TECHNOLOGIES OF COMPUTER CONTROL
DATORVADĪBAS TEHNOLOĢIJAS

VISUALIZATION OF FREE – FORM SURFACES BY A RAY TRACING METHOD

BRĪVAS FORMAS VIRSMU VIZUALIZĀCIJA AR STARU TRASĒŠANAS METODI

Aleksandr Sisojev, *PhD student*
Riga Technical University
Faculty of Computer Science and Information Technology
Institute of computer control, automation and computer engineering
Address: Meza Str. 1/3, LV-1048, Riga, Latvia
E-mail: alexiv@inbox.lv

Aleksandrs Glazs, *Dr.habil.sc.ing., professor*
Faculty of Computer Science and Information Technology
Institute of computer control, automation and computer engineering
Address: Meza Str. 1/3, LV-1048, Riga, Latvia
E-mail: glaz@egle.cs.rtu.lv.

Keywords: Ray tracing, free-form surfaces, Bézier surfaces

1. Introduction

Problem of visualization of free – form surfaces is actual in various areas of a science and engineering. One of mathematical models used for this purpose is the mathematical description of a Bézier surface [1]. Classical methods the computer graphics based on polygonal models [2], use only polygonal interpolation of a surface [1]. It inevitably conducts to occurrence of an error and as a consequence – to discrepancy of an image. Modern tools of computer graphics, such as graphic libraries (OpenGL, DirectX) give opportunities for visualization of Bézier surfaces, but they have all classical methods disadvantages. Especially it is shown in visual effects. For example, in shading, reflection and procedural texturing. Alternative of methods sold in OpenGL, is the ray tracing. Such approach has a few advantages:

- allows to render exact model of a free – form surface (Bézier),
- allows with high accuracy to realize visual effects, which rendering in classical methods is inconvenient.

A known approach that uses ray tracing is described in work [4]. But that work suggests using minimizing areas to find the hit point of ray and surface. While the approach described in given work is based on using random search and gradient method to find the hit point of a ray and surface. The given work is devoted to use of this approach for visualization of Bézier surfaces on an example of a biquadratic surface.

2. Algorithm of a Bezier surface visualization

Let's consider algorithm of visualization of a biquadric surface which described by Bézier surface. As known [1] in this case biquadric surface described as a Bézier surface:

$$Q(u,v) = \sum_{i=0}^{2}\sum_{j=0}^{2} P_{i,j} \cdot B_{2,i}(u) \cdot B_{2,j}(v) \qquad (1)$$

where:

$P_{i,j}$ – control points,

$B_{2,i}(u)$, $B_{2,j}(v)$ – the Bernstein polynomials of the 2nd power,

u, v – parameters.

By analogy a ray can be represented as:

$$Q(t) = \sum_{i=0}^{1} P_i \cdot B_{1,i}(t) \qquad (2)$$

where:

$P_{i,}$ – control points,

$B_{1,i}(t)$ – the Bernstein polynomials of the 1st power;

t – parameter.

The mathematical task of a ray and a surface hit point search can be described as the nonlinear equations system:

$$\begin{cases} Q_X(u,v) = Q_X(t) \\ Q_Y(u,v) = Q_Y(t) \\ Q_Z(u,v) = Q_Z(t) \end{cases} \qquad (3)$$

To define unknown parameters u, v and t of the above equation system should be considered a case when the ray coincides with one of coordinate axes (Oz). Then the system of the equations is defined as follows:

$$\begin{cases} Q_X(u,v) = 0 \\ Q_Y(u,v) = 0 \\ Q_Z(u,v) = Q_Z(t) \end{cases} \qquad (4)$$

The solution of the system of the equations [4] can be obtained by solving an optimization task. As for finding parameters u and v only first 2 equations to be considered, therefore the function of minimization can be described as follows:

$$w = \left(Q_X(u,v)\right)^2 + \left(Q_Y(u,v)\right)^2 \to \min_{u,v} \qquad (5)$$

Task of function w optimization should be divided into two parts: preliminary search and optimization of the parameters (specification of roots).

2.1. Preliminary search

The preliminary search is based on a method of Monte-Carlo. First of all it is necessary to generate set **M** of random points $(u_i;v_i)$, $i \in [1;N]$. From this set we choose a subset of points **M'** for the subsequent optimization, i.e. points $(u;v)\in$**M'** which can be defined according to the following inequality:

$$w(u_i,v_i)-w(u^*,v^*)\le\varepsilon_0,i\in[0,N] \qquad (6)$$

where:

u_i, v_i – coordinates of a i-th point,

u^*, v^* – coordinates of a point that can be found by following minimization function:

$w(u^*,v^*)=\min_i(w(u_i,v_i))$

ε_0 – the given accuracy,

N – number of random points.

The further optimization of function w is carried out based on these subset points

2.2. Optimization of parameters

For optimization we choose a gradient method [3]. The values for these parameters are defined at each search step as follows:

$$\begin{cases} u^{new}=u^{old}-h\dfrac{\partial w}{\partial u} \\ v^{new}=v^{old}-h\dfrac{\partial w}{\partial v} \end{cases} \qquad (7)$$

where:

$$\begin{aligned}
&\frac{\partial w}{\partial u}= \\
&=2\cdot Q_X(u,v)\cdot\left[\sum_{i=0}^{2}\sum_{j=0}^{2}P_{i,j}^X\cdot B'_{2,i}(u)\cdot B_{2,j}(v)\right]+ \\
&+2\cdot Q_Y(u,v)\cdot\left[\sum_{i=0}^{2}\sum_{j=0}^{2}P_{i,j}^Y\cdot B'_{2,i}(u)\cdot B_{2,j}(v)\right] \\
&\frac{\partial w}{\partial v}= \\
&=2\cdot Q_X(u,v)\cdot\left[\sum_{i=0}^{2}\sum_{j=0}^{2}P_{i,j}^X\cdot B_{2,i}(u)\cdot B'_{2,j}(v)\right]+ \\
&+2\cdot Q_Y(u,v)\cdot\left[\sum_{i=0}^{2}\sum_{j=0}^{2}P_{i,j}^Y\cdot B_{2,i}(u)\cdot B'_{2,j}(v)\right]
\end{aligned} \qquad (8)$$

h – weight working step.

The optimization is carried out until the condition will be executed:

$$w\le\varepsilon \qquad (9)$$

where ε – small number (in this paper we assume ε<<0,1).

After optimization it is necessary to choose from the turned out subset only one point, which will be the decision of system of the equations (3) for parameters u and v. As the decision parameters u and v have be chosen so that the value of function $Q_Z(u,v)$ will be minimal:

$$Q_Z(u,v)\to\min_{Q_Z(u,v)} \qquad (10)$$

After a finding of values of parameters u and v third parameter t can be obtained from 3rd equations of system (3), as follows:

$$t = \frac{Q_Z(u,v) - P_0^Z}{P_1^Z - P_0^Z} \qquad (11)$$

where:

$Q_Z(u,v)$ – Z coordinates of surface,
P_0^Z, P_1^Z – Z coordinates of control points of a ray.

For a finding of a hit point of a surface with any ray it is sufficient with the help of elementary geometrical transformations to transforme such a ray and control points of a surface the way that the ray would coincide with a necessary coordinate axis (in our example - axis Z).

3. Experimental results

In this section the practical results of application of the considered approaches are presented.
Fig.1 presents the images of a biquadric surface without modeling of illumination received by ray tracing and for comparison - classical methods realized in library OpenGL.
A comparison of Fig.1-a with Fig.1-b demonstrates that at construction of surface the ray tracing does not have any advantages in comparison to OpenGL library.
In Fig.2 the images biquadric of a surface with modeling of illumination received ray tracing and for comparison - with application OpenGL are given. The comparison of Fig.2-a with Fig.2-b demonstrates that at modeling of illumination the ray tracing has advantage compared to OpenGL library.
In Fig.3 the scene with visual effects

- illumination,
- shading,
- raster texturing,
- procedural texturing,

received by ray tracing is given. For comparison - classical methods realized in library OpenGL are given. The comparison of Fig.3-a with Fig.3-b demonstrates that at modeling of visual effect the ray tracing assures some advantages to compare with OpenGL

In Fig.4 the scene with realization of such visual effects as is given:

- illumination,
- shading,
- reflection,
- raster texturing,
- procedural texturing.

Considering, that the standard tools for modeling of reflection in library OpenGL are not presented, the comparisons in this problem are impossible.

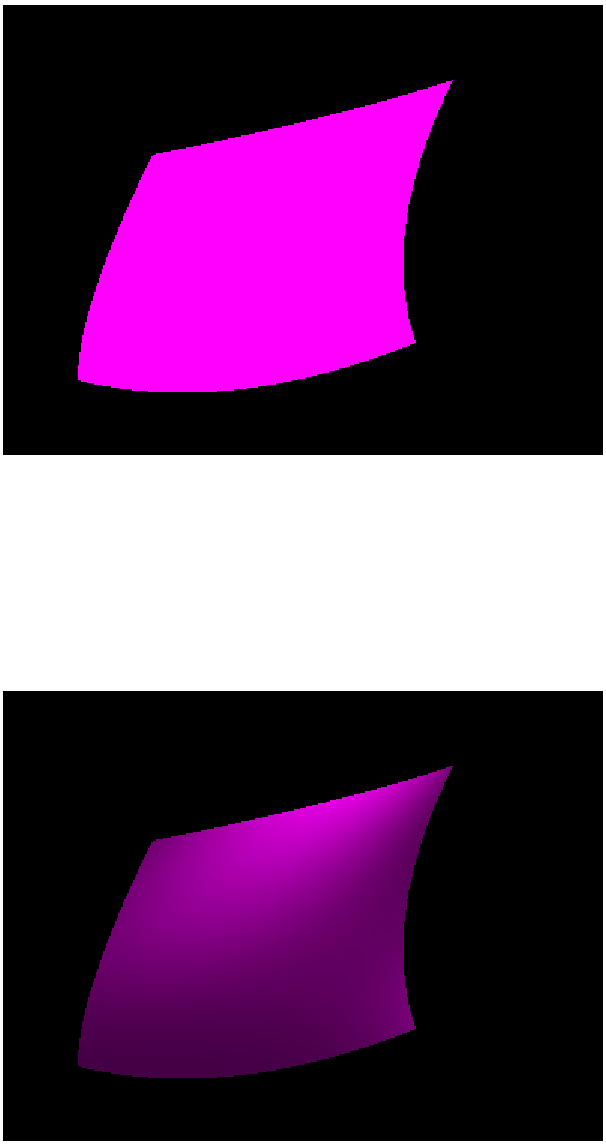


Figure 2-a: A biquadric surface with illumination (Ray Tracing).

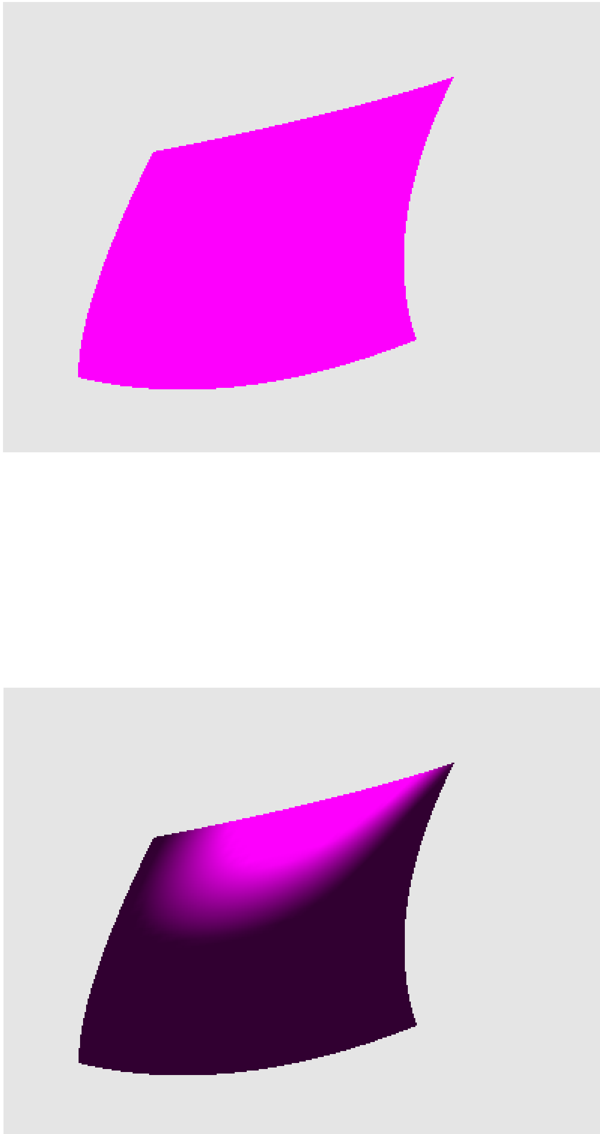
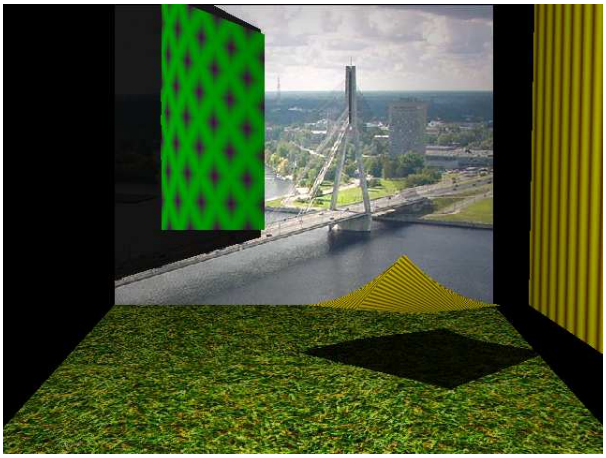


Figure 2-b: A biquadric surface with illumination (OpenGL).



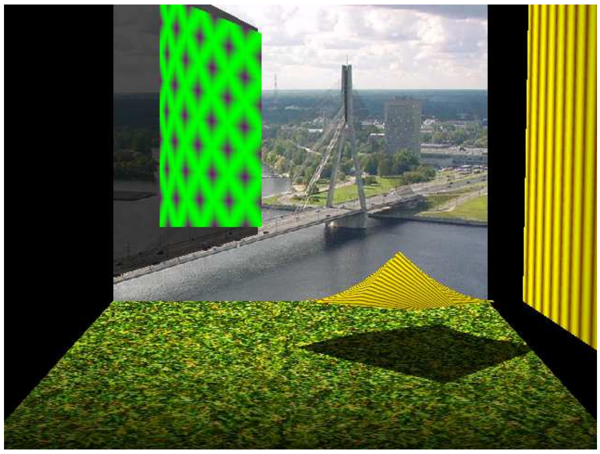


Figure 3-a: Scene with texturing and shading (Ray Tracing).

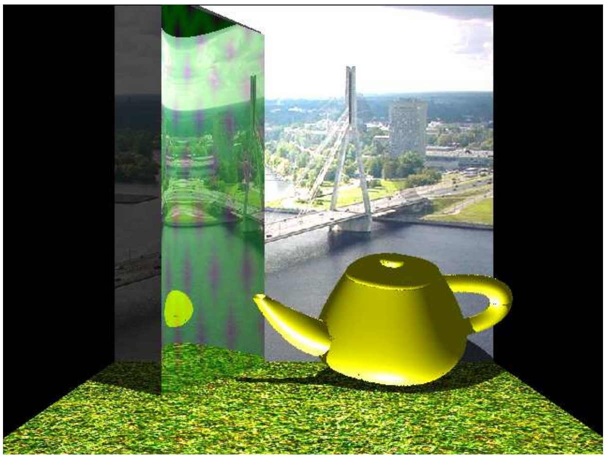


Figure 4-a: Scene with biquadric surfaces (Ray Tracing).

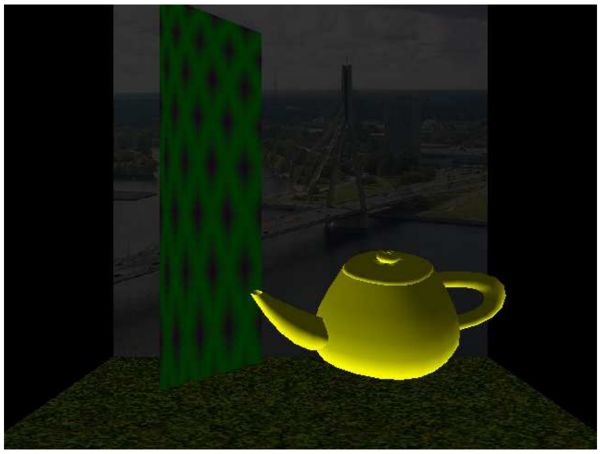


Figure 4-b: Scene with biquadric surfaces (OpenGL).

Figure 5-a: Scene with bicubic surfaces (Rat tracing)

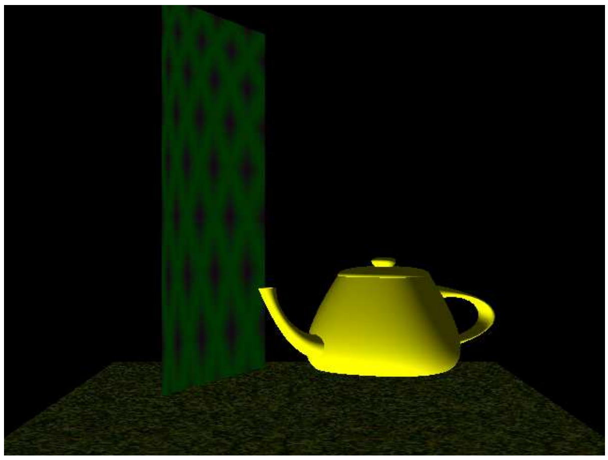


Figure 5-b: Scene with bicubic surfaces (OpenGL).

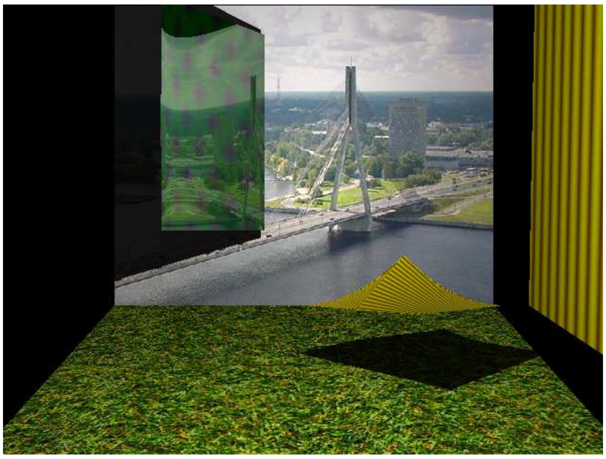


Figure 6: Scene w ith texturing, shading and reflection (Ray Tracing)

4. Conclusion

The received results reflected in a fig. 1 - 4, show, that:

- Ray tracing does not give advantages at construction of surfaces at absence of illumination. The classical methods with quite precise interpolation cope with this task.
- Ray tracing gives advantages at construction of surfaces with modeling of illumination. The classical methods solve this task with worse quality.
- Ray tracing allows correctly and precisely fulfill modeling of shading. The methods which are realized in OpenGL do not allow this and require developing of additional algorithms for this task
- Ray tracing allows correctly and precisely execute modeling reflection, including from free – form surfaces. The methods realized in OpenGL do not allow it and the development of separate algorithms for realization of this task required.

5. Acknowledgements

This work has been partly supported by the European Social Fund within the National Programme „Support for the carrying out doctoral study programm's and post-doctoral research" project „Support for the development of doctoral studies at Riga Technical University"

6. References

1. D.F. Rogers, J.A. Adams. *Mathematical Elements for Computer Graphics, 2nd Ed.*, McGraw-Hill, Boston, MA, 1990.
2. D. Hearn, M.P. Baker *Computer Graphics with OpenGL, 3rd Ed.,* Prentice Hall, 2004.

3. Д. Химмельблау *Прикладное нелинейное программирование.* – М.: Мир, 1975.
4. T. Nishita, T.W. Sederberg and M. Kakimoto. *Ray Tracing Trimmed Rational Surface Patches.* Computer Graphics, 24(4), 1990.

Sisojevs, A. Glazs, A. Brīvas formas virsmu vizualizācija ar staru trasēšanas metodi
Darbā tiek apskatītas bilineāro virsmu vizualizācijas metode un to augstas kvalitātes attēlu iegūšana. To darba metodes balstās uz staru trasēšanas modeli. Bilineāra virsma un stars tiek aprakstīti parametriskā veidā, kurš balstās uz Bernšteina bāzi. Apskatītas tās sastāvdaļas īpašības, kuras rodas staru trasēšanas izmantošanas gadījumā.

Sysojev, A. Glaz, A. Visualization of free – form surfaces by a ray tracing method
In the present work the method of rendering of free – form surfaces and creations of their high-precision images is considered. This method is based on a model of ray tracing.
A free - form surface and ray are described in a parametrical kind, which is based to base of Bernshtein. Those features component are considered which occur in case of use of ray tracing.

Сысоев, А. Глаз, А. Визуализация криволинейных поверхностей методом трассировки лучей
В работе рассмотрен метод визуализации произвольных криволинейных поверхностей и создания их высокоточных изображений. Этот метод основывается на модели трассировки лучей. Произвольная криволинейная поверхность и луч описываются в параметрическом виде на основе базиса Бернштейна. Рассмотрены те особенности составляющих, которые появляются в случае использования трассировки лучей.