Evaluation of Illumination in 3D Scenes Based on Heat Maps Comparison

Aleksandr V. Mezhenin [1,*], Vera V. Izvozchikova [2], and Ivan A. Mezhenin [1]

[1] Faculty of Software Engineering, ITMO University, St. Petersburg, Russia; Email: ivanmezhenin@gmail.com (I.A.M.)
[2] Department of Informatics, Orenburg State University, Orenburg, Russia; Email: viza-8.11@mail.ru (V.V.I.)
*Correspondence: mejenin@mail.ru (A.V.M.)

***Abstract*—The issues of assessing the quality of lighting computer 3D scenes using different lighting systems are considered. Quality lighting increases realism, immersion and improves the perception of shape, color and texture of objects in the image. Existing engineering professional lighting calculation programs are not well suited to the design, art solutions or gaming scenes. To obtain objective estimates of illumination, we propose to use metrics for evaluating the quality of rendering systems. Particular attention is paid to the use of such tools as heat maps. Their visual analysis by hue or intensity helps to compare and evaluate the quality of illumination of scenes. However, such a comparison does not give a cumulative score. A possible solution is to treat heat maps as images and use them as the basis for a generalized heat map to produce a single cumulative statistic. In order to create a generalized heat map, several ways of constructing a difference matrix based on normalization methods have been proposed. The proposed approach is implemented as a prototype application. Experiments were carried out on test scenes with different illumination systems. The generalized heat maps made it possible to obtain cumulative estimates of the comparison of different lighting approaches and to identify areas most sensitive to changes in illumination. According to the authors, the proposed approach to illuminance estimation for staged lighting can be used to improve the realism of visualization in 3D modeling.**

***Keywords*—3D graphics, photorealistic rendering, illumination, light sources, illuminance estimation, VR, metaverse, illumination assessment, heat maps**

I. Introduction

The staging of light sources in computer graphics plays a very important role, especially for obtaining photorealistic images of simulated 3D scenes and in virtual reality systems [1]. Quality lighting increases realism, immersion, and improves the perception of shape, color and texture of the objects in the image. The use of illuminance estimation makes it possible to obtain better results. There are different methods of assessing the quality of illumination of simulated scenes—subjective and objective [2]. For objective assessments, various metrics are used—standard deviation, signal-to-noise ratio, structural similarity index (SSIM—structural similarity index metric). In addition, to measure illumination levels different tools are used in 3D modeling programs, for example Exposure Control, which is used to form a heat map to estimate illumination levels in different parts of the scene. Evaluation is made on the basis of visual comparison. However, such a comparison does not result in a cumulative assessment [3, 4]. A possible solution is to consider heat maps as images and on their basis build a generalized heat map to get a single summary statistics. This approach can serve as a basis for the development of better assessment and comparison tools. On the basis of heat maps comparison authors propose to make an evaluation of illumination in 3D scenes. Several ways of making a difference matrix based on normalization methods are proposed for creating heat maps.

The proposed approach is implemented in the form of a prototype application. Experiments were carried out on test scenes with different illumination systems. As a result of building heat maps, cumulative estimates of different illumination approaches were obtained. These estimates used to identify areas most sensitive to changes in illumination. According to the authors, the proposed approach of illuminance estimation for staging light can be used to improve the realism of visualization in 3D modeling and higher immersion in VR environment and metaverse worlds. For a more complete picture of research in this area, it is necessary to note the works on the reconstruction of 3D scenes [5–7].

The main goal of our research is to develop more efficient tools for evaluating various lighting scenarios for simulated 3D computer scenes. Such a tool can be a generalized heat map. Section II contains an overview of existing methods for assessing the quality of rendering systems, which, in the opinion of the authors, can be used to assess the quality of scene lighting. These methods are based on subjective and objective comparisons of the resulting renders. The objective assessments selected for further research are considered in more detail. Section III presents the existing metrics used in the objective methods on the basis of which it is proposed to obtain aggregate estimates of the quality of lighting. In Section IV, the authors describe the algorithm for constructing heat maps. Three options for constructing comparison matrices are considered in detail, on the basis of which it is proposed to build heat maps. These matrices are used

Manuscript received December 10, 2022; revised January 28, 2023; accepted March 6, 2023; published June 29, 2023.

doi: 10.12720/jait.14.3.601-605

to build the original heat maps and the generalized heat map. Section V presents the results of the experiments. To test the proposed algorithms, we used test scenes created in a 3D modeling program with various lighting systems. The results of comparing heat maps made it possible to obtain aggregate (cumulative) estimates of the illumination of various areas of the test scenes.

II. Related Work

The issues of assessing the quality of scene lighting considered in the paper are based primarily on works devoted to assessing the quality of rendering systems [8]. Comparison of rendering systems is most often carried out on several scenes that are selected randomly—this prevents a deep analysis of the rendering stability characteristics for various scenarios, including massive geometry, optically complex materials, complex lighting, the presence of difficult-to-compute lighting phenomena [9].

Lampel [10] conducts a comparative analysis of the Cycles and Eevee rendering systems based on the Blender 3D modeling package. For comparison, different types of scenes are used, the refraction of light by different types of materials is considered, as well as the depth of shadows and different ray tracing. In addition, the author analyzes the time spent on the visualization process. The author suggests getting acquainted with various tools that allow you to create a high-quality final image that fully matches reality.

Tao Pham in “Redshift vs Octane—Which RENDER ENGINE right for Cinema 4D?” compares the Redshift and Octane rendering systems to see the benefits of using one of these programs for Cinema 4D [4]. Below are materials comparing the rendering of various textures in the Redshift and Octane render.

The article “Evaluation of the quality of rendering systems based on automatic image analysis” considers the issues of automatic evaluation of the quality of 3D rendering (visualization) of various systems [8]. The authors propose to use the values of noise levels at different rendering iterations for progressive rendering systems. The noise level is calculated based on the statistical (root mean square) deviation of the tone levels in the image Standard Deviation (Std. Dev.) (Fig. 1). According to the authors, the proposed method for estimating the level of noise in shadows makes it possible to study and compare the results of visualization on a quantitative level and to make a certain comparison of the quality of different systems [3, 4]. However, it is not entirely fair to judge the quality of the renderer only by the noise in the shadows, in addition, this approach affects the quality of the rendering of light rays with equal settings.

In addition, the authors believe that, since it is impossible to judge the quality of the rendering of global illumination, reflections, refractions, etc. by shadows, reflections and subsurface scattering are not a rendering system. So, almost all visualizers can be considered as a kind of system consisting of a core in which light is calculated and shaders that allow one or another result to be obtained.

Subjective and objective methods of comparison. Subjective methods based on the characteristics of human perception: single stimulus—respondents are presented with one image (without a sample), double stimulus—respondents are presented with a sample and a test image. Objective methods, in turn, include: Full-Reference (FR) methods—image quality measurement with full access to the sample, Reduced-Reference (RR) methods—only incomplete information about the sample is available, No-Reference (NR)—evaluation methods without access to a sample [12, 13]. In addition, it is important to mention the comparison metrics: noise. sharpness, dynamic range, tone reproduction, contrast, color accuracy, distortion, vignetting, exposure accuracy, Lateral Chromatic Aberration (LCA). Thus, the task of setting lighting in various computer graphics systems and evaluating the results of visualizations is an urgent task.

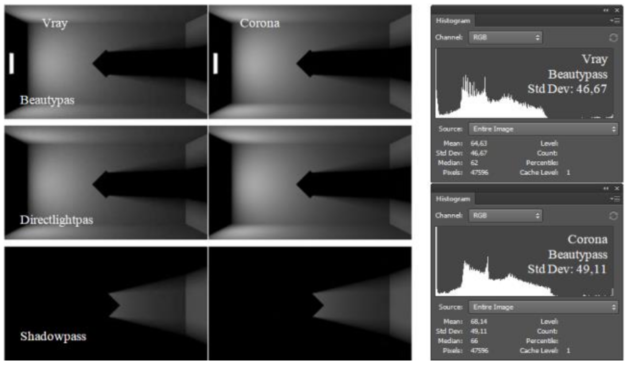

Figure 1. Noise level of Vray and Corona renderers.

III. Methods for Assessing the Visual Perception of Illumination in 3D Scenes

Let’s consider the main methods for assessing the quality of images and visualizations that can be used to assess the visual perception of the illumination of 3D scenes. One of the popular methods for obtaining an image quality score is the respondents’ rating of test scenes. The method is extremely subjective, it is difficult to determine how well the digital image processing was performed. One of the most objective and simple methods of technical assessment of image quality is the standard deviation (RMS) [14].

Take two images A and B, in this approach these images are essentially just a bunch of pixels: A = $\{a_i, i = 1, 2, \ldots, n\}$, B = $\{b_i, i = 1, 2, \ldots, n\}$, where n is the number of pixels in the image, a_i and b_i—intensity of the *i-th* pixel in the images. Moreover, if the image is an RGB image, then each pixel is a set of three numbers—the intensity of each color, and to compile the RMS, it is necessary to reduce to one number, for example, by calculating the average value using Eq. (1):

$$a_{icp} = \sum_{j=1}^{3} a_{i,j} \tag{1}$$

where $a_{i,j}$—the intensity of one of the colors (RGB). This intensity depends on the bit depth of the image, for example, if the image has a bit depth of 8 bits, then the intensity of one pixel will lie in the range from 0 to 255 (2^8–1). Using the image representation described above, the standard deviation can be determined. Let's say A is the original image, the "reference", what needs to be compared with, and B is the image that is being compared and whose quality needs to be evaluated. Then RMS is defined as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(a_i - b_i\right)^2 \tag{2}$$

With this assessment in the field of image comparison, a quality indicator is often used—the peak signal-to-noise ratio (peak signal-to-noise ratio or PSNR) [15]:

$$PSNR = 10\log_{10}\left(\frac{MAX_I^{\,2}}{MSE}\right) \tag{3}$$

where MAX_i—the maximum possible intensity value of the image pixel. However, the mutual arrangement of pixels is not taken into account in any way, and the specific size of the image is not taken into account, in fact, just sets of numbers are compared. Therefore, this method is not suitable for visualization evaluation.

The Structure Similarity Index (SSIM) method is one of the methods for measuring the similarity between two images [16, 17]. When applying the SSIM index, quality is measured based on the original image (not compressed or without distortion). This metric is more complex than Mean Square Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). Thus, the advantage of SSIM lies in the fact that the method takes into account the "perception of error" by taking into account the structural change of information. Since pixels are strongly interconnected, especially when they are spatially close, this dependence carries information about the structure of objects and about the scene as a whole.

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c1)(2\sigma_{xy} + c2)}{(\mu_x^2 + \mu_y^2 + c1)(\sigma_x^2 + \sigma_y^2 + c2)} \tag{4}$$

where μ_x—average x, μ_y—average y, σ_x^2—dispersion x, σ_y^2 —dispersion y, σ_{xy}—covariance x and y, $c_1 = (k_1L)^2$, $c_2 = (k_2L)^2$—two variables whose pixel dynamic range is $2^{(\text{bits per pixel})}$–1, k_1 and k_2 —constants.

Heat maps. A heat map is a data visualization technique that shows the scale of a phenomenon as a color in two dimensions. The hue or intensity color gamut of a heatmap can show how a certain system is grouped or changed in an image.

Fig. 2 shows the 3DS Max scene light measurement tool. Color transitions from blue to red show the illumination of objects in the scene.

A comparison of two heat maps is a comparison of two distributions [3, 4]. Thus, an alternative to this approach is to use statistical tools that calculate the difference between two distributions. The main advantages of the method of comparing two heat maps over other methods are the simplicity of calculations and brevity—a single heat map is created. A visual comparison of the ratio difference more clearly highlights sharp differences between very similar distributions, while a simple difference is more representative of overall changes. Comparison of heat maps can become the basis for the development of more advanced evaluation and comparison tools. The methods used can be expanded both in terms of usability (how good the visual representation is) and applicability to different scenarios.

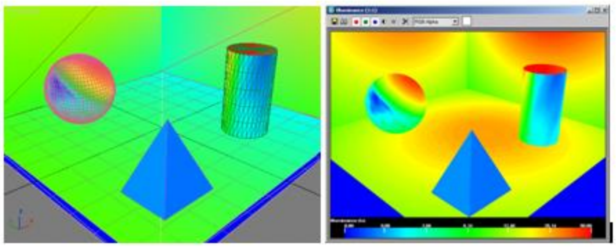

Figure 2. Light level calculation results.

IV. Proposed Approach

The read image is presented as two-dimensional arrays (matrices) of numbers. If the image is three-color (RGB-image), then it is read in the format of a three-dimensional array, which in fact is a set of three matrices. To convert an RBG image to grayscale, you need to calculate the brightness of one particular pixel in the grayscale image using Eq. (5):

$$\mathrm{Y}' = \mathrm{aR} + \mathrm{bG} + \mathrm{cB} \tag{5}$$

where R, B, G are the pixel intensities of a three-color image in the RGB palette from a three-dimensional array of intensities, a, b, c are special coefficients, the sum of which should be equal to 1. Based on the article of Sintunata and Aoki [18], the most accurate perception of the relative brightness of red, green and blue in the form of shades of gray for a person are transmitted by the following coefficients: a = 0.226, b = 0.702, c = 0.072 [19].

Now, after the transition to the mathematical space has been made and the opportunity to work with matrices has appeared, it becomes possible to compile a matrix of differences. In the simplest case, such a matrix of differences is simply a subtraction from one matrix to another. But this option is not suitable for compiling heat maps, since in such a matrix, negative (positive) values can be at a fairly large distance from each other. The formula for compiling such an n×m matrix is as follows:

$$\mathrm{D_{i,j}} = \mathrm{A_{i,j}}\text{-}\mathrm{B_{i,j}}, (\mathrm{i} = 0, \ldots, \mathrm{n}; \mathrm{j} = 0, \ldots, \mathrm{m}) \tag{6}$$

where $A_{i,j}$ and $B_{i,j}$—pixel intensities (i, j) in the first and second compared images, respectively, and D is the resulting matrix of randities, has the same shape as the original images. So, for example, if the intensity of one pixel of the image is 10, and the other 200, then their difference will be −190. In this case, if all other differences are much smaller and are in the vicinity of 0, then the image of the differences will not be indicative.

To test the program, images were used—an original photograph and an image obtained as a result of lighting correction (Fig. 3).

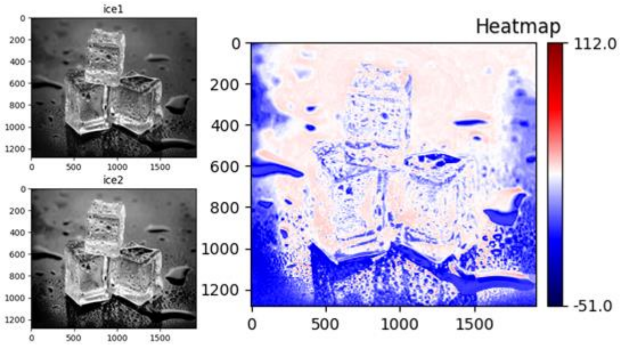

Figure 3. The result of building a heat map.

Thus, the solution proposed by the authors is based on the construction of a matrix of differences with the visualization of this matrix based on an adaptive color scale. Further, the proposed options for constructing a difference matrix are considered.

Building a matrix of differences.

A. Algorithm with Normalization Verage Deviation

This approach is one of the obvious ways to build a difference matrix with normalization. Its essence is to subtract from one matrix another modulo. Then normalization is applied in order to be able to highlight the percentage scale. In this method, normalization works like this—the maximum intensity in the resulting matrix is found, then each element of the matrix is divided by this maximum and multiplied by the image bit depth, or rather by the maximum possible intensity (for example, by 255 if the image is 8 bit).

B. Sum Matrix

The previous method has some problem, in it the normalization occurs relative to the global maximum of the entire matrix. The current method proposes to take specific pixels and normalize them with respect to their sum. Then each element of the difference matrix is divided by its corresponding element from the sum matrix. As a result, the formula for obtaining the $n \times m$ difference matrix is as follows:

$$D_{ij} = \frac{|A_{ij} - B_{ij}|}{A_{ij} + B_{ij}}, (i = 0, \dots, n; j = 0, \dots, m) \qquad (7)$$

As you can see, this method focuses on areas with a larger difference in intensities. So the darker the pixel, the smaller the value of its intensity in the matrix and, therefore, the sum is smaller, but the difference with another pixel is greater, if, on the contrary, it is light. In this approach, when comparing bright pixels with bright or dark with dark, the values in the difference matrix will be much less than if these pixels were of different brightness.

C. Relative Difference in Pixel Intensities

In this method, the pixel intensity of one image is taken and divided by the intensity of another, after which the logarithm is taken from this ratio. In this case, a situation arises when the intensity is equal to zero. To get rid of such pixels, an offset is used for both images. Offset formula:

$$Y_{ij} = \frac{X_{ij}+1}{S+M \times N} \qquad (8)$$

where X_{ij} and Y_{ij}—the intensity of one pixel, S is the sum of all image intensities, M×N is the image size. After performing the shift for both images, a matrix of differences is compiled according to the formula:

$$D_{ij} = \log\left(\frac{B_{ij}}{A_{ij}}\right), \ (i = 0, \dots, M; \ j = 0, \dots, N) \qquad (9)$$

The heat map obtained by this method (Fig. 4) is similar to that obtained by ordinary subtraction, but in this case it is used with a logarithmic scale, which indicates an increase in the order of magnitude of the difference. In this way, subtle differences between very similar images can be detected.

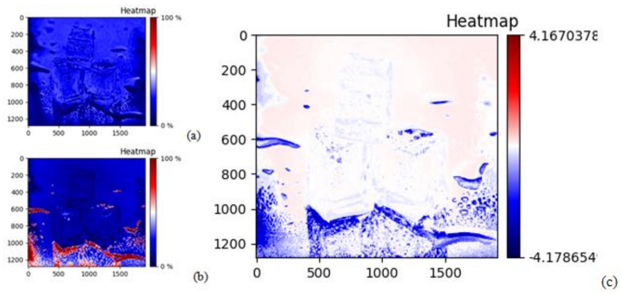

Figure 4. Algorithms with normalization (a), sum matrix (b), relative difference (c).

This section describes the algorithms, formulas and examples of the obtained heat maps for each method. The results of constructing heat maps by various methods are shown.

V. Experimental Results

To evaluate various lighting systems, a test scene was created in the Blender 3D modeling program (Fig. 5). The render without lighting is the darkest scene. We will use this image as a basis for comparison with other scenes with different lighting [20, 21].

The results are a comparison of different methods for constructing heat maps. As can be seen from the resulting heatmap, the maximum difference on the map is 95, the minimum is 0, the dark blue area indicates places where there is no lighting. The top edges of objects are the most exposed to light. As a result of comparing heat maps, a new heat map was obtained (Fig. 6). It shows which areas are more prone to changing lighting depending on the selected lighting scheme.

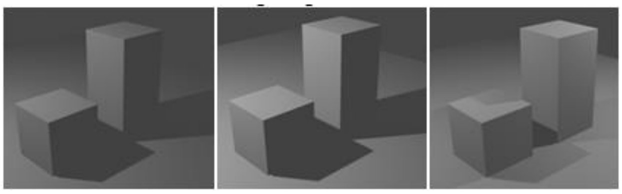

Figure 5. Test scenes.

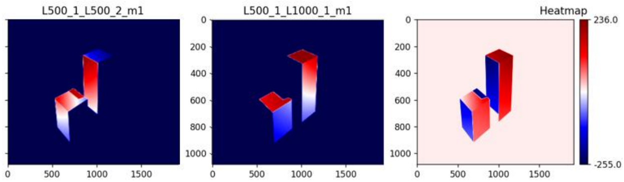

Figure 6. Results of comparison of heat maps.

Comparison of heat maps allows you to highlight areas that are most sensitive to lighting. The proposed method can be used to select the optimal lighting.

VI. Conclusion

Thus, the considered methods for estimating the illumination of 3D scenes based on the construction and comparison of thermal ones have demonstrated their operability. Various approaches for comparing heat maps are proposed and, on their basis, a prototype of a program for constructing heat maps on any images is implemented. During testing, renders were built in 3D modeling programs. At the same time, different lighting methods were used on the renders, in order to then test the developed program. As a result of building heat maps, a cumulative assessment of the comparison of different lighting approaches was obtained, and areas were found that are most sensitive to changes in lighting.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Aleksandr Mezhenin: author of the idea; Vera Izvozchikova: mathematical description of methods; Ivan Mezhenin: programming and translation of the article. All authors had approved the final version.

References

[1] P. Kán and H. Kafumann, "DeepLight: Light source estimation for augmented reality using deep learning," *The Visual Computer*, vol. 35, pp. 873–883, 2019, doi: 10.1007/s00371-019-01666-x

[2] Z. Wang, "Objective image quality assessment: Facing the real-world challenges. Image quality and system performance," *IS&T Electronic Imaging*, 2016.

[3] D. Krakov and D. G. Feitelson, "Comparing performance heatmaps," in *Proc. Job Scheduling Strategies for Parallel Processing, JSSPP 2013, Lecture Notes in Computer Scienc*e, 2013, vol. 8429.

[4] D. Krakov and D. G. Feitelson, "High-resolution analysis of parallel job workloads," in *Proc. Job Scheduling Strategies for Parallel Processing, Lect. Notes Comput. Sci.*, 2012, vol. 7698, pp. 178–195.

[5] T. Song, Z. Lyu, X. Ding, and Y. Wan, "3D surface reconstruction based on Kinect sensor," *International Journal of Computer Theory and Engineering*, vol. 5, no. 3, pp. 567–573, 2013.

[6] S. H. Kim, J. Y. Kim, and G. J. So, "Object extraction with excessive disparities in 3d stereoscopic images," *International Journal of Computer Theory and Engineering*, vol. 6, no. 4, pp. 313–318, 2014.

[7] G.-J. So, S.-H. Kim, and J.-Y. Kim, "Evaluation model of the visual fatigue on the 3D stereoscopic video," *International Journal of Computer Theory and Engineering*, vol. 8, no. 4, pp. 336–342, 2016.

[8] A. V. Mezhenin and V. A. Trushin, "Render's quality assessment of an automatic image analysis system," Scientific Works of ITMO University, 2016, pp. 119–120. (in Russian)

[9] A. Mezhenin, V. Polyakov, V. Izvozchikova, D. Burlov, and A. Zykov "The synthesis of virtual space in the context of insufficient data," *Advances in Intelligent Systems and Computing*, vol. 1127, pp. 39–46, 2020.

[10] J. Lampel. (2021). Cycles vs. Eevee—15 Limitations of real time rendering in blender. [Online]. Available: https://cgcookie.com/articles/blender-cycles-vs-eevee-15-limitations-of-real-time-rendering

[11] T. Phạm. (2021). Redshift vs Octane—Which render engine right for Cinema 4D? [Online]. Available: https://irendering.net/redshift-vs-octane-which-render-engine-right-for-cinema-4d/

[12] W. Lin and C. C. J. Kuo, "Perceptual visual quality metrics: A survey," *Journal of Visual Communication and Image Representation*, vol. 22, no. 4, pp. 297–312, 2011.

[13] Z. Wang and A. C. Bovik, "Modern image quality assessment," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 2, no. 1, pp. 1–156, 2006.

[14] V. B. Abushkevich and A. V. Mezhenin, "Research of physically correct rendering systems for object visualization tasks," Scientific Works of ITMO University, pp. 4–6, 2015. (in Russian)

[15] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. of the 20th Int. Conf. on Pattern Recognition*, 2010, pp. 2366–2369.

[16] Z. Wang and A. C. Bovik, "Modern image quality assessment," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 2, no. 1, pp. 1–156, 2006.

[17] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[18] V. Sintunata and T. Aoki, "Color segmentation based depth adjustment for 3D model reconstruction from a single input image," *International Journal of Computer Theory and Engineering*, vol. 8, no. 2, pp. 171–176, 2016.

[19] C. Saravanan, "Color image to grayscale image conversion," in *Proc. Second International Conference on Computer Engineering and Applications*, 2010, pp. 196–199, doi: 10.1109/ICCEA.2010.192

[20] A. Mezhenin, V. Izvozchikova, *et al.*, "Using virtual scenes for comparison of photogrammetry software," in *Proc. The International Symposium on Computer Science, Digital Economy and Intelligent Systems, CSDEIS 2020: Advances in Intelligent Systems, Computer Science and Digital Economics II*, pp. 57–65, 2021, doi: 10.1007/978-3-030-80478-7_7

[21] V. Polyakov and A. Mezhenin, "Procedural generation of virtual space," in *Proc. the Fourth International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'19)*, 2020, pp. 623–632.