

[Department of Computer Science](#)
[Drexel University College of Engineering](#)

The following item is made available as a courtesy to scholars by the author(s) and Drexel University Library and may contain materials and content, including computer code and tags, artwork, text, graphics, images, and illustrations (Material) which may be protected by copyright law. Unless otherwise noted, the Material is made available for non profit and educational purposes, such as research, teaching and private study. For these limited purposes, you may reproduce (print, download or make copies) the Material without prior permission. All copies must include any copyright notice originally included with the Material. **You must seek permission from the authors or copyright owners for all uses that are not allowed by fair use and other provisions of the U.S. Copyright Law.** The responsibility for making an independent legal assessment and securing any necessary permission rests with persons desiring to reproduce or use the Material.

Please direct questions to archives@drexel.edu

Drexel University Libraries
www.library.drexel.edu

University Archives and Special Collections:
<http://www.library.drexel.edu/archives/>



<http://www.drexel.edu/>

Stochastic Microgeometry for Displacement Mapping

Craig A. Schroeder, David E. Breen, Christopher D. Cera, and William C. Regli*

Drexel University, Philadelphia, PA 19104

Abstract

Creating surfaces with intricate small-scale features (microgeometry) and detail is an important task in geometric modeling and computer graphics. We present a model processing method capable of producing a wide variety of complex surface features based on displacement mapping and stochastic geometry. The latter is a branch of mathematics that analyzes and characterizes the statistical properties of spatial structures. The technique has been incorporated into an interactive modeling environment that supports the design of stochastic microgeometries. Additionally a tool has been developed that provides random exploration of the technique's entire parameter space by generating sample microgeometry over a broad range of values. We demonstrate the effectiveness of our technique by creating diverse, complex surface structures for a variety of geometric models, e.g. arrowheads, candy bars, busts, planets and coral.

1. Introduction

Generating complex surface textures with powerful and efficient techniques is an important problem in computer graphics. To solve this problem procedural modeling techniques, as well as procedural texture mapping methods, have been developed to increase the visual complexity of geometric models and computer graphics scenes. Ideally these techniques produce geometric models with intricate and realistic surface features with a concise representation and a minimum of user input, while also providing a user with flexible tools for adjusting low-level parameters that may be used to fine-tune a desired result. We describe a geometry processing method based on techniques from the field of stochastic geometry that addresses these issues and offers an approach to creating a wide variety of complex, realistically-appearing geometric models. The method produces detailed small-scale features on triangle meshes by reconstructing stochastic feature distributions on the meshes in order to define offset values for displacement maps. Given stochastic functions and statistical information that characterize and describe complex 2D patterns, the offset values are assigned to the individual vertices of the mesh, producing stochastic microgeometry on the model's surface.

The technique is useful for producing many different kinds of irregular surface detail with a single mathematically- and physically-sound computational approach; thus providing a powerful model processing tool for transforming smooth, "ideal" geometric models into convincing representations of natural or hand-made objects. See Figure 1. For example, smooth objects can be made to appear to be formed from clay, or chiseled from stone. Since the method is based on reconstruction of a statistical distribution, it may be used to create large numbers of different, but statistically and visually similar models. This allows for the automated construction of whole families of complex objects that share a common visual characteristic, but individually are quite different. This may be useful when populating a scene with numerous objects that are of the same type, while each object is still unique; for example when creating models of a gravel walk, a cinder-block wall, a coral reef, a tray of pre-historic stone implements, or a candy counter.

The statistical properties of the microgeometry can be produced from several sources. For instance they may be derived from scanning real materials [19, 34]. Alternatively the distributions and parameters that characterize these patterns may be computationally generated or interactively defined. We have created three tools that allow a user to design stochastic microgeometries. Similar to Sim's image generation technique [36] and the Design Galleries concept [22], we are able to randomly sample the parameter space that defines the resulting surface structures. For each random point in the parameter space a specific reconstruction is produced for a given model. If the exact desired displacement map is not produced and further fine-tuning is required, an interactive environment is available for adjusting the model and algorithm parameters. Finally, feature distributions may be produced by analyzing images of textured surfaces.

We demonstrate the effectiveness of our technique by creating diverse, complex surface structures for a variety of geometric models, e.g. arrowheads, candy bars, busts, planets and coral. Additionally, descriptions of our tools with representative results are presented. Once a desired displacement map is defined, it is used to randomly process a single model to produce numerous instantiations of the stochastic distribution. The stochastic displacement map may then be applied to any triangle mesh.

* e-mail: {cas43,david,cera,regli}@cs.drexel.edu

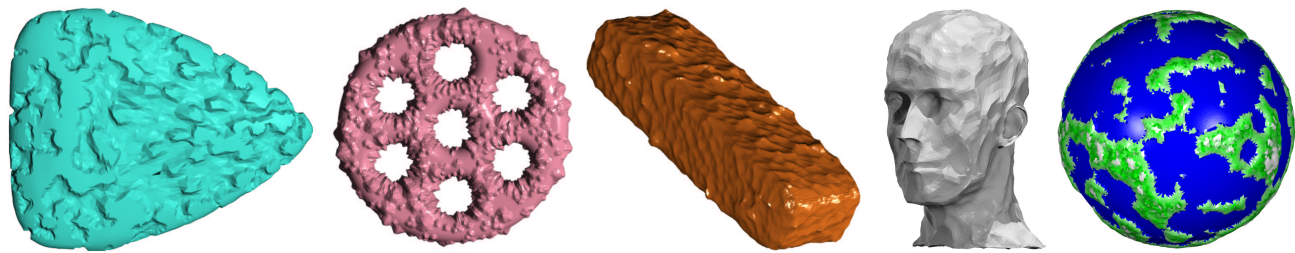


Figure 1. Turquoise arrowhead, coral disk, chocolate candy bar, Giacometti-style bust and a virtual planet. Displacement maps based on stochastic microgeometry provide the surface features.

2. Related Work

Procedural texture mapping is a common technique for computationally adding complexity to simple geometric models [13]. Most of these techniques combine noise functions and/or simulations to produce varied and realistic texture maps for surfaces [6, 14, 16, 38, 42, 43] and volumes [28, 29]. Others take similar approaches, such as the statistical learning technique of [3]. Other less general approaches have been proposed for specific modeling tasks, e.g. modeling stone facades [24], stone solid textures [19] and flow patterns [11]. Texture values may also be utilized to define offsets from the surface to produce displacement maps [8, 40]. Additionally, computational techniques have been used to tile complex models with image fragments [26, 39, 41, 45] and geometric surface textures [5].

Another related field of study is stochastic procedural modeling [12, 15, 21, 25, 27, 30]. Here stochastic techniques are used to place particle systems, subdivide surfaces and define 3D density functions. Our work differs from all the previous examples in that it brings a new mathematical modeling construct (stochastic geometry) to computer graphics and geometry processing. Reconstruction and texture mapping algorithms based on stochastic geometry provide a general and powerful approach for generating varied and intricate 2D (the work described here) and 3D [34] structures with well-defined and consistent statistical properties; thus providing a superior method for adding naturally-appearing details to smooth “ideal” geometric models.

Stochastic techniques are being widely investigated for other purposes, such as improving rendering performance [20] and reproducing characteristics of one-dimensional curves [17].

3. Stochastic Geometry

*Stochastic geometry*¹ is the study of the random processes that produce geometric structures and spatial patterns. It focuses on analyzing and understanding the con-

nections between geometry and probability in order to describe and characterize geometric small-scale features and large-scale spatial events [4, 37]. It has been used to analyze the porosity of materials like sandstone [18, 44], the spatial arrangement of particle strikes on a detector in order to detect bias or skew [37], the microstructure of biological materials [33, 34], fracture patterns in rock [2], vegetation distribution [10], human settlement patterns [7], and even bomb coverage patterns [31, 32]. For all these applications, stochastic geometry provides a rich and concise representation for complex structures and a mathematical framework for computing and analyzing aggregate statistical geometric properties. Stochastic geometry is supported by a number of key concepts.

3.1. The Stochastic Point Process

Definition A *point process* is the family of all sequences of points satisfying two special conditions: (1) the sequence is locally finite; Each bounded subset must contain a finite number of points, and (2) the sequence is simple, i.e. it contains no duplicates. These sequences can in general be considered as a random set, when order is not important. The term point process is introduced to aid in defining a contact distribution.

Stationarity and Isotropy A stationary point process is one whose properties are invariant under translation. An isotropic point process is one whose properties are invariant under rotation. A motion invariant point process is one that is both stationary and isotropic.

3.2. Contact Distributions

The *contact distribution function* $H_B(r)$ is defined as the probability that rB does not intersect Φ , where r is a non-negative scalar and B is a convex compact set. H_B may be expressed as

$$H_B(r) = 1 - P(\#\Phi(rB) = 0), \quad (1)$$

where $\#\Phi(\Lambda)$ is the number of items from point process Φ that are included in set Λ , and $P(Q)$ is the probability of event Q . If we let B be the unit disk, we are left with the radial contact distribution (or hitting distribution). The distribution $H_B(r)$ gives the probability that a disk B of radius r

¹ Stochastic geometry is a branch of mathematics. When using the term *stochastic geometry* in this paper we are referring to this distinct technical field rather than the heuristic perturbation techniques generally used in computer graphics.

can be randomly placed inside the object without intersecting (contacting) the object.

In general, there are many choices for B . If B were a cube centered at the origin, the distribution would describe the likelihood that a non-intersecting cube of a given size would fit at a point (without changing its orientation). B can be generalized to star-shaped sets in place of the convex requirement.

Radial Contact Distribution The radial contact distribution attempts to characterize the microstructure of an object by analyzing the distances associated with points outside of an object’s boundary (porous region). Additional useful information is obtained by supplementing this with a second distribution that considers points inside of an object’s boundary (solid region) in the same manner. In our 2D work the boundary defines an interface between two regions on the surface of the processed meshes, with each region containing a different displacement sub-map.

We model microstructures on the mesh with two distributions; one specifies the likelihood that points in the negative (porous) region are a certain distance from the positive (solid) region, and the other specifies the likelihood that points in the positive region are a certain distance from the negative region. The measured distances are not Euclidean; they are geodesic and are calculated over the surface of the mesh. See Section 4.2 for more detail. Our variation of the radial contact distribution is illustrated in Figure 6.

4. Stochastic Reconstruction

Stochastic displacement mapping begins with the definition of the two distributions for the positive and negative regions and the triangle mesh to be processed. The general strategy involves assigning a distance value to each triangle in the mesh, where the value corresponds to the minimum distance to the positive/negative boundary and is consistent with a given radial contact distribution. The total number of triangles in the mesh and the distributions are used to create a histogram, where each bin represents the number of triangles that should be assigned a distance value in a particular interval. As we assign distance values to triangles compatible with the constraints of the measure, we decrement the appropriate bin in the histogram. Once this process is completed, offset values for the vertices are computed by averaging the distance values of the surrounding triangles. The average value is then used to displace the vertex in the direction of the local surface normal.

4.1. Stochastic Function

The microgeometry stochastic function is a variation of the radial contact distribution for complex surfaces. The function captures information regarding the distance from random points on a surface to the nearest region boundary. In the context of a porous object, the distribution gives the probability that a random point is a distance R from the pore/material interface. This information is stored in the form of two distributions as shown in Figure 8. In our prior

work we used this information to describe the geometric properties of a porous 3D object [33, 34]. In the work described here, the distinction between pore and material is not germane. We simply define two regions with different displacement maps on a triangle mesh. As such, the use of pore and material to describe the two types of regions is abandoned. The two regions are identified by the sign of their associated distances.

4.2. Geodesic Distance

We measure distance over the surface of a triangle mesh with an approximation of the geodesic distance. Consider the dual of the mesh, where nodes are located at the centroids of triangles. Given two triangles, we compute the distance between them as the shortest path between the corresponding nodes along the dual graph. The measure is computed incrementally and is thus efficiently obtained [9].

4.3. Preparing the Distributions

The two contact distributions may be derived from real materials, randomly generated, or interactively defined by a user. The two initial distributions, which are at first defined only for non-negative distance values, are combined into a single distribution over all real numbers. One distribution remains defined for positive real numbers, and the other distribution is reflected into the negative real numbers. The frequency value for each interval (bin) in the distribution is calculated by multiplying the total number of triangles by the average probability, given by the distribution, over the interval. The transformation of two distributions into a single histogram is outlined in Figure 9. Creating a single histogram simplifies the reconstruction algorithm and its implementation, by allowing two microstructure regions to be defined with a single set of frequency values.

4.4. Processing a Triangle Mesh

Stochastic geometry reconstruction is the process that takes a contact distribution and generates a specific model with structures statistically consistent with the distribution. For stochastic displacement mapping this process involves removing distance values from the bins of the histogram and assigning them to triangles in a mesh based on a geodesic contact distribution.

Since our approach creates surface detail with microgeometry, it is sometimes necessary to first remesh and/or subdivide an input model [1] before the reconstruction step in order to produce a surface with minute, high-resolution facets. Because the reconstruction technique works at the triangle level, it is necessary to split triangles until the mesh is fine enough to obtain the desired level of detail and avoid aliasing. It is important to choose a triangulation method that produces reasonable aspect ratio triangles, since high aspect ratios could lead to undesirable artifacts. Although a

uniformly high-resolution mesh is best suited to this technique, it may be desirable to adaptively reduce the resolution of the final processed mesh based on the characteristics of the microgeometry [35].

Reconstruction We begin by identifying the bins in the histogram that have the highest absolute value. Processing begins from the outside (highest absolute value) of the histogram and terminates after the zero bin has been considered. Each iteration of the algorithm assigns values from a bin until it is empty or cannot be processed further due to the constraints described below. From our experience processing from high to low absolute values creates a more structured, efficient and less random result. The choice of direction is not arbitrary, but rather is related to the bin packing nature of the problem [23].

The assignment of these values has a geometric interpretation, and with this interpretation comes a constraint. If a triangle has been assigned the value x (which implies that the triangle is a distance x from the region boundary), another triangle located at a distance m away must be assigned a value in the range $[x - m, x + m]$. Here, distances are computed between the centroids of triangles along the surface of the mesh. For example, if a triangle has been assigned the value 5.0, a triangle that is 0.4 units away, must be assigned a value in the range [4.6, 5.4]. This constraint is global and applies to any pair of triangles. However, the constraint has little effect on triangles that are widely separated.

Each iteration of the algorithm consists of two steps. First, triangles that are constrained to have a value that falls within the interval of the current bin are identified. Values are calculated for these triangles based on the distance to the nearest assigned triangle. A range of acceptable values is maintained for each unassigned triangle.

For example, if the range of a triangle is [2.3, 5.7] and bins 3-5 are empty, the triangle must be assigned a label from bin 2. It will receive the label 2.3, the most extreme value it may be given. Note that the range of any particular triangle gives the range of labels that may be assigned to that triangle without violating any distance constraint.

When a label is assigned, the constraint is propagated outwards and the ranges are adjusted if necessary. This would normally require that every triangle be checked to see if it must have its range updated. Because the measure is geodesic, it can be computed incrementally in increasing order. In this way, the propagation can be terminated when the distance is large enough that it can no longer have an impact. More specifically, one can stop when the values assigned plus or minus the current distance both lie outside of the range of any triangle.

The next step distributes the remaining labels from the current bin randomly in a way that does not violate any constraints. If bin 4 is being processed, the value 4.0 (though any value between 4.0 and 5.0 would work) is assigned to random triangles if 4.0 is in the acceptable range for that triangle and the triangle isn't already labeled. If no triangles meet these criteria, the bin is considered finished and the next bin is processed. The algorithm terminates when

all bins have been processed. The reconstruction algorithm is illustrated in Figure 5. Note that it is this seeding process that ultimately determines the location and distribution of most of the larger positive and negative regions.

Dealing With Error Our reconstruction algorithm is effectively a form of bin packing [23], which cannot in general be accomplished in polynomial time without some error. In most cases the algorithm is unable to assign all values in the histogram to triangles. This leaves a certain number of triangles without an offset value (typically about 5% to 10%). These triangles are then assigned a value which is an average of neighboring values and is consistent with the distance constraints, regardless of the values remaining in the histogram. This heuristic guarantees that an output can be generated for any input, and that the output will not violate the distance constraint. However, this will degrade the statistical properties of the resulting structure.

4.5. Displacement Mapping

Once the triangles of the mesh are labeled with signed distance values, values for vertices are calculated by averaging the values of adjacent triangles. The user is then offered a variety of options and parameters for mapping vertex values into offset values, which may be independently chosen for each region. These choices give the user freedom to finely adjust the look and feel of the processed model and provide a powerful approach to generating a wide variety of surface detail.

First, the direction of the offset is specified. The vertices may be offset outward, inward or left untouched. The parameter s in $\{-1, 1, 0\}$ specifies the sign of the offset value, with 1 signifying an outward offset, -1 inward, and 0 no change. Second, a scale factor k that determines the magnitude of the offset is available. Finally, we have two functions $f_-(x)$ and $f_+(x)$ that map the value x into a displacement amount $f_-(|x|)$ for vertices in the negative region or $f_+(|x|)$ for vertices in the positive region. A vertex is in the positive region if the value for that vertex is positive, and similarly for negative. The final offset distance is $skf_-(|x|)$ or $skf_+(|x|)$. We initially utilize the functions $\{x, \sqrt{x}\}$ as choices for $f_-(x)$ and $f_+(x)$ to generate our examples, though other functions could be used. (The latter was chosen for its smoothing effect; it causes ridges and crevices to be more rounded). Once the parameters/options are specified, the model's vertices are offset in the direction of the local surface normal \vec{n} . An additional parameter u defines the ratio of the units of the distribution with the units of the geometric model, and controls the spatial frequency of the surface structures.

5. User Tools

A geometry processing technique that does not offer a user effective handles for controlling the resulting output has limited usefulness and value. We therefore developed three strategies for designing displacement maps based on

stochastic microgeometry. The first strategy employs random sampling of the parameter space of the contact distributions and the algorithm input values. Similar to Sim’s image generation technique [36] and the Design Galleries concept [22], a user is able to quickly view the results produced by a large number of parameter combinations. Once an interesting set of approximate parameter values are found the second strategy allows the user to interactively modify those parameters to create the final desired effect. Finally, contact distributions may be computationally generated by analyzing images of textured surfaces.

5.1. Parameter Space Exploration

The first tool is implemented with a set of scripts that can be used to explore the parameter space. These scripts allow the user to specify a subset of the entire parameter space and the number of test results to be generated for a given triangle mesh. The scripts randomly sample the selected subset and generate a large number of processed models. The models are rendered and presented on a web page for rapid viewing, along with the parameters associated with each result. Given a number of potential candidate models, a tighter search can be employed on a smaller range of the parameter space. At any time the parameters for a specific candidate model may be fed to an interactive tool for further fine-tuning. Figures 1 and 2 present several processed meshes that were randomly generated with our parameter space exploration tool.

5.2. Interactive Design

The second tool is an interactive application for fine-tuning a model’s distribution and the algorithm’s parameters in order to obtain a desired final processed mesh. The interactive program visually divides the design process into three parts. In the first part the contact distribution is designed, modified, and fine-tuned. This involves setting the number of elements in the individual bins of the histogram described in Section 4.4. Distributions may be normalized and saved for further explorations at a later time. The second part gives the user access to algorithm parameters. These parameters include mesh resolution, distribution frequency (coarse effects versus fine ones), and offsetting parameters, e.g. scale factors and displacement functions. The third part is the actual reconstruction, displacement and rendering. This stage occurs quickly, requiring only a few seconds to apply and display a new stochastic microgeometry displacement map; thus allowing the user to rapidly converge on a desired result. Figure 3 (Top) presents several screen-shot of our interactive tool while it is being used to design a specific stochastic displacement map.

5.3. Image-Based Acquisition

The radial contact distribution can be computed from a 2D image of a textured surface. Via a thresholding process the input image is first converted into a binary image of

Example	Scale Units	Displacement, k		Direction		$f(x)$	
		Neg	Pos	Neg	Pos	Neg	Pos
Fig 1, Arrow	63.27	.02876	.03489	-	in	-	\sqrt{x}
Fig 1, Disk	64.69	.01092	.05097	out	-	x	-
Fig 1, Candy	38.93	.01860	.03970	in	-	\sqrt{x}	-
Fig 1, Head	27.78	.02033	.01647	in	in	\sqrt{x}	x
Fig 1, World	36.65	.05472	.02389	-	out	-	\sqrt{x}
Fig 6, All	79.82	.04906	.03232	-	in	-	\sqrt{x}
Fig 8, Candy	71.48	.02111	.02856	out	out	\sqrt{x}	\sqrt{x}
Fig 8, Coral	57.27	.03061	.03400	in	out	\sqrt{x}	\sqrt{x}
Fig 8, Head	81.97	.01083	.01319	out	out	\sqrt{x}	x
Fig 10, All	40.00	.05000	.03000	-	in	-	x

Table 1. Parameters used to generate example models.

black and white regions. The white regions can be labeled as positive and the black as negative. The distribution generation process begins by zeroing a set of counters for each integer distance value of the histogram. A radial search is then performed at each pixel in the image until a pixel with the opposite color is found. The distance between these two pixels is computed. The unsigned distance value is negated if the original pixel is black. The counter corresponding to the distance value is incremented. Once this process is completed for all pixels, the values in every counter are divided by the total number of pixels in the image to produce the histogram for the radial contact distribution associated with the texture.

6. Implementation

Let n be the number of triangles in the mesh after subdivision has been performed. All of the initial meshing operations are $O(n)$. The reconstruction considers in the worst case every point in the mesh for every distribution value, yielding a worst case cost of $O(n)$. (The number of values in the distribution is assumed to be constant.) Because a propagation of constraint information must be performed when random placements are made, reconstruction has (worst case) time complexity of $O(n^2)$. In practice, the behavior is much faster than quadratic (after the first few passes, the propagation affects only the local portion of the mesh) and behaves linearly. The final displacement mapping and other mesh operations are all linear in the number of triangles.

Our modeling tools were implemented in C++, perl and bash. For meshes with tens of thousands of triangles, the reconstruction code completes in a few seconds on a 900MHz SPARC Solaris workstation. The parameter space exploration tools are a collection of several programs and scripts. Typically, generation of each candidate model and associated image requires a few seconds on the same 900MHz SPARC Solaris workstation.

7. Results

We have processed numerous geometric models with our stochastic displacement mapping technique. For exploration purposes, 500 candidate models were generated by

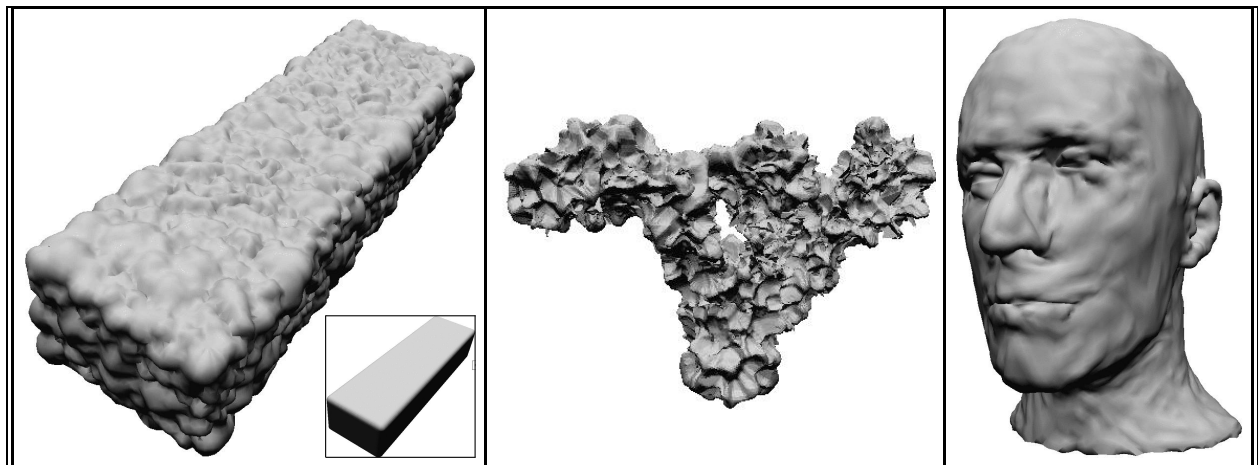
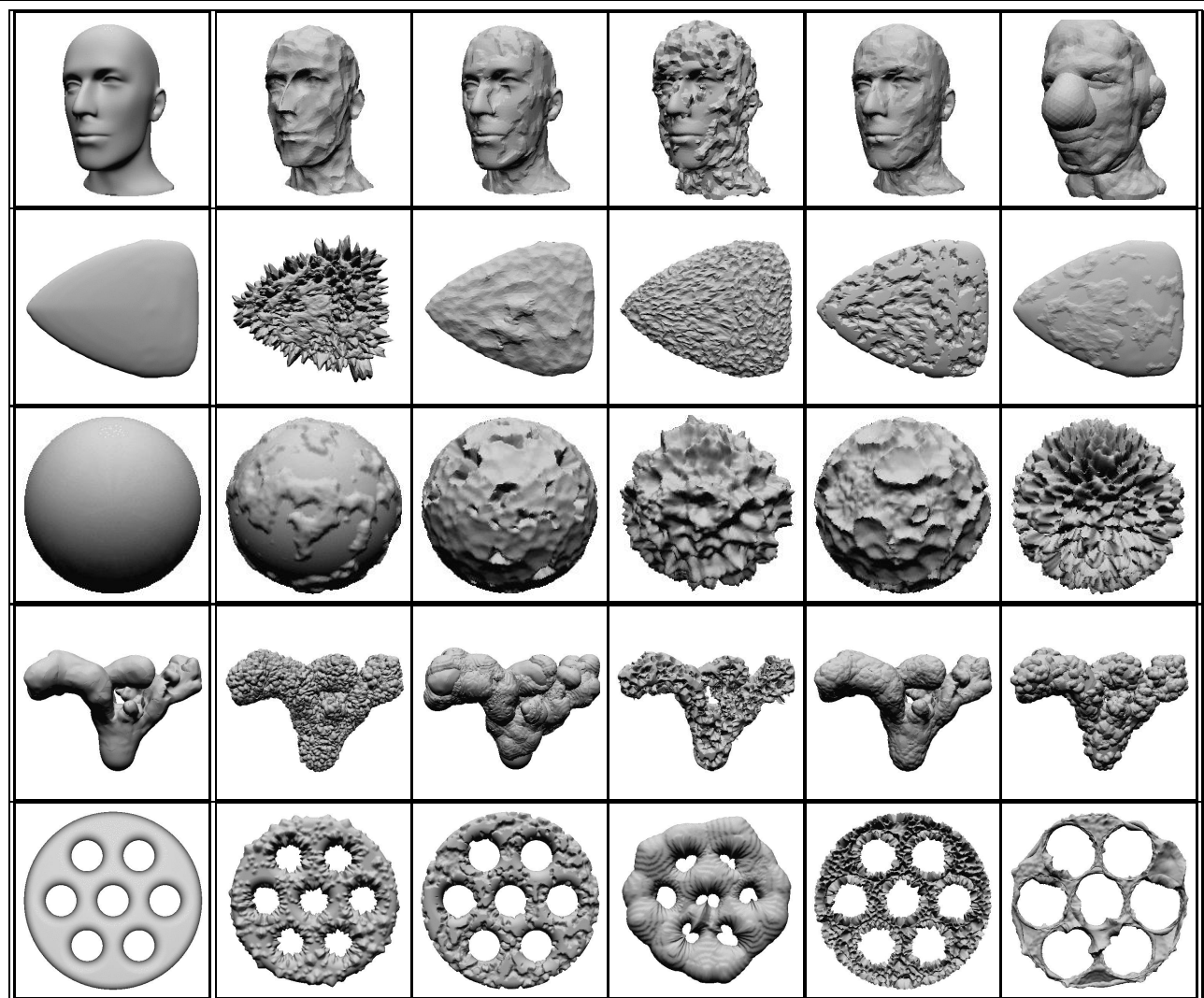


Figure 2. (Top) Five models are modified with displacement maps based on stochastic microgeometry, demonstrating a wide diversity of surface characteristics. The first image is the original mesh that is processed to produce the remaining models. (Bottom) Close-up views of a rice-covered candy bar, coral and clay head.

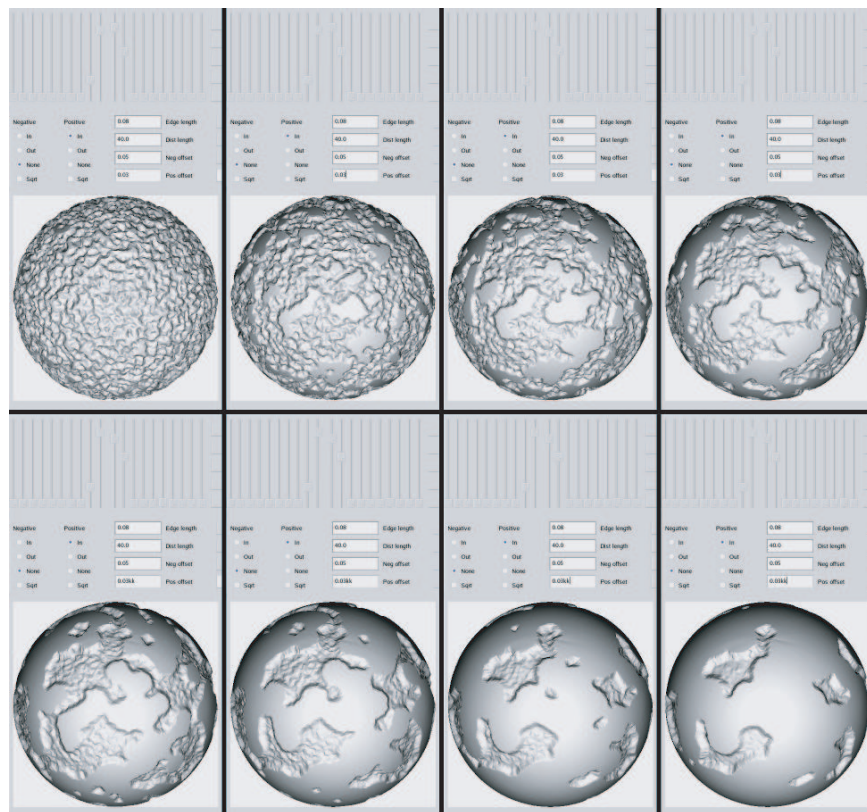


Figure 3. (Top) Designing a stochastic microgeometry displacement map. (Bottom) Twenty different textured meshes using the distribution from the bottom left distribution from the (Top) examples.

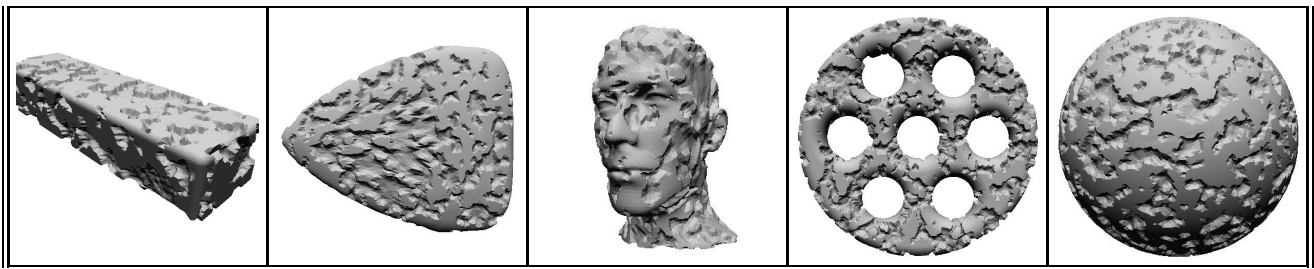


Figure 4. Processing different triangle meshes with the same stochastic microgeometry displacement map.

randomly sampling the parameter space of our algorithm (frequency values of the histogram, s , k , $f(|x|)$ and u) for each input mesh. Selected images for six of the meshes are shown in Figures 1 and 2. These images demonstrate the diversity of possible microgeometries produced by our technique. The processing loop consists of random parameter generation, reconstruction, mesh visualization, and other background processing; the entire process takes on the order of a few seconds per model. The models have on the order of tens of thousands of triangles, with the largest ones having over a hundred thousand triangles.

The parameter values and contact distributions used to generate most of these results can be found in Table 1 and Figure 7. The parameter “Scale Units” sets the scale for the model. The model is contained within a box with edge size $2(\text{Scale Units})$. Once scaled, the distances in the object correspond to distances in the distribution. Displacement parameter k is the scale factor that controls the magnitude of a vertex offset in a particular region. The function $f(x)$ influences the shape of the resulting displacements. In particular, the magnitude of the displacement is $kf(|L|)$, where L is the label assigned to that vertex. “Direction” indicates whether vertices should be displaced inward or outward. Note that vertices with positive and negative labels have their own Displacement, Direction, and Function. A dash indicates that vertices with positive or negative labels are not displaced.

While the technique may not fully reconstruct the geometric properties of the input distribution on the surface of the model, this does not affect its ability to create realistic microgeometry. Further, the technique exhibits the desirable properties that would be obtained from a faithful reconstruction, such as the consistency of statistical properties across varied geometry.

The interactive environment has been employed to design numerous stochastic microgeometry displacement maps. One design session is presented in Figure 3 (Top). These are several screen shots that demonstrate how the displacement map changes as a user modifies the individual contact distribution values. Once a desired displacement map is defined, it may be repeatedly and stochastically applied to a single mesh in order to create numerous models that have similar overall appearance, but are all individually different, as seen in Figure 3 (Bottom). The same

stochastic displacement map may also be applied to different models, as seen in Figure 4.

Despite its many benefits and useful characteristics, our modeling technique does have limitations. The contact distribution function is isotropic, which limits the type of structure it can represent. Thus, this technique cannot currently be used to recreate anisotropic textures and microstructures, but can be extended with other distribution functions to do so. Further, designing with distributions is not particularly intuitive, requiring some practice and learning in order to generate specific results.

8. Conclusion

Combining stochastic geometry with displacement mapping creates a technique for producing complex surface microgeometry that is both visually appealing and realistic. It provides a powerful and effective means for generating a diverse set of surface models by stochastically defining offset values on triangular meshes in statistically-consistent patterns. The approach illustrates the ability of stochastic geometry to supplement existing texture mapping techniques for computer graphics modeling applications.

9. Acknowledgments

The coral model was provided by Dr. Stephen Jeffrey of the University of Queensland, Australia. The mannequin head model was provided by Caltech’s Multi-Res Modeling Group. This work was supported in part by National Science Foundation Grants ITR/DMI-0219176 and CCF-0310619. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the other supporting government and corporate organizations.

References

- [1] P. Alliez, M. Meyer, and M. Desbrun. Interactive geometry remeshing. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 21(3):347–354, 2002.

- [2] G. Baecher. Statistical analysis of rock mass fracturing. *International Journal of the Association of Mathematical Geologists*, 15(2):333–352, 1984.
- [3] Z. Bar-Joseph, R. El-Yaniv, D. Lischinski, and M. Werman. Texture mixing and texture movie synthesis using statistical learning. *IEEE Trans. Visualization and Computer Graphics*, 7(2):120–35, 2001.
- [4] O. Barndorff-Nielsen, W. Kendall, and M. van Lieshout. *Stochastic Geometry: Likelihood and Computation*. Chapman & Hall / CRC, 1999.
- [5] P. Bhat, S. Ingram, and G. Turk. Geometric texture synthesis by example. In *Proc. Eurographics Symposium on Geometry Processing*, pages 43–46, July 2004.
- [6] J. S. D. Bonet. Multiresolution sampling procedure for analysis and synthesis of texture images. In *Computer Graphics*, pages 361–68. ACM SIGGRAPH, 1997.
- [7] A. Bradley and C. Small. Looking for circular structure in post hole distributions: quantitative analysis of two settlements from Bronze Age England. *Journal of Archaeological Science*, pages 285–297, 1985.
- [8] R. L. Cook. Shade trees. In *Proc. SIGGRAPH*, pages 223–231, 1984.
- [9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and Clifford. *Introduction to Algorithms, Second Edition*. MIT Press, 2001.
- [10] P. Diggle. Binary mosaics and the spatial pattern of heather. *Biometrics*, 37:531–539, 1981.
- [11] J. Dorsey, A. Edelman, H. W. Jensen, J. Legakis, and H. K. Pedersen. Modeling and rendering of weathered stone. In *Proc. SIGGRAPH*, pages 225–234, 1999.
- [12] D. Ebert, W. Carlson, and R. Parent. Solid spaces and inverse particle systems for controlling the animation of gases and fluids. *The Visual Computer*, 10(4):179–190, 1994.
- [13] D. S. Ebert, F. K. Musgrave, D. Peachey, K. Perlin, and S. Worley. *Texturing and modeling: a procedural approach*. Morgan Kaufmann, Amsterdam, 2003.
- [14] K. W. Fleischer, D. H. Laidlaw, B. L. Currin, and A. H. Barr. Cellular texture generation. In *Proc. SIGGRAPH*, pages 239–248, 1995.
- [15] A. Fournier, D. Fussell, and L. Carpenter. Computer rendering of stochastic models. *Comm. of the ACM*, 25(6):371–384, 1982.
- [16] D. J. Heeger and J. R. Bergen. Pyramid-based texture analysis/synthesis. In *Proc. SIGGRAPH*, pages 229–238, 1995.
- [17] A. Hertzmann, N. Oliver, B. Curless, and S. M. Seitz. Curve analogies. In *Proc. 13th Eurographics Workshop on Rendering*, pages 233–45, June 2002.
- [18] R. Hilfer and C. Manwart. Permeability and conductivity for reconstruction models of porous media. *Physical Review E*, 64, JUL 2001.
- [19] R. Jagnow, J. Dorsey, and H. Rushmeier. Stereological techniques for solid textures. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 23(3):329–335, 2004.
- [20] A. Kalaiah and A. Varshney. Statistical point geometry. In *Eurographics Symposium on Geometry Processing*, pages 107–15, June 2003.
- [21] J. Lewis. Generalized stochastic subdivision. *ACM Trans. on Graphics*, 6(3):167–190, 1987.
- [22] J. Marks et al. Design galleries: a general approach to setting parameters for computer graphics and animation. In *Proc. SIGGRAPH*, pages 389–400, 1997.
- [23] S. Martello and P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, 1990.
- [24] K. Miyata. A method of generating stone wall patterns. In *Proc. SIGGRAPH*, pages 387–394, 1990.
- [25] F. K. Musgrave, C. E. Kolb, and R. S. Mace. The synthesis and rendering of eroded fractal terrains. In *Proc. SIGGRAPH*, pages 41–50, 1989.
- [26] F. Neyret and M.-P. Cani. Pattern-based texturing revisited. In *Proc. SIGGRAPH*, pages 235–242, 1999.
- [27] A. Norton. Generation and display of geometric fractals in 3-D. In *Proc. SIGGRAPH*, pages 61–67, 1982.
- [28] K. Perlin. An image synthesizer. In *Proc. SIGGRAPH*, pages 287–296, 1985.
- [29] K. Perlin and E. M. Hoffert. Hypertexture. In *Proc. SIGGRAPH*, pages 253–262, 1989.
- [30] W. T. Reeves. Particle systems - a technique for modeling a class of fuzzy objects. In *Proc. SIGGRAPH*, pages 359–376, 1983.
- [31] H. Robbins. On the measure of a random set I. *Annals of Mathematical Statistics*, 15:70–74, 1944.
- [32] H. Robbins. On the measure of a random set II. *Annals of Mathematical Statistics*, 16:342–347, 1945.
- [33] C. Schroeder, W. C. Regli, A. Shokoufandeh, and W. Sun. Representation of porous artifacts for bio-medical applications. In *8th ACM Symposium on Solid Modeling and Applications, 2003*, pages 254–57, June 2003.
- [34] C. Schroeder, W. C. Regli, A. Shokoufandeh, and W. Sun. Computer-aided design of porous artifacts. *Computer-Aided Design*, 37(3):339–353, March 2005.
- [35] W. Schroeder. Decimation of triangle meshes. In *Proc. SIGGRAPH*, pages 65–70, 1992.
- [36] K. Sims. Artificial evolution for computer graphics. In *Proc. SIGGRAPH*, pages 319–328, 1991.
- [37] D. Stoyan, W. Kendall, and J. Mecke. *Stochastic Geometry and Its Applications*. John Wiley & Sons, 1987.
- [38] G. Turk. Generating textures on arbitrary surfaces using reaction-diffusion. In *Proc. SIGGRAPH*, pages 289–298, 1991.
- [39] G. Turk. Texture synthesis on surfaces. In *Proc. SIGGRAPH*, pages 347–354, 2001.
- [40] L. Wang, X. Wang, X. Tong, S. Lin, S. Hu, B. Guo, and H.-Y. Shum. View-dependent displacement mapping. *ACM Trans. Graph.*, 22(3):334–339, 2003.
- [41] L.-Y. Wei and M. Levoy. Texture synthesis over arbitrary manifold surfaces. In *Proc. SIGGRAPH*, pages 355–360, 2001.
- [42] A. Witkin and M. Kass. Reaction-diffusion textures. In *Proc. SIGGRAPH*, pages 299–308, 1991.
- [43] S. Worley. A cellular texture basis function. In *Proc. SIGGRAPH*, pages 291–294, 1996.
- [44] C. L. Y. Yeong and S. Torquato. Reconstructing random media. II. Three-dimensional media from two-dimensional cuts. *Physical Review E*, 58(1):224–33, OCT 1997.
- [45] J. Zhang, K. Zhou, L. Velho, B. Guo, and H.-Y. Shum. Synthesis of progressively-variant textures on arbitrary surfaces. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 22(3):295–302, 2003.

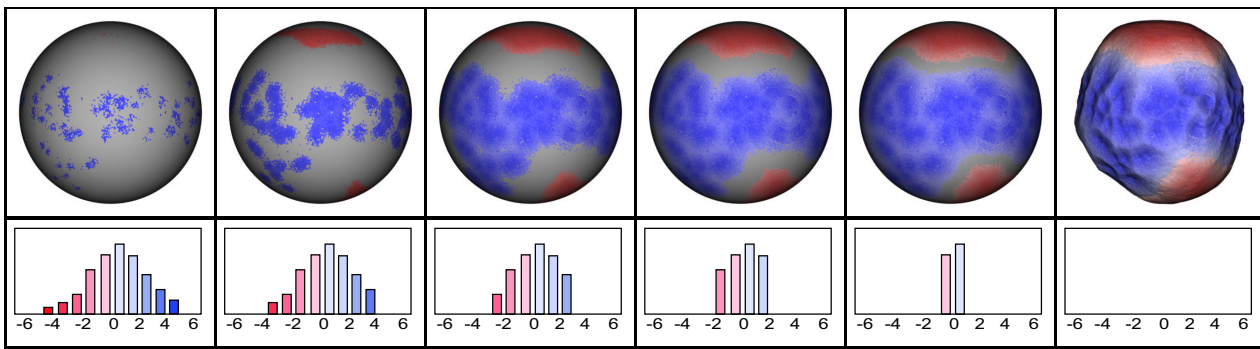


Figure 5. Labels are assigned beginning with high absolute values (red and blue), progressing towards zero (white). Labeling occurs in two ways: random placement (most evident in the first illustration), and mandatory placement (causing the spreading effect seen in the remaining stages).

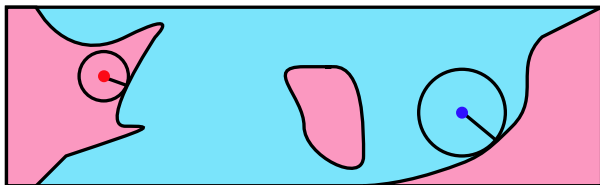


Figure 6. The radial contact distribution is based on the probability that a random point is a particular distance to a positive region (blue) and the negative region (red). Since we are taking measurements on complex surfaces, we utilize geodesic distances during this calculation.

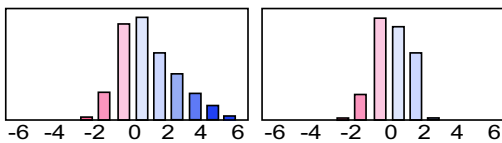


Figure 7. Distributions used to generate example models. The models in Figures 1, 2 and 4 were created with the left distribution. The models in Figure 3 (Bottom) were created with the right distribution, which was defined with the interactive tool presented in Figure 3 (Top).

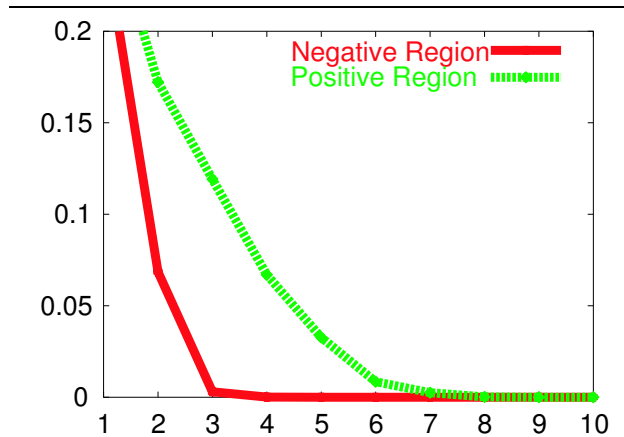


Figure 8. Distributions obtained from a porous bone sample. The horizontal axis is the distance R to the pore/material interface. The vertical axis is the probability that a point is distance R from the interface.

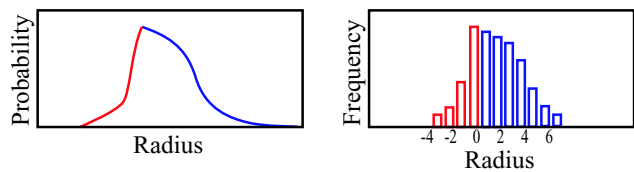


Figure 9. The distributions are originally defined separately. The distributions are merged into a single distribution that extends in both positive and negative directions. Using the mesh's triangle count, the distribution is converted into the histogram that is used by the reconstruction algorithm.