Lattice-Constrained Stratified Sampling for Point Cloud Levels of Detail

Kristian L. Damkjer, *Member, IEEE*, and Hassan Foroosh, *Senior Member, IEEE*

***Abstract*— In the last several years, airborne topographic light detection and ranging (LiDAR) has emerged as an important remote-sensing technology supporting a wide variety of applications. In that time, researchers have conducted studies to determine the optimal sampling densities required to support their respective applications. This natural progression of experimentation and analysis has resulted in several recommended sampling density stratifications for LiDAR point cloud products. Recognizing the need for point clouds of varying sample densities provides at least two motivations for creating level of details (LOD) for high-resolution point clouds. First, from the perspective of LiDAR data consumers, there is a desire to use the coarsest sampling that supports the application to reduce procurement costs, storage constraints, and processing times. Second, from the perspective of LiDAR data providers, there is a desire to collect data once at the highest supported fidelity to minimize recollection costs and redundancy in data holdings. In this article, we present an approach for generating point cloud LODs by constraining samples to regular discrete lattices to optimize the coverage of the volumes represented by each sample. We compare our approach to the two most common point cloud sampling methods: random sampling and rectangular lattice sampling. We discuss two approaches for representing the point cloud LODs. Finally, we propose an extension of our sampling approach for processing single-photon and Geiger-mode avalanche photodiode (GmAPD) LiDAR.**

***Index Terms*— Light detection and ranging (LiDAR), nonuniform sampling, point cloud, sampling methods, stratified sampling.**

I. Introduction

AIRBORNE topographic light detection and ranging (LiDAR) systems provide accurate remotely sensed 3-D georeferenced data sets for use in geographic information system (GIS) applications. LiDAR data is leveraged across several domains including wide-area mapping (WAM), corridor mapping, hazard risk assessment, vegetation mapping, resource management, building modeling, change detection, and feature classification [1], [2]. Establishing sampling optimality and sensitivity for LiDAR applications is important since the procurement of LiDAR data must balance sampling density, data size, and the reliability and accuracy of the models derived from the data with collection, processing, storage, and access costs [3]. Product accuracy and sampling fidelity requirements vary significantly across the applications supported by LiDAR data. Extensive research and analysis has been conducted in the domains of terrain mapping for flood and hazard assessment [3]–[7], forestry and land management [8]–[14], and building modeling [15]–[19] to determine the optimal sampling fidelity required to support various applications within each respective domain.

Domain studies have typically evaluated the impact of sample spacing (i.e., the horizontal distance between adjacent samples) or sample density (i.e., the number of samples per unit area) on the performance of algorithms and assessment techniques. Sample spacing and sample density are considered key characteristics of point cloud data, similar to ground sample distance (GSD) and spatial resolution for photogrammetric data. Point cloud products are regularly examined to determine a nominal point spacing (NPS), represented in this article by r, or a nominal point density (NPD), represented in this article by ρ [20]. These two metrics are frequently related to each other by

$$\rho = \left(\frac{1}{r}\right)^2. \tag{1}$$

However, this is a relationship that is only valid for samples taken on a square lattice. While LiDAR collections are typically planned to be as uniform as possible, the resulting point clouds are generally unstructured and may not agree with this underlying assumption. Furthermore, NPD is typically used to characterize data sets that have an expected sampling density greater than 1 pls/m^2, whereas NPS is used otherwise [21]. Given the assumed relationship between NPS and NPD, the complementary metric is often computed rather than measured.

The following selection of studies illustrates the stratification of optimal sampling levels that has emerged in the last several years.

Raber *et al.* [4] presented a study assessing the impact of LiDAR NPS on digital elevation model (DEM) accuracy and flood zone delineation. In this article, the researchers simulated low-density LiDAR collections by using a simple count-based decimation strategy within scan lines of linear-mode LiDAR data to thin points in the cross-track direction and dropped entire scan lines in the along-track direction to achieve a relative uniformity in their thinned data sets. This approach

Manuscript received June 21, 2019; revised November 17, 2019; accepted December 10, 2019. Date of publication February 3, 2020; date of current version July 22, 2020. This article was presented in part at the ASPRS 2019 Annual Conference, Denver, CO, January 2019. *(Corresponding author: Kristian L. Damkjer.)*

Kristian L. Damkjer is with the Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816 USA, and also with L3Harris Technologies, Melbourne, FL 32904 USA (e-mail: kristian.damkjer@l3harris.com).

Hassan Foroosh is with the Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816 USA.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2020.2967880

allowed the researchers to emulate point clouds collected with NPS of 1.35–9.64 m. Perhaps, the most important finding from their study was, "the absence of … a significant pattern relating error in DEM accuracy to post-spacing through the range of post-spacing values tested." [4, p. 802] The authors further emphasized the significance of this finding by stating, "This is an important finding since it implies that more LiDAR data is not always beneficial in the flood mapping application, especially when cost is considered. Further, these findings suggest that there may be certain cases where having more data is not only redundant, but may increase error in the final product" [4, p. 802].

Magnusson *et al.* [8] presented a study assessing the robustness of forest characterization metrics derived from LiDAR data with respect to the NPS of the collection. In this article, the researchers developed a technique for thinning point clouds to simulate low-density collections by enforcing a minimum horizontal distance between adjacent returns. The synthesized data sets in their study emulated collections with NPSs of 1–15 m and were evaluated to determine their impact on calculating various characteristic attributes of forests. This article concluded that while estimation accuracy degraded as the thinning level increased, the degradation was gradual up to an NPS of 10 m. Furthermore, the accuracy of the attributes obtained, even at an NPS of 15 m, were at least equivalent to those commonly obtained using traditional photogrammetric methods.

Ruiz *et al.* [11] independently confirmed the results of the study performed by Magnusson *et al.* [8] in their study that examined the combined effects of plot size and LiDAR density on forest structure attribute estimates. This article found that while LiDAR density has a slight influence on the derived models, plot size was the dominant factor in estimating structure attributes. This article concluded that the optimal NPD for forest structure attribute estimation is in the range of 1–5 pls/m^2.

In a similar study, Hansen *et al.* [13] examined the impacts of LiDAR density on digital terrain model (DTM) and canopy metrics in tropical rainforests. The results of the study largely echoed the findings of Ruiz *et al.* [11], with a slightly coarser NPD recommendation of 0.5 pls m^{-2}.

Vauhkonen *et al.* [9] examined the effects of LiDAR sampling density on finer structures, expanding the work previously done on forest-level estimation to tree-level estimation. The authors adapted the thinning approach introduced by Magnusson *et al.* [8] to be based on rectangular rasters of increasing cell size. The NPD of the sample data sets were significantly greater than those evaluated by Magnusson *et al.* [8], emulating collections with NPDs of 0.6–25 pls/m^2 (equivalent to NPSs of 0.2–1.3 m). The authors concluded that while NPD on the order of 12 pls/m^2 were typical for LiDAR collections supporting tree-level applications, a coarser point density on the order of 3 pls/m^2 was sufficient for species identification and calculating characteristic attributes of the tree species evaluated.

The studies presented above are not exhaustive; rather, they illustrate the formal research that has informed the establishment of NPD strata for LiDAR point cloud products based on use case and phenomenology. Isenburg [22] summarized several common product levels that have emerged through this process in Table I. Formally, NPD recommendations are being codified as one aspect of LiDAR product specifications. Both the American Society for Photogrammetry and Remote Sensing (ASPRS) [23] and the United States Geological Survey (USGS) [21] provide sampling recommendations for LiDAR products as illustrated by Tables II and III. Note that USGS distinguishes the metrics of aggregate NPS (ANPS) and aggregate NPD (ANPD) from NPS and NPD to describe the net overall pulse spacing and density, respectively, for products generated from multiple coverages. For single coverage products, the aggregate and nonaggregate metrics are identical.

Previously, the Petersen–Middleton theorem [24], [25] has been leveraged to establish sampling and reconstruction strategies in parallel domains including seismology [26], computed tomographic reconstruction [27], and periodic nonuniform sampling [28]. In this article, we apply the Petersen–Middleton theorem [24] to establish a strategy for sampling point clouds to generate level of details (LOD) which seek to

TABLE I
COMMON POINT DENSITIES [22]

Min NPD (pls/m^2)	Use Case
0.5	Basic surface model, Forest inventory
1	Flood modeling, Dam and water inundation calculations
2	Multi-purpose data sets
5	Basic 3D models
10	Detailed 3D city models

TABLE II
USGS NGP/3DEP QUALITY LEVELS [21]

Quality Level	Max ANPS (m)	Min ANPD (pls/m^2)
QL0	0.35	8.0
QL1	0.35	8.0
QL2	0.71	2.0
QL3	1.41	0.5

TABLE III
EXAMPLES OF VERTICAL ACCURACY AND RECOMMENDED LIDAR POINT DENSITY FOR DIGITAL ELEVATION DATA ACCORDING TO THE NEW ASPRS 2014 STANDARD [23]

Vertical Accuracy Class (cm)	Absolute Accuracy: RMSE$_z$ Non-Vegetated (cm)	Absolute Accuracy: NVA at 95% Confidence Level (cm)	Rec. Min NPD (pls/m^2)	Rec. Max NPS (m)
1.0	1.0	2.0	20.00	0.22
2.5	2.5	4.9	16.00	0.25
5.0	5.0	9.8	8.00	0.35
10.0	10.0	19.6	2.00	0.71
15.0	15.0	29.4	1.00	1.00
20.0	20.0	39.2	0.50	1.40
33.3	33.3	65.3	0.25	2.00
66.7	66.7	130.7	0.10	3.20
100.0	100.0	196.0	0.05	4.50
333.3	333.3	653.3	0.01	10.00

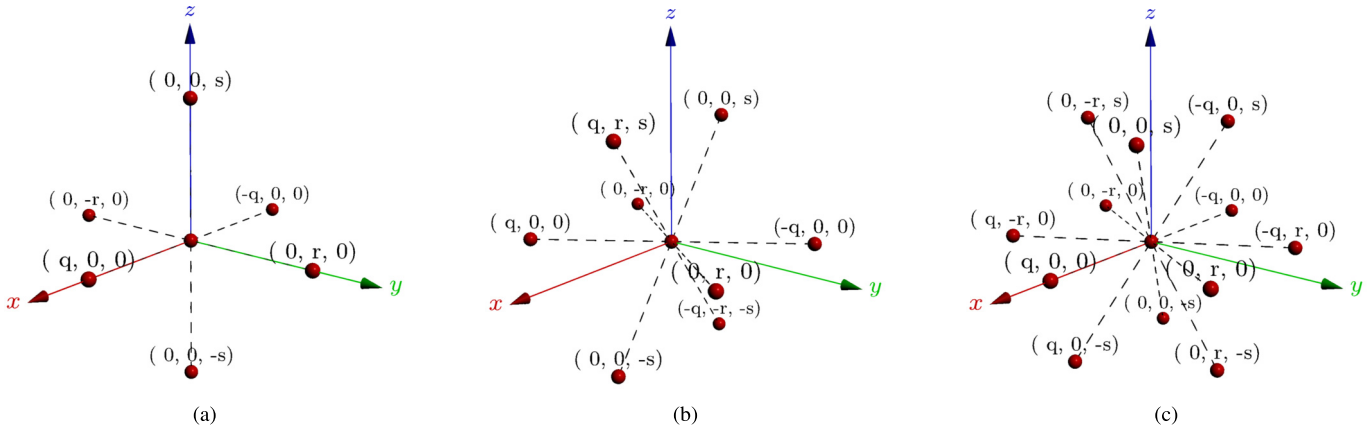


Fig. 1. Bases for (a) SC, (b) BCC, and (c) FCC lattices transformed for sampling point cloud data. Central sites are connected to NNs.

optimally enforce one of the following constraints: 1) resample to a desired interpoint spacing; 2) resample to a desired point density; 3) resample to a desired number of points; 4) resample to a desired texture resolution; and 5) resample to a desired spatial resolution. Preliminary analysis of our approach was presented via poster at the ASPRS 2019 Annual Conference [29].

The general organization of this article is as follows. Section II describes our sampling approach. Section III discusses approaches for representing the point cloud LODs. Section IV discusses our sampling approach for processing single-photon and Geiger-mode avalanche photodiode (GmAPD) LiDAR. Section V describes our evaluation approach. Section VI summarizes the comparison of our approach to the two most common point cloud sampling methods: random sampling and rectangular lattice sampling. Section VII is a summary.

II. Approach

The Petersen–Middleton theorem [24] provides the conditions for perfect reconstruction of wavenumber-limited functions from samples on regular discrete lattices. Similar to the Nyquist–Shannon sampling theorem, the conditions for perfect reconstruction are seldom realized in real-world applications; however, the theorem still provides practical guidance for establishing optimal sampling lattices for an n-dimensional field to an objective wavenumber limit. Indeed, it is this theorem that provides the result that optimal sampling for 2-D wavenumber-limited isofunctions is achieved with a hexagonal lattice and that optimal sampling for 3-D wavenumber-limited isofunctions is achieved with a body-centered cubic (BCC) lattice. These two results inform the selection of optimal sampling lattices for objective texture and spatial resolutions.

Using the same approach as Petersen and Middleton [24] to determine optimal sphere packings in the spatial domain versus the spectral domain provides a means for selecting lattices that optimize point spacing and point density metrics instead of texture and spatial resolution. Furthermore, while there are infinitely many lattices that could be used to sample 3-D spaces, we find that only three appear in the context of optimal sampling lattices in $\mathbb{R}^3$: 1) the simple cubic (SC) lattice; 2) the face-centered cubic (FCC) lattice; and 3) the BCC lattice [30]. The SC lattice is often considered to be the natural choice for a sampling lattice since it establishes a Cartesian coordinate system. However, the FCC and BCC lattices arise as alternative sampling lattices as a result of dense sphere packing in the spectral domain and the subsequent transformations to the spatial domain. The three lattices and their respective coordinate systems are illustrated in Fig. 1.

A. Basis Selection

A lattice ($\boldsymbol{\Lambda}$) that spans the n-dimensional real vector space ($\mathbb{R}^n$) is given by

$$\boldsymbol{\Lambda} = \mathcal{L}(\boldsymbol{B}) = \left\{\boldsymbol{B}\boldsymbol{c} : \boldsymbol{c} \in \mathbb{Z}^n \wedge \det(\boldsymbol{B}) \neq 0\right\} \tag{2}$$

where

$$\boldsymbol{B} = \left[\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\right] \tag{3}$$

is the matrix of basis vectors known as the generating or sampling matrix. The sampling matrix establishes a hypercubic coordinate reference frame for the vector space spanned by $\boldsymbol{\Lambda}$, where the basis vectors, $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$, are the frame axes and coordinates, $\boldsymbol{c} \in \mathbb{Z}^n$, define the sample sites. The sampling matrix and its inverse thus represent the changes of basis between the vector space representing the sampling reference frame and the vector space representing the world reference frame.

The sampling matrix for each of the lattices we use are straightforward to establish directly. The SC sampling matrix

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

establishes adjacent sample sites at adjacent cube corners, the FCC sampling matrix

$$\begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix} \tag{5}$$

establishes adjacent sample sites at adjacent cube face centers, and the BCC sampling matrix

$$\begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \tag{6}$$

establishes adjacent sample sites at adjacent cube body centers. While there are infinitely many ways to orient and scale the basis vectors for each sampling lattice, we take the approach of ensuring two of the basis vectors span the Cartesian xy-plane, which is typically parallel to the ground plane for LiDAR data sets. Furthermore, we normalize the basis vectors to establish a normal basis for the vector space so that a default scaling will preserve the same interpoint sample spacing in each sampling lattice. For the SC and FCC sampling matrices, we further constrain the basis vector arrangement by aligning one basis vector with the x-axis, which typically points in the easting direction for LiDAR data sets. This means that the SC sampling matrix is simply identity. The BCC basis vectors, however, do not form a regular tiling of the xy-plane. Thus, instead of aligning a basis vector with the x-axis, we choose to orient the basis vectors so that they are reflexively symmetric about the y-axis, which typically points in the northing direction for LiDAR data sets. The final set of fundamental sampling matrices that we use are thus

$$\boldsymbol{B}_{\mathrm{SC}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

$$\boldsymbol{B}_{\mathrm{FCC}} = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{6} \\ 0 & 0 & \frac{\sqrt{6}}{3} \end{bmatrix} \tag{8}$$

and

$$\boldsymbol{B}_{\mathrm{BCC}} = \begin{bmatrix} \frac{\sqrt{6}}{3} & -\frac{\sqrt{6}}{3} & 0 \\ \frac{\sqrt{3}}{3} & \frac{\sqrt{3}}{3} & -\frac{\sqrt{3}}{3} \\ 0 & 0 & \frac{\sqrt{6}}{3} \end{bmatrix}. \tag{9}$$

B. Spacing Scale Factors

With the sampling matrices established as described in Section II-A, lattice sites are arranged as groups of planes in $\mathbb{R}^3$. The SC lattice forms square lattices within the planar groups. The groups are arranged with lattice sites located directly above each other with a separation equal to the minimum distance of the lattice. Both the BCC and FCC lattices have planar groups that are separated by a factor of $\sqrt{6}/3$ of the minimum distance of the lattice. The BCC lattice sites form an irregular hexagon lattice within the planes and the planar groups alternate in an ABAB sequence. The FCC lattice sites form a regular hexagon lattice within the planes and the planar groups are arranged in a repeating ABCABC sequence.

This arrangement of basis vectors allows us to independently control the sampling intervals in the horizontal and vertical directions, denoted by r_h and r_v, respectively. Since each fundamental sampling matrix is designed so that the first two basis vectors span the xy-plane, the fundamental vertical separation between planar groups is always given by the sampling matrix element $b_{3,3}$. Thus, the lattice basis is scaled to achieve the desired sampling rates with the matrix

$$\boldsymbol{S} = \begin{bmatrix} r_h & 0 & 0 \\ 0 & r_h & 0 \\ 0 & 0 & \frac{1}{b_{3,3}} r_v \end{bmatrix}. \tag{10}$$

To achieve a uniform scaling of the sampling lattice, we establish the sampling interval with r_h and automatically compute the desired vertical separation of planar groups as

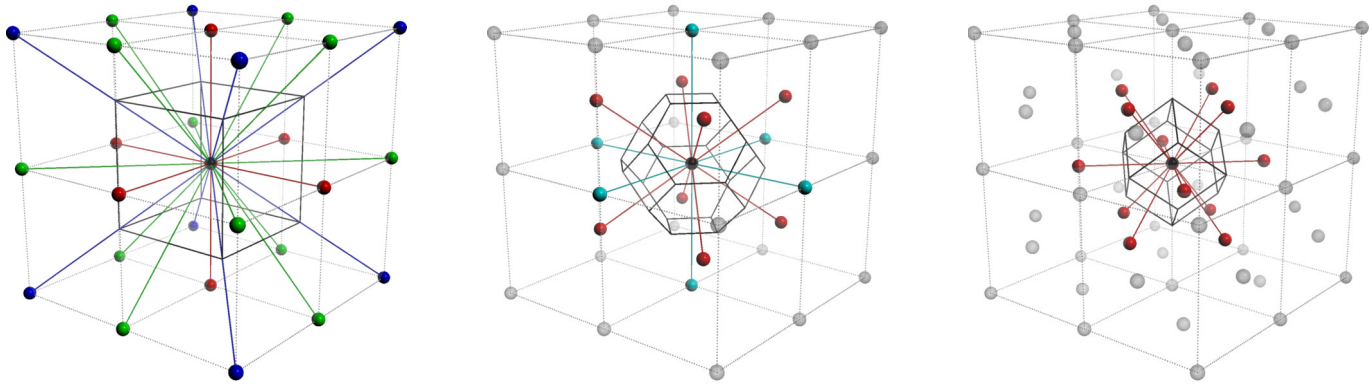
$$r_v \triangleq b_{3,3} r_h. \tag{11}$$

C. Characteristic-Preserving Scale Factors

Next, we establish a set of scale factors for each lattice that allows us to target a specific characteristic to preserve during the point cloud sampling. The scale factors we compute preserve the following characteristics: 1) interpoint spacing: scales the sampling lattices so that the interpoint distance between nearest neighbors (NNs) match; 2) ground-plane sampling density: scales the sampling lattices so that the sampling densities of the xy-plane match; 3) number of sample sites: scales the sampling lattices so that the sampling densities (thus, the expected number of sample sites) in $\mathbb{R}^3$ match; 4) texture resolution: scales the sampling lattices so that the circular support regions in the spectral domain match; and 5) spatial resolution: scales the sampling lattices so that the spherical support regions match in the spectral domain.

To preserve the interpoint spacing between the sampling lattices, the magnitudes of the basis vectors must be scaled to match. Since we establish the fundamental sampling matrices with normalized basis vectors, the spacing-preserving scale factor is simply the multiplicative identity

$$s_i = 1. \tag{12}$$



(a) (b) (c)

Fig. 2. Eight unit cells of (a) SC, (b) BCC, and (c) FCC lattice, indicated by dotted grey lines. The Voronoi cell for each of the central lattice sites is indicated with black lines. Adjacent sites are colored according to the respective descriptions.

The sampling density of a lattice is computed from the hyper-volume of its fundamental parallelepiped:

$$d(\mathbf{\Lambda}) = |\det(\boldsymbol{B})| \tag{13}$$

that is, the hyper-volume of the parallelepiped representing the region enclosed by the basis vectors. By definition, there is exactly one sample site within each fundamental parallelepiped region, which implies that the sampling density is simply the inverse of this volume

$$\rho = \frac{1}{d(\mathbf{\Lambda})}. \tag{14}$$

Thus, to preserve the sampling density in $\mathbb{R}^3$, the scale factor is set to normalize this density

$$s_p = \sqrt[3]{d(\mathbf{\Lambda})}. \tag{15}$$

To preserve the sampling density in the xy-plane between the sampling lattices, we leverage the fact that we oriented the sampling matrices so that two of the basis vectors span the xy-plane. This means that we can use the same density normalization approach as above, but restricted to just the first two basis vectors

$$s_d = \sqrt{\left|[\boldsymbol{B}]_{3,3}\right|} \tag{16}$$

where $[\boldsymbol{B}]_{3,3}$ is the $(3,3)$ minor determinant of the sampling matrix.

The final two scale factors are selected to preserve the radial support for wavenumber-limited functions in the spectral domain without aliasing and are therefore related to the reciprocal lattice:

$$\mathbf{\Gamma} = \mathcal{L}\left(\widetilde{\boldsymbol{B}}\right) = \mathcal{L}\left(\boldsymbol{B}^{-\mathsf{T}}\right) \tag{17}$$

where

$$\widetilde{\boldsymbol{B}} = \left[\widetilde{\boldsymbol{v}}_1, \ldots, \widetilde{\boldsymbol{v}}_n\right] \tag{18}$$

is the dual space sampling matrix. The scale factors are defined by the diameter of the largest n-D ball constrained to a Voronoi region of the lattice. This distance is given by the distance between the NN spectral repetitions which is simply the minimum magnitude of the dual space basis vectors. Thus, to compute the scale factor that preserves texture resolution in the xy-plane, we consider only the upper-left 2×2 submatrix

$$\boldsymbol{B}(3,3) = [\boldsymbol{u}_1, \boldsymbol{u}_2] \tag{19}$$

resulting from the elimination of the third row and columns from the matrix $\boldsymbol{B}$. We then compute the minimal magnitude of the basis vectors for the dual space

$$s_t = \min_i \|\widetilde{\boldsymbol{u}}_i\|_2. \tag{20}$$

The scale factor that preserves the spatial resolution in $\mathbb{R}^3$ is similarly computed as

$$s_s = \min_i \|\widetilde{\boldsymbol{v}}_i\|_2. \tag{21}$$

TABLE IV
FEATURE-PRESERVING SCALE FACTORS FOR SAMPLING LATTICES

Preservation Strategy	SC	FCC	BCC
Spacing (s_i)	1	1	1
Density (s_d)	1	$\frac{\sqrt[4]{108}}{3}$	$\frac{\sqrt[4]{18}}{2}$
Samples (s_p)	1	$\sqrt[6]{2}$	$\frac{\sqrt[6]{108}}{2}$
Texture Resolution (s_t)	1	$\frac{2\sqrt{3}}{3}$	$\frac{3\sqrt{2}}{4}$
Spatial Resolution (s_s)	1	$\frac{\sqrt{6}}{2}$	$\frac{\sqrt{6}}{2}$

Table IV provides the values of the scale factors described above for the SC, FCC, and BCC lattices.

The volumes represented by each sample site are given by the Voronoi tessellation of the lattice as illustrated in Fig. 2. Since each sample site is identical within $\mathbf{\Lambda}$, each volume is an identical unit cell that has the property of being a space-filling polyhedron as illustrated in Fig. 3. By comparing the relative sizes of the representative volumes under different scaling factors, we can establish the expected relative sampling efficiency between the sampling lattices. Table V

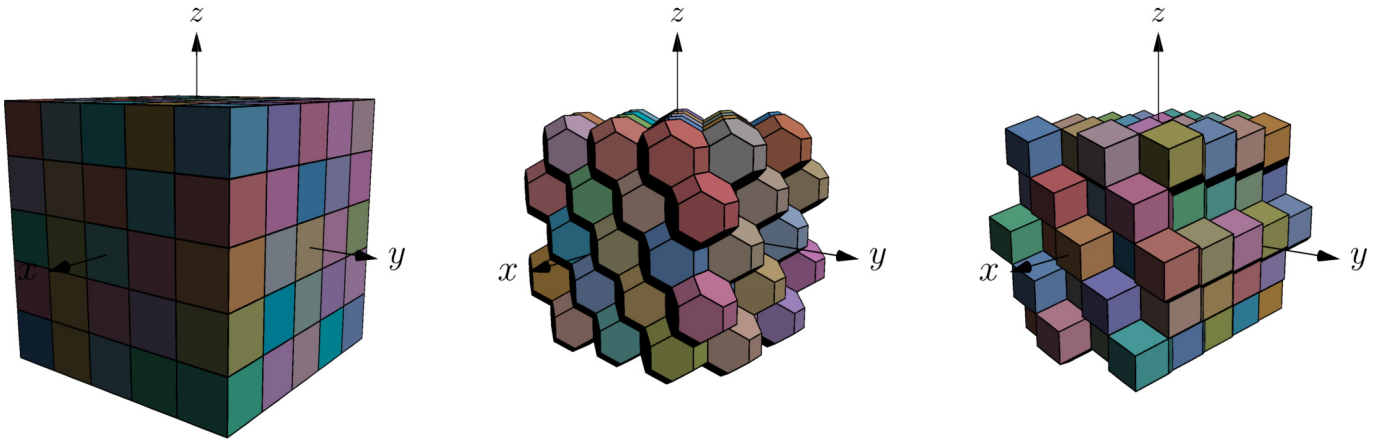


Fig. 3. Voronoi tesselations of (a) SC, (b) BCC, and (c) FCC lattices. Individual cells colored randomly to illustrate the structure.

TABLE V
DISCRETE LATTICE SAMPLES RELATIVE TO SC LATTICE ($\Delta\%$)

	3-D		2-D	
Preservation Strategy	FCC	BCC	FCC	BCC
Spacing	$+41.42$	$+29.90$	$+15.47$	$+6.07$
Density	$+13.98$	$+18.92$	—	—
Samples	—	—	-8.35	-10.91
Texture Resolution	-8.14	$+8.87$	-13.40	-5.72
Spatial Resolution	-23.02	-29.29	-23.02	-29.29

summarizes the expected relative number of samples for each preservation strategy using the SC lattice as a reference. This table illustrates that, as predicted by the Petersen–Middleton theorem, the BCC lattice provides the most efficient sampling that maintains spatial resolution; however, resolving features in texture—like changes in intensity within the ground plane—is less efficient than the FCC lattice. The FCC lattice, while not as efficient at maintaining spatial resolution, is the only lattice considered that provides improvements in sampling efficiency for both spatial and texture resolution.

D. Preliminaries

Our sampling algorithms assume the existence of NN search (NNS) algorithms in metric spaces with a run-time complexity of no worse than $\mathcal{O}(\log n)$. Specifically, we require a k-NN (k-NN) search with a Euclidean distance metric for both nearest-center NC sampling and mass point (MP) sampling. MP sampling additionally requires a radially NN (r-NN) search with a Chebyshev distance metric. We assume that the points are spatially indexed with the appropriate distance metric and that the index structure provides appropriate interfaces for the respective searches. Algorithm 1 details the behavior of the k-NN search, while Algorithm 2 details the behavior of the r-NN search. Any metric tree or binary space partitioning tree data structure can satisfy the requirements of the search algorithms. For our implementation, we leveraged vantage point trees [31] for the spatial indexing data structure.

Algorithm 1 k-NNs

Require:

▷ A database of points, $\mathcal{D}$, spatially indexed with a distance metric, $d(\boldsymbol{a}, \boldsymbol{b})$

▷ A query point, $\boldsymbol{q}$

▷ A neighborhood size, $k \in \mathbb{N}$

function KNN($\mathcal{D}$, $\boldsymbol{q}$, k)

$\mathcal{N} \leftarrow \{\mathcal{A} \subseteq \mathcal{D} : |\mathcal{A}| = k, (\forall \boldsymbol{x} \in \mathcal{A}, \boldsymbol{x}' \in \mathcal{D} \setminus \mathcal{A}) (d(\boldsymbol{x}, \boldsymbol{q}) \leq d(\boldsymbol{x}', \boldsymbol{q}))\}$

return $\mathcal{N}$

end function

Algorithm 2 r-NN

Require:

▷ A database of points, $\mathcal{D}$, spatially indexed with a distance metric, $d(\boldsymbol{a}, \boldsymbol{b})$

▷ A query point, $\boldsymbol{q}$

▷ A neighborhood reach, $r \in \mathbb{R}$

function RNN($\mathcal{D}$, $\boldsymbol{q}$, r)

$\mathcal{N} \leftarrow \{\mathcal{A} \subseteq \mathcal{D} : (\forall \boldsymbol{x} \in \mathcal{A}, \boldsymbol{x}' \in \mathcal{D} \setminus \mathcal{A}) (d(\boldsymbol{x}, \boldsymbol{q}) \leq r < d(\boldsymbol{x}', \boldsymbol{q}))\}$

return $\mathcal{N}$

end function

E. NC Sampling

The first algorithm we develop is NC sampling. This approach most closely matches uniform sampling on a regular discrete lattice with the added constraint that all samples

Algorithm 3 NC Sampling

▷ Select points from point cloud $\mathcal{P}$ nearest to lattice sites in $\mathcal{L}(\boldsymbol{B})$
function NCSAMPLE($\mathcal{P}$, $\boldsymbol{B}$)
  ▷ Change points to sampling coordinate frame
  $\mathcal{P}_{\boldsymbol{B}} \leftarrow \left\{ \boldsymbol{B}^{-1}\boldsymbol{p} : \boldsymbol{p} \in \mathcal{P} \right\}$
  ▷ Identify occupied Voronoi cells of the lattice
  $\mathcal{C} \leftarrow \{ \text{COMPONENTWISEROUND}(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{P}_{\boldsymbol{B}} \}$
  ▷ Spatially index points for efficient search
  $\mathcal{D}_{\text{E}} \leftarrow \text{EUCLIDEANINDEX}(\mathcal{P}_{\boldsymbol{B}})$
  ▷ Identify sample points
  $\mathcal{S} \leftarrow \varnothing$
  for all $\boldsymbol{c} \in \mathcal{C}$ **do**
    ▷ Find the closest point to the lattice site
    $\boldsymbol{s} \leftarrow \text{KNN}(\mathcal{D}_{\text{E}}, \boldsymbol{c}, 1)$
    ▷ Ensure adjacent sites are not aliased by the candidate
    if $\boldsymbol{c} \equiv \text{COMPONENTWISEROUND}(\boldsymbol{s})$ **then**
      $\mathcal{S} \leftarrow \mathcal{S} \cup \{ \boldsymbol{B}\boldsymbol{s} \}$
    end if
  end for
  return $\mathcal{S}$
end function

belong to the set of observations instead of to the set of lattice sites. NC sampling, as detailed in Algorithm 3, proceeds by first transforming all points into the sampling coordinate frame. Within this reference frame, the lattice sites associated with each occupied Voronoi cell are computed by performing a componentwise rounding of the point coordinates. Next, initial sample candidates are selected by identifying the points closest to each query lattice site. However, this process does not guarantee that the candidates occupy the same Voronoi cells as their respective lattice sites. Candidates appear in Voronoi cells adjacent to the query lattice site whenever two conditions are met. First, the query cell must be occupied only by points in "corners" of the cell. That is, when the occupying points fall outside the sphere inscribed within the cell (recall that within the sampling reference frame, all Voronoi cells are cubes). Second, at least one point in an adjacent cell must be within the sphere circumscribing the cell and closer to the lattice site than any point within the cell. In this case, the closest point to the lattice site will appear in an adjacent cell, thus acting as a form of aliasing. Checking for this condition is achieved by ensuring that the query lattice site and the lattice site associated with the candidate are identical. If not, then the candidate is rejected and the Voronoi cell is unrepresented in the final sample set. The final samples are produced by transforming the selected candidates back to the world coordinate reference frame. This final step is unnecessary if selected points can be identified by other means, such as a globally unique identifier.

Algorithm 4 MP Sampling

▷ Select points from point cloud $\mathcal{P}$ nearest to centers of mass of Voronoi regions in $\mathcal{L}(\boldsymbol{B})$
function MPSAMPLE($\mathcal{P}$, $\boldsymbol{B}$)
  ▷ Change points to sampling coordinate frame
  $\mathcal{P}_{\boldsymbol{B}} \leftarrow \left\{ \boldsymbol{B}^{-1}\boldsymbol{p} : \boldsymbol{p} \in \mathcal{P} \right\}$
  ▷ Identify occupied Voronoi cells of the lattice
  $\mathcal{C} \leftarrow \{ \text{COMPONENTWISEROUND}(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{P}_{\boldsymbol{B}} \}$
  ▷ Spatially index points for efficient search
  $\mathcal{D}_{\text{C}} \leftarrow \text{CHEBYSHEVINDEX}(\mathcal{P}_{\boldsymbol{B}})$
  $\mathcal{D}_{\text{E}} \leftarrow \text{EUCLIDEANINDEX}(\mathcal{P}_{\boldsymbol{B}})$
  ▷ Identify sample points
  $\mathcal{S} \leftarrow \varnothing$
  for all $\boldsymbol{c} \in \mathcal{C}$ **do**
    ▷ Find points in the Voronoi region about the site
    $\mathcal{N} \leftarrow \text{RNN}(\mathcal{D}_{\text{C}}, \boldsymbol{c}, \frac{1}{2})$
    ▷ Compute the center of mass for the region
    $\boldsymbol{m} \leftarrow \text{AVERAGE}(\mathcal{N})$
    ▷ Find the closest point to the mass-point
    $\boldsymbol{s} \leftarrow \text{KNN}(\mathcal{D}_{\text{E}}, \boldsymbol{m}, 1)$
    ▷ Ensure adjacent sites are not aliased by the candidate
    if $\boldsymbol{c} \equiv \text{COMPONENTWISEROUND}(\boldsymbol{s})$ **then**
      $\mathcal{S} \leftarrow \mathcal{S} \cup \{ \boldsymbol{B}\boldsymbol{s} \}$
    end if
  end for
  return $\mathcal{S}$
end function

F. MP Sampling

One obvious limitation of the NC sampling approach is that an observation must be sufficiently close to the lattice site, without falling outside the Voronoi cell, to be selected as a representative sample. In the sampling coordinate reference frame, only $(\pi/6) \approx 52.36\%$ of the volume of the Voronoi cell matches this constraint. Cells are not sampled when observations are biased into the "corners" of the cells which happens most often for fine linear features and along edges of bodies. By changing the query location from the lattice site to the center of mass of observations within the Voronoi cell, the number of unrepresented occupied cells is reduced. Algorithm 4 illustrates the modifications required for performing MP sampling.

The primary modifications that are required to support MP sampling are an additional spatial index to enable searches under a Chebyshev distance metric and an additional step to compute the center of mass of observations within each Voronoi cell. The Chebyshev metric is leveraged to quickly identify all observations contained within the Voronoi cell

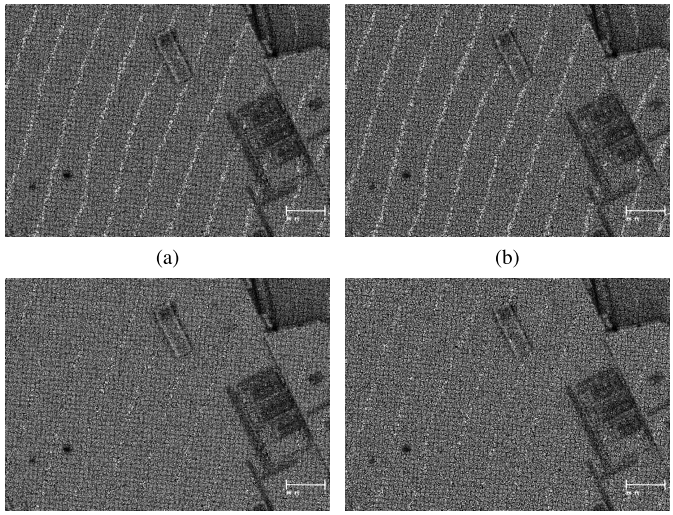


Fig. 4. (a) NC sampling with contours. (b) MP sampling with contours. (c) NC sampling with mitigation. (d) MP sampling with mitigation.

since the cells are balls of radius $(1/2)$ under this metric in the sampling coordinate reference frame. Otherwise, sampling proceeds exactly as NC sampling, including the aliasing check.

G. Novelty Versus Voxel-Based Approaches

With the primary sampling approach defined, we can now describe how our approach differs from rectangular voxel-based representations similar to the approach proposed by Stoker [32]. First, the lattice constraints are a generalization of a rectangular voxel grid with specific lattices providing key benefits for specific applications as detailed in Section II-C. Second, the LOD points are down-selected from an input cloud instead of resampled to lattice sites. This distinction may have important implications for propagation of per-point metadata. In fact, our sampling approach would be nearly identical to voxel representations if we simply generated the set of occupied voxel cells and operated only on rectangular lattices.

H. Contour-Line Sampling Artifact and Mitigation

A consequence of allowing samples to float within the Voronoi cells is the appearance of a visual artifact that we named density contours. This artifact is a post-aliasing that appears when rendering point clouds to visual displays. Since no surface is perfectly smooth and no ranging system is infinitely accurate, there is always a "thickness" for every surface observed by LiDAR systems. If the vertical spread of the surface—due to either texture or measurement uncertainty—is not entirely contained within a single sampling layer, multiple vertically adjacent sites will generate samples leading to the appearance of the contour lines. This artifact is not unique to our sampling approach. In fact, if we performed uniform discrete sampling, this artifact would manifest instead as the familiar "stair-step" aliasing associated with rasterization.

Algorithm 5 Density Contour Mitigation

▷ Mitigate the appearance of density contour artifacts that arise from sampling point cloud $\mathcal{P}$ under the constraints imposed by $\mathcal{L}(\boldsymbol{B})$

function MITIGATECONTOURS($\mathcal{P}$, $\mathcal{B}$)

▷ Change points to sampling coordinate frame

$\mathcal{P}_{\boldsymbol{B}} \leftarrow \left\{ \boldsymbol{B}^{-1}\boldsymbol{p} : \boldsymbol{p} \in \mathcal{P} \right\}$

▷ Spatially index points for efficient search

$\mathcal{D}_{\mathrm{E}} \leftarrow$ EUCLIDEANINDEX($\mathcal{P}_{\boldsymbol{B}}$)

▷ Initially, all candidates belong to the sample set

$\mathcal{S} \leftarrow \mathcal{P}$

▷ Iteratively remove aliasing candidates

for all $\boldsymbol{s} \in \mathcal{S}$ **do**

▷ Find points in cell centered at candidate

$\mathcal{N} \leftarrow$ RNN($\mathcal{D}_{\mathrm{E}}$, $\boldsymbol{B}^{-1}\boldsymbol{s}$, $\frac{1}{2}$)

▷ Remove candidate if it aliases a sample

for all $\boldsymbol{n} \in \mathcal{N}$ **do**

if $\boldsymbol{B}\boldsymbol{n} \in \mathcal{S}$ **then**

$\mathcal{S} \leftarrow \mathcal{S} \setminus \{\boldsymbol{s}\}$

break

end if

end for

end for

return $\mathcal{S}$

end function

While the artifact is purely a visual anomaly, similar to a Moiré pattern, its appearance is nonetheless unexpected and may thus be detrimental to exploitation and analysis of point cloud data. We therefore developed an approach to mitigate the appearance of density contours as detailed in Algorithm 5 that can be applied as a post-sampling process.

First, observe that the density contour artifact appears where sample points are spaced closer than predicted by the lattice constraint, that is, where the points appear to be an alias of each other. We can thus simply check whether a sample point is a part of a density contour artifact by searching for any additional samples within a ball of radius one half in the sampling coordinate reference frame. If any other samples are identified, the point is removed from the set of samples. This process is repeated for all samples. Pathological point removal is not possible because of the lattice constraint used to perform the original sampling. Fig. 4 illustrates the appearance and mitigation of the density contour post-aliasing artifact under both NC and MP sampling approaches.

The primary detriment of our mitigation approach is that it is order-dependent and thus implicitly serial. We apply a seeded random shuffle to point ordering to avoid introducing a sampling bias through the mitigation process while maintaining repeatable results. However, we recognize the implicit limitations of order-dependent processing. While we recognize

Algorithm 6 LOD Generation

▷ Select points from point cloud $\mathcal{P}$ to participate in each level of detail (LOD) $\mathcal{S} \in \mathcal{H}$ as constrained by lattice bases $\mathcal{B}$

function GENERATEDETAILLEVELS($\mathcal{P}$, $\mathcal{B}$)
  ▷ Ensure the LOD set is empty
  $\mathcal{H} \leftarrow \varnothing$
  for all $\boldsymbol{B} \in \mathcal{B}$ **do**
    $\mathcal{S} \leftarrow$ SAMPLE($\mathcal{P}$, $\boldsymbol{B}$)
    $\mathcal{H} \leftarrow \mathcal{H} \cup \{\mathcal{S}\}$
  end for
  return $\mathcal{H}$
end function

this drawback, we defer further improvement to the mitigation approach to future work.

I. LOD Generation

The lattice-constrained sampling (LCS) and contour mitigation approaches comprise the sampling methods leveraged for generating each point cloud LOD. LOD generation thus proceeds in a straightforward manner as detailed in Algorithm 6. Essentially, each LOD is an independent sampling of the input point cloud. We do not base subsequently coarser LODs on the immediately preceding finer level because the scales of the lattice constraints between LODs, as illustrated in Section I and described by (10), are usually not integer multiples of each other. This decision has direct implications on the LOD representation as described in Section III.

J. Algorithm Complexity

The overall algorithm complexity of both the NC and MP sampling approaches are very similar. Given an input point cloud ($\mathcal{P}$), for $n \leftarrow |\mathcal{P}|$, the conversion to the sampling coordinate frame and identification of occupied lattice cells ($\mathcal{C}$) both take $\mathcal{O}(n)$ time since the operations involve only simple multiplications and comparisons. Spatially indexing the input point cloud takes $\mathcal{O}(n \log n)$ time on average when metric tree data structures are used as the indexing structure. For $m \leftarrow |\mathcal{C}|$, the identification of sample points takes $\mathcal{O}(m \log n)$ time on average since locating the representative point for each occupied lattice cell involves a $\mathcal{O}(\log n)$ search of the spatial indexing structure. This means that, in general, LOD generation is expected to be no worse than $\mathcal{O}(n \log n)$ on average even when $m \to n$.

Generating multiple LODs is accomplished in a straightforward way by making multiple calls to the underlying sampling function as described in Section II-I. In this case, the overall time complexity depends largely on the number of LODs being generated (p). However, if each LOD results in a fixed magnitude reduction factor (f), then the total number of LODs required before the LOD cardinality is reduced to 1 is just $\log_f(n)$. Thus, the overall complexity is expected to be $\mathcal{O}(pn \log n)$ with a practical limit of $\mathcal{O}\left(n \log^2 n\right)$.

III. LEVEL OF DETAIL REPRESENTATION

Once the LOD point sets are identified, a decision must be made for how to store the point sets. Several obvious strategies include: 1) separating LODs into independent files; 2) segregating LODs into independent sections within a file; and 3) labeling each point with the LODs they represent.

We can make a few simplifying assumptions to help trade the various storage schemes. If we assume a single surface with uniformly distributed points, then the NPD for the base point cloud is estimated by

$$\rho_{\text{nom}} = \frac{n}{A} \tag{22}$$

where n is the number of points in the base point cloud and A is the area covered by the point cloud. In practice, this formula tends to yield inflated density estimates since there are often multiple surfaces over the least part of the area covered for natural scenes. However, it establishes a reasonable estimate for the purposes of trading LOD representations.

We can similarly leverage (14) to estimate the expected point density for each LOD, $\mathcal{L}$, based on its sampling lattice basis, $\boldsymbol{B}_{\mathcal{L}}$. Recall that since we have assumed that a single surface is being sampled, we must use the 2-D sampling density estimate

$$\rho_{\mathcal{L}} = \frac{1}{\left|[\boldsymbol{B}_{\mathcal{L}}]_{3,3}\right|}. \tag{23}$$

Given the assumption that points in the base point cloud are uniformly distributed over the sampled area, the estimated probability that a point, p, is selected to represent an LOD, $\mathcal{L}$, is given by

$$\Pr(\mathcal{L}) = \Pr(p \in \mathcal{L}) = \frac{\rho_{\mathcal{L}}}{\rho_{\text{nom}}}. \tag{24}$$

This estimate serves as the foundation for trading the various proposed storage schemes.

First, if LODs are stored in separate files or file regions, then the total expected number of points to be stored is given by

$$c_{\text{seg}} = n \sum_{i=1}^{L} \Pr(\mathcal{L}_i) \tag{25}$$

where c_{seg} is the expected total point count for the segregated schemes and L is the total number of LODs.

Otherwise, if points are tagged with the LODs they represent, then the total expected number of points to be stored is given by

$$c_{\text{tag}} = n\left(1 - \prod_{i=1}^{L} (1 - \Pr(\mathcal{L}_i))\right) \tag{26}$$

where c_{tag} is the expected total point count for tagged schemes. With this approach, there is a potential savings when points participate in multiple LODs. In fact, the optimal case is realized when all of the coarser LODs are subsets of the finest LOD. There is no requirement for this constraint, though, and it is not expected in general. Furthermore, note that as the probability of a point participating in any single level increases, c_{tag} converges to n.

Given the observations above for tagged representations, we can make a general recommendation that when the finest LOD sampling rate is close to the input point cloud NPS, the tagged scheme is likely to result in better storage efficiency. When the LOD sampling rates are all very coarse with respect to the input point cloud, there is a low probability that points will participate in multiple LODs which implies a segregated scheme will result in a better storage efficiency.

More formally, file sizes are expected to be dominated by point records. Thus, the choice between segregated and tagged representations amounts to identifying when redundancy across LODs makes it more advantageous to use a larger per-point record format. We generally assume that the per-point LOD tag, t, is a byte-aligned bit-field where each byte allows eight LODs to be tagged per point. Thus, a minimum size for t can be established as

$$t_{\min} = \left\lceil \frac{L}{8} \right\rceil. \tag{27}$$

Ignoring header data, we can establish a test variable

$$x = \frac{bc_{\text{seg}}}{(b + t_{\min})\, c_{\text{tag}}} \tag{28}$$

where b is the base per-point record format size, to test for which scheme is expected to be more efficient. When $x < 1$, the segregated scheme is expected to be more efficient. Alternatively, when $x > 1$, the tagged scheme is expected to be more efficient.

IV. Geiger-Mode and Single-Photon LiDAR Processing

In some sensing modalities, specifically single-photon sensitive systems like GmAPD LiDAR, the LOD generation can be extended to sampling the base, prefiltered, point cloud to establish the query points for filtered processing. This approach leads to less aliasing in LODs than sampling from a prefiltered, though potentially densely sampled, point cloud. As explained in Section III, this approach will consequently lead to larger point cloud sets since points that would otherwise be simply support may become query points and the potential for redundancy between LODs decreases.

V. Experiments

To evaluate our sampling methods, we leveraged NPS and NPD metrics calculated using approaches similar to those developed by Naus [33]. The analysis involves computing NPS via Delaunay triangulation of reflective surface points or ground surface points and NPD via the dual Voronoi diagram.

We made a slight modification to the NPS analysis proposed by Naus [33] to evaluate the entire population of Delaunay edges instead of just the average edge length per Voronoi cell. We made this decision because, especially in the SC and BCC cases, the distribution of Delaunay edge lengths is often bimodal which has the effect of skewing averages. The root cause of the modes is easy to visualize, especially in the SC case. The SC case is a degenerate condition for Delaunay triangulation where a hypotenuse edge must be added for every pair of edges connecting NN points. In this case, it is expected

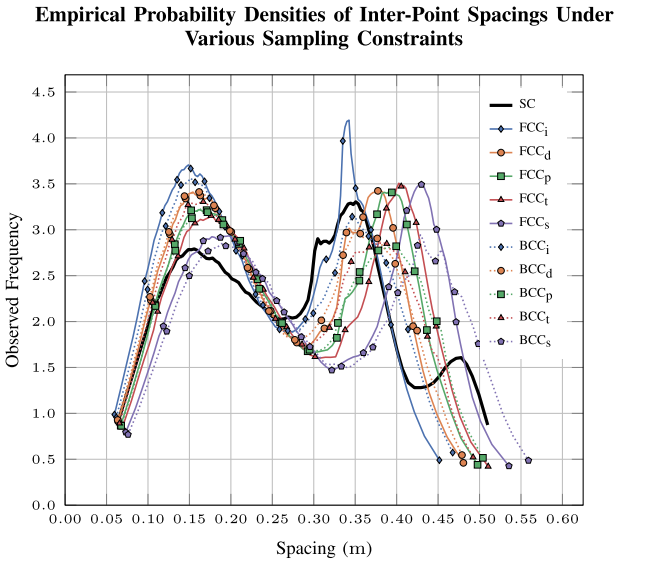


Fig. 5. Observed distribution of interpoint NN spacings for samples generated by various sampling constraints.

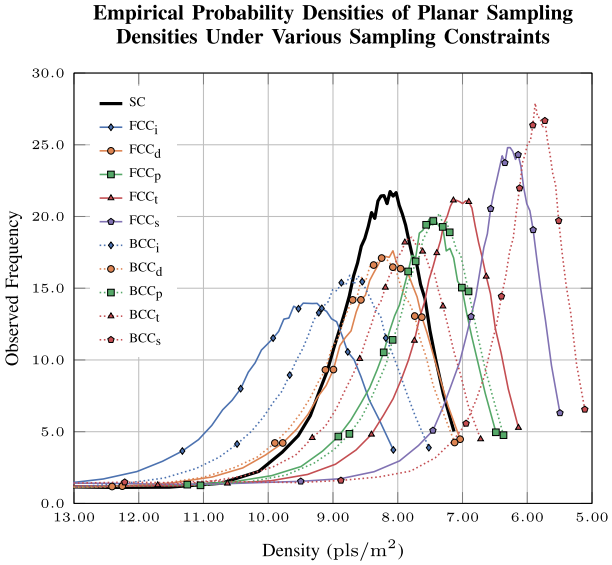


Fig. 6. Observed distribution of Voronoi cell point densities for samples generated by various sampling constraints.

that there are half as many hypotenuse edges as NN edges which is a very significant portion of the population. The BCC case is nondegenerate, but does result in an expected bimodal distribution due to the irregular shape of the hexagonal lattice cells oriented parallel to ground. In this case, however, the two modes are much closer to each other and more difficult to separate.

To evaluate our LOD generation approach, we first compare LCS to the two most common point cloud sampling methods: random sampling and rectangular lattice sampling. Rectangular lattice sampling is simply LCS using an SC lattice for the constraint. This sampling method will generally serve as our baseline for comparison purposes. We perform random sampling via random shuffle based on NPDs as estimated by (22) and (23). Since our test source is GmAPD LiDAR,

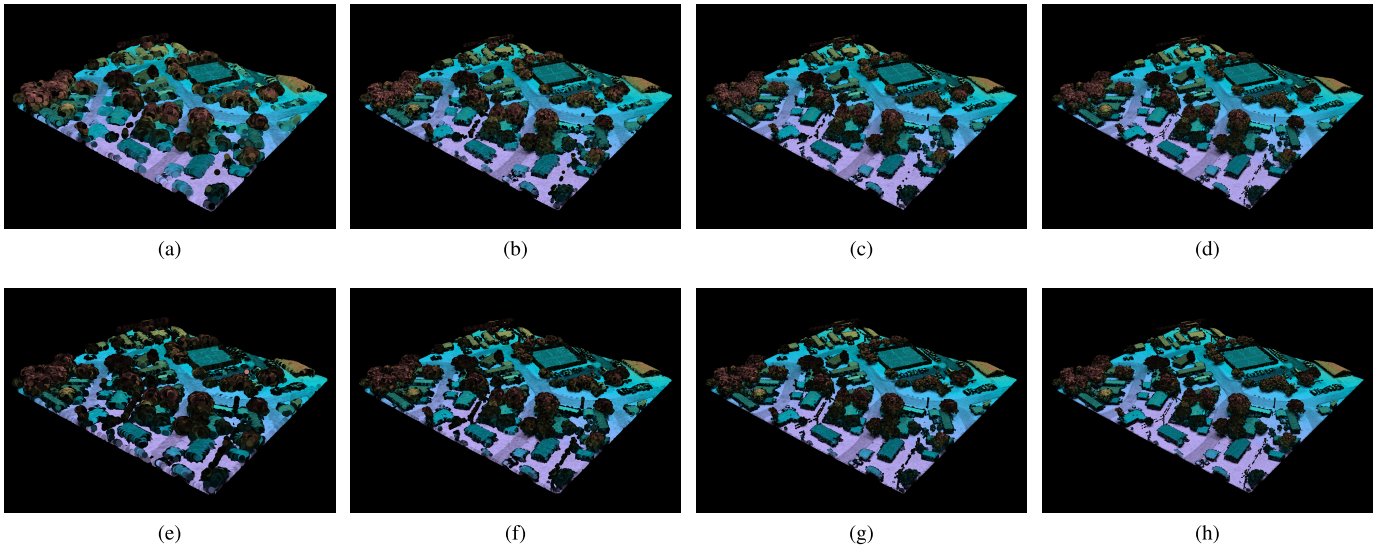


Fig. 7. Examples of directly sampled versus resampled point cloud LODs. (Top row) Directly sampled. (Bottom row) Resampled from 30 pls/m^2. (a) and (e) 1, (b) and (f) 2, (c) and (g) 4, (d) and (h) 8 pls/m^2.

we are able to test generating LODs through both the base sampling and resampling approaches described in Section IV. We provide a comparison of both approaches.

VI. Results and Discussion

We demonstrate our lattice-based sampling approaches using a GmAPD LiDAR point cloud data set with a variety of flat surfaces, buildings, and vegetation. We chose to demonstrate our approach with GmAPD LiDAR since it afforded the opportunity to compare both directly sampling and down-selecting data, which is not possible with linear-mode LiDAR. This is because GmAPD data is generated from raw measurements at several orders of magnitude more than linear-mode LiDAR. However, GmAPD data is very noisy and must be processed to determine which returns originated from actual surfaces in the scene. This process results in a large data reduction since the majority of raw measurements are used only to support the filtering process and are not included as part of the final product. However, our LOD generation approach is equally applicable to all point cloud data sets, regardless of heritage.

The directly sampled results were generated from an input raw data set consisting of two overlapping swaths collected with an elliptical scanning system. The raw data contains 23 836 717 direct ranging measurements within an approximately 22 500-m^2 area. The raw data was filtered using coincidence processing with sample sites established by our proposed lattice constraints as described in Section IV. The lattice constraints were set to target a base intersample spacing of 35 cm. The cumulative distribution of interpoint spacings was computed according to the approach outlined in Section V. Fig. 5 illustrates the observed probability densities of interpoint spacings for the point cloud data set tested. The SC sampling constraint, shown in bold black, serves as the baseline for comparison. As illustrated, interpoint spacing is best preserved among the sampling lattices with the "spacing" constraint imposed. This constraint is represented in the figure by blue lines and legend subscript "i." Observe that the central modes for each of SC, FCC$_i$, and BCC$_i$ are aligned with each other at approximately 35 cm which was the intersample spacing provided to our algorithm. Note the secondary mode in the SC case as predicted from the near-degenerate Delaunay triangulation condition imposed by the square lattice. The secondary mode is predicted to appear at approximately 49.5 cm for perfect square lattices. Our observations show that the secondary mode appears in this case with slightly shorter edge length closer to 47.5 cm.

Similarly, the cumulative distribution of point densities was computed according to the approach outlined in Section V. Fig. 6 illustrates the observed probability densities of sampling densities for the point cloud data set tested. Again, the SC sampling constraint, shown in bold black, serves as the baseline for comparison. As illustrated, sampling density is best preserved among the sampling lattices with the "density" constraint imposed. This constraint is represented in the figure by orange lines and legend subscript "d." Observe that the modes for each of the SC, FCC$_d$, and BCC$_d$ are aligned with each other at approximately 8 pls/m^2 which was the sampling density provided to our algorithm.

Next, we compared predicted point cloud sizes with observed point cloud sizes for directly sampled point clouds. Since our point clouds contain a mix of 2-D and 3-D scene content, we expect the actual samples to generally fall between the predictions given by the 2-D and 3-D constraints. The only case where this is not true is with the "spatial resolution" constraint where the 2-D and 3-D predictions are identical and thus provide an overly restrictive prediction range. Note that actual point counts may be slightly lower than predicted due in part to density contour mitigation as described in Section II-H. Table VI summarizes the results of the experiment. As expected, under all constraints, except "spatial resolution," the actual number of points generated falls

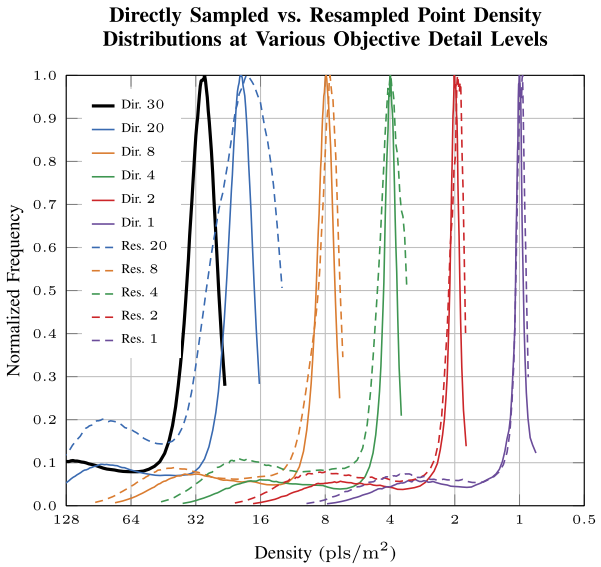


Fig. 8. LOD sample density distributions for GmAPD LiDAR point clouds directly sampled from raw data versus LODs. Generated from 30-pls/m^2 filtered data.

TABLE VI
PREDICTED VERSUS ACTUAL SAMPLES RELATIVE TO SC LATTICE (Δ%)

Lat.	Constraint	Points	Predicted 2-D	Predicted 3-D	Actual
FCC	Spacing	638 973	+15.47	+41.42	+20.95
BCC	Spacing	596 293	+6.07	+29.90	+12.87
BCC	Density	562 802	—	+18.92	+6.53
FCC	Density	554 688	—	+13.98	+4.99
BCC	Texture Res.	531 287	−5.72	+8.87	+0.56
SC		528 312	—	—	—
FCC	Samples	508 776	−8.35	—	−3.70
BCC	Samples	500 915	−10.91	—	−5.19
FCC	Texture Res.	480 464	−13.40	−8.14	−9.06
FCC	Spatial Res.	427 069	−23.02	−23.02	−19.16
BCC	Spatial Res.	397 805	−29.29	−29.29	−24.70

between the 2-D and 3-D predictions. Furthermore, the sample size is generally best preserved when the "samples" constraint is imposed. The only sample constraint that offers competitive sample size preservation is BCC$_t$. This is possibly due to the fact that its prediction range spans across zero.

We were unable to develop a method to directly measure the observed texture and spatial resolution of the point clouds under the various sampling strategies. However, in each of the three previous experiments, the trends for the texture and spatial resolution constraints followed the predictions established in Section II which leads us to believe that the performance in the spectral domain also follows as predicted.

Next, we tested our LOD generation approach and compared ANPS, ANPD, and sample sizes to directly sampled point clouds. For this experiment, we performed all sampling using an FCC lattice and density-preserving scale factor. Fig. 7 illustrates the resampled LODs compared to their directly sampled counterparts. Fig. 8 illustrates the distribution of

TABLE VII
QUALITY METRICS FOR DIRECTLY SAMPLED VERSUS RESAMPLED POINT CLOUD LODS

LOD (pls/m^2)	Method	ANPS (m)	ANPD (pls/m^2)	Points
30	Direct	0.194	28.99	1 753 740
20	Direct	0.211	19.65	1 232 853
	Resample	0.212	18.63	1 116 874
8	Direct	0.380	7.92	520 072
	Resample	0.383	7.60	527 355
4	Direct	0.538	3.98	259 957
	Resample	0.570	3.97	275 974
2	Direct	0.745	2.00	123 488
	Resample	0.772	1.94	136 745
1	Direct	1.075	1.00	56 610
	Resample	1.099	0.97	66 273

observed densities in the resulting point clouds. Table VII summarizes the point cloud metrics computed for each LOD. In general, ANPD and ANPS track very closely with the predicted values for the FCC$_d$ lattice constraint. We see that, in general, resampled LODs have slightly sparser ANPD and slightly coarser ANPS than their directly sampled counterparts. We expect this result since samples are more likely to be offset further from the lattice constraint sites for resampled data sets. One unexpected result of the experiment was that sample sizes were generally larger for the resampled LODs versus the directly sampled LODs. We expected the sample sizes to track more closely with each other because the number of lattice constraint sites is identical between the directly sampled and resampled LODs. This result may be representative of an aliasing artifact or it may simply be due to the fact that resampled points are not subjected to further filtering, unlike their directly sampled counterparts as detailed in Section IV.

Next, to demonstrate algorithm run-time scaling and application to various point cloud source types, we ran our LOD generation approach against additional data sets of 20-pls/m^2 GmAPD LiDAR and 24-pls/m^2 linear-mode LiDAR. The GmAPD LiDAR data set contains nearly 35 million points over approximately 0.58 km^2. It is a single tile from an aerial mapping survey collected as part of the North-East Illinois (NEIL) pilot project provided by Harris Corporation [34]. The GmAPD product is derived from double overlapping swaths of elliptically scanned raw data. The linear-mode data set contains nearly 124 million points over approximately 2.32 km^2. It is part of an aerial mapping survey over Hillborough County, FL collected for Southwest Florida Water Management District. The linear product is derived from an aggregation of primarily single swath coverage collected with a Riegl VQ-1560i airborne laser scanning (ALS) which collects two crossing scan lines simultaneously. These data sets were selected to illustrate both processing time scaling with respect to input data size as well as consistent performance in data reduction in resultant LODs.

Fig. 9 compares the results of down-sampling both data sets to the coarsest generated LOD of 1 pls/m^2 versus the respective native samplings. Both data sets are very dense

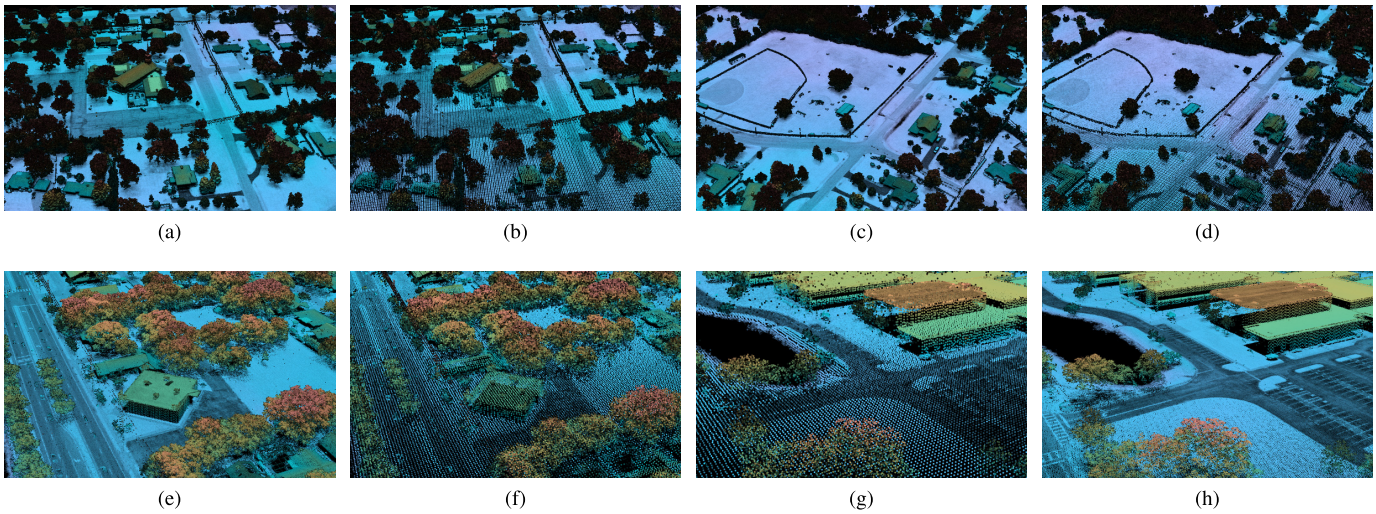


Fig. 9. GmAPD and linear LiDAR products showing original and preserved structures at native sampling and resampled to 1 pls/m^2. (Top row) GmAPD products. (Bottom row) Linear-mode products. (a) and (b) Preservation of steep vertical features like the church steeple, power lines and poles, and tree structures. (c) and (d) Preservation of fine features like the playground, swing set, and faint power lines. (e)–(h) Similar preservation of fine and vertical features in linear mode data where vertical sampling may be significantly coarser than horizontal due to line of sight (LOS) obliquity.

TABLE VIII
RESAMPLING RUN TIMES BY LOD

Source Type & ANPD	In Points	LOD (pls/m^2)	Run Time (s)	Out Points
GmAPD 30 pls/m^2	1 691 329	20	19.74	1 116 874
		8	15.52	527 355
		4	10.99	275 974
		2	10.23	136 745
		1	9.58	66 273
GmAPD 20 pls/m^2	34 985 652	8	307.02	14 116 733
		4	238.85	7 683 286
		2	209.05	3 987 454
		1	189.63	1 966 431
Linear 24 pls/m^2	123 990 155	8	1049.66	47 028 797
		4	826.01	29 004 193
		2	695.52	16 625 328
		1	630.44	8 970 506

at their native sampling with sufficient obliquity to capture vertical structures like building sides, power poles and lines, and fences. These examples show that our approach provides a natural sampling of the scene in both horizontal and vertical, preserving structures in 3-D even when down-sampling very coarsely with respect to native product sampling.

Table VIII summarizes the run times required to generate each LOD in a single-threaded process on a 3.1-GHz Red Hat Enterprise Linux (RHEL) system. The run times appear to at least meet the predicted growth from Section II-J. As expected, run times are largely dominated by the time required to spatially index and iterate over the input point cloud. This leads to largely linear growth in run time with respect to input point cloud size with run times slightly improving for coarser LODs.

The timing results also inform the design of parallel processing approaches. While the majority of our processing steps are trivially parallelizable, there are two aspects of our algorithm that do not readily support parallelization. The first challenge to parallelizing our approach is spatially indexing the input data. While possible, developing a parallel approach to building the spatial index is nontrivial. However, the larger barrier to parallelizing our approach is in Algorithm 5. The loop dependence on the candidate sample set in this algorithm would need to be removed to make any parallelization viable. The most straightforward approach to resolving this issue is to parallelize via a simple divide and conquer approach. By performing an initial spatial partitioning of the input data into equal areas, each area can be processed independently by separate threads with a trivial merge to consolidate the individual result sets. The expected speed gains from such an approach should scale approximately with the number of threads dedicated to the parallel processing.

VII. SUMMARY

The sampling rates for LiDAR point clouds may be highly variable depending on the desired application domain and support needed for downstream analysis. To date, LiDAR specifications and studies have focused on two primary metrics for quantifying point cloud sampling rates: NPS and NPD. These metrics focus only on horizontal sampling rates and provide little guidance for vertical sampling rates or optimizing point cloud sizes while maintaining the desired sampling support.

In this article, we present several approaches for extending the inherently 2-D concepts of NPS and NPD to 3-D sampling via LCS. Furthermore, we demonstrate how the sampling lattices may be scaled to preserve several desirable traits of an SC lattice while potentially offering greater sampling efficiency or robustness to voids.

We present two forms of LCS that offer subtle differences in the preservation of fine feature details in point clouds. We also

address the primary sampling artifact that manifests with LCS and provide a mitigation strategy.

Finally, we demonstrate how LCS may be employed to generate point cloud LODs with a specialized approach given for single-photon and GmAPD LiDAR systems.

We believe that there is potential work remaining to refine the measured performance of LiDAR systems with respect to the sampling rates observed in their derivative products. Specifically, there needs to be greater consensus on the definitions of NPS and NPD and the methods used to assess these metrics. Furthermore, while we were able to propose lattice constraints that would preserve texture and spatial resolution with greater sampling efficiency than an SC lattice, we have yet to establish a methodology for validating the performance of these constraints.

Finally, we believe there are potential applications of the LOD approaches presented in this article remaining to be explored. For example, given the incredible size of LiDAR data holdings, especially for high-resolution data sets, there may be applications related to data streaming, indexing, and hierarchical processing that are enabled by the LOD generation approaches we present.

ACKNOWLEDGMENT

The authors would like to thank L3Harris Technologies, Inc., for use of their GmAPD LiDAR data in support of the analysis of this article.

REFERENCES

[1] D. L. Rabine, J. L. Bufton, and C. R. Vaughn, "Development and test of a raster scanning laser altimeter for high resolution airborne measurements of topography," in *Proc. IGARSS*, vol. 1, May 1996, pp. 423–426, doi: 10.1109/IGARSS.1996.516361.

[2] M. S. Renslow, Ed., *Manual Airborne Topographic LiDAR*. Bethesda, MD, USA: ASPRS, 2012.

[3] M. T. Čekada, F. Crosilla, and M. K. Fras, "Theoretical LiDAR point density for topographic mapping in the largest scales," *Geodetski Vestnik*, vol. 54, no. 3, pp. 403–416, 2010, doi: 10.15292/geodetski-vestnik.2010.03.403-416.

[4] G. T. Raber, J. R. Jensen, M. E. Hodgson, J. A. Tullis, B. A. Davis, and J. Berglund, "Impact of LiDAR nominal post-spacing on DEM accuracy and flood zone delineation," *Photogramm. Eng. Remote Sens.*, vol. 73, no. 7, pp. 793–804, Jul. 2007, doi: 10.14358/PERS.73.7.793.

[5] T. E. Chow and M. E. Hodgson, "Effects of lidar post-spacing and DEM resolution to mean slope estimation," *Int. J. Geograph. Inf. Syst.*, vol. 23, pp. 1277–1295, Sep. 2009, doi: 10.1080/13658810802344127.

[6] A. B. Turner, J. D. Colby, R. M. Csontos, and M. Batten, "Flood modeling using a synthesis of multi-platform lidar data," *Water*, vol. 5, no. 4, pp. 1533–1560, Sep. 2013, doi: 10.3390/w5041533.

[7] J. Vukomanovic, K. K. Singh, A. Petrasova, and J. B. Vogler, "Not seeing the forest for the trees: Modeling exurban viewscapes with LiDAR," *Landscape Urban Planning*, vol. 170, pp. 169–176, Feb. 2018, doi: 10.1016/j.landurbplan.2017.10.010.

[8] M. Magnusson, J. E. S. Fransson, and J. Holmgren, "Effects on estimation accuracy of forest variables using different pulse density of laser data," *Forest Sci.*, vol. 53, no. 6, pp. 619–626, 2007, doi: 10.1093/forestscience/53.6.619.

[9] J. Vauhkonen, T. Tokola, M. Maltamo, and P. Packalén, "Effects of pulse density on predicting characteristics of individual trees of Scandinavian commercial species using alpha shape metrics based on airborne laser scanning data," *Can. J. Remote Sens.*, vol. 34, pp. S441–S459, Nov. 2008, doi: 10.5589/m08-052.

[10] J. Balsa-Barreiro and J. L. Lerma, "Empirical study of variation in lidar point density over different land covers," *Int. J. Remote Sens.*, vol. 35, no. 9, pp. 3372–3383, May 2014, doi: 10.1080/01431161.2014.903355.

[11] L. Ruiz, T. Hermosilla, F. Mauro, and M. Godino, "Analysis of the influence of plot size and LiDAR density on forest structure attribute estimates," *Forests*, vol. 5, no. 5, pp. 936–951, May 2014, doi: 10.3390/f5050936.

[12] M. S. Watt, A. Meredith, P. Watt, and A. Gunn, "The influence of LiDAR pulse density on the precision of inventory metrics in young unthinned Douglas-fir stands during initial and subsequent LiDAR acquisitions," *New Zealand J. Forestry Sci.*, vol. 44, no. 1, pp. 1–9, Aug. 2014, doi: 10.1186/s40490-014-0018-3.

[13] E. H. Hansen, T. Gobakken, and E. Næsset, "Effects of pulse density on digital terrain models and canopy metrics using airborne laser scanning in a tropical rainforest," *Remote Sens.*, vol. 7, no. 7, pp. 8453–8468, Jun. 2015, doi: 10.3390/rs70708453.

[14] S. Adnan, M. Maltamo, D. A. Coomes, and R. Valbuena, "Effects of plot size, stand density, and scan density on the relationship between airborne laser scanning metrics and the Gini coefficient of tree size inequality," *Can. J. Forest Res.*, vol. 47, no. 12, pp. 1590–1602, Dec. 2017, doi: 10.1139/cjfr-2017-0084.

[15] M. Helbich, A. Jochem, W. Mücke, and B. Höfle, "Boosting the predictive accuracy of urban hedonic house price models through airborne laser scanning," *Comput., Environ. Urban Syst.*, vol. 39, pp. 81–92, May 2013, doi: 10.1016/j.compenvurbsys.2013.01.001.

[16] Y. Wang *et al.*, "Three-dimensional reconstruction of building roofs from airborne lidar data based on a layer connection and smoothness strategy," *Remote Sens.*, vol. 8, no. 5, p. 415, 2016, doi: 10.3390/rs8050415.

[17] D. Chen, R. Wang, and J. Peethambaran, "Topologically aware building rooftop reconstruction from airborne laser scanning point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7032–7052, Dec. 2017, doi: 10.1109/TGRS.2017.2738439.

[18] K. Chen, W. Lu, F. Xue, P. Tang, and L. H. Li, "Automatic building information model reconstruction in high-density urban areas: Augmenting multi-source data with architectural knowledge," *Automat. Construct.*, vol. 93, pp. 22–34, Sep. 2018, doi: 10.1016/j.autcon.2018.05.009.

[19] S. A. N. Gilani, M. Awrangjeb, and G. Lu, "Segmentation of airborne point cloud data for automatic building roof extraction," *GISci. Remote Sens.*, vol. 55, no. 1, pp. 63–89, Jan. 2018, doi: 10.1080/15481603.2017.1361509.

[20] E. P. Baltsavias, "Airborne laser scanning: Basic relations and formulas," *ISPRS J. Photogramm. Remote Sens.*, vol. 54, no. 2, pp. 199–214, Jul. 1999, doi: 10.1016/S0924-2716(99)00015-5.

[21] H. K. Heidemann, *Lidar Base Specification (Version 1.2)*. Reston, VA, USA: United States Geological Survey, Nov. 2014, pp. 4–6, doi: 10.3133/tm11B4.

[22] M. Isenburg. *Point Density and Point Spacing*. Accessed: Feb. 3, 2017. [Online]. Available: http://felix.rohrba.ch/en/2015/point-density-and-point-spacing/

[23] ASPRS Map Accuracy Standards Working Group, "ASPRS positional accuracy standards for digital geospatial data," *Photogramm. Eng. Remote Sens.*, vol. 81, no. 3, pp. A1–A26, Mar. 2015, doi: 10.14358/PERS.81.3.A1-A26.

[24] D. P. Petersen and D. Middleton, "Sampling and reconstruction of wave-number-limited functions in *N*-dimensional Euclidean spaces," *Inf. Control*, vol. 5, no. 4, pp. 279–323, Dec. 1962, doi: 10.1016/S0019-9958(62)90633-2.

[25] D. P. Petersen and D. Middleton, "Reconstruction of multidimensional stochastic fields from discrete measurements of amplitude and gradient," *Inf. Control*, vol. 7, no. 4, pp. 445–476, Dec. 1964, doi: 10.1016/S0019-9958(64)90165-2.

[26] V. Bardan, "The Petersen–Middleton theorem and sampling of seismic data," *Geophys. Prospecting*, vol. 57, no. 5, pp. 823–834, Sep. 2009, doi: 10.1111/j.1365-2478.2008.00752.x.

[27] K. Mueller and F. Xu, *Optimal Sampling Lattices for High-Fidelity CT Reconstruction*, B. Yu, Ed. Piscataway, NJ, USA: IEEE, Oct. 2009, pp. 3048–3052, doi: 10.1109/NSSMIC.2009.5401575.

[28] S. H. Izen, "Generalized sampling expansion on lattices," *IEEE Trans. Signal Process.*, vol. 53, no. 6, pp. 1949–1963, Jun. 2005, doi: 10.1109/TSP.2005.847841.

[29] K. L. Damkjer, "Point cloud sampling and resampling strategies," presented at the ASPRS Annu. Conf., Denver, CO, USA. Bethesda, MD, USA: ASPRS, Jan. 2019. [Online]. Available: http://www.asprs.org/a/conferences/proceedings/2019/24P_380_Damkjer_Poster.pdf

[30] V. Vad, B. Csébfalvi, P. Rautek, and E. Gröller, "Towards an unbiased comparison of CC, BCC, and FCC lattices in terms of prealiasing," *Comput. Graph.*, vol. 33, no. 3, pp. 81–90, Jun. 2014, doi: 10.1111/cgf.12364.

[31] P. N. Yianilos, "Data structures and algorithms for nearest neighbor search in general metric spaces," in *Proc. SODA*, 1993, pp. 311–321.

[32] J. M. Stoker, "Volumetric visualization of multiple-return lidar data: Using voxels," *Photogramm. Eng. Remote Sens.*, vol. 75, no. 2, pp. 109–112, 2009.

[33] T. Naus. Unbiased LiDAR data measurement (draft), Fugro Horizons, 2009. [Online]. Available: http://www.asprs.org/wp-content/uploads/2010/12/Unbiased_measurement.pdf

[34] Harris Corporation. *Geiger-Mode LiDAR—Geospatial Data & Imagery|Harris Geospatial*. Accessed: Feb. 8, 2018. [Online]. Available: http://www.harrisgeospatial.com/DataImagery/Geiger-mode LiDAR.aspx

Kristian L. Damkjer (Member, IEEE) received the B.S. degree in computer and information science and the M.S. degree in digital arts and sciences from the University of Florida, Gainesville, FL, USA, in 2001 and 2003, respectively, and the M.S. degree in modeling and simulation from the University of Central Florida, Orlando, FL, USA, in 2015, where he is currently pursuing the Ph.D. degree in modeling and simulation.

From 2003 to 2005, he was an Adjunct Lecturer with the University of Florida. Since 2005, he has been a Software Engineer with L3Harris Technologies, Melbourne, FL, USA, where he is currently a Software Engineering Manager with the Geospatial Division of the Space Superiority and Geospatial Sector of the Space and Airborne Systems Segment. His work and research interests include automated processing of large-scale image mosaics, image registration, automatic tonal balancing, pan-sharpening, elevation extraction from imagery, and GmAPD LiDAR processing systems.

Mr. Damkjer is the American Society for Photogrammetry and Remote Sensing (ASPRS) Member and a Certified Mapping Scientist at LiDAR.

Hassan Foroosh (Senior Member, IEEE) is currently a CAE Link Professor of computer science with the University of Central Florida (UCF), Orlando, FL, USA. He has authored or coauthored over 150 peer-reviewed scientific articles in the areas of computer vision, image processing, and machine learning.

Dr. Foroosh received the Piero Zamperoni Award from the International Association of Pattern Recognition (IAPR) in 2004, the Best Scientific Paper Award from IAPR-International Conference on Pattern Recognition (IAPR-ICPR) in 2008, and the Best Paper Award from IEEE International Conference on Image Processing (ICIP) in 2018. He is also the Principal Investigator and the Lead of the Science Data Center of the NASA GOLD mission that launched a satellite into Earths geo-stationary orbit in 2018 to study the space weather using ultraviolet imaging. He has been serving on the editorial boards and the organizing committees of various IEEE transactions, conferences, and working groups.