IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 3, 2021, accepted June 28, 2021, date of publication July 1, 2021, date of current version July 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3094127

DALES Objects: A Large Scale Benchmark Dataset for Instance Segmentation in Aerial Lidar

NINA M. SINGER, (Graduate Student Member, IEEE), AND VIJAYAN K. ASARI, (Senior Member, IEEE)
Department of Electrical and Computer Engineering, University of Dayton, Dayton, OH 45469, USA
Corresponding author: Nina M. Singer (nsinger1@udayton.edu)

ABSTRACT We present DALES Objects, a large-scale instance segmentation benchmark dataset for aerial lidar. DALES Objects contains close to half a billion hand-labeled points, including semantic and instance segmentation labels. DALES Objects is an extension of the DALES (Varney *et al.*, 2020) dataset, adding additional intensity and instance segmentation annotation. This paper provides an overview of the data collection, preprocessing, hand-labeling strategy, and final data format. We propose relevant evaluation metrics and provide insights into potential challenges when evaluating this benchmark dataset. Finally, we provide information about how researchers can access the dataset for their use at go.udayton.edu/dales3d.

INDEX TERMS 3D data set, aerial vision, airborne system, ALS, benchmark data, data annotation, deep learning, earth scan, instance segmentation, laser scan, lidar, point cloud, semantic segmentation.

I. INTRODUCTION

Benchmark 2D image datasets like MNIST [2], CIFAR-10 [3], COCO [4], and ImageNet [5] are well known. However, in recent years, lidar sensors' advancement and increased interest in automatic driving have caused an increase in 3D datasets and expressly point clouds datasets. Research into deep learning in 3D data is not as mature as its 2D counterparts. The additional dimension increases the complexity and number of parameters in the network. The nature of point clouds also increases the difficulty. Each point cloud contains a considerable amount of individual points, all of which are unorganized and contain no formal structure, unlike their image counterparts, making direct convolution impossible. There are also considerations of occlusion and point density, which vary significantly inside a single scene depending on the sensor location and type. Because of these characteristics, a single scene can have an infinite amount of point cloud representations, making it difficult to generalize across different scenes. As a general rule, as the task's complexity increases, more data is required to produce the desired results [6]; this makes it imperative to produce large high-quality labeled datasets to train and evaluate networks for these 3D deep learning tasks.

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed A. Zaki Diab.

Lidar technology has become increasingly popular in recent years, with advancements propelled by interest in autonomous driving. However, although we have seen considerable strides in the accessibility of mobile and consumer lidar devices, high-quality geo-referenced aerial lidar can still be prohibitively expensive. Because of these costs, there are significantly fewer benchmark datasets for aerial lidar.

The two most common tasks in 3D point cloud processing are semantic and instance segmentation. These two types of segmentation can be used as the initial processing steps for all subsequent tasks. We define semantic segmentation as the labeling of each point into a general object category. These categories are typically broad and non-specific, such as ground, vegetation, or buildings. Similarly, instance segmentation is labeling each point into an object id, specifying an individual object within that category.

In 3D point cloud research, dozens of semantic segmentation benchmarks cover various scene types, and object categories [7], [8]. These benchmark datasets include indoor and outdoor scenes and different sensor types, such as lidar or RGB-D-type sensors. Unfortunately, there are not as many instance segmentation benchmarks, with only a few data sets with widespread usage. Instance segmentation is an important task when considering scene understanding.

We consider a case where a utility company is interested in using lidar to monitor a remote stretch of powerline to

perform routine maintenance, preventing deadly forest fires. An initial preprocessing step might be to perform a semantic segmentation that labels individual points into several distinct categories, like power lines, poles, and vegetation. This information is valuable, but it does not provide the complete picture. We can use an instance segmentation task to provide additional information, like the number of powerlines and poles in an area. Or the number of buildings that might be affected by a potential outage. Instance segmentation is an additional critical level of information on the way to complete scene understanding.

In this paper, we present our DALES Objects dataset. This dataset is the first of its kind, offering a meticulously hand-labeled dataset that contains eight semantic object categories and over twenty thousand hand-labeled instances. The DALES Objects dataset presents one of the most extensive instance segmentation datasets taken with aerial lidar and one of the first to include rural and urban scenes. The addition of these rural scenes provides essential information for tasks such as forestry management and utility asset monitoring. Using lidar gives a high spatial accuracy to our dataset and a unique set of viewpoints and occlusions. The outdoor dataset allows for a different set of object categories in addition to the new sensor type. It will enable researchers to test their datasets in a unique environment, giving a better understanding of network performance in a diverse group of settings.

DALES Objects covers over ten square kilometers of aerial lidar with over eight object categories; ground, vegetation, buildings, cars, trucks, powerlines, poles, and fences. We provide instance segmentation labels of each human-made object within the dataset, providing individual object id's for all buildings, cars, trucks, powerlines, poles, and fences. We can see an example of one of the DALES Objects scenes in Figure 1. We split the dataset in a rough 70/30 split between training and testing and provide the final data in two formats that match existing benchmarks for ease of use. We then offer a selection of evaluation metrics for analyzing network performance on the DALES Objects dataset. Finally, we provide several dataset statistics and identify potential challenges when working with the DALES Objects dataset.

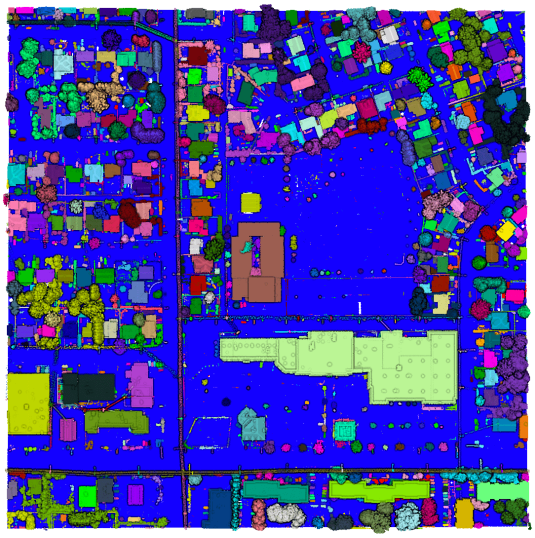

FIGURE 1. Example of a scene containing hand-labeled human-made objects from the DALES Objects dataset. We mark each instance with a random RGB color code.

II. RELATED WORKS

Benchmarking has an essential place in deep learning [24]. Because of the large amount of data required to train a supervised deep learning network, having sets of high-quality labeled data allows for training and enables us to compare and contrast different networks' performance [25]. The first well-known benchmarks were image-based, focused on image classification [26]. As the amount of benchmark datasets grows, so do the dataset-specific tasks. Recent advancements in lidar sensor research, paired with interest in autonomous driving research, have seen rapid growth in 3D point cloud datasets [27], [28]. Although the number of point cloud benchmarks is multiplying, they have primarily focused on the semantic segmentation task. This section will discuss state of the art in point cloud benchmarks and the datasets' availability for the specific task of instance segmentation. We can see a non-exhaustive list of some of the most prominent 3D point cloud benchmarks in Table 1

A. POINT CLOUD BENCHMARKS

There are a large number of semantic segmentation datasets for point clouds [29]. We separate benchmarks into two types of data; indoor and outdoor. We also note a small number of benchmark datasets covering synthetic scenes and objects such as ModelNet40 [30]; these datasets are made with generative software such as CAD. Although helpful, there are significant differences, like high density and lack of occlusions and background, which are not comparable to natural scenes. In this section, we will focus on those datasets made from real-world locations.

Indoor settings include datasets such as S3DIS [13], Matterport3D [15], and ScanNet [12]; these scans are high density but cover a relatively small area, focusing on residential or commercial settings, such as homes or offices. These datasets typically have many semantic categories spanning familiar household objects like chairs, tables, and computer monitors. Due to indoor data's nature, the object sizes are less varied than those in outdoor scenes.

Sensor types for indoor datasets are RGB-D sensors or point clouds sampled from a 3D mesh. While not as accurate as those taken with a lidar sensor, they are typically much denser. They usually provide additional features like RGB color information that is not ordinarily available in lidar scans.

A more significant portion of the semantic segmentation benchmark datasets covers outdoor scenes taken with lidar sensors. These outdoor scenes have a higher level of difficulty due to variation in scene types, class imbalances, and more

TABLE 1. A non-exhaustive summary of existing datasets for semantic and instance segmentation in 3D point clouds.

Name	Year	# Labeled Points	# Classes	RGB	Sensor	Semantic	Instance
Oakland [9]	2009	1.6M	5	✗	MLS	✓	✗
ISPRS [10]	2012	1.2M	9	✗	ALS	✓	✗
Paris-rue-Madame [11]	2014	20M	17	✗	MLS	✓	✓
ScanNet [12]	2017	-	20	✓	RGB-D	✓	✓
S3DIS [13]	2017	273M	13	✓	Matterport	✓	✓
Semantic3D [14]	2017	4000M	8	✓	TLS	✓	✗
Matterport3D [15]	2017	-	40	✓	Matterport	✓	✓
Paris-Lille-3D [16]	2018	143M	9	✗	MLS	✓	✓
Semantic KITTI [17]	2019	4549M	25	✓	MLS	✓	✗
DublinCity [18]	2019	260M	13	✗	ALS	✓	✓
LASDU [19]	2020	3.2M	5	✗	ALS	✓	✗
Toronto-3D [20]	2020	78.3M	8	✓	MLS	✓	✗
Campus3D [21]	2020	937.1M	14	✓	UAV Photogrammetry	✓	✓
H3D [22]	2021	73M	11	✓	ALS	✓	✗
ArCH [23]	2020	130M	10	✓	TLS/Photogrammetry	✓	✗
DALES Objects (Ours)	2021	492M	8	✗	ALS	✓	✓

significant differences in point density because of sensor distances.

Of these outdoor scenes, we can examine them by looking at the type of lidar; mobile, terrestrial, and aerial. Mobile lidar is the most common of these collection types because of the recent popularity of autonomous driving, with datasets like Oakland [9], Paris-rue-Madame [11], IQmulus [31], Paris-Lille 3D [16], Semantic KITTI [17], and Toronto-3D [20].

There are significantly fewer terrestrial daatasets, such as Semantic3D [14] and [23] taken with a lidar scanner from a fixed point. Aerial lidar datasets include ISPRS [10], H3D [22], DublinCity [18], and DALES [1]. Despite differences in sensor types, viewpoints and settings, these benchmark datasets contain similar categories, such as ground, vegetation, buildings, and vehicles.

B. INSTANCE SEGMENTATION BENCHMARKS

Instance segmentation in point clouds is significantly more complex than semantic segmentation because it requires a more nuanced understanding of individual points and their relationship to the scene as a whole [32]. The challenge of distinguishing between semantic categories is magnified by distinguishing between different items in the same semantic category.

Instance segmentation benchmark datasets have less representation than their semantic segmentation counterparts. In the instance segmentation space, there are far fewer available datasets. We can also split these into indoor and outdoor data types. Indoor datasets include those like S3DIS [13], ScanNet [12], [15] and SceneNN [33], taken with RGB-D or other non-lidar scanners. Outdoor datasets are mostly focused on urban scenes; with datasets including Campus3D [21], Paris-Lille3D [16] and DublinCity [18].

There is a significant gap in the number of available benchmarks and diversity of scenes when comparing semantic segmentation benchmarks and instance segmentation benchmarks. Table 1 shows a non-exhaustive comparison of point clouds datasets for segmentation. We can see that semantic labels are prevalent while instance labels are less so. With only [18] providing instance labels from an aerial lidar sensor. Because of the lack of available benchmark datasets, there are significantly fewer instance segmentation networks. On the semantic segmentation side, we see that there can be a considerable difference in the performance of a network when classifying different types of scenes. A robust network would perform equally well on indoor and outdoor settings and various kinds of sensors.

There are several reasons for the lack of instance segmentation labels. The first is that it is much more difficult to hand-label individual objects than broad object categories. Many semantic segmentation datasets have presented semi-automatic methods for labeling these object categories. These semi-automatic methods are less common in the instance segmentation space than the semantic segmentation space. Another reason for the lack of outdoor scenes is that it is easier to label human-made objects with distinct object boundaries than natural things like ground or vegetation, whose boundaries can be ambiguous or hard to distinguish.

This paper aims to increase the amount and diversity of the available instance segmentation datasets by providing our DALES Objects instance segmentation dataset. The DALES Objects dataset provides semantic and instance segmentation labels in an outdoor environment, taken with an aerial lidar sensor. We believe that it can be a valuable resource because of its size and because it contains diverse scenes, covering both rural and urban environments, in contrast to the currently available datasets.

III. DALES OBJECTS: THE DATA SET

We want to infer a class label and an object label for a given list of points in an aerial lidar tile. The class label is one of the previously defined eight semantic classes. The object label represents one individual instance of an object belonging to the semantic category. Object labels can have any object id but must contain the same defined point groupings.

A. INITIAL DATA COLLECTION

Our data was collected over the City of Surrey in British Columbia, Canada, over two days. We can see satellite imagery overlayed with the chosen tiles in Figure 2. The data was collected using a Riegl Q1560 dual-channel system with a flight altitude of 1300 meters above ground and a speed of 140kts. The sensor's scan rate was 800khz with a line spacing of 380 meters, a total line length of 1884 km, and a minimum overlap of 400%. Each area collected a minimum of 5 laser pulses per meter in the north, south, east, and west directions, with a goal of a minimum of 20 ppm and minimizing occlusions from each direction, with the possibility of multiple returns. The lidar swaths were calibrated using BayesStripAlign 2.0 software and registered, taking both relative and absolute errors into account and correcting for IMU altitude and positional errors. Each cross-section is then manually checked to verify the automatic results.

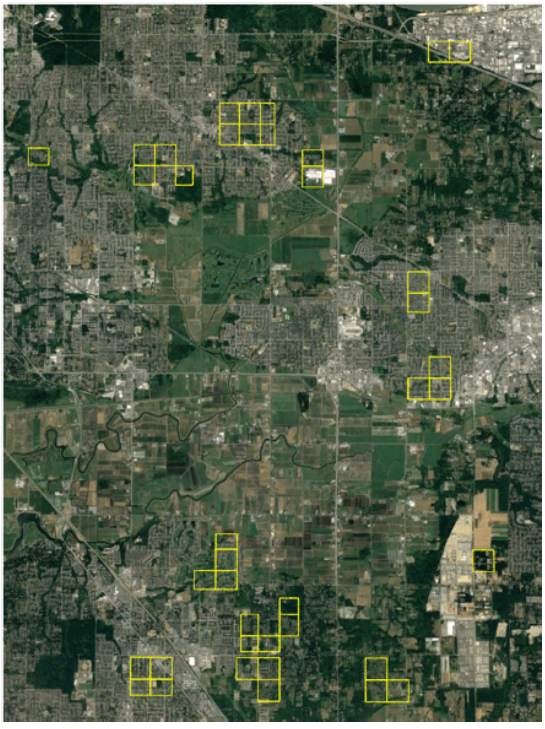

FIGURE 2. Satellite imagery of the City of Surrey, in British Columbia, Canada, overlayed with our tiles selection, chosen tiles areas are outlined in yellow.

The final data spans over 330 square kilometers and has a final data projection of UTM zone 10N with a horizontal datum of CVGD28 and uses the metro Vancouver Geoid. We performed an accuracy assessment using the ground control points and a visual inspection, matching the corners and hard surfaces from each pass. We determine the mean error to be ± 8.5 cm at 95% confidence for the hard surface vertical accuracy.

Due to the considerable distance between the sensor and the objects that occur in aerial lidar, the laser pulse diameter can become much larger by the time it hits the object. Thus it is common to have multiple returns, where a single pulse can reflect off more than one item and record two or more points. This phenomenon was prevalent in the dataset, sometimes recording up to six hits per pulse, especially in vegetation areas. The presence of multiple returns increased the resolution of our dataset. Our final average point density was slightly over 50 points per meter (ppm).

Upon receiving the total final data, we decided to focus our labeling efforts on 40 tiles, 500 meters by 500 meters, each. We examined the publically available satellite data for regions of interest. We picked these forty tiles to include diverse groups of scenes choosing to focus on commercial, urban, rural, and suburban scenes. On average, each tile contains around twelve million points. These tiles do not have any overlapping portions, with all locations being unique. We describe the four scene types below:

- Commercial: warehouses and office parks
- Urban: high rise buildings, greater than four stories
- Rural: natural objects with a few scattered buildings
- Suburban: concentration of single-family homes

B. PREPROCESSING

Our first step was to perform a noise removal on our point clouds. We found some small amounts of spectral noise throughout the cloud and used a statistical outlier removal to identify and remove these points. The filter examines each point and identifies the K nearest neighbors. After determining the neighbors, we calculate each of these neighbors' average distances to our point of interest. If any length is above a pre-determined threshold, we remove it from the point clouds. We used ten nearest neighbors for this particular dataset and chose our distance threshold to be 5 meters. This filtering only released a small number of points, on average 11 points per tile, but it successfully reduced each tile bounding box by an average of 50% by volume.

C. SEMANTIC LABELING

After the initial selection of our forty tiles, we first focused on adding the semantic labels. We will discuss the semantic labeling briefly, but more information can be found [1]. After much discussion, we decided on the following eight labels; ground(1), vegetation(2), cars(3), trucks(4), powerlines(5), fences(6), poles(7), buildings(8). When choosing these object categories, a priority was to have distinct differences between the categories. Unlike similar datasets, we do not have any ambiguous categories, such as high and low vegetation or human-made versus natural categories.

Additionally, because of the noise removal step, all of our categories are distinct objects. We avoid labeling any points that result from noise and not physically present in the original scene. A non-exhaustive list of items from each category is listed below:

- Ground: impervious surfaces, grass, rough terrain
- Vegetation: trees, shrubs, hedges, bushes
- Cars: sedans, vans, SUVs
- Trucks: semi-trucks, box-trucks, recreational vehicles

- Power lines: transmission and distribution lines
- Poles: power line poles, light poles, and transmission towers
- Fences: residential fences and highway barriers
- Buildings: residential, high-rises and warehouses

D. INSTANCE LABELING

After determining each point's semantic labels, we separated each tile into separate layers, with each layer only containing points of the same semantic class. We then performed an initial euclidean clustering on each semantic layer. We define a distinct cluster as a set of points where the minimum distance between each point in the cluster to at least one other point in the cluster is below our set distance threshold.

When this criterion is met, we determine this set of points to be a unique cluster. We estimate the distances between each point and its neighbors using a kd-tree representation as outlined here [34]. The euclidean clustering algorithm is as follows:

Algorithm 1 Rough Euclidean Clustering

Require: For each point p_i in the point cloud P we calculate k neighbors
Require: We define an empty list of clusters C, and a queue to be checked, Q
for p_i in P **do**
  Add p_i to Q
  for p in Q **do**
    Get K neighbors for p_i
    for p_k in K **do**
      if p_k not in a cluster **then**
        Add p_k to Q
      end if
    end for
  end for
  Label all points in Q as a new cluster in C and clear Q
end for

We change the euclidean clustering radius for each semantic layer based on the objects' average size within that category. Typically, more oversized items require larger radii. The radius values are as follows: buildings: 4 meters, cars, and trucks: 1.5 meters; fences: 3 meters, power lines: 0.5 meters, poles: 5 meters, vegetation: 1 meter. The results of this euclidean clustering algorithm are 'rough clusters.' We can see examples of these rough clusters from each category in Figure 3

Once we calculate the rough clusters, we proceed to the manual labeling step. For this step, we use the Point Cloud Processing ToolKit (PPTK), which uses the qT library to display the point clouds dynamically. We define the following workflow for each labeler, with each semantic tile layer defined as a ''task.''

First, the labeler loads the task into the Point Cloud Labeling Tool. The labeler will look at each object cluster one by one. For each object, we display the object cluster in red on the viewer. We also show all points within a ten-meter radius around the group; we show all other sets in a different random RGB color code. The labeler will then indicate whether he wants to accept or reject the label. If the label is validated, then we move on to the next instance. If the labeler denies the label, she can indicate whether she wants to correct it or mark it for review. If the labeler chooses to rectify the object, then the labeler will be allowed to reselect the object cluster and update the cluster list. If the labeler marks an item for review, the set of points is flagged and reviewed by another labeler.

Once each task is completed and the cluster list is updated, we give the new clusters to a second labeler who repeats the task. Once at least two labelers visit the semantic layers from each tile, the semantic layers are recombined, and the cluster object ids are updated to make our final point clouds.

We gave our labelers several key directives for object labeling that we will discuss in this paragraph. The first is that we choose only to consider object labels for human-made objects. In total, DALES Objects contains eight semantic classes, six of which are human-made (buildings, cars, trucks, fences, powerlines, and poles) and two natural categories (vegetation and ground). We can note that the ground contains both natural objects, like grass, and human-made objects like asphalt. Still, we include it in the natural category because its features more closely align with this category.

In the non-man-made categories, the object labels are less straightforward. We choose to label the ground as one object throughout the entire scene. Although our resolution is very high, we did not have the necessary density to mark individual vegetation objects. To label vegetation with a high degree of accuracy, a key element is to have enough point density to see separate tree trunks. We have found that this data does not have that resolution in all cases, especially in large forest cover areas. Because we did not have the available contextual information, we did not hand-label the vegetation layer. The object ids from the vegetation layer in this dataset will be the rough category labels from the euclidean clustering. Due to these peculiarities in the natural objects' labels, we provide the labels but do not include them in the overall evaluations.

After the labeling is completed and examined by a minimum of two human labelers, we go back and calculate an object's average size in each category. We then look again at clusters with a total number of points less than 25% of an average item in that category and look for small objects that may be artifacts from updating the labels and delete them.

We also discuss some labeling choices in the human-made categories. We chose to label free-standing structures as one object id instead of considering aspects like individual units or buildings' addresses. An example of this would be a row of six physically connected townhomes. Our labeling method would label this as one building instead of six individual units. We chose this labeling method because we did not want to rely on additional and possibly conflicting data sources like satellite imagery and address databases.

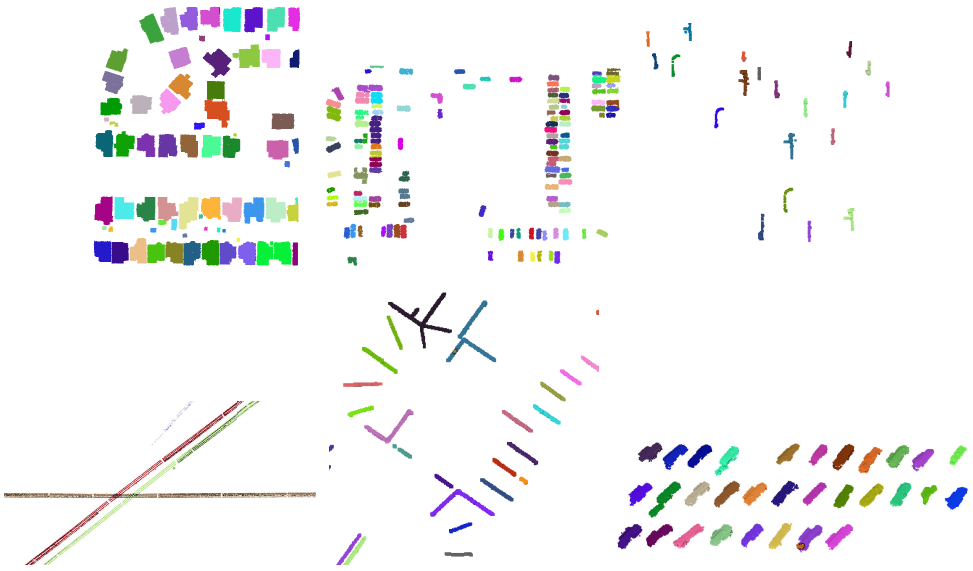

FIGURE 3. Example of the outputs of the rough euclidean clustering for the buildings and cars classes. Each separate cluster is represented by a unique RGB color code. We can see the majority of objects have good initial clustering, however many objects, especially those in close physical proximity, need further hand-labeling.

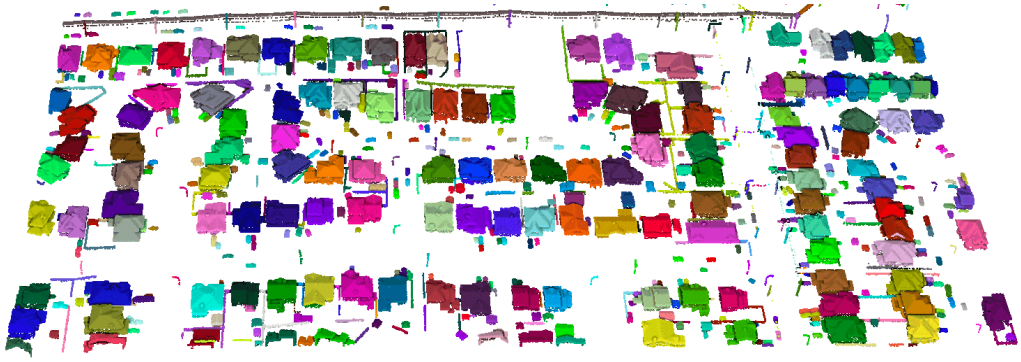

FIGURE 4. Example of a scene containing hand-labeled human-made objects from the DALES Objects dataset. We mark each instance with a random RGB color code.

The second significant labeling choice in the human-made objects was with the power lines. There were several configurations that we considered. The first was to label powerlines individually or as a group. A typical design for powerlines is to have several individual power lines in a horizontal orientation. We considered labeling each line as an object or including all of the lines as a set. The second consideration was whether to end the powerline object once it intersects with a pole or to continue the item across the entire run of the scene. After consulting with utility management professionals, we choose to label the powerlines as ''runs'' instead of individual lines. We also decided to continue the object labels through the pole intersections. A powerline object ends only when the line changes directions. Once there is a change of direction, we consider it a new powerline object. We can see an example of the labeled human-made objects in Figure 4. A selection of tiles with intensity, semantic labels, and instance labels is shown in Figure 5

E. DATA SET STATISTICS

This section provides some statistics on the dataset's overall contents and makes predictions about the potential difficulties. Overall the dataset has around 492 million points, with the ground and vegetation being the most prominent categories. The human-made types make up approximately 83 million points total, with the largest of these being the building category, followed by cars and fences. Table 2 shows the distribution of the number of points in each semantic category and the differences between training and testing.

Table 2 also examines the average size of the human-made objects in terms of the number of points; the buildings are the largest category, with the average size being around 11 thousand points. The poles and cars are the smallest with 237 and 288 points, respectively. Next, we look at the total number of objects in the human-made category. The most significant number is cars at 11 thousand individual items, and the smallest being powerlines with 228 unique things. Table 2 shows

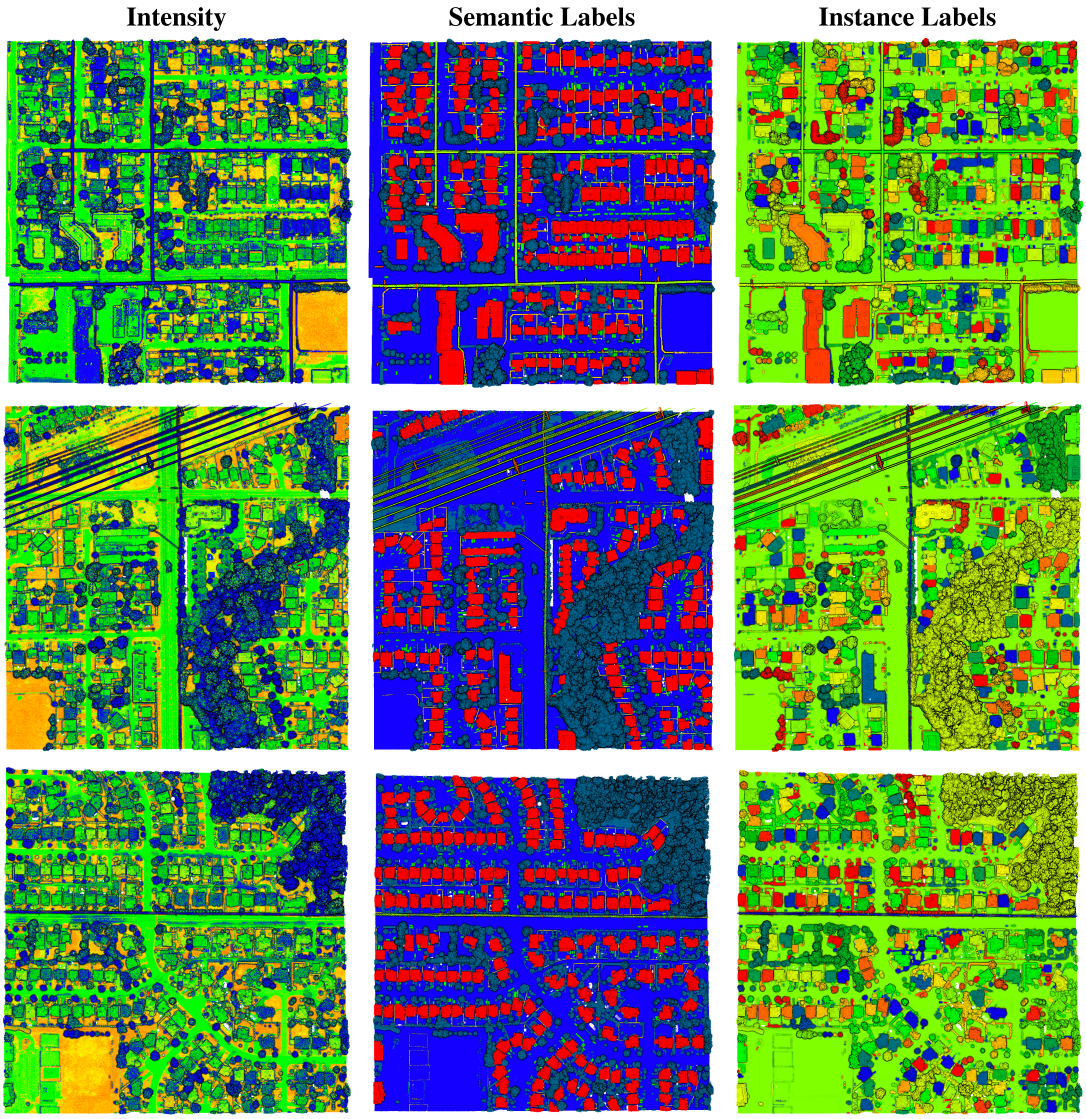

FIGURE 5. Example of our DALES tiles. Intensity images are shown in the left-most column. Semantic classes are in the middle column and are labeled by color; ground (blue), vegetation (dark green), power lines (light green), poles (orange), buildings (red), fences (light blue), trucks (yellow), cars (pink). Finally, instance labels are in the right-most column. Human-made objects are labeled with a random RGB color-code.

TABLE 2. Overview of the per class statistics of the DALES Objects dataset.

class	*# of points*	*# training points*	*# test points*	*# of objects*	*# of points/object*
ground	246.9M	178M	68.9M	–	–
vegetation	159M	118.4M	40.6M	–	–
car	4.1M	3.2M	900k	11461	288
truck	879k	728k	151k	845	1039
powerline	994k	773K	221k	228	4361
fence	2.1M	1.5M	61k	3449	599
pole	262k	200k	62k	1107	237
buildings	78.7M	55.7M	23M	6858	11483

the breakdown of the average number of objects and average object size. Since these scenes are all naturally occurring, the most challenging aspect of the DALES Objects dataset will be the significant class disparities, both in the number of overall points and the number of object instances.

F. FINAL DATA FORMAT

After labeling the dataset, we provide the entire dataset in two formats. The dataset is split randomly into a rough 70/30 training/testing split. The first is a binary ply file that contains six data categories, x,y,z, intensity, semantic class,

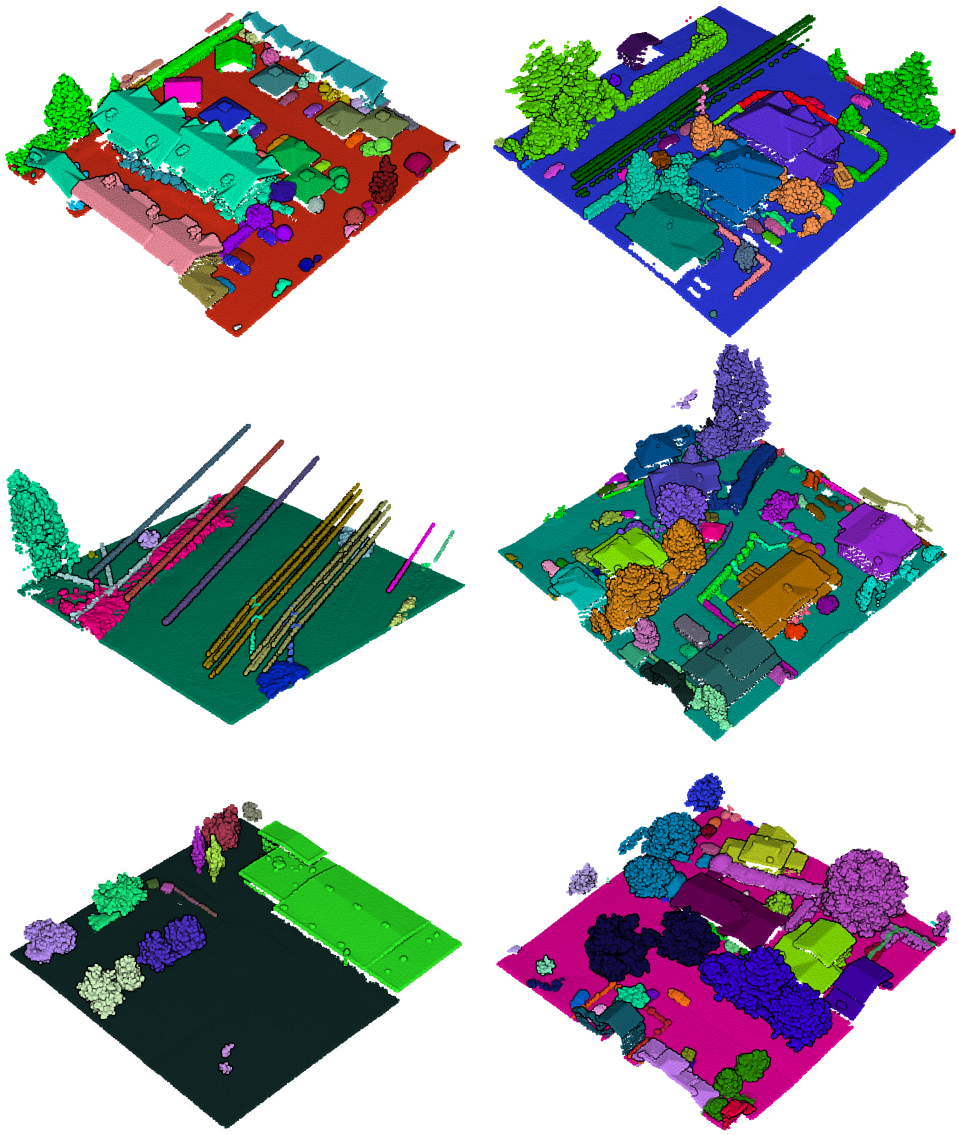

FIGURE 6. Snapshots of our DALES objects scenes. Instance labels are labeled with a random RGB color-code.

and instance class. The second format has the same points, but we construct it in the style of the S3DIS dataset, with each tile as a parent folder and each object stored as a text file within that folder. We hope that by providing this dataset in these two formats, researchers can quickly test DALES Objects on existing networks. The final point clouds, in both forms, can be found on our website: go.udayton.edu/dales3d.

IV. EVALUATION METRICS

We provide the following guidelines for evaluating network performance on our DALES Objects dataset, following the lead of other similar 3D point cloud segmentation datasets.

We assess the semantic segmentation using Intersection over Union (IoU) as our primary metric and overall accuracy as a secondary metric. We calculate the IoU per class using the following equation, where C is an NxN confusion matrix, and i represents the ground truth class, and j represents the prediction class. Once we calculate the per class IoU, we can then take the mean of each per-class IoU to form the overall metric of mean IoU.

$$IoU_i = \frac{c_{ii}}{c_{ii} + \sum_{j \neq i} c_{ij} + \sum_{k \neq i} c_{ki}} \tag{1}$$

$$\overline{IoU} = \frac{\sum_{i=1}^{N} IoU_i}{N} \tag{2}$$

We also calculate the overall accuracy according to the following formula:

$$OA = \frac{\sum_{i=1}^{N} c_{ii}}{\sum_{j=1}^{N} \sum_{k=1}^{N} c_{jk}} \tag{3}$$

For instance segmentation, we use the mean Average Precision (mAP) and mean Recall (mRec). We first calculate the Average Precision for all classes individually. We define a true positive prediction as an overlapping IoU greater or equal to 50%. Similarly, a false positive would have an IoU of less than 50%. A false negative is associated with no detection. Once we tally the true positives, false positives, and false negatives, we can construct the per class precision and recall as follows:

$$AP_i = \frac{c_{ii}}{c_{ii} + \sum_{k \neq i} c_{ki}} \tag{4}$$

$$Rec_i = \frac{c_{ii}}{c_{ii} + \sum_{j \neq i} c_{ji}} \tag{5}$$

Once we calculate the Average Precision (AP) and Average Recall (AR) for each class, we can then average the metrics across all categories to get our final mAP and mRec.

One important note for evaluation is that we only calculate and report the mAP and mRec across our six human-made categories instead of the eight categories because of our labeling strategy, which focuses on human-made objects. Researchers should feel free to evaluate natural classes. However, they should use only the mAP and mRec from the six human-made categories compared to other networks.

V. CONCLUSION

This paper presented DALES Objects, a large-scale dataset, for instance segmentation in aerial lidar. While semantic segmentation datasets have become increasingly popular, their instance segmentation counterparts are quite limited. DALES Objects is one of the most extensive lidar datasets to provide semantic and instance segmentation labels in large outdoor urban and rural scenes. In addition to the labels, we also offered the points in their original UTM Zone 10N projection and included intensity information. We discussed the challenges and difficulties of this dataset and offered suggested evaluation metrics to assess a network's performance on the DALES Objects dataset. We hope that this benchmark will be a resource for the 3D deep learning community and expand the research in the instance segmentation field to include both lidar and outdoor scenes.

ACKNOWLEDGMENT

This article's data set contains information licensed under the Open Government License – City of Surrey.

REFERENCES

[1] N. Varney, V. K. Asari, and Q. Graehling, "DALES: A large-scale aerial LiDAR data set for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 186–187.

[2] Y. LeCun. (1998). *The MNIST Database of Handwritten Digits*. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[3] A. Krizhevsky, V. Nair, and G. Hinton. (May 2010). *CIFAR-10 (Canadian Institute for Advanced Research)*. [Online]. Available: http://www.cs.toronto.edu/kriz/cifar.html

[4] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 740–755.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[6] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 843–852.

[7] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3D point clouds: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jun. 29, 2020, doi: 10.1109/TPAMI.2020.3005434.

[8] Y. Guo, J. Zhang, M. Lu, J. Wan, and Y. Ma, "Benchmark datasets for 3D computer vision," in *Proc. 9th IEEE Conf. Ind. Electron. Appl.*, Jun. 2014, pp. 1846–1851.

[9] D. Munoz, J. A. Bagnell, N. Vandapel, and M. Hebert, "Contextual classification with functional max-margin Markov networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 975–982.

[10] F. Rottensteiner, G. Sohn, J. Jung, M. Gerke, C. Baillard, S. Benitez, and U. Breitkopf, "The ISPRS benchmark on urban object classification and 3D building reconstruction," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vols. I–3, no. 1, pp. 293–298, Jul. 2012.

[11] A. Serna, B. Marcotegui, F. Goulette, and J.-E. Deschaud, "Paris-rue-Madame database: A 3D mobile laser scanner dataset for benchmarking urban detection, segmentation and classification methods," in *Proc. 4th Int. Conf. Pattern Recognit., Appl. Methods*, 2014, pp. 1–7.

[12] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3D reconstructions of indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5828–5839.

[13] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese, "3D semantic parsing of large-scale indoor spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1534–1543.

[14] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, "Semantic3D.Net: A new large-scale point cloud classification benchmark," 2017, *arXiv:1704.03847*. [Online]. Available: http://arxiv.org/abs/1704.03847

[15] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niebner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3D: Learning from RGB-D data in indoor environments," in *Proc. Int. Conf. 3D Vis. (3DV)*, Oct. 2017, pp. 1–25.

[16] X. Roynard, J.-E. Deschaud, and F. Goulette, "Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification," *Int. J. Robot. Res.*, vol. 37, no. 6, pp. 545–557, May 2018.

[17] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9297–9307.

[18] S. M. I. Zolanvari, S. Ruano, A. Rana, A. Cummins, R. E. D. Silva, M. Rahbar, and A. Smolic, "DublinCity: Annotated LiDAR point cloud and its applications," 2019, *arXiv:1909.03613*. [Online]. Available: https://arxiv.org/abs/1909.03613

[19] Z. Ye, Y. Xu, R. Huang, X. Tong, X. Li, X. Liu, K. Luan, L. Hoegner, and U. Stilla, "LASDU: A large-scale aerial LiDAR dataset for semantic labeling in dense urban areas," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 7, p. 450, 2020.

[20] W. Tan, N. Qin, L. Ma, Y. Li, J. Du, G. Cai, K. Yang, and J. Li, "Toronto-3D: A large-scale mobile LiDAR dataset for semantic segmentation of urban roadways," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 202–203.

[21] X. Li, C. Li, Z. Tong, A. Lim, J. Yuan, Y. Wu, J. Tang, and R. Huang, "Campus3D: A photogrammetry point cloud benchmark for hierarchical understanding of outdoor scene," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 238-246.

[22] M. Kölle, D. Laupheimer, S. Schmohl, N. Haala, F. Rottensteiner, J. D. Wegner, and H. Ledoux, "The Hessigheim 3D (H3D) benchmark on semantic segmentation of high-resolution 3D point clouds and textured meshes from UAV LiDAR and multi-view-stereo," 2021, *arXiv:2102.05346*. [Online]. Available: https://arxiv.org/abs/2102.05346

[23] F. Matrone, A. Lingua, R. Pierdicca, E. S. Malinverni, M. Paolanti, E. Grilli, F. Remondino, A. Murtiyoso, and T. Landes, "A benchmark for large-scale heritage point cloud semantic segmentation," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 43, pp. 1419–1426, 2020.
[24] S. Shi, Q. Wang, P. Xu, and X. Chu, "Benchmarking state-of-the-art deep learning software tools," in *Proc. 7th Int. Conf. Cloud Comput. Big Data (CCBD)*, Nov. 2016, pp. 99–104.
[25] A. Nguyen and B. Le, "3D point cloud segmentation: A survey," in *Proc. 6th IEEE Conf. Robot., Autom. Mechatronics (RAM)*, Nov. 2013, pp. 225–230.
[26] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural Comput.*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017.
[27] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: Systems and algorithms," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 163–168.
[28] T. Raj, F. H. Hashim, A. B. Huddin, M. F. Ibrahim, and A. Hussain, "A survey on LiDAR scanning mechanisms," *Electronics*, vol. 9, no. 5, p. 741, Apr. 2020.
[29] Y. Xie, J. Tian, and X. X. Zhu, "Linking points with labels in 3D: A review of point cloud semantic segmentation," 2019, *arXiv:1908.08854*. [Online]. Available: http://arxiv.org/abs/1908.08854
[30] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1912–1920.
[31] B. Vallet, M. Brédif, A. Serna, B. Marcotegui, and N. Paparoditis, "TerraMobilita/iQmulus urban point cloud analysis benchmark," *Comput. Graph.*, vol. 49, pp. 126–133, Jun. 2015.
[32] A. M. Hafiz and G. M. Bhat, "A survey on instance segmentation: State of the art," *Int. J. Multimedia Inf. Retr.*, pp. 1–19, Jul. 2020.
[33] B.-S. Hua, Q.-H. Pham, D. T. Nguyen, M.-K. Tran, L.-F. Yu, and S.-K. Yeung, "SceneNN: A scene meshes dataset with aNNotations," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 92–101.
[34] A. J. Trevor, S. Gedikli, R. B. Rusu, and H. I. Christensen, "Efficient organized point cloud segmentation with connected components," in *Proc. Semantic Perception Mapping Explor. (SPME)*, 2013, pp. 1–6.

NINA M. SINGER (Graduate Student Member, IEEE) is currently pursuing the Ph.D. degree with the University of Dayton. She is currently a Research Scientist. Her research interests include using deep learning to aid in geospatial intelligence and wide-area surveillance, with a focus on lidar and other 3D modalities.

VIJAYAN K. ASARI (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the Indian Institute of Technology, Madras, in 1994. He is currently a Professor in electrical and computer engineering and the Ohio Research Scholars Endowed Chair in wide area surveillance with the University of Dayton, Dayton, OH, USA. He is also the Director of the Vision Laboratory, University of Dayton, a Center of Excellence for Computational Intelligence and Machine Vision. He has published more than 700 research articles, including an edited book on wide area surveillance and 116 peer-reviewed journal articles in the areas of image processing, computer vision, pattern recognition, machine learning, deep learning, and artificial neural networks. He is an Elected Fellow of SPIE and a co-organizer of several IEEE and SPIE conferences and workshops. He received several awards for teaching, research, advising, and technical leadership that include the University of Dayton Vision Award for Excellence, in August 2017, the Outstanding Engineers and Scientists Award for Technical Leadership from The Affiliate Societies Council of Dayton, in April 2015, and the Sigma Xi George B. Noland Award for Outstanding Research, in April 2016.