Unsupervised three-dimensional reconstruction of small rocks from a single two-dimensional image

M. Gilardi[1,2], P. L. Watten[2] and P. Newbury[2]

[1] Doctoral School, University of Sussex, UK
[2] Dept. Engineering and Informatics, University of Sussex, UK

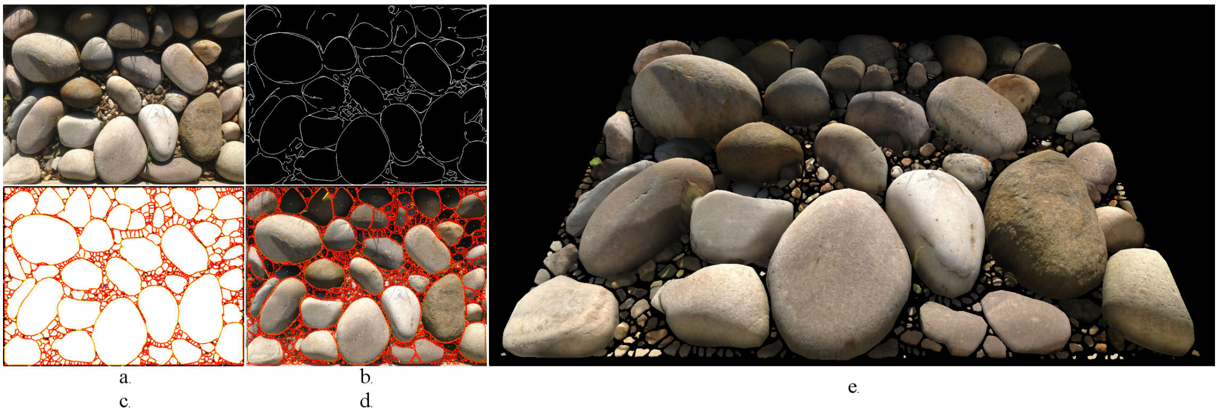

Figure 1: *a) Original image; b) Small rocks outer edges; c) Equatorial slices; d) Equatorial slices superimposed to the original image; e) Three-dimensional reconstruction*

Abstract

Surfaces covered with pebbles and small rocks can often be found in nature or in human-shaped environments. Generating an accurate three-dimensional model of these kind of surfaces from a reference image can be challenging, especially if one wants to be able to animate each pebble individually. To undertake this kind of task manually is time consuming and impossible to achieve in dynamic terrains animations.
The method described in this paper allows unsupervised automatic generation of three-dimensional textured rocks from a two-dimensional image aiming to closely match the original image as much as possible.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

1. Introduction

Procedural generation of rocks has been addressed in the literature by Peytavie, Galin et al. [PGGM09a, PGGM09b] which describe a method to aperiodically tile surfaces with rocks; on the same line of work Sakurai and Miyata [SM10] use Voronoi cells to generate and pile different kinds of rocks, while Dart, de Rossi and Togelius [DDT11] use three-dimensional L-systems to procedurally generate rocks.

Although these approaches are very effective in producing random or regular distributions of rocks, none of them use an image to guide the generation of the scene.
In contrast, Kita and Miyata [KM11] use a reference image, but only to describe the general shape of the mosaic in their pebbles mosaic generator and not to describe each rock shape and texture. Liu and Xing [LX13] model a rock starting from an image of its material interfaces. However,

DOI: 10.2312/egsh.20141007

they focus on reproducing a single rock with an high level of detail.

In this article, the focus is on reproducing a large number of small rocks that have appearance, position and shape defined by a given two-dimensional image.

Reconstructing a three-dimensional model of an object from a single two-dimensional image is challenging due to the lack of information along one of the spatial axis. In literature the problem has been often studied for architectural purposes [DT96, vdH98, SM99, JTC09], or allowing a user to guide the process [LH00, CZS*13].
The reader may refer to Bolle and Vemuri [BV91] for a survey of mathematical methods for three-dimensional surface reconstruction and to Remondino and El-Hakim [REH06] for a more recent review on techniques and problems on image based three-dimensional modeling.

In this paper, *generalized ellipsoids* are introduced as a way to generate organic small rock meshes, section 2. Moreover, a way to identify outer edges of small rocks reducing texture noise, and a method to describe the equatorial slices of a small rock given a binary image using the signed turning angle of a polygonal curve are introduced, section 3. Finally results are presented in section 4 and possible future work discussed in section 5.

2. Assumptions and definitions

Pebbles in nature usually have a smooth surface that could be easily approximated by an ellipsoid. However, not all small rocks have such property.

In the following it is assumed that the small rocks to model could be slightly concave, that they have genus zero, that the equatorial slice of a small rock is a star shaped set with respect to its center, and that the reference image is obtained so that the view direction of the camera is orthogonal to the plane on which the small rocks lies. This choice is made so that perspective can be ignored during the reconstruction process. It is also assumed that the image is packed with small rocks and no other surfaces are visible.

2.1. Generalized Ellipsoids

Following the idea of generalized cylinders, [BK85], let's define a *parametrized generalized ellipsoid*, $GE : S_p \to \mathbb{R}^3$, $S_p = [0,2\pi) \times [-\frac{\pi}{2}, \frac{\pi}{2}]$, as an extension of an ellipsoid, allowing its semi-principal axis to change for each point of the parameter space S_p accordingly with some positive smooth functions, $f(\varphi,\vartheta)$, $g(\varphi,\vartheta)$, $h(\varphi,\vartheta)$ from S_p to $\mathbb{R}^+ \backslash \{0\}$.
A parametrization for GE is given by:

$$GE(\varphi,\vartheta) = \begin{cases} x(\varphi,\vartheta) &= f(\varphi,\vartheta)\cos(\varphi)\cos(\vartheta) \\ y(\varphi,\vartheta) &= h(\varphi,\vartheta)\sin(\vartheta) \\ z(\varphi,\vartheta) &= g(\varphi,\vartheta)\sin(\varphi)\cos(\vartheta) \end{cases} \quad (1)$$

It is clear that choosing the functions $f(\varphi,\vartheta)$, $g(\varphi,\vartheta)$, $h(\varphi,\vartheta)$ as constants parametrization (1) yields a parametrization of an ellipsoid, hence the name generalized ellipsoid. In this paper, generalized ellipsoids will be preferred over generalized cylinders as the former do not require the computation of a sweep curve.

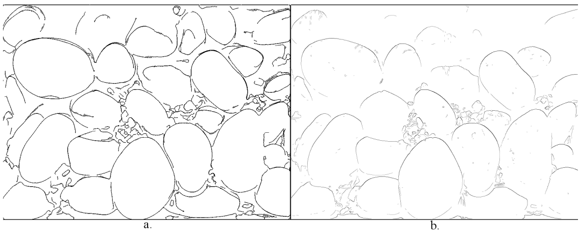

Figure 2: *(a) Edges obtained by the method presented in this paper applied to figure 1a compared with (b) edges obtained using Canny applied to the same image. Notice that using Canny some texture changes have been caught.*

3. Method

Figure 1 illustrates the steps the method uses for the reconstruction: a binary mask with the edges of each small rock is computed, each equatorial slice description in polar coordinates is then obtained from it, and finally the mesh of each small rock is computed using a generalized ellipsoid.

3.1. Mask generation

Let's represent the intensity of a $w \times h$ pixels image I as a function $f : \mathbb{R}^2 \to [0,1]$.
Edges in an image can be defined as sharp changes in the intensity function. The standard way to determine this changes is through derivation of the intensity function f or of the colour channels of the image, [Can86, AMFM11]. However, these methods detect changes in f that do not necessarily correspond to an object contour, as in figure 2b.
In order to identify and suppress all the edges which are not an object contour a median filter of radius r is applied to f, and then the notion of change in f is considered saying that f changes whenever one of the following happens:

$$\begin{array}{l} f(x,y) < \tau f(x+\varepsilon,y) \\ f(x,y) < \tau f(x-\varepsilon,y) \\ f(x,y) < \tau f(x,y+\varepsilon) \\ f(x,y) < \tau f(x,y-\varepsilon) \end{array} \quad (2)$$

where $\tau \in [0,1]$, determines the change strength, and ε is a small positive value, set to 1 in the experiments.
To identify contours of the equatorial slices the interval $[0,1]$ is subdivided in a set of L levels L_i, $i \in [0,L]$. For each level L_i the changes in the intensity f are computed as in formulation (2) . An image I_L is then initialized setting all pixels to black. If at position (x,y) a change in f for level L_i is detected a value of $\frac{1}{L}$ is summed to $I_L(x,y)$. In the final image I_L pixels with sharp changes in f are brighter than

other pixels. After a hysteresis thresholding process using two thresholds t_H and t_L, connectivity analysis for detecting and linking edges is applied. Figure 2 shows the results of this algorithm compared to Canny edge detection. Although the method proposed is less efficient than Canny, it produces contours with less texture noise.

3.2. Equatorial slices description

Form the binary mask the polar representation of each equatorial slice from its center is computed as follows.

The mask is sampled superimposing a grid of $m \times n$ points on it. Given a point not on an edge n rays are shot from it in n directions, rays-edges intersections are computed and used to build a closed polygonal curve P.
As edges in the mask are not complete those vertices of P which are not on the outer edge of the equatorial slice have to be removed. To do so a vertex is flagged as *critical* and removed from P whenever

$$\begin{array}{c} \cos(\alpha_i) < a\alpha_T \\ \sin(\alpha_i) < b\alpha_T \end{array} \tag{3}$$

Where α_i is the signed turning angle [GS08], $a, b \in \mathbb{R}$ and $\alpha_T \in (0,1)$ is a threshold value. Check (3) is repeated until no critial vertices remain in P, afterwards the center C of P is computed and P itself is recomputed resampling the mask from it. These steps are repeated until the center stabilizes. Gaps in-between vertices, generated by the removal of critical vertices, are filled using a Bezièr curve. The resulting polygonal is sampled from its center in 360 directions. Points thus obtained are then used to compute the radii r_i, $i \in \{0, 1, \ldots, 359\}$, of P, which, together with C, are stored to be used later for mesh reconstruction. Finally, the set of pixels enclosed by the resulting polygon P is flood filled and the initial samples enclosed in P are removed, figure 1c. The algorithm is repeated for the remaining initial samples.

3.3. Mesh generation

Let's consider parametrization (1) given in subsection 2.1, and let's require that $f(\varphi,\vartheta) = g(\varphi,\vartheta)$, and that f and g are constants with respect to ϑ

$$r(\varphi) = f(\varphi,\vartheta) = g(\varphi,\vartheta)$$

This is motivated by the result obtained in subsection 3.2 being r_i a discretization of $r(\varphi)$. Furthermore, let's choose the function $h(\varphi,\vartheta)$ to be constant

$$h = \min_{\varphi} r(\varphi)$$

as experiments have shown that this choice gives reasonable shapes for small rocks and in particular for pebbles. Under the choices made, vertices for a small rock can be computed using the parametrization:

$$X(\varphi,\vartheta) = \begin{cases} x(\varphi,\vartheta) & = & r(\varphi)\cos(\varphi)\cos(\vartheta) + C_x \\ y(\varphi,\vartheta) & = & h\sin(\vartheta) \\ z(\varphi,\vartheta) & = & r(\varphi)\sin(\varphi)\cos(\vartheta) + C_y \end{cases} \tag{4}$$

Where $C = \begin{pmatrix} C_x & C_y \end{pmatrix}$ is the center of the equatorial slice computed in section 3.2. Once each vertex has been computed the mesh is scaled so it matches the size in world coordinates of the plane to which the image is applied as texture. From (4) normals can be obtained normalizing:

$$N(\varphi,\vartheta) = -\begin{pmatrix} h(x(\varphi,\vartheta) - \frac{r_\varphi}{r} z(\varphi,\vartheta)) \\ \frac{r^2}{h} y(\varphi,\vartheta) \\ h(z(\varphi,\vartheta) + \frac{r_\varphi}{r} x(\varphi,\vartheta)) \end{pmatrix} \tag{5}$$

where the notation has been simplified writing $r(\varphi)$ as r and $\partial_\varphi r(\varphi)$ as r_φ. Texture coordinates are obtained using a planar projection for each hemisphere.

4. Results

Figures 1 and 3 shows that the approach described in this paper is capable of producing realistic small rocks models and distributions that match a reference image. These results have been generated using a 64 bit DELL Precision T1650, with a 4 cores Intel(R) Xeon(R) CPU at 3.4 GHz, 8 GB RAM, and an NVIDIA Quadro 2000 graphics card.

The average time for reconstructing a single image is 105.421 seconds, split as follows: 14.37% of the total time is spent for the edges extraction phase, 85.08% for the equatorial slices reconstruction phase and, 0.56% for the mesh generation phase. The bottleneck is clearly the equatorial slices reconstruction phase which takes the majority of the time.

Parameters described in section 3 have been estimated as follows. All test images have been rescaled to have width equal to 900 pixels and height determined by the image aspect ratio. For mask generation, $L = 256$, t_H has been set to the first quartile for the histogram of I_L and then scaled to stay in $[0,1]$, while, following Canny [Can86], $t_L = \frac{t_H}{3}$. For equatorial slices descriptions the initial sampling grid was set to $\frac{w}{5} \times \frac{h}{5}$, and the number of initial sampling points to 90. Experiments on a set of 80 images shown that $a = 0.65$ and $b = 0.5$ give acceptable results for most of them, and that in general $a \in [0.5, 0.75]$ is a good choice. The angle threshold α_T has been computed iteratively reducing its value from 1 toward zero by 0.01 steps, stopping whenever the minimum number of vertices of the initial polygonal curve P, which was set to 10, is reached. For the images in figure 3 the number of vertices interpolated by the Bezièr curve to close gaps has been set to 5.

The method proposed to describe equatorial slices fails to reproduce areas where edges meet with acute angles, resulting in an over-segmentation of the equatorial slice of the small rock near those areas, "crumbling" the mesh as shown by the pink rock near the lower right corner of in figure 1e.

5. Conclusions and Future Work

In this article a method for unsupervised generation and distribution of large numbers of small rock models using a single image has been described.

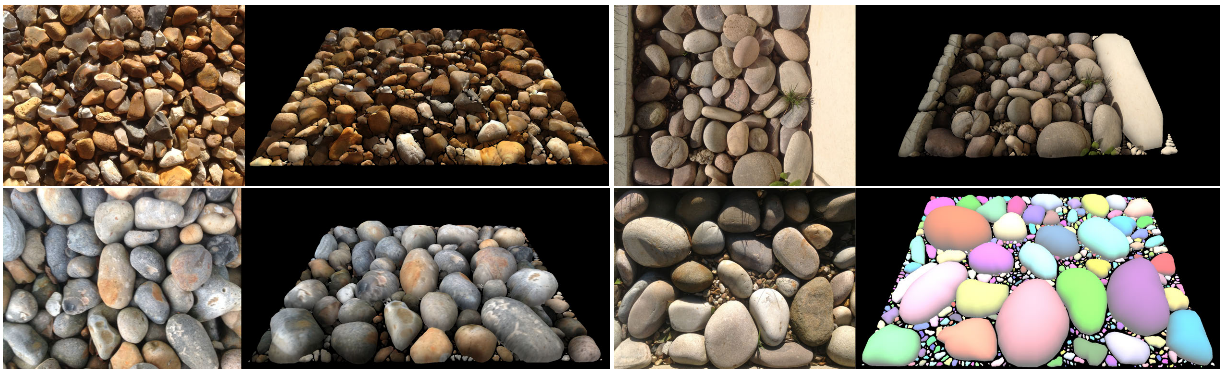

Figure 3: *Results obtained applying the method described in this article to different images of heterogeneous rocks (in sizes and textures) under different lighting conditions, last image is rendered without texturing to show the reconstructed shapes.*

An application of this method can be in the creation of dynamic terrains: changing a local section of the image based texture into small rock meshes, animating them, and changing them back to a texture image afterwards

Although the method is capable to produce realistic models and distributions of small rocks, it relies on a series of parameters that have been estimated heuristically. A possible approach to automatically estimate those parameters could involve machine learning techniques. However, it would be nice to have a mathematical relationship between them which allows automatic estimation. Such relationship will be the object of further studies. Finally speed and efficiency of the method can be improved, ideally parallelizing it for GPU implementation.

References

[AMFM11] Arbeláez P., Maire M., Fowlkes C., Malik J.: Contour Detection and Hierarchical Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 33*, 5 (2011), 898–916. 2

[BK85] Bronsvoort W. F., Klok F.: Ray Tracing Generalized Cylinders. *ACM Transactions on Graphics 4*, 4 (Oct. 1985), 291–303. doi:10.1145/6116.6118. 2

[BV91] Bolle R. M., Vemuri B. C.: On Three-Dimensional Surface Reconstruction Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence 13*, 1 (1991). 2

[Can86] Canny J.: A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-8*, 6 (1986), 679–698. 2, 3

[CZS*13] Chen T., Zhu Z., Shamir A., Hu S.-M., Cohen-Or D.: 3-Sweep : Extracting Editable Objects from a Single Photo. In *Proceedings of ACM SIGGRAPH Asia 2013* (2013). 2

[DDT11] Dart I. M., De Rossi G., Togelius J.: SpeedRock : procedural rocks through grammars and evolution. In *Proceedings of the 2nd International Workshop on Procedural Content Generation in Games* (2011), pp. 7–10. 1

[DT96] Debevec P. E., Taylor C. J.: Modeling and Rendering Architecture from Photographs : A hybrid geometry- and image-based approach. In *Proceeding SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (1996), pp. 11–20. 2

[GS08] Grinspun E., Secord A.: Introduction to Discrete Differential Geometry : The Geometry of Plane Curves. In *Proceeding SIGGRAPH Asia '08 ACM SIGGRAPH ASIA 2008 courses Article No. 5* (2008), pp. 1–4. 3

[JTC09] Jiang N., Tan P., Cheong L.-F.: Symmetric Architecture Modeling with a Single Image. *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH Asia 2009 28*, 5 (2009), 1–8. doi:10.1145/1618452.1618459. 2

[KM11] Kita N., Miyata K.: Interactive Procedural Modeling of Pebble Mosaics. In *SA '11 SIGGRAPH Asia 2011 Sketches* (2011), pp. 12–13. 1

[LH00] Liu S., Huang Z.: Interactive 3D Modeling Using Only One Image. In *VRST '00 Proceedings of the ACM symposium on Virtual reality software and technology* (2000), pp. 49–54. 2

[LX13] Liu Y., Xing H. L.: Surface mesh generation of large-scale digital rock images in 3D. *Procedia Computer Science 18* (2013), 1208–1216. doi:10.1016/j.procs.2013.05.287. 1

[PGGM09a] Peytavie A., Galin E., Grosjean J., Merillou S.: Arches : a Framework for Modeling Complex Terrains. *Computer Graphics Forum 28*, 2 (2009), 457–467. 1

[PGGM09b] Peytavie A., Galin E., Grosjean J., Merillou S.: Procedural Generation of Rock Piles using Aperiodic Tiling. *Pacific Graphics 28*, 7 (2009). 1

[REH06] Remondino F., El-Hakim S.: Image-Based 3D Modelling : A Review. *The Photogrammetric Record 21*, 115 (2006), 269–291. 2

[SM99] Sturm P. F., Maybank S. J.: A Method for Interactive 3D Reconstruction of Piecewise Planar Objects from Single Images. In *The 10th British Machine Vision Conference (BMVC '99)* (1999), vol. 89221, pp. 265–274. 2

[SM10] Sakurai K., Miyata K.: Procedural Modeling of Multiple Rocks Piled on Flat Ground. In *SIGGRAPH Asia 2010 Posters* (2010), pp. 1–2. 1

[vdH98] van den Heuvel F. A.: 3D reconstruction from a single image using geometric constraints. *ISPRS Journal of Photogrammetry and Remote Sensing 53*, 6 (Dec. 1998), 354–368. doi:10.1016/S0924-2716(98)00019-7. 2