Exploring the Advantages and Challenges of Combining ChatGPT with Blender

Exploring the Advantages and Challenges of Combining ChatGPT with Blender*

Alberto Cannavò, *Member, IEEE*, Alessandro Visconti, *Student Member, IEEE*, and Fabrizio Lamberti, *Senior Member, IEEE*

***Abstract*—Traditional graphic suites such as Blender and Autodesk Maya are widely used for 3D graphics creation, but their steep learning curve poses challenges for novice users. This complexity can make content creation time-consuming and mentally demanding. To address these issues, there is a growing interest in innovative approaches that integrate Artificial Intelligence (AI) tools into the traditional workflow. These tools promise to enhance user experience and facilitate tasks such as 3D modeling, automation, texturing, lighting setup, animation, and storytelling.**

This paper investigates the feasibility and potential strengths and weaknesses of integrating OpenAI's ChatGPT 3.5 with Blender, i.e., a prominent 3D modeling software. By combining advanced natural language processing (NLP) capabilities with the robust features of Blender, this work aims to enhance user interaction and facilitate the 3D design workflow. The paper proposes use cases that could benefit from the combination of these two technologies and warns about some limitations of this approach.

***Index Terms*—ChatGPT, Blender, 3D modeling, natural language processing, scripting**

I. Introduction

Today, graphic suites such as Blender[1] and Autodesk Maya[2] represent the most common software for creating 3D graphics assets or computer graphic animations [1]. These suites offer a range of functionalities and tools enabling users to generate entire 3D environments as well as animated scenes from scratch [2]. While these suites are generally known for their high flexibility in terms of functionalities, they are also associated with a steep learning curve [3]. The challenges in mastering such software hinder unskilled users from effectively using them [4]. Moreover, difficulties in using such software could make creating 3D content a time-intensive task and require considerable mental effort, even for professionals [5]. For this reason, it is becoming crucial to explore innovative approaches that assist users in using such software in order to simplify, e.g., 3D object modeling, automating repetitive operations, creating photorealistic textures or materials, setting up lighting, animating virtual characters, storytelling, etc. Among the potential solutions proposed in the literature, the utilization of Artificial Intelligence (AI), especially natural language processing models such as ChatGPT[3], is prominently emerging across various application domains [6]. As a matter of example, it is possible to mention tools such as Zoo Chat GPT (beta)[4] and ChatGPT for Maya [5].

This work focuses on ChatGPT 3.5, i.e., a language model developed by OpenAI based on the GPT (Generative Pre-trained Transformer) architecture [7]. Specifically, it is built on the third iteration of the GPT series. The term "chat" in its name indicates the specialization in generating human-like text responses in a conversational style. The flexibility of this model enables ChatGPT to comprehend and generate human-like text across various contexts and domains. Users can interact with ChatGPT by providing prompts or queries, and the model responds with generated text. The integration of such language models as an assisting tool within graphic suites like Blender could bring significant benefits. As a matter of example, ChatGPT's natural language processing capabilities could serve as a user-friendly interface, allowing designers to communicate their intentions to the software using human-like text. For instance, the integration between Blender and ChatGPT could simplify complex operations, such as 3D modeling, texturing, lighting adjustments, parameters optimization, or animation, by interpreting natural language commands and providing real-time guidance. Users, especially those who are novices to graphic suites, could benefit from an interactive and intuitive experience, seeking assistance, tips, or explanations. Leveraging ChatGPT's extensive language understanding has the potential to enhance user workflows, reduce learning curves, and foster a more accessible and efficient creative process.

This paper explores the possibility of integrating ChatGPT with Blender by providing a number of possible use cases that envisage the cooperation of these two technologies. The objective of this work is to enhance the user experience and assist users while operating with Blender by enabling a natural language interface for design commands and interactions within the Blender environment. Integrating these technologies may provide improvements in user efficiency, creativity, and simplification in understanding the software and its interface. As a matter of example, traditional interaction paradigms leveraged by most of the graphics suites involve the use of graphical user interfaces (GUIs), thus requiring users

*This research was developed in collaboration with VR@POLITO, and was supported by PON "Ricerca e Innovazione" 2014-2020 – DM 1062/2021 funds.

A. Cannavò, A. Visconti, and F. Lamberti are with the Politecnico di Torino, Italy (corresponding author e-mail: alberto.cannavo@polito.it).

[1]Blender: https://www.blender.org/

[2]Autodesk Maya: https://www.autodesk.com/products/maya

[3]ChatGPT: https://chat.openai.com/

[4]ZOO CHAT GPT: https://create3dcharacters.com/maya-tool-zoo-chat-gpt/

[5]ChatGPT for Maya: https://github.com/thejoltjoker/ChatGPTforMaya

to navigate through numerous menus and toolbars. Although this approach is quite effective and ensures a high level of flexibility, it may present challenges, especially for novice users. Integrating natural language interfaces can potentially address these challenges, allowing users to concentrate more on their design intent and creative process rather than on implementation details.

II. Related work

Previous research has explored solutions for facilitating the use of graphics suites with varying degrees of success. For instance, studies proposed the use of voice commands to control the 3D modeling process [8] or systems able to generate or edit 3D virtual scenes by making use of text-based interactions [9], single 2D images [10], or multi-modal inputs [11]. However, the seamless integration of a sophisticated language model, such as ChatGPT with a 3D modeling tool like Blender is relatively new and not well-explored.

One of the few examples, to the best of the authors' knowledge, is the BlenderGPT plugin[6], i.e., a Python plugin that seamlessly integrates OpenAI's GPT language model (OpenAI's GPT-4/GPT-3.5) into the Blender GUI. By leveraging the natural language processing capabilities of ChatGPT, the BlenderGPT plugin facilitates a more intuitive and interactive experience for users. More specifically, the plugin automatically generates and executes Blender Python code to make the users activate Blender's functionalities through conversational language. Sample prompts that can be issued to BlenderGPT include "*Create 100 cubes in random positions*", "*apply different scale factors to the cubes*", or "make the cube appear as glass". Besides the approach proposed by BlenderGPT, which mainly relies on the automatic generation and execution of Python scripts, language models can also be used to propose novel content and stimulate innovative ideas [12].

As a matter of example, the study in [13] investigated how ChatGPT can support developers in the design and evaluation of serious game experiences. More specifically, one of the factors highlighted in this study is the capability of ChatGPT to assist with brainstorming by suggesting game themes and mechanisms aligned with the objectives of the serious game to be developed.

Another application domain that could benefit from integrating ChatGPT with Blender (and graphics suite in general) is represented by the education field. For instance, the work in [14] investigates the application of ChatGPT to enhance the creation of educational experience within the Roblox metaverse. The principal goals of this study are to showcase how ChatGPT can simplify the design of immersive and interactive learning settings in Roblox, and to assess the effectiveness of these virtual experiences in captivating students and improving their understanding of electrical machines. The results reported in the study [14] suggest that the integration of ChatGPT into the development of Roblox-based learning experience may result in improved student engagement and understanding.

[6]BlenderGPT: https://github.com/gd3kr/BlenderGPT

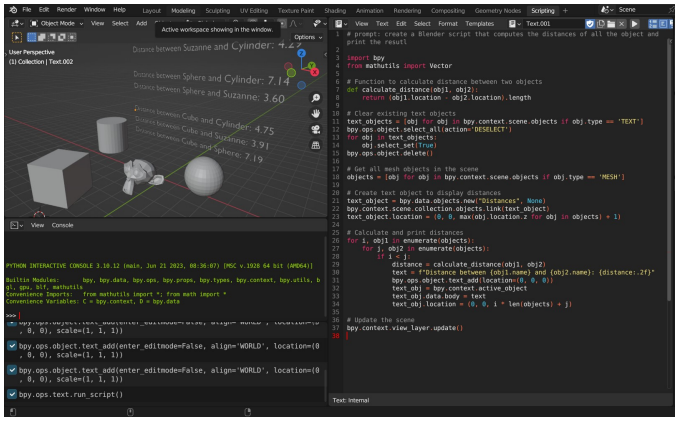


Fig. 1: The Blender scripting workspace that can be used to execute Python scripts generated by ChatGPT.

Considering the numberless benefits of integrating these two technologies, this work explores more systematically the approaches that can be used to seamlessly integrate ChatGPT into the traditional Blender workflow.

III. Methodology

In this study, several use cases have been identified that show the benefits brought by integrating ChatGPT with Blender. The use cases, detailed in the following, envisage, e.g., the possibility to enhance user interaction, creativity, and efficiency in various aspects of 3D modeling and design. The description of the use cases is complemented with sampling prompts and images that show the achieved results. These examples showcase the versatility of ChatGPT in interpreting a wide range of natural language prompts, allowing users to seamlessly interact with Blender for different purposes within the 3D modeling and design context.

A. Natural Language Design Commands & Collaborative Design

ChatGPT can be used to interpret and execute 3D design commands in Blender through natural language input, making easier and faster the design process for users. More specifically, ChatGPT can be leveraged to describe desired scenarios or environments in natural language, allowing Blender to translate these descriptions into 3D visualizations for architecture, game design, or virtual simulations.

The prompt of ChatGPT can be configured to return a Python script that can be launched in Blender's Text Editor or from the command line using Blender's Python interpreter. Once launched, the script automatically generates the 3D scene according to the issued commands (as depicted in Fig. 1).

Alternatively, the prompt can be configured to make ChatGPT directly generate the 3D model represented with an exchange file format, e.g., Wavefront (.obj and .mtl). The generated scene is usually a simplified representation of the issued prompt that can be used as a starting point for modeling

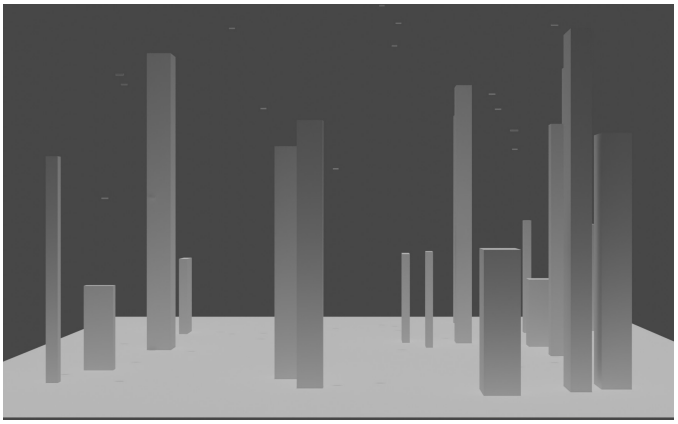


Fig. 2: Results of the prompt "*Create a 3D model of a futuristic cityscape with tall skyscrapers and flying cars in Blender*".

or as a useful visualization to foster discussions, such as scenario planning or object layout. In this respect, ChatGPT can be regarded as a useful tool to facilitate collaborative design discussions, enabling team members to discuss and refine design ideas directly within the modeling environment.

Prompt: "*Create a 3D model of a futuristic cityscape with tall skyscrapers and flying cars in Blender.*"

Fig. 2 shows the resulting rendering that is generated after launching the script generated by ChatGPT with the above prompt.

In order to instruct ChatGPT on retrieving the .obj file of a simple mesh it is possible to use the prompt: "*Generate a Wavefront .obj file representing a cube with an edge length of 3 units. Ensure that the faces of the cube have a red material assigned to them*".

B. User Guidance, Tutorials and Design Parameter Optimization

It is possible to leverage ChatGPT to provide interactive and dynamic tutorials within Blender, guiding the users through various modeling techniques, tool functionalities, and design principles.

Prompt: "*Guide me through the process of creating a realistic water simulation in Blender for an ocean scene.*"

The output of the above prompt reports two different methodologies that help the user achieve a realistic ocean scene. The first method envisioned the use of Blender's modifiers, i.e., built-in functionalities to change the geometry or properties of an object without permanently altering the base mesh. The result of the first methodology is depicted in Fig. 3. The second method guides the user through a step-by-step guide that encompasses 7 steps, i.e., set up of the initial scene, adding a plane for representing the ocean object, setting up the Blender fluid Domain and the fluid object, baking the simulation, setting up materials and lighting, rendering of the scene.

In this respect, ChatGPT can also assist in optimizing design parameters by interpreting user preferences and refining

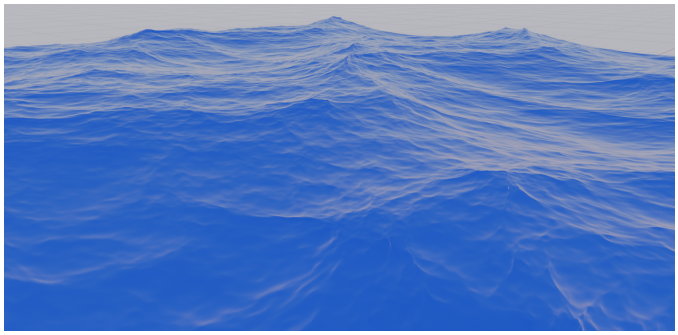


Fig. 3: Results of the prompt "*Guide me through the process of creating a realistic water simulation in Blender for an ocean scene*".

3D models based on natural language feedback, ultimately improving the iterative design process.

Prompt: "*Optimize the lighting and material parameters for a nighttime architectural visualization in Blender based on user preferences.*"

The output of ChatGPT is represented by a step-by-step guide to optimizing the lighting and material parameters for a nighttime architectural visualization in Blender. This process involves fine-tuning various settings to achieve the desired mood and aesthetics. The resulting output provides detailed descriptions and suggestions on how to tune for the following parameters: environment lighting (i.e., background, lighting intensity); artificial lighting (i.e., type and positions of the lamps, color temperature of the lights; material adjustments (i.e., emission materials; reflectivity, texture adjustments) shadows and contrast (i.e., shadow softness, contrast); atmospheric effects (i.e., fog or mist, lens flares); camera settings (i.e., the exposure and depth of field); rendering Settings (e.g., the render samples and the render resolution).

C. Idea Generation and Conceptualization

ChatGPT can be employed to assist users in brainstorming and generating creative ideas for 3D models, helping to overcome creative blocks and fostering innovation in the design process.

Prompt: "*Generate creative ideas for enhancing the interior design of a modern living room in Blender.*"

The output of the above prompt provides several indications that include: experimenting with a sophisticated color palette; designing sleek and minimalist furniture pieces to maintain a modern aesthetic; introducing unique and eye-catching lighting fixtures; mixing textures to add depth and visual interest; inserting open shelving units to showcase decorative items, books, or artwork; installing bold and contemporary artwork on the walls; adding natural elements like indoor plants, wooden accents, or stone features; creating an accent wall with textured wallpaper, a bold paint color, or even a decorative panel that can serve as a focal point; integrating smart home technology for convenience and a futuristic touch; designing and incorporating custom 3D art pieces or sculptures that give

the space a distinctive character; experiment with flooring options, such as geometric patterned tiles, hardwood floors, or a stylish rug that complements the overall design theme; ensuring that seating is not only stylish but also comfortable, like oversized sofas, plush cushions, and cozy chairs for creating a welcoming atmosphere; creating a dynamic layout by arranging furniture in unique configurations, e.g., based on asymmetric configurations to break away from traditional design norms; insert personalized items that reflect the designer personality to add a sense of individuality to the space.

D. Efficient Workflow Automation

ChatGPT can be leveraged to automate repetitive tasks in Blender by understanding and executing complex sequences of commands or providing suggestions on how to optimize the execution of the operations. In this way, the overall workflow efficiency can be enhanced.

Prompt: "*Automate the process of duplicating and arranging a series of objects along a curve in Blender.*"

The output of the above prompt provides a step-by-step guide, that assists the user in how to activate and use some Blender functionalities (e.g., Array modifier, Curve modifier, Geometry nodes) to obtain the intended result. Fig. 4 shows the nodes' architecture and the final result obtained by using the approach based on Geoemtry nodes.

E. Documentation and Annotation

Use ChatGPT to generate descriptive annotations or documentation for 3D models, automatically creating detailed textual explanations based on the visual elements present in the Blender scene.

Prompt: "*Generate detailed annotations and documentation for the components of a 3D model representing an ancient temple in Blender.*'

The output generated by ChatGPT includes annotations and documentation describing various aspects of the 3D model, such as architectural elements, materials, and historical context. More specifically, after a brief introduction, ChatGPT provides a detailed description of the main temple components, i.e., the structure of the temple, the entrance portico area, the roof and cornice details, the courtyard or altar area, the decorative elements, and the interior chambers. Then the historical context (to respect the historical accuracy) and the rendering instructions are provided.

F. Educational Simulations

Develop educational simulations within Blender by integrating ChatGPT to simulate real-world scenarios and guide users through interactive learning experiences in fields such as architecture, engineering, or biology.

Prompt: "*Create an educational simulation within Blender to demonstrate the principles of gravitational forces acting on a planetary system.*"

The output of the above prompt presents a step-by-step guide that describes how to create an educational simulation in Blender illustrating the principles of gravitational forces in a

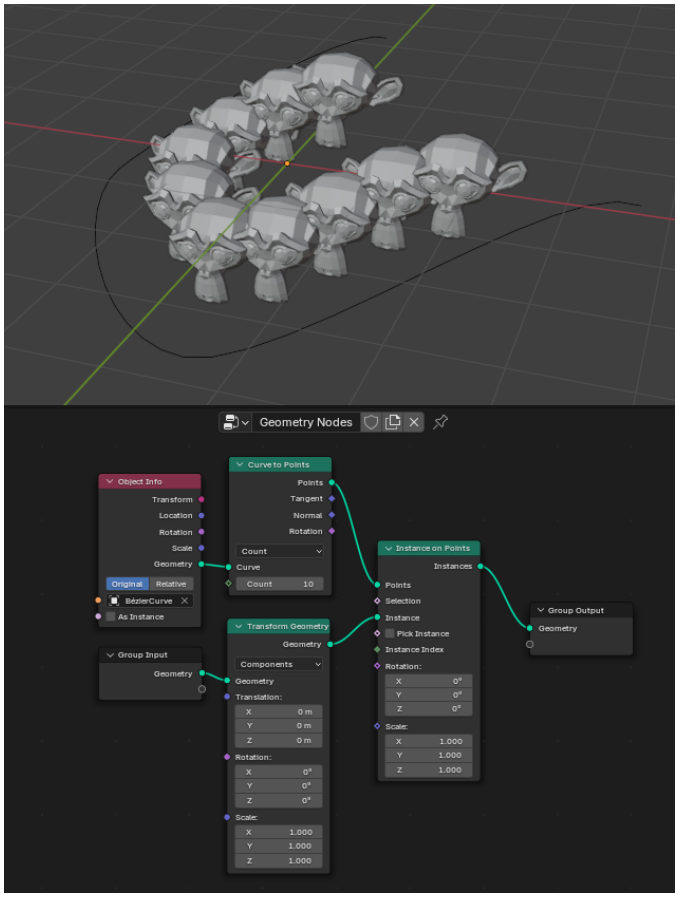


Fig. 4: Results of the prompt "*Automate the process of duplicating and arranging a series of objects along a curve in Blender.*"

planetary system. The steps include the setting up of the scene with realistic space units. Then celestial bodies, including the sun and planets, are modeled accurately, as shown in Fig. 5. Gravitational force is implemented through particle systems or constraints. The simulation is animated to showcase the orbits influenced by gravitational forces, and visual enhancements, such as moons, textures, and informative annotations, could be optionally added to increase the fidelity of the simulation. Lighting is adjusted for clarity, and optional user interaction features may be implemented for engagement. The guide also provides insights into key concepts, and user testing to ensure the accuracy and educational effectiveness of the simulation.

G. Storyboarding and Animation Scripting

Leverage ChatGPT to assist in creating storyboards and scripting animations within Blender, allowing users to describe scenes and actions in natural language for automatic translation into animation sequences.

Prompt: "*Script an animation sequence in Blender depicting a character's journey through a magical forest, including key scenes and actions.*'

With the above prompt, ChatGPT generates the animation script, titled "Enchanted Forest Adventure," that tells about

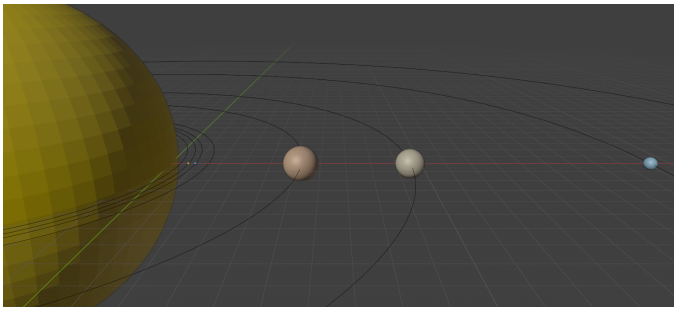


Fig. 5: Results of the prompt "*Create an educational simulation within Blender to demonstrate the principles of gravitational forces acting on a planetary system.*"

a captivating journey of a character started after his step through a magical portal into a breathtaking forest filled with mystical creatures, glowing bridges, a magical waterfall, an ancient tree grove, and a challenging enchanted maze. Each scene is carefully crafted to showcase the character's encounters and discoveries in this enchanting world, leading to a climactic revelation of a sacred grove with a powerful crystal. The character emerges from the adventure with newfound wisdom and magical artifacts, returning through the portal as the forest bathes in the morning light. The script allows for creative exploration of Blender's animation capabilities, emphasizing visual spectacle, character interactions, and the magical ambiance of the enchanted forest.

H. User Support and Troubleshooting

Implement ChatGPT as a support tool within Blender to assist users in troubleshooting issues, answering queries, and providing guidance on common challenges encountered during the modeling process.

Prompt: "*Assist the user in troubleshooting a rendering issue in Blender and provide step-by-step instructions to resolve it.*"

The resulting output with the previous prompt is a comprehensive guide that provides step-by-step instructions for diagnosing, resolving, and troubleshooting common rendering challenges in Blender. First, the issue of obtaining a black or blank render is addressed. The guide suggests ensuring proper camera visibility, checking lighting conditions, and verifying materials and textures. For grainy or noisy renders, it is suggested to increase render samples, optimize lighting, and enable denoising options. If renders take too long, optimizing render parameters, using simplified geometry, enabling denoising, and visualizing previews are the proposed solutions (as shown in Fig. 6. Another possible issue refers to objects that are missing or rendered incorrectly. In this respect, the guide indicates checking object visibility settings, adjusting camera clipping, and verifying object origins. To improve poor render quality, it is suggested to increase render resolution, adjust anti-aliasing settings, and ensure textures have sufficient resolution.

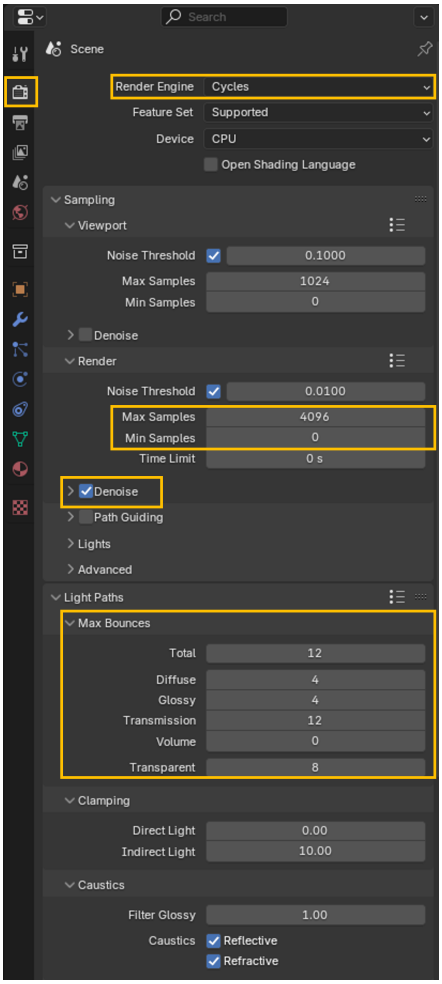
3. Long Rendering Times

Symptoms: The render takes an unusually long time to complete.

Solution:

- **Step 1**: Reduce Render Samples:
  - In the **Render Properties** panel, reduce the sample count for faster previews.
- **Step 2**: Optimize Light Paths:
  - For **Cycles**, in the **Light Paths** section, reduce the **Max Bounces** for quicker renders.
- **Step 3**: Use Simplified Geometry:
  - If using high-poly models, consider using decimated versions or lower-detail proxies during rendering.
- **Step 4**: Enable Denoising:
  - In the **Render Properties** under **Denoising**, enable denoising for faster results without sacrificing quality.
- **Step 5**: Re-render the scene (F12).

(a)

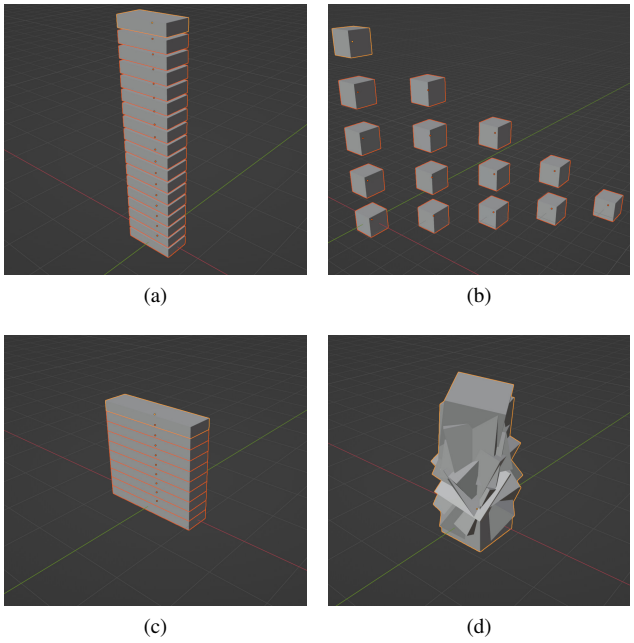


(b)

Fig. 6: Results of the prompt "*Assist the user in troubleshooting a rendering issue in Blender and provide step-by-step instructions to resolve it*": a) ChatGPT output and b) the options that are suggested to be revised to cope with long-time renderings.

TABLE I: Comparing scripts generated by the same prompts in terms of LC = Line count, FC = Function count, BW = Block width, BH = Block height, BD = Block depth, B = No. blocks, L = No. layers, CO = Clear operation, R = Randomization, IM = Imported modules, CC = Cyclomatic complexity, ND = Max nesting depth, DV = Defined variables.

	LC	FC	BW	BH	BD	B	L	CO	R	IM	CC	ND	DV
S00	42	1	1.25	0.5	0.25	54	18	Yes	No	1	4	2	16
S01	41	2	1.25	0.4	0.2	54	18	Yes	No	1	3	3	15
S02	38	2	2	1	0.3	54	18	Yes	No	1	4	3	13
S03	32	2	2	0.5	0.3	54	18	Yes	No	1	3	3	12
S04	44	0	1.25	0.4	0.2	54	18	No	No	1	4	2	16
S05	35	2	1	0.2	0.5	18	18	Yes	Yes	3	2	2	8
S06	26	1	0.5	0.5	0.5	18	18	No	No	1	2	1	10
S07	33	2	0.1	0.3	0.05	18	18	Yes	No	2	3	2	11
S08	25	2	1	0.3	0.1	18	18	Yes	No	2	3	2	7
S09	38	2	0.3	0.1	0.6	20	10	Yes	Yes	2	2	2	11
S10	36	1	0.3	0.6	0.1	54	18	No	Yes	2	3	2	14
S11	38	1	0.5	1	0.2	54	18	Yes	No	1	3	2	12
S12	32	2	0.1	0.05	0.025	30	10	No	No	1	3	3	12
S13	26	2	0.05	0.1	0.25	18	18	No	Yes	2	2	2	12
S14	41	2	0.1	0.2	0.05	18	18	Yes	No	3	3	2	10
S15	27	2	0.25	0.1	0.05	18	18	Yes	No	1	3	2	11



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

Fig. 7: Results returned by different instances of ChatGPT when issuing the prompt "*Let me know a Blender Python Script to make a tower like a Jenga*".

IV. Discussion and Limitations

Differently than BlenderGPT plugin, the prompts related to the use cases presented in this work do not require the execution of Python scripts, enabling a more creative and flexible use of the software in contexts such as brainstorming. However, a number of remarks should be made when combining Blender and ChatGPT. More specifically, the output returned by ChatGPT may change even if the same prompt is issued. For instance, Fig. 7 shows the results obtained when executing the Python script generated by different instances of ChatGPT when issuing the prompt: "*Let me know a Blender Python Script to make a tower like a Jenga*". A detailed comparison among 16 generated scripts is reported in Tab. I. More specifically, the scripts have been compared using the following objective metrics:

- Line code (LC): number of lines of code in the scprit;
- Function count (FC): number of function defined in the script;
- Dimension of the blocks: width (BW), height (BH), and depth (BD) of the generated blocks composing the Jenga tower;
- Number of blocks (B); the overall amount of blocks assembled to build the Jenga tower;
- Number of layers (L): the number of layers utilized for stacking the blocks;
- Clear operation (CO): the use of functions designed to delete pre-existing objects in the 3D scene before constructing the Jenga tower;
- Randomization (R): the use of randomized processes to assemble the tower;
- Imported modules (IM): number of modules imported by the script;
- Cyclomatic complexity (CC): the complexity of the script's control flow;
- Max nesting depth (ND): maximum number of nested structures (such as loops, conditionals, or function calls) that occur in the code of the script;
- Defined variables (DV): number of variables introduced in the script.

This behavior can hinder the integration of ChatGPT into the Blender workflow on use cases that require repeatable and uniform results. For instance, considering the use case of Educational Simulations and in particular the design of learning assessments, it should be ensured that the evaluations of the students' performance are rigorous and objective. Besides the above aspect, another possible limitation is represented by the ChatGPT's limited interaction with 3D environments which may hinder its ability to perform complex graphical operations. Moreover, ChatGPT may lack context awareness

of the specific state or content within Blender, impacting its comprehension of the 3D scene. To cope with this limitation, it is possible to provide more descriptive prompts or directly ask ChatGPT to solve the generated expectations or the observed issues. Integration complexity could also arise, since the output generally returned by ChatGPT is a Python script. This means that users need a good understanding of both natural language processing and Blender scripting. Additionally, the real-time performance of ChatGPT may be compromised depending on task complexity. ChatGPT relies on pre-existing training data, which might not cover specific terms, concepts, or workflows related to Blender. This could result in limitations when trying to generate content or answer questions specific to the Blender environment. Finally, ambiguities in natural language and potential multiple interpretations of Blender commands pose challenges in accurate response generation.

V. Conclusion

Integrating ChatGPT with Blender has the potential to revolutionize the design process, offering a more intuitive and accessible interface for users. This paper shows several use cases aimed at unlocking new possibilities in using ChatGPT together with a graphic suite such as Blender. Although this work has presented various use cases that promise to bring numerous benefits, it is also crucial to validate them through experiments. The contribution of this work serves as a starting point, but empirical testing is essential to confirm the practical applicability and real-world implications of integrating ChatGPT into the Bledner traditional workflow. Conducting experiments will not only refine and substantiate the presented ideas but will also contribute to a more robust understanding of their effectiveness in different contexts.

As a matter of example, the experiments to assess the viability of integrating ChatGPT with Blender could consider the analysis of the Command Interpretation Accuracy, i.e., the capacity of ChatGPT to interpret natural language commands related to 3D modeling tasks. In this respect, it could be studied the ability of ChatGPT to understand and execute diverse design instructions. Another aspect to be investigated could be the User Interaction Efficiency. More specifically, the efficiency of performing specific 3D modeling tasks using the traditional GUI of Blender 3.6 could be compared against a natural language interface powered by ChatGPT. To this aim, measuring the time taken by the users to accomplish the tasks, the completion rates, and user satisfaction for each method would provide useful insights.

Finally, it would be worth investigating ChatGPT's potential to provide creative suggestions during the 3D modeling process thus evaluating its ability to generate ideas, textures, or design elements based on user input.

References

[1] J. Li, A. Watkins, K. Arevalo, and M. Tovar, *Creating Games with Unity, Substance Painter, & Maya: Models, Textures, Animation, & Code*. CRC Press, 2021.

[2] B. Oliveira, D. Azulay, and P. Carvalho, "Gvrf and blender: a path for android apps and games development," in *Virtual, Augmented and Mixed Reality. Applications and Case Studies: 11th International Conference, VAMR 2019, Held as Part of the 21st HCI International Conference, HCII 2019, Orlando, FL, USA, July 26–31, 2019, Proceedings, Part II 21*. Springer, 2019, pp. 329–337.

[3] J. Lu, C. Li, C. Yin, and L. Ma, "A new framework for automatic 3d scene construction from text description," in *2010 IEEE International Conference on Progress in Informatics and Computing*, vol. 2. IEEE, 2010, pp. 964–968.

[4] M. Masoodian, A. B. M. Yusof, and B. Rogers, "Identifying problems associated with focus and context awareness in 3d modelling tasks," *Interacting with Computers*, vol. 28, no. 1, pp. 125–147, 2016.

[5] A. Cannavò, F. Lamberti *et al.*, "A virtual character posing system based on reconfigurable tangible user interfaces and immersive virtual reality," in *Proc. Smart Tools and Applications in Graphics (STAG2018)*. Eurographics, 2018, pp. 1–11.

[6] A. Nazir and Z. Wang, "A comprehensive survey of chatgpt: Advancements, applications, prospects, and challenges," *Meta-radiology*, p. 100022, 2023.

[7] K. M. Caramancion, "News verifiers showdown: a comparative performance evaluation of chatgpt 3.5, chatgpt 4.0, bing ai, and bard in news fact-checking," in *2023 IEEE Future Networks World Forum (FNWF)*. IEEE, 2023, pp. 1–6.

[8] S. Grigor, C. Nandra, and D. Gorgan, "Voice-controlled 3d modelling with an intelligent personal assistant," *Int. J. User-Syst. Interaction*, vol. 13, pp. 73–88, 2020.

[9] A. Mikaeili, O. Perel, M. Safaee, D. Cohen-Or, and A. Mahdavi-Amiri, "Sked: Sketch-guided text-based 3d editing," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 14 607–14 619.

[10] A. Cannavò, C. Bardella, L. Semeraro, F. De Lorenzis, C. Zhang, Y. Jiang, and F. Lamberti, "An automatic 3d scene generation pipeline based on a single 2d image," in *Augmented Reality, Virtual Reality, and Computer Graphics: 8th International Conference, AVR 2021, Virtual Event, September 7–10, 2021, Proceedings 8*. Springer, 2021, pp. 109–117.

[11] L. M. Seversky and L. Yin, "Real-time automatic 3d scene generation from natural language voice and text descriptions," in *Proceedings of the 14th ACM international conference on Multimedia*, 2006, pp. 61–64.

[12] D. Plate and J. Hutson, "Augmented creativity: Leveraging natural language processing for creative writing," *Art and Design Review*, 2022.

[13] W. G. Junior, E. Marasco, B. Kim, L. Behjat, and M. Eggermont, "How chatgpt can inspire and improve serious board game design," *International Journal of Serious Games*, vol. 10, no. 4, pp. 33–54, 2023.

[14] W. Ho and D. Lee, "Enhancing engineering education in the roblox metaverse: Utilizing chatgpt for game development for electrical machine course." *International Journal on Advanced Science, Engineering & Information Technology*, vol. 13, no. 3, 2023.