STAG: Smart Tools and Applications in Graphics (2021)
P. Frosini, D. Giorgi, S. Melzi, and E. Rodolà (Editors)

IMGD: Image-based Multiscale Global Descriptors of Airborne LiDAR Point Clouds Used for Comparative Analysis

J. Sreevalsan-Nair[†1] and P. Mohapatra[1] and S. Singh[1]

[1]Graphics-Visualization-Computing Lab, International Institute of Information Technology Bangalore, India

Abstract

Both geometric and semantic information are required for a complete understanding of regions acquired as three-dimensional (3D) point clouds using the Light Detection and Ranging (LiDAR) technology. However, the global descriptors of such datasets that integrate both the information types are rare. With a focus on airborne LiDAR point clouds, we propose a novel global descriptor that transforms the point cloud from Cartesian to barycentric coordinate spaces. We use both the probabilistic geometric classification, aggregated from multiple scales, and the semantic classification to construct our descriptor using point rendering. Thus, we get an image-based multiscale global descriptor, IMGD. To demonstrate its usability, we propose the use of distribution distance measures between the descriptors for comparing the point clouds. Our experimental results demonstrate the effectiveness of our descriptor, when constructed of publicly available datasets, and on applying our selected distance measures.

Keywords: Airborne LiDAR point clouds, Local geometric descriptors, Global descriptor, Classification, Barycentric coordinates, Visualization, Shannon entropy, Distribution distance measure, Uncertainty analysis, Covariance tensor, Tensor voting

CCS Concepts

• **Computing methodologies** → **Image representations;** • **Human-centered computing** → **Visualization techniques;** • **Information systems** → **Nearest-neighbor search;**

1. Introduction

Topographic Light Detection and Ranging (LiDAR) technology provides three-dimensional (3D) point clouds, which are used for extracting geometry for the 3D reconstruction of regions. The point clouds provide rich geometric information, that when integrated with imagery, provides scene understanding from observed data [Rot09]. The objects in a region are effectively extracted from a point cloud through segmentation and semantic classification, efficiently implemented using supervised and deep learning [WJHM15]. The geometric information of the point clouds is already extracted in the form of local geometric descriptors and their eigenvalue-based features. These features are used for semantic classification. Also, we can now ascertain how the uncertainty in the geometry manifests itself in the semantic classification, using entropy measures [SNM20].

However, a pertinent gap exists in comparing point clouds based on both semantic and geometric information. How can we compare two regions with similar semantic compositions but with differences in spatial layout? For instance, a planned residential area inclusive of substantial tree cover, and a forest reserve area in the proximity of a set of high-rise buildings, will have low class-distribution distances, but high geometric distances. Yet another example is the case of different compositions in regions, of trees and high-rise buildings, which tend to confuse semantic classifiers owing to their geometric similarities [RSJ*12]. We get high class-distribution distances in this case, but low geometric distances. Thus, we motivate that an integrated distance measure is required to bring out multifaceted differences between regions in a holistic manner. This brings forth the need for constructing a global descriptor for point clouds, that integrates both the geometric and semantic information. In this work, we address these gaps using a novel image-based multiscale global descriptor, IMGD, $\mathscr{S}_{Gm}$ (Figure 1).

Here, we focus on airborne LiDAR or Aerial Laser Scanning (ALS) point clouds. Our contributions are:

- A novel global descriptor, IMGD for ALS point clouds, that encode both semantic and geometric classification information,
- Identification of appropriate distribution distance measures for $\mathscr{S}_{Gm}$ for point cloud comparison, thus combining both the semantic and geometric class differences,
- A qualitative analysis of the integrated nature of $\mathscr{S}_{Gm}$ using matrix visualization of appropriate reference distance measures.

Related Work: Given that 3D LiDAR point clouds implicitly encode a surface, they are 2.5D data characterized by surface feature descriptors. Such 3D descriptors can be categorized into global and

[†] Corresponding Author

DOI: 10.2312/stag.20211475

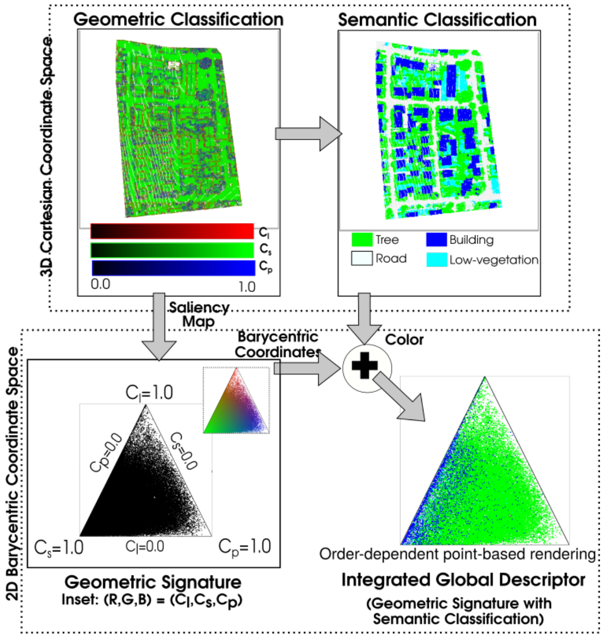
Geometric Classification
Semantic Classification
3D Cartesian Coordinate Space
C_l
C_s
C_p
0.0
1.0
Tree
Building
Road
Low-vegetation
Saliency Map
Barycentric Coordinates
Color
2D Barycentric Coordinate Space
C_l=1.0
C_p=0.0
C_s=0.0
C_s=1.0
C_l=0.0
C_p=1.0
Geometric Signature
Inset: (R,G,B) = (C_l,C_s,C_p)
Order-dependent point-based rendering
Integrated Global Descriptor
(Geometric Signature with Semantic Classification)

Figure 1: *Overview of our method of integrating probabilistic geometric and semantic classifications in IMGD $\mathscr{S}_{Gm}$ of an airborne LiDAR point cloud, using point rendering. For the 3D point cloud of Area-3 of the Vaihingen data (323,896 points) [RSJ*12], (Top) the top-view in Cartesian, and (Bottom) $\mathscr{S}_{Gm}$ in barycentric coordinate systems, generated using AvgSM T_{3DCM}. (Top-left) Mapping the saliency map $\{C_l, C_s, C_p\}$ to red, green, blue channels, and (Right) mapping the semantic classes to the selected color palette.*

local feature descriptors. The local feature descriptors use localized information, suitable for recognizing partially visible or incomplete objects in a cluttered scene with occlusions. In contrast, the global descriptors ignore shape details and require a priori segmentation of the object from the scene. Thus, they have difficulty recognizing partially visible or incomplete objects from cluttered scenes. Overall, the global features that are constructed by aggregating local features, tend to include geometric information.

Fisher vectors (FV) [PSM10] and Vector of Locally Aggregated Descriptors (VLAD) [JDSP10] are information-based aggregates used as global feature descriptors. FVs have been computed as global descriptors of images using the local descriptors and the Fisher kernel [JH99]. The FV is the normalized gradient vector of the local descriptors. The Fisher kernel of two images is a similarity measure computed as the dot product of FVs of the two images. The kernel could also be computed with a selected generative model [SPMV13]. The VLAD is a global descriptor of images, usually. It is computed using a codebook of clusters of SIFT (scale-invariant feature transform) descriptors [Low04], and accumulation of residuals between descriptors and corresponding cluster center. The VLAD has been further extended to point clouds using a deep learning network for place recognition [AUHL18]. Similar to FV used for point cloud analysis [BSLF18] and VLAD, our $\mathscr{S}_{Gm}$ is aggregated from local descriptors explicitly but uses coordinate space transformation.

Similar to our method, a barycentric coordinate representation of point-wise local geometric descriptor (LGD) has been used for the class-based analysis of point clouds in geomorphology [BL12]. The visualization referred to as the *dimensionality density diagram* (DDD), is constructed without the class information. The diagrams are then collated across multiple scales in a feature vector for semantic classification. Here, the multiscale integration occurs during the computation of eigenvalue-based saliency map [SNK17]. The visualization, equivalent to DDD, generates our descriptor $\mathscr{S}_{Gm}$, but after integrating the class information. $\mathscr{S}_{Gm}$ is further used for comparing point clouds, while DDD is used for classification.

Geometric distances between points or meshes are commonly used for point clouds. Point-to-mesh and mesh-to-mesh distances have been very well studied and demonstrated by Metro [CRS98], Mesh [ASCE02], and other tools. 3D shape reconstruction of a point cloud is computationally intensive and hence, is neither efficient nor scalable. An out-of-core method has been used for change detection in massive point clouds, without reducing the raw data [RKD13].

Direct comparison of point clouds is done by one-to-one point mapping using robust distance measures, such as the Gromov-Hausdorff (GH) distance [MS04]. GH distance generalizes the Hausdorff distance between two compact metric spaces, including probability measures and all isometric embeddings. It can be used for comparing LiDAR point clouds, which may have rotational transformations but after isometric transformations. However, computing a discrete approximation of GH distances using pair-wise geodesic distances is inefficient in large-scale point clouds.

The information from multiple scales is gathered usually by determining and using an entropy-based optimal scale [DMDV11], or by averaging features [KKV*11]. Other recent multiscale methods include a scale-invariant method for generating hierarchical point clusters by aggregating scales in a multiscale and hierarchical point classification method [WZF*14]. Latent Dirichlet Allocation (LDA) and AdaBoost classifiers are then used for feature extraction from the clusters and classification, respectively. A variant of this method uses natural exponential function thresholds for point-cluster extraction, and joint LDA and sparse coding (SCLDA) for feature extraction [ZZT*16] for ALS point clouds of complex scenes. To add contextual information, e.g., topology, in classification results, an RF classifier has been integrated into a Conditional Random Field (CRF) framework for urban object classification [NRS14].

2. Background: Multiscale Computations for Local Geometric Descriptor (LGD)

The LGD of a point in the point cloud stores the information in its local neighborhood. The shape of the local neighborhood is essential for determining semantic classification. The shape is captured in the form of a saliency map $\{C_l, C_s, C_p\}$, which gives the probability of a point belonging to the line- (linear), surface- (surface), and junction- (volumetric) type features. Hence, the saliency map is equivalent to the probabilistic geometric classification of the point [SNK17].

Computation of Local Geometric Descriptor: We use the positive semidefinite second-order tensor representation of the LGD [SNK17]. Here, we specifically use covariance tensor, T_{3DCM}, and tensor voting based descriptor, $T_{\text{3DVT-GET}}$, as computed by Sreevalsan-Nair *et al.,* [SNJK18]. For computing LGD, the first step is to identify points which will *qualify* as neighbors $\mathbb{N}(x)$ of a given point x in the point cloud $\mathscr{P}$. This is done using specific neighbor search volumes, as described later in this Section. The size of this neighbor search volume is the value for *scale*, s.

The covariance matrix is computed at a point x, as:

$$T_{\text{3DCM}}(x) = \sum_{y \in \mathbb{N}(x)} w_y t(y) t(y)^T \text{, with } w(y) = \frac{1-z(y)}{\sum_{y \in \mathbb{N}(x)} (1-z(y))},$$

where $t(y) = (y-x)$, normalized as: $z(y) = \frac{\|t(y)\|_2}{s}$.

The tensor voting field is initialized as an unoriented ball tensor at point x, using $t(y)$ as in the case of $T_{\text{3DCM}}(x)$.

$$V(x) = \sum_{y \in \mathbb{N}(x)} \mu_y . \left(I_d - \frac{t(y)t(y)^T}{t(y)^T t(y)} \right) \text{, with } \mu_y = exp\left(- \frac{\|t(y)\|_2^2}{\sigma^2} \right),$$

where I_d is identity matrix of size d based on spatial dimensionality, and $d = 3$, here. The scale value of the neighbor search volume is used as $\sigma = s$, in the Gaussian (attenuation) function μ_y. The tensor voting descriptor is computed upon applying anisotropic diffusion to the eigenvalues of tensors in the voting field $V(x)$, ordered as $\lambda_0 \geq \lambda_1 \geq \lambda_2$. For diffusion parameter $\delta = 0.16$ [SNK17, WHL*13],

$$T_{\text{3DVT}}(x) = \sum_{i=0}^{2} exp(-\frac{\lambda_i(x)}{\delta}) * e_i(x) e_i(x)^T.$$

$T_{\text{3DVT-GET}}$ is the tensor voting descriptor obtained when T_{3DVT} is further improved using 2D gradient energy tensor (GET) [SNJ17, SNJK18]. GET is computed as a Hessian to the height field in the point cloud and is used to identify points of interest (PoI) in $\mathscr{P}$, which are critical points in the GET field. $T_{\text{3DVT-GET}}$ is a modification of T_{3DVT}, where the GET is added to T_{3DVT} for the PoI only. We direct readers to the paper by Sreevalsan-Nair *et al.,* [SNJK18] for the multi-step procedure involved in computing $T_{\text{3DVT-GET}}$.

Computation of Saliency Map: The saliency map at each point is computed using the eigenvalue decomposition of its LGD. The saliency map for each scale s at each point x, with the eigenvalues of the LGD are sorted as $\lambda_0 \geq \lambda_1 \geq \lambda_2$ [SNK17], and $\mathbf{S} = (\lambda_0(x,s) + \lambda_1(x,s) + \lambda_2(x,s))^{-1}$:

$$C_l(x,s) = \mathbf{S} \cdot (\lambda_0(x,s) - \lambda_1(x,s)),$$
$$C_s(x,s) = \mathbf{S} \cdot (2(\lambda_1(x,s) - \lambda_2(x,s))),$$
$$C_p(x,s) = \mathbf{S} \cdot (3(\lambda_2(x,s))) = (1 - C_l(x,s) - C_s(x,s)),$$

This computation is applicable to all positive semidefinite second-order tensors [Kin04].

The saliency maps computed using single or multiple scales of LGDs serve as the probabilistic geometric classification of a point. This probabilistic classification can further be used for computing Shannon entropy, thus giving a measure of the *geometric uncertainty* in the data [SNM20]. This measure, referred to as the saliency map-based Shannon entropy, E_{geom}, is computed for a point $P \in \mathscr{P}$ as $E_{\text{geom}} = -C_l \cdot ln(C_l) - C_p \cdot ln(C_p) - C_s \cdot ln(C_s)$.

Multiscale Saliency Map: There are two strategies for aggregating the saliency maps across multiple scales. In the first one, the saliency maps are averaged across scales, as they are likelihoods of points belonging to a geometric class, and the different scales are mutually exclusive events [SNK17]. Thus, for N_s scales in $[s_{min}, s_{max}]$.

$$AvgSM = \{C_l, C_s, C_p\}(x) = \frac{1}{N_s} \sum_{s=s_{min}}^{s_{max}} \{C_l(x,s), C_s(x,s), C_p(x,s)\}.$$

The second strategy is to identify an optimal scale, which is determined based on the global minimum value of chosen Shannon entropy. Here, we use E_{geom}, and thus, we get optimal scale as $s_{\text{opt}} = argmin(E_{\text{geom}})$. The multiscale saliency map is:

$$OptSSM = \{C_l, C_s, C_p\}(x) = \{C_l(x, s_{\text{opt}}), C_s(x, s_{\text{opt}}), C_p(x, s_{\text{opt}})\}.$$

Neighbor Search Volume: The local neighborhood of a point is determined using the neighbor search volume. The conventionally used neighbor search volumes are spherical, cylindrical, and k-neighborhood. The cubical neighborhood is a coarse approximation of spherical neighborhood [SSN20]. The neighbor search volume is significant as its size is considered as the scale for computing LGD.

In our work, we use spherical neighborhood for LGDs for *AvgSM* saliency map, in the case of Vaihingen data [RSJ*12]. However, for *OptSSM* LGDs, we use finer-grained k-neighborhood, which is needed for finding the local minimum in Shannon entropy computed from anisotropy. Here, we have saliency map-based Shannon entropy E_{geom} for determining optimal scale, unlike the conventionally used eigen-entropy [DMDV11]. The entropies are related, nonetheless, except for the normalization factors used for computing the probabilities used in the entropy computation. In the case of large-scale datasets, such as in DALES [VAG20], we use cubical neighborhood for LGDs for reducing the computations needed in radial search [SSN20]. The reduction in computations is due to approximating Euclidean distance (L_2 norm) computation, which is required for checking for neighborhood relationship, using the Chebyshev distance (L_∞ or maximum norm).

3. Our Proposed Method

Our goal is to generate a global descriptor for a LiDAR point cloud that integrates information of both geometric and semantic classification, using our proposed workflow (Figure 2). Here, we first determine the representation form or mapping of the information, and then, the final format of the descriptor. We propose the use of visualization as an approach for representing the local descriptors of all points to construct the global one. Consequently, the

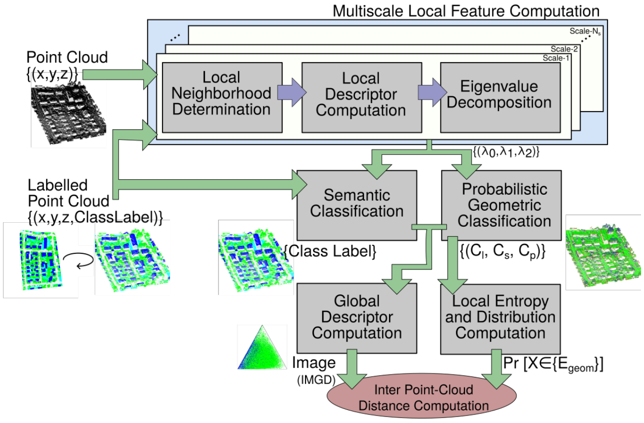
Multiscale Local Feature Computation
Scale-N_s
Scale-2
Scale-1
Point Cloud {(x,y,z)}
Local Neighborhood Determination
Local Descriptor Computation
Eigenvalue Decomposition
{(λ0,λ1,λ2)}
Labelled Point Cloud {(x,y,z,ClassLabel)}
Semantic Classification
Probabilistic Geometric Classification
{Class Label}
{(Cl, Cs, Cp)}
Global Descriptor Computation
Local Entropy and Distribution Computation
Image (IMGD)
Pr [X∈{Egeom}]
Inter Point-Cloud Distance Computation

Figure 2: *Our proposed workflow for computing the global descriptor, IMGD, for an airborne LiDAR point cloud, that is to be used for computing distances between point clouds.*

global descriptor, IMGD, is an image generated using our proposed workflow. We further introduce the appropriate distribution distance measures to be applied on the IMGDs of different ALS point clouds for comparative analysis.

3.1. Image-based Multiscale Global Descriptor (IMGD)

The geometric information of the point cloud $\mathscr{P}$ in its raw form is given by the positional information of the points. The multiscale map $\{C_l, C_s, C_p\}$ serves as the probabilistic geometric classification of each point in $\mathscr{P}$. The semantic class information of the point cloud is obtained by performing object classification. Here, we use labeled point clouds, and thus, do not perform an explicit semantic classification. The local geometric descriptor, e.g. the covariance tensor, and its eigenvalue-based features are used widely for object classification. We direct the readers to an exposition of the experiments used to identify appropriate feature vectors and classifiers [WJHM15] for a review of the state-of-the-art in supervised learning methods for semantic classification of ALS point clouds.

We observe that the probability density distributions of the saliency map, $\{C_l, C_s, C_p\}$, exhibit class-wise trends (Figure 3). As an example, this is demonstrated in building, tree, low-vegetation, and impervious surfaces/roads, for different regions of the Vaihingen site, namely, Area-2 and Area-3 [RSJ*12]. We see distinct trends in the tree and building classes, and the similarities in trends between low vegetation and road classes. This observation illustrates that the different facets of information of the point cloud, *i.e.,*geometric and semantic, are interdependent and hence, must be evaluated together. It strengthens the need for integrating the two types of information in the global descriptor of the point cloud.

Thus, we choose the probabilistic geometric and semantic object classifications as the information to be integrated into our proposed IMGD. We choose the multiscale saliency map over the raw positional data for geometric information, as the processed data encapsulates additional information of spatial locality. To integrate both saliency map and class labels, we perform two steps: (i) project the points in Cartesian coordinate space $\mathbb{R}^3$ to barycentric coordinate space $\mathbb{B}^2$ using the multiscale saliency map as its barycentric coordinates, and (ii) color the points based on its class labels.

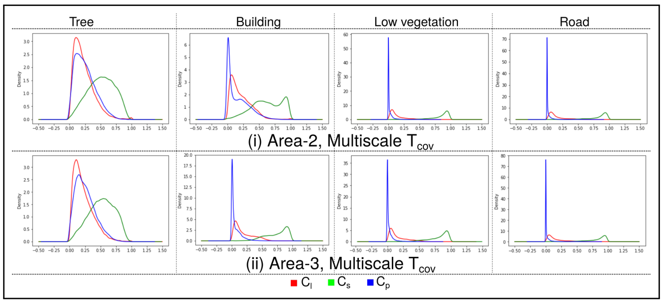

Figure 3: *Semantic class-wise probability density functions (PDFs) of saliency maps (C_l, C_s, C_p) computed using AvgSM T_{3DCM} for (i) Area-2 and (ii) Area-3 of Vaihingen site [RSJ*12]. We observe distinct trends in tree and building classes, and similar trends in low-vegetation and road.*

Barycentric Coordinate System (BCS): The two-dimensional (2D) BCS, $\mathbb{B}^2$, has two independent coordinates, and one dependent coordinate for each point. The characteristic of BCS is that, for any point $P \in \mathbb{B}^2$, with barycentric coordinates $[\alpha(P), \beta(P), \gamma(P)]$, the constraint of *partition to unity*, $\alpha(P) + \beta(P) + \gamma(P) = 1.0$, applies. This constraint is satisfied by the saliency maps, in which case the $\{C_l, C_s, C_p\}$ at P can be considered as its barycentric coordinates.

To incorporate this constraint on the coordinates, the coordinate axes of the BCS are defined by a *baseline* triangle (Figure 1). The vertices of the triangle have barycentric coordinate values, $[\alpha, \beta, \gamma]$, as $[1., 0., 0.]$, $[0., 1., 0.]$, and $[0., 0., 1.]$. The edges of the triangle opposite to these vertices are the axes, defined by functions $\alpha = 0.$, $\beta = 0.$, and $\gamma = 0.$, respectively. Any property $q(P)$ at a point P will be a *convex combination* of the property at the vertices of the triangle, e.g. the 3D position vector p, the color, etc. at A. This implies that, given the vertices of the baseline triangle, V_i, for $i = 0, 1, 2$, we get $q(P) = \alpha(P) * q(V_0) + \beta(P) * q(V_1) + \gamma(P) * q(V_2)$. Thus, the 2D position vector of each point in the ALS point cloud when projected onto the BCS, we use the weighted sum of the 2D position vectors of the vertices of the baseline triangle. The weights are the barycentric coordinates of the point, which is none other than the saliency map.

In all, the visualization of the point cloud in the BCS (Figure 2) with the position given by the saliency map, and the colors based on the semantic class labels integrate both the geometric and semantic class information of the points, respectively. The pixelated or raster image of this visualization is, thus, our proposed IMGD, $\mathscr{S}_{Gm}$, of the point cloud, $\mathscr{P}$.

Characteristics of the Global Descriptor: The probabilistic geometric classification is a many-to-one mapping, which implies that several points in the point cloud have the same saliency map. Consequently, several points can have the same barycentric coordinate representation. At the same time, *all* points in the point cloud with neighbors are guaranteed to have a saliency map, and hence a barycentric coordinate representation. The guarantee is due to the positive semi-definiteness property of the tensor, *i.e.,* the LGD [SNK17], which ensures non-negative eigenvalues and consequently, a saliency map at all non-outlier points. Thus, the mapping for generating $\mathscr{S}_{Gm}$ is a *non-injective surjective function*. The non-injective function implies that several points in the point cloud overlap in $\mathscr{S}_{Gm}$, which affects its graphical rendering.

Thus, the global descriptor is *lossy*, as its encoding of spatial locality *does not* preserve the information explicitly due to the non-linear transformation from $\mathbb{R}^3$ to $\mathbb{B}^2$. Hence, it can be used *only* for applications where analysis of the composition of a region is done using the geometric properties of the semantic classes in the region, and where spatial context is not required.

Graphical Rendering for Constructing IMGD: The construction of the global descriptor $\mathscr{S}_{Gm}$ involves graphically rendering it, usually using point objects. The visualization is improved by using appropriate point size and position vectors of the vertices of the baseline triangle. However, this rendering causes *overlap of points* based on the order of the points used in the rendering due to the non-injective mapping. Hence, point-based rendering has a limitation of hiding or obscuring previously rendered points. Improving the rendering of $\mathscr{S}_{Gm}$ is in the scope of future work.

The sampling density of the point cloud influences the overlap of points in $\mathscr{S}_{Gm}$. Thus, sparser point clouds tend to give *sparser* $\mathscr{S}_{Gm}$. However, before using downsampling as a solution, we first need to determine the *resilience* of our $\mathscr{S}_{Gm}$ to sampling density, *i.e.,* find approximately the sampling density at which the change in the point cloud size manifests as *perceptual differences* in the descriptor. We evaluate the influence of sampling density in our experiments.

We find that the point-based rendering of the descriptor is useful for visualizing the *scatter* of the point cloud in the BCS, either as a characteristic of a semantic class or for the entire cloud. For visualizing the scatter, since the color is not required, $\mathscr{S}_{Gm}$ is generated as a binarized representation, with all points rendered in black.

3.2. Distribution Distance Measures for Cloud Comparison

While visualization itself provides a qualitative comparison of the global descriptor of point clouds, we propose the use of distribution distance measures to quantify the distances. Given the image format of the descriptors, we use existing distance measures between images. We hypothesize that the difference between $\mathscr{S}_{Gm}$ of two point clouds gives the combined difference of geometric uncertainty and semantic composition between the clouds. Here, we propose using the widely used histogram-based measures, namely, bin-to-bin and cross-bin. The bin-to-bin computes the difference in the content of corresponding bins, while the latter additionally considers the neighborhood values of the concerned bins. Conventionally used Bhattacharyya distance [Bha46] is a bin-to-bin method that gives values in the range $[0,1]$, where 0 and 1 mean a perfect match and maximum difference between the images, respectively. We use the symmetrized Bhattacharyya distance:

$$d_{\text{BD-Img}}(\mathscr{S}_{Gm},\mathscr{S}'_{Gm}) = \sqrt{1-\sum_{i=1}^{m} p(i).p'(i)},$$

for m histogram bins in probability distributions p, p' corresponding to $\mathscr{S}_{Gm}, \mathscr{S}'_{Gm}$, respectively.

We also use the Earth Mover's Distance (EMD) $d_{\text{EMD-Img}}$ between the images, which is a cross-bin distance measure used for color images [RTG98]. EMD determines the minimum cost of transforming a histogram into another, and is proportional to the transformation cost. EMD is computed by solving the transportation problem using linear programming, in such a way that the overall distance is minimized. We use a faster implementation with successive short paths [PW09]:

$$d_{\text{EMD-Img}}(\mathscr{S}_{Gm},\mathscr{S}'_{Gm}) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij}f_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}}, \text{ with constraints:}$$

$$\text{(I) } f_{ij} \geq 0, \text{ for } 1 \leq i \leq m \text{ and } 1 \leq j \leq n; \qquad \text{(II) } \sum_{i=1}^{m} f_{ij} \leq P_i;$$

$$\text{(III) } \sum_{j=1}^{n} f_{ij} \leq Q_j; \qquad \text{(IV) } \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = \min(\sum_{i=1}^{m} P_i, \sum_{j=1}^{n} Q_j).$$

where f_{ij} is the flow between P_i and Q_j, for input histograms P and Q corresponding to $\mathscr{S}_{Gm}$ and $\mathscr{S}'_{Gm}$, respectively, and d_{ij} is the ground distance between i^{th} and j^{th} bins. We use the L_1 distance between indices as the inter-bin *ground distance*, *i.e.*,, $d_{ij} = \|i-j\|$ [PW09].

Increasing the image resolution of the IMGD improves the accuracy of the point cloud comparison. Owing to the use of histogram-based measures here, the images need not be of the same resolution, as ratios are used in the probability distribution. Nonetheless, using IMGDs with the same resolution improves the accuracy.

Preparing IMGDs for Point Cloud Comparison: To ensure effective comparison of point clouds using our proposed IMGD, we do the following:
• *Fixing Baseline Triangle:* For comparing the descriptors, only the interior of the baseline triangles need to be compared. The computed distances are accurate only if the regions to be compared are fixed in position across images. We achieve this by generating all descriptors using fixed position coordinates of the triangle vertices. Fixing the image regions can be alternatively done after generating $\mathscr{S}_{Gm}$ by using image processing operations, but this is prone to introducing additional computational errors.
• *Masking/Spatial Filtering*: After fixing the baseline triangle, it is sufficient to find differences of the triangle interiors. We achieve this by using the standard image processing technique of masking/spatial filtering. We extract the contour of the baseline triangle from the raster images, fill the area inside the contour to create a *mask* and find the intersection of the mask with the images. Then the distances between $\mathscr{S}_{Gm}$ are computed exclusively between the masked regions in the images.
• *Common Color Palette*: In the absence of a standard color scheme for semantic classes generally encountered in the airborne LiDAR point clouds, we narrow down specific color palettes for this work. We use the following two color palettes:

- **Dataset-specific palette:** We use the palettes as published by the data providers for the sake of familiarity of published visualizations of the datasets. Thus, when comparing datasets from different providers, this results in comparing descriptors with unmatched color palettes.
- **ColorBrewer palettes:** ColorBrewer [HB03, Bre20] is an online tool widely used for color palettes for thematic maps. We use a 12-class color palette in our experiments. This set of 12 classes can be further compacted to a minimal one based on the presence of the classes in the specific datasets being compared. Within this palette, we use specific choices of colors which have been inspired from either class-specific characteristics, e.g. green for vegetation, or from the available dataset-specific palette from providers, e.g. red for buildings.

Distance Matrices: In the absence of ground truth of distances between datasets, we first compare pair-wise distances amongst the datasets using a distance matrix, and then, compare distance matrices against those obtained from the conventionally used distance measures, which is used as a *reference*. We compute the distance matrix for each color palette, and for different LGDs used for constructing $\mathscr{S}_{Gm}$. We then visually compare these distance matrices.

3.3. Reference Distance Measures

To understand our integrated $\mathscr{S}_{Gm}$, we use reference distance measures from three separate components, namely, (a) geometric data, *i.e.*,, positional information of points and the ensuing surface mesh from the point clouds, (b) geometric classification, *i.e.*,, the saliency map and E_{geom}, and (c) semantic classification, which is the probability distribution of points across semantic classes for each dataset.

Conventional Distance Measures for Geometric Data: Hausdorff distance (d_H) and Chamfer distance (d_C) metrics are conventionally used for comparing distances between point clouds, using the positional information of the points. We use the Hausdorff distance also as a simplification of Gromov-Hausdorff distance as the latter is needed for comparing point cloud under deformation in metric space [MS04]. We use the symmetrized Hausdorff distance. Hausdorff distance is known to perform the worst for point clouds with a small number of outlier points, whereas Chamfer distance performs average, and EMD performs optimal but with a requirement of equal-sized point clouds [FSG17]. Here, we use the geometric distances only for relatively comparing the distances between point clouds, for which approximate values suffice. Hence, we use Hausdorff and Chamfer distance measures.

$$d_H(\mathscr{P}_1,\mathscr{P}_2) = \max\{\max_{x\in\mathscr{P}_1}\min_{y\in\mathscr{P}_2}\|x-y\|_2, \max_{y\in\mathscr{P}_2}\min_{x\in\mathscr{P}_1}\|x-y\|_2\}$$

$$d_C(\mathscr{P}_1,\mathscr{P}_2) = \sum_{x\in\mathscr{P}_1}\min_{y\in\mathscr{P}_2}\|x-y\|_2^2 + \sum_{y\in\mathscr{P}_2}\min_{x\in\mathscr{P}_1}\|x-y\|_2^2$$

It must be noted that the distance measures have been computed after normalizing the point cloud within a *canonical view volume*, i.e. $[-1,1]\times[-1,1]\times[-1,1]$, as used in computer graphics. The canonical view volume is also used for normalization to get the centroid of the point cloud closer to the origin (0,0,0).

Conventional Distance Measures for Geometric Classification: For obtaining the distance between two point clouds using probabilistic geometric classification, we use two methods. Firstly, we compute the EMD between distributions of saliency maps $\{C_l, C_s, C_p\}$, using 2D histograms, as there are only two independent variables in the saliency map. Secondly, we compute the EMD between univariate distributions of the Shannon entropy E_{geom}.

Conventional Distance Measures for Semantic Classification: The semantic classes are few. Hence, we use conventional distance metrics, *i.e.,* distribution distances between class probability distributions, for comparing the semantic classification of two point clouds. We use the total variation distance (d_{TVD}), and Hellinger distance (d_{HD}). We use distribution measures, such as symmetrized Kullback-Leibler divergence (d_{KL}), and Jensen-Shannon divergence (d_{JS}). d_{KL} is computed only for non-zero probability measures. For semantic class distributions, P and Q, with the same number of histogram bins (or classes) m, given by the common class order, we get:

$$\text{(I) } d_{TVD}(P,Q) = 0.5 * d_M(P,Q) \text{ for countable sets;}$$

$$\text{(II) } d_{HD}(P,Q) = \sqrt{0.5 * \sum_{i=1}^{m}(\sqrt{P_i}-\sqrt{Q_i})^2};$$

$$\text{(III) } d_{KL}(P,Q) = \sum_{i=1}^{m}(P_i - Q_i)\cdot\ln\left(\frac{P_i}{Q_i}\right), \text{ for } P_i, Q_i \neq 0;$$

$$\text{(IV) } d_{JS}(P,Q) = 0.5 * (\delta_{KL}(P,M) + \delta_{KL}(Q,M)),$$

$$\text{for } M = 0.5 * (P+Q) \text{ and } \delta_{KL}(A\|B) = \sum_{i=1}^{N} A_i \cdot \ln\left(\frac{A_i}{B_i}\right).$$

4. Experiments

We have used the airborne LiDAR dataset for residential areas, Area-2 and Area-3, in the Vaihingen site, from the ISPRS benchmark [RSJ*12], and two tiles out of 40 in the Dayton Annotated LiDAR Earth Scan (DALES) dataset of the City of Surrey [VAG20]. The DALES dataset has a higher point density of 50 ppm compared to the 4-7 ppm of the ISPRS dataset. We have divided the tiles of the DALES dataset into quarters, and treat each quarter of the tiles as separate point clouds for comparison purposes. Area-2 covers $170m\times190m$ characterized by high-rising residential buildings, surrounded by trees; and Area-3 covers $150m\times220m$ characterized by detached houses surrounded by trees. Each tile in DALES dataset covers $500m\times500m$, and hence the quarter tiles cover $250m\times250m$, each. The specifications of the 10 datasets with their size and class probability distribution, are given in Table 1. Figures 4 and 5 show the top-view of the 3D point clouds for the two regions in the Vaihingen site, Germany, and eight regions in the City of Surrey, Canada, respectively. These figures also show the probabilistic geometric classification from selected local geometric descriptors and the semantic classification of the selected point clouds.

Here, we have used the following LGDs:

- *AvgSM* T_{3DCM} using spherical neighborhood for Area-2 and Area-3 [SNK17], and cubical neighborhood for DALES data [SSN20].
- *AvgSM* $T_{\text{3DVT-GET}}$ using spherical neighborhood for Area-3 [SNJ17].
- *OptSSM* T_{3DCM} as well as *OptSSM* $T_{\text{3DVT-GET}}$, using k-neighborhood for Area-3 [SNM20].

For computing the LGDs for *AvgSM* aggregation, we have used three scales for each of the point clouds. For Area-2 and Area-3, the spherical neighborhoods of radii $r = \{0.009, 0.010, 0.011\}$ have been used for points in the canonical view volume. This is in metric scale, with step-size for radius $\Delta r = 0.21m$; we have $r = \{1.89m, 2.10m, 2.31m\}$ for Area-2; and $r = \{1.92m, 2.13m, 2.34m\}$ for Area-3. For the quarter tiles in the DALES dataset, we have used cubical neighborhoods of size $l = \{15m, 20m, 25m\}$. The value of l is equivalent to the diameter of the largest sphere contained in the cube; hence, l is comparable to $2r$. We have identified the scales based on the scale-wise distribution of entropy, E_{geom}, for each of the scales, where we have selected scales with relatively low entropy. The lowest value $E_{\text{geom}} = 0.$ at a point implies that the point belongs to one of the three geometric classes, and the highest value ($E_{\text{geom}} = \ln(3)$) implies that the point is equally likely to belong to all the three classes.

We have used k-neighborhoods for computing *OptSSM* T_{3DCM} to use finer-grained neighborhoods compared to the spherical ones. For both Area-2 and Area-3, we have used k values from 10 to 100, with step-size Δk=10, *i.e.,* we get 10 scales. This choice of a range of scales is comparable to the scales used in the spherical neighborhoods, as the area covered is comparable in both.

5. Results

We have visualized the entropy E_{geom} in the datasets to understand the spatial context of geometric uncertainty (Figure 5, Right). The entropy E_{geom} is maximum, *i.e.,* ln (3.0), at the centroid of the baseline triangle, owing to its barycentric representation $\left(\frac{1}{3},\frac{1}{3},\frac{1}{3}\right)$. E_{geom} monotonously decreases towards the vertices, where $E_{\text{geom}} = 0$, *i.e.,* ln (1.0), which is the minimum. We observe relatively high entropy for the tree or vegetation class. While high entropy points

Table 1: *The count distribution of 3D points in each semantic class, in percentage, in point clouds of two regions in Vaihingen from the ISPRS benchmark data [RSJ*12] and the quarters of two tiles from the Dayton Annotated LiDAR Earth Scan (DALES) dataset [VAG20].*

Dataset	# Points	ground/ imper.surf.	buildings/ roof	vegetation/ tree	low vegetation	unknown	cars	trucks	power-lines	fences/ hedge	poles	facade	shrub
Vaihingen-A2	266,675	0.192	0.259	0.403	0.146	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Vaihingen-A3	323,896	0.257	0.215	0.418	0.110	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Dales-95-q1	2,754,623	0.626	0.227	0.130	0.000	0.002	0.007	0.001	0.004	0.001	0.002	0.000	0.000
Dales-95-q2	2,694,980	0.667	0.139	0.175	0.000	0.003	0.002	0.002	0.004	0.004	0.003	0.000	0.000
Dales-95-q3	3,382,989	0.440	0.149	0.393	0.000	0.004	0.009	0.001	0.001	0.003	0.001	0.000	0.000
Dales-95-q4	3,098,121	0.471	0.241	0.240	0.000	0.003	0.037	0.003	0.002	0.002	0.001	0.000	0.000
Dales-60-q1	3,021,032	0.581	0.172	0.223	0.000	0.003	0.012	0.000	0.000	0.008	0.001	0.000	0.000
Dales-60-q2	4,270,369	0.309	0.121	0.553	0.000	0.004	0.008	0.001	0.001	0.005	0.000	0.000	0.000
Dales-60-q3	3,554,997	0.414	0.127	0.433	0.000	0.009	0.005	0.000	0.007	0.004	0.001	0.000	0.000
Dales-60-q4	2,937,802	0.616	0.094	0.262	0.000	0.003	0.006	0.000	0.013	0.006	0.003	0.000	0.000

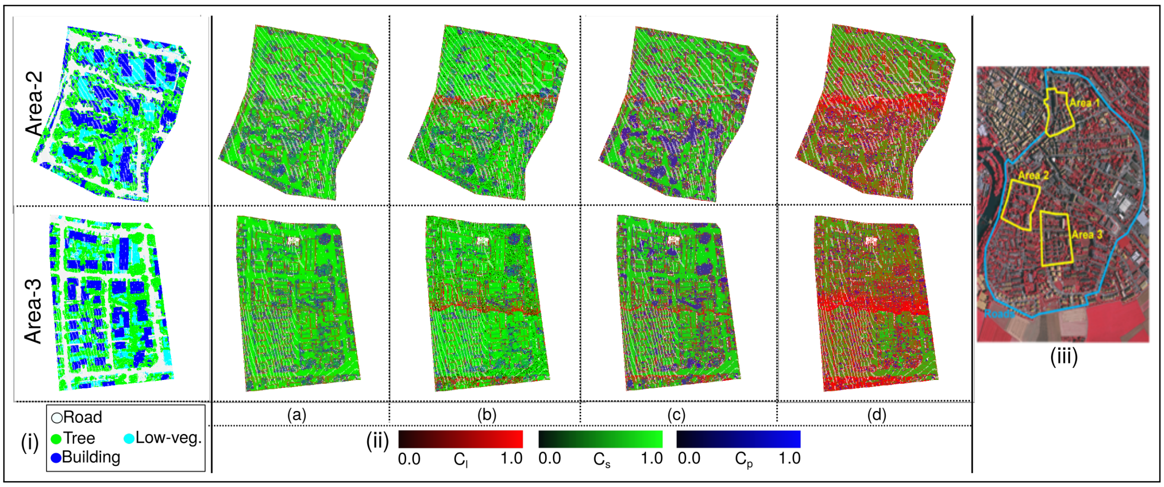

Figure 4: *Top-down views of point rendering of Area-2 and Area-3 of the Vaihingen site. They demonstrate (i) semantic classification, (ii) the variations in geometric classification for different local geometric descriptors (LGDs) with different multiscale approaches, and (iii) orthoimage of the Vaihingen site [RSJ*12]. In (ii), each point is colored with saturation in red, green, and blue channels corresponding to its saliency map, $\{C_l, C_s, C_p\}$, respectively. The following local geometric descriptors are used for generating signatures here: T_{3DCM} with (a) AvgSM, and (b) OptSSM; $T_{3DVT\text{-}GET}$ with (c) AvgSM, and (d) OptSSM.*

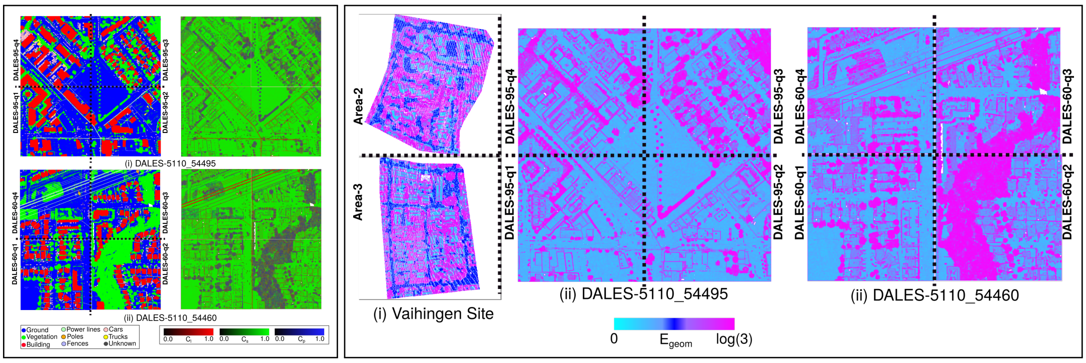

Figure 5: *(Left) Description of DALES tiles used in our experiments with its (left column) semantic classification, and (right column) probabilistic geometric classification using AvgSM T_{3DCM}. Each tile has been divided into four equal-area tiles, which are used separately here. (Right) Visualization of saliency map-based Shannon entropy, E_{geom}, computed from AvgSM T_{3DCM} as LGD.*

belong to the tree class, we also observe that the points in the tree class occupy the entire range of entropy, as seen in their global descriptors (Figure 6, Left). We have analyzed the distances between the selected datasets using distance matrices computed using EMD and BD (Figure 6, Right). We then compare these distance matrices with reference matrices for geometric data, geometric classification, and semantic classification (Figure 7).

Observations from IMGD: In point clouds of the DALES dataset, we observe that certain sparser classes, such as powerline, are promi-

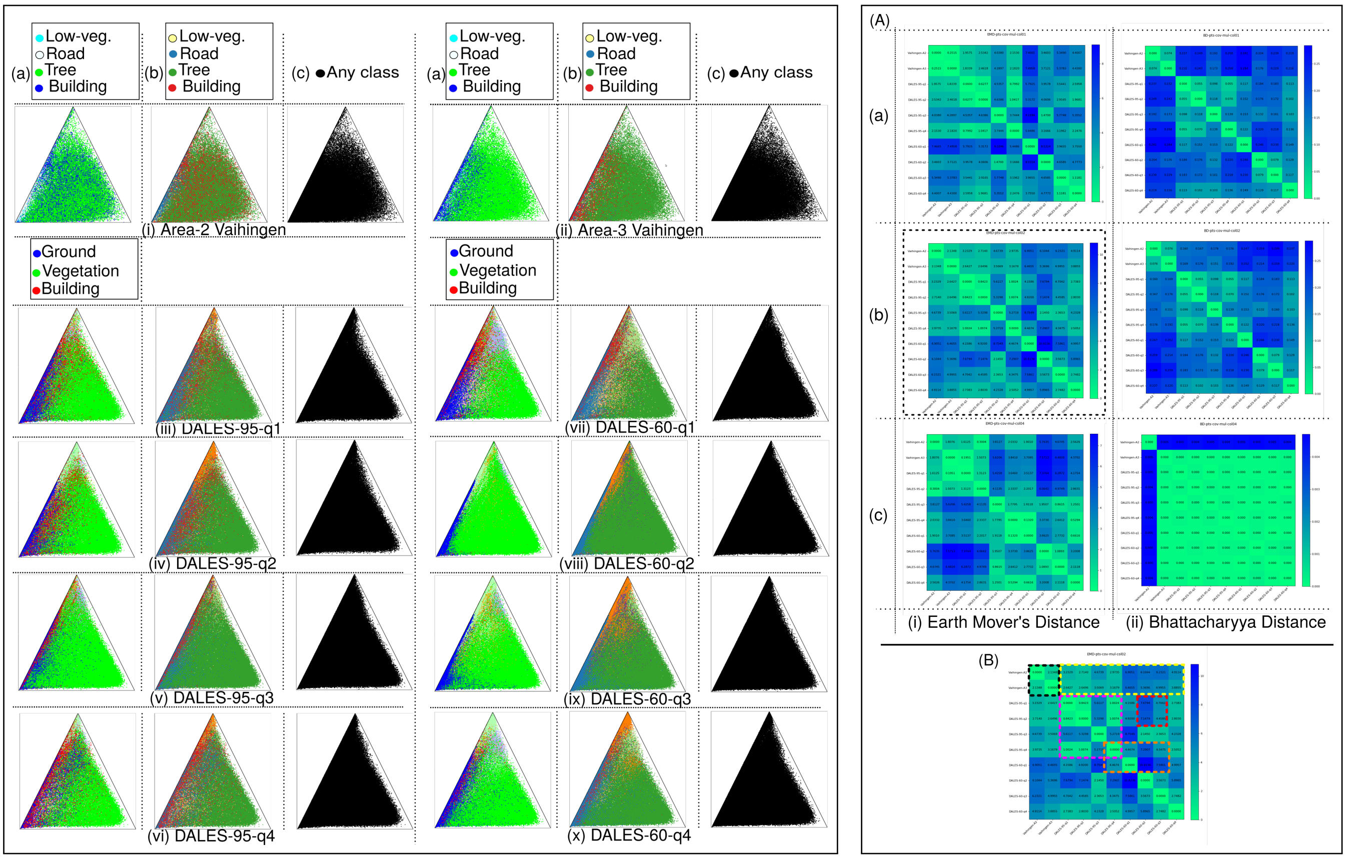

Figure 6: *IMGDs generated using three different color palettes: (a) dataset-specific, (b) ColorBrewer 12-class one, and (c) binary one. (Left) IMGDs for Vaihingen (i) Area-2, and (ii) Area-3, and the quarter tiles of (iii-vi) DALES-5110_54495 and (vii-x) DALES-5110_54460 point clouds, using point rendering,The sparser classes in (iii-x) are {Unknown, Cars, Trucks, Power lines, Fences, Poles}, correspond to colors: (a) {●, ●, ●, ●, ●, ●} (as shown in Figure 5); (b) {●, ●, ●, ●, ●, ●}. (Right) Distance matrices of IMGDs of 10 datasets, using AvgSM from T_{3DCM}. (A) shows matrices for (i) EMD, (ii) Bhattacharyya Distance – with point rendering. In (B), important aspects of the EMD matrix in ((A), (b)) are highlighted.*

nent in the global descriptors, e.g. for DALES-95-q2, DALES-60-q3, and DALES-60-q4. This is because $\mathscr{S}_{Gm}$ is designed to visualize the class distributions at each barycentric coordinate, which is different from the class distribution of the entire point cloud. We can now conclude that the powerline dominates line-type features (Figure 6, Left), and hence, the powerline points cluster around the vertex $C_l = 1.0$.

Through the binary colored IMGDs (Figure 6, Left, (c)), we observe higher *scatter* in Vaihingen datasets than in the DALES ones, due to the lower point density in the former. Area-2 and Area-3 have sparser points near the vertices, $C_l = 1$ and $C_p = 1$, indicating surface-type features. Similarly, there are sparser points along the edge, $C_s = 0$, and near the vertex $C_p = 1$, in DALES datasets.

$\mathscr{S}_{Gm}$ rendered using ColorBrewer is more reliable than the dataset-specific one by the dataset providers. The higher reliability is, by design, due to the commonality of color maps across the Vaihingen and DALES datasets. The ColorBrewer palette is also effective owing to its closeness to the dataset-specific one, without the arbitrariness in the color maps by different data providers.

Visual comparison of the distance matrices (Figure 6(A)) gives us insight into the influence of distance measures, and color palettes on the effectiveness of $\mathscr{S}_{Gm}$. We observe that EMD has higher sensitivity in the differences between $\mathscr{S}_{Gm}$ of point clouds than Bhattacharyya distance. We also find that the $d_{\text{EMD-Img}}$ integrates the separate components of the distance measures from geometric classification, geometric distances, and semantic classification (Figures 6(B) and 7). The bin-to-bin distance measures, $d_{\text{BD-Img}}$, between binarized point-based rendering show negligible differences between different point clouds, owing to which the distance matrix captures low variation (Figure 6, Right, (ii), (c)). Overall, $d_{\text{EMD-Img}}$ can be perceived to be a better distance measure than $d_{\text{BD-Img}}$.

Overall, we choose the IMGD using the ColorBrewer 12-class palette as an effective descriptor and $d_{\text{EMD-Img}}$ as a reliable distance measure between the chosen $\mathscr{S}_{Gm}$ of point clouds. This is due to its consistent results (Figure 6, Right), and confirmation to the reference component-wise distance matrices (Figure 7). We observe the following component-wise findings integrated into our chosen descriptor and its distance measure (Figure 6(B)):

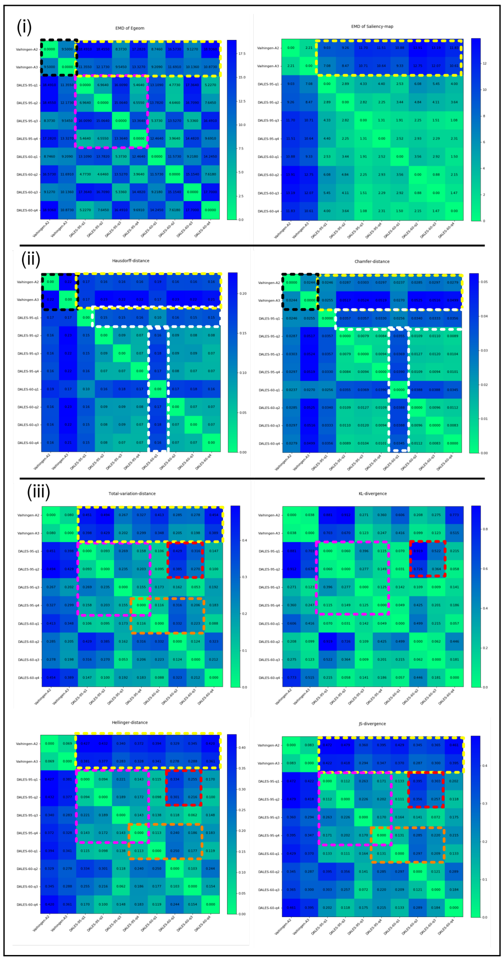

Figure 7: *Reference distance matrices of the 10 datasets. (i) Geometric classification distances using (Left-to-right) EMD of E_{geom} and EMD of $\{C_l, C_s, C_p\}$. (ii) Geometric distances using (Left-to-right) Hausdorff distance and Chamfer distance. (iii) Semantic classification distances using (Clockwise from top-left) Total variation distance, KL-divergence, Jensen-Shannon divergence, and Hellinger distance. Comparing with our distance matrices (Figure 6, (A)), we observe that the specific patterns captured in (i)-(iii) are integrated in Figure 6(B), except for the white boxes in (ii).*

- The Vaihingen datasets are different from the DALES region with respect to all three components, namely, geometric classification, geometric distances, and semantic classification. This difference is highlighted as yellow box (Figure 7). Our integrated distance measure captures the distances in the yellow box most similar to the combination observed in geometric classification distance using EMD of saliency map and semantic classification distance measure using total variation distance.
- There are dissimilarities between Area-2 and Area-3 of Vaihingen with respect to both geometric measures, *i.e.,* from geometric classification as well as geometric information, shown as a black box (Figure 7 (i), (ii)); which are also present in our distance measure (Figure 6(B)).

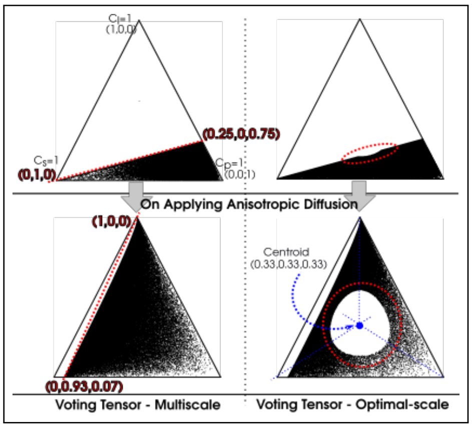

Figure 8: *Patterns observed in the IMGDs, from LGDs using tensor voting and optimal scale, are shown in red dotted lines and ellipses, respectively, in the baseline triangle in barycentric coordinate space. These images are generated for LiDAR point cloud of Area-3 of Vaihingen dataset [RSJ*12] (323,896 points).*

- There is more similarity between the regions in DALES-95 than in DALES-60 or Vaihingen, shown as a magenta box (Figure 7), in EMD of E_{geom} (geometric classification), all distance measures of semantic classification, and our distance measure.
- The region DALES-95-q1-q2 (half-tile) has relatively high dissimilarity with the region DALES-60-q2-q3, shown as a red box (Figure 7), as observed by all semantic classification distance measures and our distance measure. We know that DALES-60-q2-q3 has the highest percentage of points in tree class [45-50%], and similarly, DALES-95-q1-q2 has the highest percentage in buildings [62-66%] (Figure 5), which explains the large differences between the two half-tiles.
- The regions DALES-95-q4 and DALES-60-q1 are similar to each other, and very different from DALES-60-q2-q3 region (orange box in Figure 7). We know that DALES-60-q2-q3 has a large percentage of points in tree class [45-50%] , and DALES-95-q4 and DALES-60-q1 have a larger percentage in buildings [47-58%] (Figure 5), which explains the pattern.
- We observe that the semantic classification is prominently captured in our distance measure, which is owing to the characteristic of $\mathscr{S}_{Gm}$ being an image, with colors corresponding to the semantic classes.
- The patterns observed in geometric uncertainty are captured as well in our IMGD, but not much with geometric distances (white box in Figure 7(ii)).

Overall, we have demonstrated, by visual comparison of distance matrices, that: (a) our IMGD effectively integrates both geometric and semantic classification information, and (b) our distribution distance measure of our image-based descriptor effectively captures integrated differences between point clouds.

Discussion:

Using different LGDs for generating the IMGD, we observe two salient patterns in our global descriptor $\mathscr{S}_{Gm}$ (Figure 8). The first pattern is a distinct one in the $\mathscr{S}_{Gm}$ of the tensor voting-based LGDs,

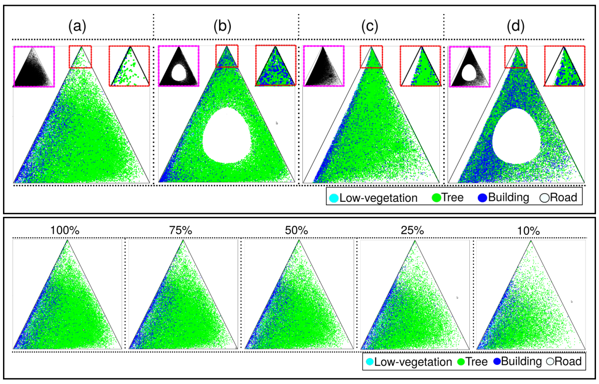

Figure 9: *IMGD of the Area-3 of Vaihingen site using different settings. (Top) $\mathscr{S}_{Gm}$ for different local geometric descriptors, namely, T_{3DCM} with (a) AvgSM, and (b) OptSSM; and $T_{3DVT\text{-}GET}$ with (c) AvgSM, and (d) OptSSM. The red insets show the zoomed-in version of the apex of the triangle, and magenta insets show the binarized variant of point-based rendering. (Bottom) Effects of sampling density on IMGD studied for $\mathscr{S}_{Gm}$ from AvgSM T_{3DCM} in the point cloud of Area-3 of the Vaihingen site. Different point densities have been generated using uniform spatial sampling in the point cloud in the Cartesian coordinate space.*

T_{3DVT} or $T_{\text{3DVT-GET}}$, and the second one is seen in *OptSSM* LGDs. The first one is explained by the characteristic of T_{3DVT} being computed using ball tensors, as an initialization. Hence, T_{3DVT} is characterized by $(C_l \leq C_p + C_s)$ in T_{3DVT}, since the ball tensor is computed in the normal space, unlike the T_{3DCM} in the tangent space [SNK17]. The computation in normal space causes the vote from each neighbor to be a curvel, *i.e.,* a plate tensor, where the minor eigenvalue is zero [TTMM04]. Thus, when these votes add up, a constraint on C_l builds up, which manifests as $(C_l \leq C_p + C_s)$. When computing the saliency map, the upper bound of this constraint, $\frac{C_s}{C_l} \leq 3.0$, appears as the *partitioning line* in the barycentric space. The partitioning line is between $(0,1,0)$ and $(0.25,0,0.75)$ (red dotted line in Figure 8). We perform anisotropic diffusion on tensor voting to strengthen weak line-type features [WHL*13] and to make the tensor voting-based LGD substitutable for T_{3DCM} [SNK17]. Now, anisotropic diffusion transforms the partitioning line in the signature of T_{3DVT} to the curved boundary between $(1,0,0)$ and $(0,0.93,0.07)$ (Figure 8), which is our first pattern. This pattern in $\mathscr{S}_{Gm}$ demonstrates two known characteristics of anisotropic diffusion on T_{3DVT} [SNK17, SNJK18], *i.e.,* (i) the increase in the number of line-type features, and (ii) decrease in the number of point-type features in T_{3DVT}. Thus, the $\mathscr{S}_{Gm}$ gives a visual validation of the characteristics of tensor voting and anisotropic diffusion.

For the second pattern in Figure 8, we observe that the $\mathscr{S}_{Gm}$ using an optimal scale based on E_{geom} shows a distinct "hole," or *i.e.,* an empty region in the neighborhood of the centroid. The optimal scale occurs at the global minimum of E_{geom}. Low-entropy regions are in the vicinity of the vertices of the baseline triangle, *i.e.,* where C_l, C_s, or C_p is 1. The entropy is maximum in the neighborhood of the centroid of the triangle. Thus, the crowding of points closer to the vertices of the baseline triangle causes the second pattern. We also observe the pattern for T_{3DCM} (Figure 9). In the case of optimal scale in tensor voting, we observe an additional hemispherical "dent" in the partitioning line in T_{3DVT}, which transforms to the "hole", upon applying anisotropic diffusion (Figure 8). Thus, the $\mathscr{S}_{Gm}$ with T_{3DVT} combines both the patterns.

We observe these patterns in $\mathscr{S}_{Gm}$ using different LGDs (Figure 9 (Top)). For Area-3, the overlap of the scatter of points of the tree and building classes has the following ascending order, with respect to its LGD: *OptSSM* T_{3DCM} < *AvgSM* T_{3DCM} < *AvgSM* $T_{\text{3DVT-GET}}$ < *OptSSM* $T_{\text{3DVT-GET}}$. The area of class overlap is inversely proportional to the extent of class separability. We thus conclude that *OptSSM* T_{3DCM} performs the best for feature extraction for semantic classification, as demonstrated by Weinmann *et al.,* [WJHM15]. Considering the *resilience* of IMGD to sampling, we observe that perceptually the descriptor remains the same until 50% uniform spatial downsampling (Figure 9 (Bottom)). This also shows that the IMGD of the point cloud can be generated using its downsampled version. Note that for this experiment, the LGDs generated from the entire cloud are used in the downsampled ones.

6. Conclusions

This paper addresses the gap in studying geometric and semantic information of ALS point clouds in an integrated manner. We have addressed the gap by constructing a novel image-based multiscale global descriptor (IMGD) of the cloud that incorporates this information and using appropriate distance measures on this descriptor for finding distances between point clouds. We have showcased the effectiveness of our global descriptor on the publicly available datasets from ISPRS [RSJ*12] and DALES [VAG20]. Our IMGD uses the transformation of the point cloud from the 3D Cartesian coordinate system to the 2D barycentric coordinate system (BCS) using probabilistic geometric classification [SNK17]. We have identified reference distances to compare our integrated distance measures, which are Bhattacharyya distance and Earth Movers Distance (EMD) of the descriptors. Our experiments have enabled us to narrow down different parameters in rendering the descriptor, namely the color palette, to obtain the descriptor most effective for computing distances. We conclude that a common color palette mapping the classes is essential for constructing our novel global descriptor. Our work shows that the point rendering with the ColorBrewer palette gives us the most effective IMGD and that EMD is a better-suited distance measure for our proposed descriptor. This work is overall a step towards formalizing perceptual differences in the visualization of point clouds as quantifiable distance measures.

Acknowledgments

This work has been supported by the Early Career Research Award, from the Science and Engineering Research Board (SERB), Government of India, and Intel India Research Fellowship. This work has used third-party software libraries – PCL [RC11], CGAL [FP09], OpenCV [Bra00], SciPy [JOP*01, VGO*20], and Open3D [ZPK18] for implementations of processing of point clouds, geometry and images; and distance measures. The Vaihingen dataset has been published by the German Society for Photogrammetry, Remote Sensing and Geoinformation [Cra10, Cav18], and the DALES dataset, under the Open Government License, City of Surrey [VAG20].

References

[ASCE02] Aspert N., Santa-Cruz D., Ebrahimi T.: Mesh: Measuring errors between surfaces using the hausdorff distance. In *Proceedings. IEEE international conference on multimedia and expo* (2002), vol. 1, IEEE, pp. 705–708. 2

[AUHL18] Angelina Uy M., Hee Lee G.: PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 4470–4479. 2

[Bha46] Bhattacharyya A.: On a measure of divergence between two multinomial populations. *Sankhyā: the indian journal of statistics* (1946), 401–406. 5

[BL12] Brodu N., Lague D.: 3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology. *ISPRS Journal of Photogrammetry and Remote Sensing 68* (2012), 121–134. 2

[Bra00] Bradski G.: OpenCV 2.4.11.0 Documentation. *Dr. Dobb's Journal of Software Tools* (2000). 10

[Bre20] Brewer C. A.: ColorBrewer 2.0: color advice for cartography, Last accessed on September 30, 2020. URL: http://colorbrewer2.org/. 5

[BSLF18] Ben-Shabat Y., Lindenbaum M., Fischer A.: 3dmfv: Three-dimensional point cloud classification in real-time using convolutional neural networks. *IEEE Robotics and Automation Letters 3*, 4 (2018), 3145–3152. 2

[Cav18] Cavallaro G.: Vaihingen dataset acquisition project. http://b2find.eudat.eu/dataset/db1d323a-7024-5883-af08-b0f9dc5b7a2b, 2018. last accessed on September 16, 2021. 10

[Cra10] Cramer M.: The DGPF-test on digital airborne camera evaluation–overview and test design. *Photogrammetrie-Fernerkundung-Geoinformation 2010*, 2 (2010), 73–82. 10

[CRS98] Cignoni P., Rocchini C., Scopigno R.: Metro: measuring error on simplified surfaces. In *Computer Graphics Forum* (1998), vol. 17, Wiley Online Library, pp. 167–174. 2

[DMDV11] Demantké J., Mallet C., David N., Vallet B.: Dimensionality based Scale Selection in 3D LiDAR Point Clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 38*, Part 5 (2011), W12. 2, 3

[FP09] Fabri A., Pion S.: CGAL: The computational geometry algorithms library. In *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems* (2009), pp. 538–539. 10

[FSG17] Fan H., Su H., Guibas L. J.: A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2017), pp. 605–613. 6

[HB03] Harrower M., Brewer C. A.: Colorbrewer.org: an online tool for selecting colour schemes for maps. *The Cartographic Journal 40*, 1 (2003), 27–37. 5

[JDSP10] Jégou H., Douze M., Schmid C., Pérez P.: Aggregating local descriptors into a compact image representation. In *CVPR 2010-23rd IEEE Conference on Computer Vision & Pattern Recognition* (2010), IEEE Computer Society, pp. 3304–3311. 2

[JH99] Jaakkola T., Haussler D.: Exploiting generative models in discriminative classifiers. In *Advances in neural information processing systems* (1999), pp. 487–493. 2

[JOP*01] Jones E., Oliphant T., Peterson P., et al.: SciPy: Open source scientific tools for Python. 10

[Kin04] Kindlmann G.: Superquadric Tensor Glyphs. In *Proceedings of the Sixth Joint Eurographics-IEEE TCVG conference on Visualization* (2004), Eurographics Association, pp. 147–154. 3

[KKV*11] Keller P., Kreylos O., Vanco M., Hering-Bertram M., Cowgill E. S., Kellogg L. H., Hamann B., Hagen H.: Extracting and visualizing structural features in environmental point cloud LiDaR data sets. In *Topological Methods in Data Analysis and Visualization*. Springer, 2011, pp. 179–192. 2

[Low04] Lowe D. G.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision 60*, 2 (2004), 91–110. 2

[MS04] Mémoli F., Sapiro G.: Comparing point clouds. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing* (2004), ACM, pp. 32–40. 2, 6

[NRS14] Niemeyer J., Rottensteiner F., Soergel U.: Contextual classification of lidar data and building object detection in urban areas. *ISPRS journal of photogrammetry and remote sensing 87* (2014), 152–165. 2

[PSM10] Perronnin F., Sánchez J., Mensink T.: Improving the fisher kernel for large-scale image classification. In *European conference on computer vision* (2010), Springer, pp. 143–156. 2

[PW09] Pele O., Werman M.: Fast and robust earth mover's distances. In *2009 IEEE 12th International Conference on Computer Vision* (September 2009), IEEE, pp. 460–467. 5

[RC11] Rusu R. B., Cousins S.: 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)* (Shanghai, China, May 9-13 2011). 10

[RKD13] Richter R., Kyprianidis J. E., Döllner J.: Out-of-Core GPU-based Change Detection in Massive 3D Point Clouds. *Transactions in GIS 17*, 5 (2013), 724–741. 2

[Rot09] Rottensteiner F.: Status and further prospects of object extraction from image and laser data. In *2009 Joint Urban Remote Sensing Event* (2009), IEEE, pp. 1–10. 1

[RSJ*12] Rottensteiner F., Sohn G., Jung J., Gerke M., Baillard C., Benitez S., Breitkopf U.: The ISPRS benchmark on urban object classification and 3D building reconstruction. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences I-3* (2012), 293–298. 1, 2, 3, 4, 6, 7, 9, 10

[RTG98] Rubner Y., Tomasi C., Guibas L. J.: A metric for distributions with applications to image databases. In *Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)* (1998), IEEE, pp. 59–66. 5

[SNJ17] Sreevalsan-Nair J., Jindal A.: Using gradients and tensor voting in 3D local geometric descriptors for feature detection in airborne lidar point clouds in urban regions. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (2017), IEEE, pp. 5881–5884. 3, 6

[SNJK18] Sreevalsan-Nair J., Jindal A., Kumari B.: Contour extraction in buildings in airborne lidar point clouds using multiscale local geometric descriptors and visual analytics. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11*, 7 (2018), 2320–2335. 3, 10

[SNK17] Sreevalsan-Nair J., Kumari B.: *Local Geometric Descriptors for Multi-Scale Probabilistic Point Classification of Airborne LiDAR Point Clouds*. Springer Cham, Mathematics and Visualization, 2017, pp. 175–200. 2, 3, 4, 6, 10

[SNM20] Sreevalsan-Nair J., Mohapatra P.: Influence of Aleatoric Uncertainty on Semantic Classification of Airborne LiDAR Point Clouds: A Case Study with Random Forest Classifier Using Multiscale Features. In *Proceedings of the IEEE International India GeoScience and Remote Sensing Symposium 2020 (to appear)* (2020). 1, 3, 6

[SPMV13] Sánchez J., Perronnin F., Mensink T., Verbeek J.: Image classification with the fisher vector: Theory and practice. *International journal of computer vision 105*, 3 (2013), 222–245. 2

[SSN20] Singh S., Sreevalsan-Nair J.: A Distributed System for Multiscale Feature Extraction and Semantic Classification of Large-scale LiDAR Point Clouds. In *Proceedings of the 2020 IEEE India GeoScience and Remote Sensing Symposium* (2020), pp. 74–77. 3, 6

[TTMM04] Tong W.-S., Tang C.-K., Mordohai P., Medioni G.: First order augmentation to tensor voting for boundary inference and multiscale analysis in 3D. *IEEE transactions on pattern analysis and machine intelligence 26*, 5 (2004), 594–611. 10

[VAG20] Varney N., Asari V. K., Graehling Q.: DALES: A Large-scale Aerial LiDAR Data Set for Semantic Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (2020), pp. 186–187. 3, 6, 7, 10

[VGO*20] Virtanen P., Gommers R., Oliphant T. E., Haberland M., Reddy T., Cournapeau D., Burovski E., Peterson P., Weckesser W., Bright J., van der Walt S. J., Brett M., Wilson J., Millman K. J., Mayorov N., Nelson A. R. J., Jones E., Kern R., Larson E., Carey C. J., Polat İ., Feng Y., Moore E. W., VanderPlas J., Laxalde D., Perktold J., Cimrman R., Henriksen I., Quintero E. A., Harris C. R., Archibald A. M., Ribeiro A. H., Pedregosa F., van Mulbregt P., SciPy 1.0 Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods 17* (2020), 261–272. doi:10.1038/s41592-019-0686-2. 10

[WHL*13] Wang S., Hou T., Li S., Su Z., Qin H.: Anisotropic Elliptic PDEs for Feature Classification. *Visualization and Computer Graphics, IEEE Transactions on 19*, 10 (2013), 1606–1618. 3, 10

[WJHM15] Weinmann M., Jutzi B., Hinz S., Mallet C.: Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing 105* (2015), 286–304. 1, 4, 10

[WZF*14] Wang Z., Zhang L., Fang T., Mathiopoulos P. T., Tong X., Qu H., Xiao Z., Li F., Chen D.: A multiscale and hierarchical feature extraction method for terrestrial laser scanning point cloud classification. *IEEE Transactions on Geoscience and Remote Sensing 53*, 5 (2014), 2409–2425. 2

[ZPK18] Zhou Q.-Y., Park J., Koltun V.: Open3D: A modern library for 3D data processing. *arXiv:1801.09847* (2018). 10

[ZZT*16] Zhang Z., Zhang L., Tong X., Mathiopoulos P. T., Guo B., Huang X., Wang Z., Wang Y.: A multilevel point-cluster-based discriminative feature for ALS point cloud classification. *IEEE Transactions on Geoscience and Remote Sensing 54*, 6 (2016), 3309–3321. 2