A Method for Modeling and Rendering Dunes with Wind-ripples

Koichi Onoue Tomoyuki Nishita
Graduate School of Frontier Sciences, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-0033, Japan
{onoue,nis}@is.s.u-tokyo.ac.jp

Abstract

This paper proposes a method for modeling and rendering realistic desert scenes. A desert terrain includes sand dunes and wind ripples. We use two types of scale models to form them. We render the dunes with the wind-ripples by bump-mapping using LODs (Levels of Detail).

Keywords: Natural phenomena, Dunes, Wind-ripples, Levels of Detail, Texture mapping, Bump mapping

1 Introduction

In computer graphics, natural phenomena such as clouds, smoke, and trees have been studied by many researchers. But the modeling and rendering desert scenes has not been studied. In this paper, we discuss a modeling method of desert scenes by simulating the motion of sand grains. Then the rendering method of simulation results is proposed.

A desert terrain includes sand dunes and ripple patterns called wind-ripples formed on a sand surface. The formation dynamics of wind-ripples and dunes have been investigated for a long time, and many of them are reviewed by Pye [1]. Nishimori et al. proposed a simple model for formation dynamics of two scales of sand features [2], but their purpose is not the rendering of desert scenes. So we propose a rendering method of the desert modeled by their method.

In rendering process, we should render dunes with wind-ripples. Because dunes and wind-ripples are formed separately in the modeling process, we must consider how to combine the two models. To reduce the calculation time and to render the scene realistically, we represent the wind-ripples by bump-mapping onto the dunes.

2 Modeling of Wind-Ripples and Dunes

In this chapter, Wind-ripple (small-scale) model is first described, then Dune (large-scale) model is explained.

(1) Wind-ripple model

The model consists of two elementary processes: saltation and creep. The saltation means the jump of sand grains caused by a wind. Sand grains may move just along the sand surface without jumping up. This movement is called creep. The model is represented as a height field. We first show the saltation dynamics expressed as

$$H_{n'}(x,y) = H_n(x,y) - q,$$

$$H_{n'}(x + L(H_n(x,y)),y) = H_n(x + L(H_n(x,y)),y) + q,$$

where $H(x,y)$ is the height of the sand surface at each cell (x,y); q is the transferred height of grains; The time step is labeled n, and n' is the intermediate time step between n and $n+1$. Wind direction is the positive x. $L(H(x,y))$ is the flight length at one saltation, approximated as $L = L_0 + bH_n(x,y)$. Here L_0 is the control parameter proportional to wind force, and b is a constant.

To calculate the creep movement, grains at a cell are distributed around it using the diffuse equation. This means that the local height of a sand hill is relaxed by gravity.

The above two processes constitute one time step. We initialize the height field using random numbers, then we repeat the calculation for each time step until wind-ripples are formed.

(2) Dune model

There are two differences between the Dune model and the Wind-ripple model.

First, the saltation length L is assumed to be

$$L = L(\nabla \cdot H_n(x,y)) = L_0 - b\tanh(\nabla \cdot H_n(x,y)),$$

where L_0 and b are control parameters. The meaning of the equation is that, when it comes to large-scale model, a grain which jump at the windward side of a hill soon clashes into the hill. Second, the quantity of the grains being ejected out is much larger at the windward side of the hill. Therefore, the transferred height of grains is described as

$$q = q(\nabla \cdot H_n(x,y)) = q_0 + b'\tanh(\nabla \cdot H_n(x,y)),$$

where q_0 and b' are constants.

3 Rendering of Dunes with Wind-Ripples

In this chapter, we describe our rendering method of dunes with wind-ripples.

The outline of the rendering process is as follows.

1. Calculate wind-ripple's normal vectors.
2. For each quadrilateral F of the height field mesh of the dune:
   (a) Perform the backface culling and the view frustum culling.
   (b) Calculate LOD by the distance from the viewpoint to F.
   (c) Calculate the wind-ripples texture. The texture resolution is determined by LOD, and the texture is mapped onto F.

To represent the wind-ripple patterns on the dune surface, we generate the textures using the bump mapping by using the normal vectors which are calculated from the wind-ripples height field. This method allows us to render a realistic image without increasing the number of polygons.

The above method needs a large computation time because each quadrilateral has a different texture. To solve this problem, we use LODs technique. The basic idea of LODs is to use simpler versions of an object as it gets farther from the viewer. For the quadrilaterals far from the viewpoint, the texture is insignificant because the quadrilateral takes only a few pixels on the screen. Therefore, we use textures of different resolutions according to the LODs.

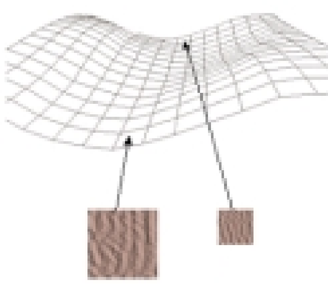


Figure 1. Bump-mapping using LOD

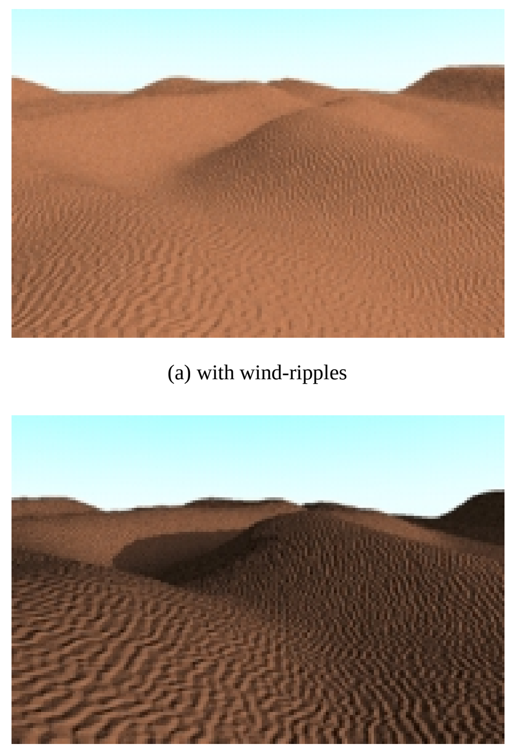
Our method is as follows. First, we calculate the normal vectors of wind-ripples. Using these vectors, we prepare the normal vectors for different resolutions. For example, when the original size of wind-ripple's height field is 256×256, we precalculate the normals of 256×256, 128×128, 64×64 and 32×32 resolutions.

Finally, we calculate the textures for each quadrilateral of the dune using the resolution determined by the quadrilateral's LOD. The LOD is calculated by the distance from the viewpoint to the quadrilateral. For example, as shown in Fig.1, a large texture is mapped onto a quadrilateral near the viewpoint, and vice versa. This method resembles mipmapping [3], but each texture is different. Therefore we cannot use mipmapping. We can save the texture calculation time by using the LODs, and we can render almost the same scene as the one without using LODs.

4 Results and Conclusion

The resulting images of the dunes with wind-ripples are shown in Fig. 2. The simulation and rendering are calculated using a Pentium III 750 MHz with a NVIDIA GeForce 256 video card. The size of the dune's height field mesh is 64×64, and the wind-ripple's size is 256×256. The size of textures for each quadrilateral of the dune is 64x64, using the repeated wind-ripples, and the image size is 720×480. The rendering time of Fig. 2(a) is 1.9 sec. A rendering result of dunes with shadows is shown in Fig. 2(b). The shadows are calculated by the hardware-accelerated shadow map method combined with the improved raytracing using height field features, and the rendering time is 3.5 sec.

The advantage points of the proposed method are as follows: (1) We use height field to represent the dunes and wind-ripples, and simulate their formation.(2) We render the dunes with wind-ripples using the bump mapping technique to efficiently obtain realistic images. (3) We employ LODs for texture/bump mapping and remove invisible polygons to reduce the calculation times.



(a) with wind-ripples

(b) with shadows (sun altitude is low)

Figure 2. Rendering results of dunes.

References

[1] K.Pye and H.Tsoar, *Aeolian Sand and Sand Dunes*, Unwin Hyman, London, pp.88-220, (1990).

[2] H. Nishimori and N. Ouchi, *Formation of Ripple Patterns and Dunes by Wind-Blown Sand*, Physical Review Letters Vol.71, No.1, pp.197-200, (July 1993).

[3] L.Williams, *Pyramidial Parametrics*, Computer Graphics, Vol.17, No.3, pp.1-11, (July 1983).