ISSN 1846-6168 (Print), ISSN 1848-5588 (Online)
https://doi.org/10.31803/tg-20240704100407

Original scientific paper
Received: 2024-07-04, Accepted: 2025-04-29

Accelerating Game Level Design with Machine Learning: A Unity Module for Procedural Kitchen Generation

Petar Pejić*, Jelena Pejić

Abstract: The video game industry faces significant challenges in content creation, with AAA games requiring extensive time and resources. This research addresses these challenges through the development of a Unity module for automatic 3D kitchen model generation using a Machine Learning-based Procedural Kitchen Generation (PKG) model. The module significantly reduces the time needed for game-level design, achieving designs over five times faster than traditional methods. A comparative study shows that the module produces results comparable to industry-standard tools in terms of user preference. This Unity module offers promising potential for commercial use, streamlining the design process and enhancing efficiency in game development.

Keywords: 3D Kitchen Models; Game Level Design; Machine Learning; Procedural Content Generation; Unity Module

1 INTRODUCTION

Computer games are increasingly present in our lives. Every day, hundreds of millions of players around the world are entertained by games [1]. It is expected that the total number of active video gamers worldwide reach 3.02 billion by 2029 [2]. Digital video games represent a large industry, generating sales of around $282.3 Billion worldwide in 2024, with annual growth rates of 8.76% between 2024 and 2027, resulting in a projected market volume of $363.2 Billion by 2027 [3].

AAA games typically take 18-36 months to develop and require the work of a few hundred people, including artists, designers, programmers, and audio engineers [4]. The cost of one AAA game development can exceed $500 million, including marketing budget [5]. The cost breakdown of an average AAA game shows that 40% of the budget is spent on marketing and 37% on generating content [6]. Since 2005, one of the significant challenges, besides cost, has been the difficulty in finding qualified individuals, even within the global workforce [7]. As a consequence, content production has grown to the point at which it has become a bottleneck in both game budgets and product time-to-market [8].

Solutions for this problem can be found in Procedural Content Generation [8, 9, 10, 11] which could eliminate the necessity for a human designer or artist to create that content. Humans are costly and slow, and there appears to be an increasing demand for more of them continuously [12]. Procedural Content Generation can be defined as the algorithmic generation of any content, with no, or limited human intervention. However, it is mostly applied to games, such as rules and mechanics [13, 14], levels [9, 15], music [16], vegetation [17], buildings [11, 17], terrains [19], story [20], and puzzles [21], for instance.

Level generation is one of the oldest and most popular challenges in the PCG domain [22]. Levels are essential content, as every game includes some form of spatial representation or virtual world where the player can engage in various actions [12].

Given their numerous similarities, the procedural generation of game levels can be approached through the perspective of procedural architecture [12]. Much like architecture, game level design must account for two key properties when creating virtual environments: aesthetics and functionality. The aesthetic properties of created virtual environments (geometry, texture, lights, etc.) may have a significant impact on the reality of the game. Besides visual impression, aesthetic also affect the navigation ability, for instance, a series of identical rooms can easily confuse the player. Functional properties are important for the generation of realistic virtual environment. They assume adherence to specific domain knowledge and principles during game level generation: urban design principles in the case of a city, architecture principles in the case of building generation, interior design in a case of furniture layout organization.

For the purpose of this research, we developed a Unity game engine module for automatic linear kitchen 3D model generation. While our earlier work [23] introduced the PKG model and demonstrated its effectiveness, the present study significantly extends that research by embedding the PKG model into the Unity engine, enabling real-time 3D kitchen generation and visualization. To evaluate the Unity module, we conducted a comparative analysis of model creation time between professional game level designers using standard industry tools and our module. Additionally, a perceptual study was performed using a subjective two-alternative forced-choice (2AFC) method, where participants compared kitchens generated by our Unity module to those designed by professionals. The results demonstrate that our system significantly accelerates the 3D modeling process while maintaining a high level of visual and functional quality.

The primary contribution of this research is the development and evaluation of a Unity-integrated module for automatic linear kitchen 3D model generation based on the PKG model. In contrast to the previous study that focused only on layout prediction, our new module offers full automation of 3D model generation inside an industry-standard game engine. Furthermore, the proposed solution is more than five times faster than traditional industry workflows and allows direct real-time integration into applications such as games, VR, and interior visualization tools.

2 RELATED WORKS

Regarding Procedural Content Generation, there exists an overwhelming amount of research papers [1, 10]. Most of them are applied to games' rules and mechanics [14]. Some of them address procedural level generation [22]. One group of research are addresses more specific area of game level design such as generation of buildings [11]. Another group of researchers addresses indoor space generation [24, 25].

There are several studies addressing even more specific problem inside the area of procedural interior generation, which is automatic layout generation. One group of authors [25, 26], uses rule based parametric principles for automatic layout creation. Another group of researchers [24, 23] is oriented toward ML-based algorithms for automatic layout generation.

Only a small number of studies [25, 27] address automatic kitchen layout generation, providing solutions with certain limitations. Several researchers developed a simple system for parametric 3D modeling specifically tailored for I-shaped kitchens [28]. This system was based on an idealistic interior space, disregarding the position of installations such as water pipes and canalization. Another study addresses the same problem by utilizing rule-based algorithms for automatic kitchen layout design, considering the placement of installations [25]. Due to the complexity and limitations of the approach used, the final system achieves only 69.2% accuracy in layout generation. Recent research presents an innovative Procedural Kitchen Generation (PKG) model for linear kitchen layout design, which utilizes data from existing layouts with the help of an advanced machine learning algorithm [23]. The developed PKG model, achieving an accuracy of 77.4%, surpasses the performance of the rule-based system. Consequently, we selected this model for generating layouts in our Unity module.

3 MATERIAL AND METHODS

3.1 PKG Model

The PKG model [23] serves as the core logic for our Unity module, utilizing a pipeline of machine learning classifiers for automatic linear kitchen layout generation. The model's input features include:

- Kitchen total length (mandatory),
- Ventilation position (optional),
- Sink drain position (optional),
- Window position (optional),
- Kitchen entry door position (optional), and
- Exit door position (optional).

The PKG model focuses on spatial and functional characteristics of kitchen elements, such as dimensions, placement constraints, and functional groupings. Visual properties like texture and color were intentionally excluded from the model's input features. This is because the primary objective of the PKG model is to generate structurally valid and functionally realistic layouts. Texture and color, while important for the player's visual experience, are handled separately during the rendering phase in Unity, where materials and appearances are applied to the 3D models. Therefore, the visual realism is achieved in the final visualization, not during the layout generation process.

The model outputs the types and positions of the three major elements:

- Washing (one sink),
- Cooking (one stove), and
- Storage (one refrigerator).

3.2 Unity Module

Game-level designers indicated that, in their practice, all input features except for kitchen length are redundant. The primary concern is the player's visual experience, which should convey the impression of a realistic interior [22]. For that reason, the developed Unity module uses the PKG model with only one input feature – kitchen total length. Using Unity game engine version 2019.3.4f1, a module for the automatic generation and presentation of 3D kitchen models was developed. The module pipeline is presented in Fig. 1.

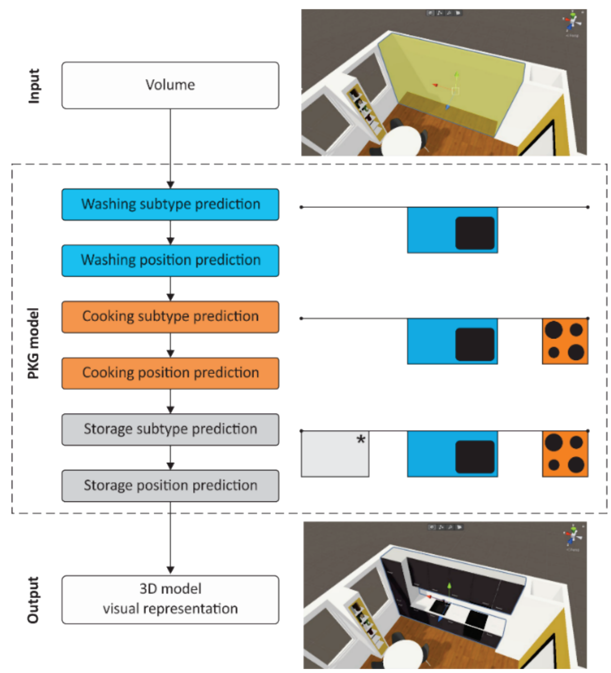

Figure 1 Unity module pipeline

The created module requires manual input of parameters inside the Unity game scene. Marking of kitchen space is done by positioning, rotating, and scaling the cube (Fig. 2a). Users can choose a kitchen design style and can see a preview of the final kitchen 3D model (Fig. 2b).

3D models of kitchen elements for the five most popular kitchen styles [30] were created using Autodesk Maya 2019 and utilized for spatial kitchen representation. These models encompass various subtypes and their respective lengths, including washing, cooking, storage, and additional working surfaces. A total of 50 kitchen element 3D models were produced, with 10 models per style. Models were exported in *.FBX file format for future use and converted inside Unity to final 3D prefabs. Their position and used subtype are determined using a service for communication with the PKG model.

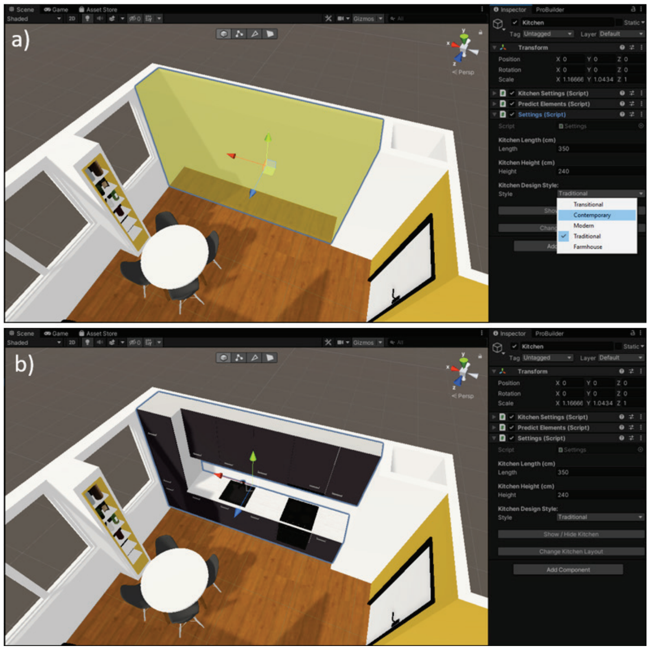

Figure 2 Unity module: a) marking of kitchen space and b) final 3D kitchen model

4 EFFICIENCY STUDY

We conducted an observational efficiency study of our Unity module by recording the time needed for professional game-level designers to create a kitchen 3D model, as well as collecting their subjective evaluations. The primary aim of this study was to explore practical usability and time savings when using our Unity module based on the PKG ML model compared to traditional pipelines.

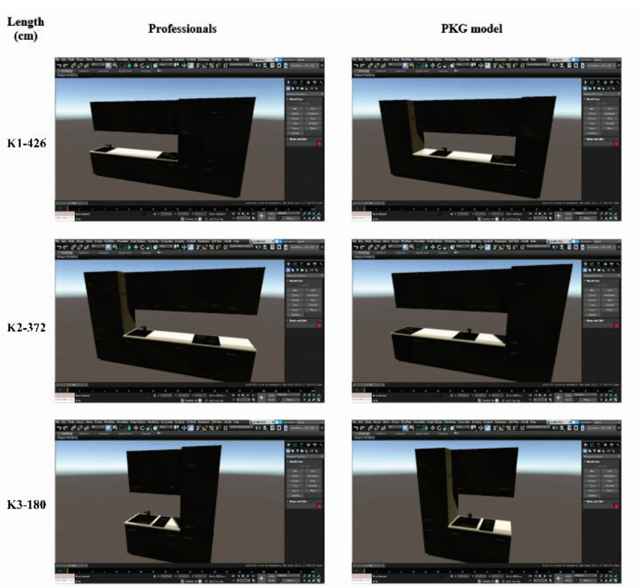

Figure 3 Final 3D models

Eight professional game-level designers participated, including six males and two females, aged 26 to 42, with an average of 11.87 years of experience in the game industry. Although the sample size is relatively small, it was composed of highly experienced professionals to ensure the credibility of the findings within the target application domain. Each participant created three different linear kitchen 3D models (lengths: K1 – 426 cm, K2 – 372 cm, and K3 – 180 cm) first using their preferred pipeline from everyday practice, and then using our Unity module. As input, they received kitchen lengths and kitchen elements in *.FBX format and as Unity prefabs, and they were instructed to present the final designs inside the Unity game engine.

Our study was performed on the same PC (Intel I7/16GB/ssd250+1T/2060gtx) equipped with all necessary software requested by game-level designers. First each of them was asked to create 3D models of three different lengths of linear kitchens using the preferred pipeline with the final result presented inside Unity. After that, they received basic instructions for Unity module use and repeated the process for the same kitchen lengths. Time was measured all 6 times and final 3D models are recorded (Fig. 3). In addition, we asked the game-level designers to give their subjective opinion regarding our Unity module and its usability in their professional work.

5 PERCEPTUAL STUDY

The perceptual experiment was conducted solely using a subjective, two-alternative forced-choice preference approach, where participants were instructed to select the image from each pair that they believed depicted a more plausible arrangement of objects in the room. Each participant completed 27 comparison tasks. 24 kitchen designs created in the "Efficiency study" (Fig. 4a), and 3 "vigilance tests" (Fig. 4b) with comparison featuring obviously wrong answers. We collected responses for each comparison from 32 users who participated in our perceptual study, discarding any responses from individuals who did not achieve 100% accuracy on the vigilance test. The test group consisted of 22 males and 10 females, aged 16 to 44.

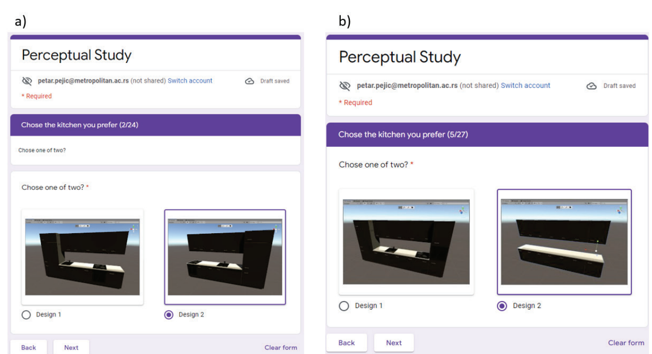

Figure 4 Perceptual study: a) efficiency study and b) vigilance tests

We aimed to determine if there was a significant difference in user preference between kitchen 3D models generated by our Unity module based on ML algorithms and those created using other game industry-standard tools and pipelines. If we found no statistically significant differences between the two methods or if our results were preferred by the users, our method could be considered successful. To validate this, we conducted an experiment using a subjective, two-alternative, forced-choice preference approach, similar to methods used in reference papers [30, 31, 32].

We conducted our user study through an online questionnaire using Google Forms. Each question presented two kitchen designs side by side for the user to compare: one created with our Unity module and the other with game industry-standard tools and pipelines. Each kitchen design was represented by a first-person perspective. Users were required to choose which of the two interior designs they would prefer to see in a game for each question. The questionnaire comprised 27 questions, with both the order of the questions and the placement (left or right) of the two compared interior designs being randomized.

6 RESULTS AND DISCUSSION

The efficiency study measured the time needed for completion of the task for each of the 8 experts (Tab. 1). Depending on their experience and preferences they chose to create a 3D kitchen model using the different pipelines. Two of them used kitchen elements directly inside Unity to create a kitchen 3D model. The rest of them used 3D modeling software to create the kitchen and then import the final 3D model into Unity. Results show significant time saving for each kitchen type using our Unity module based on the PKG ML model. In total, our module is 5.63 times faster than traditional tools and pipelines.

Table 1 Time needed for task completion (mm:ss)

		Traditional pipeline			Our Unity module		
No	Pipeline	K1	K2	K3	K1	K2	K3
1	Unity	01:32	01:52	01:59	00:24	00:22	00:22
2	Unity	01:52	02:10	02:25	00:20	00:19	00:22
3	Maya + Unity	01:52	02:10	02:25	00:28	00:26	00:28
4	3D Max + Unity	02:52	03:10	03:25	00:45	00:48	00:38
5	Maya + Unity	02:22	03:08	03:22	00:18	00:19	00:17
6	Maya + Unity	01:52	02:10	02:25	00:21	00:25	00:25
7	3D Max + Unity	02:52	03:15	03:25	00:38	00:36	00:37
8	Maya + Unity	02:02	02:53	03:21	00:20	00:20	00:19
Average		**02:09**	**02:36**	**02:51**	**00:27**	**00:27**	**00:26**
Total average			**02:32**			**00:27**	

In addition, we asked the game-level designers to give their subjective opinion regarding our Unity module and its usability in their professional work. They reported that our Unity module can be very useful for their work in the case of levels with a lot of interiors. Experts stressed that an important feature for the module to be applicable in a commercial project is a bigger kitchen elements database and the possibility for manual addition of custom-made kitchen elements. A minority of experts expressed interest in Maya and 3D Max plugins for automatic kitchen generation based on the PKG model. While some of them notice that the development of such plugins can be very helpful not only for video game developers but also for interior designers.

To evaluate the perceptual study, we employed a Chi-square nonparametric analysis, in order to identify any statistical significance across our three conditions (three different kitchens). A one-dimensional Chi-square analysis was performed separately for each condition. Each condition included 24 preference answers, which were compared against an expected frequency of 12 results. We computed the Chi-square values and tested them for significance. The results of this analysis are presented in Tab. 2.

Table 2 Perceptual study results

Kitchen	X^2-value	p-value
K1	0.302	0.583
K2	3.191	0.074
K3	0.019	0.891

Furthermore, the recorded frequencies of user preferences are presented in Fig. 5. The Chi-square analysis results indicate that there is no winner, amongst our Unity module and traditional pipelines, which would significantly outperform the other. In summary, our module achieved a higher frequency of user preference than traditional pipelines in all three conditions, but no one of them was statistically significant.

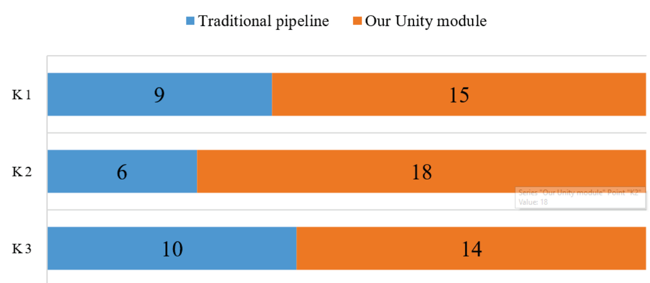

Figure 5 Frequencies of user preferences

7 CONCLUSION

The results of the efficiency and perceptual study demonstrate that our method can produce practical and livable kitchen designs that are comparable to the game industry-standard tools and pipelines. Moreover, the performance measurements show that the use of our Unity module, on average can speed up the process of 3D kitchen model creation by more than 5 times.

Using our Unity module experts need on average 27 seconds to generate a kitchen 3D model, where most of the time is spent on kitchen volumetric input and negligible time on kitchen generation using the PKG model. The fast speed of the PKG model can enable the interactive generation of VR environments during runtime, which have the potential to significantly lower the size of game levels without losing the quality of 3D models.

We find especially valuable experts' subjective opinions regarding our Unity module and its usability in their professional work. All of them found our Unity module based on the PKG model useful with potential for commercial use. The missing features they pointed out, necessary for commercial use are at the same time the tasks for future work and improvement of our Unity module.

Future work on the Unity module should firstly address the creation of a bigger kitchen elements database. Secondly, it should be upgraded to support the user-friendly addition of custom kitchen 3D elements to the database. Additionally, expanding the efficiency study with a larger and more diverse group of participants could further validate and generalize the observed benefits of the Unity module. An alternative direction of future work can be the development of a new plugin based on the PKG model for 3D modeling software such as 3D Max, Maya, and Cinema 4D.

8 REFERENCES

[1] Hendrikx, M., Meijer, S., Van der Velden, J., & Losup, A. (2013). Procedural Content Generation for Games: A Survey. *ACM Transcripts on Multimedia Computing, Communications and Applications*, 1-22. https://doi.org/10.1145/2422956.2422957

[2] Clement, J. (2024). *Number of video game users worldwide from 2019 to 2029.* New York: Statista Market Insights.

[3] Statista Inc. (2023). *Video Games: market data & analysis.* New York: Statista Market Insights.

[4] Walton, M. (2020). *Professional Techniques for Video Game Writing.* Boca Raton: CRC Press.

[5] Ivanov, M., Wittenzellner, H., & Wardaszko, M. (2019). Video Game Monetization Mechanisms in Triple A (AAA) Video Games. *Simulation Gaming Through Times and Disciplines. ISAGA 2019*, 389-404. https://doi.org/10.1007/978-3-030-72132-9_33

[6] Senchenko, A., Patterson, N., Samuel, H., & Ispir, D. (2022). SUPERNOVA: Automating Test Selection and Defect Prevention in AAA Video Games Using Risk Based Testing and Machine Learning. *2022 IEEE Conference on Software Testing, Verification and Validation (ICST).* Valencia. https://doi.org/10.1109/ICST53961.2022.00043

[7] Krueger, D., Brand, O., & Burton, D. (2005). Reinventing your company without reinventing the wheel. *Proceedings of the Game Developers Conference.*

[8] Joen, H., et al. (2023). RaidEnv: Exploring New Challenges in Automated Content Balancing for Boss Raid Games. *IEEE Transactions on Games*, 1-14.

[9] Risi, S., & Togelius, J. (2020). Increasing generality in machine learning through procedural content generation. *Nature Machine Intelligence*, 428-436. https://doi.org/10.1038/s42256-020-0208-z

[10] Amato, A. (2017). Procedural Content Generation in the Game Industry. In: Korn, O., & Lee, N. (eds). In *Game Dynamics* (pp. 15-25). Springer. https://doi.org/10.1007/978-3-319-53088-8_2

[11] Župan", R., Župan, K., Frangeš, S., & Vinković, A. (2018). Automatic procedural 3D modelling of buildings. *Tehnički glasnik, 12*(3), 166-173. https://doi.org/10.31803/tg-20180713143638

[12] Yannakakis, G., & Togelius, J. (2018). *Artificial Intelligence and Games.* Springer. https://doi.org/10.1007/978-3-319-63519-4

[13] Hom, V., & Marks, J. (2007). Automatic Design of Balanced Board Games. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 3*(1), 25-30. https://doi.org/10.1609/aiide.v3i1.18777

[14] Ruiz-Quiñones, J., & Fernández-Lei, A. (2023). Automated Videogame Mechanics Generation with XVGDL. *ICGA Journal*, 124-152. https://doi.org/10.3233/ICG-230227

[15] Morris, J. (2023). A Procedural Generation Platform to Create Randomized Gaming Maps using 2D Model and Machine Learning. *11th International Conference on Signal Image Processing and Multimedia (SIPM 2023).* Vancouver.

[16] Scirea, M., Eklund, P., Togelius, J., & Risi, S. (2018). Evolvingin-game mood - expressive music with Meta compose. *Proceedings of Audio Mostly (AM2018).* https://doi.org/10.1145/3243274.3243292

[17] Garifullin, A., Shcherbakov, A., & Frolov, V. (2022). Fitting Parameters for Procedural Plant Generation. *Computer Science Research Notes*, 282-288. https://doi.org/10.24132/CSRN.3201.35

[18] Rodrigues, F., Neto, J., & Vidal, C. (2015). Split grammar evolution for procedural modeling of virtual buildings. *XVII Symposium on Virtual and Augmented Reality.* Sao Paulo, Brazil. https://doi.org/10.1109/SVR.2015.18

[19] Ahmed, S., & Pandey, B. (2023). Procedural Terrain Generation by Sampling a 2D Monochrom Perlin Noise Map in Unity. *Asian Journal of Research in Computer Science*, 37-42. https://doi.org/10.9734/ajrcos/2023/v16i1333

[20] Cheong, Y., & Young, M. (2015). Suspenser: A Story Generation System for Suspense. *IEEE Trans. Comput. Intell. AI Games*, 39-52. https://doi.org/10.1109/TCIAIG.2014.2323894

[21] Trenquelleon, G., Choukarah, A., Roucairol , M., Addoum, M., & Cazenave, T. (2023). Procedural Generation of Rush Hour Levels. *Computers and Games. CG 2022. Lecture Notes in Computer Science vol 1386.* https://doi.org/10.1007/978-3-031-34017-8_15

[22] Khalifa, A., Perez-Liebana, D., Lucas, S., & Togelius, J. (2016). General video game level generation. *Genetic and Evolutionary Computation Conference.* https://doi.org/10.1145/2908812.2908920

[23] Pejic, J., & Pejic, P. (2022). Linear kitchen layout design via machine learning. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 1-12. https://doi.org/10.1017/S089006042100038X

[24] Wang, K., Savva, M., Chang, A., & Ritchie, D. (2018). Deep convolutional priors for indoor scene synthesis. *ACM Transactions on Graphics*. https://doi.org/10.1145/3197517.3201362

[25] Pejic, P., Mikic, M., & Milovanovic, J. (2019). Automatic Rule-based Kitchen Layout Design. *PaKSom 2019.*

[26] Stojakovic, V., Budak, I., Obradovic, R., & Koralja, D. (2017). Parametric modeling applied to the virtual reconstruction of the damaged sculpture of St John Nepomuk in Petrovaradin. *eCAADe.* Rome. https://doi.org/10.52842/conf.ecaade.2017.2.389

[27] Henderson, P., Subr, K. & Ferrari, V. (2019). Automatic Generation of Constrained Furniture Layouts. *Computer Vision and Pattern Recognition*.

[28] Pejic, P., Jovanovic, D., Marinkovic, J., & Krasic, S. (2019). Parametric 3D modeling of I-shape kitchen. *JIDEG*, 155-158.

[29] Billock, J. (2024. 5. 29.). *The 16 Most On-Trend Kitchen Styles Interior Designers Can't Get Enough Of*. (The kitchen) Retrieved 6. 12. 2024., from https://www.thekitchn.com/kitchen-styles-23661388

[30] Kan, P., & Kaufmann, H. (2018). Automatic Furniture Arrangement Using Greedy Cost Minimization. *Conference on Virtual Reality and 3D User Interfaces (VR).* https://doi.org/10.1109/VR.2018.8448291

[31] Jimenez, J., Sundstedt, V., & Gutierrez, D. (2009). Screen-space perceptual rendering of human skin. *ACM Trans. Appl. Percept*, 1-15. https://doi.org/10.1145/1609967.1609970

Authors' contacts:

Petar Pejic, Associate Professor
(Corresponding author)
Faculty of Information Technology, Belgrade Metropolitan University,
Tadeuša Košćuška 63, 11158 Belgrade, Serbia
+381 11 203 0885, petar.pejic@metropolitan.ac.rs

Jelena Pejic, Teaching Assistant
Faculty of Sciences and Mathematics, University of Niš,
Višegradska 33, 18106 Niš, Serbia
+381 18 533015, jelena.pejic@pmf.edu.rs