Variational Blue Noise Sampling

Zhonggui Chen, Zhan Yuan, Yi-King Choi, Ligang Liu, and Wenping Wang

Abstract—Blue noise point sampling is one of the core algorithms in computer graphics. In this paper we present a new and versatile variational framework for generating point distributions with high-quality blue noise characteristics while precisely adapting to given density functions. Different from previous approaches based on discrete settings of capacity-constrained Voronoi tessellation, we cast the blue noise sampling generation as a variational problem with continuous settings. Based on an accurate evaluation of the gradient of an energy function, an efficient optimization is developed which delivers significantly faster performance than the previous optimization-based methods. Our framework can easily be extended to generating blue noise point samples on manifold surfaces and for multi-class sampling. The optimization formulation also allows us to naturally deal with dynamic domains, such as deformable surfaces, and to yield blue noise samplings with temporal coherence. We present experimental results to validate the efficacy of our variational framework. Finally, we show a variety of applications of the proposed methods, including non-photorealistic image stippling, color stippling, and blue noise sampling on deformable surfaces.

Index Terms—Point sampling, blue noise, centroidal Voronoi tessellation, capacity-constrained, quasi-Newton method

1 Introduction

The problem of sampling, or point set generation, is about how to generate a point set with a certain distribution. It plays an important role in computer graphics as well as many other fields. In computer graphics, the quality of a point set distribution is of major concern in diverse contexts such as digital halftoning, point-based modeling and rendering, anti-aliasing, and distributed ray tracing. It is often desirable to have a uniformly distributed yet randomly located point set. Repetitive patterns should be avoided as they are prone to producing aliasing. A point set which is uniformly distributed without regularity artifacts is said to exhibit blue noise characteristics.

Lloyd's method [1], which iteratively moves each point to the centroid of the corresponding Voronoi cell, is a popular optimization method for enhancing the blue noise properties of a given point set [2]. This method converges to a point distribution corresponding to a centroidal Voronoi tessellation (CVT) in which each point is the centroid of its Voronoi cell. Although it produces a uniformly distributed point set, such a point distribution exhibits regular patterns because the Voronoi cells of a CVT are regular hexagons in an asymptotic sense [3]. The problem of using Lloyd's method for generating blue noise sampling is therefore the control of the number of iterations to avoid convergence. On the other hand, using Lloyd's method to compute a CVT is slow, but Liu et al. [4] showed that the computation of CVT can be accelerated by employing quasi-Newton methods.

Recently, Balzer et al. [5] presented a variant of Lloyd's method for generating point distribution by introducing the capacity-constrained Voronoi tessellation (CapVT), in which each point obtains equal capacity (i.e., the mass of its Voronoi cell). By requiring also that each point coincides with the centroid of its Voronoi cell in the CapVT, the resulting point distributions were shown to possess the blue noise properties. However, their method is implemented in the discrete setting, which is highly dependent on the resolution of the discretization of the domain, and is therefore computationally inefficient.

Inspired by [5] and [4], we propose a variational framework based on a new energy function combining the CVT energy and the CapVT energy. Both the capacity constraint and the centroid constraint are considered as soft constraints in our framework. The development of this function is based on the observation that CVT accounts for generating the uniform point distribution while CapVT tends to generate point distribution without regularity artifact.

We derive formulae for accurate evaluation of the gradient of the new energy function and present an efficient numeric approach to optimizing the energy function that integrates a fast local search based on the L-BFGS method [6]. Our method is implemented in the continuous setting and achieves significantly faster performance (generally two orders of magnitude faster for a large number of points) over the method of [5] without sacrificing blue noise proper-

- *Z. Chen is with the Department of Computer Science, Xiamen University, Xiamen, 361005, China, and The University of Hong Kong, Hong Kong, China. Part of this work was done while Z. Chen was a postdoc at The University of Hong Kong. E-mail: chenzhonggui@xmu.edu.cn*
- *Z. Yuan, Y.-K. Choi and W. Wang are with the Department of Computer Science, The University of Hong Kong, Hong Kong, China. E-mail: {zyuan, ykchoi, wenping}@cs.hku.hk*
- *L. Liu is with the School of Mathematical Sciences, University of Science and Technology of China, Hefei, China. E-mail: lgliu@ustc.edu.cn*

ties. We also study the accommodation of the density functions and reveal the relationship between the density functions used in the CVT and CapVT energy functions.

Our framework is flexible and can easily be extended to other domains such as surfaces. Temporal sampling coherence required in dynamic domains such as deformable surfaces can also naturally be dealt with within this optimization framework. We also show how our method can be extended to handle multi-class sampling in which point samples in individual classes, as well as those in their union, are required to exhibit blue noise properties simultaneously.

The contributions of this work are summarized as follows:

- A variational framework based on a new energy formulation is developed for generating point distributions with blue noise characteristics. Based on the accurate evaluation of gradient of the energy function, we present an efficient optimization approach to minimizing the energy function and gain significantly faster performance than the method of [5].
- We extend our variational framework to generating blue noise sampling points on other domains including surfaces and dynamic domains. We also develop a variational framework for handling multi-class sampling. All the variational frameworks for these different domains benefit from our efficient optimization methods and thus attain fast performances.

2 Related Work

Due to its unique spatial and spectral properties, blue noise sampling has been extensively studied in the literature [7]. We mainly review previous works which are most relevant to our work.

Blue noise sampling in 2D: A Poisson disk sampling can yield a blue noise point distribution [8]. Dart throwing is a classical method to produce Poisson disk distributed point sets, in which randomly located points are generated one by one and a newly generated point is only accepted if there is no other existing points lying within a given radius. This process is simple but very slow. There are various works on accelerating the dart throwing approach [9], [10], [11], [12], [13]. The acceleration is usually achieved by encoding the vacant regions where it is legal to place a dart, or/and by parallel implementation.

Another type of approaches [14], [15], [16] generates blue noise samples in a set of small domains, called tiles, with toroidal boundary conditions in a preprocessing step. These tiles are then used for generating a non-periodic tiling of the plane. Ostromoukhov et al. proposed methods for hierarchical importance sampling with blue-noise properties by using special tiles, Penrose tiles [17] and polyominoes [18]. These methods are generally fast and allow progressive refinement.

The relaxation approach moves the points in a given set to enhance the blue noise properties. Lloyd's method [1], [2] is commonly used due to its simplicity. However, the points converge to a regular hexagonal pattern and as pointed by Balzer et al. [5], it is hard to decide a suitable iteration number for Lloyd's method in practice. To this end, Balzer et al. [5] proposed a variant of Lloyd's approach using the capacity constraints, which makes all Voronoi cells have equal mass, to achieve blue noise sampling. However, their proposed implementation involves the discretization of the underlying domain and is therefore slow (while its acceleration [19] by parallel implementation is possible).

A recent method by Schmaltz et al. [20] models the points as charged particles which repulse each other. The uniform point distributions with high-quality blue-noise properties are obtained by simulating the Coulomb interactions between points. Fattal [21] describes an optimization-based method for blue-noise point sampling by using kernel density model. Like the CVT energy function, the global minimizer of the energy function defined in [21] is given by a hexagonal arrangement of points. To avoid the regularity artifacts, a statistical model is defined to allow solutions that are less energetically-favorable. By iteratively enlarging the minimum distance between points, Schlömer et al. [22] construct irregular distributions with a significantly higher minimum distance than previous methods.

Sampling on surfaces: Point sampling on surfaces is also crucial to many important applications in computer graphics, such as texturing, remeshing, and rendering. By means of parameterization, the relaxation method [23], [24] and tile-based method [25] can be applied to blue noise sampling on surfaces. However, these methods need pre-computed parameterizations and are thus unsuitable for applications with dynamic geometry. The extensions of dart throwing method for surfaces are described in [26], [27], [28]. The computational complexity and the approximation of the geodesic metric are the major concerns in these extensions. The recent work of Öztireli et al. [29] provides a new approach to improve quality of the sampling of a surface by maximizing measures derived from spectral analysis theorem. Although their algorithm is shown to be able to handle adaptive surface sampling, it seems hardly able to make the point distributions adapt to given density functions.

Our variational framework can easily be extended to generating blue noise samples on surfaces from a given initial configuration. We optimize the positions of points on the surface directly, and no parameterization is needed. Due to the optimization nature of our framework, it is suitable for applications that require sampling coherence on deformable surfaces.

Multi-class blue noise sampling: Recently, Wei [30] introduces the problem of multi-class blue noise sampling in which not only each individual class but also their union exhibit blue noise properties and presents two approaches to generating such multi-class blue noise samples. This is quite useful for applications (e.g., object distribution and color stippling) that require a well distribution of multiple classes of samples. Schmaltz et al. [31] generalize the electrostatic halftoning approach [20] for multi-class blue noise sampling. As pointed in [30], Lloyd relaxation fails to handle multiple classes of samples. With a variant of energy function and an elaborate initial configuration, our capacity-constrained CVT method is shown to be capable of generating high-quality multi-class blue noise sampling.

3 VARIATIONAL APPROACH

In this section we introduce our variational approach to generating blue noise samplings.

3.1 Energy formulation

Centroidal Voronoi tessellation (CVT): Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a set of points, called *sites*, in a compact domain $\Omega \in \mathbb{R}^d$. Suppose that $\rho(\mathbf{x}) \geq 0$ is a density function defined in Ω. The Voronoi cell V_i of a site $\mathbf{x}_i$ is given by

$$V_i = \{\mathbf{x} \in \Omega \mid d(\mathbf{x}, \mathbf{x}_i) \leq d(\mathbf{x}, \mathbf{x}_j),\ \forall j \neq i,\ j = 1, \ldots, n\},$$

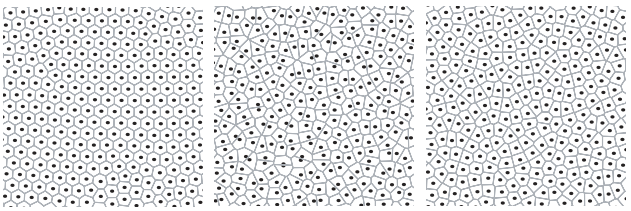
where $d(\mathbf{x}, \mathbf{y})$ is the Euclidean distance between two points $\mathbf{x}$ and $\mathbf{y}$. The collection of the Voronoi cells $\{V_i\}_{i=1}^n$ constitutes a Voronoi tessellation of Ω. If each site $\mathbf{x}_i$ coincides with the mass centroid of V_i, i.e., $\mathbf{x}_i = \frac{\int_{V_i} \rho(\mathbf{x})\mathbf{x}\mathrm{d}\sigma}{\int_{V_i} \rho(\mathbf{x})\mathrm{d}\sigma}$, where $\mathrm{d}\sigma$ is the differential area element of Ω, then $\{V_i\}_{i=1}^n$ is called a centroidal Voronoi tessellation (CVT).

From a variational standpoint, a CVT is characterized by a critical point of the following CVT energy function [32]:

$$E_{\mathrm{CVT}}(\mathbf{X}) = \sum_{i=1}^{n} \int_{V_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 \, \mathrm{d}\sigma. \tag{1}$$

The most commonly used method for minimizing $E_{\mathrm{CVT}}(\mathbf{X})$ is the Lloyd's method [1]. Recently Liu et al. [4] proved that $E_{\mathrm{CVT}}(\mathbf{X})$ is a C^2 function and proposed an efficient quasi-Newton method for minimizing it. The minimization of $E_{\mathrm{CVT}}(\mathbf{X})$ ensures that all the sites are isotropically *uniform*; however, the site distribution exhibits regular patterns as is shown in Fig. 1(a).

Capacity-constrained Voronoi tessellation (CapVT): A Voronoi tessellation $\{V_i\}_{i=1}^n$ is called a capacity-constrained Voronoi tessellation (CapVT) [5] if its Voronoi cells satisfy the constraints $|V_i| = \int_{V_i} \varrho(\mathbf{x}) \, \mathrm{d}\sigma = c_i$ $(i = 1, \ldots, n)$ where $\varrho(\mathbf{x}) \geq 0$ is a density function defined in Ω and the c_i are capacity constraints with $c_i > 0$ and $\sum_{i=1}^n c_i = \int_\Omega \varrho(\mathbf{x}) \, \mathrm{d}\sigma$.



(a) E_{CVT} (b) E_{CapVT} (c) E_{CapCVT}

Fig. 1. Site distributions and Voronoi tessellations by optimizing the three different energies. Site distribution is (a) uniform but regular; (b) irregular but nonuniform; (c) uniform and irregular. λ is set to 30 in E_{CapCVT}.

We formulate CapVT as a minimization of the function $\sum_{i=1}^n \left(\int_{V_i} \varrho(\mathbf{x}) \, \mathrm{d}\sigma - c_i \right)^2$. Assume each Voronoi cell has the same capacity, that is $c_i = c$ for all i. Hence,

$$\begin{aligned} &\sum_{i=1}^{n} \left(\int_{V_i} \varrho(\mathbf{x}) \, \mathrm{d}\sigma - c_i \right)^2 \\ =&\sum_{i=1}^{n} \left(\int_{V_i} \varrho(\mathbf{x}) \, \mathrm{d}\sigma \right)^2 - 2c \int_\Omega \varrho(\mathbf{x}) \, \mathrm{d}\sigma + \sum_{i=1}^{n} c^2. \end{aligned}$$

Since the second and the third terms on the right hand side of the above expression are constants, as far as function minimization is concerned, we can define the CapVT energy function as

$$E_{\mathrm{CapVT}}(\mathbf{X}) = \sum_{i=1}^{n} \left(\int_{V_i} \varrho(\mathbf{x}) \, \mathrm{d}\sigma \right)^2. \tag{2}$$

Balzer [33] proposes to minimize (2) using the downhill simplex method [34]. The method is derivative-free and requires only function evaluations. However, it is inefficient for large-scale optimization problem. As stated in [33], the downhill simplex method usually computes millions of function samples before it converges. Large sets of sites, with many thousands or even millions of sites, are beyond the computational feasibility of the method. We will show that (2) can be efficiently optimized by quasi-Newton method.

Starting from a random initialization (so that site capacities are not equal), minimizing $E_{\mathrm{CapVT}}(\mathbf{X})$ in (2) gives an irregular point distribution which is however nonuniform, since the energy does not regulate point positions (Fig. 1(b)).

Our variational energy function (CapCVT): We propose a new energy function by combining the CVT energy function E_{CVT} and the CapVT energy function

E_{CapVT} as

$$E_{\mathrm{CapCVT}}(\mathbf{X}) = E_{\mathrm{CVT}}(\mathbf{X}) + \lambda E_{\mathrm{CapVT}}(\mathbf{X}) = \sum_{i=1}^{n}\int_{V_i}\rho(\mathbf{x})\|\mathbf{x}-\mathbf{x}_i\|^2\mathrm{d}\sigma + \lambda\sum_{i=1}^{n}\left(\int_{V_i}\varrho(\mathbf{x})\mathrm{d}\sigma\right)^2, \tag{3}$$

where λ is a weight to balance the two energy terms. We call E_{CapCVT} *the capacity-constrained centroidal Voronoi tessellation* (CapCVT) energy function. Note that, we use different density functions $\rho(\mathbf{x})$ and $\varrho(\mathbf{x})$ for E_{CVT} and E_{CapVT}, respectively. We will discuss about choosing $\rho(\mathbf{x})$ and $\varrho(\mathbf{x})$ in Section 3.3.

The energy term E_{CapVT} in (3) functions as an equal capacity constraint and serves to introduce *irregularity* to avoid the regularity artifact in the site distribution often found in a CVT. Therefore, minimizing the new CapCVT energy function in (3) tends to generate uniform point distribution with less regularity artifact as can be seen in Fig. 1(c).

3.2 Optimization with L-BFGS method

Gradient of the energy function: The energy E_{CVT} is proved to be C^2 [4]. Similarly we can prove that E_{CapVT} is C^2 and so is E_{CapCVT}. The gradient of E_{CVT} is given in [32] as

$$\frac{\partial E_{\mathrm{CVT}}(\mathbf{X})}{\partial \mathbf{x}_i} = 2|V_i|(\mathbf{x}_i - \mathbf{c}_i), \tag{4}$$

where $|V_i| = \int_{V_i}\rho(\mathbf{x})\mathrm{d}\sigma$ and $\mathbf{c}_i = \frac{\int_{V_i}\rho(\mathbf{x})\mathbf{x}\mathrm{d}\sigma}{\int_{V_i}\rho(\mathbf{x})\mathrm{d}\sigma}$. We have derived the formulation of the gradient of E_{CapVT} which is given by (see Theorem 3 and its proof in Section 3.5):

$$\frac{\partial E_{\mathrm{CapVT}}(\mathbf{X})}{\partial \mathbf{x}_i} = 2\sum_{j\in N_i}\frac{|V_i|-|V_j|}{\|\mathbf{x}_j-\mathbf{x}_i\|}\int_{V_{ij}}\varrho(\mathbf{x})(\mathbf{x}-\mathbf{x}_i)\mathrm{d}s,$$

where N_i is the set of indices of Voronoi cells adjacent to V_i and $V_{ij} = V_i \cap V_j$ is the common face of Voronoi cells V_i and V_j. Thus the gradient of E_{CapCVT} can be written explicitly as:

$$\frac{\partial E_{\mathrm{CapCVT}}(\mathbf{X})}{\partial \mathbf{x}_i} = \frac{\partial E_{\mathrm{CVT}}(\mathbf{X})}{\partial \mathbf{x}_i} + \lambda\frac{\partial E_{\mathrm{CapVT}}(\mathbf{X})}{\partial \mathbf{x}_i}. \tag{5}$$

L-BFGS method: Since E_{CapCVT} is C^2 with an explicit gradient formula, we can use Newton-type methods to minimize it. Specifically, we apply the L-BFGS method [6], a fast local search scheme which significantly improves the space requirement of the original BFGS method [35] while preserving its super-linear convergence. Unlike the original BFGS method which uses the gradient information of all the preceding steps, the L-BFGS method computes the approximate Hessian by accumulating the gradients over a small fixed number of preceding iterations. Both the storage requirement and the computational cost of each iteration of L-BFGS are of $O(n)$, where n is the number of variables. The pseudo-code of our algorithm working on a 2D domain is given in Algorithm 1. The L-BFGS method terminates when there is no significant decrease of the energy value.

Algorithm 1 Variational CapCVT framework on 2D domain

Input: an initial point set $\mathbf{X}$ and a weight λ
Output: a new point distribution $\mathbf{X}$

$\mathbf{X}^{(0)} \leftarrow \mathbf{X}$
$\Delta E \leftarrow E_{\mathrm{CapCVT}}(\mathbf{X}^{(0)})$
$k \leftarrow 0$
while $\Delta E / E_{\mathrm{CapCVT}}(\mathbf{X}^{(k)}) > 10^{-5}$ **do**
  Compute the Voronoi tessellation of $\mathbf{X}^{(k)}$
  Compute the gradient $\nabla E_{\mathrm{CapCVT}}$
  Compute $\Delta\mathbf{X}$ using the **L-BFGS** updating rule
  $\mathbf{X}^{(k+1)} \leftarrow \mathbf{X}^{(k)} + \Delta\mathbf{X}$
  $\Delta E \leftarrow E_{\mathrm{CapCVT}}(\mathbf{X}^{(k+1)}) - E_{\mathrm{CapCVT}}(\mathbf{X}^{(k)})$
  $k \leftarrow k + 1$
end while
return $\mathbf{X}^{(k)}$

3.3 Discussions

Initialization: It is known that a local search method often ends up at a local minimum for a non-convex function like E_{CapCVT}. One may expect to use some global search scheme to find its global minimum. In general, it is hard, if not impossible, to depict a global optimizer of E_{CapCVT} as it all depends on the domain complexity, boundary effect, the number of sites, the domain density, etc.

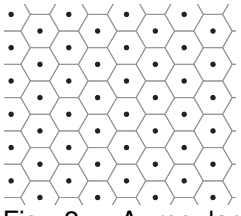


Fig. 2. A regular hexagonal lattice.

However, the global optimizer of E_{CapCVT} might not give the desirable results as far as blue noise properties are concerned. In some special cases, the global optimizer will indeed give a regular distribution of sites which is undesirable for blue noise sampling. Fig. 2 shows such an example in which the Voronoi cells form a regular hexagonal lattice and the sites are centrosymmetric. Both the CVT and CapVT energies attain their respective global minimum in this case. It is therefore expected that given such an initial point pattern, optimizing E_{CapCVT} will not break its regularity. This phenomenon is also true of several other energy function-based methods. In our case, it is found that with an arbitrary, random initialization, our method has high probability for stopping at a local minimum which yields a desirable blue noise point distribution. Hence, random point distribution (e.g., error diffusion) is used as the initialization in Algorithm 1. All the experiments we have tested show that the results of point distributions have blue noise characteristics.

Density function adaptation: In many practical applications such as importance sampling in computer graphics, the distribution of the sites is expected to adapt to some given density function $\varphi(\mathbf{x})$ in the region Ω, that is, the probability of one site locating in some region (the density of site distribution) is supposed to be equal to the density function at this site.

The density function $\rho(\mathbf{x})$ in $E_{\mathrm{CVT}}(\mathbf{X})$ is simply set to be $\varphi(\mathbf{x})$ in general. Balzer et al. [5] observed that with this density function, the CVT implicitly blurs the density function so that fewer sites than expected are found in regions of high density and more sites are found in low density regions. We also noted this issue and further discovered the relationship between $\rho(\mathbf{x})$ and $\varphi(\mathbf{x})$ in an asymptotic sense (see the proof in Section 3.5). We found that in a 2D domain $\rho(\mathbf{x}) = \varphi^2(\mathbf{x})$ should be used. This is done in our system to achieve a given point density (Fig. 3). Simply setting $\rho(\mathbf{x})$ in $E_{\mathrm{CVT}}(\mathbf{X})$ directly to be $\varphi(\mathbf{x})$ indeed accounts for the blurring by the traditional CVT reported in [5].

For the density function $\varrho(\mathbf{x})$ in $E_{\mathrm{CapVT}}(\mathbf{X})$, since the capacity of a site offers a good measurement of the quality of a density function adaption for the site distribution, it is therefore set as $\varphi(\mathbf{x})$ naturally.

Normalizing $E_{\mathrm{CVT}}(\mathbf{X})$ & $E_{\mathrm{CapVT}}(\mathbf{X})$ and the choice of λ: The two energies $E_{\mathrm{CVT}}(\mathbf{X})$ and $E_{\mathrm{CapVT}}(\mathbf{X})$ have different order of magnitudes and therefore normalization is needed. We prove that in the asymptotic case, the value ratio of $E_{\mathrm{CVT}}(\mathbf{X})$ and $E_{\mathrm{CapVT}}(\mathbf{X})$ is about $1:6$, which is independent of the number of the sites, size of the domain, and the given density function. The proof can be found in Section 3.5.

The parameter λ provides a mechanism to adjust the relative importance of $E_{\mathrm{CVT}}(\mathbf{X})$ and $E_{\mathrm{CapVT}}(\mathbf{X})$, thus affecting the irregularity of the resulting point distributions and therefore the blue noise properties. We evaluate the point distributions obtained by our method with different λ values using the spatial and spectrum analysis by [36]. For each λ value, we perform 10 runs of our method on 1,000 points with different initialization and the averaged periodograms, radial mean power, and anisotropy are computed (Fig. 4(a-e)). We also use the normalized Poisson disk radius α suggested by Lagae and Dutré [7] to evaluate the point distribution quality. The radius α is a value ranged between 0 and 1, with 0 corresponding to a distribution having two coinciding points and 1 corresponding to a regular hexagonal lattice distribution. Fig. 4(f) shows that the normalized Poisson disk radius of the distribution generated by our methods decreases gradually as λ increases. It can also be seen that, when λ is large and our method is working more towards a CapVT, the normalized radius decreases significantly.

Lagae and Dutré [7] recommend $\alpha \approx 0.75$ for high-quality blue noise point sets. However, Schlömer et al. [22] show that Poisson disk distributions can have radii up to 0.93 or higher. Considering also the spectrum properties, we found that our method with $\lambda \in [20, 100]$ gives a point distribution with good blue noise characteristics. In our system, we use a default value $\lambda = 30$.

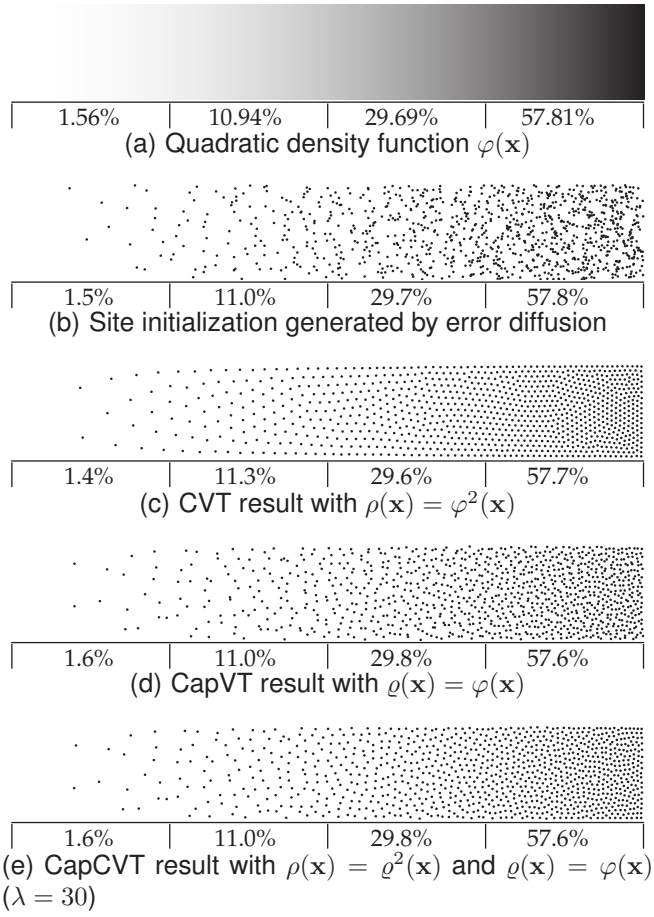
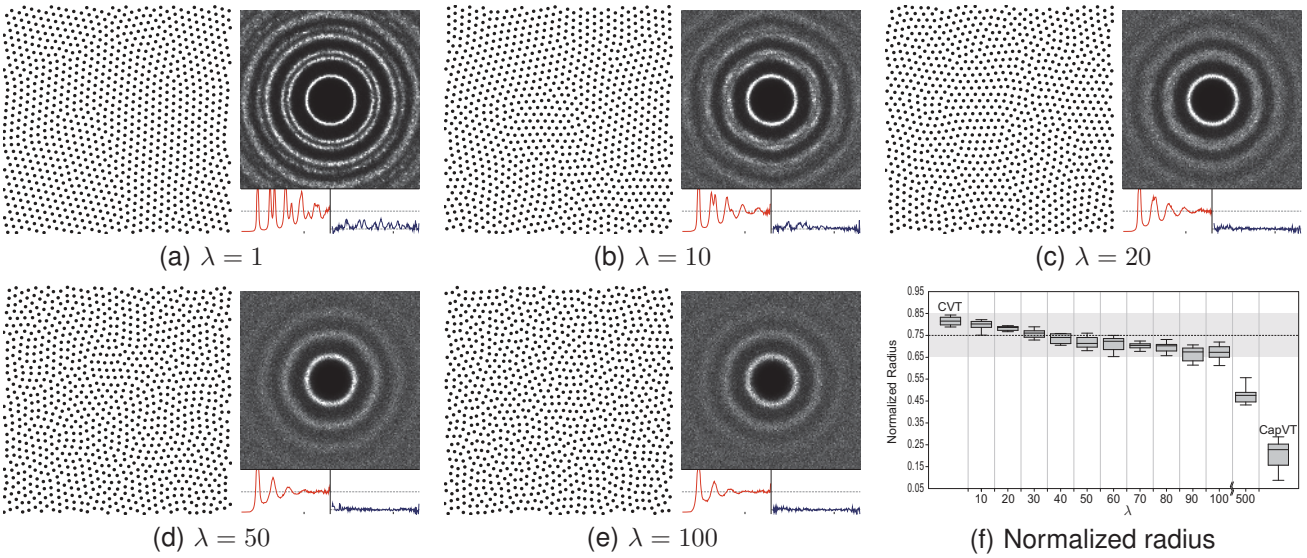


Fig. 3. The quadratic ramp in (a) is used as density function. Starting with the initial 1,000 sites in (b), our CVT, CapVT, and CapCVT results show precise adaption with the density function. The percentages indicate the density (i.e., the number of points) in each quarter.

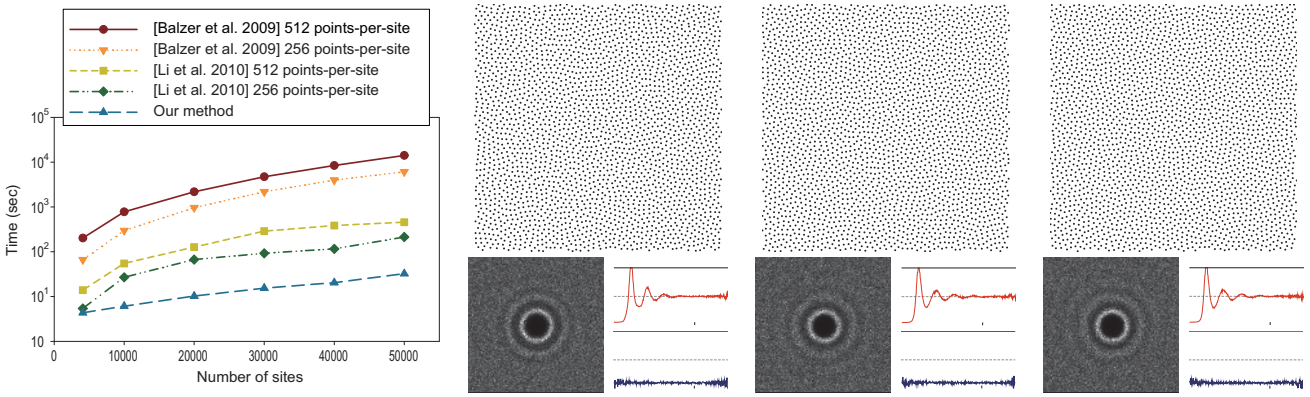
Analysis and comparisons: As shown in Algorithm 1, computing the Voronoi tessellation is the most time-consuming part of our method. Let n be the number of sites. We use CGAL [37] to compute the Delaunay triangulation first which costs $O(n \log n)$ time, and then obtain the Voronoi tessellation as the dual of Delaunay triangulation. Hence, the time complexity per iteration of our method is $O(n \log n)$. Balzer et al. [5] implement a variant of Lloyd's method in a discrete domain Ω which is represented by a set of points P. Let m be the number of points of P. Then the time complexity per iteration of [5] is $O(mn \log \frac{m}{n})$. The accelerated implementation in [19] of the same method brings down the time complexity to $O(mn)$.



(a) $\lambda = 1$ (b) $\lambda = 10$ (c) $\lambda = 20$

(d) $\lambda = 50$ (e) $\lambda = 100$ (f) Normalized radius

Fig. 4. Control of irregularity by varying λ. (a)-(e): Spectral analysis of distribution of 1,000 points obtained by CapCVT optimization. (f) Normalized radius is around 0.75 for $\lambda \in [30, 60]$.



(a) Running time in log scale (b) Our method (c) Balzer et al. 2009 (d) Li et al. 2010

Fig. 5. Comparisons of our method, the methods by Balzer et al. [5] and Li et al. [19]. The point distribution, spatial and spectral analysis of $4,096$ sites (512 points per site in discrete setting) obtained by the three methods are shown in (b), (c) and (d), respectively.

Note that m is usually much bigger than n, hence our method is expected to achieve significant acceleration.

We compare our method with [5] as well as its parallel implementation [19] in terms of sampling quality and running time, with varying number of points. The result is shown in Fig. 5. We can achieve blue noise sampling with comparable spatial and spectral properties to [5], [19]. Also, our method is more than two orders of magnitude faster than the original method proposed by [5], and is about 10 times faster than the method by [19]. All timing data is taken on a workstation with Intel Xeon 3.16GHz quad-core CPUs and 8GB memory. Both our method and the method by [5] run single-threaded, while the method by [19] runs 4-threaded.

3.4 Extension to surfaces

Our variational approach can easily be extended to surface cases. Let the input surface $S \subset \mathbb{R}^3$ be a triangular mesh surface with a set of triangles $\{\tau_k \mid k = 1, \cdots, m\}$.

Variational formulation on surface: The Voronoi diagram can be naturally defined on a surface using geodesic distance, which results in the so-called geodesic Voronoi diagram. However, it is difficult to compute an exact geodesic Voronoi diagram, and existing approximate algorithms for geodesic Voronoi diagram computation are computationally expensive [38]. Instead, we use the restricted Voronoi diagram [39] by approximating the geodesic distance between two points with their Euclidean distance in 3D space, which can be efficiently computed by Yan et al.'s method [40].

The CapCVT energy function on a surface S can be written as

$$E_{\mathrm{CapCVT}}(\mathbf{X}) = \sum_{i=1}^{n} \int_{V_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 \mathrm{d}\sigma + \lambda \sum_{i=1}^{n} \left(\int_{V_i} \varrho(\mathbf{x}) \mathrm{d}\sigma \right)^2, \tag{6}$$

where V_i is the restriction of the 3D Voronoi cell of $\mathbf{x}_i$ on S and λ is the weight, and $\mathrm{d}\sigma$ is the area element of the surface. As in 2D domains, we set $\rho(\mathbf{x}) = \varrho^2(\mathbf{x})$ where $\varrho(\mathbf{x})$ is the desired point distribution function.

Gradient of the energy function: The gradient of $E_{\mathrm{CVT}}(\mathbf{X})$ on surface domains is the same as equation (4). Please refer to [41] for the derivation. We derive the gradient of $E_{\mathrm{CapVT}}(\mathbf{X})$ on surface domain as:

$$\frac{\partial E_{\mathrm{CapVT}}(\mathbf{X})}{\partial \mathbf{x}_i} = 2 \sum_{j \in N_i} (|V_i| - |V_j|) \sum_{\tau_k \in \mathcal{T}} \int_{V_{ij} \cap \tau_k} \frac{\varrho(\mathbf{x})(\mathbf{x} - \mathbf{x}_i)}{\|\mathbf{x}_j - \mathbf{x}_i\|_{\tau_k}} \mathrm{d}s,$$

where $\mathcal{T}$ is the set of facets of S intersecting with V_{ij} (the common face of V_i and V_j), and $\|\mathbf{x}\|_{\tau_k}$ is the length of the projection of the vector $\mathbf{x}$ onto the plane τ_k. Its proof is given in Section 3.5. Thus we have the explicit computation of the gradient of $E_{\mathrm{CapCVT}}(\mathbf{X})$. As the sites have to be on the surface S, the gradient must be computed within the tangent space of S as

$$\left.\frac{\partial E_{\mathrm{CapCVT}}}{\partial \mathbf{x}_i}\right|_S = \frac{\partial E_{\mathrm{CapCVT}}}{\partial \mathbf{x}_i} - \left[\frac{\partial E_{\mathrm{CapCVT}}}{\partial \mathbf{x}_i} \cdot \mathbf{N}(\mathbf{x}_i) \right] \mathbf{N}(\mathbf{x}_i),$$

where $\mathbf{N}(\mathbf{x}_i)$ is the normal of the triangle containing $\mathbf{x}_i$.

Algorithm: The algorithm for computing a point distribution on a surface S by minimizing E_{CapCVT} is given in Algorithm 2.

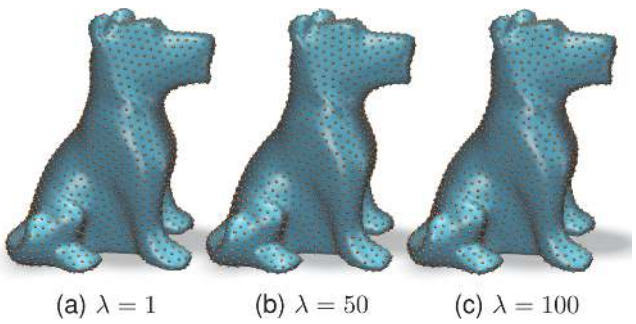
Algorithm 2 differs from Algorithm 1 in a few steps. In Step 5, we compute the restricted Voronoi tessellation of $\mathbf{X}^{(k)}$ on S which can be efficiently computed by the method presented in [40]. In Step 6, we use the gradient of E_{CapCVT} within the tangent space of S. In Step 9, we have to project the computed points onto S to guarantee that the resulting points lie on the surface.

Results and comparisons: Fig. 6 shows the results of blue noise sampling on a dog surface with different λ values. It can be seen that more irregularities are introduced to the point distribution as λ increases. In Fig. 7, we show an adaptive sampling obtained by minimizing the CapCVT function on a cat surface by the L-BFGS method. The density function is defined according to the curvature of the surface, so that regions of higher curvature are of higher density distribution. We compare our CapCVT method with other methods by the spectral analysis method proposed by Bowers et al. [28] and the results are shown in Fig. 8. Our method behaves similarly on surfaces

Algorithm 2 Variational CapCVT framework on surface

Input: an initial point set $\mathbf{X}$ on S and a weight λ
Output: a new point distribution $\mathbf{X}$ on S

1: $\mathbf{X}^{(0)} \leftarrow \mathbf{X}$
2: $\Delta E \leftarrow E_{\mathrm{CapCVT}}(\mathbf{X}^{(0)})$
3: $k \leftarrow 0$
4: **while** $\Delta E / E_{\mathrm{CapCVT}}(\mathbf{X}^{(k)}) > 10^{-5}$ **do**
5: Compute the restricted Voronoi tessellation of $\mathbf{X}^{(k)}$ on S
6: Compute the gradient $\nabla E_{\mathrm{CapCVT}}|_S$
7: Compute $\Delta \mathbf{X}$ using the **L-BFGS** updating rule
8: $\mathbf{X}^{(k+1)} \leftarrow \mathbf{X}^{(k)} + \Delta \mathbf{X}$
9: Project $\mathbf{X}^{(k+1)}$ onto surface S, denoted still by $\mathbf{X}^{(k+1)}$
10: $\Delta E \leftarrow E_{\mathrm{CapCVT}}(\mathbf{X}^{(k+1)}) - E_{\mathrm{CapCVT}}(\mathbf{X}^{(k)})$
11: $k \leftarrow k + 1$
12: **end while**
13: **return** $\mathbf{X}^{(k)}$



(a) $\lambda = 1$ (b) $\lambda = 50$ (c) $\lambda = 100$

Fig. 6. Sampling on a surface by the CapCVT method with different λ. It takes about 16 seconds to obtain a distribution with 2,000 samples.

to its counterpart on 2D domains. From the spectral analysis, we suggest that $\lambda \in [20, 100]$ is suitable for obtaining blue noise sampling on surfaces with our framework.

3.5 Properties of the CapCVT energy

In this section, we give the derivation of the gradient of the CapVT energy as well as several other properties regarding the CapCVT energy.

Gradient of $E_{\mathrm{CapVT}}(\mathbf{X})$: We derive the computation of the gradient of the CapVT energy function $E_{\mathrm{CapVT}}(\mathbf{X})$ on surface domains (with 2D domains as special cases) in this section. First, we introduce the generalized Leibniz rule [43], also called the Leibniz-Reynolds transport theorem, concerning the differentiation under the integral sign.

Theorem 1 (Leibniz-Reynolds Transport Theorem): Suppose D_t is a 2D domain changing smoothly with time t. We are given a smooth function $g(\mathbf{x}, t), \mathbf{x} \in D_t$. Denote $\mathbf{v} = \partial \mathbf{x} / \partial t$ as the velocity vector at a boundary point $\mathbf{x}$ of D_t and denote $\mathbf{b}$ as the outward unit

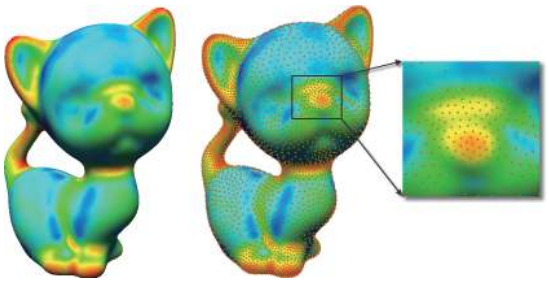


Fig. 7. Adaptive blue noise sampling on surfaces. Left: density function defined according to the curvature. The final Voronoi tessellation (middle) and the samples (right).

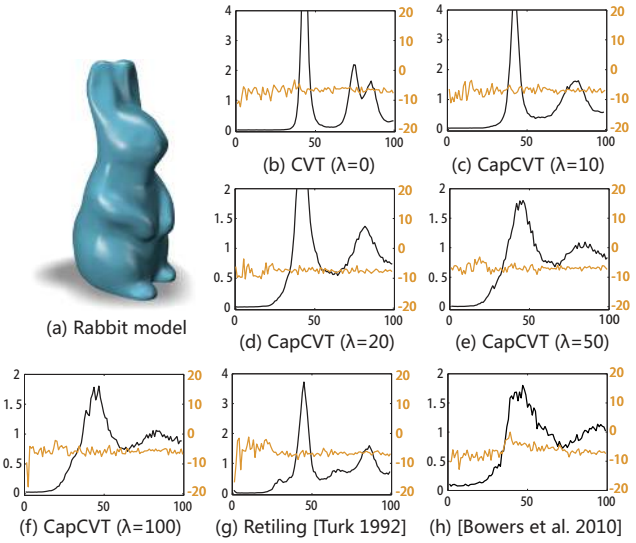


Fig. 8. Spectral comparison of the samples by our CapCVT method using different λ's, the CVT method (which is equivalent to our method with $\lambda = 0$), the retiling method [42], and Bowers et al.'s method [28]. The radial means (dark) and anisotropy (orange) are generated using the method proposed by [28].

normal at the boundary. Then we have

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{D_t} g(\mathbf{x},t)\mathrm{d}\sigma = \int_{\partial D_t} g(\mathbf{x},t)\mathbf{v}\cdot\mathbf{b}\mathrm{d}s + \int_{D_t}\frac{\partial g(\mathbf{x},t)}{\partial t}\mathrm{d}\sigma, \tag{7}$$

where $\mathrm{d}s$ is the element of arc length on the closed boundary curve ∂D_t.

Note that (7) also holds for a 3D orientable surface domain, whose closed boundary curve ∂D_t changes smoothly with t. In this case, $\mathbf{b}$ is the binormal to the space curve ∂D_t.

Theorem 2: The gradient of $E_{\mathrm{CapVT}}(\mathbf{X})$ on surface domains is

$$\frac{\partial E_{\mathrm{CapVT}}(\mathbf{X})}{\partial \mathbf{x}_i} = 2\sum_{j\in N_i}(|V_i|-|V_j|)\sum_{\tau_k\in\mathcal{T}}\int_{V_{ij}\cap\tau_k}\frac{\varrho(\mathbf{x})(\mathbf{x}-\mathbf{x}_i)}{\|\mathbf{x}_j-\mathbf{x}_i\|_{\tau_k}}\mathrm{d}s,$$

where $V_{ij} = V_i \cap V_j$ is the common face of Voronoi

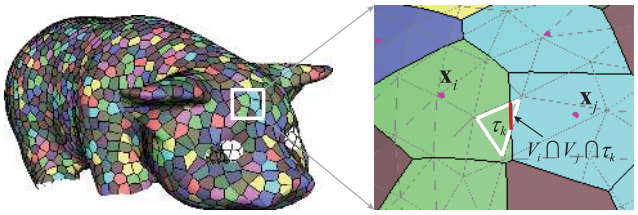


Fig. 9. Computation of the gradient of CapVT function on surface.

cells V_i and V_j, $\mathcal{T}$ is the set of triangles intersecting with V_{ij}, and $\|\mathbf{x}\|_{\tau_k}$ is the length of the projection of the vector $\mathbf{x}$ onto the plane τ_k, as shown in Fig. 9.

Proof: Assume that we apply a sufficiently small perturbation to the location of site $\mathbf{x}_i$, then only the shapes of the Voronoi region V_i and its adjacent Voronoi regions $\{V_j \mid j \in N_i\}$ will change due to the movement of site $\mathbf{x}_i$. Let the coordinates of $\mathbf{x}_i$ be $(x_{i_1}, x_{i_2}, x_{i_3})$, then

$$\frac{\partial E_{\mathrm{CapVT}}(\mathbf{X})}{\partial x_{i_l}} = 2|V_i|\frac{\partial |V_i|}{\partial x_{i_l}} + \sum_{j\in N_i} 2|V_j|\frac{\partial |V_j|}{\partial x_{i_l}}, l = 1,2,3. \tag{8}$$

By Theorem 1, we have

$$\begin{cases}\dfrac{\partial |V_i|}{\partial x_{i_l}} = \displaystyle\sum_{j\in N_i}\int_{V_{ij}}\varrho(\mathbf{x})\mathbf{v}\cdot\mathbf{b}\,\mathrm{d}s,\\ \dfrac{\partial |V_j|}{\partial x_{i_l}} = -\displaystyle\int_{V_{ij}}\varrho(\mathbf{x})\mathbf{v}\cdot\mathbf{b}\,\mathrm{d}s, \quad j\in N_i,\end{cases}$$

where $\mathbf{v} = \partial\mathbf{x}/\partial x_{i_l}$ and $\mathbf{b}$ is the outward unit binormal at boundary point $\mathbf{x}$ of the region V_i. Thus (8) can be written as

$$\frac{\partial E_{\mathrm{CapVT}}(\mathbf{X})}{\partial x_{i_l}} = \sum_{j\in N_i} 2(|V_i|-|V_j|)\int_{V_{ij}}\varrho(\mathbf{x})\mathbf{v}\cdot\mathbf{b}\,\mathrm{d}s. \tag{9}$$

We still have to compute $\mathbf{v}\cdot\mathbf{b}$ in the above equation. The common face V_{ij} of V_i and V_j is the intersection of the perpendicular bisecting plane of the segment $\overline{\mathbf{x}_i\mathbf{x}_j}$ and the surface S. Consider a segment of V_{ij} on triangle τ_k, which satisfies the equations

$$\begin{cases}(\mathbf{x}-\dfrac{\mathbf{x}_i+\mathbf{x}_j}{2})\cdot(\mathbf{x}_j-\mathbf{x}_i)=0,\\ (\mathbf{x}-\mathbf{p}_{\tau_k})\cdot\mathbf{n}_{\tau_k}=0,\end{cases}$$

where $\mathbf{n}_{\tau_k}$ is the normal of τ_k and $\mathbf{p}_{\tau_k}$ is a vertex of τ_k. By differentiating the above equations with respect to x_{i_l}, we get

$$\begin{cases}\dfrac{\partial\mathbf{x}}{\partial x_{i_l}}(\mathbf{x}_j-\mathbf{x}_i) = (\mathbf{x}-\mathbf{x}_i)\cdot\mathbf{e}_l,\\ \dfrac{\partial\mathbf{x}}{\partial x_{i_l}}\mathbf{n}_{\tau_k}=0,\end{cases}$$

where $\mathbf{e}_l$ is a 3-D vector with l-th element being 1 and other elements being 0. The unit binormal $\mathbf{b}$ is given by

$$\mathbf{b} = \frac{(\mathbf{x}_j-\mathbf{x}_i)-((\mathbf{x}_j-\mathbf{x}_i)\cdot\mathbf{n}_{\tau_k})\mathbf{n}_{\tau_k}}{\|(\mathbf{x}_j-\mathbf{x}_i)-((\mathbf{x}_j-\mathbf{x}_i)\cdot\mathbf{n}_{\tau_k})\mathbf{n}_{\tau_k}\|}.$$

Thus, we have

$$\mathbf{v}\cdot\mathbf{b}=\frac{(\mathbf{x}-\mathbf{x}_i)\cdot\mathbf{e}_k}{\|\mathbf{x}_j-\mathbf{x}_i\|_{\tau_k}}.$$

Substituting this into (9) completes the proof. □

The gradient of $E_{\mathrm{CapVT}}(\mathbf{X})$ in 2D domain can be directly obtained by replacing the triangles τ_k with a constant plane $\mathbf{x}\cdot\mathbf{e}_3=0$ in Theorem 2, which then leads to the following theorem:

Theorem 3: The gradient of $E_{\mathrm{CapVT}}(\mathbf{X})$ on 2D domains is

$$\frac{\partial E_{\mathrm{CapVT}}(\mathbf{X})}{\partial\mathbf{x}_i}=2\sum_{j\in N_i}\frac{|V_i|-|V_j|}{\|\mathbf{x}_j-\mathbf{x}_i\|}\int_{V_{ij}}\varrho(\mathbf{x})(\mathbf{x}-\mathbf{x}_i)\mathrm{d}s,$$

where N_i is the set of indices of Voronoi cells adjacent to V_i, and $V_{ij}=\partial V_i\cap\partial V_j$ is the common face of Voronoi cells V_i and V_j.

Relationship between $\rho(\mathbf{x})$ and $\varphi(\mathbf{x})$: By Gersho's conjecture [3], the energy of each site in a CVT is equal asymptotically. Then, we have $\int_{V_i}\rho(\mathbf{x})\|\mathbf{x}\|^2\mathrm{d}\sigma = \int_{V_j}\rho(\mathbf{x})\|\mathbf{x}\|^2\mathrm{d}\sigma$, for all $i\neq j$. Here we can assume $\rho(\mathbf{x})$ is constant inside a Voronoi cell, and hence $\rho(\mathbf{x}_i)\int_{V_i}\|\mathbf{x}\|^2\mathrm{d}\sigma = \rho(\mathbf{x}_j)\int_{V_j}\|\mathbf{x}\|^2\mathrm{d}\sigma$.

Denote s as the diameter of a Voronoi cell. Then $\int_{V_i}\|\mathbf{x}\|^2\mathrm{d}\sigma\propto s^4$. To keep the sites distributed in accordance with a given density φ, we need the area of a Voronoi cell being inversely proportional to φ, that is, $area(V_i)\propto 1/\varphi$. Since $area(V_i)\propto s^2$, it follows that $\int_{V_i}\|\mathbf{x}\|^2\mathrm{d}\sigma\propto(1/\varphi)^2$. We also have density being inversely proportional to $\int_{V_i}\|\mathbf{x}\|^2\mathrm{d}\sigma$. Therefore, we have $\rho\propto\varphi^2$.

Ratio of $E_{\mathrm{CVT}}(\mathbf{X})$ and $E_{\mathrm{CapVT}}(\mathbf{X})$: Assume that each Voronoi region is a regular hexagon with edge length r and the density function $\varrho(\mathbf{x})$ is constant for each Voronoi region. The CVT energy and CapVT energy in V_i are given by

$$\begin{cases}E_{\mathrm{CVT}}(V_i)=\displaystyle\int_{V_i}\varrho^2\|\mathbf{x}-\mathbf{x}_i\|^2\mathrm{d}\sigma=\frac{5\sqrt{3}}{8}r^4\varrho^2,\\ E_{\mathrm{CapVT}}(V_i)=\displaystyle\left(\int_{V_i}\varrho\,\mathrm{d}\sigma\right)^2=\frac{27}{4}r^4\varrho^2.\end{cases}$$

Thus, we get

$$\frac{E_{\mathrm{CVT}}(\mathbf{X})}{E_{\mathrm{CapVT}}(\mathbf{X})}=\frac{5\sqrt{3}}{54}\approx\frac{1}{6}.$$

Normalization of $E_{\mathrm{CapCVT}}(\mathbf{X})$: When the capacity constraint of the CapCVT energy is satisfied, each Voronoi cell has the same CapVT energy. Thus we have

$$E_{\mathrm{CapVT}}(\mathbf{X})\approx\sum_{i=1}^{n}\left(\frac{|\Omega|}{n}\right)^2=\frac{|\Omega|^2}{n},$$

where $|\Omega|=\int_\Omega\varrho(\mathbf{x})\mathrm{d}\mathbf{x}$ and n is the number of the sites. Then, we have $nE_{\mathrm{CapVT}}(\mathbf{X})\propto|\Omega|^2$. Since $E_{\mathrm{CapVT}}(\mathbf{X})\propto E_{\mathrm{CVT}}(\mathbf{X})$, we have

$$nE_{\mathrm{CapCVT}}(\mathbf{X})\propto|\Omega|^2.$$

We call the above equation $nE_{\mathrm{CapCVT}}(\mathbf{X})$ the normalization of $E_{\mathrm{CapCVT}}(\mathbf{X})$, whose value is independent of the number of sites n.

4 Variational multi-class blue noise sampling

Wei [30] first introduces the notion of multi-class blue noise sampling which requires that blue noise property be observed not only by the samples in the union of all classes, but also by the samples in the individual classes simultaneously. In this regard, we devise a multi-class CapCVT energy function which accounts for the CapCVT energies of the union and the individual classes.

4.1 Variational formulation

Given N classes of points $\mathbf{X}_i$, $i=1,\ldots,N$, with $\mathbf{X}=\cup_i\mathbf{X}_i$, the multi-class CapCVT energy function of $\mathbf{X}$, $E^{\mathrm{M}}_{\mathrm{CapCVT}}(\mathbf{X})$, is defined as:

$$E^{\mathrm{M}}_{\mathrm{CapCVT}}(\mathbf{X})=\mu\cdot N\cdot E^{*}_{\mathrm{CapCVT}}(\mathbf{X})+\sum_{i=1}^{N}E^{*}_{\mathrm{CapCVT}}(\mathbf{X}_i),\tag{10}$$

where $\mu>0$ is a weight. The term $E^{*}_{\mathrm{CapCVT}}(\mathbf{X})$ is the normalized CapCVT energy function which equals $nE_{\mathrm{CapCVT}}(\mathbf{X})$, since the value of $E_{\mathrm{CapCVT}}(\mathbf{X})$ is inversely proportional to n, where n is the number of seeds in $\mathbf{X}$ (see Section 3.5 for the proof).

4.2 Algorithm

The multi-class CapCVT energy in (10) is simply a combination of the CapCVT energies of the union set and the individual classes. A naïve application of minimizing (10) of a multi-class point set at first glance seems to be able to optimize the point distribution of the union as well as the individual classes all at the same time. However, due to the diversified interests

Algorithm 3 Variational multi-class CapCVT framework

Input: the initial N classes of points $\mathbf{X}_i$ $(i=1,\ldots,N)$ and a weight $\mu>0$

Output: new distributions of points $\hat{\mathbf{X}}_i$ for the N classes

1: **for each** class i **do**
2: Apply Algorithm 1 (for 2D) or 2 (for surface) on $\mathbf{X}_i$ to obtain a blue noise sampling $\mathbf{X}'_i$
3: **end for**
4: $\mathbf{X}\leftarrow\cup_i\mathbf{X}'_i$
5: Apply Algorithm 1 (for 2D) or 2 (for surface) on $\mathbf{X}$, with the energy function replaced by $E^{\mathrm{M}}_{\mathrm{CapCVT}}(\mathbf{X})$ to obtain the blue noise sampling of the union as $\hat{\mathbf{X}}$ as well as that of the individual classes as $\hat{\mathbf{X}}_i$.
6: **return** $\hat{\mathbf{X}}_i$ $(i=1,\ldots,N)$

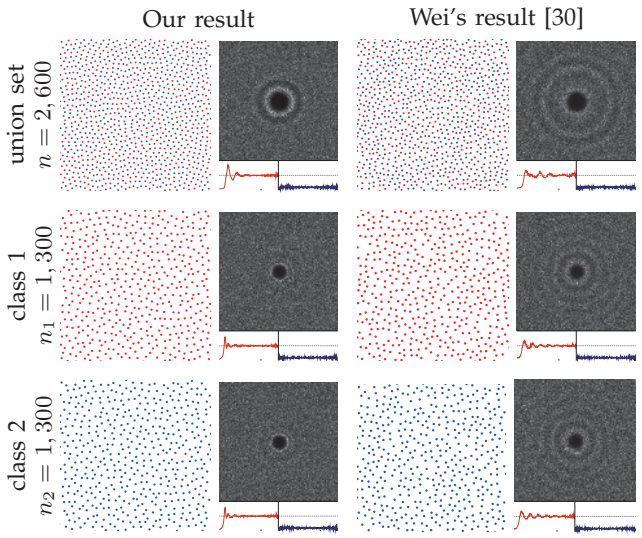


Fig. 10. Multi-class blue noise sampling by variational CapCVT with $N = 2$. All classes have the same number of samples. Our result is shown in the left column and Wei's result [30] is in the right column. Only a central portion of the 2D domain is displayed.

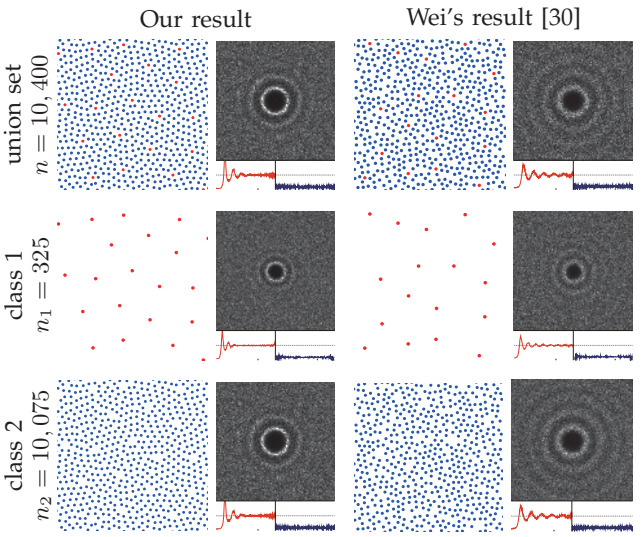


Fig. 11. Multi-class blue noise sampling by variational CapCVT with $N = 2$. Each class has different number of samples. Only a central portion of the 2D domain is displayed.

of these different considerations, the optimal gradient directions for each energy term may conflict with each other, thereby introducing a combined gradient which in general does not favor any of the sampling distributions, neither in the union set nor in the individual classes. This behavior is also confirmed by our experiments that blue noise sampling in neither the union class nor the individual classes can be accomplished by directly minimizing (10) for a multi-class point set.

Surprisingly, if we start with individual classes $\mathbf{X}_i$ of point distribution with minimized $E_{\mathrm{CapCVT}}(\mathbf{X}_i)$, (10) can now be interpreted with a different perspective. The first term of (10) serves to achieve a good point distribution with respect to the CapCVT energy for the union set, while the latter terms are responsible for preserving the optimal distributions of the individual classes. Based on this understanding, we devise a two-stage algorithm: the first stage is to obtain optimal blue noise sampling for each individual class, while the second stage is to achieve multi-class blue noise sampling by minimizing (10). The detailed steps are given in Algorithm 3.

4.3 Results and comparisons

We evaluate the quality of our multi-class sampling method on 2D domain using the same number of points in each class as in the example sets given by Wei [30]. The results of three example sets are shown in Fig. 10 and Fig. 11 (for $N = 2$) and Fig. 12 (for $N = 3$). The spatial and spectral analysis show that our multi-class sampling results are comparable to that of Wei's multi-class sampling by dart throwing [30].

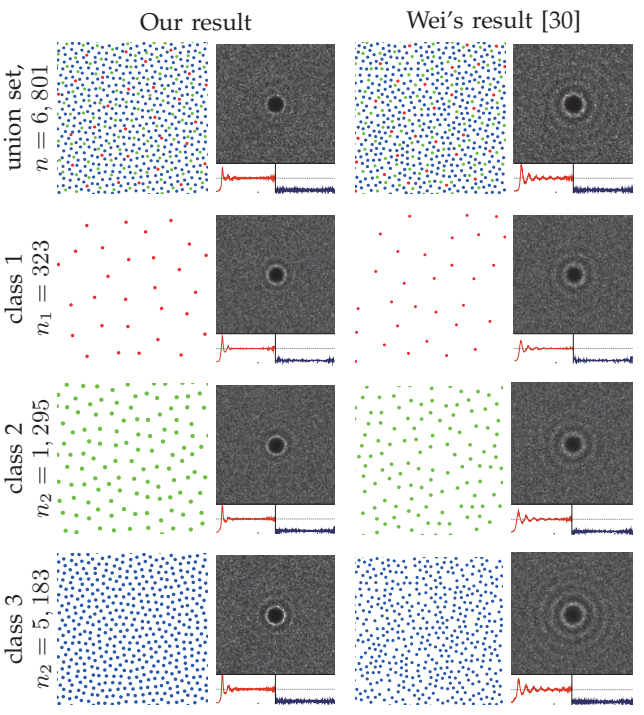


Fig. 12. Multi-class ($N = 3$) blue noise sampling by variational CapCVT. The three classes have different numbers of samples. Only a central portion of the 2D domain is displayed.

We also apply Algorithm 3 for generating multi-class blue noise samples on surfaces. Fig. 13 shows a 2-class sampling on the surface of a 3D model.

More examples of our multi-class method working on applications such as color stippling can be found in Section 5.

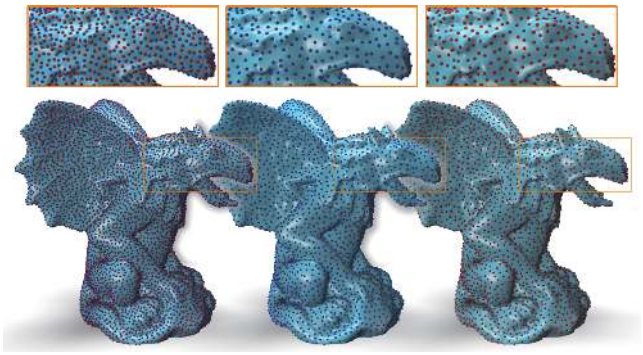


Fig. 13. Multi-class blue noise sampling on a surface. From left to right: union point distribution, point distributions for classes 1 and 2.

5 Applications

In this section, we demonstrate some applications and evaluate the performance of our variational sampling framework in different contexts.

Image stippling: Fig. 14 gives the result of our method for stippling of a gray scale image using 10,000 points. The density function is defined based on the intensity values of the input image. We first obtain an initial point distribution that adapts to the density function using error diffusion (Fig. 14(b)). Our method can then achieve a desired sampling using only 30 iterations of optimization (Fig. 14(c)). The overall time taken is about 20 seconds.

Color stippling: We apply our multi-class CapCVT method for color stippling with 7 classes of colored dots (red, green, blue, cyan, magenta, yellow and black). Fig. 15 shows our result against the multi-class sampling by dart throwing [30] on a color image. We define a particular density function for each individual class based on the color intensities of the image (see Appendix A). We aim to have more dots in the area where the corresponding color is dominant and fewer dots in the other regions. It can be seen that our method can achieve comparable results as [30] in terms of point distribution. Note that the visual quality depends also on the color decomposition (i.e., how the underlying densities for the color classes are defined), for which we adopt differently from that used in [30].

Object placement on surfaces: Object placement is also an important application in which objects are required to arrange in a visually pleasing layout without regularity artifacts. Fig. 16 shows our method working for object placement. The box surface in Fig. 16(a) is decorated with a mosaic of tiles which is the resulting Voronoi tessellation of our CapCVT method. Another example shows a texture of two objects on a surface generated using our multi-class sampling results.

Dynamic domains: For sampling in dynamic domains such as deformable surfaces, a major issue is to ensure temporal coherence of samples when the underlying domain changes against time. A sample may well represent an object which are expected to move smoothly on a surface so as to avoid flickering. Our variational framework is very suitable for sampling with temporal coherence in the dynamic domains (see Fig. 17 and the accompanying video). By taking the result of the previous frame as the initialization of the next frame, we can generate blue noise sampling in only a few iterations. For the sequence in Fig. 17, it takes 38 iterations to generate the blue noise sampling in the first frame. Only 5-10 iterations are then needed to produce each of the subsequent frames. Due to the optimization nature of our framework, the identity of each sample can be carried forward to the next time instant naturally. The sampling therefore guarantees temporal coherence to be observed over time change. This property is not possessed automatically by any stochastic blue noise sampling method, such as dart throwing, without the use of auxiliary information or processing (such as optical flow).

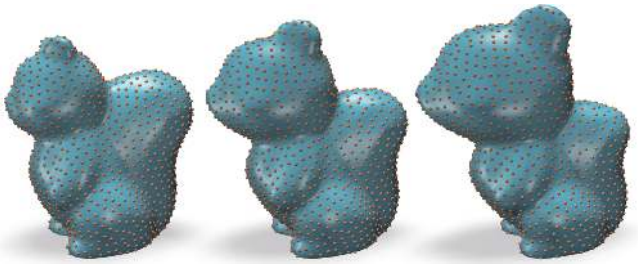


Fig. 17. Object placement on a deformable surface. Our method naturally provides temporal coherence while maintaining a desired density distribution when the domain changes continuously. See also the accompanying video.

6 Conclusion

We present a novel variational approach to generating blue noise sampling points. The energy function is a combination of the CVT energy function and the CapVT energy. Based on the derivation of the gradient of the energy function, we propose an efficient L-BFGS method to minimize the energy function. Our algorithm achieves better results and runs two orders of magnitude faster than the previous optimization-based method [5]. Our variational framework is flexible and can easily be extended to surface cases. We also develop a new method for generating multi-class blue noise sampling based on the variational framework. We demonstrate the applicability of our variational approach in various applications including image stippling, color stippling, and generating dynamic blue noise samplings in dynamic domains.

It is promising to further study this variational approach for generating blue noise sampling. Firstly, it is

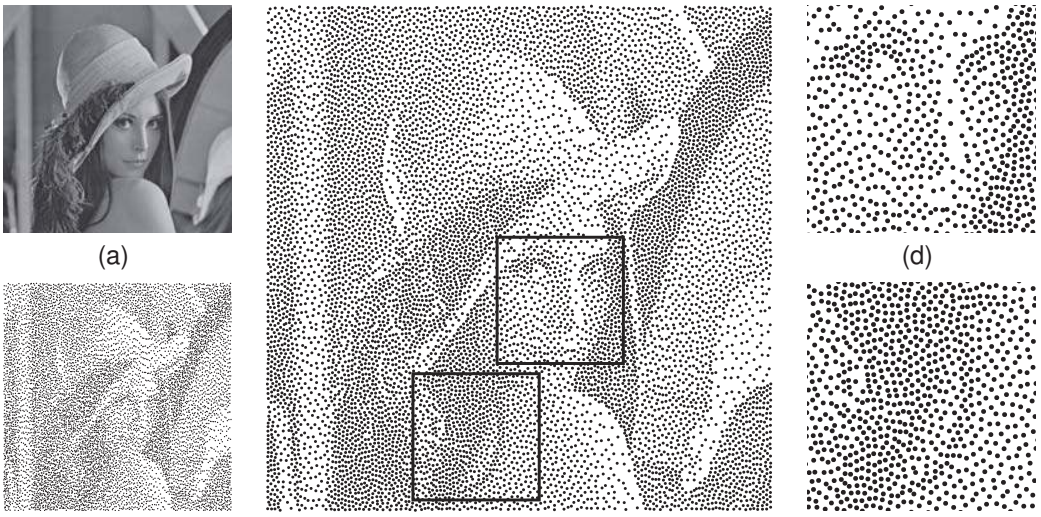


Fig. 14. Image stippling. (a) Input image as a density map; (b) initial point set obtained by error diffusion; (c) the result after 30 iterations, overall taking 22.0s; (d) & (e) zoom-in views in the hair and face regions highlighted with the boxes in (c).

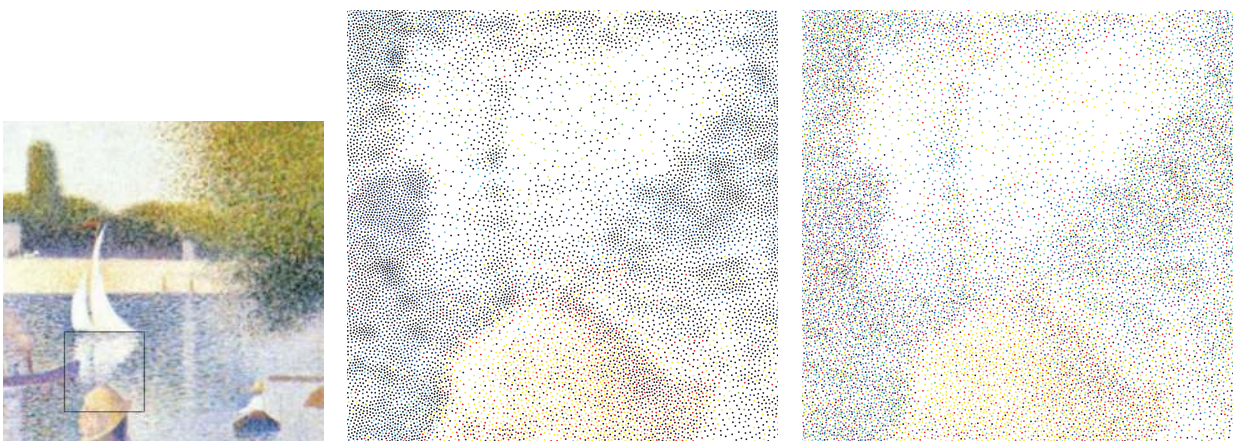


Fig. 15. Color stippling with 7 classes of color dots: red, green, blue, cyan, magenta, yellow and black. A total of 280k samples are used in the 7 classes following the color decomposition described in Appendix A. Our result using the CapCVT method in (b) is comparable to that in (c) by the dart throwing method [30] in terms of point distribution. We note here that the two methods use different color decomposition functions. (b) & (c) are zoom-in views of the region highlighted by the black box in (a). The stippling result of the entire image in vector format is provided in Appendix B.

Fig. 16. Object placement on surfaces. (a) The mosaic tile placement on surface by using the CapCVT Voronoi tessellation; (b) our multi-class blue noise samples are used to guide the placement of textures.

straightforward to extend the variational framework to blue noise sampling in 3D volumetric domain. The energy function is the same except that the domain is in 3D. Secondly, we will investigate speedup of Voronoi tessellation (or Delaunay triangulation) computation using GPU, for which there has been some related work in the literature. Thirdly, it is worthwhile to develop a variational approach to generating anisotropic blue noise sampling [44], which we feel is possible but not straightforward.

ACKNOWLEDGMENTS

The work of Wenping Wang was partially supported by the National Basic Research Program of China (2011CB302400), the Research Grant Council of Hong Kong (718209 and 718010), and the State Key Program of NSFC project (60933008). Ligang Liu was partially supported by NSFC (61070071). Zhonggui Chen was supported by NSFC (61100105, 61100107) and Natural Science Foundation of Fujian Province of China (2011J05007). The authors would like to thank Li-Yi Wei, Hongwei Li, and Rui Wang for sharing their code or results. The surface models are courtesy of Aim@Shape.

REFERENCES

[1] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[2] M. McCool and F. Eugene, "Hierarchical Poisson disk sampling distributions," in *Proceedings of the conference on Graphics interface '92*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1992, pp. 94–105.

[3] A. Gersho, "Asymptotically optimal block quantization," *IEEE Transactions on Information Theory*, vol. 25, no. 4, pp. 373–380, 1979.

[4] 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. Graph.*, vol. 28, no. 4, pp. 1–17, 2009.

[5] M. Balzer, T. Schlömer, and O. Deussen, "Capacity-constrained point distributions: a variant of Lloyd's method," in *SIGGRAPH '09: ACM SIGGRAPH 2009 papers*. New York, NY, USA: ACM, 2009, pp. 1–8.

[6] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming: Series A and B*, vol. 45, no. 3, pp. 503–528, 1989.

[7] 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.

[8] R. L. Cook, "Stochastic sampling in computer graphics," *ACM Trans. Graph.*, vol. 5, no. 1, pp. 51–72, 1986.

[9] T. R. Jones, "Efficient generation of Poisson-disk sampling patterns," *journal of graphics, gpu, and game tools*, vol. 11, no. 2, pp. 27–36, 2006.

[10] D. Dunbar and G. Humphreys, "A spatial data structure for fast Poisson-disk sample generation," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 503–508, 2006.

[11] K. B. White, D. Cline, and P. K. Egbert, "Poisson disk point sets by hierarchical dart throwing," in *Proc. IEEE Symposium on Interactive Ray Tracing*, 2007, pp. 129–132.

[12] L.-Y. Wei, "Parallel Poisson disk sampling," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 27, no. 3, pp. 20:1–20:9, 2008.

[13] M. N. Gamito and S. C. Maddock, "Accurate multidimensional Poisson-disk sampling," *ACM Trans. Graph.*, vol. 29, pp. 8:1–8:19, December 2009.

[14] M. F. Cohen, J. Shade, S. Hiller, and O. Deussen, "Wang tiles for image and texture generation," in *ACM SIGGRAPH 2003 Papers*, ser. SIGGRAPH '03. New York, NY, USA: ACM, 2003, pp. 287–294.

[15] A. Lagae and P. Dutré, "An alternative for Wang tiles: colored edges versus colored corners," *ACM Trans. Graph.*, vol. 25, no. 4, pp. 1442–1459, 2006.

[16] J. Kopf, D. Cohen-Or, O. Deussen, and D. Lischinski, "Recursive Wang tiles for real-time blue noise," *ACM Trans. Graph.*, vol. 25, pp. 509–518, 2006.

[17] V. Ostromoukhov, C. Donohue, and P.-M. Jodoin, "Fast hierarchical importance sampling with blue noise properties," in *ACM SIGGRAPH 2004 Papers*. New York, NY, USA: ACM, 2004, pp. 488–495.

[18] V. Ostromoukhov, "Sampling with polyominoes," *ACM Trans. Graph.*, vol. 26, pp. 78:1–78:6, Jul. 2007.

[19] H. Li, D. Nehaby, L.-Y. Wei, P. V. Sander, and C.-W. Fu, "Fast capacity constrained Voronoi tessellation," in *Proc. Symposium on Interactive 3D graphics and Games*, 2010, pp. 13:1–13:7.

[20] C. Schmaltz, P. Gwosdek, A. Bruhn, and J. Weickert, "Electrostatic halftoning," *Computer Graphics Forum*, vol. 29, no. 8, pp. 2313–2327, 2010.

[21] R. Fattal, "Blue-noise point sampling using kernel density model," *ACM Trans. Graph.*, vol. 30, pp. 48:1–48:12, Aug. 2011.

[22] T. Schlömer, D. Heck, and O. Deussen, "Farthest-point optimized point sets with maximized minimum distance," in *Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics*, ser. HPG '11. New York, NY, USA: ACM, 2011, pp. 135–142.

[23] V. Surazhsky, P. Alliez, and C. Gotsman, "Isotropic remeshing of surfaces: a local parameterization approach," in *Proceedings of 12th International Meshing Roundtable*, 2003.

[24] G. Rong, Y. Liu, W. Wang, X. Yin, X. D. Gu, and X. Guo, "Gpu-assisted computation of centroidal Voronoi tessellation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, pp. 345–356, 2011.

[25] H. Li, K.-Y. Lo, M.-K. Leung, and C.-W. Fu, "Dual Poisson-disk tiling: An efficient method for distributing features on arbitrary surfaces," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 5, pp. 982–998, 2008.

[26] Y. Fu and B. Zhou, "Direct sampling on surfaces for high quality remeshing," *Computer Aided Geometric Design*, vol. 26, no. 6, pp. 711–723, 2009.

[27] D. Cline, S. Jeschke, K. White, A. Razdan, and P. Wonka, "Dart throwing on surfaces," *Computer Graphics Forum*, vol. 24, no. 4, p. 1217C1226, 2009.

[28] J. Bowers, R. Wang, L.-Y. Wei, and D. Maletz, "Parallel Poisson disk sampling with spectrum analysis on surfaces," *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, vol. 29, no. 6, pp. 166:1–166:12, 2010.

[29] A. Öztireli, M. Alexa, and M. Gross, "Spectral sampling of manifolds," *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, vol. 29, no. 6, pp. 168:1–168:8, 2010.

[30] L.-Y. Wei, "Multi-class blue noise sampling," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 29, no. 4, pp. 79:1–79:8, 2010.

[31] C. Schmaltz, P. Gwosdek, and J. Weickert, "Multi-class electrostatic halftoning," Department of Mathematics, Saarland University, Saarbrcken, Germany, Tech. Rep., Oct. 2011.

[32] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi tessellations: applications and algorithms," *SIAM Review*, vol. 41, pp. 637–676, 1999.

[33] M. Balzer, "Capacity-constrained Voronoi diagrams in continuous spaces," in *International Symposium on Voronoi Diagrams in Science and Engineering*. Los Alamitos, CA, USA: IEEE Computer Society, 2009, pp. 79–88.

[34] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, January 1965.

[35] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. Springer, 2006.

[36] T. Schlömer and O. Deussen, "Towards a standardized spectral analysis of point sets with applications in graphics," University of Konstanz, Tech. Rep., 2010.

[37] "CGAL, Computational Geometry Algorithms Library," http://www.cgal.org.

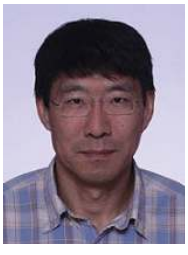
[38] G. Peyré and L. D. Cohen, "Geodesic remeshing using front propagation," *Int. J. Comput. Vision*, vol. 69, pp. 145–156, August 2006.
[39] H. Edelsbrunner and N. R. Shah, "Triangulating topological spaces," *International Journal of Computational Geometry and Applications*, vol. 7, no. 4, pp. 365–378, 1997.
[40] D.-M. Yan, B. Lévy, Y. Liu, F. Sun, and W. Wang, "Isotropic remeshing with fast and exact computation of restricted Voronoi diagram," in *Proc. of Symposium on Geometry Processing*, 2009, pp. 1445–1454.
[41] Q. Du, M. D. Gunzburger, and L. Ju, "Constrained centroidal Voronoi tessellations for surfaces," *SIAM J. Sci. Comput.*, vol. 24, pp. 1488–1506, May 2002.
[42] G. Turk, "Re-tiling polygonal surfaces," in *Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH '92. New York, NY, USA: ACM, 1992, pp. 55–64.
[43] H. Flanders, "Differentiation under the integral sign," *The American Mathematical Monthly*, vol. 80, no. 6, pp. 615–627, 1973.
[44] H. Li, L.-Y. Wei, P. V. Sander, and C.-W. Fu, "Anisotropic blue noise sampling," *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, vol. 29, no. 6, pp. 167:1–167:10, 2010.

Zhonggui Chen is an assistant professor in Department of Computer Sciences, School of Information Science and Technology, Xiamen University, China. He received his BSc and PhD degrees in Applied Mathematics from Zhejiang University in 2004 and 2009 respectively. His research interests include computer graphics and computational geometry.

Zhan Yuan received his B.Eng. degree in computer science and technology from Shandong University. He is a Ph.D. student in the Computer Graphics Group headed by Professor Wenping Wang at The University of Hong Kong. His research interests include computer graphics, geometric computing, visualization and GPGPU.

Yi-King Choi is a research assistant professor in the Department of Computer Science at The University of Hong Kong. She received her B.Sc. (1996), M.Phil. (2000) and Ph.D. (2008) in computer science from The University of Hong Kong, China. Her research interests include computer graphics, geometric computing, and visualization.

Ligang Liu is a professor at the School of Mathematical Sciences, University of Science and Technology of China. He received a Bachelor degree and a PhD degree in applied mathematics from Zhejiang University in 1996 and 2001 respectively. He worked at Microsoft Research Asia during 2001 and 2004 and worked at Department of Mathematics, Zhejiang University during 2004 and 2012. For more information, please visit http://staff.ustc.edu.cn/∼lgliu.



Wenping Wang is Professor of Computer Science at The University of Hong Kong. His research covers computer graphics, visualization, and geometric computing. He got his B.Sc. (1983) and M.Eng. (1986) at Shandong University, China, and Ph.D. (1992) at University of Alberta, Canada, all in Computer Science. He is currently Associate Editor of the Springer journal Computer Aided Geometric Design and IEEE Transactions on Visualization and Computer Graphics. He is program co-chair of several international conferences, including Geometric Modeling and Processing (GMP 2000), Pacific Graphics 2003, ACM Symposium on Physical and Solid Modeling (SPM 2006), and International Conference on Shape Modeling (SMI 2009). For more information, please visit http://www.cs.hku.hk/∼wenping.