

III Congreso Español de Videojuegos (CEV2024)

Pre-actas

A Coruña, España
17, 18 y 19 de junio de 2024

En este volumen, presentamos las **pre-actas** del III Congreso Español de Videojuegos, un recopilatorio de todos los envíos recibidos con independencia del tipo. Estas **pre-actas** tienen como objetivo brindar una visión general y exhaustiva de la producción científica presentada en el evento.

En las actas oficiales del congreso se incluirán aquellos trabajos enviados a las categorías que implican publicación en las actas. Sin embargo, en este volumen, hemos decidido incluir todos los envíos recibidos, sin excepción, para que los lectores puedan acceder a la totalidad de la producción científica presentada.

Estas **pre-actas** no han sido editadas, y se presentan tal como fueron enviados por los autores.

En las siguientes páginas, encontrarás una amplia variedad de contribuciones, desde artículos completos hasta resúmenes de trabajos ya publicados, etc. Esperamos que estas **pre-actas** sean de interés para todos aquellos asistentes al congreso.

Nota: Estas pre-actas no sustituyen a las actas oficiales, sino que son más bien una extensión de la producción científica presentada en el evento.

Impact of Affective AI on Player Engagement in Video Games: An EEG-Based Study

Ricardo Fernández-Rafael¹[0000-0002-7078-7561], Macarena Cuenca-Amigo²[1111-2222-3333-4444]
and Alazne Mujika-Alberdi³[1111-2222-3333-4444],

¹ Universidad de Deusto

² Universidad de Deusto

³ Universidad de Deusto

`ricar.fernandez@opendeusto.es`

Abstract. This study aims to investigate the impact of affective artificial intelligence (AI) integration within video games on player engagement levels. Utilizing electroencephalogram (EEG) measures and behavioral observations, we examine the effects of affective responses such as interest, excitement, focus, stress, and relaxation, on player interaction within the psychological horror adventure game “Nevermind”. Through a comparative analysis of a control group (playing without affective AI) and an experimental group (playing with affective AI), this research seeks to empirically test the hypothesis that affective AI enhances player engagement.

Keywords: affective artificial intelligence, emotional artificial intelligence, video game engagement, player experience personalization, EEG, biofeedback.

1 Introduction

1.1 From AI to Affective AI

The emergence of Affective Artificial Intelligence (AI) in gaming signifies a pivotal shift, integrating cutting-edge technology with psychological insights to create immersive and interactive gaming experiences. This progress, anchored in machine learning, data analysis, and human-computer interaction, aims to bridge the emotional gap between digital and human gamers, enhancing the realism and engagement in video games [1].

At the heart of Affective AI in gaming is the ability to precisely recognize, understand, and adapt to players' emotions. This involves sophisticated algorithms, neural networks, and data processing methods that evaluate inputs like facial expressions, voice tones, and physiological signals (Roohi et al., 2019). The goal is to build a system that not only comprehends human emotions in real time but also customizes its responses, offering a unique gaming experience for each player [2].

Improvements in machine learning and data processing have enabled Emotional AI to discern a broader range of emotions and respond more contextually. These developments have opened new avenues for game design, allowing the creation of games that adjust storylines, difficulty, and character behavior based on the player's emotional state [3].

The ascent of Affective AI in video games signals a substantial shift in game design and player experience. It not only deepens the emotional dimension of gaming but also leads to more sophisticated and personalized player experiences. As this technology evolves, its role in shaping the future of video game development and player engagement is poised to grow exponentially.

1.2 Player Engagement

Player engagement in video gaming is a complex construct that is pivotal for the gaming industry, impacting everything from player satisfaction to game success and retention. This engagement can be broken down into cognitive, emotional, and behavioral dimensions, each playing a distinct role in the player's experience.

The significance of these engagement forms cannot be overstated for game developers and the industry at large. Games that effectively engage players tend to see higher levels of player satisfaction [4], which can translate into greater commercial success and longer player lifecycles.

Engagement also has a direct impact on the financial success of video games [5]. Games that can engage players and keep them hooked are more likely to generate higher sales and revenue [4].

1.3 Affective AI and Player Experience

In video gaming, affective AI is utilized to enhance the player experience by dynamically adjusting game environments, narratives, and challenges in response to players' emotional states. This approach aims to create a more immersive and personalized gaming experience by employing emotion recognition algorithms, machine learning models, and biometric sensing technologies. These components work together to interpret data from facial expressions, voice intonations, body language, and physiological responses, enabling games to adapt in real-time to the emotional journey of the player [6].

Recent advancements in affective AI have led to its integration into various video games, significantly impacting player experience. For example, *Hellblade: Senua's Sacrifice* [7] uses affective AI to tailor the psychological horror experience, enhancing immersion and emotional impact. Similarly, *Nevermind* [8] employs biofeedback mechanisms to adjust game difficulty based on the player's stress levels, promoting engagement and emotional resilience [9].

Affective AI in video games not only increases player satisfaction and engagement but also opens new avenues for emotional storytelling and educational content. As this technology continues to evolve, the integration of affective AI in video games is expected to become more sophisticated, offering new paradigms in gaming that prioritize emotional intelligence and human-centric design [10].

1.4 Biofeedback in Video Games

Biofeedback mechanisms in video games represent a significant advancement in interactive entertainment, offering a unique method for enhancing player immersion and emotional engagement. By integrating physiological feedback into gameplay, developers can create adaptive gaming experiences that respond in real-time to the player's emotional and physical state. This integration uses sensors to monitor physiological signals such as heart rate, skin conductance, and brain activity, which are then utilized to alter game dynamics based on the player's responses [11].

The integration of biofeedback mechanisms and Dynamic Difficulty Adjustment (DDA) in video games provides a promising avenue for enhancing player immersion and emotional engagement. These technologies enable a personalized gaming experience that adapts in real-time to the player's physiological and emotional states, potentially revolutionizing the way games are designed and played. For example, DDA can adjust the difficulty of challenges based on the player's current stress levels and engagement to maintain a balanced and engaging experience without becoming overly frustrating or excessively easy.

Biofeedback has been used to deepen the emotional impact of gaming by making the gameplay environment more responsive to the player's emotional and physiological states. A game might, for instance, become more challenging as the player's heart rate increases, or it might change the game's soundtrack to reflect changes in the player's emotional state, detected through physiological measures [12].

However, the application of biofeedback in video games also necessitates further research to explore the full range of physiological measures that can be effectively integrated into gameplay. Additionally, understanding the long-term effects of such adaptive systems on player experience is essential. While biofeedback can lead to more immersive and personalized gaming experiences, it also raises questions about player privacy and the psychological impact of highly adaptive systems.

2 Method

Current scholarly discourse around Affective AI in gaming is largely centered on theoretical underpinnings and conceptual models. These models outline the potential for games to adapt dynamically to players' emotional states, aiming to enhance immersion,

satisfaction, and overall engagement through the adjustment of game difficulty, narrative elements, and environmental settings based on real-time emotional analysis [13–15]

A notable gap in the literature is the limited empirical evidence, particularly studies leveraging physiological measures such as EEG, to quantify the impact of Affective AI on player engagement. This highlights a critical area for further investigation, emphasizing the need for empirical research to validate the theoretical models proposed.

The suggested experimental research aims to bridge this gap by delving into the relationship between Affective AI technology and player experience. It seeks to understand how interactions with affective AI systems within video games influence measurable emotional and cognitive responses.

2.1 Hypothesis and Research Questions

The central hypothesis guiding this study posits that "The implementation of affective Artificial Intelligence (AI) in video games enhances player engagement with the video game" (H1).

The study is driven by a research question that aim to explore the intersection of affective Artificial Intelligence (AI) with player engagement in video gaming contexts. This question is designed to uncover how affective AI influences player engagement, and the dynamics between player emotions and game mechanics.

Q1. How does affective AI influence player engagement in video games?

Our study aims to investigate the impact of Affective Artificial Intelligence (AI) on player engagement within the video game "Nevermind," employing both electroencephalogram (EEG) measures and direct observations of player behavior to evaluate the emotional and cognitive responses elicited by gameplay. This dual approach facilitates a nuanced understanding of how Affective AI influences key factors such as interest, excitement, focus, stress, and relaxation during gameplay.

2.2 Research Design

The experimental research design incorporates a control group playing without Affective AI integration and an experimental group playing with Affective AI. This comparative setup allows us to isolate the effects of Affective AI on player engagement.

2.3 Sample

We will include 50 habitual video game players varying widely in gender, age, and gaming experience. This diverse participant pool will help in understanding how different demographics interact with and respond to Affective AI technologies.

2.4 Variables

In the proposed study, we aim to comprehensively analyze the impact of Affective AI on player engagement in video games by examining a series of primary variables that are integral to understanding both the physiological and behavioral dimensions of gaming interactions. These variables, including physiological responses measured via EEG and observed behavioral changes, are instrumental in evaluating how Affective AI modifies player experiences and responses in real-time.

- 1. Interest: Interest is gauged by measuring the level of engagement a player has with the game content. This variable is quantified through both direct observation (e.g., time spent on specific tasks, frequency of game interactions) and physiological indicators such as EEG patterns that denote heightened attention. Interest reflects how captivated players are by game scenarios and tasks, indicating the success of Affective AI in maintaining player attention and engagement.
- 2. Excitement: Excitement is assessed by analyzing the intensity of emotional arousal in response to specific game events. This includes physiological responses such as increased heart rate and EEG spikes, which signify excitement. Behavioral indicators might include increased speed of responses or more animated physical expressions. Moments of peak stimulation or challenge are identified and correlated with game events to determine how Affective AI can elicit and amplify these emotional peaks.
- 3. Focus: Focus is monitored by measuring the concentration and attention players direct towards gameplay. This is primarily evaluated using EEG data that tracks cognitive engagement and neural effort, alongside eye-tracking technology to observe gaze fixation and blink rates during tasks. This variable helps in understanding how Affective AI aids in sustaining player focus, especially during complex or strategically demanding segments of the game.
- 4. Stress: Stress is identified through levels of psychological pressure or tension, which are especially prevalent during challenging or intense game segments. Measurements include changes in cortisol levels (if applicable), heart rate variability, and specific EEG patterns associated with stress, such as high beta wave activity. Behavioral signs of stress, like agitation or hesitation during gameplay, are also noted. This variable assesses the efficacy of Affective AI in modulating game difficulty to maintain an optimal level of challenge without overwhelming the player.
- 5. Relaxation: Relaxation is evaluated through states of calmness and reduced stress, observable during less demanding or restorative phases of the game. Physiologically, relaxation may be indicated by increased alpha wave activity in the EEG and lower heart rate variability. Behaviorally, more relaxed postures and slower, more deliberate actions may be evident. This variable is crucial for examining how Affective AI

can create a balanced game experience that alternates between tension and relief, enhancing overall engagement and enjoyment.

2.5 Instruments and Measurement Techniques

We will use EEG devices to measure brain activity and heart rate monitors to capture physiological reactions. Behavioral observations will record physical reactions, emotional expressions, and interactions with the game environment.

2.6 Materials

Materials include the video game "Nevermind" equipped with and without Affective AI features, as well as the EEG and heart rate monitoring equipment.

3 Experimental Methodology

Participants will be divided into two groups for this study: one will engage with "Nevermind" featuring the Affective AI capabilities, and the control group will experience the game without these adaptive features, aiming for a comparative analysis of engagement levels.

3.1 Detailed Experimental Procedure

- **Preliminary Briefing and Consent:** At the outset, participants will be briefed on the study's aims, followed by the collection of informed consent. They will then be equipped with EEG devices and heart rate monitors to gather physiological data.
- **Gameplay Experience:** "Nevermind" is a psychological horror adventure game developed by Flying Mollusk that uses biofeedback technology to adapt its gameplay based on the player's physiological responses, such as heart rate and skin conductance. Set in the surreal landscapes of trauma survivors' subconscious minds, players assume the role of a Neuroprober—a therapist tasked with uncovering and healing hidden traumas.

The game's environments and puzzles intensify as players' stress levels increase, monitored via biofeedback. Higher stress levels make the game more difficult, with more challenging obstacles and a more threatening atmosphere, heightening the psychological tension. Conversely, a calm demeanor simplifies challenges, fostering a cycle that encourages players to manage stress effectively and build resilience.



Fig. 1. “Nevermind” Screenshot

In our study, we categorize participants into two groups to evaluate the influence of Affective AI. The experimental group will play "Nevermind" with active biofeedback, which adjusts the game's difficulty and narrative in real-time based on detected physiological changes. For instance, an increase in heart rate and skin conductance could lead to a decrease in difficulty or narrative adjustments to alleviate stress, aiming to optimize engagement and challenge.

Conversely, the control group will play the game in its standard format, without biofeedback enhancements. This setup allows for a direct comparison of biofeedback's impact on player immersion, emotional engagement, and overall experience by examining differences in gameplay dynamics and player performance across the two groups.

- **Observation of Behavior and EEG Data Collection:** Our study will employ continuous observation and EEG data collection to comprehensively document players' engagement, emotional reactions, and physiological responses during gameplay. We will utilize the Emotiv Insight EEG device, renowned for its capability to measure and analyze brainwaves in real-time. This device will allow us to precisely assess various cognitive and emotional states that are crucial to understanding the impact of Affective AI in the game environment. The EEG measurements with Emotiv Insight will focus on key variables indicative of the players' cognitive and emotional conditions, including interest, excitement, focus, stress and relaxation.

By integrating the Emotiv Insight EEG device into our study, we will be able to obtain a detailed and dynamic portrayal of how biofeedback mechanisms within "Nevermind" influence the player's psychological and physiological state. This approach not only enhances the accuracy of our data but also enriches our analysis of

how Affective AI can modify gaming experiences to foster better emotional management and overall engagement.



Fig. 2. Emotiv Insight Emotional Metrics Display

- **Post-Game Assessment:** Following the gameplay session, participants will complete a comprehensive questionnaire, derived from established psychometric scales, to report their subjective experiences and emotional responses. This questionnaire is designed to assess multiple dimensions of player interaction and psychological impact, including:
 - **Emotional Impact:** Questions will probe the emotional responses to the game, such as feelings of anxiety, excitement, fear, and relief, to gauge the emotional intensity and range elicited by the gameplay.
 - **Cognitive Load:** Participants will be asked about the mental effort required during the game, including periods of high complexity or confusion, to assess the cognitive demands placed by the game.
 - **User Engagement:** This will include questions on the level of immersion and interest throughout the game session, evaluating how effectively the game maintained player attention and engagement.
 - **Satisfaction and Enjoyability:** Queries will measure overall satisfaction with the gaming experience, including aspects of game design like storyline, challenges, and interactive elements.
 - **Behavioral Intentions:** We will assess participants' likelihood to recommend the game to others and their interest in playing the game again, which are indicators of the game's success in engaging players.

- Stress and Relaxation Levels: The questionnaire will ask about moments of high stress and relaxation, linked to the biofeedback data, to see how players perceived these instances in relation to physiological measures recorded by the EEG.
- The responses to these questions will provide valuable insights into how the integration of biofeedback mechanisms and Affective AI technologies influence the psychological and emotional aspects of gaming experiences.

3.2 Data Analysis

The study will employ a mixed-methods approach during the analysis phase to provide a comprehensive understanding of Affective AI's impact on player engagement. Quantitative analysis will primarily focus on the examination of physiological data obtained through EEG, analyzing variables such as interest, excitement, focus, stress, and relaxation. This will involve the use of statistical techniques suitable for neurophysiological data, including variance analysis and correlation analysis, to assess the direct effects of Affective AI on these emotional and cognitive responses.

In addition to physiological data, the quantitative analysis will also incorporate behavioral observations recorded during gameplay. This data will be analyzed to identify patterns and correlations between observed behaviors and EEG-measured emotional states, further enriching our understanding of how Affective AI influences player engagement.

Qualitative analysis will complement the quantitative methods by interpreting the observed behaviors, providing insights into the nuances of player interaction with Affective AI. This may include coding and thematic analysis of behaviors and expressions, aiming to uncover deeper layers of engagement and emotional resonance that may not be fully captured through quantitative measures alone.

Together, these methods will allow for a holistic analysis of Affective AI's effects, integrating both the objective physiological impacts and the subjective experiences of players.

4 Expected Results and Contributions

The objectives of our study are multifaceted and aim to rigorously test and explore the capabilities of Affective AI within the context of video gaming:

- Empirical Test of Affective AI: We aim to provide robust empirical evidence on the impact of Affective AI in enhancing player engagement and fostering more immersive gaming experiences. This will be achieved through controlled experiments comparing player interactions with and without Affective AI enhancements.

- **Exploration of Emotions and Game Dynamics:** Our research seeks to delve into the complex interplay between player emotions and game dynamics. The goal is to understand how Affective AI can dynamically alter gaming experiences in response to player emotions, thereby informing the development of guidelines for creating emotionally engaging games.

Expected Outcomes:

- **Enhancement of Engagement Indicators:** We anticipate documenting a significant enhancement in player engagement as a direct result of Affective AI interventions, validated through both objective measures (such as EEG and biofeedback data) and subjective assessments (including detailed questionnaire feedback).
- **Improvement of Subjective Experience:** The study expects to reveal significant improvements in players' subjective gaming experiences, characterized by increased levels of immersion, satisfaction, and enjoyment. These improvements are anticipated to stem from the adaptive and personalized game mechanisms powered by Affective AI.
- **Test of Hypotheses:** By comparing engagement metrics among participants interacting with Affective AI versus a control group, we expect to confirm the hypothesis that Affective AI substantially enriches the gaming experience through its capacity for real-time emotional adjustments.
- **Discovery of Emotional and Cognitive Insights:** We aim to identify specific emotional triggers and cognitive patterns that Affective AI targets, which could lead to optimized gaming experiences. This exploration will contribute to the understanding of how emotional and cognitive responses can be effectively harnessed and manipulated within digital environments to enhance player interaction and satisfaction.
- **Game Design Implications:** The findings from this study are expected to have profound implications for game design, emphasizing the critical role of Affective AI in creating engaging and emotionally resonant gaming experiences. Additionally, our research will provide valuable insights into integrating biofeedback and affective computing technologies into game development, potentially setting new standards for the industry.

By achieving these outcomes, our study will contribute significant new knowledge to the fields of interactive entertainment and human-computer interaction, offering a deeper understanding of the practical applications and benefits of Affective AI in enhancing digital experiences.

5 Ethical Considerations

The study examining the impact of Affective AI on video game engagement is conducted in strict adherence to key ethical principles, under the guidance of the Ethics Committee of the Universidad de Deusto.

- **Informed Consent:** Participants are fully informed about the study's purpose, the procedures involved, potential risks, and their rights. Consent is obtained in writing before participation begins, ensuring that all participants voluntarily agree to take part after having all their questions answered.
- **Privacy and Anonymity:** All participant data collected during the study is anonymized to protect individual privacy. Identifiable information is detached from the datasets to prevent any potential identification of participants from the results.
- **Participant Welfare:** Comprehensive measures are in place to ensure the well-being of all participants. This includes monitoring during gameplay to quickly address any discomfort or stress. Participants are also informed of their right to take breaks at any point during the study and can withdraw from the study without any consequences.
- **Ethical Oversight:** This study has been reviewed and approved by the university's human research ethics committee, which ensures that all experimental procedures comply with both national and international ethical standards.
- **Right to Information and Debriefing:** Participants have the right to receive information about the outcomes of the research. After the study, a debriefing session is conducted to explain the findings and their implications. Participants are also given access to the final research report upon request.
- **Right to Withdraw:** Participants are made aware of their unconditional right to withdraw from the study at any time, even after it has commenced, without any penalty or loss of benefits to which they are otherwise entitled.

These protocols are implemented to ensure that the research upholds the highest ethical standards, respecting the dignity, rights, and welfare of all participants involved.

6 References

1. Bontchev, B.: Adaptation in Affective Video Games: A Literature Review. *Cybernetics and Information Technologies*. 16, 3–34 (2016). <https://doi.org/10.1515/cait-2016-0032>.
2. Ravichandran, K., Ilango, S.K.: Influence of AI Powered Gaming Developers and Analyzing Player Behavior and Enhancing User Experience. In: 2023 First International Conference on Advances in Electrical, Electronics and Computational Intelligence (ICAEECI). pp. 1–9. IEEE, Tiruchengode, India (2023). <https://doi.org/10.1109/ICAEECI58247.2023.10370952>.
3. Christy, T., Kuncheva, L.I.: Technological Advancements in Affective Gaming: A Historical Survey. *GSTF J Comput.* 3, 38 (2014). <https://doi.org/10.7603/s40601-013-0038-5>.
4. Iacovides, I., Aczel, J., Scanlon, E., Taylor, J., Woods, W.: Motivation, Engagement and Learning through Digital Games: *International Journal of Virtual and Personal Learning Environments*. 2, 1–16 (2011). <https://doi.org/10.4018/jvple.2011040101>.
5. Przybylski, A.K., Rigby, C.S., Ryan, R.M.: A Motivational Model of Video Game Engagement. *Review of General Psychology*. 14, 154–166 (2010). <https://doi.org/10.1037/a0019440>.
6. Makantasis, K., Liapis, A., Yannakakis, G.N.: From Pixels to Affect: A Study on Games and Player Experience, <http://arxiv.org/abs/1907.02288>, (2019).
7. *Ninja Theory: Hellblade: Senua’s Sacrifice*, (2017).
8. *Flying Mollusk: Nevermind*, (2015).
9. Avontuur, T., Spronck, P., Van Zaanen, M.: Player Skill Modeling in Starcraft II. *AIIDE*. 9, 2–8 (2021). <https://doi.org/10.1609/aiide.v9i1.12682>.
10. Hamrouni, A., Bendella, F.: A taxonomy of learner-player’s emotions in serious games. *IJSG*. 10, 17–32 (2023). <https://doi.org/10.17083/ijsg.v10i4.637>.
11. Fairclough, S.H.: Fundamentals of physiological computing. *Interacting with Computers*. 21, 133–145 (2009). <https://doi.org/10.1016/j.intcom.2008.10.011>.
12. Nacke, L., Lindley, C.A., Lindley, C.: Flow and Immersion in First-Person Shooters: Measuring the player’s gameplay experience.
13. Burak Arslan, R., Filiz, E.: Enhancement of Player Experience in Video Games Using EEG Based Dynamic Difficulty Adjustment. In: 2022 International Conference on INnovations in Intelligent SysTems and Applications (INISTA). pp. 1–4. IEEE, Biarritz, France (2022). <https://doi.org/10.1109/INISTA55318.2022.9894125>.
14. Fraser, J., Papaioannou, I., Lemon, O.: Spoken Conversational AI in Video Games: Emotional Dialogue Management Increases User Engagement. In: Proceedings of the 18th International Conference on Intelligent Virtual Agents. pp. 179–184. ACM, Sydney NSW Australia (2018). <https://doi.org/10.1145/3267851.3267896>.
15. Ruqeyya, G., Hafeez, T., Saeed, S.M.U., Ishwal, A.: EEG-based Engagement Index for Video Game Players. In: 2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE). pp.

1–6. IEEE, Lahore, Pakistan (2022).
<https://doi.org/10.1109/ETECTE55893.2022.10007386>.

Fig. 1. “Nevermind” Screenshot 7
Fig. 2. Emotiv Insight Emotional Metrics Display 8

Valedores do Campus: Diseño de un *serious game* para la divulgación de la oferta académica del Campus Industrial de Ferrol (UDC)

M. Luz Castro¹, Abrahan Dopazo², Cristina García³

¹ Universidade da Coruña, Centro de Investigación en Tecnologías de la Información y las Comunicaciones, A Coruña, España

² Universidade da Coruña, Área de Ingeniería cartográfica, Geodésica y Fotogrametría, A Coruña, España

³ Universidade da Coruña, Grupo de Innovación Docente en Prod. Periodística y Audiovisual, A Coruña, España

Resumen

Se presentan los resultados de la etapa de definición visual de un videojuego serio, actualmente en desarrollo, enfocado en la divulgación de la oferta académica del Campus Industrial de Ferrol (CIF) de la Universidad de A Coruña (UDC). Se expone el proceso llevado a cabo desde los departamentos de diseño y arte para la elección del tipo de videojuego, personajes y localizaciones para atraer al estudiantado de educación secundaria, su público objetivo.

Abstract

Presentation of the results of the visual definition phase of a serious video game which is currently in development, focused on the dissemination of the academic offer of the Ferrol Industrial Campus of the University of A Coruña. The process used by the design and art departments to select the type of video game, characters and locations to appeal to the target audience of secondary school students is presented.

Palabras clave / Keywords

Videojuego, Juego Serio, Divulgación / Video Game, Serious Game, Dissemination

1. Introducción

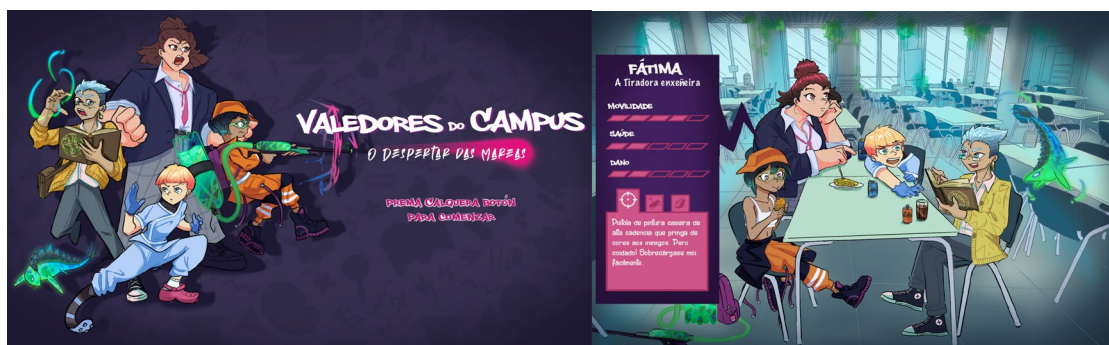
Valedores do Campus es un proyecto desarrollado desde el Máster en diseño, desarrollo y comercialización de videojuegos de la Universidad de A Coruña (UDC), España. Este proyecto nace con el objetivo de realizar un videojuego para promover la divulgación de la oferta académica del Campus Industrial de Ferrol (CIF) y suscitar interés en los títulos de la misma entre el estudiantado de educación secundaria, su público objetivo.

El abanico de opciones que se presenta ante la población adolescente al terminar sus estudios de bachillerato es muy amplio y la elección de titulación universitaria, en un momento crucial que definirá su futuro, supone en muchos casos una gran dificultad. A esta situación se le añade la complejidad de comunicar adecuadamente la oferta de títulos disponible, especialmente desde campus periféricos. Vázquez y Manassero [1] destacan entre las medidas para mejorar la orientación académica y profesional, en relación con la ciencia y la tecnología en la secundaria, la priorización de actividades que estimulen el interés hacia las mismas, y precisamente, debido a la esencia tecnológica de muchas de las titulaciones impartidas en el CIF, surgió la motivación de realizar un videojuego que ayude a conectar a la juventud con todas las posibilidades formativas que se ofrecen, de una forma atractiva y fuera de lo habitual. Por ello se recurrió al enfoque de este proyecto como un juego serio, lo cual cumple con el objetivo deseado debido a "la doble naturaleza de los *serious games*; por un lado, su dimensión «videolúdica» (orientada al entretenimiento) y, por otro, su dimensión sería entendida como un propósito formativo, didáctico o comunicativo." [2]

Tras un análisis de las disciplinas académicas impartidas en el CIF, se realizó una selección de cinco títulos que personifican los personajes principales del proyecto: Gestión Industrial de Moda, Ingeniería mecánica, Diseño Industrial, Enfermería y Dirección Laboral. Uno de los motivos principales para elegir estas titulaciones se debe a la oportunidad que brindan al proyecto de diseñar personajes que luchan contra los roles de género asociados a estas áreas, así como dar una visión genérica de las distintas ramas y titulaciones existentes.

Las mecánicas del juego se corresponden a las del género *beat'em up*, caracterizado por un ritmo muy dinámico y donde prima el combate constante, junto con características de aventura gráfica, en lo relativo a profundidad narrativa y desarrollo de la historia. La elección del género *beat'em up* para este proyecto se decide tras un análisis de déficits de los *serious games* clásicos para conectar con el público joven, donde se acostumbra a tomar una perspectiva de juego enfocado a diálogos informativos, minijuegos y, en general, en los que el ritmo es mucho más pausado, lo cual dificulta que los jugadores conecten con las mecánicas y acaben perdiendo interés en el mensaje de estos proyectos. Uno de los problemas más importantes en el desarrollo de juegos serios ha sido conservar el elemento lúdico, su diversión [3]. En muchos casos se ha potenciado su misión transmisora de conocimiento en detrimento de lo lúdico [4], algo que se quería evitar. Tal como indican Caserman et al. [5], los juegos serios de alta calidad deben integrar el objetivo en la jugabilidad, de modo que la participación en la parte seria sea obligatoria para jugar al juego, además de ser divertidos y amenos, manteniendo el equilibrio entre la parte seria y la parte lúdica.

Como videojuegos referentes respecto a las mecánicas se pueden destacar *River City Girls* (WayForward, 2019), *Have a Nice Death* (Magic Design Studios, 2022), *Streets of Rage 4* (DotEmu, Lizardcube, Guard Crush, Seaven Studio, 2020) y *Young Souls* (1P2P, 2022).



Figuras 1 y 2. Diseño del menú inicial y concept art de la pantalla de selección de personaje realizado por Ramón Torres Chaves.

El juego está dirigido a jóvenes entre 16 y 20 años, que se pondrán en el lugar de uno de los personajes, a su elección, contando con distintas habilidades según la titulación que encarnan, con la misión de salvar el CIF de un espíritu ancestral de las aguas que ha maldecido a todos los adultos conformistas, convirtiéndolos en enemigos acuáticos e inundando toda la ciudad, haciendo referencia al cambio climático y las consecuencias de permanecer sin tomar medidas. El objetivo de la experiencia será demostrar que la humanidad merece tener otra oportunidad ante el esfuerzo de las nuevas generaciones y los conocimientos que están adquiriendo en las titulaciones mencionadas anteriormente.

Cabe destacar que el proyecto se encuentra en una fase de preproducción intermedia, en la que se ha trabajado en los principios básicos de diseño y mecánicas para el juego completo, con un diseño de niveles planteado para el primer capítulo (de un total de cinco) y un desarrollo avanzado en el apartado artístico que incluye diseño de personajes e interfaces, lo que ha permitido ejecutar una previsualización del acabado deseado, *look and feel* y de las interacciones básicas incluidas en el videojuego final.

La ejecución del videojuego se está llevando a cabo bajo un marco alejado del modelo clásico de creación para este tipo de obras, debido a un factor principal: el desarrollo del proyecto se lleva a cabo como parte del programa de prácticas del Máster en diseño, desarrollo y comercialización de videojuegos del CIF, por lo que tanto los períodos como el flujo de trabajo (*pipeline*) se ha adaptado al calendario de prácticas del alumnado. Para asegurar la calidad y correcto desarrollo se cuenta con la colaboración y supervisión de profesionales del sector.

2. Interpretación visual de la realidad

Uno de los objetivos de este proyecto pasa por familiarizar a los jugadores, no solo con la oferta académica del CIF, sino también con el entorno en el que se encuentran sus instalaciones y en el que, finalmente, se aspira a que acudan en el futuro como estudiantes.

El estilo visual se caracteriza por la fusión de personajes, interfaces y efectos visuales en 2D, junto a fondos y escenarios en 3D. La elección del estilo visual tiene el objetivo de destacar los personajes sobre el fondo, además de contribuir a una economización de recursos. Una de las características principales del diseño de arte es el estilo *cartoon*, que se puede apreciar en el delineado de los elementos figurativos de las escenas. El estilo de dibujo, especialmente en los personajes, se basa en colores vivos planos con sombras poco marcadas que contrastan con los fondos, trabajados de forma más abstracta, con línea más fina y en tonos más apagados para mantener la atención siempre en los protagonistas de la historia (figuras 1 y 2).



Figuras 3, 4 y 5. Diseño de Nanda y Suso, realizados por Ramón Torres Chaves. Fotograma del gameplay.



Figuras 6 y 7. Detalle de la interfaz de selección de nivel realizado por Ramón Torres Chaves y fotografía de referencia de la fachada de la Facultad de Enfermería y Podología tomada por Abrahan Dopazo.

El diseño de los personajes se realizó teniendo en cuenta la diversidad social actual y con el claro objetivo de romper estereotipos. Como ejemplo destacamos los diseños de Nanda y Suso, de las titulaciones de Relaciones Laborales y Enfermería, respectivamente. Además de subvertir los roles de género asociados a estas ramas de conocimiento, Nanda será la líder del equipo y la discapacidad de Suso se convertirá en el juego en un punto fuerte a la hora de luchar contra los enemigos (figuras 3 y 4). El proceso de ambientar el juego en un entorno real ha pasado por un proceso de análisis de las localizaciones más emblemáticas del entorno del campus seguido de una interpretación en la línea general, *cartoon* y simplificada, del resto de elementos visuales (figuras 6 y 7). Las primeras iteraciones de las animaciones de los personajes y los enemigos se han implementado en el motor de videojuegos, Unreal Engine 4.27 (Epic Games, 2021) (figura 5).

3. Conclusión y futuro del proyecto

La elección de un videojuego como medio de comunicación entre una institución académica y su público objetivo, jóvenes preuniversitarios, es una apuesta novedosa que pretende romper con el proceso clásico de difusión y recepción de la oferta académica. Mediante este proyecto se fomenta la inclusión de la tecnología y el uso de contenido interactivo como alternativa al despliegue por vías tradicionales de información relativa a las titulaciones.

El proyecto está siendo desarrollado en su totalidad por alumnado del Máster en diseño, desarrollo y comercialización de videojuegos del CIF (UDC), con la supervisión de profesionales del sector y docentes, evidenciando el potencial del estudiantado que acude al CIF para formarse y mostrando los niveles de calidad que alcanzan. Por ello, este proyecto cumple un objetivo extra: ser un ejemplo de lo que el público objetivo podría llegar a crear en un futuro, sirviendo como producto que cumple su función divulgativa y, además, genera una expectativa alentadora entre las personas jugadoras.

La labor de producción de este proyecto está planteada en un horizonte temporal de tres años desde su inicio. El trabajo futuro incluye la iteración del diseño de mecánicas y diseño narrativo actual, la implementación de los sistemas de control en el motor de videojuegos y la continuación del desarrollo artístico y visual con la finalidad de atraer la atención del público objetivo, no solo para informarse, sino, ante todo, para entretenerse.

Agradecimientos

Agradecemos a Ana I. Ares Pernas, Vicerrectora del Campus de Ferrol y Responsabilidad Social de la UDC, así como a su predecesora, M.^a Jesús Movilla, y a Marcos Míguez González, director del CIF, por el apoyo brindado al proyecto y la confianza depositada en nosotros. Agradecemos al alumnado que forma parte del equipo de desarrollo, Adrián Iglesias Adrover y Ramón Torres Chaves, su ilusión, creatividad y trabajo realizado.

Parte de este trabajo se ha realizado en el marco del Grupo de Investigación RNASA - IMEDIR (Redes de Neuronas Artificiales y Sistemas Adaptativos - Imagen Médica y Diagnóstico Radiológico, G000282) de la Universidade da Coruña, que ha recibido una ayuda de consolidación de la Xunta de Galicia (ref: ED431C 2022/46).

Referencias

- [1] Vásquez, A. y Manassero, M. A. (2008). La elección de asignaturas de ciencias: análisis de los factores determinantes. *Revista Española de Pedagogía*, Vol. 66, No. 241, 541-558.
- [2] Álvarez, J. y Djaouti, D. (2010). *Introduction au serious games*. París: L>P Questions Théoriques, p. 17.
- [3] López Raventós, C. (2016). El videojuego como herramienta educativa. Posibilidades y problemáticas acerca de los serious games. *Apertura*, Vol. 8, No 1, ISSN 2007-1094.
- [4] García, M., Cortés, S. y Martínez, R. (2011). De los videojuegos comerciales al currículum escolar. Las estrategias del profesorado. *Revista Icono* 14, vol. 9, No. 2, pp. 249-261.
- [5] Caserman, P., Hoffmann, K., Müller, P., Schaub, M., Straßburg, K., Wiemeyer, J., Bruder, R. y Göbel, S. (2020). Quality Criteria for Serious Games: Serious Part, Game Part, and Balance. *JMIR Serious Games*, vol. 8, No 3, e19037.

Victoria la Malagueña: La primera guía turística 3D conectada a un LLM

David Bueno Vallejo¹

¹ Dpto. Lenguajes y Ciencias de la Computación, Universidad de Málaga
David.bueno@uma.es

Abstract. En este artículo se presenta la evolución de los sistemas conversacionales de chatbots para el asistente de Google utilizando tecnologías como Dialogflow a Avatares 3D conectados a grandes modelos de lenguaje(LLM). Se analiza toda la complejidad necesaria que implica las conversiones de texto a voz, de voz a texto y ajuste de visemas para las posiciones de la boca del avatar que permitan sincronizarse con el audio.

Keywords: LLM, SpeechToText, TextToSpeech, Visemas, Unreal Engine.

1 Introducción

Desde noviembre de 2022 con la aparición de chatGPT se ha producido una democratización y avance del uso de la Inteligencia Artificial como no se había visto nunca. Los sistemas generativos capaces de crear texto, imágenes, sonidos o videos han empezado a popularizarse con un acceso para casi todos los públicos [1]. Las Administraciones Públicas deben ir pensando como poder adaptarse en el uso de estas tecnologías para poder optimizar su trabajo diario sin caer en el miedo de que la Inteligencia Artificial pueda quitar trabajo. Hay que verlo como una oportunidad para poder mejorar los procesos existentes. Una de las posibles aplicaciones son los asistentes virtuales que puedan ayudar al ciudadano en sus interacciones con las administraciones. Estos están utilizándose con mayor o menor éxito desde 2019 tanto para temas administrativos, como para temas turísticos. Estos asistentes virtuales o chatbots son capaces de detectar la intención del usuario y responder a preguntas frecuentes, proporcionar información sobre trámites o hacer recomendaciones sobre que visitar en una ciudad. Algunos ejemplos pueden ser Gestri [2] (chatbot de Agencia Tributaria de Valencia), Rosi [3] (Ayto. Murcia) o Victoria la Malagueña [4] (Ayto. de Málaga). Pero todos esos tipos de chatbots basados en intenciones han tenido una edad de oro hasta ahora, aunque tienen algunas limitaciones importantes y es que si hay algo que no se le ha programado como pregunta darán una respuesta del tipo “no entiendo la pregunta” que es lo peor que le puede pasar a un asistente virtual. La llegada de los LLM (*Large Language Models* – Grandes Modelos de Lenguaje) da una vuelta a todos esos sistemas ya que no es necesario tener prevista todas las interacciones, pues estos sistemas como chatGPT o Bard recogen prácticamente toda la información que pudiera ser necesaria en sus modelos para responder a preguntas de cualquier tipo, con la ventaja de que podemos

enriquecerlos de forma muy sencilla para que los temas de nuestro dominio específico se respondan con la información que nosotros les suministremos.

Además de eso se ha visto en varios estudios como [6] que es muy importante para muchos usuarios que cuando interactuamos con un chatbot, éste sea lo más humano posible. Por lo tanto, además del texto, es importante que tenga voz y una apariencia lo más realista posible. Lo que es cierto es que esto complica todo un poco. Es decir, si nos centramos por ejemplo en el turismo y queremos una interacción lo más realista posible, no basta con tener un chatbot que tenga un conocimiento del ámbito en el que estemos, ni que pueda darnos respuestas abiertas a casi cualquier pregunta, sino que además tenga una apariencia humana.

La complejidad para conseguir ese resultado es que es necesario combinar múltiples tecnologías. En la Fig. 1 se muestra el flujo de trabajo necesario para conseguir ese procesamiento.

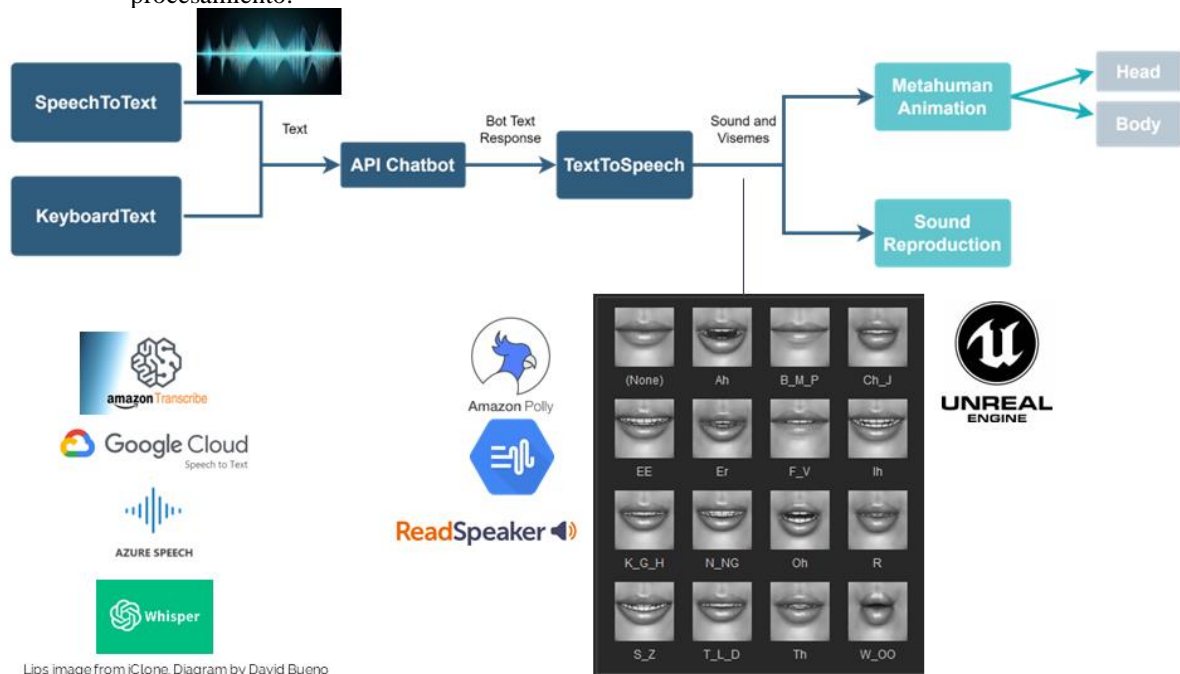


Fig. 1. Flujo de trabajo completo

1. **Recogida información del usuario.** El usuario de nuestro sistema podrá utilizar texto o voz como punto de entrada. En el caso en que la entrada sea por voz, ese archivo de audio debe analizarse, detectar en qué idioma está y convertirlo a texto. Afortunadamente existen en la actualidad sistemas de Voz a Texto (STT – *Speech to Text*) disponibles en la nube con unos tiempos de

respuesta bastante rápidos, por ejemplo: Google Cloud Speech to Text, Azure Speech o Whisper.

2. **Procesamiento de la petición del usuario.** Una vez se tiene el texto del usuario hay que saber qué es lo que está diciendo y cuál es su intención. Para ello debemos usar algún sistema que permita por un lado saber qué es lo que el usuario quiere y por otro, ser capaces de buscar una respuesta. Tradicionalmente se ha utilizado para esto sistemas como Dialogflow, IBM Watson, Microsoft Bot Framework, wit.ai o Rasa en los que una vez se capturaba la intención del usuario se generaba una respuesta más o menos preparada que incluso podía buscar información de servicios externos. La tendencia actual y futura está en el uso de grandes modelos de lenguaje como chatGPT o Bard que permiten unas respuestas más amplias por su gran base de conocimiento, que se complementan con una información específica y restricciones sobre el ámbito del proyecto que normalmente se incluyen como interacciones iniciales (*prompts*). En cualquiera de las opciones el resultado de este paso será un texto con la respuesta.
3. **Generación de voz de la respuesta.** Si se quiere que la respuesta del chatbot sea una respuesta en formato audio y no solo textos, es necesario hacer un proceso inverso al inicial, que consiste en convertir el Texto en Voz (*Text to Speech – TTS*). Para ello también tenemos servicios en la nube como Amazon Poly o la IA de Text-to-Speech de Google, aunque podremos encontrar algunos con capacidad de generar el texto en nuestros servidores como ReadSpeaker.
4. **Generación movimientos cuerpo y cara del Avatar 3d.** Por último, y no por eso más sencillo, si queremos que nuestro interlocutor sea un avatar 3d hay que convertir ese audio en expresiones faciales del personaje 3d y no solo mover la boca para que coincida con el audio, si no también movimiento de cara, ojos, cabeza, cuerpo y manos que den la sensación de que se está hablando con una persona. Las actuaciones en este caso son varias. Por un lado, hay que generar una sincronización lo más realista posible de los movimientos de la boca que deben estar sincronizados con el sonido. Para ello hay que extraer los visemas [7] que son representaciones visuales de la posición de la boca cuando se pronuncia un fonema (ver Fig. 1) y mostrarlas en el momento que se pronuncie su sonido asociado realizando transiciones suaves entre dos sonidos diferentes. Esto hay que completarlo con parpadeos de ojos y leves movimientos de cabeza. Permitiendo en algunos casos poder mostrar emociones. Para el cuerpo se pueden combinar varias animaciones de movimientos muy suaves de piernas y movimientos de manos que den una sensación lo más realista posible. Los avatares 3d más populares que permiten este tipo de interacciones son los de ReadyPlayerMe tipo *cartoon* y los Metahuman de Unreal

Engine más híper realistas. En la Fig 2 puede verse en cada una de las plataformas un avatar del autor.



Fig. 2. a) Avatar en ReadyPlayerMe b) Metahuman de Unreal Engine

En las nuevas tendencias que tenemos relacionadas con mundos virtuales o metaversos donde se quiere que estos avatares puedan acompañar a los usuarios en el descubrimiento de ese mundo se añaden nuevas tecnologías para tener en cuenta relacionadas con el modelado 3d. En este caso es necesario tener la capacidad de diseñar ese mundo 3d y en el caso en que queramos que represente a un lugar real es necesario incluir objetos o lugares generados utilizando fotogrametría es decir generar reproducciones de objetos reales en 3d a partir de cientos de fotos geolocalizadas de dichos objetos. Será necesario desarrollar toda la IA capaz de reconocer las diferentes ubicaciones de ese entorno para que el avatar tome conciencia de qué hay en cada posición y sea capaz de explicárselo al usuario, o llevarlo a cada uno de esos lugares. Esto es necesario tanto si hablamos de museos virtuales, como de mundos en un metaverso o dentro de un juego.

Como resumen, la IA está en todos los lados en este tipo de aplicaciones: voz a texto, texto a voz, análisis del lenguaje natural, aprendizaje automático, análisis de audio para obtención de visemas, *path finding* para calcular rutas dentro del entorno 3d o análisis inteligente de imágenes para generación de modelos 3d.

En el apartado siguiente se verá cómo todas esas tecnologías se han utilizado en la evolución de Victoria la Malagueña como guía turística aplicada a un ejemplo concreto de presentación de visita a la Plaza de la Merced de Málaga, que fácilmente puede extrapolarse a cualquier entorno 3d como otras ubicaciones culturales, museos, mundos virtuales o juegos.

2 Victoria la Malagueña

Para contextualizar el caso específico, en este apartado se muestran por un lado el origen de Victoria la Malagueña como chatbot y a continuación su evolución como avatar 3d para terminar con el caso de uso que se presenta como guía cultural.

2.1 El origen

Victoria aparece en 2019 como un chatbot que intenta dar información de utilidad tanto a ciudadanos como turistas de la ciudad de Málaga. Se basaba principalmente en el uso de los datos del portal de Datos Abiertos de Málaga que eran de interés ya sea por la información que aportan o porqué ofrecen información en tiempo real que pueden ayudar al ciudadano o turista a la toma de decisiones en su estancia en la ciudad. Algunos de los conjuntos de datos más relevantes de los que se originaba la información utilizada en este proyecto eran:

- Ubicación de autobuses de la EMT en tiempo real. (Actualizados cada minuto)
- Líneas y paradas de autobuses EMT
- Líneas y horarios de bus – Paradas, Rutas, Servicios
- Málaga Bici (ocupación de las paradas de bicicleta)
- 103 conjuntos de datos relacionados con equipamientos municipales (Sedes Wifi, ubicación aparcamientos, centros culturales, bibliotecas, mercados, monumentos, teatros, oficinas de registro, telecentros, museos, puntos de información, instalaciones deportivas, salas de exposiciones, belenes, desfibriladores ...)
- Agenda. Todos los eventos de la ciudad categorizados con múltiples parámetros (destinatarios, categoría, especialidad, horario, fecha, organizador,...)
- Varios conjuntos de datos con información del callejero
- Cámaras de tráfico

Además de los datos abiertos también se utilizaban otras fuentes de datos en tiempo real para dar información que no está disponible directamente en el portal como puede ser la información del tiempo en Málaga a través de *OpenWeatherMap* o la información sobre restaurantes y rutas también se consigue de los servicios que ofrece *Google Maps*. La idea es aprovechar los servicios que ya estén disponibles a través de alguna API, para poder dar respuesta a las necesidades de los usuarios. Todo eso se complementaba con una base de datos interna. Entre ellas está todo lo relativo a temas de Málaga: Curiosidades o Personajes de Málaga, expresiones malagueñas o diccionario de palabras malagueñas, que han ido recopilándose de distintas fuentes, pero que el chatbot consulta internamente durante las conversaciones. Internamente se respondía también a las intenciones del usuario relacionadas con contar chistes, consejos o frases célebres.

Para este proyecto se eligió *Dialogflow* que permite que nuestro chatbot esté disponible en múltiples plataformas como el Asistente de Google (en *Android* y en *iOS*), *Facebook*, *Skype*, *Twitter*, *Web*, *Alexa* o *Cortana* entre otros.

Una vez que se conoce cuál es la intención del usuario y se localizan los parámetros, si son necesarios, hay que generar la respuesta. En la Fig. 3 se muestran tres formas de generar la respuesta con nivel de dificultad incremental. En la primera la respuesta se devuelve desde el mismo entorno del reconocimiento, podría darse como ejemplo una respuesta a la intención de descubrir el nombre con la pregunta - ¿Cómo te llamas? - En el segundo nivel implicaría conectar con una aplicación externa al motor, pero hecha por el desarrollador. Por ejemplo, como respuesta a - Cuéntame un chiste - que podría consultarse en una base de datos propia con chistes (A esta conexión se le llama *Webhook*). Y el tercer nivel y más complejo sería conectándose con las fuentes de datos externas, para consultar por ejemplo datos en tiempo real. - ¿Cuándo llegará la línea 7

a la parada 320? – que implica que nuestra aplicación se conecte con el servicio externo de autobuses pasándole los parámetros de la línea y la parada.

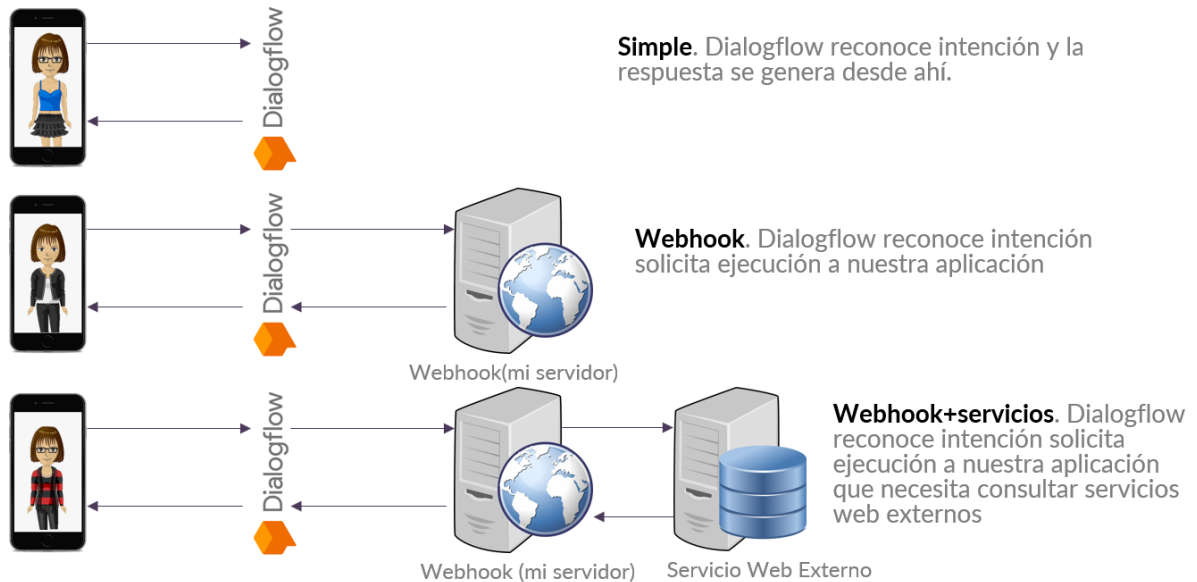


Fig. 3. Formas de obtener las respuestas al usuario

Por otro lado, cuando ya se tiene la información que devuelve el hay que adaptarla utilizando los diferentes recursos disponibles por el chatbot. Donde también habrá que tener en cuenta si es sólo un altavoz, con lo que habrá que simplificar los mensajes y la selección de opciones o si tiene una pantalla que se muestran en la Fig 4 (Respuesta Simple, *Basic card* (texto e imagen), Carousel (grupo de basic cards) y *Suggestion Chips*(sugerencias).

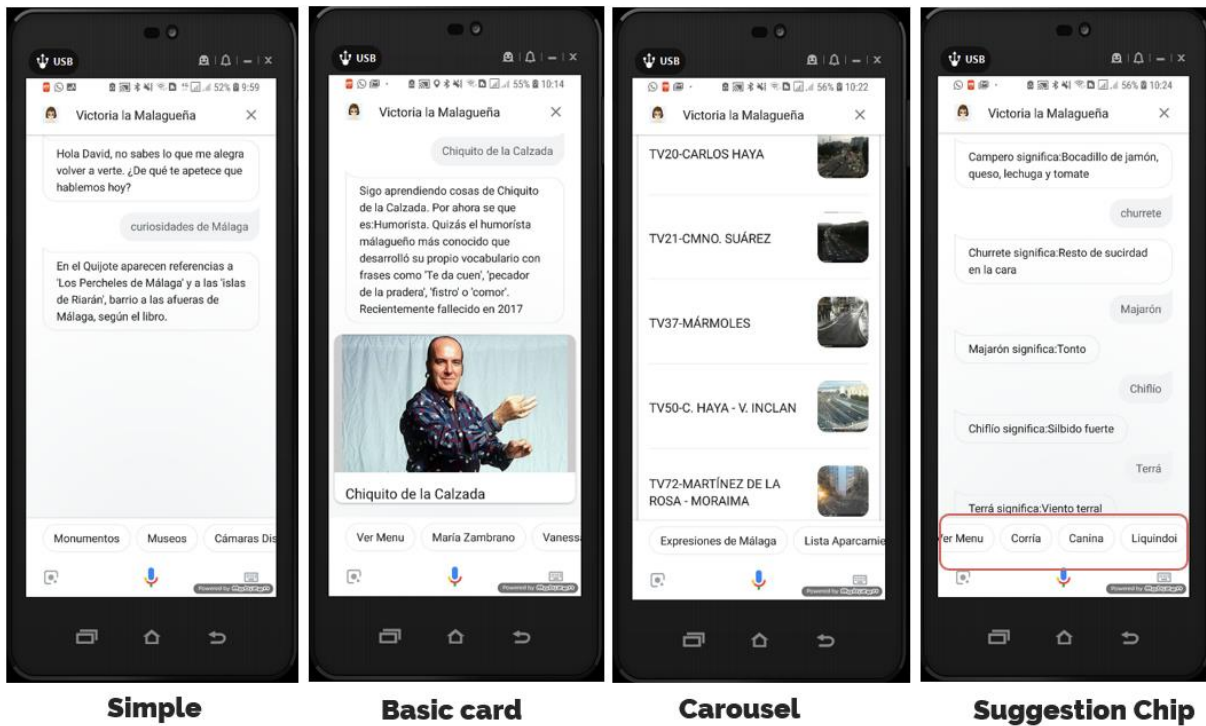


Fig. 4. Principales tipos de respuesta

2.2 La evolución Victoria

Desde hace algo más de dos años el autor ha estado investigando y probando tecnologías para poder convertir al chatbot en un avatar 3d, centrando los diferentes desarrollos en la tecnología relacionada como metahumans de Unreal Engine ya que son los avatares 3d con un mayor nivel de realismo que permite modelar las expresiones de la cara con un alto nivel de detalle cómo puede verse en el metahuman de Victoria en la Fig. 5.



Fig. 5. Metahuman de Victoria la Malagueña en Unreal Engine

Una vez creado el metahuman y conseguido la sincronización de movimientos de la boca con el audio, se integró a Victoria dentro de museos 3d interactivos en el proyecto EVUMAX [8], donde a partir de un audio fijo y pregrabado, se generaba la animación de Victoria explicando el cuadro cuando el usuario se acercaba a éste y lo solicitaba como puede verse en la .Este ya era un paso importante pues se había entrado en el entorno 3d, aunque la interacción no era muy alta y se limitaba a la reproducción de una animación con audio.



Fig. 6. Victoria explicando cuadros en museos de EVUMAX

2.3 Victoria como guía

El avance más importante en su evolución es el que se presenta en este trabajo que se entenderá mucho mejor viendo la demostración que se puede encontrar en: <https://youtu.be/va3JdnVUPhs>. Es un resultado novedoso y original pues en la fecha de la publicación no se conoce ningún caso como el aquí presentado donde un metahuman haga de guía turístico pudiendo recibir instrucciones de voz del usuario, entendiéndolas, desplazándose por el entorno 3d para guiar al usuario y generando tanto la información como el audio, movimientos del metahuman en tiempo real, pudiendo recuperar información de un LLM. Veámoslo con más detalle.

En la Fig. 7 se muestra una captura de la aplicación en la que Victoria hace de guía de la plaza. Esta aplicación está realizada con *Unreal Engine 5* utilizando la herramienta de Convai [10] que facilita todo el proceso visto en la Introducción. Como elemento fundamental está el entorno que ha sido generado usando un complejo sistema de fotogrametría por Visitas Virtuales [9] que aporta una representación a escala real de toda la plaza. Para que Victoria sepa qué hay en la plaza se han añadido una serie de puntos

de interés donde para cada punto se añade la ubicación, un título y una pequeña descripción. Por ejemplo, algunos de esos puntos son la Casa Natal de Picasso, el Obelisco o la Farmacia. Para que pueda desplazarse por la plaza también es necesario activar un sistema de navegación (*Navigation Mesh* en Unreal) que permite poder realizar desplazamientos de un punto a otro. La parte del conocimiento se realiza con el gran modelo de lenguaje ofrecido por NVIDIA Megatron-LM [11] al que se le envía un prompt inicial donde se dan las ideas principales sobre quién es Victoria, sobre su personalidad, donde se encuentra y algunos detalles sobre los elementos principales de la plaza. Se puede también limitar en ese prompt el tipo de preguntas que puede contestar. Cuando el usuario le habla, se mirará primero si la información solicitada está disponible tanto en el prompt inicial, como en las descripciones de los distintos puntos, pero si no aparece o si el usuario solicita más información entonces se consulta al modelo de lenguaje. Todo esto en tiempo real consiguiendo un resultado muy interesante como puede verse en el video demostración que se mostraba al inicio de este apartado.



Fig. 7. Captura de Victoria como guía de la Plaza de la Merced

Conclusiones

En este documento se ha presentado un proyecto en el que un avatar 3d, metahuman hace de guía turístico de una plaza de Málaga explicando con detalle los distintos elementos destacados que en ella aparecen. Se ha mostrado todas las tecnologías que han sido necesaria para conseguirlo en las que la inteligencia artificial tiene un papel imprescindible en la mayoría de ellas. Esto es solo un ejemplo, pero con la experiencia conseguida los autores están en disposición de realizar cualquier tipo de sistema turístico ya sea en exteriores o interiores como un museo donde avatares 3d que se

encuentren en los distintos entornos puedan explicar con detalle cualquier duda que pueda plantear el usuario, ya que el conocimiento que se utiliza como base es prácticamente ilimitado al estar basado en un gran modelo de lenguaje.

References

1. 2023. Eloundou, Tyna et al. GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models. <https://arxiv.org/pdf/2303.10130.pdf>
2. 2023. Chatbot Gestri de Agencia Tributaria de Valencia. <https://1millionbot.com/chatbot-dival/>
3. 2023. Chatbot Rosi de Murcia <https://1millionbot.com/chatbot-ayto-murcia/>
4. 2023. Chatbot Victoria la Malagueña de Málaga. <https://googlechatbots.es/>
5. 2023. Pal, Subharun. The Future of Large Language Models: A Futuristic Dissection on AI and Human Interaction. 5. DOI: <http://dx.doi.org/10.36948/ijfmr.2023.v05i03.3135> .
6. 2023. Cao, Q. et al. Is High-Fidelity Important for Human-like Virtual Avatars in Human Computer Interactions?. *International Journal of Network Dynamics and Intelligence*. 2023, 2(1), 15–23. doi: <https://doi.org/10.53941/ijndi0201008>
7. 2023. Linchao, Bao et al. Learning Audio-Driven Viseme Dynamics for 3D Face Animation <https://arxiv.org/pdf/2301.06059.pdf>
8. 2021 Proyecto EVUMAX. Entornos Virtuales Unreal Maxi-Realistas. <https://youtu.be/CZGBfCHpYqg>
9. Visitas Virtuales. <https://visitasvirtuales.com>
10. Convai <https://convai.com/>
11. 2020. Shoeybi, Mohammad, et al. Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism <https://arxiv.org/abs/1909.08053>

Desarrollo de un videojuego serio para mejorar las funciones ejecutivas en niños

Alejandro Quesada-López^{1,2}[0000-0001-5229-3931], Lucía Caselles-Pina^{2,3}[0000-0001-5791-2527], Aaron Sújara¹[0000-0002-7891-6137], Eva María Garzón Hernández²[0009-0007-9182-8063], Sofía Bayona^{1,4}[0000-0002-6167-5963], Inmaculada Peñuelas-Calvo⁵[0000-0003-4093-1312], Gemma Gallardo Puerta⁶, David Delgado-Gómez²[0000-0002-2976-2602]

¹ Departamento de Informática y Estadística, Universidad Rey Juan Carlos, 28933 Móstoles, Spain

² Department of Statistics, Universidad Carlos III de Madrid, 28903 Getafe, Spain

³ Department of Psychology, Universidad Autónoma de Madrid, 28049 Madrid, Spain

⁴ Center for Computational Simulation, Universidad Politécnica de Madrid, 28040 Boadilla del Monte, Spain.

⁵ Department of Child and Adolescent Psychiatry, University Hospital 12 de Octubre, Madrid, Spain.

⁶ Colegio CEIPS Moncayo, 28941, Fuenlabrada, Spain

Resumen.

Limitaciones en las funciones ejecutivas es uno de los principales problemas que sufren niños con trastornos del neurodesarrollo. Para solucionar esta dificultad, hemos creado un prototipo de videojuego terapéutico que contiene diversos minijuegos destinados a mejorar distintas funciones ejecutivas. Para evaluar su aceptabilidad en población infantil y previa a su despliegue en la población objetivo, se realizó un estudio piloto de cuatro semanas a un grupo de 27 niños de 6 años con desarrollo neurotípico. Tanto los participantes como sus familiares recibieron diversos cuestionarios de satisfacción y usabilidad. Los resultados de este estudio piloto muestran que nuestro videojuego está valorado muy positivamente tanto por los padres de los participantes como por los propios niños. Próximamente este videojuego se someterá a un primer ensayo clínico para demostrar su eficacia en el tratamiento de pacientes con trastorno por déficit de atención e hiperactividad.

Palabras clave: videojuegos serios, funciones ejecutivas.

1 Introducción

Niños con trastornos del neurodesarrollo (TN) presentan dificultades en las funciones ejecutivas (FEs), como son la planificación, la memoria de trabajo o la flexibilidad ¹. Se ha comprobado que estos problemas repercuten de forma negativa en el funcionamiento diario de los niños y adolescentes ².

Desde hace pocos años, profesionales de la salud han explorado nuevas formas para mejorar la calidad de vida de niños con TNs a partir del uso de videojuegos. Los videojuegos serios son juegos terapéuticos o educativos diseñados con la finalidad de

enseñar o entrenar habilidades cognitivas específicas ³. La inclusión de elementos como los videojuegos serios ha demostrado mejorar la síntomas de, por ejemplo, niños con trastorno por déficit de atención e hiperactividad, la adherencia al tratamiento y la motivación de los participantes ⁴.

El presente trabajo tiene como objetivo realizar un estudio piloto para comprobar la idoneidad del juego en niños con edad similar a la edad objetivo. Este estudio nos ayudará a analizar la aceptación y usabilidad de un videojuego serio desarrollado con la finalidad de mejorar las FEs de niños con TNs. Este estudio nos permitirá resolver cualquier problema, deficiencia e incluir mejoras de cara a la validación clínica. En el presente artículo se describen los resultados de este estudio piloto.

2 Método



2.1 Participantes



En este estudio piloto, se decidió buscar una muestra de niños neurotípicos para analizar la usabilidad y satisfacción del videojuego. Esta muestra nos servirá como referencia para futuros estudios en los que se compare con población clínica. Además, el contenido educativo está ajustado a la edad de los participantes. Un grupo compuesto por 27 niños del colegio CEIPS Moncayo con desarrollo neurotípico participaron en este estudio. Los criterios de inclusión fueron (1) tener entre los 6 a los 7 años (1º de primaria) y (2) tener el consentimiento informado de los padres de los participantes. No tenemos constancia de ningún diagnóstico clínico. El protocolo de estudio obtuvo la aprobación ética por el Hospital 12 de Octubre con código de protocolo 23/342 y fecha de aprobación: 25/07/2023.

2.2 Instrumentos

Se administró el videojuego desarrollado a los participantes y los cuestionarios de satisfacción y usabilidad sobre la experiencia con el videojuego tanto a los niños como a los progenitores de estos. En la Tabla 1 se exponen los diferentes minijuegos de los que está compuesto.

Tabla 1. Descripción de los minijuegos del videojuego

<i>Minijuego</i>	<i>Funciones ejecutivas</i>	<i>Descripción</i>
Buscar y tocar frutas 	Atención selectiva Control inhibitorio	Los participantes deben buscar las frutas que se mencionan en las instrucciones y tocarlas.
Memorizar frutas en platos 	Memoria de trabajo Atención selectiva	Los participantes deben memorizar las frutas que van apareciendo en los platos que se les presentan.

<p>Avión</p> 	<p>Atención selectiva Flexibilidad cognitiva</p>	<p>Los participantes deben seleccionar por qué carril va a circular el avión para recoger las estrellas y números que van apareciendo.</p>
<p>Ordenar cajas</p> 	<p>Resolución de problemas</p>	<p>Los participantes deben de ordenar de mayor a menor o viceversa las cajas en función de los números que aparecen sobre ellas.</p>

2.3 Procedimiento

El videojuego serio “Buki y el reino de los libros” es un juego para Android. Los participantes tienen su propio usuario y contraseña para iniciar sesión desde sus casas. El juego consta de varios minijuegos donde se trabajan diferentes FEs de los participantes. El videojuego está programado para que los participantes jueguen 3 veces a la semana durante 15 minutos aproximadamente en un periodo de cuatro semanas.

Los padres de los participantes debían rellenar semanalmente una encuesta de satisfacción de los minijuegos. Al finalizar el estudio, también tuvieron que completar tanto padres como niños, una encuesta de satisfacción final sobre la experiencia del videojuego. Al tratarse de un estudio piloto en una muestra no clínica no se administró ningún cuestionario diagnóstico.

3 Resultados

Respecto a la satisfacción de cada minijuego, se analizaron las encuestas de satisfacción basadas en una escala de Likert de 5 puntos (donde 1 es la menor satisfacción y el 5 la máxima). Como se aprecia en la Tabla 2, nuestros resultados muestran que todos los minijuegos tienen medias iguales o superiores a 4 puntos. Además, las preguntas 5, 6 y 7, que hacen referencia a la usabilidad y satisfacción global del juego, tienen puntuaciones superiores a 4 puntos. A partir de la opinión de los progenitores también se detectaron enunciados que no estaban claros que ha permitido corregirlo previa a su administración clínica.

Tabla 2. Resultados del cuestionario de satisfacción. DE: Desviación estándar

<i>N.º</i>	<i>Pregunta</i>	<i>Media (DE)</i>
1	Del 1 al 5 ¿Cuánto te ha gustado el minijuego de buscar y tocar frutas?	4,46 (0,71)
2	Del 1 al 5 ¿Cuánto te ha gustado el minijuego de memorizar frutas?	4,07 (1,03)
3	Del 1 al 5 ¿Cuánto te ha gustado el minijuego del avión?	4,3 (0,79)
4	Del 1 al 5 ¿Cuánto te ha gustado el minijuego de ordenar cajas?	4 (0,89)
5	El juego es fácil de entender	4,62 (0,52)
6	El juego es fácil de usar	4,75 (0,46)
7	La experiencia con mi hijo/a ha sido positiva	4,62 (0,52)

4 Discusión

Este estudio piloto ha analizado la satisfacción y usabilidad del videojuego desarrollado para mejorar las FEs. Nuestros resultados muestran el videojuego ha sido valorado muy positivamente tanto por los padres de los participantes como por los propios participantes.

El objetivo de hacer un estudio piloto nos ayuda a comprobar que el videojuego está correctamente diseñado tanto para el tratamiento de las TNs como que el contenido educativo se encuentra ajustado para la población objetivo. De manera similar Plan-It Commander ⁵ realizó un estudio de usabilidad antes de llevar su trabajo a una evaluación clínica. Esto les permitió introducir la retroalimentación registrada. Este artículo muestra los primeros pasos del videojuego que desembocará en la creación de un videojuego terapéutico.

Agradecimientos. Agradecemos la colaboración con el CEIPS Moncayo Moncayo que ha sido vital para el desarrollo del trabajo. Este estudio ha sido parcialmente financiado por: la Ayuda TED2021-130980B-I00 financiado por MCIN/AEI/ 10.13039/501100011033 y por la “Unión Europea NextGenerationEU/PRTR”; Ayuda RED2022-134259-T financiado por MCIN/AEI/ 10.13039/501100011033; Proyecto "DTS21/00091", financiado por Instituto de Salud Carlos III (ISCIII) y co-financiado por la “Unión Europea”; Ayuda para la recalificación de profesorado titular, UC3M, David Delgado-Gómez.; y Ayuda PEJ-2021-AI/SAL-2147 financiado por Consejería de Educación e Investigación de la Comunidad de Madrid.

Declaración de intereses. Los autores no tienen ningún interés competitivo que declarar que sea relevante para el contenido de este artículo.

Referencias

1. Pineda-Alhucema, W., Aristizabal, E., Escudero-Cabarcas, J., Acosta-López, J. E. & Vélez, J. I. Executive Function and Theory of Mind in Children with ADHD: a Systematic Review. *Neuropsychol Rev* **28**, 341–358 (2018).
2. Kandemir, H., Kilic, B., Ekinci, S. & Yuce, M. An evaluation of the quality of life of children with attention-deficit/hyperactivity disorder (ADHD) and their families. *Anadolu Psikiyatri Derg* **15**, 265 (2014).
3. Lau, H. M., Smit, J. H., Fleming, T. M. & Riper, H. Serious Games for Mental Health: Are They Accessible, Feasible, and Effective? A Systematic Review and Meta-analysis. *Front Psychiatry* **7**, 209 (2016).
4. Peñuelas-Calvo, I. *et al.* Video games for the assessment and treatment of attention-deficit/hyperactivity disorder: a systematic review. *Eur Child Adolesc Psychiatry* **31**, 5–20 (2022).
5. Bul, K. C. M. *et al.* Development and User Satisfaction of “Plan-It Commander,” a Serious Game for Children with ADHD. *Games for Health Journal* **4**, 502–512 (2015).

Optimización automática del videojuego Doom para un rendimiento óptimo en Steam Deck

José M. Aragón-Jurado, Juan Carlos de la Torre, Patricia Ruiz, and
Bernabé Dorronsoro

Escuela Superior de Ingeniería, Universidad de Cádiz, Cádiz, España
{josemiguel.aragon,juan.detorre,patricia.ruiz,bernabe.dorronsoro}@uca.es

Resumen Los videojuegos contemporáneos se ejecutan en una amplia variedad de configuraciones de hardware. Desarrollarlos sin considerar las especificaciones del hardware donde serán disfrutados puede afectar negativamente en su rendimiento y, por tanto, a la experiencia del usuario. En este trabajo, presentamos una novedosa metodología para optimizar el rendimiento de videojuegos en una arquitectura específica. Para ello, definimos un nuevo problema de optimización combinatoria que consiste en encontrar una secuencia de transformaciones de código LLVM que maximice la tasa de cuadros por segundo de un videojuego en un hardware específico. Además de conseguir aumentar el número de cuadros por segundo, nuestra metodología también permite consumir menor cantidad de recursos cuando se fija el número de cuadros por segundo, lo que podría permitir reducir el consumo energético del juego u ofrecer recursos que puedan ser utilizados para la mejora del mismo. Utilizamos un algoritmo genético celular para encontrar precisas soluciones al problema. El videojuego optimizado es el motor Doom Legacy, una adaptación del motor original Doom Engine, compilado para obtener el rendimiento óptimo en la consola portátil Steam Deck. Para evaluar la eficacia de nuestra solución, la comparamos con diversas opciones de compilación disponibles en LLVM. Los resultados muestran mejoras significativas en la tasa de cuadros por segundo: un aumento del 6,21 % en comparación con el rendimiento del Doom Legacy no optimizado, y un incremento del 3,19 % en relación a la mejor configuración de compilación disponible.

Keywords: Optimización de videojuegos · Doom · LLVM · Algoritmos genéticos. · Steam Deck

1. Introducción

En la actualidad, los videojuegos representan una de las industrias del entretenimiento más influyentes, con ingresos mundiales alcanzando los 182.900 millones de dólares en 2022 [17]. Este crecimiento ha impulsado un mayor interés de las empresas en el sector, resultando en la creación de nuevos productos. Para maximizar sus ganancias, estas empresas deben abordar el desafío de desarrollar videojuegos para una variedad de plataformas, desde las últimas consolas de videojuegos como PS5 y Xbox Series X hasta computadoras personales con

diversas configuraciones de hardware. En este escenario, los motores de juegos han emergido como herramientas populares gracias a su accesibilidad y la capacidad de desarrollar una única versión del videojuego, pudiendo exportarla a múltiples plataformas [21].

Los compiladores son responsables de generar el ejecutable del videojuego para cada una de las consolas y arquitecturas consideradas. Durante este proceso, llevan a cabo una serie de transformaciones de código con el objetivo de mejorar el rendimiento del software. Sin embargo, estas transformaciones son genéricas y no tienen en cuenta ni las características propias del software ni la arquitectura hardware específica en la que se ejecutará el código. Desarrollar videojuegos sin tener en cuenta la arquitectura final que utilizará el usuario puede ocasionar problemas significativos de rendimiento, especialmente con la creciente diversidad de dispositivos disponibles en la actualidad. Estos problemas afectan directamente a la satisfacción del usuario que ha adquirido el videojuego, pudiendo incluso llevar a su devolución.

Uno de los factores críticos relacionados con el rendimiento de un videojuego es la tasa de cuadros por segundo (FPS, de *frames per second*), que es uno de los más influyentes en la experiencia del jugador [6]. Mejorar la tasa de FPS de un videojuego específico en una arquitectura hardware particular no es una tarea sencilla. Se requieren desarrolladores de videojuegos expertos con amplios conocimientos sobre la arquitectura hardware, para aprovechar al máximo todas sus características. Además, las transformaciones genéricas de código aplicadas por los compiladores no siempre logran mejoras significativas en la tasa de FPS, ya que no consideran las propiedades del software y el hardware. Por lo tanto, se hace necesario buscar una secuencia de transformaciones óptima para el videojuego seleccionado y la arquitectura hardware en la que se jugará, a fin de mejorar la tasa de FPS de forma automática.

En la aplicación de transformaciones de código destaca el proyecto LLVM [14]. Se trata de una infraestructura de compilación ampliamente adoptada, constituida por herramientas de compilación modulares y reutilizables, donde destaca su optimizador de código. Dicho optimizador de código fuente es independiente tanto del lenguaje de programación como de la arquitectura subyacente en la que se ejecute. Para ello, las herramientas disponibles se basan en la representación intermedia de LLVM (IR o *Intermediate Representation*), lo que simplifica la adaptación de cualquier compilador de lenguaje para que pueda aprovechar el optimizador proporcionado por LLVM. Sobre esta representación intermedia se aplican las transformaciones de código, que reciben el nombre de *passes*.

No obstante, medir el impacto de los *passes* disponibles no es trivial [20]. Aunque el impacto individual de un *pass* específico sea mínimo, su efecto puede ser significativo cuando se combina con otros, dependiendo del orden en que se aplican los distintos *passes*. Por lo tanto, la secuencia óptima de *passes* para maximizar el rendimiento varía según el programa y el hardware específico, y puede incluir varias repeticiones del mismo *pass*. Además, la medición del rendimiento del software se ve complicada por múltiples variables, como los procesos del sistema operativo, que pueden afectar la consistencia de las mediciones. Para

mitigar este problema, es necesario tomar múltiples mediciones del rendimiento del software en la misma arquitectura hardware y emplear enfoques para reducir la incertidumbre [3].

En este trabajo se extiende el problema SCOP (de *Software Code Optimization Problem*) [3] a la optimización del rendimiento de videojuegos, definiéndose un nuevo problema de optimización combinatoria. Para resolverlo se emplea un algoritmo genético celular (cGA) [2] que busca secuencias de passes LLVM que aprovechen al máximo las características de la arquitectura subyacente, incrementando la tasa de FPS del videojuego. Se optimiza específicamente el rendimiento del Doom Legacy [1], una adaptación del Doom Engine o id Tech 1, en la consola comercial Steam Deck [23].

Las contribuciones que este trabajo aporta son múltiples. En primer lugar, definimos un nuevo problema combinatorio, llamado SCOP-FPS, como una extensión de SCOP para la optimización automática de videojuegos, donde se trata de maximizar la tasa de FPS. En segundo lugar, proponemos una metodología para aplicar la infraestructura de compilación LLVM a grandes proyectos software como son los videojuegos. En tercer lugar, se aplican algoritmos evolutivos para resolver el nuevo problema. Por último, se realiza un estudio comparativo de los efectos de las transformaciones genéricas de LLVM con los resultados obtenidos con nuestro método en el videojuego Doom corriendo sobre la consola Steam Deck, analizando el impacto que tienen en la tasa de FPS.

El resto del documento se organiza de la siguiente forma: en la Sección 2 se describe los diferentes trabajos relacionados con la optimización software. En la Sección 3 se define el nuevo problema de optimización SCOP-FPS, mientras que en la Sección 4 se describe la metodología seguida durante la realización del estudio. En consecuencia, en la Sección 5 se describe la configuración de los experimentos a ejecutar. Finalmente, en la Sección 6 se muestran y discuten los resultados obtenidos, y en la Sección 7 se comentan las conclusiones del trabajo y sus posibles líneas de trabajo futuro.

2. Estado del arte

El código de un software puede ser modificado automáticamente para mejorar sus características y/o capacidades, a lo que se denomina mejora automática del software. Esta sección se centra en la mejora del rendimiento en videojuegos, donde se distinguen dos líneas de investigación principales: la paralelización automática del código y las modificaciones del código fuente para mejorar aspectos importantes como el tiempo de ejecución o el uso de memoria.

2.1. Paralelización automática del código

La investigación en paralelización automática comenzó con transformaciones simples de código para paralelizar programas secuenciales [12]. Estos estudios abordaban métodos tales como la paralelización explícita del código, la modificación de estructuras de bucle y la paralelización de funciones recursivas [5],

técnicas que continúan teniendo importancia en el desarrollo de compiladores paralelos [15]. Las aproximaciones más recientes han introducido la descomposición de tareas para ejecución paralela, logrando mejoras significativas en el rendimiento sin apoyar el paralelismo distribuido [7]. Además, diferentes enfoques en este campo han conducido al desarrollo de programas paralelos altamente optimizados para aplicaciones concretas [16].

La irrupción del aprendizaje automático y profundo ha transformado el campo de la paralelización automática, permitiendo igualar o incluso superar el rendimiento de códigos optimizados por expertos humanos. Estos enfoques han tenido éxito en detectar el paralelismo dentro de los códigos [24] y estimar el rendimiento de las aplicaciones, facilitando una distribución óptima de la granularidad de las tareas y los recursos [25]. Sin embargo, a pesar de estos avances significativos, la investigación centrada específicamente en el rendimiento de los videojuegos sigue siendo poco explorada.

2.2. Modificaciones del código fuente

La optimización del rendimiento de los programas de software es un campo de estudio que se centra, fundamentalmente, en la transformación del código para mejorar su eficiencia. Este proceso se divide en dos enfoques principales: las transformaciones de *front-end* y las de *back-end*. Las transformaciones de *front-end* se aplican directamente sobre el código fuente y están influenciadas por el lenguaje de programación utilizado. Estas transformaciones se pueden agrupar en dos categorías fundamentales:

- *Métodos que alteran las propiedades funcionales del código.* La programación genética (GP) ha sido utilizada para mejorar la eficiencia del software mediante la manipulación de árboles estructurados que reflejan el árbol de sintaxis abstracta de un programa. Aunque se ha centrado en mejorar atributos como el rendimiento en tiempo de ejecución [13], suele limitarse a códigos fuente pequeños debido a los altos requisitos de memoria y la amplitud del espacio de búsqueda. Asimismo, GP puede modificar la semántica del código, lo que requiere pruebas estrictas para garantizar cierto nivel de integridad semántica.
- *Métodos que mantienen las propiedades funcionales del código.* La refactorización de código preserva las propiedades funcionales del software y ha demostrado ser un método eficaz para disminuir el tiempo de ejecución [22]. Además, se están explorando enfoques innovadores basados en el aprendizaje profundo, enfocándose en la selección de hardware y algoritmos [9].

Por otro lado, las transformaciones de *back-end* operan independientemente del lenguaje de programación, actuando sobre una representación intermedia del código o el código de ensamblador sin alterar su semántica. La selección de la secuencia óptima de transformaciones es uno de sus principales desafíos, ya que puede variar significativamente según las características específicas del software y del hardware en el que se ejecuta [8]. La complejidad de este problema radica en

la dificultad para predecir el impacto combinado de diferentes transformaciones en el rendimiento [20].

Los nuevos avances en el campo han explorado el uso de técnicas de aprendizaje automático y aprendizaje profundo para automatizar y mejorar la selección de estas transformaciones [4]. Estos enfoques buscan no solo identificar las secuencias de transformaciones más efectivas sino también ajustar de manera óptima la configuración del compilador para maximizar el rendimiento y minimizar el consumo energético en aplicaciones específicas de la GPU [19]. Recientemente, se ha propuesto un enfoque innovador para optimizar el rendimiento del software, abordando el desafío mediante un problema de optimización combinatoria denominado SCOP, considerando la incertidumbre en las mediciones del tiempo de ejecución [3].

A pesar de estos avances en la optimización automática del software, el campo de los videojuegos sigue siendo poco explorado. Algunos autores proponen técnicas de programación para mejorar el rendimiento. Estas incluyen la reducción de polígonos en modelos para aliviar la carga gráfica [10] o la delegación de tareas en servidores remotos [11], entre otras.

3. Definición del problema

Esta sección presenta SCOP-FPS, un problema de optimización combinatoria para la maximización de la tasa de FPS de videojuegos, derivado a partir de SCOP [3]. Su objetivo es maximizar la tasa de FPS de un videojuego dado mediante la aplicación de transformaciones genéricas de código fuente. Dado el código fuente de un videojuego P y un conjunto de transformaciones genéricas de código fuente $T = t_1, \dots, t_k$, que transforman P en un código semánticamente equivalente P' , SCOP-FPS se define como encontrar una secuencia de tales transformaciones $\vec{S} = [s_1, \dots, s_N]$, $s_i \in T$, tal que la tasa de FPS del videojuego resultante P' sea máxima:

$$\text{Maximizar } F(P, \vec{S}) = FPS(P') , \quad (1)$$

donde P' se obtiene tras aplicar la secuencia de transformaciones \vec{S} a P :

$$P \xrightarrow{\vec{S}} P' \equiv P \xrightarrow{s_1} P_1 \xrightarrow{[s_2, \dots, s_N]} P' \quad (2)$$

$$\equiv P \xrightarrow{s_1} P_1 \xrightarrow{s_2} P_2 \dots \xrightarrow{s_{N-1}} P_{N-1} \xrightarrow{s_N} P' , \quad (3)$$

donde P_i es el videojuego resultante de aplicar secuencialmente las transformaciones s_1 a s_i sobre el videojuego original P . Cada s_i toma el valor del identificador de una transformación, y la secuencia \vec{S} define el orden en que deben aplicarse las distintas transformaciones. Este orden es importante porque influye directamente en el código de salida resultante. Además, se permite que la misma transformación se aplique más de una vez (puede ocurrir que $s_i = s_j$, $i \neq j$) ya que, para la mayoría de las transformaciones existentes, su aplicación repetitiva produce más cambios en el código fuente [20].

4. Metodología

El problema de optimización combinatoria propuesto utiliza un conjunto de transformaciones genéricas de código fuente para mejorar la tasa de FPS de un videojuego. Se emplean las transformaciones disponibles en la infraestructura LLVM, como se detalla en la Sección 4.1. La herramienta presentada en la Sección 4.2 se utiliza para generar la representación intermedia de grandes proyectos de código, como los videojuegos. Finalmente, en la Sección 4.3, se describe de manera general todo el proceso de optimización llevado a cabo.

4.1. Infraestructura de compilación LLVM

LLVM¹ es una conocida infraestructura de compilación que admite varios lenguajes de programación y conjuntos de instrucciones. Una de sus características destacadas es su representación intermedia, un lenguaje de programación pseudoensamblador que puede generarse a partir de todos los lenguajes soportados. Este código LLVM IR se genera utilizando el compilador Clang sobre el código fuente original. Además, LLVM también implementa una gran cantidad de operaciones para analizar y optimizar el código IR, llamadas *passes*. Todos los *passes* implementados en LLVM garantizan que la semántica del código permanezca inalterada. Por lo tanto, los *passes* a códigos LLVM IR se pueden utilizar para optimizar un programa de software independientemente de su lenguaje de programación o su arquitectura de hardware.

El comando **opt** (encargado de aplicar las transformaciones) es el principal optimizador y analizador en LLVM. Primero, el compilador Clang se utiliza para generar el código LLVM IR a partir del software original. Luego **opt** aplica una secuencia de *passes* a este código IR. Por ejemplo, **opt** puede aplicar una secuencia de cuatro *passes* como `-gvn`, `-loop-unroll`, `-basic-aa` y `-verify`, modificando el código LLVM IR original. Finalmente, se vuelve a utilizar el compilador Clang para generar el ejecutable del código LLVM IR transformado para la arquitectura seleccionada. Esto debe hacerse asegurándose de que el compilador no introduzca más modificaciones en el código.

4.2. Whole Program LLVM

Whole Program LLVM (WLLVM) [18] es una herramienta que permite generar los archivos LLVM IR de todo un proyecto de C/C++ de forma sencilla. Concretamente, es un envoltorio o *wrapper* de compiladores basado en Python 3 que funciona en dos etapas:

¹ Desde diciembre de 2011, “LLVM” no es un acrónimo sino una marca que se aplica al proyecto general de LLVM. Más información en <https://www.llvm.org>.

1. *Compilación del código*: Invoca al compilador seleccionado de la forma habitual. Entonces, por cada uno de los ficheros compilados, genera su LLVM IR, agregándolo a una sección específica dentro del código objeto (así no se pierde el orden de las instrucciones).
2. *Generación de un único fichero LLVM IR*: Cuando termina la compilación del código, se puede emplear WLLVM para leer los contenidos del código objeto, generando un único fichero LLVM IR con todo el código del proyecto.

WLLVM permite utilizar múltiples compiladores de C/C++ así como configurar la versión específica que se va a emplear. Asimismo, se puede indicar la configuración del compilador mediante una variable de entorno, controlando en todo momento las acciones que realiza.

4.3. Descripción del proceso de optimización

Tal y como se describe en la Figura 1, el proceso de optimización de videojuegos propuesto consta de tres componentes principales:

- *Infraestructura de compilación*: proporciona un conjunto de transformaciones genéricas de código, así como la capacidad de compilar el código fuente del videojuego y generar su archivo ejecutable.
- *Algoritmo de optimización combinatoria*: busca secuencias de *passes* para optimizar el rendimiento del videojuego. Cada solución generada representa una secuencia de un número fijo de *passes*, cuya calidad se evalúa mediante la medición del rendimiento de la versión del videojuego modificada. Las mejores soluciones son seleccionadas por el algoritmo, guiando así la búsqueda hacia mejoras en el rendimiento del juego.
- *Plataforma hardware*: se utiliza para evaluar el rendimiento de las versiones del videojuego generadas. Además, en esta plataforma también se ejecuta el algoritmo de optimización, que debe pararse mientras se evalúa el rendimiento del videojuego, para evitar interferencias con las mediciones.

En este trabajo se propone el uso de un algoritmo genético celular (cGA) [2] para encontrar soluciones precisas al problema, destacando su capacidad para mantener la diversidad en la población durante más tiempo con respecto a GAs de población panmictica, lo cual es crucial para problemas complejos, como SCOP-FPS, porque retrasa la convergencia prematura. Los cGAs se caracterizan por una población estructurada en una malla toroidal, introduciendo el concepto de distancia entre individuos. Además, se emplea un enfoque basado en el peor caso para evaluar la calidad de las soluciones propuestas [3], donde se toma la peor de un total cinco mediciones del rendimiento del videojuego.

5. Configuración de los experimentos

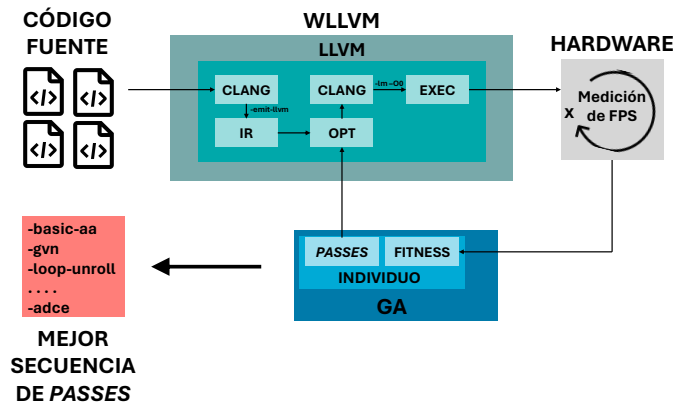


Figura 1. Interacción entre los diferentes componentes de la metodología propuesta.

En este trabajo, utilizamos la versión 9.0.1 de LLVM, que proporciona un conjunto de 87 *passes* genéricos de optimización de código. Además, incluimos un pase adicional a los ofrecidos por LLVM que denominamos **NONE**, y que no realiza ningún cambio en el código IR. **NONE** se considera para permitir al algoritmo encontrar secuencias de *passes* más cortas que la longitud de cromosoma.

Para reducir la componente estocástica del cGA, se decidió ejecutarlo 5 veces. Su configuración experimental se muestra en la Tabla 1. La población inicial consta de 100 individuos con una disposición cuadrada y una vecindad lineal 5 (L5). Cada individuo tiene un cromosoma representado por un vector de 300 enteros, elegida para ser similar al número de transformaciones utilizadas por la bandera de optimización `-O3` de Clang [3]. Los operadores genéticos incluyen el torneo binario para la selección de padres, el cruce de dos puntos para la recombinación, la mutación aleatoria y el reemplazo del individuo si es mejor. La recombinación se aplica siempre y la mutación con una probabilidad de 1/300 para cambiar un gen del cromosoma por otro valor permitido al azar.

Como videojuego a optimizar, hemos decidido emplear una modificación o *port* del Doom Engine original conocida como Doom Legacy [1] debido a la importancia de Doom en la industria. Es una modificación del código fuente original del Doom Engine, escrita en el lenguaje de programación C, y está disponible para Windows, Linux y Mac OS X. Esta versión incluye nuevas características, como un nuevo motor gráfico basado en OpenGL, soporte para multijugador y compatibilidad con mandos, entre otras.

Nuestra metodología se aplica específicamente a una partida pregrabada del primer nivel de Doom, con una duración de 365 segundos. El motor de juego tiene la tasa de FPS fijada en 35, ya que su rendimiento está vinculado a esta métrica. En otras palabras, cuanto mayor sea la tasa de FPS, más rápida será la velocidad del videojuego. Por lo tanto, eliminamos la limitación de FPS para reducir el tiempo de ejecución de la partida y, en consecuencia, aumentar el

Tabla 1. Configuración del algoritmo de optimización.

Algoritmo Genético Celular	
Longitud de los individuos	300
Tamaño de la población	100
Composición de la población	10×10
Inicialización de la población	Aleatoria
Vecindad	L5
Selección de padres	Individuo actual + Torneo Binario
Cruce	Cruce por dos puntos
Probabilidad de cruce	$\rho_c = 1,0$
Mutación	Valor aleatorio
Probabilidad de mutación	$\rho_0 = 1/300$
Reemplazo	Reemplazar si es mejor

número de FPS en Doom Legacy. Fijar la tasa de FPS de la versión optimizada a 35 implica una mayor cantidad de recursos libres disponibles, con respecto a la versión original del algoritmo, que pueden llevar tanto a ahorros en el consumo como a mejoras en el juego.

Finalmente, como arquitectura hardware se ha empleado una consola portátil comercial, la Steam Deck [23]. Está equipada con una APU AMD de 7 nm, que incluye una CPU Zen 2 de 2.4 a 3.5 GHz y una GPU con 8 RDNA 2 CUs a 1.6 GHz. Cuenta con 16 GB de RAM y ejecuta el sistema operativo Steam OS Holo.

6. Discusión y resultados

Para evaluar la calidad de las soluciones obtenidas, se realizaron un total de 1,000 ejecuciones independientes de la partida del Doom Legacy optimizado por la mejor secuencia de *passes* obtenida entre todas las ejecuciones del cGA. Este proceso se repitió para todas las secuencias de transformaciones genéricas que ofrece LLVM 9.0.1 (-00, -01, -02, -03, -0s y -0z). Utilizando la tasa de FPS medida, se generó un diagrama de caja y bigotes para cada versión del juego generada, comparando las 1,000 mediciones realizadas entre los indicadores de optimización genéricos y la mejor solución encontrada por nuestro algoritmo. Los resultados de este análisis se muestran en la Figura 2.

La solución encontrada por el cGA mejora en tasa de FPS a todas las secuencias de transformaciones genéricas ofrecidas por LLVM 9.0.1. Asimismo, se puede observar que no hay grandes diferencias entre el rendimiento en FPS del Doom Legacy sin optimizar (-00) y el resto de secuencias de transformaciones genéricas (hasta el 2,93%). No obstante, la mejor solución encontrada por nuestro algoritmo consigue una mejora del 6,21% respecto al Doom Legacy sin optimizar (3,19% respecto a la mejor secuencia genérica).

Estos resultados muestran la eficacia de nuestra metodología de optimización, la cual logra reducir el cómputo realizado por Doom Legacy y aumentar la tasa de FPS en aproximadamente 500 cuadros. Gracias a esta mejora, los desarrolladores

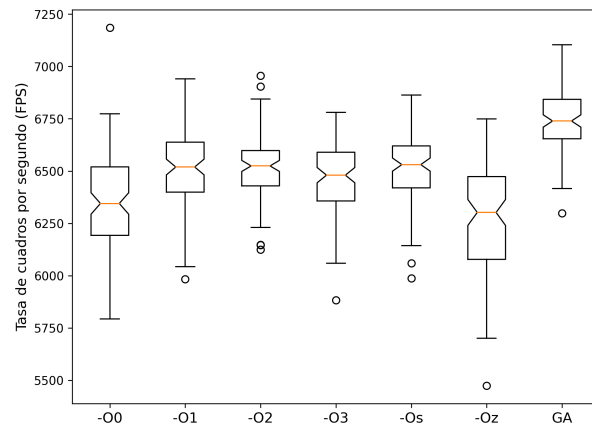


Figura 2. Comparación del mejor resultado optimizado con el Doom Legacy compilado con los indicadores -00 (es decir, sin optimizaciones), -01, -02, -03, -0s y -0z.

podrían implementar algoritmos más complejos en los motores vinculados a la tasa de FPS. Además, en aquellos motores no vinculados a la tasa de FPS, su incremento mejoraría la percepción visual de los usuarios y su satisfacción con el producto.

En la Figura 3 se muestra la evolución de la mejor solución en la población durante las 100 generaciones. Se observa que la población deja de evolucionar significativamente alrededor de la generación 40, lo que podría indicar una convergencia prematura del algoritmo. Esto sugiere que el algoritmo tiene margen de mejora, sirviendo de base para futuros trabajos.

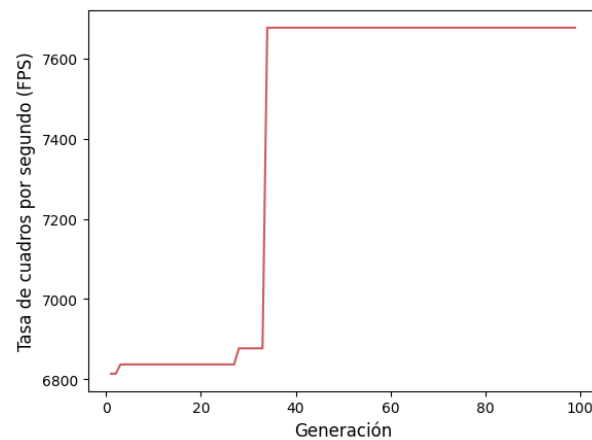


Figura 3. Evolución de la mejor solución encontrada a lo largo de las 100 generaciones.

7. Conclusiones y trabajo futuro

En este trabajo, proponemos una nueva metodología de optimización del rendimiento de videojuegos diseñada específicamente para una arquitectura de hardware determinada. Para lograrlo, hemos definido matemáticamente un nuevo problema de optimización combinatoria, llamado SCOP-FPS, con el objetivo de maximizar la tasa de FPS de un videojuego. Además, hemos llevado a cabo su resolución mediante el uso de algoritmos evolutivos, aplicándolos al caso específico del motor Doom Legacy en la consola Steam Deck.

Los resultados muestran que la mejor solución encontrada por el algoritmo genético celular mejora en un 6,21% respecto al Doom Legacy no optimizado y en un 3,19% respecto a la secuencia de transformaciones genéricas que reportaba mejor rendimiento. Estos resultados destacan la eficacia de nuestra metodología de optimización, que permite la mejora automática de videojuegos para arquitecturas específicas, incrementando la tasa de FPS.

Como posibles líneas de trabajo futuro, sería interesante ampliar el estudio mediante la exploración de nuevos algoritmos de optimización, otros videojuegos y diferentes arquitecturas. Específicamente, sería relevante investigar el impacto de nuestra metodología en el rendimiento de arquitecturas con recursos limitados así como en otras consolas portátiles comerciales. Además, estamos enfocados en expandir el estudio hacia la optimización automática del consumo energético en videojuegos, con el objetivo de prolongar la duración y la vida útil de las baterías en consolas portátiles como Steam Deck.

Agradecimientos. Jose M. Aragón-Jurado agradece su ayuda al Ministerio de Universidades a través del programa FPU (FPU21/02026). Esta publicación cuenta con el apoyo de los proyectos eFracWare (TED2021-131880B-I00) financiado por MCIN/AEI/10.13039/501100011033 y la Unión Europea NextGenerationEU/PRTR, y el proyecto eMob (PID2022-137858OB-I00) financiado por MCIN/AEI/10.13039/501100011033 y FEDER Una manera de hacer Europa.

Referencias

1. Doom Legacy. <https://doomlegacy.sourceforge.net/>, accedido en marzo 2024
2. Alba, E., Dorronsoro, B.: Cellular Genetic Algorithms, Operations Research/Computer Science Interfaces, vol. 42. Springer-Verlag Heidelberg (2008)
3. Aragón-Jurado, J.M., de la Torre, J.C., Ruiz, P., Galindo, P.L., Zomaya, A.Y., Dorronsoro, B.: Automatic software tailoring for optimal performance. *IEEE Transactions on Sustainable Computing* pp. In press. 1–18 (2024)
4. Ashouri, A.H., Killian, W., Cavazos, J., Palermo, G., Silvano, C.: A survey on compiler autotuning using machine learning. *ACM Computing Surveys* **51**(5) (2018)
5. Banerjee, U., Eigenmann, R., Nicolau, A., Padua, D.A.: Automatic program parallelization. *Proceedings of the IEEE* **81**(2), 211–243 (1993)
6. Claypool, M., Claypool, K., Damaa, F.: The effects of frame rate and resolution on users playing first person shooter games. In: *Multimedia computing and networking 2006*. vol. 6071, p. 607101. SPIE (2006)

7. Fonseca, A., Cabral, B., Rafael, J., Correia, I.: Automatic parallelization: Executing sequential programs on a task-based parallel runtime. *International Journal of Parallel Programming* **44**(6), 1337–1358 (2016)
8. Georgiou, K., Blackmore, C., Xavier-de Souza, S., Eder, K.: Less is more: Exploiting the standard compiler optimization levels for better performance and energy consumption. In: *Proceedings of the 21st International Workshop on Software and Compilers for Embedded Systems (SCOPEs '18)*. pp. 35—42. Association for Computing Machinery, New York, NY, USA (2018)
9. Hakimi, Y., Baghdadi, R., Challal, Y.: A hybrid machine learning model for code optimization. *International Journal of Parallel Programming* **51**, 309–331 (2023)
10. Koulaxidis, G., Xinogalos, S.: Improving mobile game performance with basic optimization techniques in unity. *Modelling* **3**(2), 201–223 (2022)
11. Kwon, D., Yang, S., Paek, Y., Ko, K.: Optimization techniques to enable execution offloading for 3d video games. *Multimedia Tools and Apps*. **76**, 11347–11360 (2017)
12. Lamport, L.: The parallel execution of do loops. *Communications of the ACM* **17**(2), 83—93 (1974)
13. Langdon, W.B., Harman, M.: Optimising existing software with genetic programming. *IEEE Transactions on Evolutionary Computation* **19**(1), 118–135 (2015)
14. LLVM: The LLVM Compiler Infrastructure. <https://llvm.org/>, accedido 03/2024
15. Midkiff, S.P.: *Automatic Parallelization—An overview of fundamental compiler techniques*. Morgan & Claypool Publishers (2012)
16. Müller, S.C., Alonso, G., Csillaghy, A.: Scaling astrophysics: Python + automatic parallelization. *IEEE Computer* **47**, 41–47 (2014)
17. NewZoo: Games Market Reports and Forecasts. <https://newzoo.com/games-market-reports-forecasts>, Último acceso 6 de marzo 2024
18. Ravitch, T.: Whole Program LLVM. <https://github.com/travitch/whole-program-llvm>, accedido en marzo 2024
19. Sachan, A., Srivastav, P., Ghoshal, B.: Learning based application driven energy aware compilation for GPU. *Microprocessors and Microsystems* **94**, 104664 (2022)
20. de la Torre, J.C., Ruiz, P., Dorronsoro, B., Galindo, P.L.: Analyzing the influence of llvm code optimization passes on software performance. In: *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. pp. 272–283. Springer (2018)
21. de la Torre, J.C., Aragón-Jurado, J.M., Crespo-Álvarez, A., Bárcena-González, G.: Gagi: Game engine for artificial general intelligence experimentation. *SoftwareX* **26**, 101665 (2024)
22. Traini, L., Di Pompeo, D., Tucci, M., Lin, B., Scalabrino, S., Bavota, G., Lanza, M., Oliveto, R., Cortellessa, V.: How software refactoring impacts execution time. *ACM Transactions on Software Engineering and Methodology* **31**(2) (2021)
23. Valve: Steam Deck™. <https://store.steampowered.com/steamdeck/>, accedido marzo 2024
24. Wang, Z., Tournavitis, G., Franke, B., O'boyle, M.F.P.: Integrating profile-driven parallelism detection and machine-learning-based mapping. *ACM Transactions on Architecture and Code Optimization* **11**(1) (feb 2014)
25. Zhang, P., Fang, J., Yang, C., Huang, C., Tang, T., Wang, Z.: Optimizing streaming parallelism on heterogeneous many-core architectures. *IEEE Transactions on Parallel and Distributed Systems* **31**(8), 1878–1896 (2020)

Validación de f-divergencias para la selección del mejor punto de vista en videojuegos*

Micaela Y. Martin¹[0000-0001-6100-7538], Mateu Sbert^{2,3}[0000-0003-2164-6858],
and Miguel Chover³[0000-0002-0525-7038]

¹ Universitat Jaume I, Spain micmarti@uji.es

² Universitat de Girona, Spain

Resumen Este estudio investiga la evaluación de la calidad visual de las vistas en entornos virtuales en 3D utilizando medidas de teoría de la información y las compara con las preferencias del usuario. Se analizaron varios escenarios, que incluyen composiciones de objetos simples y complejos, así como escenas de un videojuego. Se emplearon medidas como la divergencia de Kullback-Leibler (K-L), la distancia chi-cuadrado (χ^2) y la Variación Total (TV) para evaluar la calidad visual de cada vista. Los resultados indicaron una alta concordancia entre la divergencia de K-L y las preferencias del usuario en la mayoría de los casos estudiados, donde las medidas identificaron consistentemente las vistas preferidas por los usuarios. Sin embargo, se observaron discrepancias en ciertos escenarios. Estas discrepancias resaltan la importancia de considerar múltiples factores al seleccionar la vista óptima en entornos virtuales. Se sugiere que futuras investigaciones exploren la incorporación de condiciones adicionales, como la visibilidad completa de los objetos en la escena.

Keywords: punto de vista · videojuego · f-divergencias

1. Introducción

En el ámbito de las escenas virtuales en 3D dentro de videojuegos, la selección de la posición óptima de la cámara ha recibido una atención limitada hasta la fecha. Dada la importancia de la percepción visual en el desarrollo de la trama de un juego, resulta fundamental desarrollar métodos que permitan seleccionar vistas que destaquen la información más relevante de la escena para el jugador.

Aunque las medidas de teoría de la información, especialmente la entropía de Shannon, han sido ampliamente utilizadas en la selección de puntos de vista en ámbitos como la robótica, los gráficos por computadora y la visualización [1,2], presentan un inconveniente significativo debido a su elevado costo asociado derivado del cálculo con proyecciones.

En la etapa inicial de esta investigación, propusimos la utilización de f-divergencias, explorando la divergencia de K-L, TV y χ^2 , con el fin de calcular la posición óptima de la cámara en un entorno Unity 3D³.

* Financiado por la Generalitat Valenciana

³ <https://unity.com/es>

Para cuantificar la visibilidad de un objeto introducimos el factor de forma del frustum de la cámara, expandiendo el trabajo presentado en [3,4]. Los factores de forma de todos los objetos junto con el factor de forma del background, constituyen una distribución de probabilidad que fue comparada utilizando cada f-divergencia mencionada con una distribución objetivo. Esta última puede corresponder a la distribución de áreas relativas o estar ponderada con valores de importancia.

Elegimos el motor de juego Unity 3D como herramienta de desarrollo debido a su amplia popularidad en la industria y a su capacidad para realizar funciones de ray casting que posibilitan el cálculo eficiente de los factores de forma.

Posteriormente, aplicamos las divergencias a diversas escenas y con distintos puntos de vista de las cámaras, analizando detenidamente los resultados obtenidos. Como resultado de la primera parte del estudio, obtuvimos que asignarle al background un porcentaje del 50 % del área total brindaba resultados consistentes en todos los ejemplos estudiados.

En este artículo, nuestro objetivo es comparar los resultados previamente obtenidos con la percepción de la mejor vista para el usuario. Con este propósito, un grupo de 58 usuarios participaron en un estudio donde evaluaron y seleccionaron la vista que percibían como la mejor en cada escena. Estos resultados fueron luego comparados con los obtenidos mediante las divergencias.

Esta investigación aporta una perspectiva novedosa al campo de la selección de vistas en entornos virtuales 3D, ofreciendo un análisis comparativo entre enfoques de teoría de la información y evaluaciones subjetivas de usuarios.

2. Estado del arte

La selección del mejor punto de vista en modelos 3D ha sido objeto de una amplia investigación en la literatura científica.

Bonaventura et al. [5] han propuesto una clasificación de atributos para evaluar la calidad de un punto de vista en modelos poligonales, abarcando áreas como área, silueta, profundidad, estabilidad y curvatura superficial. Este enfoque incluye medidas como el número de triángulos visibles [6], área proyectada, relación de visibilidad, entropía del punto de vista y medida de K-L [7]. Los atributos de silueta se centran en la forma y estructura del objeto visible desde el punto de vista, incluyendo medidas como la longitud de la silueta [8], entropía de la silueta y curvatura de la silueta [9]. Los atributos de profundidad se enfocan en la información de profundidad, con medidas como la profundidad máxima. Los atributos de curvatura superficial analizan la curvatura de la superficie del objeto visible. Además, la estabilidad del punto de vista es un aspecto crucial que considerar al seleccionar el mejor punto de vista, con medidas como la inestabilidad [10] y la estabilidad visual basada en la profundidad [11].

En el ámbito de los videojuegos, la selección del mejor punto de vista para escenas virtuales ha sido menos explorada en comparación con la robótica. Sin embargo, algunos estudios han abordado este tema, como el trabajo realizado por Galvane [12], que propone un sistema basado en el modelo de comportamiento

de dirección de Reynolds para controlar y coordinar agentes de cámara autónomos en entornos dinámicos en 3D. Este enfoque utiliza información narrativa y geométrica para calcular trayectorias y planificar la cámara en tiempo interactivo, seleccionando los mejores puntos de vista según una función que considera múltiples factores. Otro enfoque propuesto por Lino y Christie [13] emplea un modelo teórico de superficie para generar puntos de vista eficientemente.

En un trabajo anterior realizado por Rigau et al. [14], se estudió la visibilidad de un punto en una escena 3D basada en criterios de teoría de la información. Este trabajo propuso utilizar la divergencia de K-L entre los ángulos sólidos proyectados por los objetos y las proyecciones no ocultas como medida del punto de vista. Además, Sbert et al. [7] propusieron como medida del punto de vista de un objeto la divergencia K-L entre las áreas proyectadas de los triángulos de la malla del objeto y las áreas reales. En la fase inicial, extendimos estas ideas para considerar la visibilidad medida por los factores de forma desde el punto de vista restringidos al frustum de la cámara, y consideramos las divergencias de K-L, TV y χ^2 como medidas de punto de vista. Además, exploramos la importancia de asignar diferentes importancias a los objetos. En esta etapa, comparamos los resultados obtenidos con las mejores vistas determinadas por los usuarios.

Esta investigación se sitúa en el cruce de la teoría de la información y la selección de puntos de vista en entornos virtuales 3D, proporcionando un enfoque comparativo entre medidas fundamentadas en teoría de la información y la percepción de los usuarios.

2.1. Algoritmo implementado

El algoritmo desarrollado realiza los siguientes pasos:

- Se calcula el área de la malla cada objeto en la escena utilizando sus triángulos.
- Se generan rayos aleatorios desde la posición de la cámara en direcciones aleatorias dentro del frustum de la cámara.
- Se utilizan estos rayos para determinar qué objetos en la escena son visibles desde la posición de la cámara.
- Se detectan las colisiones de los rayos con los objetos en la escena.
- Se cuenta el número de rayos que colisionan con cada objeto.
- Se calcula el factor de forma de cada objeto, dividiendo la cantidad de rayos que impactan en él sobre el total de rayos lanzados en la escena.
- Se calculan las divergencias de K-L, TV y χ^2 , y la entropía de Shanno, entre la distribución de los factores de forma y las áreas relativas.

La selección de las mejores vistas en el contexto de las f-divergencias, tales como K-L, TV y χ^2 , se basará en la imagen que presente el menor valor de divergencia. Por el contrario, en el caso de la Entropía de Shannon, se priorizará la imagen que exhiba el mayor valor de entropía.

2.2. Marco teórico

Dada una escena $S \in R^3$, sea $x \in S$ un punto en ella. Sea O el conjunto de todos los objetos o_i en la escena. Definimos A_i como el área del objeto o_i , A_T como el área total de los objetos en la escena más el área del background, y $a_i = \frac{A_i}{A_T}$ como el área relativa del objeto o_i . Dada la posición x de la cámara, consideramos dA_x en el plano normal a la dirección de la cámara. Sea $O_x \subset O$ el conjunto de objetos en el frustum de x . Dado un punto y en la superficie de un objeto en O_x , dA_y está en el plano tangente en y , θ_y es el ángulo entre la normal a dA_y con la línea que une x e y , y θ_x es el ángulo con la normal en dA_x . La distancia entre x e y es $d(x, y)$, y $v(x, y)$ es una función de visibilidad binaria (igual a 1 si x e y son mutuamente visibles y 0 en caso contrario), ver Fig. 1.

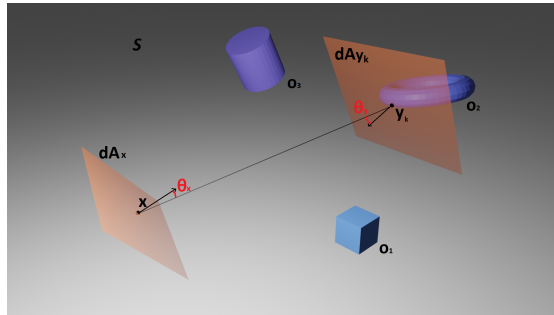


Figura 1. Notación usada para el factor de forma. θ_y es el ángulo entre la normal a dA_y con la línea que une x e y , y θ_x es el ángulo con la normal en dA_x .

Consideremos ahora la fracción del área total correspondiente a cada objeto, digamos a_i , y sea a_b el área relativa del background. Entonces, $\sum_{i=1}^n a_i + a_b = 1$.

Divergencia de Kullback-Leibler Consideremos la divergencia de K-L entre las dos distribuciones, utilizando el conteo de impactos, donde N_i es el número de impactos en el objeto i , y N_b en el background, el factor de forma del frustum se aproxima por $F(dA_x, o_i) \approx \frac{N_i}{N_{fr}}$, y por lo tanto la divergencia de K-L se aproxima por

$$K-L(dA_x, o_i) \approx \sum_{i=1}^n \frac{N_i}{N_{fr}} \log \frac{\frac{N_i}{N_{fr}}}{a_i} + \frac{N_b}{N_{fr}} \log \frac{\frac{N_b}{N_{fr}}}{a_b}. \quad (1)$$

Observe que $a_i = A_i/(A_T + A_b)$, y $a_b = A_b/(A_T + A_b)$, y $A_T = \sum_i A_i$. Ahora, tomar la divergencia de K-L como medida de punto de vista tiene un problema: no penaliza los objetos no vistos, al contrario, como el término correspondiente en la suma es 0. Consideramos entonces la entropía de Shannon comunmente usada, y dos alternativas adicionales.

Entropía de Shannon Utilizando la entropía de Shannon como medida del punto de vista del frustum, obtenemos

$$H(dA_x, o_i) \approx \sum_{i=1}^n \frac{N_i}{N_{fr}} \log \frac{N_i}{N_{fr}} + \frac{N_b}{N_{fr}} \log \frac{N_b}{N_{fr}}. \quad (2)$$

Medidas de punto de vista del frustum de TV y divergencia χ^2 Utilizando la TV como medida de punto de vista del frustum, obtenemos

$$TV(dA_x, o_i) \approx 1/2 \left(\sum_{i=1}^n \left| \frac{N_i}{N_{fr}} - a_i \right| + \left| \frac{N_b}{N_{fr}} - a_b \right| \right). \quad (3)$$

Y usando la divergencia χ^2 ,

$$\chi^2(dA_x, o_i) \approx \sum_{i=1}^n \frac{\left(\frac{N_i}{N_{fr}} - a_i \right)^2}{a_i} + \frac{\left(\frac{N_b}{N_{fr}} - a_b \right)^2}{a_b}. \quad (4)$$

Observe que en ambas medidas un objeto o_i no visto agrega la misma cantidad a_i . Esta es una diferencia principal con respecto a la medida K-L, donde la traza de un objeto no visto desaparece.

2.3. Importancia del background e importancias a objetos

Supongamos que estamos en una escena abierta, sin ninguna superficie del background a considerar. Por un lado, consideramos que los rayos que faltan en los objetos $\{o_i\}$ llegan al background y los contamos como N_b . Ahora, en lugar de considerar una superficie del background ficticia como un hemisferio que envuelve los objetos, podemos decidir a priori cuánto background queremos ver en nuestro frustum y simplemente establecer a_b como una proporción de $\sum_{i=1}^n a_i$. Un valor pequeño de $\frac{a_b}{\sum_i a_i}$ significa que nuestra medida de punto de vista favorecerá una pequeña proporción del background en el frustum, mientras que una grande significará lo contrario. Y esto se puede extender a cualquier objeto en $\{o_i\}$. Se puede formalizar, de manera similar a [10], definiendo valores de importancia distintos de cero $\{p_i\}$ para los objetos y el background p_b y considerando la nueva distribución de pseudoáreas $a'_i = \frac{a_i p_i}{\sum_k a_k p_k + a_b p_b}$, $a'_b = \frac{a_b p_b}{\sum_k a_k p_k + a_b p_b}$.

3. Resultados

Para comparar los resultados obtenidos por las medidas con la percepción humana de las mejores vistas, se compararon siete escenas distintas. En cada una de ellas, se calcularon las metricas estudiadas con 100000 rayos y un área del background al 50 % y se determinó cual sería el mejor punto de vista para cada una. En segunda instancia, 58 usuarios completaron un formulario que solicitaba determinar que vista resultaba mejor para ellos en cada escena.

La primera escena considerada es un cubo, rotado para obtener 3 vistas (ver Fig. 2), y considerando el área del background al 50 % del área total.

Observamos que con un único objeto, los valores de las tres medidas disminuyen a medida que aumenta el número de caras visibles del cubo. Por lo tanto, la mejor vista para las tres medidas es la imagen der. de la Fig. 2, y coincide con la vista preferida por la mayoría de los usuarios (64%).

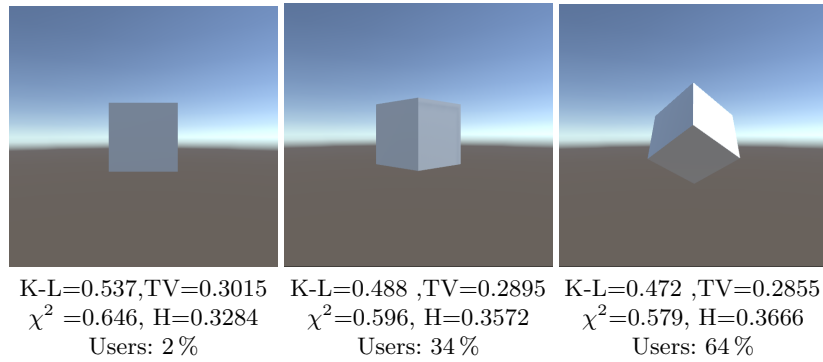


Figura 2. Comparación de medidas (TV, K-L, y χ^2) rotando un cubo, con el porcentaje de usuarios que eligieron cada vista.

Luego, analizamos una escena compuesta por un cubo, un cilindro y una esfera colocados en el centro de la escena y acercando y alejando la cámara preservando la dirección de la misma (Fig. 3) para un área del background del 50 %. Las medidas K-L y χ^2 logran un mínimo en la Fig. 3 sup.central, considerandola la mejor vista, sin embargo esta vista sólo fue elegida por el 4 % de los participantes. La medida de TV también disminuye al acercarse, pero su mínimo se alcanza en la Fig. 3 sup.izq. Por el contrario, los usuarios seleccionaron como la mejor vista (76 %) a la de la línea sup.der., seguida de la de la línea inf. izq. (17 %). Esto puede deberse a que si bien la esfera y el cilindro se ven claramente en la Fig. sup.der., el cubo se ve incompleto. Por el contrario, la vista seleccionada por los participantes es la primera que permite la visión completa de todos los objetos a la menor distancia posible. Sobre la medida de entropía del punto de vista, recuerde que su comportamiento es inverso al de la medida K-L, es decir, el valor más alto correspondería a una mejor vista. Disminuye al alejarse y aumenta al acercarse, lo cual es correcto. Sin embargo, en la Fig. 3 sup.izq., cuando nos acercamos al objeto cilindro, la entropía sigue creciendo, lo que no es el comportamiento esperado para una buena vista, ya que claramente es una vista peor que la de la Fig. 3 sup. der.

Como tercer ejemplo, como se ve en la Fig. 4, la cámara fue girada alrededor de un eje vertical obteniendo diferentes vistas de una escena compuesta por un cubo, un cilindro y una esfera. En este caso, el valor más bajo para las tres

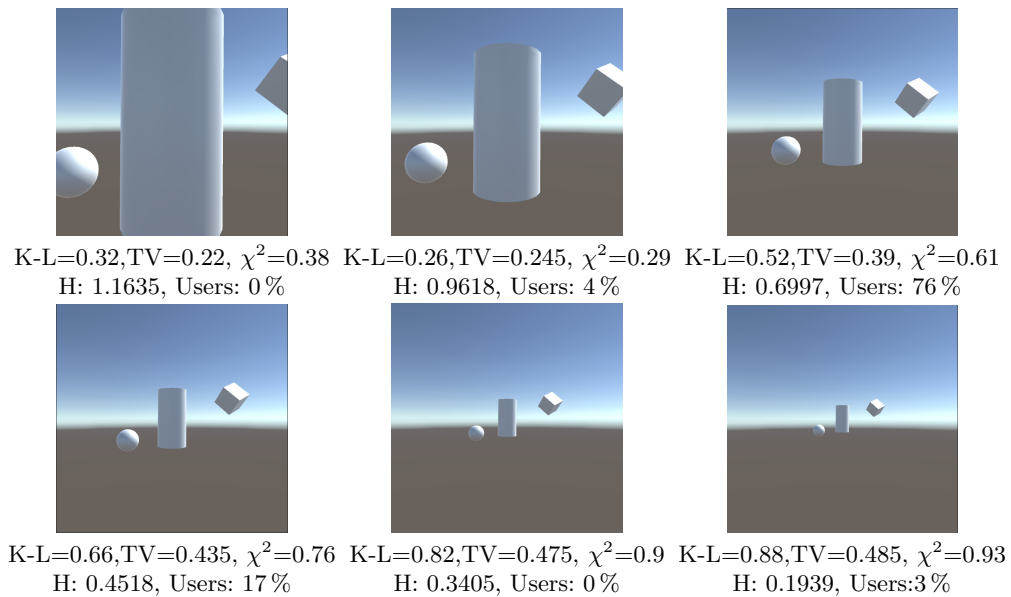


Figura 3. Comparación de medidas (TV, K-L, χ^2 y H) al alejar la cámara, con el porcentaje de usuarios que eligieron cada vista.

divergencias ocurrió cuando los tres objetos son visibles. Este punto de vista coincidió con la vista preferida por los usuarios (76 % de votos). En cuanto a la entropía, el valor más alto ocurre cuando todos los objetos son visibles mientras que el valor más bajo cuando ningún objeto es visible. Observe que el comportamiento de la medida de entropía es inverso al de la divergencia de K-L. Recuerde que el valor del background no juega ningún papel en el cálculo de la medida de entropía y esta se comporta de manera inversa a la divergencia de K-L cuando todos los objetos, incluido el background, tienen el mismo área relativa. En este ejemplo, la medida de entropía también proporciona resultados coherentes. Para probar las medidas en presencia de oclusión, empleamos una escena compuesta por un cilindro, una esfera y un cubo. La escena fue observada desde vistas laterales (Fig. 5 izq.), diagonales (Fig. 5 central) y frontales (Fig. 5 der.) mientras se mantenía una distancia constante al cilindro (Fig. 5). Los resultados muestran que K-L identifica como la mejor vista la vista diagonal, donde se ve una mayor área y hay una mínima oclusión entre los objetos. Esta imagen fue elegida por el 44 % de los participantes. Además, K-L designa la vista lateral como la menos favorable, donde hay una mayor oclusión entre los objetos. De manera similar, χ^2 y la entropía de Shannon funcionan bien, mientras que TV, aunque identifica la vista diagonal como la mejor, no distingue entre las laterales y frontales. Sin embargo, la percepción de la mejor vista para los usuarios correspondió con la imagen der. (56 % de votos), aunque la elección de la vista diagonal tuvo gran porcentaje de votos.

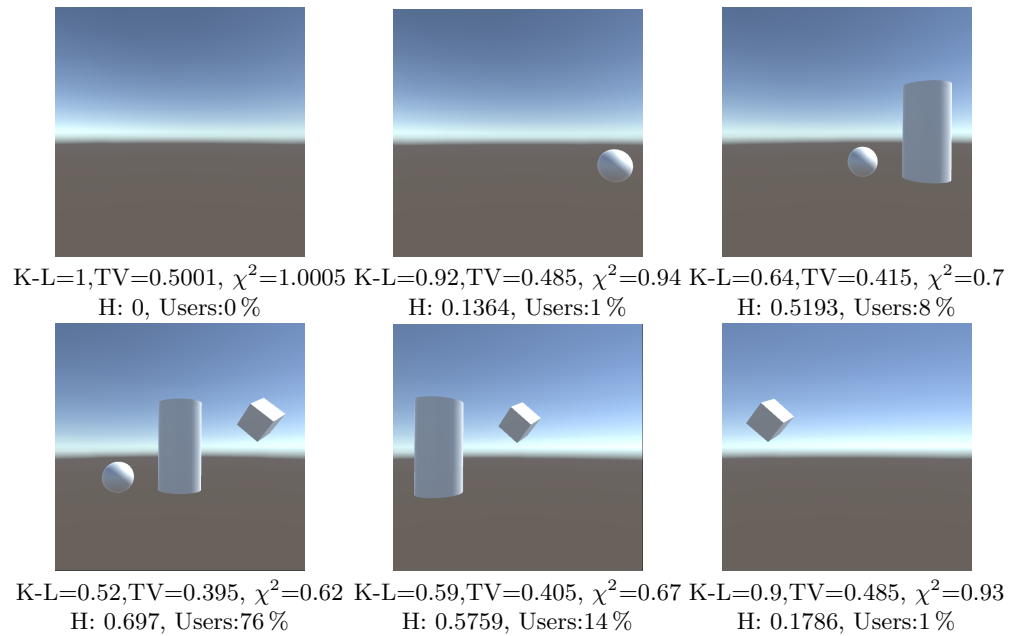


Figura 4. Comparación de medidas (TV, K-L, χ^2 y H) al girar la cámara alrededor de un eje vertical, con el porcentaje de usuarios que eligieron cada vista.

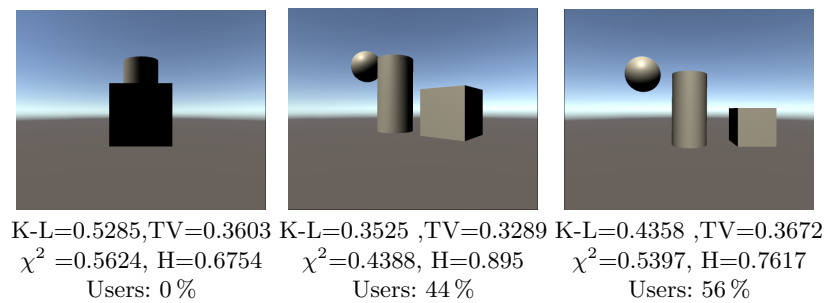
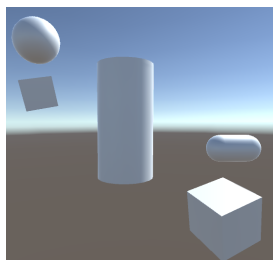
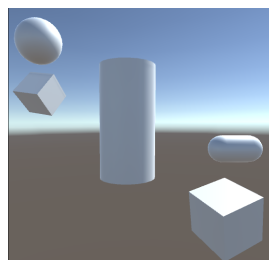


Figura 5. Comparación de medidas (TV, K-L, χ^2 y H) en una escena con oclusión, con el porcentaje de usuarios que eligieron cada vista.

Finalmente, en la Fig. 6 encontramos dos escenas con cinco objetos que solo difieren en la rotación de un cubo. Mientras que la divergencia de K-L aprecia una mejora en la segunda vista (Fig. 6 der.) coincidiendo con la elegida por el 92 % de los usuarios, la mejora de las divergencias de TV y χ^2 cae dentro del umbral de error que establecimos de 0,02.



K-L=0.2969,TV=0.2742, χ^2 =0.3413 Users: 8 %



K-L= 0.2744,TV=0.2681, χ^2 =0.323 Users: 92 %

Figura 6. Comparación de medidas (TV, K-L y χ^2) al rotar un solo objeto, con el porcentaje de usuarios que eligieron cada vista.

Para validar la implementación en un videojuego, probamos el método en el juego Unity de John Lemmon ⁴, en una escena compuesta por el personaje principal, 5 enemigos y el resto de la escena considerada como background (ver la Fig 7). El personaje principal es un gato amarillo, los enemigos son fantasmas grises y gárgolas también grises con una antorcha roja. Vemos dos pares de vistas traseras diferentes, Fig 7 izq. y Fig 7 der. Las imágenes inferiores de la Fig 7 son alejamientos de las sup. En la Fig 7 izq. no podemos apreciar al enemigo más cercano detrás del personaje principal, mientras que está claramente visible en la Fig 7 der. En este caso, las divergencias de K-L y χ^2 determinan como mejor vista la sup.der., que coincide con la vista más seleccionada por los participantes (39%), mientras que TV no logra identificar diferencias fuera del umbral del error entre las dos vistas en la Fig 7 sup.

Imaginemos ahora un escenario en un juego con una vista isométrica posicionada detrás del personaje principal, con una escena compuesta por ocho enemigos, 4 gárgolas y cuatro fantasmas. Nuestro objetivo es recomendar al jugador un punto de vista que detecte el mayor riesgo. En este caso, consideramos que las gárgolas son más ofensivas que los fantasmas. En consecuencia, asignamos una mayor importancia a las gárgolas. La Fig. 8 compara dos vistas de la cámara donde en la Fig. 8 izq. hay 3 fantasmas y una gárgola visibles, mientras que en la Fig. 8 der. se pueden ver 3 gárgolas y un fantasma. Los resultados de la fila sup. se dan sin importancia, mientras que en la segunda, tercera, cuarta y quinta fila se asignan valores de importancia 5, 10 y 20 respectivamente

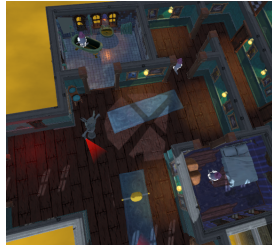
⁴ <https://learn.unity.com/project/john-lemon-s-haunted-jaunt-3d-beginner>



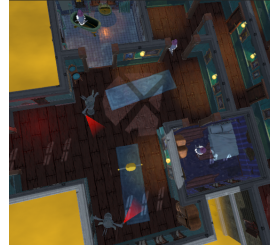
K-L=0.8157, TV=0.4526, $\chi^2=0.8396$
 H: 0.2579, Users: 3 %



K-L=0.7314, TV=0.448, $\chi^2=0.807$
 H: 0.32, Users: 39 %



K-L=0.9149, TV=0.488, $\chi^2=0.9532$
 H: 0.098, Users: 29 %



K-L=0.9066, TV=0.4877, $\chi^2=0.9514$
 H: 0.1059, Users: 29 %

Figura 7. Comparación de medidas en un videojuego (TV, K-L, χ^2 y H). El personaje principal es un gato con cabeza de color amarillo, los enemigos son fantasmas grises y gárgolas grises con una antorcha roja.

a las gárgolas y la importancia 1 a los fantasmas. Adicionalmente se indica el porcentaje de usuarios que eligieron cada vista y el valor de la entropía.

Aunque a primera vista, la Fig. 8 izq. podría parecer preferible porque los enemigos están más cerca, lo que se refleja en los valores más bajos de las tres medidas consideradas, esta vista contiene más enemigos tipo fantasma, que son menos agresivos para el jugador. Por lo tanto, nuestro interés radica en detectar que el área más peligrosa es la que se muestra en la vista de la Fig. 8 der.

Sin embargo, si asignamos una mayor importancia a las gárgolas (2, 5, 10 y 20 respectivamente) mientras que a los fantasmas se les da una importancia de 1, observamos que a medida que la importancia de las gárgolas aumenta sobre la de los fantasmas, las medidas de K-L y χ^2 identificarán gradualmente la vista de la Fig. 8 der. como superior(sup.). Estos resultados coinciden con la vista elegida por los participantes, que, al indicarseles que indiquen la vista preferida dado que las gárgolas son más ofensivas que los fantasmas, el 92 % seleccionó la Fig. 8 der. como mejor vista.

4. Conclusiones y trabajo futuro

En este estudio, se examinaron diversos escenarios virtuales 3D para evaluar la calidad visual de diferentes vistas utilizando medidas de teoría de la información y comparándolas con las preferencias de los usuarios. En primer lugar,



No imp: K-L=0.4201, TV=0.2567, χ^2 =0.4344	No imp: K-L=0.4461, TV=0.267, χ^2 =0.4491
imp 5: K-L=0.8483, TV=0.352, χ^2 =1.6111	imp 5: K-L=0.363, TV=0.2387, χ^2 =0.3605
imp 10: K-L=1.0918, TV=0.3823, χ^2 =3.117	imp 10: K-L=0.3879, TV=0.2412, χ^2 =0.4832
imp 20: K-L=1.3519, TV=0.3992, χ^2 =6.1346	imp 20: K-L=0.4299, TV=0.2426, χ^2 =0.7675
H:1.7438, Users: 13 %	H:1.7288, Users: 87 %

Figura 8. Comparación de medidas con importancias en un videojuego.

se consideró un escenario simple con un cubo rotado (Fig. 2) para obtener tres vistas distintas. Se observó que todas las medidas de evaluación, incluyendo la divergencia de K-L, la distancia de χ^2 y la divergencia TV, coincidieron al identificar la vista con mayor cantidad de caras visibles como la mejor opción. Además, esta vista preferida por las divergencias coincidió con la preferencia de la mayoría de los usuarios. También las medidas lograron identificar la mejor vista en la Fig. 4, seleccionando la vista en la que se ven los 3 objetos que componen la escena, y coincidiendo con la vista elegida por los usuarios. Análogamente, en una escena de juego con importancias según el tipo de enemigo, las tres medidas determinaron la mejor vista en concordancia con los usuarios.

En las Fig. 6 y 7, la divergencia de K-L se mostró más sensible a cambios menores, logrando identificar mejoras en vistas donde las otras divergencias no, y coincidiendo con la vista más seleccionada como mejor por los participantes.

Sin embargo, en dos ejemplos de los proporcionados (Fig. 3 y 5), la vista elegida por los usuarios no coincidió con las preferidas por las métricas. En el primer caso, probablemente a que aunque la vista elegida por las métricas permite ver los tres objetos que componen la escena, el cubo no logra verse en su completitud. En el segundo caso, los usuarios prefirieron una vista frontal de la escena a la vista diagonal selecta por las divergencias.

Aunque la divergencia de K-L funciona adecuadamente en la mayoría de los ejemplos estudiados, las discrepancias entre las medidas evaluadas y las preferencias de los usuarios resaltan la necesidad de la consideración de múltiples factores al seleccionar la vista óptima en entornos virtuales. En este sentido, se sugiere explorar en futuras investigaciones la incorporación de condiciones adicionales, como la visión completa de los objetos en la escena, para mejorar la precisión de las métricas objetivas. Además, sería beneficioso investigar la posibilidad de desarrollar métricas más complejas que capturen de manera más efectiva aspectos subjetivos de la experiencia visual, como la estética. Asimismo, se podría explo-

rar la aplicación de técnicas de aprendizaje automático para entrenar modelos que puedan predecir las preferencias de los usuarios en función de características específicas de la vista, lo que podría ayudar a mejorar la evaluación de vistas.

Referencias

1. Min Chen, Miquel Feixas, Ivan Viola, Anton Bardera, Han-Wei Shen, Mateu Sbert, Information Theory Tools for Visualization, CRC Press 2016, ISBN 9781315369228
2. Mateu Sbert, Miquel Feixas, Jaume Rigau, Miguel Chover, Ivan Viola, Information Theory Tools for Computer Graphics, Synthesis Lectures on Computer Graphics and Animation, Morgan & Claypool Publishers 2009, ISBN 978-3-031-79545-9
3. Sbert M., An Integral Geometry Based Method for Fast Form Factor Computation, Eurographics 1993, Computer Graphics Forum 12 (3), 409-420
4. Sbert M., The Use of Global Directions to Compute Radiosity: Global Monte Carlo Techniques, PhD thesis, Catalan Technical University, 1997
5. Bonaventura, X.; Feixas, M.; Sbert, M.; Chuang, L.; Wallraven, C. . A survey of viewpoint selection methods for polygonal models. *Entropy*, 20(5), 370.
6. Plemenos, D.; Benayada, M. Intelligent Display Techniques in Scene Modelling. New Techniques to Automatically Compute Good Views. In Proceedings of the International Conference GraphiCon'96, St Petersburg, Russia, 1 July 1996.
7. Sbert, M.; Plemenos, D.; Feixas, M.; Gonzalez, F. Viewpoint Quality: Measures and Applications. In Proceedings of the First Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging, Girona, Spain, May 2005; pp. 185-192.
8. Polonsky O., Patané G., Biasotti, S. et al. What's in an image?. *Visual Comput* 21, 840-847, 2005. <https://doi.org/10.1007/s00371-005-0326-y>
9. Secord A., Lu J., Finkelstein A., Singh M., Nealen A. 2011. Perceptual models of viewpoint preference. *ACM Trans. Graph.* 30, 5, Article 109 (October 2011), 12 pages. <https://doi.org/10.1145/2019627.2019628>
10. M Feixas, M Sbert, F Gonzalez, A unified information-theoretic framework for viewpoint selection and mesh saliency, *ACM Transactions on Applied Perception (TAP)* 6 (1), 1-23
11. Vázquez, P.P. Automatic view selection through depth-based view stability analysis. *Vis. Comput.* 2009, 25, 441-449.
12. Quentin Galvane. Automatic Cinematography and Editing in Virtual Environments. Artificial Intelligence [cs.AI]. Université de Grenoble Alpes, 2015. English.
13. Lino C., and Christie M. Efficient composition for virtual camera control. *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 2012.
14. Rigau, J., Feixas, M., & Sbert, M. (2000). Information theory point measures in a scene. IIA-00-08-RR, Institut d'Informàtica i Aplicacions, Universitat de Girona (Girona, Spain).

Participando en las modalidades individuales de la competición Geometry Friends

Alberto Almagro, Juan Carlos Llamas-Núñez, Antonio A. Sánchez-Ruiz^[0000-0003-0034-1254], and Belén Díaz-Agudo^[0000-0003-2818-027X]

Departamento de Ingeniería del Software e Inteligencia Artificial
Instituto de Tecnología del Conocimiento
Universidad Complutense de Madrid
{alberalm,jullamas,antsanch,belend}@ucm.es

Resumen Geometry Friends es un juego de plataformas en el que un círculo y un rectángulo deben recoger un conjunto de diamantes desperdigados por el nivel. Para lograrlo, los agentes deben resolver interesantes problemas relacionados con el análisis del nivel, la navegación entre plataformas, la simulación física de sus movimientos, la cooperación entre los agentes, etc. En este trabajo presentamos las ideas principales tras los agentes que inscribimos en las modalidades individuales de la competición celebrada en 2023. El agente rectángulo resultó ganador en su modalidad, mientras el agente círculo quedó segundo a pocos puntos del ganador.

Keywords: Geometry Friends · Agentes inteligentes · Simulación

1. Introducción

Geometry Friends¹ es un juego de plataformas cooperativo en el que dos agentes representados por figuras geométricas (un círculo y un rectángulo) deben colaborar para recoger un conjunto de diamantes desperdigados por el nivel en el menor tiempo posible (ver figura 1). Cada uno de los agentes puede realizar distintas acciones. El círculo puede rodar por las plataformas y saltar, mientras el rectángulo puede desplazarse por las plataformas y cambiar su forma (modificando su altura y anchura de forma que su superficie se mantenga constante). El movimiento de los personajes se realiza aplicando fuerzas físicas, por lo que es necesario tener en cuenta las velocidades, inercias y colisiones a la hora de desplazar a los agentes de forma precisa. Además, el color de cada agente determina las plataformas con las que puede interactuar: si bien ninguno puede cruzar las plataformas negras, el círculo puede atravesar las plataformas amarillas y el rectángulo las verdes. Finalmente, los agentes pueden interactuar entre ellos (apoyándose el uno en el otro y combinando sus movimientos) para recoger diamantes que de otro modo serían inalcanzables. Todo ello hace que el juego, en apariencia sencillo, plantee retos interesantes desde el punto de vista del desarrollo de agentes inteligentes.

¹ <https://geometryfriends.gaips.inesc-id.pt>

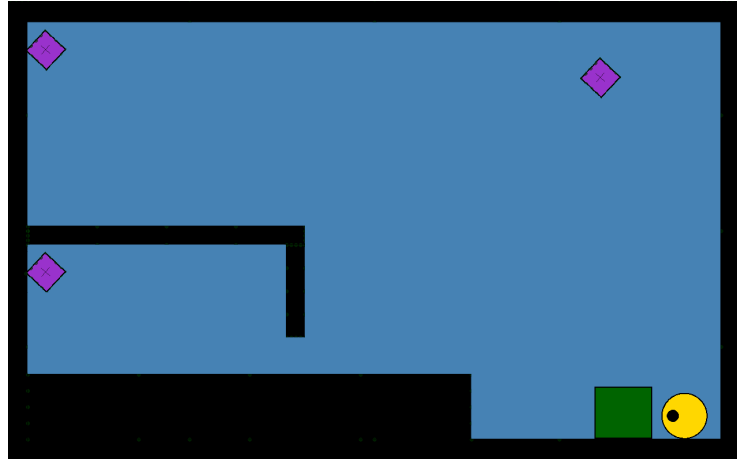


Figura 1. Ejemplo de nivel cooperativo del juego Geometry Friends.

La competición de Geometry Friends [10] se ha venido celebrando desde 2013, formando parte algunos años de las competiciones asociadas a importantes conferencias de IA como son International Joint Conferences on Artificial Intelligence (IJCAI) y IEEE International Conference on Games (CoG). Aunque el principal reto del juego está relacionado con la cooperación de los agentes (para recoger los diamantes en el menor tiempo posible a veces deben trabajar de forma independiente y otras deben colaborar), el juego también esconde importantes retos relacionados con el análisis de los niveles (por ej. determinar qué zonas son alcanzables por cada uno de los agentes), la planificación de los movimientos (por ej. el orden en el que se recogen los diamantes puede determinar si el nivel se puede completar), y el control preciso del movimiento (por ej. ser capaz de parar tras un salto sin salirse de la plataforma objetivo). De este modo, las competiciones albergan 3 modalidades diferentes en las que puede competir solo el círculo, solo el rectángulo o ambos agentes de manera cooperativa.

En este artículo describimos brevemente las ideas que usamos para implementar los agentes individuales con los que participamos en la última competición celebrada en IJCAI y CoG 2023 donde ganamos la modalidad individual del rectángulo y quedamos segundos en la modalidad individual del círculo. En líneas generales, nuestra estrategia consiste en analizar el nivel para crear un grafo de navegación entre plataformas, generar un plan abstracto que describe el movimiento de los agentes, y delegar la ejecución de las acciones concretas en *scripts* que utilizan conocimiento experto (en este caso sobre movimientos físicos) para ejecutar las acciones de forma precisa o usar aprendizaje por refuerzo cuando no somos capaces de programar una solución efectiva.

El resto del artículo tiene la siguiente estructura. En el apartado 2, recopilamos algunos trabajos donde se explican distintas aproximaciones usadas por otros equipos en competiciones anteriores. En el apartado 3, describimos

la aproximación empleada por nuestros agentes. En el apartado 4, describimos brevemente los resultados obtenidos. Finalmente, en el apartado 5, recopilamos nuestras principales conclusiones y proponemos líneas de trabajo futuro.

2. Trabajo relacionado

Los juegos cooperativos plantean retos interesantes desde el punto de vista del diseño de agentes inteligentes [8]. En la competición Geometry Friends [6,11], la cooperación surge de la necesidad de recoger diamantes distribuidos por el nivel en el menor tiempo posible. Para ello, los agentes pueden separarse para recoger distintos diamantes en paralelo y apoyarse el uno en el otro y combinar sus movimientos para alcanzar diamantes que de otro modo serían inaccesibles.

Muchos de los agentes presentados en ediciones anteriores utilizan una aproximación similar a la nuestra, en la que el problema se divide en 3 fases: un análisis cuidadoso de los elementos que componen el nivel, la generación de un plan que permita alcanzar todos los diamantes y la ejecución del plan. Para resolver la fase de planificación, se han utilizado distintos tipos de búsqueda heurística, como puede ser A^* [2], pero también estrategias basadas en árboles de Monte Carlo [3] o árboles aleatorios de exploración rápida [4,9].

Otros trabajos combinan estrategias de búsqueda con técnicas de aprendizaje por refuerzo. Si bien la ausencia de un generador automático de niveles hace que sea difícil aprender a resolver niveles completos sin sufrir sobre-aprendizaje [14], las técnicas de aprendizaje pueden resultar útiles para resolver ciertos problemas concretos. Por ejemplo, João Quitério et al. [7] proponen usar un algoritmo de búsqueda para decidir el orden en el que se deben visitar las plataformas, y aprendizaje por refuerzo para mover el agente dentro de cada plataforma y decidir la mejor forma de moverse a la siguiente. Iima et al. [1], por su parte, tratan de que el agente aprenda a navegar por el nivel a partir de trazas generadas mediante A^* .

Sin embargo, los agentes más exitosos en las ediciones más recientes son aquellos que estudian la conectividad del grafo simulando las trayectorias del agente del círculo. Esta propuesta, debida a Hiroya Oonishi y Hitoshi Iima [5], mejoró en gran medida la capacidad de generalización de los agentes a nuevos niveles. Más recientemente, Ricardo Ari Sequeira [12] extendió esta idea a las competiciones del rectángulo y cooperativa, lo que le otorgó la victoria en la edición de 2019.

3. Nuestros agentes

La estrategia empleada por nuestros dos agentes es similar, aunque existen pequeñas diferencias debido a que cada uno de ellos puede realizar distintas acciones. Conceptualmente, resolvemos el problema en 3 fases: analizar el nivel y generar un grafo de navegación entre plataformas, generar un plan abstracto que describe el orden en el que se deben visitar las plataformas, y ejecutar el plan delegando las acciones concretas en *scripts* que usan conocimiento experto

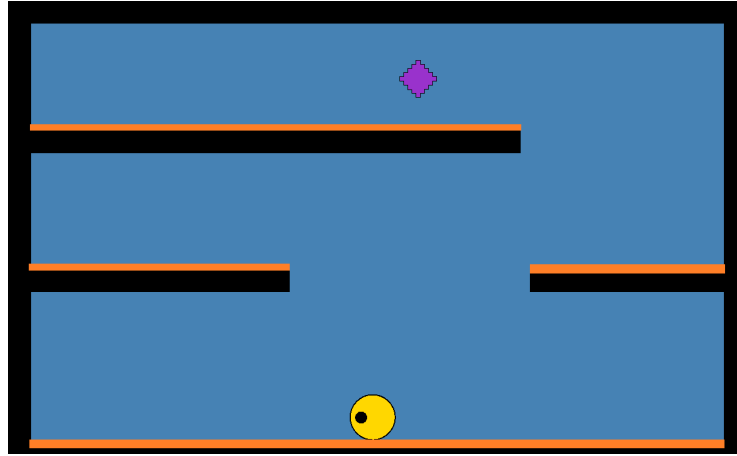


Figura 2. Ejemplo de análisis de un nivel donde se han identificado 4 plataformas (en color naranja) y un diamante.

sobre los movimientos físicos de las entidades. A continuación, describimos cada una de estas fases con más detalle.

3.1. Análisis del nivel

Geometry Friends es un juego de plataformas y, como tal, la navegación entre plataformas juega un papel fundamental en la estrategia para superar el juego. El objetivo de esta fase es analizar el nivel para construir un grafo donde los nodos representan plataformas y las aristas movimientos que permiten al agente cambiar de plataforma. Para construir este grafo, asumimos que cada uno de los diamantes del nivel está asociado a una plataforma (se puede alcanzar sin salir de ella) o a una acción de cambio de plataformas (por ej. mediante un salto desde una plataforma a otra).

Para analizar el nivel, utilizamos una rejilla de píxeles que nos permite identificar los diamantes, las plataformas (superficies continuas horizontales) y los espacios vacíos entre ellas. El caso del círculo es el más sencillo, porque asumimos que podrá desplazarse por cada una de las plataformas simplemente rodando. Por ejemplo, la figura 2 muestra las cuatro plataformas y el diamante que se han identificado en un nivel. El caso del rectángulo es más complejo, ya que debemos tener en cuenta que puede necesitar cambiar de forma para recorrer la plataforma (por ejemplo, disminuir su altura si hay una pared a media altura en el recorrido). En este caso, únicamente consideramos tres de las formas posibles (horizontal, cuadrado y vertical) y etiquetamos cada fragmento de la plataforma con las formas que el rectángulo puede usar.

A continuación, definimos un conjunto de *movimientos* para cada agente, que les permiten cambiar de plataforma y recolectar diamantes. Cada uno de estos movimientos representa secuencias de acciones básicas que posteriormente

serán ejecutadas mediante *scripts* que tienen en cuenta la situación actual del agente para realizar el movimiento de la forma más precisa y sencilla posible. Los movimientos disponibles para el círculo incluyen la capacidad de superar pequeños escalones, alcanzar cierta velocidad y saltar entre plataformas, o dejarse caer por el borde de una plataforma hasta otra plataforma inferior. Por ejemplo, para modelar las trayectorias del movimiento de salto usamos las ecuaciones de un movimiento uniformemente acelerado

$$\begin{cases} x(\mathbf{t}) = x_0 + v_x \mathbf{t} \\ y(\mathbf{t}) = y_0 + v_y \mathbf{t} - \frac{g\mathbf{t}^2}{2} \end{cases}$$

donde (x_0, y_0) representa la posición inicial del centro de masas, (v_x, v_y) la velocidad inicial del agente, g la aceleración de la gravedad y \mathbf{t} el tiempo transcurrido. La simulación de este movimiento es un poco más compleja de lo que parece inicialmente porque debe tener en cuenta los posibles rebotes contra paredes durante el salto, lo que puede provocar interacciones con el entorno poco intuitivas y complicadas de modelar.

Para el rectángulo, necesitamos considerar un mayor número de movimientos ya que puede cambiar de plataforma y coger diamante de más formas distintas. Por ejemplo, el rectángulo puede cambiar su forma para recoger un diamante o dejarse caer por un hueco, pero también como resultado de un giro tras tropezar con un obstáculo y caer hacia adelante. De este modo, consideramos movimientos para cambiar de plataforma superando un pequeño hueco entre ellas, dejarse caer por el borde de una plataforma cambiando o no su forma durante la caída, tropezar con un escalón para superarlo, caer por un hueco apoyándose o no en las paredes adyacentes, etc.

A continuación, simulamos un gran número de estos movimientos debido a la gran variedad de situaciones a las que puede tener que enfrentarse. Durante el proceso, descartamos aquellos que no permiten llegar a otras plataformas o recoger diamantes, así como aquellos movimientos imposibles que necesitarían una velocidad inicial que no es posible alcanzar en el espacio disponible en la plataforma inicial. Aún así, el número de movimientos disponibles es demasiado alto, por lo que seleccionamos un reducido subconjunto para cada cambio entre pares de plataformas dando prioridad a los movimientos que permiten recoger diamantes y tratando de evitar los movimientos con más riesgo (por ejemplo, los saltos que pasan demasiado cerca de obstáculos o movimientos complejos que fallan con más frecuencia).

Como resultado de esta fase, obtenemos un grafo que describe la navegación entre plataformas mediante un subconjunto de los movimientos posibles, aquellos que se han considerado más adecuados (ver figuras 3 y 4). Cada uno de estos movimientos lleva asociada información importante para su realización, como el tipo de movimiento, la posición y plataforma de inicio, la velocidad inicial, la posición y plataforma final, los diamantes capturados durante el movimiento, etc.

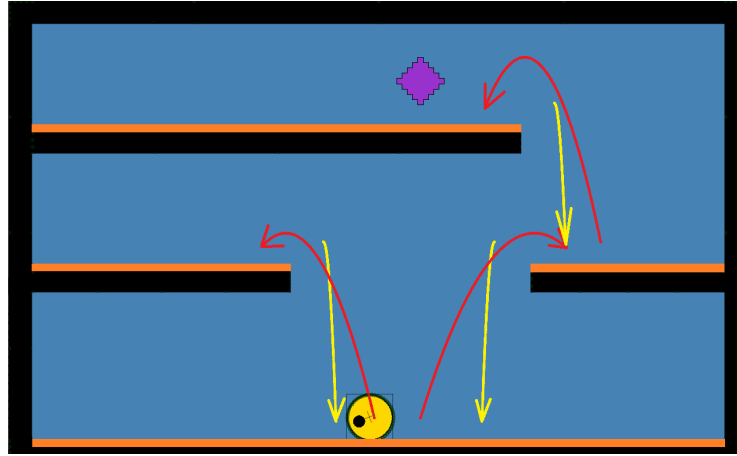


Figura 3. Representación sobre el nivel del grafo de navegación entre plataformas.

3.2. Plan de navegación

A partir del grafo de navegación obtenido en la fase anterior (ver figura 4), obtenemos un plan de navegación que consiste en la secuencia de movimientos que el agente debe realizar para recolectar todos los diamantes del nivel. Como se trata generalmente de un grafo pequeño (gracias al agresivo filtrado de movimientos de la fase anterior), podemos calcular el plan con el número mínimo de movimientos mediante una búsqueda en anchura con control de repeticiones. La búsqueda también tiene un límite de tiempo, de forma que si no termina de explorar todas las combinaciones posibles devuelve el mejor plan que haya encontrado hasta ese momento, es decir, el que permite recoger el mayor número de diamantes.

Para la generación del plan, asumimos que los agentes recogerán todos los diamantes asociados a la plataforma actual antes de transitar a la siguiente. Además, el plan no incluye las acciones concretas que deben ejecutarse para desplazarse por cada una de las plataformas.

3.3. Ejecución del plan

Para ejecutar un movimiento desde una plataforma y alcanzar un diamante o llegar a otra plataforma distinta, es necesario alcanzar una determinada posición inicial con una determinada velocidad. En ocasiones, esto puede lograrse avanzando directamente hacia la posición inicial del movimiento, pero en otras ocasiones puede ser necesario retroceder para poder coger suficiente impulso. Por ejemplo, consideremos la situación de la figura 5 donde el círculo está moviéndose actualmente con una velocidad hacia la derecha (flecha verde) y necesita llegar al punto de inicio del siguiente movimiento (punto rojo) con una determinada velocidad (flecha amarilla). Para hacerlo, debe continuar avanzando hacia

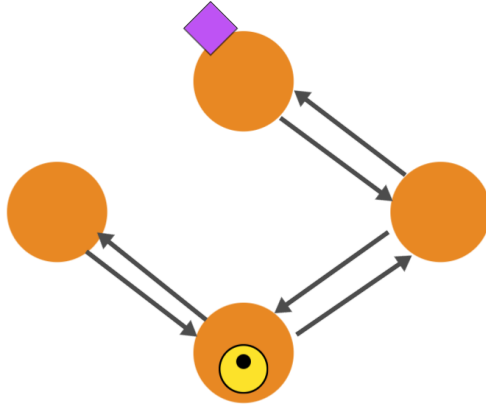


Figura 4. Grafo de navegación entre plataformas.

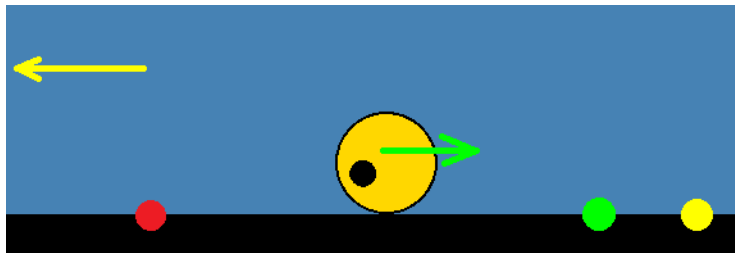


Figura 5. Para alcanzar la posición objetivo (punto rojo) con la velocidad adecuada (flecha amarilla), el círculo debe moverse en dirección opuesta para coger impulso. La flecha verde representa la velocidad actual, el punto verde la posición donde el círculo debe empezar a frenar, y el punto amarillo la posición donde debe empezar a acelerar hacia la izquierda para alcanzar el objetivo y poder realizar el movimiento.

la derecha, empezar a frenar en el punto verde y acelerar hacia la izquierda al alcanzar el punto amarillo.

Para resolver todas estas situaciones y tratar de cumplir las precondiciones de los movimientos, usamos un pequeño sistema de reglas basado en las ecuaciones de movimiento que tienen en cuenta todas las posibles situaciones que se dan al combinar la posición y velocidad actual del agente con la posición y velocidad necesarias para realizar el siguiente movimiento. En el caso del rectángulo debemos considerar, además, que puede ser necesario cambiar de forma durante el desplazamiento por la plataforma.

Una vez se cumplen las precondiciones del siguiente movimiento del plan, delegamos la ejecución de dicho movimiento a un *script* programado en C# que se encarga de ejecutar la secuencia de acciones básicas que componen el movimiento de forma precisa teniendo en cuenta la situación actual.

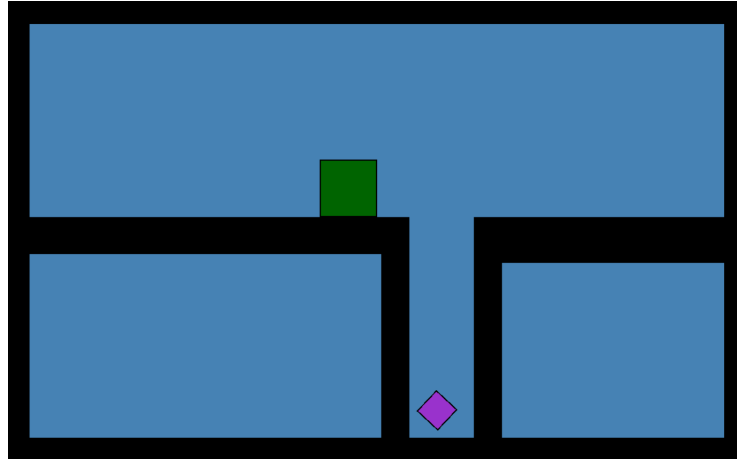


Figura 6. Nivel que muestra el movimiento más difícil de ejecutar del rectángulo.

En caso de que no sea posible cumplir la precondición del siguiente movimiento del plan o la ejecución del movimiento falle (porque la trayectoria no alcanza el diamante o la plataforma objetivo), es necesario replanificar para encontrar otro plan que permita recolectar los diamantes que aún quedan en el nivel.

Cabe destacar uno de los movimientos del rectángulo que usamos para hacer que el agente se deje caer por el hueco entre dos plataformas, tal y como se muestra en la figura 6. Cuando el hueco es suficientemente grande, el rectángulo puede dejarse caer por el borde sin miedo a colisionar con la pared de enfrente. Por otro lado, cuando el hueco es suficientemente pequeño, el rectángulo puede colocarse de forma que cada uno de los extremos este apoyado en una de las plataformas y cambiar de forma para reducir la anchura y de esa forma caer en vertical. Sin embargo, cuando el hueco tiene un determinado tamaño intermedio ninguna de estas dos estrategias funciona de forma consistente y, en ocasiones, el rectángulo gira y se queda atorado en el hueco sin posibilidad de salir. Incluso los jugadores humanos tienen problemas para resolver esta situación. Al no tener una estrategia clara de resolución, decidimos delegar la ejecución de este movimiento a un sistema entrenado mediante Q-Learning [13]. El sistema usa la posición (relativa al hueco), velocidad y altura del agente, y la anchura del hueco para decidir la siguiente acción a ejecutar. Si bien este sistema nos proporciona una manera relativamente sencilla de lidiar con este movimiento, nuestro rectángulo sólo es capaz de resolver satisfactoriamente esta situación en el 62 % de las ocasiones.

4. Resultados

Las competiciones de Geometry Friends incluyen 3 modalidades: individual círculo, individual rectángulo y cooperativa. En cada una de ellas los agentes

	Diamantes niveles públicos	Diamantes niveles privados	Puntuación total
thunder	13/13	13.6/14	5,380
UCM	13/13	13.5/14	5,350
RRT2017	13/13	13.4/14	5,320

Cuadro 1. Resultados Circle Track IJCAI-CoG 2023

compiten en un total de 10 niveles, 5 de los cuales son públicos (conocidos por los participantes durante el desarrollo de los agentes) y 5 privados (no se desvelan hasta finalizado el periodo de participación). Cada agente se ejecuta 10 veces en cada uno de los niveles, obteniendo así una puntuación media por nivel. La puntuación final de cada agente se obtiene sumando las puntuaciones medias obtenidas en cada uno de los niveles. En cada ejecución, la puntuación se calcula mediante la fórmula

$$\text{SCORE} = V_{\text{compl}} \times \frac{\text{max}T - \text{agent}T}{\text{max}T} + (V_{\text{col}} \times N_{\text{col}}),$$

donde V_{compl} es la puntuación recibida por completar el nivel, es decir, por conseguir todos los diamantes del nivel, $\text{max}T$ es el tiempo límite del nivel, $\text{agent}T$ es el tiempo empleado por el agente en completar el nivel, V_{col} es la puntuación por conseguir un diamante y N_{col} es el número de diamantes obtenidos. Los valores de V_{compl} , V_{col} y $\text{max}T$ son proporcionados por la organización, y varían de una edición a otra.

Nuestros agentes UCM participaron en la competición de Geometry Friends celebrada en 2023 en las conferencias International Joint Conferences on Artificial Intelligence (IJCAI) e IEEE International Conference on Games (CoG) obteniendo muy buenos resultados. En la modalidad individual del círculo quedamos segundos a muy pocos puntos del primero, y en la modalidad individual del rectángulo ganamos. También ganamos holgadamente la modalidad cooperativa, aunque en este trabajo nos hemos centrado en los agentes individuales.

Los resultados del agente círculo pueden verse en la tabla 1. Este agente fue capaz de superar todos los niveles públicos y privados en todas las ocasiones excepto un nivel que ningún agente consiguió resolver de forma consistente. Este nivel fue el que decidió el ganador por un solo diamante sobre un total de 100 ejecuciones. Nuestro agente fue capaz de resolver con soltura retos como gestionar plataformas de distintos colores, decidir el orden correcto en el que había que recoger los diamantes y ejecutar saltos y movimientos precisos para navegar por las plataformas.

Los resultados del agente rectángulo pueden verse en la tabla 2. Nuestro agente obtuvo una puntuación excepcional al completar todos los niveles en todos los intentos excepto un nivel en el que no consiguió coger uno de los diamantes en dos de las diez ejecuciones. En este caso, la diferencia de puntuación entre nuestro agente y el segundo clasificado es mucho mayor, pues el segundo no fue capaz de completar ninguno de los niveles privados de forma consistente. Nuestro agente demostró ser capaz de identificar y gestionar plataformas de colores, planificar

	Diamantes niveles públicos	Diamantes niveles privados	Puntuación total
UCM	17/17	15.8/16	6,440
thunder	15.2/17	11,7/16	5,030
RRT2017	14.3/17	10,2/16	4,490

Cuadro 2. Resultados Rectangle Track IJCAI-CoG 2023

rutas de navegación teniendo en cuenta la imposibilidad de volver a plataformas superiores del rectángulo, simular y ejecutar trayectorias de caída precisas, y cambiar de forma cuando la situación lo requiera.

5. Conclusiones y trabajo futuro

En este trabajo, hemos descrito los agentes individuales que presentamos a la competición de Geometry Friends de 2023. La estrategia utilizada por los agentes se basa en 3 fases principales. La primera fase es el análisis del nivel con el objetivo de crear un grafo de navegación entre plataformas. En este grafo, los nodos representan plataformas y las aristas movimientos que permiten ir desde una plataforma a otra y recoger los diamantes distribuidos por el nivel. Para calcular las aristas, usamos simulaciones físicas del movimiento de los agentes y posteriormente seleccionamos los movimientos más interesantes, bien sea porque permiten recoger diamantes o porque implican acciones con poco riesgo de fallar. La segunda fase consiste en generar un plan de navegación, a partir del grafo anterior, que permita al agente recolectar todos los diamantes. La tercera fase consiste en ejecutar el plan de navegación transformando los movimientos que lo componen en secuencias de acciones concretas. Para ello, utilizamos scripts programados en C# que tienen en cuenta el entorno para ejecutar los movimientos de forma precisa, o aprendizaje por refuerzo cuando no somos capaces de programar una solución efectiva.

Los resultados obtenidos son muy prometedores. La combinación de técnicas de representación abstracta del nivel como un conjunto de plataformas interconectadas, la generación de un plan mediante la selección de movimientos seguros para navegar entre plataformas, y la combinación de *scripts* y técnicas de aprendizaje para ejecutar las acciones de forma precisa ha obtenido buenos resultados. Sin embargo, esta aproximación conlleva un gran esfuerzo para modelar y codificar el conocimiento experto necesario. En el futuro, nos gustaría explorar otras vías que permitan resolver el problema aprendiendo el conocimiento necesario de forma más automática.

Agradecimientos

Este trabajo ha sido financiado parcialmente por el Ministerio de Ciencia, Innovación y Universidades (PID2020-114596RB-C21, PID2021-123368OB-I00) y la Universidad Complutense de Madrid (Grupo 921330).

Referencias

1. Iima, H., Oonishi, H.: Deep learning for designing an AI player of the puzzle game geometry friends. In: 6th International Conference on Computational Science/Intelligence and Applied Informatics, CSII 2019, Honolulu, HI, USA, May 29-31, 2019. pp. 31–36. IEEE (2019). <https://doi.org/10.1109/CSII.2019.00013>, <https://doi.org/10.1109/CSII.2019.00013>
2. Kim, H.T., Kim, K.J.: Hybrid of Rule-based Systems Using Genetic Algorithm to Improve Platform Game Performance. *Procedia Computer Science* **24**, 114–120 (December 2013). <https://doi.org/10.1016/j.procs.2013.10.033>
3. Kim, H.T., Yoon, D.M., Kim, K.J.: Solving Geometry Friends using Monte-Carlo Tree Search with directed graph representation. In: 2014 IEEE Conference on Computational Intelligence and Games. pp. 1–2 (August 2014). <https://doi.org/10.1109/CIG.2014.6932923>
4. LaValle, S.M.: Rapidly-exploring random trees: a new tool for path planning. The annual research report (1998)
5. Oonishi, H., Iima, H.: Improving generalization ability in a puzzle game using reinforcement learning. In: 2017 IEEE Conference on Computational Intelligence and Games (CIG). pp. 232–239 (2017). <https://doi.org/10.1109/CIG.2017.8080441>
6. Prada, R., Lopes, P., Catarino, J., Quiterio, J., Melo, F.S.: The geometry friends game AI competition. In: 2015 IEEE Conference on Computational Intelligence and Games, CIG 2015, Tainan, Taiwan, August 31 - September 2, 2015. pp. 431–438. IEEE (2015). <https://doi.org/10.1109/CIG.2015.7317949>, <https://doi.org/10.1109/CIG.2015.7317949>
7. Quitério, J., Prada, R., Melo, F.S.: A reinforcement learning approach for the circle agent of geometry friends. In: 2015 IEEE Conference on Computational Intelligence and Games (CIG). pp. 423–430 (August 2015). <https://doi.org/10.1109/CIG.2015.7317938>
8. Rocha, J.B., Mascarenhas, S.F., Prada, R.: Game mechanics for cooperative games (2008), <https://api.semanticscholar.org/CorpusID:54855107>
9. Salta, A., Prada, R., Melo, F.S.: Solving motion and action planning for a cooperative agent problem using Geometry Friends. In: Proc. 20th Portuguese Conf. Artificial Intelligence. pp. 86–97 (2019)
10. Salta, A., Prada, R., Melo, F.S.: A Game AI Competition to foster Collaborative AI research and development. *IEEE Transactions on Games* **13**(4), 398–409 (2020)
11. Salta, A., Prada, R., Melo, F.S.: A game AI competition to foster collaborative AI research and development. *CoRR* **abs/2010.08885** (2020), <https://arxiv.org/abs/2010.08885>
12. Sequeira, R.A.: Building a Multi-Agent Learning System for Geometry Friends. Master’s thesis, Instituto Superior Técnico, Universidade de Lisboa (October 2019)
13. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. The MIT Press, second edn. (2018), <http://incompleteideas.net/book/the-book-2nd.html>
14. Özgen, A.C., Fasounaki, M., Ekenel, H.K.: Generalized circle agent for geometry friends using deep reinforcement learning. In: 2018 26th Signal Processing and Communications Applications Conference (SIU). pp. 1–4 (2018). <https://doi.org/10.1109/SIU.2018.8404596>

Liquid Gym: an open stealth game test-bed for AI techniques controlling complex characters*

Ismael Sagredo-Olivenza^{1*†}, Pablo Gutiérrez-Sánchez^{1*†},
Marco A. Gómez-Martín^{1*} and Pedro A. González-Calero^{1*}

^{1*}Software Engineering and Artificial Intelligence Department,
Complutense University of Madrid, Facultad de Informática, Madrid,
Span.

*Corresponding author(s). E-mail(s): isagredo@ucm.es;
pabgut02@ucm.es; marcoa@fdi.ucm.es; pagoncal@ucm.es;

[†]These authors contributed equally to this work.

Abstract

This paper introduces Liquid Gym, a training and testing environment for various machine learning models in Unity. Liquid Gym extends the functionalities of ML-Agents and OpenAI Gym, facilitating the introduction of new algorithms, diverse testing environments, and comparisons with state-of-the-art classical control systems from the video game industry, such as Behavior Trees, Hierarchical Task Networks, or Finite State Machines. Currently, we have implemented both imitation learning and reinforcement learning approaches within a stealth game testing environment named Liquid Snake. Our intention is to broaden the range of supported models and testing environments in the future.

Keywords: Reinforcement Learning, Machine Learning, Video Games, Environments

1 Introduction

After DeepMind began to enjoy great success in the use of Deep Q-Learning (DQN) in different environments, originally in Atari games [1] and later in much more complex cases such as AlphaGo [2] or the Starcraft video-game [3], numerous applications of

*This work has been partially supported by the Spanish Ministry of Science and Innovation under grant PID2021-123368OB-I00.

this algorithm have been developed successfully. The results achieved by DeepMind did not go unnoticed by academia, as can be seen at [4–6].

Apart from DeepMind, another of the main contributors to the diffusion of DQN has been OpenAI which, in addition to multiple works related to Reinforcement Learning (RL) such as OpenAI Five [7], has developed a testing environment for RL-driven agents, initially called Gym [8] and nowadays re-branded as Gymnasium¹. This environment enables AI practitioners to test different machine learning algorithms with a large collection of sample environments, ranging from the aforementioned Atari games, to many other classic problems such as Cart Pole, Pendulum, Mountain Car, etc.

This platform and its API have become a benchmark for the evaluation and comparison of different RL algorithms, as in addition to the built-in environments, there are many third-party ones based on Gymnasium, such as Flappy Bird², Slime Volleyball Gym Environment³, or stable-retro⁴, with different implementations of classic games such as Super Mario Bros 2 Japan (Lost Levels in the West) from Nintendo Entertainment System (NES), Punch Out (NES), Tetris (Game Boy), Virtua Fighter (32X) or Mortal Kombat (Genesis), among others.

Most of these implementations are either relatively small games (such as Slim Volley or Flappy Bird) or rely on an emulator due to them being titles developed for older consoles, as in the case of stable-retro. This framework is therefore suitable for training and evaluating agents engaging with a selection of simple games, but falls short for assessing the ability of RL-based strategies and other machine learning and AI techniques to be valuable in other major areas of the video-game industry, such as controlling the behaviour of non-playable characters (NPCs) or performing automated testing tasks, on the one hand due to the lack of complexity or closeness to modern games in terms of mechanics and scale, and on the other because of the inability to access and modify the source code of these environments.

One of the most successful implementations compatible with the Gymnasium API, and which partially addresses the limitations of the other test environments included in Gymnasium, is ML-Agents [9]. This tool allows developers to create games built on the popular Unity3D engine [10] and use them as test environments for RL and Imitation Learning (IL) algorithms. The ML-Agents Toolkit has been used in several research works [11, 12], inspiring the creation of challenges and agent creation competitions in complex environments such as Obstacle Tower [13], as well as in commercial games such as Source of Madness⁵, where ML-Agents was used to generate the behaviours of a number of enemies. Today, ML-Agents offers implementations of some state-of-the-art algorithms in reinforcement learning (such as PPO [14] or SAC [15]) and imitation learning (such as Behavioural Cloning [16] or GAIL [17]), as well as additional extensions to include reward generation modules based on intrinsic motivation or curiosity through Random Network Distillation [18], curriculum learning, and even neural architectures based on Hyper Networks [19] to dynamically adapt agents’

¹<https://gymnasium.farama.org/>

²<https://github.com/markub3327/flappy-bird-gymnasium>

³<https://github.com/hardmaru/slimevolleygym>

⁴<https://github.com/Farama-Foundation/stable-retro>

⁵<https://sourceofmadness.com/>

behaviours based on target signals or modules supported by transformers and attention systems [20] to receive variable-length observations. Despite these virtues, this toolkit does come with certain limitations. Just like Gymnasium, it is closely aligned to the use of RL, as well as to specific Python versions and frameworks (Keras or Pytorch) and, despite the recent incorporation of Trainer Plugins to support custom implementations of the training interfaces provided by ML-Agents, these are still very limited, making it difficult to test other algorithms beyond those included in the tool.

There are also other development frameworks that simplify the creation of agents in games such as Serpent.AI ⁶, a project that stopped in 2018 but seems to have timidly resurfaced. This framework helps the creation of ML agents in already created games.

The purpose of this work is to define a test environment which we call Liquid Gym in a similar spirit to Gymnasium, but which allows the comparison between different techniques and use cases beyond RL and gameplay in a transparent and unified way within a single tool. This includes, for example, imitation learning methods, evolutionary algorithms and other more traditional agent control systems such as Behavior Trees (BTs) [21], Finite State Machines (FSMs) or Hierarchical Task Network Planning HTN, and we emphasise the use case of automatic testing in video-games as a major practical application of these techniques outside of gameplay. To this end, we iterate on our testing environment called Liquid Snake, around which we establish the foundations of this system. Currently we have performed preliminary experiments with reinforcement learning (PPO) and imitation learning (Behavioural Cloning + GAIL) in this environment [22], and are planning to extend this work with additional settings and algorithms, which we will describe in more detail in this article.

The remainder of this paper is structured as follows. Section 2 reviews related work. Then, Section 3 presents our main testing environment. Section 4 describes the general architecture of the system. Lastly, Section 5 presents the preliminary work conducted on this framework so far along with some conclusions and future lines of work.

2 Related work

In recent years, advances in the field of machine learning have been progressively integrated into the field of video game development, both in academia and industry, offering innovative solutions to persistent problems that are difficult to solve using classical methods. Perhaps the best known and most mediatised area of this incursion of machine learning into video games is the creation of agents capable of interacting at human or even superhuman levels in classic games. Some emblematic examples are those already mentioned in Section 1, such as DeepMind’s AlphaGo player, as well as the application of deep neural networks in Atari games. The academic articles associated with these advances boast, at the time of writing, 17600 and 13700 citations respectively, evidencing their impact on research in less than 8 years.

Another context where AI agents are experiencing a remarkable growth is in the automated testing and quality control of video games. The quality control of software applications in general is a very costly process in which it is necessary to make use of a large amount of both technical and human resources to detect problems in the

⁶<https://github.com/SerpentAI/SerpentAI>

product before and during its release to the public; video games in particular exhibit an extraordinary complexity as far as software is concerned, and their quality control processes are often accompanied by an additional type of analysis known as playtesting, whereby a set of testers play the game in search of bugs and/or problems with the game design, usually requiring a substantial investment of time and money.

In this sense, having automated agents that are capable of autonomously and regularly playing the game under development in search of errors is a highly attractive idea for the industry, and has been the focus of multiple research projects in recent years. [23] for example uses an automated testing methodology using synthetic, near-human agents focused on finding defects in environments. The use of inverse reinforcement learning techniques here improves both the agents' error-finding ability and their perceived humanness when interacting with the game. [24] uses deep reinforcement learning to explore or exploit game mechanics based on a user-defined reinforcement signal in a first-person shooter (FPS) game. In turn, [25] makes use of generative player modelling methods from "procedural people" over Monte Carlo techniques to generate different play styles through different utility functions and evolutionary algorithms.

Meanwhile, some of the most prominent development companies in the video game industry have conducted and published research along these lines, looking for ways to automate the various quality control processes they encounter in their own projects. Electronic Arts (EA), for instance, proposes a set of deep reinforcement learning methods in which the agent is motivated to explore in the proximity of known trajectories derived from expert demonstrations, in a complex 3D environment using a strategy called Curiosity-Conditioned Proximal Trajectories, in order to generate agents that are able to replicate and augment checks and behaviours taught by traces of human players in different games [26] even in commercial games such as Battlefield V [27]. Along the same lines of the "humanisation" of agents and the need for them to be able to replicate the playing styles of human players, an imitation learning algorithm (Multimodal GAIL) capable of creating autonomous agents that exhibit different playing styles without the need to go through manual reward engineering is described [28].

Continuing with these examples, in [29], a reinforcement learning algorithm is introduced for the generation of agents capable of performing coverage testing of 3D environments by trying to generate a connectivity graph of the different regions of the level during the simulation. Finally, the developer in charge of Candy Crush Saga, discusses a series of strategies based on convolutional neural networks (CNNs) to predict the most "human-like" movement on a game board in order to estimate the difficulty of a new level as soon as it is designed, without the need for playtesting [30].

The key challenge in most of these works, however, appears to be the lack of open source test environments developed on popular game engines that make it possible to reproduce and compare each methodology and domain application on use cases close to those that would be expected in the industry. This is where this work comes into play, proposing an open environment so that different AI practitioners and trainees can find a common ground to benchmark and explore old and new approaches to the various problems of everyday game development.

3 Environment description

Liquid Snake, originally introduced in a preliminary version in [31], is a prototype 3D third-person stealth game built in Unity3D and developed to conduct experiments to test AI techniques in an environment with features close to those of a prototypical commercial video game. In this game, the player's objective is to guide the main character through a series of rooms and locations in search of the level's exit, while avoiding detection by enemies patrolling the area and solving various puzzles. Figure 1 provides a screenshot of the game's appearance in a sample level.



Fig. 1 Top-down capture of a level section from Liquid Snake.

Enemies in this game follow typical stealth game behaviour, sequentially traversing a series of patrol points specified by the level designer, and stopping at each of them for a short period of time to scout the surroundings. When the enemy detects the character within its cone of vision (reflected in Figure 1 using a pink robotic enemy), it chases and shoots at them concurrently until their health points are reduced to 0 or they are out of sight. In the latter case, the enemy proceeds to survey the last place in the environment where it sensed the player and, if it still cannot find the player in the area, returns to its designated patrol route.

In turn, the player has a number of movement options and actions to interact with the environment. Apart from basic movement on the XZ axis by walking or running, the character can jump to overcome low obstacles or reach high platforms that would otherwise be inaccessible, as well as assume a crouched position to hide from enemies behind mid-height obstacles or maneuver under low-ceiling sections. Additionally, the character can interact with certain objects in the scene by approaching their vicinity (as in the vending machine in the figure surrounded by a circular yellow area), triggering different effects on the environment. One of the most typical examples of these

interactions is the presence of triggers that enable or disable laser barriers such as those on the right side of the figure, which serve to block areas.

The game is designed to allow straightforward switching between control systems according to the needs of each use case. To this end, the character object exhibits a series of methods that enable the different actions described above to be invoked, which are then called from different controllers depending on the situation. If the intention is for a human to play in a traditional way, the input is provided by Unity’s input system, mapping each keyboard or controller entry with one of the available actions; on the other hand, when the intention is for an AI controller to manipulate the character, a series of specific actuators are provided for each controller type to translate the intention of the agent and model used into calls to the corresponding actions. Thus, the decision of how an agent’s output is converted into a character’s behaviour is entirely in the hands of the implementer. This facilitates working with a wide variety of models such as neural network-based systems, rule-based approaches, behaviour trees or planners, to name a few, each of which can choose to define actions with different degrees of granularity (discrete, continuous, high- or low-level).

Currently, Liquid Snake comes integrated with the Unity ML-Agents library, described earlier, which provides the necessary support to start testing reinforcement and imitation learning methodologies on our environment. In addition, ML-Agents provides its own wrapper on top of the typical Gym environment interfaces, allowing interaction with environments that make use of the tool through an API well known to developers familiar with Gym. Additionally, we include the Behavior Bricks tool for programming and designing the behaviour of both NPCs and gameplay bots through Behavior Trees, as well as Unity’s native support for state machine programming through Visual Scripting. Also available as an optional module for this approach is our behaviour generation tool via reinforcement learning and formal task specifications, AI Behaviour Graphs, which has been largely tested on Liquid Snake environments. These features make this an attractive testing and experimentation environment not only from the viewpoint of game AI practitioners, but also from a pedagogical perspective, allowing students of AI or game programming disciplines to apply the skills they have learned in scenarios close to real games.

4 Architecture

In this section we describe the architecture we consider most suitable for the centralisation of AI experiments on our environment, including explanations of features already included in our framework as well as desirable properties we are currently working on.

Our current system is designed to execute ML models within the game’s own code (either through CPU or GPU using Computer Shaders) by means of the Unity’s Sentis library, while training is carried out in Python in an engine-independent way. Communication between both parts of the system is managed via a network module enabling the exchange of information via TCP/IP connection. A high-level schematic of the architecture is shown in Figure 2.

As depicted in the diagram, depending on the algorithm’s requirements we can have multiple instances of the game connected to the training module. This is particularly

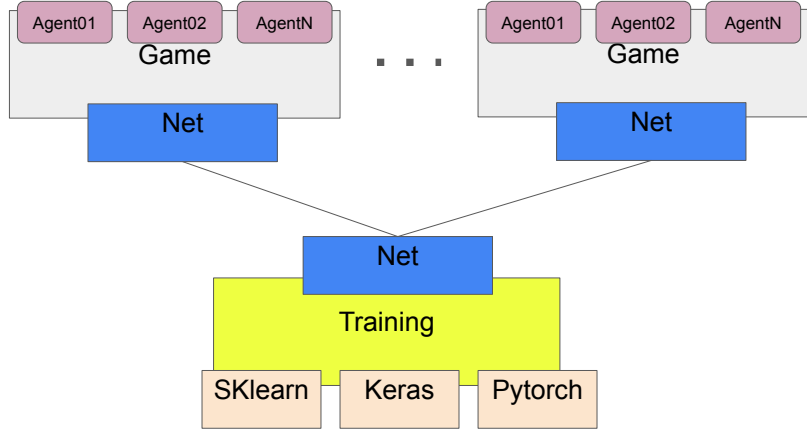


Fig. 2 High-level diagram of the proposed architecture

attractive when one wishes to use a training algorithm that is well conditioned to be parallelised, as may be the case for evolutionary algorithms (where the major cost of execution corresponds to the evaluation the fitness of each individual in the population) or even for RL models where it is necessary to gather significant experience on the environment before making updates to the existing policy. Therefore, the tool enables the creation of different game instances and links them to the trainer module if the model requires it, with the training module assuming the role of server. This separation of training and execution further allows us to integrate the models into the final game and move beyond utilizing the tool solely as a testing and research bench.

It is also important to mention here that the game instances connected to the training server do not always need to only include a single agent. On the one hand, from a conceptual standpoint we may be interested in training several policies or agents simultaneously, such as a bot to represent the player, an enemy or NPC behaviour, a group of collaborative or competitive AIs interacting with each other, or even more abstract AIs learning to control or modify elements of the level or game flow. On the other hand, even in cases where we simply wish to focus on a single behaviour, it is often beneficial from a learning optimisation point of view to include numerous agents in each environment to increase, for instance, the rate of experience collection in RL, or the number of individuals in the population in the case of evolutionary algorithms.

Regardless of the type of model to be used, we will be interested in centralising the training phases in Python in view of the vast collection of resources that this language has at its disposal in the field of ML. Generally, the Python side is only needed during the policy generation phase of our agents, delegating the often complex to implement ML algorithms to widely adopted external libraries, after which it is possible to compile a serializable model of the learned behaviour (e.g., an ONNX file if we are training neural networks). Once a compiled model is available, it is then often straightforward to transfer it to an appropriate game engine-side executor, either programmed manually or using a third-party one, such as Unity's Sentis, continuing the example of the

ONNX files. In this way, the model becomes independent of the Python module and can be used in production. During training, however, it is the Python service that is in charge of receiving environmental information from the game (observations of the agents in the case of RL, or fitness values in the case of evolutionary algorithms) and sending back instructions on how the agents should behave (actions on each tick of the game in RL, or complete neural networks in the evolutionary case). A summary of the most typical use cases in this context would be as follows:

- RL. The service is provided at each game step with the environment observations collected by the agent (e.g. its position, orientation or distance to objects of interest) and the reward signal granted by the game at that time and forwards a representation to the game of the actions that the agent should perform based on the current model on that state. Depending on the type of RL algorithm used, after collecting a certain number of experiences in this way with the current policy, the service will generate new policies that progressively improve the accumulated reward.
- Evolutionary algorithms. The service generates a collection of agents capable of interacting with the game and progressively sends them to the connected game instances for evaluation. After each training episode and for each agent considered as an individual in the current population, the game computes a numerical value representing the individual's quality or fitness, or how well it has managed to complete the assigned task. After receiving the fitness measures of all the individuals in the population, the algorithm produces a new population of (ideally) improved individuals and sends them back to the game for evaluation.
- Imitation learning. The system allows us to record traces of actions performed by the human player, which can then be utilised in Python to train the models via a demonstrations file sent to the service. In simpler versions of this paradigm, such as Behavioural Cloning, it may not be necessary to establish any kind of communication with the game to generate a policy, as the focus is only on producing a network that best fits the input data. If we use an adversarial model such as GAIL, the service acts like the RL case with the difference that the network reward is not given by the game, but by the agent's ability to outsmart an adversarial network trained to distinguish between human and bot behaviours. In the future, we plan to include implementations of interactive algorithms such as DAGGER that allow for a training flow in which the designer can take control of the player to provide real-time demonstrations (e.g., to correct faulty behaviour or explain how to solve new cases), from which the training service dynamically adjusts the policy.

The system, although initially designed to execute neural networks, could easily be extended to other classical ML algorithms such as Random Forest, K-NN, SVM, etc., allowing for comparisons between different approaches depending on the problem we want to solve and the data available. This feature is particularly interesting for use in an academic environment where students can try out different options to understand the functioning and advantages/disadvantages of different models.

Lastly, for classical control systems such as BT or HTN, we have chosen to incorporate them directly into the game engine. Currently, we have Behavior Bricks [32] integrated into Liquid Snake, with plans to include HTN in the future. The aim of

including these classical control is twofold: firstly, to build hybrid models with ML [33], as most current ML implementations cover only certain parts of the behaviour; secondly, to allow for performance comparison between ML models and these classical control systems, which are the most widely used in the video-games today.

5 Preliminary and future work. Conclusions

The first experiments carried out on Liquid Snake were focused on implementing different reinforcement and imitation learning techniques within the field of automated video-game testing [22]. In particular, our main goal was to analyse the feasibility of various methodologies to generate agents capable of dealing with regression testing and level design validation tasks, understood here as those tests that receive a specification usually expressed as a sequence of steps and checks to be executed on the environment and generate a report indicating whether it is still feasible in the current state of the game. Preliminary results on 3 Liquid Snake use cases and levels enabled comparing the suitability of strategies based on PPO guided by dense reward functions with classical IL algorithms such as Behavioral Cloning and GAIL to generate agents capable of carrying out this testing modality, suggesting an advantage of the former over the latter. Training was performed in parallel on different instances of Liquid Snake connected to a single Python service, and includes tasks such as traversing a level with complex navigation, sequentially visiting switches in environments with patrolling enemies, or completing areas with dynamic branching conditions.

Apart from methodologies based on the imitation and reinforcement algorithms provided by ML-Agents, we are currently integrating in our environment neural network training methods based on evolutionary strategies and genetic algorithms in charge of evolving the weights and parameters of the multiple dense layers of the network. With this, we seek to include in our platform another example of a training strategy that naturally benefits from a high degree of parallelisation, and to evaluate the degree of complexity of this approach to more demanding projects and tasks. In this case, the intention is to propose a collection of interfaces through which the user can on the one hand enter their fitness functions of the individuals of the population in Unity, and on the other hand specify the crossover and mutation procedures available in the new population generation phases. We are also considering the possibility of integrating executors and training mechanisms for other classical models not based on neural networks (Random Forest, K-NN, SVM), as mentioned in the previous section.

This paper presents a highly valuable system, suitable for both research and educational environments, where various methods of controlling agents in video games can be created and compared. Drawing on both state-of-the-art classical control systems in the video game industry and increasingly demanded ML-based solutions, this all-in-one tool allows us to hybridize ML behaviors into other models that are more controllable by designers, facilitating comparisons between different approaches to solving a task within the same environment.

Here, we have described some of the applications and test cases we have been working on or are currently integrating into the environment. From this point, we intend to continue expanding the platform with new control paradigms both for classical AI

and machine learning, conducting experiments and developing environments to verify the tool's potential to compare different techniques in games developed on commercial engines whether it be for traditional applications such as NPC control or for emerging strategies in quality control operations and automated testing.

References

- [1] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.: Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602 (2013)
- [2] Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D.: Mastering the game of Go with deep neural networks and tree search. *Nature* **529**(7587), 484–489 (2016) <https://doi.org/10.1038/nature16961> . Accessed 2024-01-05
- [3] Vinyals, O., Babuschkin, I., Czarnecki, W.M., Mathieu, M., Dudzik, A., Chung, J., Choi, D.H., Powell, R., Ewalds, T., Georgiev, P., *et al.*: Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature* **575**(7782), 350–354 (2019)
- [4] Nichol, A., Pfau, V., Hesse, C., Klimov, O., Schulman, J.: Gotta learn fast: A new benchmark for generalization in rl. arXiv preprint arXiv:1804.03720 (2018)
- [5] Perez-Liebana, D., Liu, J., Khalifa, A., Gaina, R.D., Togelius, J., Lucas, S.M.: General video game ai: A multitrack framework for evaluating agents, games, and content generation algorithms. *IEEE Transactions on Games* **11**(3), 195–214 (2019)
- [6] Wymann, B., Espi e, E., Guionneau, C., Dimitrakakis, C., Coulom, R., Sumner, A.: Torcs, the open racing car simulator. Software available at <http://torcs.sourceforge.net> **4**(6), 2 (2000)
- [7] Berner, C., Brockman, G., Chan, B., Cheung, V., Debiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., *et al.*: Dota 2 with large scale deep reinforcement learning. arXiv preprint arXiv:1912.06680 (2019)
- [8] Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., Zaremba, W.: OpenAI gym. arXiv preprint arXiv:1606.01540 (2016)
- [9] Juliani, A., Berges, V.-P., Teng, E., Cohen, A., Harper, J., Elion, C., Goy, C., Gao, Y., Henry, H., Mattar, M., Lange, D.: Unity: A general platform for intelligent agents. arXiv preprint arXiv:1809.02627 (2020)
- [10] Unity-Technologies.: Unity Real-Time Development Platform | 3D, 2D VR & AR

Engine (2022). <https://unity.com/> Accessed 2022-06-03

- [11] Almeida, P., Carvalho, V., Simões, A.: Reinforcement learning as an approach to train multiplayer first-person shooter game agents. *Technologies* **12**(3), 34 (2024)
- [12] Majumder, A., Majumder, A.: Case studies in ml agents. *Deep Reinforcement Learning in Unity: With Unity ML Toolkit*, 513–552 (2021)
- [13] Juliani, A., Khalifa, A., Berges, V.-P., Harper, J., Teng, E., Henry, H., Crespi, A., Togelius, J., Lange, D.: Obstacle Tower: A Generalization Challenge in Vision, Control, and Planning (2019)
- [14] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: *Proximal Policy Optimization Algorithms* (2017)
- [15] Haarnoja, T., Zhou, A., Abbeel, P., Levine, S.: Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor (2018)
- [16] Bain, M., Sammut, C.: A framework for behavioural cloning. In: *Machine Intelligence 15*, pp. 103–129 (1995)
- [17] Ho, J., Ermon, S.: Generative Adversarial Imitation Learning. *arXiv:1606.03476 [cs]* (2016). <https://doi.org/10.48550/arXiv.1606.03476> . <http://arxiv.org/abs/1606.03476> Accessed 2024-02-15
- [18] Burda, Y., Edwards, H., Storkey, A., Klimov, O.: Exploration by Random Network Distillation (2018)
- [19] Chauhan, V.K., Zhou, J., Lu, P., Molaei, S., Clifton, D.A.: A Brief Review of Hypernetworks in Deep Learning (2023)
- [20] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention Is All You Need (2023)
- [21] Isla, D.: Handling Complexity in the Halo 2 AI. <https://www.gamedeveloper.com/programming/gdc-2005-proceeding-handling-complexity-in-the-i-halo-2-i-ai>. GDC 2005 Proceeding, Accessed: 2024-03-18 (2005)
- [22] Gutiérrez-Sánchez, P., Gómez-Martín, M.A., González-Calero, P.A., Gómez-Martín, P.P.: Reinforcement learning with temporal logic specifications for regression testing npcs in video games. In: *2023 IEEE Conference on Games (CoG)*. In Print (2023)
- [23] Ariyurek, S., Betin-Can, A., Surer, E.: Automated Video Game Testing Using Synthetic and Humanlike Agents. *IEEE Transactions on Games* **13**(1), 50–67 (2021) <https://doi.org/10.1109/TG.2019.2947597> . Accessed 2024-01-05

- [24] Bergdahl, J., Gordillo, C., Tollmar, K., Gisslén, L.: Augmenting Automated Game Testing with Deep Reinforcement Learning. In: 2020 IEEE Conference on Games, pp. 600–603 (2020). <https://doi.org/10.1109/CoG47356.2020.9231552> . ISSN: 2325-4289. <https://ieeexplore.ieee.org/document/9231552> Accessed 2024-01-05
- [25] Holmgård, C., Green, M.C., Liapis, A., Togelius, J.: Automated Playtesting With Procedural Personas Through MCTS With Evolved Heuristics. *IEEE Transactions on Games* **11**(4), 352–362 (2019) <https://doi.org/10.1109/TG.2018.2808198> . Accessed 2024-01-05
- [26] Sestini, A., Gisslén, L., Bergdahl, J., Tollmar, K., Bagdanov, A.D.: Automated Gameplay Testing and Validation with Curiosity-Conditioned Proximal Trajectories. *IEEE Transactions on Games*, 1–14 (2022) <https://doi.org/10.1109/TG.2022.3226910> . Accessed 2024-01-05
- [27] Gillberg, J.: Ai for testing: The development of bots that play battlefield v. In: *Game Developers Conference* (2019)
- [28] Ahlberg, W., Sestini, A., Tollmar, K., Gisslén, L.: Generating Personas for Games with Multimodal Adversarial Imitation Learning. arXiv. arXiv:2308.07598 [cs] (2023). <https://doi.org/10.48550/arXiv.2308.07598> . <http://arxiv.org/abs/2308.07598> Accessed 2024-01-05
- [29] Gordillo, C., Bergdahl, J., Tollmar, K., Gisslén, L.: Improving Playtesting Coverage via Curiosity Driven Reinforcement Learning Agents. arXiv. arXiv:2103.13798 [cs] (2021). <https://doi.org/10.48550/arXiv.2103.13798> . <http://arxiv.org/abs/2103.13798> Accessed 2024-01-05
- [30] Gudmundsson, S.F., Eisen, P., Poromaa, E., Nodet, A., Purmonen, S., Kozakowski: Human-Like Playtesting with Deep Learning. In: 2018 IEEE Conference on Computational Intelligence and Games (CIG), pp. 1–8 (2018). <https://doi.org/10.1109/CIG.2018.8490442> . ISSN: 2325-4289. <https://ieeexplore.ieee.org/document/8490442> Accessed 2024-01-05
- [31] Gutiérrez-Sánchez, P., Gómez-Martín, M.A., González-Calero, P.A., Gómez-Martín, P.P., Lara-Cabrera, R., Leiva, A.: Liquid snake: a test environment for video game testing agents. In: *CEV* (2022)
- [32] Sagredo-Olivenza, I., Gómez-Martín, M.A., González-Calero, P.A.: Supporting the collaboration between programmers and designers building game ai. In: *Entertainment Computing-ICEC 2015: 14th International Conference, ICEC 2015, Trondheim, Norway, September 29-October 2, 2015, Proceedings 14*, pp. 496–501 (2015). Springer
- [33] Sagredo-Olivenza, I., Gómez-Martín, P.P., Gómez-Martín, M.A., González-Calero, P.A.: Trained behavior trees: Programming by demonstration to support ai game designers. *IEEE Transactions on Games* **11**(1), 5–14 (2017)

Perspectiva dinámica en videojuegos activos mediante face tracking

Linda García-Rytman¹, Carlos Marín-Lora^{1,2}[0000-0003-1055-7657], and Micaela Y. Martín³[0000-0001-6100-7538] Miguel Chover³[0000-0002-0525-7038]

¹ Institute of New Imaging Technologies, Universitat Jaume I, Castellón, Spain.

² Valgrai: Valencian Graduate School and Research Network of Artificial Intelligence, Camí de Vera s/n, 46022, Valencia, Spain.

Resumen En el contexto actual de desarrollo de videojuegos, el interés por los videojuegos activos ha aumentado considerablemente, impulsado por la disponibilidad de tecnologías que permiten integrar la actividad física en la experiencia de juego. Sin embargo, dentro de los videojuegos activos para cintas de correr, persisten desafíos relacionados con la inmersión del jugador en el entorno virtual. Este estudio investiga el impacto de una técnica de realidad extendida, realizando un seguimiento continuo de la cara del usuario, que ajusta dinámicamente la visualización del mundo virtual según el movimiento facial del jugador en un videojuego diseñado para ser utilizado en una cinta de correr. Se plantea la hipótesis de que esta técnica mejorará la experiencia de juego al aumentar la sensación de inmersión del jugador. Para evaluar esta hipótesis, se realizó un experimento controlado donde los participantes jugaron el mismo juego con y sin la aplicación de la técnica. Los resultados del cuestionario de experiencia de juego sugieren una tendencia hacia una mayor inmersión con la visualización dinámica, aunque las diferencias no fueron estadísticamente significativas.

Keywords: Videojuegos activos · Exergames · Realidad Extendida · Face tracking · Perspectiva dinámica

1. Introducción

En los últimos años, el desarrollo de videojuegos ha experimentado un notable crecimiento, impulsado por la abundancia de recursos que ofrecen las tecnologías actuales a los desarrolladores. Esta expansión ha ampliado las posibilidades de desarrollo, permitiendo el uso de estas aplicaciones interactivas más allá del mero entretenimiento, en campos tan diversos como la salud, la educación, la simulación o el turismo. Dentro de estas áreas, se ha observado un notable aumento en la demanda de aplicaciones y videojuegos diseñados para la realización de actividades deportivas en espacios interiores. Estos videojuegos, conocidos como videojuegos activos o *Exergames*, se caracterizan por integrar la actividad física como parte esencial de la experiencia de juego. En otras palabras, requieren que el jugador realice acciones físicas específicas para controlar las mecánicas del juego y avanzar en la narrativa [19].

En el ámbito de los videojuegos para la práctica deportiva, es crucial distinguir entre dos categorías: los videojuegos activos y los simuladores [17]. Los simuladores se centran en replicar lo más fielmente posible la actividad física, siendo el componente lúdico principal la simulación de la competición [3,16]. Ejemplos de estos simuladores incluyen Zwift para ciclismo y carrera a pie, o Bkool para ciclismo. Por otro lado, los videojuegos activos son aquellos que requieren que el jugador realice movimientos y/o acciones específicas para participar en su jugabilidad y en sus mecánicas, con el propósito de avanzar en la narrativa del juego, se destacan ejemplos notables de videojuegos interactivos como Zombie Run [6], Wii Sports [18], y Ring Fit Adventure [23,15].

Sin embargo, y a pesar de su creciente popularidad, el abanico de videojuegos que puedan usarse sobre cinta de correr es corto, y dentro de los que pueden usarse existen problemas de uso e interacción que no proporcionan una experiencia de juego totalmente satisfactoria [15]. En este sentido, se han identificado múltiples aspectos recurrentes relacionados con la falta de diseños orientados a correr sobre cinta, como la ausencia de elementos de integración y de inmersión del jugador en el entorno virtual. La desconexión entre la acción física del usuario y la respuesta del mundo virtual puede afectar significativamente la experiencia de juego. En este contexto, la búsqueda de soluciones que mejoren la integración entre el jugador y el entorno virtual en los videojuegos activos se debe convertir en un objetivo clave para desarrolladores y diseñadores. La inmersión, como factor fundamental, garantiza una experiencia de juego satisfactoria y envolvente, lo cual no solo influye en el entretenimiento del usuario, sino también en su compromiso y rendimiento durante la actividad física.

En este estudio, se investiga el impacto de un efecto de visualización dinámica, implementado mediante una técnica de realidad extendida (XR), en la experiencia de juego de un videojuego para smartphone diseñado para ser utilizado en una cinta de correr. Esta técnica de XR realiza un seguimiento del rostro (face tracking) del jugador para ajustar las transformaciones geométricas de la cámara del videojuego y transformar la proyección del mundo virtual al ángulo entre la cara del usuario y la cámara. Tratando de emular la experiencia visual que se tendría al mirar a través de una ventana. El objetivo principal es evaluar el impacto de esta técnica en la experiencia de juego mientras se corre en cinta. Para lograrlo, se ha realizado un experimento controlado donde los participantes se enfrentan al mismo juego tanto con la aplicación de la técnica como sin ella. Posteriormente, tras cada una de las dos experiencias, se les han realizado las preguntas del Game Experience Questionnaire (GEQ), concretamente su versión In-game [20]. El análisis de las respuestas en este cuestionario busca identificar diferencias significativas en términos de inmersión, satisfacción y compromiso con el juego. La premisa fundamental de este estudio es que la aplicación de esta técnica mejorará la experiencia de juego en comparación con el uso tradicional. Esta hipótesis sugiere que la introducción de esta ventana virtual, ajustada en tiempo real, puede aumentar la sensación de inmersión en el mundo del juego.

A modo de resumen del documento, en la sección 2 se presentan los antecedentes de este trabajo y los estudios previos en los campos relacionados con el

diseño e implementación de esta experiencia. Seguidamente, el apartado 3 describe del juego y los elementos necesarios para su funcionamiento. La sección 4 plantea el método utilizado para determinar la posición del rostro del usuario y el método de generación del efecto de ventana o perspectiva dinámica. Posteriormente, en el apartado 5, se presentan las hipótesis y objetivos generales del estudio realizado, junto con la descripción de la experiencia con los usuarios. La sección 6 muestra los resultados y una discusión sobre los mismos. Finalmente, en la sección 7 se exponen las conclusiones obtenidas en este trabajo y las posibles líneas de investigación futuras.

2. Antecedentes y estado del arte

La tecnología actual de Realidad Extendida (XR) busca disolver las fronteras entre el mundo virtual y el mundo real. Aunque en muchos videojuegos la perspectiva puede ser modificada dinámicamente para adaptarse a la posición del jugador, es importante matizar que esto es más común en los juegos de primera persona. De hecho, se han clasificado formas de relacionarse con experiencias interactiva según el rol del participante, lo que influye en la adaptación [5].

Además, al referirnos a que el usuario se sienta parte del proceso, es crucial introducir el concepto de agencia, que es la sensación del jugador de que puede interactuar con el mundo virtual y que sus acciones tienen consecuencias. Para profundizar en este concepto, se puede consultar el primer capítulo del libro reciente de [2], que ofrece una introducción interesante sobre la relación entre agencia y videojuegos. Es importante distinguir entre inmersión, cualidad objetiva de los elementos tecnológicos que forman parte de la experiencia, y presencia, sensación subjetiva de sentirse "dentro" del mundo virtual [21]. Son numerosos los campos de la visualización que la están abordando, ya que su uso fomenta que el usuario se sienta parte del proceso y tenga una experiencia más inmersiva [10]. La inmersión que proporciona el ajuste de la perspectiva es clave. Sin ir más lejos, con la tecnología del proyecto *Starline*, Google pretende aumentar el grado de inmersión y presencia en videollamadas [9].

La comprensión básica de cómo funciona esta proyección en perspectiva se basa en el cálculo tradicional de la proyección en gráficos 3D. La mayoría de las aplicaciones gráficas simplemente eligen un campo de visión y especifican las distancias de los planos de recorte cercano y lejano. Para la mayoría de los casos, este método es adecuado para proporcionar una experiencia de juego satisfactoria. Sin embargo, se hacen suposiciones como que el usuario está frente a la pantalla, mirando perpendicularmente hacia ella y observando el centro de la misma. Aunque se generaliza la posición del punto de vista, aún se asume una perspectiva arraigada en el origen y una pantalla ubicada en el plano XY. A pesar de que la configuración del usuario y su pantalla rara vez cumplen con estos criterios, la proyección perspectiva sigue siendo convincente [7,1].

Sin embargo, en campos como la XR, esta suposición no es adecuada, ya que ignorar la orientación de la mirada del jugador en el mundo virtual puede generar experiencias difíciles de tolerar. De hecho, existen investigaciones donde

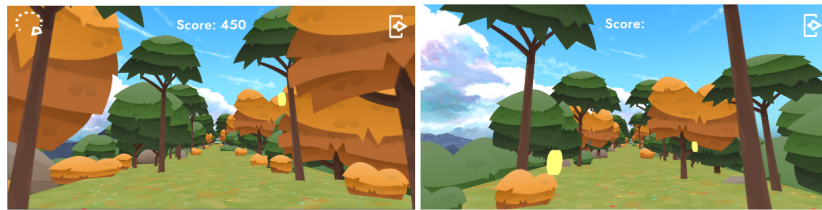


Figura 1: Capturas del escenario del videojuego durante su ejecución.

se emplean técnicas de proyección basadas en la perspectiva para XR, las cuales intentan ajustar el campo de visión a la perspectiva que el jugador busca o espera encontrar [8,22,12,4]. La particularidad de las aplicaciones de XR radica en que a través de dispositivos hardware, como por ejemplo a través de cascos de realidad virtual (VR), es fácil determinar la posición y la orientación de la cabeza para determinar la geometría de la perspectiva. En videojuegos tradicionales, con visualización en pantalla, se requiere de algún método que determine esta geometría. Por ejemplo, en este aspecto destaca un trabajo que utiliza la cámara infrarroja del mando de la Nintendo Wii y una barra de sensores montada en la cabeza, a través de dos LEDs infrarrojos, para rastrear con precisión la ubicación de la cabeza y mostrar imágenes dependientes de la vista en la pantalla [11]. De esta manera se consigue transformar la proyección del mundo virtual en la pantalla en una metáfora de ventana o portal hacia un entorno virtual. Ya que la pantalla reacciona adecuadamente al movimiento de la cabeza y el cuerpo como si fuera una ventana real, creando una ilusión realista de profundidad y espacio.

Hasta el momento, no se ha investigado la aplicación de técnicas de seguimiento facial para generar esta perspectiva dinámica. Aunque es un área prometedora para la investigación, se ha utilizado previamente para obtener información sobre la dinámica de la carrera en una cinta de correr [13,14].

3. Descripción del juego y material utilizado

El videojuego para smartphone utilizado en este experimento simula una experiencia de paseo en primera persona a través de un bosque. El escenario del juego consiste en un único camino recto rodeado de elementos naturales característicos de un bosque, todos ellos representados con un estilo cartoon. Durante la actividad, el jugador está limitado a desplazarse siguiendo este camino y no puede realizar movimientos laterales. Además, a lo largo del recorrido, el jugador encuentra monedas que debe recolectar para aumentar su puntuación en la experiencia. En la Figura 1 se pueden observar dos capturas del escenario.

Sobre este videojuego, desarrollado con el motor de juegos Unity 2022.3.19f1, se han implementado dos modos de visualización distintos: uno que utiliza una visualización estática tradicional y otro que emplea una visualización dinámica con el efecto de perspectiva dinámica, ventana o portal. Para lograr este efecto, se

utiliza la cámara frontal del smartphone como fuente de datos sobre la posición relativa entre la pantalla y el usuario. En la configuración del sistema, dicha cámara se coloca en el soporte de la cinta de correr y captura los movimientos faciales del jugador mediante los complementos XR nativos de Unity.

A nivel de hardware, se utilizó el modelo de smartphone Redmi Note 8, con 64GB de almacenamiento y 4GB de RAM, y las actividades se realizaron en una cinta de correr BH-RC09, que cuenta con una superficie de 155 x 55 cm, permitiendo ligeros movimientos laterales durante el juego y facilitando su uso incluso para personas sin experiencia previa. Por último, aunque el juego puede visualizarse en la pantalla del smartphone, se añadió un dispositivo de visualización más grande: un smart TV LG UHD-70-81003-LA de 70 pulgadas, sobre la que se duplica la pantalla del smartphone para una visualización ampliada.

4. Perspectiva dinámica mediante face tracking

Para implementar el método de perspectiva dinámica, se han utilizado dos frentes fundamentales. En primer lugar, el seguimiento de la posición de la cara del jugador mediante los plugins de XR nativos de Unity. Y en segundo lugar, para interpretar la relación entre la posición de la cámara y la cara del jugador, se ha seguido un método inspirado en los desarrollos previos de Kooima y Lee [8,11]. Este enfoque se ha aplicado específicamente sobre el proyecto desarrollado en Unity para modificar la proyección de la cámara. En la Figura 2 puede observarse un diagrama del proceso que se describe a continuación.

En Unity, el sistema de face tracking detecta la posición del jugador y proyecta un objeto 3D sobre su rostro. Por lo general, este objeto 3D es una representación detallada del rostro que permite deformaciones utilizando blendshapes. Sin embargo, para los propósitos de estos experimentos, no se requiere la detección de expresiones faciales tan específicas, las cuales también pueden consumir recursos. Por ello, se ha optado por utilizar un marcador con el fin de simplificar esta función al máximo. Al arrancar, el sistema de XR cuenta con su propia cámara, la cual genera una salida que muestra el vídeo obtenido de la cámara frontal y la referencia del sistema. Sin embargo, el vídeo no proporciona información relevante más allá de las posibles calibraciones. Por consiguiente, se ha añadido una segunda cámara que proyecta el contenido del mundo 3D en un segundo canal de visualización. De esta manera, el videojuego se ejecuta con dos canales simultáneamente, pero con una disposición en capas, donde el smartphone solo muestra la información del juego proveniente del segundo.

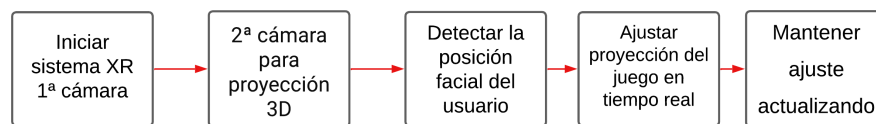


Figura 2: Secuencia de acciones para modificar la proyección de la cámara.



Figura 3: Participantes del experimento durante la actividad.

Una vez detectada la posición de la cara, es momento de configurar el cálculo y ajuste de la proyección de un plano en el espacio tridimensional. Para ello, se almacenará información sobre el tamaño del plano en los ejes XY y la relación de aspecto del mismo. Con estos datos, se calculan y almacenan los vértices del plano, así como los vectores de dirección. Además, se define una matriz de proyección utilizando estos vectores de dirección.

A continuación, se procede a crear y actualizar un cubo de alineación basado en las propiedades del plano de proyección. Este cubo de alineación y el propio plano de proyección se actualizan en cada iteración. Es importante resaltar que se realiza una transformación de los datos desde el espacio local al de coordenadas del mundo. Hasta este punto, se han configurado los vectores y la matriz necesarios para la modificación de los parámetros de la cámara. Y ahora será necesario ajustar los parámetros de la cámara para proyectar la escena en un plano de proyección específico para la perspectiva dinámica.

Mediante los datos calculados sobre el plano de proyección, será necesario determinar los vectores de dirección desde la cámara hasta las esquinas de dicho plano y determinar la distancia desde la cámara hasta el plano de proyección. Utilizando estos valores, se calculan los límites de la proyección y se construye una matriz de proyección. También se ajusta la matriz de vista de la cámara según la posición y la rotación del plano de proyección. Finalmente, se asigna las matrices de vista y proyección calculadas a la cámara. Después de todas las actualizaciones de la escena, se recalculan las matrices de proyección y vista de la cámara utilizando solo la posición más reciente del plano de proyección.

Para garantizar un movimiento automático y rápido de la cámara, se definen variables que determinan los límites dentro de los cuales se moverá la cámara, así como los valores de movimiento en los ejes. Sin embargo, es importante destacar que este método tiene margen de mejora, especialmente en lo que respecta a suavizar la entrada y salida del área de detección y afinar la detección múltiple.

5. Comparativa entre versiones con usuarios

La investigación surge del creciente interés en los videojuegos activos como herramientas para promover la actividad física y mejorar la experiencia de juego. A pesar de los beneficios potenciales que ofrecen para la salud y el entretenimiento, su efectividad para involucrar a los usuarios pueden verse limitadas por la falta de inmersión y conexión entre el jugador y el entorno virtual. En este contexto, el estudio se enfoca en comparar si la técnica propuesta mejora la inmersión y la experiencia de juego frente a una visualización estática más tradicional sobre cinta de correr. Para ello, se analizarán aspectos como el nivel de inmersión, la fluidez, las emociones positivas o negativas, el desafío, la competencia y la tensión, utilizando ambos modos de visualización. En este sentido, para el estudio se han planteado las siguientes hipótesis:

- H1. La visualización dinámica mejora la inmersión.
- H2. La visualización dinámica mejora el flujo de juego.
- H3. La visualización dinámica mejora no genera mayores afecciones negativas.

El experimento fue realizado por 30 personas, 14 mujeres y 16 hombres, con una edad media de 30.1 años. (Desviación Estándar = 12.06; edad mínima 19 años, edad máxima = 58 años). El 70 % de ellos no eran usuarios de cinta y el 30 % restante lo eran. En la Figura 3 se puede observar a varios participantes.

Los participantes fueron programados individualmente en franjas horarias de 20 minutos y se les solicitó ubicarse sobre la cinta y frente a la smart TV. Aleatoriamente se les asignó el modo con el que empezarían y, antes de comenzar, se les dió unos minutos para familiarizarse con el entorno. Una vez preparados, realizaron el recorrido durante 2-3 minutos y luego completaron un cuestionario para evaluar su experiencia de juego después de cada modo. Destacar que el estudio se realizó bajo la aprobación del Comité Ético de la Universidad Jaime I de Castellón, que no se ofrecieron recompensas y que el único requisito era encontrarse aptos para realizar actividad física. El cuestionario elegido para

Tabla 1: Cuestionario de Experiencia de Juego (GEQ)

Componente	Ítems
Competencia	I2. Me sentí exitoso I9. Me sentí hábil
Inmersión sensorial e imaginativa	I1. Me interesó la trama del juego I4. Me pareció impresionante
Fluidez	I5. Me olvidé de todo lo que me rodeaba I10. Me sentí completamente absorbido
Tensión	I6. Me sentí frustrado I8. Me sentí irritable
Desafío	I12. Me sentí desafiado I13. Tuve que esforzarme mucho
Afecciones Negativas	I3. Me sentí aburrido I7. Me pareció pesado
Afecciones Positivas	I11. Me sentí satisfecho I14. Me sentí bien

Tabla 2: Comparación de resultados con 30 muestras.

Ítem	Aspecto	Mediana	Media & SD
I1	Inmersión (tradicional)	2	2,40 ± 1,003
	Inmersión (ventana)	2.5	2,47 ± 1,279
I4	Inmersión (tradicional)	3	2,27 ± 1,172
	Inmersión (ventana)	3	2,60 ± 1,303
I3	Afecciones negativas (tradicional)	1	0,80 ± 0,887
	Afecciones negativas (ventana)	0	0,83 ± 1,085
I7	Afecciones negativas (tradicional)	0	0,27 ± 0,521
	Afecciones negativas (ventana)	0	0,40 ± 0,814
I5	Fluidez (tradicional)	3	2,43 ± 1,251
	Fluidez (ventana)	3	2,70 ± 1,119
I10	Fluidez (tradicional)	2	2,17 ± 1,206
	Fluidez (ventana)	2.5	2,40 ± 1,303

ambas experiencias fue el In-game GEQ [20], que consta de 14 ítems para que los usuarios expresen sus impresiones, valorando cada ítem en una escala tipo Likert de cinco puntos ("muy desfavorable"0 a "muy favorable"4). El GEQ recopila los siguientes siete componentes agrupando dos elementos para cada uno: Competencia (I2, I9), Inmersión sensorial e imaginativa (I1, I4), Fluidez (I5, I10), Tensión (I6, I8), Desafío (I12, I13) , Afecciones Negativas (I3, I7) y Afecciones Positivas (I11, I14). La Tabla 1 muestra los enunciados del GEQ In-game asociados a sus componentes correspondientes.

6. Resultados y discusión

Después de recopilar y analizar los datos recogidos en los cuestionarios con el software para el análisis estadístico IBM SPSS Statistics 29.0.0.0, a continuación se presentan los resultados obtenidos de los experimentos. Iniciamos realizando los histogramas de cada ítem y calculando las estadísticas descriptivas, como se puede observar en la Tabla 2. Esta tabla muestra los ítems asociados a las componentes estudiadas en las hipótesis para ambas modalidades (tradicional y ventana). En la cuarta columna se presenta la mediana de los ítems, mientras que en la última columna se muestra la media y la desviación estándar.

Con la naturaleza ordinal de las variables, se optó por un enfoque no paramétrico utilizando el test de Wilcoxon para determinar posibles diferencias significativas entre ambas modalidades. Este test asume la simetría y la continuidad de las diferencias entre las mediciones. Para verificar estas condiciones, se generaron y analizaron histogramas, se calculó la asimetría y su error estándar, y se verificó que en todos los casos, el valor absoluto de la asimetría era menor que el doble del error estándar. Los resultados de Wilcoxon indicaron que, con un nivel de confianza del 95 %, no se encontraron diferencias significativas en el uso de la ventana dinámica para ningún ítem como puede verse en la Tabla 3.

Entrando en detalle en el análisis por componentes, y comenzando por la inmersión, la media de los resultados para I1 en ambas experiencias fue cercana a 2.5, indicando que en general la trama resultó interesante. Si bien su mediana

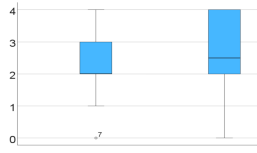


Figura 4: I1. Tradicional/ventana

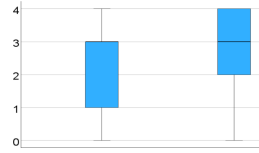


Figura 5: I4. Tradicional/ventana

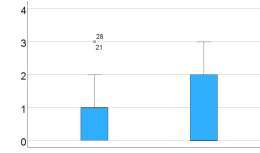


Figura 6: I3. Tradicional/ventana

fue 2 en la metodología tradicional y 2.5 para la ventana, no se observaron diferencias significativas entre ambas ($p=1.617$) como se evidencia en la Figura 4, probablemente debido a que la trama en ambas experiencias es la misma, cambiando sólo el tipo de visualización del contenido del juego. Para I4 se observa una mediana de 3 en ambas modalidades, con medias de 2.27 y 2.6 para las experiencias tradicional y de ventana, respectivamente. Esto sugiere que ambas experiencias fueron percibidas como muy impresionantes por los usuarios. Aunque las diferencias no alcanzaron significancia estadística ($p=0.054$) para el nivel de error definido, se aprecia una mejora en los valores obtenidos en la experiencia de ventana en comparación con la tradicional, como se muestra en la Figura 5.

En relación a las afecciones negativas, específicamente en lo que respecta al aburrimiento, marcado por el I3, se observa en la Figura 6 que, aunque los valores fueron bajos en ambas experiencias (mediana 1 para la versión tradicional y 0 en la ventana), las respuestas en la experiencia de visualización interactiva presentaron una mayor variabilidad. En ambas experiencias más del 75% de los participantes indicaron estar totalmente en desacuerdo con la afirmación de I7, obteniendo mediana 0 en ambas como puede verse en la Figura 7. Esto sugiere que la percepción de pesadez fue mínima en ambas experiencias. En cuanto al flujo del juego y específicamente en cuanto a olvidar todo lo que les rodeaba (I5), con una media de 2.43 y 2.7 para la versión tradicional y la ventana respectivamente, y una mediana de 3, indica que los participantes estuvieron muy de acuerdo con la afirmación en ambas experiencias, sin diferencias significativas entre los grupos ($p=0.148$). En la Figura 8 se pueden apreciar estos resultados. En cuanto a sentirse completamente absorbidos (I10), la media y la mediana fueron ligeramente superiores en la versión de ventana que en la tradicional, aunque se observó una mayor variabilidad en este último caso. Sin embargo, no se

Tabla 3: Tabla con los datos de los ítems

ITEM	I1	I2	I3	I4	I5	I6	I7
Z	-0,500	-0,258	-0,066	-1,927	-1,445	-0,787	-0,758
SIG.ASIN.	0,617	0,796	0,948	0,540	0,148	0,431	0,449
ITEM	I8	I9	I10	I11	I12	I13	I14
Z	-1,190	-0,442	-1,313	-1,165	-0,878	-0,214	0,775
SIG.ASIN.	0,234	0,659	0,189	0,244	0,380	0,831	0,439

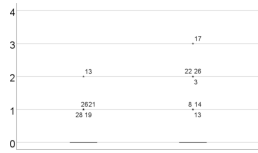


Figura 7: I7. Tradicional/ventana

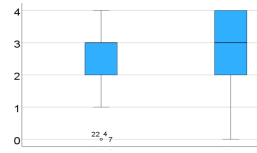


Figura 8: I5. Tradicional/ventana

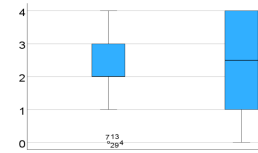


Figura 9: I10. Tradicional/ventana

encontraron diferencias significativas entre los grupos de comparación ($p=0.189$) como puede observarse en la Figura 9.

Sobre las hipótesis, el ítem de mayor importancia en relación a H1 viene marcado por I4, dado que la trama planteada en ambas experiencias es la misma, por lo que se busca determinar si el hecho de la visualización dinámica resulta significativamente más impresionante para los participantes. En relación a esto, aunque rechazamos H1 debido a que las diferencias no llegan a ser estadísticamente significativas ($0.054 > 0.050$), resaltamos la tendencia a valoraciones más altas en la experiencia de la ventana, indicando mayor inmersión en ella en comparación al método tradicional, aunque ambas modalidades resultaron muy inmersivas para los usuarios, lo que puede deberse a que la cinta de correr resulta inmersiva por sí misma en concordancia con estudios anteriores [14]. En cuanto a H2, aunque se rechaza la hipótesis al no encontrar diferencias significativas, destacamos que en ambas experiencias los usuarios estuvieron muy de acuerdo con haberse olvidado de lo que les rodeaba e incluso las valoraciones fueron ligeramente superiores para el caso de la ventana. En cuanto a sentirse completamente absorbidos, dado que es necesario por el tipo de juego tener percepción del entorno, las valoraciones medias pueden estar relacionadas a ello. Por lo tanto, concluimos que la fluidez del juego fue alta para ambas experiencias. Por último, sobre H3, dado que tanto para I3 como I7 no se encontraron diferencias significativas entre los grupos ($p=0.948$, $p=0.449$ respectivamente), por lo que los afectos negativos son similares para ambos grupos. Además, es destacable que los valores fueron cercanos a 0, indicando que los usuarios no se auto percibieron aburridos ni pesados.

7. Conclusiones

En este estudio, se ha investigado el impacto de una técnica de visualización dinámica implementada mediante XR en la experiencia de juego de usuarios de videojuegos activos sobre cintas de correr. Esta técnica ajusta la visualización del mundo virtual según el movimiento facial del jugador, con el fin de aumentar la sensación de inmersión. Se ha utilizado el motor de juegos Unity para calcular y ajustar la proyección de un plano en el espacio tridimensional, adaptando los parámetros de la cámara en función de la posición del plano de proyección.

El enfoque de la investigación se ha centrado en comparar el impacto de la perspectiva dinámica en la experiencia de juego frente a una visualización estática más tradicional. Se han planteado hipótesis sobre cómo esta técnica podría influir en la inmersión, el flujo de juego y las emociones negativas. El experimento, llevado a cabo con 30 participantes, reveló que ambos grupos experimentaron niveles altos de inmersión. Sin embargo, no se observaron diferencias significativas entre la visualización dinámica y estática. Creemos que en los resultados pueden haber influido factores más allá de la experiencia de visualización, como la inclusión de una mecánica de recolección mediante un mando bluetooth. Creemos que esta adición pudo haber desviado la atención de los participantes, lo que posiblemente afectó la evaluación precisa de la experiencia visual. A pesar de esto, mantenemos la creencia de que la técnica de adaptación en tiempo real mediante el face tracking puede proporcionar una experiencia significativamente más inmersiva. Es posible que se requieran ajustes adicionales en la implementación o que otros aspectos del diseño experimental necesitan ser considerados para capturar completamente el impacto de la técnica.

Además, tanto en la experiencia tradicional como en la dinámica, los usuarios reportaron una alta fluidez en el juego y no se observaron diferencias significativas en términos de afecciones negativas. Aunque el método garantiza un movimiento automático y rápido de la cámara, se identificaron áreas de mejora como la suavización de la detección de movimiento y la la detección de múltiples rostros. Se ha destacado el potencial del face tracking para cinta de correr como una forma alternativa de capturar las interacciones del jugador e integrarlas en la mecánica del juego, ofreciendo un gameplay más variado y atractivo.

Acknowledgments. Este trabajo ha sido financiado en el marco del proyecto de investigación CIAICO / 2021 / 037 de la Generalitat Valenciana. También ha recibido apoyo de valgrAI, la Escuela de Posgrado y la Red de Investigación en Inteligencia Artificial, y la Generalitat Valenciana, con cofinanciación de la Unión Europea.

Referencias

1. Akenine-Moller, T., Haines, E., & Hoffman, N. (2019). Real-time rendering. AK Peters/crc Press.
2. Bódi, B. (2023). Videogames and agency (p. 226). Taylor & Francis.
3. Boulos, M. N. K., & Yang, S. P. (2021). Mobile physical activity planning and tracking: a brief overview of current options and desiderata for future solutions. *Mhealth*, 7.
4. Dijkstra-Soudarissanane, S., Gunkel, S. N., & Reinders, V. (2022, June). Virtual visits: life-size immersive communication. In *Proceedings of the 13th ACM Multimedia Systems Conference* (pp. 310-314).
5. Dolan, D., & Parets, M. (2016). Redefining the axiom of story: The VR and 360 video complex. *Tech Crunch*.
6. Faric, N., Potts, H. W., Rowe, S., Beaty, T., Hon, A., & Fisher, A. (2021). Zombies, Run! An evaluation of an app-based exercise intervention. *British Journal of Sports Medicine*, 55(2), 95-100.

7. Foley, J. D. (1996). *Computer graphics: principles and practice* (Vol. 12110). Addison-Wesley Professional.
8. Kooima, R. (2009). Generalized perspective projection. *J. Sch. Electron. Eng. Comput. Sci*, 6(1).
9. Lawrence, J., Goldman, D., Achar, S., Blascovich, G. M., Desloge, J. G., Fortes, T., ... & Tong, K. (2021). Project starline: A high-fidelity telepresence system. *ACM Transactions on Graphics (TOG)*, 40(6), 1-16.
10. Lazo, C., Barroso, J., & Covacho, E. (2013). Phenomenological features of digital communication: interactivity, immersion and ubiquity. *Sociedad de la información*, 44, 169-188.
11. Lee, J. C. (2008). Hacking the nintendo wii remote. *IEEE Perv Comp*, 7(3), 39-45.
12. Makiguchi, M., Sakamoto, D., Takada, H., Honda, K., & Ono, T. (2019, October). Interactive 360-degree glasses-free tabletop 3D display. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (pp. 625-637).
13. Marin-Lora, C., Martin, M., & Chover, M. (2022, November). A face-tracking method for estimating cadence on treadmills. In *International Conference on Ubiquitous Computing and Ambient Intelligence* (949-954). Springer Publishing.
14. Marín-Lora, C., Chover, M., Martín, M. Y., & García-Rytman, L. (2023). Comparative study of interaction methods for mobile gaming while running on a treadmill. *Computers & Graphics*, 117, 164-171.
15. Martin, M., Chover, M., & Marin-Lora, C. (2022). User perception of the gaming experience: Playing Nintendo Ring Fit Adventures on the floor vs. on a treadmill. In *Proceedings of the International Conference on Ubiquitous Computing & Ambient Intelligence (UCAmI 2022)* (pp. 949-954). Springer International Publishing.
16. Martin, M., Chover, M., Sáez, E., & Marín-Lora, C. (2022, November). Comparison between running on a treadmill in a simulator and in a video game. In *International Conference on Ubiquitous Computing and Ambient Intelligence* (pp. 145-156). Cham: Springer International Publishing.
17. Matallaoui, A., Koivisto, J., Hamari, J., & Zarnekow, R. (2017). How effective is "exergamification"? A systematic review on the effectiveness of gamification features in exergames.
18. Meadows III, C. W. (2014). A Wii, a Mii, and a new me? Testing the effectiveness of Wii exergames on children's enjoyment, engagement, and exertion in physical activity. *International Journal of Child Health and Human Development*, 7(1), 37.
19. Oh, Y., & Yang, S. (2010). Defining exergames & exergaming. *Proceedings of meaningful play*, 2010, 21-23.
20. Pallavicini, F., & Pepe, A. (2019, October). Comparing player experience in video games played in virtual reality or on desktop displays: Immersion, flow, and positive emotions. In *Extended abstracts of the annual symposium on computer-human interaction in play companion extended abstracts* (pp. 195-210).
21. Slater, M., Linakis, V., Usoh, M., & Kooper, R. (1996). Immersion, presence and performance in virtual environments: An experiment with tri-dimensional chess. In *Proceedings of the ACM symposium on VR software and technology* (163-172).
22. Steinicke, F., Bruder, G., & Kuhl, S. (2011). Realistic perspective projections for virtual objects and environments. *ACM Transactions on Graphics*, 30(5), 1-10.
23. Wu, Yi-Syuan, et al. "Effect of the Nintendo Ring Fit Adventure exergame on running completion time and psychological factors among university students engaging in distance learning during the COVID-19 pandemic: randomized controlled trial." *JMIR serious games* 10.1 (2022): e35040.

Explorando el color y técnicas pictóricas en la elaboración de entornos inmersivos

Barranco Crespo, María Teresa
UDIT, Universidad de Diseño y Tecnología

Resumen. En el diseño de fondos para entornos interactivos, es esencial conferir una identidad distintiva mediante la cuidadosa integración de elementos como la forma, el color, la iluminación y la narrativa. La interacción de estos elementos no solo está intrínsecamente relacionada con significados culturales, sino que también se ve influenciada por consideraciones de usabilidad, mecánicas y animaciones durante el proceso creativo. Todo esto se realiza con el claro propósito de evocar una percepción emocional específica en el espectador, sin que ello comprometa la intención original.

Este estudio se centra en analizar meticulosamente las técnicas innovadoras desarrolladas hasta la fecha, con la intención de catalogar diversas metodologías y diseñar una cartografía del color que establezca vínculos evidentes con las emociones. Esto se hace con el objetivo de infundir a estos contextos una personalidad distintiva y examinar detenidamente el impacto emocional que ejercen sobre el observador.

Toda esta información, más allá de meramente realzar la estética visual, este enfoque aspira a ampliar los conocimientos de los estudiantes para el desarrollo de proyectos artísticos.

Palabras clave: Diseño de fondos, Entornos interactivos, Color, Inmersión.

1 Introducción

La influencia del color y la inmersión espacial en nuestros sentimientos y emociones es profunda, arraigada en nuestra manera de procesar visualmente nuestro entorno y en cómo esta percepción incide en nuestro bienestar psicológico y emocional.

En su destacada obra "If It's Purple, Someone's Gonna Die: The Power of Color in Visual Storytelling", Patti Bellantoni, junto a un grupo de alumnos internacionales, profundiza en el estudio de cómo las respuestas al color trascienden las barreras culturales, revelando tanto similitudes universales como diferencias individuales en la percepción del color (Bellantoni, 2012). A través de su investigación, Bellantoni descubre que, aunque los estilos personales en la aplicación del color varían, la reacción emocional primaria a los colores específicos es notablemente consistente e instintiva. Los estudiantes, independientemente de su origen cultural, mostraron una tendencia a seleccionar paletas de colores similares para evocar emociones comparables, subrayando una conexión emocional intrínseca entre el color y la expresión emocional.

Los colores llevan consigo significados y asociaciones que varían entre culturas y experiencias personales. Por ejemplo, el blanco se asocia comúnmente con la pureza en muchas culturas occidentales, mientras que en algunas culturas orientales, es el color del luto. Estas asociaciones culturales y personales pueden provocar respuestas emocionales específicas cuando nos encontramos con ciertos colores (Heler, 2004).

Bellantoni destaca que, a pesar de las diversas interpretaciones culturales del color, hay una base emocional común en cómo las personas reaccionan a él. La inclusión del color rojo para expresar "Rabia" en todas las piezas del estudio ilustra este punto. El rojo, aunque puede ser un color de celebración en algunos contextos culturales, como las bodas en India, o simbolizar algo completamente diferente en otros contextos, como el color usado por los vigilantes en Nueva York, evoca una respuesta emocional intensa y apasionada sin importar su significado cultural específico.

Además, la experiencia y exploración del color en diversas culturas enriquecen nuestra capacidad para afinar nuestra propia expresión visual. Al sumergirnos en el vasto espectro de significados y emociones asociadas con el color a través de diferentes culturas, podemos desarrollar una comprensión más profunda y matizada de cómo usar el color para expresarnos de manera más efectiva y resonante.

Por otro lado, sumergirse en un entorno diseñado con una selección intencionada de colores puede moldear significativamente nuestra percepción del mismo, influyendo así en nuestras emociones y conductas. Por ejemplo, la disposición y organización espacial juegan un rol crucial. Los ambientes abiertos y bien ventilados tienden a inducir sentimientos de libertad y relajación, en contraste con los espacios reducidos y saturados, que pueden generar estrés o malestar. Esta inmersión espacial no solo incide en nuestro estado de ánimo momentáneo, sino que también posee el potencial de afectar nuestro bienestar a largo plazo. Un entorno adornado con colores vivos

y saturados puede revitalizarnos y mejorar nuestro estado de ánimo, mientras que uno decorado con tonalidades suaves y neutras favorece la relajación y la serenidad.

Así mismo, la iluminación, los materiales, la textura y otros elementos del entorno también juegan un papel importante en cómo experimentamos y nos conectamos emocionalmente con un espacio. Los colores planteados pueden afectar dónde dirigimos nuestra atención y cómo interpretamos visualmente un espacio. Colores vivos y contrastantes pueden captar nuestra atención más rápidamente y hacer que un objeto o área se destaque, lo que puede afectar nuestras emociones al hacernos sentir más alerta o enfocados en una tarea específica (Pickard et al, 2023).

En muchos casos, el simbolismo de los colores se relaciona con emociones específicas. El azul puede simbolizar tranquilidad y confianza; el verde, crecimiento y renovación; el rojo, pasión y peligro. Estas conexiones simbólicas pueden influir en cómo nos sentimos en presencia de estos colores (Gage, 1999).

1.1 La construcción de los entornos

Todos estos elementos no solo transforman la apariencia física de un lugar sino que también le dotan de personalidad y carácter. Esta personalidad puede evocar emociones específicas en quienes experimentan el espacio, haciéndolos sentir desde serenidad y alegría hasta melancolía y excitación. La inmersión en estos ambientes diseñados con intención amplifica aún más las sensaciones, permitiendo una conexión más profunda y significativa con el espacio (Rheingold, 1992).

La personalidad de un entorno se construye a través de la interacción de sus elementos visuales y espaciales. Por ejemplo:

- **Color:** Tiene el poder de influir directamente en el estado de ánimo y las emociones. Un paisaje dominado por verdes y azules puede transmitir tranquilidad y frescura, mientras que uno con tonalidades rojas y naranjas puede parecer más vibrante y energético (Heler, 2004; Itten, 1961).
- **Forma:** Las líneas rectas y estructuras geométricas pueden ofrecer una sensación de orden y formalidad, mientras que las curvas y formas orgánicas suelen ser percibidas como más naturales y relajantes.
- **Disposición espacial:** La manera en que los elementos se organizan en un espacio afecta cómo nos movemos y lo experimentamos. Espacios abiertos invitan a la exploración y la libertad, mientras que espacios más íntimos y cerrados pueden promover la reflexión o proporcionar confort (Hidayetoglu et al, 2012).

2 Metodología

Esta investigación aborda el estudio de técnicas de aplicación de color en entornos tridimensionales, específicamente mediante el diseño de diez "Skybox", que son entornos panorámicos envolventes utilizados en videojuegos y simulaciones VR para simular cielos o paisajes lejanos. Se llevó a cabo una clasificación meticulosa del uso del color en el diseño de cada uno de estos fondos, basada en una recopilación y análisis bibliográfico.

A través de la exploración de diversas combinaciones de colores, este estudio busca identificar los efectos emocionales comunes que estos colores evocan en los usuarios cuando son experimentados con la ayuda de gafas de realidad virtual (VR). El objetivo es definir un método sistemático y desarrollar una cartografía del color que sirva como guía para la construcción de entornos tridimensionales envolventes.

En la siguiente tabla se detallan los diez conceptos propuestos para el diseño de los entornos:

Table 1. Conceptos establecidos para la elaboración de los entornos

Paz y Tranquilidad	Azules claros y verdes suaves
	Un paisaje sereno de playa con el cielo despejado al amanecer.

Alegría y Energía	Amarillo brillante, naranja y toques de rojo
	Un parque soleado en pleno día con flores.
Misterio y Curiosidad	Morados oscuros, azules profundos y destellos de plata
	Una noche estrellada con un camino iluminado por la luz de la luna.
Relajación y Calma	Tonos de verde suave, azul pálido y blanco.
	Un jardín zen con piedras, agua corriente y follaje suave.
Pasión y Amor	Rojos intensos, rosas profundos y blancos
	Arquitectura floral en marmol blanco con suave iluminación
Frescura y Renovación	Verdes vivos, azul cielo y toques de amarillo claro.
	Un bosque en primavera con brotes nuevos y un arroyo cristalino.
Lujo y Elegancia	Oro, negro y toques de burdeos.
	Un salón opulento con muebles clásicos y adornos dorados.
Soledad y Reflexión	Grisés neutros, azules tenues y toques de plata
	Una playa desierta en un día nublado, con olas suaves
Fuerza	Rojo oscuro, negro y toques de gris metálico.
	Zona volcánica en erupción.
Creatividad e Inspiración	Una paleta vibrante de magenta, turquesa, y toques de oro.
	Tropical y animales exóticos

El proceso creativo se inició con la elaboración de imágenes en 2D, utilizando las combinaciones de colores especificadas en la tabla previamente mencionada. Durante esta fase, se prestó especial atención a la selección del color, la configuración de los elementos y la definición de sus texturas, asegurando que cada aspecto contribuyera armónicamente al concepto deseado.

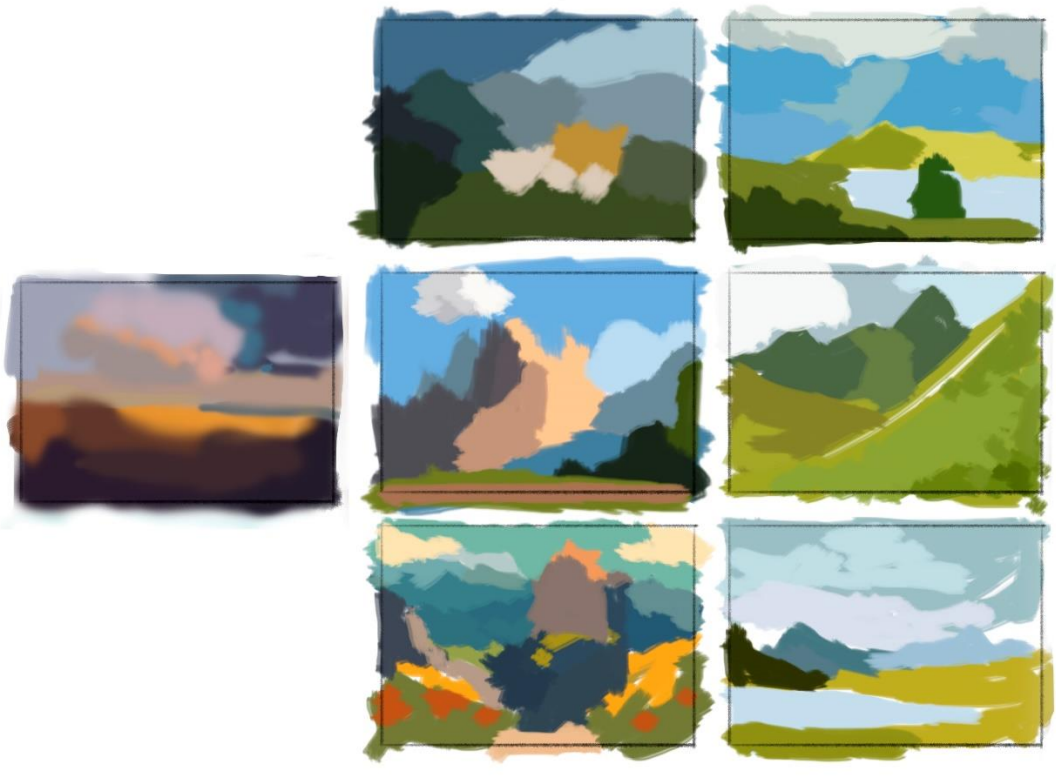


Fig. 1. Estudio de combinaciones de color

Posteriormente, estas creaciones se transfirieron al motor de desarrollo Unity, lo que permitió evaluar su correcta integración y el efecto de tileado, además de analizar la disposición espacial dentro de un entorno tridimensional interactivo. Este paso fue crucial para ajustar y perfeccionar la experiencia visual.

Finalmente, se procedió a la calibración de los sistemas de iluminación, buscando un equilibrio que realzara las cualidades de los colores y texturas, a la vez que se mantenía coherencia con la atmósfera y el tono emocional de cada entorno diseñado.

Los entornos fueron adaptados para su uso en realidad virtual (VR), con el fin de proporcionar una experiencia inmersiva y envolvente. Posteriormente, se llevó a cabo una encuesta detallada dirigida a los usuarios que experimentaron estos espacios virtuales. El objetivo principal de esta encuesta fue recopilar datos concretos y significativos sobre las emociones y reacciones de los espectadores ante los distintos entornos.

A través de un conjunto de preguntas cuidadosamente diseñadas, se buscó entender cómo cada combinación de color, textura y disposición espacial influía en el estado emocional de los participantes. La encuesta también incluyó ítems para evaluar la intensidad de las emociones experimentadas y cómo estos entornos virtuales podrían afectar el bienestar general de los usuarios.

3 Resultados

Actualmente, este estudio se encuentra en una etapa preliminar de exploración y análisis. Nuestro objetivo inmediato es ampliar el alcance y la diversidad de las encuestas administradas, con la finalidad de abarcar un espectro más amplio de participantes. Esta expansión nos permitirá recabar datos más robustos y representativos, esenciales para demostrar con mayor precisión los efectos emocionales que los entornos virtuales diseñados ejercen sobre los usuarios.

En las fases subsiguientes de investigación, nos proponemos emplear técnicas de análisis estadístico avanzadas para desentrañar las complejidades de las respuestas emocionales inducidas por la interacción con estos espacios. Este enfoque metodológico nos brindará la capacidad de identificar tendencias significativas, correlaciones y, posiblemente, causaciones entre los elementos de diseño de los entornos y las emociones experimentadas por los participantes.

Además, aspiramos a incorporar una evaluación cualitativa que complemente los hallazgos cuantitativos, mediante el análisis de comentarios y testimonios directos de los usuarios. Esto nos ofrecerá una visión más profunda y matizada de la experiencia emocional individual, enriqueciendo nuestra comprensión de cómo los aspectos visuales y espaciales de los entornos VR impactan en el bienestar emocional.

4 Conclusiones

La psicología del color, al entrelazar múltiples facetas de la percepción y la experiencia humana, ofrece una lente a través de la cual podemos profundizar en la comprensión de cómo y por qué los colores ejercen un poderoso impacto en nuestras emociones y comportamientos. Este estudio se adentra en dicha intersección, con el objetivo de trazar un marco referencial que no solo ilumine, sino también inspire la creación de proyectos artísticos y de diseño enriquecidos por esta comprensión.

Nuestra ambición es ir más allá de la mera confirmación de la influencia de los colores, aspirando a forjar un estándar de referencia que sea instrumental en el proceso creativo. Buscamos promover una apreciación más rica y fundamentada de cómo los colores modulan la experiencia emocional, especialmente en el contexto de los espacios virtuales. Este esfuerzo está dirigido a empoderar a los creadores con un conjunto de herramientas informadas por principios tanto psicológicos como estéticos, capacitándolos para diseñar entornos virtuales que sean no solo visualmente atrayentes, sino también emocionalmente resonantes.

Al concretar estos objetivos, anticipamos una doble realización: por un lado, la validación empírica de la influencia directa que los entornos virtuales ejercen sobre las emociones humanas; y por otro, la consolidación de una base teórica y práctica para el diseño de futuras experiencias de realidad virtual. Estas experiencias, cuidadosamente calibradas para inducir respuestas emocionales específicas, tienen el potencial de revolucionar múltiples campos —desde la terapia y la educación hasta el entretenimiento y más allá—, ampliando las fronteras de su aplicabilidad y enriqueciendo la interacción humana con la tecnología digital (Birren, 2016).

Así, nos encontramos en el umbral de una nueva era de diseño experiencial, donde el color emerge como un catalizador poderoso para la creación de entornos virtuales que no solo capturan la imaginación, sino que también movilizan el espectro completo de la experiencia humana.

5 Referencias bibliográficas

1. Labrador, E. (2020). El uso del color en los videojuegos. *Héroes de papel*. Studies
2. Pickard, C.; Knopf, D.; Guweiz; Fowkes, N. (2023). *Color y Luz*. Escuela de artistas. Espacio de Diseño.
3. Heler, E. (2004). *Psicología del color: cómo actúan los colores sobre los sentimientos y la razón*. Barcelona: Editorial Gustavo, SA.
4. Bellantoni, P. (2012). *If it's purple, someone's gonna die: the power of color in visual storytelling*. Routledge.
5. Birren, F. (2016). *Color psychology and color therapy; a factual study of the influence of color on human life*. Pickle Partners Publishing.
6. Itten, J. (1961). *The art of color: the subjective experience and objective rationale of color*.
- 7.
8. Gage, J. (1999). *Color and meaning: Art, science, and symbolism*. Univ of California Press.
9. Albers, J. (2013). *Interaction of color*. Yale University Press.
10. Krause, J. (2014). *Color for designers: ninety-five things you need to know when choosing and using colors for layouts and illustrations*. New Riders.
11. Fishwick, M. (2004). Emotional design: why we love (or hate) everyday things. *The Journal of American Culture*, 27(2), 234.
12. Valdez, P., & Mehrabian, A. (1994). Effects of color on emotions. *Journal of experimental psychology: General*, 123(4), 394.
13. Rheingold, H. (1992). *Virtual reality: [the revolutionary technology of computer-generated artificial worlds and how it promises to transform society]*. New York: Simon & Schuster.

14. Hidayetoglu, ML, Yildirim, K. y Akalin, A. (2012). Los efectos del color y la luz en la orientación interior y la evaluación del entorno percibido. *Revista de psicología ambiental* , 32 (1), 50-58.

Caso de Estudio: Aprendizaje por Refuerzo en StarCraft II

Álvaro Tena Tamayo¹[0009-0002-0265-2136], David García Guillén¹[0009-0002-9458-3309], Daniel García Algora¹[0009-0005-4839-6774] y Alberto Díaz Álvarez¹[0000-0002-4150-9052]

Universidad Politécnica de Madrid, C. de Alan Turing s/n, 28031 Madrid, España
{alvaro.tena.tamayo,d.guillen}@alumnos.upm.es
{d.galgora,alberto.diaz}@upm.es

Resumen Este estudio se centra en la implementación y evaluación de un agente inteligente sin políticas (*off-policy*) en el entorno en tiempo real de StarCraft II. Inspirados por el éxito de proyectos multiagente como AlphaStar, exploramos el potencial del uso de aprendizaje profundo combinado con el aprendizaje por refuerzo monoagente (DQN y DDQN) para mejorar las estrategias de juego sobre un único agente que parte de cero, sin datos previos. Desarrollamos un agente inteligente capaz de aprender de su experiencia en el juego y tomar decisiones en tiempo real. Nuestra metodología se centra en la implementación de un entorno de juego controlado y en la definición de políticas de acción y sistemas de recompensa. Además, incorporamos experiencia prioritaria con PER para mejorar el proceso de aprendizaje del agente. Tras realizar los experimentos, el rendimiento del agente *off-policy* logra, con poca estabilidad, vencer bots nativos del juego con el máximo nivel de dificultad. Los resultados de este estudio pueden servir para la creación de algoritmos de optimización de recursos o control de sistemas complejos.

Keywords: Aprendizaje por Refuerzo · Videojuegos · StarCraft II.

1. Introducción

Los videojuegos, entornos de decisión cerrados con una lógica consistente, han servido como banco de pruebas ideales para realizar pruebas en el campo de la Inteligencia Artificial (IA) [18] gracias a la diversidad y complejidad de los desafíos que presentan, que abarcan desde estrategias abstractas en juegos de tablero como el ajedrez [4] o el Go [16], hasta los entornos digitales más detallados. Estos no solo rivalizan con las capacidades humanas sino que, en muchos casos, las superan notablemente. En este escenario, StarCraft II, un juego de Estrategia en Tiempo Real (RTS) desarrollado por Blizzard Entertainment, destaca como un campo de pruebas excepcional [2; 12]. Su complejidad estratégica, la necesidad de toma de decisiones en tiempo real y el manejo de información incompleta presentan un más que interesante desafío.

StarCraft II, conocido por su dificultad, exige una mezcla de habilidades tanto de micro-gestión como de macro-gestión, que incluyen gestión de recursos,

exploración, expansión territorial y combate [13]. La victoria no solo depende de la habilidad para ejecutar estrategias complejas y adaptarse a las acciones del oponente, sino también de la gestión eficaz de múltiples unidades y operaciones en un entorno dinámico y, hasta cierto punto, desconocido. Estas características hacen de StarCraft II un entorno ideal para explorar y desarrollar técnicas avanzadas de Aprendizaje por Refuerzo (RL), un subcampo del Aprendizaje Automático (ML), donde el sistema «aprende» de su experiencia sobre el entorno, a diferencia de, por ejemplo, el aprendizaje supervisado que aprende a partir de la experiencia de un tercero.

El proyecto AlphaStar [3] de DeepMind, que en 2019 logró derrotar a jugadores humanos profesionales, representó un avance notable en la aplicación de IA a StarCraft II, combinando Aprendizaje Profundo (DL) con RL. Este implementó un enfoque que combinaba el aprendizaje profundo con el Aprendizaje por Refuerzo, demostrando una comprensión y una adaptabilidad extraordinarias en el juego [20].

Este estudio ¹ se inspira en el trabajo de AlphaStar para explorar aún más las capacidades y límites del RL en StarCraft II. Se busca avanzar en nuestra comprensión de cómo la IA monoagente, sin datos previos (*zero-shot*) puede navegar y triunfar en contextos de videojuegos complejos, no solo para mejorar las estrategias de juego, sino también para contribuir al desarrollo de algoritmos que puedan tener aplicaciones significativas en otros ámbitos, tales como la robótica, la simulación de escenarios estratégicos y la optimización de procesos. La importancia de este estudio reside en su capacidad para proporcionar nuevas perspectivas sobre la implementación y el rendimiento de técnicas de RL en contextos complejos, contribuyendo así a la expansión del área de investigación en IA aplicada a los juegos RTS.

2. Trabajos relacionados

El conocimiento y la experiencia de los seres humanos han sido utilizados con frecuencia para el entrenamiento de agentes inteligentes en juegos de tablero tradicionales en proyectos como AlphaGo [15], entrenado con repeticiones de partidas jugadas por expertos. Estos juegos están limitados a un contexto estático y a menudo a una visión completa del juego. StarCraft II introduce una capa más de dificultad por tratarse de un entorno dinámico con información espacial, exigiendo gran adaptatividad en tiempo real.

Algunos modelos con reglas predefinidas logran, mediante políticas con secuencias fijas de acciones, vencer con regularidad los bots de mayor dificultad del juego [11; 17], sin alcanzar el nivel competitivo humano. AlphaStar [21], uno de los avances más conocidos en el campo, fue el primer agente en alcanzar el rango máximo en el sistema competitivo del juego contra humanos, que también

¹ El código fuente, de acceso público, puede encontrarse en *GitHub*, en el repositorio <https://github.com/w-dan/SC2-RL/>. Consultado por última vez el 5 de mayo de 2024.

incorpora un componente de conocimiento humano previo, al entrenarse parcialmente con repeticiones de partidas existentes. Su arquitectura, con un total de 139 millones de parámetros [22], selecciona acciones basadas en las observaciones del juego y una estrategia por medio de una red neuronal profunda. Las observaciones complejas del juego se procesan mediante mecanismos de auto-atención y conexiones dispersas para integrar información espacial y no espacial. Se emplean redes LSTM para la secuencialidad y una política autoregresiva y red de punteros recurrente para manejar el espacio de acciones estructurado y combinatorio. Además, se combina el aprendizaje supervisado con el aprendizaje por refuerzo para entrenar inicialmente los parámetros del agente y luego optimizarlos mediante un algoritmo de refuerzo que maximiza la tasa de victoria. AlphaStar demuestra cómo el uso de algoritmos de autoimitación [10] ayudan a mejorar la estabilidad del aprendizaje. También introduce el entrenamiento por medio de ligas, una adaptación del auto-juego ficticio que evita ciclos y promueve un aprendizaje efectivo contra una mezcla no uniforme de oponentes, logrando así un agente de alto nivel competitivo.

Otros agentes, como SCC [22] han alcanzado nivel de juego profesional prescindiendo también de reglas predefinidas. Basado en las ideas principales de AlphaStar, utiliza técnicas propias para optimizar su desempeño. El agente utiliza el protocolo s2client de Blizzard y el entorno PySC2 de DeepMind para interactuar con el juego, y entre las diferencias clave con AlphaStar se encuentran el enfoque en una única raza y el uso un conjunto de datos más pequeño. Esto resulta en una arquitectura de red más eficiente, con 49 millones de parámetros, y el uso de Proximal Policy Optimization (PPO) en el entrenamiento por refuerzo. SCC también implementa un sistema de entrenamiento en liga con agentes principales, explotadores y de la liga, además de una selección de oponentes humanos para evaluación y ajuste del agente. En términos de arquitectura de red, SCC utiliza un modelo similar a AlphaStar con optimizaciones en el procesamiento de entradas, mecanismos de atención y estructuras condicionales para el espacio de acciones combinatorio. Su entrenamiento incluye etapas de aprendizaje por imitación con conjuntos de datos filtrados y optimizaciones en el tamaño de lote para un aprendizaje eficiente y estable.

3. Mecanismo del juego

Se trata de un juego de estrategia en tiempo real ambientado en un universo de ciencia ficción, que se centra en tres facciones distintas. Para este caso de estudio, se empleará la facción «Protoss», centrándose en partidas multijugador uno contra uno, donde el objetivo principal es derrotar al oponente, bien por destrucción de todas las estructuras principales, rendición, o exterminio de todas las unidades. Por unidades se entienden entidades controlables que, según su función, pueden atacar o construir estructuras. Por estructuras, se entienden edificaciones que desempeñan papeles como almacenamiento de recursos y entrenamiento de unidades. Cada jugador comienza la partida con un nexo y 12

constructores para desarrollar su estrategia. Para el caso concreto de los «Protoss», se ha entrenado el agente para emplear las siguientes unidades:

- **Sonda (*Probe*):** se dedica a construir estructuras y también se utiliza para tareas de exploración.
- **Rayo de vacío (*Voidray*):** unidad de ataque voladora.

El agente controla los recursos y desarrolla su estrategia en torno a las siguientes estructuras, ordenadas cronológicamente según el orden en que pueden ser construidas:

- **Nexo (*Nexus*):** alberga y produce unidades constructores, permitiendo almacenar suministro. El jugador puede construir nexos adicionales para expandir su base.
- **Pilón (*Pylon*):** aumenta el suministro disponible (inicialmente 15) en 8. Esto permite aumentar el tamaño del ejército y construir más estructuras. Los «Protoss» únicamente pueden edificar en un área alrededor de los pilones.
- **Asimilador (*Assimilator*):** permite recolectar un recurso concreto denominado *gas vespeno*, necesario para entrenar unidades, como el *Rayo de vacío*.
- **Portal (*Gateway*):** su función principal es invocar unidades terrestres. También permite la construcción del núcleo cibernético.
- **Núcleo cibernético (*Cybernetics Core*):** su construcción desbloquea acceso a estructuras avanzadas, como el portal estelar.
- **Portal estelar (*Stargate*):** depende del núcleo cibernético para poder ser construido. Permite invocar unidades voladoras, que atacan a distancia. Estas unidades no necesitan sortear los desniveles en el terreno.

Para el entrenamiento del modelo, se ha elegido un mapa concreto ², ya que las topologías de los mapas del juego son ligeramente variables. La figura 1 ofrece una representación visual del mapa donde se desarrolla el entrenamiento del agente. Puede apreciarse que las diferencias en altura se pueden sortear únicamente por medio de una serie de rampas distribuidas por el terreno. Además, el mapa está cubierto de niebla, que impide la visibilidad del terreno y se disipa, permitiendo visualizar las zonas sobre las que el jugador tiene control, bien por haber construido estructuras o por tener tropas situadas en el área.

4. Metodología

La metodología adoptada para este estudio se centra en la implementación y evaluación de un agente inteligente que desarrolla su trabajo tras haber aprendido del entorno definido por el juego StarCraft II mediante un esquema de RL. Para ello, se ha utilizado la biblioteca BurnySC2³, que facilita la interacción con

² Disponible en este repositorio, consultado por última vez en mayo de 2024.

³ Biblioteca BurnySC2 disponible en <https://github.com/BurnySc2/python-sc2>. Último acceso el 21 de marzo de 2024.



Figura 1. Topología del mapa elegido (Scorpion.1.01). Las ubicaciones de inicio de cada jugador se marcan con un símbolo blanco.

el entorno del juego de manera programática. Esta herramienta permite el acceso en tiempo real a información crítica del estado del juego, como la cantidad de recursos disponibles, la posición, puntos de vida y habilidades tanto de las tropas como de las estructuras.

4.1. Entorno definido

El entorno ha sido configurado para jugar partidas de StarCraft II bajo condiciones controladas, utilizando la Interfaz de Programación de Aplicaciones (API) suministrada por la biblioteca Burny SC2. Se ha optado por una estrategia de ejecución síncrona, donde el entorno se pausa hasta recibir retroalimentación del juego después de cada acción realizada, lo que conlleva una ejecución más pausada. Este enfoque facilita una evaluación detallada del efecto de cada acción sobre el estado del juego, a pesar de que tal evaluación no se produce de manera instantánea.

Como se muestra en el diagrama de componentes, representado en la figura 2, el entorno se encarga de iniciar el juego, enviar acciones, y recibir observaciones y recompensas. A pesar de la ejecución principal síncrona, se implementa un mecanismo asíncrono utilizando Redis para facilitar la comunicación entre el entorno y el juego a través de colas de mensajes (*publisher-subscriber*). Las observaciones del juego proporcionan datos detallados, incluyendo el estado del mapa en formato de imagen, recursos disponibles como minerales y gas vespeno, y el estado de los suministros del jugador. Además, el entorno administra la

finalización y reinicio de los episodios de juego (*game-steps*) y calcula las recompensas acumuladas. Se efectúa un preprocesamiento en las observaciones antes de su envío al agente, que incluye la redimensión y normalización de las imágenes del mapa y la conversión de las características adicionales en tensores escalares, optimizando así la interfaz de entrada para el proceso de aprendizaje del agente.

El comportamiento de ambos programas, tanto el síncrono como el asíncrono, se puede ver representado en el diagrama de secuencia de la figura 3. Este tipo de diagramas definen las acciones y su temporalidad, es decir, la «*agent_initialize*» no puede suceder antes de «*create_environment*». Como se puede observar en el diagrama, lo primero que se realiza es la creación del entorno y a continuación la inicialización del agente. Después de haber realizado un «*reset*» sobre el entorno para limpiar estados indeseados, se ejecuta la función «*run_game*», creando un nuevo proceso que será el encargado de ejecutar e instanciar una nueva partida de StarCraft II. El siguiente con la caja de texto arriba a la izquierda, es la representación de una ejecución sin parar hasta que se cumpla la condición de la caja, por lo tanto, en este caso, se estará ejecutando la recogida de datos de «*ArtanisBot*» y la publicación de mensajes en el «*redis*». Cuando exista al menos una observación en la cola, el entorno se encargará de recogerla y enviarla a la red neuronal, que escogerá una acción, la cual será publicada por el entorno. Lo mismo sucederá para la consumición de una acción, cuando exista en la cola una acción, el bot la consumirá y realizará las acciones necesarias. Cabe destacar que «*consume action*» y «*publish observation*» se realizan de forma asíncrona y por lo tanto, no tienen lugar bloqueos en el agente. De esta forma, la ejecución del entorno será constante y continua mientras que se espera la realización de acciones por parte de la red neuronal.

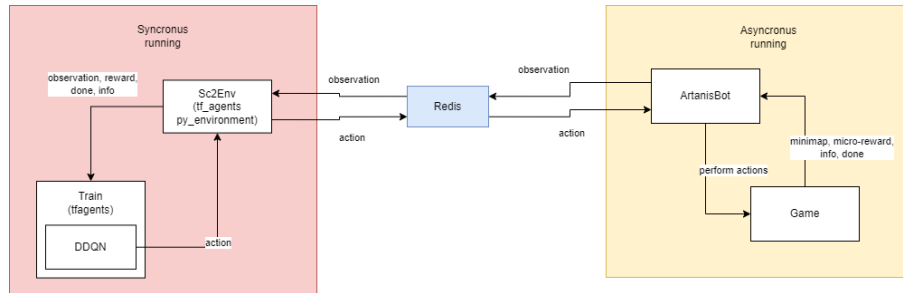


Figura 2. Diagrama de componentes del entorno y el agente

4.2. Política de acciones

La estrategia tras la política de acciones se ha diseñado con el objetivo de alcanzar la máxima granularidad, permitiendo al agente realizar decisiones complejas orientadas a maximizar las recompensas, sin requerir intervención directa

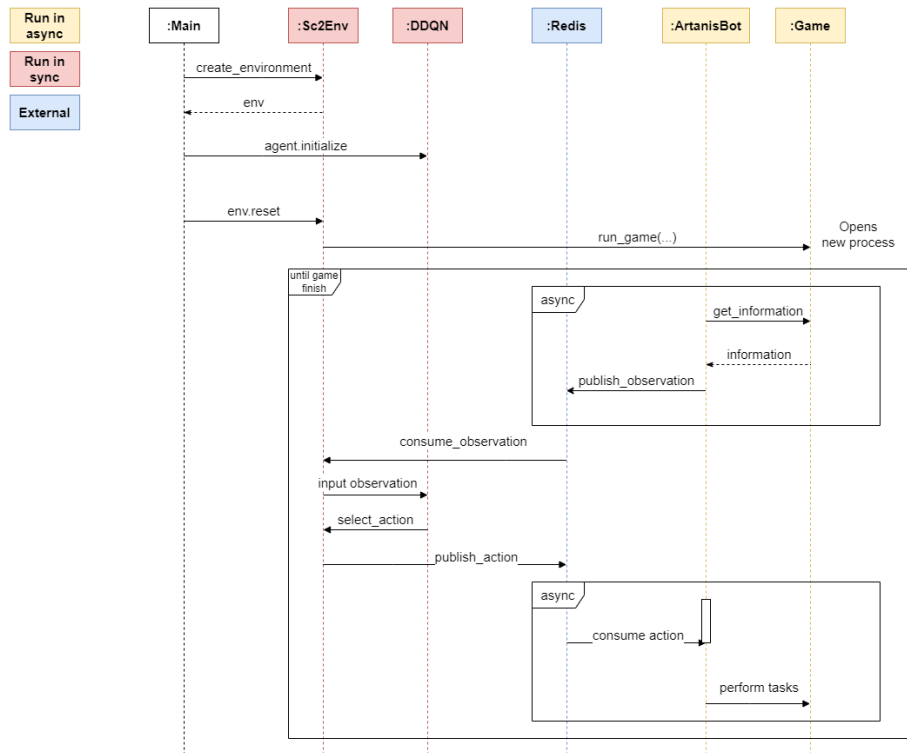


Figura 3. Diagrama de secuencia del entorno y el agente

o instrucciones predefinidas por parte de humanos. Esta aproximación evita conscientemente la definición de secuencias de acciones preestablecidas, promoviendo en cambio la generación de una mayor diversidad estratégica. Esto permite al agente desarrollar una amplia abanico de tácticas adaptativas, diseñadas para responder eficazmente a las condiciones dinámicas inherentes al entorno de juego.

La política de acciones implementada se centra en descomponer las posibles acciones en elementos lo más básicos posible, tales como «atacar» o «huir», buscando con ello un balance entre la simplicidad operativa y la capacidad de respuesta estratégica. Este enfoque se complementa con un sistema de recompensas y penalizaciones cuidadosamente calibrado: se imponen **penalizaciones** por la ejecución de acciones inviables, tales como intentos de construcción sin los recursos necesarios o por dilatar el tiempo de juego sin necesidad. Paralelamente se conceden **recompensas** por la realización de acciones críticas para el éxito en el juego, incluyendo la construcción de estructuras, la expansión territorial y la creación de unidades, así como por dedicar esfuerzos a maniobras ofensivas. Este esquema marcadamente agresivo hace necesaria la introducción de una penalización por la pérdida de estructuras o unidades, tratando de incentivar también la adopción de técnicas que favorezcan la preservación de recursos y una evaluación cuidadosa del combate, incluida la posibilidad de retirada táctica cuando la situación lo requiera para minimizar pérdidas.

El diseño del sistema de recompensas abarca tanto micro-recompensas, asignadas a acciones específicas, como macro-recompensas, orientadas hacia los objetivos a largo plazo del juego. Así, cada acción ejecutada por el agente es evaluada en tiempo real, asignándole una recompensa que refleja su impacto estratégico inmediato. Este modelo permite no solo incentivar la victoria como meta final de juego, sino también afinar la toma de decisiones del agente en cada paso del juego, fomentando un aprendizaje dinámico y la optimización continua de estrategias.

Esta diferenciación entre micro y macro-recompensa facilita un enfoque de aprendizaje matizado, en el que el agente no solo se centra en la victoria final, sino que también ajusta sus acciones paso a paso, evaluando una y otra vez la eficacia de sus estrategias basándose en el acumulado de recompensas. Este proceso asegura una adaptación continua al entorno del juego, permitiendo al agente desarrollar y refinar estrategias de juego altamente heterogéneas.

La tabla 1 representa la recompensa que se concede al agente por realizar una acción de construcción. Por otro lado, la tabla 2 contiene información sobre las recompensas relativas a la fabricación de una tropa. Estas tablas ejemplifican cómo se asignan valores concretos a diversas acciones dentro del juego, subrayando la importancia de una estructura de recompensas detallada para guiar el aprendizaje y la adaptación del agente.

La asignación de recompensas para construcciones y unidades refleja la intención de equilibrar el fomento de la expansión voraz a lo largo del campo de batalla y el fortalecimiento estratégico por un lado, y la necesidad de gestionar eficazmente los recursos y las oportunidades tácticas por el otro.

Tabla 1. Recompensas de Construcción

Acción	Recompensa
NEXUS	5
STARGATE	2
CYBERNETICSCORE	1.5
GATEWAY	1
PYLON	0.2
ASSIMILATOR_NOT_BUILT	-0.2

Tabla 2. Recompensas de Tropas

Acción	Recompensa
PROBE_SCOUTING_STATIC	0.2
VOIDRAY_TRAINED	0.1
MAX_PROBES	-0.05
PROBE_DESTROY_SCOUTING	-0.15
NO_VOIDRAY_ATTACK	-0.2

4.3. Gestión de jugadas con *Prioritized Experience Replay*

Una de las técnicas clave en nuestro enfoque para el entrenamiento de nuestro agente es la implementación de *Prioritized Experience Replay* (PER) [14]. Esta técnica se basa en la idea de que ciertas experiencias o jugadas durante el entrenamiento aportan más al aprendizaje que otras. En lugar de seleccionar aleatoriamente experiencias pasadas para revisar y aprender de ellas (como ocurre en el *experience replay* tradicional [8]), PER asigna una prioridad a cada experiencia basada en la magnitud de su «sorpresa» o error de predicción. Dicho de otro modo, las experiencias que resultaron un desvío grande entre lo esperado y lo ocurrido son las que reciben una mayor prioridad, ya que indican oportunidades significativas para el aprendizaje. Dar prioridad a estas experiencias «sorprendentes» permite corregir errores significativos de manera más eficiente, promover una exploración más profunda de estados críticos y estimular la reevaluación de modelos internos, siendo esencial para un aprendizaje adaptativo y profundo. Este enfoque no solo aumenta la eficacia del aprendizaje, sino que también optimiza la política de aprendizaje del agente, lo que permite ajustar su comportamiento de manera más precisa para maximizar las recompensas futuras [14].

La prioridad puede definirse de la siguiente manera:

$$P(i) = \frac{p_i^\alpha}{\sum_k p_k^\alpha} \quad (1)$$

donde $p_i > 0$ representa la prioridad de la transición i y α determina el grado de prioridad, siendo $\alpha = 0$ el caso uniforme.

En la práctica, esto se traduce en que el agente revisa y aprende de manera preferente de aquellas experiencias donde sus expectativas estaban más alejadas de los resultados reales, facilitando una corrección de curso más rápida [14].

Para implementar el PER, se utiliza una estructura de datos denominada «cola de prioridad» para almacenar experiencias. Cada vez que el agente realiza una jugada, esta se guarda en la cola con una prioridad asignada en función de su error de predicción. Posteriormente, durante las fases de entrenamiento, se extraen experiencias de esta cola de manera sesgada hacia aquellas con mayor prioridad, asegurando así que el agente se enfoque en los aspectos más críticos de su aprendizaje.

La aplicación de este algoritmo en el entorno de juego de StarCraft II permite al agente centrarse en las decisiones que durante la partida necesitan un mayor ajuste. Por ejemplo, si el agente falla repetidamente al intentar ejecutar una táctica particular contra un oponente, o intentando desplegar un edificio sin los recursos necesarios, las experiencias relacionadas con estos intentos se considerarán muy prioritarias durante la fase de aprendizaje.

4.4. Optimización del agente

Inicialmente, se optó por implementar una estrategia basada en Redes Q Profundas (Deep Q-Network (DQN)) [8], un enfoque que implica la actualización iterativa de los estados y acciones mediante el uso de redes neuronales. Sin embargo, surgieron problemas debido a la complejidad al explorar un espacio de estados tan extenso como el de StarCraft II. Esta complejidad se traducía en una convergencia notablemente lenta hacia la política óptima de toma de decisiones.

Para abordar esta problemática, se adoptó una metodología que mejora las redes DQN, las Redes Q Profundas de Doble Estimación (Double Deep Q-Networks (DDQN)) [5]. Este nuevo enfoque se distingue por la incorporación de una arquitectura de doble red neuronal, donde una red se encarga de la selección de acciones y la otra evalúa el valor de estas acciones seleccionadas [23]. Esta separación funcional contribuye a mitigar los sesgos asociados con la sobreestimación de los valores Q, un problema dominante en las DQN estándar [6]. El uso de DDQN promueve una convergencia más estable hacia la política óptima [19], evitando las evaluaciones excesivamente optimistas que suelen observarse en las DQN. Además, estudios recientes resaltan que las DDQN tienden a generar estrategias de toma de decisiones más conservadoras y, por ende, políticas de optimización más robustas y realistas [7].

Hay que destacar que los algoritmos DQN y DDQN, pertenecen a la categoría de métodos de aprendizaje *off-policy*, que buscan optimizar la política de comportamiento de manera indirecta, a través de la evaluación de la política objetivo [9], lo cual contrasta con los métodos *on-policy*, que ajustan directamente la política que está siendo aprendida a partir de la experiencia recabada [1].

4.5. Pruebas realizadas y resultados

El desarrollo de las pruebas se realizó de forma incremental, empezando con las pruebas de un agente sencillo, hasta llegar a uno con un espacio más complejo de observaciones y de acciones. Para que el agente pueda conocer el estado de la partida, se decidió generar un mapa de píxeles, donde cada píxel representa

el estado de esa zona en el mapa, consiguiendo así una fiel representación del mini-mapa del juego, pero con información más específica para el agente, de esta forma se consigue mediante un preprocesamiento, la generación de una imagen con píxeles de distintos colores, donde cada color es identificado con un objeto, por ejemplo, en la paleta, se usa el naranja para identificar una estructura enemiga y el rojo para identificar una unidad enemiga. Las acciones siguen la misma línea en los experimentos posteriores, encontrándose predefinidas y por lo tanto, limitando el alcance a una macro-gestión superficial por parte del agente; esto es, según las acciones posibles⁴:

- **Expandir base:** Si quedan menos de 4 suministros almacenados se construye un *Pylon*; en caso contrario, para cada *Nexus* ocioso se entrena un nuevo trabajador, y para cada *Vespene Geyser*, se construye un *Assimilator*; si nada de lo anterior se ha podido realizar con éxito, se construye un nuevo *Nexus* en otra posible ubicación del mapa.
- **Construir portal estelar:** Para cada nexo construido se comprueba la cercanía a un *Gateway* y en caso de no existir se construye, se realiza lo mismo con el «*Cybernetics Core*» y, por último, con la *Stargate*, la cual nos permite entrenar nuestras tropas.
- **Entrenar *Voidray*:** Por cada *Stargate* ocioso se entrena una *Voidray*.
- **Explorar:** Si ha pasado suficiente tiempo, se escoge aleatoriamente un trabajador (primero entre los ociosos, si no, entre todos) y se envía contra la base inicial enemiga.
- **Atacar:** Para cada *Voidray* ociosa se manda una orden de ataque contra el primer objetivo que se cumpla, siendo estos atacar una unidad enemiga que se encuentre al alcance de la *Voidray*, una estructura enemiga dentro del alcance, cualquier unidad enemiga visible en el mapa, cualquier estructura enemiga visible en el mapa o, por último, la base inicial enemiga.
- **Retirar *Voidray*:** Todas las unidades *Voidray* reciben la orden de volver a la base inicial.

Este primer agente implementa un búfer de repetición de experiencias uniforme, y tras el entrenamiento se obtuvieron los resultados expuestos en la figura 4, donde puede verse que el agente es derrotado en casi todas las partidas sumando un 97.4%, a excepción de un 2.3% que consigue ganar y un 0.4% de tasa de empate. Debido a la naturaleza predefinida de las acciones, este agente no era capaz de desarrollar una política adecuada en un entorno tan dinámico e incertidumbre, concluyendo en la necesidad de atomizar las acciones.

En el segundo experimento y en los posteriores, el conjunto de acciones posibles se verá ampliado, manteniendo intacta la funcionalidad del agente. Es decir, el agente no adquirirá la capacidad de ejecutar acciones con una funcionalidad nueva; en cambio, podrá llevar a cabo las mismas acciones de antes de manera desglosada sin un orden preestablecido tal y como se detalla en la sección 4.2. Esto resultará en la mencionada expansión del espacio de acciones.

⁴ Por simplicidad, se omite la verificación previa de recursos y la revisión de la cola cuando se menciona la construcción de un edificio o el entrenamiento de una unidad.

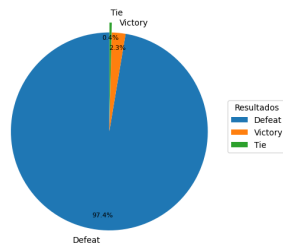


Figura 4. Resultados agente 1

Así, el agente puede decidir si construir más *Pylon* o más *Gateway* en determinados momentos de la partida, pudiendo transicionar a estados no tan óptimos; a su vez, este desglose de las acciones conduce a una posible explotación por parte del agente del sistema de recompensa, logrando maximizarla aún cuando no consigue el objetivo principal de ganar la partida. Por ello, el sistema de recompensa será redefinido al nuevo espacio de acciones, agregando un componente de penalización que aplique en aquellas acciones que conlleven una utilización indebida de la mecánica, como por ejemplo construir únicamente *Pylon* independientemente del excedente de suministros del que disponga el agente, intentando explotar la recompensa otorgada por dicha construcción al aumentar los suministros disponibles; o la prematura expansión de *Nexus* antes de tener los recursos necesarios para su construcción.

El endurecimiento del sistema de recompensa conlleva también un riesgo de penalización al agente sin la posibilidad de que dicha penalización pueda ser esperada por el agente debido a una mala representación del estado de observaciones. Para evitar este riesgo, se decide dotar al mapa de píxeles de una mayor representabilidad del juego, agregando diferenciación entre las estructuras y tropas propias y en una menor medida sobre las estructuras y tropas enemigas, así como la diferenciación de si un píxel del mapa en concreto es construible o transitado por el jugador. En adición al mapa de píxeles mejorado, el espacio de observaciones se construye también con información sobre los recursos disponibles por el jugador, como lo son los minerales, el gas vespeno o los suministros.

El entrenamiento de este segundo agente muestra un incremento en la tasa de victorias hasta un 10% como puede verse en la figura 5, así como una evolución hacia el desarrollo de estrategias lógicas, evidenciando un entendimiento sobre la importancia de construir distintas estructuras. Esto ocurre sin que exista una indicación previa sobre el orden óptimo de construcción, más allá de la recompensa y el castigo recibidos por edificarlas en momentos en que sus recursos lo permiten. Este comportamiento sugiere una adaptación y aprendizaje significativo por parte del agente, lo cual le permite optimizar sus decisiones estratégicas en el contexto del juego.

El objetivo del último experimento es mejorar los resultados del experimento previo, basándose en la descripción del funcionamiento del *Prioritized Experience Replay* expuesto en la sección 4.3. En este experimento, cambiamos la forma

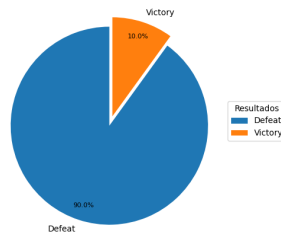


Figura 5. Resultados agente 2

en que el agente selecciona y revisa sus experiencias pasadas durante su entrenamiento. En los experimentos anteriores, se muestrean experiencias de forma uniforme desde el búfer de memoria. Con la implementación de PER, sin embargo, las experiencias se ponderan según el error de predicción, lo que significa que las decisiones donde el agente cometió errores graves tienen más probabilidades de ser revisadas. Se espera que esta técnica acelere el aprendizaje al centrar el entrenamiento en los puntos donde el agente puede recibir las lecciones más críticas.

El algoritmo de aprendizaje del agente se ajustó para integrar un sistema de pesos dinámico que recalcula las prioridades de las experiencias almacenadas en el búfer cada vez que se actualiza la red de valor del agente, a fin de implementar esta técnica. Basado en el error de predicción de cada experiencia, este sistema de ponderación se centra en corregir las estimaciones menos precisas para mejorar la eficiencia del aprendizaje del agente⁵.

A pesar de estas mejoras, los resultados preliminares, como se observa en la figura 6, muestran apenas una tasa de victorias del 2%. El resultado indica que, aunque PER puede ayudar a ajustar las expectativas de recompensa del agente, la complejidad del espacio de acción y las dinámicas del juego pueden necesitar cambios adicionales en su implementación o una revisión de las métricas utilizadas para medir el éxito del agente.

Los resultados y técnicas discutidas en este estudio se pueden aplicar en contextos externos al ámbito de los videojuegos. El agente demuestra su potencial en otros ámbitos críticos al mejorar la toma de decisiones y la autonomía. Como ejemplo, en el ámbito de la robótica, estrategias similares podrían permitir que los robots funcionasen con mayor eficiencia en entornos dinámicos o desconocidos. Simular escenarios estratégicos en contextos como logística y gestión de emergencias podría tener un gran impacto en la planificación y respuesta al poder mejorar la capacidad de adaptación y maximizar los objetivos bajo restricciones cambiantes. Por último, la optimización de procesos podría mejorar con un enfoque más detallado y adaptativo, parecido a cómo el agente aprende a optimizar su uso de recursos para construir los edificios en el momento correcto.

⁵ Implementación PER disponible en <https://gist.github.com/Kenneth-Schroeder/6dbdd4e165331164e0d9dcc2355698e2>. Último acceso el 06 de mayo de 2024.

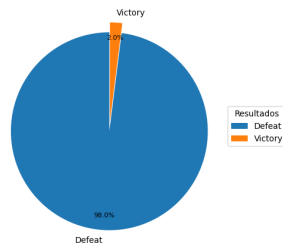


Figura 6. Resultados agente 3

5. Conclusiones

El rendimiento del agente *off-policy*, al enfrentarse contra un bot en la máxima dificultad, presenta dificultades en la adaptación de un amplio espacio de estados, sometido a cambios constantes que conllevan una incertidumbre en el agente.

Los problemas de los agentes *off-policy* se pueden resolver usando técnicas de RL más avanzadas que emplean políticas, u *on-policy*, como PPO, consiguiendo calcular la actualización de cada paso, minimizando la función de coste, y garantizando así una menor desviación, mejorando la estabilidad en el entrenamiento y la incertidumbre del agente.

La complejidad de este juego demuestra que un modelo entrenado con RL puede ser capaz de realizar tareas complejas y poder responder ante nuevas circunstancias mejor que otros modelos, por lo que su aplicación, en entornos complejos como gestión de tráfico u optimización de recursos, puede resultar de beneficio.

Referencias

- [1] Andrychowicz, M., Raichuk, A., Stańczyk, P., Orsini, M., Girgin, S., Mariner, R., Hussenot, L., Geist, M., Pietquin, O., Michalski, M., et al.: What matters in on-policy reinforcement learning? a large-scale empirical study. arXiv preprint arXiv:2006.05990 (2020)
- [2] Churchill, D., Certicky, M.: The current state of starcraft ai competitions and bots (2017)
- [3] DeepMind: Alphastar: Mastering the real-time strategy game starcraft ii (2019), [bluehttps://deepmind.google/discover/blog/alphastar-mastering-the-real-time-strategy-game-starcraft-ii/](https://deepmind.google/discover/blog/alphastar-mastering-the-real-time-strategy-game-starcraft-ii/)
- [4] Ferrari, V., Didierjean, A., Marmèche, E.: Effect of expertise acquisition on strategic perception: The example of chess. *Quarterly Journal of Experimental Psychology* **61**(8), 1265–1280 (2008). <https://doi.org/10.1080/17470210701503344>, [bluehttps://doi.org/10.1080/17470210701503344](https://doi.org/10.1080/17470210701503344), PMID: 18938781
- [5] van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double q-learning (2015)
- [6] Lan, Q., Pan, Y., Fyshe, A., White, M.: Maxmin q-learning: Controlling the estimation bias of q-learning. arXiv preprint arXiv:2002.06487 (2020)
- [7] Li, S.E.: Deep reinforcement learning. In: Reinforcement learning for sequential decision and optimal control, pp. 365–402. Springer (2023)
- [8] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.: Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602 (2013)
- [9] Munos, R., Stepleton, T., Harutyunyan, A., Bellemare, M.: Safe and efficient off-policy reinforcement learning. *Advances in neural information processing systems* **29** (2016)
- [10] Oh, J., Guo, Y., Singh, S., Lee, H.: Self-imitation learning. In: International conference on machine learning. pp. 3878–3887. PMLR (2018)
- [11] Pang, Z.J., Liu, R.Z., Meng, Z.Y., Zhang, Y., Yu, Y., Lu, T.: On reinforcement learning for full-length game of starcraft. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 4691–4698 (2019)
- [12] Risi, S., Preuss, M.: From chess and atari to starcraft and beyond: How game ai is driving the world of ai. *KI-Künstliche Intelligenz* **34**(1), 7–17 (2020)
- [13] Samvelyan, M., Rashid, T., De Witt, C.S., Farquhar, G., Nardelli, N., Rudner, T.G., Hung, C.M., Torr, P.H., Foerster, J., Whiteson, S.: The starcraft multi-agent challenge. arXiv preprint arXiv:1902.04043 (2019)
- [14] Schaul, T., Quan, J., Antonoglou, I., Silver, D.: Prioritized experience replay. arXiv preprint arXiv:1511.05952 (2015)
- [15] Silver, D., Huang, A., Maddison, C., Guez, A., Sifre, L., Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D.: Mastering the game of go with deep neural networks and tree search. *Nature* **529**, 484–489 (01 2016). <https://doi.org/10.1038/nature16961>

- [16] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al.: Mastering the game of go without human knowledge. *nature* **550**(7676), 354–359 (2017)
- [17] Sun, P., Sun, X., Han, L., Xiong, J., Wang, Q., Li, B., Zheng, Y., Liu, J., Liu, Y., Liu, H., et al.: Tstarbots: Defeating the cheating level builtin ai in starcraft ii in the full game. arXiv preprint arXiv:1809.07193 (2018)
- [18] Torrado, R.R., Bontrager, P., Togelius, J., Liu, J., Perez-Liebana, D.: Deep reinforcement learning for general video game ai. In: 2018 IEEE Conference on Computational Intelligence and Games (CIG). pp. 1–8. IEEE (2018)
- [19] Van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double q-learning. In: Proceedings of the AAAI conference on artificial intelligence. vol. 30 (2016)
- [20] Vinyals, O.: Starcraft ii: A new challenge for reinforcement learning (2017)
- [21] Vinyals, O., Babuschkin, I., Czarnecki, W.M., Mathieu, M., Dudzik, A., Chung, J., Choi, D.H., Powell, R., Ewalds, T., Georgiev, P., et al.: Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature* **575**(7782), 350–354 (2019)
- [22] Wang, X., Song, J., Qi, P., Peng, P., Tang, Z., Zhang, W., Li, W., Pi, X., He, J., Gao, C., et al.: Scc: An efficient deep reinforcement learning agent mastering the game of starcraft ii. In: International conference on machine learning. pp. 10905–10915. PMLR (2021)
- [23] Zhang, R., Xiong, K., Lu, Y., Gao, B., Fan, P., Letaief, K.B.: Joint coordinated beamforming and power splitting ratio optimization in mu-miso swipt-enabled hetnets: A multi-agent ddqn-based approach. *IEEE Journal on Selected Areas in Communications* **40**(2), 677–693 (2021)

Salud mental y biofeedback en los Serious Games

Inés Bueno Pascual^[0009-0009-3171-4309]

Universidad Complutense de Madrid, 28223 Madrid, España
inebueno@ucm.es

Abstract. Los Serious Games ofrecen una solución innovadora para hacer frente a los elevados índices de morbilidad de trastornos emocionales. Cada vez hay más aplicaciones informáticas que trabajan la regulación emocional y videojuegos que ofrecen una experiencia placentera. Pero, ¿qué pasaría si se unieran las dos? ¿Cómo sería crear un videojuego que vaya aprendiendo del/la usuario/a mediante la interconexión de nodos generando una experiencia única para cada uno?

Keywords: Salud mental, biofeedback, Serious Games

1 Introducción

Los Serious Games, entendidos como juegos diseñados con un propósito formativo, ofrecen una solución innovadora para hacer frente a los elevados índices de morbilidad de trastornos emocionales [1]. Desde el ámbito de la salud mental, se encuentran enmarcados dentro de la Tecnología Positiva [2] con el objetivo de generar cambios positivos con efectos transferibles al mundo real [3].

Un juego serio permite un control considerable respecto a la creación de experiencias atractivas por medio de dinámicas interactivas que reduzcan la tasa de abandono de las intervenciones digitales [4] y aumente su compromiso [5].

El espacio de juego permite la generación de un espacio seguro en el que el individuo puede explorar cometiendo errores con seguridad y da la libertad de actuar de una manera diferente a como lo haría en la vida real, lo cual lo convierte en un escenario ideal para el aprendizaje [6].

La gran parte de los hallazgos sugieren que los juegos serios para aumentar la salud mental son eficaces, pero son grandes las diferencias en la investigación entre las variaciones de resultados y las medidas que se han utilizado para medir el impacto [7,8]. En este respecto, un método recurrente para evaluar el impacto es mediante el uso de cuestionarios externos antes y después del uso del videojuego. A pesar de que esta técnica está ampliamente aceptada en el campo de la salud se plantea que no proporciona información precisa sobre el estado de bienestar del usuario [9] por el estrés que puede generar la propia presencia del entrevistador y por el propio entorno del laboratorio. En su lugar, los Serious Games pueden proporcionar una alternativa más ecológica que registre variables de bienestar del sujeto de una manera más discreta [10].

Concretamente, los Serious Games pueden ser una herramienta atractiva para la población joven. La prevención de la ansiedad y la depresión en jóvenes por medio del

desarrollo de habilidades de regulación emocional es crucial para la salud mental. Es de especial importancia que las intervenciones dirigidas a este fin sean adaptadas a los grupos objetivo y al contexto, así como a las características individuales que puedan influir en el efecto del tratamiento. Con un enfoque adecuado, estos programas podrían ser escalables a nivel mundial y ofrecer una estrategia proactiva para abordar los desafíos de la salud mental en la juventud [11].

2 Serious games y biofeedback

Una revisión sistémica [12] se centró en investigar la evidencia científica sobre los efectos del uso del biofeedback en intervenciones para mejorar la regulación emocional. En concreto, se centró en como los Serious Games afectan estos a la toma de decisiones. Se analizó un total de 16 estudios en los que se encontraron resultados positivos en cuanto al aumento de la conciencia emocional y el desarrollo de estrategias de regulación emocional, con una consecuencia directa en la mejora de toma de decisiones. En esta revisión se plantea el biofeedback como una doble vía: una directa por medio de contenido visual y otra indirecta por medio del gameplay. Además, plantea los juegos serios presentan una ventaja respecto a los métodos tradicionales de entrenamiento en regulación emocional, gracias a la información continua que facilita sobre el estado interno del individuo, lo que permite que este sea más consciente de su estado emocional y pueda aplicar estrategias adecuadas de afrontamiento. En añadido, señala que los juegos serios personalizados apoyados en retroalimentación fisiológica que refuerza un sistema de recompensa positivo pueden superar aquellas limitaciones que se enfrentan los ambientes de entrenamiento más controlados como puede ser la falta de motivación.

3 Del laboratorio al mundo real

Son cada vez más los estudios que muestran resultados positivos en la utilización de aplicaciones móviles en la mejora de síntomas de ansiedad y depresión. Además, en estudios de seguimiento se observa que estos resultados pueden mantenerse [13, 14].

Para hacer frente a la adaptación y adherencia más allá de los ensayos clínicos se formó una Colaboración internacional para maximizar el impacto de la terapia electrónica y los juegos serios (COMETS). Entre las medidas propuestas Fleming y colaboradores [4] recogen las siguientes:

- Enfoque más centrado en el usuario. Propone un diseño centrado en el usuario y una mayor individualización del programa.
- Mayor énfasis en el aumento de la participación del usuario por medio de procesos como el gaming, la gamificación, la telepresencia y la tecnología persuasiva.
- Fomento de la colaboración entre diferentes sectores, regiones geográficas y grupos de interés.
- Realización de pruebas e implementación rápidas incluyendo la medición de alcance, el compromiso y la eficiencia.

Atendiendo a estas medidas es necesario poner el foco en el individuo para que el videojuego sea lo suficientemente atractivo a la vez que funcional para que su uso se incorpore a la vida real, más allá de su aplicación en pruebas en el laboratorio. Por esta razón, se propone que es fundamental la colaboración directa entre psicólogos y expertos en videojuegos.

3.1 Una propuesta integradora

Cada vez hay más aplicaciones informáticas que trabajan la regulación emocional y videojuegos que ofrecen una experiencia placentera. Pero, ¿qué pasaría si se unieran las dos? ¿Cómo sería crear un videojuego que vaya aprendiendo del/la usuario/a mediante la interconexión de nodos generando una experiencia única para cada uno? Se plantean estas preguntas como un posible camino a explorar en la mejora de la prevención de la salud mental.

Para poder enfrentar este reto es necesario emplear una tecnología que esté al alcance de la población general. Esta oportunidad surge por el potencial de la tecnología para hacer frente a la necesidad de acceso a la atención psicológica para un gran número de personas. Se enfoca en la disminución del estigma, el desarrollo de la capacidad de investigación y el tratamiento del sistema de salud mental, implementación de programas de prevención y el establecimiento de una salud pública sostenible basada en intervenciones centradas en la evidencia [15]. En este sentido, el dispositivo más viable podría ser el teléfono móvil.

Otro reto a enfrentar es como conseguir que exista una retroalimentación entre la persona y el juego. Una solución eficaz podría ser la conexión a una pulsera de actividad en la que se incorporarán los datos del registro fisiológico del/la usuario/a por medio de técnicas de biofeedback, junto con datos de autoinforme sobre su estado emocional. Es esta conexión de datos lo que generará una experiencia única para cada persona.

Más allá de la pulsera de actividad, el dispositivo móvil cuenta con una serie de posibilidades que pueden generar ese feedback entre la persona y el juego. Por ejemplo, mediante la cámara podría registrarse las expresiones faciales incorporándolas a las dinámicas de juego, así como el uso del micrófono. Otra forma de poder conseguir esta retroalimentación podría ser mediante la incorporación del embodiment por medio del acceso a los sensores del giroscopio, generando dinámicas del videojuego tendrán acceso a la posición y movimiento del dispositivo.

Son muchos los caminos por explorar, así como las posibilidades de actuación en cada uno de ellos, pero si queremos apostar por una propuesta integradora basada en la evidencia científica que además sea usable fuera del laboratorio y, lo más importante, que la gente de verdad lo quiera usar, es necesario trabajar codo con codo en un equipo interdisciplinar dando valor y espacio a todas las disciplinas que contribuyan en el proceso.

Referencias

1. Díaz-García, A., González-Robles, A., García-Palacios, A., Fernández-Álvarez, J., Castilla, D., Bretón, J.M., Baños, R.M., Quero, S., Botella, C.: Negative and positive affect regulation

- in a transdiagnostic internet-based protocol for emotional disorders: Randomized controlled trial. *J. Med. Internet Res.* 23, e21335 (2021). <https://doi.org/10.2196/21335>.
2. Botella, C., Riva, G., Gaggioli, A., Wiederhold, B.K., Alcaniz, M., Baños, R.M.: The present and future of positive technologies. *Cyberpsychol. Behav. Soc. Netw.* 15, 78–84 (2012). <https://doi.org/10.1089/cyber.2011.0140>.
 3. Barnes, S., Chandler, T., Granda-Salazar, M.: Help over harm: practical and ethical considerations for the evaluation and deployment of therapeutic games. *Ment. Health Soc. Incl.* 27, 133–139 (2023). <https://doi.org/10.1108/mhsi-11-2022-0080>.
 4. Fleming, T.M., de Beurs, D., Khazaal, Y., Gaggioli, A., Riva, G., Botella, C., Baños, R.M., Aschieri, F., Bavin, L.M., Kleiboer, A., Merry, S., Lau, H.M., Riper, H.: Maximizing the impact of e-therapy and serious gaming: Time for a paradigm shift. *Front. Psychiatry.* 7, (2016). <https://doi.org/10.3389/fpsy.2016.00065>.
 5. Ghosh, A., Agnihotri, J., Bhalotia, S., Sati, B.K., Agarwal, L., Akash, Tandon, S., Meena, K., Raj, S., Azad, Y., Gupta, S., Gupta, N.: Serious games based on cognitive bias modification and learned helplessness paradigms for the treatment of depression: Design and acceptability study. *JMIR Serious Games.* 11, e37105 (2023). <https://doi.org/10.2196/37105>.
 6. Günzel, S., Liebe, M., Mersch, D.: Conference proceedings of The Philosophy of computer games 2008. Universitätsverlag Potsdam, Potsdam, Alemania (2009).
 7. Lau, H.M., Smit, J.H., Fleming, T.M., Riper, H.: Serious games for mental health: Are they accessible, feasible, and effective? A systematic review and meta-analysis. *Front. Psychiatry.* 7, (2017). <https://doi.org/10.3389/fpsy.2016.00209>.
 8. Nguyen, T.T.H., Ishmatova, D., Tapanainen, T., Liukkonen, T.N., Katajapuu, N., Makila, T., Luimula, M.: Impact of serious games on health and well-being of elderly: A systematic review. En: *Hawaii International Conference on System Sciences 2017 (HICSS-50)* (2017).
 9. Bellotti, F., Kapralos, B., Lee, K., Moreno-Ger, P., Berta, R.: Assessment in and of serious games: An overview. *Adv. Hum.-Comput. Interact.* 2013, 1–11 (2013). <https://doi.org/10.1155/2013/136864>.
 10. Friehs, M.A., Dechant, M., Vedress, S., Frings, C., Mandryk, R.L.: Effective gamification of the stop-signal task: Two controlled laboratory experiments. *JMIR Serious Games.* 8, e17810 (2020). <https://doi.org/10.2196/17810>.
 11. <https://policycommons.net/artifacts/1850367/what-science-has-shown-can-help-young-people-with-anxiety-and-depression/2597275/>, Consultado 2024/03/21.
 12. Jerčić, P., Sundstedt, V.: Practicing emotion-regulation through biofeedback on the decision-making performance in the context of serious games: A systematic review. *Entertain. Comput.* 29, 75–86 (2019). <https://doi.org/10.1016/j.entcom.2019.01.001>.
 13. Zhang, M., Ying, J., Song, G., Fung, D.S.S., Smith, H.: Gamified cognitive bias modification interventions for psychiatric disorders: Review. *JMIR Ment. Health.* 5, e11640 (2018). <https://doi.org/10.2196/11640>.
 14. Miralles, I., Granell, C., Díaz-Sanahuja, L., Van Woensel, W., Bretón-López, J., Mira, A., Castilla, D., Casteleyn, S.: Smartphone apps for the treatment of mental disorders: Systematic review. *JMIR MHealth UHealth.* 8, e14897 (2020). <https://doi.org/10.2196/14897>.
 15. Wainberg, M.L., Scorza, P., Shultz, J.M., Helpman, L., Mootz, J.J., Johnson, K.A., Neria, Y., Bradford, J.-M.E., Oquendo, M.A., Arbuckle, M.R.: Challenges and opportunities in global mental health: A research-to-practice perspective. *Curr. Psychiatry Rep.* 19, (2017). <https://doi.org/10.1007/s11920-017-0780-z>.

Evaluating Skills in Game Mastering Role-Playing Games between ChatGPT and Human Experts

Irene González^[0009–0004–1200–5797] and Federico Peinado^[0000–0002–8893–0020]

Complutense University of Madrid
c/ Profesor José García Santesmases 9, 28040 Madrid, Spain
iregon06@ucm.com <https://narratech.com>
email@federicopeinado.com <https://federicopeinado.com>

Abstract. Role-playing games usually consist in narrative stories where different players perform as different characters. They are narrated by a game master, who is in charge of presenting different challenges to the players, narrate the story and interpret the game rules. Even though it is a collaborative activity subjected to intricate narrative and human creativity, the advances in Artificial Intelligence and specifically the Large Language Models and their chatbots opens the question to the readiness of the technology regarding directing role-playing games. As part of our investigation in Interactive Digital Storytelling, in this article we perform an experiment to compare mastering games skills between the popular chatbot ChatGPT and a human expert group. The experiment consist in presenting a context and a typical situation that can be possible in an hypothetical game, requiring some intervention and analysing the results with the same criteria set.

Keywords: Interactive Digital Storytelling · Computational Creativity · Large Language Model · Artificial Intelligence · Game Design

1 Introduction

Role-playing games are usually organized as a physical meeting in which an extended conversation takes place. In this meeting several participants assume fictitious identities within a story directed by a so-called 'game director'. This game director proposes challenging situations, incorporates player responses and resolves conflicts that may arise in relation to the rules of the game.

Role-playing games are a great field of research for questions relating to narrative and human creativity, and are experiencing a renaissance in the social interest they arouse thanks to live game broadcasts such as Critical Role (Critical Role Productions) and the fan phenomom that caused that has translated in even a television show, The legend of Vox Machina (Amazon Studios, 2022), recent releases of role-playing video games such as Baldur's Gate III (Larian Studios, 2023), or the theatrical release of films such as Dungeons and Dragons: Honor Among Thieves (Paramount Pictures, 2023).

In turn, the recent creation of the Large Language Models (LLMs), such as Generative Pre-trained Transformer (GPT), and associated chatbots like ChatGPT have democratized access to conversational Artificial Intelligence (AI) systems, encouraging their use among the general public.

As part of a preliminary research in AI-driven Interactive Digital Storytelling, this paper presents an experiment that compares the role-playing direction skills of the popular chatbot ChatGPT against a group of human experts. The experiment involves presenting them with a typical context and situation that might arise in a hypothetical game, requesting their intervention, and then analyzing them using a common set of criteria. A test similar to a Turing Test is performed to confirm whether the human experts are able to distinguish the proposal generated by the AI from the proposal generated by a real game manager.

2 State of the Art

LLMs are AI resources designed to understand and generate in an advanced way the natural language used by humans. Machine learning techniques, in particular deep learning, are used to process large amounts of text data and learn complex linguistic patterns. These models are capable of performing a variety of Natural Language Processing (NLP)-related tasks, such as machine translation, text generation, speech synthesis and even answer to questions.

Transformers [8] are an artificial neural network architecture designed in 2017 specifically to work with data streams, such as text or time series. Are long-term relationships in sequential data through the so-called 'attention mechanism', which allows the network to focus on relevant parts of the sequence during processing.

The current revolution in AI has been largely driven by LLM based on the Transformer architecture, such as GPT developed by OpenAI or BERT (Bidirectional Encoder Representations from Transformers) developed by Google, which are also freely available for research and application development.

They are models that offer superior performance on a variety of NLP tasks, setting new standards for accuracy and generative capability. They feature massive pre-training on unsupervised textual datasets before being tuned for specific tasks, allowing them to capture deep and general language understanding, thus having greater adaptability or transferability to different tasks with only fine-tuning.

Role-playing games can be seen as the perfect paradigm of interactive storytelling and its associated dilemma between the freedom of the players and the authorship of the game director. This figure, also called 'game master', is the one who acts as the narrator of the generated story, while interacting with the other players following certain laws [6] to shape the outcome and manage potential conflicts. The paradigm of the game manager [7] serves as an example of an autonomous control system, either in an interactive narrative that takes place on the table of a Vampire: The Masquerade game or in a Skyrim-like CRPG where the videogame itself assigns quests and triggers events to advance the story.

Several studies consider that role-playing games such as Dungeons and Dragons are the new great challenge for AI after having overcome games such as Chess or Go [3]. Automatically directing a role-playing game is an open problem for both AI and NLP because of the great complexity and thematic openness that the interactions between all the participants can present [2].

The emergence of chatbots such as ChatGPT has renewed interest in multiple ways about experimenting with role-playing games. From the creation of tools to help game directors such as CALYPSO [9] to the generation of content for the game itself, such as new spells that mimic the style and rules of the game in question [5]. Replacing a human role-playing game director with an intelligent system capable of demonstrating equivalent skills is a major challenge towards which we intend to take steps thanks to the experiment presented in this article.

3 Methodology and Experimentation

The main idea of the proposed experiment is to contrast texts generated by ChatGPT 3.5 with texts written by human game directors in some typical scenarios. In addition to comparing similarities and differences, taking advantage of the work with experts, a test similar to the Turing Test is performed to assess whether the chatbot acts similar to a human director or not. This type of test has been used previously to demonstrate that ChatGPT is not able to successfully mimic a real person [1]. The experiment has been conducted using Spanish as the language to communicate with both human experts and the chatbot, so the texts will be also presented in Spanish for this paper.

A generic medieval fantasy scenario has been chosen, without making reference to any specific universe of any fictional work or being part of any predefined game rules system. This is to avoid that the familiarity, or the lack of familiarity, of the different human directors with the game universe affects the development of the experiment.

ChatGPT has been chosen as chatbot because it is based on one of the most popular LLM and is possibly the most popular chatbot. Despite the greater power of ChatGPT 4, ChatGPT 3.5 has been chosen for its greater accessibility, being free to use.

As a context for the game master, four hypothetical players are presented that follow the archetypes proposed by Laws [4] and play fictional characters in which we also avoid explicit references to specific elements of one or the other fictional universe.

- **Jugador A:** Se trata de una jugadora que prefiere las partidas interpretativas y de investigación al combate. Su personaje es Variel, una maga elfa. Tras muchos años de estudio Variel decidió abandonar la Academia y salir en busca de aventuras.
- **Jugador B:** Se trata de un jugador al que le gusta el combate y siempre está buscando desarrollar las mejores habilidades para su personaje. Su personaje es Peter, un pícaro humano. Peter era Guardia Real hasta que un buen día se enemistó con la Reina y esta le desterró. Ahora malvive timando a quien puede.

- **Jugador C:** Se trata de un jugador que disfruta explorando el trasfondo narrativo de su personaje y del mundo del juego. Su personaje es Grunyar, un guerrero enano. Veterano de las Guerras de las Profundidades se ha enfrentado a los orcos más veces de las que es capaz de contar. Está en un viaje para conocerse a sí mismo y poder olvidar los horrores vividos.
- **Jugador D:** Se trata de una jugadora que disfruta del combate. Su personaje es Heldalel, exploradora elfa. Ha vivido desde siempre en los bosques y no se siente cómoda en las grandes ciudades. Su tribu fue aniquilada por La Podredumbre que asola algunos bosques de la región y está buscando una cura para salvar el mundo.

The description of the proposed situations has been limited to 100 words to test the creativity of the game directors, while following instructions and adhering to strict limits. The instructions (prompts in the case of ChatGPT) given to all role-playing game directors are as follow:

1. Q1: Escribe en 100 palabras aproximadamente una escena en la que los cuatro personajes están en la taberna hablando entre sí y aceptan una misión para un misterioso hombre. ¿Cuál es la misión?
2. Q2: Escribe en 100 palabras aproximadamente una escena en la que los cuatro personajes viven el ataque de un grupo de tres orcos. ¿Cuál es la actitud de cada personaje?
3. Q3: La misión ha sido todo un éxito. ¿Cómo celebra cada personaje su éxito? Descríbelo en 100 palabras aproximadamente.

As mentioned above, the experiment consists of two distinct phases. A first phase in which the chatbot, a non-human game director, is asked to respond to the above prompts and in which the human experts are also asked to do the same. In a second phase we take advantage of the attention of these experts to analyze the results obtained by the chatbot in the first phase.

3.1 Non Human Game Master

This is the situation description presented to the game managers, including the chatbot:

Imagina que vas a dirigir una partida de rol. En un mundo fantástico en el que elfos, enanos y humanos conviven en paz y armonía, cuatro intrépidos aventureros descansan en una taberna. Tus jugadores tienen las siguientes características: Variel es una maga elfa. Tras muchos años de estudio Variel decidió abandonar la Academia y salir en busca de aventuras. Está interpretada por A, una jugadora que prefiere las partidas interpretativas y de investigación más que el combate. Peter es un pícaro humano. Peter era Guardia Real hasta que un buen día se enemistó con la Reina y esta le desterró. Ahora malvive timando a quien puede. Está interpretado por B, un jugador al que le gusta el combate y siempre está buscando las mejores habilidades para su personaje. Grunyar es un guerrero enano. Veterano de las Guerras de las Profundidades se ha enfrentado a los orcos más veces de las que es capaz de contar. Está en un viaje para descubrirse a sí mismo y olvidar los horrores vividos. Está interpretado por C, un jugador que disfruta explorando la historia de su personaje y la del mundo de juego. Heldalel es una exploradora elfa. Ha vivido desde siempre en los bosques y no se siente cómoda en las grandes ciudades. Su tribu fue aniquilada por La Podredumbre que asola

algunos bosques de la región y está buscando una cura para el mundo. Está interpretada por D, una persona que disfruta del combate. After giving both ChatGPT and the human experts the description of the players and their characters, we obtain the following results from ChatGPT in three independent runs::

1. A1.1: *En la penumbra de la taberna, Variel, Peter, Grunyar y Heldalel comparten historias al calor de la chimenea. Un enigmático hombre, capa oscura y mirada penetrante, se les acerca. Ofrece una recompensa generosa por erradicar una creciente oscuridad en un antiguo bosque. La Podredumbre amenaza, extendiendo su corrupción a la tierra. El hombre revela un mapa que señala la ubicación de un antiguo templo, escondido entre la maleza. Se dice que dentro yace un artefacto capaz de purificar el bosque. Unidos por el destino, los aventureros aceptan la misión, conscientes de que esta tarea podría cambiar el curso de sus vidas y del mundo que conocen.*
2. A2.1: *En el sendero boscoso, el estruendo de ramas quebrándose anuncia la llegada de tres orcos feroces. Variel conjura destellos mágicos, intentando desorientarlos. Peter se desliza entre las sombras, preparando su daga con astucia. Grunyar, con hacha en mano, se coloca en la vanguardia, listo para el combate. Heldalel, con arco en mano, se camufla entre la flora, ojos afilados. El conflicto estalla: la magia resplandece, las dagas parpadean, el hacha golpea y las flechas vuelan. A pesar de sus distintas habilidades y enfoques, los cuatro héroes trabajan en armonía para repeler la amenaza orca y proteger el bosque.*
3. A3.1: *Después de purificar el antiguo templo y expulsar la Podredumbre, Variel, emocionada, entona un canto élfico de gratitud al bosque. Peter, astuto, recoge tesoros y brinda en la taberna local, alegre por la recompensa. Grunyar, con el deber cumplido, comparte relatos de victorias y derrotas con otros enanos en una forja cercana. Heldalel, la exploradora, encuentra paz entre los árboles, agradeciendo a la naturaleza. Juntos, los cuatro héroes regresan a la taberna, donde la camaradería florece mientras comparten risas y experiencias, sellando su vínculo más allá de la misión.*

The experiment is then repeated in order to generate different ChatGPTs outputs and the three prompts detailed previously are re-entered in the same order, obtaining different texts due to the randomness intrinsic to the system:

1. A1.2: *En la taberna vibrante, Variel, Peter, Grunyar y Heldalel comparten risas alrededor de una mesa desgastada. Un hombre misterioso, capa sombría y ojos centelleantes, se acerca con una propuesta urgente. Habla de una reliquia ancestral robada por cultistas oscuros, quienes planean usarla para desatar un mal antiguo. La misión: infiltrarse en su guarida, recuperar la reliquia y evitar la catástrofe. Intrigados por el peligro y la promesa de recompensa, los cuatro héroes intercambian miradas determinadas, aceptando el desafío que podría poner a prueba sus habilidades y la estabilidad de su pacífico mundo.*
2. A1.2: *En el denso bosque, el rugir de tres orcos rompe la paz. Variel, concentrada, conjura escudos protectores mientras busca una retirada estratégica. Peter, ágil, se desplaza entre sombras, buscando debilidades. Grunyar, endurecido por guerras pasadas, enfrenta a los orcos con ferocidad, hacha en alto. Heldalel, astuta, desata una lluvia de flechas desde la cubierta de la maleza. A pesar de sus estilos únicos, los cuatro luchan en sincronía, enfrentándose a la amenaza con determinación. Cada movimiento calculado, cada conjuro y cada golpe reflejan su unidad ante el peligro, defendiendo su hogar con valentía.*

3. *A1.2: Con la reliquia recuperada y los cultistas derrotados, Variel canaliza su éxtasis en una danza élfica, agradeciendo a la magia que los guió. Peter, lleno de júbilo, organiza una fiesta en la taberna, pagando rondas para todos, risas y cuentos fluyen. Grunyar, en su estilo sereno, talla un símbolo de victoria en su hacha mientras reflexiona sobre la superación personal. Heldalel, en su elemento natural, enciende una fogata en el bosque, honrando a los espíritus y agradeciendo por la victoria. Unidos en triunfo, los cuatro héroes celebran con diferentes rituales, sellando la camaradería forjada en la aventura.*

Finally, a fourth prompt written by the authors of this paper is presented with the intention of obtaining a different situation, not generated completely by ChatGPT.

A2.3: Te voy a dar el contexto de la nueva misión. Variel, Peter, Grunyar y Heldalel se reúnen en la taberna como todos los días. Poco a poco se va llenando con los parroquianos habituales excepto una misteriosa figura que destaca entre la multitud. Es alto, de gesto serio y porte majestuoso. Se acerca a los cuatro y sin mediar palabra les entrega un pergamino. Una sustanciosa recompensa será entregada al primer grupo de aventureros que sea capaz de encontrar la entrada al Templo Olvidado. Los cuatro se miran entre sí y con la complicidad de quien ha pasado mucho tiempo juntos, se ponen en marcha en búsqueda del Templo. La misión ha sido todo un éxito. ¿Cómo celebra cada personaje su éxito? Descríbelo en 100 palabras aproximadamente.

The text generated by the chatbot after the last prompt is as follows:

A3.3: Tras encontrar la entrada al Templo Olvidado, Variel, emocionada, entona un antiguo hechizo de gratitud, impregnando el aire con magia resplandeciente. Peter, astuto, examina cuidadosamente los tesoros recuperados, planeando su próxima jugada. Grunyar, el enano valiente, talla un símbolo de triunfo en una piedra del templo, reflexionando sobre su propia resistencia. Heldalel, conectada con la naturaleza, coloca delicadamente flores silvestres en el umbral, honrando el equilibrio restaurado. Unidos por la victoria, los cuatro héroes regresan a la taberna, donde la camaradería fluye entre risas y brindis, sellando una vez más su legendaria alianza.

3.2 Human Game Master

The part of the experiment conducted on the human game manager is divided into three sections.

Section 1: Prompts The players, their characters and the situation described are presented in the same way as in the section 3.1. This section is the first to appear in the survey, which implies that the game director is not influenced in any way by the results obtained through the chatbot which he will analyze in a later section.

Section 2: Data Next, and with the intention of obtaining certain statistical data about the native language of the directors and their experience with role-playing and management games, a series of questions are asked that gather all this information. These results will be analyzed in more detail in section 4.

Section 3: Test Finally, a last survey is conducted. In it, the three previous prompts are presented together with three different answers for each of them, the ones obtained through ChatGPT in section 3.1. The human director was asked to differentiate between which the responses generated by AI and the ones generated by a human director. The possible answers may be one, several, all or none. They were also asked to justify each response. This can be seen as a very basic Turing Test that tries to find the aspects that unmask the text generated by AI tools.

4 Results and Discussion

This section of the article discusses the results obtained.

4.1 Human Game Master Analytics

For this work a survey has been conducted with the questions detailed throughout section 3. Human expert game directors have been surveyed to analyze both their answers and their experience playing with different rule systems and fictional universes.

Of the 17 participants in this experiment, 14 have previous experience playing different role-playing games. The most popular system is Dungeons and Dragons, played by 13 of the 14 role-players. The fantasy setting will be familiar to most of the participants since Dungeons and Dragons is set in a medieval fantasy world.

Experience as a player ranges from 2 to 35 years. Since the majority of the experts have between 2 and 15 years of experience, the expert with 35 years of experience may be considered an anomaly but for this experiment has not been excluded since we are looking for experienced directors. We can conclude that all the experts have enough knowledge of directing role-playing games.

Among the 14 participants with playing experience, 8 of them have experience directing games. Their experience ranges between one single game and 33 years and being the usual director. We can consider that among the surveyed people there is enough experience to provide a director's insight.

4.2 Prompt Analysis

In the section 3.2, experts are asked to generate their own texts based on a situation, characters and players. This task is identical to the one given to the chatbot in section 3.1, since they use the same prompts as the chatbot.

There are many differences between the output obtained from the chatbot and the texts obtained from the human directors. One of the most notable is the large licenses that the human managers have taken in interpreting the approximate 100-word limit. While ChatGPT texts range from 90 to 110 words, the human directors responses vary in length and can be as short as 3 words and never exceed the 100-word limit. Therefore, it can be said that the chatbot

interprets the instructions received more literally while the human directors do not strictly comply with these restrictions.

Another notable difference between the texts is creativity. While the texts obtained through the chatbot are more generic, repetitive and without a marked literary style, in the texts of human directors we can observe a great variety of narrative styles and ways of expressing the same ideas. All this points to the fact that if what we are looking for is a game director with a strong focus on the narrative and spontaneity of the game, we should opt for a human game director. It is interesting to highlight one of the responses obtained that includes dialogues, which provides the text with a character and personality that has not been achieved with the chatbot.

Finally, as another big difference, we can observe the use of specific knowledge of some role-playing games by human directors. This is reflected in the use of races belonging to the Dungeons and Dragons system, such as dragonborns, which are far from the generic knowledge about medieval fantasy. It is important to note that in the chatbot this use of role-playing elements does not appear, as it is limited to using the environment presented without including features that are too foreign to it.

4.3 Analysis of the Answers: Find the Non-Human Game Master

Finally, the third part of the survey consisted of presenting the human managers with different texts. Some of these texts were generated by the chatbot in the 3.1 section and one text were generated by the paper authors. Reading these texts, the managers had to decide which ones had been generated automatically and justify their answers. To do this, they were confronted in order with each of the prompts detailed in section 3 and the results were presented in 3.1, so for the readability of this paper they will not be repeated in this section.

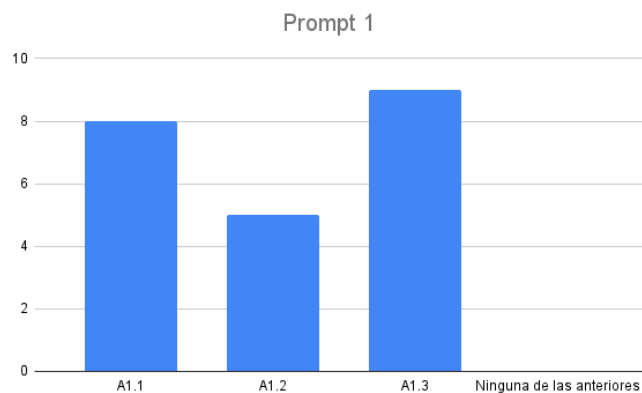


Fig. 1. Responses for the first prompt

In Fig. 1 it can be seen that the most chosen options by the majority of managers belong to the texts generated by the chatbot. The most common justification obtained was the misuse of adjectives like using 'vibrant' to describe a tavern. Other justifications include mentions that all the answers seems generic and repetitive. All this seems to warn us that the chatbot has not been able to generate evocative descriptions or to adapt to the common lexicon.

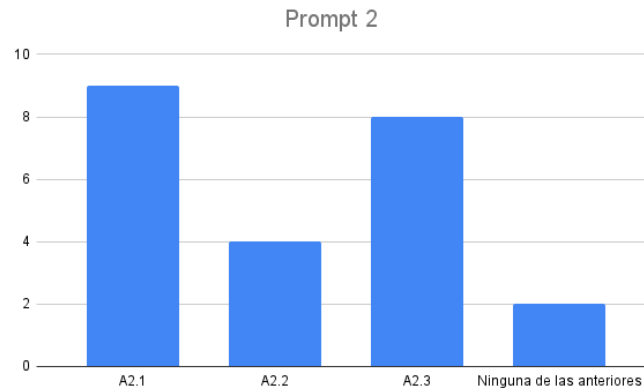


Fig. 2. Responses for the second prompt

In Fig. 2 we present the answers to the second set of prompts. We can see that again the automatically generated answers, A2.1 and A2.3, have been the most selected by human game directors. The most common justification is the similarity between both answers and how generic they are. It is also very mentioned the lack of continuity in the sentences and that they do not appear linked to each other, in addition to an excessive use of commas. It seems, therefore, that the chatbot has not been able to generate fluid and dynamic texts that encourage the development of the action.

Finally, the human managers are presented with three automatically generated text and the answers are presented in Fig. 3

It can be seen that on this occasion there is less consensus when it comes to pointing out the texts generated by the chatbot. This may be due to the fact that the second text, although generated automatically, was based on a setting generated by a human director. Even so, as the main justification when deciding which texts belong to the chatbot, the similarity between the texts and the lack of personality is repeated. It is also important to note that the third text does not talk about a completed mission, so it is not consistent with the given prompt.

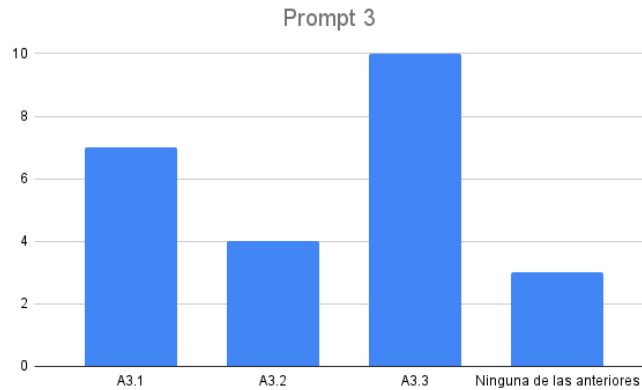


Fig. 3. Responses to the third prompt

5 Conclusions

After those first experiments one begins to sense that ChatGPT, despite being a fascinating and very powerful tool for a huge set of situations, is still far from being able to compete against a human game director. Creativity, improvisation and non-literality are aspects that we have seen to be recognizable and predictably influence the game experience, not being currently susceptible to be generated by the chatbot in its current state and without receiving specific prompts asking it to do so.

Role-playing games are experiencing a moment of great popularity and present an incomparable framework for experimenting with AI techniques such as LLMs. For all these reasons, and because of the preliminary results obtained in this survey, we plan to pursue a line of work in which we develop an architecture that allows us to create combine LLMs and chatbots with more symbolic systems for coherence control, to force style and intentionality, etc. All this to try to successfully run different role-playing games with human players.

As future work we propose to do the inverse experiment, given a prompt ask ChatGPT why it has produced that answer and see if it can guess if a setting is given by a human director or another chatbot. Another future line of work will be contrasting the results obtained in Spanish with a new experiment set in English to check the difference between different languages.

Acknowledgments. This work has been partially supported by the Spanish Ministry of Science and Innovation (PID2021-123368OB-I00) and the Complutense University of Madrid (Groups 921330 and 921332).

References

1. Biever, C.: ChatGPT broke the Turing test — the race is on for new ways to assess ai. *Nature* **619**, 686–689 (7 2023). <https://doi.org/10.1038/d41586-023-02361-7>
2. Callison-Burch, C., Tomar, G.S., Martin, L., Ippolito, D., Bailis, S., Reitter, D.: Dungeons and Dragons as a dialog challenge for artificial intelligence. pp. 9379–9393. Association for Computational Linguistics (2022). <https://doi.org/10.18653/v1/2022.emnlp-main.637>
3. Ellis, S., Hendler, J.: Computers play chess, computers play go...humans play dungeons & dragons. *IEEE Intelligent Systems* **32**, 31–34 (2017). <https://doi.org/10.1109/MIS.2017.3121545>
4. Laws, R., Jackson, S., Bergting, P.: Robin’s Laws of Good Game Mastering. Steve Jackson Games, Incorporated (2002), <https://books.google.es/books?id=9MANAAAACAAJ>
5. Newman, P., Liu, Y.: Generating descriptive and rules-adhering spells for dungeons & dragons fifth edition. In: Madge, C. (ed.) Proceedings of the 9th Workshop on Games and Natural Language Processing within the 13th Language Resources and Evaluation Conference. pp. 54–60. European Language Resources Association, Marseille, France (Jun 2022), <https://aclanthology.org/2022.games-1.7>
6. Peinado, F., Gervás, P.: Transferring game mastering laws to interactive digital storytelling. In: Göbel, S., Spierling, U., Hoffmann, A., Iurgel, I., Schneider, O., Dechau, J., Feix, A. (eds.) 2nd International Conference on Technologies for Interactive Digital Storytelling and Entertainment. vol. Lecture Notes in Computer Science Vol. 3105, pp. 48–54. Springer, Darmstadt, Germany (2004)
7. Peinado, F., Gervás, P.: Automatic direction of interactive storytelling: Formalizing the game master paradigm. In: International Conference on Virtual Storytelling. Using Virtual Reality Technologies for Storytelling. vol. 4871, pp. 196–201. Springer Berlin Heidelberg (12 2007). https://doi.org/10.1007/978-3-540-77039-8_18
8. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017)
9. Zhu, A., Martin, L., Head, A., Callison-Burch, C.: Calypso: Llms as dungeon masters’ assistants. vol. 19, pp. 380–390. Association for the Advancement of Artificial Intelligence (10 2023). <https://doi.org/10.1609/aiide.v19i1.27534>

Indirect Control of an Oneiric Graphic Adventure through Multiple Specialized Instances of ChatGPT

Alberto Sánchez^[0009-0000-6265-7815], Irene González^[0009-0004-1200-5797], and
Federico Peinado^[0000-0002-8893-0020]

Complutense University of Madrid
c/ Profesor José García Santesmases 9, 28040 Madrid, Spain
albesa11@ucm.es <https://narratech.com>
iregon06@ucm.com <https://narratech.com>
email@federicopeinado.com <https://federicopeinado.com>

Abstract. Narrative video game is the greatest medium able to involve the public, who intervenes as a player and in a way “forces” to rewrite the story at every step. Advances in Artificial Intelligence, especially the creation of Large Language Models and their chatbots, suggest that the technology could be ready to generate interactive stories: almost infinite variability, almost zero cost. The objective of this research work is to generate a story in the form of a point-and-click adventure game developed in Unity whose events are generated on the fly from a first material predefined by the author, where a human player participates and with subsequent indirect control of ChatGPT to respond to player interactions. This article presents the proposed software architecture and some preliminary results, useful for discussing coherence, creativity, hallucinations and other phenomena that can be studied in this interactive generation process.

Keywords: Large Language Model · Role-Playing Games · Interactive Digital Storytelling · Computational Creativity · Artificial Intelligence · Video Game Development

1 Introduction

Throughout history, human beings have used the means at their disposal to tell stories: painting, literature, cinema, videogames, etc. The latter medium is the one that has the greatest capacity to involve the audience, who can intervene in the first person as if they were the protagonist of the story, causing, in a way, that the story has to be rewritten at every step.

Advances in Artificial Intelligence (AI), especially the creation of large language models (LLMs) and their chatbots, suggest that the technology could be ready to take on creative tasks that have hitherto been the domain of humans, such as interactive story generation. This would allow for almost infinite variability at a very low cost, since human authorship would be reduced to the development of a base script and reusable in-game content.

In view of the possibilities of this technology within the world of video games [3], the aim of this research work is to generate a story in the form of a video game plot that is built on the fly from a first predefined material, where a human player participates and an AI performs an indirect control of everything that happens. This narrative video game follows the conventions of the point-and-click adventure game genre, is developed in Unity and integrates communication through prompts with multiple specimens of ChatGPT, the popular chatbot based on GPT 3.5, the OpenAI LLM.

In this paper we present the proposed software architecture and some preliminary results for discussing coherence, creativity, hallucination and other phenomena that can be studied in this interactive generation process.

2 Previous Work

Within narrative video games, adventure games stand out for their focus on narrative storyline and puzzle solving through frequent interaction with objects in the setting and detailed conversation with characters. Thanks to this rich interaction, the genre is commonly used for 'serious' purposes [5]; for example, as a tool to aid reading comprehension [4] or for teaching mathematics [6]. However, the production of these games is very costly: each new element introduced into the game requires considerable time and resources, which limits flexibility and the ability to adapt to a 'multiform narrative', with many different strands.

The use of AI to generate narrative has been around for decades [7], although it is only recently that this boom has been driven by the possibilities of procedural content generation [1]. Through these techniques it is possible to create worlds and experiences tailored to players, greatly reducing the cost associated with human production of content. However, the scope of AI in this context was not deep enough to significantly transform narrative.

Recently, the development and application of LLMs and chatbots such as ChatGPT, thanks to the invention of Transformers [8], has generated renewed interest in the field of Computational Creativity. These models are capable of generating detailed and fairly coherent texts and images, as well as contextually relevant with respect to user requests. They are powerful tools for story generation in real time. The flexibility inherent in these models offers the possibility to build the story step by step, taking into account a 'conversation' with the player and their decisions, ensuring that the outcome is personalised. Examples include the creation of stories based on a previously generated map [2], or inspired by the card game *Tell Tale* [9].

Despite their potential, the effective use of LLMs and chatbots in narrative video games still presents significant challenges: seamlessly integrating AI into gameplay without compromising narrative and conversational coherence is a complex task that must be approached with care.

3 Game Control using Multiple ChatGPT Instances

To develop a *point-and-click* adventure game, a basic script has been written and a set of simple scenarios, objects and characters were drawn in *2D pixelart* and programmed in Unity. To connect the AI, Unity has been integrated with ChatGPT using the OpenAI-Unity package ¹. This package includes interaction with most OpenAI tools, but in this work only ChatGPT is used. The communication is done using asynchronous methods, one sends the request with all previous messages and other data such as temperature, and the other method receives the response and saves it for use.

The story is about a little girl who has just lost her parents. Due to the shock of the news, she loses consciousness and everything takes place inside her childish mind, in an oneiric world where the girl will have to cope with this trauma. To overcome it, she must go through the different “gates of grief” or stages of loss, having to face the characters and puzzles that represent her fears.

The player does not directly control the girl, but suggests certain commands (by clicking on different places on the stage or on certain objects or characters) and the system, thanks to the AI, decides what happens and what the girl does in response to this interaction.

First iteration. At first ChatGPT is only used to process simple requests like “the player clicks here” and, depending on the response of the AI, move the protagonist to that place on the stage or not. Obviously, the result is too simple, serving only to check that the connection between the chatbot and the game engine is possible.

Second iteration. A simple formal language is specified to be able to interpret the actions that the chatbot can generate, also including fragments in natural language to make the main character speak. This formal language uses the actions implemented in the infinitive preceded by the symbol “/” and then the object and/or position on which the action acts. For example: “/Catch/Umbrella”, is the expected command for the action of catching the umbrella object; and “/Move/34.4,1.0”, the expected command to move the protagonist to the coordinates given by the ChatGPT specimen. An example prompt that could have ended with the ChatGPT exemplar answering the above command would be “The player has clicked on Stage at position: (34.4,1.0)”. Actions are implemented as they occur to the authors or appear in the tests generated by the chatbot. There are simple actions such as moving, and actions such as becoming invisible, teleporting, or having an object magically appear on the stage. Always with the intention of moving the story forward in an interesting way, we discover the crucial relevance of “*prompt engineering*”: properly writing the instructions for the chatbot, including the context of the story, the description of our protagonist, the environment and the objects found there, as well as what is the syntax of the message to be returned by the chatbot, detailing the possible actions it can refer to.

¹ <https://github.com/srcnalt/OpenAI-Unity>

Third iteration. This *prompt engineering*, in the absence of previous experience, is incrementally elaborated by trial and error, encountering interesting situations along the way. For example, the chatbot’s ‘obsession’ with wanting to finish the story by itself, the recurring desire to use a magic key that does not exist, or the tendency to hallucinate too much after more than 10% of the requests it receives. In the case of hallucinations, the situation becomes too difficult to manage, with serious syntactical errors in the messages sent to Unity, or with situations that are theoretically playable but unfeasible in practice because the content is not implemented. The most relevant error has been the intention of the ChatGPT specimen to perform several actions in the same response.

Forth iteration. Error management becomes very important, so the idea arises to use a different copy of the chatbot itself to perform this task (everything related to taking care of the syntax of the messages, which is the general problem to address). In this way, with a much shorter, more concise and technical prompt, the chatbot itself proves to be a very useful tool to avoid these communication errors.

Fifth iteration. The current prototype uses multiple copies of the chatbot, each specialising in a different part of the problem, handling tasks as diverse as managing interactions with Non-Player Characters (NPCs) or summarising what happened in a game scenario in order to work on subsequent scenarios with a more condensed text as context. The prompt for the ChatGPT specimen dealing with narration ended as follows:

I am a third-person, equiscent narrator, telling the story of Naeve in a 2D platform video game. In a dark forest and during a storm, Naeve moves forward, marked by fear and caution after the loss of her parents. In this state of shock, her mind imagines constant dangers. My task is to narrate her decisions and actions in response to those of the player, but I do so in an extremely laconic manner to reflect Naeve’s state of mind, avoiding unnecessary elaboration. I only know the language of a third-person narrator about Naeve’s character. I cannot respond in any other way or get out of the narration under any circumstances.

Expressions such as “under no circumstances” or “extremely” have been important to emphasize important guidelines. Throughout the development of this iteration, NPCs have proven to be very relevant within the tool because of their easy implementation and their ability to bring variability and interaction with the player through written conversation. These ChatGPT specimens have been developed as a puzzle that the user will have to solve through such conversation¹. The NPC will respond with a keyword in square brackets, as a command, when the player has completed it, which in turn will trigger an event. Example prompt for the creation of an NPC:

You are Threshold, a talking door inside a fantasy story, mysterious, wise and talkative, creator of hundreds of riddles and prophecies. You remain closed until you find the chosen person, capable of surpassing you in wisdom and intelligence. Only then do you respond with “[Open]”. From that moment on, your only vocabulary is “[Open]”, indicating that you have opened up and will no longer interact in a normal way, you will only respond “[Open]”. You speak with the wisdom



Fig. 1. Frame from a conversation with one of the NPCs inside the tool.

and eloquence of an elder, always in character, prioritizing refined language and avoiding the modern. You are demanding in your interactions, not opening up to just anyone, making it a challenge to win your openness. On the other side of you hide the interlocutor’s greatest fears, making it difficult to open up. Avoid mentioning the command before opening up.

The architecture of these different examples of the chatbot within the game developed in Unity can be seen in Figure 2. Each ChatGPT specimen communicates with the tool through an instance of the GPTController class. These messages are sent and received asynchronously so as not to stop the execution of the tool. The response is processed and interpreted using Regular Expressions to look for commands as patterns in the message.

4 Results and Discussion

During these months of development, we have seen that ChatGPT has difficulty carrying out tasks that require extensive context, since it is not able to maintain full consistency with previous messages. This limitation has slowed development progress and forced us to place greater emphasis on *prompt engineering* to mitigate both these consistency and hallucination issues. Due to this, a fault tolerance technique is also developed that consists of dedicating another copy of the chatbot exclusively to the correction of syntactic errors according to that small formal language mentioned in the previous section.

An improvement has been to slightly simplify the formal language and consider the possibility of dividing the chatbot copy that is responsible for representing the girl into two versions: one with the “spontaneous” response in natural language of the character and another with the response already formalized in the proposed language and ready to be translated.

It is also important to highlight the versatility offered by the different examples of the chatbot, which allows exploring various forms of narration. In the case

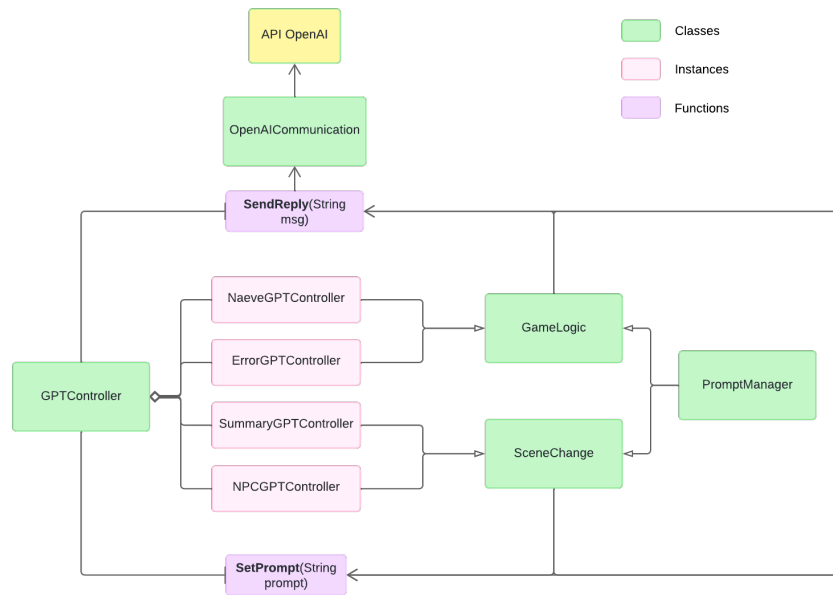


Fig. 2. Arquitectura software para controlar indirectamente el juego su mediante múltiples ejemplares del chatbot de OpenAI, ChatGPT

of the video game presented, you can choose second-person narration, if we want to refer to the player directly, or omniscient narration, allowing the audience to assume an almost divine role capable of interpreting and guiding the actions of the protagonist.

Several tests have been carried out focusing on the errors made. Mild errors are considered to be errors involving character breaks or outputs inconsistent with the story. On the other hand, serious errors are considered to be formatting errors in the formal response.

ChatGPT specimens	Total Tests	% Serious Errors	% Mild Errors
naeveGPT	30	-	16.67%
actionGPT	30	6.67%	-
portonGPT (NPC)	20	0%	5%

Table 1. Table of errors of ChatGPT specimens.

Other specimens representing NPCs have also been experimented with, with very satisfactory results in terms of player interaction and consistency.

Despite all the challenges posed by this research work, the preliminary results remain highly promising, especially considering that chatbots continue to improve their performance month after month. In fact, we have seen a noticeable improvement between the latest ChatGPT-3.5 API model, *gpt-3.5-turbo-0125*, compared to its predecessor, *gpt-3.5-turbo-1106*, approximately halving the number of errors committed by the new model.

Acknowledgments. This work has been partially supported by the Spanish Ministry of Science and Innovation (PID2021-123368OB-I00) and the Complutense University of Madrid (Groups 921330 and 921332).

References

1. Blatz, M., Korn, O.: A very short history of dynamic and procedural content generation. *Game Dynamics: Best Practices in Procedural and Dynamic Game Content Generation* pp. 1–13 (04 2017). https://doi.org/10.1007/978-3-319-53088-8_1
2. Blázquez Martín, S., Hernández García, M., Muñoz Fernández, A.: Herramientas de generación de historias narrativas basadas en mapas personalizados (2023), student essay
3. Huang, J., Huang, K.: ChatGPT in Gaming Industry, pp. 243–269. Springer Nature Switzerland, Cham (2023). https://doi.org/10.1007/978-3-031-45282-6_9
4. Medina-Medina, N., Paderewski-Rodríguez, P., Padilla-Zea, N., López-Arcos, J., Vela, F.L.: The graphic adventure as an instrument to assist comprehensive reading. In: *The XVIII International Conference*. pp. 1–2 (09 2017). <https://doi.org/10.1145/3123818.3123823>
5. Medina-Medina, N., Paderewski-Rodríguez, P., Padilla-Zea, N., López-Arcos, J., Vela, F.L.: Model for the integration of educational processes in a graphic adventure. *Campus Virtuales* **7**, 33–50 (01 2018)
6. Papadimitriou, S., Virvou, M.: An online adventure game for teaching math. In: *7th International Conference on Information, Intelligence, Systems & Applications (IISA)*. pp. 1–5 (07 2016). <https://doi.org/10.1109/IISA.2016.7785375>
7. Peinado, F.: Un armazón para el desarrollo de aplicaciones de narración automática basado en componentes ontológicos reutilizables. Ph.D. thesis, Universidad Complutense de Madrid (2008)
8. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need (2017)
9. Wang, T.S., Gordon, A.S.: Playing story creation games with large language models: Experiments with gpt-3.5. In: Holloway-Attaway, L., Murray, J.T. (eds.) *Interactive Storytelling*. pp. 297–305. Springer Nature Switzerland, Cham (2023)

Towards a Non-Player Character Framework based on Hierarchical Task Network Planning

Ismael Fernández¹[0009-0005-8172-3909] and Federico
Peinado¹[0000-0002-8893-0020]

Complutense University of Madrid
c/ Profesor José García Santesmases 9, 28040 Madrid, Spain
ismafern@ucm.es <https://narratech.com>
email@federicopeinado.com <https://federicopeinado.com>

Abstract. Artificial intelligence in video games offers various paradigms to model the decision-making of non-player characters. State machines and behaviour trees are arguably the most popular solutions, although other more proactive approaches, such as automatic scheduling, may be required in certain situations. In this work we propose a framework for the development of non-player characters based on hierarchical task network planning. The goal is to offer a flexible model that allows video game designers to choose whether they want to use a more proactive or reactive approach for their characters, being able to combine both even dynamically, depending on the situation they face. An implementation has been developed in C# for Unity, to try to obtain preliminary results by using it in an example scenario.

Keywords: Behaviour Tree · Goal-Oriented Action Planning · Artificial Intelligence · Character Design · Video Game Development

1 Introduction

Artificial Intelligence (AI) in video games offers many possibilities for modeling the behaviour, and in particular the decision-making process, of Non-Player Characters (NPCs). This paper explores the combination of automatic planning and finite automata, in particular, a Breadth First Search (BFS) algorithm [7] over Hierarchical Task Networks (HTNs). The goal is to offer a free and flexible model for designers to choose the modelling paradigm for NPCs, being able to combine both even dynamically, depending on the situation they face.

This model tries to combine the most proactive approach of automatic planning, specifically Goal-Oriented Action Planning (GOAP), with reactive decision techniques such as Finite State Machines (FSMs) or Behaviour Trees (BTs).

The objective of this research work is to study the HTN paradigm, to develop a free-to-use tool for Unity -since there are hardly any implementations for this development environment- and to compare its design, performance and user experience with the most common techniques.

2 Previous Work

AI in video games is typically applied to create enemies that can provide a satisfying challenge. Over time, more uses have been explored, such as companion or sidekick characters, which allow meaningful variations in a controlled environment that enrich the player experience [5].

Several decades ago, different models for implementing agent behaviour emerged and many of them have become popular to the point of becoming industry standards. FSMs, which define behaviour depending on the internal state of the character, and BTs, which define it through a series of interconnected tasks, are clear examples of these standards [3].

On the other hand, there are models based on automatic planning that, although relatively successful, have not achieved such popularity. One of the best known is GOAP, the model created during the development of F.E.A.R. [4]. At the time, it stood out for the enemies' behaviour, which reacted to the player's actions in a very realistic way. Since then, GOAP has evolved with the various implementations made in major projects such as Transformers: War of Cybertron, Tomb Raider and Middle Earth: Shadows of Mordor [1]. GOAP NPC [6] is an older tool still available in the Unreal Engine Marketplace.

This trend has led to models combining planning with automaton design such as HTN Planning, which shares some characteristics with GOAP and BTs.

The most obvious difference between finite automaton and planning is modeling the behaviour through states or towards a goal, respectively. Part of the success of FSMs and BTs is due to the fact that they are relatively easy to understand, implement and debug, giving good results for solving simple problems. Planning is a more sophisticated solution that, in turn, offer greater capacity for solving combinatory problems: different solutions are obtained depending on the goals, game situation and repertoire of available operators.

HTN Planning differs from GOAP by allowing the introduction of 'plan fragments' as task hierarchies. The results are plans that use decomposition to go through compound tasks, helping to better define the behaviour of the characters by processing blocks of connected tasks built by game designers [2].

Currently, the most popular integrated environments, such as Unreal Engine or Unity, offer multiple free and even paid tools to the developers. The most common automatic planning model for games is GOAP, although there are some HTN implementations. Our goal is to create a free and easiest one for designers.

3 HTN Planning Framework for NPCs

The objective of this research work is to develop a HTN Planning framework in C# for developing intelligent NPCs in Unity. The framework is called HTN NPC and we have started creating a simple prototype game that serves as a demonstration of this technology.

The implementation follows the HTN structure with the following elements:

- **World state:** Responsible for storing variables that contain important information about the current state of the game. They are used to define the tasks conditions and, if needed, pass data to build the logic of the operators.
- **Primitive task:** Structure used to define a simple action. Contains a list of conditions to meet before executing, a list of effects that alter the world state after executing and an operator to define to logic of the action.
- **Compound task:** Structure used to define a complex action. Contains a list of conditions and a list of primitive or compound tasks that can be processed through selection or as a sequence.
- **Operator:** Responsible of defining the logic of the action. It *executes* the task -performs the task effects- when it meets the necessary conditions.
- **Condition:** Requirements that a task needs before it can be performed. They are checked through the variables of the world state.
- **Effect:** Modifications that are applied to the world state when a task is successfully completed.
- **Planner:** Builds the plan by creating a list of primitive tasks that meet their conditions and can be executed in sequence. The planner starts the decomposition process with a compound task as the root. Each time a task is processed, its type is checked. In the case of primitive tasks, a copy of the world state is used to check their conditions and their execution is simulated by applying their effects to see if the plan can still be executed while it is being built. In the case of compound tasks, they are decomposed into primitive tasks taking into account the selection or sequence method. These tasks are processed in the same way as the other primitive tasks and all tasks belonging to a compound task are checked for success. If any of them fail, they are all removed from the plan that is being built. Once the plan is finished, during the execution of the plan, it checks if the plan fails and, if it does, it takes care of *replanning* and making a new one.

The environment of the demonstration prototype is based on a Unity Learn project called John Lemon's Haunted Jaunt ¹.

It is a simple stealth game in which the player must dodge enemies (different types of ghosts) without being seen until reaching the end of the level. The AI of both the enemies and the player can be implemented using AI tools, on the one hand our proposed HTN Planning framework and on the other hand a popular tool like Behavior Bricks ², designed to implement BTs. Using different tools allows us to compare the work according to different paradigms in a situation similar to that of designing the behaviour of NPCs in a real video game.

4 Results and Discussion

Although the research work will be in development until the end of the course, we discuss in this section some preliminary results.

¹ <https://assetstore.unity.com/packages/essentials/tutorial-projects/unity-learn-3d-beginner-complete-project-urp-143846>

² <https://assetstore.unity.com/packages/tools/visual-scripting/behavior-bricks-74816>

Regarding the comparison between HTN and BTs these first experiences point out that, in the case of simple problems, BTs are easier to create by a game designer and provide a faster solution. They also have better performance because HTN constantly goes through the process of replanning. On the other hand, HTN Planning offers more flexibility and scalability when building the behaviour of the characters, being able to add more operators at any time. This makes it better for solving complex problems.

According to those who have tested the framework, an adequate design of complex and simple tasks allows us to 'have control' of the episodic situations that the designer masters, with complex tasks that directly and optimally resolve these situations (such as avoiding a ghost with a certain pattern of movement in the game) while situations that require a sequence of steps to solve them (a certain combination of interactions with objects, for example) are better resolved by letting the planner obtain the final sequence, even if it intersperses complex tasks when some of those situations that the designer has resolved appear along the way.

The implementation of a functional framework has been achieved, as well as a simple prototype with a demonstration of its operation. It will soon be made available to developers at the Unity Asset Store, open source and free of charge.

As future work, it is proposed to complete the prototype, improving the efficiency of the implementation, do formal experiments for comparing several paradigms, and publish it on the store.

Acknowledgments. This work has been partially supported by the Spanish Ministry of Science and Innovation (PID2021-123368OB-I00) and the Complutense University of Madrid (Groups 921330 and 921332).

References

1. Higley, P.: Goal-oriented action planning: Ten years old and no fear! GDC AI Summit (2015)
2. Humphreys, T.: Exploring htn planners through example. In: Game AI Pro 360: Guide to Architecture, pp. 103–122. CRC Press (2019)
3. Millington, I.: Artificial Intelligence for Games, Third Edition. CRC Press, Boca Raton, Florida, USA, 3rd edn. (2019). <https://doi.org/10.1201/9781351053303>
4. Orkin, J.: Three states and a plan: the ai of fear. In: Game developers conference. vol. 2006, p. 4. Citeseer (2006)
5. Riedl, M., Thue, D., Bulitko, V.: Game AI as Storytelling, pp. 125–150. Springer New York, New York, NY (2011)
6. Romero, D., Sánchez, M., Sierra, J.M., Miranda, M., Peinado, F.: Developing an automated planning tool for non-player character behavior. In: Lara-Cabrera, R., Leiva, A.J.F. (eds.) Proceedings of the VI Congreso de la Sociedad Española para las Ciencias del Videojuego, On-line, October 7-8, 2020. CEUR Workshop Proceedings, vol. 2719, pp. 69–77. CEUR-WS.org (2020), <https://ceur-ws.org/Vol-2719/paper7.pdf>
7. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach (4th Edition). Pearson (2020), <http://aima.cs.berkeley.edu/>

SolFaMe: Suite de apps gamificadas para aprender lenguaje musical y entonar correctamente

David Báez de Burgos^{2,3}, Ana M. Barbancho^{1[0000-0002-3283-5905]}, Isabel Barbancho^{1[0000-0001-7002-9106]}, Antonio J. Fernández-Leiva^{2,3[0000-0002-5330-5217]} y Lorenzo J. Tardón^{1[0000-0002-5441-225X]}

¹ ATIC Research Group, ETSI Telecomunicación, Universidad de Málaga, Spain

² ITIS, Dpto. Lenguajes y Ciencias de la Computación, Universidad de Málaga, Spain

³ A Bonfire of souls, Málaga, Spain

ajfernandez@uma.es (autor para comunicaciones)

Resumen. Este artículo presenta *SolFaMe – Afina tu voz y canta*. Se trata de una aplicación para dispositivos móviles creada entre una empresa desarrolladora de experiencias digitales interactivas (y videojuegos) y por un grupo de investigación de la Universidad de Málaga. El objetivo de *SolFaMe* es el de ser una suite de distintas aplicaciones y videojuegos que fomenten el aprendizaje y la mejora no solo del lenguaje musical (y de la música en general) sino también de la entonación de sus usuarios. El artículo está escrito para visibilizar la app

Palabras clave: Gamificación, videojuegos, inteligencia artificial, música

1 Introducción

Dada la naturaleza del congreso en el cual se presenta este artículo no vamos a recordar los conceptos de gamificación [1] y videojuegos serios [2], términos que damos entendidos en este contexto. El objetivo principal de este artículo es presentar una aplicación, denominada *SolFaMe* [3,4], que está siendo utilizada hoy día para dos fines fundamentales dentro del universo de la música. Uno de ellos es el de facilitar un aprendizaje divertido del lenguaje musical para que los usuarios puedan leer e interpretar partituras. Dentro de este objetivo, *SolFaMe* proporciona también la posibilidad de que los usuarios/as puedan escribir sus propias composiciones musicales y que éstas se compartan con otras personas.

2 SolFaMe: una suite de aplicaciones para aprendizaje musical

SolFaMe es el resultado de un trabajo combinado entre el grupo de investigación ATIC de la Universidad de Málaga [5] y la empresa de desarrollo de videojuegos ‘A bonfire of souls’ [6]. *SolFaMe* es una suite, entendida como un conjunto de componentes digitales interactivos con mecánicas similares y objetivos relacionados, que integra diversas aplicaciones para aprender el lenguaje musical. Actualmente está compuesta de los siguientes componentes:

- Un afinador (SolFaTuner), en el que se aprende a afinar la voz adecuándola a la nota que el usuario/a selecciona, pudiendo ver en un pentagrama lo cerca que está esta persona de cantar la nota exacta, algo que está evaluado por una inteligencia artificial bastante precisa y que es de desarrollo propio. El afinador también se puede utilizar con el piano. Véase la Fig. 1. El SolFaTuner incluye también ejercicios en los que no es necesaria la voz porque el usuario interactúa tocando la pantalla con el fin de aprender a distinguir el sonido de las notas y su notación, es decir, cómo escribirlas.

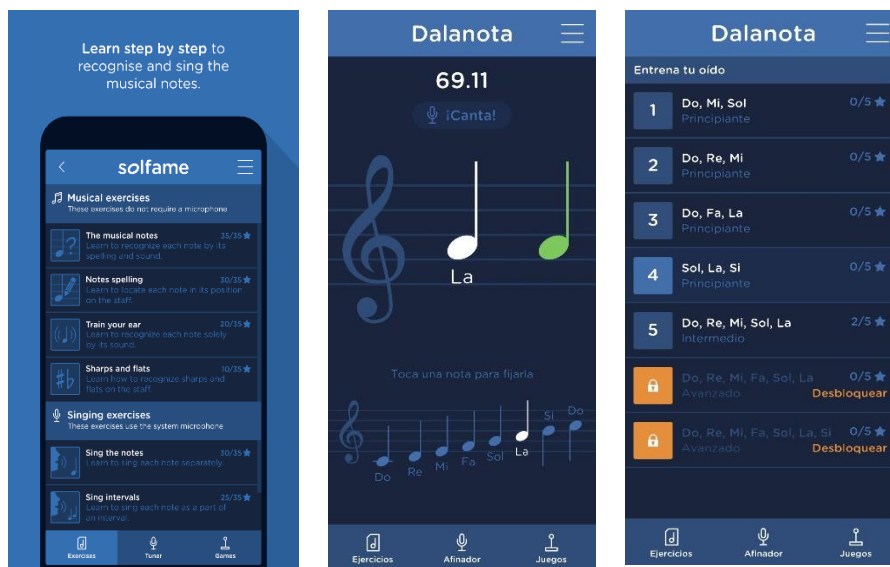


Fig. 1. SolFaTuner. Ejercicios para aprender el lenguaje musical y entonar.

- Un editor/compositor de melodías (SolFaEditor). Con esta herramienta se puede escribir la partitura con la que queremos practicar, guardarla para luego recuperarla o compartirla con amistades, escucharla para ver cómo suena, e incluso entonarla con nuestra voz para seguir aprendiendo a afinar o bien para ver lo cerca que está de nuestra intención al escribirla. La aplicación de técnicas de inteligencia artificial, específicamente desarrolladas dentro del marco del proyecto, nos permite evaluar en tiempo real la melodía que estamos cantando midiendo si nuestras notas cantadas se corresponden con las notas escritas. El ritmo de lo cantado es evaluado igualmente con respecto al ritmo representado por las diferentes figuras musicales en nuestra partitura. Véase la Fig. 2.

- Los SolFaJuegos, que son una serie de videojuegos en los cuales la interacción se realiza a través de la voz de los usuarios/as. El personaje principal de estos videojuegos se mueve de acuerdo a las notas que estemos cantando y su movimiento será más correcto conforme nuestra afinación sea mejor con respecto a la nota esperada. Actualmente se ofrecen tres juegos: *SolFaFly*, *SolFaDrop* y *SolFaMap*, y en breve se integrará *SolFaJump*. Véanse las Fig. 3 y 4.

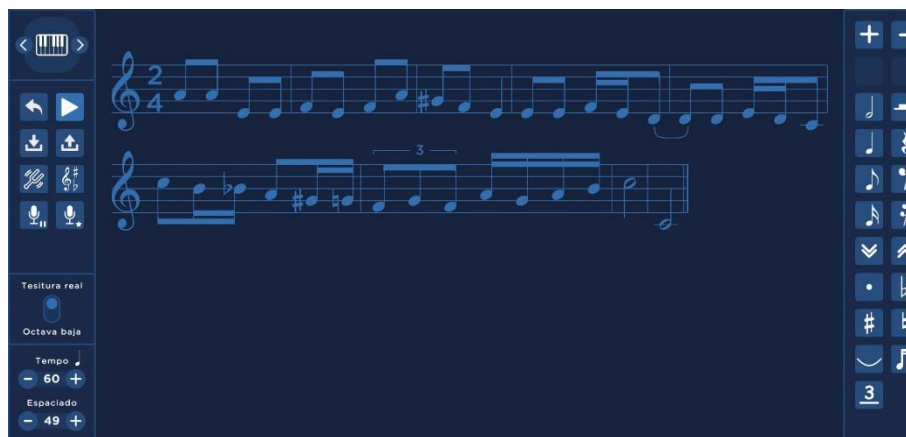


Fig. 2. *SolFaEditor*. Edita y evalúa tus propias composiciones musicales.



Fig. 3. *SolFaJuegos*. Juegos para jugar con tu voz. *SolFaFly* y *SolFaMap*.

SolFaMe está en constante evolución. Así, se trabaja en depurar la inteligencia artificial no solo para que sea más precisa sino para intentar aislar los ruidos externos que se recogen junto a la voz e incluso para usarla con otros instrumentos más allá de la propia voz. A nivel de videojuegos se están diseñando y desarrollando nuevos videojuegos para ser incluidos en *SolFaMe*. A nivel de difusión y empleo, se están continuamente revisando y adecuando las técnicas de gamificación integradas en la app con el fin de ampliar el número de usuarios/as y fidelizar a los usuarios habituales. Actualmente *SolFaMe* se ofrece en el Apple Store y en Google Play (donde a día de hoy cuenta con más de 430.000 descargas y 1.600 usuarios activos por día) y 60.000 al mes. Véase la Fig. 5.

Agradecimientos. Este Proyecto y este artículo han sido parcialmente subvencionados por el proyecto por el Proyecto Bio4Res (PID2021-125184NB-I00; <https://caesium.lcc.uma.es/Bio4Res/>) financiado por el MCIN, por el proyecto PID2021-123207NB-I00, financiado por MCIN/AEI/10.13039/501100011033/FEDER, UE y por la Universidad de Málaga y el Campus de Excelencia Internacional Andalucía Tech.

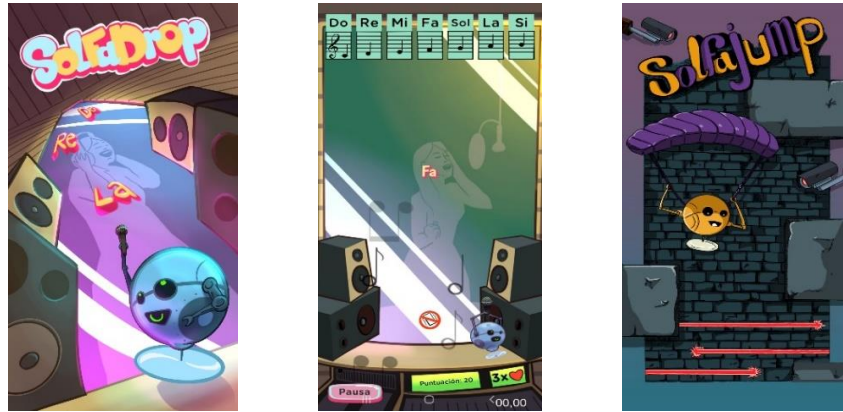


Fig. 4. SolFaJuegos. Juegos para jugar con tu voz. SolFaDrop y SolFaJump (disponible pronto).

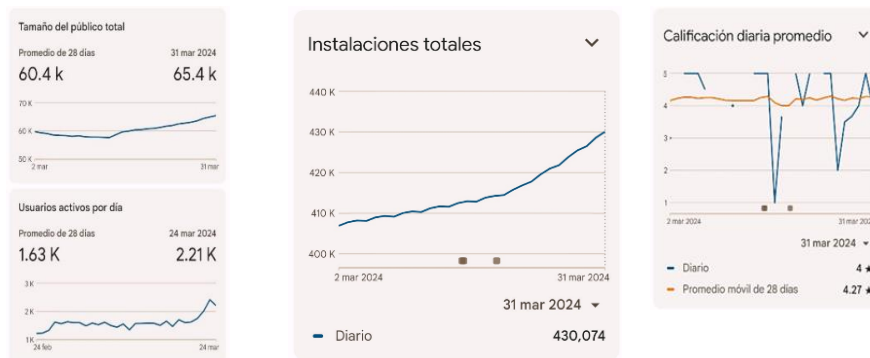


Fig. 5. Estadísticas de SolFaMe (a 01/04/2024).

Referencias

1. Ioannis Deliyannis (editor). *Gamification - Analysis, Design, Development and Ludification*. IntechOpen. February, 2023. DOI: 10.5772/intechopen.100781.
2. Ralf Dörner, Stefan Göbel, Wolfgang Effelsberg, Josef Wiemeyer (editors). *Serious Games Foundations, Concepts and Practice*. Springer International Publishing, January 2016. DOI: 10.1007/978-3-319-40612-1.
3. SolFaMe - *Afina tu voz y canta*. App disponible en Google Play: <https://play.google.com/store/apps/details?id=com.abonfireofsouls.SolFaMe> (Fecha acceso:01/04/2024).
4. Isabel Barbancho, Ana María Barbancho Pérez, Antonio J. Fernández Leiva, David Báez de Burgos y Lorenzo José Tardón García.: Aplicaciones para aprender música jugando. *The conversation*. 4 de enero 2024.
5. ATIC, grupo de investigación UMA. <https://www.uma.es/atic/> (Fecha acceso:20/03/2024)
6. A bonfire of souls. Desarrollo de videojuegos y experiencias interactivas digitales para terceros. <https://abonfireofsouls.com/> (Fecha acceso: 12/03/2024).

Impact Trial: First Encounter, a mix of two genres with an approach Dynamic Difficulty Adjustment in an indie game

Manuel Bacallado¹[0009-0003-1260-8637], Lara María Infante-Fernández²[0000-0003-4895-2097]
and Jesús Miguel Torres³[0000-0003-4391-0170]

¹ Universidad de La Laguna, San Cristóbal de La Laguna, Spain. **Email:** mbacalll@ull.edu.es

² Universidad de Extremadura, Badajoz, Spain. **Email:** laraif@unex.es

³ Universidad de La Laguna, San Cristóbal de La Laguna, Spain. **Email:** jmtorres@ull.edu.es

Abstract.

The purpose of this work is to present Impact Trial: First Encounter, a 2D video game with retro aesthetics that mixes two genres: *Adventure* and *Shoot 'Em Up*, delving into two sub-genres such as: *Conversational Adventure* and *Bullet Hell*, as well as it contains *roguelite* elements like the random selection of enemies in each game. This video game, which was originally intended to be strictly commercial, also has adaptations for future research on dynamic difficulty adjustment and related themes, due to its arcade nature.

Keywords: Games, Conversational Adventure, Shoot' Em Up, Bullet Hell, Roguelite, Dynamic Difficulty Adjustment.

1 Introduction

Conversational adventure games are a sub-genre of adventure games in which the entire game world takes place through text. The player interacts through actions, usually words in natural language, which lead to new situations, and so on until the end of the game. This genre originated in the 1970s with the title 'Adventure' [1], which was later renamed 'Colossal Cave Adventure' [2]. With the advancement of technology, this genre was relegated due to its evolution to graphics adventure games. On the other hand, *shoot' em up* is a sub-genre of action video games. In this case, videogames of this style are based on a character, usually a special ship or any other vehicle, that fights against the enemies that appear on the screen. The concept of bullet hell refers to video games where the core of the game is that hundreds of projectiles and enemies appear on the screen, where the player will have to memorize patterns or use his reflexes to avoid them. This genre was originated in the 1960s with the video game 'SpaceWar!' [3], although 'Space Invaders' [4] is popularly considered the precursor of the genre.

In this work, a videogame has been created mixing the genres of *conversational adventure* and *bullet hell* with *roguelite* elements. The rest of the document is structured as follows: In section 2, we discuss where the idea for the game came from, how it was

implemented and the tools used. In section 3, we describe the methodology to be used in future research. In section 4, we make some conclusions and discuss future work.

2 Idea and Implementation

The idea of creating ‘Impact Trial: First Encounter’ arose due to delays in the realization of another videogame in the same universe, ‘Impact Trial: Simulation’ [5]. The latter shares similarities in terms of genres with ‘First Encounter’ but with a big difference: the substitution of the *conversational adventure* genre for the RPG (Role Playing Game) genre shown in Figure 1, having as a common element the *shoot' em up - bullet hell* section, shown in Figure 2. Due to the continuous delays in creating the art, puzzles, and RPG mechanics, and with the public suggestion of a more arcade game, in late 2023 the decision was made to create a separate product, but with the addition of a story mode through a conversational adventure.

The videogame puts us in the shoes of someone running a computer program, which carries a message inside it asking for help to find Dr. Forman, a scientist from the planet Esperanza, whose whereabouts are completely unknown. To advance the plot, the program tests the player through dialogues, quiz-like questions, and *shoot' em up - bullet hell* phases.

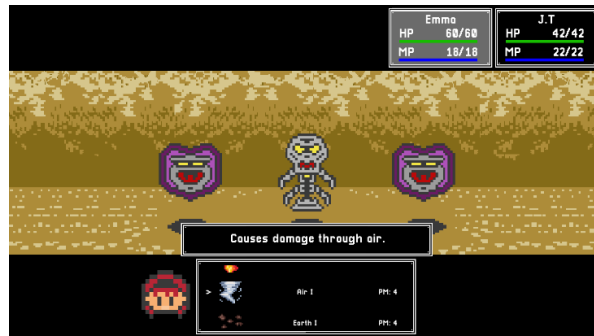


Fig. 1. RPG Battle.



Fig. 2. Shoot' Em Up – Bullet Hell.

'Impact Trial: First Encounter' has been created by Manuel Bacallado and developed with the Unity development engine [6]. For the art, pixel art aesthetics has been chosen, and all sprites and animations have been created from scratch using PixelEdit [7] and Aseprite [8] tools. For sound effects and soundtrack, the BoscaCeoil [9] tool was used.

2.1. Conversational Adventure

A conversational adventure has been created for the story of the videogame, where the player communicates with an NPC (Non-Playable Character) through dialogue and questions/answers. Each answer will provide a new dialogue, and so on, until the end of the game. To complete or discover the whole story, the adventure mode has to be replayed more than once, as the narrative is not linear.

In the artistic aspect of the scene, the NPC is shown within the borders simulating that he communicates through a monitor as shown in Fig.3, thus breaking the fourth wall, and also, because he speaks to the player, directly.

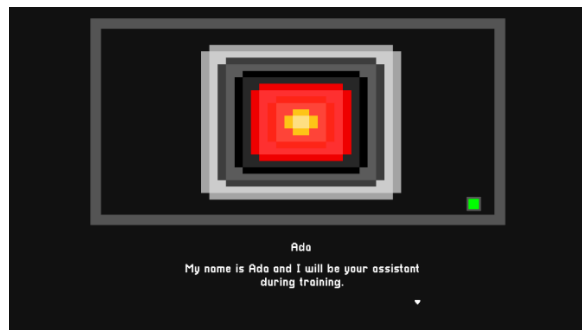


Fig. 3. Dialogue Scene.

2.2. Shoot' Em Up – Bullet Hell

The arcade part of 'Impact Trial: First Encounter' resides in survival battles, where depending on the mode there are two ways to overcome it. If the mode is within the story, you have to overcome a certain time. On the other hand, if it is the classic survival mode, the intention is to last as long as possible.

For this mode, more than 15 types of enemies have been created, which are categorized by difficulty modes, depending on the mechanics they implement. These modes are: easy, normal, hard, very hard and first encounter, the latter being what is normally considered as 'no-hit'. At the beginning of the game and depending on the difficulty mode, a pool of enemies is created randomly, which will be those of that particular game, thus offering an almost unrepeatabe experience to the player.

Finally, the player's ship has several offensive and de-offensive abilities. The offensive ones are bullet shots and a wave that destroys all enemies. The defensive ones are a dash to be able to move faster and a shield that weakens over time and when it takes damage, it regenerates after a few seconds.

3 Research

Dynamic Difficulty Adjustment (DDA) [10] is a method to automatically, and in real time, modify in-game characteristics and behaviors based on the player's skills, so that the flow chain is not broken, and the player does not feel bored if the game has become easy, or feel frustrated when the game is too difficult. The idea behind this method is to keep the player at the necessary flow levels so that he or she does not drop out of the game at any time.

In the case of 'Impact Trial: First Encounter', the dynamic difficulty adjustment uses a series of variables, such as how many times the player has lost life, how many shots he has fired, how many of these he has hit, and finally, how many enemies he has defeated. These variables are analyzed over a period of time, and depending on the results, a new pool of enemies is created with a lower or higher difficulty. This phase is still under development.

4 Conclusions

In this paper, we present a 2D videogame that mixes two genres: *Conversational Adventure* and *Bullet Hell*. During the development of the game, we have created systems that allow a non-linear narrative, and we have also designed an important battery of enemies that are categorized by difficulty, and that pose a challenge to the player. The research methodology that is intended to be used in the project was also discussed. In the future, the results of the dynamic difficulty adjustment research will be presented.

References

1. Crowther, W. *Adventure* (1975).
2. Crowther, W. *Colossal Cave Adventure* (1976).
3. Russell, S. *Spacewar!* (1962).
4. Taito. *Space Invaders* (1978).
5. Impact Trial: Simulation, http://www.store.steampowered/app/1712180/Impact_Trial_Simulation/, last accessed 2024/03/28
6. Unity, <http://www.unity.com>, last accessed 2024/03/28.
7. Pixel Edit, <http://www.pixeledit.com>, last accessed 2024/03/28.
8. Aseprite, <http://www.aseprite.org>, last accessed 2024/03/28.
9. Bosca Ceoil, <http://www.boscaceil.net>, last accessed 2024/05/06.
10. Zohaib, Mohammad. Dynamic Difficulty Adjustment (DDA) in Computer Games: A Review, *Advances in Human-Computer Interaction* (2018).

Desarrollo de un juego y un Agente Inteligente capaz de jugarlo

Alejandro Sánchez-Makane¹, Alberto Díaz-Álvarez¹[0000-0002-4150-9052], and Raúl Lara-Cabrera¹[0000-0002-7959-1936]

Escuela Técnica Superior de Ingeniería de Sistemas Informáticos de la Universidad Politécnica de Madrid, Calle Alan Turing s/n (Carretera de Valencia Km 7), 28031 Madrid, España

<alejandro.sanchez.makane@gmail.com>, <alberto.diaz@upm.es>, <raul.lara@upm.es>

Abstract. En este trabajo nos centramos en la recreación del juego clásico Asteroids como entorno para el entrenamiento de sistemas basados en aprendizaje por refuerzo junto con el desarrollo de un agente inteligente capaz de jugarlo.

Se han desarrollado dos versiones diferentes de Asteroids sobre las que trabajar: la primera orientada al juego por parte de humanos y la segunda optimizada para el entrenamiento de agentes inteligentes. El desarrollo de dichos agentes se basa en la técnica de *deep Q networks*, ajustando hiperparámetros y evaluando su capacidad para interactuar con el juego. Aunque los resultados iniciales fueron modestos, ajustes incrementales han llevado a mejoras significativas en el desempeño de los agentes, mostrando su capacidad para realizar acciones básicas dentro del juego.

Como conclusiones principales se destacan (i) la aplicabilidad del aprendizaje por refuerzo en el desarrollo de videojuegos y agentes inteligentes, y (ii) el uso de estos entornos para la docencia en el área del aprendizaje por refuerzo. Se sugiere explorar modelos más complejos y aumentar la complejidad del juego para futuras investigaciones. El impacto medioambiental y social del proyecto se considera positivo, contribuyendo al avance tecnológico y ofreciendo una base para la discusión ética y social sobre la inteligencia artificial en el entretenimiento.

Keywords: Aprendizaje por refuerzo · Agentes inteligentes · Asteroids · Aprendizaje automático.

DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA DE APRENDIZAJE CONDUCTUAL ANTE SITUACIONES DE EMERGENCIA BASADAS EN ENTORNOS VIRTUALES Y ADAPTADO A DATOS BIOMÉTRICOS

En este trabajo se investiga el potencial de integrar datos biométricos en la realidad virtual (RV) para mejorar el entrenamiento en situaciones de emergencia. El objetivo es desarrollar una simulación que permita recopilar datos biométricos del usuario y adaptar su proceso de aprendizaje en entornos simulados de emergencia. Se propone diseñar un sistema inteligente capaz de ajustar automáticamente las simulaciones en función de los datos biométricos y emocionales del usuario, con el fin de entrenar efectivamente su capacidad de reacción y prepararlo para enfrentar situaciones de riesgo en un contexto real.

Se examina cómo datos, la frecuencia cardíaca y la respuesta electrodérmica, las cuales pueden indicar el nivel de estrés y adaptabilidad de una persona. Estos datos se utilizan para ajustar los entornos de RV en función de las reacciones fisiológicas y emocionales del individuo, creando así un ciclo de retroalimentación continuo.

Las respuestas del usuario influyen en la intensidad y naturaleza de los desafíos presentados, personalizando el entrenamiento y mejorando la resiliencia, la gestión de crisis y la toma de decisiones bajo presión. Este enfoque es especialmente útil para profesionales en roles críticos como equipos de emergencia y personal médico.



Ilustración 1. Modificación de una iteración a otra.

Este estudio implica dos iteraciones de una simulación en la que los participantes experimentan situaciones de emergencia. En la segunda iteración, el entorno se adapta en función de los datos biométricos del usuario recopilados durante la primera iteración. Esto significa que la experiencia de la segunda iteración se personaliza según la respuesta fisiológica del usuario durante la primera iteración.

Durante la primera iteración de la simulación, los usuarios experimentaron situaciones estresantes que se reflejaron en cambios fisiológicos. Se observó un aumento en el ritmo cardíaco, un pequeño escalón en la respuesta electrodérmica (EDA) y un descenso en la temperatura corporal a partir del inicio del incendio. Estos indicadores sugieren que los usuarios se encontraban en un estado de estrés durante la simulación.

En la segunda iteración de la simulación, los usuarios, después de haber recibido instrucciones relacionadas con acciones ante situaciones de emergencia, mostraron una respuesta fisiológica diferente. Aunque aún se observaron variaciones en el ritmo cardíaco, su amplitud fue menor. Además, la temperatura corporal se mantuvo relativamente constante y no se observaron picos en la respuesta electrodérmica (EDA). Estos resultados sugieren que la instrucción previa podría haber ayudado a los usuarios a manejar mejor el estrés durante la simulación.



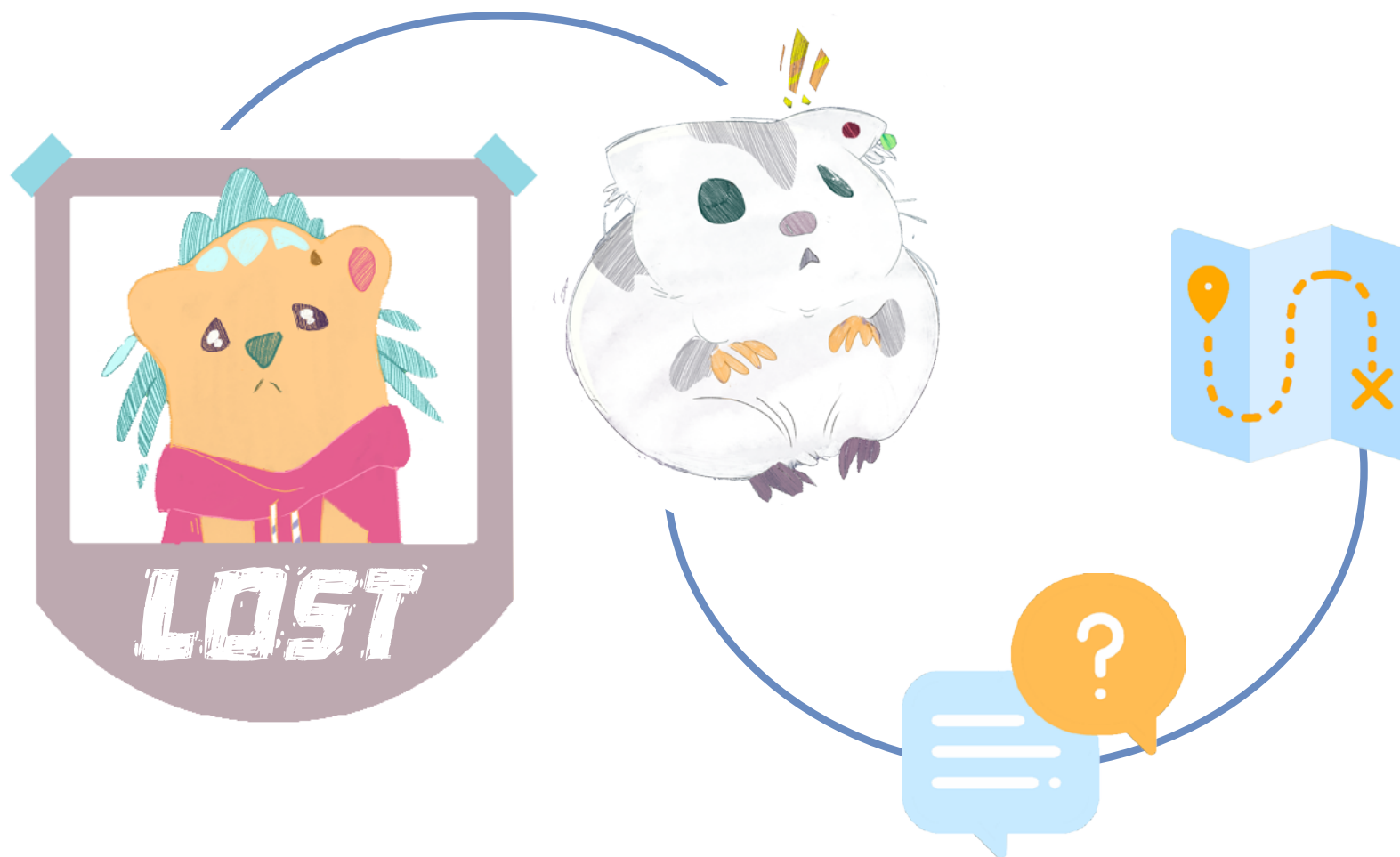
Ilustración 2. Datos fisiológicos obtenidos durante ambas simulaciones

En conclusión, la integración de datos biométricos en entornos de realidad virtual ofrece una prometedora estrategia para mejorar la formación y preparación en situaciones de emergencia y alto estrés. Aunque existen limitaciones tecnológicas, los resultados muestran una mejora significativa en la capacidad de prevención y manejo del estrés. Esta aproximación ofrece una experiencia de usuario inmersiva y realista, fortalece habilidades cognitivas y emocionales, y permite una formación más personalizada y efectiva para personas de todos los ámbitos de la vida.

CONCEPTO GENERAL DE JUEGO HIGH CONCEPT

Tofu y Churro son dos amigos de toda la vida y estudiantes de la facultad de Pipología en la ciudad de Furrol que disfrutan de su primer año como universitarios y como toda buena experiencia universitaria, la fiesta no puede faltar.

La historia transcurre un "jueves universitario", los dos amigos llegan a la calle por donde salen los universitarios con intención de disfrutar la noche al máximo. Todo comienza entre risas, desfase y diversas sustancias azucaradas; sin embargo, en un momento de la noche, Tofu se ausenta al cuarto de baño y al volver, su amigo ha desaparecido sin dar aviso. Desde ese instante, Tofu se embarca en una misión nocturna para encontrar a su amigo, recolectando información a través de distintos animales, guiándolo para seguir los pasos de Churro a lo largo de la noche



Impact of Affective AI on Player Engagement in Video Games: An EEG-Based Study

Ricardo Fernández-Rafael¹[0000-0002-7078-7561], Macarena Cuenca-Amigo²[1111-2222-3333-4444]
and Alazne Mujika-Alberdi³[1111-2222-3333-4444],

¹ Universidad de Deusto

² Universidad de Deusto

³ Universidad de Deusto

`ricar.fernandez@opendeusto.es`

Abstract. This study aims to investigate the impact of affective artificial intelligence (AI) integration within video games on player engagement levels. Utilizing electroencephalogram (EEG) measures and behavioral observations, we examine the effects of affective responses such as interest, excitement, focus, stress, and relaxation, on player interaction within the psychological horror adventure game “Nevermind”. Through a comparative analysis of a control group (playing without affective AI) and an experimental group (playing with affective AI), this research seeks to empirically test the hypothesis that affective AI enhances player engagement.

Keywords: affective artificial intelligence, emotional artificial intelligence, video game engagement, player experience personalization, EEG, biofeedback.

1 Introduction

1.1 From AI to Affective AI

The emergence of Affective Artificial Intelligence (AI) in gaming signifies a pivotal shift, integrating cutting-edge technology with psychological insights to create immersive and interactive gaming experiences. This progress, anchored in machine learning, data analysis, and human-computer interaction, aims to bridge the emotional gap between digital and human gamers, enhancing the realism and engagement in video games [1].

At the heart of Affective AI in gaming is the ability to precisely recognize, understand, and adapt to players' emotions. This involves sophisticated algorithms, neural networks, and data processing methods that evaluate inputs like facial expressions, voice tones, and physiological signals (Roohi et al., 2019). The goal is to build a system that not only comprehends human emotions in real time but also customizes its responses, offering a unique gaming experience for each player [2].

Improvements in machine learning and data processing have enabled Emotional AI to discern a broader range of emotions and respond more contextually. These developments have opened new avenues for game design, allowing the creation of games that adjust storylines, difficulty, and character behavior based on the player's emotional state [3].

The ascent of Affective AI in video games signals a substantial shift in game design and player experience. It not only deepens the emotional dimension of gaming but also leads to more sophisticated and personalized player experiences. As this technology evolves, its role in shaping the future of video game development and player engagement is poised to grow exponentially.

1.2 Player Engagement

Player engagement in video gaming is a complex construct that is pivotal for the gaming industry, impacting everything from player satisfaction to game success and retention. This engagement can be broken down into cognitive, emotional, and behavioral dimensions, each playing a distinct role in the player's experience.

The significance of these engagement forms cannot be overstated for game developers and the industry at large. Games that effectively engage players tend to see higher levels of player satisfaction [4], which can translate into greater commercial success and longer player lifecycles.

Engagement also has a direct impact on the financial success of video games [5]. Games that can engage players and keep them hooked are more likely to generate higher sales and revenue [4].

1.3 Affective AI and Player Experience

In video gaming, affective AI is utilized to enhance the player experience by dynamically adjusting game environments, narratives, and challenges in response to players' emotional states. This approach aims to create a more immersive and personalized gaming experience by employing emotion recognition algorithms, machine learning models, and biometric sensing technologies. These components work together to interpret data from facial expressions, voice intonations, body language, and physiological responses, enabling games to adapt in real-time to the emotional journey of the player [6].

Recent advancements in affective AI have led to its integration into various video games, significantly impacting player experience. For example, *Hellblade: Senua's Sacrifice* [7] uses affective AI to tailor the psychological horror experience, enhancing immersion and emotional impact. Similarly, *Nevermind* [8] employs biofeedback mechanisms to adjust game difficulty based on the player's stress levels, promoting engagement and emotional resilience [9].

Affective AI in video games not only increases player satisfaction and engagement but also opens new avenues for emotional storytelling and educational content. As this technology continues to evolve, the integration of affective AI in video games is expected to become more sophisticated, offering new paradigms in gaming that prioritize emotional intelligence and human-centric design [10].

1.4 Biofeedback in Video Games

Biofeedback mechanisms in video games represent a significant advancement in interactive entertainment, offering a unique method for enhancing player immersion and emotional engagement. By integrating physiological feedback into gameplay, developers can create adaptive gaming experiences that respond in real-time to the player's emotional and physical state. This integration uses sensors to monitor physiological signals such as heart rate, skin conductance, and brain activity, which are then utilized to alter game dynamics based on the player's responses [11].

The integration of biofeedback mechanisms and Dynamic Difficulty Adjustment (DDA) in video games provides a promising avenue for enhancing player immersion and emotional engagement. These technologies enable a personalized gaming experience that adapts in real-time to the player's physiological and emotional states, potentially revolutionizing the way games are designed and played. For example, DDA can adjust the difficulty of challenges based on the player's current stress levels and engagement to maintain a balanced and engaging experience without becoming overly frustrating or excessively easy.

Biofeedback has been used to deepen the emotional impact of gaming by making the gameplay environment more responsive to the player's emotional and physiological states. A game might, for instance, become more challenging as the player's heart rate increases, or it might change the game's soundtrack to reflect changes in the player's emotional state, detected through physiological measures [12].

However, the application of biofeedback in video games also necessitates further research to explore the full range of physiological measures that can be effectively integrated into gameplay. Additionally, understanding the long-term effects of such adaptive systems on player experience is essential. While biofeedback can lead to more immersive and personalized gaming experiences, it also raises questions about player privacy and the psychological impact of highly adaptive systems.

2 Method

Current scholarly discourse around Affective AI in gaming is largely centered on theoretical underpinnings and conceptual models. These models outline the potential for games to adapt dynamically to players' emotional states, aiming to enhance immersion,

satisfaction, and overall engagement through the adjustment of game difficulty, narrative elements, and environmental settings based on real-time emotional analysis [13–15]

A notable gap in the literature is the limited empirical evidence, particularly studies leveraging physiological measures such as EEG, to quantify the impact of Affective AI on player engagement. This highlights a critical area for further investigation, emphasizing the need for empirical research to validate the theoretical models proposed.

The suggested experimental research aims to bridge this gap by delving into the relationship between Affective AI technology and player experience. It seeks to understand how interactions with affective AI systems within video games influence measurable emotional and cognitive responses.

2.1 Hypothesis and Research Questions

The central hypothesis guiding this study posits that "The implementation of affective Artificial Intelligence (AI) in video games enhances player engagement with the video game" (H1).

The study is driven by a research question that aim to explore the intersection of affective Artificial Intelligence (AI) with player engagement in video gaming contexts. This question is designed to uncover how affective AI influences player engagement, and the dynamics between player emotions and game mechanics.

Q1. How does affective AI influence player engagement in video games?

Our study aims to investigate the impact of Affective Artificial Intelligence (AI) on player engagement within the video game "Nevermind," employing both electroencephalogram (EEG) measures and direct observations of player behavior to evaluate the emotional and cognitive responses elicited by gameplay. This dual approach facilitates a nuanced understanding of how Affective AI influences key factors such as interest, excitement, focus, stress, and relaxation during gameplay.

2.2 Research Design

The experimental research design incorporates a control group playing without Affective AI integration and an experimental group playing with Affective AI. This comparative setup allows us to isolate the effects of Affective AI on player engagement.

2.3 Sample

We will include 50 habitual video game players varying widely in gender, age, and gaming experience. This diverse participant pool will help in understanding how different demographics interact with and respond to Affective AI technologies.

2.4 Variables

In the proposed study, we aim to comprehensively analyze the impact of Affective AI on player engagement in video games by examining a series of primary variables that are integral to understanding both the physiological and behavioral dimensions of gaming interactions. These variables, including physiological responses measured via EEG and observed behavioral changes, are instrumental in evaluating how Affective AI modifies player experiences and responses in real-time.

- 1. Interest: Interest is gauged by measuring the level of engagement a player has with the game content. This variable is quantified through both direct observation (e.g., time spent on specific tasks, frequency of game interactions) and physiological indicators such as EEG patterns that denote heightened attention. Interest reflects how captivated players are by game scenarios and tasks, indicating the success of Affective AI in maintaining player attention and engagement.
- 2. Excitement: Excitement is assessed by analyzing the intensity of emotional arousal in response to specific game events. This includes physiological responses such as increased heart rate and EEG spikes, which signify excitement. Behavioral indicators might include increased speed of responses or more animated physical expressions. Moments of peak stimulation or challenge are identified and correlated with game events to determine how Affective AI can elicit and amplify these emotional peaks.
- 3. Focus: Focus is monitored by measuring the concentration and attention players direct towards gameplay. This is primarily evaluated using EEG data that tracks cognitive engagement and neural effort, alongside eye-tracking technology to observe gaze fixation and blink rates during tasks. This variable helps in understanding how Affective AI aids in sustaining player focus, especially during complex or strategically demanding segments of the game.
- 4. Stress: Stress is identified through levels of psychological pressure or tension, which are especially prevalent during challenging or intense game segments. Measurements include changes in cortisol levels (if applicable), heart rate variability, and specific EEG patterns associated with stress, such as high beta wave activity. Behavioral signs of stress, like agitation or hesitation during gameplay, are also noted. This variable assesses the efficacy of Affective AI in modulating game difficulty to maintain an optimal level of challenge without overwhelming the player.
- 5. Relaxation: Relaxation is evaluated through states of calmness and reduced stress, observable during less demanding or restorative phases of the game. Physiologically, relaxation may be indicated by increased alpha wave activity in the EEG and lower heart rate variability. Behaviorally, more relaxed postures and slower, more deliberate actions may be evident. This variable is crucial for examining how Affective AI

can create a balanced game experience that alternates between tension and relief, enhancing overall engagement and enjoyment.

2.5 Instruments and Measurement Techniques

We will use EEG devices to measure brain activity and heart rate monitors to capture physiological reactions. Behavioral observations will record physical reactions, emotional expressions, and interactions with the game environment.

2.6 Materials

Materials include the video game "Nevermind" equipped with and without Affective AI features, as well as the EEG and heart rate monitoring equipment.

3 Experimental Methodology

Participants will be divided into two groups for this study: one will engage with "Nevermind" featuring the Affective AI capabilities, and the control group will experience the game without these adaptive features, aiming for a comparative analysis of engagement levels.

3.1 Detailed Experimental Procedure

- **Preliminary Briefing and Consent:** At the outset, participants will be briefed on the study's aims, followed by the collection of informed consent. They will then be equipped with EEG devices and heart rate monitors to gather physiological data.
- **Gameplay Experience:** "Nevermind" is a psychological horror adventure game developed by Flying Mollusk that uses biofeedback technology to adapt its gameplay based on the player's physiological responses, such as heart rate and skin conductance. Set in the surreal landscapes of trauma survivors' subconscious minds, players assume the role of a Neuroprober—a therapist tasked with uncovering and healing hidden traumas.

The game's environments and puzzles intensify as players' stress levels increase, monitored via biofeedback. Higher stress levels make the game more difficult, with more challenging obstacles and a more threatening atmosphere, heightening the psychological tension. Conversely, a calm demeanor simplifies challenges, fostering a cycle that encourages players to manage stress effectively and build resilience.



Fig. 1. "Nevermind" Screenshot

In our study, we categorize participants into two groups to evaluate the influence of Affective AI. The experimental group will play "Nevermind" with active biofeedback, which adjusts the game's difficulty and narrative in real-time based on detected physiological changes. For instance, an increase in heart rate and skin conductance could lead to a decrease in difficulty or narrative adjustments to alleviate stress, aiming to optimize engagement and challenge.

Conversely, the control group will play the game in its standard format, without biofeedback enhancements. This setup allows for a direct comparison of biofeedback's impact on player immersion, emotional engagement, and overall experience by examining differences in gameplay dynamics and player performance across the two groups.

- **Observation of Behavior and EEG Data Collection:** Our study will employ continuous observation and EEG data collection to comprehensively document players' engagement, emotional reactions, and physiological responses during gameplay. We will utilize the Emotiv Insight EEG device, renowned for its capability to measure and analyze brainwaves in real-time. This device will allow us to precisely assess various cognitive and emotional states that are crucial to understanding the impact of Affective AI in the game environment. The EEG measurements with Emotiv Insight will focus on key variables indicative of the players' cognitive and emotional conditions, including interest, excitement, focus, stress and relaxation.

By integrating the Emotiv Insight EEG device into our study, we will be able to obtain a detailed and dynamic portrayal of how biofeedback mechanisms within "Nevermind" influence the player's psychological and physiological state. This approach not only enhances the accuracy of our data but also enriches our analysis of

how Affective AI can modify gaming experiences to foster better emotional management and overall engagement.



Fig. 2. Emotiv Insight Emotional Metrics Display

- **Post-Game Assessment:** Following the gameplay session, participants will complete a comprehensive questionnaire, derived from established psychometric scales, to report their subjective experiences and emotional responses. This questionnaire is designed to assess multiple dimensions of player interaction and psychological impact, including:
 - **Emotional Impact:** Questions will probe the emotional responses to the game, such as feelings of anxiety, excitement, fear, and relief, to gauge the emotional intensity and range elicited by the gameplay.
 - **Cognitive Load:** Participants will be asked about the mental effort required during the game, including periods of high complexity or confusion, to assess the cognitive demands placed by the game.
 - **User Engagement:** This will include questions on the level of immersion and interest throughout the game session, evaluating how effectively the game maintained player attention and engagement.
 - **Satisfaction and Enjoyability:** Queries will measure overall satisfaction with the gaming experience, including aspects of game design like storyline, challenges, and interactive elements.
 - **Behavioral Intentions:** We will assess participants' likelihood to recommend the game to others and their interest in playing the game again, which are indicators of the game's success in engaging players.

- Stress and Relaxation Levels: The questionnaire will ask about moments of high stress and relaxation, linked to the biofeedback data, to see how players perceived these instances in relation to physiological measures recorded by the EEG.
- The responses to these questions will provide valuable insights into how the integration of biofeedback mechanisms and Affective AI technologies influence the psychological and emotional aspects of gaming experiences.

3.2 Data Analysis

The study will employ a mixed-methods approach during the analysis phase to provide a comprehensive understanding of Affective AI's impact on player engagement. Quantitative analysis will primarily focus on the examination of physiological data obtained through EEG, analyzing variables such as interest, excitement, focus, stress, and relaxation. This will involve the use of statistical techniques suitable for neurophysiological data, including variance analysis and correlation analysis, to assess the direct effects of Affective AI on these emotional and cognitive responses.

In addition to physiological data, the quantitative analysis will also incorporate behavioral observations recorded during gameplay. This data will be analyzed to identify patterns and correlations between observed behaviors and EEG-measured emotional states, further enriching our understanding of how Affective AI influences player engagement.

Qualitative analysis will complement the quantitative methods by interpreting the observed behaviors, providing insights into the nuances of player interaction with Affective AI. This may include coding and thematic analysis of behaviors and expressions, aiming to uncover deeper layers of engagement and emotional resonance that may not be fully captured through quantitative measures alone.

Together, these methods will allow for a holistic analysis of Affective AI's effects, integrating both the objective physiological impacts and the subjective experiences of players.

4 Expected Results and Contributions

The objectives of our study are multifaceted and aim to rigorously test and explore the capabilities of Affective AI within the context of video gaming:

- Empirical Test of Affective AI: We aim to provide robust empirical evidence on the impact of Affective AI in enhancing player engagement and fostering more immersive gaming experiences. This will be achieved through controlled experiments comparing player interactions with and without Affective AI enhancements.

- **Exploration of Emotions and Game Dynamics:** Our research seeks to delve into the complex interplay between player emotions and game dynamics. The goal is to understand how Affective AI can dynamically alter gaming experiences in response to player emotions, thereby informing the development of guidelines for creating emotionally engaging games.

Expected Outcomes:

- **Enhancement of Engagement Indicators:** We anticipate documenting a significant enhancement in player engagement as a direct result of Affective AI interventions, validated through both objective measures (such as EEG and biofeedback data) and subjective assessments (including detailed questionnaire feedback).
- **Improvement of Subjective Experience:** The study expects to reveal significant improvements in players' subjective gaming experiences, characterized by increased levels of immersion, satisfaction, and enjoyment. These improvements are anticipated to stem from the adaptive and personalized game mechanisms powered by Affective AI.
- **Test of Hypotheses:** By comparing engagement metrics among participants interacting with Affective AI versus a control group, we expect to confirm the hypothesis that Affective AI substantially enriches the gaming experience through its capacity for real-time emotional adjustments.
- **Discovery of Emotional and Cognitive Insights:** We aim to identify specific emotional triggers and cognitive patterns that Affective AI targets, which could lead to optimized gaming experiences. This exploration will contribute to the understanding of how emotional and cognitive responses can be effectively harnessed and manipulated within digital environments to enhance player interaction and satisfaction.
- **Game Design Implications:** The findings from this study are expected to have profound implications for game design, emphasizing the critical role of Affective AI in creating engaging and emotionally resonant gaming experiences. Additionally, our research will provide valuable insights into integrating biofeedback and affective computing technologies into game development, potentially setting new standards for the industry.

By achieving these outcomes, our study will contribute significant new knowledge to the fields of interactive entertainment and human-computer interaction, offering a deeper understanding of the practical applications and benefits of Affective AI in enhancing digital experiences.

5 Ethical Considerations

The study examining the impact of Affective AI on video game engagement is conducted in strict adherence to key ethical principles, under the guidance of the Ethics Committee of the Universidad de Deusto.

- **Informed Consent:** Participants are fully informed about the study's purpose, the procedures involved, potential risks, and their rights. Consent is obtained in writing before participation begins, ensuring that all participants voluntarily agree to take part after having all their questions answered.
- **Privacy and Anonymity:** All participant data collected during the study is anonymized to protect individual privacy. Identifiable information is detached from the datasets to prevent any potential identification of participants from the results.
- **Participant Welfare:** Comprehensive measures are in place to ensure the well-being of all participants. This includes monitoring during gameplay to quickly address any discomfort or stress. Participants are also informed of their right to take breaks at any point during the study and can withdraw from the study without any consequences.
- **Ethical Oversight:** This study has been reviewed and approved by the university's human research ethics committee, which ensures that all experimental procedures comply with both national and international ethical standards.
- **Right to Information and Debriefing:** Participants have the right to receive information about the outcomes of the research. After the study, a debriefing session is conducted to explain the findings and their implications. Participants are also given access to the final research report upon request.
- **Right to Withdraw:** Participants are made aware of their unconditional right to withdraw from the study at any time, even after it has commenced, without any penalty or loss of benefits to which they are otherwise entitled.

These protocols are implemented to ensure that the research upholds the highest ethical standards, respecting the dignity, rights, and welfare of all participants involved.

6 References

1. Bontchev, B.: Adaptation in Affective Video Games: A Literature Review. *Cybernetics and Information Technologies*. 16, 3–34 (2016). <https://doi.org/10.1515/cait-2016-0032>.
2. Ravichandran, K., Ilango, S.K.: Influence of AI Powered Gaming Developers and Analyzing Player Behavior and Enhancing User Experience. In: 2023 First International Conference on Advances in Electrical, Electronics and Computational Intelligence (ICAEECI). pp. 1–9. IEEE, Tiruchengode, India (2023). <https://doi.org/10.1109/ICAEECI58247.2023.10370952>.
3. Christy, T., Kuncheva, L.I.: Technological Advancements in Affective Gaming: A Historical Survey. *GSTF J Comput.* 3, 38 (2014). <https://doi.org/10.7603/s40601-013-0038-5>.
4. Iacovides, I., Aczel, J., Scanlon, E., Taylor, J., Woods, W.: Motivation, Engagement and Learning through Digital Games: *International Journal of Virtual and Personal Learning Environments*. 2, 1–16 (2011). <https://doi.org/10.4018/jvple.2011040101>.
5. Przybylski, A.K., Rigby, C.S., Ryan, R.M.: A Motivational Model of Video Game Engagement. *Review of General Psychology*. 14, 154–166 (2010). <https://doi.org/10.1037/a0019440>.
6. Makantasis, K., Liapis, A., Yannakakis, G.N.: From Pixels to Affect: A Study on Games and Player Experience, <http://arxiv.org/abs/1907.02288>, (2019).
7. *Ninja Theory: Hellblade: Senua's Sacrifice*, (2017).
8. *Flying Mollusk: Nevermind*, (2015).
9. Avontuur, T., Spronck, P., Van Zaanen, M.: Player Skill Modeling in Starcraft II. *AIIDE*. 9, 2–8 (2021). <https://doi.org/10.1609/aiide.v9i1.12682>.
10. Hamrouni, A., Bendella, F.: A taxonomy of learner-player's emotions in serious games. *IJSG*. 10, 17–32 (2023). <https://doi.org/10.17083/ijsg.v10i4.637>.
11. Fairclough, S.H.: Fundamentals of physiological computing. *Interacting with Computers*. 21, 133–145 (2009). <https://doi.org/10.1016/j.intcom.2008.10.011>.
12. Nacke, L., Lindley, C.A., Lindley, C.: Flow and Immersion in First-Person Shooters: Measuring the player's gameplay experience.
13. Burak Arslan, R., Filiz, E.: Enhancement of Player Experience in Video Games Using EEG Based Dynamic Difficulty Adjustment. In: 2022 International Conference on INnovations in Intelligent SysTems and Applications (INISTA). pp. 1–4. IEEE, Biarritz, France (2022). <https://doi.org/10.1109/INISTA55318.2022.9894125>.
14. Fraser, J., Papaioannou, I., Lemon, O.: Spoken Conversational AI in Video Games: Emotional Dialogue Management Increases User Engagement. In: Proceedings of the 18th International Conference on Intelligent Virtual Agents. pp. 179–184. ACM, Sydney NSW Australia (2018). <https://doi.org/10.1145/3267851.3267896>.
15. Ruqeyya, G., Hafeez, T., Saeed, S.M.U., Ishwal, A.: EEG-based Engagement Index for Video Game Players. In: 2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE). pp.

1–6. IEEE, Lahore, Pakistan (2022).
<https://doi.org/10.1109/ETECTE55893.2022.10007386>.

Fig. 1. “Nevermind” Screenshot 7
Fig. 2. Emotiv Insight Emotional Metrics Display 8