Using a Space Colonization Algorithm for Lightning Simulation

Nuno Reis
Department of Informatics
Universidade do Minho
Braga, Portugal
a77310@alunos.uminho.pt

António Ramires Fernandes
Algoritmi Centre/Department of Informatics
Universidade of Minho
Braga, Portugal
arf@di.uminho.pt

***Abstract*—We present a method for the procedural generation and rendering of lightning bolts. The geometric definition of the bolt is based on space colonization, exploring the similarity between natural trees and lightning bolts. Regarding rendering we aimed to show the life cycle of a lightning bolt, encompassing the three phases of its formation: stepped leader, return stroke, and dart leader. The strobing effect is also detailed in our proposal. With this work, we create an alternative to the current state of the art modeling, a physically-inspired method, using an heuristic iterative algorithm. Notably, the goal was to create a solution of rivaling quality, while being capable of real-time applications, and fully customizable, both in terms of the geometric definition as well as the final appearance and life cycle animation.**

***Index Terms*—lightning, rendering, space colonization, procedural modeling**

I. Introduction

Lightning occurs when electrically charged regions in the atmosphere, generally negatively charged, and the ground, generally positively charged, temporarily equalize themselves, causing an instantaneous release of energy. The amount of energy carried by this stream of electrons is so incomprehensibly vast that it is able to directly effect the molecular structure of the atmosphere it propagates through, turning the gaseous vastness of the skies into a brightly lit bolt of plasma. In fact, the increase in pressure from this sudden transition is so tremendous that it results in the shock wave we call thunder. These discharges from cloud to ground, labeled downward negative lightning, account for up to ninety percent of all naturally occurring lightning strikes [1].

From the inception of Computer Graphics, the simulation of physical phenomena has been a pursuit of choice for academics within the field. However, the reproduction of lightning, rather than the effect of its strikes, has been woefully underappreciated. While one of the most recent models [2] focuses on approximating physical reality as accurately as possible, the overhead that came with the associated calculations was obvious.

Parallel to the academic pursuit were the reproduction of lightning in games and movies which rely primarily on specially crafted renditions. In the former, specifically, it is not uncommon for lightning to be represented by a sprite randomly selected from a predetermined pool which leads to potential repetition.

Fig. 1. An example of a bolt constructed and rendered using our methods at the moment of maximum brightness.

With this in mind, we sought a manner to approximate the appearance of a lightning bolt while doing away with the expensive computation associated with a physical simulation. Something which might be appealing for real-time rendering while still maintaining both a degree of unpredictability and

978-8-3503-4573-57/22/$31.00

approximation to the real life phenomena.

To this end, we explored the similarity between natural trees and lightning bolts, repurposing an existing algorithm for procedural tree generation: the Space Colonization (SC) algorithm [3]. As it stood, this algorithm had its inception in botany, first used to emulate the venation patterns of leaves [4] and later to simulate the growth of several different types of flora. The goal was to use SC as a basis through which to develop an adaptation capable of producing adequate results, as seen in Fig. 1. Besides the geometric definition, we also present the steps that allow for the rendering of the bolt full life cycle, including the strobing effect. Finally, being an heuristic method, it is fully customizable allowing for a large degree of freedom when creating and rendering bolts.

II. Literature Review

This section serves as a review of relevant literature which inspired or otherwise contributed to our method.

A. Simulating Lightning

When it comes to the manner in which lightning is simulated and its shape drawn, the observed literature can be rather cleanly split into two categories: Physically-Inspired Methods and Qualitative Methods, the latter of which mainly makes use of iterative or procedural algorithms to achieve a desired result. The former, on the other hand, makes use of mathematical and computational manipulation to generate algorithms used to simulate the phenomenon that births lighting as accurately as possible.

The first method to ever research and construct a lightning bolt in the realm of Computer Graphics was conceived by Reed and Wyvill in 1994 [5]. Through the careful observation of countless images, they were able to discern that each segment and, subsequently, each branch incurs on average a $16°$ rotation from the segment that preceded it. By using this value, a repeated randomization of the angle between segments results in a pattern resembling that of the observed lightning bolts.

Sosorbaram et al. [6] use the, at the time, newly discovered dielectric breakdown model (DBM), to generate lightning shapes. Rather than solving the exact equations, they made use of an approximation based on local values. This quickly became the main point of focus in physically-inspired approaches: to better solve or approximate the Laplace equations found in the DBM while sacrificing as little performance as possible.

Kim and Lin [7] propose a different method based on a conjugate gradient. However, despite their best efforts, the iterative nature of this solution still presented a considerable toll on the computational requirements. As such, they revised their work [2], and proposed a distance-based method. This approach makes use of adaptive meshes such as quadtrees and octrees, in conjunction with electrical potential equations, to produce a faster alternative.

The DBM also inspired the method developed by Bryan et al. [8] which combines a qualitative approach with a physically-based approach. Employing Cellular Automata in conjunction with potential differential equations. This was the first method able to resolve lightning in real time.

Yun et al. [9] presents a method for physically-inspired simulation. The authors discovered two integral properties present in electric potentials which can be easily described via cell-types in a grid. Combining this approximation with the DBM, allowed them to solve a gradient between each neighboring voxel and a given point, dramatically reducing the computational requirements. However, there were still times in which their method encountered local minima, a problem they solved by generating waypoints via path-finding algorithms.

Fig. 2 shows the results obtained by the works presented in this section. The result by Kim and Lin [2] is the one which better resembles real life lightning.

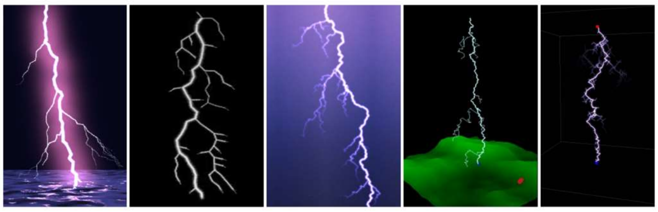


Fig. 2. Results from previous works. From left to right: [5], [6], [2], [8], [9]

B. Rendering Effects

The rendering of lightning has been a topic of contention amid those involved in its simulation. Most resort to ray-tracing to simulate the dispersion of its light across a scene while others focus on the effects of atmospheric scattering observed when it strikes, including the manner in which it interacts with clouds.

Reed and Wyvill [5] pioneered a method in which a blur is applied over the lightning's geometry, simulating the glowing body of hot plasma that is easily observed whenever lightning strikes. A technique that later evolved into bloom, however, at the time they used ray-tracing to simulate the resulting light.

Dobashi et al. [10] focused on these ray-traced effects, looking for an alternate method with which to simulate atmospheric scattering and cloud simulation. By using metaballs, precomputation and, most importantly, billboard techniques to render 3D spaces into 2D scenes before applying lighting effects, they were able to severely reduce the computational load that had been previously observed. For the lightning itself, however, a similar process to Reed and Wyvill was used.

Kim and Lin [2], [7] made use of a similar post-processing technique to simulate glow. Additionally, an atmospheric point spread function was employed to illuminate the surrounding scene. This function, when combined with a convolution kernel, can be used to emulate the scattering of light in a surrounding medium.

Yun et al [9] alter the brightness and thickness of each branch, extending billboard techniques seen in Dobashi et al. Much like the other methods previously described, a blur filter is used to simulate glow. Due to the specific nature of their method, a soft jittering is introduced to mask any artifacts.

Notably, none of the methods seem to focus on any effects other than the light emanated by the lightning itself at its maximum brightness point.

III. Space Colonization

Space Colonization (SC) is an algorithm developed by Runions et. Al [3], [4]. It was first conceptualized as a method for simulating the venation patterns [4] in tree leaves, and latter was expanded to simulate the growth of various types of flora from trees to algae [3]. The method, outlined in Fig. 3 works by treating competition for space as a key factor in determining branch growth, relying on the concept of attractors and nodes, the former competing with each other to influence the latter's growth direction.

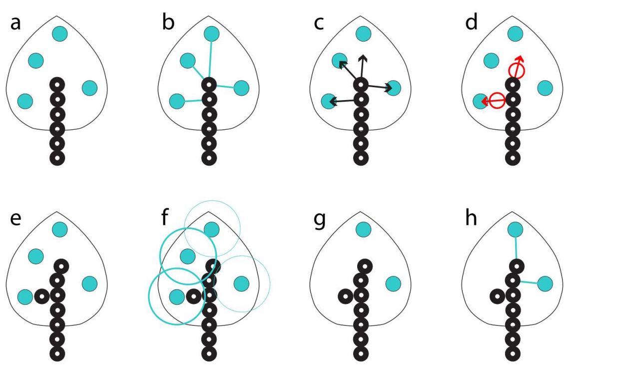


Fig. 3. The mechanism through which the space colonization algorithm expands. **A** presents the initial state of the system. Should the distance between a given node and an attractor become smaller than its radius of influence, **B** follows, representing the process after said relationship is formed. Shown in **C**, is the fact each node can have multiple vectors as it can be influenced by multiple attractors. These vectors are then added and the result is normalized, as seen in **D**, representing the growth direction, resulting in **E**. **F** show the attractor removal process, in which the minimum deletion distance is used to identify if any nodes exist within it, resulting in the removal of affected attractors. With the attractors removed, **G**, the algorithm begins anew, as seen in **H**. Source [4]

A. Properties and Traits

One of the core advantages of space colonization as a model for lightning, is that it was built to simulate natural growth. As it stands, in nature, all that exists must compete for space with those that share similar traits, a space which is necessary for something to further grow, be it flora or fauna. It stands then that structures resulting from a method employing this algorithm as a basis exhibit very natural shapes, making SC well-suited for our goal. Secondly, the manner in which it was conceptualized allows for extensive parametrization, not only that, it facilitates modification of the algorithm down to its very core behavior.

What follows from that mechanism of action are a number of entangled parameters and variables which can be used to achieve an emulation of a given piece of flora. The number of attractors and the distance at which their deletion is triggered have a tremendous influence on the shape of the tree and its branches as well as their density and own offspring. Concretely, the amount of attractors available directly influences the frequency and length of branches.

As for the deletion radius, larger numbers result in sparser structures. This is due to the decreased lifespan of each individual attractor resulting in less chances for a new node to be influenced by an attractor that had been previously reached, shortening the overall width and length of the structure.

Lastly, envelope shape, the volume in which these attractors are distributed, is directly tied to the shape of each structure. This degree of influence is directly tied to the attraction and deletion radii chosen at a given time as the algorithm is programmed to iterate until no new attractor is found within effective range.

IV. Methodology

The formation of a Lightning Bolt can be separated into three distinct phases, as seen in Fig. 4: stepped leader, return stroke and dart leaders. The stepped leader phase encompasses the formation of the bolt itself in which an imperceptible stream of negatively charged particles descends from the clouds, extending tendrils which propagate outwards. These 'feelers', as they are called, search for the path of least resistance, allowing the bolt to expand into several branches though only one results in a finished stroke. Once one of these feelers touches the ground, a return stroke manifests towards the cloud itself, what we associate with the striking of lightning and subsequently, the iconic thunder. From hereon, it is often that dart leaders travel downwards, discharging any excess negative charge to achieve equilibrium, resulting in a strobing effect.

A complete overview of this process is essential in understanding why lightning exists, how it forms and, crucially, how to best emulate them. Albeit normally imperceptible, there are times in which the stepped leader and its feelers are visible to the naked eye. Similarly, there are times in which the strobing behavior is easily observed, something that we wished to capture with the manner in which we rendered lightning. Furthermore, understanding the process behind a stepped leader's descent granted us some insight into how the space colonization algorithm could be molded to better fit the desired outcome.

The more natural interpretation for the propagation of feelers during the stepped leader phase is that of a path-finding algorithm. This is seen in Yun's [9] work and the manner in which the DBM operates, however there are other ways in which this process could be interpreted: particle density, competition for space and interference. These three concepts are emulated in SC, with attractors serving as positive charges competing for space and aggregating with a given density, interfering with one another via competing attraction vectors.

A. Reproducing the Main Channel

Our first focus was solely the reproduction of the main channel, the stepped leader, without worrying about emulating any of the surrounding branches.

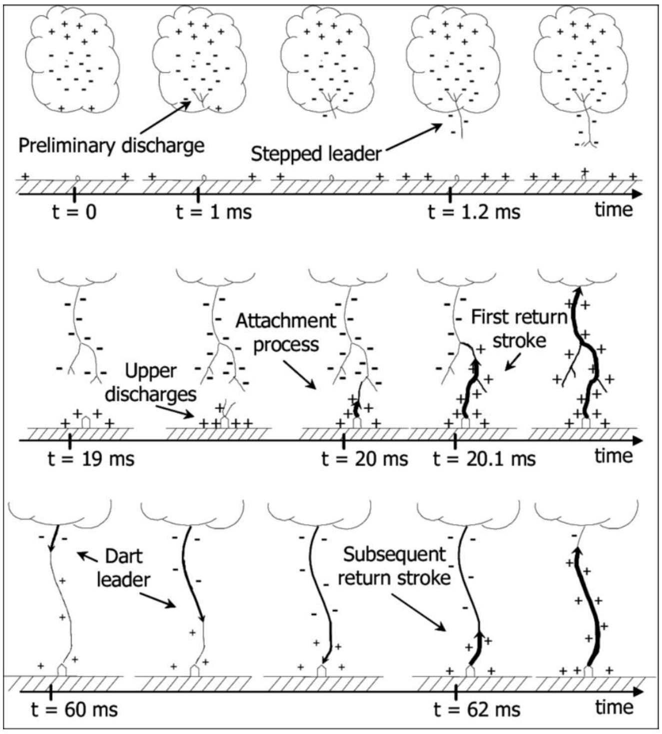


Fig. 4. The process by which a lightning bolt forms [11].

The main channel is built using a modified version of the standard SC algorithm. Firstly, we turned the ability to grow into a trait available only for the most recent node, effectively removing the algorithm's natural ability to produce branches.

Considering a beginning and an end point, the goal is to propagate from a starting point and build the main channel iteratively until the end point is reached. This is another departure from the original SC algorithm. Rather than waiting for all available attractors to be consumed, the algorithm checks for the existence of a specific attractor representing the endpoint. Once this attractor is reached, that is to say, once that specific attractor is consumed, that given iteration of the algorithm ceases.

Attractors are randomly placed inside an envelope volume containing these two points. Arbitrary shapes can be considered for the envelope. For illustrative purposes consider an oriented cylinder where the center of each cap is one such point. The manner in which nodes and attractors interact remains largely unchanged.

Due to the way the algorithm was conceived, any structures made with the original algorithm tends to resemble some form of flora. In order to restrict any behavior that would be considered erratic when considering lightning simulation, such as looping directions and reversing strokes, a method briefly mentioned in Runions et al.'s 2007 article is used [3]. A predetermined weight in order to simulate the elasticity of bark or weight of the tree's boughs. This direction, pointing from the very first node towards an end goal position, served as a way to direct the bolt's propagation. Furthermore, by assigning it a given weight, it was possible to tune the influence of this pre-calculated direction to a fine degree, affecting the overall straightness of the resulting bolt, see in Fig. 5.

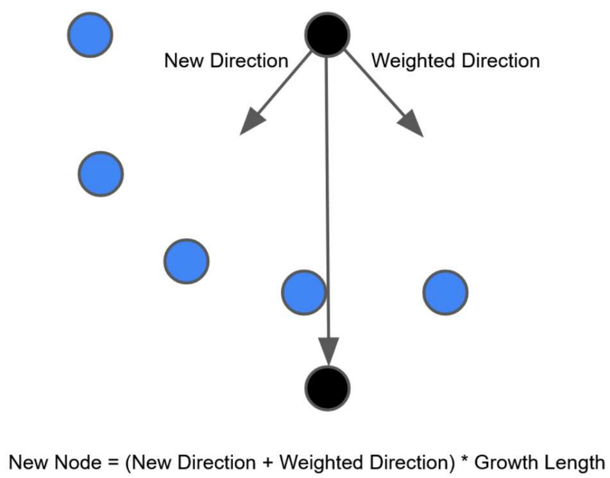


Fig. 5. Weighted growth can be used to influence the manner in which nodes propagate. In this case, half of the growth direction is dictated by normal calculation while the other half is dictated by the weighted direction.

Borrowing from the concept of waypoints explored by Yun et al. [9], a chain of end points can be defined, allowing further path customization. In such a scenario, the algorithm is applied sequentially to each segment, i.e., the algorithm will take the last grown node as the first node in a new iteration and begin propagating towards the next endpoint.

This approach allowed us to introduce the concept of finality to the algorithm, of reaching a goal such as the floor or a lightning arrester, while also providing the flexibility of a waypoint system which lets the user construct a path for their lightning to traverse, an example can be seen in Fig. 6.

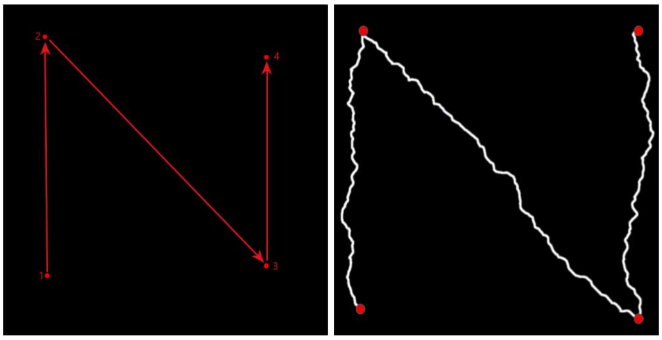


Fig. 6. Waypoints can be used to dictate the path which a lightning bolt takes, drawing out shapes or directing it towards elements.

The algorithm, as described above, can face a situation where there are no attractors available to influence a node, hence prematurely stopping the algorithm. To circumvent this issue a fallback approach was implemented, based in Reed and Wyvill's [5], where the direction of growth is the line linking the last grown node and the end point, rotated by an average of

16° at each step. Employing this method allows the algorithm to propagate until either the new node is within the influence of an attractor or the end attractor is reached.

B. Generating Branches

Through observation of images and videos of lightning, bolts primarily form with three types of channels: primary, secondary and tertiary. The primary branch, resulting from the stepped leader which reaches the ground, was already covered, leaving this section to focus on the secondary and tertiary channels, the branches of the lightning bolt.

In order to finely control the amount of branches produced and how they manifested, we maintained the inability for them to be produced and progressed naturally during the algorithm's execution. The solution found was instead conjured sampling from a probability distribution, employed each time a new node is created. This method allows us to flag nodes as capable or incapable of producing branches, serving as the root node for said structure, while also promoting more granular manipulation of branch density when compared to its SC counterpart. After a root node is identified, the next step in branch creation lies with settling a suitable endpoint.

To allow for an easier tweaking of parameters, spherical coordinates are employed, a decision which also simplified the randomization process tremendously. The azimuthal angle is determined via sampling from a uniform probability distribution function without any restraints. The inclination is decided as a function of the distance from the root node to the main channel's endpoint.

Based on observation of real footage and imagery, it was found that, in general, the angle between the parent and child channel is relatively wide initially, sometimes even growing upwards, becoming narrower the closer the main channel is to the ground. As such, by default, we settled the lower limit for the inclination at 30° while the upper limit may vary up to 135°. Nevertheless, these are just parameters that can be tweaked by the user.

The final parameter, length, is chosen as a random percentage of a user defined length. The usage of probability distribution functions allows for an increase or decrease in disparity between branch lengths.

With an ending selected and a root node already available, the ensuing branch is formed by employing the same method outlined for the main stroke with tertiary branches being the result of a further reduced probability function. These are then also created in the same manner, though their ability to produce further branches is disabled. An example of a result with the three types of branches can be seen in Fig. 7.

Finally, in the event a user defines a secondary branch, parameters such as the root node, the endpoint, the envelope type and width, charge density, the ability to sprout tertiary branches and their density can all be manually defined separately from the natural branches.

Fig. 7. A lightning bolt generated via our method, exhibiting all three types of channel.

C. Rendering

When it came to rendering, we focused solely on the lightning bolt and the effects which surrounded it. This includes the animation of the lightning bolt life cycle.

Our first effort began with the iterative rendering of the structure that would become our bolt. To build it line by line, as if a stepped leader or feeler propagating through the air. The first step to achieve the desired result was found in properly sorting the generated nodes, necessary due to the way branches are built during the algorithm's lifetime. Once they are sorted by hierarchical order, in a manner similar to a data tree, the program can calculate the amount of nodes to render at any given point.

To achieve this, we had to determine how long, at default speed, this section of the rendering should take. With that number, we can then use the percentage of time elapsed to gather an equal percentage of the data structure which contains every point to render in that given frame. This results in a slowly growing bolt, as in Fig. 8 which emulates many of the quirks found in slow motion footage.

The next effect to tackle was the bolt's luminosity. The manner in which it glows once it strikes and illuminates all

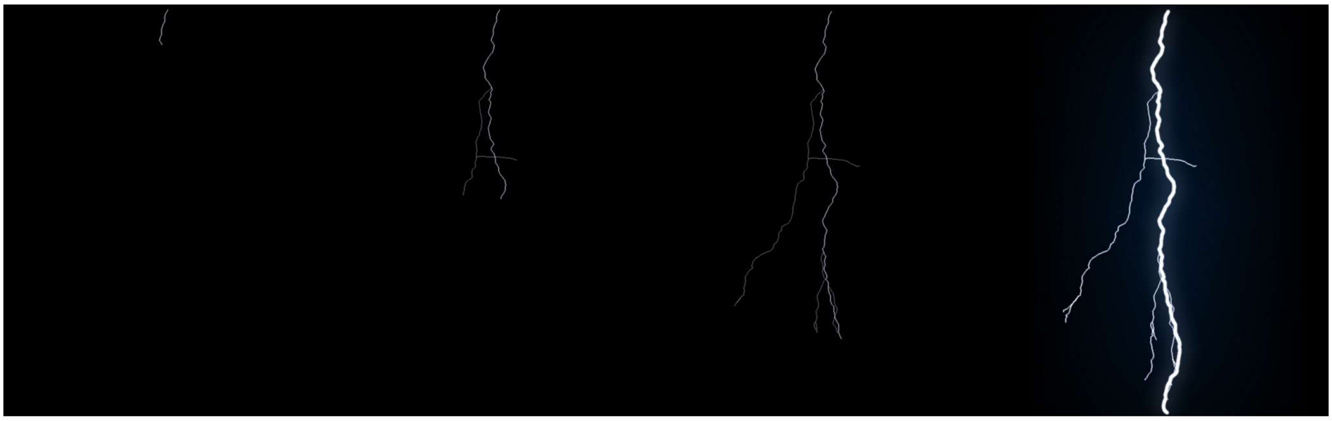


Fig. 8. Snapshots of the animation of the iterative rendering of geometry.

around it. To emulate the former, a dilation shader is applied to the main channel, used to emulate the thickness seen once the bolt hits the ground. Afterwards, a separable Gaussian blur with a small kernel size is applied to the entirety of the bolt, including branches. This helps in concealing artifacts formed by the geometry as well as giving a smoother appearance to the bolt itself.

Lastly, two different blurs are applied. The one applied to the branches is given a slightly larger kernel than the previous one. Combining these two allows for a large contrast in luminosity as it nears the branch structure itself, more accurately reflecting the real life phenomenon. The main stroke, however, is given a much larger kernel to reflect its overwhelming presence.

This however, did not fully represent the luminosity seen in the presence of a lightning bolt. To further enhance the rendering, a mipmap of the final blurred texture was created. Fetching one of the lower levels of this map and overlaying the blurred texture on top of it allows for a soft distribution of color that can be easily interpreted as an overall glow.

The last effect we wished to emulate was the strobing of the lightning bolt, caused by subsequent dart leaders and return strokes as well as the effect seen whenever a bolt entirely dissipates, vanishing from sight. To do this, we altered the blending process by defining it as a function of time.

By dividing the strobing effect into separate waxing and waning phases, we can make the blended image increase or decrease in intensity as time elapses within these phases, making use of the same concept explored when iteratively rendering the geometry. Then, by chaining these phases in pairs, the bolt can be made to strobe a number of times, Fig. 9, until it finally fades in its entirety.

V. Results of Implementation

The results achieved from our implementation reflected our efforts to promote user-freedom and parameterize as much of our method as we possibly could. From the listing of waypoints to the definition of individual branches and their associated parameters, the user is allowed the freedom to design their own lightning bolt but also to change its appearance as they see fit. While not all of the possible user parameters were implemented, such as the individual selection of probability functions, most of them and, most importantly, the ones which would most effect the generated bolt are present and as follows:

- Toggle creation of secondary branches.
- Toggle creation of tertiary branches.
- Adjust the probability of secondary branch creation.
- Adjust the probability of tertiary branch creation.
- Adjust the average and maximum length of secondary branches.
- Adjust the average and maximum length of tertiary branches.
- Select envelope type for charge distribution.
- Adjust maximum envelope radius for charge distribution.
- Adjust distance between nodes.
- Adjust the influence of the weighted direction.
- Adjust the color of the lightning bolt.

Most of all, however, we wished to see just how well the results of our method would stack up when directly compared to real lightning bolts we attempted to emulate. To achieve this, we made full use of the tools at our disposal, including the definition of waypoints and branches and the fine-tuning of individual parameters until the resulting simulation was sufficiently close to the real phenomenon. Figures 10-14 show the results we achieved by attempting to emulate different real life lightning. See the acknowledgments section for the source of the real imagery used in this comparison.

From a graphical point of view we believe to have achieved results which are similar in quality to Kim and Lin [2] (see Fig. 2 for a result in [2]), and closely emulate real lightning strikes. Regarding customization, our method offers users a wide range of control, allowing, for instance, for separate configuration for primary and secondary branches.

VI. Conclusion

Throughout this article we detailed a procedural algorithm that can be used for the generation and rendering of lightning.

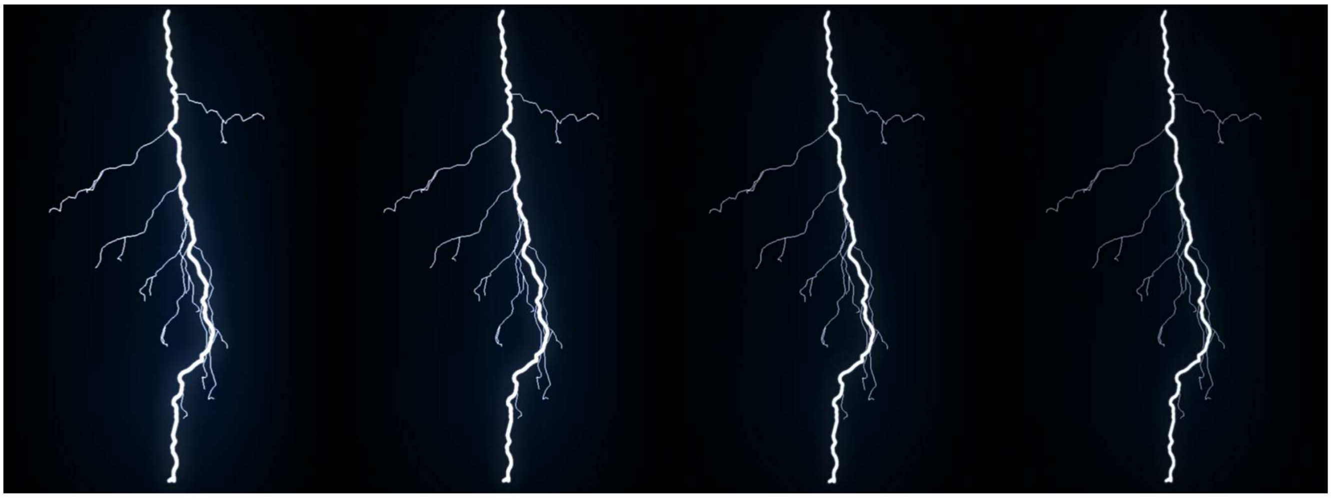


Fig. 9. Snapshots of the animation of a generated lightning bolt waning.

The geometric procedure is based on the Space Colonization algorithm, and it is fully heuristic. Rendering goes one step further than previous attempts by providing a visualization of the bolt's life cycle.

We believe that this algorithm could already be used in real-time media to great effect, allowing for a game engine, or the users themselves, to fully control and design their lightning as they see fit. Furthermore, the amount of user parameters that were made available makes this method fit even for artistic purposes.

The way the light emanating from a bolt interacts with a 3D scene, including clouds, is a promising avenue for future work, complementing this work, and allowing for a complete solution for lightning simulation in 3D real-time engines.

Further experimentation could be done in an effort to achieve plausible simulation of rare and esoteric events related to lightning such as jets, elves .

A final avenue that might be worth pursuing in the future is an attempt to integrate this manner of procedural generation with the DBM. The DBM could be used to dictate the distribution of the attractors, amongst other settings, causing the creation and definition of a bolt to be based on meteorological conditions, while keeping the SC based approach as defined.

ACKNOWLEDGEMENTS

This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the RD Units Project Scope: UIDB/00319/2020.

Lighting images from real scenarios taken from:

1) https://www.flickr.com/photos/eclipsx/5751977702
2) https://www.flickr.com/photos/veggiefrog/2573076436
3) https://www.flickr.com/photos/hunty66/390350345
4) https://www.flickr.com/photos/snowpeak/3761397491
5) https://www.flickr.com/photos/snowpeak/3761397565

REFERENCES

[1] N. NSSL, "Lightning basics," 2021. [Online]. Available: https://www.nssl.noaa.gov/education/svrwx101/lightning/

[2] T. Kim and M. C. Lin, "Fast animation of lightning using an adaptive mesh," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 2, pp. 390–402, March 2007.

[3] A. Runions, B. Lane, and P. Prusinkiewicz, "Modeling trees with a space colonization algorithm." *NPH*, vol. 7, no. 63-70, p. 6, 2007.

[4] A. Runions, M. Fuhrer, B. Lane, P. Federl, A.-G. Rolland-Lagan, and P. Prusinkiewicz, "Modeling and visualization of leaf venation patterns," in *ACM SIGGRAPH 2005 Papers*, ser. SIGGRAPH '05. New York, NY, USA: Association for Computing Machinery, 2005, p. 702–711. [Online]. Available: https://doi.org/10.1145/1186822.1073251

[5] T. Reed and B. Wyvill, "Visual simulation of lightning," in *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '94. New York, NY, USA: Association for Computing Machinery, 1994, p. 359–364. [Online]. Available: https://doi.org/10.1145/192161.192256

[6] B. Sosorbaram, T. Fujimoto, K. Muraoka, and N. Chiba, "Visual simulation of lightning taking into account cloud growth," in *Proceedings. Computer Graphics International 2001*, July 2001, pp. 89–95.

[7] T. Kim and M. Lin, "Physically based animation and rendering of lightning," in *12th Pacific Conference on Computer Graphics and Applications, 2004. PG 2004. Proceedings.*, Oct 2004, pp. 267–275.

[8] J. L. Bryan, S. K. Semwal, C. E. Chow, and T. Chamillard, "Fast simulation of lightning for 3d games," Ph.D. dissertation, University of Colorado at Colorado Springs, 2005.

[9] J. Yun, M. Son, B. Choi, T. Kim, and S.-E. Yoon, "Physically inspired, interactive lightning generation," *Computer Animation and Virtual Worlds*, vol. 28, no. 3-4, p. e1760, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cav.1760

[10] Y. Dobashi, T. Yamamoto, and T. Nishita, "Efficient rendering of lightning taking into account scattering effects due to clouds and atmospheric particles," in *Proceedings Ninth Pacific Conference on Computer Graphics and Applications. Pacific Graphics 2001*, Oct 2001, pp. 390–399.

[11] J. Bermudez, F. Rachidi, M. Rubinstein, W. Janischewskyj, V. Shostak, D. Pavanello, J. S. Chang, A. Hussein, C. Nucci, and M. Paolone, "Far-field-current relationship based on the tl model for lightning return strokes to elevated strike objects," *IEEE Transactions on Electromagnetic Compatibility*, vol. 47, no. 1, pp. 146–159, Feb 2005.

Fig. 10. Our lightning vs. Real Phenomena (1).

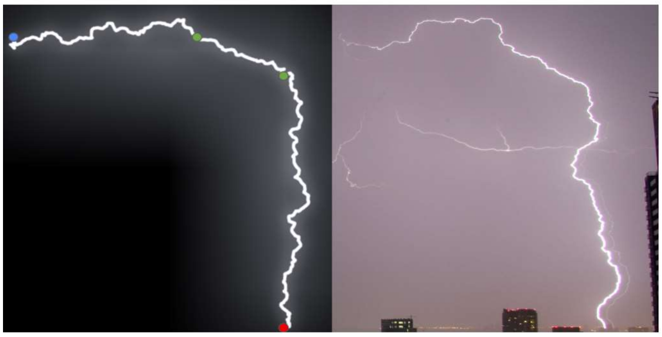


Fig. 11. Our lightning vs. Real Phenomena (2).

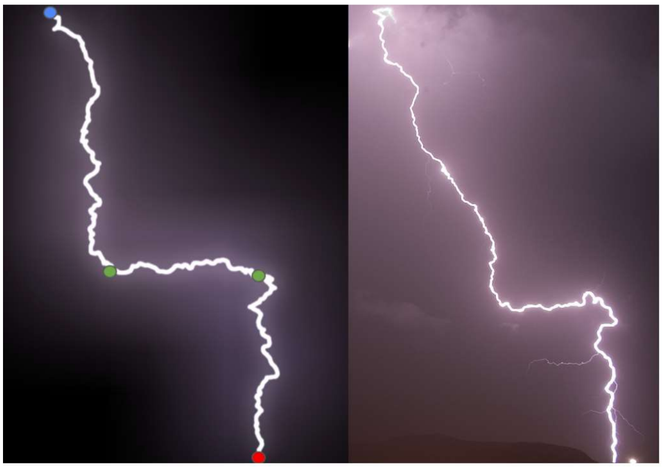


Fig. 12. Our lightning vs. Real Phenomena (3).

Fig. 13. Our lightning vs. Real Phenomena (4).

Fig. 14. Our lightning vs. Real Phenomena (5).

Character Simulation Using Imitation Learning With Game Engine Physics

João Rodrigues
NOVA LINCS
NOVA School of Science and Technology
Caparica, Portugal
jfpe.rodrigues@campus.fct.unl.pt

Rui Nóbrega
NOVA LINCS
NOVA School of Science and Technology
Caparica, Portugal
rui.nobrega@fct.unl.pt

***Abstract*—Creating visual 3D sensing characters that interact with AI peers and virtual environments can be a difficult task for those with less experience in using learning algorithms or creating visual environments to execute an agent-based simulation. In this paper, the use of game engines as a tool to create and execute graphic simulations with 3D sensing characters is being explored with plugins such as ML-Agents for the Unity3D game engine. This allows the simulation of agents using off-the-shelf algorithms and using the game engine's motor for the visualizations of these agents. We explore the use of these tools to create visual bots for games, and teach them how to play the game until they reach a level where they can serve as adversaries for real-life players in interactive games.**

***Index Terms*—Interaction, Games, ML-Agents, Imitation Learning**

I. Introduction

Games have always used interactive characters and Non-Playable Characters (NPCs) to make the game more interesting, appealing and relatable. Agent and Character Simulation has been used over time for many different purposes [1], going from scientific research, such as predicting the spread of pandemics [2], to leisure [3] with games having NPCs to add more interactivity or make the game harder. Using these simulations requires a vast knowledge of the different AI, machine learning, and deep learning algorithms needed for these simulations. Additionally, many of these simulations also need graphic environments to show the results with rendering techniques and physics simulations. All these techniques require specialized knowledge that for a computer graphics developer may not be available, specially in the AI and Machine Learning fields.

Another factor that is also present in agent simulation, is the need for create sensing characters. Many of the Agent-Based Modelling (ABM) and Simulation scenarios require characters that interact with the environments or with their peers. To achieve this, the creation of characters that can sense their surroundings by simulating vision, touch, and distance among other factors is needed. Creating these senses from scratch also needs advanced programming knowledge such as ray tracing for vision and distance, or physics for touch.

All these needs constitute a problem for those who want to create character simulations but do not have the technical expertise to develop all the aforementioned technologies. To solve the problems created by the need for a graphical environment with sensing characters, it can be interesting to use game engines to visualize the environment and simulate the senses. Being in constant development to facilitate the process of creating games, game engines seem to be a good tool to create and train characters and environments since they already have render and physics engines that implement virtual environment rules sush as ray cast distance, gravity or colisions.

Besides the use of game engines, over the years many tools have been created to facilitate the creation and execution of these simulations by providing utilities that facilitate the testing and sharing of different machine and deep learning algorithms such as OpenAI Gym [4] and PettingZoo [5]. One tool that has special interest when considering the use of game engines is the ML-Agents plugin [3], since it was created specifically to integrate with the Unity3D game engine and aims to ease the creation and learning of game bots.

This last tool, the ML-Agents, contains a utility that can be helpful in teaching bots how to play which is the demonstration recorder. This tool allows recording the steps of a human playing in the same environment as the bots, collect the observations the bot would receive during the playtime, and generate a demo file that can be later used to teach bots how to play.

In this paper we propose to create a set of game scenarios and use them to teach bots how to play. More than this, we will study the use of human play recordings as a way of teaching the bots how to play. Afterwards we aim to compare if this method shows results similar to human level of play.

To have a better grasp of how game engines can be used for a better graphical interface for agent-based simulation we are going to look at some of the related work done in these areas to understand better a few of the concepts of agent simulation, game engines, and some libraries that can be used as a tool to connect both. After this, the idea and concepts behind the work done are going to be exposed for a better understanding of how ML-Agents works, and how we can use it to improve learning bots' performance.

II. Related Work

A game engine's main use is to produce games by rendering its graphics, producing music and sound effects, and allowing

978-8-3503-4573-57/22/$31.00 ©2022 IEEE

external manipulation of the scene by reading inputs from an input device. Usually, a game engine is made up of Rendering Engine, Animation Engine, Physics Engine, Artificial Intelligence Engine, Network Engine, 3D Sound Engine, and a Map Editor [6].

Although the main purpose of a game engine is, as its name says, to create games, its features allow for more uses such as animations and simulations. The render, animation, and physics engines can facilitate the production of animation. With these engines one can just pick up a scene to animate, code the intended movements, and then just play out the result which will follow physics without having to animate frame by frame every movement, and collision [6]. In the same line as animation, these tools can be used to produce simulations.

What makes the use of a game engine alluring for animations and simulations is how it allows a user to view these creations, without having to make low-level code to render and animate the scene since the game engine deals with this internally [7]. For agent simulation, it interests the most how a game engine allows manipulating the motion of the agent's models and how to create a simulation environment for the agents to play out simply by adding game objects to a scene, and manipulating its properties with no need for extensive lines of code describing the shape, physics, and motion of every object in the simulation. Some of the works done in using game engines for simulations are [8]–[10].

A. Graphical Agent Simulation

Graphical Agent-Based Modeling and simulation is a paradigm in which simulated humans, animals, or other forms of beings are modeled as agents that interact with their peers as well as their environment [11]. In these agent-based simulations, sometimes called multi-agent simulation, the environment plays a crucial role since it influences all the agents and their interactions and therefore must be carefully taken into account.

In order to create an agent-based simulation, four elements have to be taken into consideration: the set of agents, set of interactions, the environment and the simulation infrastructure [12].

Self-play reinforcement learning is when agents learn by exploring and playing with themselves. It has seen success when applied in many game scenarios, however, the process for self-play learning is unstable and more sample-inefficient than general reinforcement learning, especially when used in imperfect information games [13]. Multi-agents auto-curricula have been used to solve the various type of multiplayer games both in classic discrete games, such as Backgammon [14] and Chess [15], and continuous real-time games like Dota and Starcraft. Illustrative examples of works done with agents simulation and self-play reinforcement learning using visual scenarios are given by [13], [16]–[18].

B. Agent Simulation Libraries

In order to create sensing agents for game engine environments, it is crucial to find a library that allows simulating the agent's actions without the need to write complex algorithms. Luckily, nowadays, there is an increasing number of these that are open source and allow for indiscriminate integration in any kind of simulation as long as it complies with the library specifications.

Unity Machine Learning Agents [3] (ML-Agents) is a toolkit that allows games and simulation to serve as environments for training intelligent agents. This library is specific for Unity's 3D game engine and provides implementations of state-of-the-art that enable game developers and researchers to easily train agents for 2D, 3D, and VR/AR games. With ML-Agents, agents can be trained using reinforcement learning, imitation learning, neuroevolution, and some other methods with the provided simple-to-use Python API.

III. Simulation and Training Workflow

As mentioned before, the goal is to get records of humans playing and using the obtained data to train bots.

In the context of this paper these records will be called demonstrations. A demonstration is a data structure that saves the observations humans obtained at each moment, and maps it to the actions they took, so that it is possible for bots to decide their actions, by finding the actions in the demonstrations that have similar observations to what they have at that point.

To better understand how the demonstration records can be used to improve the learning speed of bots, first it is good to understand how to create scenarios with learning bots and how ML-Agents teach them. Creating the scenarios is quite simple for those with game development experience. The first step is to envision the desired interactive scene, then, build that scene in Unity3D by placing different game objects and scripts that control the desired interactions.

After this, each bot connects with the ML-Agents interface, its behaviour takes into consideration the actions that it receives from the system and the observations from the environment it interacts with. Each bot sends a set of variables and values that constitute an observation, and a hit history from ray-traces from the bot to the world.

Outside of Unity3D, the ML-Agents library is executed using a configuration file that defines: run id, flags and the specific training algorithm. The training algorithms can be split into two categories, the Reinforcement Learning algorithms, and the Imitation Algorithms. For RL algorithms, ML-

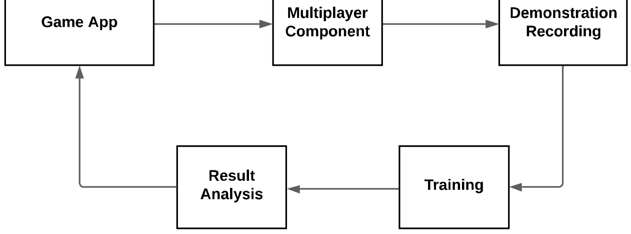


Fig. 1. Demonstration Record process flowchart

Agents offers Proximal Policy Optimization (PPO) and Soft Actor-Critic (SAC) for single agent scenarios, Multi-Agent Posthumous Credit Assignment (MA-POCA) for multi agent cooperative/competitive scenarios, and self-play which can be used in both situations.

For the imitation algorithms, the ML-Agents library[1] offers Behavioral Cloning and Generative Adversarial Imitation Learning (GAIL) which use the demonstration records to perform imitation learning. These algorithms can be used together as a better way to reinforce the imitation learning, and can also be used with RL Algorithms as a way to learn past the demonstrations and use them only as a base for learning.

Now that a broad look at how ML-Agents can be used to create self-learning bots in Unity3D was seen, we can focus on the question of how can demonstration records help bots learn. First, let's look at what exactly are these demonstration records. With the ML-Agents package in unity, it is possible to attach a demonstration recorder to an agent. When attached, this recorder will save a data structure with all the actions the humans who are playing did mapped to all the variables and ray trace hits provided as observations at each action point. This forms a demonstration for imitation learning because the agents can then use these data structures to decide what actions to take, by comparing their observations with the observations present on the demo file, and see which one looks the most similar and perform the action took with that observation on the demo. This means that to generate demos that can be used for learning, we can have human users playing the scenarios and complete the tasks to serve as inspiration for the bots. This is possible because the ML-Agents interface allows an heuristic function which can be used to create human controls for the bots.

The idea with this is that for more complex scenarios that take very long for bots to learn, and that the tuning of the configuration file becomes harder, we can take advantage of these demonstrations to accelerate the process and create usable bots faster. For this, first we have to create a game that is designed for both the agents and users to play it out. We also need to define the observations and rewards the agent gets. If this game involves multiple agents simultaneously it has to support multiplayer so multiple users can also play it out simultaneously. After all this is done, we proceed to have users play the game to record the demonstrations. The more users and the more time the better. Upon collecting all the demonstrations, the training of the bots can be started. Then the final result of the bots can be analyzed, and if it's still not up to expectations, the process can be repeated to collect more demonstrations with the possibility of using the current bots as adversaries to the users. A flowchart of this process can be seen in Figure 1.

To test this idea of using demonstration records to improve the learning speed of bots we decided the use a game of capture the flag in a $3vs3$ player competitive environment as seen in Figure 2. This game was chosen because we wanted a simultaneously competitive/cooperative game, with a decent amount of complexity, so that the performance level differences are easily distinguishable, but also it's not extremely hard for bots and humans to learn how to play the game.

In this game, 2 teams (team Blue and team White) compete to try and capture the enemy flag, and take it all the way back to their teams field location. Each iteration of capturing a flag and retrieving it to the teams field is, for training purposes, considered a round. In addition to this, the field is spread with a few walls to create obstacles for the players, and balls the players can catch and throw to attempt to hit other players. If a players gets with while holding a flag, they will drop it and be stunned for 3 seconds. While a flag is dropped out of place, both teams can attempt to catch it, to either recapture it or make it go back to spawn depending if it is the enemy or ally flag respectively. An image illustrating the described game can be seen in Figure 2. This game supports that multiple users play at the same time, meaning that it is necessary to implement a multiplayer component for this game. This is needed so that the human users can play and record demonstrations at a competitive level by playing other humans.

IV. Preliminary Results

For this initial experiment, 3 play sessions with human users were played to record demonstrations. During these play sessions a total of 10 demonstrations were obtained. After the recording sessions the bots were trained several times using different numbers of demos.

[1]ML-Agents, https://github.com/Unity-Technologies/ML-Agents/blob/main/docs/Training-ML-Agents.md

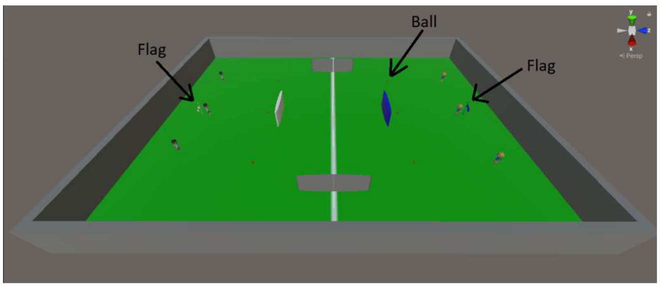


Fig. 2. Capture the Flag Scene where players have to capture enemy flag and retrieve it to their own field

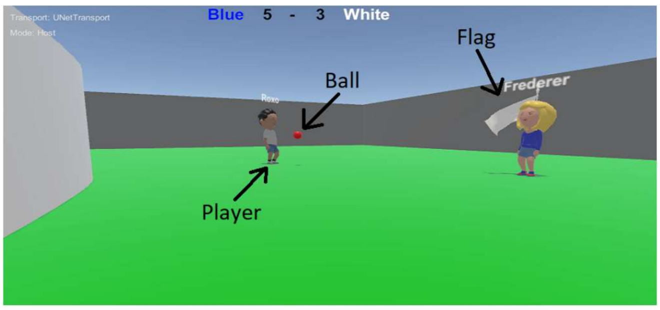


Fig. 3. Game-play screenshot displaying all Network Objects

In these initial results, it was possible to see that the bots got better as more demonstrations were used. The amount of time needed to complete each round of the game dropped from an average of 517 seconds with bots trained using 1 demo to 165 seconds when 10 demos were used. The number of draws (when neither team is able to win after 1000 seconds) also lowers as more demos are used, having a ratio of 20% of games drawn with 1 demo and only 7% of games drawn with 10 demos.

During the recording sessions, it was noted that the team playing the left side of the field (Blue team) won overall 68.9% of all rounds played, having more rounds won in all three of the play sessions. This dominance was also translated to the bots and had increased strength as more demonstrations were used. When only 1 demo was used to train the bots, the blue team won 57% of games having a slight superiority, but when the 10 demonstrations were used to train the bots, the blue team won 81% having a huge superiority, even bigger than the one the human blue team had.

Looking back at times per round, we analyzed how quickly each of the teams finished the game in their winning rounds. Against expectations, on their wins the White team had an average of 50 seconds and were able to finish the game, 50 seconds faster than the Blue team which had an average time on wins of 100 seconds. A first explanation for this event is that the White team is only able to win on ideal circumstances, while the blue team can find ways to win even if the game drags a bit longer.

V. Conclusions

In this paper, we present a method to create game engine self-learning bots in Unity3D that use imitation algorithms. This technique was achieved using the described ML-Agents plugin. Then we focused on the hypothesis that imitation learning methods using demonstration records can be a good way to create bots for games with complex environments. For this, we created a $3vs3$ game of capture the flag and had multiple human users record demonstrations simultaneously.

With the results obtained we conclude that imitation learning can be a good way to train game-ready bots and that further investigation is necessary. Firstly, the recording of many more demonstration recordings is needed so that it is possible to analyze the level of the bots when a big sample of demonstrations is available and compare it to the human level by games between humans and bots. Following the increase in the sample size, it will also be interesting to see the results if a more balanced set of results is used and see if then both teams have a similar win ratio or if one team keeps dominating regardless. Lastly, it is left to expand this method of teaching to other games and see the results and the level of trained bots in games with varying levels of complexity and see what variables tend to facilitate or difficult the bots learning process through imitation.

Acknowledgment

This work was supported by NOVA.ID.FCT/NOVA LINCS (UIDB/04516/2020)

References

[1] C. Macal and M. North, "Agent-based modeling and simulation," 12 2009.
[2] K. M. Khalil, M. Abdel-Aziz, T. T. Nazmy, and A.-B. M. Salem, *An Agent-Based Modeling for Pandemic Influenza in Egypt*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 205–218. [Online]. Available: https://doi.org/10.1007/978-3-642-25755-1_11
[3] A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, and D. Lange, "Unity: A general platform for intelligent agents," 9 2018. [Online]. Available: http://arxiv.org/abs/1809.02627
[4] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," 6 2016. [Online]. Available: http://arxiv.org/abs/1606.01540
[5] J. K. Terry, B. Black, N. Grammel, M. Jayakumar, A. Hari, R. Sullivan, L. Santos, R. Perez, C. Horsch, C. Dieffendahl, N. L. Williams, Y. Lokesh, and P. Ravi, "Pettingzoo: Gym for multi-agent reinforcement learning," *35th Conference on Neural Information Processing Systems*, 9 2020. [Online]. Available: http://arxiv.org/abs/2009.14471
[6] S. S. Noh, S. D. Hong, and J. W. Park, "Using a game engine technique to produce 3d entertainment contents," in *16th International Conference on Artificial Reality and Telexistence–Workshops (ICAT'06)*, 2006, pp. 246–251.
[7] A. Bak and M. Wojciechowska, "Using the game engine in the animation production process," pp. 209–220, 2020.
[8] C. Bartneck, M. Soucy, K. Fleuret, and E. B. Sandoval, "The robot engine - making the unity 3d game engine work for hri," *24th IEEE International Symposium on Robot and Human Interactive Communication*, 2015. [Online]. Available: https://www.openrobots.org/wiki/morse/
[9] M. Pasternak, N. Kahani, M. Bagherzadeh, J. Dingel, and J. R. Cordy, "Simgen: A tool for generating simulations and visualizations of embedded systems on the unity game engine," in *MODELS 2018*, ser. MODELS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 42–46. [Online]. Available: https://doi.org/10.1145/3270112.3270135
[10] A. Carlos, A. Mól, C. Alexandre, F. Jorge, and P. M. Couto, "Using a game engine for vr simulations in evacuation planning," *IEEE Computer Society*, 2008. [Online]. Available: http://udn.
[11] F. Klügl and A. L. C. Bazzan, "Agent-based modeling and simulation," *AI Magazine*, vol. 33, no. 3, p. 29, Sep. 2012. [Online]. Available: https://ojs.aaai.org/index.php/aimagazine/article/view/2425
[12] F. Klügl, M. Fehler, and R. Herrler, "About the role of the environment in multi-agent simulations," vol. 3374. Springer Verlag, 2005, pp. 127–149.
[13] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," 2016. [Online]. Available: https://arxiv.org/abs/1606.01540
[14] G. Tesauro, "Temporal difference learning and td-gammon," *Commun. ACM*, vol. 38, no. 3, p. 58–68, mar 1995. [Online]. Available: https://doi.org/10.1145/203330.203343
[15] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. P. Lillicrap, K. Simonyan, and D. Hassabis, "Mastering chess and shogi by self-play with a general reinforcement learning algorithm," *CoRR*, vol. abs/1712.01815, 2017. [Online]. Available: http://arxiv.org/abs/1712.01815
[16] J. Yang, A. Nakhaei, D. Isele, K. Fujimura, and H. Zha, "Cm3: Cooperative multi-goal multi-stage multi-agent reinforcement learning," *ICLR 2020*, 2020.
[17] OpenAI, "Openai five," https://blog.openai.com/openai-five/, 2018.
[18] O. Vinyals, I. Babuschkin, J. Chung, M. Mathieu, M. Jaderberg, and W. Czarnecki, "Alphastar: Mastering the real-time strategy game starcraft ii." [Online]. Available: https://www.deepmind.com/blog/alphastar-mastering-the-real-time-strategy-game-starcraft-ii

Exploring Player Adaptivity through Level Design: A Platformer Case Study

Pedro Esteves
DEI, Faculty of Engineering, University of Porto
Porto, Portugal
0000-0002-9128-543X

João Jacob
Engmindera, Software Engineering, S.A and LIACC, University of Porto
Porto, Portugal
0000-0003-2078-3375

Rui Rodrigues
DEI, Faculty of Engineering, University of Porto and INESC TEC
Porto, Portugal
0000-0003-4883-1375

***Abstract*—Creating a game that engages everyone is not trivial and, ultimately, impossible. However, generating content adapted to players could improve their experience, thus maximizing the potential player base. With adaptivity, games can be designed to act differently according to several actions without developing different versions. Although it can help the designers, creating a generic approach that adapts content for every game and genre is not trivial. To tackle this problem, this work proposes the exploration of player adaptivity in level design, following three stages: (i) selection of a genre and a game; (ii) study to correlate players' profiles, preferences, and actions with single elements; (iii) study, based on the previous, to evaluate the impact of generating and adapting game levels. Using Infinite Mario Bros, the first results show that correlating players' profiles with game elements is not possible. However, preferences for specific levels' configurations were spotted and used for the second study's adaptation, which did not impact the players' experience. In the future, other analyses of the study's data should be performed, and the methodology should be applied to other games, ensuring similar results are obtained. Lastly, another study should explore correlations between players' profiles and groups of game elements.**

***Index Terms*—Game Adaptivity, Procedural Content Generation, Level Design, Player Profiling, Player Personality**

I. Introduction

People play games because of experiences, so game designers need to know the preferences of their audience. Their job can be difficult when dealing with games with **static elements**, i.e., games where the content is predefined by designers, maintaining it testable and controllable [1]. Alternatively, there are **adaptive games**, i.e., games with some adaptation to players, either through content creation adapted to their profiles or in adjusting parameters or content to these profiles [1]. With this approach, game designers can improve the players' experience and increase the target audience. However, creating these games can be very time-consuming, increasing the planning and development time and the game's budget. Since each person is unique, designers need a way to group players, which can be achieved through **player profiling**, "the categorization of players based on static information that does not alter during gameplay" [2, p. 2], which can include, for example, age range or personality, and could be related to certain in-game behaviors or preferences for elements. However, not enough research relates these behaviors with the said personality traits.

Since there are many game genres, this work focused only on platform single-player games to avoid dealing with multiple players' preferences. Several elements can be adapted in each genre, so the work focused only on level design.

Two research questions were posed to guide this research:

Q1: Do players' profiles correlate with their preference for specific game elements?

Q2: Does automatically generating and adapting levels to the overall preferences of players impact their experience?

The work contributes with: (i) a meta-analysis on platform levels' elements and ways to generate or adapt them, (ii) a new methodology to correlate players' profiles, actions, and preferences with game elements and evaluate the impact of generating and adapting game levels, (iii) a Super Mario level generator, adapted from a *JavaScript* port of Infinite Mario Bros [3] that allows creating and importing JSON files describing the levels, which store only their relevant information, (iv) a gameplay metrics collector, capable of registering every game element by ID and collecting users' data, allowing game designers to improve the players' experience later, and (v) two datasets [4], [5] allowing other researchers to iterate over them and investigate other correlations with distinct gameplay data.

Lastly, this document is divided into several sections. Section III explains the work's methodology, developed to answer Q1 and Q2, and the solution's architecture. Sections IV and V present the two studies conducted during the work. Section VI discusses several testers' comments, how they could influence the results and what could be done to avoid users' mistakes while answering the survey. Finally, some conclusions are presented in section VII, summarizing the work done and showing work that can and should be done in the future.

II. Adaptivity through Level Design: a review

Game Adaptivity has been an increasing topic of research in previous years [1], [6]. Since this work focuses on adapting a game by generating levels according to players' profiles, the research explored works related to Game Adaptivity, Level Design, and Player Modeling and Profiling.

A. Game Adaptivity

Game Adaptivity consists of creating or adjusting content to make games more challenging, unpredictable, and player-

978-8-3503-4573-57/22/$31.00 ©2022 IEEE

oriented [1]. It has two main types: (i) offline, where content is adapted to a model before the gameplay starts, and (ii) online, where content is adjusted during the gameplay, either by creating new content or tweaking some parameters [1]. Both offline and online adaptivity can be influenced by gameplay expectations, learning preferences, or assessment data [1].

Research around Game Adaptivity has increased in previous years, so many works can help follow the approach proposed in section III. The works reviewed [7]–[9] show potential for using *Infinite Mario Bros* [3] as a case study. Yannakakis and Togelius [10] show how to assess the quality of generated content in three ways, which can contribute to the proposed generator. Shu, Liu, and Yannakakis [7] demonstrate how to define fun as a function that can influence the content generator proposed. Shaker et al. [9] present platform levels' features that can be adapted and gameplay events and questionnaires, which can be used to assess players' engagement, challenge, and frustration. Lastly, Sarkar and Cooper [11] show a tendency to incorporate machine learning in game design.

In conclusion, game adaptivity is still an area with much research but has been moved towards other subareas.

B. Player Modeling and Profiling

Works in the Game Adaptivity area often define models of the player to generate or adapt their content, such as modeling players' experience [7]–[9], [12] or their personalities [13]. Player modeling and profiling sometimes are used as synonyms. However, in most works, player profiling is defined as a subset of modeling, in which players are categorized according to information that does not change during gameplay, such as personality, cultural background, or demographic data, as stated by Yannakakis et al. [2].

Hooshyar, Yousefi, and Lim [14] define two approaches to modeling players, theory-driven and data-driven. Yannakakis et al. [2] also establish three main types of inputs to these models, behavioral data, objective data, and game context. This work follows a **data-driven** approach since a map between the input and a player state representation was made. Its model takes behavioral data, and the game context as input since the previous mapping was made according to these data. Considering the approaches for player experience modeling defined by Yannakakis and Togelius [10], it can be stated that the model follows an objective and subjective approach, although it does not model players' experience.

According to the work of Karpinskyj, Zambetta, and Cavedon [15], there was no game personalization application using personality until 2014. Meanwhile, some works were exploring this application [16]. Most of these recent works model the Big Five factors [17], with the BFI-10 [18] and the TIPI [19] questionnaires, but the first scores a little better in most parameters. However, there is no application in the platform's level design. So, it can be explored in this work.

Moreover, this work's methodology was based on the work of Freilão [20], by adapting content, in this case, levels, to players given their personality and a default range of levels' parameters as input. Then the players play a default and an adapted version, answering questionnaires about their experience after playing each. Finally, the players answer a questionnaire about their general experience.

In conclusion, besides adapting levels, the proposed solution could also be used as a recommendation system by adjusting and suggesting content with or without the player's knowledge.

C. Video Games Level Design

Over the years, there have been attempts to categorize game content. Henrikx et al. [21] proposed a taxonomy with six layers of game content: (i) bits, (ii) space, (iii) systems, (iv) scenarios, (v) design, and (vi) derived. Liapis et al. [22] defined a taxonomy of six creative facets: (i) visuals, (ii) audio, (iii) narrative, (iv) levels, (v) rules, and (vi) gameplay.

The levels facet is considered in this work. Levels are virtual spaces where the gameplay occurs, ranging from extremely simple to highly complex areas [22]. Searching for Level Design works, there are mainly two types that can be found: some are more focused on design, and others are more focused on the generation. For this work, the most important is the generation works, where tools such as *Tanagra* [23] can fit.

Smith et al. [24] propose a taxonomy of five categories for components observed in industry platform levels: (i) platforms, (ii) obstacles (e.g., gaps, enemies, stompers), (iii) movement aids (e.g., springs), (iv) collectible items (e.g., coins), and (v) triggers, which add puzzle elements into platformers.

Sorenson et al. [25] identified *Infinite Mario Bros* design elements, i.e., the elements and constraints for these elements necessary to build a level of these games. Moreover, the authors show that each genre has distinct elements.

1) Adaptive Platform Levels' Elements: Many elements can be generated or adapted in platform games, as seen in table I. Most adaptations change objects' placement or number, but the width and height of gaps can also be found [9], [12], [23] by changing their average, minimum, or maximum values. Stompers and springs always have preset sizes in platformers, but varying them could improve players' engagement.

Other elements or ways to adapt them are not seen in the literature and could be explored in an initial stage of development. For instance, the speed of the enemies or the player could be varied to hamper or ease the players' progression in a level. However, with these adaptations, the game can lose some of its essences, so they should be used carefully.

III. Exploring Player Adaptivity in Level Design

To answer Q1 and Q2, a generic methodology was elaborated. In this work, it was implemented with a specific game genre and specific game elements, but it can and should, be applied to others. The following subsections describe the methodology allowing to replicate it in other works and the case study created to infer correlations between players' personalities and game elements.

A. Methodology

As stated above, this methodology (figure 1) is generic and can be applied to distinct game genres and players' profiling

Element	Ways to generate and adapt them
Blocks / Boxes	Number [9] Placement [25]
Coins	Number [24] Placement [24]
Enemies	Number [9], [24], [26], [27] Placement [23], [24], [26], [27]
Gaps / Holes, where the character may fall and die	Height [23] Number [9], [12], [24], [26], [27] Placement [12], [23], [24], [27] Width [9], [12], [23], [26]
Platforms	Movement Direction [24] Number [24] Placement [23], [24] Slope [24]
Powerups	Number [9]
Spawn Points	Number [9]
Spikes	Number [24], [27] Placement [24], [27]
Springs	Number [24] Placement [23], [24]
Stompers	Number [24] Placement [23], [24]

TABLE I
PLATFORM LEVELS' ELEMENTS, AND WAYS TO GENERATE OR ADAPT THEM.

methods. It is divided mainly into three phases: (i) **Game Selection**, where the genre and game within it are selected, (ii) **First Study**, where correlations between profiles and game elements are analyzed, and (iii) **Second Study**, where an adaptation is created, and its impact is measured.

In the Game Selection stage, the game genre is firstly selected, and a game within it that allows for content generation should be chosen. If no game is found, a new one should be created, with the awareness that the generator can influence people's experiences, yielding unexpected results.

After this stage, the first study can be prepared by modifying the game to track game elements and gameplay metrics and measure player profiles and preferences, which can be done via questionnaires. The study should include pre-generated levels, enabling researchers to control the obtainable data. With a new dataset, researchers can analyze the data and answer Q1. Supposing the results are insufficient for an adaptation, the study should be reconducted.

Upon the first study, researchers can use its results to modify the prototype. An A/B methodology should be used to study the adaptation's impact by showing the players' adaptive and non-adaptive versions. With the analysis of these results, researchers can conclude if the adaptation impacted the players' experience and measure its impact, answering Q2.

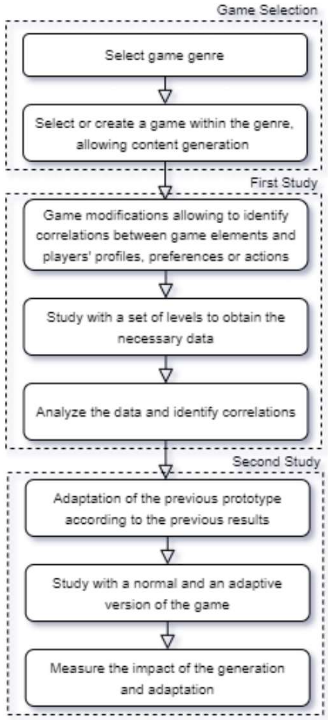


Fig. 1. Flowchart with the proposed methodology split into phases.

B. Platformer Case Study

This subsection describes the case study created from the previous methodology to infer correlations between players' personalities and game elements and measure the impact of levels' adaptations, i.e., the game selection stage and the modifications to this architecture shared by both studies.

1) Game Selection: After selecting the genre, research was conducted for platform games allowing for content generation. Several games were found, but only some meet all the requirements, where *Infinite Mario Bros* [3] stood out since it already contained a port for *JavaScript*[1], allowing it to run on browsers. Besides, it has been used in previous works in the game adaptivity field.

2) Game Architecture Modifications: The original *Infinite Mario Bros* (IMB) [3] contained six states[2], but half of them were unnecessary to present a single level to the players each time. Therefore, the map, win, and lose states were removed from the game. The game states remained relatively similar, but with some changing in their names. The Title state led to the **Predefined Title** state, allowing to show the game's controls. The Level state led to the **Predefined Level** state, allowing players to enter a previously generated level and play it with only one life and redirecting them to a new questionnaire or level when completing or losing the level.

While entering the Level state, IMB creates, with the LevelGenerator, a new Level with a random configuration. In these prototypes, these classes were also extended, allowing not only to generate but also store and load levels to and from the *JavaScript Object Notation* (JSON) format. Furthermore, by decoupling the game sections (jump, tube, straight, hill, and cannon) and the enemies array from the levels map and data, the generator could efficiently store the levels and send them with HTTP requests to a server, thus allowing researchers to reproduce the users' experimentations.

A crucial class in the game is *Mario*, which contains a reference to the character and the global map state. A *PLAY_PROTOTYPE* boolean was added to this class, allowing players to play the actual game or the prototypes. Besides, the

[1]https://github.com/OpenHTML5Games/games-mirror/tree/gh-pages/dist/mariohtml5
[2]https://github.com/pemesteves/game-adaptivity/wiki/IMB-Architecture

MarioCharacter variable points to a *Character* class, which was expanded to store the gameplay metrics, one of the main contributions of this work. The *GameplayMetrics* class stores data from the gameplay, which is used to analyze players' behaviors, containing: (i) arrays with the jumps and landings distances to the nearest gaps, (ii) an array with the wall jumps positions, (iii) arrays with the IDs of coins, powerups, and enemies collected and killed, (iv) the number of coins, powerups, and enemies present in the level, (v) the players' cause of death, represented by the *CauseOfDeath* enumeration, (vi) the time left in the clock when the player leaves the level, by dying or completing it, (vii) an array with the players' actions, containing the keys, ticks, and input events, which allows reproducing the experimentations, and (viii) the level state to access the level's content.

Moreover, three classes were developed to store and reproduce players' actions: *Agent*, *PlayerAgent*, and *AIAgent*. The *Agent* class is abstract and allows storing the actions during the game's execution, the number of ticks, and the time since the game started. The other classes have opposite purposes: (i) the *PlayerAgent* records the players' actions by registering events of the game's input keys, and (ii) the *AIAgent* reproduces an array of actions by triggering the specified events.

IV. Correlating Content with Profiles

This section describes the second stage of the methodology (First Study), thus answering Q1. However, since the study's results can be connected with the results from the second study, their discussion is made in section VI.

Most of the preliminary work made for this study was composed of the game selection stage (subsection III-B1), where the platform game genre was chosen as well as IMB since it included a level generator and a port for *JavaScript* (JS). Furthermore, there was the need to create and distribute questionnaires about profiles and the experience in the levels. The questionnaires should be embedded in the game or vice-versa to allow players to answer them during the playthrough. So, *SurveyJS* was used, embedding the game's levels in the survey's pages, and combining the questionnaires with the levels.

A. Survey and Game Modifications

Using *SurveyJS*, a survey was created to correlate players' profiles, particularly players' personalities, with game elements. It was divided into three main sections, each representing an HTML page: (i) player profiling (index page), with questionnaires about demographics, personality (BFI-10 [18]), and gaming experience, (ii) levels (mario page), wherein the user played a set of twelve pre-generated levels and answered questions concerning their experience in that level, and (iii) comments (comments page), where players could provide feedback regarding their experience in the survey.

Each level was pre-generated to evaluate a specific element by drastically increasing or decreasing its chance to spawn, as seen in table II. Since only the elements' chance to spawn was changed, most variables changed across the levels. However, assuring that only a single variable change would require a more extensive set of levels. Moreover, these levels would be highly similar, probably resulting in a worse experience.

The study focused on elements common to most platformers, testing the enemies' number (ET) and types (ETT), the gaps' width (GT), and the collectibles' number (CT). Each element has an identifier, allowing to gather data about them and understand which influenced the players' experience more.

Moreover, to show every level to the player and ensure the data was correctly collected, other values were stored during the experience, such as (i) the users' global unique identifier (GUID), (ii) the pre-generated levels, (iii) the levels' order, randomly generated before starting any level, and (iv) the current level users are playing.

Level	Purpose	Enemies	Enemy Types	Gaps	Pwps.	Coins
1 - Avg. Num. Enemies	ET/ ETT	17	3	1	1	77
2 - Avg. Gap Width	GWT	0	0	25	0	0
3 - Fewer Enemies	ET	6	4	2	5	51
4 - Fewer Enemy Types	ETT	10	2	1	2	41
5 - More Collectibles	CT	14	4	0	6	70
6 - More Enemies	ET	30	7	9	1	21
7 - More Enemy Types	ETT	27	7	3	0	54
8 - Narrower Gaps	GWT	0	0	14	0	0
9 - No Collectibles	CT	13	5	1	0	0
10 - No Enemies	ET	0	0	3	6	41
11 - No Enemies, No Gaps	ET/ GWT	0	0	0	8	52
12 - Wider Gaps	GWT	0	0	19	0	0

TABLE II
First study pre-generated levels and their elements' configuration.

B. Experience Deployment and Data Collection

The game was deployed on an *itch.io* page [3]. Since it was developed using *JavaScript*, everyone with a computer and an Internet connection could run it on a browser. After the deployment, the link for the game's page was spread around several communities, reaching a good sample size of 282 people. Since the questionnaires were integrated with the game levels and *itch.io*, *Google App Scripts* was used to deploy a web application capable of receiving requests from the game and storing them in a spreadsheet, allowing to collect and store the players' data reliably.

C. Results

The survey was available for three weeks, receiving 308 answers [4], where 26 were duplicates. In other words, 282 people contributed to the study, where 115 completed the questionnaire, and 37 commented on the overall experience. These comments are discussed in section VI.

The sample's demographics did not allow to draw conclusions since people were not well-distributed by gender, age, or education. For instance, 68% of the sample were men while 29.4% were women, which could result in biased results.

[3]https://pemesteves.itch.io/game-design-adaptivity

Due to an error in the data collection, data about the users' gaming experience were not stored for a large part of the population, more precisely 72.7% of the sample, so conclusions concerning gaming experience cannot be drawn.

Besides the questionnaires about player preferences, some gameplay metrics and elements data were collected and used to produce statistics and correlate elements with players' profiles. Subsubsection IV-C1 shows these results for each level, while subsubsection IV-C2 compares levels with the same purpose.

1) Level Data: After carefully analyzing each level's data, using mainly *RapidMiner*, no correlation between players' personalities and game elements could be found. However, some results related to the players' stated preferences were deeply investigated. A decision tree and a correlation matrix between the overall experience, other game properties, and players' preferences were generated for each level. With this tree, a cross-validation strategy could be applied to predict its values. The accuracy of this strategy and statistics about the overall experience in the levels are shown in table III. Although users needed to state their experience on a scale of 1 (Very dissatisfied) to 5 (Very satisfied), these values were grouped into three categories since there was no need to distinguish between a good and an excellent one.

Level	**Satisfied**	**Neutral**	**Dissatisfied**	**Total**	**CV accuracy**
1	80 (51.6%)	58 (37.4%)	17 (11.0%)	155	43.6%
2	33 (22.0%)	53 (35.3%)	64 (42.7%)	150	42.7%
3	102 (67.6%)	36 (23.8%)	13 (8.6%)	151	68.2%
4	66 (48.9%)	52 (38.5%)	17 (12.6%)	135	52.6%
5	98 (73.1%)	25 (18.7%)	11 (8.2%)	134	74.8%
6	53 (41.7%)	43 (33.9%)	31 (24.4%)	127	37.8%
7	67 (48.2%)	50 (36.0%)	22 (15.8%)	139	44.6%
8	24 (18.8%)	33 (25.8%)	71 (55.5%)	128	54.5%
9	48 (40.3%)	41 (42.9%)	20 (16.8%)	129	44.6%
10	40 (33.5%)	47 (35.9%)	44 (33.6%)	131	41.2%
11	48 (34.0%)	38 (27.0%)	55 (39.0%)	141	34.1%
12	19 (16.1%)	34 (28.8%)	65 (55.1%)	118	53.4%

TABLE III
OVERALL SATISFACTION AND CROSS-VALIDATION (CV) PREDICTION ACCURACY FOR EACH LEVEL.

Other analysis methods, such as the Principal Component Analysis and clustering, were explored. However, they were discarded for not yielding relevant results.

2) Level Comparison: Besides analyzing each level individually, levels testing a specific element were grouped, and the satisfaction and dissatisfaction degrees, i.e., the amount of satisfied and dissatisfied answers divided by the total number of responses, were calculated.

a) Gaps: The gaps testing comprised levels 2, 8, and 12, as seen in table IV. The basis produced greater satisfaction and lesser dissatisfaction, which could be due to the lack of challenge with small gaps and a massive challenge with big holes. Levels containing only gaps are not good in general, since they need more elements and sections without jumps to make the players feel they are playing a Super Mario level.

	8 - Narrower	**2 - Average Width**	**12 - Wider**
Sat./Total	0.188	0.220	0.161
Dis./Total	0.555	0.427	0.551

TABLE IV
SATISFACTION COMPARISON BETWEEN GWT LEVELS.

b) Number of Enemies: The enemies' number was tested with levels 1, 3, and 6, but level 10 was added since it did not contain enemies. Table V shows that decreasing the number of enemies produces more satisfaction in the players. However, removing every enemy makes the experience neutral. So, it can be concluded that it is good to have challenges by introducing enemies in levels, but they should not be exaggerated, i.e., their number must remain relatively low.

	10 - None	**3 - Fewer**	**1 - Average Number**	**6 - More**
Sat./Total	0.305	0.675	0.516	0.417
Dis./Total	0.336	0.086	0.107	0.244

TABLE V
SATISFACTION COMPARISON BETWEEN ET LEVELS.

c) Number of Enemy Types: The variety of enemies was tested with levels 1, 4, and 7. Analyzing table VI, there does not seem to be much difference between each level.

	4 - Fewer	**1 - Average Number**	**7 - More**
Sat./Total	0.489	0.516	0.482
Dis./Total	0.126	0.107	0.158

TABLE VI
SATISFACTION COMPARISON BETWEEN ETT LEVELS.

d) Number of Collectibles (Coins and Powerups): The number of collectibles was tested with levels 1, 5, and 9. By analyzing table VII, it can be concluded that increasing the number of collectibles produces more satisfaction in the players. A possible explanation is that coins are fun to collect and a goal whereby people play, while powerups make the game easier by helping defeat enemies, potentially helping newbies and also motivating experienced players to continue playing.

	9 - None	**1 - Average Number**	**5 - More**
Sat./Total	0.403	0.516	0.731
Dis./Total	0.168	0.107	0.082

TABLE VII
SATISFACTION COMPARISON BETWEEN CT LEVELS.

e) Level without challenges (enemies and gaps): Level 11 represents the last test, and it neither has enemies nor gaps, so everyone could quickly complete. Table VIII shows the lack of challenges leads to a neutral experience since players only need to run to complete the level. This can be good for new players to learn some of the basic game's mechanics, but bad for the ones that want to be challenged by the game.

	11 - No Enemies, No Gaps
Sat./Total	0.340
Dis./Total	0.390

TABLE VIII
DATA FROM THE LEVEL WITHOUT CHALLENGES.

3) Summary: With the completion of this analysis, it was possible to answer Q1: Do players' profiles correlate with their preference for specific game elements? Although several analyses were made with the data collected from the study, it was not possible to correlate players' profiles with their preferences for specific game elements.

So, the original objectives were changed, leading the second study to focus on generating levels that improved the overall experience instead of developing them according to the players' profiles. By analyzing the levels by purpose, it can be concluded that the players' overall experience did not improve when changing the variety of enemy types and gaps' average width. However, the experience improved when the number of enemies decreased, without eliminating them all, and when the number of collectibles increased, which could be used in the second study's adaptation.

Since some results did not yield the expected outcome, section VI discusses possible reasons for them and additional ways to analyze the data collected in future studies.

V. Improving Experience through Adaptivity

This section describes the third and last stage of the methodology (Second Study), thus answering Q2. This stage depends on the previous since it consists of developing an adaptive version of the game based on the first study's results. This way, the prototype was modified, allowing to generate and adapt levels. Since the study's results can be connected with the previous results, their discussion is made in section VI.

A. Survey and Game Modifications

As for the previous study, a survey was created with *SurveyJS* to measure the impact of generating and adapting levels to the overall players' preferences for game elements. The survey was divided into three main sections: (i) player profiling (index page), equal to the previous survey, (ii) levels (mario page), where the user played an automatically generated level and an adapted version of it, and (iii) preferences and comments (comments page), where players could state their preferences and provide feedback about the whole experience.

To create the adaptation, the levels' generator was extended to generate and adapt the same level by deleting 1/3 of the enemies, adding coins to decorative sections, and changing empty blocks for collectible blocks with a 2/3 chance. These values were chosen based on the authors' expectations and serve as a starting point for future iterations, as they may be excessive or not nearly enough to influence the game experience.

Moreover, to show the same level to the player twice (standard and adapted version) and ensure the data was correctly collected, other values had been stored in the browser's local storage during the experience, such as (i) the users' GUID, (ii) an identifier, ensuring the playing order does not affect the results of the A/B testing, (iii) the level number users are playing, and (iv) the standard level.

The main differences between the surveys' questionnaires were the level and the comments sections. In this study, there was no level questionnaire, leaving the questions related to the level, such as which level had more of a particular element and which level the player preferred, for the end.

B. Experience Deployment and Data Collection

As in the first study, the game was deployed on an *itch.io* page[4], allowing everyone that has a computer and an internet connection to run and answer it using only a browser. The link for the game's page was spread around several communities, reaching a good sample size of 178 people. To collect the questionnaires' answers and gameplay data, the first study's script was used to deploy a web application capable of receiving requests from the game and storing them in a spreadsheet.

[4]https://pemesteves.itch.io/game-design-adaptivity-2

C. Results

The survey was available for two weeks, receiving 189 answers [5], where 11 were duplicates, possibly from people experiencing issues during the game. In other words, 178 people contributed to the study, where 142 completed the questionnaire, playing both levels and answering the final questionnaire. Of these 178, 28 commented on the overall experience. This time, most people commented on issues related to the game mechanics or explained their preferences.

As seen in table IX, the adaptation worked as expected, consistently decreasing the enemies' number and increasing the number of collectibles, coins, and powerups, except for one case where the powerups' number was not increased.

Element	Min.	Max.	Avg.	Std. Dev.
Coins	19	77	44.605	14.688
Enemies	-5	-17	-9.111	2.208
Powerups	0	13	4.407	2.519

TABLE IX
Difference in the number of elements between the levels.

The sample's demographics did not allow to draw conclusions since people were not well-distributed by gender, age, or education. For instance, 68% of the sample were men while 29.4% were women, which could result in biased results.

This time, the users' gaming experience was well stored for every user. The survey counted on people more experienced in video games since some have participated in the first study.

Besides the questionnaires about player preferences, some gameplay metrics and data about elements were collected and used to produce statistics and remove outliers. However, the time left in the clock was not collected due to an error, which was solved with the number of update ticks. Both levels were analyzed according to the ticks data to remove other outliers. Experiences with values greater than or equal to 6000 ticks and lesser than the first quartile (368 in level 1, 937.25 in level 2) were deleted since these corresponded to people who died by reaching the time limit (inactive player) or died too soon in a level, respectively. This resulted in a sample of 81 people, which was used for the other analyses.

After removing the outliers, a *RapidMiner* process was used to analyze the preference for each level and the players' feelings about the difference in the elements' amounts. Table X shows these preferences, where unexpected results can be spotted, especially on the gaps questions since these elements were not changed. Moreover, regarding the adapted elements, i.e., coins, powerups, and enemies, testers seem to know more or less what levels contain more of these elements. However, some could not identify these differences, possibly due to death in different parts of the levels.

Furthermore, a correlation matrix for each preference was generated, showing no significant coefficient. However, it suggests that, for example, women tend to prefer the adapted level, while men tend to prefer the standard level.

Question	Level 1 (Regular)	Level 2 (Adapted)	None
Level Preference	36 (44.44%)	35 (43.21%)	10 (12.35%)
Feel had more Coins	12 (14.81%)	49 (60.49%)	20 (24.69%)
Feel had more Enemies	69 (85.19%)	7 (8.64%)	5 (6.17%)
Feel had more Gaps	18 (22.22%)	21 (25.93%)	42 (51.85%)
Feel had more Powerups	4 (4.94%)	70 (86.42%)	7 (8.64%)

TABLE X
PLAYERS' PREFERENCES AND FEELING ABOUT THE ELEMENTS' AMOUNT.

As in the first study, a decision tree was created to understand if a model like this could be used as a recommendation system. Although the tree created several clusters, it only reached an accuracy of 53.19% in the cross-validation test, which is not enough to recommend levels.

With the conclusion of the second study, it was possible to answer Q2: Does automatically generating and adapting levels to the overall preferences of players impact their experience? As concluded by looking at table X, the adaptation did not improve or worsen the overall players' experience, i.e., adapting levels did not impact players' overall experience.

Since some results were unexpected, section VI outlines them by discussing possible reasons for those results and other ways to analyze the data collected in future studies.

VI. Discussion

Chapters IV and V presented several results, some unexpected. So, this section presents possible reasons for it, targeting both studies and the game selected, considering the comments provided by players in the two surveys conducted.

A. Infinite Mario Bros

The chosen game was not developed solely for these studies, and it was only changed to allow storing and loading levels. So, some issues were spotted by players while answering the surveys: (i) in the second study, the generated levels were "very bland and effectively empty"; (ii) the game ran at a lower frame rate in some browsers, such as *Mozilla Firefox*; (iii) the character's acceleration was higher than the original; (iv) the game did not register some inputs, making it difficult to jump; (v) some mechanics, like pick up and throw shells, were missing; (vi) the game's controls were "not much intuitive at first," and the game should allow changing them. Besides, some mechanics, such as the wall sliding and jumping (not present in the original game), got some positive feedback.

Since the work was focused on understand how the game's elements impacted the players' experience, and if this impact could be correlated to the players' profiles, the questionnaires were made to ignore these issues, focusing on the game elements. Some people may have given wrong answers just because of these issues, so deeper analyses of the studies' data should be conducted in later studies, by eliminating more outliers and expand it to the machine learning area.

B. First Study

Many comments about the questionnaire and the game were made during the first study. Most people complained about the only life given and suggested that the first level "should be a test, only to test the time of response of the game." Some suggested substituting the gaming experience's questions with yes/no questions, and the levels' questions scale from the feeling of satisfaction to the feeling about the elements' amount. Some people commented on their preferences, for example, "all the levels without enemies were very boring and not enjoyable." A person said that the "frequency of gaps and obstacles around them affect difficulty more than the width," which, although not considered in this study, should be studied in future work. Moreover, people commented on how well they performed and answered the questionnaire, some with mistakes that were not used to spot outliers since others could make the same mistakes without commenting. Other testers said what their feeling of satisfaction meant in terms of the elements' amount. Therefore, the data should be better analyzed in the future, removing any outliers.

Despite these comments, some people "liked to answer this survey in this way" (translated from Portuguese), which can be because the survey incorporated a game without forcing people to download it and play it on their computer.

To conclude, it was not possible to correlate players' profiles with their preferences for specific game elements. However, groups of elements should be considered, and a question for future research can be raised here: Do players' profiles correlate with their preference for groups of game elements?

C. Second Study

Most people (16/28) commented on the lives given in each level, stating they lost too early on one or both. Besides, one person suggested the game should give three lives for each level, but most players who completed the first study could give up early, not finishing every level due to the survey's length. Before starting the survey, players could be presented at a tutorial level to bypass these issues. Nevertheless, a survey with three lives should also be conducted to conclude if this issue is relevant for evaluating the experience.

Beyond lives, seven testers commented on their preferences, showing they preferred the standard level since the other was "a little poorly designed," or "non-challenging, and [...] less enjoyable." Since each player experienced a different level, this could be related to these players' levels, or the adapted level could not fit the players' skills.

To conclude, generating and adapting levels to the overall players' preferences does not impact their experience. However, further analyses should be made with the data, eliminating more outliers and applying machine learning.

VII. Conclusions

Game Adaptivity was addressed through the platform levels' generation problem. Solving this problem could help game designers in the development process, enabling faster development and possibly better results.

Works related to Game Adaptivity, Level Design, and Player Modeling and Profiling were reviewed, establishing a lack of research in platform game adaptivity using players' personalities as input to the players' models, and highlighting elements that can be adapted or generated and ways to do it.

A methodology that can be extrapolated to other games and genres was designed to answer Q1 and Q2, and it is divided into three main stages: (i) game selection, (ii) first study, and (iii) second study. Based on the methodology, a case study was created with a port of *Infinite Mario Bros* [3].

After selecting the game, a study was conducted to correlate game content with players' profiles. A survey was developed by integrating *SurveyJS* with the game engine, resulting in a prototype launched at *itch.io*, counting 282 responses, of which 115 completed every level. However, it was not possible to find correlations between single game elements and players' profiles. Nevertheless, the survey showed intriguing results when comparing levels by purpose. So, it could be concluded that overall the players' experience improved when decreasing the number of enemies, without eliminating them all, and increasing the number of collectibles.

Based on the previous results, another study was conducted to improve players' experience through offline adaptivity, counting 178 responses, where 142 completed both levels and answered about their preferences. However, its results show it did not improve or worsen the overall players' experience.

After both studies were presented, those results were discussed through the players' comments. Several issues were spotted, mainly on the selected game. Other players thought some questions might be asked differently.

In the future, the issues spotted by the survey testers should be tackled. A new iteration of the studies could be conducted by applying other algorithms and machine learning, ensuring the data was adequately collected and matched the players' feelings and beliefs. Another game could be selected to conclude if it influences the results. Lastly, the connections between elements should be explored, possibly producing different results than analyzing single elements.

REFERENCES

[1] Ricardo Lopes and Rafael Bidarra. Adaptivity challenges in games and simulations: a survey. *IEEE Transactions on Computational Intelligence and AI in Games*, 3:85–99, 2011.

[2] Georgios N Yannakakis, Pieter Spronck, Daniele Loiacono, and Elisabeth Andre. Player modeling. *Artificial and Computational Intelligence in Games*, 2013.

[3] Markus Persson. Infinite mario bros, 2006.

[4] Pedro Esteves. Players' profiles and satisfaction for game elements across levels: Dataset. https://doi.org/10.5281/zenodo.6623375, 2022.

[5] Pedro Esteves. Players' profiles and preferences in standard and adapted levels: Dataset. https://doi.org/10.5281/zenodo.6677476, 2022.

[6] Sahraoui Dhelim, Nyothiri Aung, Mohammed Amine Bouras, Huansheng Ning, and Erik Cambria. A survey on personality-aware recommendation systems. *Artificial Intelligence Review*, 2021.

[7] Tianye Shu, Jialin Liu, and Georgios N. Yannakakis. Experience-driven pcg via reinforcement learning: A super mario bros study. In *2021 IEEE Conference on Games (CoG)*, pages 1–9, 6 2021.

[8] Noor Shaker, Georgios N Yannakakis, Julian Togelius, Miguel Nicolau, and Michael O'neill. Evolving personalized content for super mario bros using grammatical evolution. In *Eighth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2012.

[9] Noor Shaker, Georgios N. Yannakakis, and Julian Togelius. Crowd-sourcing the aesthetics of platform games. *IEEE Transactions on Computational Intelligence and AI in Games*, 5:276–290, 2013.

[10] Georgios N. Yannakakis and Julian Togelius. Experience-driven procedural content generation. *IEEE Transactions on Affective Computing*, 2:147–161, 7 2011.

[11] Anurag Sarkar and Seth Cooper. Towards game design via creative machine learning (gdcml). In *2020 IEEE Conference on Games (CoG)*, pages 744–751. IEEE, 8 2020.

[12] Christopher Pedersen, Julian Togelius, and Georgios N. Yannakakis. Modeling player experience for content creation. *IEEE Transactions on Computational Intelligence and AI in Games*, 2:54–67, 3 2010.

[13] Carlos Pereira Santos, Kevin Hutchinson, Vassilis Javed Khan, and Panos Markopoulos. Profiling personality traits with games. *ACM Transactions on Interactive Intelligent Systems*, 9, 3 2019.

[14] Danial Hooshyar, Moslem Yousefi, and Heuiseok Lim. Data-driven approaches to game player modeling: A systematic literature review. *ACM Computing Surveys*, 50, 1 2018.

[15] Stephen Karpinskyj, Fabio Zambetta, and Lawrence Cavedon. Video game personalisation techniques: A comprehensive survey. *Entertainment Computing*, 5:211–218, 9 2014.

[16] Edirlei Soares de Lima, Bruno Feijó, and Antonio L. Furtado. Player behavior and personality modeling for interactive storytelling in games. *Entertainment Computing*, 28:32–48, 12 2018.

[17] Oliver P. John and Sanjay Srivastava. The big five trait taxonomy: History, measurement, and theoretical perspectives. In *Handbook of personality: Theory and research*, pages 102–138. Guilford Press, 1999.

[18] Beatrice Rammstedt and Oliver P. John. Measuring personality in one minute or less: A 10-item short version of the big five inventory in english and german. *Journal of Research in Personality*, 41:203–212, 2 2007.

[19] Samuel D. Gosling, Peter J. Rentfrow, and William B. Swann. A very brief measure of the big-five personality domains. *Journal of Research in Personality*, 37:504–528, 2003.

[20] Matilde Freilão. Affective narratives for engagement in digital games. Master's thesis, Faculdade de Engenharia da Universidade do Porto, 2020.

[21] Mark Hendrikx, Sebastiaan Meijer, Joeri Van Der Velden, and Alexandru Iosup. Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications and Applications*, 9, 2 2013.

[22] Antonios Liapis, Georgios N. Yannakakis, Mark J. Nelson, Mike Preuss, and Rafael Bidarra. Orchestrating game generation. *IEEE Transactions on Games*, 11:48–68, 3 2019.

[23] Gillian Smith, Jim Whitehead, and Michael Mateas. Tanagra: A mixed-initiative level design tool. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games - FDG '10*, pages 209–216. ACM Press, 2010.

[24] Gillian Smith, Jim Whitehead, Michael Mateas, Mike Treanor, Jameka March, and Mee Cha. Launchpad: A rhythm-based level generator for 2-d platformers. *IEEE Transactions on Computational Intelligence and AI in Games*, 3:1–16, 3 2011.

[25] Nathan Sorenson, Philippe Pasquier, and Steve DiPaola. A generic approach to challenge modeling for the procedural creation of video game levels. In *IEEE Transactions on Computational Intelligence and AI in Games*, volume 3, pages 229–244, 9 2011.

[26] Elizabeth Camilleri, Georgios N. Yannakakis, and Alexiei Dingli. Platformer level design for player believability. In *IEEE Conference on Computatonal Intelligence and Games, CIG*, volume 0. IEEE Computer Society, 7 2016.

[27] Henry D. Fernández, Mikami Koji, and Kunio Kondo. Adaptable game experience based on player's performance and eeg. In *Proceedings - 2017 NICOGRAPH International, NICOInt 2017*, pages 1–8. Institute of Electrical and Electronics Engineers Inc., 9 2017.

Session 5: Virtual and Augmented reality II

chair: Daniel Mendes

Supporting Research in Memory and Contamination through a Virtual Reality Approach

Diana Silva, Lisandra Fernandes
IEETA / DETI
University of Aveiro
Aveiro, Portugal
drgs@ua.pt, natalia.fernandes@ua.pt

Sónia Santos, Beatriz Pedro
WJCR / DEP
University of Aveiro
Aveiro, Portugal
santos.s@ua.pt, beatriznpedro@ua.pt

Bernardo Marques
DigiMedia / DeCA, IEETA / DETI
University of Aveiro
Aveiro, Portugal
bernardo.marques@ua.pt

Beatriz Sousa Santos
IEETA / DETI
University of Aveiro
Aveiro, Portugal
bss@ua.pt

Josefa Pandeirada
WJCR / DEP
University of Aveiro
Aveiro, Portugal
josefa@ua.pt

Samuel Silva
IEETA / DETI
University of Aveiro
Aveiro, Portugal
sss@ua.pt

***Abstract*—Biologically, humans have evolved to adapt against pathogens but, since fighting them can be costly, a set of mechanisms have emerged to avoid contact, called the Behavioral Immune System (BIS). Experimental Psychology studies the BIS, for instance, by conducting experiments where a participant is exposed to several objects some of which are presented as having been in contact with a source of contamination (e.g., a sick person). To move the research on BIS and memory into a potentially more ecologically-valid setting, the current work proposes a Virtual Reality (VR) based approach where 360º videos are used as stimuli, during the experiments. At its current stage of development, the proposed framework has already served a first pilot study to assess the overall adequateness of the proposal and validate acquisition parameters for the stimuli with encouraging results.**

***Index Terms*—behavioral immune system, memory, experimental psychology, virtual reality, 360º video, user-centered design**

I. Introduction

Since the beginning of time, humans have recurrently faced various problems, including the threat posed by pathogens. To respond to these threats, humans have developed a series of biological responses aimed at eliminating these pathogens that invade the body, which is called the Biological Immune System [1]. Some authors defend that natural selection has also favored the development of the Behavioral Immune System (BIS), which consists of a set of psychological mechanisms that prevent contact with infectious pathogens [1]. For example, disgust has long been considered an adaptive affective response to help us avoid diseases [2]. More recently, Fernandes and collaborators proposed that memory is also a key component of the BIS [3]. In their studies, they found a mnemonic advantage for contamination, i.e., an enhanced retention for potential contaminated objects (e.g., touched by sick people) compared to non-contaminated objects (e.g., touched by healthy people). Such studies rely on the participants imagination, i.e., visual or verbal stimuli are presented accompanied by descriptive scenarios that participants are asked to imagine.

Considerable efforts have already been made to increase the ecological validity of the experimental procedures (e.g., [4]), but much can still be improved to reduce the amount of the procedure relying on the participant's imagination and increase the sense of presence and immersion in situations of potential contamination. In a first analysis, these aspects may be tackled by moving from static stimuli to more dynamic scenes and by drifting away from the traditional form of presenting them: the computer monitor. In this regard, Virtual Reality (VR) has been considered to increase ecological validity and experiment control in Experimental Psychology with positive outcomes [5], [6] and seems to offer the overall setting required to improve the experimental approach to the research on the Mnemonic Tuning for Contamination (MTC). To this end, in this work we adopt a user-centered design approach, in collaboration with domain experts, and propose a VR-based approach to research on MTC, supported on 360º video stimuli. The main goals of this first stage of the work were to: (a) establish the set of features required to support the experimental workflow for MTC research; (b) design and develop a first proof-of-concept of the core system to validate the overall approach (c) have a first understanding of the impact of the devised setting on participants, e.g., regarding how natural did the environment feel to them, and if they could examine objects and people well.

The remainder of this article is organized as follows: section II provides an overview of current practices adopted for experimental research on mnemonic tuning for contamination along with notable examples on how VR is being considered in Experimental Psychology; Section III describe the overall aspects of the design and development of a first functional prototype; Then, a first test of the platform is presented and

978-8-3503-4573-57/22/$31.00 ©2022 IEEE

discussed in section IV; Finally, section V presents some conclusions and a set of ideas informing future work.

II. Background and Related Work

The research on the Mnemonic Tuning for Contamination has resorted to experimental protocols that, in simplistic terms, establish the potential contamination of certain objects presented to a participant and, then, assesses how this affects the way these objects are remembered, afterwards. In this context, different approaches have been adopted to establish the connection between the presented objects and potential sources of contamination. For instance, in Fernandes et al. [3], line drawings of objects with possible signs of contamination were used along with descriptions or faces (describing the health status of the person who came into contact with the object) as supplements, followed by an immediate memory test using the same object images. Nevertheless, this approach entailed that participants had to rely on their imagination to establish contact between objects and signs of contamination (descriptions or faces) due to the fact that both objects and signs of contamination were displayed side by side without visible contact. For this reason, recent studies considered objects shown in direct contact with potential sources of contamination, followed by an immediate memory test. The photographs used showed two hands holding an object [7], as shown in Figure 1. However, the static presentation of the stimuli in the computer monitor still leaves room for a certain notion of detachment/distance towards the objects, an aspect that VR might help tackle.

Fig. 1. Examples of stimuli used in the different experiments. (A) Used associated with cues (both in the item presentation and the immediate memory phase); (B) Used as the contaminated items; (C) Used to describe hands covered with chocolate spread (non-disease context) or as covered with diarrhea (disease context); and (D) Used in the immediate memory test in (B) and (C).

Virtual environments have allowed researchers in the field of Psychology greater experimental control over what the subject sees and/or hears. These advantages have led several

TABLE I
LIST OF REQUIREMENTS FOR THE TARGET PLATFORM GROUPED BY OVERALL STAGE.

Stimulus Preparation	**Experiment Controller**
– 360º video recording	– define participant information
– Trim videos	– control experiment flow,
– Insert stimuli in a catalog	– show current experiment status
– Remove stimuli from catalog.	– show participant's performance
– Associate tags with videos	– allow annotations
Experiment Configuration	– save experiment data
– browse stimuli catalog	– save participant inputs/responses
– filter stimuli by tag and/or keyword	**Experiment Rendering**
– configure stimuli to be presented	– receive configuration from controller
– preview the experiment	– display stimuli
– save configuration	– control stimuli sequence and flow
– load configuration	– show questionnaires inside VE
– allow presenting questionnaires	– send participant inputs to controller

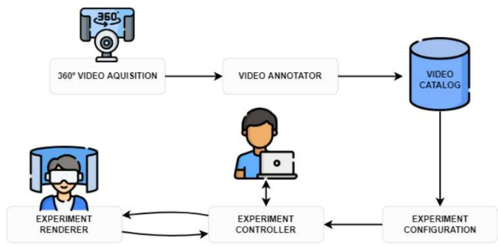


Fig. 2. Overall diagram depicting the different high-level modules envisaged to address the requirements established.

fields of Psychology to use this technology to study human behavior [8] and intervention, such as: treatment of phobias, anxiety, rehabilitation, and others [9], showing a strong potential of VR in helping pave the road to raise experiment ecological validity while keeping control of the experimental tasks [10]. In this regard, one relevant aspect is that 360º video has successfully been considered, in this context [11]. This has the advantage of providing a faster way to create a more realistically looking environment avoiding the tiresome task of modeling it, particularly in conditions where the subject does not need to navigate and/or interact with the environment.

III. Development of First Proof-of-Concept

In this work, we adopt a User-Centered Design (UCD) approach including Participatory Design, working closely with experimental psychologists. Frequent discussions with these experts started by helping to understand the current experimental practice for studying MTC [7]. To understand user motivations and explain the proposed solution, we considered Personas and a set of scenarios [12] to determine functional/non-functional requirements. The functional requirements were divided by four important stages: stimulus preparation, experiment configuration, experiment controller, and experiment rendering. The functional requirements are presented in Table I, organized by the overall stage they refer to. Considering all the requirements for the development of the different phases of this platform, we have created a diagram depicting the different high-level modules as presented in Figure 2.

A. Overall Infrastructure

Considering the requirements presented in Table I and the feedback obtained during the discussions with the domain experts, we defined an overall system architecture.The first step is the acquisition of the 360º videos (in our case, using a Samsung Gear 360º camera). Considering that the technical requirements to develop the video annotator, experiment configuration, and experiment controller are identical, it was decided to develop everything in the same application using Node.js adopting the React.js framework, thus addressing the multiplatform requirement. To store the data required to support the envisaged features (e.g., the information about the stimuli, and experiment configurations) and since the data comes from different sources — its structure is not the same for all data, there are no relationships between the data and they are stored in JSON files in order to facilitate communication between applications — we decided to use semi-structured data, and for this reason a NoSQL database adopting MongoDB Atlas.

Finally, to execute the experiment, the experiment renderer, implemented using Unity, needs to retrieve the information required from the database, and display it in the VR equipment (in our case, the Oculus Quest 2). The information about which videos to play or other information that needs to be exchanged between the experiment controller and the renderer, e.g., to pause the experiment or update experiment status, is sent through streams implemented using Kafka.

B. Current Stage of Development

For the initial development, requirements were analyzed and prioritized based on the system components that would allow the conceptual approach to be tested, as early as possible, to ensure that it was feasible and had the overall expected features and impact on the participants. At its core, the proposed approach has the Experiment Renderer and the control over it, provided by the Experiment Controller, making them the two most important modules for first testing of the concept.

One of the most important features in this project is the ability to exhibit the stimuli in VR. Therefore, this feature was developed first and for testing, a set of 360° videos in a folder were played as soon as the app was opened. Then, in order to have control over which videos were played and when, a web-based control dashboard was created. In creating this tool, message exchange was added for communication between the two modules, making it possible to select, by name or folder, which videos to play. Figure 3 shows the Experiment Controller user interface and an example of a video exhibited in the Experiment Renderer.

IV. Preliminary Evaluation

Two main goals motivated the first preliminary evaluation of the proposed system. The first was to validate the overall features of the current stage of the prototype to assert the ability to: (a) display a set of 360º videos in the headset; and (b) control the experiment (e.g., which videos to present) from the desktop. The second goal was to test which video capturing conditions are the most successful in approaching the virtual environment to real-word experiences. Specifically, we wanted to explore how the camera height and distance between the camera and a table (where the object would be placed) influenced the participant's experience.

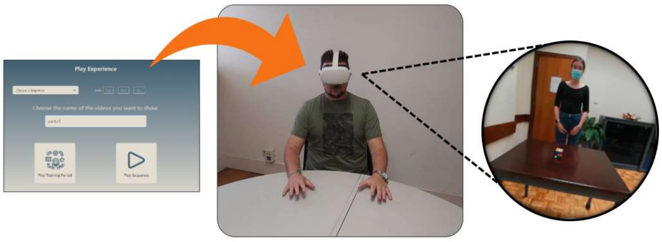


Fig. 3. Example of the current state of the platforms: left, Experiment Controller user interface; center, a participant wearing the Oculus Quest 2 during an experiment; and right, the video content that is being shown by the Experiment Renderer.

A. Experimental Conditions

For recording the different 360º environments a Samsung Gear 360 camera was used mounted on a tripod. Six environments (2 camera heights $\times$ 3 distances to the table), each comprising three 25-sec 360º videos, were created (total duration of each environment was 75-sec). Each video depicted a similar set of actions: a person approaching and putting an object on top of a table positioned in front of the participant who was seated on a chair. Videos were recorded at different heights (110 and 115 cm), with a table positioned at different distances from the camera (35, 45, and 55 cm). Selection of these conditions was based on a literature review exploring the effect of camera height on virtual environments [13] and on preliminary testing of different conditions by our team. The prototype was subject to a series of tests before the pilot study was carried out and adjustments were made accordingly (e.g., a 1-second break between consecutive videos was occurring and interrupted the sense of realism and presence).

B. Protocol

Twelve participants participated in this pilot study (females = 7). Participants were asked to carefully explore each environment. While seated on a chair, they could look around, in the virtual environment, to inspect the room around them. The order of the three videos within each environment was counterbalanced across participants. The order of the presentation of the six environments was randomly determined for each participant. At the end of each environment, participants rated 12 items on a 5-point Likert-type scale ranging from 1 (nothing) to 5 (very much). Some of these questions were from the Presence Questionnaire [14], and others were created specifically for this study (e.g., "How natural did the environment feel to you?","How well were you able to examine the objects?","To what extent did you feel that you were sitting at the table?"). Other subjective feedback was collected (e.g.,

TABLE II
RESULTS OF THE VALIDATION STUDY: AVERAGE AND STANDARD DEVIATION FOR THE RESPONSES GIVEN BY THE PARTICIPANTS TO QUERIES ABOUT THE NATURALNESS OF DIFFERENT CHARACTERISTICS OF THE VR ENVIRONMENT.

experimental conditions	height (cm)	110			115		
	distance (cm)	35	45	55	35	45	55
responses	average	3.98	4.03	4.08	3.88	3.96	4.11
	std. dev.	0.52	0.58	0.53	0.62	0.54	0.61

"How comfortable did you feel in the virtual environment?", "How involved you felt in the virtual environment?"). The questions were presented verbally so that participants would not have to remove the Oculus.

C. Results

The average values (and standard deviations) obtained on the considered set of items for each environment are presented in Table II. Overall, when looking at the average values, the highest values were obtained for the environment captured at the height of 115 cm and the distance of 55 cm.

When we analyze the average rating provided by each participant to each environment, we verify that this same environment obtained the highest rating for 5 of the 12 participants. The next best rated environment was recorded at a height of 110 cm and distance of 55 cm. Thus, there is a consistent choice on the distance (55 cm); the preference for different heights (110 or 115 cm) might be due to variability on the participant's own height. However, a Pearson correlation showed no correlation between participant's height and their average ratings either for environments at a height of 110 cm or environment at a height of 115 cm (highest p = .090).

At the end of the task, participants verbally expressed a higher preference for the environments presented last. Because the order of presentation was determined randomly for each participant, we opted to formally explore a possible influence of the order of presentation of the environment and the preference rank ordering derived from the participants' ratings. A Spearman's correlation revealed a moderate significant correlation effect between the presentation and the rating rank order, $rs = .403$, $p < .001$. This suggests that, indeed, participants tended to rate higher those that were presented last. It also hints that the repeated exposure (or habituation) to these types of environments tended to increase the sense of presence which is a positive indication for using it in the envisaged experimental settings.

V. CONCLUSIONS

To move towards increased ecology and realism in studying the role of the Mnemonic Effect of Contamination in the BIS, this work proposes an approach based on VR and the use of 360º videos as stimuli during the experiments. To this end, and in collaboration with domain experts, a set of tools was conceptualized to support creating and running the experimental studies adopting this novel paradigm. In the initial evaluation, the results show that the system is adequate for supporting the experiment, being able to present the desired sequences of videos in a credible manner, and already allowed the validation of the acquisition parameters for the 360º videos.

The gathered results and feedback are very encouraging and future work includes optimizing the experiment control, adding the ability for a participant to answer questionnaires without leaving the virtual environment, and integrating the remaining modules, such as the experiment annotator, video catalog database, and experiment configuration.

ACKNOWLEDGMENT

This research was performed in the scope of FCT project Prevent2Protect [EXPL/PSI-GER/1008/2021]. It was also supported by IEETA, funded by National Funds through FCT, in the context of project UIDB/00127/2020.

REFERENCES

[1] M. Schaller and J. H. Park, "The behavioral immune system (and why it matters)," *Curr. Dir. Psychol. Sci.*, vol. 20, pp. 99–103, 2011.

[2] V. Curtis, M. De Barra, and R. Aunger, "Disgust as an adaptive system for disease avoidance behaviour," *Philos. Trans. R. Soc. Lond., B, Biol. Sci.*, vol. 366, pp. 389–401, 2011.

[3] N. L. Fernandes, J. N. Pandeirada, S. C. Soares, and J. S. Nairne, "Adaptive memory: The mnemonic value of contamination," *Evol. Hum. Behav.*, vol. 38, pp. 451–460, 2017.

[4] N. L. Fernandes, J. N. Pandeirada, and J. S. Nairne, "Presenting new stimuli to study emotion: Development and validation of the objects-on-hands picture database," *PLoS One*, vol. 14, p. e0219615, 2019.

[5] P. Kourtesis, S. Collina, L. A. Doumas, and S. E. MacPherson, "Validation of the virtual reality everyday assessment lab (VR-EAL): an immersive virtual reality neuropsychological battery with enhanced ecological validity," *J Int Neuropsychol Soc*, vol. 27, pp. 181–196, 2021.

[6] N. Reggente, J. K.-Y. Essoe, Z. M. Aghajan, A. V. Tavakoli, J. F. McGuire, N. A. Suthana, and J. Rissman, "Enhancing the ecological validity of fMRI memory research using virtual reality," *Frontiers in neuroscience*, vol. 12, p. 408, 2018.

[7] N. Fernandes, J. Pandeirada, and J. Nairne, "The mnemonic tuning for contamination: A replication and extension study using more ecologically valid stimuli," *J. Evol. Psychol.*, vol. 19, pp. 1–13, 2020.

[8] J. Brookes, M. Warburton, M. Alghadier, M. Mon-Williams, and F. Mushtaq, "Studying human behavior with virtual reality: The Unity experiment framework," *Behavior research methods*, vol. 52, no. 2, pp. 455–463, 2020.

[9] T. D. Parsons, G. Riva, S. Parsons, F. Mantovani, N. Newbutt, L. Lin, E. Venturini, and T. Hall, "Virtual reality in pediatric psychology," *Pediatrics*, vol. 140, no. Supplement 2, pp. S86–S91, 2017.

[10] N. Foreman, "Virtual reality in psychology," *J Sci Educ Technol*, vol. 2, pp. 225–252, 2010.

[11] A. Ionescu, T. Van Daele, A. Rizzo, C. Blair, and P. Best, "360° videos for immersive mental health interventions: a systematic review," *Journal of Technology in Behavioral Science*, vol. 6, no. 4, pp. 631–651, 2021.

[12] A. Cooper, R. Reimann, and D. Cronin, *About face 3: the essentials of interaction design*. John Wiley & Sons, 2007.

[13] T. Keskinen, V. Mäkelä, P. Kallioniemi, J. Hakulinen, J. Karhu, K. Ronkainen, J. Mäkelä, and M. Turunen, "The effect of camera height, actor behavior, and viewer position on the user experience of 360 videos," in *2019 IEEE conference on virtual reality and 3D user interfaces (VR)*, pp. 423–430, IEEE, 2019.

[14] J. Vasconcelos-Raposo, M. Melo, L. Barbosa, C. Teixeira, L. Cabral, and M. Bessa, "Assessing presence in virtual environments: adaptation of the psychometric properties of the presence questionnaire to the portuguese populations," *Behav. Inf. Technol.*, vol. 40, pp. 1417–1427, 2021.

Authoring tool for creating immersive virtual experiences expeditiously for training

Rui Machado
UTAD
Vila Real, Portugal
rui-12@hotmail.com

Ricardo Rodrigues
UTAD
Vila Real, Portugal
rrodrigues1999.rr@gmail.com

Hugo Coelho
UTAD and INESC TEC
Vila Real and Porto, Portugal
hugo.r.mendes@inesctec.pt

Miguel Melo
INESC TEC
Porto, Portugal
mcmelo@inesctec.pt

Luís Barbosa
UTAD and INESC TEC
Vila Real and Porto, Portugal
lfb@utad.pt

Maximino Bessa
UTAD and INESC TEC
Vila Real and Porto, Portugal
maxbessa@utad.pt

***Abstract*—Virtual reality (VR) is still a field that is in constant development, and people are trying to use it to have a close representation of reality by creating immersive environments. However, despite the existence of some tools that have been adapted to work with VR, they require some experience to work with, and there is a considerable amount of resources that need to be spent to create and maintain the VR experiences, which prevents the adoption and use of all the benefits that VR can bring. This work proposes an architecture for an authoring tool that allows users to create their own virtual experiences without the need for an extensive understanding of it and use them to create a virtual training exercise. This paper uses a case study built upon a real training context scenario applied to the agroforestry field.**

To validate this proposal, a prototype was built and subject to usability and satisfaction tests that demonstrated the ease of understanding and learning of the interfaces and all the functionalities implemented.

***Index Terms*—immersive environments, virtual reality, authoring tool, immersive training**

I. INTRODUCTION

Over the years, Virtual Reality (VR) has been increasing in popularity. Few people are aware that VR first appeared in 1960, when the equipment was expensive and the computational power did not allow interactive experiences which made VR popularity at the time not so great but, with the emergence of lower prices for gaming and entertainment, VR experienced a rise in interest. As a consequence, the VR market has been rising as powerful companies want to revolutionize the gaming and entertainment industry [1].

Alongside VR technology, there have been multiple VR definitions one of which is Kirner et al. [2], which states that VR is an interface that allows users to access applications being executed on a computer. This interface provides real-time visualization, movement, and interaction with the generated 3D environments.

A considerable number of fields are using VR, including the creation of video games [3], education [4], and the training of professionals, which can involve military training and doctors. Many 3D resources have been developed for this technology to train users on how to behave in real-life situations. Using these 3D resources increases the interest in learning and can be used to represent real-life situations [5]. However, VR must constantly be updated to provide a good solution for each scenario. The tools used to create VR experiences are designed to be used by people with enough knowledge. Companies usually hire professionals with the knowledge to create the necessary experiences to solve this issue. In addition, these experiences will probably need some maintenance and updates throughout the time so they can work properly and keep providing a good experience to their users. Ultimately, the creation of VR experiences can be expensive and, because of that, there is a need to create a tool that allows the creation of such experiences without spending too many resources.

Training in VR brings several advantages, like specific situations can be simulated inside a controlled environment using detailed training conditions at any time; they can repeat the training the number of times necessary to understand the procedure; a large number of people can participate without spending many resources. However, as mentioned before, creating and maintaining these virtual training environments (VR experiences) can be expensive. Therefore it is necessary to create an authoring tool to overcome such obstacles.

This work proposes an authoring tool that creates VR experiences for training, developed with a real case study, a partnership with the company VICORT which fits within an ongoing project called SMARTCUT. The authoring tool allows users to create VR experiences that can be saved and opened as many times as needed to perform specific training. After its implementation, the authoring tool is properly evaluated and validated.

II. BACKGROUND CHARACTERIZATION

VR is a computing field that allows users to emerge and interact with a virtual world while using specific devices to

978-8-3503-4573-57/22/$31.00 ©2022 IEEE

simulate the environment and stimulate the user by providing feedback. This makes the virtual experience as close as possible to reality. One needs to address two main concepts about VR: immersion and perception. Immersion is how much a system can abstract its user from reality. Perception is the user's awareness of the surroundings through their senses [6].

A few different types of systems allow people to experience VR. The context of the experience causes the differences between these systems. Three types of systems can be defined by their immersion and perception: immersive, semi-immersive, and non-immersive systems [6], [7]. Non-immersive systems interact with the virtual environment presented to him on a computer screen using a mouse and a keyboard [8]. Semi-immersive systems allow users to interact with the virtual world while performing activities in the real world [9]. Fully immersive systems use a Head Mounted Display (HMD) to present the virtual environment to the user and where they can interact with it using specific controllers [10].

Over the years, VR has been evolving, and its involvement in different fields is increasing. Fields like video games, education, medicine, military, and psychology, among others, are being studied to understand what is the best approach to using VR in these fields. In medicine, VR can solve the interrelation between anatomic structures in the 3D space; a trainee can explore and interact with these structures anytime [11]. In the military, virtual training is used to improve individual and group abilities. A high-resolution image and advanced controls are used to be as close as possible to reality. Without a geographical limitation, many individuals can be involved in simulated confrontations based on real situations and changes in the virtual environment. The train's durability and the resources used to perform this train will decrease, and new operation ideas can appear [12]. Various applications exist, in the field of education, to help students learn about a specific matter. In NewtonWorld, students can learn about the laws of motion by interacting with objects' various states of non-gravity [13]. In MaxwellWorld, they can learn about electrostatic fields and forces with different polarities and intensities [14]. In PaulingWorld, the student can interact with a structure of atoms and see the consequences when one or more atoms are replaced [15].

A. *Authoring tools for creating VR experiences*

Even though tools like Unity [16], Unreal Engine [17] or Cry Engine [18]] allow the creation of virtual environments, they were not developed with such purpose. Instead, they were developed to create games and adapted to these situations over the years. One thing that did not change was the knowledge one has to have to use them. To create these environments, there are a few important characteristics to be taken into account: if there is an objective or not, the visual feedback, the light and sounds from the environment, and interfaces to help navigate through the experience [19].

Unity, being a game engine, gives the user several tools that he can use to create 3D virtual environments. The inspector interface gives information like the position, rotation, and scale of an object, and the number of assets (objects, sound, textures, etc.) the project browser has available to use. The hierarchy interface gives the objects the user uses in the virtual environment. All this does not allow the user to create a virtual environment, so there are plugins like SteamVr [20] that help content creators create the virtual environment. This tool brings new assets that can be used and new interactions, such as teleport, so the user can move around the virtual world without moving in the real world or grabbing and throwing objects. New interactions can also be created, and the user can change them anytime. When the virtual environment is completed, he can run the application, use the necessary equipment, and interact with the environment. In Unity, any user can use a vast selection of assets (some free, others need to be paid for) to implement in their experience. Still, when there is the need to create some logic with those assets, they will need some programmatic knowledge.

Despite how good tools like Unity are, they are not ideal for a person that does not have the essential knowledge to use them. One tool was proposed by Coelho et al. [21], which allows users to create a 3D virtual environment for multisensory experiences, and is based on 360º videos in two ways: with a desktop and with an immersive interface to test which one is the best. This tool allows users to select any frame from the video timeline and add/remove a stimulus. At any time, the user can visualize the experience on the desktop or with the help of HMD. The only difference is the interface presented to the user and how he interacts with it. Some tests were made, and the team concluded that both interfaces were acceptable and preferred the immersive interface to the desktop interface. This tool is simple, and everyone can use it, but it is limited to using a 360º video to create a virtual environment.

VRIA is another tool for creating a virtual environment, and it is a web-based framework for creating 2D or 3D environments. This framework allows users to create and test prototypes without leaving the web application, making this tool easier, quick, and more accessible to a larger audience [22]. The graphic can be constructed by importing a file in JSON or CSV or creating one. Even though it is impossible to watch these creations with the HMD in the web application, the user can use a mouse and a keyboard to navigate the 3D environment. There are available some examples that we can see and even change. To visualize the created environment in a virtual world, using NodeJs and VR equipment is necessary. This tool is simple, and people who know about JSON can benefit from it.

Both these tools that were developed to facilitate the creation of VR experiences have limitations. The authoring tool proposed by Coelho et al. needs 360º videos to create a virtual environment, and the VRIA tool will need to have some data to create a graphic. Even further, these tools were developed to create a more visualization environment than an interactive one. The tool presented next does not need information from the outside to create a virtual environment, and the interaction levels are higher.

III. Proposal for an Authoring Tool for Immersive Experiences

The development of this authoring tool considers SMART-CUT project, which is connected to company VICORT. This company is dedicated to manufacturing, commercializing, and assisting agricultural, forestry, and environmental equipment. They produce forest machines and harvesting heads in the forest area that will be used in specific work situations. All this equipment needs maintenance to verify if it will work correctly to prevent possible accidents. Trained professionals do the maintenance; hiring or training them can be expensive. And so, a discussion was held with VICORT, and it retrieved their expectations, which the authoring tool should address when creating immersive experiences intended for training. With their feedback, functional and non-functional requirements were defined, representing the functionalities the authoring tool should allow users to perform and are presented by the following requirements:

- Create environments - The authoring tool should allow the creation of an environment where it will be executed the different tasks by the operators during the training they are performing.
- Save environments - The authoring tool should allow storage of the environments created to be loaded whenever needed.
- Load environments - The authoring tool should allow loading environments previously created and saved to be edited or executed in a training.
- Define the terrain deformations - The authoring tool should allow the creation of the terrain, being plane, tilted, or on terraces.
- Define the type of trees - The authoring tool should allow the introduction, in the terrain, of different trees being straight, irregular, single-foot trees or multiple-foot trees.
- Distribution of objects - The authoring tool should allow the position and movement of every object available in the environment.
- Create training - The authoring tool should allow the creation of training where it will be defined the tasks that the operator needs to perform.
- Save training - The authoring tool should allow storing the created training to be loaded whenever needed
- Load training - The authoring tool should allow loading training previously created and saved to be edited or executed.
- Define the schedule - The authoring tool should allow the definition of the time of the day when the train will be executed.
- Choose a harvesting head - The authoring tool should choose which harvesting head will be used in the train that will be executed.

This authoring tool aims to create specific real-life situations in the deforestation area that can be used at any time and as many times as needed, so the users can use these scenarios to train procedures used in the day-by day job that this area requires. To do this, there are a few requirements that need to be accounted for: the creation of an environment that represents the real one, where in this part, the definition of the terrain is what is most important to have when it comes to representing the reality; the creation of training that represents a real day of work. Thus, this authoring tool allows any user to create training scenarios where they can choose to deform the terrain and the life forms it contains. Furthermore, users can also choose which harvesting head they want to use in the environment that they create because every harvesting head has its purpose for a specific situation. Before loading the created training, the user can choose between a non-immersive or an immersive training environment.

A. *Authoring tool architecture*

The authoring tool should allow the user to create an experience and, with that experience, create a training environment. The authoring tool comprises two editors: an environment editor and a training editor. The environments created will be used on the training editor because training cannot be executed without an environment. Several pieces of training can occur using the same environment, allowing reuse instead of creating new ones. Whenever the user wants to access the existing environments or training, a communication is made to retrieve some data (if there is any). The user can then use that information to find an environment or training to be executed or edited. The authoring tool architecture is presented in Fig. 1.

The environments will be created by a user in the environment editor where, after he executes all the interactions available and uses the available components (assets repository), this will suffer a serialization in a specific format (JSON format) and be transformed into a string. All this information will then be saved into a database, and the list that contains the available environments will be updated. If the user wants to edit an environment or execute training, the serialized components need to be deserialized to build the environment correctly. After the environments are all updated, the user can use them to choose one for his training.

The training scenarios are created in the training editor by a user. This editor has an interface where the user configures the scenario, assigning it a name and description, defining the time of day, the environment previously created in the environment editor, and the harvesting head. The environment and the harvesting head are mandatory parameters so the training can be executed later. In the end, the training will be saved into the database, and the list that contains the training environments will be updated. Every training has a scenario identifier so it can be loaded when the training is executed or if the user wants to edit it. This information can be consulted with the same editor if the user wants to change some training parameters.

B. *Authoring tool functionalities*

The objective of this work was to create an authoring tool that allows users to create their experiences, immersive or non-immersive. This authoring tool is not immersive but allows

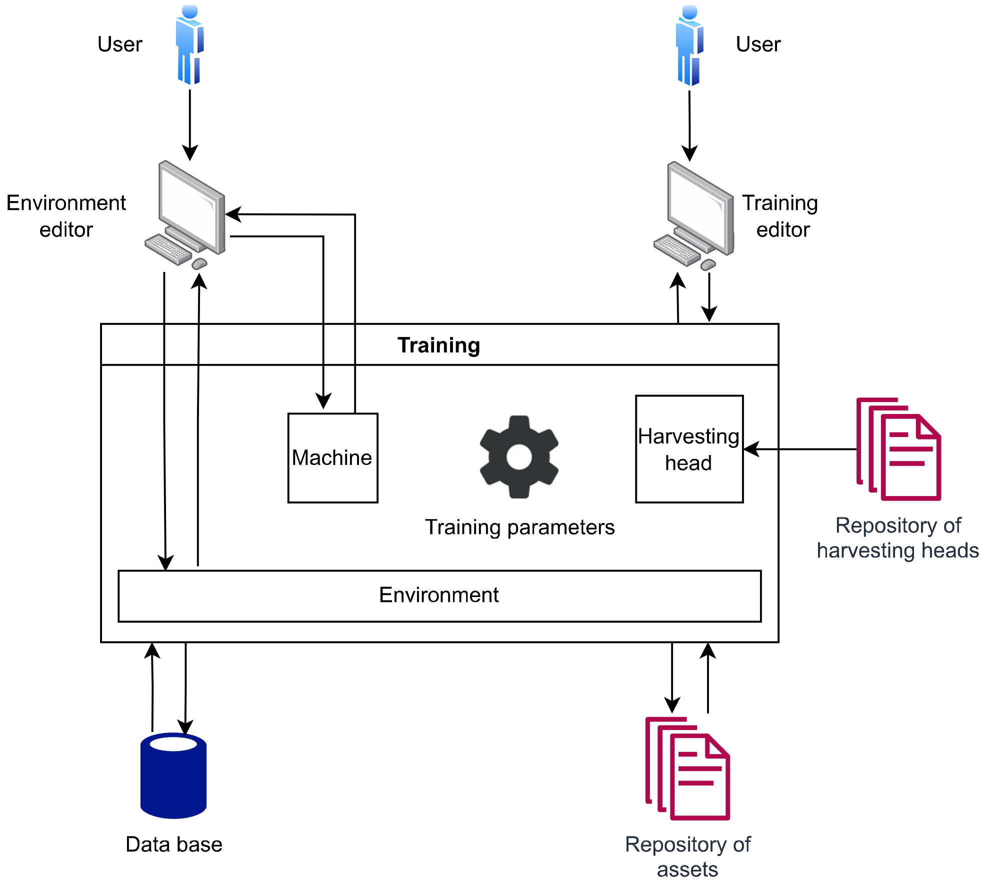


Fig. 1. Authoring tool architecture

any user to create a training where, before being executed, they can choose if they want to be in a non-immersive or an immersive environment. Because of this, this tool was divided into two editors: an environment editor and a training editor. In the environment editor, the user can create or edit several environments with some available objects and with the functionalities implemented. In the training editor, users can create or edit several courses where they can choose in which environment they want to execute the training. A database was implemented to store every necessary information to have a more accessible way to access and save the experiences.

In the environment editor, there are three main functionalities. First, the user can deform the terrain (Fig. 2) in several ways depending on the chosen mode. Despite the mode, he also can choose the area of effectiveness, represented by a white brush following the mouse movement, and the strength that will be used to deform the terrain. After that, he can raise or lower the terrain, flatten, which allows the user to choose the target height he wants to apply to deform the terrain, and smooth, which allows the user to soften some parts of the terrain. The authoring tool also allows for generating a random terrain, with all deformations in just one button. When it comes to the objects, the user can choose three different modes: selection, generate one and generate in the area (Fig. 3). In the selection mode, the user can select the objects already positioned in the terrain (the selected ones will change the colour to green). He can select one, select one by one, create a group, or select several simultaneously. After that, he can change the position and the rotation of the selected objects or even delete them. In the generate one mode, the user can select one object, and the object will follow the mouse movement until it is placed in the terrain. In the generate in area mode, the user can see a white brush that will follow the mouse. The brush size can be chosen, and the objects will be placed inside. The last functionality available is textures paint (Fig. 4), where the user can choose between vegetation or details to paint the terrain. Just like the deform terrain functionality, the user can choose an area and a strength to paint the terrain, or we can generate a detail all over the terrain with the random generation option. Ultimately, the environment can be saved to be used later on.

The training editor (Fig. 5) is more straightforward than the environment editor. In this one, the user only needs to fulfill a few fields and choose which harvesting head he wants to use in his training and in what environment he wants to execute it. One important functionality is the possibility to edit the environment associated with the training being created/edited. In this way, the user does not need to look for that specific environment if he wants to edit it. To interact with the authoring tool, the user needs to use a mouse and a keyboard, and the interfaces will be presented on a monitor.

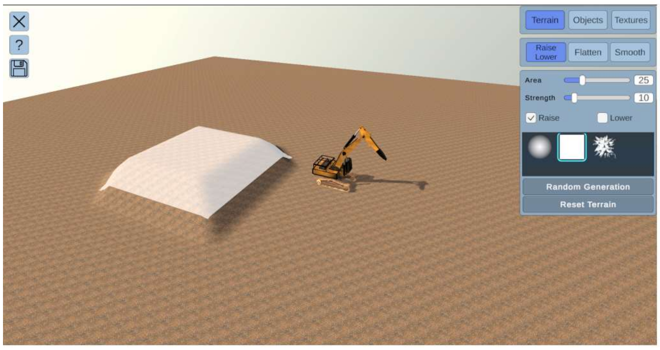


Fig. 2. Interface for editing the terrain in the environment editor.

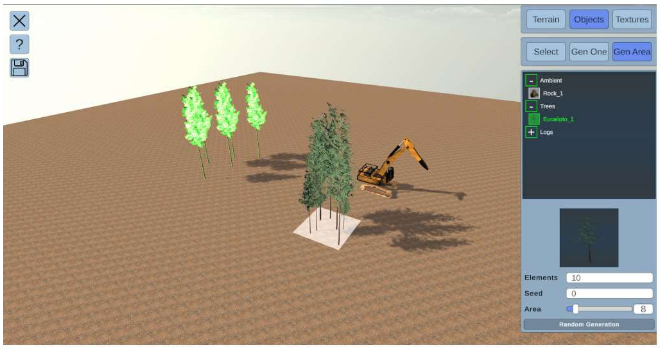


Fig. 3. Interface for editing the objects in the environment editor.

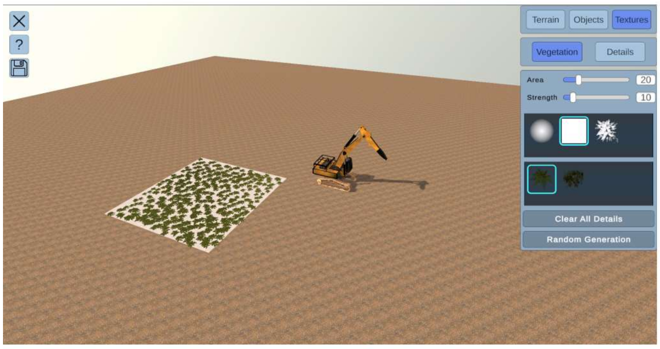


Fig. 4. Interface for editing the textures in the environment editor.

IV. Usability evaluation

A user usability study was conducted to evaluate and validate the authoring tool. Namely, usability and satisfaction evaluations were conducted for the authoring tool, as described next.

Fig. 5. Interface for the training editor.

A. Sample

Twenty participants (17 males and 3 females) tested all the authoring tool's features. The participants were between 20 and 30 years old (M = 21.3, SD = 2.20), and everyone reported normal or corrected-to-normal vision. According to the socio-demographic questionnaire introduced to the participants before they started the protocol, they all use computers in their everyday lives.

B. Instruments

To test the authoring tool and evaluate the usability of the interfaces and its functionalities, a protocol was defined with a total of 33 tasks involving authoring a virtual environment and a training scenario (28 tasks for the environment editor and five tasks for the training editor). In addition, a socio-demographic questionnaire was applied to gather information about them. As this was a study to evaluate the usability of the authoring tool, a System Usability Scale (SUS) for the overall system usability and the After Scenario Questionnaire (ASQ) for the overall satisfaction were used. The SUS scale is composed of 10 questions that evaluate the usability of the authoring tool. The ASQ scale is composed of 3 questions that evaluate the participant's satisfaction with the experience.

C. Materials

All the experiences that were made ran on a desktop computer with an Intel I7-8700K CPU, NVIDIA Geforce 2080 TI, 32GB of RAM, and Windows 10. As an interface display, the monitor used was a DELL S2340T display with 60HZ. With the desktop, a keyboard and a mouse were used to interact with the interfaces of the authoring tool.

D. Variables

The variables defined in this study were: system usability (SUS scores), experience satisfaction (ASQ scores), the number of keyboard clicks, the number of mouse clicks, the total number of clicks combined, the number of help requests, and the number of errors.

E. Procedure

The studies were conducted in a research laboratory in which the research team controlled the environment. The room has acoustic treatment and there was no noise from outside sources, so the participants could not be distracted. The first process in the experience was to receive the participants and give them a small explanation about the experience they were about to perform. It also explained the purpose of the authoring tool, why it was built and what it would be used for. After that, it was asked to the participant to read a consent paper and sign to express his agreement to participate in the study. The participant was asked to fill out a concise socio-demographic questionnaire after the signing. Next, it was given to the participant a small tutorial about one of the parts of the authoring tool (environment editor), so he would get used to the navigation of the interface and the different functionalities implemented. Following that the test protocol was given to the participant, and he was informed that if he needed any help, he could request it if he had any doubt about the functionalities of the tasks defined during the experience. During the experience, the researcher registers the number of help requests from each participant.

After completing the test protocol, it was asked to the participant to fulfil the SUS and the ASQ questionnaire. Finally, it was given the opportunity to each participant to test the environment that he created using the HMD.

F. Results

When it comes to usability and satisfaction, the values that were obtained were the following: for usability, the minimum value was 72.5, the maximum value was 100, and the average value was 87.13 (Table I); for satisfaction, the minimum value was 5.8, the maximum value was seven and the average value was 6,68 (Table I).

Regarding the number of clicks that each participant had on the keyboard, mouse, and in total, the values are between: for the keyboard, the minimum value is 83 clicks, the maximum value is 204 clicks, and the average value is 104.5 clicks; for the mouse clicks the minimum value is 91 clicks, the maximum value is 216 clicks, and the average value is 162.35 clicks; for the total number of clicks, the minimum value is 192 clicks, the maximum value is 415 clicks, and the average value is 302.85 clicks (Table II). For the number of help requests and number of errors, the values are the following: the average value for the number of help requests is 0.4, and the total is 8; the average value for the number of errors is 0.15, and the total is 3 (Table III).

V. Discussion

The values obtained from the experiences were used to see if the interaction with the authoring tool was easy to learn, understand, and interact with every functionality implemented. The results showed that all the functionalities and interfaces were easy to learn, interact with, and understand. As one can see by the usability number, this result can be found in the A percentile (if the result is bigger than 85, it can be classified as A, which equals excellent), and it tells us that the authoring tool is easy to interact with, and the participants will most likely recommend it to other users. As for the satisfaction results, the average number is superior to 6, which can be classified as positive (on a scale of 1 to 7). This can explain the low average number of help requests and errors, where the participants finished the protocol without any difficulties. The information provided to them to complete the protocol and learn the interactions with the authoring tool was helpful.

For the number of keyboard clicks, mouse clicks, and total clicks, mouse clicks are higher than the keyboard clicks. However, this should not be the case. The research team expected the number of keyboard clicks to be higher than the number of mouse clicks. The number obtained is also too high then was expected. The keyboard clicks should have been between 100 - 110 clicks, the mouse clicks should have been between

TABLE I
RESULT OF THE USABILITY AND SATISFACTION TESTS

Participants	SUS scores	ASQ scores
Participant 1	95	7
Participant 2	85	6,166666667
Participant 3	90	6,666666667
Participant 4	80	6,5
Participant 5	90	6,833333333
Participant 6	75	6,666666667
Participant 7	95	7
Participant 8	90	7
Participant 9	77,5	5,833333333
Participant 10	90	7
Participant 11	77,5	6,833333333
Participant 12	87,5	7
Participant 13	87,5	7
Participant 14	97,5	7
Participant 15	77,5	5,833333333
Participant 16	100	7
Participant 17	97,5	NR
Participant 18	92,5	6,5
Participant 19	72,5	6,166666667
Participant 20	85	7
Mean	87.13	6.68

TABLE II
NUMBER OF CLICKS OF THE KEYBOARD, MOUSE AND TOTAL

	M	SD
Keyboard	140.5	34.45
Mouse	162.35	36.99
Total	302.85	60.47

TABLE III
RESULTS OF THE NUMBER OF HELP REQUESTS AND NUMBER OF ERRORS

	M	SD
Requests	0.4	0.6
Errors	0.15	0.67

80 - 90 clicks, which will make the total between 180 - 200 clicks. These are the values that were expected if the protocol was complete almost without mistakes. During the protocol participants made mistakes and realizing what they had done, they redo some tasks which can explain why the number of clicks is higher than expected. Not all the participants had these types of results. In some situations, the keyboard clicks were higher than the mouse clicks. Some participants used the keyboard to move around the environment so they could have a full view of the terrain. Others just left the camera in the initial position. Despite almost every interaction with the authoring tool being made with the mouse, the number of keyboard clicks should still be higher than the number of mouse clicks. It was noticed, during the experiences, that some participants had some difficulties in selecting the objects (which use the mouse) that were presented in the environment. So, they redo that specific task multiple times, which explains why the number of mouse clicks is higher than the number of keyboard clicks.

One of the problems many participants refer to was the lack of objects available to position in the environment. Still, as this authoring tool is inside project SMARTCUT it was developed only to create environments that could be used to train people in the deforestation area. This makes the tool created restricted to only one area of work, so the tool will not be helpful for any creation that does not have this purpose. Besides that, the tool could be improved by adding new objects later on and new functionalities so that it could be used in other areas and would not be restricted to only one.

VI. Conclusion

The purpose of this authoring tool is to find a way of creating VR experiences without having too much trouble maintaining them functional and spending too many resources creating them. With the tools available to create these experiences, the users need to have considerable knowledge to perform these creations and make them as close as possible to reality. This tool will help users create their own experiences without spending too many resources and without needing vast knowledge. These experiences can then be executed in a non-immersive or an immersive environment (the user will have the possibility to choose that before he executes the experience).

Based on the project's requirements and the defined architecture, a prototype was built to be tested and to understand if such a tool was easy to understand and interact with every functionality. The study revealed that usability and satisfaction had a higher score, meaning the authoring tool interfaces were easy to learn and interact with. Furthermore, despite the authoring tool being restricted to the project SMARTCUT, and, as a consequence, restricted to one application area, the code that was used to create some parts of the authoring tool (where the user can create, save and edit an environment and training) can be reused and adapt to other working areas and so, the objective is to increment its functionalities and expand the application area to other areas of work where VR training can be a benefit.

To complement the authoring tool, a functionality that can improve it would be the addition of newer 3D models (assets) that can be used to populate the environment that is being created. This will make the authoring tool more expeditiously and allow the user to have more options when it comes to populate the environment and not be restricted to the ones that he already has. When a user is performing training, in the end, he should be able to see how good he was and what he needs to do to improve so. Implementing Key Performance Indicators could help the creator of the training to establish some objectives so the trainee can learn and improve in all kinds of situations.

Acknowledgment

This work is co-financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 under the PORTUGAL 2020 Partnership Agreement, and through the Portuguese National Innovation Agency (ANI) as a part of project "SMARTCUT - Diagnóstico e Manutenção Remota e Simuladores para Formação de operação e manutenção de Máquinas Florestais: POCI-01-0247-FEDER-048183".

Appendix

Protocol

- Access the environment menu.
- Choose the environment "Experimental terrain" and press the option edit terrain.
- Create a terrain using the random generation with 8555 seed.
- Select the tool raise/lower in the terrain section.
- In the frontal position of the machine, raise the terrain using an area of 50 and a strength of 65.
- Lower the terrain around the machine.
- Select the flatten tool in the terrain section.
- In the right lower corner of the terrain (use the machine as a reference), flatten the terrain to a height of zero, with an area of 20 and a strength of 10.
- Smooth the deformation created in that same corner.
- Select the category environment in the option "Objects".
- Add the available object with 1000 elements and seed 23.
- Add a trunk with the size 100 in the right lower corner that was deformed.
- Add a tree with the size 100 by the side of the machine.
- Select the option "Add multiple objects".
- Add 10 trees by the side of the previous one, with the area 10 and seed zero.
- Select all trees and try to position them inside the machine.
- Undo the previous action.
- Select the option "Textures" and the first detail.
- Add the detail to the terrain using the random generation and the strength value to the max.
- Delete the detail around the machine.
- Select the second texture.

- Add the texture underneath the machine with a strength of 100.
- Select the trunk that was added to the terrain.
- Rotate it using any axis available.
- Select two trees and delete them.
- Save the environment with the name "Experimental terrain V.2".
- Press the button "Save as".
- Close the environment terrain.
- Access the course menu and create a new course.
- Use the name "Experimental train" with the description "My first train".
- Change the schedule to five PM and choose the harvesting head 340-HD.
- Choose the environment that you just create.
- Save the course that you just create by pressing the button "Save".

REFERENCES

[1] I. Wohlgenannt, A. Simons, and S. Stieglitz, "Virtual Reality," *Business and Information Systems Engineering*, vol. 62, no. 5, pp. 455–461, 2020.

[2] C. Kirner, *Realidade Aumentada : Conceitos , Projeto e Aplicações*, 2007, vol. 1.

[3] J. van den Heuvel, J. Cunliffe, and E. Parker, "The Right Foot in the Wrong Place," pp. 1–2, 2021.

[4] H. Rodrigues Batista, P. Ricardo Batista Mesquita, and M. Antonio Gaspar, "Simulador de Realidade Virtual Aplicado à Educação Patrimonial para Experiências Imersivas Gamificadas," *Renote*, vol. 16, no. 2, pp. 230–242, 2018.

[5] T. Monahan, G. McArdle, and M. Bertolotto, "Virtual reality for collaborative e-learning," *Computers and Education*, vol. 50, no. 4, pp. 1339–1353, 2008.

[6] Y. A. G. V. Boas, "Overview of Virtual Reality Technologies," *Interactive Multimedia Conference*, p. 4, 2013.

[7] H. García-López, E. Obrero-Gaitán, A. M. Castro-Sánchez, I. C. Lara-Palomo, F. A. Nieto-Escamez, and I. Cortés-Pérez, "Non-immersive virtual reality to improve balance and reduce risk of falls in people diagnosed with parkinson's disease: A systematic review," *Brain Sciences*, vol. 11, no. 11, 2021.

[8] R. Bevilacqua, E. Maranesi, G. R. Riccardi, V. Di Donna, P. Pelliccioni, R. Luzi, F. Lattanzio, and G. Pelliccioni, "Non-immersive virtual reality for rehabilitation of the older people: A systematic review into efficacy and effectiveness," *Journal of Clinical Medicine*, vol. 8, no. 11, pp. 1–14, 2019.

[9] S. B. Moro, M. Carrieri, D. Avola, S. Brigadoi, S. Lancia, A. Petracca, M. Spezialetti, M. Ferrari, G. Placidi, and V. Quaresima, "A novel semi-immersive virtual reality visuo-motor task activates ventrolateral prefrontal cortex: A functional near-infrared spectroscopy study," *Journal of Neural Engineering*, vol. 13, no. 3, 2016.

[10] T. S. Mujber, T. Szecsi, and M. S. Hashmi, "Virtual reality applications in manufacturing process simulation," *Journal of Materials Processing Technology*, vol. 155-156, no. 1-3, pp. 1834–1838, 2004.

[11] A. Cardoso, E. Júnior, C. Kirner, and J. Kelner, "Tecnologias e Ferramentas para o Desenvolvimento de Sistemas de Realidade Virtual e Aumentada," *UFPE, Recife*, pp. 1–19, 2007.

[12] X. Liu, J. Zhang, G. Hou, and Z. Wang, "Virtual Reality and Its Application in Military," *IOP Conference Series: Earth and Environmental Science*, vol. 170, no. 3, pp. 1–8, 2018.

[13] E. A.-L. Lee and K. W. Wong, *A Review of Using Virtual Reality for Learning*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 231–241.

[14] C. Dede, M. C. Salzman, and R. B. Loftin, "MaxwellWorld: learning complex scientific concepts via immersion in virtual reality," pp. 22–29, 1996.

[15] S. Su, R. Loftin, D. Chen, and Y. Fang, "Distributed collaborative virtual environment: Paulingworld," *Proceedings of the 10th International Conference on Artificial Reality and Telexistence*, pp. 112–117, 2000.

[16] U. Technologies. (2022) Unity. [Online]. Available: https://unity.com/

[17] E. Games. (2022) Unreal engine 5. [Online]. Available: https://www.unrealengine.com/en-US/unreal-engine-5

[18] Crytek. (2022) Cry engine. [Online]. Available: https://www.cryengine.com/

[19] J. Novak, *Game Development Essentials Third Edition*, 2012.

[20] J. W. Murray, *Building Virtual Reality with Unity and SteamVR*. CRC Press, 02 2020.

[21] H. Coelho, M. Melo, J. Martins, and M. Bessa, "Collaborative immersive authoring tool for real-time creation of multisensory VR experiences," *Multimedia Tools and Applications*, vol. 78, no. 14, pp. 19 473–19 493, 2019.

[22] P. W. Butcher, N. W. John, and P. D. Ritsos, "VRIA: A Web-Based Framework for Creating Immersive Analytics Experiences," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 7, pp. 3213–3225, 2021.

Virtual reality for validation of automatic bone fracture reduction algorithms.

Juan José Jiménez-Delgado
Computer Science Department.
University of Jaén.
Campus las Lagunillas s/n, 23071 Jaén, Spain
juanjo@ujaen.es

Gema Parra-Cabrera
Computer Science Department.
University of Jaén.
Campus las Lagunillas s/n, 23071 Jaén, Spain
gparras@ujaen.es

Francisco Daniel Pérez-Cano
Computer Science Department.
University of Jaén.
Campus las Lagunillas s/n, 23071 Jaén, Spain
fdperez@ujaen.es

Augusto Silva
Department of Electronics, Telecommunications and Informatics
University of Aveiro
Campus Universitário de Santiago, 3810-193 Aveiro, Portugal
augusto.silva@ua.pt

***Abstract*—This paper presents a virtual world for bone fracture reduction. The environment is designed to facilitate interaction and immersion of the specialist in a realistic ambience improving fracture visualization and facilitating fracture reduction. Subsequently, the reduction data is collected and compared with the data provided by the automatic reduction algorithms. It is intended to provide an environment for the validation of the reductions where the expert establishes the optimal reduction. The system has been tested by specialists in the field of traumatology and computer science familiar with the subject of study and is an improvement on a first version in which the degree of realism of the interaction was lower.**

***Index Terms*—Fracture Reduction, Virtual Reality, Validation, Usability, Interaction.**

I. Introduction

Bone fractures follow a physical pattern of fracture that differs according to the conditions under which it occurs. The purpose of fracture reduction is to repair a fracture. Bone reduction does not imply any type of elimination or quantitative reduction. It is important to take in consideration that fragments of microscopic size can become detached. The bone reduction just requires the union of the larger fragments, ignoring the smaller ones. Therefore, the reduction does not imply that the fragments must be completely joined. In addition to the splitting into different fragments, the fracture often also causes a loss of alignment in the form of displacement or angulation so the specialists must also take care of these problems during the operation.

The increase in life expectancy in recent years has increased the frequency of bone fractures caused by the wear and tear of daily life. Blood interferes and also makes this difficult, and in many cases, a second surgery is required because the fragments are not aligned correctly. The older adults are particularly susceptible to hip fractures, which can have serious consequences. Therefore, fractures are a very common element of daily life that anyone can suffer so automatic fracture reduction systems are increasing in popularity.

The aim of this paper is to present a virtual reality environment for the reduction of bone fractures carried by experts and the extraction of data from it for comparison with calculations performed by automatic reduction algorithms in order to validate them.

The summary of this article is as follows. In the next section we analyze the previous work on reduction of bone fractures and medical applications in virtual reality. After the background, the materials used and the system developed are detailed. The following section presents and discusses the results obtained in the process of using the simulator for validation. Finally, the conclusions summarize the most important advances and the lines of work for the future.

II. Background

Bone fractures follow a pattern that varies according to the conditions that produce them. AO Trauma International and Orthopaedic Trauma Association Representatives (AO/OTA) presented a compendium for the classification of fractures and dislocations [1]. Figure 1 represents the main types of fractures from the AO/OTA compendium. The number of pieces of bone detached, the angle and the shape of the fracture lines are the elements that allow differentiating the different types of fractures.

Automatic fracture reduction algorithms have emerged as a solution to this challenge because during operation the specialist has little visibility over the work area. The automatic reduction algorithms focus mainly on the alignment of the generated fragments and the reduction of the distance between fragments using some modified version of the iterative closest point (ICP) algorithm. Chowdhury et al. [2], [3] proposed an algorithm based on the manual identification of some significant points in the fracture zone for fragment alignment. Other authors have presented a curvature-based procedure to obtain fracture lines through an interactive process [4]–[6]. Liu et al. [7] suggested a method based on the calculation of the

978-8-3503-4573-57/22/$31.00 ©2022 IEEE

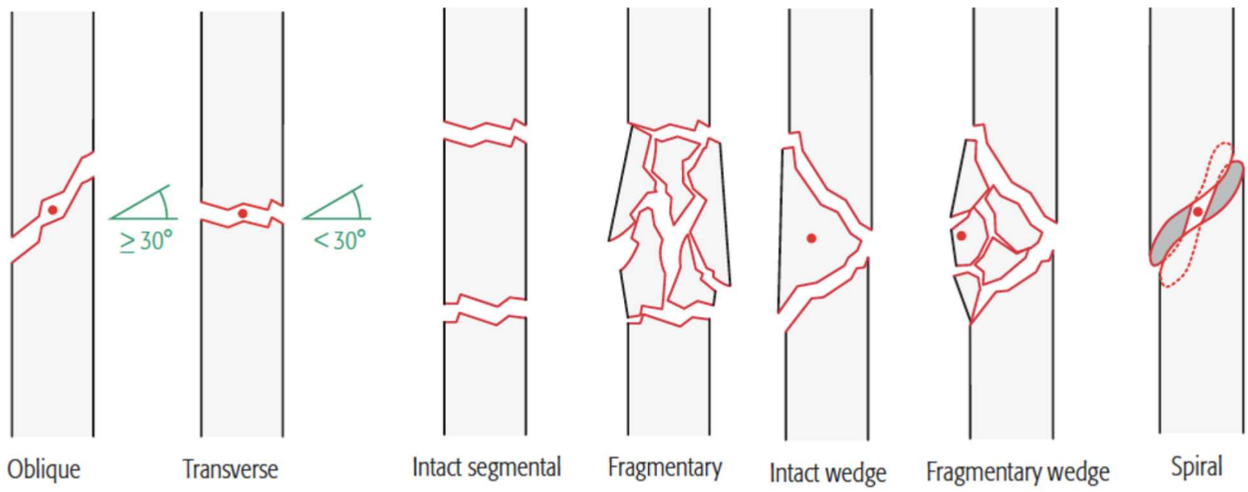


Fig. 1. AO/OTA fracture and dislocation classification [1].

bone axis and the elimination of points with deviation of the curvature and normal with respect to this axis. In a previous work, we developed a method where the entire fracture zone is automatically identified regardless of the orientation of the fragments [8].

These algorithms use as input 3D models of fractured pieces of bones. The most accurate way is to reconstruct the fracture fragments from a CT scan. Paulano et al. [9] have developed a method for segmentation of the images obtained from these scans based on the 2D region growing method [10], [11]. In contrast to the original method, the proposed method allows the identification of different fragments of the fracture although it is sensitive to certain irregularities of the bones. Fedorov et al. [12] have developed a platform for medical image computation that allows segmentation of geometric models in 3D from images efficiently. Shadid and Willis [13] have also proposed a segmentation algorithm for bone models using a probabilistic model based on the variation of the watershed transform.

The main problem encountered in fracture reduction automation work is that there is no standardized way to validate the results obtained. The most common and obvious method is visual assessment, but it is subjective and does not provide quantifiable results. Fürnstahl et al. [14] proposed a symmetrical comparison with the bone of the same patient. This approach may not always be possible since in clinical cases only the fracture zone is studied for economic and time reasons. In addition, it must be taken into account that in some cases the bones are not symmetrical or may have anomalies or trauma from other fractures that have modified their original shape. In a study performed by Paulano et al. [5] a process of validation is carried out based on the comparison of the measurements obtained after an automatic reduction and a manual reduction performed by an expert.

Tsai et al. [15] were some of the first to conclude that the visual enhancement of a virtual environment helps to improve planning of the bone fracture reduction. Citak et al. [16] emphasizes the use of new technologies to improve visualization and interaction in order to improve planning for fracture reduction and, consequently, obtain better fracture results. Gusai et al. [17] analyzed the behavior and experience of users in a virtual world in terms of interaction. Negrillo et al. [18] developed a system for fracture reduction, using a contralateral bone pattern as a target, for surgical training. Vaughan et al. [19] reviewed the main existing fracture reduction simulation environments as well as the problems of some of them.

III. Material and Method

The development of this tool has been carried out using the Unity graphics engine and the HTC Vive Pro kit made up of a VR headset and its sensors. HTC Vive is designed to be used preferably in a room, allowing you to immerse yourself in the virtual world, walk through space, and use the controllers to interact with several objects. Thanks to the base stations or lighthouse tracking system, a 360-degree virtual space of up to 15x15 feet in radius can be created. The base stations emit infrared pulses at 60 pulses per second which are then picked up by the helmet and controllers with sub-millimeter precision. That is why the use of this kit is appropriate in the simulation of the developed operating room. This hardware has allowed total immersion, by having a wide variety of sensors such as proximity or gyroscope, as well as infrared sensors that manage to determine the position in the space. It also has controllers that facilitate the control of the environment and that include feedback through vibration of the same, allowing the fragments to be manipulated in three-dimensional space. The process carried out to interact with the bone fragments is based on the user's interaction with the two controllers, which he will have in his right and left hand, respectively. The equipment we have used to work with virtual reality is a computer equipped with a first generation i7 microprocessor

Fig. 2. Scene designed to perform fracture reduction.

with 8GB of RAM, an NVidia 1060 graphics and the virtual reality glasses, HTC Vive.

The virtual environment design consists of an operating room (Fig. 2). A familiar space has been recreated for the surgeons with the aim of increasing the degree of immersion at the time of fracture reduction, as well as the objective of focusing attention on this task. The main objective of the developed system is the reduction of a fracture. For this, the system is capable of showing the user the different fragments generated in a certain fracture. The user selects the fragments, works with them by offering them the appropriate arrangement in space and, finally, performs the reduction of the fracture by joining all the fragments together.

The developed system only allows the reduction of two-fragment fractures, although in cases where a fracture with multiple fragments occurs, it can be reduced by pairs of two fragments, that means, first two fragments are reduced, one to each other, and then the obtained fragment is reduced with the remaining fragment. In addition, the bone models used during experimentation are loaded in the natural position in which the bone fracture occurred, so the user is in charge of aligning the fragments and performing the reduction.

The first time the user interacts with the fragments of a certain fracture, he has to position the first fragment in front of him in order to start the reduction task. To do this, the user selects the fragment to be treated and, by pressing the trigger of the controller on the right, positions it in front of the controller, being linked to the movement of the controller on the right(Fig. 3). If the user presses the controller's trigger on the right again, the fragment changes its state to unselected and starts following the controller on the left. This allows unselected fragments to be moved and rotated independently with the left controller, while with the right controller we can interact with other fragments that we can select. It is also possible to leave the controller in a fixed position which would reduce the complexity of the reduction since we only have to rotate and move one of the fragments to perform the reduction. The process can be viewed from various angles that improve the results of the reduction.

The reduction of a fracture is determined by the correct interaction of the user with the bone fragments, joining the fragments together. For the correct union, the user has continuous feedback from the system that, by means of the vibration of the controls, indicates if there is a collision between the fragments. In addition to vibration, when the fragments collide with each other, the system marks the area visually indicating where the collision occurs with yellow spheres Fig 4). This is done to indicate the area that is being affected by the collision. The way to mark the collision zone is through Unity's colliders (Fig. 5). An algorithm is used to create several cube-shaped colliders suited to the shape of the fracture since the default colliders in Unity do not fit to the models perfectly and the accuracy is inferior. This allows the collision to be perceived as real and avoids the overlapping of the models during reduction. In addition, it must be considered that the bones are connected to each other and that there are certain limitations when performing a fracture reduction. Every articulation of

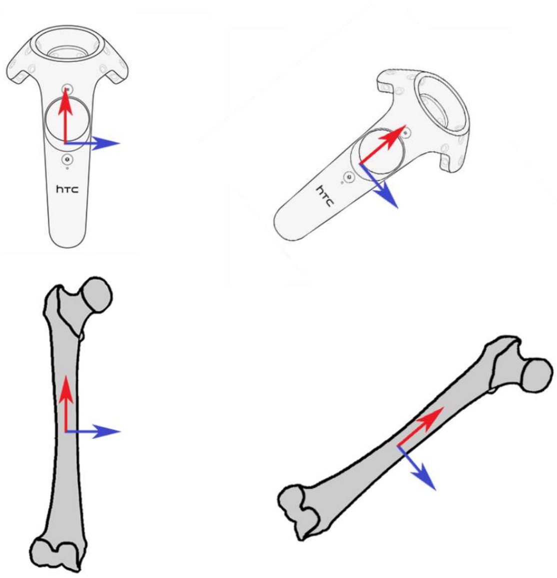


Fig. 3. Schematic representation of the interaction paradigm with bone models.

the body has a different number of degrees of freedom and there is the added problem that mobility also depends on the articulation and on the individual [20]. Consequently, in order to increase the degree of realism and to take under consideration this aspect, we have established a limitation of rotation of the fragments of 180 degrees with respect to the longitudinal axis of the fragment. These limits are applied to the initial position of the bone fragment and are applied both clockwise and counterclockwise. Therefore the limit is ± 90 degrees. As soon as this limit is exceeded, the controller starts to vibrate to alert the user that the bone is being placed in an unnatural position at bone fracture reduction. Concerning the rotation with respect to the vertical axis of the model, a rotation limitation of ± 45 degrees has also been established. As for displacement, in a real simulation both bones would be very close together, but no such constraints have been added to allow separation of the fragments and facilitate examination of the fracture zone in detail.

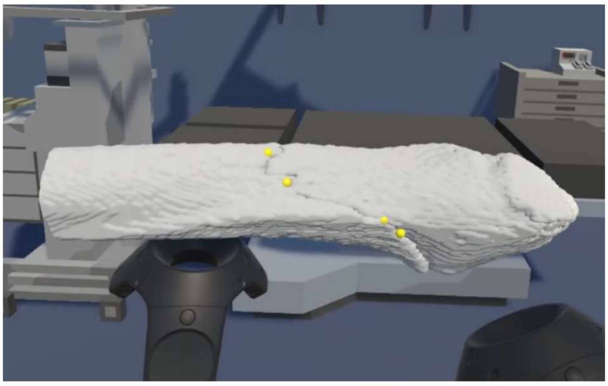


Fig. 4. Collision points are shown as small yellow spheres that generate haptic feedback in the controllers.

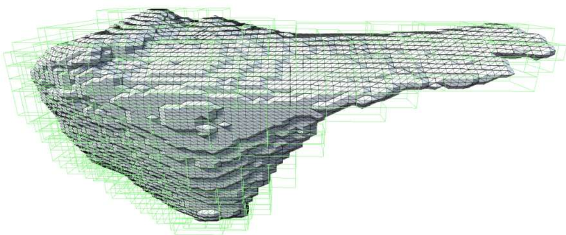


Fig. 5. Each piece of the fracture has a group of colliders generated through a shape algorithm that improves the accuracy of collision management.

IV. Results and Discussion

The previous section has detailed the development of a virtual reality system for bone fracture reduction with the objective of evaluating automatic fracture reduction algorithms. In this section we will focus on the evaluation of the results obtained and how to analyze them to validate the quality of the algorithms. For this purpose, we have relied on two experts with experience in fracture reduction, two users with experience in fracture reduction tools and a radiologist expert in the analysis of traumatological bone images. The validation of the system has been carried out through two studies. The first allows us to check the validity of the results obtained through fracture reduction carried out by experts as ground truth. The second has allowed us to verify the usability of the system.

A. Fracture reduction

To evaluate a fracture reduction, the error is calculated as the absolute value of the difference between the result obtained with a fracture reduction performed by an automatic algorithm and the reduction performed by an expert. Thus, the closer each value is to zero, the more accurate it is. For each reduction, the transformation matrix of each piece that makes up the fracture is extracted. In this way we can study the rotations and displacements that are applied to the models when aligning and reducing them. The distance error is measured as the difference between the center of mass of each fragment. The rotation error is calculated in two ways, as the average difference around the fragments three 2nd moment vectors, proposed by Paulano et al. [5], and as α and β errors, used in the work of Fürnstahl [14]. α is the difference around the two largest 2nd moment vectors and β represents the rotational difference around the smallest 2nd moment vectors. Figure 6 ilustrates the difference between a manual reduction and an automatic fracture reduction algorithm.

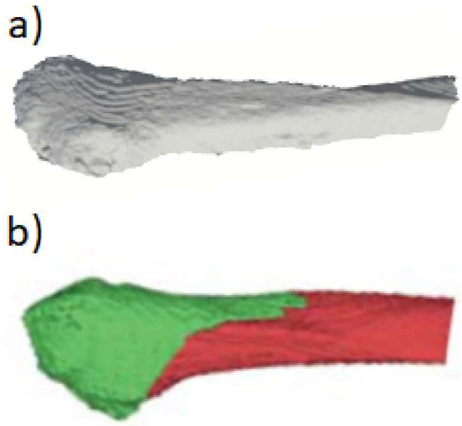


Fig. 6. Result obtained after fracture reduction performed manually in the simulator (a) and by an automatic reduction algorithm (b).

Tests were performed on fractures provided by the Complejo Hospitalario de Jaén. In these tests, a multiple fracture was included. Table I shows the average translational and rotational error for different automatic fracture reduction algorithms in the experimental stage. As shown in the results in the table, the reduction results can be improved, as they may result in faster patient recovery, although the margins of error are small and acceptable.

The error of the fracture reduction algorithms can be minimized with the measurements of this system. After achieving

TABLE I
AVERAGE RESULTS OF DIFFERENT ALGORITHMS WHEN EVALUATING ERRORS.

Study	Traslation	Rotation	Alpha	Beta
Fürnstahl et al. [14]	1,11	—	3,10	3,51
Paulano et al. [5]	1,80	3,25	—	—
Luque et al. [8]	0,58	2,41	3,30	0,92

a certain level of accuracy in fracture reduction, it is quite difficult to visually differentiate whether one result is better than another. This is why it is important to create an environment and a standardized process in which the results in fractures can be compared.

B. Usability

To verify the user experience, a two-stage process has been carried out. The first phase consists of the immersion of the user in the virtual world created without prior explanation of the workflow with the tool. This step was evaluated through an individual survey using a 5-item Likert scale. The objective was to evaluate the experience of users when they face the application for the first time. It allowed us to detect problems in the system and evaluate the quality of its features.

TABLE II
RESULTS OF THE USER EXPERIENCE SURVEY.

Feature	Rating
Image quality	4.83±0.37
Immersive sensation	4.83±0.37
Realism	3.83±0.69
Graphics fluidity	4.50±0.50
Intuitive control	3.66±0.94
Learning curve	4.33±0.47

Table II shows the average results obtained through the survey in the different aspects considered. An average higher than 4 out of 5 has been obtained. The results of the survey corroborate that the learning curve of the developed application is small. This conclusion is in line with the previous literature, which shows that the use of familiar environments facilitates the interaction with the user, in contrast to conventional user interfaces, which use a large number of buttons and menus that make them difficult to use. The worst result obtained in this section was for the degree of realism. Future versions could use more complex models as well as different textures and materials to increase the degree of realism of the environment surrounding the user. In addition, the bone models are from segmentations of bone images so the irregularities of their surface are identified with an excessive level of detail. Therefore, smoothing filters could also be applied on the models to improve the realism.

The second phase of the process consists of a brief explanation of the fracture reduction process and how to carry it out with the application. Then, the user was invited to perform different bone reductions and to fill out a new survey. This survey also was based on a 5-item Likert scale. This questionnaire was based on the study of the quality of the results obtained after the reduction, analyzing parameters such as precision, feedback, ease of the reduction process or the selection and movement of the fragments. The results obtained in this second phase are based on the perception of the users after the reduction for each bone reduction case.

Table III shows the results of the survey after carrying out fracture reductions in different bones. An average higher than 4 out of 5 is obtained, so it is verified that the perception of the users is that they can easily carry out the reduction task satisfactorily. The cases with the lowest score has been analyzed in depth. It is mainly due to the impossibility of performing a perfect fracture reduction of that model, because the model is damaged, presenting loss of bone material in some areas (Fig. 7). However, it can be seen that the system allows the fracture reduction task to be carried out as expected. In addition, although visual and vibration-based feedback to identify collisions have been included, the users have scored this feature lower than the others. Thus, we can conclude that the feedback provided can be improved and does not replace the physical contact that takes place between two real models.

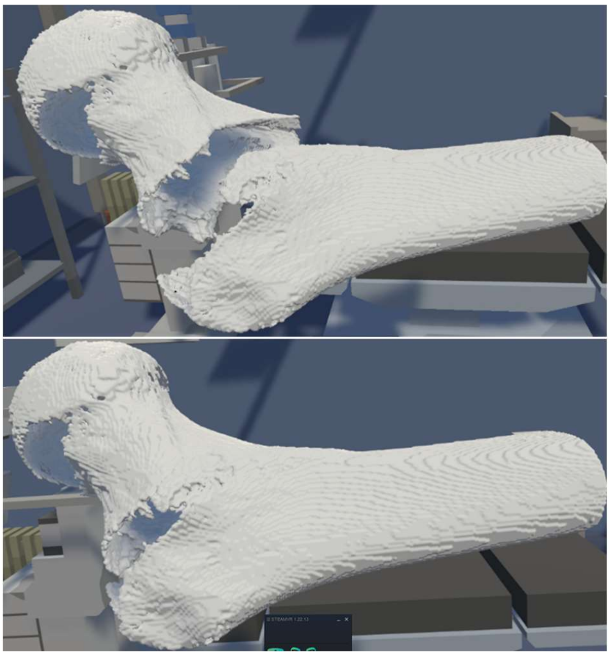


Fig. 7. Case of fracture reduction in which there are areas of bone loss.

V. CONCLUSIONS

A virtual system for fracture reduction has been presented in this work, achieving a tool capable of obtaining bone fragments united quickly and accurately. This system has made it possible to obtain a tool that allows the validation and comparison of automatic fracture reduction systems in a simple way. This is due to the fact that the implemented system obtains the matrix with the position and rotation of each of the fragments once a fracture has been reduced, as other automatic reduction algorithms do. In addition, the data obtained can also assist to identify the key problems of the

TABLE III
RESULTS OF THE QUESTIONNAIRE ABOUT THE REDUCTION PROCESS.

Fracture	Ease of selection and movement of fragments	Accuracy and feedback of the collisions	Ease of the reduction process	Accuracy
Fibula	4.66±0.62	3.83±0.99	4.16±0.69	4.16±0.69
Femur 1	4.75±0.43	3.83±0.80	3.66±0.75	4.00±0.58
Femur 2	4.66±0.47	3.75±0.43	3.33±1.11	3.50±0.76
Femur 3	4.83±0.37	3.58±0.49	3.16±0.69	3.33±0.47

algorithms under development as well as help to improve the existing ones.

The creation of a virtual environment has made the process more realistic by providing immersion and allowing the perception of collisions between the different bone fragments. The users are provided with visual feedback through coloured spheres, that indicate the points of collision, and vibrations that warn them during the fracture reduction process. However, after evaluating user experience, the freedom of movement of the controls means that the feedback should be improved because there is no physical barrier when performing a movement during the fracture reduction.

In the future, the visual quality of the fractured models will be increased, so that they look more realistic using textures and smoothing techniques. The simulation environment will also be improved, in search of a more realistic appearance of the room. Moreover, it would be interesting to advance in the creation of an external system with integrated controllers that limits the movements in a physical way to improve the user experience.

REFERENCES

[1] E. Meinberg, J. Agel, C. Roberts, M. Karam, and J. Kellam, "Fracture and dislocation classification compendium—2018," *Journal of Orthopaedic Trauma*, vol. 32, pp. S1–S10, Jan. 2018.

[2] A. Chowdhury, S. Bhandarkar, R. Robinson, and J. Yu, "Virtual craniofacial reconstruction from computed tomography image sequences exhibiting multiple fractures," in *2006 International Conference on Image Processing*, pp. 1173–1176, 2006.

[3] A. S. Chowdhury, S. M. Bhandarkar, R. W. Robinson, and J. C. Yu, "Virtual multi-fracture craniofacial reconstruction using computer vision and graph matching," *Computerized Medical Imaging and Graphics*, vol. 33, no. 5, pp. 333–342, 2009.

[4] T. Okada, Y. Iwasaki, T. Koyama, N. Sugano, Y.-W. Chen, K. Yonenobu, and Y. Sato *, "Computer-assisted preoperative planning for reduction of proximal femoral fracture using 3-d-ct data," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 3, pp. 749–759, 2009.

[5] F. Paulano-Godino and J. J. Jiménez-Delgado, "Identification of fracture zones and its application in automatic bone fracture reduction," *Computer Methods and Programs in Biomedicine*, vol. 141, pp. 93–104, 2017.

[6] J. Buschbaum, R. Fremd, T. Pohlemann, and A. Kristen, "Computer-assisted fracture reduction: a new approach for repositioning femoral fractures and planning reduction paths," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 10, pp. 149–159, Feb. 2015.

[7] B. Liu, S. Zhang, J. Zhang, Z. Xu, Y. Chen, S. Liu, W. Qi, and L. Yang, "A personalized preoperative modeling system for internal fixation plates in long bone fracture surgery—a straightforward way from CT images to plate model," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 15, sep 2019.

[8] A. Luque-Luque, F. D. Pérez-Cano, and J. J. Jiménez-Delgado, "Complex fracture reduction by exact identification of the fracture zone," *Medical Image Analysis*, vol. 72, p. 102120, aug 2021.

[9] F. Paulano, J. J. Jiménez, and R. Pulido, "3d segmentation and labeling of fractured bone from CT images," *The Visual Computer*, vol. 30, pp. 939–948, May 2014.

[10] R. K. Justice, E. M. Stokely, J. S. Strobel, R. E. Ideker, and W. M. Smith, "Medical image segmentation using 3d seeded region growing," in *Medical Imaging 1997: Image Processing* (K. M. Hanson, ed.), SPIE, Apr. 1997.

[11] J. Fan, G. Zeng, M. Body, and M.-S. Hacid, "Seeded region growing: an extensive and comparative study," *Pattern Recognition Letters*, vol. 26, pp. 1139–1156, June 2005.

[12] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J.-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F. Fennessy, M. Sonka, *et al.*, "3d slicer as an image computing platform for the quantitative imaging network," *Magnetic resonance imaging*, vol. 30, no. 9, pp. 1323–1341, 2012.

[13] W. G. Shadid and A. Willis, "Bone fragment segmentation from 3d ct imagery," *Computerized Medical Imaging and Graphics*, vol. 66, pp. 14–27, 2018.

[14] P. Fürnstahl, G. Székely, C. Gerber, J. Hodler, J. G. Snedeker, and M. Harders, "Computer assisted reconstruction of complex proximal humerus fractures for preoperative planning," *Medical Image Analysis*, vol. 16, pp. 704–720, Apr. 2012.

[15] M.-D. Tsai, M.-S. Hsieh, and S.-B. Jou, "Virtual reality orthopedic surgery simulator," *Computers in Biology and Medicine*, vol. 31, pp. 333–351, Sept. 2001.

[16] M. Citak, M. J. Gardner, D. Kendoff, S. Tarte, C. Krettek, L.-P. Nolte, and T. Hüfner, "Virtual 3d planning of acetabular fracture reduction," *Journal of Orthopaedic Research*, vol. 26, no. 4, pp. 547–552, 2008.

[17] E. Gusai, C. Bassano, F. Solari, and M. Chessa, "Interaction in an immersive collaborative virtual reality environment: A comparison between leap motion and HTC controllers," in *New Trends in Image Analysis and Processing – ICIAP 2017*, pp. 290–300, Springer International Publishing, 2017.

[18] J. Negrillo-Cárdenas, J.-R. Jiménez-Pérez, J. Madeira, and F. R. Feito, "A virtual reality simulator for training the surgical reduction of patient-specific supracondylar humerus fractures," *International Journal of Computer Assisted Radiology and Surgery*, vol. 17, pp. 65–73, aug 2021.

[19] N. Vaughan, V. N. Dubey, T. W. Wainwright, and R. G. Middleton, "A review of virtual reality based training simulators for orthopaedic surgery," *Medical Engineering & Physics*, vol. 38, pp. 59–71, feb 2016.

[20] N. Pollard, J. Hodgins, M. Riley, and C. Atkeson, "Adapting human motion for the control of a humanoid robot," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 2, pp. 1390–1397 vol.2, 2002.

Understanding of the advantages of Augmented Reality in Patients with Autism

Anabela Marto
CIIC-ESTG-Polytechnic of Leiria
Leiria, Portugal
anabela.marto@ipleiria.pt

Henrique Almeida
ESTG-Polytechnic of Leiria
Leiria, Portugal
henrique.almeida@ipleiria.pt

Alexandrino Gonçalves
CIIC-ESTG-Polytechnic of Leiria
Leiria, Portugal
alex@ipleiria.pt

***Abstract*—Augmented Reality (AR) technology has been used in a wide range of areas uncovering diverse benefits. Focusing on medicine, AR has been used for the last few decades in several different tasks – from surgery support to treatment of phobias and other cognitive or physiological disorders. This study intends to focus on the usage of AR in one specific pathology: rehabilitation of people suffering from autism. This paper will present an overview of the usage of AR technology, from a patient's perspective, where strengths and weaknesses of current approaches will be pointed out, to better understand good practices to follow. The results will provide valuable information for further ideas and solutions regarding enhancing Autism Spectrum Disorder patients' well-being.**

Keywords — Augmented Reality; Autism; Rehabilitation

I. INTRODUCTION

Augmented Reality (AR) technology, has been used in a wide range of areas, uncovering diverse benefits regarding its usage, such as publicity [1], entertainment [2], education [3], [4], architecture [5], manufacturing [6], cultural heritage [7]–[9], or medicine [10]–[12]. Within each of these areas where we can find the use of AR and its features, several approaches are noticed with different purposes. Focusing on medicine, where AR is being used for the last few decades, gaining preponderance and with some visible results to enhance several different tasks – from surgery support [10] to treatment of phobias [13], this study intends to focus on the use of AR in one specific pathology: the rehabilitation of people that suffer from autism.

Valuable systematic reviews of Virtual Reality (VR) effectiveness for particular health conditions are identified, such as brain injury rehabilitation [14] or improvements in Parkinson's patients [15]. Regarding autism, a recent systematic review of autistic children and adolescents with VR-based treatments was performed [16]. The authors concluded that most of the studies are related to the improvement of activities in daily life and communication, especially social and emotional skills.

Following the VR approach for autism patients, the current review will present an overview of the usage of AR technology, from a patient's perspective, where strengths and weaknesses of current approaches will be pointed out, to better understand good practices to follow when using AR for autism rehabilitation.

Thus, based on the works identified for this systematic review, it is intended to ascertain: 1) which studies covered the usage of AR to improve autism patients' wellbeing; 2) how did they improve autism patients' lives – identifying the rehabilitation variables, such as social skills, learnability, or ability to complete tasks; 3) what AR technologies have been used; and 4) what were the main limitations of the applied methodologies.

The results from this systematic review provide valuable information for further ideas and solutions to enhance autistic patients' well-being.

A. *Autism Brief Description*

Autism Spectrum Disorder (ASD) is a disorder related to neurodevelopment in a wide range of cognitive and physiological conditions. Being a spectrum disorder, autism affects people differently, varying from genius level to very low operation with mental retardation [17].

There are many subtypes of autism spectrum disorders. Some cases are identified with concreteness and rigidity of thinking having true anomalies in their thinking patterns, others may appear retarded due to problems with sensory processing. Autism may be divided into two extremes in a continuum: 1) Kanner/Asperger types, who have rigid, concrete thinking patterns and firm problems with particular types of cognitive processing; 2) Nonverbal, low functioning Regressive/Epileptic, who lose the power of speech and tend to have lower mental development.

Without a groundwork targeted for their condition, people with this disorder will not develop effective social skills and may speak or behave in ways creating severe challenges for everyone around them. A wide variety of medical treatment options have been developed over the years to help these individuals and the people who deal with them. This is a very active field of work and research with distinct strategies aiming to ascertain possible causes, attainable treatments, and rehabilitation approaches. Regarding this review, the usage of augmented reality for autistic patients has been revised in the current study.

B. *AR Technology Characterization*

To better understand the possibilities provided by AR, a brief characterization of this technology is necessary. While using VR, people can be fully immersed in a virtual environment, with AR applications, the virtual information is mixed with the real scenario, where they perceive both real and virtual information in what was called by Milgram et al. [18], a *Virtuality Continuum*. AR technology is used to provide a given quantity of virtual elements that are being blended with reality. Consequently, AR is a collection of interactive technologies that merge these two elements – virtual and real – in real-time, providing accurate registration in all three dimensions [19]. Several types of displays may be used to experience AR, such as head-mounted displays, hand-held displays, or spatial displays [20], [21].

The definition of AR technology helps to better understand why it has been sparking interest in its usage for autism

978-8-3503-4573-57/22/$31.00

patients. Evidence shows that children with autism suffer a deficit in sharing imaginative spontaneous pretend play, which is believed to result in reduced social interaction, known to be developmental links to key competencies of their future lives [22]. Thus, an AR technology approach is found to promote spontaneous responses in children with autism in both solitary and sociodramatic pretend play [23]. Kerdvibulvech et al. presented a three-dimensional human-computer interaction application based on AR technology for assisting children with special problems in communication for social innovation [24]. Lumbreras et al. have undergone the evaluation of the learning process through mobile augmented reality applications, to help them in their relationships with the outside world [25]. Using AR systems as part of the teaching process, the ability to perform a chain task is studied on three elementary-age students with ASD [26]. Given the difficulty that children with ASD have to recognize facial expressions and understand the associated emotions, a mobile AR application, implemented as a game, is evaluated to observe its impact on children's interaction [27].

II. Methods

The systematic review method used for the current systematic review was based on the PRISMA methodology [28] to ensure a transparent and complete reporting of the surveyed topics.

A. Eligibility Criteria

For the study, it was considered studies where AR technology was used to help or assist autism patients. The inclusion criteria are listed hereinafter.

- The paper has the one of following terms in the title, abstract, or keywords: "Augmented Reality", "AR", or "Mixed Reality", along with one of the terms "Autism" or "ASD", and with one of the terms "Rehabilitation" or "Rehab" or "Assist" or "Help" or "Aid".
- The paper is published in a refereed journal or conference.
- The paper is written in English, Portuguese, or Spanish

The search made according to the inclusion criteria revealed a long list of studies where, in some cases, they didn't fit the purpose of this systematic review. Thus, the exclusion criteria are listed hereinafter.

- The paper is not available.
- The paper is written in another language than English, Portuguese, or Spanish.
- The paper is a technical report, an abstract, a conference proceeding, a workshop, a conference review, an editor note, a call, a book, or a thesis.
- The paper does not use AR to assist/help autism patients.
- The paper does not implement the AR system, only presenting a theoretical approach to the system.
- The study does not test the application with autist patients
- The study is not targeted for the wellbeing of the patient, e.g., studies intended to evaluate usability, tolerance, acceptance, etc. of the developed app.

B. Search Strategy and Study Selection

The literature was identified through online searches, by conducting an extensive search in the databases Science Citation Index on Web of Science (Clarivate), Elsevier Scopus, ACM Digital Library, IEEE Xplore, Wiley Online Library and ScienceDirect. The search was performed to be equivalent to the following logical expression: Title/Keywords/Abstract contains ("Augmented Reality" OR "AR" OR "Mixed Reality") AND ("Autism" or "ASD") AND ("Rehabilitation" OR "Rehab" OR "Assist" OR "Help" OR "Aid").

Following the search method, whereby a total amount of 67 records were obtained, an eligibility assessment was performed independently in a conventional unblinded standardised manner by two reviewers (A.M. and H.A.). Each paper was reviewed by these two reviewers to decide its eligibility, based on the title and abstract of each study, taking into consideration the exclusion criteria. When a record was rejected by one reviewer and accepted by the other, that record was kept for eligibility.

C. Data Collection Process and Quality Assessment

Previous literature, when analysing the effectiveness of VR systems for children and adolescents with autism [16], distinguished six areas: social skills, emotional skills, daily living skills, communication ability, attention, physical activity, and phobia or fear. Thus, the eligibility assessment performed across the 16 records included the referred areas for variables analysis. When analysing the records, a new area was established, namely, cognitive skills.

According to analysed records, emotional skills are stated to improve social skills [29], thus, these two areas previously proposed are collapsed into only one area named social skills. Also included in social skills are studies targeted at social communication, with non-verbal communication, such as eye contact. The communication ability area includes specific features related to verbal communication, such as speech-language [30]. Some examples of daily living skills are studies which aimed to improve social cues when meeting and greeting [31].

Due to the intention of the current study to present a systematic review and analysis of AR implementations for autism patients' wellbeing, a quality assessment was carried out. Following the guidelines to create a quality assessment checklist [32], and based on STROBE checklists [33], a set of questions was created to which predefined answers were settled – Yes (weight of 1), Partially (weight of 0.5), No (weight of 0).

The questions established for this quality assessment were: 1) Clearly define all outcomes; 2) Clearly describe used technologies; 3) Report numbers of individuals at each stage of study; 4) Present variables to be handled in the study; 5) Describe all statistical methods; 6) Show and discuss results; 7) Report outcome events; 8) Discuss limitations of the study.

An average score of 0.8 was obtained, having an average deviation of 0.2. The high-quality papers having into account this quality assessment (with an average of 0.8 or higher) are marked with an asterisk in the table that summarises the 16 studies found applying AR for autistic patients presented in Table I.

TABLE I. A SUMMARY OF THE 16 STUDIES FOUND APPLYING AR FOR AUTIST PATIENTS.

Reference	*Patients*	*Variables*	*Technology*	*User sample*	*Positive findings*
[34] *	Children	Social skills	Smartphone	3	Yes
[35]	Children	Daily living skills	Smartphone	11	Yes
[36] *	Children	Attention	Smartphone	12	Yes
[37]	*n.s.*	Cognitive skills	Video projector	*n.s.*	Yes
[30] *	Children	Communication ability	Computer	4	Yes
[29] *	Adolescents	Social skills	Tablet	6	Yes
[38]	Children	Social skills	Computer	*n.s.*	Yes
[39] *	Children	Social skills	Smartglasses	2	Yes
[40]	Children	Social and cognitive skills	Google Cardboard	4	Yes
[41]	Children	Social skills	Computer	10	Yes
[42]	Children	Social skills	Video projector	10	Yes
[43] *	Children	Daily living skills	Computer	7	Yes
[44] *	Adolescents	Social skills and attention	Smartglasses	1	Yes
[31] *	Children	Social skills	Tablet	3	Yes
[45] *	Children and adults	Social skills	Smartglasses	18	Yes
[46] *	Children	Social skills	Smartglasses	4	Yes

n.s. *stands for "not specified".*

* high-quality papers with an average of 0.8 or higher in the quality assessment score.

III. RESULTS

The search in the identified databases returned a total of 77 records, of which 2 were identified as duplicates and were consequently removed, resulting in a total of 75 records. The title and abstracts of the unique 75 records were analysed, considering the eligibility criteria defined being that 28 records were excluded for not meetingsuch criteria. This resulted in 47 records eligible for full-text analysis. From those 47 records, 31 full-paper records were excluded based on the previously defined exclusion criteria, resulting in a total of 16 records that were included in the qualitative synthesis.

A total of 16 AR applications targeted at autistic patients were analysed, from which it was possible to extract valuable information regarding its implementation, main findings obtained, and limitations found.

A. *Qualitative and Quantitative Analysis*

According to the presented data, we observe that, as noticed in VR applications [16], most AR applications were focused on improving patients' social skills. User tests were mainly conducted with children's patients.

A graphic representation of the most evaluated variables is shown in Figure 1. None of the analysed studies was focused on physical activity nor phobia or fear. Regarding applied technologies and technological devices, Figure 2 resumes the AR setups used in the analysed studies.

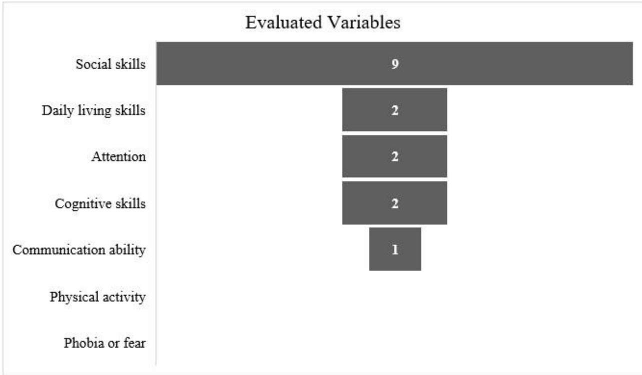


Fig. 1. Evaluated variables among the AR apps analysed.

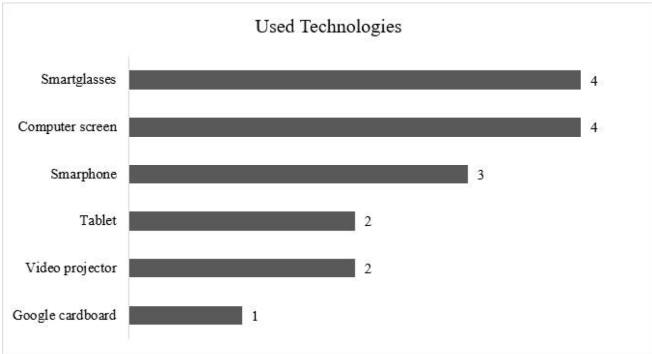


Fig. 2. Used technologies among the AR apps analysed.

B. *Discussion*

Aiming to find AR applications used for enhancing the well-being of autism patients, 16 studies were found according to the eligibility criteria. Most of these studies presented social skills as the variable to improve with the AR technology (56%). Daily living skills were identified when discriminating objects through object recognition and interaction gestures with them [35], as well as the ability to pretend to play with toys [43]. Communication ability, for verbal communication, was evaluated by analysing the benefits of speech-language [30]. Attention [36], [44], and Cognitive skills [37] which included psycho-pedagogical tasks [40], were also stated as getting improved when using AR systems.

The technologies used for the AR systems are diverse, having smartglasses (Empowered Brain [39], [44]–[46]) and Computer screens [30], [43] (specifying the use of a webcam [38], [41]) as the more common technologies used, where 4 cases of each technology were observed. Mobile devices were also common to find (3 smartphones [34]–[36] and 2 tablets [29], [31]), followed by 2 video projections [37] (specifying the use of a Kinect [42]) and one that implemented the AR using the Google cardboard [40].

The data collection was obtained through interviews (31%), video recording (22%), direct observation (17%), questionnaires (17%), and parental reports (4%). About 9% of the reviewed studies did not specify the instrument used in their evaluation. Most of the studies were focused on children and adolescents with autism (88%), with ages between 3 and 14 years old.

All reviewed records presented positive main findings regarding their presented goals when using the AR technology. Among these main findings, we found that the AR system facilitates practising, and learning social skills [31], [34], [38], [39], [44], [46], improves students' selective attention for more time [29], [30], [36], [43], helps children to learn new things [30], [37], and helps children to better understand the facial expressions and emotions [29], [41].

Limitations presented within these studies are mainly focused on involving larger samples [31], [43], [44], [46], since these studies presented sample sizes between 1 and 12 participants, with an average sample size among all scanned records is 6.8, with an average deviation of 3.9. Other limitations point out the importance of considering individual differences in further research [31], [44], and the need to explore the longer-term effects of using AR smartglasses [45].

IV. Conclusion

AR technology has been used for the last few decades in several medical applications – from surgery support to treatment of phobias and other cognitive or physiological disorders. This study focused on the usage of AR in the rehabilitation of people that suffer from autism. This systematic review presents an overview of the usage of AR technology, from a patient's perspective. Strengths and weaknesses of current approaches were identified, providing significant data to better understand how AR can be used for mitigating different distresses experienced by patients with autism.

A systematic review regarding the usage of AR applications in patients with autism is presented, with a total of 16 implementations found in the literature. All analysed records presented positive main findings regarding this technology with autistic patients among children – the most common evaluated sample found. The studies also state that social skills, communication ability, attention and cognitive skills improved with the usage of AR technologies.

Some limitations are presented, namely the influence of the individual differences in the study along with the small sample sizes from 1 single participant, up to 12.

The results from this systematic review provide valuable information for further ideas and solutions regarding enhancing ASD patients' well-being with the usage of AR technologies.

Acknowledgements

This work was supported by national funds through the Portuguese Foundation for Science and Technology (FCT) under the project UIDB/04524/2020.

References

[1] Y. Kim and W. Kim, "Implementation of Augmented Reality System for Smartphone Advertisements," Int. J. Multimed. Ubiquitous Eng., vol. 9, no. 2, pp. 385–392, 2014.

[2] M. Hamasaki, H. Takeda, and T. Nishimura, "Network analysis of massively collaborative creation of multimedia contents," in UXTV '08 Proceedings of the 1st international conference on Designing interactive user experiences for TV and video, 2008, pp. 165–168.

[3] S. C. Yuen, G. Yaoyuneyong, and E. Johnson, "Augmented Reality : An Overview and Five Directions for AR in Education," J. Educ. Technol. Dev. Exch., vol. 4, no. 1, pp. 119–140, 2011.

[4] M. Bower, C. Howe, N. McCredie, A. Robinson, and D. Grover, "Augmented Reality in Education – Cases, Places and Potentials," EMI. Educ. Media Int., vol. 51, no. 1, pp. 1–15, 2014.

[5] S. Meža, Ž. Turk, and M. Dolenc, "Measuring the Potential of Augmented Reality in Civil Engineering," Adv. Eng. Softw., vol. 90, pp. 1–10, 2015.

[6] G. Baratoff, W. Wilke, H. Artificial, and J. C. Engineering, "Augmented Reality Projects in Automotive and Aerospace Industry," IEEE Comput. Graph. Appl., vol. 25, no. 6, pp. 58–56, 2005.

[7] M. Canciani, E. Conigliaro, M. Del Grasso, P. Papalini, and M. Saccone, "3D Survey and Augmented Reality for Cultural Heritage. The Case Study of Aurelian Wall at Castra Praetoria in Rome," Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci., vol. 41, pp. 931–937, 2016.

[8] C. Fidas, C. Sintoris, N. Yiannoutsou, and N. Avouris, "A Survey on Tools for End User Authoring of Mobile Applications for Cultural Heritage," in Proceedings of the 6th International Conference on Information, Intelligence, Systems and Applications (IISA), 2015, pp. 1–5.

[9] A. Han, D., Jung, T., Gibson, "Dublin AR: Implementing Augmented Reality (AR) in Tourism," in Proceedings of the Information and Communication Technologies in Tourism, 2014, vol. 523, pp. 511–523.

[10] L. T. De Paolis, "Augmented Visualization as Surgical Support in the Treatment of Tumors," in Proceedings of the International Conference on Bioinformatics and Biomedical Engineering, 2017, pp. 432–443.

[11] E. Quintana and J. Favela, "Augmented reality annotations to assist persons with Alzheimers and their caregivers," Pers. Ubiquitous Comput., vol. 17, no. 6, pp. 1105–1116, 2013.

[12] A. Garcıa, "Interactive Augmented Reality As a Support Tool for Parkinson' S Disease Rehabilitation Programs," no. September, p. Paper 862., 2012.

[13] M. C. Juan, M. Alcaniz, C. Monserrat, C. Botella, R. M. Baños, and B. Guerrero, "Using AugmentedReality to TreatPhobias.pdf," IEEE Comput. Graph. Appl., vol. 25, no. 6, pp. 31–37, 2005.

[14] F. D. Rose, B. M. Brooks, and A. A. Rizzo, "Virtual reality in brain damage rehabilitation: Review," Cyberpsychology Behav., vol. 8, no. 3, pp. 241–262, 2005.

[15] B. Wang, M. Shen, Y. Wang, and Z. He, "Effect of virtual reality on balance and gait ability in patients with Parkinson's disease: a systematic review and meta-analysis," Clin. Rehabil. 1, vol. 1, no. 9, 2019.

[16] P. Mesa-Gresa et al., "Effectiveness of Virtual Reality for Children and Adolescents with Autism Spectrum Disorder: An Evidence-Based Systematic Review," Sensors, vol. 18, no. 8, 2018.

[17] Rutter, Michael. "Autism research: Prospects and priorities." Journal of Autism and Developmental Disorders 26.2 (1996): 257-275.

[18] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino, "Augmented reality: A class of displays on the reality-virtuality continuum," in SPIE - The International Society for Optical Engineering, 1994, pp. 282–292.

[19] R. Azuma, "A survey of augmented reality," Presence Teleoperators Virtual Environ., vol. 6, no. 4, pp. 355–385, 1997.

[20] R. Azuma, Y. Baillot, S. Feiner, S. Julier, R. Behringer, and B. Macintyre, "Recent Advances in Augmented Reality," IEEE Comput. Graph. Appl., vol. 21, no. 6, pp. 34–47, 2001.

[21] O. Bimber and R. Raskar, Spatial Augmented Reality: Merging Real and Virtual Worlds. A K Peters, 2005.

[22] Z. Bai, A. F. Blackwell, and G. Coulouris, "Making Pretense Visible and Graspable: An augmented reality approach to promote pretend play," in ISMAR 2012 - 11th IEEE International Symposium on Mixed and Augmented Reality 2012, Science and Technology Papers, 2012, pp. 267–268.

[23] Z. Bai, "Augmenting imagination for children with autism," in ACM International Conference Proceeding Series, 2012, pp. 327–330.

[24] C. Kerdvibulvech and C.-C. Wang, "A New 3D Augmented Reality Application for Educational Games to Help Children in Communication Interactively," in Computational Science and Its Applications – ICCSA 2016, 2016, pp. 465–473.

[25] M. A. M. Lumbreras, M. T. M. de Lourdes, and S. R. Ariel, "Aura: Augmented reality in mobile devices for the learning of children with ASD-augmented reality in the learning of children with autism," in Augmented Reality for Enhanced Learning Environments, 2018, pp. 142–169.

[26] D. D. McMahon, E. J. Moore, R. E. Wright, D. F. Cihak, M. M. Gibbons, and C. Smith, "Evaluating Augmented Reality to Complete a Chain Task for Elementary Students With Autism," J. Spec. Educ. Technol., vol. 31, no. 2, pp. 99–108, 2016.

[27] P. Cunha, J. Brandão, J. Vasconcelos, F. Soares, and V. Carvalho, "Augmented reality for cognitive and social skills improvement in children with ASD," in Proceedings of 2016 13th International Conference on Remote Engineering and Virtual Instrumentation, REV 2016, 2016, pp. 334–335.

[28] A. Liberati et al., "The PRISMA Statement for Reporting Systematic Reviews and Meta-Analyses of Studies That Evaluate Health Care Interventions: Explanation and Elaboration," PLoS Med., vol. 6, no. 7, pp. 1–28, 2009.

[29] C.-H. Chen, I. J. Lee, and L.-Y. Lin, "Augmented reality-based video-modeling storybook of nonverbal facial cues for children with autism spectrum disorder to improve their perceptions and judgments of facial expressions and emotions," Comput. Human Behav., vol. 55, pp. 477–485, 2016.

[30] C. A. da Silva, A. R. Fernandes, and A. P. Grohmann, "Star: Speech therapy with augmented reality for children with autism spectrum disorders," Lecture Notes in Business Information Processing, vol. 227. pp. 379–396, 2015.

[31] I. J. Lee, C. H. Chen, C. P. Wang, and C. H. Chung, "Augmented Reality Plus Concept Map Technique to Teach Children with ASD to Use Social Cues When Meeting and Greeting," Asia-Pacific Educ. Res., vol. 27, no. 3, pp. 227–243, 2018.

[32] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," 2007.

[33] University of Bern, "STROBE Statement." [Online]. Available: https://www.strobe-statement.org/index.php?id=strobe-home. [Accessed: 25- Jul-2022].

[34] L. Escobedo et al., "MOSOCO: A mobile assistive tool to support children with autism practicing social skills in real-life situations," in Conference on Human Factors in Computing Systems - Proceedings, 2012, pp. 2589–2598.

[35] E. Quintana, C. Ibarra, L. Escobedo, M. Tentori, and J. Favela, "Object and gesture recognition to assist children with autism during the discrimination training," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 7441 LNCS. Department of Computer Science, CICESE, Mexico, pp. 877–884, 2012.

[36] L. Escobedo, M. Tentori, E. Quintana, J. Favela, and D. Garcia-Rosas, "Using Augmented Reality to Help Children with Autism Stay Focused," Ieee Pervasive Comput., vol. 13, no. 1, pp. 38–46, 2014.

[37] N. S. Lakshmiprabha, A. Santos, D. Mladenov, and O. Beltramello, "An augmented and virtual reality system for training autistic children," in ISMAR 2014 - IEEE International Symposium on Mixed and Augmented Reality - Science and Technology 2014, Proceedings, 2014, pp. 277–278.

[38] C. H. Chung and C. H. Chen, "Augmented reality based social stories training system for promoting the social skills of children with autism," Advances in Intelligent Systems and Computing, vol. 486. pp. 495–505, 2017.

[39] R. P. Liu, J. P. Salisbury, A. Vahabzadeh, and N. T. Sahin, "Feasibility of an Autism-Focused Augmented Reality Smartglasses System for Social communication and Behavioral coaching," Front. Pediatr., vol. 5, 2017.

[40] K. P. Soares, A. M. F. Burlamaqui, L. M. G. Gonçalves, V. F. Da Costa, M. E. Cunha, and A. A. R. S. Da Silva Burlamaqui, "Preliminary Studies with Augmented Reality Tool to Help in Psycho-pedagogical Tasks with Children Belonging to Autism Spectrum Disorder," Ieee Lat. Am. Trans., vol. 15, no. 10, pp. 2017–2023, 2017.

[41] C. Zheng et al., "Toon-chat: A cartoon-masked chat system for children with autism," 2017.

[42] C. Zheng et al., "KinToon: A Kinect facial projector for communication enhancement for ASD children," 2017, pp. 201–203.

[43] M. Dragomir, A. Manches, S. Fletcher-Watson, and H. Pain, "Facilitating pretend play in autistic children: Results from an augmented reality app evaluation," in ASSETS 2018 - Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility, 2018, pp. 407–409.

[44] N. U. Keshav et al., "Longitudinal socio-emotional learning intervention for autism via smartglasses: Qualitative school teacher descriptions of practicality, usability, and efficacy in general and special education classroom settings," Educ. Sci., vol. 8, no. 3, 2018.

[45] N. T. Sahin, N. U. Keshav, J. P. Salisbury, and A. Vahabzadeh, "Second version of google glass as a wearable socio-Affective aid: Positive school desirability, high usability, and theoretical framework in a sample of children with autism," J. Med. Internet Res., vol. 20, no. 1, 2018.

[46] A. Vahabzadeh, N. Keshav, R. Abdus-Sabur, K. Huey, R. Liu, and N. Sahin, "Improved Socio-Emotional and Behavioral Functioning in Students with Autism Following School-Based Smartglasses Intervention: Multi-Stage Feasibility and Controlled Efficacy Study," Behav. Sci. (Basel)., vol. 8, no. 10, p. 85, 2018.

IS3TA - A mixed reality tool for exposure-based therapies

Marta Nunes
University of Coimbra
Dept. Electrical and Computer Engineering
Institute of Systems and Robotics
Coimbra, Portugal
marta.nunes@isr.uc.pt

Paulo Menezes
University of Coimbra
Dept. Electrical and Computer Engineering
Institute of Systems and Robotics
Coimbra, Portugal
PauloMenezes@isr.uc.pt

***Abstract*—*In vivo* exposure is the most common treatment for phobias, although it has several drawbacks that can be mitigated by adopting technological alternatives such as virtual reality or augmented reality. Augmented reality provides some advantages over virtual reality, including fewer modelling costs and a higher level of realism. As a result, the goal is to develop an alternate treatment to exposure *in vivo* using augmented reality with Hololens 2. The proposed approach allows the patient to interact with the phobic elements while simultaneously giving the psychologist complete control over them, allowing each person to have a unique and personalized experience based on their phobias. Implementation and preliminary analysis results are presented.**

***Index Terms*—Augmented Reality, Immersive Systems, Phobias, Anxiety, Exposure therapy**

I. Introduction

Anxiety affects 18.1% of adults, making it one of the most common mental diseases [1]. When it reaches excessive levels, it causes agitation, muscle tension, and difficulty concentrating, among other things. Some anxiety disorders are associated with phobias, which are aversions or fears of specific items or circumstances that cause significant discomfort to the individual who comes into contact with them [2] [3].

There are various treatments for these diseases, and while no single treatment works for all phobias, *in vivo* exposure is the most robust treatment applicable to most of them. In this type of treatment, the patient directly confronts his phobic stimulus to overcome it [4]. However, this therapy has many problems associated with it, such as low acceptance and high dropout rates [14], [16], complications related to the presence of the phobic element (especially if it is an animal) [11], [12], [14], the impossibility of performing this therapy in the usual office environment [7], and the therapist's inability to control the phobic elements.

To overcome the drawbacks of *in vivo* exposure, technological solutions have been introduced in therapies using virtual and augmented reality. As virtual reality (VR) creates simulations of the real-life feared environment/elements, several studies have found that using it can be just as successful as using *in vivo* exposure [4]–[8], [13], [14]. Being relatively new in this application field, through augmented reality (AR) the patient may see the real world with phobic elements added. Despite being newer, this treatment method yields positive results comparable to *in vivo* exposure [6], [9]–[12]. Both treatments minimize the problems discussed above and may allow therapists to have complete control of the phobic elements, allowing gradual and tailored exposures based on patients' anxieties [5], [9]–[11], [14]. Plus, technological solutions allow a wider variety of exercises and scenarios offered to the patient [11], [12].

In the case of exposure therapy applications, augmented reality has some advantages over virtual reality, which goes from less demanding modelling needs, to improved user experiences. In fact, in augmented reality, the environment does not need modelling, and as such, it has lower programming and modelling requirements [6], [11], [12]. On the other side, the integration of virtual items in the actual world contributes to a more natural integration and acceptance from the user's point of view, also contributing to a stronger sense of presence [9], [11].

However, both virtual and augmented reality demand caution in their application, as these aim at producing experiences that convey sensations comparable to those found in the real world. If successful they produce realistic impressions which may be in turn susceptible to inducing emotional involvement, discomfort levels, and even anxiety [10].

We can better understand which applications are best for virtual or augmented reality by simply considering that the former "transports the user to another virtual place". In contrast, the latter brings virtual elements to the user space, or eventually modifies some of the existing ones. From this, it is simple to understand that virtual reality is particularly appropriate for cases such as claustrophobia and acrophobia. On the other hand, augmented reality can be very effective for animal phobias and other elements that can be integrated naturally into the user's surroundings and familiar environment, without the risk of cybersickness.

In line with the above, this paper proposes an alternative to *in vivo* exposure that makes use of augmented reality. This solution consists of a new tool that offers the possibility of presenting a plethora of stimuli/phobic elements in a fully

This research has been partially supported by Fundação para a Ciência e a Tecnologia (FCT), under the project FCT/MCTES UIDB/00048/2020.

978-8-3503-4573-57/22/$31.00 ©2022 IEEE

controlled way, but still keeping the therapist present, aware and in control of the exposure session. This represents a step forward with respect to many other VR/AR based approaches that commonly do not enable the therapist to be a full player in control at real-time stimulation level. In our proposal, the therapist may not only tailor each session but also add or remove elements according to the observed reactions of the patients.

The remainder of this paper is structured as follows: (1) some of the most inspiring related works, (2) the proposed concept and approach, (3) the implementation and results, and the concluding remarks.

II. Related Work

A. Virtual Reality

The first alternatives to *in vivo* exposure have emerged using virtual reality. The work of Rothbaum et al. [20] (1995) was one of the first to present an experiment with virtual reality using a headset. It proposed a treatment for acrophobia using various virtual environments: various bridges, outdoor balconies at different heights, and one elevator simulation. Although there were no treatment comparison group and the subjects were not patients seeking treatment, the study successfully showed that virtual reality could be used to treat anxiety disorders.

In Côté and Bouchard's work [21] (2005), the authors focus on treating arachnophobia with a VR environment consisting of 2 apartments with many rooms with different levels. There are few static spiders in the lower levels, while the spiders were bigger and moving in the most difficult levels. The study showed improvement in participants' phobia symptoms.

Pitti et al. [22] (2015) presented an alternative treatment to agoraphobia with VR, using polarized glasses and projecting the image on a special screen. In this experience, there were various scenarios: a square and street, an airport building and plane, a bank office, an elevator and underground car park, a beach, a highway, and a cableway. This study concludes that the patients that received combined treatment (VR and psycho drugs) improved significantly more than the others. However, the dropout rates with VR and *in vivo* exposure were similar.

There are several other examples of the use of virtual reality to this end, but with the appearance of usable augmented reality headsets, these have lately received increasing interest in particular due to the fact that it enables to explore the familiarity of the local environment to avoid unnecessary anxiety induced just by the unfamiliar virtual environments themselves.

B. Augmented Reality

Juan et al. [23] (2005) presented one of the first attempts to use AR as an alternative to *in vivo* exposure, using a VR HMD and a camera connected to a computer. This work, although exploratory, already supported the idea of having the therapist watch the exposure scenario on the computer monitor. The virtual elements used were spiders (three types: small, medium, and large) and cockroaches (one type), and these could move, be static, or even dead, if the patient killed them. The interaction provided to the therapist was based on markers, controlling this way where the objects would appear, controlling how many animals appear, their size, and movements. The patients to "kill" the animals also needed a specific marker.

Bretón-López et al. [25] (2010) developed an experience using an AR system to treat cockroach phobia. The system ran on a PC, and similarly, the video was captured by a camera attached to an HMD and connected to the PC via USB. The therapist could control the cockroaches' number, movement, and sizes. The experience reported good results, however, the intricate cabled connection caused discomfort and tiredness.

The work of Wrzesien et al. [11] (2011) and Botella et al. [12] (2016) also aimed to treat spider and cockroach phobias using AR solutions. The patients could observe and interact with the scene through HMDs, and the therapists controlled the phobic triggers' sizes, number, and movement. Despite the success of the therapy, [11] it is mentioned that the systems have functions (like the non-optimal user-interface dialogue) that are not useful or not optimal, and [12] tiredness and dizziness were reported due to the HMD.

De Witte et al. [10] (2020) focus on ensuring the validity of using an AR application for smartphone or tablet to induce fear and anxiety, an important prerequisite for effective treatments with AR. The user could choose between various animals (rats, pigeons, snakes, cockroaches, spiders, frogs, and dogs). The study shows that the application induces fear and anxiety, even though the use of a smartphone or a tablet also reduces the realism and perception of the virtual elements, and makes it difficult to interact with these elements.

Patrão, Menezes and Gonçalves [24] (2020) presented an AR experiment with spiders. The application uses tablets, one for the therapist and another for the patient, and both can control and add elements to the environment. The study reveals good usability and interaction, yet, the use of tablets reduces the perception of the virtual elements and makes it difficult the interaction with them.

Wrzesien et al. [26] (2013) approach AR therapy with a new device, the Therapeutic Lamp (TL), which projects the elements on a surface. This experiment focuses on cockroach and spider phobias, with three types of each animal, and the therapist controls the number, size, and movement of the elements. The experience shows that the TL is an effective tool for small animal phobia treatment, however, the fact that it uses projection instead of holograms can affect the sense of presence and realism.

After reviewing these articles, we realized that in many of the approaches for therapy with AR, the devices used are also not always the most appropriate, as tablets, smartphones, and TL are likely to affect the realism of the therapy elements and the sense of presence. Previously used HMDs can become uncomfortable and cumbersome due to their weight and wires. Our alternative, presented below, is therefore intended to overcome these problems.

III. Proposal

As with other papers, the goal of this proposal focuses on developing a passive augmented reality solution to be used in a clinical setting to replace *in vivo* exposure. This resulted in the development of an *Immersive System for Sensory Stimulation and Therapeutic Applications (IS3TA)* that is expected to enable the therapist to bring into the office space virtual versions of elements associated with specific phobias of each patient. This system should be integrated into a typical therapeutic office space, where a table can be used in front of the patient, on which the exposure elements will appear, and with which the patient will be asked to approach (when deemed appropriate by the therapist), or even interact. This approach may also allow us to explore the surrounding space, floor, walls, and ceiling to introduce these types of elements. For example, it is more natural a dog to appear on the floor than on a table, whereas a spider might be on the wall or on the ceiling.

A. Putting the therapist in control

One of the innovations introduced in this approach is the possibility that the therapist can control the whole exhibit through a computer, selecting the elements that should appear, their location and characteristics, etc. Similarly, they can be removed if it is found that there is some overexposure effect. It will also be possible to choose the most appropriate phobic element for each patient, allowing a single device to be versatile for several patients with different phobias.

B. Integrating the therapist's workspace

One of our objectives is to take advantage of the space where the therapeutic session takes place and integrate the virtual triggering elements in an ecological way. This is to say that the virtual elements must be added in a coherent way to the space and existing furniture. For the therapist to take control, as above referred, there must be a common reference frame between his/her view and the patient's. This common frame enables the introduction of a virtual table on the therapist's view that corresponds to the physical table viewed by the patient. This allows the therapist to precisely choose where to place the virtual elements on this surface. As the 3D reconstruction of the environment in real-time can require a significant amount of computational power and may result in the appearance of disturbing artefacts, the choice fell on the superposition of models with some of the furniture elements. This way these elements will be represented in the therapist's view so that they can be used to precisely place those referred elements, while "invisible" in the patient's, as there is already the physical counterpart. This idea is accomplished with the use of fiducial markers (in this case, QR codes).

C. Solution Architecture

The solution is divided into two parts: the patient's device and the therapist's device. First, the patient is able to see his phobic triggers through an augmented reality device. In our case, the Hololens 2 (figure 1) was picked since it is wire-free, lightweight, portable, and provides a high level of immersion. Besides these advantages it allows us to explore other aspects such as hand tracking capabilities, built-in voice commands, eye tracking, and spatial mapping.

Fig. 1. A user interacting with AR via Hololens 2.

On the other hand, the psychologist can control everything the patient sees in real-time. This includes managing the type, quantity, and movement of the phobic aspects visible to the user at every moment, allowing the creation of experiences tailored to each patient's needs. With the ability to choose the type of phobic trigger presented to the patient, the treatments of numerous phobias will also be possible with the same equipment.

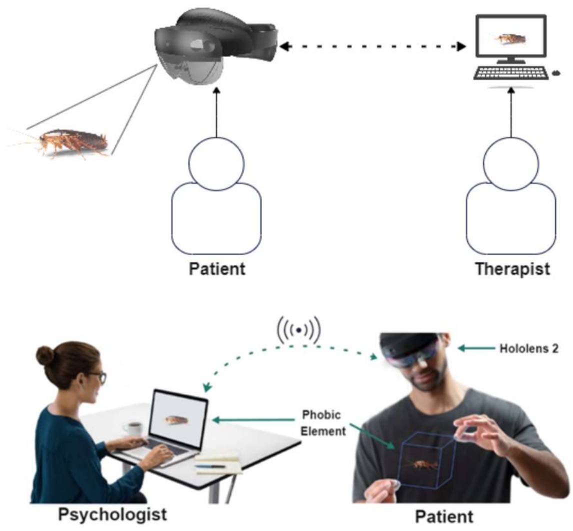


Fig. 2. Illustration of the principles supported in the application.

Figure 2 shows two similar diagrams portraying the application's division into the two parts explained above. The patient has an immersive experience using the Hololens 2, viewing his phobic element in numerous ways. The psychologist controls everything the patient sees via a computer.

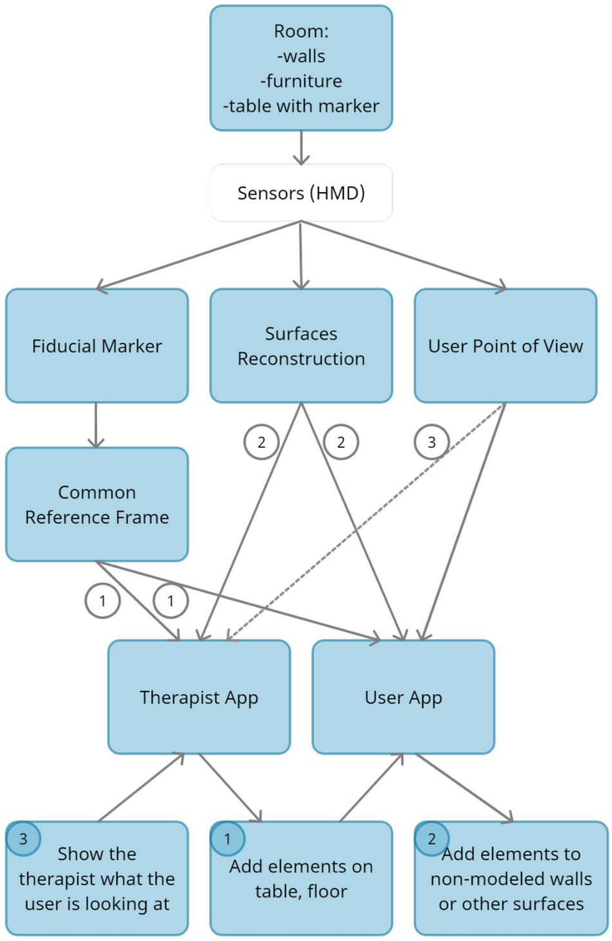


Fig. 3. Flow diagram of the solution.

In figure 3 we can see the flow diagram of our solution. The HMD and its sensors can acquire a model of the room, the user's point of view, and perform surface reconstruction. The application includes fiducial marker detection and poses extraction to support the establishment of a common reference frame for both the HMD and the desktop applications. The reconstruction of the surfaces gives realism to the scene because the phobic elements can be put realistically on the objects in the room. The user's point of view is constantly monitored and used for the therapist to analyse in real time what is in the patient's field of view and if he/she tries to avoid looking at, or staring at some of the elements. All these features support this therapeutic system composed of the two already mentioned applications. This is made possible through the inclusion of appropriate communication support that will be described later. By putting all together, the virtual elements may be added, removed and controlled by the therapist, according to the interaction and intended stimulation for each patient at each specific moment of the exposure sessions.

IV. Implementation and Results

With the solution explained before in mind, we will now look at how we are carrying out the implementation of the project, explaining the modules used to put the ideas discussed into practice.

A. Development

As already mentioned, the target device chosen was Microsoft Hololens 2, which has an interesting set of sensors and capabilities, accessible through the Mixed Reality Toolkit (MRTK). The development for this device can be made directly using MRTK, or using a third party platform commonly used in game development such as Unreal or Unity. The choice fell on the latter, mostly driven by the existent experience within the research group. The development based on unity can take advantage of the vast number of assets available, both proprietary and opensource that support different types of functionalities like model imports, communications, or others. Examples of these assets are those provided by hardware vendors that support the development of applications for their specific platforms through some sort of hardware abstraction layer that hides the particular details. These assets, therefore, provide the necessary interfaces for the developer to extend them through his/her own assets and scripts.

These include scripts and components that help create mixed reality apps in an easy and quick way. This gives the developer freedom to use them or write his/her own assets and scripts, in the current case, build support for multiple device communication and the establishment of consistent points of view between an AR application and a 3D screen-based application, so the therapist can control the timing, location and intensity of stimuli generation.

B. Multiplatform communication

To allow communication between Hololens 2 and the therapist's computer, we are using Unity Multiplayer Networking Netcode, a Unity networking library that allows faster creation of networked applications. This uses UNet transport that relies on UDP transport layer to support the timed synchronization of the information between the devices.

Usability and flexibility concerns led to the implementation of a local network device discovery, to enable highly flexible usage of the setup, without relying on directory/game servers that would need permanent Internet connectivity. This discovery step is based on an initial broadcast of a discovery message initiated by the AR device (the client), expecting to receive a response from the therapist's computer (the server). After this initial phase, the peers are mutually identified, the connection is established supporting all the necessary communications used to place/move/destroy the phobic elements, parameter change, or other.

C. Establishment of a common reference frame

To enable the therapist to have a view of the patient's workspace/environment and interact with it while disposing of the elements as appropriate, a common reference frame should be established. It became also clear that some of the important physical elements should have a digital representation to make the above interaction possible. As the Hololens device establishes its own coordinate frame, it is necessary to infer one transformation per physical object that will have a virtual counterpart representation in order to support the proper display of the virtual representations in the therapist's view. Starting with a table, typically placed between the patient and the therapist, and where most of the interaction and exposure will be happening, a simple marker purposely placed on it can be enough support to the alignment of the digital model. This marker is detected on the images captured by one of the device's cameras and its pose estimation is consequently performed, enabling the necessary adjustment to the coherent placement of the virtual elements in front of the patient, through the therapist interface. In reality, although the marker detection and pose estimation requires that the patient "looks at it", once detected, and given that the Hololens has a quite stable pose estimation, the relative transformation is not expected to change even if the marker does not return into the camera field of view for a period of time.

D. *Addition of the patient's hand reference point*

Since the therapist cannot directly see the patient's view, it is difficult for the therapist to analyze the patient's interaction with the phobic elements. To mitigate this problem, it was necessary to detect the poses of the patient's hands and pass them to the therapist's application. Using this, the therapist observes in real time the patient's hands and their interaction with the phobic elements. In this way, the therapist can better analyze the patient's reactions and interactions with the phobic elements, adapting the exposure as needed.

E. *Results and preliminary analysis*

In figure 4 we can observe the result of the implementation explained above. In this image, we can understand the therapist's view of the exposure surface, and the patient's interactions, and therefore chose where to place the phobic elements. Both user and therapist can interact with the virtual objects, knowing that the position is updated for both perspectives. The fiducial marker is placed on an out of sight corner, and does not interfere with the experience. This is possible since the detection of the marker is only required in the beginning for the establishment of a common reference frame for the exposure table.

The design of the patient-side application was made to be simple and to minimize distractions for the patient, allowing the user to focus on the phobic elements, and on the therapist's instructions. The goal of the therapist-side application is to be easy to operate and intuitive. So, to add the animals just right-click on the place where you want them to appear. To remove and move the animals, we perform the same operation on the animal we want to control at the moment. In figures 5 and 6 we can see the view from the patient with the two animals current present in the solution, frogs and spiders. The developed framework enables the easy addition of other types of phobic-triggering elements.

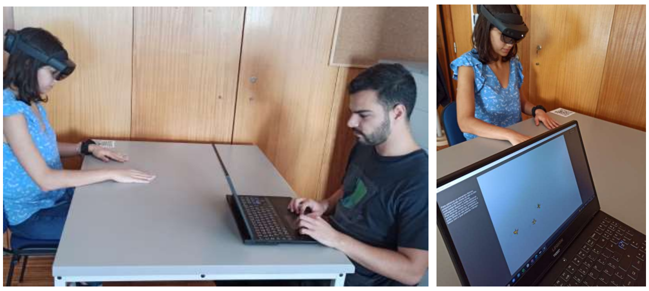


Fig. 4. Representation of a possible therapy session with a patient using IS3TA on Hololens 2, whose exposure to the phobic elements is controlled by the therapist through the personal computer interface.

Fig. 5. View from Hololens 2 with spiders as the phobic elements.

Although the system is still under development it was considered important to validate its current state. For that a preliminary study was made, where eleven non-phobic people were asked to try the AR application (i.e. the patient side of the solution) and answer a questionnaire about the immersion and quality of the AR environment. Although this test was performed with a reduced number of participants, it follows the ideas of Nielsen and Landauer [28], who stated that with only 5 users it is possible to find 85% of the problems in interactive

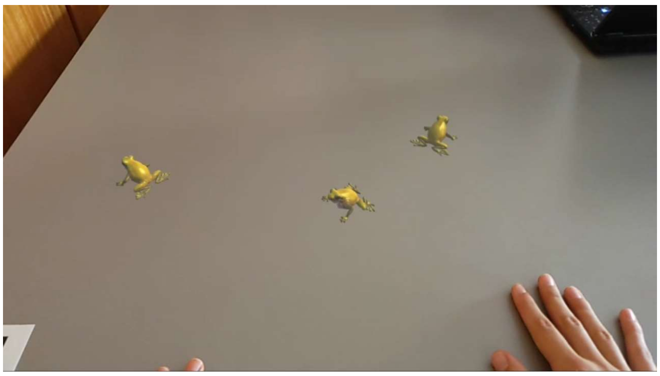


Fig. 6. View from Hololens 2 with frogs as the phobic elements.

systems. In fact, early identification of potential problems typically brings an enormous advantage, as corrections tend to be simpler to apply at this phase, rather than over a final product. The questions for this evaluation were chosen from the ”reality judgement and presence questionnaire” of Regenbrecht and Schubert [27] and are presented in the table below.

Q1 - Was watching the virtual objects just as natural as watching the real world?	1= No	5= Yes
Q2 - Did you have the impression that the virtual objects belonged to the real object, or did they seem separate from it?	1= Separate objects	5= Same object
Q3 - Did you have the impression that you could have touched and grasped the virtual objects?	1= No	5= Yes
Q4 - Did the virtual objects appear to be on a screen, or did you have the impression that they were located in space?	1= On screen	5= In space
Q5 - Did you have the impression of seeing the virtual objects as merely flat images or as three-dimensional objects?	1= Flat images	5= Three-dimensional objects
Q6 - Did you pay attention at all to the difference between real and virtual objects?	1= I Saw no difference	5= The difference was clear
Q7 - Did you have to make an effort to recognize the virtual objects as being three-dimensional?	1= Yes an effort was needed	5= No effort was needed

As explained before, the study was made with ten males and one female divided into two groups: the first group of six people who interacted with frogs and the second group of five people who interacted with spiders. The participants have ages ranging from 20 to 30 years old. The conditions of light were similar, and all people saw three animals on a table (first static and then animated to look alive). The Hololens 2 was also calibrated before every test to adjust the view of the models to each user. Figure 7 shows the average and standard deviation plots computed for the answers to the questionnaires separated into three series: total, frogs only, and spiders only.

From the analysis of the results obtained, we can conclude that the models are realistic and produce the desired effect of appearing to be part of the environment. From the observation of the values of questions Q1, Q2, Q3, and Q5, we can observe that the participants found the objects to have a three-dimensional and natural appearance, well integrated into the environment, and likely or appealing to be touched. There was no visible difference in the results between the two animal models used in the trial. Regarding question Q4, the volunteers who interacted with the frog models seemed to be somewhat aware of the screen used to create the holographic objects. When analyzing the values of question Q6, we understand that the difference between real objects and virtual objects was clearly perceived, however, the impact of this in the therapy has yet to be studied in a therapeutic context. At last, reviewing the results obtained in question Q7 we can conclude that, the virtual objects were recognized as being three-dimensional, especially for the spider models. In addition to these answers, it was also observed that as soon as the volunteers saw the models, most of them had an instinctive reaction to try to touch or smashed them, showing that the animals seem realistic.

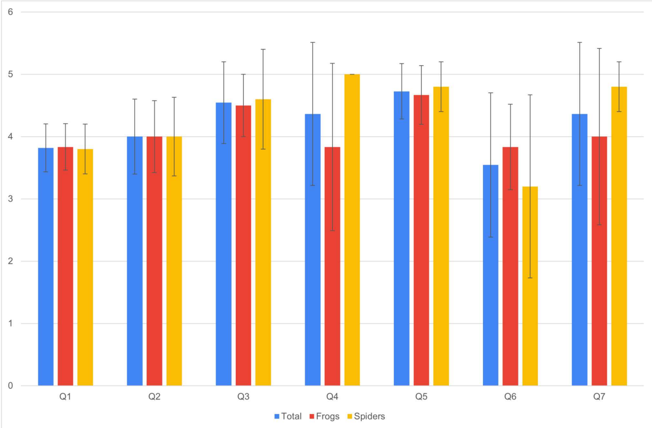


Fig. 7. Average and standard deviation values for all the answers, frogs-only answers, and spiders-only answers.

One of the aspects reported, although already known to us, was the fact that the perception of the virtual elements depends on the ambient lighting. In reality, the holographic nature of the device is based on ”adding light”, and does not dispose of any means to block the light that comes from the objects. As a result, if the environment itself is too bright, the virtual elements tend to fade out.

Although this study does not address the therapeutic application context, it already brings some exciting results. The forthcoming steps will include the analysis made by a therapist and then, after incorporating possible recommendations, a clinical study will also be performed.

V. Conclusion

This paper proposes a solution to replace the exposure *in vivo*, using augmented reality. This method can suppress the problems associated with *in vivo* exposure, like the low acceptance and high dropout rates, the complications related to the presence of the phobic trigger and the therapist’s inability to control it, as well as the impossibility of performing this therapy in the usual office environment.

Despite not being yet tested on people with phobias, this work shows promising results. In future work, we plan to improve it by adding additional features, like eye tracking and improved interaction.

References

[1] R.C. Kessler, W. T. Chiu, O. Demler, K. R. Merikangas, E. E. Walters,"Prevalence, Severity, and Comorbidity of 12-Month DSM-IV Disorders in the National Comorbidity Survey Replication", Archives of General Psychiatry, vol. 62, pp. 617-627, June 2005.

[2] U.S. Department of Health and Human Services,"Anxiety disorders", National Institute of Mental Health, Last accessed 4 January 2021, 2018, Avaliable online: https://www.nimh.nih.gov/health/topics/anxiety-disorders.

[3] S. Dattani, H. Ritchie, M. Roser,"Mental Health", Our World in Data, 2021.

[4] Y. Choy, A. J. Fyer, J. D. Lipsitz, "Treatment of specific phobia in adults", Clinical Psychology Review, vol. 27, pp. 266-286, 2007.

[5] N. Morina, H. Ijntema, K. Meyerbröker, P. M.G. Emmelkamp,"Can virtual reality exposure therapy gains be generalized to real-life? A meta-analysis of studies applying behavioral assessments", Behaviour Research and Therapy, vol. 74, pp. 18-24, 2015.

[6] C. Suso-Ribera, et al.,"Virtual Reality, Augmented Reality, and In Vivo Exposure Therapy: A Preliminary Comparison of Treatment Efficacy in Small Animal Phobia", Cyberpsychology, Behavior, and Social Networking, vol. 22, pp. 266-286, 2018.

[7] P.M.G. Emmelkamp, M. Krijn, A. M. Hulsbosch, S. de Vries, M. J. Schuemie, C. A. P. G. van der Mast,,"Virtual reality treatment versus exposure in vivo: a comparative evaluation in acrophobia", Behaviour Research and Therapy, vol. 40, pp. 509-516, 2002.

[8] T. D. Parsons, A. A. Rizzo,"Affective outcomes of virtual reality exposure therapy for anxiety and specific phobias: A meta-analysis", Journal of Behavior Therapy and Experimental Psychiatry, vol. 39, pp. 250-261, 2008.

[9] I. C. Giglioli, F. Pallavicini, E. Pedroli, S. Serino, G. Riva, "Augmented Reality: A Brand New Challenge for the Assessment and Treatment of Psychological Disorders", Computational and Mathematical Methods in Medicine, pp. 1-12, 2015

[10] N. A. J. De Witte, S. Scheveneels, R. Sels, G. Debard, D. Hermans, T. Van Daele, "Augmenting Exposure Therapy: Mobile Augmented Reality for Specific Phobia", Frontiers in Virtual Reality, vol. 1, pp. 8, 2020.

[11] M. Wrzesien, J. Burkhardt, M. Alcañiz, C. Botella, "How Technology Influences the Therapeutic Process: A Comparative Field Evaluation of Augmented Reality and In Vivo Exposure Therapy for Phobia of Small Animals", Springer Berlin Heidelberg, pp.523-540, 2011.

[12] C. Botella, W. A. Pérez-Ara, J. Bretón-López, S. Quero, A. García-Palacios, R. M. Baños, "In Vivo versus Augmented Reality Exposure in the Treatment of Small Animal Phobia: A Randomized Controlled Trial", PLOS ONE, vol. 11, pp. 1-22, February 2016.

[13] C. Botella, J. Fernández-Álvarez, V. Guillén, A. García-Palacios, R. Baños, "Recent Progress in Virtual Reality Exposure Therapy for Phobias: A Systematic Review", Current Psychiatry Reports, vol. 19, pp. 1-12, 2017.

[14] D. Boeldt, E. McMahon, M. McFaul, W. Greenleaf, "Using Virtual Reality Exposure Therapy to Enhance Treatment of Anxiety Disorders: Identifying Areas of Clinical Adoption and Potential Obstacles", Frontiers in Psychiatry, vol. 10, pp. 773, 2019.

[15] A. Garcia-Palacios, C. Botella, H. Hoffman, S. Fabregat, "Comparing Acceptance and Refusal Rates of Virtual Reality Exposure vs. In Vivo Exposure by Patients with Specific Phobias", CyberPsychology & Behavior, vol. 10, pp. 722-724, 2007.

[16] B. O. Olatunji, B. J. Deacon, J. S. Abramowitz,"The Cruelest Cure? Ethical Issues in the Implementation of Exposure-Based Treatments", Cognitive and Behavioral Practice, vol. 16, pp. 172-180, 2009.

[17] J. Szymanski, W. O'Donohue,"Fear of Spiders Questionnaire", Journal of Behavior Therapy and Experimental Psychiatry, vol. 26, pp. 31-34, 1995.

[18] R. M. Baños, C. Botella, M. Alcañiz, V. Liaño, B. Guerrero, B. Rey, "Immersion and Emotion: Their Impact on the Sense of Presence", CyberPsychology & Behavior, vol. 7, pp. 734-741, 2006.

[19] A. Bangor, P. T. Kortum, J. T. Miller,"An Empirical Evaluation of the System Usability Scale", International Journal of Human–Computer Interaction, vol. 24, pp. 574-594, 2008.

[20] B. O. Rothbaum, L. F. Hodges, R. Kooper, D. Opdyke, J. S. Williford, M. North, "Effectiveness of computer-generated (virtual reality) graded exposure in the treatment of acrophobia." American Journal of Psychiatry, vol. 152, pp. 626-628, April 1995.

[21] S. Côté, S. Bouchard, "Documenting the efficacy of virtual reality exposure with psychophysiological and information processing measures." Applied Psychophysiology and Biofeedback, vol. 30, pp.217-232, September 2005.

[22] C. T. Pitti, et al., "The combined use of virtual reality exposure in the treatment of agoraphobia.", Actas Esp Psiquiatr, vol. 43, pp. 133-141, July-August 2015.

[23] M. C. Juan, M. Alcaniz, C. Monserrat, C. Botella, R. M. Baños and B. Guerrero, "Using augmented reality to treat phobias," IEEE Computer Graphics and Applications, vol. 25, pp. 31-37, November-December 2005.

[24] B. Patrão, P. Menezes, N. Gonçalves, "Augmented Shared Spaces: An Application for Exposure Psychotherapy.", International Journal of Online and Biomedical Engineering (iJOE), vol. 16, pp. 43-50, 2020

[25] J. Bretón-López, S. Quero, C. Botella, A. García-Palacios, R. M. Baños, M. Alcañiz, "An Augmented Reality System Validation for the Treatment of Cockroach Phobia." Cyberpsychology, Behavior, and Social Networking, vol. 13, pp. 705-710, December 2010.

[26] M. Wrzesien, M. Alcañiz, C. Botella, J. M. Burkhardt, J. Bretón-López, M. Ortega , D. B. Brotons, "The therapeutic lamp: treating small-animal phobias." IEEE Comput Graph Appl, vol. 33, pp. 80-86, January-February 2013.

[27] H. Regenbrecht, T. Schubert, "Measuring Presence in Augmented Reality Environments: Design and a First Test of a Questionnaire", arXiv, 2021.

[28] J. Nielsen, T. K. Landauer, "A mathematical model of the finding of usability problems. In Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93). Association for Computing Machinery", pp. 206-213, 1993.

3D-Based Pairwise Color Correction Approach for Texture Mapping Applications

Lucas Dal'Col
IEETA, DEM
University of Aveiro
Aveiro, Portugal
lucasrdalcol@ua.pt

Daniel Coelho
IEETA, DEM
University of Aveiro
Aveiro, Portugal
danielsilveiracoelho@ua.pt

Tiago Madeira
IEETA, DETI
University of Aveiro
Aveiro, Portugal
tiagomadeira@ua.pt

Paulo Dias
IEETA, DETI
University of Aveiro
Aveiro, Portugal
paulo.dias@ua.pt

Miguel Oliveira
IEETA, DEM
University of Aveiro
Aveiro, Portugal
mriem@ua.pt

***Abstract*—Texture mapping of 3D models using multiple images often results in textured meshes with unappealing visual artifacts known as texture seams. These artifacts can be more or less visible, depending on the color similarity between the images. The main goal of this paper is to produce seamlessly textured meshes from indoor scenarios through a process of color correcting images of the scene. We use a pairwise-based methodology, capable of color correcting multiple images from the same scene. Moreover, we use 3D information from the scene, namely meshes and point clouds, to build a filtering procedure, in order to produce a more reliable spatial registration between images, thereby increasing the robustness of the color correction procedure. This work presents a color correction algorithm integrated with a texture mapping pipeline that receives non-corrected images, a 3D mesh, and point clouds as inputs, producing a textured mesh and color corrected images as output. Results include a comparison between the color similarity of the uncorrected images and of the corrected images, showing that the color correction algorithm enhances the visual quality of textured meshes by eliminating most of the texture seams.**

***Index Terms*—Color Correction, Texture Mapping, Joint Image Histogram, Color Mapping Function.**

I. Introduction

The creation of 3D models from the acquisition of the shape and appearance of real world objects is known as 3D reconstruction. It is an extensively researched topic in computer graphics [1] and computer vision [2], and has recently gained attention in others, for example autonomous driving [3], robotics [4], medical applications [5], agriculture [6], and cultural heritage [7].

The model acquisition technologies are based typically on RGB Cameras [8], RGB-D Cameras [9], and Light Detection and Ranging (LiDAR) [10]. Point clouds from LiDAR sensors contain relatively precise depth information but suffer from problems such as occlusions, sparsity, and noise, whereas images from RGB Cameras provide the color and high-resolution data but no depth information [11]. Nonetheless, the fusion of LiDAR point clouds and RGB images can achieve better 3D reconstruction results than a single data type [12].

Texture mapping is the colorization of a 3D mesh using one image or several overlapping images [13]–[15]. The overlap will often occur in irregular regions, which creates an intricate combination of dependencies and redundant photometric information. This problem is generally solved by selecting a single image from the set of possible images — winner-take-all approach. This approach may generate visual seam artifacts in the textured 3D mesh, which are not visually appealing [16]. The seams are more or less visible depending on the similarity of colors in the images, which is why a proper color correction of the images is crucial to achieve seamless textured 3D models. Color correction can be defined as the general problem of compensating the photometrical disparities between two coarsely geometrically registered images. In other words, color correction consists of transferring the color palette of a reference image, usually called source image (**S**), to a target image (**T**) [17].

This work presents a method that aims to make use of these additional information, namely point clouds and meshes, in order to improve the effectiveness of the color correction procedure, by filtering the noise from the photometrical information shared between images. Furthermore, we present a pairwise-based methodology that is able to tackle the problem of color correcting multiple images from the same scene, according to a reference image. We have also integrated the color correction algorithm in a texture mapping pipeline to assess the performance of the color correction algorithm in producing high-quality textured 3D meshes from indoor scenarios.

The remaining of this paper is structured as follows: in section II, the detailed process to acquire the dataset is described; section III presents the 3D-Based Pairwise Color Correction Approach; in section IV, the results are discussed; finally, section V presents the conclusions and raise some

978-8-3503-4573-57/22/$31.00 ©2022 IEEE

issues to be investigated as future work.

II. Data Acquisition Process

This section describes the hardware and software used in the acquisition process of the dataset used in this paper. The device used is the Leica BLK360[1], which is a 3D LiDAR fixed laser scanner equipped with a spherical imaging system and thermography panorama sensor system. This 3D LiDAR is capable of producing full-color panoramic images geometrically registered with a high-accuracy point cloud. The BLK360 produces as output of each scan a .blk file, which includes the point cloud and the images, as well as some metadata. Despite its high-cost, the accuracy achieved by this equipment is a key aspect to increase the robustness and efficiency of the 3D-based color correction approach presented in this paper.

In an acquisition using a fixed laser scanner, one single scan position in a room will likely not capture every feature of the environment due to occlusions from the laser scanner point of view. The solution is to select several scan locations to acquire most parts of the room and produce a more complete point cloud including more images from different viewpoints. The scans taken must be geometrically registered with respect to each other to define the scan locations within the 3D environment. Since this paper is focused on texture mapping applications, a meshing software is used to produce the 3D meshes from the point clouds. In this context, Leica Geosystems, the manufacturer of the device, provides two software products to process the data produced by the Leica BLK360: Leica Cyclone REGISTER 360[2] and Leica Cyclone 3DR[3].

The Leica Cyclone REGISTER 360 is a point cloud registration software. The software receives as input the .blk raw files from the BLK360, one per scan, geometrically registers the point clouds with respect to each other, and produces a single point cloud file as output. The program supports several file formats of point clouds, and the E57 file format was selected. The E57 is an open-source format and is able to store 3D point data (point clouds), its attributes such as color and intensity, and the 2D images in a cubemap representation.

The Leica Cyclone 3DR is a 3D modeling and meshing software. The software receives as input the .e57 file and then creates a 3D mesh. It is possible to modify several aspects, such as the density of the mesh. It would also be possible to texturize the mesh inside the Leica Cyclone 3DR using the images from the .e57 file, but we prefer to implement our own algorithm to have full flexibility in the texture mapping applications. In the end, the program returns as output a 3D mesh file in OBJ format, one of the several file formats supported.

The dataset used in this paper is from a laboratory at IEETA — Institute of Electronics and Informatics Engineering of Aveiro in University of Aveiro, Portugal and was acquired using Leica's device and software programs, as described above. The dataset is presented in Figure 1 and results from 9 captures at different locations in the room. We geometrically registered the 9 point clouds from those captures, that are used to create a 3D mesh with 47,253 faces and 24,928 vertices of a corner of the room. Additionally, there are 10 images available from those captures to use in the texture mapping process.

[1]https://leica-geosystems.com/products/laser-scanners/scanners/blk360
[2]https://leica-geosystems.com/products/laser-scanners/software/leica-cyclone/leica-cyclone-register-360
[3]https://leica-geosystems.com/products/laser-scanners/software/leica-cyclone/leica-cyclone-3dr

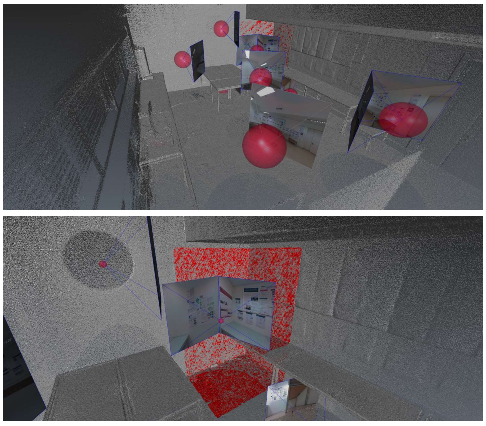


Fig. 1. Two viewpoints from the dataset. The gray points represent the point clouds, the red surfaces represent the 3D mesh, and the red spheres are the scan acquisition locations. The images are also shown, with their frustum in blue.

III. 3D-based Pairwise Color Correction Approach

The architecture of the 3D-based pairwise color correction approach is depicted in Figure 2. As input, the system takes RGB images, registered point clouds from different scans, and a 3D mesh obtained from those point clouds. All RGB images must be geometrically registered with respect to the 3D model, given that the main objective of this paper is to use 3D information to enhance the color similarity between the images. Firstly, the faces present in the 3D mesh are projected onto the images, to create what we call pairwise mappings. These mappings correspond to the same projected vertices of a face on both images from an image pair, that are the correspondent pixel coordinates of the projection on both images. The pairwise mappings are computed for all combinations of image pairs. Then, three techniques are applied to filter the noisy pairwise mappings that would undermine both color correction and texture mapping processes: Z-Buffering filtering, Depth Consistency filtering and Camera Viewpoint filtering. The Z-Buffering filtering and Depth Consistency filtering remove the most common source of noise in datasets from indoor scenarios, which are occluded faces. The Camera Viewpoint filtering discards the pairwise mappings of faces that have an excessive oblique angle with respect to the camera focal axis because the impact of the registration error on the accuracy of these pairwise mappings is amplified. Each computed pairwise mapping is submitted to this procedure, and is only considered valid if it passes the 3 filtering steps. After a successful removal of the noisy pairwise mappings, we compute a Joint

Image Histogram (JIH) for every pair of images and estimate a Color Mapping Function (CMF) that best fits each JIH data. The CMF is estimated using a regression analysis called Support Vector Regressor Composed (SVRC). This composed model mixes the linear kernel and radius basis function kernel, and was created because the radius basis function kernel is not able to extrapolate in the columns of the JIH where there are no observations. To finish the color correction procedure, we perform the pairwise color correction by using the estimated CMFs in all pixels of the images, effectively color correcting all images with respect to a reference image arbitrarily selected by the user. The selection of the reference image by the user allows him/her to have control over the overall appearance of the color corrected mesh. At the end of this stage, the corrected RGB images are produced. To analyze the impact of this 3D-based color correction in the visual quality of textured 3D meshes, the corrected images are used to colorize each face of the 3D mesh using two different techniques: random image selection and largest project area image selection. Besides using color corrected images to colorize the 3D mesh, we also make use of the information from the pairwise mappings filtering component to increase the robustness of the image selection technique. At the end of the pipeline, the textured 3D mesh is produced.

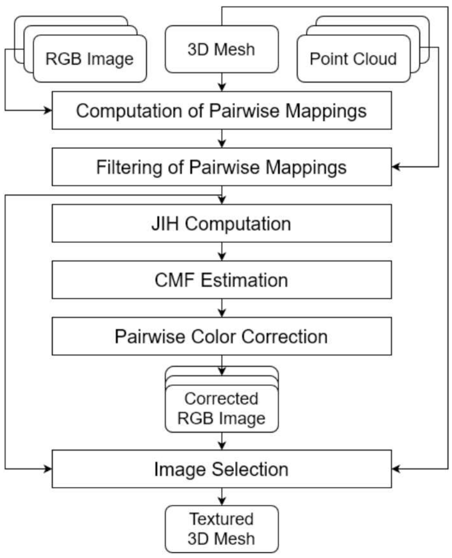


Fig. 2. Architecture of the 3D-based pairwise color correction approach. As input, our system receives RGB images, a 3D mesh and registered point clouds from different scans. In the first part, seven steps are performed to produce the corrected RGB images. Subsequently, we execute one last step to produce the textured 3D mesh.

The detailed description about each step of the pipeline presented above is presented in our paper called "A Robust 3D-based Color Correction Approach for Texture Mapping Applications" [18] in the Scientific Journal called Sensors from MDPI publisher. In this paper, we aim to prove the robustness of the 3D-based pairwise color correction approach by applying it in a more complex dataset, which has a high range of colors. This approach was implemented using the open-source programming language Python jointly with several libraries, such as OpenCV, NumPy, Pandas, Seaborn, SciPy, Matplotlib, Networkx, Graphviz, Pye57, and Open-Mesh.

IV. Results

In this section, we present two comparisons to evaluate the effectiveness of the color correction procedure: Image-Based Quantitative Evaluation and Mesh-Based Qualitative Evaluation. In the first evaluation, we compare the similarity in color between the original images (uncorrected) and the color corrected images using two image similarity metrics over the filtered pairwise mappings: peak signal-to-noise ratio (PSNR) and CIEDE2000. The second evaluation presents a comparison between the original and the color corrected images to analyze their influence on the visual quality of the textured 3D meshes from different viewpoints using both image selection criteria: random image selection and largest projection area image selection. To simplify the identification of the used images, the original images are represented by algorithm #1 and the color corrected images are represented by algorithm #2.

Image-Based Quantitative Evaluation

Table I shows the scores of the original (algorithm #1) and color corrected images (algorithm #2) using the PSNR and CIEDE2000 image similarity metrics. The PSNR is a similarity metrics, which means the higher the score, the higher is the color similarity. The CIEDE2000 is a dissimilarity metrics, meaning that the lower the score, the higher is the color similarity. The 3D-Based Pairwise Color Correction approach increased the similarity in color between the images after the color correction procedure by 21.06% and 34.57% relatively to the original images in the PSNR and CIEDE2000 metrics, respectively.

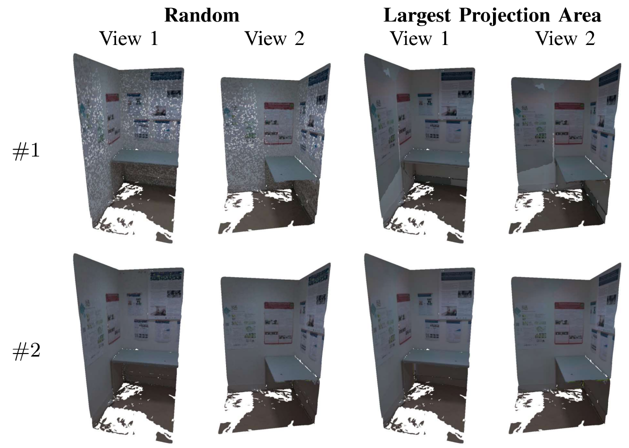
TABLE I
MEAN AND STANDARD DEVIATIONS OF THE PSNR AND CIEDE2000 SCORES USING THE ORIGINAL IMAGES (#1) AND THE COLOR CORRECTED IMAGES (#2). THE BEST RESULTS ARE HIGHLIGHTED IN BOLD.

	PSNR		**CIEDE2000**	
Alg.	μ	σ	μ	σ
#1	19.94	6.39	11.34	6.66
#2	**24.14**	5.26	**7.42**	5.66

Mesh-Based Qualitative Evaluation

Table II presents the textured meshes using the original images (algorithm #1) and the color corrected images (algorithm #2), using the two image selection techniques, from two different viewpoints. Since algorithm #1 uses the original images, the visual artifacts created by the changes in the selected image are noticeable. The random image selection criterion aggravates the effect of this problem, but it can be mitigated by color correcting the images. The color corrected images (#2) produced by the 3D-Based Pairwise Color Correction approach increased significantly the visual quality of the textured mesh. Even when using the random image selection criterion, the texture seams are less noticeable, verifying that indeed the images become more similar in color with respect to each other. Moreover, with the largest projection area selection criterion, the visual quality of the mesh improves even more, displaying almost no texture seams.

TABLE II
TEXTURED MESHES PRODUCED BY THE ORIGINAL IMAGES (#1) AND THE COLOR CORRECTED IMAGES (#2) FROM TWO DIFFERENT VIEWPOINTS AND USING TWO DIFFERENT IMAGE SELECTION TECHNIQUES: RANDOM SELECTION AND LARGEST PROJECTION AREA SELECTION.



V. CONCLUSIONS AND FUTURE WORK

This paper presented the 3D-Based Pairwise Color Correction approach that uses 3D information, namely point clouds and 3D meshes, to filter out noisy mappings between pairs of images through a filtering procedure and uses a pairwise-based methodology capable of color correcting multiple images from the same scene.

Results demonstrate that the 3D-Based Pairwise Color Correction approach increased the similarity in color of all the images from an indoor dataset in both image similarity metrics, compared with the original images. Also, the color corrected images improved the visual quality of the produced textured meshes using both image selection criteria, displaying almost no texture seams.

For future work, we aim to develop an automatic process to select the reference image, enhancing the flexibility of our approach. In addition, we intend to integrate other color correction algorithms with our filtering procedure and evaluate the impact that it has on the color correction outcome. We also plan to develop a multi-image color correction approach based on optimization methods integrated with the pairwise mappings and the filtering procedure. Furthermore, we aim to develop an algorithm to create the geometry of the 3D meshes from the point clouds, and therefore be independent of third-party programs and have full flexibility in the entire pipeline. With this additional algorithm, we can improve the geometry of the 3D mesh to produce 3D models with high quality both in the geometry and the texture.

ACKNOWLEDGMENTS

We thank everyone involved for their time and expertise. This work has been partially supported by Portugal 2020, project I&DT nº 45382 in collaboration with VR360, by National Funds through the FCT - Foundation for Science and Technology, in the context of the project UIDB/00127/2020, and by FCT - Portuguese Foundation for Science and Technology, in the context of Ph.D. scholarship 2020.07345.BD.

REFERENCES

[1] R. Ziegler, W. Matusik, H. Pfister, and L. McMillan, "3D reconstruction using labeled image regions," *Proceedings of the Eurographics Symposium on Geometry Processing*, pp. 248–260, 2003.
[2] W.-S. Jang and Y.-S. Ho, "3-d object reconstruction from multiple 2-d images," *3D Research*, vol. 1, no. 2, pp. 1–5, 2010.
[3] Y. Zhang, H. Chen, S. L. Waslander, T. Yang, S. Zhang, G. Xiong, and K. Liu, "Toward a More Complete, Flexible, and Safer Speed Planning for Autonomous Driving via Convex Optimization," *Sensors*, vol. 18, no. 7, 2018.
[4] L. Pérez, Í. Rodríguez, N. Rodríguez, R. Usamentiaga, and D. F. García, "Robot Guidance Using Machine Vision Techniques in Industrial Environments: A Comparative Review," *Sensors*, vol. 16, no. 3, 2016.
[5] M. Trinidad-Fernández, D. Beckwée, A. Cuesta-Vargas, M. González-Sánchez, F.-A. Moreno, J. González-Jiménez, E. Joos, and P. Vaes, "Validation, Reliability, and Responsiveness Outcomes of Kinematic Assessment with an RGB-D Camera to Analyze Movement in Subacute and Chronic Low Back Pain," *Sensors*, vol. 20, no. 3, 2020.
[6] M. Vázquez-Arellano, H. W. Griepentrog, D. Reiser, and D. S. Paraforos, "3-D Imaging Systems for Agricultural Applications—A Review," *Sensors*, vol. 16, no. 5, 2016.
[7] L. Di Angelo, P. Di Stefano, E. Guardiani, A. E. Morabito, and C. Pane, "3D Virtual Reconstruction of the Ancient Roman Incile of the Fucino Lake," *Sensors*, vol. 19, no. 16, 2019.
[8] I. Afanasyev, A. Sagitov, and E. Magid, "ROS-based SLAM for a gazebo-simulated mobile robot in image-based 3D model of Indoor environment," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9386, no. October, pp. 273–283, 2015.
[9] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with Microsoft Kinect sensor: A review," *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1318–1334, 2013.
[10] H. Fan, W. Yao, and Q. Fu, "Segmentation of sloped roofs from airborne LiDAR point clouds using ridge-based hierarchical decomposition," *Remote Sensing*, vol. 6, no. 4, pp. 3284–3301, 2014.
[11] H. Xu and C.-Y. Chen, "3D Facade Reconstruction Using the Fusion of Images and LiDAR: A Review," in *New Trends in Computer Technologies and Applications* (C.-Y. Chang, C.-C. Lin, and H.-H. Lin, eds.), (Singapore), pp. 178–185, Springer Singapore, 2019.
[12] S. Becker and N. Haala, "Refinement of building fassades by integrated processing of LIDAR and image data," *PIA 2007 - Photogrammetric Image Analysis, Proceedings*, no. September, pp. 7–12, 2007.
[13] A. Bhattad, A. Dundar, G. Liu, A. Tao, and B. Catanzaro, "View Generalization for Single Image Textured 3D Models," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6077–6086, IEEE, jun 2021.
[14] Y. Fu, Q. Yan, J. Liao, H. Zhou, J. Tang, and C. Xiao, "Seamless Texture Optimization for RGB-D Reconstruction," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 8, pp. 1–1, 2021.
[15] H. He, J. Yu, P. Cheng, Y. Wang, Y. Zhu, T. Lin, and G. Dai, "Automatic, Multiview, Coplanar Extraction for CityGML Building Model Texture Mapping," *Remote Sensing*, vol. 14, p. 50, dec 2021.
[16] M. Oliveira, G. H. Lim, T. Madeira, P. Dias, and V. Santos, "Robust texture mapping using rgb-d cameras," *Sensors*, vol. 21, no. 9, pp. 1–22, 2021.
[17] W. Xu and J. Mulligan, "Performance evaluation of color correction approaches for automatic multi-view image and video stitching," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 263–270, 2010.
[18] D. Coelho, L. Dal'Col, T. Madeira, P. Dias, and M. Oliveira, "A Robust 3D-Based Color Correction Approach for Texture Mapping Applications," *Sensors*, vol. 22, p. 1730, feb 2022.

Posters

Dam Health Monitoring with VR

Pedro Leitão
Instituto Superior Técnico, University of Lisbon
Lisbon, Portugal
pedro.de.leitao@tecnico.ulisboa.pt

Nuno Verdelho Trindade
INESC-ID, Instituto Superior Técnico, University of Lisbon
Lisbon, Portugal
mail@nunotrindade.com

Sérgio Oliveira
Laboratório Nacional de Engenharia Civil (LNEC)
Lisbon, Portugal
soliveira@lnec.pt

Alfredo Ferreira
INESC-ID, Instituto Superior Técnico, University of Lisbon
Lisbon, Portugal
alfredo.ferreira@tecnico.ulisboa.pt

***Abstract*—Structural Health Monitoring (SHM) is a crucial procedure when it comes to ensuring the safety of the structures. SHM is mainly done through traditional methods, but with the evolution of immersive technologies such as Virtual Reality (VR), this is rapidly changing.**

The goal of this work is to explore the benefits and challenges of applying VR in the SHM of dams. For this, related work concerning applications of VR in different areas was explored and discussed. An application, named VRCabrilAnalysis, was also developed and tested.

VRCabrilAnalysis is a VR application where users can move through a model of the Cabril Dam and interact with the sensors that are present there. It is possible to visualize the data measured by the sensors and analyze the damage evolution of the dam over time.

***Index Terms*—virtual reality, digital shadow, immersive environments, structural health monitoring for dams, dam analysis over time, damage evolution**

I. Introduction

Structural Health Monitoring (SHM) is the evaluation in real-time of a physical structure and the materials that compose it, to detect failures as early as possible. This minimizes the risks and ensures the safety of the structure [1].

Laboratório Nacional de Engenharia Civil (LNEC) performs SHM analysis of dams through the use of traditional methods, needing to be on-site to analyze and monitor the safety of the dam in real-time.

Immersive analytics is a field that combines data visualization, visual analytics, human-computer interaction, and immersive technologies such as virtual reality (VR) and mixed reality (MR). It has the goal of facilitating the connection between the data, the users, and the tools used for analysis [2].

To study the benefits and consequences of applying immersive analytics in the SHM of dams, we developed a VR application, called VRCabrilAnalysis, where the users can analyze the measurements from the sensors installed at the Cabril Dam.

This work is a first step towards a digital twin VR application of the Cabril Dam, allowing users to analyze and monitor the health of the Cabril Dam and then act accordingly.

II. Related Work

Augmented Reality (AR) has previously been applied in the inspection and monitoring of dams, with a system named DamAR [3]. With VRCabrilAnalysis, we will be testing the application of VR in the same field.

To support the development of our application, we analyzed related work concerning VR applications in architecture, engineering, and construction (AEC) industry and the benefits they provide.

One example of this was British Columbia Hydro Corporation's Life Safety Model (LSM) [4], a VR computer system where users could generate and analyze different dam emergency scenarios. Geographic Information Systems (GIS) and census data were used to provide the necessary means to develop this system.

Hoover Dam: IndustrialVR [5] is a VR serious game that allows its users to explore the Hoover Dam and learn about its components and how they work. An innovative documentary-style approach was used in the development of this application.

Another VR serious game applied in the exploration of the Hoover Dam was VR Hoover Dam [6]. This historically based VR serious game allows its users to explore the Hoover Dam while they learn about its construction and its impact on the population, the economy, and the natural environments.

Web-Based Game-Like VR Construction Site Simulator (WWGVRSS) [7] is an interactive and collaborative VR rule-based system. For its development, a Computer-Aided Design (CAD) interface and a web-based VR interface were used to support an online BIM platform for the integration of projects in the AEC industry.

A VR-integrated workflow in BIM-based projects [8] was developed to allow multiple users to collaboratively visualize and analyze a specific BIM model in the same virtual environment.

III. Prototype

VRCabrilAnalysis is a VR application where a specific user can see a representation of the Cabril Dam 1(a) and its surrounding area 1(b). The model contains different types of sensors, which the user can interact with.

978-8-3503-4573-57/22/$31.00

(a) The model of the Cabril Dam (b) The surrounding area

Fig. 1. The virtual environment of VRCabrilAnalysis

The types of sensors present in VRCabrilAnalysis are geodetic marks, benchmarks, plumblines, the GNSS antenna, the water tower, and accelerometers (either uniaxial or triaxial, as well as the data acquisition units). The sensors become highlighted when the user points to them, with a color that depends on whether or not they have their acquired data present in the application.

The user can select the sensors and teleport to a certain part of the scene. When selecting a certain sensor, an interactive menu appears on top of the controller that was used for that. The interactive menu contains information about the selected sensor. It also contains options for plotting the time evolution of the data acquired by the sensor and displaying additional information about the sensor.

The additional information depends on what type the selected sensor is. If the sensor is an accelerometer, it consists of information about seismic accelerations and about the corresponding earthquake such as its date and epicenter. If the selected sensor is not an accelerometer, the additional information consists of the last recorded values for each attribute the sensor measures, the names of those attributes, and the date and hour when those last values were registered.

The interactive menu automatically disappears when the option for plotting the time evolution of data is selected. It is possible to toggle between the menu being visible or invisible as well as the information tables.

The data charts differ depending on the type of the selected sensor. In case it is an accelerometer, the table contains one line chart, containing the time evolution of the accelerations, which can be the radial, tangential and vertical accelerations.

If the sensor is not an accelerometer, the table contains three line charts, one for the time evolution of the temperature, another for the time evolution of the water elevation, and the other for the time evolution of the displacements. The displacements that can be present in the chart are the radial, tangential and vertical displacements.

The user can interact with the data charts by using the controllers. When the user points to a certain part of a certain line chart, a tooltip appears, displaying the attribute, its value, and the date when that value was recorded.

With the buttons, from the left controller, the user can use the Pan feature by pressing the primary button and Zoom Out by clicking on the secondary button. From the right controller, the user can select a certain interval of data with the Brush feature by pressing the primary button and Zoom In by clicking on the secondary button.

In case a certain line chart has more than one attribute, it contains a legend on top that displays the name of each different variable and which line color represents it. The user can select which attributes he wants to see in the chart by clicking on the legend of that specific variable.

Fig. 2. Interacting with VRCabrilAnalysis.

IV. Future Work

VRCabrilAnalysis has the goal of obtaining results that help in our study. Those results are gathered through user tests. This application was developed as a proof of concept and will be continued with future projects. The results gathered by testing this version of the prototype will also be used to know the necessary adaptations that should be made in the application. The ultimate goal of this project is to have a digital twin of the dam, where it would be possible for multiple users to collaboratively perform the SHM of the Cabril Dam.

Acknowledgment

This work was supported by national funds and the European Social Fund (ESF) through Fundação para a Ciência e a Tecnologia (FCT), under projects UIDB/50021/2020 and PDTC/ECI-ECG/5332/2020 and grant 2021.07266.BD.

References

[1] D. Balageas, C.-P. Fritzen, and A. Guemes, Structural health monitoring. John Wiley & Sons, 2010, vol. 90.

[2] S. Malik, A. Ranjan, and R. Balakrishnan, "Interacting with large displays from a distance with vision-tracked multi-finger gestural input," UIST: Proceedings of the Annual ACM Symposium on User Interface Software and Technology, pp. 43–52, 2005. [Online]. Available: https://books.google.com/books/about/Immersive Analytics.html?id=vaVyDwAAQBAJ

[3] N. Verdelho Trindade, A. Ferreira, and S. Oliveira, "DamAR: Augmented reality in dam safety control," International Journal on Hydropower and Dams, vol. 26, no. 5, pp. 56–62, 2019. [Online]. Available: https://www.hydropower-dams.com/articles/damar-augmented-reality-in-dam-safety-control

[4] H. Assaf and D. Hartford, "A virtual reality approach to public protection and emergency preparedness planning in dam safety analysis," in Proceedings of the Canadian dam association conference, Victoria, 2002. [Online]. Available: https://www.researchgate.net/publication/258513135

[5] Hoover Dam: IndustrialVR on Oculus Rift, https://www.oculus.com/experiences/rift/1297777480332012/

[6] VR Hoover Dam – A Virtual Reality Game Exploring the History and Construction of an American Icon, https://vrhooverdam.com/

[7] J. S. Goulding, F. P. Rahimian, and X. Wang, "Virtual reality-based cloud bim platform for integrated aec projects," Journal of Information Technology in Construction, vol. 19, pp. 308–325, 2014. [Online]. Available: https://strathprints.strath.ac.uk/58018

[8] R. Zaker and E. Coloma, "Virtual reality-integrated workflow in bim-enabled projects collaboration and design review: a case study," Visualization in Engineering, vol. 6, 2018.

Two-Dimensional Scatterplots and Parallel Coordinates Plots in VR

Paulo Moutinho, Daniel Mendes, Rui Rodrigues, Alexandre Carvalho
INESC TEC, Faculdade de Engenharia, Universidade do Porto
up201704710@edu.fe.up.pt, {danielmendes, rui.rodrigues, alexandre.valle}@fe.up.pt

***Abstract*—Nowadays, people have access to a wide range of data. However, understanding how it relates to real-life events is an elaborate and time-consuming process. We developed an approach for immersive data analysis using 2D Scatterplots to represent time-dependent continuous data, and Parallel Coordinates technique to track specific data points across several dimensions. A Timeline visualization connects the information to discrete events making it easier to correlate different types of information. We conducted a user evaluation that highlighted the potential of the proposed approach.**

***Index Terms*—Information Visualization, Virtual Reality, Immersive Analysis, 2D Scatterplots, Parallel Coordinates Plots**

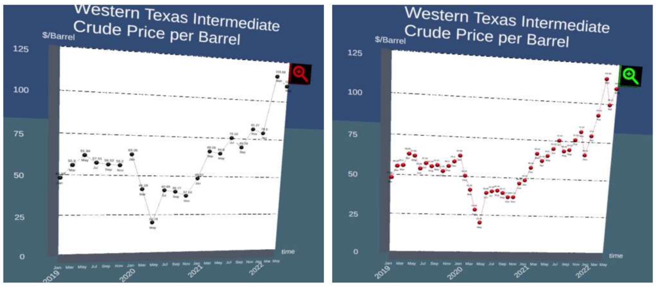


Fig. 1. Temporal data represented by scatterplots, with normal (left) and denser time sampling (right).

I. Introduction

The concept of Information Visualization, commonly referred to as InfoVis, has long piqued people's interest [2], [4]: creating interactive and visual representations of complex data to reinforce human cognition by allowing the user to better understand the complex data structures, to discover causal patterns and to have a better understanding of the information, leading to assessments that could be missed otherwise.

InfoVis' exploration of novel ways for people to view and interact with abstract data, combined with Virtual Reality's ability to provide an immersive experience that may combine 3D and 2D visualizations [9], lead to new ways of representing information and interacting with it [5], [8], [10], and to the field of Immersive Analytics [7].

In this work, we present an approach in the realm of Immersive Analytics that combines techniques from both the field of Information Visualization and the field of Immersive Environments. To that end, we use as case study an example pertaining to the financial domain. In this field, understanding how geopolitical, social and economic events are related is very important, but it is a complex and time-consuming process. We therefore proposed and evaluated an immersive visualization of both asset performance and relevant financial indicators to help discern what is influencing value change. To this end, we allow users to explore 2D Scatterplots both individually as well as connected in a 3D Parallel Coordinates visual idiom.

II. Approach

We proposed the implementation of more than one visualization technique in immersive environments with several interaction methods thought to provide familiarity and immersion. Since our goal is to enable users to find possible correlation between different sources of information, we believe a visualization encompassing multiple representations and a natural and immersive interaction with heterogeneous data might help us more easily accomplish this goal.

As the main display, 2D Scatterplots are used to represent high-dimensional temporal data regarding the performance of assets and economic relevant markers (Fig. 1). The users are able to interact with these plots by getting closer and selecting the points that they intend to analyze. The plots are placed in circle around the user (Fig. 2), facing the center. Users can rotate them around 360º to better position the active plots in the direction they are looking. The plot the user is looking towards gets closer to the center, allowing the user to get in range of selecting the points with just one or two steps. Each plot has the option to maximize the temporal scale in the top right

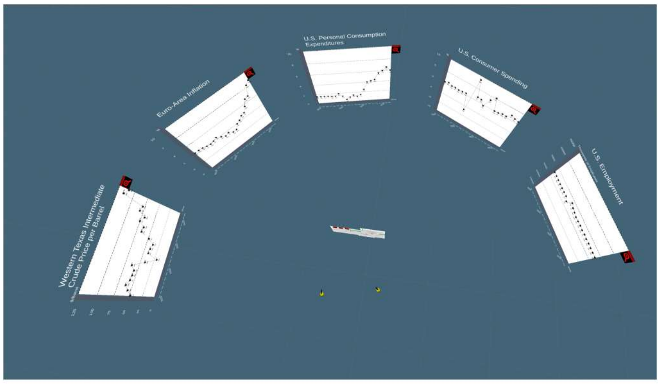


Fig. 2. Scatterplots radial placement around the user.

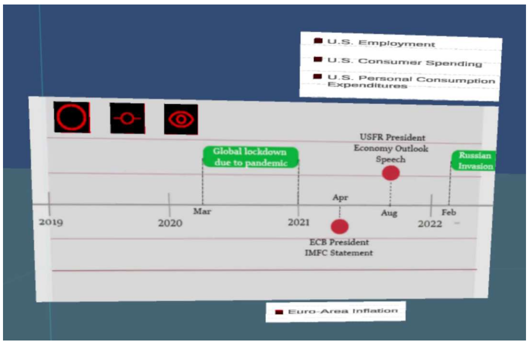


Fig. 3. Timeline of discrete events.

corner, accessing more detailed information about a specific metric or asset (Fig. 1 right).

A second visualization will dictate which plots are visible to the user. The solution includes a Timeline representing geopolitical, economic and social events in a temporal scale (Fig. 3). The user can select which events are active, dynamically updating the Scatterplots display. For each event, several relevant financial indicators are displayed and if chosen, the respective plot will be added to the pool of Scatterplots present in the scene.

A third visualization technique, Parallel Coordinates, is also available. After selecting the intended plots, the user can opt to link the plots, better following individual points through several charts (Fig. 4). Any selected point will also highlight points of different Scatterplots in the same time scale, including plots represented in the Parallel Coordinates visualization. Also, users can freely move, rotate, and scale the whole visualization object directly with their hands.

As mentioned before, interaction techniques are crucial to immerse the user into the environment and, consequently, the visualization of data. Given that these representations are not far and unreachable from the user original position, it makes sense to choose the simple virtual hand technique as well as

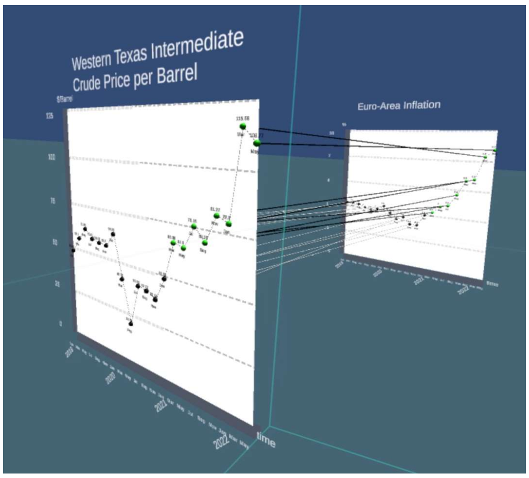


Fig. 4. Parallel Coordinates visualization with some points selected.

real walking as the selection and navigation techniques [6], respectively.

III. Evaluation and Results

We conducted a user evaluation, where the VR prototype was compared to a Desktop setting. The Desktop setting resorted to the Bloomberg platform [1], which provides global financial indicators and assets charts. Every necessary web page was opened in a browser tab before the test and participants were given the opportunity to explore the multiple plots.

The tests consisted in executing four tasks, each composed of two to three financial questions promoting data analysis. Both the questions and the answer were given verbally. Since the presented data did not change between setups, the datapoints asked in each differed slightly (e.g. maximum values instead of minimum values, different dates) without changing the difficulty and the essence of the tasks.

At the end of each setup, participants were asked to fill out the System Usability Scale questionnaire [3]. To avoid bias, participants alternated between setups and corresponding tasks, following a Latin square design.

A total of 20 participants (13 males and 7 females) between 16 and 35 years old took part in the tests. While the time measured during each task determined that our solution did not provide the fastest data analysis between both setups, the results from the system usability questionnaire and the user feedback showed that our solution is more appealing, intuitive, and easy to use than the Desktop baseline.

References

[1] Bloomberg.com. https://www.bloomberg.com/europe. Online; accessed 23 February 2022.

[2] B. B. Bederson, P. B. B. Bederson, B. Shneiderman, and U. P. B. Shneiderman. *The Craft of Information Visualization: Readings and Reflections*. Morgan Kaufmann, 2003. Google-Books-ID: IDaFnCl6TvIC.

[3] J. Brooke. SUS: A quick and dirty usability scale. *Usability Eval. Ind.*, 189, Nov. 1995.

[4] S. Card. *INFORMATION VISUALIZATION*. CRC Press, Mar. 2009. Pages: 199-234 Publication Title: Human-Computer Interaction.

[5] C. Donalek, S. G. Djorgovski, S. Davidoff, A. Cioc, A. Wang, G. Longo, J. S. Norris, J. Zhang, E. Lawler, S. Yeh, A. Mahabal, M. Graham, and A. Drake. Immersive and Collaborative Data Visualization Using Virtual Reality Platforms. *2014 IEEE International Conference on Big Data (Big Data)*, pages 609–614, Oct. 2014. arXiv: 1410.7670.

[6] J. J. LaViola Jr, E. Kruijff, R. P. McMahan, D. Bowman, and I. P. Poupyrev. *3D user interfaces: theory and practice*. Addison-Wesley Professional, 2017.

[7] K. Marriott, F. Schreiber, T. Dwyer, K. Klein, N. H. Riche, T. Itoh, W. Stuerzlinger, and B. H. Thomas. *Immersive Analytics*. Springer, Oct. 2018. Google-Books-ID: vaVyDwAAQBAJ.

[8] S. K. Tadeja, T. Kipouros, and P. O. Kristensson. Exploring Parallel Coordinates Plots in Virtual Reality. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI EA '19, pages 1–6, New York, NY, USA, May 2019. Association for Computing Machinery.

[9] B. H. Thomas. Virtual Reality for Information Visualization Might Just Work This Time. *Frontiers in Robotics and AI*, 6, 2019.

[10] Y. Yang, T. Dwyer, B. Jenny, K. Marriott, M. Cordeil, and H. Chen. Origin-Destination Flow Maps in Immersive Environments. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):693–703, Jan. 2019. Conference Name: IEEE Transactions on Visualization and Computer Graphics.

Framework para Reconstrução e Visualização de Ambientes 3D através de Geolocalização

Roberto Ribeiro, Nuno Rodrigues
CIIC-ESTG-Politécnico de Leiria
roberto.ribeiro@ipleiria.pt, nunorod@ipleiria.pt

Filipe Sousa, Isac Amado
ESTG-Politécnico de Leiria
2191168@my.ipleiria.pt, 2191908@my.ipleiria.pt

***Resumo*—É notável o crescente interesse na reconstrução de ambientes 3D em temas como a simulação realista, os Digital Twins e a Realidade Virtual. Como tal, este trabalho propõe e descreve uma *framework* para a reconstrução digital simplificada de uma área exterior, através de GPS. Esta alternativa acessível e de baixo custo permite ao utilizador comum criar um mapa e registar os seus pontos de interesse (PoI) com um dispositivo móvel. É apresentada a concretização desta proposta, que é constituída por uma *app* Android, um servidor alojado na *cloud* e a plataforma Unity. A sua usabilidade foi avaliada por dez participantes, através da *System Usability Scale* (SUS).**

Palavras-chave—Reconstrução de Ambientes 3D, Simulação de Ambientes, Geolocalização, Pontos de Interesse, Aplicações Móveis, Plataforma de Desenvolvimento Unity

I. Introdução

A reconstrução de ambientes 3D tem sido explorada para diversos fins. Entre eles, a simulação realista para o treino de *robots* e agentes interativos em ambientes seguros [1], o registo e visualização de PoI culturais (e.g., monumentos) [2], planeamento para terrenos de construção [3] e planeamento urbano com desenvolvimento de Digital Twins [4]. São várias as técnicas utilizadas para a reconstrução digital de ambientes 3D. Por exemplo, os sensores LiDAR [5] e as câmaras estereoscópicas [3] são tecnologias que permitem estimar distâncias e, por consequência, a reconstrução baseada na geração de malhas poligonais. Apesar de funcionarem com precisão, estes equipamentos podem atingir custos elevados.

Como tal, este trabalho apresenta a proposta, descrição, implementação e testes de uma alternativa de baixo custo e acessível ao utilizador comum, para a reconstrução de ambientes 3D, através de geolocalização. O objetivo é a criação de um mapa em tempo real, através de um dispositivo móvel (e.g., smartphone), para visualização e edição numa plataforma 3D.

II. Proposta da Arquitetura da *Framework*

Para cumprir os objetivos propostos, foi criada uma arquitetura tripartida nos sistemas apresentados de seguida.

A. Sistema de Criação de Mapas

O Sistema de Criação de Mapas (SCM) considera um dispositivo móvel que permite estar presencialmente numa área exterior e iniciar a reconstrução desse ambiente 3D através da criação de um mapa, de forma intuitiva e em tempo real. Depois de criar o mapa, o utilizador deverá registar os PoI presentes nessa área, bem como a sua geolocalização (i.e., latitude, longitude, altitude), obtida através dos sistemas de navegação do dispositivo utilizado. O SCM permite também atualizar mapas já criados através da adição, edição e remoção de PoI.

B. Sistema de Armazenamento de Mapas

O Sistema de Armazenamento de Mapas (SAM) é responsável pela persistência dos mapas criados pelo SCM e pelo seu acesso. Este sistema inclui a base de dados da *framework* e deve garantir a disponibilidade dos vários mapas, através da Internet, a qualquer momento e em qualquer lugar. É ao SAM que os outros Sistemas solicitam os dados para poder visualizar, editar, ou eliminar os mapas.

C. Sistema de Edição e Visualização de Mapas

O Sistema de Edição e Visualização de Mapas (SEVM), que deve ser suportado por uma plataforma de desenvolvimento 3D, integra a edição de um mapa e a visualização da reconstrução digital da sua área. Para isso, deve pedir ao SAM uma listagem dos mapas persistidos e escolher um deles. Consoante a geolocalização e categoria dos PoI, o SEVM converte as coordenadas numa posição relativa a um ponto de origem e escolhe um modelo 3D para apresentar. Esta reconstrução assemelha-se ao posicionamento de miniaturas num jogo de tabuleiro. Quanto à edição de um mapa, considera-se a alteração das características de pelo menos um dos seus PoI. Esta edição pode resultar numa alteração do nome, descrição, categoria e geolocalização definida pelo SCM. Esta edição deve ser convertida novamente em coordenadas e guardada no SAM.

III. Implementação dos Módulos da *Framework*

A implementação segue a seguinte arquitetura (Fig. 1).

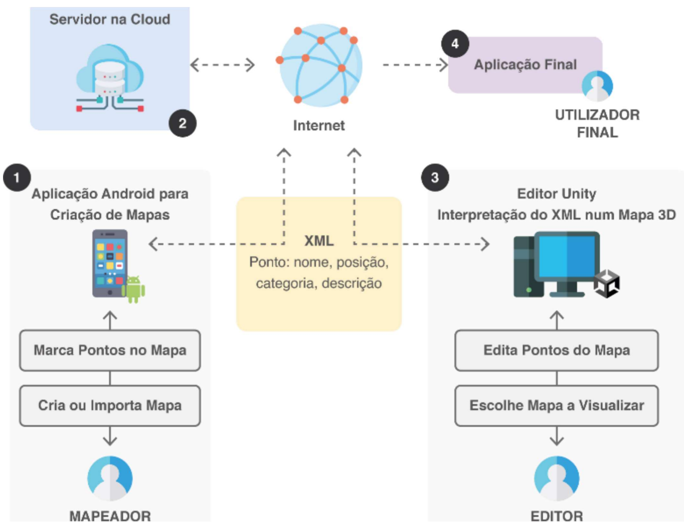


Fig. 1. Arquitetura Funcional da Implementação

978-8-3503-4573-57/22/$31.00 ©2022 IEEE

Para desenvolver os sistemas propostos, foram criadas três entidades que comunicam através da Internet. Estas entidades correspondem a uma *app* móvel Android, um servidor alojado na *cloud* e uma interface customizada para a plataforma Unity. A implementação pressupões três perfis de utilizador. São eles o Mapeador, que cria mapas e marca pontos; o Editor, que visualiza e edita o mapa já criado; e o Utilizador Final.

A. App Móvel Android

O SCM foi desenvolvido no formato de uma *app* móvel Android (1). Para usar esta *app*, o Mapeador dirige-se à área a reconstruir, inicia a criação do mapa e regista cada um dos PoI (Fig. 2). Este registo inclui a atribuição de nome, descrição e a classificação baseada num conjunto de categorias pré-definidas (e.g., banco, estátua, árvore). Aqui, o GPS é utilizado para atribuir automaticamente a posição do Mapeador a esse ponto. Durante a criação de um mapa, a *app* mostra um mapa com todos os PoI registados. Finalmente, o mapa é guardado através de XML e transmitido ao SAM, através da Internet, para persistir. Além disso, a *app* permite importar um mapa já guardado no SAM, para posterior atualização. Esta funcionalidade é útil para casos em que a estrutura da mesma área exterior seja variável.

B. Servidor Web para Persistência dos Mapas

Para persistir os mapas criados e permitir o seu acesso, a qualquer momento e em qualquer lugar, foi alojado um servidor na *cloud*, através da plataforma Heroku. Este servidor, criado com a *framework* PHP Slim, suportou uma *api* REST, usada para gerir os mapas através da comunicação das mensagens XML. A persistência e gestão dos mapas foi garantida com uma base de dados MySQL e com o *add-on* ClearDB.

C. Editor Unity para Gestão e Visualização dos Mapas

A plataforma Unity permite criar e exportar experiências interativas em ambientes 3D. Como tal, é no Unity que o utilizador Editor visualiza a reconstrução dos mapas criados pelo Mapeador. Para tornar eficiente a gestão do mapa, foi implementado um editor customizado para Unity, em C#. Este editor corresponde a uma interface gráfica que permite escolher um mapa, gerar a sua reconstrução digital e gerir os seus pontos. Este mapa é obtido em XML, a partir do servidor *web*.

A reconstrução do mapa divide-se em duas partes. Primeiro, as coordenadas GPS de cada PoI são convertidas em posições, à escala real. Depois, nessas posições, são instanciados modelos 3D representativos das categorias de cada PoI (Fig. 3). Finalmente, o utilizador Editor pode atualizar as coordenadas de um PoI, ao alterar a posição do seu modelo 3D. Para persistir as alterações, a posição relativa do PoI no Unity é convertida para coordenadas GPS. Através da *api* REST, o Unity atualiza o mapa editado no SAM.

Fig. 2. Vista de Criação de um Ponto na App Móvel

Fig. 3. Representação de um mapa na plataforma de desenvolvimento Unity.

IV. Testes e Resultados

Para garantir a correta operação de cada um dos sistemas que representam a *framework*, foram feitos testes funcionais em cenários reais. Uma vez que este trabalho pretende disponibilizar uma alternativa simples e intuitiva para a reconstrução de ambientes 3D, criámos um cenário de teste para avaliar o perfil do mapeador ao usar a *app* móvel. Estes testes decorreram numa área exterior, com dez participantes, entre os 18 e os 52 anos, com diferentes níveis de experiência no uso de *apps* móveis. Depois de criar o mapa, os participantes avaliaram a experiência com a SUS [6]. A média das respostas dos dez participantes foi 75.75, o que representa um resultado qualitativo de "bom" na SUS.

V. Conclusão

Este trabalho apresentou a proposta e implementação de uma *framework* acessível e de baixo custo, composta por três sistemas que comunicam através da Internet para reconstruir e visualizar ambientes 3D. A reconstrução, baseada em geolocalização, torna possível a visualização digital simplificada de áreas exteriores. A usabilidade foi avaliada positivamente através da SUS. Ainda, o Unity, escolhido para a visualização e edição dos mapas, possibilita a criação de experiências interativas sobre a reconstrução apresentada. São exemplos, um videojogo baseado num cenário real, um ambiente para simulação de controlo e navegação, ou um Digital Twin. Sugere-se assim o potencial desta *framework* para agilizar o desenvolvimento de experiências baseadas em cenários reais.

Agradecimentos

Este trabalho apresenta ícones obtidos na plataforma web "flaticon" e desenvolvidos pelo autor "Freepik" e foi financiado por fundos nacionais através da FCT – Fundação para a Ciência e a Tecnologia, I.P., no âmbito do projeto UIDB/04524/2020.

Referências

[1] B. Shen et al., "iGibson 1.0: A Simulation Environment for Interactive Tasks in Large Realistic Scenes," in 2021 IROS, Sep. 2021, pp. 7520–7527. doi: 10.1109/IROS51168.2021.9636667.

[2] I. Liritzis et al., "3D Reconstruction of Cultural Heritage Sites as an Educational Approach. The Sanctuary of Delphi," Applied Sciences, vol. 11, no. 8, p. 3635, Apr. 2021, doi: 10.3390/app11083635.

[3] C. Sung and P. Y. Kim, "3D terrain reconstruction of construction sites using a stereo camera," Automation in Construction, vol. 64, pp. 65–77, Apr. 2016, doi: 10.1016/j.autcon.2015.12.022.

[4] G. Schrotter and C. Hürzeler, "The Digital Twin of the City of Zurich for Urban Planning," PFG, vol. 88, no. 1, pp. 99–112, Feb. 2020, doi: 10.1007/s41064-020-00092-2.

[5] Y.-C. Fan, et al., "3D Environment Measurement and Reconstruction Based on LiDAR," in 2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Houston, TX, May 2018, pp. 1–4. doi: 10.1109/I2MTC.2018.8409759.

[6] A. I. Martins et al., "European Portuguese Validation of the System Usability Scale (SUS)," Procedia Computer Science, vol. 67, pp. 293–300, Jan. 2015, doi: 10.1016/j.procs.2015.09.273.