

# A Survey of Blue-Noise Sampling and Its Applications

Dong-Ming Yan(严冬明)<sup>1,2</sup>, Jianwei Guo(郭建伟)<sup>2</sup>, Bin Wang(王斌)<sup>3</sup>, Xiaopeng Zhang(张晓鹏)<sup>2</sup>, Peter Wonka<sup>1,4</sup>

1. *Visual Computing Center, King Abdullah University of Science and Technology*

2. *National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences*

3. *School of Software, Tsinghua University*

4. *Department of Computer Science and Engineering, Arizona State University*

**Abstract** In this paper, we survey recent approaches to blue-noise sampling and discuss their beneficial applications. We discuss the sampling algorithms that use points as sampling primitives and classify the sampling algorithms based on various aspects, e.g., the sampling domain and the type of algorithm. We demonstrate several well-known applications that can be improved by recent blue-noise sampling techniques, as well as some new applications such as dynamic sampling and blue-noise remeshing.

**Keywords** Blue-Noise Sampling, Poisson-disk Sampling, Lloyd Relaxation, Rendering, Remeshing

## 1 Introduction

Sampling is an essential technique in computer science. Sampling translates a continuous signal into its discrete counterpart or selects a subset from a discrete set of signals, such that the signal can be represented and processed by computers efficiently. For example, one-dimensional (1D) sound waves, two-dimensional (2D) images,

and three-dimensional (3D) polygonal meshes are captured by discrete sampling from continuous signals.

In computer graphics, sampling plays an important role in many applications, such as rendering [1], stippling [2], texture synthesis [3], object distribution [4], and simulation [5]. Among all the sampling techniques, *blue-noise sampling* is the most popular method in recent papers. The term “blue-noise” refers to any

noise with minimal low-frequency components and no concentrated spikes in energy. Intuitively, blue-noise sampling generates randomized uniform distributions. Fig. 1 shows a typical point set with blue-noise properties generated by Poisson-disk sampling.

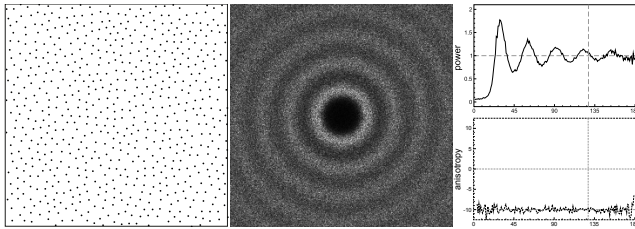


Fig. 1. An example of Poisson-disk sampling and its spectral analysis. Left: a sampled point set; middle: the power spectrum from this point set; and right: radial means and normal anisotropy.

In this paper, we first discuss the color of noise in Section 2, and then summarize recent techniques for blue-noise sampling in Section 3. We focus on recent approaches in our survey as a complement to the comprehensive survey by Lagae and Dutré [6]. Furthermore, we discuss several existing and new applications that benefit from the blue-noise sampling techniques in Section 4.

## 2 The Color of Noise

We can classify point distributions by looking at the Fourier spectrum. Different spectra are associated with different colors .

For example, white noise refers to noise with a flat spectrum, which contains an equal amount of energy in all frequency bands. It is usually used in random number generators [7]. Blue noise refers to point distributions with weak low-frequency energy, but strong high-frequency energy. Pink noise is the complement of blue-noise and its spectral energy is concentrated in the low-frequency bands. Pink noise occurs very frequently in nature and thus is used for physical simulation and biological distributions [8, 9]. Green noise is an uncommon term that can refer to the mid-frequencies of white noise. It characterizes the distributions of a variety of natural phenomena and has been used for digital halftoning [10].

There are also many other colors used for noise, with or without precise definitions. In this paper, we focus only on techniques that generate blue-noise sampling patterns.

## 3 Blue-Noise Sampling

There are various ways to characterize existing blue-noise sampling techniques. For example, the sampling algorithms can be classified by the type of sampling domain (2D, 3D, or surfaces), the metric used in the domain (geodesic or Euclidean), the shape of the sampling primitives (point, line, ball, etc.), the

properties of the sampling results (isotropic or anisotropic), the style of the algorithm (dart throwing, relaxation, or tiling) and so on.

### 3.1 Sampling Domain

Here, we briefly describe the common input domains.

**Euclidean domain.** Most previous blue-noise sampling algorithms were first developed to handle the 2D or 3D Euclidean space. The traditional domain is the unit torus (i.e., the unit square and cube in 2D and 3D, respectively, with periodic boundary conditions) [11], in which the distance between two points is measured using the Euclidean metric. Some work also addresses more complicated domains, such as non-convex polygons with holes [12, 13].

**High dimensions.** Since high-dimensional point distributions have special applications, some approaches are able to generalize blue-noise sampling to high dimensions [14, 15, 16, 13]. However, many high-dimensional sampling methods do not scale well with high dimensions because they typically suffer from the curse-of-dimensionality, which means that the effectiveness deteriorates very rapidly as the dimensions increase. Ideas to overcome this problem have been suggested by Ebeida et al. [17].

**3D Surface.** Blue-noise sampling has also been extended to mesh surfaces. In this case,

the input is usually a two-manifold triangular mesh surface, that consists of a set of triangles. The sampled points should be located exactly on the surface.

### 3.2 Sampling Algorithm

Blue-noise sampling methods can be roughly classified into three types according to the techniques they use: (1) Poisson-disk sampling and its variations; (2) relaxation-based sampling; and (3) patch/tile-based sampling. In this section, we give an overview of these methods and discuss several classic algorithms in detail. Each type of algorithm is further classified by the sampling domain. We emphasize methods that use points as primitives for isotropic blue-noise sampling in lower dimensions.

#### 3.2.1 Poisson-disk Sampling

Poisson-disk sampling is a classic technique that generates uniformly random distributed point sets. An ideal Poisson-disk sampled point set,  $\mathbf{X} = \{(\mathbf{x}_i, r_i)\}_{i=1}^n$ , in sampling domain  $\Omega$  should satisfy the following three properties: 1) the *minimal distance property*, which requires that the distance between any two disk centers should be larger than the sampling radius, i.e.,  $\forall \mathbf{x}_i, \mathbf{x}_j \in P, \|\mathbf{x}_i, \mathbf{x}_j\| \geq \min(r_i, r_j)$ ; 2) the *unbiased sampling property*,

which requires that each point in the domain has a probability that is proportional to the sizing at this point to receive a sampling point; and 3) the *maximal sampling property*, which requires that the union of the disks covers the entire sampling domain, i.e.,  $\bigcup(\mathbf{x}_i, r_i) \supseteq \Omega$ . The sampling is uniform if the sampling radius,  $r_i$ , is constant. Otherwise, it becomes adaptive sampling.

**Euclidean domain.** The traditional method for Poisson-disk sampling is called dart-throwing and was first proposed by Cook [18]. Given a sampling domain and a sampling radius, the algorithm generates disks in the sampling domain randomly. If the current generated disk conflicts with any previous sampled disk, then it is rejected; otherwise, it is accepted. This process is repeated until a continuous number of rejections is observed. The algorithm complexity of the original dart-throwing algorithm is  $O(n^2)$ . However, this approach is inefficient to achieve the maximal property. Therefore, a lot of work has been expended on generalizing and accelerating this algorithm.

Most recent work aiming for efficient Poisson-disk sampling maintains a data structure to track and sample the empty regions (also called gap primitives). Dunbar and Humphreys [19] described an efficient implementation of the dart-throwing algorithm for

maximal Poisson-disk sampling (Fig. 2(a)). They applied a data structure called scalloped sectors to record the active front of the sampled disk set. Their algorithm runs in  $O(n \log(n))$  time, but the sampling is biased.

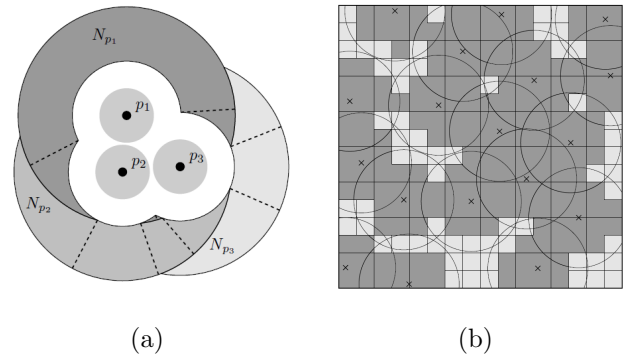


Fig. 2. Data structures used for accelerating Poisson-disk sampling. (a) Scalloped sectors [19]. (b) Quad-tree [13].

The simplest data structure for uniform Poisson-disk sampling is the quad-tree. White et al. [20] first proposed to use such a data structure for acceleration. The cell size of the base grid equals  $\frac{r}{\sqrt{2}}$ , such that each grid cell can at most receive one disk with radius  $r$ . During the sampling process, the partially covered cells are subdivided into smaller fragments in a quad-tree manner. Gamito and Maddock [16] extended White et al.'s algorithm to higher dimensions. Later, Jones and Karger [21] reported that they reduced the time complexity of Poisson-disk sampling to linear time. The follow-up work of Ebeida et al. [13] further accelerated the sampling process by sampling a



flat fragment array instead of using the hierarchical quad-tree, as shown in Fig. 2(b). The grid-based sampling algorithm has been implemented on recent *Graphics Processing Units* (GPUs) [15, 22, 23]. But the GPU extensions cannot guarantee the unbiased sampling property.

Jones [24] first proposed an algorithm for unbiased maximal Poisson-disk sampling. He used a Voronoi diagram to extract the uncovered regions, called gaps in the sampling domain. These uncovered regions were further re-sampled to achieve the maximal sampling property in an unbiased manner. The core idea is that maximal sampling can be obtained if and only if the Voronoi cell of each vertex is fully covered by the disk centered at the vertex. In the sampling process, Jones repeatedly generated disks and inserted them in a global Voronoi diagram one by one, until the sampling became maximal. Each vertex in the Voronoi diagram recorded a value that indicated the area of the empty region of the corresponding Voronoi cell. A new sample was generated by first selecting a Voronoi cell based on the empty area. Once a new sample was generated, the Voronoi diagram and the value of each vertex were updated.

Ebeida *et al.* [12] proposed a hybrid approach that first used squares and later con-

vex polygons bounding the intersections of a square and multiple circles as gap primitives. A two-step unbiased maximal sampling framework was developed. They first performed classic dart-throwing on a uniform grid and then switched to computing and filling empty regions by clipping the grid cells against the neighboring disks.

However, most algorithms mentioned above can only handle uniform sampling. Yan and Wonka [25] presented an algorithm for the generation of maximal Poisson-disk sets with varying radii. Built on the regular triangulation and the power diagram, they conducted a theoretical analysis of gaps in such disk sets. Then they designed efficient algorithms and data structures for gap detection and gap updates when the disks were changed. Their method works well in both Euclidean space and on manifolds. Fig. 3 compares the gap extraction algorithms of [24], [12], and [25].

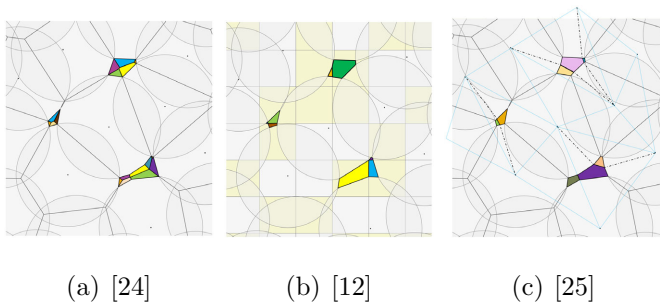


Fig. 3. Comparison of three representative algorithms for gap computation. (a) Voronoi diagram [24]. (b) Uniform grid [12]. (c) Regular triangulation and the power diagram [25].

Mitchell et al. [26] studied 2D Poisson-disk sampling with various radii. They analyzed the conflicting condition under which a maximal sampling can be achieved. [27] introduces a sifted disk technique for locally resampling a point cloud to reduce the number of points. The essence of this algorithm is still maximal Poisson-disk sampling. More recently, Yuksel [28] proposed a point set resampling approach for fast Poisson-disk sampling; however, this approach is biased and not maximal.

**Surface sampling.** The classic dart-throwing algorithm has been extended to mesh surfaces [29, 30, 31, 32, 33, 25, 34, 35, 36, 28] as well as to isosurfaces [37]. Euclidean distance is used in these approaches. Besides the Euclidean metric, the geodesic metric is also used for Poisson-disk sampling [38, 39, 40, 41]. Surface sampling techniques can be also used for surface remeshing by computing the restricted Delaunay tri-

*J. Comput. Sci. & Technol., Mon.. Year, ,* angulation [42]. We address this issue as an application of blue-noise sampling in Section 4.4.

### 3.2.2 Relaxation-based Sampling

Iterative relaxation is another important technique for generating point distributions. This type of method usually consists of two steps: (1) generating an initial point set, and (2) optimizing the point positions using Lloyd iterations [43] until convergence. In this type of approach, the points  $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$  to be optimized are called *sites*.

The methods are differentiated by different objective functions. For example, the original Lloyd algorithm minimizes the quantization error in signal processing. It is known as the *Centroidal Voronoi Tessellation* (CVT) in computer graphics [44]. The energy function of CVT can be formulated as

$$E_{CVT}(\mathbf{X}) = \sum_{i=1}^n \int_{V_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 dx, \quad (1)$$

where  $\{V_i\}_{i=1}^n$  is the Voronoi diagram of the points in the sampling domain  $\Omega$  and  $\rho(\mathbf{x})$  is a density function defined over  $\Omega$ .

The key ingredient of computing a CVT is to construct the Voronoi diagram. The Voronoi diagram in Euclidean spaces is well studied. It has also been generalized on manifold surfaces. There are multiple ways to compute CVT on mesh surfaces, e.g., parameterization-based ap-

proaches [45, 46], discrete clustering [47], and the exact computation of the Voronoi diagram on surfaces [42, 34]. We compare these different approaches in Section 4.4.

However, CVT tends to generate a point distribution with regular patterns (i.e., hexagonal arrangements) that lacks some blue-noise properties. Several methods have been proposed to modify CVT to obtain better blue-noise properties.

Balzer et al. [48] introduced a *capacity constrained Voronoi tessellation* (CCVT) to generate point sets with excellent blue-noise properties. In this method, Voronoi cells should satisfy the constraints  $\|V_i\| = \int_{V_i} \rho(\mathbf{x})dx = c_i$ , where  $c_i$  are capacity constraints with  $c_i > 0$ . Intuitively, the capacity of a site can be understood as the area of its corresponding Voronoi region weighted by the density function. The energy function of CCVT is given as follows:

$$E_{CCVT}(\mathbf{X}) = \sum_{i=1}^n \left\| \int_{V_i} \rho(\mathbf{x})dx - c_i \right\|^2. \quad (2)$$

However, since this method relies on a discretization of the capacities, it suffers from quadratic complexity and converges slowly. Even when implemented on a GPU [49], it is still inefficient for large-scale optimization problems. Three variants were further proposed to improve the algorithm performance.

Xu et al. [50] proposed *capacity-constrained Delaunay triangulation* (CCDT)

for blue-noise sampling and generalized the concept of CCDT to mesh surfaces [51].

Chen et al. [52] combined CCVT [48] with the CVT framework in [42] for surface blue-noise sampling, which they called *capacity-constrained centroidal Voronoi tessellation* (CapCVT). Their energy function is defined as:

$$E_{CapCVT}(\mathbf{X}) = E_{CVT}(\mathbf{X}) + \lambda E_{CapVT}(\mathbf{X}), \quad (3)$$

where  $E_{CVT}$  is the same as Eqn.(1) and  $E_{CapVT}(\mathbf{X}) = \sum_{i=1}^n \left\| \int_{V_i} \rho(\mathbf{x})dx \right\|^2$ . Then, using an efficient optimization framework based on the L-BFGS method [53], they achieved significant performance improvement.

de Goes et al. [54] reformulated CCVT as a continuous constrained minimization problem based on optimal transport, instead of the discretized approximation suggested in [48]. In addition, they used a power diagram, as opposed to a Voronoi diagram, to perform density-adapted sampling. They formulated their constrained minimization as:

$$E_{BNOT}(\mathbf{X}, \mathbf{W}) = \varepsilon(\mathbf{X}, \mathbf{W}) + \sum_{i=1}^n \lambda_i (c_i - c), \quad (4)$$

where  $\varepsilon(\mathbf{X}, \mathbf{W}) = \sum_{i=1}^n \int_{V_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 dx$ ,  $\mathbf{W} = \{w_i\}_{i=1}^n$  is the weight defined at each site.

Another kind of relaxation technique, called *farthest point optimization* (FPO), maximizes the minimal distance of a given point

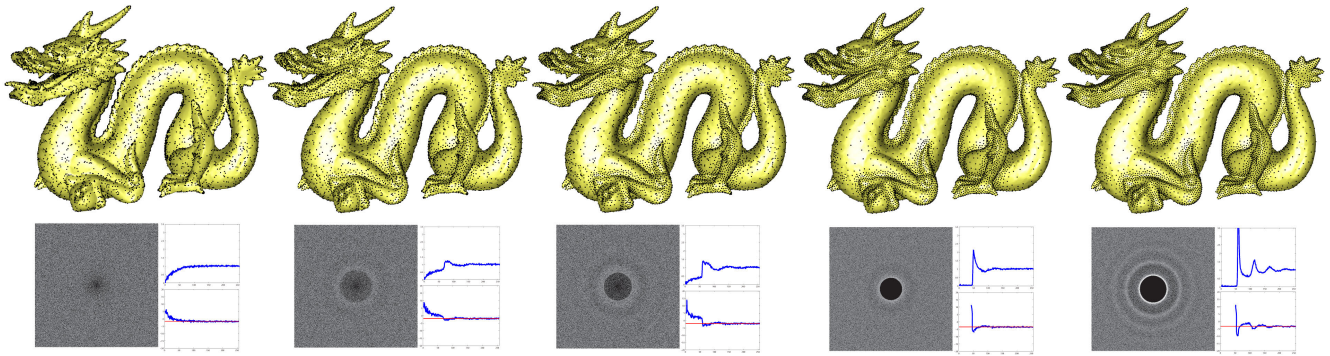


Fig. 4. An example of relaxation-based optimization [55]. From left to right: the results of increasing number of iterations. The top row are the sampling results and the bottom row are the results of spectral analysis.

set. The energy function of FPO is discrete and can be optimized only by discrete optimization method. The original FPO algorithm was proposed by Schlömer et al. [11]. An equivalent algorithm was proposed by Kanamori et al. [56] which is based on Delaunay triangulation. They successively moved each point to the farthest point (i.e., the Voronoi vertex farthest from its immediate neighbors), by removing it and reinserting it at the farthest point. Chen and Gotsman [57] parallelized the FPO framework of [11] via local Delaunay triangulation. However, these two FPO approaches could handle only 2D uniform sampling. By introducing the regular triangulation and the power diagram, Yan et al. [55] proposed two important generalizations of the original FPO framework: adaptive sampling and sampling on surfaces. Fig. 4 shows an example of applying FPO to a mesh surface.

Apart from the above two categories, there are many other approaches that aim at high-quality sampling based on relaxation techniques. Öztireli et al. [58] solved the problem of finding optimal sampling conditions based on the spectral analysis of manifolds. Fattal [2] presented an adaptive sampling algorithm based on kernel density estimation. Chen et al. [59] introduced bilateral blue-noise sampling that is suitable for dense point set sub-sampling. Ebeida et al. [60] proposed an iterative optimization method to improve blue-noise properties starting from a Poisson-disk sampled point set.

### 3.2.3 Patch/Tile-based Sampling

Poisson-disk sampling and relaxation-based methods can generate high-quality point sets, but the computational overhead can become an issue for real-time applications. Patch/Tile-based Sampling is able to generate

large point sets in real-time while sacrificing the sampling quality.

The core of a tile-based sampling method is that one or more tiles are pre-computed and then placed next to each other to form point sets of arbitrary sizes. Hiller *et al.* [61] first utilized Wang tiles to generate non-periodic point sets with blue-noise properties. Lagae and Dutré [62] extended Wang tiles to Poisson-disk tiles aiming at the rapid generation of Poisson-disk distributed point sets. A recursive tile subdivision proposed by Kopf *et al.* [63] also used Wang tiles to produce a higher level of noise. However, Wang-tile approaches usually generate sampling artifacts because the low count of prototiles and their placement on a square lattice induce a grid of peaks in Fourier spectra.

Ostromoukhov *et al.* [64] proposed to utilize non-periodic Penrose tiling for generating blue-noise patterns on 2D domains. They hierarchically subdivided a Penrose tiling and used the Fibonacci number system to label the sampled point. This technique is well suited for generation of non-uniform sampling patterns. However, this method yields rather strong artifacts in the spectral domain. Ostromoukhov then improved their work [64] for general fast hierarchical importance sampling [65]. This approach is built on self-similar tiling of the plane

or the surface of a sphere with rectifiable polyominoes, as opposed to Penrose tiling. Each polyomino contains just one sample and it is recursively subdivided until the desired local density of samples is reached. The exact position of the sampling point within the polyomino is determined by a pre-computed structural index. However, these two approaches result in poor Fourier spectra, as the single-sample tiles they use exacerbate the presence of tiling structures that are very simple or regular. Considering the drawbacks of Wang tiles and single-sample tiles, Wachtel *et al.* [66] proposed a new fast tile-based method for adaptive 2D sampling. At its heart is an adaptive non-periodic tiling with a deterministic, hierarchical construction of self-similar, equal-area, tri-hex tiles. Then an offline computation of a lookup table of optimized (spectrally controlled) point sets is used to populate the tiles.

Kalantari *et al.* [67, 68] proposed other efficient methods for fast generation of a large number of blue-noise samples. Their main idea was to generate an initial set of Poisson-disk samples first using any existing approach, and then replicate this set at various locations in the final space using the convolution theorem. This method is very fast, but its blue-noise properties are not as good as the previously mentioned Poisson-disk sampling methods s-

ince it is approximate.

### 3.2.4 Other approaches

There are several other approaches that cannot be classified by the categories discussed above. For example, Zhou et al. [69] and Heck et al. [70] studied the relationships between spatial statistics and spectral properties of point distributions, and they proposed efficient methods to generate point sets that match the given spectra. Mitchell et al. [71] generated a blue-noise quad-mesh using two-color Poisson-disk sampling. Beyond the point dart, high-dimensional darts were also investigated for blue-noise sampling [72, 73, 74, 17]. Other sampling methods focused on anisotropic sampling instead of isotropic sampling [75, 76, 77]. We do not discuss the details of these approaches here because they are out of the scope of this paper.

## 3.3 Evaluation

Different sampling algorithms result in point distributions that have different characteristics. Hence, how to choose the right method for a given application is very important. So far, there are mainly two methodologies for evaluating the quality of samples: spectral properties and geometric analysis.

**Spectral evaluation.** Spectral analysis is a

common method for evaluating the quality of point distributions and has been demonstrated to be effective in detecting sampling artifacts. The first technique was introduced by Ulichney [78] to study dither patterns. They estimated the power spectrum by averaging the periodogram of distributions, determined by Fourier transforms. Then, two useful one-dimensional statistics from the power spectrum were derived from the power spectrum. The first is the *radially averaged power spectrum*, in which the typical blue-noise characteristic should start in a sharp transition region, with a low-frequency cutoff and a flatter, high-frequency region. The second one is *Anisotropy*, that measures the radial symmetry of the power spectrum. This tool was used by [6] to compare different methods for generating Poisson-disk distributions. Schlömer et al. [79] extended the work of [78] and [6] to investigate accuracy issues regarding the spectral analysis of 2D point sets. In addition, [80] and [81] proposed approaches to analyze the quality of samples. The former was based on the statistical measure *Pair Correlation Function* (PCF), while the later utilized the amplitude and variance of the sampling spectrum.

The evaluation of blue-noise sampling on surfaces is difficult since a typical Fourier analysis cannot be directly used. Bowers et al. [30]



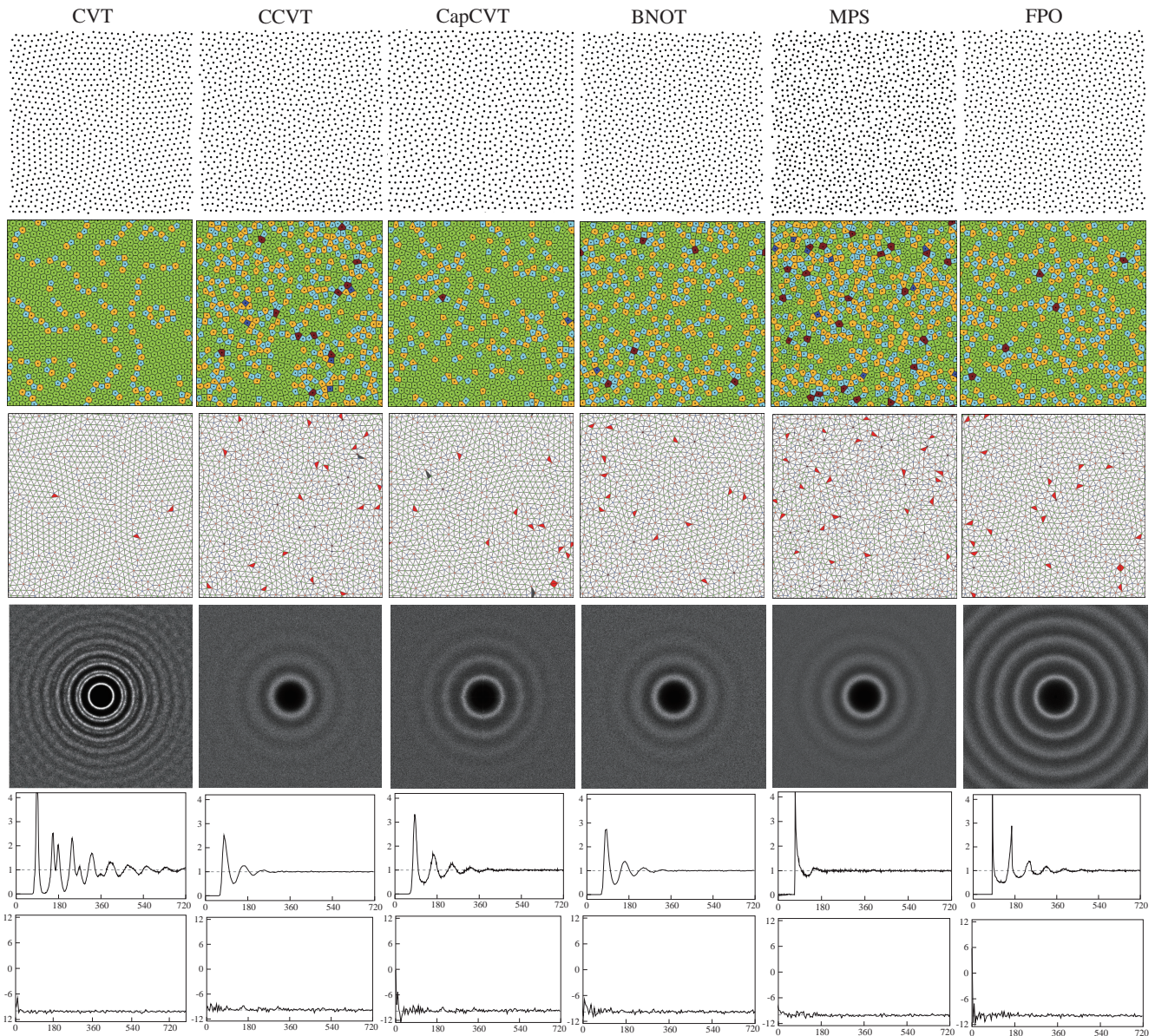


Fig. 5. Comparison of recent 2D blue-noise sampling algorithms, including CVT [44, 53], CCVT [48], CapCVT [52], BNOT [54], MPS [25], and FPO [11]. From top to bottom: distributions of 1024 points in a periodic square, Voronoi cells (each cell is color-coded by its degree: green is valence 6, orange is valence 7, light blue is valence 5, dark blue is valence 4 and brown is valence larger than 7), Delaunay triangulation (triangles with the minimal angle less than  $30^\circ$  is shown in dark gray and the obtuse triangles are shown in red), the power spectrum, radial means and anisotropy.

first proposed a spectral analysis method for surface sampling, but it could only be used

for analyzing uniform sampling. Wei and Wang [82] introduced the *Differential Domain*

Method	$\delta_{\mathbf{X}}$	$Q_{min}$	$Q_{avg}$	$\theta_{min}$	$\bar{\theta}_{min}$	$\theta_{max}$	$\theta_{<30^\circ}\%$	$\theta_{>90^\circ}\%$	$V_{567}\%$
CVT	0.793	<b>0.643</b>	<b>0.932</b>	<b>38.97</b>	<b>54.37</b>	<b>98.28</b>	<b>0</b>	<b>0.19</b>	<b>100</b>
CCVT	0.778	0.518	0.832	28.82	47.13	113.26	0.05	8.74	98.54
CapCVT	0.741	0.512	0.847	28.67	48.44	116.83	0.17	6.81	99.91
BNOT	0.766	0.570	0.848	31.67	48.23	107.08	<b>0</b>	6.34	99.02
MPS	0.781	0.487	0.806	30.19	45.30	117.11	<b>0</b>	15.07	96.53
FPO	<b>0.925</b>	0.567	0.856	35.12	50.90	107.51	<b>0</b>	6.50	99.61

**Table 1.** Statistics of 2D sampling and meshing qualities.

Model	Method	$ \mathbf{X} $	$Q_{min}$	$Q_{avg}$	$\theta_{min}$	$\bar{\theta}_{min}$	$\theta_{max}$	$\theta_{<30^\circ}\%$	$\theta_{>90^\circ}\%$	$V_{567}\%$	$d_{RMS}(\times 10^{-3})$	$d_H(\times 10^{-2})$
Venus	MPS	3.0K	<b>0.67</b>	0.85	32.7	48.6	<b>90.0</b>	<b>0</b>	<b>0</b>	<b>100</b>	0.67	0.68
	FPO	3.0K	0.57	0.85	34.2	50.8	107.1	<b>0</b>	6.29	99.7	<b>0.64</b>	0.71
	CapCVT	3.0K	0.39	0.78	20.5	43.3	128.9	4.41	17.7	98.8	0.68	0.61
	CVT	3.0K	0.65	<b>0.93</b>	<b>39.5</b>	<b>54.5</b>	97.3	<b>0</b>	0.25	<b>100</b>	0.76	<b>0.59</b>
Bunny	MPS	8.3K	0.40	0.83	33.9	53.9	103.0	<b>0</b>	0.29	<b>100</b>	0.47	0.34
	FPO	8.0K	0.39	0.84	22.6	48.7	128.5	0.56	0.29	98.0	0.45	0.34
	CapCVT	8.0K	0.39	0.84	15.6	47.3	125.1	1.04	8.46	98.7	<b>0.43</b>	<b>0.24</b>
	CVT	8.0K	<b>0.64</b>	<b>0.93</b>	<b>34.8</b>	<b>54.2</b>	<b>98.8</b>	<b>0</b>	<b>0.06</b>	99.9	0.53	0.37

**Table 2.** Statistics of remeshing qualities.

*Analysis* (DDA) technique for analyzing the spectral properties of non-uniformly sampled point sets, as well as for surface sampling. With this tool, it is possible to analyze the blue-noise properties of various methods on surfaces.

Fig. 5 compares selected blue-noise sampling algorithms by using the spectral analysis tool *PSA* provided by Schlömer et al. [79]. Fig. 6 does the comparison on mesh surfaces

by applying the differential domain analysis of Wei and Wang [82].

**Geometric evaluation.** Various spatial quantities have been proposed to measure the spatial distribution properties of samples. One common choice is the relative radius,  $\delta_{\mathbf{X}} = d_{min}/d_{max}$ , defined in [6, 79], where  $d_{min}$  is the global minimum distance for any pair of points in point set  $\mathbf{X}$  and  $d_{max}$  is the theoretical-



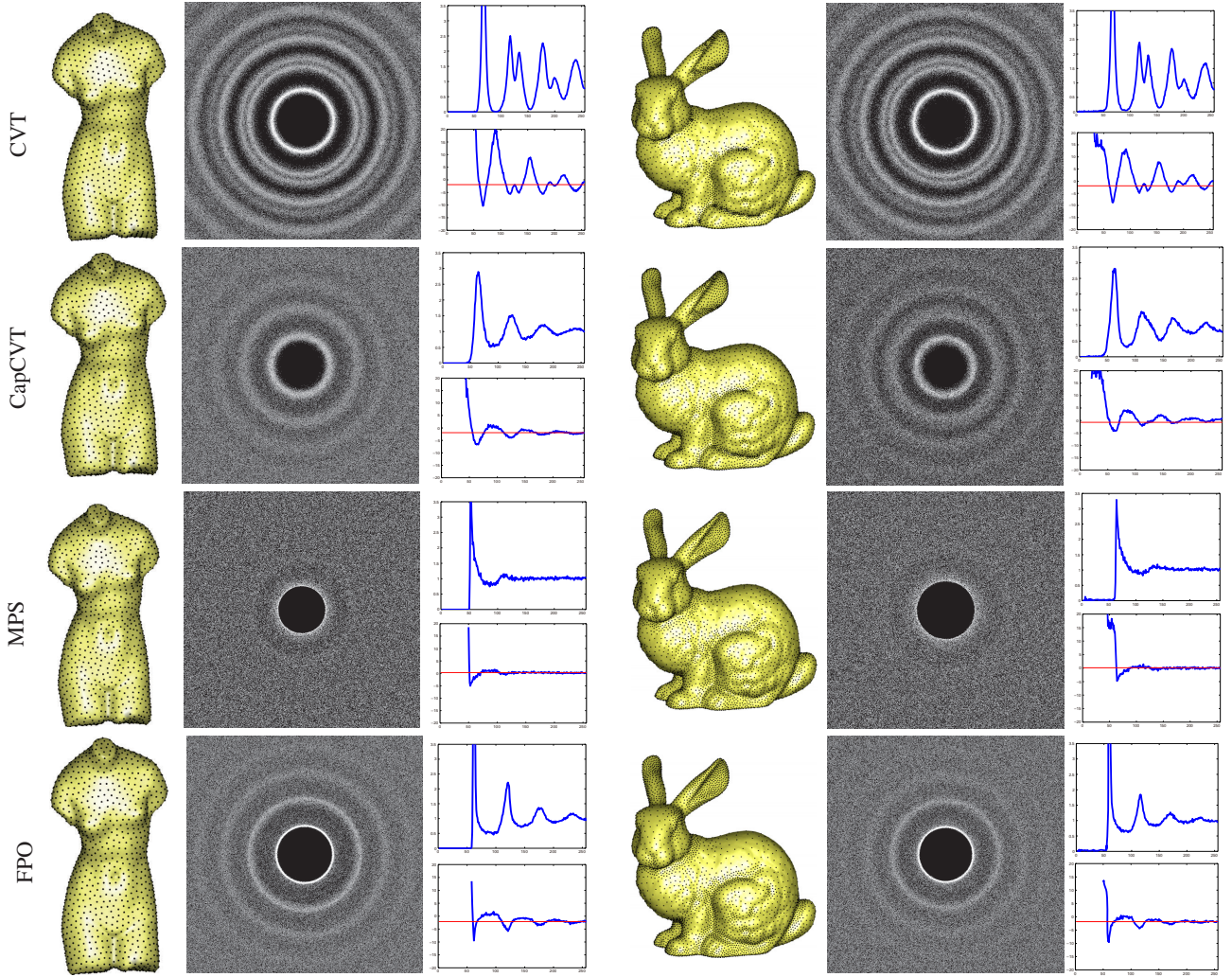


Fig. 6. Comparison of surface sampling and remeshing using CVT [42], CapCVT [52], MPS [25], FPO [55]. From left to right: sampling points, the power spectrum, radial means and anisotropy. For each method, first image shows uniform sampling on the Venus model with 1800 samples, and fourth image from left shows adaptive sampling on the Bunny model with about 6300 samples.

ly largest minimum distance between any two points (i.e.,  $d_{max} = \sqrt{(2/\sqrt{3}n)}$ ). Other spatial measures used in recent meshing/remeshing papers are listed in Tables 1 and 2. The quality of a triangle is measured by  $Q_t = \frac{6}{\sqrt{3}} \frac{s_t}{p_t h_t}$ , where  $s_t$  is the area of  $t$ ,  $p_t$  is the half-perimeter of  $t$  and  $h_t$  is the longest edge length of  $t$  [83]. Here,

$Q_{min}$  and  $Q_{avg}$  are respectively the minimal and average triangle quality;  $\theta_{min}$  and  $\theta_{max}$  are respectively the minimal and maximal angle, and  $\bar{\theta}_{min}$  is the average of the minimal angles of all triangles;  $\theta_{<30^\circ}$  and  $\theta_{>90^\circ}$  are the ratios of the triangles with  $\theta_{min}$  smaller than  $30^\circ$  and with  $\theta_{max}$  larger than  $90^\circ$ ;  $V_{567}$  is the percentage of

vertices with valences 5, 6 and 7;  $d_H$  and  $d_{RMS}$  are the Hausdorff distance and the root mean squared distance between the input mesh and the remeshing result (divided by the diagonal length of the input mesh bounding box), measured with the Metro tool [84].

## 4 Applications

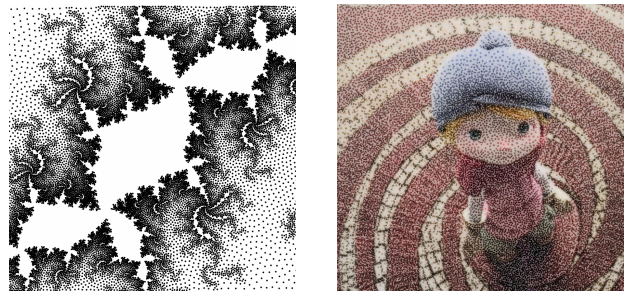
In this section, we discuss several applications that can benefit from blue-noise sampling.

### 4.1 Rendering

Rendering algorithms typically face the challenge of numerically computing high dimensional integrals. Most of the time, sampling algorithms are used and therefore sample generation is a core problem in rendering. Due to its low discrepancy and randomness, blue-noise sampling has been exploited to improve rendering quality and efficiency. Spencer and Jones [85] applied blue-noise sampling to caustics rendering powered by photon mapping. They presented a method that progressively removes noise from photon maps, which are view-independent. The resulting photon distribution holds blue-noise properties, which improve the rendering quality while avoiding a huge amount of photons. Chen et al. [59] further improved the blue-noise distribution of the photons. The so-called “bilateral blue-

noise sampling” method [59] considered not only the photon positions, but also the photon properties during relaxation, resulting in a better quality both in smooth regions and sharp features. A comparison of the rendering results is shown in Fig. 7.

### 4.2 Image/Video Stippling



(a) Image stippling [54]      (b) Video stippling [86]

Fig. 8. Blue-noise sampling for image and video stippling.

Stippling is a kind of art form that uses points to represent a drawing/painting. The contrast of the image is controlled by using different densities of the point distribution in different regions. The distribution of the points cannot be regular or there would be visual artifacts. Blue-noise sampling is well suited for this application. Secord [87] first used a weighted Voronoi diagram for image stippling. Then, various techniques were proposed to improve the quality of the stippling [88, 48, 2, 54, 52]. More recently, Ge et al. [86] used bilateral blue-noise sampling [59] for video stippling. Fig. 8

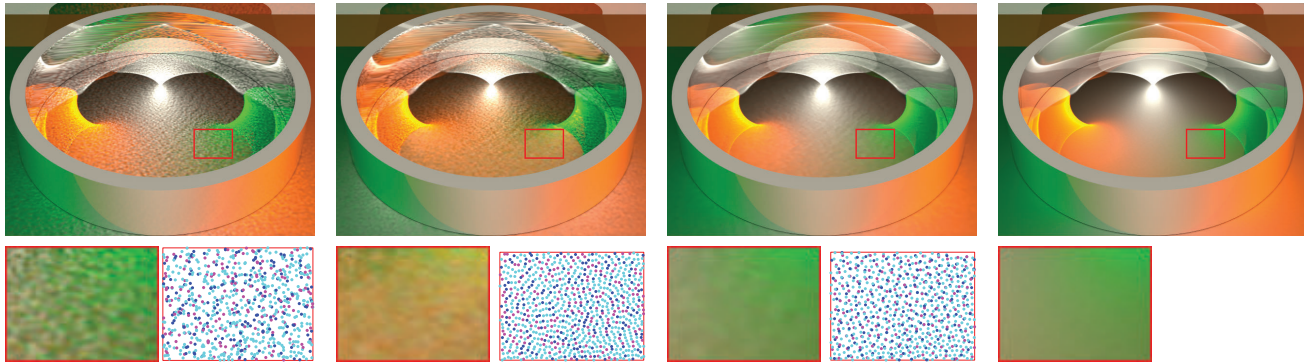


Fig. 7. A comparison of photon mapping results of different methods. (a) Unrelaxed result, (b) result of [85], (c) result of [59], (d) reference 40x photons.

shows two recent examples of image and video stippling.

### 4.3 Dynamic Sampling

We propose two new applications for dynamic sampling based on maximal Poisson-disk sampling [25].

**Plant growth simulation.** We adapt a plant growth simulation [4] to formulate it as an adaptive Poisson-disk sampling problem [25]. Starting with a set of seeds (disks with minimal radii), the radius of each plant (disk) increases over time. Different species have varying growth speeds. The plants whose centers are covered by others are removed due to insufficient light. New gaps are created and filled by new seeds repeatedly. Fig. 9 shows the simulation in different stages.

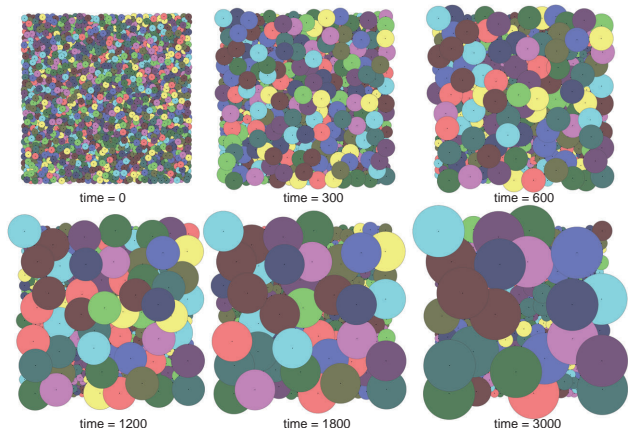


Fig. 9. Plant simulation with Poisson-disk sampling.



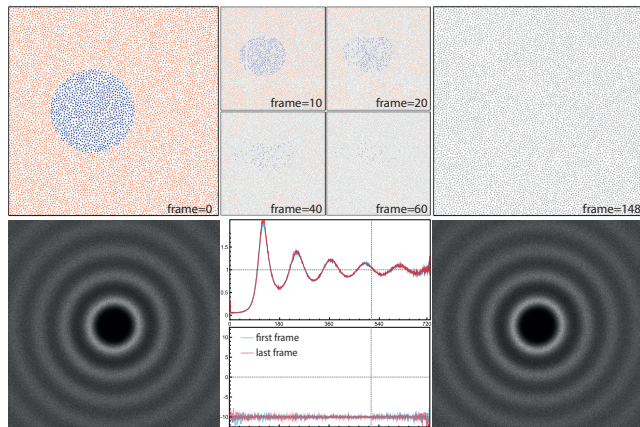


Fig. 10. Dynamic point set with Poisson-disk sampling.

**Dynamic point set.** We consider time varying blue-noise triangulations as another application of dynamic sampling. We choose to represent the changes over time as a dynamical system where the vertex positions are integrated according to a vector field. Fig. 10 (top) shows an example. This application uses efficient operations for inserting, deleting, and moving points as proposed in [25]. The spectral analysis of the first and last frame, shows that the blue-noise properties of the triangulation are maintained over time (Fig. 10 (bottom)).

#### 4.4 (Re)meshing

As another new application, we show here that blue-noise sampling can be used directly for high-quality surface remeshing and 2D/3D mesh generation.

**Surface remeshing.** Recent work of Ebei-

da et al. [89, 90] showed that the triangulation of a maximal Poisson-disk set has many elegant geometric properties, such as the edge-length bound, the angle bound, etc., which coincide with the theoretical analysis of [91]. Yan and Wonka proposed to use MPS for blue-noise surface remeshing [25]. Yan et al. [55] further improved the blue-noise properties by generalizing farthest point optimization for surface remeshing. In this section, we compare the remeshing quality of different blue-noise sampling approaches, including CVT [42], CapCVT [52], MPS [25], FPO [55]. Fig. 11 illustrates the remeshing results of the different approaches. The recent papers [25, 55] present more details on the quality comparison between the different methods.

**2D/3D meshing.** We slightly modified the sampling framework for mesh generation. The boundary of the input 2D polygon or the 3D mesh is first sampled, and then the interior of the domain is further sampled. Finally, the samples are triangulated with respect to the boundaries. The triangulation can be further optimized using the randomized optimization operators proposed by [25]. Fig. 12 and Fig. 13 show two examples of 2D and 3D blue-noise mesh generation, respectively.

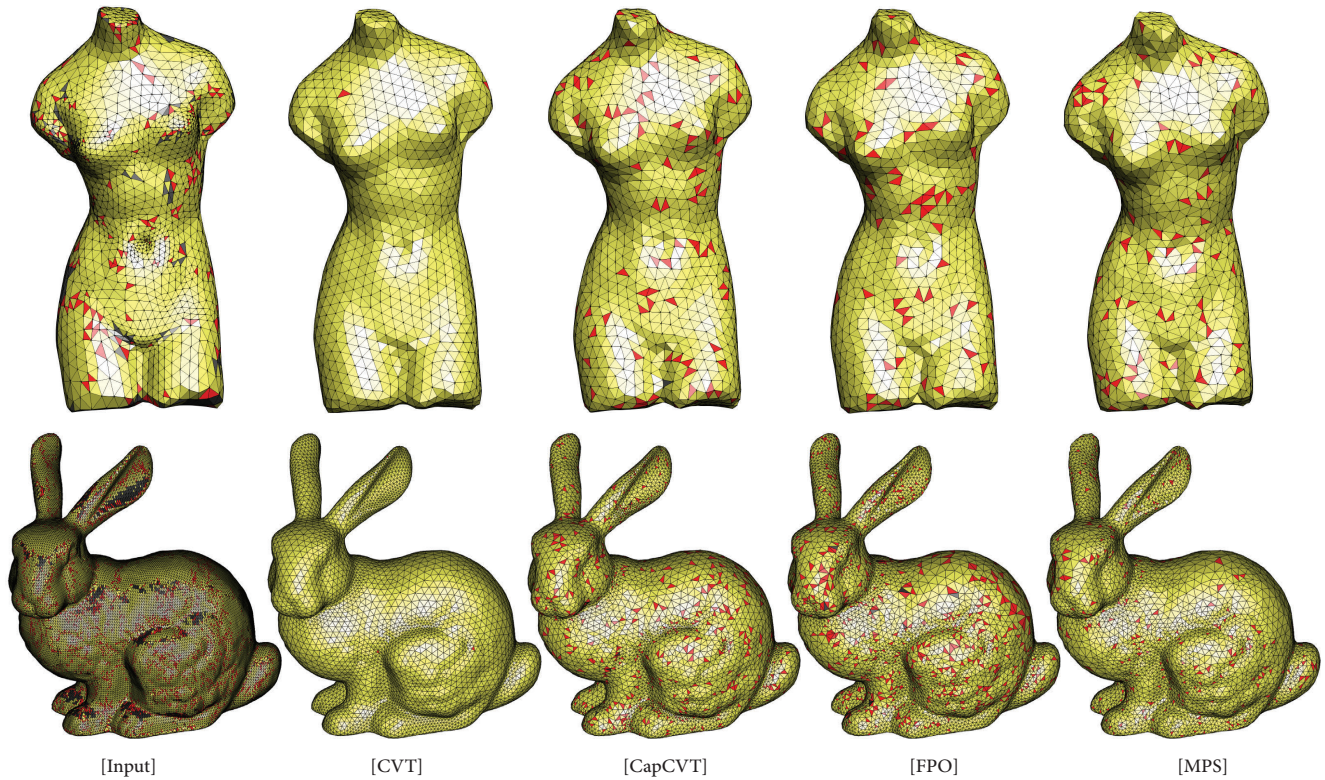


Fig. 11. Blue-noise surface remeshing. Top row: uniform remeshing. Bottom row: adaptive remeshing. The result of CVT always has the best meshing quality but lacks blue-noise features, while the other blue-noise remeshing methods are able to generate competitive results. The red triangles have angles larger than  $90^\circ$ , and the gray triangles have angles smaller than  $30^\circ$ .

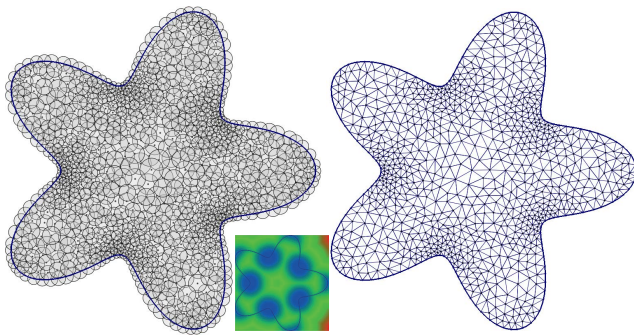


Fig. 12. 2D meshing with blue-noise sampling [25]. The angle bounds of the triangles are  $[30^\circ, 120^\circ]$ .

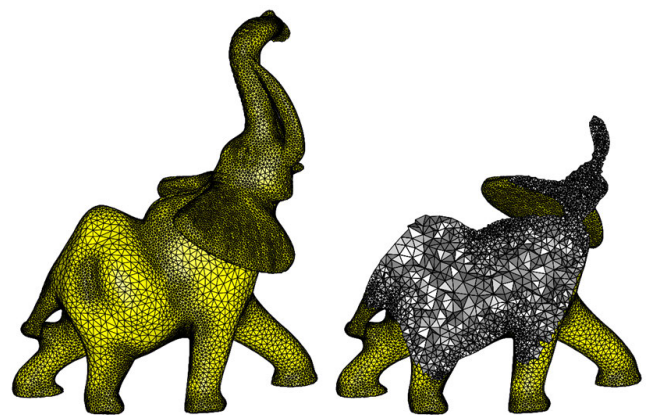


Fig. 13. 3D tetrahedral meshing with blue-noise sampling. The dihedral angle bounds of the tetrahedra are  $[15.5^\circ, 156.8^\circ]$ .

## 5 Conclusion and Future Work

This paper reviews recent works on blue-noise sampling and related applications. In this section, we briefly summarize these methods and discuss several future research topics.

Poisson-disk sampling is the traditional algorithm for blue-noise sampling and has been studied extensively. Most previous approaches cannot guarantee all three sampling criteria, especially the maximal sampling property, except for the recent works of [24, 12, 25]. It has been shown that the lack of maximal sampling has a drastic influence on the meshing quality [12, 25], an important aspect for applications like physical simulations.

Relaxation-based methods are able to generate high-quality point distributions. These methods are most suitable for applications like stippling and remeshing. However, relaxation-based methods are time consuming and not suitable for real-time applications.

Tile-based approaches can generate large-scale point sets in real time, but they sacrifice blue-noise properties, and it is not clear whether this type of method can be used for mesh generation. This is an interesting topic for further study.

As shown in Section 4.4, CVT-based remeshing generates meshes with the best meshing quality, whereas other approaches re-

*J. Comput. Sci. & Technol., Mon.. Year, ,*  
sult in meshes with better blue-noise characteristics. The question of which applications need the meshes with such blue-noise properties is worth exploring in the future.

## 6 Acknowledgement

We would like to thank Fernando de Goes, Zhonggui Chen, Li-Yi Wei and Rui Wang for making their software publicly available. This work was partially funded by National Natural Science Foundation of China (61372168, 61373071, 61372190, and 61331018), the Scientific Research Foundation for the Returned Overseas Chinese Scholars of State Education Ministry of China, the Visual Computing Center at King Abudullah University of Science and Technology (KAUST), and the open funding project of the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University (Grant No. BUAA-VR-14KF-10).

## References

- [1] D. P. Mitchell, “Generating antialiased images at low sampling densities,” in *Proc. ACM SIGGRAPH*, pp. 65–72, 1987.
- [2] R. Fattal, “Blue-noise point sampling using kernel density model,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 28, no. 3, pp. 48:1–48:10, 2011.

- [3] F. Wu, W. Dong, Y. Kong, X. Mei, D.-M. Yan, X. Zhang, and J.-C. Paul, “Feature-aware natural texture synthesis,” *The Visual Computer*, pp. 1–13, 2014. accepted.
- [4] O. Deussen, P. Hanrahan, B. Lintermann, R. Mech, M. Pharr, and P. Prusinkiewicz, “Realistic modeling and rendering of plant ecosystems,” *Proc. ACM SIGGRAPH*, pp. 275–286, 1998.
- [5] H. Schechter and R. Bridson, “Ghost SPH for animating water,” *ACM Transactions on Graphics (Proc. SIGGRAPH)*, vol. 31, no. 4, pp. 61:1–61:8, 2012.
- [6] A. Lagae and P. Dutré, “A comparison of methods for generating Poisson disk distributions,” *Computer Graphics Forum*, vol. 27, no. 1, pp. 114–129, 2008.
- [7] S. Tzeng and L.-Y. Wei, “Parallel white noise generation on a GPU via cryptographic hash,” in *Proceedings of the 2008 symposium on Interactive 3D graphics and games*, pp. 79–87, ACM, 2008.
- [8] R. Condit, P. S. Ashton, P. Baker, S. Bunyavejchewin, S. Gunatilleke, N. Gunatilleke, S. P. Hubbell, R. B. Foster, A. Itoh, J. V. LaFrankie, *et al.*, “Spatial patterns in the distribution of tropical tree species,” *Science*, vol. 288, no. 5470, pp. 1414–1418, 2000.
- [9] A. Ostling, J. Harte, and J. Green, “Self-similarity and clustering in the spatial distribution of species,” *Science*, vol. 290, no. 5492, pp. 671–671, 2000.
- [10] D. L. Lau, R. Ulichney, G. R. Arce, *et al.*, “Fundamental characteristics of halftone textures: blue noise and green noise,” *IEEE Signal Processing Magazine*, vol. 20, no. 4, pp. 28–38, 2003.
- [11] T. Schlömer, D. Heck, and O. Deussen, “Farthest-point optimized point sets with maximized minimum distance,” in *High Performance Graphics Proceedings*, pp. 135–142, 2011.
- [12] M. S. Ebeida, A. Patney, S. A. Mitchell, P. M. K. Andrew Davidson, and J. D. Owens, “Efficient maximal Poisson-disk sampling,” *ACM Transactions on Graphics (Proc. SIGGRAPH)*, vol. 30, no. 4, pp. 49:1–49:12, 2011.
- [13] M. S. Ebeida, S. A. Mitchell, A. Patney, A. A. Davidson, and J. D. Owens, “A simple algorithm for maximal Poisson-disk sampling in high dimensions,” *Computer Graphics Forum (Proc. EUROGRAPHICS)*, vol. 31, no. 2, pp. 785–794, 2012.

- [14] R. Bridson, “Fast Poisson disk sampling in arbitrary dimensions,” in *ACM SIGGRAPH 2007 Sketches*, pp. 22:1–22:5, 2007.
- [15] L.-Y. Wei, “Parallel Poisson disk sampling,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 27, no. 3, pp. 20:1–20:9, 2008.
- [16] M. N. Gamito and S. C. Maddock, “Accurate multidimensional Poisson-disk sampling,” *ACM Trans. on Graphics*, vol. 29, no. 1, pp. 8:1–8:19, 2009.
- [17] M. S. Ebeida, S. A. Mitchell, A. Patney, A. A. Davidson, S. Tzeng, M. A. Awad, A. H. Mahmoud, and J. D. Owens, *Topological and Statistical Methods for Complex Data Mathematics and Visualization*, ch. High-Dimensional Sampling: Maximal Poisson-Disk Sampling and k-d Darts, pp. 221–238. 2015.
- [18] R. L. Cook, “Stochastic sampling in computer graphics,” *ACM Trans. on Graphics*, vol. 5, no. 1, pp. 69–78, 1986.
- [19] D. Dunbar and G. Humphreys, “A spatial data structure for fast Poisson-disk sample generation,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 25, no. 3, pp. 503–508, 2006.
- [20] K. B. White, D. Cline, and P. K. Egbert, “Poisson disk point sets by hierarchical dart throwing,” in *Proceedings of the IEEE Symposium on Interactive Ray Tracing*, pp. 129–132, 2007.
- [21] T. R. Jones and D. R. Karger, “Linear-time Poisson-disk patterns,” *Journal of Graphics, GPU, and Game Tools*, vol. 15, no. 3, pp. 177–182, 2011.
- [22] L.-Y. Wei, “Multi-class blue noise sampling,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 29, no. 4, pp. 79:1–79:8, 2010.
- [23] C. Y. Ip, M. A. Yalçın, D. Luebke, and A. Varshney, “Pixelpie: Maximal Poisson-disk sampling with rasterization,” in *Proceedings of the 5th High-Performance Graphics Conference*, pp. 17–26, 2013.
- [24] T. R. Jones, “Efficient generation of Poisson-disk sampling patterns,” *Journal of Graphics Tools*, vol. 11, no. 2, pp. 27–36, 2006.
- [25] D.-M. Yan and P. Wonka, “Gap processing for adaptive maximal Poisson-disk sampling,” *ACM Trans. on Graphics*, vol. 32, no. 5, pp. 148:1–148:15, 2013.
- [26] S. A. Mitchell, A. Rand, M. S. Ebeida, and C. L. Bajaj, “Variable radii Poisson disk



- sampling,” in *24th Canadian Conference on Computational Geometry (CCCG)*, pp. 185–190, 2012.
- [27] M. S. Ebeida, A. H. Mahmoud, M. A. Awad, M. A. Mohammed, S. A. Mitchell, A. Rand, and J. D. Owens, “Sifted disks,” *Computer Graphics Forum (Proc. EUROGRAPHICS)*, vol. 32, no. 2, pp. 509–518, 2013.
- [28] C. Yuksel, “Sample elimination for generating Poisson disk sample sets,” *Computer Graphics Forum (Proceedings of EUROGRAPHICS 2015)*, vol. 37, no. 2, 2015. to appear.
- [29] D. Cline, S. Jeschke, A. Razdan, K. White, and P. Wonka, “Dart throwing on surfaces,” *Computer Graphics Forum (Proc. EGSR)*, vol. 28, no. 4, pp. 1217–1226, 2009.
- [30] J. Bowers, R. Wang, L.-Y. Wei, and D. Maletz, “Parallel Poisson disk sampling with spectrum analysis on surfaces,” *ACM Trans. on Graphics (Proc. SIGGRAPH Asia)*, vol. 29, no. 6, pp. 166:1–166:10, 2010.
- [31] M. Corsini, P. Cignoni, and R. Scopigno, “Efficient and flexible sampling with blue noise properties of triangular meshes,” *IEEE Trans. on Vis. and Comp. Graphics*, vol. 18, no. 6, pp. 914–924, 2012.
- [32] B. Geng, H. Zhang, H. Wang, and G. Wang, “Approximate Poisson disk sampling on mesh,” *SCIENCE CHINA Information Sciences*, vol. 56, no. 9, pp. 1–12, 2013.
- [33] D.-M. Yan and P. Wonka, “Adaptive maximal Poisson-disk sampling on surfaces,” in *ACM SIGGRAPH Asia Technical Briefs*, pp. 21:1–21:4, 2012.
- [34] D.-M. Yan, G.-B. Bao, X. Zhang, and P. Wonka, “Low-resolution remeshing using the localized restricted Voronoi diagram,” *IEEE Trans. on Vis. and Comp. Graphics*, vol. 20, no. 10, pp. 1418–1427, 2014.
- [35] E. Medeiros, L. Ingrid, S. Pesco, and C. Silva, “Fast adaptive blue noise on polygonal surfaces,” *Graphical Models*, vol. 76, no. 1, pp. 17–29, 2014.
- [36] J. Guo, D.-M. Yan, X. Jia, and X. Zhang, “Efficient maximal Poisson-disk sampling and remeshing on surfaces,” *Computers & Graphics*, vol. 46, no. 6-8, pp. 72–79, 2015.
- [37] D.-M. Yan, J. Wallner, and P. Wonka, “Unbiased sampling and meshing of iso-surfaces,” *IEEE Trans. on Vis. and Comp.*

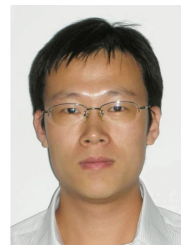
- Graphics*, vol. 20, no. 11, pp. 1579–1589, 2014.
- [38] Y. Fu and B. Zhou, “Direct sampling on surfaces for high quality remeshing,” in *ACM symposium on Solid and physical modeling*, pp. 115–124, 2008.
- [39] X. Ying, S.-Q. Xin, Q. Sun, and Y. He, “An intrinsic algorithm for parallel Poisson disk sampling on arbitrary surfaces,” *IEEE Trans. on Vis. and Comp. Graphics*, vol. 19, no. 9, pp. 1425–1437, 2013.
- [40] X. Ying, Z. Li, and Y. He, “A parallel algorithm for improving the maximal property of Poisson disk sampling,” *Computer-Aided Design*, vol. 46, no. 9, pp. 37–44, 2014.
- [41] J.-L. Peyrot, F. Payan, and M. Antonini, “Direct blue noise resampling of meshes of arbitrary topology,” *The Visual Computer*, p. accepted, 2014.
- [42] D.-M. Yan, B. Lévy, Y. Liu, F. Sun, and W. Wang, “Isotropic remeshing with fast and exact computation of restricted Voronoi diagram,” *Computer Graphics Forum*, vol. 28, no. 5, pp. 1445–1454, 2009.
- [43] S. A. Lloyd, “Least squares quantization in PCM,” *IEEE Transactions on Informa-*  
*J. Comput. Sci. & Technol., Mon.. Year, ,*  
*tion Theory*, vol. 28, no. 2, pp. 129–137, 1982.
- [44] Q. Du, V. Faber, and M. Gunzburger, “Centroidal Voronoi tessellations: applications and algorithms,” *SIAM Review*, vol. 41, pp. 637–676, 1999.
- [45] P. Alliez, É. C. d. Verdière, O. Devillers, and M. Isenburg, “Centroidal Voronoi diagrams for isotropic surface remeshing,” *Graphical Models*, vol. 67, no. 3, pp. 204–231, 2005.
- [46] G. Rong, M. Jin, L. Shuai, and X. Guo, “Centroidal Voronoi tessellation in universal covering space of manifold surfaces,” *Comp. Aided Geom. Design*, vol. 28, no. 8, pp. 475–496, 2011.
- [47] S. Valette, J.-M. Chassery, and R. Prost, “Generic remeshing of 3D triangular meshes with metric-dependent discrete Voronoi diagrams,” *IEEE Trans. on Vis. and Comp. Graphics*, vol. 14, no. 2, pp. 369–381, 2008.
- [48] M. Balzer, T. Schlömer, and O. Deussen, “Capacity-constrained point distributions: A variant of Lloyd’s method,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 28, no. 6, pp. 86:1–86:8, 2009.

- [49] H. Li, D. Nehab, L.-Y. Wei, P. V. Sander, and C.-W. Fu, “Fast capacity constrained Voronoi tessellation,” in *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pp. 13:1–13:1, 2010.
- [50] Y. Xu, L. Liu, C. Gotsman, and S. J. Gortler, “Capacity-constrained Delaunay triangulation for point distributions,” *Computers & Graphics (Proc. SMI)*, vol. 35, no. 3, pp. 510–516, 2011.
- [51] Y. Xu, R. Hu, C. Gotsman, and L. Liu, “Blue noise sampling of surfaces,” *Computers & Graphics*, vol. 36, no. 4, pp. 232–240, 2012.
- [52] Z. Chen, Z. Yuan, Y.-K. Choi, L. Liu, and W. Wang, “Variational blue noise sampling,” *IEEE Trans. on Vis. and Comp. Graphics*, vol. 18, no. 10, pp. 1784–1796, 2012.
- [53] Y. Liu, W. Wang, B. Lévy, F. Sun, D.-M. Yan, L. Lu, and C. Yang, “On centroidal Voronoi tessellation - energy smoothness and fast computation,” *ACM Trans. on Graphics*, vol. 28, no. 4, pp. 101:1–17, 2009.
- [54] F. de Goes, K. Breeden, V. Ostrovskiy, and M. Desbrun, “Blue noise through optimal transport,” *ACM Trans. on Graphics (Proc. SIGGRAPH Asia)*, vol. 31, pp. 171:1–171:12, 2012.
- [55] D.-M. Yan, J. Guo, X. Jia, X. Zhang, and P. Wonka, “Blue-noise remeshing with farthest point optimization,” *Computer Graphics Forum (Proc. SGP)*, vol. 33, no. 5, pp. 167–176, 2014.
- [56] Y. Kanamori, Z. Szego, and T. Nishita, “Deterministic blue noise sampling by solving largest empty circle problems,” *The Journal of the Institute of Image Electronics Engineers of Japan*, vol. 40, no. 1, pp. 6–13, 2011.
- [57] R. Chen and C. Gotsman, “Parallel blue-noise sampling by constrained farthest point optimization,” *Computer Graphics Forum (Proc. SGP)*, vol. 31, no. 5, pp. 1775–1785, 2012.
- [58] A. C. Öztireli, M. Alexa, and M. Gross, “Spectral sampling of manifolds,” *ACM Trans. on Graphics (Proc. SIGGRAPH Asia)*, vol. 29, no. 6, pp. 168:1–168:8, 2010.
- [59] J. Chen, X. Ge, L.-Y. Wei, B. Wang, Y. Wang, H. Wang, Y. Fei, K.-L. Qian, J.-H. Yong, and W. Wang, “Bilateral blue noise sampling,” *ACM Trans. on Graphics*

- s (*Proc. SIGGRAPH Asia*), vol. 32, no. 6, pp. 216:1–216:11, 2013.
- [60] M. S. Ebeida, M. A. Awad, X. Ge, A. H. Mahmoud, S. A. Mitchell, P. M. Knupp, and L.-Y. Wei, “Improving spatial coverage while preserving the blue noise of point sets,” *Computer-Aided Design*, vol. 46, pp. 25–36, 2014.
- [61] S. Hiller, O. Deussen, and A. Keller, “Tiled blue noise samples,” in *Proceedings of the Vision Modeling and Visualization Conference 2001*, pp. 265–272, 2001.
- [62] A. Lagae and P. Dutré, “A procedural object distribution function,” *ACM Trans. on Graphics*, vol. 24, no. 4, pp. 1442–1461, 2005.
- [63] J. Kopf, D. Cohen-Or, O. Deussen, and D. Lischinski, “Recursive Wang tiles for real-time blue noise,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 11, no. 2, pp. 509–518, 2006.
- [64] V. Ostromoukhov, C. Donohue, and P.-M. Jodoin, “Fast hierarchical importance sampling with blue noise properties,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 23, no. 3, pp. 488–495, 2004.
- J. Comput. Sci. & Technol., Mon.. Year, ,*
- [65] Ostromoukhov, “Sampling with polyominoes,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 26, no. 3, pp. 78:1–78:6, 2007.
- [66] F. Wachtel, A. Pilleboue, D. Coeurjoly, K. Breeden, G. Singh, G. Cathelin, F. de Goes, M. Desbrun, and V. Ostromoukhov, “Fast tile-based adaptive sampling with user-specified fourier spectra,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 33, no. 4, pp. 56:1–56:11, 2014.
- [67] N. K. Kalantari and P. Sen, “Fast generation of approximate blue noise point sets,” *Computer Graphics Forum (Proc. EGSR)*, vol. 31, no. 4, pp. 1529–1535, 2012.
- [68] N. K. Kalantari and P. Sen, “Efficient computation of blue noise point sets through importance sampling,” *Computer Graphics Forum (Proc. EGSR)*, vol. 30, no. 4, pp. 1215–1221, 2011.
- [69] Y. Zhou, H. Huang, L.-Y. Wei, and R. Wang, “Point sampling with general noise spectrum,” *ACM Trans. Graph.*, vol. 31, pp. 76:1–76:11, July 2012.
- [70] D. Heck, T. Schlömer, and O. Deussen, “Blue noise sampling with controlled alias-

- ing,” *ACM Trans. on Graphics*, vol. 32, no. 3, pp. 25:1–25:12, 2013.
- [71] S. A. Mitchell, M. A. Mohammed, A. H. Mahmoud, and M. S. Ebeida, “Delaunay quadrangulations by two-coloring vertices,” in *23rd International Meshing Roundtable*, pp. 364–376, 2014.
- [72] S. Tzeng, A. Patney, A. Davidson, M. S. Ebeida, S. A. Mitchell, and J. D. Owens, “High-quality parallel depth-of-field using line samples,” in *the Fourth ACM SIGGRAPH / Eurographics conference on High-Performance Graphics*, pp. 23–31, 2012.
- [73] X. Sun, K. Zhou, J. Guo, G. Xie, J. Pan, W. Wang, and B. Guo, “Line segment sampling with blue-noise properties,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 32, no. 4, pp. 127:1–127:14, 2013.
- [74] M. S. Ebeida, A. Patney, S. A. Mitchell, K. Dalbey, A. Davidson, and J. Owens, “k-d darts: Sampling by k-dimensional flat searches,” *ACM Trans. on Graphics*, vol. 33, pp. 3:1–3:16, 2014.
- [75] L. Feng, I. Hotz, B. Hamann, and K. I. Joy, “Anisotropic noise samples,” *IEEE Trans. on Vis. and Comp. Graphics*, vol. 14, no. 2, pp. 342–354, 2008.
- [76] H. Li, L.-Y. Wei, P. V. Sander, and C.-W. Fu, “Anisotropic blue noise sampling,” *ACM Trans. on Graphics (Proc. SIGGRAPH Asia)*, vol. 29, no. 6, pp. 167:1–167:12, 2010.
- [77] J. Quinn, F. Langbein, Y.-K. Lai, and R. Martin, “Generalized anisotropic stratified surface sampling,” *IEEE Trans. on Vis. and Comp. Graphics*, vol. 19, no. 7, pp. 1143–1157, 2013.
- [78] R. Ulichney, *Digital Halftoning*. MIT Press, 1987.
- [79] T. Schlömer and O. Deussen, “Accurate spectral analysis of two-dimensional point sets,” *Journal of Graphics, GPU, and Game Tools*, vol. 15, no. 3, pp. 152–160, 2011.
- [80] A. C. Öztireli and M. Gross, “Analysis and synthesis of point distributions based on pair correlation,” *ACM Trans. on Graphics (Proc. SIGGRAPH Asia)*, vol. 31, no. 6, p. 170, 2012.
- [81] K. Subr and J. Kautz, “Fourier analysis of stochastic sampling strategies for assessing bias and variance in integration,” *ACM*

- Trans. on Graphics (Proc. SIGGRAPH)*, no. 4, pp. 905–914.
- [82] L.-Y. Wei and R. Wang, “Differential domain analysis for non-uniform sampling,” *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 30, no. 4, pp. 50:1–50:8, 2011.
- [83] P. Frey and H. Borouchaki, “Surface mesh evaluation,” in *6th Intl. Meshing Roundtable*, pp. 363–374, 1997.
- [84] P. Cignoni, C. Rocchini, and R. Scopigno, “Metro: measuring error on simplified surfaces,” in *Computer Graphics Forum*, vol. 17, pp. 167–174, Wiley Online Library, 1998.
- [85] B. Spencer and M. W. Jones, “Progressive photon relaxation,” *ACM Trans. on Graphics*, vol. 32, no. 1, pp. No.7:1–7:11, 2013.
- [86] X. Ge, L.-Y. Wei, Y. Wang, and H. Wang, “Bilateral blue noise sampling: Additional algorithms and applications,” osu-cisrc-8/13-tr17, The Ohio State University, 2013.
- [87] A. Secord, “Weighted Voronoi stippling,” in *2nd International Symposium on Non-Photorealistic Animation and Rendering*, pp. 37–43, 2002.
- J. Comput. Sci. & Technol., Mon.. Year, ,*
- [88] I. Ascencio-Lopez, O. Meruvia-Pastor, and H. Hidalgo-Silva, “Adaptive incremental stippling using the Poisson-disk distribution,” *Journal of Graphics, GPU, and Game Tools*, vol. 15, no. 1, pp. 29–47, 2010.
- [89] M. S. Ebeida, S. A. Mitchell, A. A. Davidson, A. Patney, P. M. Knupp, and J. D. Owens, “Efficient and good Delaunay meshes from random points,” *Computer-Aided Design*, vol. 43, no. 11, pp. 1506–1515, 2011.
- [90] J. Guo, D.-M. Yan, G. Bao, W. Dong, X. Zhang, and P. Wonka, “Efficient triangulation of Poisson-disk sampled point sets,” *The Visual Computer*, vol. 30, no. 6–8, pp. 773–785, 2014.
- [91] L. P. Chew, “Guaranteed-quality triangular meshes,” department of computer science tech report 89-983, Cornell University, 1989.



**Dong-Ming Yan** is a research scientist at King Abdullah University of Science and Technology (KAUST), and he is an Associate Professor at National Laboratory of Pattern

Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences. He received his PhD in computer science from Hong Kong University in 2010, and his Master and Bachelor degrees in computer science and technology from Tsinghua University in 2005 and 2002, respectively. His research interests include computer graphics, geometric processing and visualization.



**Jianwei Guo** received his bachelor degree from Shandong University in 2011, and he is currently a PhD candidate in National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences. His research interests include 3D shape analysis and geometry processing.



**Bin Wang** is currently an associate professor at School of Software, Tsinghua University, China. He received his B.Sc. degree in Chemistry in 1999, and his Ph.D. in Computer Science from Tsinghua University in

2005. He was a research assistant at Department of Computer Science, Hong Kong University, and had postdoctoral research training at ISA/ALICE Research Group, INRIA-LORIA, France. His research interests include photorealistic rendering, non-photorealistic rendering, and image and video processing.



**Xiaopeng Zhang** is a Professor in National Laboratory of Pattern Recognition at Institute of Automation, Chinese Academic of Sciences (CAS). He received his Ph.D. degree in Computer Science from Institute of Software, CAS in 1999. He received the National Scientific and Technological Progress Prize (second class) in 2004. His main research interests include computer graphics and image processing.



**Peter Wonka** received the Ph.D. degree in computer science and the M.S. degree in urban planning from the Technical University of Vienna, Vienna, Austria, in 2001 and 2002, respectively. He was a Postdoctoral

Researcher with the Georgia Institute of Technology, Atlanta, GA, USA, for two years. He is currently a Professor with the Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and

*J. Comput. Sci. & Technol., Mon.. Year, ,*  
Technology, Thuwal, Saudi Arabia, and also an Associate Professor with Arizona State University, Tempe, AZ, USA. His research interests include topics in computer graphics, visualization, computer vision, remote sensing, image processing, and machine learning.