

Elaboration of Visualization Modes for Lighting Simulation in CATIA CAD System

E.D. Birukov¹, A.G. Voloboy², E.Yu. Denisov³

Keldysh Institute of the Applied Mathematics RAS, Moscow, Russia

¹ ORCID: 0000-0003-4297-6813, birukov@gin.keldysh.ru

² ORCID: 0000-0003-1252-8294, voloboy@gin.keldysh.ru

³ ORCID: 0000-0002-0614-9100, denisov@gin.keldysh.ru

Abstract

The CATIA CAD system was not initially intended for lighting simulation. However, modern engineering tasks require the lighting simulation to be a part of a new product design process. This justifies the necessity of integration of lighting simulation systems into CATIA CAD and adding special visualization modes that are initially absent in this CAD. This article describes elaboration of such visualization directly in CATIA workspace window using CATIA's native CAA RADE programming environment. We propose the visualization modes that significantly simplify preparation of initial data for lighting simulation. Three new modes are considered here: visualization of geometry tessellation in form of a wireframe triangular mesh, heat map display of the optical parameters distribution over the scene, and spherical panorama preview. Implementation of the new visualization may cause the problems for re-draw of the additional data in real time. The paper describes the proposed methods of solving or bypassing them. Using the new display modes provides a possibility of tuning some scene parameters and viewing preliminary results immediately while rendering of such scenes may take several hours. Such additional functionality can significantly improve the process of design of a new product.

Keywords: Visualization Modes, Lighting Simulation, Computer-aided Design, CATIA, Triangle Mesh, Heat Map, Spherical Panorama.

1. Introduction

Different modes of scientific visualization are deployed for interrogation of the numerical or physical experiment results as a part of visual analytics [1, 2]. They allow scientist to comprehend obtained data. However visual representation can help not only in resultant data analysis but also during initialization process of numerical experiment or specification of input data for simulation.

Lighting simulation systems allow synthesizing photorealistic images of virtual 3D scenes as well as obtaining numerical values of various optical parameters of virtual objects, with sufficient speed and accuracy [3]. Both, realistic images and numerical values are used in design of a new product. For making simulated images more realistic, such systems require high-quality and precise input data specification. The most convenient approach to address the problem is integration of lighting simulation systems with existing computer-aided design systems [4]. Such integration allows usage of ready object models which were designed at the earlier stages in the corresponding CAD systems but they should be enlarged by new attributes related to optical and lighting simulation.

CATIA system is one of the most popular CAD systems, especially in automotive and aviation industry. Physical accuracy of the simulation results is especially important for CATIA users

[4], so deep integration with this CAD is very important for engineers who design new airplanes and automobiles.

At the initial stage of the integration of optical simulation systems with existing CAD it is sufficient to develop an algorithm of data conversion from the CAD internal data format to some format suitable for processing in the lighting simulation system. However, the most convenient for end-users approach is direct integration of the user interface controls intended for lighting simulation in the native user interface of the target CAD. It allows user to consider the combined complex of the initial CAD system and lighting simulation system integrated into it as a monolith program [5]. User gets a possibility of multiple changes in the initial object model and immediate execution of lighting simulation functionality without additional actions for converting of saved files to another format and opening them in other program. Still, user may experience inconvenience if user interface of the target CAD doesn't provide some specific modes which are important for optical simulation.

There are some examples of adding new visualization modes in existent system [6, 7]. The specific of our task is that CATIA is a huge system with closed source codes. Even if you are a partner of Dassault Systemes (creator and owner of CATIA) and can use their programming system named CAA RADE you still have not access to all internal information needed for programming of your module. So programming in CATIA you have to get data in some indirect way or sometimes even to repeat data processing coded in the system but not accessible for you.

2. Specifics of integration of the lighting simulation module to the CATIA CAD system

CATIA Workspace is a document window where all scene objects are shown and which provides possibility of their visual editing. As workspace requires real-time update of the shown data during editing, usually it displays a scene with draft quality, while some details with potentially long rendering time are omitted. The CATIA CAD system provides different modes and additional possibilities for displaying scene objects in its workspace. However, CATIA system is primarily intended for geometric modeling of parts in the mechanical engineering. Creation of plug-ins for CATIA intended for some other purposes, for example, for lighting simulation tasks, requires additional display modes which are not provided by the built-in functionality of the CATIA system. In this work we describe insertion of such modes using the CATIA's own programming system – CAA RADE (integrated with Microsoft Visual Studio development environment) [8].

Among the view modes absent in the CATIA CAD system but important for lighting and optical simulation tasks most often it is required to display tessellation of geometrical objects into triangular meshes with different subdivision accuracy, as well as show optical parameters at the surfaces of the geometrical objects in the form of a heat map. Also a functionality of showing spherical panorama used as a background image directly in the CATIA workspace viewport is described in this article.

3. Displaying tessellation of CATIA geometrical objects

Geometrical objects in CATIA usually are presented by analytically defined surfaces. To be processed by computational optics, in most cases such surfaces must be tessellated into triangular mesh. Size of the obtained triangles can be different for achievement of balance between calculation speed and accuracy. Tessellation procedure may often take much time [9]. Further calculations with too accurate geometry presentation formed by very dense mesh may take even more time. So it is necessary to select the tessellation parameters which will provide satisfactory compromise between calculation accuracy and time. Besides it, it may be a problem with an object which has mostly simple shape but also contains some small important parts. After tessellation with parameters selected according to the entire object size the subdivision of mesh can be insufficient for correct displaying of small parts [10]. For verification

that tessellation parameters are appropriate it is often advisable to draw the calculated triangle mesh directly in CATIA workspace so user will be able to see if current parameters satisfy his goals or not.

We use the built-in CATIA tessellator. It has the following parameters: maximal sag, maximal step and maximal angle. Sag is maximal distance between the initial analytically defined curve and tessellated line segment. Step is a length of generated line segments relative to the initial curve length. Angle is measured between the tangents at the start and the end points of the segment.

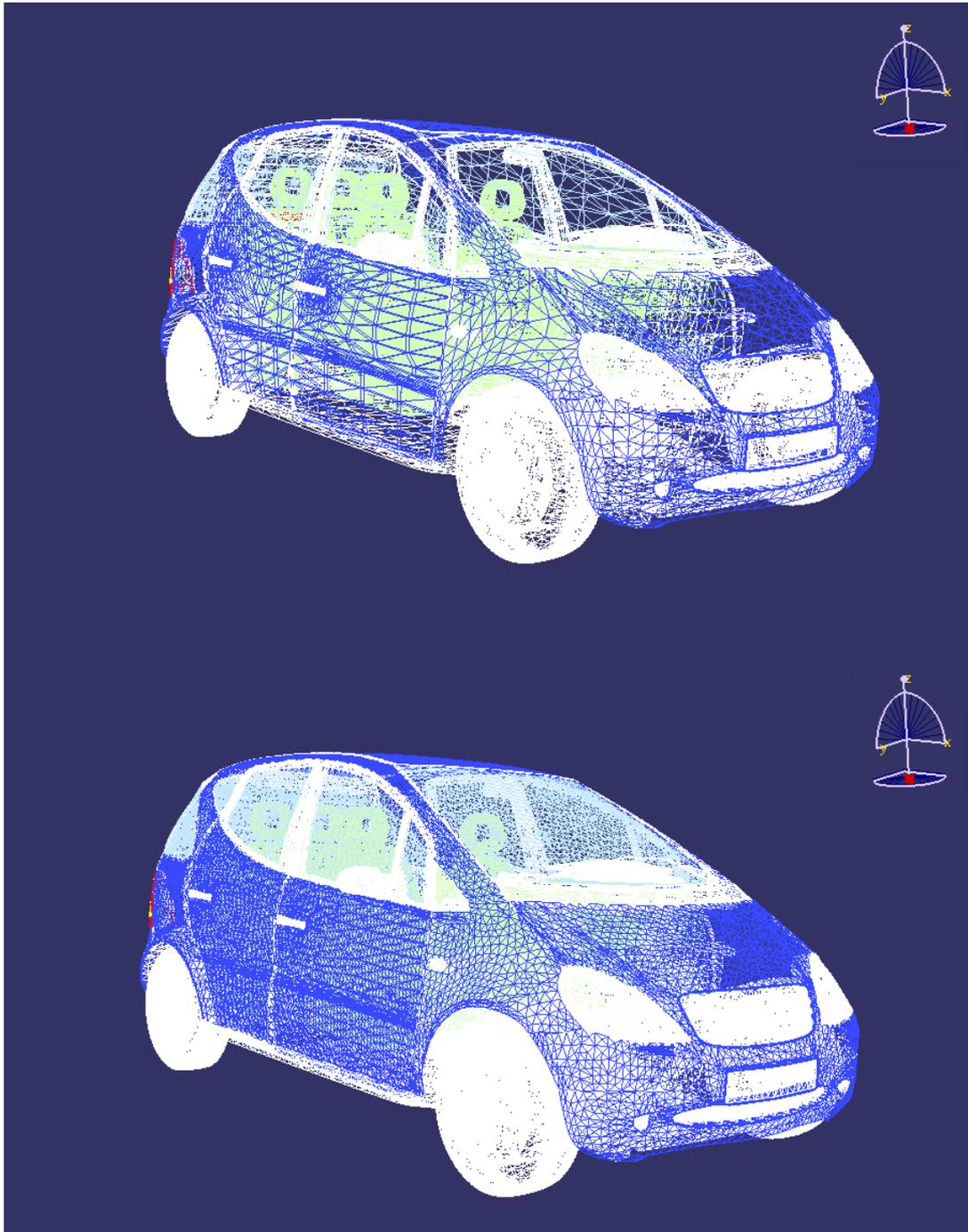


Fig. 1. Tessellated mesh shown in CATIA viewport.

The CATIA system provides a possibility of creating special temporary geometric models which are shown in its workspace but are not inserted into the virtual scene and are not saved to files. For triangular mesh display we use such temporary objects. There is also possibility to set an own viewing mode (wireframe, shaded etc.) for each object in CATIA system. So we

always show tessellation results in wireframe mode independently on the current global viewing mode for scene objects. It should be noticed that if objects in the workspace are painted in the wireframe mode and their coordinates coincide with coordinates of other objects painted in some other mode, than the first ones are not overlapped with the last ones. So there is no need in some additional manipulations with depth buffer to guarantee permanent visibility of triangle meshes in wireframe.

Viewport orientation parameters in CATIA have an influence not only to the scene objects, but to the temporary visualized objects too. So there are not needed in any additional actions to provide correspondence between the source object and its tessellated result during viewport transformations. However, in some cases the tessellated model may not correspond to the initial object. For example if the object has been changed since the last tessellation procedure. Currently we have decided not to add any automatic rebuilding of the tessellation data because this process may take significant time. So it will embarrass user's editing of the scene objects.

The CATIA viewport with tessellated mesh for two different tessellation sag values is shown on Fig. 1. Tessellation parameters at the upper part of the image are insufficient, so tessellation sag has been reduced and the new result is shown at the lower part of the figure.

4. Visualization of optical parameters as a heat map

The so-called heat map mode means graphical representation of the function of several variables. Value of the displayed quantity is expressed with color of the corresponding pixel [11]. There are different color models of heat map display. A "rainbow" is a one of the popular color models. Despite its disadvantages from the scientific point of view [12] it still is very vivid for user. In the rainbow mode parts with minimal value of the quantity have blue color, parts with maximal value have red color, and intermediate value parts have green and yellow colors. Usually heat map is just a 2-dimensional diagram, but similar color shading can be applied to any surface. Quantities which are suitable for heat map display include luminance, illuminance, radiance, irradiance and other quantities used in the optical simulation. Values of the optical quantities calculated for node points of the tessellated mesh correspond to definite colors of the heat map. Such mode is especially convenient if it is necessary to mark parts where the displayed quantity significantly excess average value.

CATIA system does not have built-in functionality for displaying heat maps. We had to develop such functionality from scratch by extending the existing display modes. It had been decided to create a special texture for scene objects and use Shading with Material for displaying it. Shading with Material mode allows displaying scene and temporary objects with applied textures directly in the workspace.

We have elaborated the method which significantly simplified applying the heat map to the meshes in the viewport. For objects displayed in the workspace we have selected the one-dimensional texture coordinates mode. Then we created a texture with single-pixel width which represents smooth color gradient corresponding to the colors of the heat map. Value of the single texture coordinates for each mesh vertex is calculated using the following formula:

$$C = \frac{Val_{cur} - Val_{min}}{Val_{max} - Val_{min}}$$

Here Val_{cur} is the luminance/illuminance value of the current vertex, Val_{max} is the maximal value presented in the scene, Val_{min} is the minimal value in the scene. So, provided that the color scale is linear, the texture pixel color at the calculated texture coordinate will correspond to the required value of the displayed quantity. The linear interpolation for pixels located between vertices with explicitly set luminance/illuminance value provides satisfactory quality of shading. The described texture is being generated automatically depending on the current luminance/illuminance data and heat map parameters.

We use only the rainbow color model for heat map display. Customizing of color models is not supported. Also there are supported only linear and logarithmic scales with equal dis-

tance between all intermediate values, without a possibility to move the intermediate values separately. These restrictions significantly simplify specifications of the heat map mode parameters while provided functionality is sufficient for the CATIA users. A dialog shown on Fig. 2 was provided to set the parameters. It is opened automatically when heat map display is turned on.

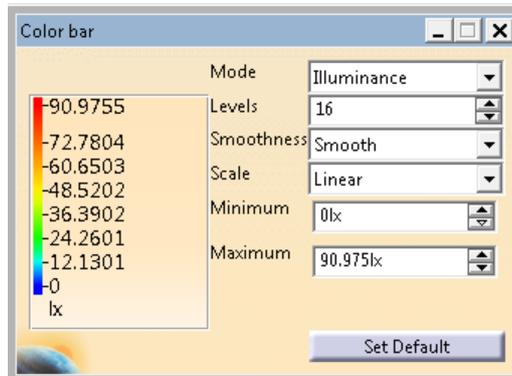


Fig. 2. Heat map parameters dialog

This dialog provides setting the following parameters:

- Current phenomenon for showing (luminance or illuminance).
- Minimal and maximal values. Areas with values below minimal are painted with color corresponding to minimal value (currently it is blue) and areas with values over maximal are painted with color corresponding to maximal (currently it is red).
- Smoothness. It can be smooth or discrete. In the Discrete mode there are sharp borders between areas painted with different colors, and number of intermediate colors is equal to number of levels.
- Number of intermediate levels. If it is less than 8, the real color model will differ from the ideal rainbow model.
- Scale (can be linear or logarithmic).

Applying the generated textures directly to the CATIA scene objects is hardly possible. The tree of representations of CATIA scene objects can have complicated structure which does not always correspond to tree structure of the scene objects themselves. Besides this, we use tessellation (conversion of initial CATIA scene objects to triangle meshes according to specified parameters). Thus the calculated luminance/illuminance values significantly depend on these parameters because tessellated meshes may have noticeable difference from initial scene objects. That is why we display heat map upon the additionally created temporary geometrical meshes similar to the mesh demonstrating tessellation result.

Another problem appears there – it is necessary to display these newly created objects in front of the initial CATIA scene objects. Because we were unable to manipulate the depth buffer of the CATIA main viewport, we solved this problem by hiding the initial geometry in the workspace. To do so, we obtained pointer to the graphic representation (a CATRep object) [8] corresponding to the scene root object and changed its view mode to wireframe. This forcedly set new view mode bypasses the view mode previously set by user with the UI.

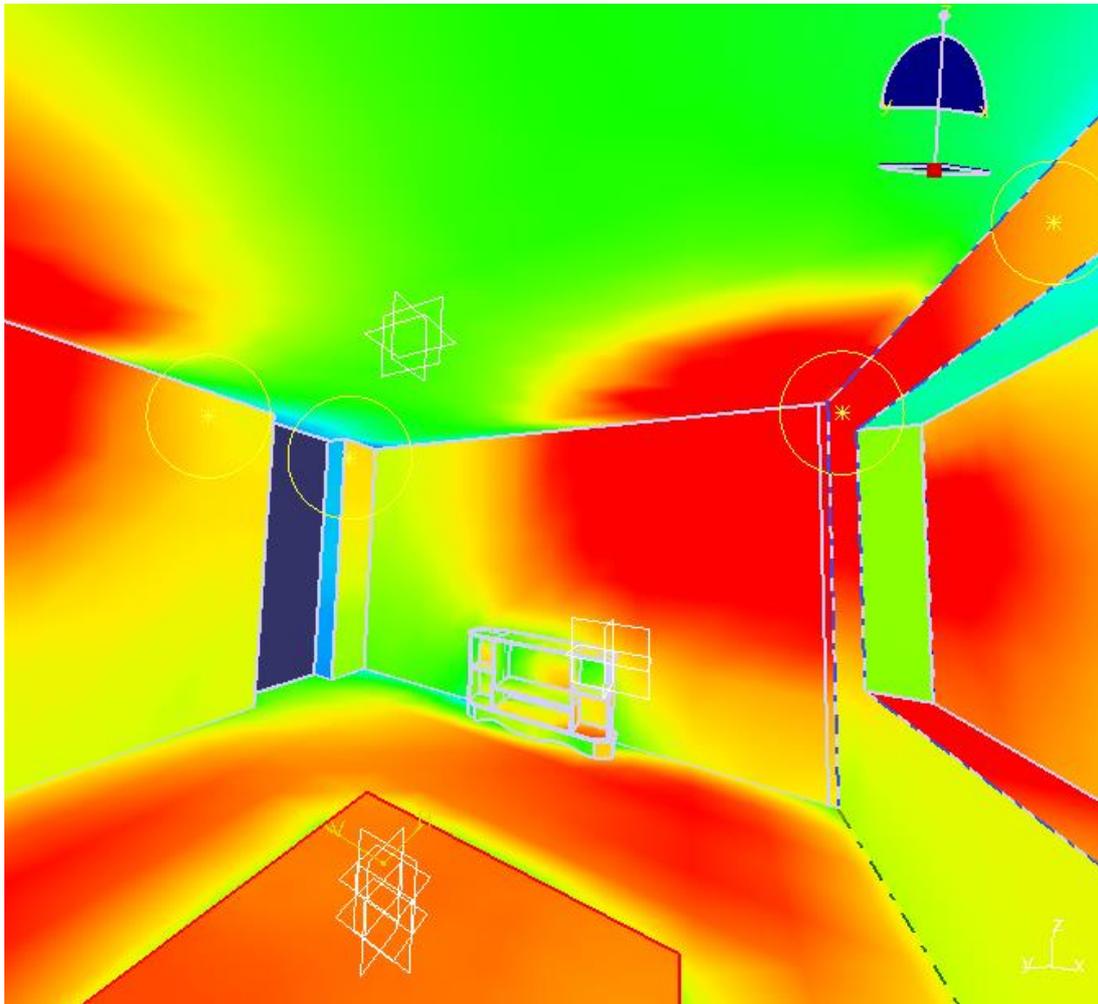


Fig. 3. CATIA workspace containing the scene with heat map applied to the surfaces.

As displaying any bitmap textures in the CATIA workspace viewport is possible only in the Shading with Material display mode, we disable turning on heat map mode in case if any other view mode is currently active. But user also can switch from Shading with Material view mode to other view mode already after heat map display had been turned on. For automatic turning off heat map display in this case, we had to set up the callback for the VIEWER_UPDATE event. This event is being activated each time after any redraw of the viewport. The event handler checks the current view mode and turns off heat map display if the mode is not the Shading with Material.

CATIA workspace with heat map display turned on and dialog box containing controls of this mode is shown on Fig 3. The color bar shown at the left of the dialog box shown on Fig. 2 corresponds to the 1-dimensional texture applied to the surfaces.

5. Showing a spherical panorama as a background image

Spherical panorama is an image combined from several photos taken with special equipment which display a view from some point in all directions (360 x 180 degrees). Usage of a spherical panorama as a background image allows increasing realism of representation of artificially created model. The model is inserted in panorama. Such panorama can also be a light source for the created models if it is HDRI (High Dynamic Range Image) [13].

We had provided different parameters of HDRI panorama which can be set up by user. It is possible to set geographical location and time of the shot panorama, and to specify its orientation relatively to the scene.

The HDRI lighting reconstruction page allows reconstruction of HDRI luminance using different methods. HDRI provides luminance for each direction of panorama. But due to com-

plexity of HDRI acquisition process in many cases the luminance recorded in a pixel does not equal to the luminance of this direction in reality. In such a case reconstruction methods allow to find scaling coefficient for all pixels of HDRI [13]. The bright regions of the image can be represented as parallel lights, if Virtual lights switch is On. It allows to increase the speed of MCRT calculations and provide highlights from these lights in BRT. The brightest parallel light is considered as Sun. Sun flux is usually underestimated in HDRI due to overflow of measurement device. Reconstruction allows to calculate Sun flux separately.

It is desirable to show the panorama preview directly in the CATIA viewport for setting correct orientation of the spherical panorama relatively to the virtual object or model. User should have a possibility to change common parameters of the panorama and its orientation and see the results immediately on the screen.

For solving this task we created the temporary geometrical representation with spherical shape in CATIA viewport. The entire scene built by user is to be inscribed into this sphere. The simplest way to place the sphere visually behind all other scene objects would be forced changing of the workspace depth buffer. But unfortunately, as in the previous cases, absence of direct access to the CATIA depth buffer forced us to invent some indirect way for bypassing this problem. So the sphere was created just as a usual temporary geometric representation in the workspace. The center and radius of the sphere are set automatically and are changed corresponding to viewpoint orientation change.

Initially the sphere center is set equal to the current camera position and sphere radius is set equal to some constant. Then, in the callback of the viewport redraw event, the current radius of the scene bounding sphere (which includes the radius of the newly created sphere object) is obtained. If the obtained radius is more than in 1.00001 times larger than the radius of the panorama object, then the panorama radius is doubled. Such stepped radius change with additional threshold allows to avoid infinite looping which may appear because global bounding sphere radius obtained through CATIA API functions includes the newly created sphere radius too.

The function which prepares initial arrays for the sphere representation is called only once after the first call of the Show Panorama Preview command. The callback procedure which handle camera position change and correspondingly sets sphere center and radius should not rebuild the entire geometry. It just sets a new transformation matrix to the already created graphic representation object. Also, the texture data is being changed only if the current HDRI panorama has been changed. It makes the redraw procedure faster. Another callback handles the current view mode. As panorama representation with texture can be painted only in the Shading with Material view mode, the callback should handle view mode changing and hide the panorama representation if the current view mode is not the Shading with Material. Besides HDRI panorama, also similar algorithm is used for drawing the daylight preview representation in the viewport. In this case panoramic image is generated dynamically using the current daylight parameters (sun position, cloudiness etc).



Fig. 4. Preview of artificially created car model inserted into spherical panorama.



Fig. 5. Rendering result of the car model with HDRI panorama used as a background.

The preview of a spherical panorama together with a car model in the CATIA workspace is shown on Fig.4. User can change camera orientation in real time and see the preview immediately while high quality scene rendering may take several hours. Realistic rendering results of the scene is shown on Fig. 5.

6. Conclusion

We have implemented the three new visualization modes for CATIA system. The modes are a part of commercial Lumicept/CA product. During development many problems were related to closed system interfaces of CATIA system. This required invention of the ways to get data necessary for implementation of our algorithms.

As a result the additional visualization modes provide better integration of lighting simulation and physically accurate rendering module into the CATIA CAD. They make more convenient the work of end user with the program complex. Lighting engineers who use extended CATIA program complex are now able to edit the required models dynamically, in real time. Added visualization modes allow them to view immediately the reflectance of these changes to the results of lighting simulation as well as preview of the changes without repeating the time-consuming procedures of high quality rendering after each adjustment of scene parameters.

References

1. Bondarev, A.E., Bondarenko, A.V., Galaktionov, V.A.: Visual analysis procedures for multidimensional data. *Scientific Visualization*, 2018, **10**(4), pp. 120-133, DOI: 10.26583/sv.10.4.09
2. Ryabinin, K.V., Belousov, K.I., Chuprina, S.I., Shchebetenko, S.A., Permyakov, S.S.: Visual Analytics Tools for Systematic Exploration of Multi-Parameter Data of Social Web-Based Service Users. *Scientific Visualization*, 2018, **10**(4), pp. 82-99, DOI: 10.26583/sv.10.4.07
3. Barladian, B.Kh., Voloboy, A.G., Galaktionov, V.A., Shapiro L.Z.: Integration of Realistic Computer Graphics into Computer-Aided Design and Product Lifecycle Management Systems. *Programming and Computer Software*. 2018, **44**(4), 225–232.
4. Barladian, B. Kh., Voloboy, A.G., Shapiro, L.Z.: Integration of illumination simulation by ray tracing method into CAD systems, In: Proc. of the 16th Int. Conf. GraphiCon'2006, Novosibirsk, Russia, 2006, pp. 275–278.
5. Barladian, B. Kh., Voloboy, A.G., Shapiro, L.Z.: Generating realistic images in CAD systems, In: Proc. Of the 23rd Int. Conf. GraphiCon'2013, Vladivostok, Russia, 2013, pp. 186–190.
6. Padolski, S., Korchuganova, T., Wenaus, T., Grigorieva, M., Alexeev, A., Titov, M., Klimentov, A.: Data visualization and representation in ATLAS BigPanDA monitoring. *Scientific Visualization*, 2018, **10**(1), pp. 69-76, DOI: 10.26583/sv.10.1.05
7. Grigorieva, E.G., Klyachin, A.A., Klyachin, V.A.: Boundary Identification Algorithms and Visualization of the Solution of the Plateau Problem in a Blender Environment. *Scientific Visualization*, 2017, **9**(4), pp. 13-25, DOI: 10.26583/sv.9.4.02
8. Dassault Systemes, Inc. CATIA Version 5-6 R2015 Documentation. <http://media.3ds.com>. Last accessed 05 Aug 2018
9. Loop, C., Schaeffer, S.: Approximating Catmull-Clark subdivision surfaces with bicubic patches. *ACM Trans. Graph.* **27**(1), 2008, 8:1–8:11.
10. Yuan, Y., Wang, R., Huang, J., Jia, Y., Bao, H.: Simplified and tessellated mesh for realtime high quality rendering. *Computers & Graphics*, 2016, **54**, 135-144.
11. Weinstein, J.: A Postgenomic Visual Icon, *Science*, 2008, **319**, 1772–1773.
12. Borland, D., Taylor, R.: Rainbow Color Map (Still) Considered Harmful. *IEEE Computer Graphics and Applications*, 2007, **27**(2), 14–7.
13. Voloboi, A.G., Galaktionov, V.A., Kopylov, E.A. and Shapiro, L.Z.: Simulation of natural daylight illumination determined by a high dynamic range image, *Programming and Computer Software*, 2006, **32**(5), 284-297.