Animation of shadow - Advantages and disadvantages when rendering 3D project

Student authors: Valentina Acevska[1], Ilche Acevski[2],

Mentor: Igor Nedelkovski

***Abstract* – Animators use 3D animation software to produce the special effects that can be seen in movies, as such as this film of Clock Tower in Bitola. The ultimate goal is to gain animation of the shadow of the Clock Tower, during one day of 24 hours, developed in Maya 2010 and then imported into 3ds Max where the final product was obtained. There will be reviewed and discussed various results when rendering with mental ray.**

***Keywords* – 3d, animation, graphic, rendering, shadow**

I. Introduction

3D computer animation software is used to produce many of the special effects that can be seen in this movie, as well as in many movies. The architectural model of the Clock Tower represents the shape of the building and its parts that contains some of its important features. For this purpose polygonal modeling is used where points in 3D space, called vertices, are connected by line segments to form a polygonal mesh [3].

The clock tower is surrounded by nature and street lights. Modelling of nature such as trees, plants and flowers is by placing hundreds, even thousands of branches and leaves of the tree, but this approach requires great dedication, effort and time. The techniques of algorithmic and procedural generation of shapes allow 3D graphics to create abstract, but realistic geometric shapes. Fractals and graftals can create convincing plants or trees by copying a few simple geometric shapes. Contrary to the forms of nature that are modeled manually (which can have thousands of geometric shapes), these methods involve a small memory space, and both techniques are visually effective and technically efficient [2].

The texture represents shadowing or some other attribute in computer graphics that adds the area of graphic images to get an impression of physical matter. For example you can achieve a surface look like it was made of metal or glass and have their ability to reflect light. For example, scanned image or photograph of a wall, can be applied to any shape and also to simulate the material from which the wall is made [1].

Student autors:

[1]Valentina Acevska is with the Faculty of technical sciences, Ivo Lola Ribar bb, 7000 Bitola,…Macedonia, E-mail: valle.mk@gmail.com.

[2]Ilche Acevski is with the Faculty of technical sciences, Ivo Lola Ribar bb, 7000 Bitola, Macedonia, E-mail: iacevski74@gmail.com.

Mentor: Prof. Igor Nedelkovski is with the Faculty of technical sciences, Ivo Lola Ribar bb, 7000 Bitola, Macedonia, E-mail: igor.nedelkovski@gmail.com.

Animation is derived from the Latin anima and means the act, process, or result of imparting life, interest, spirit, motion, or activity. Techniques and algorithms directly used to create and manipulate motion are: keyframing (gives the most direct control to the animator who provides necessary data at some moments in time and the computer fills in the rest), procedural animation (involves specially designed, often empirical, mathematical functions and procedures whose output resembles some particular motion), physics-based techniques (solve differential equation of motion) and motion capture (uses special equipment or techniques to record real-world motion and then transfers this motion into that of computer models) [6].

Animations can be divided into: animation of a still object, clear structural animation, dynamic simulation, individual animation and behavior animation. Animation of a still object is the easiest and most present form. It is one of the oldest and most popular forms of computer animation. It can be described as a basic requirement for animation and can be used in any form by any other categories. It can produce animated sequences rendering the scene at different positions, with light from different positions or from different sources, or by moving the point of view (virtual camera) [8].

Light objects in 3D software can be divided into: point/omni light or volume light in Maya (casts rays in all directions from a single source), spot light (casts a beam of light only in one direction and have properties to control the area affected by light), directional light (also called “Infinite” or “Sun” light cast parallel light rays in a single direction, as the sun does at the surface of the earth) and ambient light (provides perfectly even lighting throughout the entire scene, with no shading). Some of the lighting attributes are: diffuse and specular, color, intensity, decay, and attenuation. If shadows are not enabled, light goes through objects. There are depth mapped shadows (use texture maps to store information about which objects are in the shadows, that usually render quite quickly, don't support transparency and they produce are not physically correct, and will not match up with the way soft shadows look in real life) and raytraced shadows (support transparent shadows and correct soft shadows, but they are usually slow when rendering) [4].

Last stage for the final product of each 3D programming package is processing all elements (surfaces, materials, lights and movements) in the 2D image or sequence of images. One of the most important things is to synchronize the time of rendering with the image quality. There are several types of rendering: software, hardware, vector, V-Ray, finalRender, RenderMan and Mental ray [1].

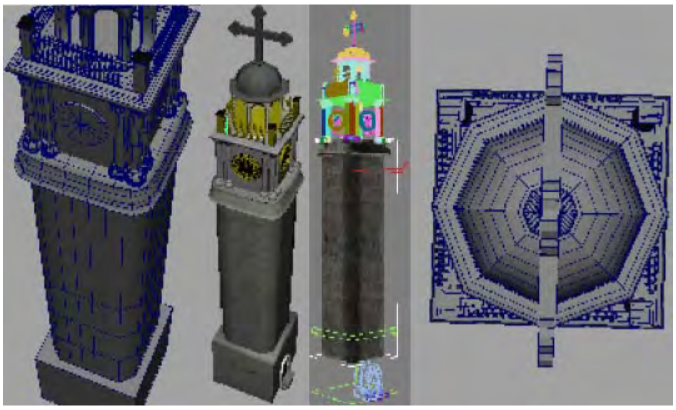

Fig. 1. Model of the Clock Tower in Maya 2010

In this case the ultimate goal is to gain animation of the shadow of the Clock Tower together with all its surroundings, during one day of 24 hours. For this purpose the methodology for the final animation will be briefly discussed. That would cover techniques and software used for modeling, texture, lighting, animation and rendering. Mental ray is mostly used tehnique in this kind of a project. There will be presented results when rendering with Mental Ray using different settings, and they will be compared with the results from the previous researches.

II. TEHNIQUES

A. Methodology

Fig. 1 illustrates the process of developing the Clock Tower that was modeled in Maya 2010. Textures for this model shown at Fig. 2 and Fig. 3 is taken from the actual construction in order to be able as close as possible to represent the actual characteristics of the real building blocks used in construction of the object. Some parts of the nature such as trees are taken as finished products from the web. Although the textures are used to add fine detail to the surface, this usually is not enough for modeling rough surfaces that appear on the object, such as the walls, trees or grass. The intensity of illumination of the details to such objects are adjusted independently of the lighting parameters such as direction of the light rays. Better way for modeling the bumps at the surface is applying the function of the noise to a normal surface and then uses the normal vector of rough parts when calculating the brightness of the model. Fig. 3 illustrates this technique that is called bump mapping. Bump mapping gives the illusion of the presence of bumps, holes, carving, scales, and other small surface irregularities. Moreover on a brick wall, a texture map will provide the shape, color, and approximate roughness of the bricks.

Before proceeding further it must be said that a single scene in this project, with most objects and directional light as lighting that gives the sun, modeled in Maya, takes several hours for rendering with Mental Ray. With importing of the same objects in 3ds Max, and replacement of directional light with 3ds Max built-in sunlight system, rendering of the same scene with a few settings is done in a minute.

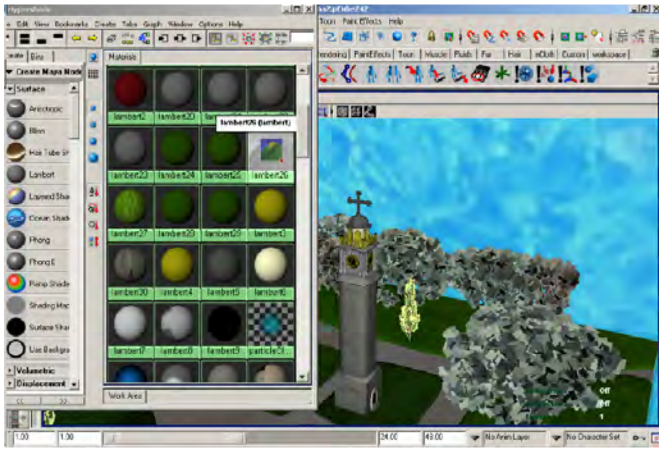

Fig. 2. Textures in Maya 2010

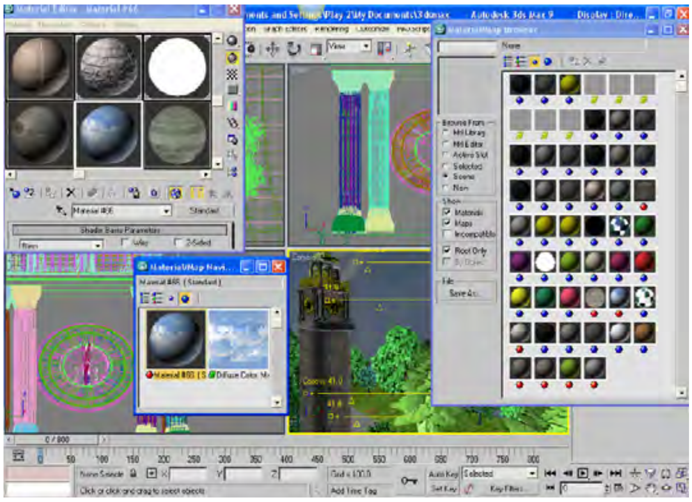

Fig. 3. Textures in 3ds Max

Therefore majority of the further project, part of the texture, animations, and additional street lights (omni light) for nighttime lighting of the park and reflector lights (spot light) for the clocks are prepared in 3ds Max.

Animation of a still object is mostly used in this project. During this, animation of the sun that moves around the planet is prepared, and as a result of that a whole day or 24 hours is obtained. In Maya, that can be done with directional light and animation of that light, while 3ds Max incorporates a special system of functions known as Daylight System. There is also animation of other light sources (street lights and reflectors) that are activated by darkening, or when the sun is on the other side of the planet. There are animations of the arrows of four clocks from the Clock Tower, mostly obtained by rotation on some of the axes, depending on the side of the world. This can be seen at Figs. 4, 5, 6 and 7. The background is also animated as you can see by the position of the sky and clouds on the same Figs. However the biggest impression leaves the camera as animated objects. Some or all external parameters of camera can be animated, where the camera flies through a mostly static environment. As the real camera, the virtual camera makes images based on its position relative to the scene. Which part of the scene will be captured in the image depends on how much the scene is in the scope of view.

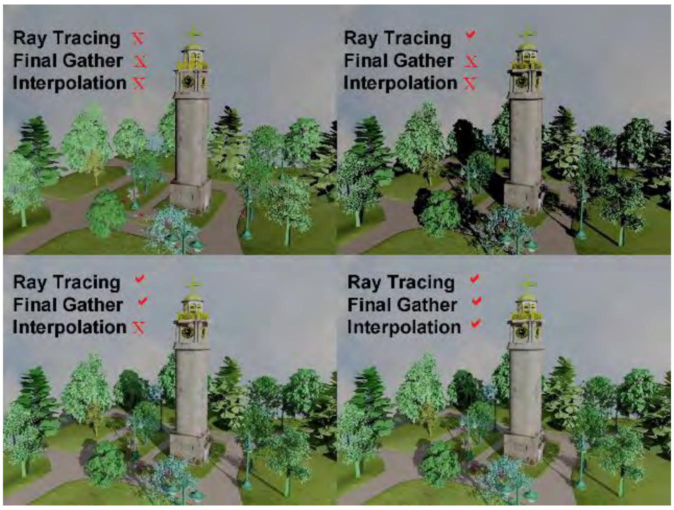

Fig. 4. Results from the 100-th scene (morning)

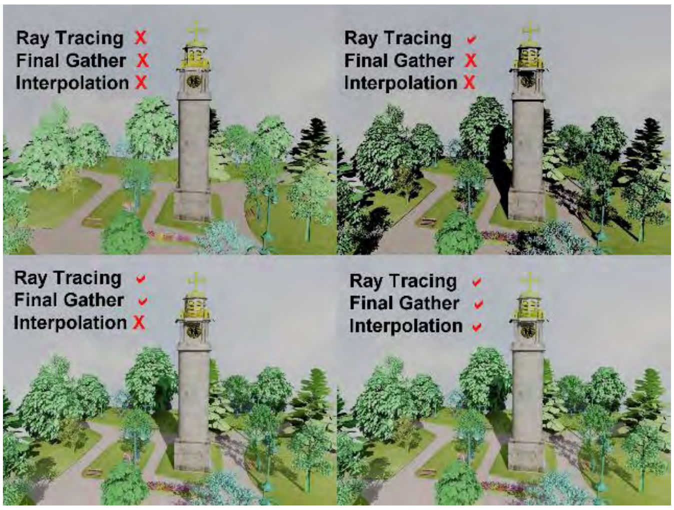

Fig. 5. Results from the 200-th scene (before noon)

By tuning the parameters of the scope of view, the user can simulate many aspects that are encountered in the real camera, including: location (location of the camera in the scene), direction (direction in which the camera is focused), width of the lens or viewing angle (angle in which objects can be captured - depends from the distance between camera and the object), angle or orientation of the camera (rotation angle or the scope of view) and focal distance (distance from the lens in which the object is captured with the highest clarity - objects that are "out of focus" look blurred). The real camera always creates perspective projection represented by a single point of view, objects are decreased with their distancing, and parallel lines that are parallel to the plane converge (gather) to the point of vanishing. When the camera is animated object, it is necessary to control the viewpoint and viewing direction, or two degrees of freedom. The camera must always show the object of interest, which usually can be static, movable, or with movable parts on it, as in this case with the clock arrows.

Mental ray includes many features (Global Illumination, Final Gather, reflection, refraction etc.), most of them used to obtain photorealistic rendered results and is a production quality rendering application that supports ray tracing to generate images.

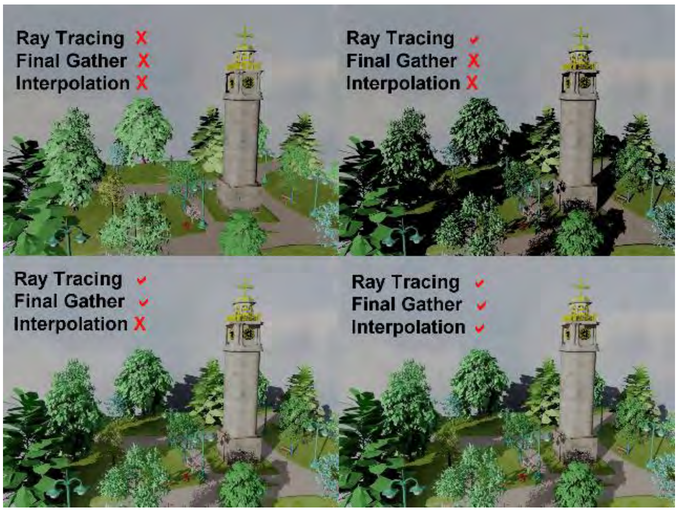

Fig. 6. Results from the 410-th scene (afternoon)

Fig. 7. Results from the7-th scene (night)

The software uses acceleration techniques such as scanline for the determination of the primary visible surfaces and binary space partitioning for secondary rays. Also supports simulation of global illumination including photon mapping. Any combination of diffuse, glossy (soft or scattered), and specular reflection and transmission can be simulated. For the final product this tehnique is used. Resolution of 640x480px is used for the final product. Rendering the scenes with very high resolution can be a challenge, not only because the rendering time increases by a quadratic function, but also because large memory is required. Memory problems will become bigger by using Global Illumination and/or Final Gather, because they use extra memory to calculate the indirect effects of lighting. The project will be rendered four times with diferent settings. The project will be rendered four times with diferent features: with Ray Tracing disabled, than with Ray Tracing enabled and Final Gather disabled, then with both enabled but with Interpolation over Num. FG points, and for the end with Radius Interpolation Method instead of Num. FG points.

The hardware configuration used for this project is: AMD Athlon 64 Processor 3000+ 1.81GHz, 1.5 GB DDR2 RAM, NVIDIA GeForce 6600 256MB on Windows XP.

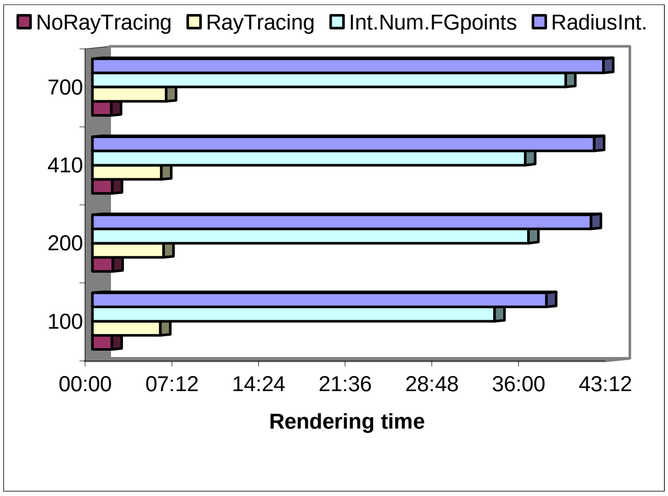

Fig. 8. Results from the rendering

TABLE I
RENDERING TIME

Scene	NoRayTr.	RayTr.	Int.N.FGP.	R.Int.
100	01:36	05:38	33:24	41:42
200	01:41	05:54	36:13	41:25
410	01:38	05:43	35:57	41:40
700	01:33	06:07	39:18	42:26

B. Results

The results of rendering can be seen at the Fig. 4 for the 100-th scene (morning), Fig. 5 for the 200-th scene (before noon), Fig. 6 for the 410-th scene (afternoon) and Fig. 7 for the 700-th scene (night) from a total of 800 scenes. The quality of the scenes with the selected features can be seen from the above mentioned Figs. But another important feature is the duration of rendering. Chart from Fig. 8 and Table I illustrates the time consumed at each rendering on each scene. It also supports caustics and physically correct simulation of global illumination employing photon maps. When rendering some of the scenes with caustics and global illumination enabled, time consuming is 1 hour 22 minutes and 29 seconds. In some cases with some options like Atmosferics enabled Mental Ray will run out of memory followed by error message and with image which contains a black squares in render buckets or 3ds Max might fail.

C. Discusion

From the Fig. 4, 5, 6, 7 and 8 and from Table I can be concluded that if Ray Tracing is disabled, there are no shadow at the scenes, although duration of rendering is about half past one minute. With Ray Tracing enabled there are dark shadows (no so realistic), but good enough for the time consuming of about 6 minutes for a scene. With Final Gather enabled with Interpolation over Num. FG points there are so realistic soft shadows with realistic form, but with time consuming of about 35 minutes for a scene. Almost the same quality is obtained with Final Gather enabled but now with Radius Interpolation Method instead of Num. FG points, and the time consuming is about 42 minutes for a scene. For a total of 800 scenes in the animated movie with the first method, time needed for rendering of all scenes is less than a day. With the second method time needed is about 3 days, with third method is about 20 days, and with last method is about 23 days. Certainly with a better hardware configuration, these times will be improved. The primary feature of mental ray is the achievement of high performance through parallelism on both multiprocessor machines and across render farms There are, also, many other features that can be researched with Mental Ray. Scanline acceleration technique is used for primary visible surface determination and binary space partitioning (BSP) is used for secondary rays. Here is some other features that can be set: Sampling Quality, Rendering Algorithms (Scanline, Raytrace Acceleration, Trace Depth), Camera Effects (Motion Blur, Contours, Camera Shaders, Depth of Field), Shadows&Displacement (Shadow, Shadow Maps, Displacement), Memory options etc.

In [5] and [7] a variety of techniques are discussed for ray tracing dynamic scenes, and some ideas and experimental results a given, but there are no real implementation yet [5], or there are no realistic shadow from the lights [7].

III. CONCLUSION

Finally can be concluded that with current hardware configuration the best way is to complete the project only with the Ray Tracing enabled, which takes 3 days. Complet project can be seen at http://gorginaumov.edu.mk/?page_id=1713. Of course that still remains a big space for further working with other features, with better hardware, and also with other algorithms and new render techniques such as iRay.

REFERENCES

[1] ALAN WATT, 3D Computer Graphic-Third Edition, Skopje, 2010 ISBN 978-608-4606-02-4
[2] Anne Morgan Spalter, THE COMPUTER IN THE VISUAL ARTS, Brown University, 1999 ISBN 0-201-38600-3
[3] Dr Dragan Cvetkovic, RACUNARSKA GRAFIKA, Beograd, 2006 ISBN 86-7991-287-5
[4] Jeremy Birn, Digital Lighting and Rendering-Second Edition, New Riders, Berkeley, 2006 ISBN: 0-321-31631-2
[5] Peter Djeu, Warren Hunt, Rui Wang, Ikrima Elhassan, Gordon Stoll, William R. Mark, *Razor: An Architecture for Dynamic Multiresolution Ray Tracing*, January 24, 2007, Available: http://www-csl.csres.utexas.edu/gps/publications/tr07-razor [Accessed 2 Mart 2012]
[6] Peter Shirley, Michael Ashikhmin, Michael Gleicher, Stephen R. Marschner, Erik Reinhard, Kelvin Sung, William B. Thompson, Peter Willemsen AK Peters, FUNDAMENTALS OF COMPUTER GRAPHICS-Second Edition, Wellesley, Massachusetts, 2005 ISBN 1-56881-269-8
[7] Ufuk Tiryaki, *Design and Implementation of Rendering System with Ray Tracing and Ambient Occlusion*, 2009, Available: http://www.cmpe.boun.edu.tr/medialab/VW/ufuktiryaki.pdf [Accessed 2 Mart 2012]
[8] Проф. Др Игор Неделковски, КОМПЈУТЕРСКА ГРАФИКА (3D МОДЕЛИРАЊЕ И АНИМАЦИЈА) Битола, 2008 ISBN 9989786-02-X