

# TEHNIKE RENDERIRANJA I RAČUNALNOG OSVJETLJENJA

## RENDERING AND COMPUTER LIGHTING TECHNIQUES

*Andrija Bernik, Goran Blažuc*

Stručni članak

**Sažetak:** Ovaj rad opisuje tehnike renderiranja, render pogone, te vrste osvjetljenja scene u programu Autodesk Maya. Postoji više algoritama renderiranja koje se koriste u računalnoj grafici, a u ovom radu su opisani oni koji se najviše koriste ili oni algoritmi koji su bili važni za razvoj računalne grafike kakvu danas poznajemo. Opisan je mental ray koji je zaseban program, a u Mayi se koristi kao dodatak, pomoću kojega se Mayi omogućuje napredno i realistično renderiranje. Uz to se mental ray uspoređuje sa svojim najvećim konkurentom, V-rayom. Također su navedeni direktni izvori svjetlosti koje nam nudi Maya, te koje su to opcije za indirektno ili globalno osvjetljenje, koje nam nudi mental ray.

**Ključne riječi:** Autodesk Maya, mental ray, osvjetljenje, renderiranje, V-ray

Professional paper

**Abstract:** This paper describes the rendering techniques, render engines and types of lighting in the Autodesk Maya software. There are many types of algorithms in the computer graphics area, but this paper describes the most commonly used or historically important ones for the development of computer graphics. Mental ray, which is used as a plug-in, enables the photorealistic rendering for Maya. Mental ray is compared with its biggest rival – V-ray. Furthermore, this paper describes the sources of direct lighting which Maya offers, as well as the options that are available from mental ray for indirect or global lighting.

**Key words:** Autodesk Maya, mental ray, lighting, rendering, V-ray

### 1. INTRODUCTION

Rendering is the process of transforming or drawing a 3D scene into a 2D object, i.e. image. Rendering can be photorealistic, in which complicated algorithms are used and real-life effects are simulated, which results in a very long rendering period. When fast and interactive, i.e. real-time rendering is required, simpler and less demanding algorithms are used that can be calculated quickly, but the final product is of lower quality.

Mental ray is a special rendering application that is integrated in Maya. For years this program has represented a symbol of realistic rendering. Along with the fact that indirect lighting methods are implemented in it, it also contains a large number of options that can be used for adjusting even the smallest details in final rendering. As rendering is actually a type of photographing of a created 3D scene, many options that are found in mental ray and similar programs may also be found in professional cameras. V-ray is the strongest competitor of mental ray and these two rendering programs together dominate the market. Mental ray allows for very realistic photographs, as it simulates light rays and due to the fact that they react with a certain object and environment. In this process light flow, i.e. the amount of light that comes from a source of light to the object which it reflects from is very important. The

quality of light in this reflection depends on the material assigned to this object, and it determines how it is dispersed, reflected or refracted. This article is based on the final thesis 86/MM/2012, Polytechnic of Varaždin; which deals with the types of light sources in Maya, how they behave and which are their advantages and disadvantages. [1]

Along with direct light, i.e. the light that comes from a light source, there is also indirect light. This light is obtained by the reflection of light rays from the objects in the scene and, in this way, the parts of the scene which are not reached by direct light from the source are illuminated as well, just as in the real world. Global Illumination and Final Gather are methods of indirect illumination that come with mental ray and they simulate secondary rays that reflect from an object's surface.

In practical work a complete new scene is created and steps in obtaining a final product, the animation that is subsequently further elaborated in the Adobe After Effects program, are described.

### 2. RENDERING

In the introduction it has already been mentioned that rendering or drawing results in a final product of a virtual

3D scene, which previously has to be elaborated and modeled.

Before rendering it is necessary to set up a camera that will determine the location and perspective from which necessary information from the 3D scene will be taken and used for drawing the scene in the 2D form. Depending on the demands set by the industry in which rendering is used, it may be photorealistic or real-time. Thus, for the gaming industry a quick algorithm is necessary, as interactive rendering is required, but in that case details are lost. When the rendering time is not that important and if the computer is strong enough, as in the film industry for creating special effects, algorithms of greater precision and quality may be used. Such algorithms usually simulate light according to the laws of physics that are valid in the real world as well. It should be mentioned that each of these algorithms has its advantages and disadvantages, due to which render engines are in practice combined with two or more algorithms for more efficient rendering. [3]

### 2.1. Scanline rendering algorithm

The Scanline rendering algorithm is one of the oldest algorithms used in computer graphics, and it is still widely used. Scanline may be used for 2D and 3D rendering. [3] This algorithm is quick and allows for real-time rendering. It is applied in videogames and some scientific visualization programs. This method uses geometrical shapes, triangles and polygons that are called primitives. Before the rendering itself, the analysis phase is carried out. All objects are projected in a single two dimensional plane and then sorted according to their Y and X coordinates in that very order. In the analysis phase the information about pixels is recorded, so that by the time the final phase is reached, there are all necessary pieces of information, and only color is to be added to pixels using the raster method. In rendering, the image is drawn horizontally, line by line, and only the pieces of information obtained in the analysis phase are used thereat. A disadvantage of this method is that it is very difficult to get some photorealistic images using it. It is impossible to simulate reflection and refraction in some materials, and the use of special effects related to camera (lens shader usage) is disabled as well. Nevertheless, due to the speed of rendering, this method is still popular. The Autodesk 3ds Max uses the scanline render as the initial render engine. With a high degree of knowledge it is possible to achieve good results. [4]

### 2.2. Radiosity rendering algorithm

Radiosity is the algorithm that works on the principle of light reflection from one surface to another, due to which it is also one of the global illumination algorithms. The radiosity algorithm was named after the unit for thermal radiation intensity emitted from the surface. In this algorithm there are no typical light sources, but light is emitted from surfaces. As opposed to some other methods used for all light rays in the scene, radiosity only deals with rays leaving the light source and reflecting in a diffused manner. How many times will the

rays reflect, depends on settings that were selected, bearing in mind that each additional reflection increases the rendering time. Each surface being rendered is divided into several smaller patches. For each patch the form factor is calculated, which refers to the coefficient that describes how well is one patch seen by another patch. Logically observed, patches that are more distant and at a greater angle will have a lower coefficient. These coefficients are then inserted in linear rendering equations, and solving this equation system results in the brightness of each patch, bearing in mind shadows and reflections. As radiosity imitates a phenomenon from the real world, it may be used for obtaining quality and realistic images. Despite this, its quality falls behind the algorithms that fully track light rays. Recently this algorithm has become very popular, due to the advancements in computer technology, so powerful configurations that can use this algorithm in real time without any problems are available to regular users. Enlighten is a well-known graphic engine that uses this algorithm, and is used in famous games such as Battlefield 3 and Need for Speed: The Run. These games have become well-known because of their supreme reality in comparison with their competition. Lightscape is another well-known render engine that uses this method, and recently it has been incorporated in the Autodesk 3ds Max as an internal render engine. [5]

### 2.3. Ray casting rendering algorithm

Ray casting is an algorithm whose greatest advantage refers to its drawing speed. Ray casting was used in the 1990s for real-time rendering. Although this method is lapsed and almost not used anymore, it is very important for the development of the 3D computer graphics. First of all, the ray casting algorithm is a predecessor of the currently very popular ray tracing described in 3.4. Another significant feature is that, by using this algorithm, the modern gaming industry has developed. The most credit for this goes to the games Wolfenstein 3D and Doom, ones of the first 3D games. Due to the computer system limitations in the 1990s, the simplicity and rendering speed of ray casting allowed for smooth motions and interaction with the scene. [20]

Ray casting works on the principle of sending rays from the observation point (camera) towards the scene, after which it tracks these rays and calculates the interaction with the object that comes in the way. In reflection from the object its color is calculated based on illumination characteristics of the object. The rendering process ends there and further reflection of light rays from the object is not monitored, which means that it is impossible to simulate reflections, refractions and mild shadow decline. These effects may only be artificially created using textures. Ray casting sends the mentioned rays in groups, and they are grouped in vertical columns. Therefore, for the resolution of 320x200 320 rays will be emitted. For the purpose of comparison, the ray tracing method for the same resolution emits as many as 64,000 rays, which clearly shows that the later method is more detailed. [6]

## 2.4. Ray tracing rendering algorithm

It was mentioned that ray tracing is an advanced version of the ray casting algorithm. This algorithm is today widely spread in the world of computer graphics, but its application is different than in the ray casting algorithm. Ray tracing is not used for real-time rendering, but for detailed, mostly photorealistic rendering. The main principle of the ray tracing operation is kept from ray casting, but it also has a large number of additional options and calculations by means of which a high level of realism is achieved. Therefore, ray tracing is a rendering algorithm that calculates what will be the color of a certain pixel in the way that it emits light rays from the camera into the scene and observes which objects are illuminated by these rays and what happens with them as they travel through empty space. When an object is illuminated, its color is calculated based on the material (texture) assigned to it, as well as the amount and color of the light that reaches that object. [7]

Thereafter there is the possibility of emitting secondary rays, depending on whether or not there are reflections or refractions in the material definition. Rays emitted from the camera or observation point are also used for the calculation of shadows in the way that it is checked whether there is an object between the light source and the point being rendered that obstructs the light. The described method of ray tracing can be called backwards ray tracing, as rays are sent from the camera into the scene, which is contrary to the real world where rays of light originate from a light source. [7]

The reason for not using a physically correct method of ray tracing or forward ray tracing, lies in the fact that much more ray reflections would be required in order for them to, while being emitted from a light source, reach the camera, and as much as 99 % of these rays would never reach the camera. A convincing and physically accurate illumination, as well as the possibility of tracing secondary rays of light that come to existence in the contact with an object, are the main advantages of this photorealistic rendering method. Ray tracing is a method that dominates in render engines when quality and realism come first. Among them is, along with others, mental ray, a render engine described in the text to follow. [7]

## 3. MENTAL RAY

Mental ray is a render engine that is used for rendering high-quality and realistic images. Along with a physically accurate simulation of light that may be adjusted to the tiniest details, it is possible to achieve any imaginable lighting effect using it. The reason for its popularity and wide usage lies in its features that allow for rendering carried out by both advanced and regular users, without the need for a computer with high performances. It is achieved through advanced techniques of acceleration and sampling. Along with this, mental ray allows for and utilizes parallelism on multi-processor machines, as well as the usage of render farms (groups of computers connected into a cluster), which is

especially useful for advanced users. On the market mental ray may be found as a separate program or as a plug-in, i.e. library of other programs. [8]

Today mental ray is used in all industries that use computer graphics, from visual effects in many movies (The Matrix, Tron: Legacy, Hulk, Spiderman 3...), videogames, architectural design, prototype design in the car industry, to rendering in CAD programs for the purpose of it being used as a support for NURBS surfaces. [8]

### 3.1. Materials and shading

In order to render a geometrical shape, it is necessary to add a material definition to its surface. This material is defined separately from the object, and it can be directly assigned to a geometrical shape or connected to the graphic hierarchy that is intended for the shape. Shaders are small plug-in programs and may be selected from shader libraries that come in package with mental ray. Furthermore, if there is a need for new shaders, it is possible to program them in C or C++ program. A material may consist of nine shaders:

**Material shader** – defines the appearance of the object's surface when a ray of light falls onto it. It controls illumination, reflection, transparency and other features of surfaces. This is the only compulsory shader, while the rest of them are optional.

**Shadow shader** – controls the way in which transparent objects throw shadows, e.g. dirty glasses. Non-transparent objects have no need to use this shader.

**Photon shader** – is used only when Global Illumination or the caustics option is on. It controls the way in which the object influences the effects from the Global Illumination, such as diffusion, brightness, reflection and refraction. This usually results in color bleeding (the surroundings takes the colors of the illuminated object) and other lighting effects.

**Volume shader** – defines of which colors visible rays of light will be after they pass through the empty space behind the object's surface, taking into account that it is transparent. It is used when there is a smoke-filled volume, for visible rays of light and other similar effects.

**Photon volume shader** – is similar to volume shader, but instead of working on the principle of light rays from the camera, it traces the photons emitted from the source of light. Furthermore, it is used only when Global Illumination or the caustics options are off.

**Displacement shader** – using this shader details are pressed onto a geometrical surface. This shader is not used in rendering, but in the process of tessellation of objects.

**Environment shader** – is used if the material shader uses the effects of atmospheric mapping such as false chromium reflection.

**Contour shader** – controls the style of outlines if this rendering style is used. Outline rendering is used when a cartoon-like image is to be obtained or when glow is added to objects.

**Lightmap shader** – creates the texture map in the way that it scans the whole surface of the object. The collected information is saved in the texture map or another place where it may be used during subsequent rendering.

**Hardware shader** – is used when the mental ray automatic selection of shader hardware is to be bypassed. This has an effect only in hardware rendering that uses a graphic card for generating images.

Not only does mental ray operate well with regular Maya materials, but it is equipped with its additional materials. Various materials for various applications may be found here, from architectural materials and car paint to leather and similar materials. Some materials have changed and been upgraded through versions of mental ray, so there are more versions, e.g. `mia_material`, `mia_material_x` and `mia_material_x_passes` are the same material, but each of them has some additional options, so `mia_material_x` represents a newer version of `mia_material` and is delivered with some additional new options and altered interface for easier adjustments. `Mia_material_x_passes` is the latest version of this material, but it is completely the same as the previous version. Its only advantage is that the material works with a new rendering system (passage system) that was released with Maya 2009.

### 3.2. Work algorithm

As it may be concluded from its name, mental ray primarily uses the ray tracing methods. Ray tracing is the reason why images rendered by mental ray can achieve a high degree of realism. This method is described in chapter 3.4. Furthermore, it has been mentioned that each rendering algorithm had its advantages and disadvantages. The main disadvantage of key tracing is its construction. As opposed to other algorithms that use previous calculations, ray tracing treats each ray separately and starts a new calculation. Along with this, the way in which ray tracing seeks objects is inefficient and slows down rendering without any purpose. Due to these reasons mental ray uses another rendering method – scanline rendering. [2] It has already been mentioned that scanline rendering is very limited. The first condition for even using it is that there is no lens shader (shader applied on the camera lens) that deforms the image, and the second condition is that there are no reflections or refractions in the scene. Without reflections and refractions drastic losses are made to the rendering realism, so ray tracing is used. Along with the aforementioned condition that lens shaders are off, in order to maximize performances, mental ray first uses scanline rendering in order to calculate the color and make a sample of the image. When a change in the ray direction due to reflection or refraction occurs, it is automatically transferred to ray tracing during the rendering process. Once transferred to ray tracing, it does not return to scanline as long as the other ray is not emitted from the camera. [2]

### 3.3. Texturing and mapping methods

Texture mapping is a method of adding details, surface texture or color to a 3D model. Texture is applied on the surface of that 3D model. Every texture has its UV coordinate system. In this system each vertex is set from the polygon on which texture is applied. In this way the image is spread over the polygon and the face and gives a much richer visual result than the one that would be obtained with a limited number of polygons. In order to achieve more detailed and realistic results, often more than one texture is applied on the object's surface. This process is called multitexturing. There are many types and methods of multitexturing. On the object's surface concavities or convexities may be simulated, its lighting properties may be defined and it is even possible to generate a real convexity on the object's surface by using displacement texture. Which map type may be applied on the surface depends also on the definition of the material itself, so if a material has no reflection or glow, it is not possible to use maps that control these two properties. Diffuse map is the main texture that determines what will be the color of an object, while in the text to follow maps that are used only to add details to the initial texture are described.

#### 3.3.1. Bump and Normal mapping

Bump mapping is a technique in computer graphics by means of which cavities and convexities on the surface of the object being rendered are simulated. This method adds reality to objects without modifying their actual geometry. This is achieved by deforming the object's normal surfaces in the same way as if the surface was deformed. Modified normal surfaces of the surface are then used for common light calculations and, as the result, a smooth surface is given the effect of having cavities and convexities. Bump mapping uses a grayscale image for deforming normal surfaces. This grayscale image is actually a height map, which determines which part of the image will have concavities and convexities. Black color represents concavities, and white color represents convexities. Normal mapping also fakes concavities and convexities, as well as bump mapping, but the principle is somewhat different. Normal mapping, instead of grayscale, uses a RGB image that may contain information for the 3D coordinate system. Instead of deforming the existing geometrical normal surfaces, they are completely replaced by new normal surfaces generated from the aforementioned image. Normal maps are more detailed in comparison with regular bump maps, which is especially visible in shadows when the source of light moves in the scene. [3]

#### 3.3.2. Displacement mapping

Displacement creates what is only simulated by the bump and normal map. By using displacement mapping real geometry is created while rendering the object's surface, i.e. concavities and convexities that are obtained based on a grayscale image, the same as in bump mapping. Shadows thrown by these convexities are real and do not

need to be faked, which allows for a high level of realism, which is also the main advantage of this method. Displacement mapping offers the best results, but also requires many computer resources, so that rendering time is drastically prolonged. Displacement mapping was for a long time reserved only for advanced users of mental ray and was used mostly in film industry or in studios where more powerful configurations were available. However, as the computer technology progresses every day, with the new generation of graphic processors, displacement mapping could be used in the environment where real-time rendering is required, such as videogames. [3]

### 3.4. V-ray – an alternative to mental ray

Although there are several other render engines that could oppose mental ray, such as Renderman, Maxwell render etc., V-ray was selected for the comparison with mental ray due to their similar construction. V-ray is simultaneously the strongest competitor of mental ray, as it is available as a plug-in for a wide range of programs such as Autodesk 3DS Max and Maya. V-ray is a render engine based on the ray tracing algorithm, which is used for obtaining quality and photorealistic images. It was developed in 1997 by Vladimir Koylazov and Peter Mitev who founded Chaos Software that owns V-ray. Although there is a separate version of it, V-ray is mostly used as a plug-in in other computer graphics programs. It adapts to each of these programs, so its version for Maya is not completely the same as the one for 3DS Max, but the main features and functions remain the same. [9]

In its version for Maya V-ray sustains its basic primitives, as well as some of the basic shaders, but it cannot use Maya's materials, lights and some shaders. V-ray emulates their functionality by means of its plug-ins, which means that results may vary. Regarding the main features of V-ray, they are almost the same as the ones of mental ray, so V-ray also has the Global Illumination option, HDR image illumination, V-ray Sun and Sky, which is equivalent to Maya Sun and Sky that offers mental ray. Due to the aforementioned quality and realism, V-ray is used in the film and gaming industry, as well as for rendering realistic scenes for the architecture. V-ray played its role in the animated film "Avatar", which is the most expensive film of all times. Its previously unseen realism made the film famous, highlighting the fact accepted by many that it could set a new standard in the animated film industry. [9]

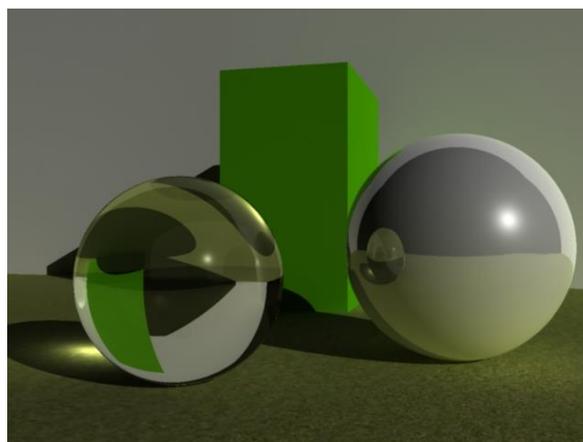
### 3.5. The comparison of mental ray and V-ray

In the last few years a lot has changed in the area of computer graphics, but one issue repeats again and again. Mental ray or V-ray, which render engine is to be used? These two render engines have separated themselves from the competition based on their quality and functionality, so the render engine selection for a quality and realistic project almost always goes down to these two engines. The parameters used in this comparison are quality and rendering speed, as well as the usage and user interface simplicity. It has already been mentioned that

these two engines are very similar, at least regarding the work principle, as both of them use the ray tracing algorithm. Due to this fact, most options available to mental ray are also available to V-ray and vice versa. The transition from one render engine to another one is considerably simplified due to this fact. In this case the versions of these two engines created for Autodesk Maya are compared. For this purpose a scene was created in which both render engines were tested.

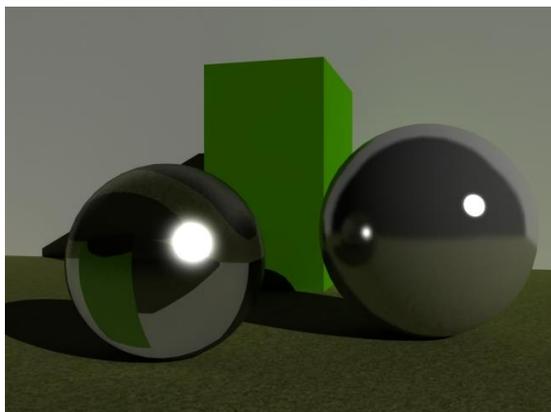
The scene consists of three geometrical shapes (one cube and two spheres) onto which various materials were applied. The Maya Lambert material was used (material with no reflections and refractions), as well as a reflecting and refracting material with the refraction index amounting to 1.52. The scene is illuminated by the *Directional Light*, and photons for the Global Illumination are emitted by a special *Spot Light*. The interface for both engines is located in Maya render settings, so at first sight they are not so different, but options are differently positioned and often they are not named in the same way. The mental ray interface is quite simplified, but a good knowledge is required in order to obtain quality results. At first sight V-ray does not have a simple interface, but its advantage lies in the fact that quality settings can be obtained with a lower degree of knowledge related to the render engine.

In figures 1 and 2 scenes with approximately equal mental ray and V-ray settings are rendered. Maya's materials Blinn and Lambert were used. It has been mentioned that a disadvantage of V-ray is the fact that it does not sustain Maya's standard materials, so it has to emulate them, which is visible from this comparison. V-ray could not calculate caustics with Maya's Blinn material and Directional Light.



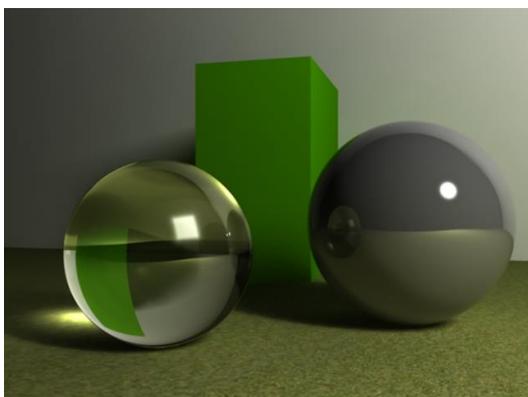
**Figure 1.** Scene rendered by mental ray and Maya materials

The reason for this is the fact that V-ray requires its own light that will emit photons necessary for them, as it does not sustain the photons emitted by Maya. The mental ray rendering lasted for around two minutes, and in V-ray a minute and 20 seconds, but the quality of mental ray is much higher.

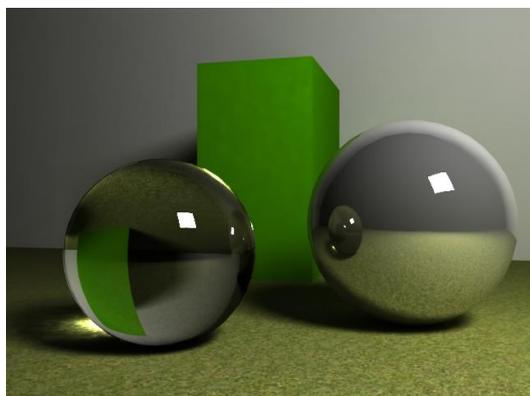


**Figure 2.** Scene rendered by V-ray and Maya materials

In order to fully test and use the possibilities of both render engines, in figures 3 and 4 the same scene is rendered, but with materials that come with these two engines. Regarding V-ray, its special light source was used, which allows for caustics as well. Mental ray does not have its own light sources, so Maya Area Light was used, which mostly resembles the used source in V-ray. In mental ray rendering (figure 3) its mia\_material\_x material for reflections and Maya Blinn material for refractions were used. Image rendering lasted for 6 minutes and 13 seconds. In V-ray rendering (figure 4) its VrayMtl material was used for reflections and refractions. V-ray rendering lasted for 7 minutes and 40 seconds.



**Figure 3.** Scene rendered by mental ray and its materials



**Figure 4.** Scene rendered by V-ray, its materials and a light source

When these two images are compared, differences are obvious, but both images are of high quality, so it would be difficult to decide which of the render engines is better. There is a significant difference in the rendering time, where mental ray beats V-ray by a minute. However, adjusting the settings to be approximately equal in both engines is quite a difficult task, so not much attention is to be paid to this segment. The advantage of mental ray is the fact that it works well with Maya shaders and lights, while V-ray with its emulating attempts clearly falls behind regarding quality. V-ray catches up with its own materials and lights by means of which it is easy to obtain quality results, which is not the case with mental ray. The quality obtained is more or less the same, but much more effort is to be invested into this process with mental ray.

Furthermore, while adjusting the Global Illumination and caustics in V-ray, quality results can be achieved much faster, e.g. in order to obtain quality global illumination with mental ray, it is almost always necessary to add an additional source of light that emits photons for Global Illumination, while with V-ray this is solved by using a single source of light. Thus, it can be concluded that very good results may be achieved much faster with V-ray, but mental ray has more depth, and results of a somewhat better quality may be achieved.

## 4. ILLUMINATION

### 4.1. Direct sources of light

There are six types of light sources in Maya. Each of them provides different options for the scene illumination. In the text to follow each of the light sources are described. For testing them the scene from the previous section was used. In this case Maya's Blinn and Lambert materials and mental ray's material mia\_material\_x were used. It should be noted that Final Gather was used as well as an indirect illumination method, and mental ray settings were set to Production Quality and resolution 640x480.

#### 4.1.1. Directional Light

Directional Light is based on a single vector and, depending on its direction, light illuminates every object at the same angle. The position of this source in the scene, as opposed to other sources, is irrelevant. The only relevant features are its direction and angle on the x-axis, which is controlled by rotation tools. Shadows cast by objects illuminated by light are orthogonal projections of their shapes. Directional Light is usually used as an auxiliary light and is useful when illumination of larger surfaces is required as a substitute for global illumination. The adjustments of this source are quite limited and there aren't many available options for detailed adjustment. [3]

#### 4.1.2. Point Light

In this source light is emitted from an infinitely small point in space and is equally dispersed in all directions. Although in the real world there is no light with perfectly uniform radiation, this source can be compared with a light bulb or a star in the space. This non-realistic radiation uniformity can easily be corrected or limited by means of other objects that cast shadows or by adding texture maps to that light. From a number of options of this light Decay Rate should be mentioned, which determines how the intensity of light decreases when moving away from the source, which affects the realism quite strongly. [3]

#### 4.1.3. Spot Light

In this source the light that is emitted from an infinitely small point in space is simulated too, but as opposed to Point Light, light is limited within a defined cone. In the real world this light could be connected with a type of spot light, as light beams are focused on one spot.

This type of source is most widely used not only in Maya, but in computer graphic technology in general. It can easily be manipulated, desired objects can be emphasized, and it can be grouped with other 3D models such as flashlight or car lights. As with Point Light, it is possible to use the Decay Rate option, and with Spot Light there are additional options for controlling the cone through which light is focused. [3]

#### 4.1.4. Area Light

Area Light is a two-dimensional source of light that simulates light from a certain shape, usually a square. These sources are used when light coming from a window, TV, neon lights etc. is to be simulated. Light can be formed in five shapes: square, disc, circle, cylinder and a user-defined shape. As opposed to other sources, Area Light requires more time for rendering, and the products of this are light and shadows of a better quality. [3]

#### 4.1.5. Volume Light

Volume Light works in the way that it retains light within a certain volume. This type of light is usually used only for some special effects. It is possible to choose between four shapes of Volume Light: sphere, box, cylinder and cone. Furthermore, there are three types of illumination: Outward, Inward and Down Axis. Outward is similar to Point Light, from one point light is dispersed to the outside, and is retained within the volume. Inward is reversed to Outward, i.e. the light is uniformly dispersed from the outside towards a single point within the volume. Down Axis or Downward behaves in a similar way as Directional Light, but the light is retained within the volume. [3]

#### 4.1.6. Ambient Light

In order to explain this type of light in Maya, first it is necessary to explain what ambient light is in the real world. Ambient light is an indirect light that reflected

from or passed through an object. It illuminates parts of a room that are not directly affected by light. When reflecting from objects, ambient light assumes some of their features. In Ambient Light in Maya this is unfortunately not the case, so this type of light is rarely used. Ambient Light adds the same color and intensity to all objects and from all sides of objects, which is not realistic and usually only impoverishes the scene in some details. In some realistic render engines this source is almost never used, but it is applied in cartoon-like rendering. Adjustments to this light are also quite modest and there are not many adjustment options. [3]

## 4.2. Indirect or global sources of light

Maya's direct sources of light are by themselves not enough to obtain a convincing and photorealistic image, so mental ray is used. Mental ray allows for several indirect lighting methods, including two environment lighting methods, Image Based Lighting and Physical Sun and Sky. There is also Global Illumination that physically accurately simulates ambient lighting and Final Gather that was used in previous images as well. [3] In figure 5 options available for indirect lighting in mental ray are shown.

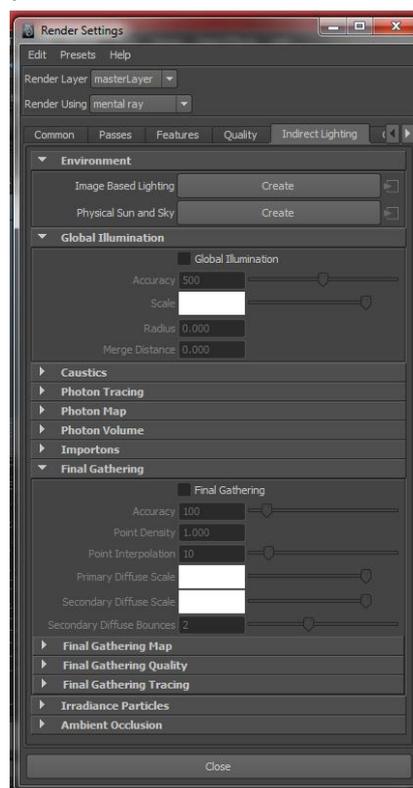
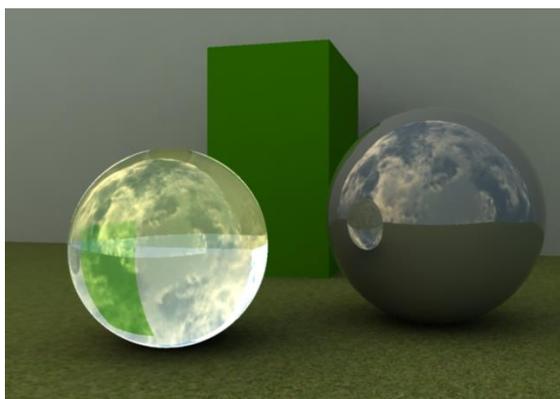


Figure 5. Options for indirect lighting

#### 4.2.1. Image Based Lighting (IBL)

Image Based Lighting allows for HDR lighting in mental ray. For HDR lighting high-dynamic range images are used, usually photographs from the real world, for illuminating the environment. Such image contains and can transport the original energy and color of light from the real world into the 3D scene. This way of lighting allows for high quality and a lot of freedom in adjusting

the illumination details. HDR images sustain decimal values in the standard contrast range from 0 to 255, which contributes to a higher precision by means of which the image contrast may be defined. Image Based Lighting in Maya is created by pressing only one button in the indirect lighting section in mental ray. When created, environment shader is automatically connected to the camera, and mental ray creates a special internal source of light that illuminates the scene from the environment. Around the scene a large sphere is created onto which the desired scene is applied. It is recommendable to combine IBL with Global Illumination and/or Final Gathering, because the results obtained without one of these options are very dark and unrealistic. [3]



**Figure 6.** Scene rendered by using Image Based Lighting

#### 4.2.2. Maya physical Sun and Sky

Physical Sun and Sky is used when a scene requires lighting that simulates real conditions of the relation between the Sun and the sky. As well as IBL, it is created by pressing a single button in the indirect lighting options. In this lighting Directional Light and three shaders are created (*mia\_physicalsky*, *mia\_physicalsun*, and *mia\_exposure\_simple*), and Final Gather is automatically on. The Sun is controlled very simply, there is one Directional Light in the scene and by using its x-axis the time of day is controlled, whereas by using the y-axis and z-axis the position of the Sun in the sky is controlled. Regarding the *mia\_physicalsky* shader, it is used for controlling almost the entire scene lighting, including the Sun, as options with *mia\_physicalsun* are directly connected to it. There are many adjustment options. It is possible to adjust the entire light intensity, color of light, saturation, color and position of the horizon, size and intensity of the Sun etc. Another shader being used is *mia\_exposure\_simple*, which is actually a Lens Shader that is applied to the camera lens.

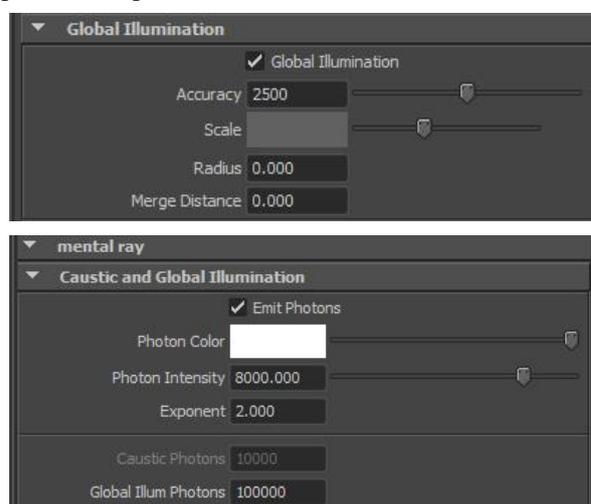
#### 4.2.3. Global Illumination

Global Illumination is a physically accurate simulation of indirect light that is reflected from objects and assumes some of their properties. Global Illumination works on the principle of photons emitted from a source of light and calculates their reaction in contact with objects, i.e. materials assigned to the objects' surface. Scenes illuminated only by direct light sources seem unrealistic,

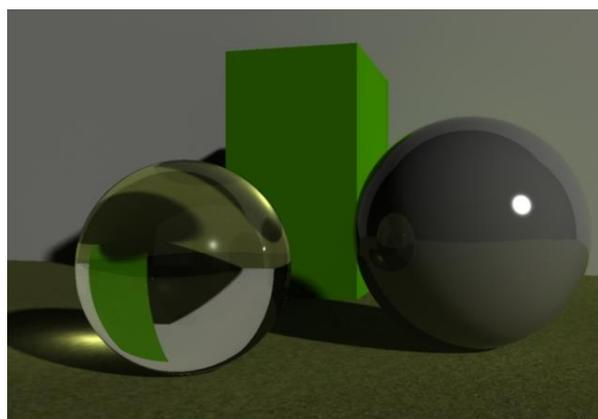
and Global Illumination allows for this level of realism with high precision. However, this method has a huge disadvantage, as it requires a lot of computer resources in order to be used and obtain smooth and realistic results. It should be taken into consideration while creating animations, as this option can prolong rendering quite a lot. Global Illumination has its settings located in the mental ray render engine, under indirect lighting.

In order to obtain the best results from indirect lighting, the best option is to use Global Illumination with Final Gather, which is described in the text to follow.

It should be taken into account that we need a quality computer system, as it requires additional resources. Global Illumination works on the principle of photons and, as the caustics option was not on, mental ray could have not precisely simulated that transparent object, because photons did not pass through it. In figure 8 a scene with Global Illumination and caustics on is shown, from which all options available from working with photons are presented.



**Figure 7.** Global Illumination settings



**Figure 8.** Render scenes with Global Illumination and caustics on

#### 4.2.4. Final Gather

As previously mentioned, Final Gather is also a method of indirect lighting that simulates secondary rays of light reflected from objects. As opposed to Global Illumination which uses photons emitted by a source of

light, Final Gather uses a camera from which it emits rays and then calculates their behavior in contact with the object's surface. This method is physically not accurate and does not exist in the real world, but it saves computer resources and adds some details that Global Illumination cannot. The reason for combining Global Illumination with Final Gather is the fact that Final Gather provides better illumination for angles that are reached by light from the source more difficultly. Final Gather sends rays from the camera; the camera sees these angles and illuminates them.

Final Gather settings are located in the same place as the ones for Global Illumination. For Final Gather it was not necessary to use an additional source that would emit photons, as this method does not use photons. It is necessary to note that setting and adjusting the Final Gather is much simpler than adjusting Global Illumination.

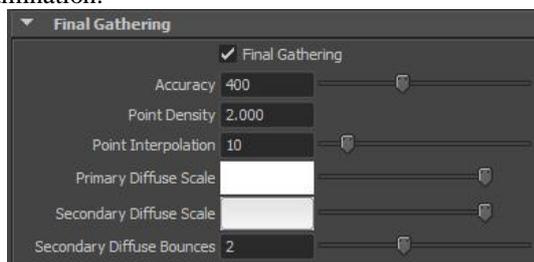


Figure 9. Final Gather options

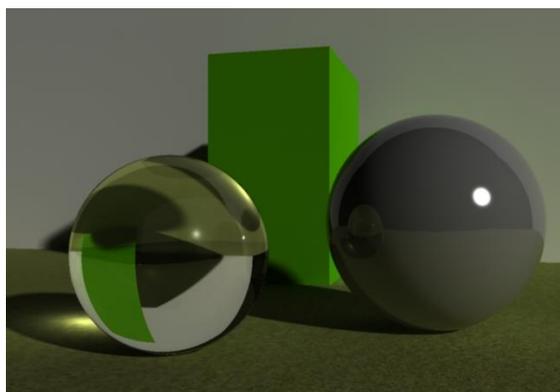


Figure 10. Scene with the Final Gather, Global Illumination and caustics on

## 5. PROJECTING

Making the animation is divided into several phases. Animation is conceived as a sort of a walk or seeing around a castle that has medieval elements combined with one modern central building. After the animation theme has been conceived, the realization consisting of the following phases followed.

### Modeling

Models were made exclusively with the polygon method. Regarding the polygon manipulation tools, along with the basic ones (Move, Scale and Rotate), Extrude, Split Polygon, Bevel, Bridge and others were used. Modeling the walls and buildings always started from the simplest geometrical shape, usually a square. This shape was then attributed with details and manipulated until the desired

result was achieved. For field modeling the Sculpt Geometry tool was used by means of which concavities and hills were made. In the end, by using the Paint Effect tool trees and grass were added.

### Model texturing

After modeling, model texturing followed. In order to save system resources and unburden the computer a little, for building and wall models repeating textures of low resolutions (700x700) were used. Each texture was also attributed with its bump map, which provides the illusion that the object's surface is uneven. For the field texture a bit more complicated texturing model with layer textures was used. With layer textures it is possible to add more textures to a single surface. By using alpha maps, which are drawn with the 3D Paint tool, it was determined in which field part which texture will be visible, and the texture intensity is determined by the alpha map color. The texture is not visible if the alpha map is black, while it is 100 % visible if the alpha map is white. For easier field drawing the reversed alpha map model was used, which was subsequently returned to the desired format by using Reversenode.

### Animating

Along with basic animations, such as door opening, animations of trees, grass, flag and lake were carried out as well. For the animation of grass and trees the turbulence option was used that comes in the set with Paint Effects models. For the animation of the flag nCloth and nucleus force were used, and the result was a quite realistic fabric animation that flutters in the breeze. For the simulation of the lake Maya's Ocean shaders was used with reflections and refractions on.

### Preparation for rendering

The illumination of this scene was easily adjusted. The Physical Sun and Sky system was used and there was no need for additional sources of light, as the entire animation action occurs in open space. Due to the limitations of the computer system that was used for rendering, only Final Gather was used for indirect lighting. The grass was used only in initial scenes, also due to the limitations of the system and reducing the rendering time. In the end the animation was rendered for almost two months with the resolution of 1024 x 768 pixels.



Figure 11. Final scene

## 5.1. Practical application

Computer 3D animations are mostly used for the purpose of entertainment. Today there are many animated films completely generated by computers, and creating special effects has become unthinkable without them. However, the usage of 3D animations is not limited only to the entertainment industry. They can be used in architecture for the presentation of existing or future objects. Archeologists use them for reconstructing ruins of some historical objects. They are used for design in the car, and even fashion industry. This animation was created for the purpose of entertainment. It was conceived as a virtual walk through some imagined castle in nature. However, in conceiving the theme, there were several options for the purpose of which the animation could have been used. There was the possibility of creating the model of an existing castle, so the animation could have been used for presentation or marketing purposes. Furthermore, there was the possibility of reconstructing ruins of a castle, in order to present its state in the past. There is also the possibility to upgrade the scene from which an animation was made or to change its purpose, so it could be used for some other, larger projects.

## 6. CONCLUSION

Rendering is a complicated process that requires a powerful computer system in order to obtain the most precise and realistic results. This is the final step in the process of creating images and animations obtained from 3D scenes, and also the most important one, as the quality of the final product depends on it. With suitable computer equipment it is possible to obtain photorealistic images, where it is not possible to notice that the image is virtual. It is fascinating that programs used, including Autodesk Maya, are simple for use, but also detailed enough for them to be used by the most demanding professionals. Rendering algorithms of these images heavily depend on the illumination of the scene, which does not differ much from taking a photograph. In the real world a photograph is obtained by rays of light entering the camera lens, and the brightness of an image depends on the amount of light. The same case is with rendering, with the difference that the camera is replaced by an imaginary camera in the scene, and light is simulated by using complicated and detailed algorithms that are valid in the real world as well. Rendering of greater quality and realism in Maya is achieved by using mental ray. The mere switching from the Maya Software render engine to mental ray with initial settings provides excellent results. Among many options available with mental ray the most important ones are to be mentioned. Global Illumination simulates indirect light, i.e. the reflection of rays from objects, and in this way illuminates scene parts that are not affected by direct light. Illumination by means of HDR images allows for precise adjustments of light and reflections in an open or closed space. Along with this, the usage of more advanced textures is allowed for, by means of which plane surfaces are provided with the illusion of having concavities or convexities. V-ray is a render engine that

strongly resembles mental ray and is also its greatest competitor on the market. This rivalry has lasted for years and in the future only positive outcomes of this rivalry are to be expected, as it leads to new ideas and perfection of the leading render engines in the computer graphics industry.

## 7. REFERENCES

- [1] Blažuc G.: Tehnike renderiranja i računalnog osvjetljenja, Final thesis 86/MM/2012, Polytechnic of Varaždin, Varaždin, 2012
- [2] Rendering with mental ray (Vol.1) - Thomas Driemeyer, Springer-Verlag New York, Inc. Secaucus, NJ, USA 2000
- [3] Keller E.: Mastering Autodesk Maya 2011, Wiley Publishing, Indianapolis, Indiana, 2010
- [4] <http://lets-do-blogging.blogspot.com/2007/10/scanline-rendering.html> (Available on 03.09.2012)
- [5] [www.flipcode.com/archives/Radiosity\\_In\\_English-The\\_Basics.shtml](http://www.flipcode.com/archives/Radiosity_In_English-The_Basics.shtml), (Available on 06.09.2012)
- [6] <http://www.permadi.com/tutorial/raycast/index.html> (Available on 06.09.2012)
- [7] [www.3dkingdom.org/tutorial/3ds-max-Lighting-with-standard-max-lights.html](http://www.3dkingdom.org/tutorial/3ds-max-Lighting-with-standard-max-lights.html) (Available on 03.09.2012)
- [8] [www.mentalimages.com/products/mental-ray/about-mental-ray.html](http://www.mentalimages.com/products/mental-ray/about-mental-ray.html) (Available on 03.09.2012)
- [9] [www.spot3d.com/vray/help/maya/150R1/render\\_maya.htm](http://www.spot3d.com/vray/help/maya/150R1/render_maya.htm) (Available on 06.09.2012)
- [10] [www.impresszio.hu/szabolcs/MentalRay/MentalRayHide\\_Shaders/MentalrayHidesshaders.html](http://www.impresszio.hu/szabolcs/MentalRay/MentalRayHide_Shaders/MentalrayHidesshaders.html) (Available on 03.09.2012)
- [11] [www.cs.washington.edu/education/courses/458/05au/reading/lighting\\_tutorial/lighttypes.html](http://www.cs.washington.edu/education/courses/458/05au/reading/lighting_tutorial/lighttypes.html) (Available on 06.09.2012)
- [12] <http://fuzzyphoton.tripod.com/whatisrt.htm> (Available on 06.09.2012)

### Contact:

**Andrija Bernik, dipl.inf./graduate computer scientist**

Polytechnic of Varaždin  
J. Križanića 33/6, 42000 Varaždin  
091 8991882