STAG: Smart Tools and Applications in Graphics (2020)
S. Biasotti, R. Pintus and S. Berretti (Editors)

Kernel-Based Sampling of Arbitrary Data

S. Cammarasana[1], and G. Patanè[1]

[1]Istituto di Matematica Applicata e Tecnologie Informatiche 'E. Magenes' - CNR, Italy

Abstract

Point sampling is widely used in several Computer Graphics' applications, such as point-based modelling and rendering, image and geometric processing. Starting from the kernel-based sampling of signals defined on a regular grid, which generates adaptive distributions of samples with blue-noise property, we specialise this sampling to arbitrary data in terms of dimension and structure, such as signals, vector fields, curves, and surfaces. To demonstrate the novelties and benefits of the proposed approach, we discuss its applications to the resampling of 2D/3D domains according to the distribution of physical quantities computed as solutions to PDEs, and to the sampling of vector fields, 2D curves and 3D point sets. According to our experiments, the proposed sampling achieves a high approximation accuracy, preserves the features of the input data, and is computationally efficient.

CCS Concepts

• ***Computing methodologies*** → ***Point-based models; Mesh models; Image processing;***

1. Introduction

Measuring data and simulating complex phenomena typically generate a large amount of unstructured and noisy scalar and/or vector data (e.g., natural events, diffusive processes, electric and electromagnetic fields), which are a challenging input for sampling and for the reconstruction of a global representation. In these examples, we typically have heterogeneous data (e.g., scalar values, vectors) in terms of structures (e.g., vector fields, scalar functions), spatial distribution or resolution (e.g., regular grids, meshes, sparse/dense samples) and dimension. Then, the design of sampling algorithms that can be applied to arbitrary data, in terms of structure, dimension, and resolution is generally difficult and time-consuming.

Point sampling is widely used in several Computer Graphics' applications, such as point-based modelling [PKKG03] and rendering [SP04], image and geometric processing [PJH16, Mit87]. Point sampling is strictly related to image half-toning [LA01], which consists in simulating the full tone range of an image through a proper pattern of dots. Further applications are image reconstruction [TV91], anti-aliasing [KH01], dithering [FH02], QR codes reconstruction [CCLM13], and artistic visualisation [OH95].

Important aspects of point sampling are the *adaptation* to the input signal in order to guarantee that the sampling density is proportional to the image density or to the complexity of the signal in a given area; *feature preservation* without either over-smoothing or artefacts in the sampling or in the reconstructed signal; *spectral properties* (e.g., blue-noise property) that allow us to achieve visually superior images, as the distribution of photoreceptors in a primate eye possesses the blue-noise characteristic [Yel83].

Overview and contribution Our starting point is the kernel-based sampling [ZH16] (Sect. 2), which approximates an input signal (e.g., a 2D or 3D image) as the sum of Gaussian kernels with a fixed support, whose centres are computed through the minimisation of an energy functional. The kernel-based sampling has been applied to signals defined on regular 2D/3D grids and generally well preserves the underlying geometries and features (Fig. 1). However, this method is typically affected by outliers or pixel changes, or when the geometries of the input image are too complex.

In this context, our overall goal is to increase the range of applicability of the kernel-based sampling to arbitrary data (e.g., signals, curves, 2D/3D point sets, vector fields) in terms of data structures (e.g., vector fields, scalar functions), spatial distribution or resolution (e.g., regular grids, meshes, dense samples), and dimension. To this end (Sect. 3), we define a *signal-based sampling* of a 2D/3D domain according to the distribution of physical quantities, computed as solutions to PDEs, and without constraints on the regularity of the input grid. In this way, the sampling density is higher in those regions where the signal is more significant and sparser where its behaviour is almost constant. For instance, if the PDE models a mechanical stress, then we expect to place a higher number of samples in those regions of the domain where the simulated stress is higher. Following an analogous approach, we define the *energy-based sampling* of a vector field that is guided by high-energy values and recovers the potential of its conservative component.

To sample a 2D curve, we apply the kernel-based sampling to the 2D image defined by evaluating a meshless approximation of the input curve on a regular 2D grid. Since the image-based curve sampling well preserves the geometry of the input points but the

DOI: 10.2312/stag.20201252

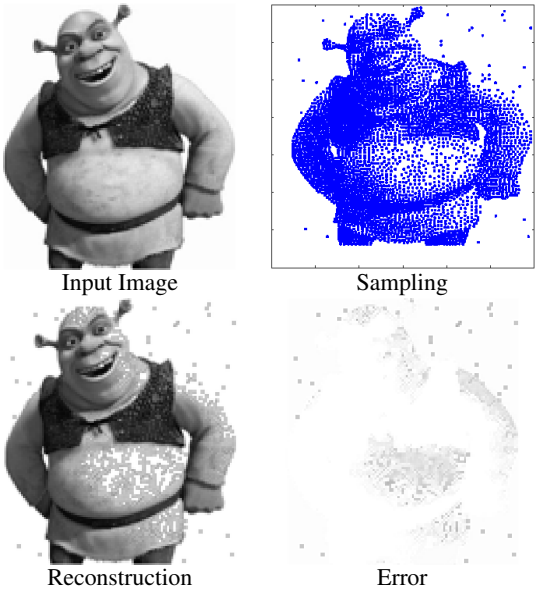


Figure 1: *Input* 256×256 *image and sampling with 5K samples, reconstruction, and reconstruction error; the white colour represents the regions where the error is lower.*

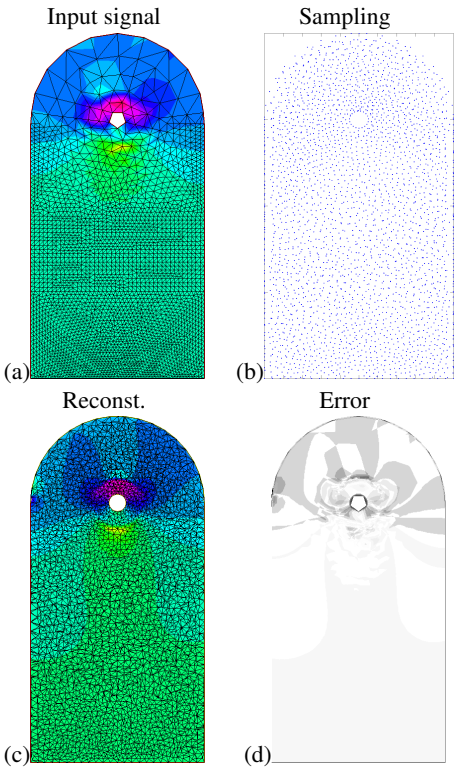


Figure 2: *(a) Input signal as solution to the Navier-Stoke equation on a 2D domain (25K input points), (b) kernel-based sampling ($n = 4K$ samples), (c) reconstructed solution recomputed on this new sampling, (d) distribution of the approximation error (white colour is associated with a lower error).*

sampling has a poor blue-noise property, we propose an isoparametric curve sampling. In this case, we maximise the minimum distance among the samples with a constraint that forces the points to belong to the underlying curve. According to our tests, the reconstructed curve accurately interpolates the input points, the samples belong to the underlying curve and are uniformly spaced along it. The sampling correctly extrapolates incomplete input data, is generally robust with respect to the input sampling density and to noise. The isoparametric curve sampling provides the best results in terms of spatial distribution of the samples and preservation of geometric features.

Finally, we generalise these results to an energy-based sampling of 2D/3D point sets through the minimisation of an energy functional, whose terms force the samples to be close to the input points and not to overlap. The resulting sampling is generally accurate and able to preserve the global and local shape features. According to our tests, the sampling results are comparable with respect to state-of-the-art methods, such as the Poisson-disk point sampling [Bri07].

2. Related work

The *kernel-based sampling* [ZH16] approximates an input signal as a linear combination of Gaussian kernel functions, whose centres (i.e., the samples) are computed through the minimisation of the energy functional

$$E(\mu) = \int_{\Omega} |C(\mathbf{x}) - C_{\text{recon}}(\mathbf{x},\mu)|^2 d\mathbf{s}, \tag{1}$$

where $C(\mathbf{x})$ is the input signal on a domain Ω of $\mathbb{R}^d$ and the *reconstructed function* is defined as

$$\begin{cases} C_{\text{recon}}(\mathbf{x},\mu) = k\sum_{j=1}^{n} G(\mathbf{x},\mu_j), \\ G(\mathbf{x},\mu_j) := \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp(-\frac{\|\mathbf{x}-\mu_j\|_2^2}{2\sigma^2}), \end{cases} \tag{2}$$

where $\mu = \{\mu_j\}_{j=1}^n$ is the set of n samples, $k = \int_\Omega C(\mathbf{x})d\mathbf{s}/n$ is a constant term, $\sigma = c_\sigma \sqrt[d]{|\Omega|/n}$ is the kernel width, c_σ is a constant term, and $|\Omega|$ is the area of Ω. The minima of $E(\cdot)$ can be computed as the roots of its partial derivatives with respect to μ, i.e.,

$$\frac{\partial E(\mu)}{\partial \mu_i} = -2\int_\Omega [C(\mathbf{x}) - k\sum_{j=1}^{n} \frac{1}{\sqrt{2\pi}\sigma^d} \exp(-\frac{\|\mathbf{x}-\mu_j\|_2^2}{2\sigma^2})] \times \frac{k(\mathbf{x}-\mu_i)}{(\sqrt{2\pi})^d \sigma^{d+2}} \exp(-\frac{\|\mathbf{x}-\mu_i\|_2^2}{2\sigma^2}) d\mathbf{s}.$$

In the discrete setting, the input signal is known at a set of points $\mathcal{P} = \{\mathbf{x}_i\}_{i=1}^m$ and the integral in Eq. (1) is discretised as a finite sum over the input points. The minimum of the discrete energy functional is computed through the iterative optimisation method L-BFGS (Limited-memory Broyden–Fletcher–Goldfarb–Shanno) [ZBLN97], which finds the roots of the derivative of the energy functional. For the L-BFGS method,

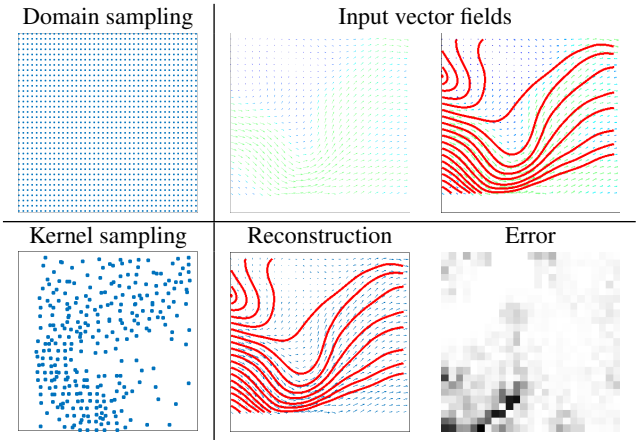


Figure 3: *Input wind flow field on a regular grid with 625 input points, $d = 2$, kernel-based sampling ($n = 120$ samples), reconstruction, and approximation error (white colour is associated with a lower error).*

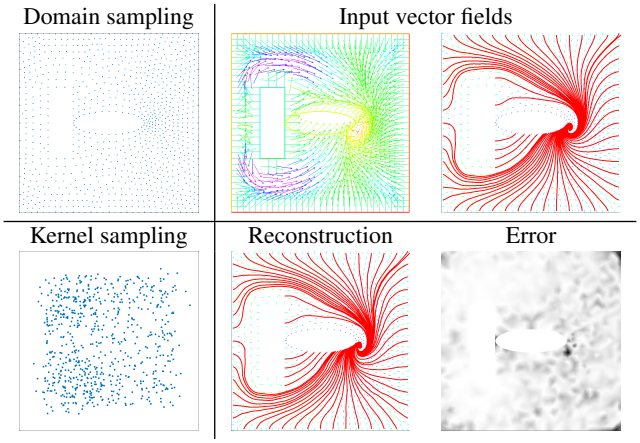


Figure 4: *Input vector field solution to the steady Navier-Stokes equation on an irregular grid with 1000 input points, kernel-based sampling ($n = 200$), reconstruction, and approximation error.*

the memory storage is $\mathcal{O}(u^2)$, and the computational cost is $\mathcal{O}(uv)$ at each iteration, where u is the total number of variables, and v is the number of steps stored in memory. The computational cost of the kernel-based sampling is $\mathcal{O}(u+m)$, where u is the number of variables, and m is the number of input points, and the memory storage is $\mathcal{O}(u^2)$.

In [Fat11], the input signal is approximated in order to generate a point set with blue-noise properties. Each kernel function has a predefined support, and a statistical model is defined to allow solutions that further reduce the minimum of the energy functional. In [ÖAG10], the optimal sampling conditions are obtained by combining spectral analysis with kernel functions, and the samples are used to reconstruct a continuous surface with the desired smoothness. In [CP99], super-resolution images are reconstructed by applying a convolution operation and assuming a similarity among correlated neighbours. In [CGW*13], a sampling method with blue-noise properties is defined by considering both spatial and non-spatial properties and modulating the samples' position with a domain-independent similarity. In [Han14], Gaussian kernels are used for approximating a continuous probability density function.

Physics-based sampling is driven by physical equations, such as electrostatics [SGBW10], compressible flow [FHHA19] or mechanical laws [Bos96], principles of natural selection for biological organisms [FT16], equilibrium in a truss structure [PS04]. *Stochastic sampling* applies a probabilistic approach to generate a sampling where points are tightly-packed, with a minimum distance constraint. Main examples include (i) dart throwing [Coo86, WCE07], which has been specialised to surface sampling [BWWM10] and image rendering [DW85], and (ii) Poisson-disk sampling [DH06]. *Lloyd's method and Voronoi tessellation* compute a centroidal Voronoi tessellation, whose vertices provide a sampling of the input domain [Llo82b, DO07, DHVOS00, BSD09, Sec02, CYC*12]. In [DHVOS00], the Lloyd's algorithm is specialised to half-toning applications. Finally, the *kernel-based image sampling* aims at optimising the approximation accuracy of the input image, through the minimisation of an energy functional, such as luminance, contrast, and structure [PQW*08], perceptually-based metrics [FP04], spatial frequency bands [PFFG98], or constraints on edge preservation [Li06] or accuracy of the reconstructed image [AA92].

Clustering (e.g., k-means clustering [Llo82a], PCA - Principal Component Analysis [Jol86]) is applied to group those points that satisfy a common "property" (e.g., planarity, closeness) and each basis function is centred at a representative point of each cluster. *Kernel-baed clustering* [CV95] (e.g., kernel PCA) evaluate the correlation among points with respect to the scalar product induced by a positive-definite kernel. *Sparsification* selects the basis functions through a basis pursuit de-noising [CDS98], standard and orthogonality matching pursuit methods [CBL89, MZ93], or regularised logistic regression [Ng04].

For the *error evaluation*, we reconstruct the input signal at any point $\mathbf{y}$ as a linear combination of the kernel functions, by computing $C_{\text{recon}}(\mathbf{y})$ in Eq. (2). We evaluate the accuracy of the reconstruction as the difference $|C_{\text{recon}}(\mathbf{x}_i) - C(\mathbf{x}_i)|$ between the input and the reconstructed signals at $\mathbf{x}_i$, $\forall i$. Given m input points, we evaluate the *normalised cross correlation*

$$NCC = \frac{\sum_{i=1}^{m}[C_{\text{recon}}(\mathbf{x}_i) - \bar{C}_{\text{recon}}][C(\mathbf{x}_i) - \bar{C}]}{\left[\sum_{i=1}^{m}[C_{\text{recon}}(\mathbf{x}_i) - \bar{C}_{\text{recon}}]^2\right]^{1/2}\left[\sum_{i=1}^{m}[C(\mathbf{x}_i) - \bar{C}]^2\right]^{1/2}},$$

where $\bar{C}_{recon}$, $\bar{C}$ are the average of the reconstructed and input signal values, and the *normalised mean square error* $NRMSE^2 = \sum_{i=1}^{m}[C_{\text{recon}}(\mathbf{x}_i) - C(\mathbf{x}_i)]^2 / \sum_{i=1}^{m}[C(\mathbf{x}_i)]^2$. We introduce the P_k*-percentile*, as the percentage $P_k = \#\{i : |C_{\text{recon}}(\mathbf{x}_i) - C(\mathbf{x}_i)| < k\}/m$ of input points whose reconstruction error is lower than k. For the comparison of the samplings $\mathcal{X}, \mathcal{Y}$, we compute the *Hausdorff distance* $d(\mathcal{X},\mathcal{Y}) := \max\{d_{\mathcal{X}}(\mathcal{Y}), d_{\mathcal{Y}}(\mathcal{X})\}$, with *one-side Hausdorff distance* $d_{\mathcal{X}}(\mathcal{Y}) := \max_{\mathbf{x}\in\mathcal{X}}\{\min_{\mathbf{y}\in\mathcal{Y}}\{\|\mathbf{x}-\mathbf{y}\|_2\}\}$.

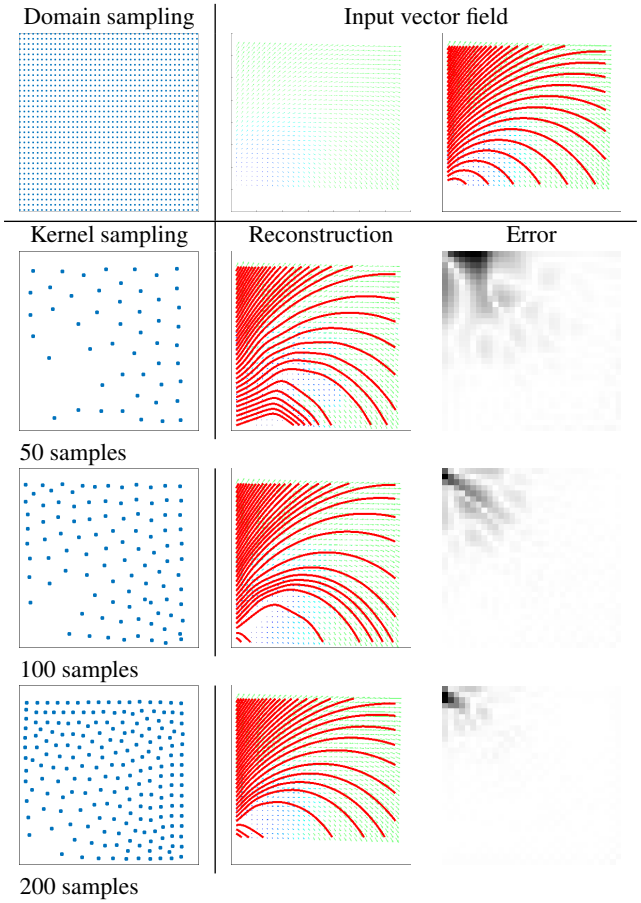


Figure 5: *Input vector field on a regular grid with 1024 input points. Energy-based sampling with an increasing number of samples, reconstruction, and approximation error.*

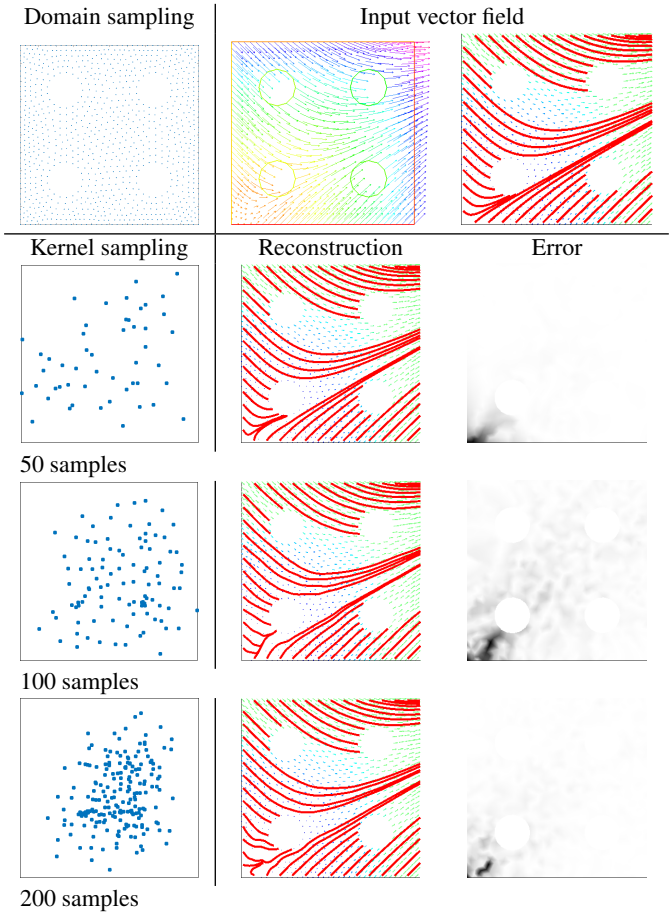


Figure 6: *Input vector field on a regular grid with 996 input points. Energy-based sampling with an increasing number of samples, reconstruction, and approximation error*

3. Proposed variants of the kernel-based sampling

We discuss novel applications of the kernel-based method to the sampling of the solution of PDEs (Sect. 3.1), vector fields (Sect. 3.2), curves and point sets (Sect. 3.3).

3.1. Signal-based sampling

The kernel-based sampling allows us to adaptively sample a domain according to the distribution of physical quantities, which are computed as solutions to PDEs and without constraints on the regularity of the input grid. In this way, we define a *signal-based sampling* of the input domain, thus improving the sampling density where the signal is more significant. For instance, if the PDE models a mechanical stress, then we expect to place a higher number of samples in those regions of the domain where the simulated stress is higher.

As test, we consider the solution $f : \Omega \rightarrow \mathbb{R}$ of the steady Navier-Stokes equation on a domain Ω (Fig. 2a), which is discretised as an unstructured grid of 25K points. The kernel-based sampling is applied to f and (Fig. 2b) places more samples in those regions of the domain where the signal intensity is higher. Reconstructing the signal (Fig. 2c) as solution to the Navier-Stokes equations on the mesh whose vertices are the computed samples, the solution accuracy is preserved in those regions where we have higher signal values, due to the density distribution of the samples. To evaluate and visualise the approximation accuracy of the reconstructed signal (Fig. 2d), we compute the difference between the solution at each point in (a) and the solution at its closest point in (c).

3.2. Kernel-based sampling of vector fields

To sample a vector field $\mathbf{v}(\mathbf{x}) : \Omega \rightarrow \mathbb{R}^d$ on an arbitrary input domain, we define the *energy-based sampling* as the kernel-based sampling of its norm $C(\mathbf{x}) := \|\mathbf{v}(\mathbf{x})\|_2$. In this way, we compute a set of samples that are then used to recover the reconstructed function in Eq. (2), which defines the conservative component $\nabla C_{\text{recon}}(\mathbf{x})$ of $\mathbf{v}(\mathbf{x})$. The resulting computational cost is equal to the kernel-based sampling (Sect. 2).

In the following, we discuss the results of the energy-based sampling of 2D vector fields defined on regular and irregular grids and

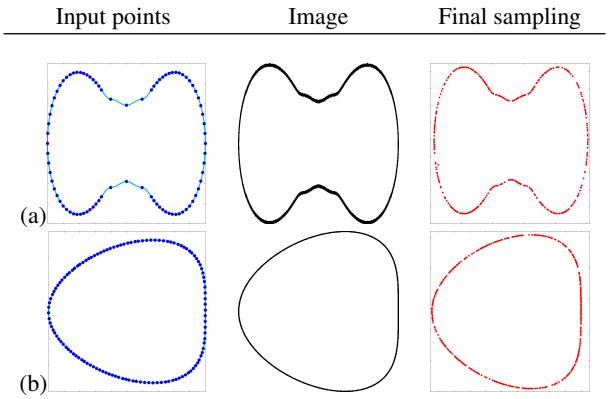


Figure 7: *Input points (blue), approximating curve (cyan), converted image (black and white), and final sampling (red) for two examples: (a) 100 input points,* 256×256 *image, 500 samples; (b) 500 input points,* 256×256 *image, 500 samples.*

analyse its performances with respect to the number of samples. For each test, we show the domain grid, and the input vector field with the related streamlines. Then, we plot the sampling, the reconstructed signal in terms of vector representation and streamlines, and the reconstruction error, which is computed as the angle between the input and the reconstructed vectors, for each point of the input domain.

Vector field on a regular grid Applying the energy-based sampling to a wind field defined on a regular grid (Fig. 3), the samples are denser in those regions where the intensity of the vector field is higher, the reconstruction is very accurate and preserves the flow lines of the vector field. The reconstruction is also more precise where the intensity of the signal is higher; for example, in the central region of the input domain. The error is generally low, and it is mainly localised at the boundary regions, where we have only a partial information on the behaviour of the input vector field. Finally, we measure the error of the reconstructed vector field as the average error on all the components of the vector field, i.e., $\overline{NCC} = \sum_{i=1}^{d} NCC_i/d$, where NCC_i is the NCC metric of the reconstruction of the i-th component of the vector field. On regular grids (Table 1), the energy-based sampling has a NCC value of 0.991 and a $P_{0.05}$ value of 97.8%.

Vector field on an irregular grid Considering a vector field computed as the solution of the steady Navier-Stokes equations on an irregular discretisation of a 2D domain (Fig. 4), the samples are denser in those regions where the intensity of the vector field is higher (e.g., around the top and bottom side of the rectangular hole), the reconstruction is very accurate and preserves the flow lines of the vector field. The error is generally low, localised at the boundary regions and at the bottom of the ellipse, due to the complexity of the flow lines. The error metrics (Table 1) show a lower accuracy than the regular grid case, with a NCC value of 0.945 and a $P_{0.05}$ value of 75.3%.

Examples Fig. 5 and Table 2 show the sampling, reconstruction, and error of the energy-based sampling applied to a vector field defined on a regular grid, when varying the number of samples. The reconstruction error is located mainly at the top-left corner. The $P_{0.05}$ value increases from 66.1% to 97.8%, when passing from 50 to 200 samples. Fig. 6 and Table 2 show the sampling, reconstruction, and error of the energy-based sampling of a vector field defined on an irregular grid, by varying the number of samples. The reconstruction error is localised mainly around the bottom-left corner; increasing the number of samples, the reduction of the reconstruction error is less evident than the regular grid case, e.g., the $P_{0.05}$ value increases from 99.0% to 99.9%, when passing from 50 to 200 samples.

Table 1: *Accuracy of the energy-based sampling applied to a vector field defined on a regular (Fig. 3) and an irregular (Fig. 4) grid.*

Test	$\overline{\text{NCC}}$	$\overline{\text{NRMSE}}$	$\overline{\text{P}_{0.05}}$	$\overline{\text{P}_{0.10}}$
Fig. 3	0.991	0.074	97.8%	100%
Fig. 4	0.945	0.32	75.3%	91.8%

Table 2: *Approximation accuracy of the energy-based sampling method, applied to a vector field defined on a regular (Fig. 5) and irregular (Fig. 6) grid, varying the number of samples.*

Samples	$\overline{\text{NCC}}$	$\overline{\text{NRMSE}}$	$\overline{\text{P}_{0.05}}$	$\overline{\text{P}_{0.10}}$
Fig. 5				
50	0.964	0.147	66.1%	87.2%
100	0.987	0.096	81.7%	95.7%
200	0.997	0.038	97.8%	99.9%
Fig. 6				
50	0.997	0.060	99.0%	100%
100	0.996	0.062	98.7%	100%
200	0.998	0.047	99.9%	100%

3.3. Point set sampling

To address the sampling of a 2D curve, we convert the input curve to an image through a meshless approximation (Sect. 3.3.1), which is improved by the isoparametric curve sampling (Sect. 3.3.2). Then, we introduce the energy-based (Sect. 3.3.3) sampling of 2D/3D point clouds.

3.3.1. Image-based curve sampling

We sample a discrete curve by applying the kernel-based sampling in Eq. (1) to a proper underlying image. More precisely, given a set of input points that represent a discrete curve, we compute the interpolating curve as the level-set $\gamma(\mathbf{x}) = \sum_j \beta_j \phi(\mathbf{x} - \mathbf{c}_j) = 0$, where the centres $\mathbf{c}_j$ are the input points and β_j are the coefficients computed through the solution of a linear system [TO99]. Once the implicit representation of the curve has been computed, we transform it to an image by sampling the implicit function γ on a regular grid and apply the kernel-based sampling to this image. Selecting compactly-supported radial basis functions, the evaluation of the curve γ is $\mathcal{O}(n \log n)$, where n is the number of input samples. Then, the computational cost of the kernel-based sampling $\mathcal{O}(2n+m)$ is determined by the dimension $m = a \times b$ is the sampling grid (e.g., 128×128, in our experiments) and by the number n of 2D sam-

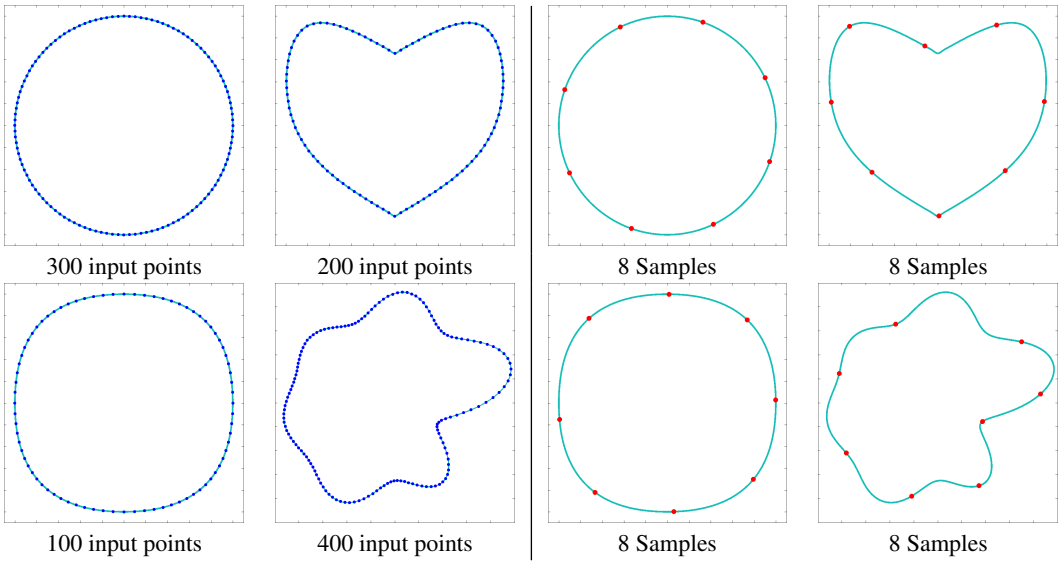


Figure 8: *Input point set (blue), approximating curve (cyan), and sampling (red).*

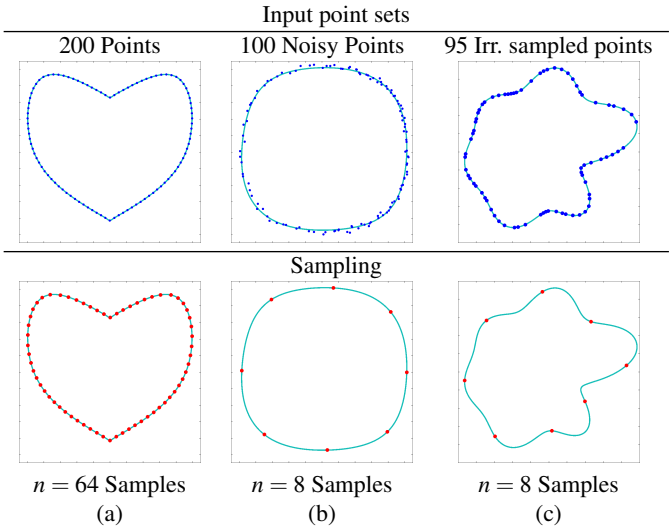


Figure 9: *Input point set (blue), approximating curve (cyan), and sampling (red).*

ples. In Fig. 7, the sampling well preserves the geometry of the input points; however, the poor blue-noise property of the sampling motivates the following definition of the isoparametric curve sampling.

3.3.2. Isoparametric curve sampling

We introduce the sampling of a 2D point set, that represents a closed curve, also satisfying the blue-noise property. Firstly, we compute the interpolating curve $\{\mathbf{x} : \gamma(\mathbf{x}) = 0\}$ (Sect. 3.3.1); then, we determine the optimal position of n samples $(\mu_i)_{i=1}^{n}$ through the solution of the following optimisation problem

$$\begin{cases} \max_{\mu_1,\dots,\mu_n} \{\min_{\substack{i,j=1,\dots,n \\ i\neq j}} \text{dist}_\gamma(\mu_i,\mu_j)\}, & (a) \\ \text{subject to} \quad \gamma(\mu_1,\dots,\mu_n) = 0, & (b) \end{cases} \tag{3}$$

where $\text{dist}_\gamma(\mu_i,\mu_j)$ is the distance along the curve γ between the samples μ_i, μ_j. In (a), we maximise the minimum distance among the unknown samples and the constraint (b) forces the points to belong to the curve. The condition $\min_{i\neq j} \text{dist}_\gamma(\mu_i,\mu_j)$ associates each sample with its closest (distinct) sample; then, the corresponding distance is maximised in order to uniformly distribute the samples along the curve. For the solution to Eq. (3), we apply a non-linear programming solver [Bro70] based on a quasi-Newton algorithm, which starts from an inital point μ_0 and iteratively computes a minimum of the constrained objective function.

According to Fig. 8, the reconstructed curve accurately interpolates the input points, and the samples computed through the proposed algorithm belong to the underlying curve, and are uniformly spaced along it. Fig. 9 shows an analysis of the sampling accuracy with respect to a different number of samples, and its robustness with respect to noise. In (a), changing the number of samples (i.e., $n = 64$) does not affect the uniform distribution of the samples. In case of noisy input data (b), the approximating curve is built by considering a small subset of the input points as centres of the approximating function. The resulting curve well interpolates the input points, and the samples are uniformly spaced. Finally, we take the input point cloud introduced in Fig. 8(bottom right) and remove half of the points; the approximating curve (c) still correctly fits the incomplete input data, and the samples are uniformly spaced along the curve.

Analysis of the initial sampling Since the initial position of the samples has a significant impact on their final position, we have tested four different initial configurations (Fig. 10), showing the initial position, the path (i.e., the movement of each sample from

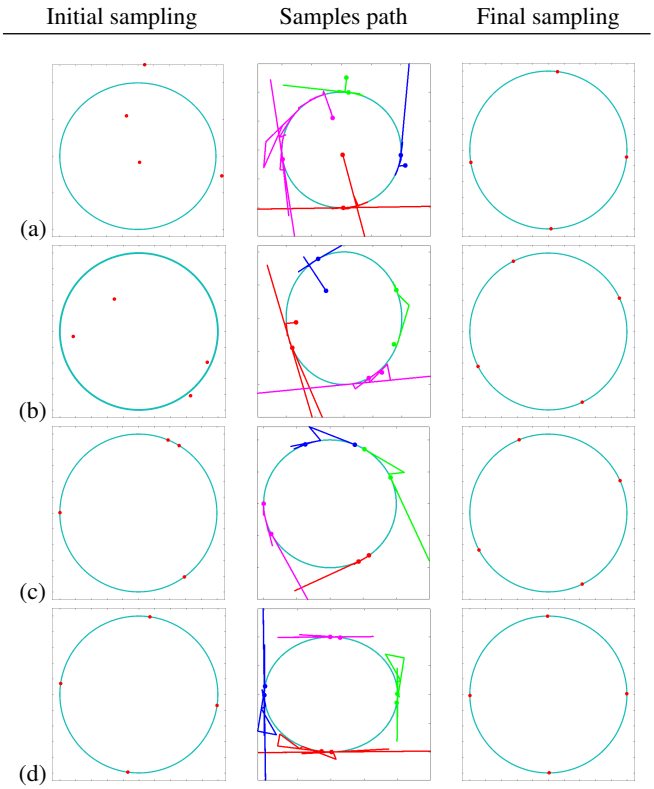


Figure 10: *Initial position of the samples used as input of the kernel-based sampling, path of the samples from the initial to the optimal position, and final sampling computed through the minimisation of the energy functional in Eq. (3).*

the initial to the final position), and the final position of the samples. In (a), the samples have been placed randomly. In (b), each new sample is proposed randomly, and it is accepted with more probability as it is closer to the approximating curve γ, and farther to its closest sample, among the ones already placed. In (c), the samples have been randomly placed along the curve γ; in (d), the samples have been placed along the curve γ, with a control on their closeness.

Table 3 reports the values of the minimum distance among the samples, with the four samples initialisation, and since we are using four samples on a circle, the ideal distance is $\pi/2$. The error is computed as the norm of the distance among the samples and the ideal distance. All the approaches give good results; the random initialisation has the higher error, while placing the samples on the curve and sufficiently spaced (d) leads to an almost perfect solution, with an error lower than 10^{-5}.

3.3.3. Energy-based sampling of 2D/3D point clouds

For the sampling of an arbitrary point cloud, the samples are computed through the minimisation of the energy functional

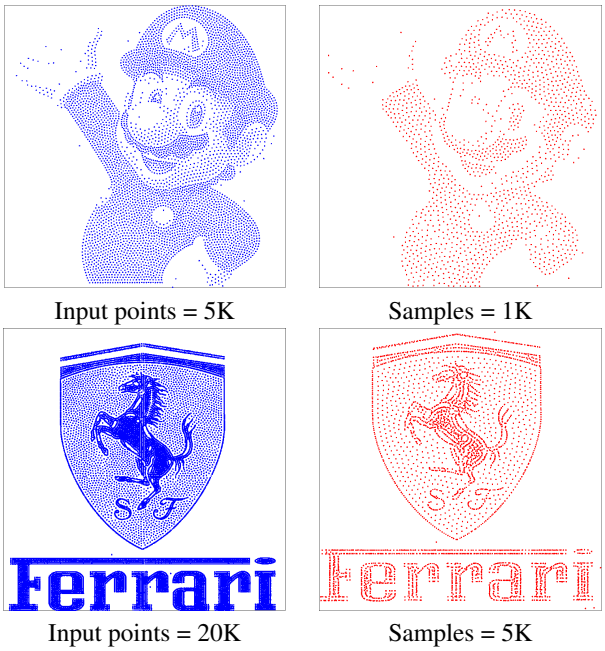


Figure 11: *Sampling of 2D point clouds. Input points (blue) and samples (red): (first row) Supermario (5K input points, 1K samples); (second row) Ferrari logo (20K input points, 5K samples).*

$E(\mu) = w_1 E_1(\mu) + w_2 E_2(\mu) + E_3(\mu)$, where

$$E_1(\mu) = \int_\Omega \sum_{i=1}^{n_1} \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp(\frac{\|\mathbf{x}-\mu_i\|_2^2}{2\sigma^2}) d\mathbf{s};$$

$$E_2(\mu) = \int_\Omega \sum_{i=1}^{n_2} \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp(\frac{\|\mathbf{x}-\mu_i\|_2^2}{2\sigma^2}) d\mathbf{s};$$

$$E_3(\mu) = \sum_{s=1}^{n} \sum_{i=1}^{n_3} \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp(-\frac{\|\mu_s-\mu_i\|_2^2}{2\sigma^2}).$$

The terms $E_1(\mu)$ and $E_2(\mu)$ force the samples to be close to the input points. $E_3(\mu)$ forces the samples not to overlap. Here, w_1 and w_2 are the weights of the components of the energy functional. The difference between the definition of $E_1(\mu)$ and $E_2(\mu)$ is in the way the neighbours are computed; in $E_1(\mu)$, we search the n_1 nearest samples for each input point, and in $E_2(\mu)$ we search the n_2 nearest input points for each sample. The derivative of the terms E_1, E_2, and E_3 are

$$\begin{cases} \frac{\partial E_h(\mu)}{\partial \mu_i} = \frac{-1}{(\sqrt{2\pi})^d \sigma^{d+2}} \int_\Omega \exp(\frac{\|\mathbf{x}-\mu_i\|_2^2}{2\sigma^2})(\mathbf{x}-\mu_i) d\mathbf{s}, \; h = 1,2, \\ \frac{\partial E_3(\mu)}{\partial \mu_i} = \frac{1}{(\sqrt{2\pi}\sigma)^d} \sum_{s=1}^{n} [\exp(-\frac{\|\mu_s-\mu_i\|_2^2}{2\sigma^2})(-2)(\mu_s-\mu_i) + \\ \qquad \sum_{\substack{h=1 \\ h\neq i}}^{n_3} \exp(-\frac{\|\mu_s-\mu_h\|_2^2}{2\sigma^2}) 2(\mu_i-\mu_h)]. \end{cases}$$

For the minimisation of the energy, we apply a non-linear programming solver [BGN00] based on a sequential quadratic programming method with an accurate estimation of the gradient and of the Hessian of the Lagrangian at each iteration.

Table 3: *With reference to Fig. 10, we report the minimum distance among the samples with four different samples initialisation.*

Initialisation	(a)	(b)	(c)	(d)
Distance	1.295	1.429	1.453	1.570
Error	0.275	0.141	0.117	$< 10^{-5}$

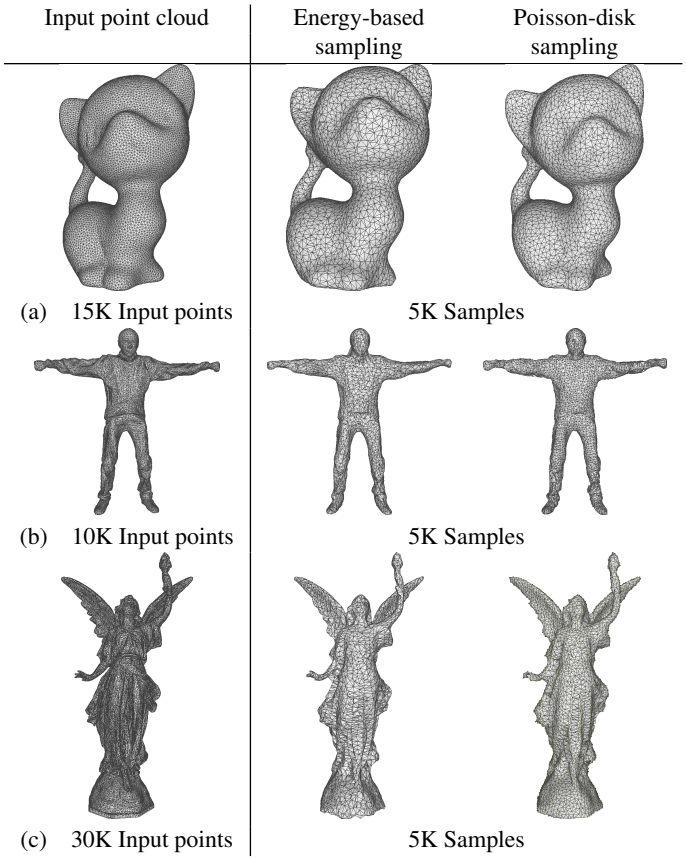


Figure 12: *Energy-based and Poisson-disk sampling of 3D point clouds: (a) kitten (15K input points, 5K samples); (b) human (10K input points, 5K samples); (c) Lucy (30K input points, 5K samples).*

Sampling 3D point sets In Fig. 11(first row), the sampling of Supermario (5K input points, 1K samples), well preserves the main features (e.g., the profiles of the body, arm and right hand) of the input point cloud. In (Fig. 11,second row), the sampling of the Ferrari logo (20K input points, 5K samples) well preserves the label and the horse shape.

In Fig. 12, we compare the sampling of 3D point clouds with the proposed energy-based sampling and the Poisson-disk sampling; the corresponding meshes are generated through the ball-pivoting algorithm [BMR*99]. We briefly recall that the Poisson-disk sampling method generates a distribution of samples of an arbitrary input domain, by complying with the blue-noise property and proposing new samples candidates only from a region near already placed samples. Our method has good results in terms of preserved features and satifies the blue-noise property; in particular, it well reconstructs the tail and the ears of the kitten, the legs of the human, and the complex geometries of Lucy's wings.

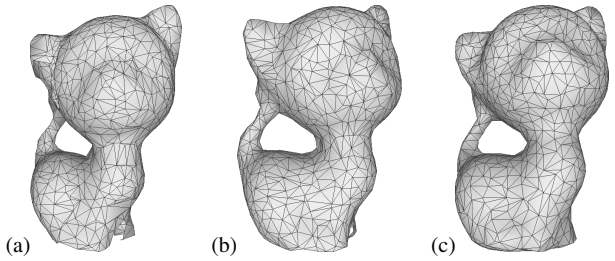


Figure 13: *With reference to Table 5, examples of the energy-based sampling: (a) 2.5K input points, 1K samples; (b) 5K input points, 1K samples; (c) 15K input points, 1K samples.*

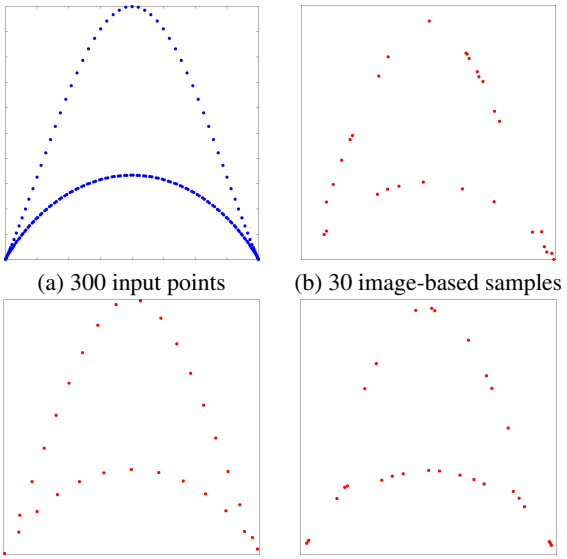
Table 4: *Hausdorff distance between three different point clouds (i.e., kitten, human, Lucy) and the respective sampling, performed with the energy-based method and the Poisson-disk method, both with 5K samples.*

Hausdorff distance	**Kitten 15K**	**Lucy 30K**	**Human 10K**
Energy-based sampling	0.015	0.035	0.010
Poisson-disk sampling	0.009	0.008	0.006

According to the values of the Hausdorff distance between the input point cloud and the energy-based and Poisson-disk sampling methods (Table 4), the accuracy of our sampling method is comparable with the results of the Poisson-disk method. Event though the Poisson-disk sampling has a lower Hausdorff distance value, due to the specificity of this method for the 3D sampling application, the energy-based sampling method is enough general to be applied to different data structures, such as 3D point clouds.

Table 5 reports the execution time of the energy-based sampling method, applied to the kitten 3D point cloud (Fig. 13), with a different number of input points and samples. Tests have been performed with Matlab R2020a, on a workstation with 2 Intel i9-9900KF CPUs (3.60GHz), and 32 GB RAM. Our method takes about 5 minutes to sample an input point cloud of 15K points, with 5K samples. The sampling well preserves the main features of the kitten, even when we reduce the number of input points (e.g., 2.5K), and the number of samples (e.g., 1K).

Comparison of samplings of point sets representing 2D curves To compare the three sampling methods, we compute the average distance between the samples and the curve [Law13] underlying the input point, and the minimum distance between each sample and its closest one (i.e., a measure of the blue-noise property). According to our tests (Fig. 14, Table 6), the isoparametric curve sampling provides the best results in terms of spatial distribution of the samples, and preservation of the input geometry. In fact, the distance between the samples and the input points is very small, while the distance between each sample and its closest one is large.



(a) 300 input points (b) 30 image-based samples

(c) 30 iso-par. samples (d) 30 energy-based samples

Figure 14: *Curve sampling with the (b) image-based, (c) isoparametric curve, and (d) energy-based samplings.*

Table 5: *Execution time of the energy-based sampling applied to the kitten 3D point cloud, with respect to a different number of input points and samples.*

Time [s]		Input points			
		500	**2.5K**	**5K**	**15K**
Samples	**200**	1.1	1.9	2.6	8.6
	1K	-	9.3	13.2	19.5
	5K	-	-	257.4	277.0

4. Conclusions and future work

In this paper, we have extended the kernel-based sampling of signals defined on regular grids to arbitrary signals in terms of dimensionality and structure, e.g., signals, vector fields, point sets, etc. The main pros of the proposed approach are the generality with respect to the input data, its simple implementation, and an approximation accuracy of the same order of state-of-the-art methods (e.g., the Poisson-disk sampling). The main limitation of our method is the higher computational cost with respect to standard sampling methods (e.g., Lloyd's method, Voronoi tessellation, clustering, Poisson-disk sampling). Indeed, as future work we plan to modify the kernel-based sampling method in order to improve the sampling and the reconstruction accuracy, also reducing its computational cost (e.g., through parallelisation) for real-time applications.

Acknowledgements We thank the Reviewers for their comments, which helped us to improve the technical and experimental parts of the article.

Table 6: *With reference to Fig. 14, we report the average distance of the samples with respect to the bicorn curve, and the minimum distance between the samples, for the three methods, with best result in bold.*

Method	Image-based sampl. (Sect. 3.3.1)	Isoparam. curve sampl. (Sect. 3.3.2)	Energy based sampl. (Sect. 3.3.3)
Curve-sample distance	0.4940	**0.0045**	0.0318
Sample-sample distance	0.0082	**0.3886**	0.0063

References

[AA92] ANALOUI M., ALLEBACH J.: New results on reconstruction of continuous-tone from halftone. In *Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing* (1992), vol. 3, pp. 313–316. 3

[BGN00] BYRD R. H., GILBERT J. C., NOCEDAL J.: A trust region method based on interior point techniques for nonlinear programming. *Mathematical programming 89*, 1 (2000), 149–185. 7

[BMR*99] BERNARDINI F., MITTLEMAN J., RUSHMEIER H., SILVA C., TAUBIN G.: The ball-pivoting algorithm for surface reconstruction. *Trans. on Visualization and Computer Graphics 5*, 4 (1999), 349–359. 8

[Bos96] BOSSEN F.: *Anisotropic Mesh Generation with Particles*. Tech. rep., Carnegie-Mellon Univ. Pittsburgh, 1996. 3

[Bri07] BRIDSON R.: Fast poisson disk sampling in arbitrary dimensions. *SIGGRAPH sketches 10* (2007), 1. 2

[Bro70] BROYDEN C. G.: The convergence of a class of double-rank minimization algorithms 1. general considerations. *IMA Journal of Applied Mathematics 6*, 1 (1970), 76–90. 6

[BSD09] BALZER M., SCHLÖMER T., DEUSSEN O.: *Capacity-constrained point distributions: a variant of Lloyd's method*, vol. 28. ACM, 2009. 3

[BWWM10] BOWERS J., WANG R., WEI L.-Y., MALETZ D.: Parallel poisson disk sampling with spectrum analysis on surfaces. In *ACM Trans. on Graphics* (2010), vol. 29, p. 166. 3

[CBL89] CHEN S., BILLINGS A., LUO W.: Orthogonal least-squares methods and their applications to non-linear system identification. *Intern. Journal of Control 50*, 5 (1989), 1873–1896. 3

[CCLM13] CHU H.-K., CHANG C.-S., LEE R.-R., MITRA N. J.: Halftone qr codes. *ACM Trans. on Graphics 32*, 6 (2013), 217. 1

[CDS98] CHEN S. S., DONOHO D. L., SAUNDERS M. A.: Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing 20*, 1 (1998), 33–61. 3

[CGW*13] CHEN J., GE X., WEI L.-Y., WANG B., WANG Y., WANG H., FEI Y., QIAN K.-L., YONG J.-H., WANG W.: Bilateral blue noise sampling. *ACM Trans. on Graphics 32*, 6 (2013), 216. 3

[Coo86] COOK R. L.: Stochastic sampling in computer graphics. *ACM Trans. on Graphics 5*, 1 (1986), 51–72. 3

[CP99] CANDOCIA F. M., PRINCIPE J. C.: Super-resolution of images based on local correlations. *IEEE Trans. on Neural Networks 10*, 2 (1999), 372–380. 3

[CV95] CORTES C., VAPNIK V.: Support-vector networks. *Machine Learning 20*, 3 (1995), 273–297. 3

[CYC*12] CHEN Z., YUAN Z., CHOI Y.-K., LIU L., WANG W.: Variational blue noise sampling. *IEEE Trans. on Visualization and Computer Graphics 18*, 10 (2012), 1784–1796. 3

[DH06] DUNBAR D., HUMPHREYS G.: A spatial data structure for fast poisson-disk sample generation. In *ACM Trans. on Graphics* (2006), vol. 25, ACM, pp. 503–508. 3

[DHVOS00] DEUSSEN O., HILLER S., VAN OVERVELD C., STROTHOTTE T.: Floating points: A method for computing stipple drawings. In *Computer Graphics Forum* (2000), vol. 19, pp. 41–50. 3

[DO07] DONOHUE C., OSTROMOUKHOV V.: Fast generation of importance-sampled point sets with associated Delaunay triangulation. In *Proc. of GRAPHICON* (2007), pp. 125–130. 3

[DW85] DIPPÉ M. A., WOLD E. H.: Antialiasing through stochastic sampling. *ACM Siggraph 19*, 3 (1985), 69–78. 3

[Fat11] FATTAL R.: Blue-noise point sampling using kernel density model. In *ACM Trans. on Graphics* (2011), vol. 30, p. 48. 3

[FH02] FRUCHTER A., HOOK R.: Drizzle: A method for the linear reconstruction of undersampled images. *Publications of the Astronomical Society of the Pacific 114*, 792 (2002), 144. 1

[FHHA19] FU L., HAN L., HU X. Y., ADAMS N. A.: An isotropic unstructured mesh generation method based on a fluid relaxation analogy. *Computer Methods in Applied Mechanics and Engineering 350* (2019), 396–431. 3

[FP04] FARRUGIA J.-P., PÉROCHE B.: A progressive rendering algorithm using an adaptive perceptually based image metric. In *Computer Graphics Forum* (2004), vol. 23, pp. 605–614. 3

[FT16] FABRITIUS B., TABOR G.: Improving the quality of finite volume meshes through genetic optimisation. *Engineering with Computers 32*, 3 (2016), 425–440. 3

[Han14] HANEBECK U. D.: Kernel-based deterministic blue-noise sampling of arbitrary probability density functions. In *Conf. on Information Sciences and Systems* (2014), pp. 1–6. 3

[Jol86] JOLLIFFE I. T.: Principal components in regression analysis. In *Principal Component Analysis*. Springer Verlag, 1986, pp. 129–155. 3

[KH01] KELLER A., HEIDRICH W.: Interleaved sampling. In *Rendering Techniques 2001*. Springer, 2001, pp. 269–276. 1

[LA01] LAU D. L., ARCE G. R.: *Modern digital halftoning*. CRC Press, 2001. 1

[Law13] LAWRENCE J. D.: *A catalog of special plane curves*. Courier Corporation, 2013. 8

[Li06] LI X.: Edge-directed error diffusion halftoning. *IEEE Signal Processing Letters 13*, 11 (2006), 688–690. 3

[Llo82a] LLOYD S.: An algorithm for vector quantizer design. *IEEE Trans. on Communications 28*, 7 (1982), 84–95. 3

[Llo82b] LLOYD S.: Least-squares quantization in PCM. *IEEE Trans. on Information Theory 28*, 2 (1982), 129–137. 3

[Mit87] MITCHELL D. P.: Generating antialiased images at low sampling densities. *ACM SIGGRAPH Computer Graphics 21*, 4 (1987), 65–72. 1

[MZ93] MALLAT S., ZHANG Z.: Matching pursuit with time-frequency dictionaries. *IEEE Trans. on Signal Processing 41*, 12 (1993), 3397–3415. 3

[Ng04] NG A. Y.: Feature selection, l_1 vs. l_2 regularization, and rotational invariance. In *ACM Proc. of the Intern. Conf. on Machine Learning* (2004), p. 78. 3

[ÖAG10] ÖZTIRELI A. C., ALEXA M., GROSS M.: Spectral sampling of manifolds. *ACM Trans. on Graphics 29*, 6 (2010), 168. 3

[OH95] OSTROMOUKHOV V., HERSCH R. D.: *Artistic screening*. Tech. rep., ACM, 1995. 1

[PFFG98] PATTANAIK S. N., FERWERDA J. A., FAIRCHILD M. D., GREENBERG D. P.: A multiscale model of adaptation and spatial vision for realistic image display. In *Int. Conf. on Computer Graphics and Interactive Techniques* (1998), pp. 287–298. 3

[PJH16] PHARR M., JAKOB W., HUMPHREYS G.: *Physically based rendering: from theory to implementation*. Morgan Kaufmann, 2016. 1

[PKKG03] PAULY M., KEISER R., KOBBELT L. P., GROSS M.: Shape modeling with point-sampled geometry. In *ACM Trans. on Graphics* (2003), vol. 22, pp. 641–650. 1

[PQW*08] PANG W.-M., QU Y., WONG T.-T., COHEN-OR D., HENG P.-A.: Structure-aware halftoning. In *ACM Trans. on Graphics* (2008), vol. 27, p. 89. 3

[PS04] PERSSON P.-O., STRANG G.: A simple mesh generator in matlab. *SIAM Review 46*, 2 (2004), 329–345. 3

[Sec02] SECORD A.: Weighted Voronoi stippling. In *Proc. of the Int. Symp. on Non-photorealistic Animation and Rendering* (2002), pp. 37–43. 3

[SGBW10] SCHMALTZ C., GWOSDEK P., BRUHN A., WEICKERT J.: Electrostatic halftoning. In *Computer Graphics Forum* (2010), vol. 29, pp. 2313–2327. 3

[SP04] SAINZ M., PAJAROLA R.: Point-based rendering techniques. *Computers & Graphics 28*, 6 (2004), 869–879. 1

[TO99] TURK G., O'BRIEN J. F.: *Variational implicit surfaces*. Tech. rep., Georgia Institute of Technology, 1999. 5

[TV91] TERZOPOULOS D., VASILESCU M.: Sampling and reconstruction with adaptive meshes. In *Conf. on Computer Vision and Pattern Recognition* (1991), pp. 70–75. 1

[WCE07] WHITE K. B., CLINE D., EGBERT P. K.: Poisson disk point sets by hierarchical dart throwing. In *Symp. on Interactive Ray Tracing* (2007), pp. 129–132. 3

[Yel83] YELLOTT J. I.: Spectral consequences of photoreceptor sampling in the rhesus retina. *Science 221*, 4608 (1983), 382–385. 1

[ZBLN97] ZHU C., BYRD R. H., LU P., NOCEDAL J.: Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Trans. on Mathematical Software 23*, 4 (1997), 550–560. 2

[ZH16] ZHONG Z., HUA J.: Kernel-based adaptive sampling for image reconstruction and meshing. *Computer-Aided Geometric Design 43* (2016), 68–81. 1, 2