One Approach for Overlaying with Polygon Meshes

Emiliyan Petkov[1]

***Abstract* – Covering of three-dimensional objects is a part of the modeling process and building virtual three-dimensional scenes in the graphics systems. This research examines a case in which faces of a freely located 3D object must be overlaid with a 3D object as well. The last one should be duplicated and arranged over the faces.**

***Keywords* – Computer, Graphics, Systems, Modelling, Overlay, Polygon, Mesh.**

I. Introduction

From almost the very inception of computer graphics techniques for covering two-dimensional (2D) and three-dimensional (3D) objects in computer generated virtual spaces with bitmap images and textures are developed [1]. This is a good and relatively simple way to achieve greater realism in the output images. More over, covering with textures achieves the effect of three-dimensional relief on the otherwise flat faces of the objects (Fig. 1). This significantly reduces the number of vertices which leads to faster rendering [4]. So, when we look for fast rendering and a small volume of information in the representation of the models in the graphics systems the best approach is to cover with textures.

Fig. 1. A 2D texture that is placed over a 3D tore.

However, there are tasks and applications which seek maximum realism in computer synthesized images. This means that each object must be represented by three-dimensional model in the scene [2, 4]. Such a high level of realism is sought in fields such as architecture design, film industry, simulations, virtual reality, etc. When creating images comply with this requirement the rendering of the scenes is not needed to be in real time, i.e. rendering of one frame can be carried out within a few minutes and if necessary even hours [6]. The problem here is the volume of information that must be processed and stored. But when the applied field does not require this to be done in real time then this volume can be as much as the computer system allows. This research examines cases that require maximum realism – in which "just an effect" is not enough. These are the cases in which the time for rendering and the volume of information of the 3D objects in the scene are not critical.

There are several types of representation of 3D objects in computer graphics and graphics systems [1, 4, 6]. The most popular are: polygon mesh, patch, NURBS, Bezier, etc. And the most natural one is polygon mesh. Objects of all other types uniquely are converted to polygon mesh as well. Under polygon mesh or mesh we understand: collection of vertices, edges and faces that defines the shape of a polyhedral object in 3D computer graphics and solid modelling [4].

This report presents a solution of a problem which was given by a business company called Monblan Design Ltd., Bulgaria. This problem concerns overlaying faces of 3D objects with polygon meshes in graphics systems. For the last few years the problem stands before many architectures, designers, animators, etc. who work on computer graphics software.

The problem could be defined as follows: A mesh is given that is freely located in space (Fig. 2). We may call it *base*. A face or a group of faces that belong to the base have to be overlaid with another three-dimensional object. The last one must be polygon mesh as well and we may call it *target*. The target must be duplicated and arranged in two dimensions orthogonal each other over the face. We may call these new objects *instances*. If a set of faces are selected they must lie in one plane and we may call them *polygon*.

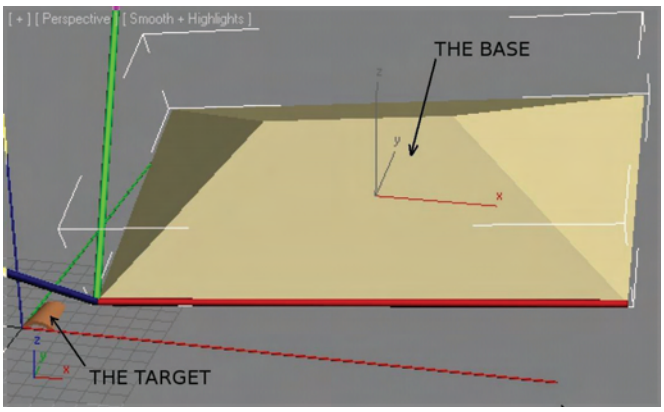


Fig. 2. A base and a target in 3D space.

If a face or a group of faces which belong to a base in 3D space and a coordinate system is fixed in it are collinear to one of the coordinate planes Oxy, Oxz or Oyz, then the overlaying can be accomplished only with translations along

[1]Emiliyan Petkov is with the Faculty of Mathematics and Informatics, St. Cyril and St. Methodius University, 2 Theodosii Turnovski Str., 5000 Veliko Turnovo, Bulgaria, e-mail: EPetkov@abv.bg.

the *X* axis, *Y* axis or *Z* axis. This could be easily realized in any graphics system. But these are only individual cases. There is a problem when the base is freely located in space. There are no tools in the graphics applications that can solve this problem.

II. THE SOLUTION

A. The main idea for solving the problem

Creation and setting up the base in a 3D space and creation of the target and applying transformations over it, as well as defining how the target will be positioned over the polygon, must be done in a fixed coordinate space. In computer graphics and graphics systems an orthonormal coordinate system is chosen generally [3, 5, 7].

Overlaying of the base (Fig. 2) with the target must be done in two directions. Let mark them with $\vec{u}$ and $\vec{v}$. In Fig. 2 $\vec{u}$ and $\vec{v}$ are drawn as red and green respectively. These directions could be determined as follows: one vertex from the polygon is determined as origin, an edge which comes from it determines $\vec{u}$ direction, and $\vec{v}$ is determined as an orthogonal vector to $\vec{u}$, where $\vec{u}$ and $\vec{v}$ belong to the plane that is defined by the faces (polygon). Then we could define the values of the distances between the duplicated instances of the target in both $\vec{u}$ and $\vec{v}$ directions.

There is one more question: How to define the position and orientation of the target over the polygon that we want to overlay? The most intuitive way (according to the author) is to locate and orientate the target in relation to the Oxy plane near to the origin. In Fig. 2 the target which is placed near the origin could be seen. In accordance to how the target is put to the origin in Oxy plane it will be located and arranged over the polygon. But this raises the following question: What is the transformation that can do this?

The idea is to find first the transformation that transforms every point from the polygon in Oxy plane where the target lies. After this we can take the reverse transformation which can transport the target over the polygon.

B. Mathematical formulation of the problem

Let an orthonormal coordinate system $K = \{O, \vec{e_1}, \vec{e_2}, \vec{e_3}\}$ be fixed in space (Fig. 3). The group of faces which have to be overlaid must lie in one plane. Let this plane be β. Let all vertexes that belong to the polygon be $A_i(x_i, y_i, z_i)$, $i = 0,..,n$, $n \geq 2$ (every face is composed by three vertexes). Let mark $\sphericalangle\left(\left|\overrightarrow{A_0A_1}\right|\left|\overrightarrow{A_0A_n}\right|\right)$ with α. For β we have:

$$\beta : Ax + By + Cz + D = 0 \tag{1}$$

$$\overrightarrow{A_0A_1} \times \overrightarrow{A_0A_n} = \vec{N} = (A, B, C) \tag{2}$$

$$\vec{n} = \frac{\vec{N}}{-sign(D).\left|\vec{N}\right|} = (n_1, n_2, n_3) \tag{3}$$

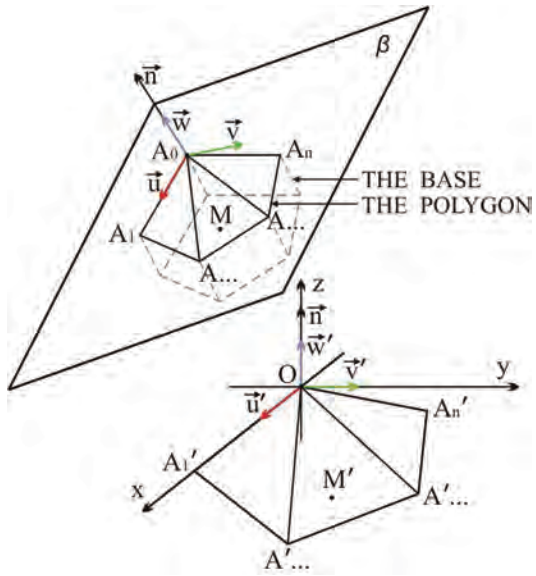


Fig. 3. The selected faces (polygon) that lie in β and their transformation by φ in Oxy.

We are seeking for a transformation of uniformity φ which is determined as follows:

$$\begin{aligned}
&\beta \xrightarrow{\varphi} Oxy \\
&A_0 \xrightarrow{\varphi} O(0,0,0) \\
&A_1 \xrightarrow{\varphi} A_1^{'}(x_1^{'}, 0, 0) \\
&A_n \xrightarrow{\varphi} A_n^{'}(x_n^{'}, y_n^{'}, 0) \\
&N(n_1, n_2, n_3) \xrightarrow{\varphi} N^{'}(0,0,1)
\end{aligned} \tag{4}$$

where:

$$x_1^{'} = \left|\overrightarrow{A_0A_1}\right| = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2 + (z_1 - z_0)^2}$$

$$\sphericalangle\left(\left|\overrightarrow{A_0A_1^{'}}\right|\left|\overrightarrow{A_0A_n^{'}}\right|\right) = \alpha$$

$$\left|\overrightarrow{OA_n^{'}}\right| = \left|\overrightarrow{A_0A_n}\right| \tag{5}$$

To find φ, $A_1^{'}$ and $A_n^{'}$ have to be uniquely determined. But Eqs. (4) and (5) shows that $x_1^{'}$ and α are known. In the conditions that were set in Eqs. (4) two parameters are unknown. These are the coordinates of $A_n^{'}$: $x_n^{'}$ and $y_n^{'}$. For α we have:

$$\cos\alpha = \frac{\overrightarrow{A_0A_1}.\overrightarrow{A_0A_n}}{\left|\overrightarrow{A_0A_1}\right|.\left|\overrightarrow{A_0A_n}\right|} = b \text{ – known,}$$

$$\alpha = \arccos\frac{\overrightarrow{A_0A_1}.\overrightarrow{A_0A_n}}{\left|\overrightarrow{A_0A_1}\right|.\left|\overrightarrow{A_0A_n}\right|}. \tag{6}$$

Considering the Eqs (6) and the conditions from Eqs (4) a system of equations is set:

$$\begin{cases} \cos\alpha = \dfrac{\overrightarrow{A_0A_1'}.\overrightarrow{A_0A_n'}}{\left|\overrightarrow{A_0A_1'}\right|.\left|\overrightarrow{A_0A_n'}\right|} = b \\ \sqrt{x_n'^2 + y_n'^2} = \left|\overrightarrow{A_0A_n}\right| \end{cases} \quad (7)$$

After elementary transformations over the Eqs. (7) the result is:

$$\begin{cases} x_n' = \left|\overrightarrow{A_0A_n}\right|.\cos\alpha \\ y_n' = \pm\left|A_0A_n\right|.\sin\alpha \end{cases} \quad (8)$$

C. Solution of the problem

Every transformation in space could be presented in matrix mode. Therefore, we are seeking of a transformation matrix A and a vector of translation $\vec{p}$ such that

$$\varphi : \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \underbrace{\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}}_{A} . \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \underbrace{\begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix}}_{\vec{p}} \quad (9)$$

and the conditions from Eqs. (4) to be performed. So, we are receiving the following system of four equations:

$$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = A. \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} + \vec{p} \quad (10)$$

$$\begin{pmatrix} x_1' \\ 0 \\ 0 \end{pmatrix} = A. \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + \vec{p} \quad (11)$$

$$\begin{pmatrix} x_n' \\ y_n' \\ 0 \end{pmatrix} = A. \begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} + \vec{p} \quad (12)$$

$$\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = A. \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} + \vec{p} \quad (13)$$

Considering Eqs. (10), (11), (12) and (13) as a system an elementary transformations are made and the result is:

$$\underbrace{\begin{pmatrix} x_1' & x_n' & 0 \\ 0 & y_n' & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{C} = A. \underbrace{\begin{pmatrix} x_1 - x_0 & x_n - x_0 & n_1 - x_0 \\ y_1 - y_0 & y_n - y_0 & n_2 - y_0 \\ z_1 - z_0 & z_n - z_0 & n_3 - z_0 \end{pmatrix}}_{B} \quad (14)$$

Matrixes B and C are known. We are seeking for matrix A and a vector $\vec{p}$. The matrix is found as follows:

$$A.B = C \quad \Leftrightarrow \quad A = C.B^{-1} \quad (15)$$

The translation vector $\vec{p}$ is found by replacing A in Eqs. (10):

$$\vec{p} = -A. \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} \quad (16)$$

Therefore, the transformation of uniformity φ is already known. Now we are ready to find the reverse transformation of φ. This will be φ^{-1} that is determined as follows:

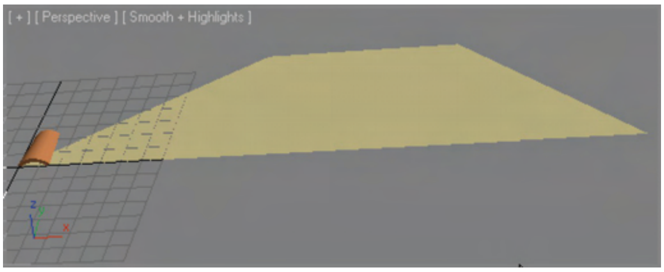
$$\varphi^{-1} : \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = A^{-1}. \begin{pmatrix} m_1' \\ m_2' \\ m_3' \end{pmatrix} - A^{-1}. \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix} \quad (17)$$

For every point $M'\left(m_1', m_2', 0\right)$ that belongs to Oxy we have a transformation φ^{-1} which transforms M' (Fig. 3) in point $M\left(m_1, m_2, m_3\right)$ which belongs to plane β and φ^{-1} satisfies the conditions in Eqs. (4). So, we have:

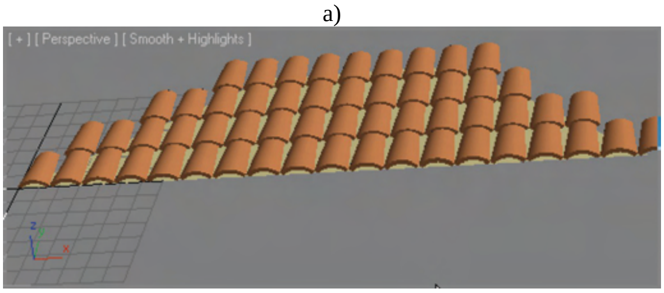
$$A_i \xrightarrow{\varphi} A_i'\left(x_i', y_i', 0\right), \quad i = 0,..,n, \quad n \geq 2 \quad (18)$$

Transformation φ is a transformation of uniformity, therefore, we have the polygon in Oxy (Fig. 4a). The polygon has dimensions along x and y axis that define a rectangular area. Le this area be determined by $x_{\min}$, $x_{\max}$, $y_{\min}$ and $y_{\max}$ that are determined as follows:

$$\begin{aligned} x_{\min} &= \min\left\{x_i, i = 0,..,n\right\} \\ x_{\max} &= \max\left\{x_i, i = 0,..,n\right\} \\ y_{\min} &= \min\left\{y_i, i = 0,..,n\right\} \\ y_{\max} &= \max\left\{y_i, i = 0,..,n\right\} \end{aligned} \quad (19)$$



a)



b)

Fig. 4 – a) the polygon is transformed by φ;
b) the polygon is overlaid with the instances.

Now the target could be duplicated and arranged in this rectangular area along both x and y axis. The instances that do not belong to the polygon could be removed (Fig. 5b). Thus we have a overlaying of the polygon but it lies in Oxy. Using transformation φ^{-1} the polygon together with the instances is transformed to its primary position in space. The result could be seen in Fig. 5.

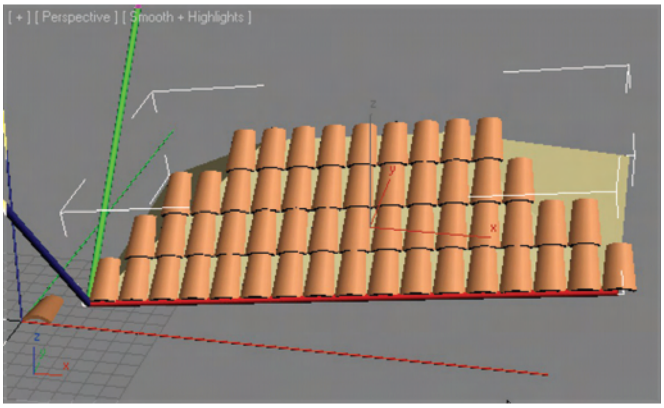


Fig. 5. The target has been duplicated and arranged over the polygon.

After we have the result a Boolean operation "cutting" could be applied over the instances. They may be cut with the outline of the polygon. This is shown in Fig. 6.

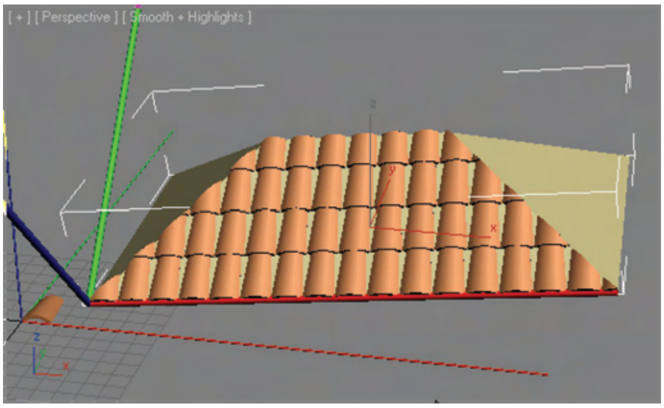


Fig. 6 – The instances of the target are cut.

III. Conclusion

In this paper an approach for overlaying faces of 3D objects with polygon meshes in computer graphics and graphics systems has been presented. It is well mathematical grounded and could be applied in any graphics system as additional software tool.

Figures 2, 4, 5 and 6 have been generated in Autodesk 3ds Max using Overlay software module (plug-in) which has been developed using the presented here solution. Overlay plug-in overview and trial version are available at http://www.creativecrash.com.

Acknowledgement

This paper is financed by project: Creative Development Support of Doctoral Students, Post-Doctoral and Young Researches in the Field of Computer Science, BG 051PO001-3.3.04/13, European Social Fund 2007–2013, Operational Program "Human Resources Development".

References

[1] Buss, S. R. 3-D Computer Graphics - A Mathematical Introduction with OpenGL. Cambridge University Press, UK, 2003.

[2] Gallier, J. Curves and Surfaces in Geometric modelling – theory and algorithms. Morgan Kaufmann Publishers, USA, 2000.

[3] Lowther, J. L., Ching-Kuang Shene. Computing with Geometry as an Undergraduate Course: A Three-Year Experience. ACM: SIGCSE 2001 2/01 Charlotte, NC, USA, 2001.

[4] Hill F. S., Stephen M. Kelley. Computer graphics using OpenGL. Pearson International edition, Prentice Hall, NJ 07458, USA, 2007.

[5] Rogers, D. F., J. A. Adams. Mathematical elements for computer graphics, second edition. McGraw-Hill, USA, 1990.

[6] Wang, H., J. Kearney, K. Atkinson. Curve and Surface Design. Saint-Malo 2002, Nashboro Press, Brentwood, TN, 2002, pp. 387–396.

[7] Weisstein, W. Eric. MathWorld – Web comprehensive and interactive mathematics encyclopedia. http://mathworld.wolfram.com.