ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20240306001380
Original scientific paper

A Monte Carlo Simulation-Based Method for Generating Virtual Test Scenarios in Virtual Reality Environments

Mingju YAO*, Zhiyuan LI

Abstract: Virtual test scenario generation plays a crucial role in enhancing user experience and authenticity in virtual reality systems. This paper proposes a Monte Carlo simulation-based method for generating realistic virtual test scenarios by integrating various techniques, including Meanshift segmentation, depth map generation, transmittance estimation, gradient fusion, and support vector machine learning. The method effectively segments the input image, generates accurate depth maps, and optimizes the visual quality of the virtual test scene. Experimental results demonstrate the proposed method's superiority in terms of scene generation accuracy and fidelity compared to existing methods. The generated virtual test scenes exhibit minimal errors and high realism, making this method a valuable tool for virtual reality applications. Future research directions include exploring advanced techniques for further enhancing the precision and detail of the generated scenes and extending the method to other domains.

Keywords: gradient fusion method; Monte Carlo simulation; transmittance estimation method; virtual test scene generation

1 INTRODUCTION

Virtual reality technology can be used to realize the interaction between people and reality [1]. Virtual test scene modeling is very important in virtual reality system. It is found that the fidelity of scene generation directly affects the user's experience and immersion in virtual reality system. Imagine a world where car manufacturers test the safety of their latest models through harrowing weather conditions without stepping outside, where aerospace engineers trial spacecraft in the punishing cosmos without a single launch, and where game developers create immersive worlds that feel as tangible as our own-all made possible with the magic of a Monte Carlo simulation-based method in virtual reality environments. In the automotive industry, virtual test scenarios can simulate a spectrum of real-world conditions-icy roads that challenge a vehicle's traction, dense fog that tests its lighting systems, or emergency braking in wet conditions. This not only accelerates the development cycle by allowing rapid prototyping and testing but also enhances safety by reducing the need for risky physical tests. For aerospace, the stakes are even higher. Virtual testing allows for meticulous simulation of flight before an aircraft ever leaves the ground. It can replicate the void of space, the fierce re-entry heat, or the unpredictable conditions on other planets, ensuring that every contingency is planned for and every system is robust against the extraordinary. In the gaming universe, the method brings fantasies to life with an authenticity that captivates the player. Game environments undergo trials that push the boundaries of reality; battles are waged in ancient realms constructed with historical accuracy, and alien worlds are painted with the hues of physics that govern galaxies far from our own. This not only enriches the gamer's experience but also saves countless hours and resources in creating these expansive virtual worlds. Each of these applications stands as a testament to the practical significance of Monte Carlo simulation-based virtual test scenarios, offering a glimpse into a future where innovation is limitless and safety and authenticity go hand in hand. The research on virtual test scene generation method in foreign countries is earlier. Restricted by the hardware environment, the research on virtual test scene generation method in China started later. However, in recent years, with the strong support of the state, some research results have been obtained in real-time scene rendering, graphics generation, geospatial information and other aspects [2]. In the field of virtual reality, it is very important to create a realistic virtual test scene that can be displayed dynamically in real time. With the development of virtual reality technology, people's requirements for the realism of virtual test scenarios are constantly improving. Therefore, it is of great significance to study the generation method of virtual test scenarios [3].

Cai Kai [4] and his team first obtain the depth map of the scene, and input the obtained image into the full convolutional network to generate the test scene. This method has a large error in scene generation. Tian Xin [5] and his team standardized the semantic space of the scene graph through the common-sense knowledge in the external knowledge base, balanced the distribution of data in the data set, extracted the image features through the residual scrambling method, and integrated the common-sense knowledge into the image features to generate the scene graph. The scene graph generated by this method has a lot of noise, which reduces the fidelity of the scene. In order to solve the problems in the above methods, a virtual test scenario generation method based on Monte Carlo simulation is proposed.

2 LITERATURE REVIEW

With the rapid advancement of virtual reality technology, the generation of virtual test scenarios has emerged as a crucial research direction within the field. The objective is to provide more immersive and dynamic virtual environments, leading researchers to constantly pursue more effective methods for scene generation. Among these methods, the Monte Carlo simulation-based approach has gradually emerged as a research focus due to its unique advantages.

The Monte Carlo simulation is a probabilistic random sampling technique that approximates the true solution through the statistical analysis of a large number of random samples [6-9]. In the context of virtual test scenario generation, the Monte Carlo simulation can be employed

to simulate the motion and collision behavior of objects within the scene, as well as the propagation and scattering of light. The strength of this method lies in its ability to generate highly authentic dynamic scenes with exceptional controllability and flexibility.

In recent years, the Monte Carlo simulation-based approach for generating virtual test scenarios has witnessed extensive research and application. For instance, some studies have utilized the Monte Carlo simulation techniques to generate virtual scenes with intricate physical behaviors. By simulating the motion trajectories and collision responses of objects, highly immersive virtual environments have been achieved. Simultaneously, other studies have combined the Monte Carlo simulation with machine learning algorithms to optimize the scene generation process through training learning models, thus enhancing generation efficiency and quality.

Moreover, beyond its application in virtual test scenario generation, Monte Carlo simulation has also demonstrated widespread application potential in other fields. For instance, in computer graphics, Monte Carlo simulation is utilized to generate realistic lighting effects and textures; in game development, it facilitates the implementation of dynamic physical interactions and combat systems; in architecture and urban planning, it aids in simulating urban development and planning outcomes.

However, despite the progress achieved with the Monte Carlo simulation-based approach for generating virtual test scenarios, several challenges and issues still require further exploration and resolution. For instance, strategies to enhance generation efficiency improve the realism and fidelity of scenes and address the simulation of lighting and shadow effects, are areas that require additional research.

In summary, the Monte Carlo simulation-based approach for generating virtual test scenarios represents a technology with vast application potential. Future research efforts should aim to deepen our understanding and utilization of the Monte Carlo simulation, refine and optimize this method, and provide further support for the development and application of virtual reality technology.

3 RESEARCH METHODS

In order to generate virtual test scenarios to enhance user experience and increase the authenticity of virtual test environments, this article employs the following research methodology.

(1) Meanshift Image segmentation and fusion: The input image is segmented using the Meanshift segmentation method, and the image blocks obtained after segmentation are fused to mark the image regions [10]. This step aims to obtain the depth map of the virtual test scenario. Meanshift segmentation offers superior performance in handling various types of data distributions and is robust against noise. Its adaptive nature ensures optimal segmentation even in complex scenarios, outperforming conventional techniques such as thresholding or clustering.

(2) Depth map generation: By fusing the depth maps obtained above, the depth map of the virtual test scenario is obtained. This step is to further enhance the realism of the virtual test scenario. Depth map fusion enhances scene depth perception by mitigating artifacts and inconsistencies often encountered in traditional depth estimation approaches [11]. Compared to stereo matching or structure-from-motion techniques, this fusion method exhibits superior robustness in diverse environmental conditions.

(3) Transmittance estimation and gradient fusion: The transmittance estimation method is used to fuse the depth map of the virtual test scenario, and the gradient fusion method is used to fuse the normalized low-illumination image. This step aims to optimize the visual effect of the scene [12]. This approach excels in preserving scene details while effectively handling illumination variations and atmospheric effects. Unlike simplistic blending methods, such as alpha compositing, transmittance estimation coupled with gradient fusion ensures seamless integration of visual elements, resulting in visually appealing virtual environments.

(4) Introduction of support vector machine learning method: The support vector machine learning method is introduced into the scene generation process to control the scene generation process. SVM-based control offers superior adaptability and scalability, surpassing rule-based or heuristic approaches [13]. Its ability to learn from data enhances scene generation accuracy and flexibility, leading to more coherent and contextually relevant virtual scenarios.

(5) Monte Carlo simulation: The Monte Carlo simulation method is used to eliminate noise spots in the scene and optimize the generated virtual test scenario. This step aims to improve the realism and authenticity of the scene. Monte Carlo simulation excels in probabilistic modeling and statistical inference, enabling robust noise suppression and artifact elimination [14]. Unlike deterministic filtering methods, such as median or mean filtering, Monte Carlo simulation offers superior adaptability to complex noise distributions and scene dynamics, resulting in more natural and lifelike virtual environments.

Through the above steps, this research method can generate virtual test scenarios with higher authenticity and realism, thereby enhancing users' experience in virtual test scenarios.

3.1 Image Segmentation and Depth Map Acquisition

In the virtual test scene image segmentation [15], the image is divided into the sky part, the ground part and the building part. The specific process is shown in Fig. 1.

The input image is segmented through the Meanshift algorithm [16, 17] to obtain a plurality of small regions to form a set D_r . There are M image regions in the set. Suppose that the $t_m \in D_r$ representative is the m-th region, which exists in the set D_r , and the marked image is obtained through the following process:

(1) Different types of test scenes correspond to different effective line segments, and the effective line segments are used as a basis to judge the building area.

(2) Usually, the upper and lower parts of the image are the sky and the ground respectively. Therefore, search the areas with the most pixels in the upper and lower parts of the image, and mark them as the sky and the ground respectively.

The histogram of RGB color space is used as the feature vector of the image region [18], and the histogram is used to fuse multiple image regions. The gray value of the pixel is generally taken in the interval [0, 255], in which there are three color components, namely *R* component, *G* component and *B* component. The gray value of the pixel is processed by unified quantization to obtain 16 gray levels, and the color components are processed by fusion.

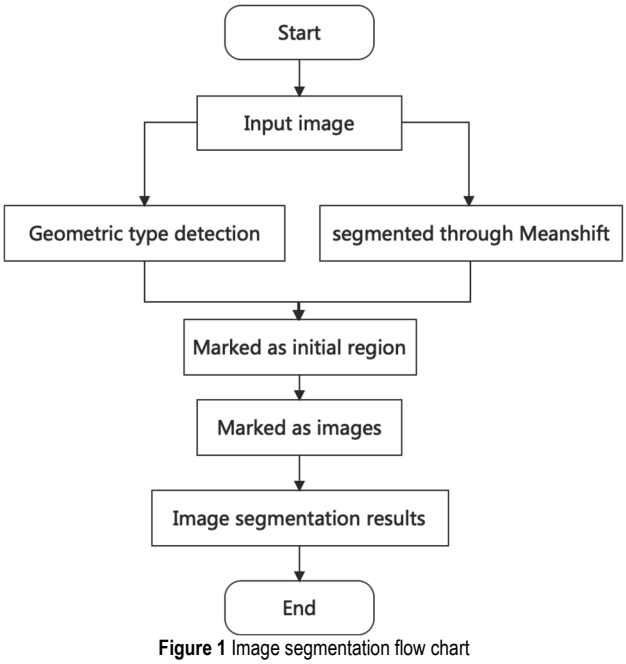

Figure 1 Image segmentation flow chart

A neighborhood set of the marked region t_j in the input image is represented by DM_i, and a region t_k with the highest similarity to the region t_j is selected from the set DM_i through the following formula:

$$t_k = \arg\max_{t_j \in DM_i} \{\sigma(t_j, t_i)\} = \arg\max_{t_j \in DM_i} \sum_{m=1}^{n} \sqrt{b^m_j b^m_i} \tag{1}$$

In the formula, b_j and b_i respectively represent the feature vectors [19] corresponding to the regions t_j and t_i. $\sigma(t_j, t_i)$ represents the Bhattacharyya coefficient, and the distance between the two regions in the image can be measured by the Bhattacharyya coefficient [20], and the larger the value of the Bhattach coefficient is, the higher the similarity between the two regions is; *m* describes the components existing in the feature vectors.

And fusing the region t_j and the region t_i to obtain a new region, obtaining the feature vector of the new region in the input image by using the formula, marking the region, repeating the process, completing the marking of the image region, and obtaining a building region, a sky region and a ground region.

After the segmentation of the input image is completed on the basis of the geometric complexity, the depth value corresponding to the image region is calculated according to the geometric characteristics, and the depth map *F* of the input image is obtained according to the depth value. According to the color characteristics of the input image, the depth values are assigned to small regions in the image to obtain a depth map F_r embodying the details of the input image, and the depth map *F* and the depth map F_r are fused to obtain a final depth map of the input images.

The depth value of the sky area is usually fixed; the depth value of the ground area in the image is determined according to the principle of from near too far, and the calculation principle of the pixel depth value in the building area is the same as that of the ground area, and the depth map F_g is obtained through the above process.

Let $v \in \{1, 2, 3, 4\}$ represent that *O* scene type of the input image, let F_c represent the standard depth map of the input image *O*, segment the image through a Meanshift algorithm to obtain a region t_m, let a_k represent the pixel points exist in the region, let $M(t_m)$ represent a number of pixel points in the region t_m, and let $F_r(a_k)$ represent a depth value corresponding to the pixel point a_k in the region t_m. It can be calculated by the following formula:

$$F_r(a_k) = \frac{\sum_{k=1}^{n} F_c(a_k)}{M(t_m)} \tag{2}$$

According to the calculation result, a depth map F_r with detail information is obtained, and the depth map F_r and the depth map F_g are input in a joint filter [21] to obtain the final depth map *F* of the input image:

$$F = \frac{F_r + F_g}{2} \tag{3}$$

3.2 Scenario Generation

The transmittance estimation method [22] is used to fuse the scene depth map obtained by the above process to construct the virtual test scene reconstruction model $O(m_1, m_2)$:

$$O(m_1, m_2) = \sum_{s_1=0}^{n} \sum_{s_2=0}^{n} O - (2m_1 + s_1, 2m_2 + s_2) \tag{4}$$

In the formula, m_1 and m_2 represent the coordinates of the pixel point in the virtual test scene; s_1 and s_2 represent the variation of the pixel point coordinates in the virtual test scene.

Match the points existing in the virtual test scene in the pixel feature matching window by using the fuzzy noise distribution area analysis method [23], and construct the following point distribution model $S(m_1, m_2)$:

$$S(m_1, m_2) = \frac{\sum_{k} T_x e}{I} \tag{5}$$

In the formula, *k* represents the gray variation of the virtual test scene; T_x represents the gray invariant function; *e* is a constant; and *I* represents the feature difference value.

On the basis of the above point distribution model $S(m_1, m_2)$, the similarity characteristic quantity $H(y_1, y_2)$ of the virtual test scene is obtained, and its expression is:

$$H(y_1, y_2) = e^{\cos \vartheta(y_1, y_2)} \tag{6}$$

where, ϑ represents the angle formed by (y_1, y_2) in the virtual reality scene.

g is used to represent a virtual visual image of a virtual test scene, a weighted control method [24] is used to calculate the row frequency and column frequency distribution $N = \{N_z\}_{z=1}^{M}$ of the pixel (a,b) in the virtual visual image *g* by adopting a weighted control method, and fuses the point distribution areas of the virtual reality scene to obtain a detail characteristic quantity $\lambda_N^A(g)$:

$$\begin{cases} \lim\limits_{A \to +\infty} \lambda_N^A(g)(a,b) = \varepsilon_N(g)(a,b) \\ \lim\limits_{A \to -\infty} \lambda_N^A(g)(a,b) = \phi_N(g)(a,b) \end{cases} \tag{7}$$

where, ε_N represents the similarity coefficient and ϕ_N represents the ambiguity coefficient.

The gradient fusion method [25] is used to carry out normalization fusion processing [26] on the low illumination image in the virtual test scene generation process, and set the following normalization probability distribution function is $\Gamma(\xi)$:

$$\Gamma(\xi) = \int_{-\infty}^{+\infty} g(x) e^{j\xi x} \tag{8}$$

where ξ represents the white spot parameter and $g(x)$ represents the average gradient coefficient.

Analyzing the change characteristics of the gradient in the virtual test scene by using the function, and calculating the pixel value $\psi(\xi)$ of the pixel in the image according to the analysis result:

$$\psi(\xi) = \ln[\Gamma(\xi)] \tag{9}$$

Setting the gray feature distribution set of *v* as $\vec{p}(v)$, and obtaining the transformation intensity feature distribution of the virtual reality test scene by an edge intensity estimation method [27].

$$|\vec{d}(v,u)| = |\vec{p}(u) - \vec{p}(v)| \tag{10}$$

In the formula, $|\vec{d}(v,u)|$ represents the absolute distance between *v* and *u*; $\vec{p}(u)$ describes the normalized canonical eigen solution corresponding to *u*, and the normalized eigen solution *K* is obtained by normalizing the estimated eigen set as follow:

$$K = \Gamma(\xi) - |\vec{d}(v,u)| \tag{11}$$

The set formed by the connected components of the virtual test scene in the connected domain is $W = \Gamma(\xi) - \vec{p}(u)$, the characteristic solution of the virtual test scene is calculated by using the minimum value, and the following similarity characteristic distribution function is constructed according to the calculation result:

$$Q = H - \Gamma(\xi) \tag{12}$$

In the background area of the generated image, the dark primary color in the image is analyzed, and for the generated low illumination image, the normalization processing is carried out. After the above processing, the image is fused. The support vector machine learning method [28] is introduced into the virtual test scene generation process, and the learning function $U(e^{j\xi}) = e^{j\Gamma(\xi)}$ is constructed in combination with the similarity feature comparison method. According to the marker analysis results, generate and reconstruct virtual test scenarios.

3.3 Scenario Optimization

The virtual test scene generated in Section 3.2 is prone to noise spots. Therefore, the Monte Carlo simulation method [29] is used to denoise the generated virtual test scene to optimize the scene.

The multiplicative noise $n(r,\omega)$ is separated from the virtual test scene image by the following logarithmic operation to obtain $f_0(r,\omega)$ which is a noise-free image:

$$\ln\{f(r,\omega)\} = \ln\{f_0(r,\omega)\} + \ln\{n(r,\omega)\} \tag{13}$$

where, $f(r,\omega)$ represents the virtual test scene image with noise.

Sample sequence xx3 corresponding to the point xx2 is extracted from the noisy image xx1, and the sequence satisfies the two-dimensional Gaussian distribution xx5 with the point xx4 as the center Extracting a sample sequence $\{f(\zeta_1), f(\zeta_2), \cdots, f(\zeta_k)\}$ corresponding to the point $\hat{f}_0(r_s, \omega_s)$ is extracted from the noisy image $f(r,\omega)$, where in the sequence satisfies a two-dimensional Gaussian distribution $P(\zeta_{r_s, w_s}(r_i, \omega_i))$ taking the point (r_s, ω_s) as the center:

$$P(\zeta_{r_s, w_s}(r_i, \omega_i)) = \frac{\exp[-(r_i - r_s)^2 + (\omega_i - \omega_s)^2] / 2\sigma^2}{2\pi\sigma^2} \tag{14}$$

where, σ represents the standard deviation.

Estimate the point $\hat{f}_0(r_s, \omega_s)$ according to the sequence $\{f(\zeta_1), f(\zeta_2), \cdots, f(\zeta_k)\}$ obtained by the above process using the Monte Carlo simulation method:

$$\hat{f}_0(\zeta_s)=\frac{\sum_{i=1}^{k}\hat{f}(\zeta_i)}{k} \tag{15}$$

ϑ represents the spatial correlation coefficient between the sample point $f(r_i,\omega_i)$ and the estimated point $\hat{f}_0(r_s,\omega_s)$, and its calculation formula is as follows:

$$\vartheta=\frac{1}{\exp\{\upsilon\sqrt{r_s^2+r_i^2-2r_sr_i\cos(\omega_s-\omega_i)}\}} \tag{16}$$

where υ represents the convergence control parameter. When the value of the spatial correlation coefficient ϑ approaches zero, it indicates that the distance between the estimation point and the sample point is large, and when the value of the spatial correlation coefficient ϑ approaches one, it indicates that the distance between the estimation point and the sample point is small.

Let $T(\zeta_i\,|\,\zeta_s)$ represent the weight factor existing between the estimation point and the sample point [30], and obtain the likelihood weighted Monte Carlo estimation value of the estimation point $\hat{f}_0(r_s,\omega_s)$ by the following formula:

$$\hat{f}_0(\zeta_s)=\frac{\sum_{i=1}^{k}T(\zeta_i\,|\,\zeta_s)\hat{f}(\zeta_i)}{\sum_{i=1}^{k}T(\zeta_i\,|\,\zeta_s)} \tag{17}$$

The natural index $\hat{f}_0(\zeta_s)$ is calculated, and the virtual test scene image is restored according to the calculation result to obtain the de-noised virtual reality scene $f_0(\zeta_s)=\exp\{\hat{f}_0(\zeta_s)\}$.

Overall，the Monte Carlo simulation-based method for generating virtual test scenarios applies multiple techniques and methods such as Meanshift Image segmentation and fusion, Depth map generation, Transmittance estimation and gradient fusion, introduction of support vector machine learning and Monte Carlo simulation. The specific generation process is shown in Fig. 2.

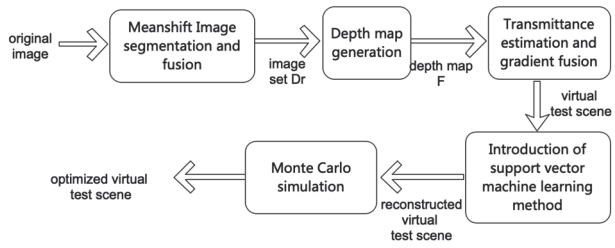

Figure 2 Virtual test scenarios generation process

4 EXPERIMENT AND ANALYSIS

In order to verify the overall effectiveness of the virtual test scenario generation method based on Monte Carlo simulation, the scenario generation test is carried out on the research method, the method in reference [4] and the method in reference [5].

4.1 Experimental Data Source and Evaluation Index

The experimental images are from the CIFAR 10 dataset (https://www.cs.toronto.edu/ ~ kriz/cifar. HTML), in which traffic test scenes are extracted. In order to ensure the accuracy of the test results, the image specifications are consistent. And MATLAB/Simulink software is used to process.

Absolute error and relative error are used as indicators to test the scene generation accuracy of different methods.

The absolute error ATE is calculated as follow:

$$ATE=\sqrt{\frac{1}{N}\sum_{i=1}^{n}[Y_i-f(x_i)]^2} \tag{18}$$

where, $f(x_i)$ represents the generated value, Y_i represents the actual value, and N represents the number of pixels.

The relative error RPE is calculated as follow:

$$\begin{cases}RPE=\sqrt{\frac{1}{N}\sum_{i=1}^{n}[E_i]^2}\\ E_i=(Q_i^{-1}Q_{i+\Delta})^{-1}(P_i^{-1}P_{i+\Delta})\end{cases} \tag{19}$$

where, Q_i represents the standard deviation and E_i: escribes the relative error corresponding to the frame i.

4.2 Analysis of Test Results

5000 images of traffic scenes from the CIFAR-10 dataset were selected. Virtual test scenes for 5 types of traffic scenarios were generated using methods from references [4] and [5], as well as the Monte Carlo testing method proposed in this paper. Finally, the absolute error, relative error, and the quality of the generated virtual test scenes from the three methods were compared.

The absolute errors of the research method, the method of reference [4] and the method of reference [5] are calculated by Eq. (18), and the results are shown in Fig. 3.

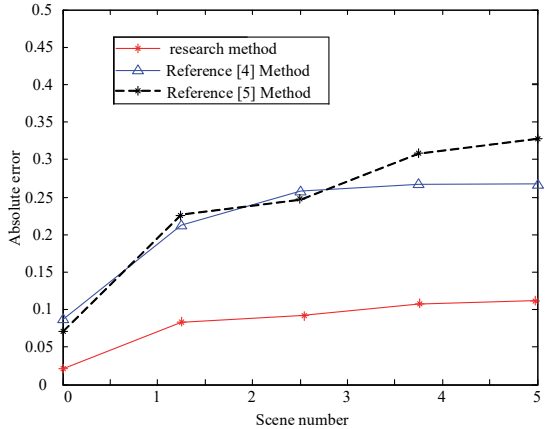

Figure 3 Absolute error of the three methods

The calculation results of the relative errors of the three methods are shown in Tab. 1.

It can be seen from the data in Fig. 2 and Tab. 1 that, compared with the method in reference [4] and the method in reference [5], the absolute error and relative error of the research method are the smallest, indicating that the scene generation accuracy of the research method is higher, because after the virtual test scene is generated, the Monte Carlo simulation method is used to optimize the scene, which improves the scene generation accuracy.

Table 1 Relative error test results

Scenes	Relative error / mm		
	Research methods	Method of reference [4]	Method of reference [5]
Scenario 1	1.2	6.8	5.5
Scene 2	0.7	5.7	2.3
Scene 3	0.9	1.2	2.8
Scene 4	1.1	3.4	1.9
Scene 5	0.5	0.8	4.6

In order to show the scene generation effect more vividly, the virtual test scene generation effect picture shown in Fig. 4 is given.

(a) Virtual test scenario generated by the research method

(b) Virtual test scenario generated by the method of [4]

(c) Virtual test scenario generated by the method of [5]

Figure 4 Virtual test scenarios generated by different methods

It can be seen from Fig. 3 that the virtual test scene generated by the research method has high definition, the virtual test scene generated by the method of reference [4] has obvious stitching traces and low fidelity, and the virtual test scenes generated by the method of reference [5] are better than that generated by reference [4], but there are still some unnatural textures. The virtual test scene generated by the research method has high fidelity and sense of reality, and can meet the needs of people for the sense of reality of the virtual test scene.

5 DISCUSSION

The Monte Carlo simulation-based method for generating virtual test scenarios introduced in this study offers a unique approach to enhancing user experience and increasing the authenticity of virtual test environments. The integration of various techniques, including Meanshift segmentation, depth map generation, transmittance estimation, gradient fusion, and support vector machine learning, demonstrate the method's versatility and adaptability to handle the complexities of virtual test scene generation.

The Meanshift segmentation technique effectively segments the input image into distinct image blocks, aiding in the identification of areas for obtaining the corresponding depth map. The depth map generation process further refines these areas, producing a highly detailed and accurate representation of the virtual test scene. The transmittance estimation method effectively fuses the depth map, enhancing its overall quality and realism.

The gradient fusion method used in this study offers an effective solution for normalizing low-light images, ensuring that the virtual test scene maintains consistent and realistic visual qualities. The integration of support vector machine learning into the scene generation process allows for greater control and adaptability, enabling the elimination of noise speckles and further optimizing the generated virtual test scene.

The Monte Carlo simulation method employed in this study serves to refine and enhance the generated virtual test scene by eliminating noise speckles and improving overall visual quality. The experimental results demonstrate the effectiveness of this research method, producing virtual test scenes with minimal generation errors and high fidelity.

Overall, this study presents a comprehensive and robust approach to generating virtual test scenarios using the Monte Carlo simulation. The integration of multiple techniques allows for the creation of highly realistic and immersive virtual test environments that can significantly enhance user experience in virtual reality systems. Overall, this study proposes a comprehensive and powerful method for generating virtual test scenarios based on the Monte Carlo simulation. The integration of multiple techniques enables the creation of highly realistic and immersive virtual test environments, significantly enhancing user experience in virtual reality systems. In the current landscape of ultra-large and hyper-realistic virtual scenes, such as autonomous driving, digital twins and metaverses, the application of the Monte Carlo simulation-based virtual test scenario generation method is particularly suitable for achieving more realistic and efficient generation of virtual test scenes. However, to apply this method in such complex and massive digital systems, further exploration and improvement are needed in the future, including the exploration of other techniques such as image segmentation, depth map generation, and noise reduction.

6 CONCLUSION

This paper presents a comprehensive and accurate method for generating virtual test scenarios using the Monte Carlo simulation. The proposed approach integrates various techniques, including Meanshift segmentation, depth map generation, transmittance estimation, gradient fusion, and support vector machine learning, to generate realistic and high-fidelity virtual test scenes. Experimental results demonstrate the superiority of the proposed method compared to existing approaches in terms of scene

generation accuracy and visual quality. The generated virtual test scenarios exhibit minimal errors and high realism, making this method a valuable tool for enhancing user experience and authenticity in virtual reality applications.

Despite its contributions, the proposed method has some limitations, such as computational complexity and potential challenges in real-time implementation. Future research directions include exploring advanced techniques, such as deep learning, to further enhance the precision and detail of the generated scenes, optimizing the method for real-time performance, and extending its application to other domains beyond virtual reality.

In conclusion, the Monte Carlo simulation-based method presented in this paper offers a promising approach for generating realistic and accurate virtual test scenarios. With further improvements and adaptations, this method has the potential to significantly advance the field of virtual reality and contribute to the development of more immersive and authentic virtual experiences.

7 REFERENCES

[1] Lei, L., Liu, J., & Yang, X. (2012). Research of the remote experiment system based on virtual reality. *Physics Procedia*, *24*, 1199-1206. https://doi.org/10.1016/j.phpro.2012.02.179

[2] Kastner, J. &Fischer, P. M. (2023). Detecting and Analyzing Fine-Grained User Roles in Social Media. *Computer Science and Information Systems*, *20*(3),1263-1287. https://doi.org/10.2298/CSIS220110006K

[3] Poth, A., Kottke, M., & Riel, A. (2023). Self-Service Kits to scale knowledge to autonomous teams - concept, application and limitations. *Computer Science and Information Systems*, *20*(1), 229-249. https://doi.org/10.2298/CSIS211112048P

[4] Cai, K., Li, X., & Tian, X. (2020). 3D Content Generation Method Based on Visual Attention Analysis. *Computer Engineering*, *46*(4), 7.

[5] Lin, X., Tian, X., Ji, Y., Xu, Y., & Liu, C. (2021). Scene Graph Generation Method Based on External Information Guidance and Residual Scrambling. *Journal of Frontiers of Computer Science & Technology*, *15*(10),1958.

[6] Wang, Y.D., Lu, X.C., Song, Y.M., Feng, Y., & Shen, J.R. (2022). Monte Carlo Tree Search improved Genetic Algorithm for unmanned vehicle routing problem with path flexibility, *Advances in Production Engineering & Management, 17*(4), 425-438. https://doi.org/10.14743/apem2022.4.446

[7] Tobisova, A., Kalavsky, P., Senova, A., & Rozenberg, R. (2024). Application of Simulation Models for Decision-Making Processes in Aviation Companies. *International Journal of Simulation Modelling, 23*(2), 299-310. https://doi.org/10.2507/IJSIMM23-2-686

[8] Yaghoubi, D., Dorodiyan, M., & Adibi, M. A. (2022). Time-Cost Estimation Probabilistic Model Using MCS in Quantitative Risk Analysis in BOT Renewable Energy Projects. *International Journal of Industrial Engineering and Management, 13*(4), 250–264. https://doi.org/10.24867/IJIEM-2022-4-317

[9] Zhang, H.Q. (2024). Optimizing obstacle avoidance path planning for intelligent mobile robots in multi-obstacle environments, *Advances in Production Engineering & Management, 19*(3), 358-370. https://doi.org/10.14743/apem2024.3.512

[10] Su, X., Wu, C., Gao, W., & Huang, W. (2020). Interior layout generation based on scene graph and graph generation model. *Design Computing and Cognition*, *20*, 267-282. https://doi.org/10.1007/978-3-030-90625-2_15

[11] Nafei, A., Huang, C., Azizi, S. P., & Chen, S. (2022). An optimized method for solving membership-based neutrosophic linear programming problems. *Studies in Informatics and Control*, *31*(4), 45-52. https://doi.org/10.24846/v31i4y202205

[12] Long, Y., Xiao, X., Shu, X., & Chen, S. (2010). Vehicle tracking method using background subtraction and meanshift algorithm. *International Conference on E-Product E-Service and E-Entertainment* , 1-4. https://doi.org/10.1109/ICEEE.2010.5661108

[13] Petrovas, A., Bausys, R., Zavadskas, E. K., & Smarandache, F. (2022). Generation of creative game scene patterns by the Neutrosophic Genetic COCOSO Method. *Studies in Informatics and Control*, *31*(4), 5-11. https://doi.org/10.24846/v31i4y202201

[14] Belean, B. (2021). Active contours driven by cellular neural networks for image segmentation in biomedical application. *Studies in Informatics and Control*, *30*(3), 109-119. https://doi.org/10.24846/v30i3y202110

[15] Song, D. T., Han, W. Z., Chen, S. H., & Qiu, Z. P. (2015). Simplified calculation of eigenvector derivatives with repeated eigenvalues. *AIAA Journal*, *34*(4), 859-862. https://doi.org/10.2514/3.13156

[16] Liu, N., Zhang, D., Xu, X., Liu, W., & Chen, L. (2017). An iterative refinement framework for image document binarization with Bhattacharyya similarity measure. *2017 14th IAPR International Conference on Document Analysis and Recognition*, 93-98. https://doi.org/10.1109/ICDAR.2017.24

[17] Alizadeh, M. & Ma, J. (2023). High-dimensional time series analysis and anomaly detection: a case study of vehicle behavior modeling and unhealthy state detection. *Advanced engineering informatics*, 57. https://doi.org/10.1016/j.aei.2023.102041

[18] Eltrass, A. & Khalil, M. (2018). An automotive radar system for multiple-vehicle detection and tracking in urban environments. *IET Intelligent Transport Systems*, *12*(8), 783-792. https://doi.org/10.1049/iet-its.2017.0370

[19] Jonsson, D., Kronander, J., Unger, J., Schon,T. B., & Wrenninge, M. (2020). Direct transmittance estimation in heterogeneous participating media using approximated taylor expansions. *IEEE Transactions on Visualization and Computer Graphics*, *28*(7), 99. https://doi.org/10.1109/TVCG.2020.3035516

[20] Yamada, R., Enomoto, Y., Watanabe, I., Nagano, K., & Kawamura, S. (2021). Reduction of quantum noise using the quantum locking with an optical spring for gravitational wave detectors. *Physics Letters A*, 402. https://doi.org/10.1016/j.physleta.2021.127365

[21] Minewaki, S., Yoshida, T., Takei, Y., Iwahashi, M., & Kiya, H. (2019). Noise bias compensation for tone mapped noisy image using prior knowledge. *APSIPA Transactions on Signal and Information Processing*, *8*(1). https://doi.org/10.1017/ATSIP.2018.29

[22] Xu, S., Ji, L., Wang, Z., Li, P., & Zhang, J. (2020). Towards reducing severe defocus spread effects for multi-focus image fusion via an optimization based strategy. *IEEE Transactions on Computational Imaging*, *6*, 1561-1570. https://doi.org/10.1109/TCI.2020.3039564

[23] Olayemi, A. D., Dharmaratne, A., & Pasha, M. F. (2020). Deep learning and late fusion technique in medical x-ray image. *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV).* https://doi.org/10.1109/ICARCV50220.2020.9305346

[24] Christensen, K. & Hebert, M. (2019). Edge-direct visual odometry. *Computer Vision and Pattern Recognition.* https://doi.org/10.48550/arXiv.1906.04838

[25] Lehtokangas, M. (2000). Pattern recognition with novel support vector machine learning method. *2000 10th European Signal Processing Conference*, *2000*, 1-4.
[26] Salehi, A., Lotfi, F. H., & Malkhalifeh, M. R. (2022). A new method for recognizing the input congestion in data envelopment analysis. *Studies in Informatics and Control*, *31*(2), 15-24. https://doi.org/10.24846/v31i2y202202
[27] Wang, Z. & Zhu, C. (2023). Application Research of Computer Virtual Reality Technology in Digital Media System Design. *2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)*, 460-465. https://doi.org/10.1109/EEBDA56825.2023.10090807
[28] Sportillo, D., Paljic, A., Boukhris, M., Fuchs, P., & Roussarie, V. (2017). An immersive Virtual Reality system for semi-autonomous driving simulation: a comparison between realistic and 6-DoF controller-based interaction. *Proceedings of the 9th international conference on computer and automation engineering*, *2*, 6-10. https://doi.org/10.1145/3057039.3057079
[29] Athilakshmi, R., Jacob, S. G., & Rajavel, R. (2023). Automatic Detection of Biomarker Genes through Deep Learning Techniques: A Research Perspective. *Studies in Informatics and Control*, *32*(2), 51-61. https://doi.org/10.24846/v32i2y202305
[30] Indriyani, T., Utoyo, M. I., & Rulaningtyas, R. (2021). A new watershed algorithm for pothole image segmentation. *Studies in Informatics and Control*, *30*(3), 131-139. https://doi.org/10.24846/v30i3y202112

Contact information:

Mingju YAO, Master Lecturer
(Corresponding author)
School of Intelligence Technology,
Geely University of China,
Chengdu, Sichuan, 641423, P. R. China,
No. 123, SEC. 2, Chengjian Avenue, Eastern New District,
Chengdu City, Sichuan Province
E-mail: 370604206@qq.com

Zhiyuan LI, Associate Professor
School of Intelligence Technology,
Geely University of China,
Chengdu, Sichuan, 641423, P. R. China,
No. 123, SEC. 2, Chengjian Avenue, Eastern New District,
Chengdu City, Sichuan Province
E-mail: lizhiyuan@guc.edu.cn