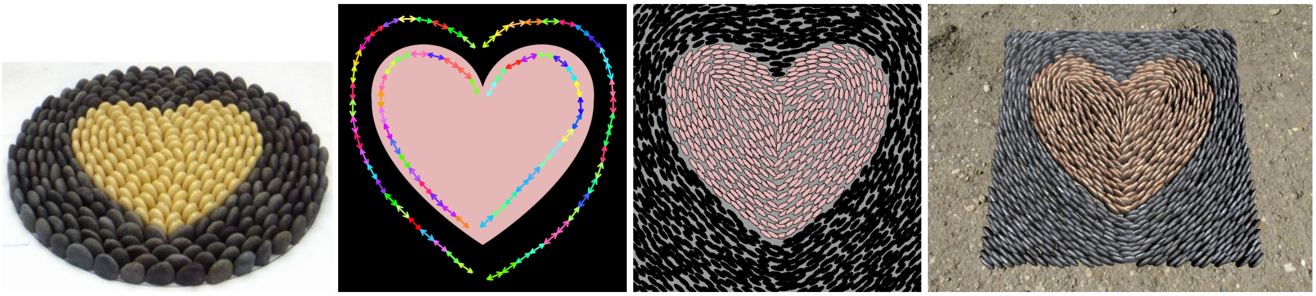
Interactive Procedural Modeling of Pebble Mosaics

Naoki Kita* Kazunori Miyata†
Japan Advanced Institute of Science and Technology



(a) Reference image (b) Stream curves (c) Generated pattern (d) Result

Figure 1: *Example: Heart design*

1 Introduction

In computer graphics, with improvement in hardware performance, more detailed production of three-dimensional (3D) models is required. However, the modeling process becomes tedious if it is carried out manually, in particular for natural objects because of their complexities. Therefore, it is important to generate content procedurally or automatically in order to efficiently produce such models. Furthermore, a purely procedural approach prevents the user from exercising total control over the outcome. The solution we proposed here is to combine the power of procedural modeling with the flexibility of manual editing. In this paper, we focus on the generation of pebble mosaics to achieve the distribution of numerous pebbles while maintaining distribution controllability. Existing research on rock modeling and digital mosaics has not focused on user-specified arrangements, whereas in this work, we present an interactive method for generating pebble mosaics. This is achieved through designing an underlying tensor field that specifies the pattern of pebble arrangement. The experimental results show that a realistic visual quality is obtained using our proposed method.

2 Related Work

Miyata et al. proposed a method that packs rectangles into two-dimensional (2D) specified regions and then creates rock shapes by subdividing the surface in order to generate a pavement [Miyata 01]. A method presented by Peytavie et al. [Peytavie 09] can generate aperiodic rock piles by using tiling, a 3D Voronoi diagram, and an erosion simulation. Hausner proposed a method for simulating decorative tile mosaics using a centroidal Voronoi diagram [Hausner 01]. The method is used to place tiles along a specified direction field; therefore, the generated mosaic images represent a fine flow of tiling.

3 Process Overview

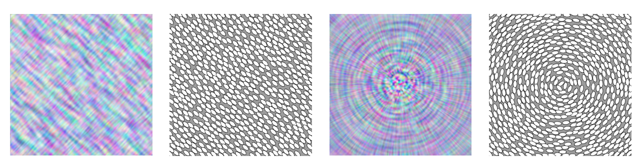
The proposed method generates an initial flow field on the basis of reference image gradients. Next, the user can edit the flow field via a stroke interface. Considering the tensor values of the flow field, site distribution is generated by a Poisson disk distribution using an elliptical disk. Pebble volumes are generated in each disk region. Therefore, our method generates 3D pebble mosaics by means of user-specified arrangements.

3.1 Tensor Field Generation

Most previously proposed methods employ a vector field for the flow field at which elements such as mosaic tiles are directed. Instead, of a vector field, we employ a tensor field for the flow field [Chen 08; Zhang 07], which allows a user to interactively edit the flow field. In our method, we employ a 2×2 symmetric tensor, t , given by Eq. 1. Here $\mu \geq 0$, $\theta \in [0, 2\pi)$, and t have two orthogonal eigenvectors.

$$t = \mu \begin{pmatrix} \cos 2\theta & \sin 2\theta \\ \sin 2\theta & -\cos 2\theta \end{pmatrix} \tag{1}$$

We employ two types of tensor fields to generate a pebble mosaic: a *grid pattern* for regular flow and a *radial pattern* for revolving flow. Fig. 2 shows examples of the tensor fields and the generated pebble patterns represented by elliptical disks. The long axis of an elliptical disk is placed so as to follow the major eigenvector of its underlying tensor field.



(a) Grid pattern (b) Grid result (c) Radial pattern (d) Radial result

Figure 2: *Tensor fields and generated pebble patterns*

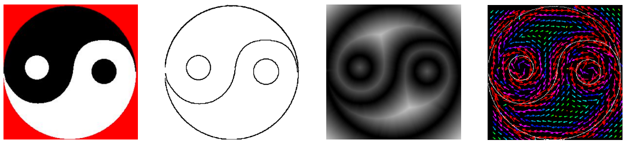
3.2 Initial Flow Field

For a given input image I (Fig. 3(a)), the procedure computes its gradient ΔI (Fig. 3(c)) from the edge curves (Fig. 3(b)) and normal vectors $s \cdot \nabla p$ at each pixel p from ΔI, and then rotates them counterclockwise by $\pi/2$ to obtain the tangent vectors. Here s is scaling factor $s = e^{-dD(p)}$, where $D(p)$ is the distance value from its nearest edge point and d is the scalar scaling parameter. From

*e-mail: naoki.nth@gmail.com
†e-mail:miyatak@acm.org

SIGGRAPH Asia 2011, Hong Kong, China, December 12 – 15, 2011.
ISBN 978-1-4503-0807-6/11/0012

the tangent vector, the tensor value at pixel p is calculated using the *grid pattern* described in Section 3.1.



(a) Input image (b) Edge curve (c) Distance map (d) Initial field

Figure 3: *Initial flow field generation*

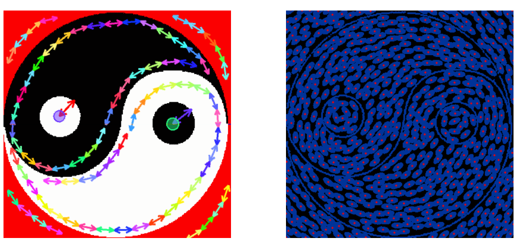
3.3 Editing Flow Field

A user can edit the initial flow field by drawing arbitral curves on it. The drawn curves are divided into uniform segments, and tensor value $T_i(p)$ at pixel p is calculated for each i-th segment s_i. Final value $T(p)$ is calculated by summing up $T_i(p)$ weighted with the distance between s_i and p, as given by Eq. 2.

$$T(p) = \sum_i e^{-d\|p-s_i\|^2} T_i(p) \qquad (2)$$

3.4 Cell Distribution

Each cell of a pebble is placed along the generated stream field. To distribute the cells, our method employs the Poisson disk distribution using an elliptical disk instead of a circular one. The method also distributes the cells so as to avoid the edge curves. Fig. 4(b) shows an example of the distributed cells.



(a) Stream curves (b) Distributed cells

Figure 4: *Poisson disk distribution*

3.5 Pebble Mosaic Generation

A pebble mosaic is obtained by generating a pebble for each cell distributed by the aforementioned method. Each pebble is generated in two steps:

1. Generation of an initial mesh for each cell
2. Smoothing of the initial mesh by the subdivision surface method

An initial mesh is generated according to the geometry of each cell. A 3D prism shape is obtained by sweeping the extent rectangle of each cell to a given height in the cell's normal vector direction. The initial mesh is then generated by deforming this prism by displacing each top corner at random and tapering the prism. After the initial meshes have been generated, they are subdivided and refined by using the Loop's subdivision method.

4 Results

Fig. 1(d) and Fig. 5(a) are generated from 2D images. It can be seen that the pebbles are aligned to the specified stream curves. Fig. 5(b) and Fig. 6(b) are generated from 3D models. Here the initial flow field is calculated from the normal map and the depth map of the 3D model, as shown in 6(a). The pebbles are scaled in proportion to the depth value of the model. In addition, we can obtain an alternative decoration pattern, whose background is generated with rocks from Voronoi diagrams instead of pebbles, as shown in Fig. 5(b).

(a) Yin-yang symbol (b) Bunny

Figure 5: *Examples*

(a) Data (b) Result

Figure 6: *Example: Horse model*

5 Conclusion

We proposed an interactive method for generating pebble mosaics that enables the generation of an intended pattern freely by drawing stream curves. In the future, we would like to solve the problem of noticeable gaps among pebbles by applying a relaxation process.

References

CHEN, G., ESCH, G., WONKA, P., MÜLLER, P., AND ZHANG, E. 2008. Interactive procedural street modeling. In *Proc. of SIGGRAPH 2008*, 103:1–10.

HAUSNER, A. 2001. Simulating decorative mosaics. In *Proc. of SIGGRAPH 2001*, 573–580.

MIYATA, K., ITOH, T., AND SHIMADA, K. 2001. A method for generating pavement textures using the square packing technique. *The Visual Computer*, 17(8): 475–490.

PEYTAVIE, A., GALIN, E., GROSJEAN, J., AND S. MÈRILLOU 2009. Procedural generation of rock piles using aperiodic tiling. *Computer Graphics Forum*, 28(7): 1801–1809.

ZHANG, E., HAYS, J., AND TURK., G 2007 Interactive tensor field design and visualization on surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 13: 94–107.