Real-time formation of photorealistic images for architectural environment visualization in virtual reality systems

A V Chistyakov, S G Shabiev
Department of Architecture, Institute of Architecture and Construction, South Ural State University, 76, Lenin Avenue, Chelyabinsk 454080, Russia
E-mail: perfidem@list.ru

Abstract. Interactive visualization is an innovative way of presenting architectural projects, while at the same time using virtual reality systems. The aim of this study was to determine the optimal real-time rendering techniques of photorealistic images. This article is devoted to the main principles of forming physically based images and the features of rendering techniques. Comparative analysis was used to examine visualization time of images obtained by various rendering techniques. In the course of the experiment we revealed that the Image-based lighting technique together with the other methods are the best effective tools. The analysis suggests that the identified method is suitable for rendering, including physically based visualization of the architectural environment in real time and real scale and will be used for further research and software development for interactive prototyping of the architectural environment with the use of virtual reality systems.

1. Introduction
To create photorealistic physically based images, a modern architect needs to understand the main principles of the forming such images and the features of various rendering techniques. Static images of ready-made design solutions are increasingly being replaced with virtual interactive animated scenes, which make it possible to clearly visualize the project features in various conditions close to reality [1]. Therefore, it is important to choose the effective way of physically based rendering which is applicable for virtual reality systems [2].

In most cases, the process of forming a physically correct image involves the solution of several tasks: the development and preparation of scene models, the adjustment of software image generation algorithms and the solution of the rendering equation for each point of the generated raster image, taking into account all the components of the three-dimensional scene model, setting software algorithms for image enhancement (for instance, color correction and exposure, lens effects, elimination of graininess) [3]. The solution of the tasks makes the process of creating a physically correct image quite long, especially for setting software algorithms to create an image and solving the rendering equation [4] for each point of the generated raster image. Besides that, the problem is the need for large power and energy costs of the hardware (computer) to calculate the rendering equation [5].

Therefore, it is necessary to develop an optimal rendering algorithm for all existing techniques. Thus, it is also worthwhile to investigate the basic rendering equation and to find a way to reduce the
time and energy resources for calculations, taking into account the lighting factors of the microsurfaces.

The main techniques and the notion of rendering, the basic rendering equation and the aspects of creating photorealistic images using mathematical modeling of the physically based distribution of light rays on the surfaces of three-dimensional objects are considered. The modern tendencies of applying the rendering technique for architectural visualization and visualization of elements in architectural environment using modern virtual reality systems are revealed.

My article discusses the following main problems: Rendering Techniques, Basic Rendering Equation, Microsurfaces Theory. The procedure for the experiment with calculating various rendering techniques and the comparison of the results obtained are described. The tendencies for rendering techniques application in architectural visualization are revealed.

2. Methodology
The rendering process can be performed by various types of techniques. In my research work, I will focus on the most commonly used techniques.

For example, rasterization (or rasterization) is the task of taking an image described in a vector graphics format (shapes) and converting it into a raster image (pixels or dots) for output on a video display or printer, or for storage in a bitmap file format. It refers to both rasterization of models and 2D rendering primitives such as polygons, line segments, etc. Rasterization is currently the most popular technique for producing real-time 3D computer graphics. Real-time applications are used to respond immediately to user input, and generally are used to produce frame rates of at least 30 frames per second to achieve smooth animation. Compared to other rendering techniques, such as ray tracing, rasterization is extremely fast. However, rasterization is simply a process of computing the mapping from scene geometry to pixels and does not prescribe a particular way to compute the color of those pixels.

The other technique, which was considered, is Ray casting. Ray casting is the use of ray–surface intersection tests to solve a variety of problems in computer graphics and computational geometry. The principle of the technique is the following: from the virtual point of observation in the direction of the pixel of the image plane, the color of which is determined, the calculated ray is constructed. If the ray crosses the object of the three-dimensional scene, then the pixel of the image takes the corresponding color.

Ray tracing is a rendering technique for generating an image by tracing the path of light as pixels in an image plane and simulating the effects of its encounters with virtual objects. The technique can produce a very high degree of visual realism, usually higher than that of typical scanline rendering methods, but at a greater computational cost. This makes ray tracing best suitable for the applications where the image can be rendered slowly ahead of time, such as in still images, film and television visual effects, and more poorly suitable for real-time applications, like video games, where speed is critical [6]. Ray tracing can simulate a wide variety of optical effects, such as reflection and refraction, scattering, and dispersion phenomena (e.g. chromatic aberration).

Global illumination (GI) is a general name for a set of algorithms to add more realistic lighting to 3D scenes [7]. Such algorithms take into account not only the light that comes directly from a light source (direct illumination), but also subsequent cases in which light rays from the same source are reflected by other surfaces in the scene, whether reflective or not (indirect illumination) [8]. This technique allows modeling the path of light reflected from the surface which in reality is reflected not in one direction, but is scattered along the directions of the spatial hemisphere at the point of incidence, illuminating the area around. In this case, the color of the scattered light changes in accordance with the color of the surface at the point of incidence. Global Illumination is a generalized name for algorithms for solving the rendering equation that consider both the direct radiation of light sources and those reflected by other surfaces in the scene [9].

BRDF is the main tool for modeling rough surfaces with specified properties, such as: the required reflection angles, the angles of microgranhes inclination of rough surfaces [10] and their light
absorbing [11] and reflective abilities [12]. These are aspects of the microsurfaces theory that I have studied [13]. Moreover, I realized that BRDF uses different approximations that accelerate the process of solving the basic rendering equation but the image remains photorealistic [14].

Another way to simplify the solution of the basic rendering equation is Image-based lighting (IBL) [15]. It is a rendering technique which involves capturing an omnidirectional representation of real-world light information as an image typically using a specialized camera [16]. This image is then projected onto a dome or sphere analogously to environment mapping [17], and this is used to simulate the lighting for the objects in the scene. This allows highly detailed real-world lighting to be used for lighting a scene instead of trying to accurately model illumination using an existing rendering technique [18]. To reveal the features of each rendering technique an experiment was carried out. The essence of the experiment is a comparative analyzing the features of various rendering techniques comparing the time of visualizing the images for each rendering technique and comparing the photorealism of these images.

Using various rendering techniques, images of the same 3D model of the interior were made in the software package Autodesk 3DMax and their further comparison with interactive visualization – in the software package Unity3D.

The obtained data show that Rasterization and Ray casting are very fast rendering techniques, but they don’t allow you to visualize a physically correct image. On the one hand, Ray tracing and Global Illumination allow you to create photorealistic and physically correct images, on the other hand it take a lot of time and computational resources. However, the integrated use of IBL with other rendering techniques in the Unity3D software development environment allows you to render a beautiful photorealistic image with some presurmise. Various approximations of the main directions of physically based rendering are used for fast photorealistic visualization in real time [19].

As a result of the experiment, it was considered that the use of IBL together with other rendering techniques is optimal for real-time photorealistic visualization with a high frame rate, which allows using this method for interactive visualization of the architectural environment and software development based on Unity3D for virtual reality systems.

This study was done to demonstrate the principle of operation and the differences of each rendering technique and to justify the further use in the developing of software for interactive virtual prototyping of the architectural environment in virtual reality systems. 3D modeling software packages use additional plug-ins to render static images, but for real-time photorealistic interactive visualization programs based on game engines such as Unity3D are usually used [20].

During the experiment when rasterizing and re-staging the viewport window of the program and the value of Framerate were used to get the frame rendering time. Image visualization of by the technicians of raytracing and Global Illumination was carried out with the help of plugins Corona and V-Ray with the appropriate settings.

3. Results
The main result was a visual demonstrating the principles of work and the difference of each rendering technique. Also, further software development for interactive virtual prototyping the architectural environment in virtual reality systems was adopted.

Table 1. The results of rendering different techniques with the same parameters of the processor power and graphics.

| Evaluation criteria | Rasterization | Ray casting | Ray tracing | Global Illumination | Image-based lighting |
|---------------------|---------------|-------------|-------------|---------------------|---------------------|
| rendering time, sec | 0,004         | 0,016       | 480         | 1080                | 0,01                |
| photorealism, %     | 0-20          | 20-50       | 50-80       | 99                  | 80-90               |
Visualizing the Rasterization technique was carried out without considering the effect of perspective relative to the observer and did not take into account the physically correct light behavior. As a result, a simplified visualization was obtained.

The Ray casting technique doesn’t calculate new tangents of light rays. This feature did not allow accurate rendering of reflections, refractions and the natural projection of shadows.

The computational complexity of Ray tracing technique is that it solve a basic rendering equation is very correct. Thus, a high level realism of the obtained images is possible.

Global Illumination. When solving the equation of rendering, GI technique takes into account only those rays that give up the light source and are reflected diffusely for some number of times from surfaces in the scene. They came to the point of observation. GI technique allowed creating photorealistic images which can be taken as 100%. However, this technique required high computational power and time.

IBL combined with other rendering techniques allowed to visualize beautiful photorealistic images, taking minimal time and using various approximations of the basic equations rendering for photorealistic visualization in real time.

The result of the experiment is visually demonstrated in the graph below.

![Comparison of rendering techniques](image_url)

**Figure 1.** Comparison of rendering techniques at the same CPU and graphics card parameters.

The comparison shows that the use of IBL combined with other rendering techniques demonstrated the optimum values to create photorealistic real-time visualizations of high frame rate, while spending the minimum amount of time.

4. Discussions
The main result of this research work was to investigate the advantages of the IBL technique. The main features are that there is no need to solve the integral over the hemisphere of all incoming radiations of the basic rendering equation. Besides that, the data for lighting is taken from the texture map of the environment. As a result, the IBL technique together with other methods contributes to the creation of realistic images and is an effective tool for integrating computer graphics objects into virtual reality systems, including physically based 3D visualization of the architectural environment in real time and real scale.

Processes behind the result: experimental rendering of five images by different techniques, their comparison and analysis, as well as the identification of the most optimal method for physically correct 3D visualization of the architectural environment in virtual reality systems. All the results were expected. A possible explanation for this statement is that I have studied the mathematical aspects of all kinds of rendering before the experiment. Nevertheless, compared to other studies in this field, I took into account the factors necessary for virtual reality systems. However, during the experiment,
many plug-ins are not taken into account to create a physically correct rendering and other game engines. Only the most common rendering techniques are considered. The images obtained during the experiment have a size of 1280x853 pixels, in view of the limited computing capabilities of the workstation.

5. Conclusions
The main techniques and the notion of rendering, the basic rendering equation and the aspects of creating photorealistic images using mathematical modeling of the physically based distribution of light rays on the surfaces of three-dimensional objects are considered. The most optimal method of applying rendering techniques for architectural visualization and visualization of elements of the architectural environment using modern virtual reality systems is revealed.

The process of methods developing for solving the rendering equation is being constantly improved, every year the algorithms are changed, different approximations are used, but the accuracy of solving the rendering equation that has already been achieved. It is quite enough for obtaining photorealistic and physically correct images in real time. It all boils down to the fact that the power of software and hardware for images modeling and calculation are actually performed. The architecture of microprocessors is improved, which allows to perform calculations of various assignments faster using less energy. There are also new methods for visualizing projects and trends in using the modern information technologies in the construction industry and architectural design of buildings and structures. The toolkit of the architect in the information environment is being improved.

My further research work is to analyze the existing software solutions for interactive visualization of the architectural environment and to create my own software for prototyping the architectural environment in real time using virtual reality systems.

References
[1] LaValle S M 2016 Virtual reality Cambridge University Press pp 93–115
[2] Asgary Z Rahimian F P 2017 Advanced Virtual Reality Applications and Intelligent Agents for Construction Process Optimization and Defect Prevention Procedia Engineering 196 pp 1130–1137
[3] Okura F 2015 Mixed-Reality World Exploration Using Image-Based Rendering Journal on Computing and Cultural Heritage 9
[4] Kajiya J T 1986 The rendering equation Siggraph Proceedings of the 13th annual conference on Computer graphics and interactive techniques pp 143–150
[5] Ramamoorthi R 2009 Precomputation-Based Rendering Foundations and Trends in Computer Graphics and Vision pp 281–369
[6] Hwang S J 2015 A Mobile Ray Tracing Engine With Hybrid Number Representations Siggraph Asia 2015 Mobile Graphics and Interactive Applications 3
[7] Immel D S 1986 A radiosity method for non-diffuse environments Siggraph Proceedings of the 13th annual conference on Computer graphics and interactive techniques pp 133–142
[8] Schied Ch 2017 Spatiotemporal Variance-Guided Filtering Real-Time Reconstruction for Path-Traced Global Illumination Proceedings of High Performance Graphics 2 p 21
[9] Stokes W A Perceptual Illumination Components a New Approach to Efficient, High Quality Global Illumination Rendering ACM Siggraph Transactions on Graphics pp 742–749
[10] Guo Jie 2017 A Physically-based BRDF Model for Retroreflection Proceedings of the Computer Graphics International Conference 36
[11] Claustres L 2007 Wavelet Encoding of BRDFs for Real-time Rendering Proceedings of Graphics Interface 2007 pp 169–176
[12] McDermott W 2015 The Comprehensive PBR Guide vol 1 Light and Matter The theory of Physically-Based Rendering and Shading Allegorithmic innovation matters pp 2–10
[13] Park H 2015 Adaptive B-spline volume representation of measured BRDF data for photorealistic rendering Journal of Computational Design and Engineering 2 pp 1–15
[14] McDermott W 2015 The Comprehensive PBR Guide vol 2 Light and Matter Practical guidelines for creating PBR textures Allegorithmic innovation matters pp 2–28
[15] Debevec P 2016 Image-Based Lighting IEEE Computer Graphics and Applications pp 26–33
[16] Kronander J 2015 Physically Based Rendering of Synthetic Objects in Real Environments Linköping University Electronic Press pp 55–67
[17] Yang L 2011 Image-based Bidirectional Scene Reprojection Proceedings of the 2011 SIGGRAPH Asia Conference 150
[18] Poullis Ch 2004 Photogrammetric modeling and image-based rendering for rapid virtual environment creation Proceedings of the 24th Army Science Conference p 58
[19] Sloan P 2002 Precomputed Radiance Transfer for Real-Time Rendering in Dynamic Low Frequency Lighting Environments Computer Graphics Proceedings of the 29th annual conference on Computer graphics and interactive techniques pp 527–536
[20] Ren Z 2006 Real-time Soft Shadows in Dynamic Scenes using Spherical Harmonic Exponentiation ACM Siggraph Transactions on Graphics pp 977–986

Acknowledgment
The work was supported by Act 211 Government of the Russian Federation, contract no.02.A03.21.0011.