MESH-FREE SPARSE REPRESENTATION OF MULTIDIMENSIONAL LIDAR DATA

Kristian L. Damkjer, Hassan Foroosh

University of Central Florida
Department of Electrical Engineering and Computer Science
Orlando, Florida

ABSTRACT

Modern LiDAR collection systems generate very large data sets approaching several million to billions of point samples per product. Compression techniques have been developed to help manage the large data sets. However, sparsifying LiDAR survey data by means other than random decimation remains largely unexplored. In contrast, surface model simplification algorithms are well-established, especially with respect to the complementary problem of surface reconstruction. Unfortunately, surface model simplification algorithms are often not directly applicable to LiDAR survey data due to the true 3D nature of the data sets. Further, LiDAR data is often attributed with additional user data that should be considered as potentially salient information. This paper makes the following main contributions in this area: (i) We generalize some features defined on spatial coordinates to arbitrary dimensions and extend these features to provide local multidimensional statistics. (ii) We propose an approach for sparsifying point clouds similar to mesh-free surface simplification that preserves saliency with respect to the multidimensional information content. (iii) We show direct application to LiDAR data and evaluate the benefits in terms of level of sparsity versus entropy.

Index Terms— LiDAR, multidimensional systems, point cloud, mesh-free simplification, principal component analysis

1. INTRODUCTION

Mapping and surveying Light Detection and Ranging (LiDAR) systems produce large amounts of true three-dimensional (3D) data. Modern systems sample several thousand to over a million points per second resulting in several million to billions of point samples per product to be stored, processed, analyzed and distributed [1, 2, 3].

Managing such large data sets presents a host of challenges to content providers. Production strategies have been developed to mitigate data management issues inherent in processing large-scale projects [4]. However, user demands for simultaneous wide-area coverage, high-fidelity scene content, and low-latency access keep data sizing considerations at the forefront of content provider concerns.

The LAS file format was developed to facilitate the exchange of LiDAR data [5]. Extensions to the LAS format, *e.g.* LASzip, and generic exchange formats, *e.g.* HDF5, further address data sizing concerns by offering support for lossless compression with typical performance yielding files between 10 and 20 percent of the original file size [6, 7]. However, even with an effective compression strategy, explicit data reduction is often necessary to support users in bandwidth-limited and mobile device environments. It is therefore necessary to establish approaches to intelligently reduce point data in a manner that preserves information content. Current approaches focus primarily on preserving the surface structures represented by the spatial coordinates [8]. We describe an approach that also allows for the preservation of non-surface structures and includes point attribution in the salience criterion.

2. NOVELTY AND RELATIONSHIP TO PRIOR WORK

Simplification of LiDAR survey data remains largely unexplored, however point-based surface model simplification algorithms are well-established, especially with respect to the complementary problem of surface reconstruction. We refer to the survey conducted by Pauly *et al.* for an overview of point-based surface simplification [8]. In this problem domain, there is an underlying assumption that points in the cloud all belong to surfaces embedded in the spatial dimensions. This assumption is frequently violated in LiDAR data where points often belong to non-surface features. Further, survey data is often attributed with additional information that should be considered in the simplification process lest salient information be lost [4]. Regardless of these limitations, we draw inspiration for our approach from mesh-free surface simplification approaches.

Dyn *et al.* [9] present an iterative sub-sampling approach supported by local surface approximation. Their approach operates in a fine-to-coarse manner terminated by a desired point set size, τ. Their point selection is solely based on the input point cloud geometry, $\mathcal{P} \subset \mathbb{R}^3$, and a salience criterion, $s\colon \mathcal{T} \subseteq \mathcal{P} \setminus \{\varnothing\} \to \mathbb{R}$. An important aspect of s is that it updates with respect to the current subset $\mathcal{T} \subseteq \mathcal{P}$ throughout the point removal process.

Yu *et al.* [10] present a similar approach that enforces a post-condition of a terminal point set size and operates in an adaptive manner driven by point clustering and a user-specified simplification criteria and optimization process.

While these approaches operate without generating an explicit mesh surface, they carry forward the legacy of mesh-based approaches by limiting their analysis to spatial coordinates and operating under the assumption that points locally approximate a surface. In contrast, natural scenes are complex and contain significant points belonging to linear, planar, and isotropic structures. LiDAR survey data is also frequently attributed with intensity or color data, classification, or other user-defined features. These additional dimensions may contain content that is salient to end-user applications which suggests the need for a multidimensional approach to point removal.

The primary goal of this paper, therefore, is to create a data sparsifying algorithm by developing a multidimensional salience measure, $s\colon \mathcal{P} \to \mathbb{R}$, and therefore demonstrate that such multidimensional approach produces sparse point representations that preserve salience. Several approaches have been developed to identify salient points based solely on 3D spatial coordinates. West *et al.* introduce features based on structure-tensor–eigenvalue analysis of local point

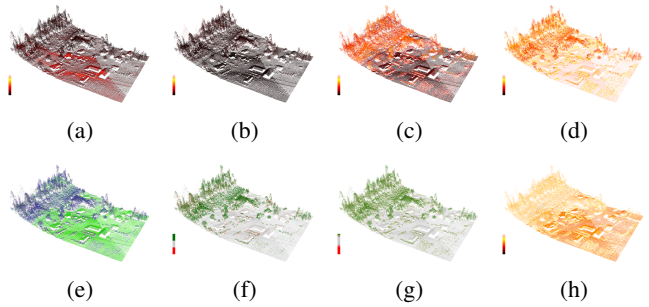


Fig. 1: Visualization of Neighborhood Features on section of *Armstrong/Enderby* data set from Applied Imagery. (a) Density, (b) Omnivariance, (c) Isotropy, (d) Anisotropy, (e) Dimensionality, (f) Dimension Label, (g), Component Entropy, (h) Dimensional Entropy

neighborhoods [11]. These feature descriptors have been enhanced to extract strong spatially linear features to support scene modeling applications [12]. Methods have also been developed to direct optimal neighborhood scale selection for feature attribution [13]. Next, we generalize these attribute definitions to arbitrary dimensions to serve as the basis for measuring salience.

3. LOCAL STATISTIC ATTRIBUTION

Our salience measure is based on attributes defined by neighborhoods in arbitrary dimensions. In this section, we establish our definition for locality in arbitrary dimensions and generalize the definitions for previously-established features in the spatial domain to arbitrary dimensions. Figure 1 illustrates the features we consider based on evaluation of 3D spatial point data.

3.1. Data Conditioning

Our attributes are based on principal components analysis which is sensitive to differences in scale within the feature space. The source data should therefore be conditioned prior to analysis so that different classes of attributes have approximately the same precision scale or measurement resolution. Without this adjustment, insignificant variations within one dimension can easily dominate significant variations in another. We perform this conditioning by first decentering the data then normalizing each class by an estimate of the measurement resolution for the class. We estimate the measurement resolution by computing the standard deviation within a flat response region for each attribute in the class. We then take the minimum class attribute standard deviation as the measurement resolution for the class.

3.2. Locality

We consider the analysis of multidimensional points, $\boldsymbol{x} \in \mathbb{R}^n$, where $\mathcal{N}$ is the set of native attributes for the point and $|\mathcal{N}| = n$ is the dimension of the native feature space. All attributes are assumed to be real-valued. While boolean and finite-class attributes may be simply represented by an appropriate integer enumeration, our approach is unlikely to yield meaningful results with such classes due to the conditioning issues mentioned previously. Our definition of a point cloud, $\mathcal{D} \subset \mathbb{R}^n$, then is simply a database of real-valued multidimensional points with consistent feature space definition.

In most cases, it is desirable to restrict neighborhood definition to a subset of the available native feature space. To support this capability, we establish a database of query points, $\mathcal{Q} \subset \mathbb{R}^m$, where $\mathcal{M} \subseteq \mathcal{N}$ is the search space of attributes for the determination of locality and $|\mathcal{M}| = m$ is the dimension of the search space.

We proceed by analyzing the neighborhoods of points about the query points, $\mathcal{V}_{\boldsymbol{q}} \subseteq \mathcal{D}$. The neighborhoods are defined by an m-dimensional distance metric, δ, between the query points, $\boldsymbol{q} \in \mathcal{Q}$, and the data points, $\boldsymbol{x} \in \mathcal{D}$. For point cloud simplification, we treat each $\boldsymbol{x} \in \mathcal{D}$ as a query location (*i.e.*, $\mathcal{Q} = \mathcal{D}$). This approach requires a reasonable all nearest-neighbor search algorithm to be practical, that is one with complexity no worse than $O(p \log p)$ where $p = |\mathcal{D}|$.

We investigated two neighborhood definitions that each present merits. The k-nearest neighborhood, $\mathcal{V}_{\boldsymbol{q}}^k$, consists of the k closest points to $\boldsymbol{q}$ in $\mathcal{D}$ whereas the fixed-radius neighborhood, $\mathcal{V}_{\boldsymbol{q}}^r$, consists of all points in $\mathcal{D}$ within the ball of radius r centered at $\boldsymbol{q}$. Similar to Dyn *et al.*, we enforce the condition that $\boldsymbol{q} \notin \mathcal{V}_{\boldsymbol{q}}$ [9]. This condition is imposed so that $\mathcal{V}_{\boldsymbol{q}}$ can be used to estimate the effects of eliminating $\boldsymbol{q}$ during the simplification process.

3.3. Structure Features

West *et al.* and Demantké *et al.* define several features for describing 3D point neighborhoods. In this section, we generalize, and in some cases modify, their proposed features to support multidimensional analysis and interpretability. The generalized features are summarized in table 1.

Name	Equation	
Omnivariance	$Omni: \left(\prod_{d=1}^{n} \lambda_d\right)^{\frac{1}{n}}$	(1)
Isotropy	$Iso: \frac{\sigma_n}{\sigma_1}$	(2)
Anisotropy	$Ani: \frac{\sigma_1 - \sigma_n}{\sigma_1}$	(3)
Dimensionality	$\alpha_d: \begin{cases} \frac{\sigma_d - \sigma_{d+1}}{\sigma_1} & , d < n \\ Iso & , d = n \end{cases}$	(4)
Dimension Label	$d^*: \underset{d \in \{1,\ldots,n\}}{\operatorname{argmax}} \alpha_d$	(5)
Component Entropy	$H_\sigma: -\sum_{d=1}^{n} \hat{\sigma}_d \log_n \hat{\sigma}_d$	(6)
Dimensional Entropy	$H_\alpha: -\sum_{d=1}^{n} \alpha_d \log_n \alpha_d$	(7)

Table 1: Features defined on $\mathcal{V}_{\boldsymbol{q}}$

West *et al.* present six features that proved to be most applicable to their work in segmentation and object recognition: *omnivariance*, *anisotropy*, *linearity*, *planarity*, *sphericity*, and *eigenentropy* [11]. Each of the features they describe are derived from the eigenvalues resulting from the principal components analysis of the query neighborhoods, $\mathcal{V}_{\boldsymbol{q}}$. However, while West *et al.* define the features with respect to the eigenvalues, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, we generally prefer to use the singular values, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$, as demonstrated by Demantké *et al.* [13]. The sole exception to this recommendation is the *omnivariance* feature which is used to meaningfully compare the total variance of the neighborhoods to each other. Re-defining

the feature with respect to the singular values, while still meaningful, would be more directly related to the standard deviation.

Linearity, *planarity*, and *sphericity* are closely related features that each represent the concept of the neigborhood's participation in subsequently higher dimensions. That is, the values attempt to capture the degree to which the local neighborhood spreads into each of the respective dimensions [11]. We generalize this concept as *dimensionality* and define the family of features by equation (4). We feel that it is worth considering the highest order dimensionality of the data set as a unique feature as well and generalize the concept to *isotropy* as defined by equation (2). The complement of this value, *anisotropy*, is thus easily understood and maintains a definition consistent with West *et al.* as expressed by equation (3)

Eigenentropy is a feature based on the Shannon entropy of the principal component eigenvalues. It describes the dimensional participation of the neighborhood. That is, higher values imply greater participation across more of the available dimensions [11]. We generalize this feature by modifying the logarithmic base to the number of dimensions, n, and operating on normalized singular values, $\hat{\sigma}_d$, instead of raw eigenvalues. We normalize the singular values by the sum over all singular values for the neighborhood so that each value can be treated as a probability that a point in the neighborhood has the respective eigenvector as its dominant local coordinate axis. The resulting feature, which we call *component entropy*, describes the unpredictability of the neighborhood in the n-dimensional space and is expressed by equation (6).

Demantké *et al.* introduce two additional features to support automated neighborhood scale selection: *dimensionality labeling* and *dimensional entropy* [13]. The *dimension label* is simply the dimension that maximizes equation (4). We use this feature to establish an equivalence relation on $\mathcal{D} \times \mathcal{D}, \boldsymbol{x} \sim \boldsymbol{y} \iff d^*(\mathcal{V}_{\boldsymbol{x}}) = d^*(\mathcal{V}_{\boldsymbol{y}})$. This equivalence relation creates a partition on $\mathcal{D}$ that we leverage as part of our simplification algorithm as described in section 4. *Dimensional entropy* is very similar in concept to the *component entropy*, with the exception that it describes the Shannon entropy of the *dimensionality* feature. This feature describes the unpredictability of the *dimension label* feature and acts as a figure of merit for the selected label.

4. APPROACH

In this section, we describe a general point cloud sparsifying algorithm, derive the multidimensional salience measure, and describe the update operations that must take place per iteration to enforce the correct dynamic behavior of the salience measure. Algorithm 1 describes our solution that supports sparsifying points in arbitrary dimensions. Our objective is to remove least salient points, while preserving the proportional distribution of dimension labels in the final point set. We also wish to maintain the behavior that the algorithm computes a unique nested sequence of subsets that can be used to define a multiresolution model.

The dimensional partitioning at line 3 of algorithm 1 is simply achieved by segregating points according to equivalence relation established by equation (5). This partitioning only happens once to establish the apparent local dimension of the point neighborhoods. Points are not moved out of their initial partition, regardless of how their descriptive features evolve through the sparsifying process.

We simultaneously enforce the proportional sparsifying constraint and the nested subset constraint by removing points from the partitions in an interleaved manner. We order the partitions so that $|\mathcal{M}_1| \geq \cdots \geq |\mathcal{M}_n|$. The pre-computed priorities for each partition are given by equation (8) where $M = \max_{d \in \{1,\ldots,n\}} |\mathcal{M}_d|$.

Algorithm 1 Multidimensional Point Cloud Simplification

Require: $\mathcal{D} \subset \mathbb{R}^n \setminus \{\varnothing\}, |\mathcal{D}| = N, \tau \in \mathbb{Z}_N$
Ensure: $\mathcal{T} \subset \mathcal{D}, |\mathcal{T}| = \tau$
1: **function** MULTIDIMREMOVEPOINTS($\mathcal{D}, \tau$)
2: $\quad\mathcal{T} \leftarrow \mathcal{D}$
3: $\quad\mathcal{M} \leftarrow \mathcal{D}/\sim$
4: $\quad$**while** $|\mathcal{T}| > \tau$ **do**
5: $\quad\quad d^* \leftarrow \underset{d \in \{1,\ldots,n\}}{\operatorname{argmin}} \min \mathcal{P}_d$
6: $\quad\quad \mathcal{P}_{d^*} \leftarrow \mathcal{P}_{d^*} \setminus \{\min \mathcal{P}_{d^*}\}$
7: $\quad\quad \boldsymbol{x}^* \leftarrow \underset{\boldsymbol{x} \in \mathcal{M}_{d^*}}{\operatorname{argmin}} s(\boldsymbol{x})$
8: $\quad\quad \mathcal{M}_{d^*} \leftarrow \mathcal{M}_{d^*} \setminus \{\boldsymbol{x}^*\}$
9: $\quad\quad \mathcal{T} \leftarrow \mathcal{T} \setminus \{\boldsymbol{x}^*\}$
10: $\quad$**return** $\mathcal{T}$

$$\mathcal{P}_d = \left\{ \frac{mM}{|\mathcal{M}_d|} + \frac{d-1}{n} : \forall m \in 1, \ldots, |\mathcal{M}_d| \right\} \tag{8}$$

In each iteration, we seek to select the point that minimizes the change of information content in the point cloud. Dyn *et al.* use a salience measure that increases in value as points in the neighborhood diverge from the local fit of a smoothed surface [9]. Obviously, we are unable to use a similar model for salience since our measure must be defined for arbitrary dimension. However, recall from section 3.3 that equation 6 describes the unpredictability of the neighborhood and acts as a measure of information content in the local neighborhood. We therefore select this feature, which is defined for arbitrary dimension, as the basis for our salience measure.

To estimate the change of information content caused by the removal of a point, we first establish a baseline estimate. The baseline, $H_{\sigma,0}$, is based on the *component entropy* of the initial point neighborhoods as described by equation (9).

$$H_{\sigma,0}(\boldsymbol{x}) = H_\sigma(\mathcal{V}_{\boldsymbol{x}} \cup \{\boldsymbol{x}\}) \tag{9}$$

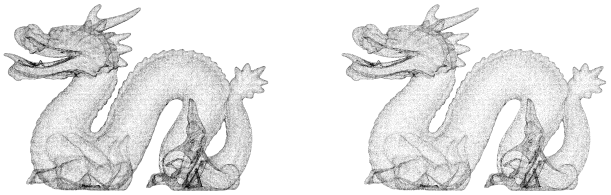
We estimate the change of information content caused by the removal of a point as the maximum absolute deviation of the neighborhood component entropy from the component baselines as described by equation (10). This measure acts as the salience function for our sparsifying process.

$$s(\boldsymbol{x}) = \max_{\boldsymbol{y} \in \mathcal{C}_{\boldsymbol{x}}} |H_{\sigma,0}(\boldsymbol{y}) - H_\sigma(\mathcal{V}_{\boldsymbol{x}})| \tag{10}$$

In each iteration, the point, $\boldsymbol{x}^*$, that minimizes equation (10) is selected for removal. To ensure that removed points continue to influence the sparsifying process, we maintain a constituency, $\mathcal{C}_{\boldsymbol{x}}$, for each $\boldsymbol{x} \in \mathcal{D}$. The constituency sets serve an identical function to the test sets described by Dyn *et al.* and are updated in a similar manner [9].

The constituency contains the set of points represented by $\boldsymbol{x}$. Initially, each point represents only itself, *i.e.* $\mathcal{C}_{\boldsymbol{x}} = \{\boldsymbol{x}\}$. When a point, $\boldsymbol{x}^*$, is selected for removal, its constituency, $\mathcal{C}_{\boldsymbol{x}^*}$, is distributed among its neighbors' constituencies, $\{\mathcal{C}_{\boldsymbol{y}} : \boldsymbol{y} \in \mathcal{V}_{\boldsymbol{x}^*}\}$, by selecting the closest $\boldsymbol{y} \in \mathcal{V}_{\boldsymbol{x}^*}$ as a representative for each $\boldsymbol{z} \in \mathcal{C}_{\boldsymbol{x}^*}$.

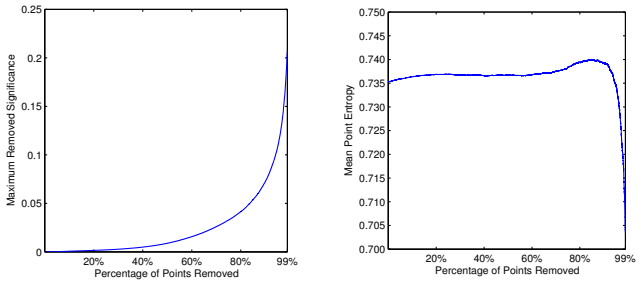
In addition to updating the constituencies, we must also update the neighborhoods containing the removed point to make sure that it does not continue to influence estimates of the current point cloud state. The set of back-references to the neighborhoods containing each point, $\mathcal{B}_{\boldsymbol{x}} = \{\mathcal{V}_{\boldsymbol{y}} : \boldsymbol{x} \in \mathcal{V}_{\boldsymbol{y}}\}$, are maintained to keep this update operation efficient. The neighborhoods containing the removed



(a) 75%, $\overline{H}_{\sigma,0} \approx 0.7369$ (b) 50%, $\overline{H}_{\sigma,0} \approx 0.7367$

(c) 25%, $\overline{H}_{\sigma,0} \approx 0.7384$ (d) 10%, $\overline{H}_{\sigma,0} \approx 0.7393$

Fig. 2: Data output by our approach on *Dragon* from the Stanford 3D Scanning Repository



(a) Least Salience (b) Model Entropy

Fig. 3: Least salience and mean baseline entropy trends during simplification of *Dragon* to 1% of the original point cloud size

point replace it with a new closest point from their neighbor's neighborhoods. That is, from the set

$$\bigcup_{\boldsymbol{z} \in \mathcal{V}_{\boldsymbol{y}}} \mathcal{V}_{\boldsymbol{z}} \setminus (\{\boldsymbol{x}^*\} \cup \mathcal{V}_{\boldsymbol{y}}) \tag{11}$$

If the set described by equation (11) is empty, a closest point from the current set of remaining points is selected instead.

Finally, the salience measures for each $\boldsymbol{x} \in \mathcal{V}_{\boldsymbol{x}^*} \cup \mathcal{B}_{\boldsymbol{x}^*}$ are updated according to equation (10).

5. RESULTS AND DISCUSSION

We have implemented our approach using *vantage point* tree [14] for the spatial indexing structure in support of all nearest neighbor searching and *splay* tree [15] for managing the salience heap. The selection of these data structures maintains aymptotic complexity equivalent to the approach proposed by Dyn *et al.* while compensating for higher dimensional data.

To illustrate the effectiveness of our approach, we first applied our algorithm to the standard *Dragon* data set from the Stanford 3D scanning repository which contains only spatial coordinates with no additional attribution. Figure 2 shows results for data sparsified to 75%, 50%, 25% and 10% of the original point cloud size, $|\mathcal{D}| = 435545$. This test case demonstrates that our approach produces a sparse representation of the original data that preserves features that are salient with respect to representing the original surface. Figure 3 illustrates the behavior of the algorithm during the sparsifying process. The salience measure does not increase monotonically throughout the sparsifying process since the point removal and update process does not enforce any guarantees on the entropies of the affected neighborhoods. However, figure 3a illustrates that the least salience trend increases monotonically throughout the sparsifying process. Figure 3b illustrates the effect of our salience measure on the mean baseline entropy for the model. Since we define salience to minimize change in entropy, the mean entropy remains very flat through most of the sparsifying process and in fact increases slightly as redundant points are removed. However, there is a point beyond which significant points are removed and mean entropy drops sharply as a result. For the *Dragon* test case, this occurs once approximately 90% of the original points have been removed.

(a) 100% (b) 10% (c) 1%

Fig. 4: Mesh reconstruction from data output by our approach on *Vellum Manuscript* from the Stanford 3D Scanning Repository

Next, to illustrate the effectiveness of our approach on multidimensional data, we applied our algorithm to the *Vellum Manuscript* data set from the Stanford 3D scanning repository which contains spatial coordinates with color attribution per point. Figure 4 shows mesh reconstructions of data sparsified to 100%, 10%, and 1% of the original point cloud size, $|\mathcal{D}| = 2155617$. This test case demonstrates that our salience measure generalizes to multidimensional data. The example illustrates preservation of fine features in the nD data set up to high levels of sparsity. The thin red margin lines are visible and paper edges are preserved even when data is sparsified to just 1% of the original data size. Our approach is lossy, though, and significant degradation is noticeable at the 1% level. However, we are able to create a very faithful reconstruction of the data set with just 10% of the original data.

6. CONCLUSION AND FUTURE WORK

In this paper, we develop extensions of established 3D features to arbitrary dimensions and present an application to sparse representation of point clouds. We believe that this approach may be further enhanced by better selecting the initial neighborhood sizes using an approach such as the one proposed by Demantké *et al.* [13] We also believe that there are other potentially interesting applications of these features that warrant investigation, for example as features that support correlation and registration algorithms.

7. REFERENCES

[1] Christopher E. Parrish, "Full-waveform lidar," In Renslow [16], chapter 2.4, pp. 54–61.

[2] Philip W. Smith, "Geiger mode lidar," In Renslow [16], chapter 2.6, pp. 91–97.

[3] Jamie Young, "Key elements of ALS technology," In Renslow [16], chapter 2.2, pp. 17–37.

[4] Nicolas David, Clément Mallet, and Frédéric Bretar, "Library concept and design for LiDAR data processing," in *GEOBIA 2008 - Pixels, Objects, Intelligence. GEOgraphic Object Based Image Analysis for the 21st Century*, Geoffrey J. Hay, Thomas Blaschke, and Danielle Marceau, Eds., Calgary, Alberta, Canada, Aug. 2008, ISPRS, vol. XXXVIII-4/C1 of *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, University of Calgary, Calgary, Alberta, Canada.

[5] American Society for Photogrammetry and Remote Sensing, Bethesda, MD, USA, *LAS Specification Version 1.4 R12*, 2012.

[6] Martin Isenburg, "LASzip: lossless compression of LiDAR data," in *Proceedings of the 2011 European LiDAR Mapping Forum*, Salzburg, Austria, 2011, Intelligent Exhibitions Limited, pp. 1–9.

[7] The HDF Group, "Hierarchical data format version 5," http://www.hdfgroup.org/HDF5/, 2000-2014, [February 12, 2014].

[8] Mark Pauly, Markus Gross, and Leif P. Kobbelt, "Efficient simplification of point-sampled surfaces," in *Proceedings of the Conference on Visualization '02*, Washington, DC, USA, 2002, VIS '02, pp. 163–170, IEEE Computer Society.

[9] Nira Dyn, Armin Iske, and Holger Wendland, "Meshfree thinning of 3D point clouds," *Foundations of Computational Mathematics*, vol. 8, no. 4, pp. 409–425, 2008.

[10] Zhiwen Yu, Hau-San Wong, Hong Peng, and Qianli Ma, "ASM: An adaptive simplification method for 3d point-based models," *Computer-Aided Design*, vol. 42, no. 7, pp. 598–612, 2010.

[11] Karen F. West, Brian N. Webb, James R. Lersch, Steven Pothier, Joseph M. Triscari, and A. Evan Iverson, "Context-driven automated target detection in 3D data," in *Proceeding of SPIE*, Firooz A. Sadjadi, Ed., Bellingham, WA, 2004, SPIE, vol. 5426, pp. 133–143, SPIE.

[12] Hermann Gross and Ulrich Thoennessen, "Extraction of lines from laser point clouds," in *Symposium of ISPRS Commission III Photogrammetric Computer Vision PCV '06*, Wolfgang Förstner and Richard Steffen, Eds., Göttingen, Germany, 2006, ISPRS, vol. XXXVI-3 of *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 86–91, Copernicus Publications.

[13] Jérôme Demantké, Clément Mallet, Nicolas David, and Bruno Vallet, "Dimensionality based scale selection in 3D LIDAR point clouds," in *WG V/3, I/3, I/2, III/2, III/4, VII/7, V/1 ISPRS Workshop Laser Scanning 2011*, Derek D. Lichti and Ayman F. Habib, Eds., Göttingen, Germany, Aug. 2011, ISPRS, vol. XXXVIII-5/W12 of *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 97–102, Copernicus Publications.

[14] Peter N. Yianilos, "Data structures and algorithms for nearest neighbor search in general metric spaces," in *Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, Philadelphia, PA, USA, 1993, Association for Computing Machinery and Society for Industrial and Applied Mathematics, SODA '93, pp. 311–321, Society for Industrial and Applied Mathematics.

[15] Daniel Dominic Sleator and Robert Endre Tarjan, "Self-adjusting binary search trees," *J. ACM*, vol. 32, no. 3, pp. 652–686, July 1985.

[16] Michael S. Renslow, Ed., *Manual of Airborne Topographic Lidar*, American Society for Photogrammetry and Remote Sensing, Bethesda, MD, USA, 2012.