Formation of Microrelief on the Base of Multilevel Elevations Maps

Sergei I.VYATKIN[1], Oksana V. ROMANIUK[2]
[1]Institute of Automation and Electrometry, SB RAS,1 prosp. Koptiuga, Novosibirsk Russia,
[2]Vinnitsa National Technical University, Ivana Bohuna street, 119, Vinnitsa-21100, Ukraine

***Abstract* — There had been considered the issues on usage of perturbation functions for microrelief rendering. There had been suggested the methods for prescribing and rendering of microrelief on the base of multi level elevation map. There had been shown the advantages before the existent methods of microrelief rendering. In comparison with the known methods in the suggested methods, the time of calculations during the relief generation practically does not depend on elevation map resolution. In the suggested method of microrelief generation for the microrelief representation and levels of detail changing the same mechanism is used as for the usual texture.**

***Index Terms* — microrelief, multilevel elevation maps, multilevel ray-tracing, perturbation function, rendering**

I. INTRODUCTION

Contemporary graphic processors use the collection of methods for rendering microrelief of 3D objects surfaces [1]. Their application allows to represent the structure of an object surface more realistically.

There are two main methods for microrelief rendering: on the geometry level and on the level of flat surface rendering. Microrelief representation on the geometry level is reached due to object triangulation with further displacing of new vertices to needed position (displacement mapping). Second method of microrelief rendering is based on changing the illumination of bumps and displacing texture coordinates for sampling from material textures. Today the following varieties of such method are popular: normal mapping [2]), parallax mapping [3] and relief mapping [4, 5]. There are also many modifications of them, which eliminate their different drawbacks, for example spherical harmonic relief mapping and others.

Normal mapping method uses the special texture that contains information on normals (direction vectors) at the points of object surface. Such a method allows approximately to reproduce surface relief by visible distinctions in illuminations of bumps.

Parallax mapping uses normal mapping and one-channel height map which, as a rule, is stored as fourth component of normal map. For relief imitation at visible points the value of texture coordinates offset is calculated. Then new coordinates are used for sampling from the texture. Offset value is calculated from data in height map and viewer position.

Relief mapping is the form of simplified local ray-tracing. Tracing is used for finding the nearest visible point in the direction from viewing point. This point determines texture coordinates which are used as in parallax mapping. There are different ways to find visible point. One of them is a linear search with further binary search which finds more accurate value.

This paper considers the method of microrelief formation based on multilevel elevation maps. The principle distinction of suggested method before known methods is unpolygonal surface representation and application of multilevel ray-tracing for scene rendering. Application of multilevel ray-tracing method eliminates disadvantages which are typical for systems of rasterization on the plane and voxel-based systems.

II. MICRORELIEF BASED ON MULTILEVEL ELEVATION MAPS

Considering the geometric model structure, microrelief is created with the help of base surface and perturbation function [6] which is determined inside the infinite long parallelepiped. The values of perturbation function are set in the cross-section of the parallelepiped with two-dimensional elevation map. The direction of the normal to the base surface should coincide with longitudinal direction of the parallelepiped that is definitional domain of perturbation function according to fig.1.

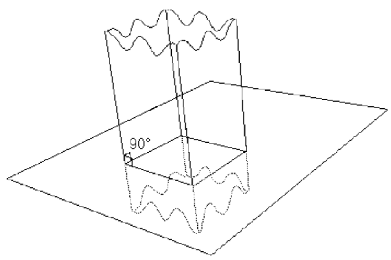

Figure 1. Definitional domain of perturbation function.

Realization of the perturbation function requires to create an object of geometric model, initial data for which is a sample of values located in the points of rectangular mesh which is formed on the base of elevation measurements. As during the rasterization the perturbation should evaluate the maximum of its function in three-dimensional or one-dimensional domain, then for the productive calculations levels-of-detail maps are formed preliminarily. Initial data form level n if mesh dimension is 2^nx2^n. Data for level $n-1$ are got by selection the maximum value from four adjoining values of level n, the rest three values are not considered then, i.e. we get mesh dimension 2^{n-1}x2^{n-1} . Level 0 consists of single value – the maximum over the whole elevation map. The process of preparing levels-of-detail maps is shown on fig.2.

During the determination of the perturbation maximum, the characteristic size of projection of current domain is calculated, proceeding from which the level of detail is chosen. For larger interval the rougher approximation of input function is chosen. If more precise presentation is needed, bi-linear or bi-quadratic interpolation of elevation values, which form last level of detail, is conducted. Such an approach allows to decrease the number of calculations due to increasing memory capacity for storing additional data and mainly due to dynamic adjusting complexity of calculations under criterion of required accuracy of result.

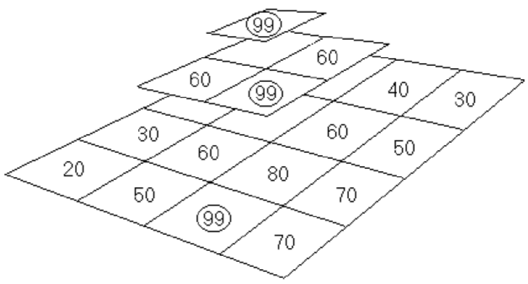

Figure 2. Levels-of-detail maps.

Let's determine the concepts and terms which will be used further. We call the open simply connected set of points on the plane as a domain of the plane. Let D be the plane domain and $\overline{D}$ its closure. We enter the coordinate system (u,v) on the plane. Let x, y, z be the rectangular Cartesian coordinates of the points in the 3D Euclidean space E^3 . Prescribe three continuous functions on the set $\overline{D}$:

$$x = \varphi(u,v), \ \ y = \psi(u,v), \ \ z = \chi(u,v). \tag{1}$$

We assume that functions (1) have the following property. If (u_1,v_1) and (u_2,v_2) are different points of the set $\overline{D}$, then $M_1(x_1,y_1,z_1)$ and $M_2(x_2,y_2,z_2)$ of the space E^3, whose coordinates were calculated by formulas (1), are also different:

$$x_1 = \varphi(u_1,v_1), \ \ y_1 = \psi(u_1,v_1), \ \ z_1 = \chi(u_1,v_1),$$
$$x_2 = \varphi(u_2,v_2), \ \ y_2 = \psi(u_2,v_2), \ \ z_2 = \chi(u_2,v_2). \tag{2}$$

The set S of the points $M(x,y,z)$ which coordinates x,y,z are determined by (1), where the functions φ, ψ and χ in the closure $\overline{D}$ of the domain D possess the described property, is called a simple surface.

The simple surface that is a plot of the function defined in the 3D space $z = f(x,y)$ is referred to as the freeform surface F .

The freeform surface representation based on the scalar field is a totality of a base surface P (in the same coordinate system as F) and the related elevation map.

We may use any surface as the base surface, however, surface used in practice are simple surfaces such as planes, ellipsoids, or cylinders.

The elevation map is a 2D rectangle called hereafter a perturbation domain D_P of the base surface P , and the perturbation function $h(u,v)$ is given inside this rectangle. The elevation map in turn determines the perturbation. The domain of $h(u,v)$ is $D_{h(u,v)} = \{U,V\}$, where U and V are the size of the rectangle.

The elevation map and the base surface are related as follows: there exists a transformation $G(\Re^3 \Rightarrow \Re^2)$ from the coordinate system of F and P to coordinate system of the map. This transformation is usually a parallel projection.

The value of $h(G(d_F))$ characterizes the deviation of the point d_F , on the surface F from the point, from the point d_P that is the projection of this point onto the surface P . In other words, the value of $h(G(d_F))$ is equal to the scalar of vector

$$\vec{v} = (d_F - d_P). \tag{3}$$

Therefore, the domain of the freeform surface can be defined as a set of point in $\Re^3$, which are defined by the vector equation

$$\vec{F} = G(\vec{v}) + \vec{n} \cdot h(G(\vec{v})); \forall \vec{v} \in \Re,^3 \tag{4}$$

where $\vec{n}$ is the normal to the base surface.

If the vector $\vec{v}$ is outside the perturbation domain, the vector $\vec{n} \cdot h(G(\vec{v})) = 0$, and $\vec{F}$ is the vector on the base surface. Thus, for prescribing the form of the perturbing surface we can use a table of numbers, and the function h can be represented by a function of interpolation by pivotal values taken from the table (fig. 3).

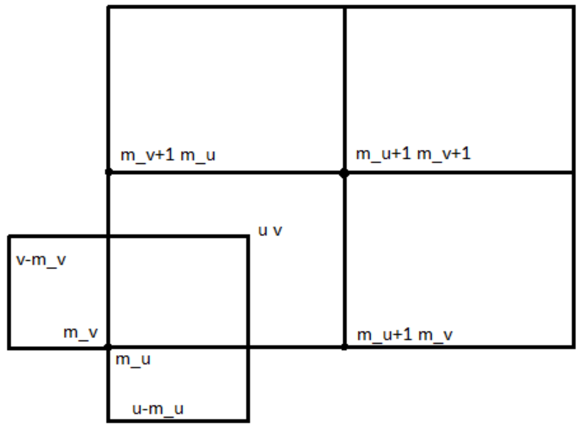

Figure 3. The table of elevation map numbers.

In this case, we may assume that a scalar field is given in the perturbation domain D_P .

Thus, for prescribing the freeform surface F based on the scalar field (fig. 4) there are needed the base surface, transformation from coordinate system of the base surface into coordinate system of the elevation map, the perturbation function and the table of numbers, which characterizes the deviation of the surface F from the base one at check points.

III. RENDERING

For rendering the method of multi-level ray-tracing [7,8] is used. That is through each pixel of the plane the ray from the viewer through the viewing pyramid (object space) is sent. These rays are projected onto the base surface e. Isolation of the surface shape is realized of the stage of ray division (voxel bounded by front and back edges of the viewing pyramid) over *z*-coordinate. One-dimension bar (voxel) V_0 is set by pair of vectors $P_0 = (x_0, y_0, z_0)$ and $P_1 = (x_1, y_1, z_1)$, $V_0 = \{P_0, P_1\}$. Then, the coordinates of voxel V_0 are transformed into coordinate system of elevation map using G transformation.

$$\{(x_0, y_0, z_0), (x_1, y_1, z_1)\} \Rightarrow \{(u_0, v_0, h_0), (u_1, v_1, h_1)\} \quad (5)$$

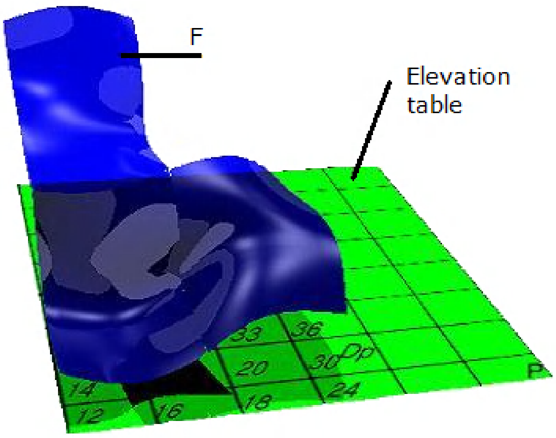

Figure 4. The perturbation function and the base surface.

For transformation into coordinate system of elevation map the transformation matrix T is used, which being multiplied on matrix M of geometric transformations gives resulting transformation matrix $G = T \times M$.

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Then the transformed coordinates (u, v, h, a) of the voxel in the coordinate system of elevation map are calculated from the (x, y, z) coordinates of the voxel in object space by multiplying the vector of the point in object space on matrix G .

$$G \times \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ h \\ a \end{bmatrix} \quad (7)$$

Then division of the voxel over *z*-coordinate or binary division of the voxel is performed (fig.5).

On this stage for current level of recursion the vector of the end of the voxel that is nearest to the viewer is equal the vector of nearest end of the voxel of previous level of division. The vector of distant end of the voxel is calculated as a half a sum of nearest and distant ends of the voxel of previous level of division.

$$P_{ni} = P_{ni-1},\ P_{fi} = \frac{(P_{ni-1} + P_{fi-1})}{2},\ V_i = \{P_{ni}, P_{fi}\} \quad (8)$$

where V_i is a voxel of the *i-th* level of recursion, and P_{ni}, P_{fi} are coordinates of nearest to the viewer and distant ends of the voxel on the *i-th* level of recursion. Thus, geometric transformations for a voxel are done only once on the first level of binary division, and then the process of voxel division into two parts and choosing the nearest intersection of the voxel is performed.

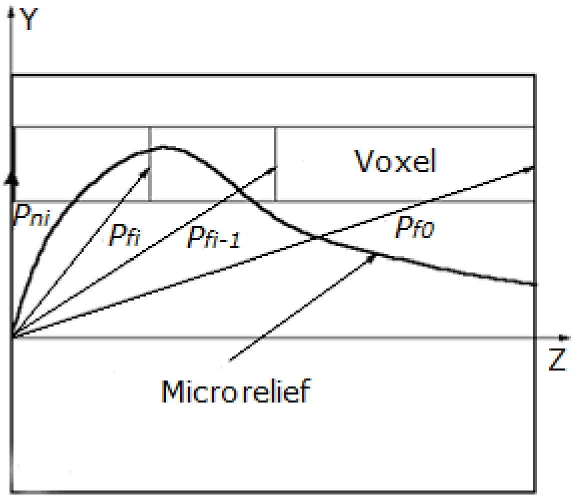

Figure 5. Binary division of the voxel.

By the size of the voxel projection on appropriate level of recursion the level of detail is calculated which is also an elevation map, but of lower resolution and by the u and v coordinates of P_{ni} and P_{fi} points the selection of maximum value from the table of numbers is performed, which represent this level of detail.

On the each stage of voxel binary division the level of detail *"level"* is calculated by the size of the voxel. If *'level'* is not the last level of detail, then obtained elevation *'h'* is compared with elevation value of given level *'Hmax'*, and if *'h>Hmax'*, then division of the voxel breaks. Thus, calculations are done in two stages.

1. Determine dimension of the rectangle which is the voxel projection onto elevation map as maximum distance from point $\{u_0, v_0\}$ to point $\{u_1, v_1\}$ – L_P (fig.6).

2. From the inequality $\frac{1}{2^{level}} < L_P < \frac{1}{2^{level+1}}$ determine level of detail *'level'*.

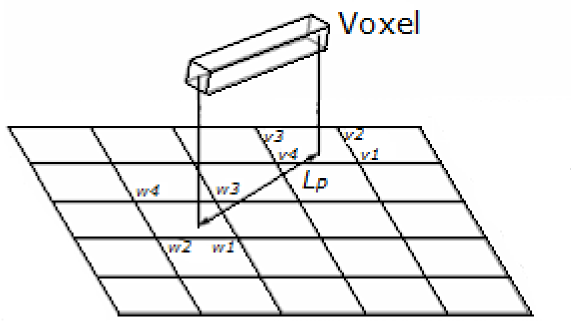

Figure 6. Determination of level of detail by the voxel projection of current level.

The results of microrelief modeling is shown on fig. 7 and fig. 8.

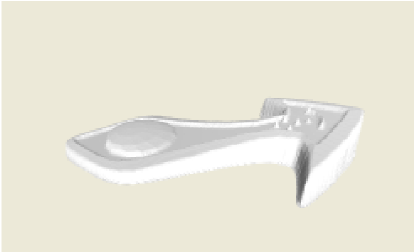

Figure 7. Microrelief.

Figure 8. Front view of microrelief.

IV. CONCLUSIONS

The conducted researches on technology of microrelief rendering allowed to reveal a number of advantages both in method of microrelief prescribing and in rendering methods. Transfer from rendering in image plane to volume rendering in combination with the suggested method of microrelief prescribing leads to positive moments which increase the realism of the rendered objects, although it increases the number of calculations performed in real time.

In the method of relief representation and levels of detail changing the same mechanism is used as for usual texture. Time of calculations during relief generation practically does not depend on elevation map resolution (fig. 9 and 10).

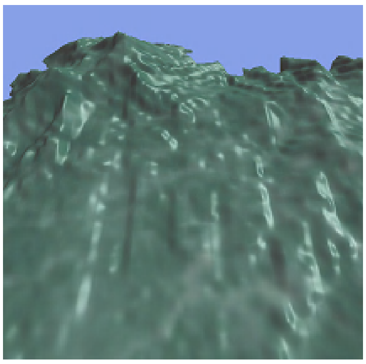

Figure 9. Microrelief. Elevation map with resolution 64x64.

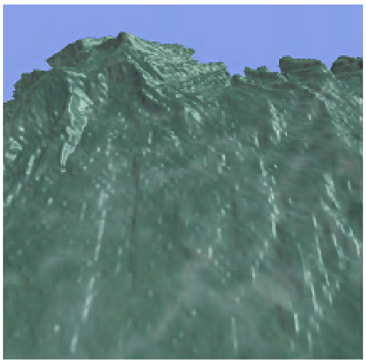

Figure 10. Microrelief. Elevation map with resolution 128x128.

REFERENCES

[1] T. Akenine-Möller, E. Haines, N. Hoffman. "Real-Time Rendering", third edition. – A K Peters, 2008.

[2] I. Ernst, D. Jackèl, H. Rüsseler, O. Wittig. "Hardware Supported Bump Mapping: A Step towards Higher Quality Real-Time Rendering". – Proceedings of 10th Eurographics Workshop on Graphics Hardware, 1995. – P. 63-70.

[3] T. Welsh. "Parallax Mapping with Offset Limiting: A PerPixel. Approximation of Uneven Surfaces". – Infiscape Corp. Tech Report, 2004.

[4] Manuel M. Oliveira, Gary Bishop, David McAllister. "Relief Texture Mapping". Proc. SIGGRAPH 2000. - P. 324 - 331.

[5] F. Policarpo, M. Olivers, J. Comba. "Real-Time Relief Mapping on Arbitrary Polygonal Surfaces". – Proceedings of ACM SIGGRAPH 2005. – P. 155-162.

[6] Vyatkin S.I, Romaniuk O.V. "The Method of Relief Surface Image Formation" // Data Rec., Storage & Processing. – 2009. – Vol. 11, №4. – P. 51-58.

[7] Vyatkin S.I., Dolgovesov B.S., Yesin A.V. et al. "Voxel Volumes volume-oriented visualization system" // International Conference on Shape Modeling and Applications (March 1-4, 1999, Aizu-Wakamatsu, Japan) IEEE Computer Society, Los Alamitos, California, 1999. - p.234-241.

[8] Vyatkin S.I., Dolgovesov B.S., Guimaoutdinov O.Y. "Synthesis of virtual environment using perturbation functions" // volume III (Emergent Computing and Virtual Engineering), World Multiconference on Systemics, Cybernetics and Informatics Proceedings, Orlando, FL, USA, July 22-25, 2001. - p.350-355.