of the roles played in context [7]. This can be cumbersome and resource intensive to game designers.

Scripted dialogue interactions in VEs are typically used to help alleviate this burden. This is presented to the player as a menu of choices to prompt a response. This affords the player some discrete level of expression along the continuum of responses and is quite flexible. However, the resource cost in delivering bespoke options in a dialogue tree limits this interaction and inhibits the ability for growth to dynamic interactions expected of a player [8].

In order to better bridge the gap between scripted authoring, whether branching or linear, and natural dialogue responses for social agents in virtual worlds it is important to be able to evaluate dialogue responses, moving towards an automatic Turing Test [22], [11]. This paper thus presents a framework for autonomous dialogue responses for social agents under different themes by fine-tuning an existing model and conducts an evaluation of these thematic dialog responses vs. a baseline model that is not fine-tuned, across model sizes of GPT-2 using the ADEM metric. The contributions of this work are as follows:

- A framework for text generation models in narrative authoring for VEs;
- A platform for interfacing with contextual trained models via web requests;
- A procedure for evaluating response quality from a semantic NLP model output against ground truth human-sourced responses.

2 Related Work

Considerable research has been conducted into generating interactive dialogue systems and narrative authoring applications [13], [15], [12]. There exists a common interest in the community in using natural language processing (NLP) techniques to manage and mediate plausible and contextual interactions in VEs. Comparative to the work that has been conducted in managed scripted systems, less research exists in the field of autonomous natural language interfaces [8].

Generative pre-training has been used to empower natural language generation across a range of tasks [14]. This approach, referred to as GPT-2, uses abundant unlabelled text corpora to build a language model and then uses a transfer learning approach to fine-tune this model to a particular context. This has recently been extended to GPT-3 [2], where 175B parameters are used in the language model and is shown to be able to generate text that human evaluators have difficulty in distinguishing from human written. There are ethical concerns surrounding this improvement in the state-of-the-art.

There exists a need to evaluate the efficacy of generative text for a particular context. Erkel *et. al* performed a study utilising the Bystander Turing Test paradigm to establish if subjects rated dialogue in tutoring transcripts where generated by a computer or by a human [4]. Results indicated that subjects were incapable of correctly judging by what means the text was generated. Adversarial training has been investigated in the context of evaluating open-domain dialogue generation [9]. In this work, Li *et. al.* train a system to generate utterances that are indistinguishable from human-generated sequences using reinforcement learning and both a generator (to create response sequences) with a discriminator (to evaluate the efficacy of the responses). The discriminator is used as a reward in the reinforcement learning system for the generator. Other recent advances have empowered machine generated text evaluation to be performed automatically [11]. Lowe *et. al.* proposed ADEM, to allow objective scores to be created in the evaluation procedure. This model learns to predict human-like scores to input responses, using a dataset created of human response scores. The predictions from this system correlate significantly with human judgements for machine generated text allowing for its use in objective assessment processes.

3 Methodology

This Section introduces the overarching methodology of the presented framework as well as dataset generation, training procedures and evaluation methods. The framework methodology is functionally shown in Figure 1.

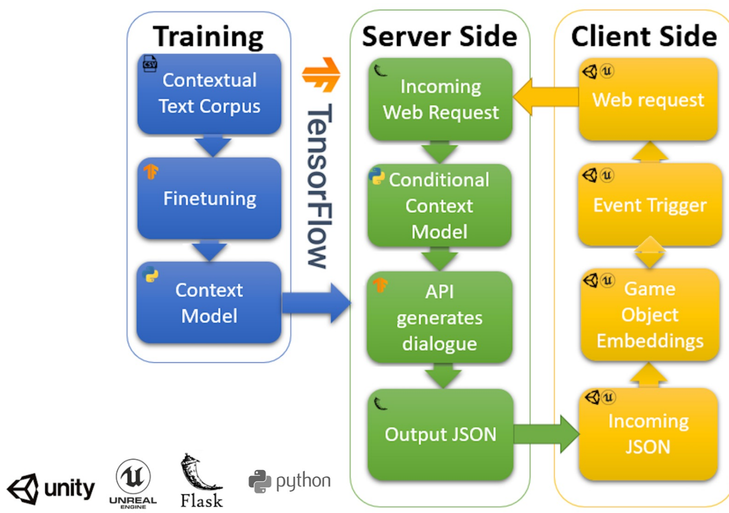


Fig. 1: Demonstrating the processes and pipelines of interaction between the training of the models, the server side processing and request handling with the client side interaction. We indicate the software and tools used in this pipeline, it should be noted that these are interchangeable with other options.

3.1 Web API

It is possible to run a system similar to this on a local machine, however the requirement of GPU compute required impacts upon the game play performance of the end user. As a result this system has been packaged in the form of an Application Programmable Interface (API). A versatile and system agnostic method is developed serving and allowing for interfacing with an API which interconnects with a verity of different endpoint platforms. The API is used to evaluate contextually trained models using a combination of both HTTP Requests and the prevalent JSON format. The API is built in the micro web framework, Flask which in turn allows it to be developed and run within Python. Flask utilises the HTTP Request functionality GET in order to set the parameters of the text generation functionality within the API. Using common web functionality enabled a platform agnostic system that in turn can be used within any game engine that allows HTTP Web Calls to be made within it.

Once the web requests are made, the API loads the required pre-processed and fine tuned model in Tensorflow, sets the parameters sent with the request and processes the request. This generates a string of text that will be returned to Flask in the form of an enumerated array that contains the original request along with the parameters and prefix provided to Tensorflow. Flask then compiles the array into the data interchange format, JSON allowing the requesting application to process the data. Parameters are evaluated as follows `https://server/?speech=str&length=int&truncate=str&style=str`. A copy of the code is available here: [3].

3.2 Dataset Generation

For proof-of-concept we chose to create three datasets (*norse*, *pirates* and *sci-fi*) to finetune generation for 4 scenes. Our fourth scene would use the base model without finetuning, called *modern*. The scene coupled with dataset pairings are presented as follows in the format $\longrightarrow$ [theme: corpus: text-lines: size (kb)]. These are [modern: none: n/a: none], [norse: vikings: 8232: 1071], [sci-fi: altered carbon, lost in space, star trek, the expanse: 21205: 2216], and [pirates: black sails: 8794, 854].

Datasets were harvested to provide contextual dialogue for each game thematic. We used a subtitle hosting service to extract dialogue from relevant media shows [20], this text was cleaned using the ftfy library [21]. This removed generic advertising embeddings, emojis and also standardised punctuation and whitespace. Once cleaned the text was exported to a single column csv where each sequence of tokens is annotated with beginning and end of sentence tokens, <|startoftext|> and <|endoftext|> respectively, empowering the traversal-style approach [17]. This allows for fine-tuning across tasks such as text classification, question answering or textual entailment. The task for this proof of concept is one of semantic NPC dialogue generation.

Finally, in order to make the context generic, it was necessary to remove instances of fictional characters and names and replace with tokens that can be parsed client side to convey the narrative of the world being developed. For example, references to nominals that are franchise related were replaced with npc_n or $place_n$ where n is the current counter of novel instances of the type we are looking for. These special tokens can then be decoded client-side and substituted with NPC names or locations relevant to the world.

3.3 Model Training

Here we present the method for finetuning the model including loss per semantic category with timings. The original Generative Pre-Training paper uses an unsupervised pre-training to produce models with pre-trained weights [14]. The number of parameters for each released model under GPT-2 are 124M: 12, 355M: 24, 774M: 36 and 1558M: 48. This is in stark contrast to the potential of models such as GPT-3 [2]. Reported to be using 175B parameters, a significant step in non-sparse autoregressive language models.

The training process adopts a transfer learning paradigm whereby unsupervised pre-training is conducted on a generic text corpus of tokens $U = \{u_1, ..., u_n\}$, attempting to maximise the likelihood:

$$L_1(U) = \sum_i logP(u_i | u_{i-k}, ..., u_{i-1}; \theta) \quad (1)$$

where k is the size of the context window and P is the conditional probability being modelled by the neural network controlled with parameters θ. This then uses a multi-layer Transformer decoder as per the original implementation [14], [10]. We use pre-trained models from the process in Equation 1 and adapt the parameters for different sized models using a process of supervised fine-tuning. Assuming a contextual and labelled dataset C and each instance of text within C comprises a sequence of input tokens $x^1, ..., x^m$ with a label y. These input tokens $x^1, ..., x^m$ are passed through the pre-trained model which gives the final transformer block's activation h_l^m. This activation is passed into a linear output layer which has a parameter, W_y used to predict the value of the label y. This is shown in Equation 2:

$$P(y|x^1, ..., x^m) = \texttt{softmax}(h_l^m W_y) \quad (2)$$

Following on from the unsupervised process that yields a generic model, to finetune for a purpose it is necessary to maximise for the following objective:

$$L_2(C) = \sum_{(x,y)} logP(y|x^1, ..., x^m) \quad (3)$$

We test fine-tuning the 124M, 355M and 762M pre-trained models using methods provided by the gpt-2-simple interface [23]. The 1558M model did not fit into our hardware memory. We trained the models on different setups including using a NVIDIA Titan V in conjunction with a NVIDIA GTX 1080 in a multi-gpu setup and a singular NVIDIA Titan Xp. For fine-tuning details, we use 500 steps, with a batch size of 1, with a learning rate of 1e-4 and an adam optimiser. These trained, contextual models of varying model parameter sizes, are then uploaded to the server for host evaluation calls from a client. Training and loss times per theme and model are shown in Table 1.

3.4 Game Framework

A 'trade scene' was created in various styles, each depicting one of the four chosen themes as shown in Figure 2. This is a style typical of any given role-playing-game. This approach is motivated by example in the work carried out by [7] to

Table 1: Timings and loss per model size against each semantic text corpus for 500 steps of fine tuning training. μ represents average loss over the 500 steps, t is the time for training in seconds along with a comparisons between the different models based upon the time taken to generate a response.. * required a multi-gpu approach to training due to memory requirements.

	Training Times						Evaluation Time	
	Norse		Scifi		Pirates		All Themes	
Size	μ	t	μ	t	μ	t	Mean	Min-Max: Range
124M	1.06	567.16	1.74	595.17	1.03	576.13	5.4353	4.29-6.93: 2.64
355M	1.46	387.74	1.88	406.46	1.57	399.45	9.9565	7.96-12.12: 4.16
774M	1.27	2253.37	1.74	2256.23	1.28	2276.14	15.5606	12.90-19.84: 6.94

aid in creating believable characters in an immersive world. With the advantage of a trade scene being commonplace in video games, it is also a setting that is agnostic to a specific genre, making it a more than ideal locale to showcase the thematic fluidity of the dialogue generation that has been created.

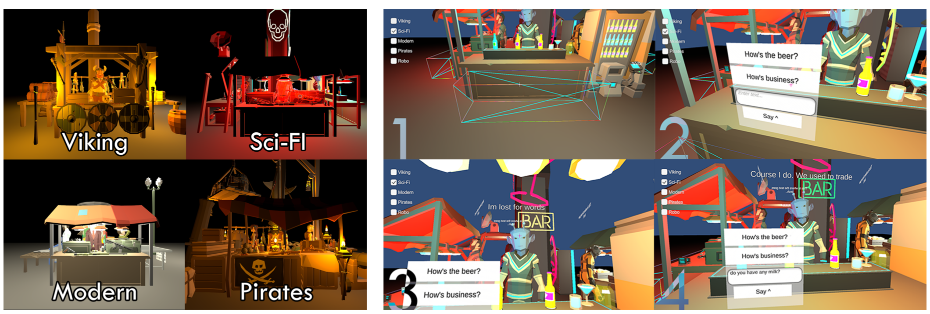


Fig. 2: Left: Illustrating the four types of scene presented in the similar setting of a vending stall. The scenes shown: Norse, Sci-fi, Modern and Pirates. Right: A typical interaction paradigm in the virtual trade scenario for a specific theme (Sci-fi). The storyboard shown, from 1 to 4: walk the player character into the trigger box for interaction with the NPC; a dialogue selection box appears, choose from scripted text or enter your own; a response from the NPC is generated and displayed; if a response cannot be displayed, a generic response is displayed.

Each trading stall is equipped with a trigger. This trigger is used to execute UI interactions, UI interactions can then be used to send a web request to the address where the dialogue system is being hosted. Any form of interaction can be added in between the trigger being used and the web request being sent. This could be in the form of a dialogue tree, an input text field or perhaps even a physical action within the game. This interaction window should be used to seed the dialogue model with a phrase or question, to be sent along with specifying the requested theme data model, as per parameters. It is also possible to instigate a response without a seed pre-text, by defaulting to `<|startoftext|>`. A typical interaction with this system can be seen via storyboard presented in Figure 2.

3.5 Evaluation

To better understand the merits of different model sizes and thematic fine tuning on the performance of the dialogue when evaluated, we use the Automatic Dialogue Evaluation Model (ADEM) [11]. To facilitate this study we perform a 3×4 factorial design study, investigating independent variables (IVs) of *model* $\times$ *theme*. The dependent variable (DV) in this study is the ADEM score, where machine generated responses, $\hat{r}$, are evaluated against a pretext seed, c, in comparison to a human-generated response, r, to the same pretext:

$$\texttt{adem}(c, r, \hat{r}) = (\mathbf{c}^T M \hat{\mathbf{r}} + \mathbf{r}^T N \hat{\mathbf{r}} - \alpha)/\beta \tag{4}$$

where $\mathbf{c}$, $\mathbf{r}$ and $\hat{\mathbf{r}}$ are vector representations of c, r and $\hat{r}$ respectively, transformed by a hierarchical recurrent neural network. $M, N \in \mathbb{R}^n$ are learned matrices which are trained to minimise the squared error between the machine predictions and human scores using L2 regularisation. We motivate this choice despite recent work showing that targeted attacks can systematically exploit weaknesses in the ADEM score [18]. This work showed that word order can confuse the metric as well as other slight modifications such as removing punctuation, simplifying the response and creating generic responses. The machine generated text of GPT-2 formulates a likely probability of the next token in a sequence, creating a smaller likelihood of out of sequence errors. Punctuation is controlled for by preprocessing via *ftfy* and generic responses are accounted for via fine tuning of the models. As such, ADEM is still an appropriate choice for automatic evaluation of the dialogue responses. To perform the analysis we consider 3 pretexts, 14 human-generated responses to each pre-text, 10 machine generated responses, 3 model sizes and 4 themes. This gives us 5040 scores when exploring all permutations. It should be noted for clarity that human-generated responses were generated per theme by asking for performant responses from contributors for each theme.

4 Results

In order to establish the quality of dialogue responses, we adopt the ADEM evaluation model that predicts human-like scores to input responses [11]. This approach is more appropriate to dialogue utterances. This method allows for objective results for automatic dialogue evaluation, given a seed and a truth against the machine generated dialogue. For a number of seeds shown, we generate human responses to these and compare and contrast the automatically generated equivalents using ADEM. This is shown in Table 3. Using the ADEM score to evaluate results in comparison with human generated text allows us to facilitate the objective evaluation of the machine generated responses, this in turn would be apparent if a human were to interact with the platform within a game as we could evaluate multiple outputs from the platform against ADEM scores to provide the most human-like response.

As can be displayed in Table 3, examples have been provided showcasing different responses generated based upon the different models, scenes and seeds input to trigger the API response. Each response is evaluated against 14 different human generated responses to the same seed and theme. Once the response has been calculated against the 14 human responses it takes an average, using an average ADEM score removes potential erroneous out of range results which in turn could result in a poorly rated response not achieving the required level of human-like responses that the user would be expecting.

Table 1 also shows the processing time for each response, with differences between each of the base models that have been trained and built upon. Although for testing purposes the speed of the response does not provide a significant problem, the increase in time to generate caused by the increased model size would reduce overall immersion. This can be overcome by pre-generating responses for deterministic seeds, for example from dialogue trees or scripts and weighting appropriate responses by ADEM score. However, is still a caveat when interaction occurs naturally. With further development and optimisation the time taken to generate a response can be improved, though for this work it was decided to concentrate on evaluating the quality of the responses over the generation speed.

To present a subset of the permutations explored in the evaluation, Table 3 in Section 6 shows for a theme and a model, sample text seeds accompanied by model machine responses and human generated performer responses to this seed alongside the ADEM score. To explore contrasts and to test the IVs against the DVs a two-way univariate ANOVA is conducted against the DV of ADEM. Shown in Table 2 is this analysis, demonstrating significance of *theme* but not of *model* suggesting the contextual fine-tuning performed has an influence on the ADEM score. This motivates an exploration via pairwise comparisons to elucidate inter-theme contrasts.

It is also shown that the *model* $\times$ *theme* contrast demonstrates an interaction effect meaning that the effect of *model* depends upon *theme*: or, model sizes perform differently depending upon the theme on which they are fine-tuned. This interaction effect is demonstrated in 3, where lines do not run parallel.

Pairwise comparisons are used to explore the permutations of *theme* and identify where the significant effect occurs against the DV, ADEM. As can be seen in Table 2, the sci-fi theme is significantly different from all other themes. We attribute this observation to the larger dataset that was used in the fine-tuning process and the variety of narrative that exists across the corpora used.

Table 2: (a) Significance testing across model size and theme factors showing significance difference exists across themes, warranting exploration with pairwise comparisons. No significant difference exists across model, so no pairwise comparisons are performed. (b) Pairwise comparisons between the different themes based upon the ADEM Score. I and J are Themes and I-J indicates the difference between the theme average ADEM scores. * indicates the mean difference is significant at the .05 level.

(a)

	df	F	Sig.
ModelSize	2	1.539	.215
Theme	3	10.432	.000
ModelSize * Theme	6	5.147	.000

(b)

(I)	(J)	Diff (I-J)	Sig.
Modern	Pirates	-.0149	.431
	Sci-fi	-.0508*	.000
	Norse	-.0074	.875
Pirates	Modern	.0149	.431
	Sci-fi	-.0359*	.002
	Norse	.0075	.874
Sci-fi	Modern	.0508*	.000
	Pirates	.0359*	.002
	Norse	.0434*	.000
Norse	Modern	.0074	.875
	Pirates	-.0075	.874
	Sci-fi	-.0434*	.000

5 Discussion

The results yield a number of useful findings aligned to NLP for immersive worlds. As discussed earlier, NPC dialogue is commonly perceived as being predictable or scripted, using NLP to generate the dialogue has produced thematically seeded language that will in the future allow games to include a more dynamic and rich dialogue that will help increase player immersion. Dependant upon the game type, machine generated dialogue can also help to remove a significant workload from the developer and in turn allow for more development time to be focused upon game mechanics and story.

Whilst this technology is still in its infancy its many uses have already become apparent. Research into the area of video games and their uses continually grows to show they are useful beyond entertainment, video games are now being used regularly in areas such as education and professional training. Role-playing games have been used to facilitate therapy and education to young adults, [6], and there has been an increase in the use of video games and virtual reality to train officers in police forces across the world, as outlined in a study about training officers within virtual environments [1]. What all of these techniques require to be effective is immersion within the given scenario, a topic that established its own line of research, studies have also shown and spoken directly about the

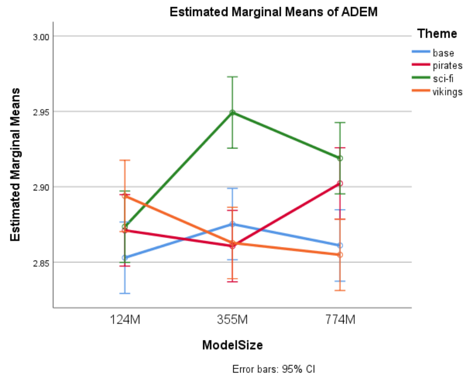


Fig. 3: Estimated Marginal Means of ADEM score across model size considering theme. This shows the influence of theme fine-tuning on dialogue generation performance and also illustrates that there is an interaction effect between these two IVs.

importance of immersion techniques to increase engagement and realism within these virtual worlds [19], [5]. Moving forward, the system of dialogue generation through natural language processing explored in this paper can be built upon and incorporated into any nature of projects that either rely upon immersive narrative as a method for increased engagement or at least would benefit from an enriched experienced through the use of believable characters that provide the user or player with unpredictable dialogue and narrative. This could be for entertainment purposes through video games or for more immersive and believable training scenarios within virtual simulations.

6 Conclusion, Limitations and Future Work

This paper presents experimental studies with the ultimate goal of demonstrating a practical framework for the generation of NPC dialogue in virtual environments. It has successfully showcased a platform agnostic API that has the ability to generate thematically correct dialogue within a game environment.

The current limitations of the framework are the slow processing times based upon the generation of dialogue via the models, further development could include potentially pre-processing responses in advanced to reduce the initial response time significantly, this could potentially help build a higher level of realism within the game and in turn make it more of a viable solution for future applications. Another limitation was the size of the initial data sets used to fine tune each of the thematic models. Using larger, richer data sets will provide richer and more thematically accurate responses, as alluded to in Section 4. This was proven with a net higher average ADEM result within our sci-fi model, this can be seen in Figure 3. Initially and for proof-of-concept, we fine-tuned the models on scripts from popular TV shows that aligned with the thematic required. Dedicated bodies of text or larger data sets more targeted around each of the themes would greatly improve not only the quality of responses on a whole, but the reliability of each response fitting with the theme set out in the developers story arc. Naturally, these themes can be extracted from the narrative of appropriate target media, for example games and also be supplemented by story writers aligned to a particular development exercise. Accounting for these limitations in future developments would create a more robust framework for adoption in the field.

References

1. Bertram, J., Moskaliuk, J., Cress, U.: Virtual police: Acquiring knowledge-in-use in virtual training environments. In: 2011 IEEE International Symposium on VR Innovation. pp. 341–342 (2011)
2. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners (2020)
3. Davies, R., Harvey, C., Dewell, N.: GPT-2 unity thematic dialogue api, `https://github.com/RichardTHF/GPT-2-Thematic-Narrative-Generation-for-Unity-and-Unreal`
4. Erkel, M., Person, N.: Autotutor passes the bystander turing test. In: Proceedings of E-Learn 2002–World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education. pp. 778–784. Association for the Advancement of Computing in Education (AACE) (2002)
5. Gorini, A., Capideville, C.S., De Leo, G., Mantovani, F., Riva, G.: The role of immersion and narrative in mediated presence: the virtual hospital experience. Cyberpsychology, Behavior, and Social Networking **14**(3), 99–105 (2011)
6. Hawkes-Robinson, W.: Role-playing games used as educational and therapeutic tools for youth and adults. Rpgresearch.com (12 2008)
7. Jenny, B., Staffan, B.: Gameplay design patterns for game dialogues. In: DiGRA - Proceedings of the 2009 DiGRA International Conference: Breaking New Ground: Innovation in Games, Play, Practice and Theory. Brunel University (September 2009), `http://www.digra.org/wp-content/uploads/digital-library/09287.59480.pdf`
8. Kacmarcik, G.: Using natural language to manage npc dialog. In: Proceedings of the Second AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment. p. 115–117. AIIDE'06, AAAI Press (2006)
9. Li, J., Monroe, W., Shi, T., Ritter, A., Jurafsky, D.: Adversarial learning for neural dialogue generation. CoRR **abs/1701.06547** (2017). https://doi.org/10.18653/v1/D17-1230, `http://arxiv.org/abs/1701.06547`
10. Liu, P.J., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L., Shazeer, N.: Generating wikipedia by summarizing long sequences. CoRR **abs/1801.10198** (2018), `http://arxiv.org/abs/1801.10198`

11. Lowe, R., Noseworthy, M., Serban, I.V., Angelard-Gontier, N., Bengio, Y., Pineau, J.: Towards an automatic Turing test: Learning to evaluate dialogue responses. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1116–1126. Association for Computational Linguistics, Vancouver, Canada (Jul 2017). https://doi.org/10.18653/v1/P17-1103, https://www.aclweb.org/anthology/P17-1103
12. Mccoy, J., Treanor, M., Samuel, B., Tearse, O., Mateas, M., Wardrip-fruin, N.: N.: Authoring game-based interactive narrative using social games and comme il faut. In: In: Proceedings of the 4th International Conference Festival of the Electronic Literature Organization: Archive Innovate. Providence (2010)
13. Orkin, J., Roy, D.: Semi-Automated Dialogue Act Classification for Situated Social Agents in Games, pp. 148–162. Springer Berlin Heidelberg, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-18181-$8_1$1, https://doi.org/10.1007/978-3-642-18181-8_11
14. Radford, A., Narasimhan, K., Salimans, T., , Sutskever, I.: Improving language understanding by generative pre-training. Tech. rep., OpenAI (2018)
15. Rich, C., Sidner, C.L.: Using collaborative discourse theory to partially automate dialogue tree authoring. In: Nakano, Y., Neff, M., Paiva, A., Walker, M. (eds.) Intelligent Virtual Agents. pp. 327–340. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)
16. Rieser, V., Lemon, O.: Reinforcement Learning for Adaptive Dialogue Systems: A Data-driven Methodology for Dialogue Management and Natural Language Generation. Springer (2011)
17. Rocktäschel, T., Grefenstette, E., Hermann, K.M., Kočiský, T., Blunsom, P.: Reasoning about entailment with neural attention (2015)
18. Sai, A.B., Gupta, M.D., Khapra, M.M., Srinivasan, M.: Re-evaluating adem: A deeper look at scoring dialogue responses (2019)
19. Silva, R.A.: Offland: Narratives, Immersion and Interaction In Virtual Reality. Master's thesis (2020)
20. Subtitles, O.: opensubtitles.org (2020), https://www.opensubtitles.org/en/search/subs
21. Technologies, L.: ftfy: fixes text for you (2020), https://github.com/LuminosoInsight/python-ftfy
22. Turing, A.: Computing machinery and intelligence. Mind **59**(236), 433–460 (10 1950). https://doi.org/10.1093/mind/LIX.236.433, https://doi.org/10.1093/mind/LIX.236.433
23. Woolf, M.: Gpt-2-simple (2019), https://github.com/minimaxir/gpt-2-simple

A Appendix

Table 3: Examples of seeds fed into the model and evaluation responses for permutations of scenes and model sizes. t is the time taken in seconds for the model evaluation and response generation. ADEM* is an average across 14 evaluations based upon human response

	Model	Seed	Model Response	Human Response	t	ADEM*
Norse	124M	I want a drink!	A drink of wine!	You want a drink? How about some Mead?	4.46	2.83
	355M	Get out of my way!	Fight! Mmm. Sure thing. It's time!	Are you looking for a fight?; I would choose your words more wisely next time;	9.26	2.88
	762M	What weapons do you sell?	There are many things here for sale.	I've got plenty of stuff about the shop, just right for making your enemies scream.	14.70	2.84
Sci-fi	124M	Get out of my way!	Please don't hit me. You're not in your right mind.	No need to be so pushy!	5.93	3.21
	355M	What weapons do you sell?	sir. We've got quite the arsenal. How do you even know where to start?	Anything you could possibly need	9.37	2.72
	762M	I want a drink!	He wanted one and he got one!	The bar is down the road	16.41	2.91
Modern	124M	Get out of my way!	All you dudes get out of my way!	My weapons work very well on people like you.	6.14	2.94
	355M	I want a drink!	Do you need any help?	look at the menu	11.39	2.87
	762M	What weapons do you sell?	There are currently no weapons available for sale.	All sorts, depends what you're after	15.49	2.74
Pirates	124M	I want a drink!	When you need a drink! Come back later!	Your looking for in the wrong place matey try another deck	5.33	2.93
	355M	Get out of my way!	Get me a sail. Take me up the trail. All is well here?	Aarrrghhh this was the last tale you be telling!	10.35	2.96
	762M	What weapons do you sell?	Anything. I've got a supply caravan moving cargoes up from the beach	anything you want lad!; guns, pistols, rum anything!;	13.07	3.01

Interactive Self-Assembling Agent Ensembles

Samuel Truman and Sebastian von Mammen

Games Engineering, Julius-Maximilians University, Würzburg, Germany
{samuel.truman,sebastian.von.mammen}@uni-wuerzburg.de

Abstract. In this paper, we bridge the gap between procedural content generation (PCG) and user-generated content (UGC) by proposing and demonstrating an interactive agent-based model of self-assembling ensembles that can be directed though user input. We motivate these efforts by considering the opportunities technology provides to pursue game designs based on according game design frameworks. We present three different use cases of the proposed model that emphasize its potential to (1) self-assemble into predefined 3D graphical assets, (2) define new structures in the context of virtual environments by self-assembling layers on the surfaces of arbitrary 3D objects, and (3) allow novel structures to self-assemble only considering the model's configuration and no external dependencies. To address the performance restrictions in computer games, we realized the prototypical model implementation by means of an efficient entity component system (ECS). We conclude the paper with an outlook on future steps to further explore novel interactive, dynamic PCG mechanics and to ensure their efficiency.

Keywords: Procedural Content Generation · User-Generated Content · Game Mechanics · Agent-based Models · Self-Assembly

1 INTRODUCTION

As an extension to the broadly applied foundation of the MDA framework (mechanics, dynamics, aesthetics) on game design [12], the DPE framework (design, play and experience) emphasizes the technological basis of game design as it determines the potential design space in terms of user experience, gameplay, storytelling, and learning in a pedagogical sense, if applicable [22]. The technological perspective and its tight relationship with game design is especially obvious in the context of PCG, where various elements of a game immediately emerge from algorithmic instructions. In particular, PCG has been applied to small *bits* of a game such as graphical assets, behavioral descriptions or local effects, their combination into *spaces*, most prominently defined through maps, *systems* that put bits into relation (also spatially), *scenarios* that introduce gameplay challenges, *designs* that combine various elements into playable experiences, and *content* which drives the experiences [4]. More recent advances of the MDA framework and its successors, e.g. the DDE framework (design, dynamics, experience) [18] shift the focus towards the possible and emerging relationships between the player subject and the antagonist that considers the entirety of designed aspects the player is exposed to. The essence of the DDE framework is, accordingly, the need to iteratively consider all the different, tightly interwoven design aspects to arrive at a wholistic design of an interactive experience. The resulting increase in the frequency of playtests culminates in the idea of PCG that adapts game design to specific player models in advance or during play, see e.g. [23,14,8,10].

Different from such *adaptive* PCG approaches, in this paper we consider another direction that *dynamic* PCG can take[1]. In particular, we demonstrate how ensembles [5] of agents [2] can implement rather versatile interfaces to inform dynamic PCG processes. In the following section, we refer to several concepts that motivated and approaches that we built an exemplary model on, which we detail in Section 3. We showcase its capabilities and discuss its limitations. We conclude this paper with a short summary and considerations for future work.

2 RELATED WORK

PCG refers to algorithmically generated content [13], whereas UGC refers to content that platform users create, a.o. players of virtual worlds [7]. Such binary distinction quickly fades away when looking at crafting systems which provide mechanics for players to proactively contribute to the generation of game-related assets [3]. In [6], a 5-type taxonomy for crafting systems is introduced: type 1 implies that certain resources are traded for items (or upgrades of items) which could be reduced to the use of currency. Type 2 requires the player to uncover and follow a specific recipe to yield certain items. Type 3 crafting systems allow the player to explore the space of possible (pre-defined) recipes by themselves, whereas type 4 systems allow the player to make several choices which result in custom items, e.g. by combining different materials. Finally, type 5 are considered “true” crafting systems in the sense that minute details that the player determines interact with each other in a complex manner. Type 5 systems challenge the player in acquiring the knowledge how these interactions play out and in crafting solutions that meet some given requirements. As type 5 crafting systems generate outcomes based on computing the interplay of some inputs, they also qualify as procedural systems, bridging the gap between UGC and PCG, and potentially elevating “the crafting system to being the primary game mechanic” [6].

Arbitrary generative models can serve as a basis for systems at the boundary between UGC and PCG. Individually designed components might, for instance, *self-assemble* “into patterns or structures without human intervention” [21]. Simple models of self-assembly can be implemented by modeling virtual agents [2] that attract each other and stick together (*static* self-assembly), yielding higher-level artifacts. These artifacts may exhibit properties that *emerge* from the in-

[1] We refer to dynamic PCG to particularly highlight the need and capability to procedurally generate on-the-fly to yield new solutions dynamically as, for instance, pursued in [24,20,19].

terplay of the self-assembled *ensemble* of lower-level agents, i.e. that none of the lower-level agents exhibit by themselves [5]. In this paper, we present an interactive generative model that implements according self-assembling agent ensembles. We do not focus on the specific requirements of the user interface which arise when instructing large numbers of self-organizing agents [16] but rather on a description of the model and its implementation concept to achieve real-time performance.

One of the authors previously presented works in which agent ensembles were evolved and guided to grow three-dimensional structures based on tracing the agents' trajectories, whereas some agents proliferated to yield branching structures [9] and others concerted their flight to generate braids [17]. These approaches featured *reactive* agents [15] that preform actions based on their perceived environment and internal states. In particular, they extended the original "boids" flocking model based on neighboring agents' states [11] in terms of the aforementioned structural trace and behavioral augmentations. Reactive agents have also been used to interactively design large biomolecular models [1].

3 MODEL

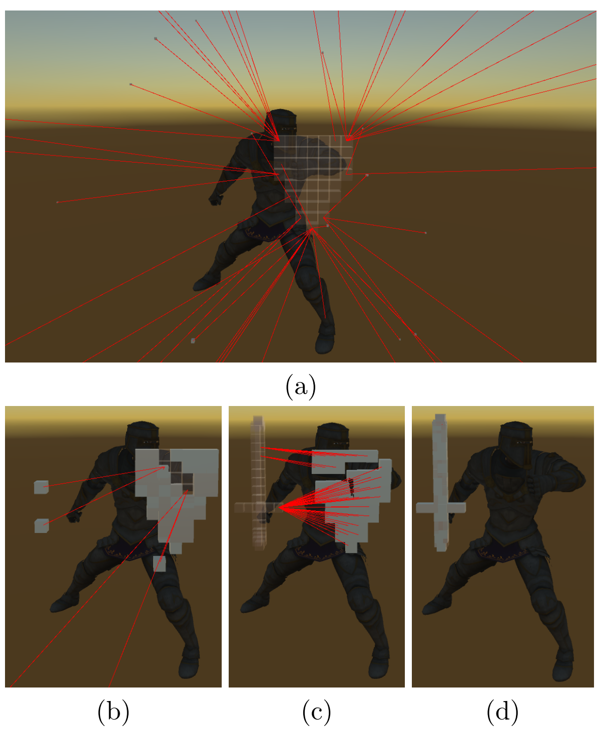
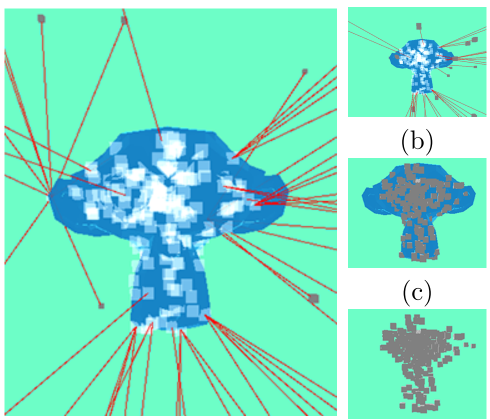


Fig. 1. (a) Agent ensembles that are distributed across the scene (b) assemble into a shield, and (c) transition into (d) a sword.

In the preceding sections, we elaborated about PCG in the context of game design and motivated agent-based PCG mechanics directly made available to the player. To further this concept in the context of computer games, we implemented a simple agent-based model that can implement rules of self-assembly but can also be guided by the player subject or the antagonist (following the terminology of the DDE framework [18]). We realized this by agents (represented as grey cubes) that are stationary and exhibit ports (semi-transparent grey cubes) for other mobile agents to lock on. In the mobile state, an agent does not exhibit ports but seeks them out within its vicinity to proactively come into port. Agents will move to their target position in a straight line and are allowed to pass through each other. Currently, all agents share the same appearance, dimensions and speed. While several agents may move toward the same port, agents encountering an occupied port either stop moving or choose an alternative destination. Spatial conflicts are resolved by automatic offsets to the side.

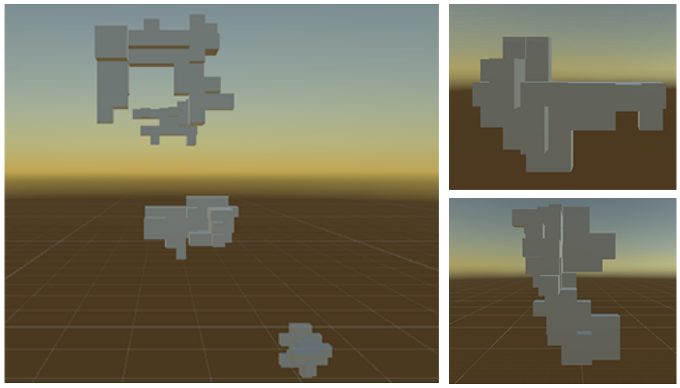


(a) (b) (c) (d)

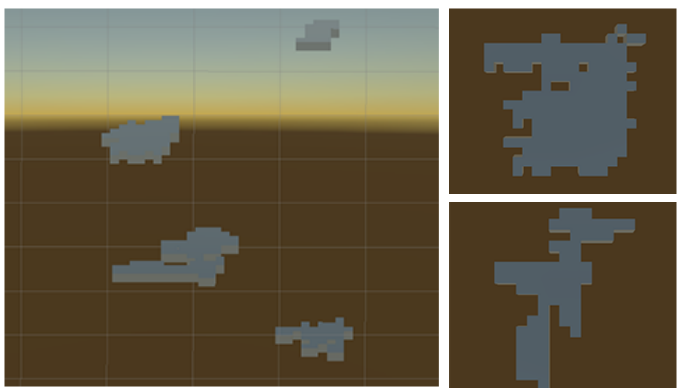
Fig. 2. The agents lock onto the closest ports that were randomly generated on the surface.

We implemented three concrete model instances, each promoting a different degree of user interaction, to demonstrate some of the capabilities of our model: Use case 1 shows how the agents can assemble into different assets over time. Use case 2 shows how the agents can dynamically occupy a surface. Use case 3 highlights the agents' capacity to self-assemble without the need for a shape that provides context. At the beginning of each assembly process, the available agents are randomly distributed within a well-defined box that is centered at the origin of the scene. The initial distribution of agents can have a major impact on the outcome as their placement determines their locally perceived information. Figure 1 shows use case 1, i.e. how the agents first assemble into a warrior's shield and transition into his sword. The shapes' port information is precomputed, whereas the agents' destinies are computed during runtime, once the player triggers the assembly of one or the other shape.

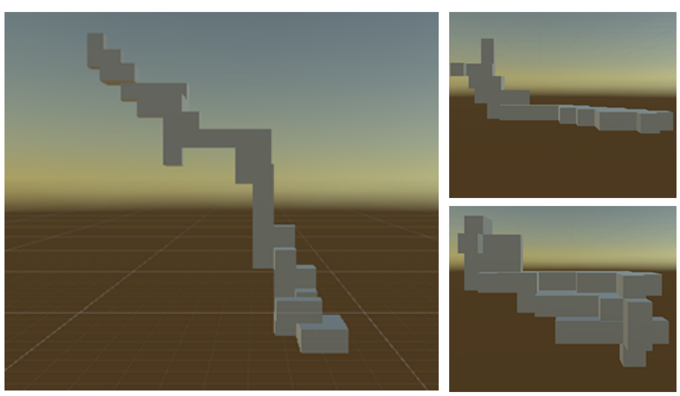
Use case 2 is demonstrated in Figure 2. It shows how an assembly instruction is dynamically applied to the environment without pre-computation: The user designates an object which triggers the random generation of ports on its surface and the agents' embarkment. We consider the relative surface area of each triangle when calculating the probability of its occupation. However, complex meshes with double-sided faces or inner faces might skew the odds. Figure 3 displays outcomes of use case 3, i.e. self-assembly without contextual port definitions as in use cases 1 and 2. Rather, the agents stick together and grow into clusters by themselves. With a given probability, each agent determines whether to remain mobile or switch into the stationary state at the beginning of the simulation. The mobile agents dock onto free ports based on proximity, turn static and open up new ports themselves. The three examples in Figure 3 have been obtained by different rule sets that specify port availability.



(a)



(b)



(c)

Fig. 3. (a) Agent ensembles can dock onto free ports without restrictions. (b) Port generation by stationary agents is restricted to two dimensions. (c) Agent ensembles can only dock to the most recently generated ports.

In the first example (Figure 3(a)), stationary agents open up ports on each vacant side and, as a consequence, mobile agents are not limited to grow the cluster into any direction. In the second example (Figure 3(b)), only ports in the xy-plane are generated by the stationary agents which limits the growth to flat platforms. In the third example (Figure 3(c)), mobile agents can only dock onto ports opened by those agents that docked onto a cluster most recently. All the ports previously opened by agents that had docked on before, but one random one, are removed.

Efficiency is a great challenge when considering interactive PCG in the context of games considering the multifaceted intertwined processes that make for a proper, immersive experience. Therefore, we designed our implementation accordingly and built it on top of Unity's Data Oriented Tech Stack (DOTS). It combines an entity component system (ECS), a job system, and a performant compiler especially to promote large numbers of interacting entities. Implementing the presented agents as entities, each one is composed of a set of uniquely identifiable data components, which can be iterated at high speed, especially due to a systematic avoidance of cache misses.

4 SUMMARY AND FUTURE WORK

We motivated the use of interactive, dynamic PCG as a play mechanic based on broad perspectives of game design. Based on examples of agent ensembles that have previously been presented in interactive generative and simulative contexts, we proposed a model of agent ensembles that self-assemble based on player input. We demonstrated three according use cases where concrete model instances lead to self-assembly of different structures constrained by docking rules (use case 3), to agent coverage of arbitrary surfaces to highlight the flexibility of the model in a given virtual environment context (use case 2), and to player-triggered dynamic transformation of the ensemble into different predefined assets. We addressed the high performance requirements of interactive, dynamic PCG by realizing our model by means of an efficient ECS system.

In order to push interactive PCG in games, further interactions and especially their impact on play need to be explored. To this end, small whitebox prototypes could unearth some new directions where such game elements could lead. Models, technological realization and game mechanics should be systematically embedded into game design frameworks such as MDA, DPE or DDE to better explore their impact and the space for design opportunities they unfold. With the first steps of "true" design capabilities of interactive PCG—analogous to the quote about "true" crafting systems in Section 2—larger cascades and cycles of interlocking mechanics should be considered at a systematic, abstract level as well. For instance, based on the model instances presented in this paper, one could investigate the impact of dynamic assets on ecosystems of online multiplayer platforms that might correlate to actual processing and storage capacities. Or, as another example, one could investigate the physical impact of user-created assets in the context of competitive fighting, racing or sports games, etc.

As mentioned above, the performance requirements are also high, both in terms of real time capabilities at small scopes of interactive, dynamic PCG or, if those can be ensured, for scaling up the scope, e.g. the numbers of agents and their capabilities. To this end, various development challenges can be overcome. For instance, our implementation could have harnessed the power of the utilized ECS framework more rigorously by identifying and setting up more jobs that can be run in parallel or by using ECS during the initialization step or when novel agents are spawned during runtime. Well-established acceleration algorithms such as spatial data structures could further reduce the number calculations to determine interactions. The performance impact of structural changes, i.e. changing an entity's archetype by adding or removing components, should be evaluated. In case structural changes are too costly, boolean flags might be used.

References

1. Davison, T., Samavati, F., Jacob, C.: Lifebrush: Painting interactive agent-based simulations. In: 2018 International Conference on Cyberworlds (CW). pp. 17–24. IEEE (2018)
2. Denzinger, J., Kordt, M.: Evolutionary on-line learning of cooperative behavior with situation-action-pairs. In: ICMAS. pp. 103–110 (2000)
3. Francis, B.: 7 crafting systems game designers should study. Online at `https://www.gamasutra.com` (July 2015)
4. Hendrikx, M., Meijer, S., Van Der Velden, J., Iosup, A.: Procedural content generation for games: A survey. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) **9**(1), 1–22 (2013)
5. Jun, H., Liu, Z., Reed, G.M., Sanders, J.W.: Ensemble engineering and emergence. In: Software-Intensive Systems and New Computing Paradigms, pp. 162–178. Springer (2008)
6. King, A.: 5 approaches to crafting systems in games (and where to use them). Online at `https://gamedevelopment.tutsplus.com` (January 2015)
7. Lastowka, G.: User-generated content and virtual worlds. Vand. J. Ent. & Tech. L. **10**, 893 (2007)
8. Lopes, R., Hilf, K., Jayapalan, L., Bidarra, R.: Mobile adaptive procedural content generation. In: Proceedings of the fourth workshop on Procedural Content Generation in Games (PCG 2013), Chania, Crete, Greece (2013)
9. von Mammen, S., Jacob, C.: Genetic swarm grammar programming: Ecological breeding like a gardener. In: Srinivasan, D., Wang, L. (eds.) CEC 2007, IEEE Congress on Evolutionary Computation. pp. 851–858. IEEE Press, Singapore (2007)
10. Oliveira, S., Magalhães, L.: Adaptive content generation for games. In: 2017 24º Encontro Português de Computação Gráfica e Interação (EPCGI). pp. 1–8. IEEE (2017)
11. Reynolds, C.W.: Flocks, herds, and schools: A distributed behavioral model. Computer Graphics **21**(4), 25–34 (1987)

12. Robin Hunicke, Marc LeBlanc, R.Z.: Mda: A formal approach to game design and game research. In: Proceedings of the AAAI-04 Workshop on Challenges in Game AI. pp. 1–5 (2004)
13. Shaker, N., Togelius, J., Nelson, M.J.: Procedural Content Generation In Games. Springer (2014)
14. Shaker, N., Yannakakis, G.N., Togelius, J.: Towards player-driven procedural content generation. In: Proceedings of the 9th conference on Computing Frontiers. pp. 237–240 (2012)
15. Stuart, R., Peter, N.: Artificial Intelligence: A Modern Approach. Prentice Hall, Upper Saddle River, NJ, 4th edn. (2020)
16. von Mammen, S.: Self-organisation in games, games on self-organisation. In: Games and Virtual Worlds for Serious Applications (VS-Games), 2016 8th International Conference on. pp. 1–8. IEEE (2016)
17. Wagner, D., Hofmann, C., Hamann, H., von Mammen, S.: Design and exploration of braiding swarms in vr. In: Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology. p. 13. Gothenborg, Sweden (November 2017)
18. Walk, W., Görlich, D., Barrett, M.: Design, dynamics, experience (dde): an advancement of the mda framework for game design. In: Game Dynamics, pp. 27–45. Springer (2017)
19. Washburn, M., Khosmood, F.: Dynamic procedural music generation from npc attributes. In: International Conference on the Foundations of Digital Games. pp. 1–4 (2020)
20. Wheat, D., Masek, M., Lam, C.P., Hingston, P.: Dynamic difficulty adjustment in 2d platformers through agent-based procedural level generation. In: 2015 IEEE International Conference on Systems, Man, and Cybernetics. pp. 2778–2785. IEEE (2015)
21. Whitesides, G.M., Grzybowski, B.: Self-assembly at all scales. Science **295**(5564), 2418–2421 (2002)
22. Winn, B.M.: The design, play, and experience framework. In: Handbook of research on effective electronic gaming in education, pp. 1010–1024. IGI Global (2009)
23. Yannakakis, G.N., Togelius, J.: Experience-driven procedural content generation. IEEE Transactions on Affective Computing **2**(3), 147–161 (2011)
24. Zook, A., Riedl, M.: A temporal data-driven player model for dynamic difficulty adjustment. In: Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment. vol. 8 (2012)

Overlapping Architecture: Implementation of Impossible Spaces in Virtual Reality Games

Rafael Epplée[1] and Eike Langbehn[3,2]

[1] Universität Hamburg, Germany
[2] Curvature Games, Germany
https://curvaturegames.com
[3] Hamburg University of Applied Sciences, Germany
eike.langbehn@haw-hamburg.de

Abstract. Natural walking in virtual reality games is constrained by the physical boundaries defined by the size of the player's tracking space. Impossible spaces, a redirected walking technique, enlarge the virtual environment by creating overlapping architecture and letting multiple locations occupy the same physical space. Within certain thresholds, this is subtle to the player. In this paper, we present our approach to implement such impossible spaces and describe how we handled challenges like objects with simulated physics or precomputed global illumination.

Keywords: Virtual Reality · Games · Locomotion.

1 Introduction

Virtual Reality (VR) enables immersive gaming experiences which provide more natural spatial cues than games on a 2D screen. Natural locomotion further increases the sense of presence when exploring these virtual environments (VEs) [8], but is limited by the size of the physical tracking space of the player.

Since VEs are often larger than the available tracking space, different locomotion techniques have been employed to help the user navigate these VEs. Among these are joystick-based continuous motion [12], teleportation [1], and redirected walking (RDW) [9].

Impossible spaces are another technique that leverages self-overlapping architecture to build layouts that would be impossible in the real world. By making virtual rooms partially overlap with each other, the available virtual space can be enlarged without users noticing. However, if the overlap is too large, some users will start to detect it [11]. Figure 1 shows an example room layout using impossible spaces.

By using impossible spaces, players can explore larger virtual worlds in confined real-world play spaces by natural walking which is known to be more presence-enhancing and causes less disorientation [12]. Hence, the whole player experience can be improved with impossible spaces. However, there are some technical obstacles in the implementation of this technique.

In this paper, we present our approach of implementing impossible spaces in the Unity engine. There are a couple of studies that evaluate the perceptibility of impossible spaces, but to our knowledge there is no publication that discusses benefits and drawbacks of different implementations. Section 2 describes related work in the field. Section 3 outlines our actual implementation of impossible spaces, detailing interesting edge cases and how we handled them. Section 4 concludes the paper.

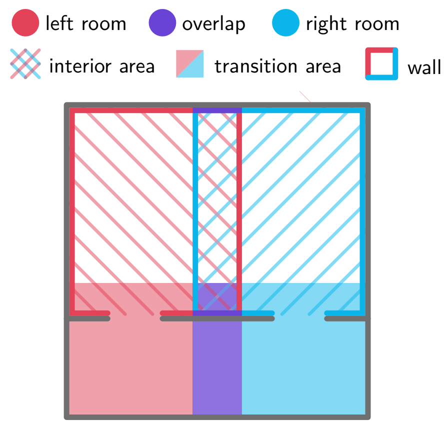


Fig. 1. An illustration of two overlapping rooms with their respective interior and transition areas.

2 Related Work

Impossible spaces were introduced by Suma et al. [11,10], who demonstrated their effectiveness in two experiments that investigated the possible amount of overlap.

In their experiments, Suma et al. reported that participants without prior knowledge of impossible spaces noticed them hardly ever, even with overlaps of 75% (which exceeds the absolute detection threshold mentioned in the experiment by far) [11]. Even when participants noticed the overlap, they only referred to them as "weird" or "strange", and none reported a negative effect on their experience.

In terms of manipulation detection, a corridor with additional turns is even more effective than a longer corridor [13], and curved corridors between the overlapping rooms are more beneficial than right-angled corridors [14].

In addition, there were also approaches to generate overlapping rooms automatically at runtime, called flexible spaces [15], and experiments that combined impossible spaces and redirected walking techniques [2].

Impossible spaces have already been used in games as well, for example by the VR game Unseen Diplomacy [6]. It arranges the (self-overlapping) rooms in such a way that the player does not recognize that she is only walking back and forth in a 3m x 4m large area. The game Tea for God even creates their impossible spaces procedurally for each run and can handle flexible tracking space sizes [7]. There is a toolkit for the Unity engine that enables procedural environment generation and manual world-building for impossible spaces[4].

2.1 Implementation with Portals

A popular technique for implementing impossible spaces are portals, virtual "windows" that teleport users to a different location when stepped through. This technique was used to great effect in the 2007 video game "Portal"[5], which made portals an explicit gameplay element used for solving puzzles. It is also the technique underlying the TraVRsal toolkit for building worlds with impossible spaces [16] which uses portals imperceptibly to maintain the illusion of a normal, non-overlapping world. Previously, non-VR games such as the 2013 video game "Antichamber" used portals like this as well to create unusual, challenging puzzles [4].

While portals promise a conceptually simple, elegant implementation of impossible spaces, they are technically complex and hard to implement. The game "Portal" and the TraVRsal toolkit implemented them using the stencil buffer [3], essentially having a fragment shader render pixels depicting portals' surfaces using a camera with a transformed location. This needs additional work to correctly simulate light and objects traveling through portals. Tricks such as duplicating light sources on both sides of the portal help, but often still leave deficiencies, such as the lack of support for light transfer when precomputing global illumination [17]. Stereographic rendering can lead to rendering artifacts that require further workarounds ([17]). Portals visible through other portals require special consideration and are usually limited in their depth. For this reason, the TraVRsal toolkit places tight constraints on the locations and number of portals, and does not support portals visible through other portals at all ([18]).

In summary, it is difficult to implement portals in a truly imperceptible way. There are many situations and interactions with other parts of the application that need to be considered, e.g. other shaders, objects controlled by players, lighting, rendering of portals visible through other portals, and more. Stylized graphics as found in the earlier mentioned game Antichamber might make it easier, making it favorable for certain scenarios. But even Antichamber contained some situations in which the limitations of the implementation became apparent, breaking the illusion of a continuous non-euclidean space ([4]). Removing other features like physics simulation from an application might make implementing portals easier as well, but this limits the technique to certain environments.

[4] https://blog.wetzold.com/2020/07/02/an-editor-for-impossible-spaces-in-virtual-reality/

[5] `https://en.wikipedia.org/wiki/Portal_(video_game)#Gameplay`

Implementations usually work by repeating the render process to render the view into a portal, which incurs a manageable, but significant performance overhead. Using portals requires overlapping rooms to be placed at a different location than the rest of the level, causing inconvenience for level designers.

3 Implementation

Besides portals, which work by imperceptibly teleporting users, there are also other methods of implementing impossible spaces. The following section outlines our approach to implement impossible spaces along with its advantages and disadvantages: manipulating room visibility, which works by dynamically hiding geometry that would reveal an overlap to users. We provide the rationale for choosing this approach for our implementation and describe the way it works, including special cases it handles and limitations of its capabilities. We explain the mechanics behind additional features, namely the handling of objects with simulated physics, precomputed global illumination (GI), and room doors. The source code is permissively licensed and available for download.[6]

3.1 Overview

Our implementation method of impossible spaces is to remove overlapping rooms from the world when the player is not near them. By keeping track of the player's current location in the world, which overlapping rooms are visible from that location, and which overlapping rooms that location is inside of, the implementation can reveal and hide the appropriate rooms at the right time.

To achieve this, the space from which an overlapping room is visible, and the space encompassing its interior, are marked by the level designer. Detecting the player entering and leaving these spaces and hiding rooms from view have a negligible performance impact, making this technique feasible in many scenarios, even with lots of overlapping rooms. Implementing this technique is straightforward compared to the portal technique. A mechanism for designating room boundaries makes no assumptions of a project's rendering pipeline and level layout, making it more flexible and easier to integrate than fragment shaders using stencil buffers.

The show-and-hide approach has some drawbacks as well. Similar to the portal technique, hiding rooms requires some additional work to correctly handle objects with simulated physics, and when precalculating global illumination using path tracing, overlapping rooms have to be considered in isolation from the rest of the level to prevent artifacts in areas where they overlap. This imposes some restrictions on developers' workflows. Additionally, level designers have to be careful to prevent situations in which users witness rooms suddenly appearing or disappearing.

While portals might be a favorable approach for some specific virtual experiences, we wanted a generic solution usable for a wide range of applications.

[6] `https://gitlab.com/raffomania/impossible-spaces`

The use of portals imposes restrictions performance budgets, render pipelines and development workflows, and easily leads to edge cases where the technique breaks down. Because the reveal-and-hide approach imposes fewer restrictions and is more robust, we chose it for our implementation.

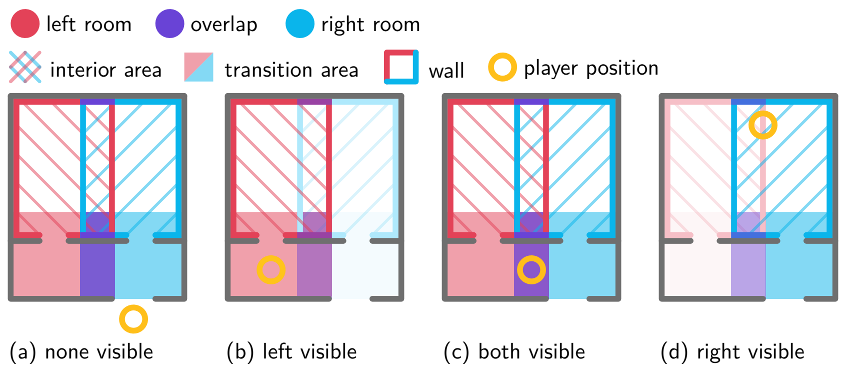


Fig. 2. Examples of different player positions relative to overlapping rooms and their respective visibility states.

3.2 Transition between rooms

In our implementation, overlapping rooms are represented by three components.

A transition area A trigger collider covering the area outside the room. This is the area in which the room should still be visible to the player. When the player leaves this area, it is considered safe to hide the associated room.

An interior area A trigger collider covering the interior of the room. When the player enters this collider, they are considered to be inside the room.

A room container An object containing everything in the room that should be hidden.

See Figure 1 for an example of two overlapping rooms with their interior and transition areas. Figure 2 shows the different visibility states of the same rooms for some player positions. To detect which area the player is currently in, an arbitrary object associated with the player is assigned a collider that triggers collision events when entering and leaving overlapping room areas. An accompanying script processes these events to keep track of the player's location, which room they are currently in and which rooms they can see, activating and deactivating them appropriately. To function correctly, it is important to distinguish when a player is inside a room and when they are outside. Simply looking at one area collision event at a time results in ambiguous situations: for example, figure 2 (d), the player is in the interior area of both rooms. To know

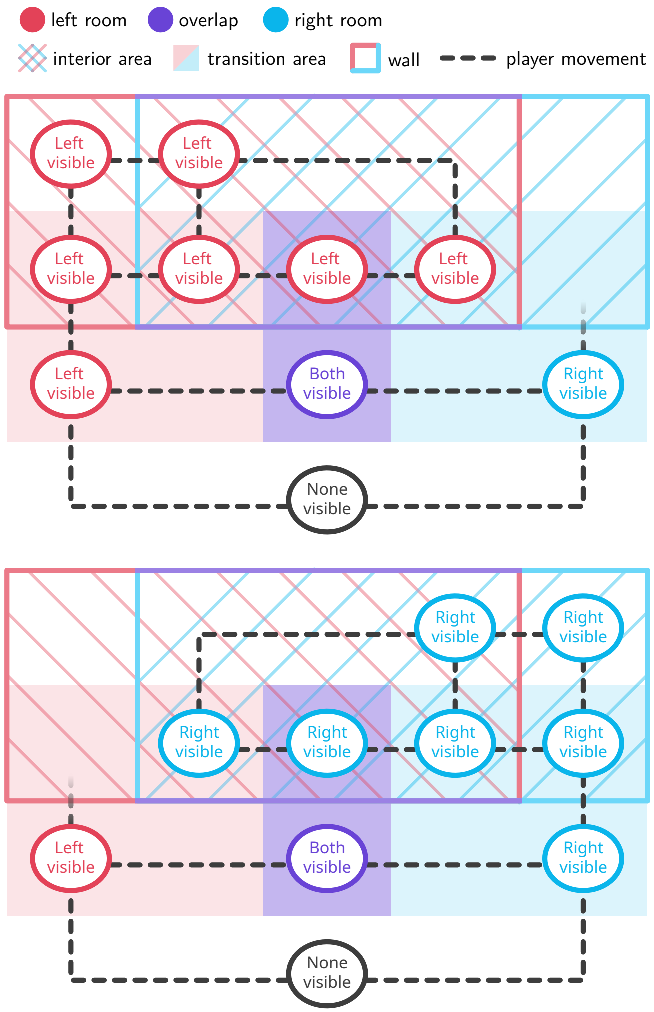


Fig. 3. The behavior of our implementation for a simple case. Execution starts at the node labeled "none visible." Depending on which room's interior players enter first, the top or the bottom graph is traversed, with each node indicating the visibility for both rooms.

which one to show and which one to hide, the implementation has to keep track of the entrance through which a player entered a room. This is why both figure 1 and 2 show the transition area overlapping with part of the interior area of the rooms. This way, the overlap between transition and interior area is treated as the entrance to the room. The implementation knows that a player has entered when they are in the interior area of a room while they are simultaneously in its transition area.

Figure 3 details the implementation's behavior in the context of a simple example. Depending on which room's interior area players enter first, one of the two graphs depicted becomes "active," determining the visibility of both rooms. As the graphs show, once players enter an interior area of a room, that room stays visible until players exit via the corresponding transition area. This behavior is a design decision coming with advantages and disadvantages which we discuss later in this section.

Note that nodes in the overlapping transition areas have less outgoing edges than most other nodes. The player position is determined by discrete events sent to the script by the Unity engine, signaling whether the player has entered or left a specific area. Since only one of these events can arrive at a time, it is not possible to enter or exit two areas at once, e.g. going directly from the "none visible" node to the node labeled "both visible." Specifically, if players moved straight between those two locations, it would result in two successive events, and for a short time only one of the two rooms would be visible.

So far, we have only covered "legal" transitions, that is, transitions where players obey the rules of the physical world when inside the virtual environment. However, in VR, players can move through walls. This means that figure 3 doesn't show every possible transition our implementation had to handle. Figure 4 adds highlighted transitions and their resulting nodes that occur when players move their head across the defined areas in ways that are considered exceptions to the normal behavior. This doesn't necessarily have to be malicious action. Since VR can be disorienting, in some moments players might fail to recognize walls and move through them. At other times, they might be unaware of the volume of their heads, not noticing it intersecting with a wall. In these situations, hiding a room from view the instant a player moves their head outside its boundaries would only add to their confusion. Hence, we decided to keep a room always visible as long as players are inside, meaning as long as they didn't exit via the corresponding transition area. As figure 4 shows, this can have some unfavorable consequences. Once players are outside a room's interior area, if they move towards a different part of the world, e.g. a different room, the implementation is now "stuck," not showing any other room than the one players originally entered. However, we consider this behavior highly unlikely in real-world situations.

A solution to this problem might be to "reset" the state and show a different room once players enter its transition area. However, since transition areas might overlap with interiors of rooms they don't belong to (see figure 4), if a player is inside one room and walks into the transition area of another, this would lead to undesired behavior. As mentioned above, simply hiding rooms once players exit their interior areas would cause undesired states as well. Considering the alternative choices, we judge the current behavior to be an acceptable tradeoff.

3.3 Handling Objects With Simulated Physics

When a room gets hidden, it is temporarily removed from the game. This means that any behavioral components such as bounding boxes for physics simulation

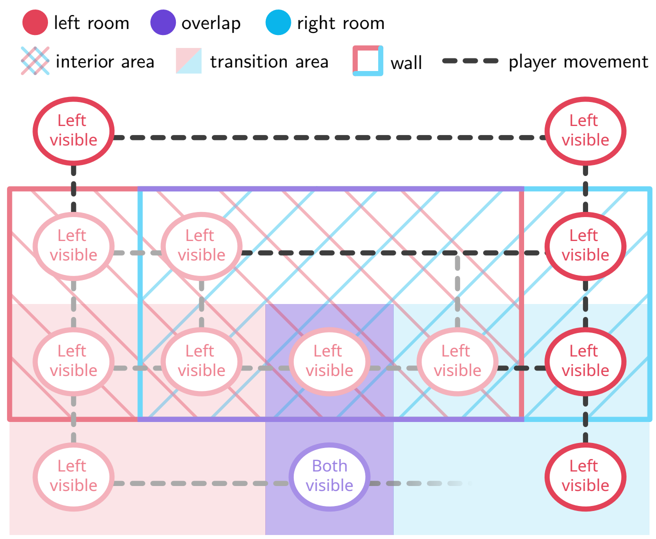


Fig. 4. The behavior of our implementation in the case that players walk through walls. Transitions that lead through walls are highlighted in deep black. States that result from these transitions are highlighted in deep red.

get disabled as well, which poses a problem for dynamic physics objects players can carry around with them. As these are not children of the overlapping room in the scene tree, they won't get hidden along with the room, and might react to the removal of the room. For example, a flashlight left on a table by a player might drop onto the floor. Additionally, objects at the overlapping space between two rooms might react to another room returning to existence.

To prevent this, each overlapping room tracks objects with simulated physics entering and exiting the room, hiding them when another room becomes visible or when the room itself gets hidden.

3.4 Precomputed Global Illumination

A popular technique for modern games is precomputed global illumination (GI). Performance-intense ray tracing computations are executed during development, before running the application. They are encoded in "lightmaps," textures that allow performance-efficient rendering of indirect lighting at runtime. This process is also known as "baking" lightmaps. Baking lightmaps for overlapping rooms can introduce artifacts in the areas they overlap. Since lightmaps do not get updated at runtime, when one room gets hidden from view, the indirect lighting still looks as if both rooms were visible at the same time.

To prevent this, overlapping rooms have to be baked separately. In the Unity engine, this means putting them into different scenes, since that is the only way to restrict the baking process to the subset of the virtual world. This doesn't mean that each and every room needs its own scene, but rather that each area where rooms overlap requires two scenes, one for each room. Rooms that don't overlap with each other can then be grouped in the same scene. As a result, the number of scenes needed for overlapping rooms is equal to the maximum number of rooms that overlap each other in any given situation.

During development, care was taken to support applications with multiple simultaneously loaded scenes by not relying on any compile-time references between rooms or the player component, as these are not supported across scenes in Unity. Users of our implementation will have to be careful about cross-scene references as well if they want to use precomputed GI, which we consider a limitation of our current implementation. To allow per-room lightmap baking without separate scenes, we investigated Unity's render layers. They can be used to draw objects only on certain cameras and restrict physics ray casting to specific groups of objects. We tried to apply this restriction to the lightmap calculation process, but GI calculation is based on path *tracing*, a completely different implementation from physics ray casting and the camera drawing logic, without support for the render layer functionality.

A notable consequence of restricting lightmap baking to a single scene is that no lights from other scenes will contribute to the GI calculated for that scene. This means that a corridor between two overlapping rooms might need its lightmaps to be baked together with both rooms first, to prevent seams at the entrances of rooms, where light might shine from the rooms' interiors into the corridor. Afterwards, lightmaps for both rooms will have to be baked separately, overriding the lightmaps previously generated when baking all scenes at the same time. As a result, lights from inside a room can affect GI in a corridor through entrances, but rays from lights in a corridor will not affect GI inside an overlapping room. Following this procedure might require an additional scene to accommodate the geometry outside overlapping rooms.

4 Conclusion

In this paper, we described problems and challenges related to the implementation of impossible spaces in VR games. We presented our approaches that is available as an open source plugin for Unity and was already used for two different projects: a scientific study [7] and a commercial VR game [8]. The experiences of working with the implementation on this game are documented by Paulmann et al. [5].

References

1. Bozgeyikli, E., Raij, A., Katkoori, S., Dubey, R.: Point & teleport locomotion technique for virtual reality. In: Proceedings of ACM Symposium on Computer-Human Interaction in Play (CHI Play). pp. 205–216 (2016)

[7] https://www.xrdrn.org/2020/12/dead-science-orientation-and-minimaps-in-small-spaces/

[8] https://www.youtube.com/watch?v=eBXya_LiwFY/

2. Langbehn, E., Steinicke, F.: Space walk: a combination of subtle redirected walking techniques integrated with gameplay and narration. In: ACM SIGGRAPH 2019 Emerging Technologies, pp. 1–2 (2019)
3. Murray, T., Vigentini, L.: A study protocol to research and improve presence and vection in vr with a non-euclidean approach. In: 2019 IEEE International Conference on Engineering, Technology and Education (TALE). pp. 1–5 (2019)
4. Möller, H.: Antichamber: A strange world, `https://hendrik.fam-moe.de/wp-content/uploads/2019/02/Antichamber_Analysis.pdf`
5. Paulmann, H., Mayer, T., Barnes, M., Briddigkeit, D., Steinicke, F., Langbehn, E.: Combining natural techniques to achieve seamless locomotion in consumer vr spaces. In: 2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW). pp. 383–384. IEEE (2021)
6. Pixels, T.: *Unseen Diplomacy*. Game [HTC Vive] (April 2016), triangular Pixels. Played November 2016.
7. Room, V.: *Tea for God*. Game [SteamVR] (October 2018), void Room. Played October 2018.
8. Slater, M.: Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. Philosophical Transactions of the Royal Society B: Biological Sciences **364**(1535), 3549–3557 (2009)
9. Suma, E.A., Bruder, G., Steinicke, F., Krum, D.M., Bolas, M.: A taxonomy for deploying redirection techniques in immersive virtual environments. 2012 IEEE Virtual Reality Workshops (VRW) pp. 43–46 (2012)
10. Suma, E.A., Clark, S., Krum, D.M., Finkelstein, S., Bolas, M., Wartell, Z.: Leveraging change blindness for redirection in virtual environments. 2011 IEEE Virtual Reality Conference pp. 159–166 (2011)
11. Suma, E.A., Lipps, Z., Finkelstein, S., Krum, D.M., Bolas, M.: Impossible spaces: Maximizing natural walking in virtual environments with self-overlapping architecture. IEEE Transactions on Visualization and Computer Graphics **18**, 555–564 (2012)
12. Usoh, M., Arthur, K., Whitton, M.C., Bastos, R., Steed, A., Slater, M., Brooks, Jr., F.P.: Walking > Walking-in-Place > Flying, in Virtual Environments. In: Proceedings of ACM SIGGRAPH. pp. 359–364 (1999)
13. Vasylevska, K., Kaufmann, H.: Influence of path complexity on spatial overlap perception in virtual environments. In: Proceedings of the 25th International Conference on Artificial Reality and Telexistence and 20th Eurographics Symposium on Virtual Environments. pp. 159–166. Eurographics Association (2015)
14. Vasylevska, K., Kaufmann, H.: Towards efficient spatial compression in self-overlapping virtual environments. In: Symposium on 3D User Interfaces (3DUI) (2017)
15. Vasylevska, K., Kaufmann, H., Bolas, M., Suma, E.A.: Flexible spaces: Dynamic layout generation for infinite walking in virtual environments. In: 3D User Interfaces (3DUI), 2013 IEEE Symposium on. pp. 39–42. IEEE (2013)
16. Wetzold, R.: An editor for impossible spaces in virtual reality, `https://blog.wetzold.com/2020/07/02/an-editor-for-impossible-spaces-in-virtual-reality/`
17. Wetzold, R.: Non-euclidean stencil portals in virtual reality, `https://blog.wetzold.com/2020/01/04/non-euclidean-stencil-portals-in-virtual-reality/`
18. Wetzold, R.: Stencils sorting geometry, `https://blog.wetzold.com/2019/11/03/stencils-sorting-geometry/`