NEEDLE IN A HAYSTACK: FEASIBILITY OF IDENTIFYING SMALL SAFETY ASSETS FROM POINT CLOUDS USING DEEP LEARNING

G. Anjanappa[a], S. Nikoohemat[a], S. Oude Elberink[a], R. L. Voûte[b], V. V. Lehtola[a, *]

[a] Dept. of Earth Observation Science, Faculty ITC, University of Twente, Enschede, The Netherlands
[b] CGI Inc, The Netherlands

KEY WORDS: Asset management, point clouds, Deep learning, indoor, Scene segmentation

ABSTRACT:

Asset management systems are beneficial for maintaining building infrastructure and can be used to keep up-to-date records of relevant safety assets, such as smoke detectors, exit signs, and fire extinguishers. Existing methods for locating and identifying these assets in buildings primarily rely on surveys and images, which only provide 2D locations and can be tedious for large-scale structures. Indoor point clouds, which can be captured quickly for buildings using mobile scanning techniques, can provide us with 3D asset locations. In this paper, we study the feasibility of using 3D point clouds of buildings combined with deep learning techniques to identify safety-related assets, particularly small-sized assets like fire switches and exit signs. We adopt the state-of-the-art Deep Learning network, Kernel Point-Fully Convolutional Network (KP-FCNN), to identify these assets through semantic segmentation. Using the obtained results, we create a 3D-geometry model of the building with assets pinpointed, providing scene semantics and delivering more value. Our method is tested using three different point cloud datasets obtained from a depth camera, a mobile laser scanner, and an iPhone lidar sensor.

1. INTRODUCTION

Asset identification is essential for asset management systems of buildings, which in particular must be performed regularly to maintain up-to-date information (name and location) of relevant safety-related assets. Existing methods like manual surveys and automated techniques using images only provide 2D locations of assets, which are also tedious and time-consuming for large-scale buildings [Warsop and Singh, 2010, Kostoeva et al., 2019]. However, using point clouds, a de-facto data for three-dimensional (3D) representation of the real world, we can obtain precise 3D locations of the assets within a building [Chen, 2019]. Further, the availability of low-cost 3D sensors makes it easier to acquire point clouds of large-scale buildings recurrently with minimal difficulty [Lehtola et al., 2017].

In using deep learning (DL) methods with 3D point clouds, we see the possibility of automating asset identification by learning essential features through the colors and shapes of assets [Goodfellow et al., 2016, p. 96]. In particular, object detection and segmentation (instance or semantic) techniques can obtain both the 3D location and name of assets from point clouds at once [Guo et al., 2020]. In our case, the segmentation method would identify assets and also provide scene semantics, delivering more value than just object detection [Anjanappa, 2022]. Since the safety-related assets are standalone objects, semantic segmentation would be enough to identify them and would not require instance segmentation. Also, semantic segmentation fits well to process large point clouds, which often have noise and other scanning imperfections [Lehtola et al., 2017].

Existing DL networks for semantic segmentation [Qi et al., 2016, Qi et al., 2017, Wang et al., 2019] for buildings mainly focus on planar structures and big objects. Recently, [Hossain. et al., 2021] used PointNet++ [Qi et al., 2017] and point clouds of buildings to identify safety-related assets. However, they were unsuccessful in identifying small-sized assets directly, leading them to use images first to detect assets and then transfer the labels to point clouds to locate them in 3D.

*Corresponding author: v.v.lehtola(at)utwente.nl

On the data level, the total safety-related assets present throughout a building are limited, which likely constrains the points representing them in the building's point cloud, as shown in Figure 1. The point counts for small assets like fire switches and exit signs are further reduced due to their smaller physical sizes. While the point cloud of a building can be huge and irregular [Lehtola et al., 2017], for safety-related assets, we have the following challenges:

- *"Finding-the-needle-in-a-haystack"*: Processing huge point clouds of buildings (haystack) to find and identify various small-sized assets (needles) as seen in Figure 1 [Anjanappa, 2022].
- *"Class-imbalanced"* data: Point cloud of a building with the majority (ceiling, floor, and wall) and minority (safety-related assets) classes, affecting the performance of DL networks [Johnson and Khoshgoftaar, 2019].

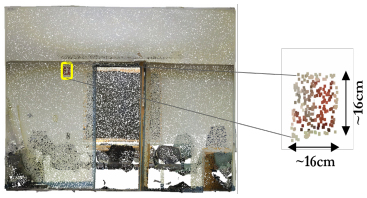


Figure 1: Point cloud of a room with a fire switch (small asset) in the yellow bounding box, measuring approximately 16cmx16cm.

As our first contribution, we present a study on the feasibility of identifying small-sized assets in 3D point clouds of buildings with deep learning, based on an MSc thesis work [Anjanappa, 2022]. We adopt KP-FCNN [Thomas et al., 2019] semantic segmentation network to predict point-wise labels and then cluster them to get asset instances. Our hypothesis is that the introduction of kernels to unordered point clouds brings a necessary amount of order to succeed. Specifically, the study focuses

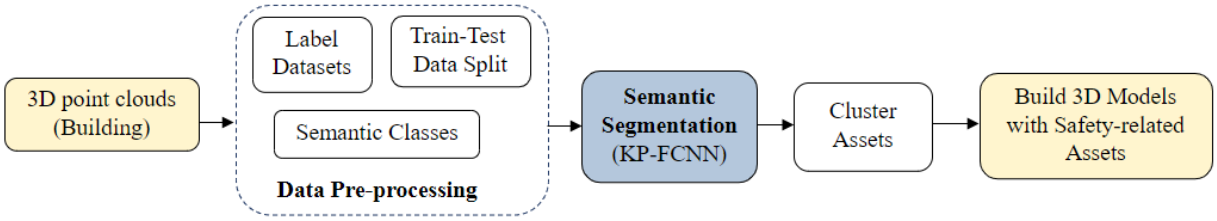


Figure 2: Methodology - Overall workflow.

on safety-related assets commonly found in buildings, like, ceiling light, fire switch, fire extinguisher, exit sign, ceiling ventilation duct, window, and door. In our case, the smallest safety-related asset identified is a fire switch measuring approximately 16cmx16cm in Stanford indoor dataset (S3DIS) [Armeni et al., 2016] as shown in Figure 1.

As this is a feasibility study, we carefully evaluate the robustness of the designed method on point cloud datasets obtained from three different 3D sensors, namely, depth camera, mobile laser scanner, and iPhone lidar sensor. Further, we estimate the acceptable point cloud density (or sparsity) for the proposed method to be able to detect small safety-related assets. So, we artificially decreased the points on these assets by down-sampling the point cloud and then tested the outcome.

As our second contribution, using the enriched point clouds from test datasets, we build sample 3D geometry models with assets pin-pointed. This delivers more value and provides contextual information about the scene [Anjanappa, 2022]. As a result, for example, the height of a fire switch from the floor is measurable, and such 3D models are likely beneficial for (i) safety inspectors or facility managers to monitor the safety infrastructure, and for (ii) first responders during emergencies.

Additionally, our efforts include manual labeling work. Available labeled point cloud datasets like ShapeNet Part [Yi et al., 2016] and S3DIS [Armeni et al., 2016] do not have labels for safety-related assets and, therefore, can not be directly used for deep learning to identify these assets [Hossain. et al., 2021]. Hence, we manually labeled the datasets used for training and testing the network.

The paper is organized as follows: Section 2. presents the related work. In Sections 3. and 4., we present our proposed method and the data used for evaluating it, respectively. Section 5. describes the experiments conducted with their results. The paper ends with the discussion and conclusion in Sections 6. and 7..

2. RELATED WORK

Following the keywords on 3D point cloud processing methods with deep learning, We review the works of object detection, semantic segmentation, and instance segmentation. Based on [Liu et al., 2019, Bello et al., 2020, Guo et al., 2020], for point clouds, *Object detection* identifies and localizes objects through 3D bounding boxes and assigns labels to only the detected point sets; *Semantic segmentation* is a process where each point in the point cloud is assigned to a semantic class; *Instance segmentation* distinguishes each point into different semantic class, and the semantic classes are further separated into individual instances.

Object detection networks: They are mainly categorized into single-shot [Lang et al., 2019] and region proposal (RP) based [Shi et al., 2019] methods. RP-based networks sometimes use point-wise classification networks as the first step to generate object proposals [Guo et al., 2020]. Though object detection methods would identify safety-related assets, they would not provide any information about the scene.

Semantic segmentation networks: Based on how the DL networks consume and process a point cloud, these networks can broadly be of two types, indirect and direct methods. Indirect methods like [Tchapmi et al., 2017] use transformations and intermediate representations of point clouds with DL networks resulting in data loss and discretization errors [Guo et al., 2020]. In contrast, direct methods do not use data conversions, among which the point-based methods work directly on 3D point clouds [Qi et al., 2016, Qi et al., 2017, Wang et al., 2019, Thomas et al., 2019, Hu et al., 2020] and hybrid methods use multi-modal 2D and 3D information [Jaritz et al., 2019].

Point clouds of buildings are inherently huge and irregular with sparse and dense regions, which also vary based on sensors and acquisition methods [Lehtola et al., 2017]. As discussed in Section 1., safety-related assets in point clouds have *"find-the-needle-in-a-haystack"* and *"class-imbalanced data"* setbacks. Hence, for this study, we would need a network that manages these scenarios efficiently, learns features of small-sized assets, and directly operates on point clouds. Thus, narrowing our focus to point-based direct methods.

In this context, we use KP-FCNN, a point-based method that uses a robust kernel-point convolution (KPConv) method and directly processes huge point clouds. KP-FCNN processes point clouds on multiple levels using different kernel points for each level capturing fine-grained features, helping feature learning for small-sized assets [Thomas et al., 2019]. Further, it uses grid subsampling and radius neighborhoods strategies to manage varying point densities, reducing the computational cost without compromising feature learning [Thomas et al., 2019]. In addition, the random-picking sampling strategy of KP-FCNN gives equal importance to all semantic classes during training, hence tackling class-imbalanced data issues with deep learning.

Instance segmentation networks: They are mainly categorized into proposal-based [Yang et al., 2019] and proposal-free [Wang et al., 2018, Vu et al., 2022] methods. Proposal-based methods use 3D object detection techniques to generate proposals and then predict masks to separate instances, whereas proposal-free methods use semantic segmentation techniques to extract instances based on feature similarities of points [Guo et al., 2020]. In our case, the safety-related assets are clearly standalone objects and in distinct locations; hence there is no risk of instance confusion.

3. METHOD

We use KP-FCNN and 3D point cloud datasets to identify safety-related assets within a building through steps illustrated in Figure 2. First, we define semantic classes and prepare the data for

Figure 3: Screenshots of safety-related assets from the datasets in section 4. including, exit sign; fire switch; ceiling lights; ceiling ventilation ducts; fire extinguisher (Image Source: [Armeni et al., 2017, Guzov et al., 2021]. Here, areas within S3DIS have different assets, and further assets among the three datasets also have visual variations like wall-embedded and hand-useable fire extinguishers.

the DL network. Next, we choose parameters and strategies to perform semantic segmentation with KP-FCNN. Lastly, we post-process the results to refine them and build 3D-geometry models.

3.1 Data preparation

Buildings have various safety-related assets based on access, utilities, and fire suppression functionalities [NAPSG, 2020,Hossain. et al., 2021]. For this study, we chose semantic classes from commonly found assets in buildings, namely, ceiling light, fire switch, fire extinguisher, exit sign, ceiling ventilation duct, door, and window. Further, to provide complete scene semantics and build 3D models, we have ceiling, floor, wall, stairway, furniture, and clutter classes. Here, the clutter includes any object that does not belong to any other chosen semantic class. Therefore, in total, we have 13 semantic classes. Using CloudCompare [GPL software, 2022], we manually labeled the point clouds for the chosen semantic classes. Figure 3 shows some examples of the chosen safety-related assets.

For training and testing, the labeled datasets are divided based on visual representations of assets. For testing the network's generalization ability, we perform experiments with train-test sets having buildings from different areas with visually different-looking assets [Goodfellow et al., 2016, p. 108]. Further, we demonstrate domain adaptation, a transfer learning approach, where data from different domains are used to adapt network's learning for improved generalization [Goodfellow et al., 2016, p. 534]. We discuss this in detail in Section 5.

3.2 Semantic segmentation

We use KP-FCNN, a fully convolutional neural network (CNN) with connected and multi-layered encoder and decoder modules. The encoder has five layers with two convolutional blocks per layer consisting of a strided KPConv (a KPConv block used for pooling) as the first block, except the first layer. The other convolutional block consists of a KPConv, batch normalization, and leaky ReLu activation modules. The features between the intermediate layers of the encoder and decoder are passed using skip links. The decoder derives the point-wise features using the nearest upsampling method. The upsampled features are combined with those obtained from the skip links and are further processed using a unary convolution.

The open-source PyTorch implementation of KP-FCNN is used for this study with area-wise point clouds with geometry, color, and class fields as input [Thomas, 2020]. We chose five input features (D_{in}=5), having a constant feature encoding input's geometry, color (Red-Green-Blue), and Z geometry field. In our case, using the XY position of assets as a feature is not helpful, as assets like an exit sign or a fire switch could be anywhere in the building and have no fixed XY location. However, the Z values can provide a spatial context of the assets through information like height above ground [Anjanappa, 2022].

By design, KP-FCNN uses random or regular picking strategies to choose small spherical sub-clouds across the scenes to process the point clouds [Thomas et al., 2019]. For training, we use the random picking strategy to tackle the class imbalances in the used datasets. This strategy chooses the same number of spheres centered on each semantic class, ensuring the network sees minority classes (small-sized assets) more often, ensuring efficient feature learning. For testing, we pick the spheres regularly for spatial regularity so that each point is evaluated multiple times at different sphere locations.

Parameter	Description	Value
$\mathbf{D_{in}}$	Input feature dimension.	5
R	Spherical sub cloud radius (m)	0.7
$\mathbf{dl_0}$	First subsampling grid size (m)	0.01

Table 1: *Chosen KP-FCNN parameters for training.*

Further, to boost the sensitivity of the network for minority classes, we use class weights in training to reshape the cross-entropy loss [Johnson and Khoshgoftaar, 2019]. These weights are dynamically calculated using Equation 1 based on KP-FCNN's original work, where P is the total point proportions for all semantic classes per training data batch chosen.

$$Class\ Weights\ = \sqrt{\frac{100}{P}} \tag{1}$$

We use a batch size of 6, a learning rate of 0.001, and train the network for 500 epochs with a 500-optimizer step for all the experiments. To increase the data variability during training, we utilize the data augmentation strategies available in KP-FCNN, like scaling, rotation, flipping, noise addition, and color annealing. In particular, we use the color annealing probability of 0.8,

which erases color features of input clouds, occasionally using only geometry for feature learning.

3.3 Post-processing and 3D-geometry models

For post-processing, we adopt the detection workflow of PointNet [Qi et al., 2016]. For all the safety-related asset classes, we generate sub-clouds based on the predicted labels from KP-FCNN. Using the Connected Component Algorithm in CloudCompare, we generate asset instances with minimum points of 50. These instances are then merged with the indexes enabled to generate a cluster for each class with a unique segment number per instance as a scalar field. Therefore, all the points for a given instance in the cluster will have the same segment number. Further, the minimum number of points criteria removes random noisy predictions from the results.

From the processed point clouds, we separate wall, floor, and ceiling classes to reconstruct planar geometries as polyhedrons using Polyfit [Nan and Wonka, 2017], a polygonal surface reconstruction tool [Nikoohemat et al., 2021]. Then the identified safety-related assets are added to the obtained mesh to construct the final 3D-geometry model. For constructing models for an entire test area, we recommend generating per-room models first and then combining them to obtain an area-wise model.

3.4 Evaluation Metrics

Metrics to evaluate semantic segmentation on point clouds generally use per-point labels. For example, the widely used metric Intersection over Union (IoU) uses per-point labels to measure the overlap between the prediction and ground truth bounding boxes of objects [Liu et al., 2019, Guo et al., 2020]. However, per-point metrics derive less meaning for an asset management system or a prospect operator using the system. Instead, they would benefit from finding the asset as an object within the building, like

- Total assets as true positives (TP) + false negatives (FN).
- The presence or absence of an asset at a location.
- Reduced false asset locations as False Positives (FP).

Therefore, we utilize conventional metrics based on per-point labels to compute object-level metrics for each asset class to evaluate the performance of our method.

Figure 4: Example Door Instance (left to right): RGB, ground truth, and predictions from KP-FCNN with correctly identified points in brown and others in yellow.

For asset identification, We focus on finding whether or not an asset is correctly identified rather than the accuracy of its predicted shape, like the door example in Figure 4. Hence, the recall rate would be a suitable metric to categorize the instances into TP, FP, or FN. For all the safety-related asset classes, we calculate the recall rate per instance using Equation 2 using the clustered sub-clouds from the post-processing step. Using a threshold-based method on the calculated recall rates, we group the asset instances into TP, FP, or FN.

$$Instance\ Recall\ Rate = \frac{Correct\ Point\ Predictions}{Total\ Points} \quad (2)$$

For assets that require the highest reliability, like the fire switches and extinguishers, we use a threshold of recall$>$70% to categorize an asset as TP. For other classes, we use a threshold of recall$>$50%. Using the obtained TP, FP, and FN counts for all the asset classes, we calculate instance-level precision, recall, and F1 score per class.

4. DATA

We use colored point cloud datasets acquired from three different 3D sensors: depth camera, mobile laser scanner, and consumer-grade lidar.

1. **Stanford 3D Indoor Scene Dataset (S3DIS)**, a benchmark dataset with point clouds generated using images from a depth camera [Armeni et al., 2016, Armeni et al., 2017]. It covers six large-scale indoor areas from different educational and office buildings, namely Areas 1 to 6. Architecturally, Areas 1, 3, and 6 and Areas 2 and 4 are similar-looking. However, Area-5 is captured from a different building compared to the other areas.

2. **Human POSEitioning System (HPS) Dataset** contains six 3D indoor scans covering large working spaces, namely, Mpi-biblio-ug, Mpi-biblio-eg, Mpi-biblio-og, Mpi-Etage6, Mpi-kino, and Mpi-eg. Acquired with a mobile laser scanner, these point clouds are huge and dense with geometry, colors, surface normals, curvature, and camera-specific attributes [Guzov et al., 2021].

3. **iPhone Dataset** was acquired with an iPhone 12 Pro embedded with a lidar sensor [Apple, 2020]. We used Scaniverse [Toolbox AI, 2022] and Polycam [Polycam, 2022] mobile applications to scan different areas of a university building covering lecture rooms, hallways, and lobby.

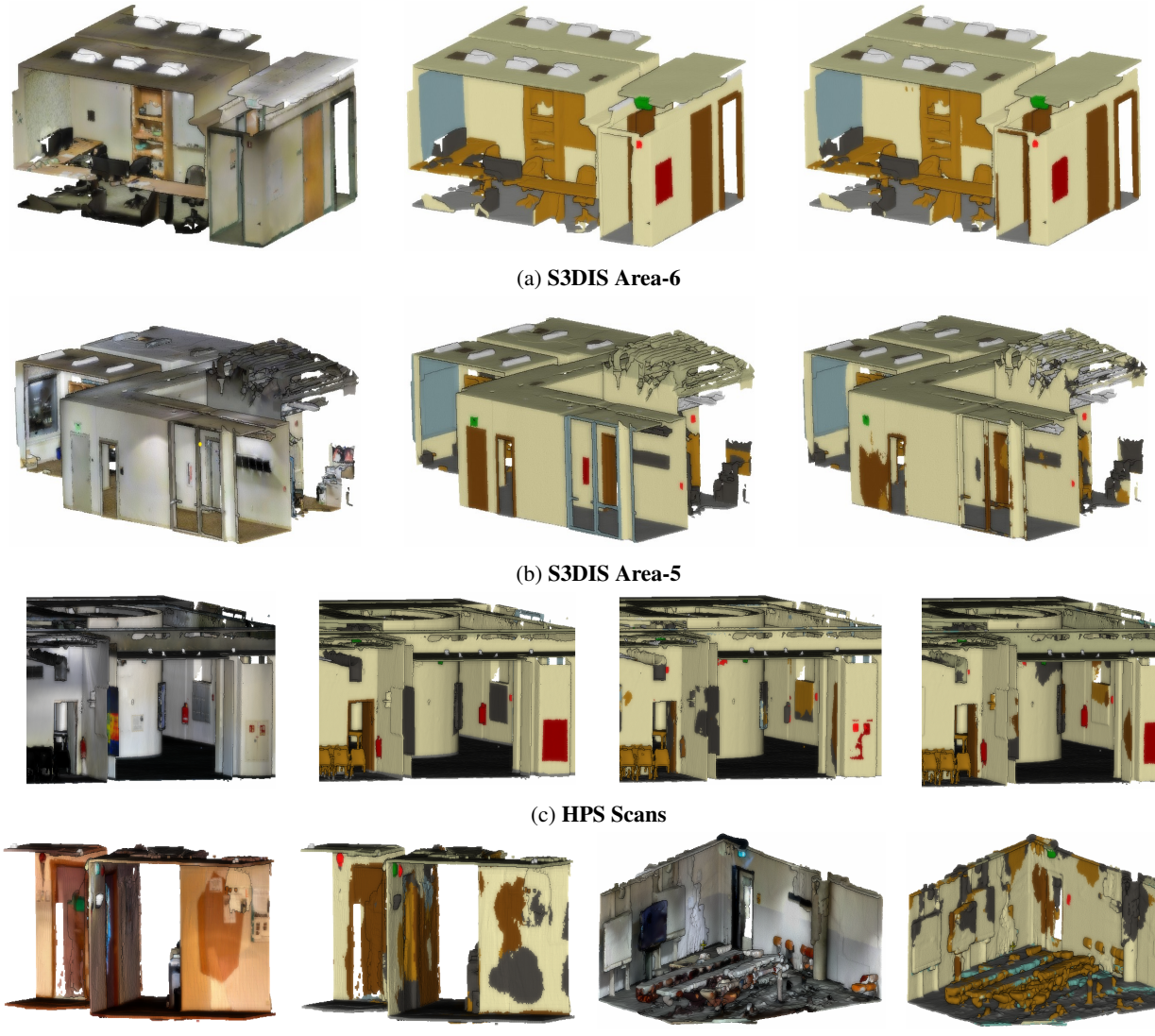
All the above datasets contain various safety-related assets like temperature controllers, smoke detectors, lights, exit signs, fire alarms, sprinklers, and extinguishers. Out of the 13 chosen semantic classes in section 3.1, the S3DIS dataset had labels for the permanent structures and some safety-related assets like doors and windows. However, the remaining asset classes were labeled manually. For HPS dataset, we manually labeled the point clouds for all 13 classes. Further, the iPhone dataset is not labeled, as we used it only for qualitative assessment.

5. EXPERIMENTS AND RESULTS

5.1 Case-study: S3DIS

Experiments: We performed two experiments with Area-6 and Area-5 as test areas, using the respective remaining areas for training. As the main goal was to determine the feasibility of identifying small assets, we first tested the method on Area-6, which had assets similar to training Areas 1 and 3 (see Figure 3). Though the train and test areas here share a few similarities among the assets, they are spatially non-overlapping. Next, we use one of the S3DIS's standard split with test Area-5 to evaluate model generalization. [Armeni et al., 2017].

Further, to estimate the smallest feasible point cloud resolution for the proposed network configurations discussed in Section 3.2, we repeated the first experiment with sub-sampled point clouds for both training and testing. We used the spatial sub-sample



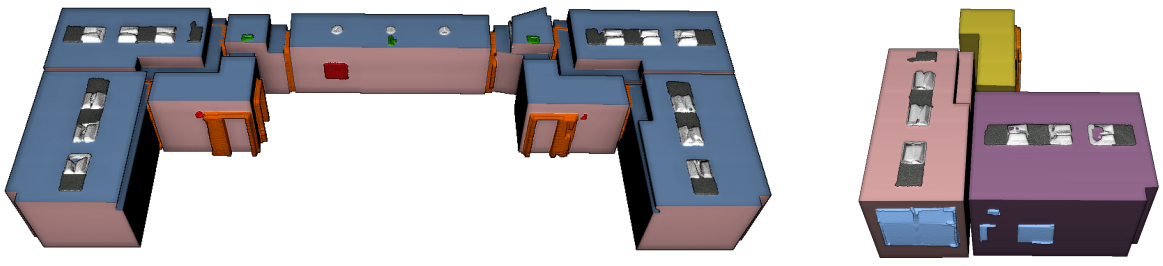
(a) **S3DIS Area-6**

(b) **S3DIS Area-5**

(c) **HPS Scans**

(d) **iPhone - Hallway and Room**

Figure 5: Point clouds with scene segmentation results for all three datasets. Ordering from left to right: (i) S3DIS (a) and (b) - RGB, ground truth, and KP-FCNN predictions; (ii) iPhone (c) - RGB and KP-FCNN predictions; (iii) HPS scans (d) - RGB, ground truth, and KP-FCNN predictions for model generalization and domain adaptation



(a) (b)

Figure 6: Polyhedron 3D-geometry model for selected part of S3DIS test Area-6. (a) Area-wise model showing structural semantics of selected rooms and hallways (wall in pink and ceiling in dark blue) with safety-related assets. (b) Room-wise model with multiple rooms in different colors containing safety-related assets. Safety-related assets in the models: lights (white), fire switches (bright red), a fire extinguisher (dark red), ventilation ducts (grey), exit signs (green), doors (orange), and windows (light blue).

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-1-W1-2023-461-2023 | © Author(s) 2023. CC BY 4.0 License.

Test Area-6							
Class	**GT**	**TP**	**FP**	**FN**	**Prec**	**Rec**	**F1**
Light	148	136	7	12	95.1	91.9	93.5
Fire Swi	10	10	2	0	83.3	100	91
Fire Ext	5	5	1	0	83.3	100	91
Ven Duct	116	112	0	4	100	96.6	98.2
Exit Sign	5	5	0	0	100	100	100
Door	48	48	4	0	92.3	100	96
Window	31	27	1	4	96.4	87.1	91.5

Model Generalization - Test Area-5							
Class	**GT**	**TP**	**FP**	**FN**	**Prec**	**Rec**	**F1**
Light	210	162	43	48	79	77.1	78.1
Fire Swi	32	31	8	1	79.5	96.9	87.3
Fire Ext	6	0	4	6	0	0	-
Ven Duct	124	117	3	7	97.5	94.4	95.9
Exit Sign	14	1	0	13	100	7.1	13.3
Door	69	60	5	9	92.3	87	89.6
Window	48	40	2	8	95.2	83.3	88.9

Table 2: Precision, Recall and F1-score of safety-related assets on test areas from S3DIS dataset (in $\%$).

Model Generalization - Mpi-biblio-og and Mpi-eg							
Class	**GT**	**TP**	**FP**	**FN**	**Prec**	**Rec**	**F1**
Light	217	140	78	77	64.2	64.5	64.4
Fire Swi	15	15	13	0	53.6	100	69.8
Fire Ext	10	1	1	9	50	10	16.7
Ven Duct	34	16	17	18	48.5	47.1	47.8
Exit Sign	19	13	13	6	50	68.4	57.8
Door	28	18	27	10	40	64.3	49.3
Window	8	6	23	2	20.7	75	32.4

Domain Adaptation - Mpi-biblio-og and Mpi-eg							
Class	**GT**	**TP**	**FP**	**FN**	**Prec**	**Rec**	**F1**
Light	217	177	71	40	71.4	81.6	76.1
Fire Swi	15	14	5	1	73.7	93.3	82.4
Fire Ext	10	10	0	0	100	100	100
Ven Duct	34	23	0	11	100	67.7	80.7
Exit Sign	19	15	3	4	83.3	79	81.1
Door	28	27	13	1	67.5	96.4	79.4
Window	8	5	0	3	100	62.5	76.9

Table 3: Precision, Recall, and F1-score of safety-related assets on test areas from HPS dataset (in $\%$).

method from CloudCompare to generate the sub-sampled point clouds at an interval of $0.01m$.

Results: The results for both experiments with S3DIS test areas are shown in Table 2. Figures 5a and 5b show the segmentation results and the reconstructed 3D polyhedron model for a part of test Area-6 in Figure 6.

For test Area-6, Table 2 shows that all the assets achieved a recall and precision rate greater than 80% as the network is familiar with the asset representations due to its train-test setup. However, test Area-5 had visually different fire extinguishers and exit signs compared to the areas used for training, as seen in Figure 3, making them unfamiliar to the network, which explains why they are not detected in the dataset, as seen in Table 2. Further, assets like lights are only identified partially in this experiment, among which most unidentified lights are round-shaped that are not prominently found in the training areas. The probability of picking such samples during training using the random picking method in KP-FCNN is unpredictable, affecting feature learning for such cases.

Further, we were able to identify all the chosen safety-related assets with sub-sampled point clouds with a point spacing of $0.01m$ and $0.02m$ [Anjanappa, 2022]. However, when sub-sampled beyond $0.02m$, the network failed to converge at neighborhood computation during training because of insufficient points available per input batch [Thomas et al., 2019].

5.2 Case-study: HPS

Experiments: We performed one experiment each for model generalization and domain adaptation using Mpi-biblio-og and Mpi-eg as test areas. First, the S3DIS-only trained network was used with test areas to evaluate the network's ability to identify assets when trained and tested on point clouds from different 3D sensors. Next, for domain adaptation, we re-trained the S3DIS-only network with the remaining HPS scans to examine the network's ability to adapt and improve on new buildings. In this experiment, the HPS scans used as train-test areas are spatially non-overlapping.

Results: Results for HPS experiments are shown in Table 3 and Figure 5c. Though S3DIS and HPS datasets have varying point densities and visually different assets (see Figure 3), for model generalization, small-sized assets like fire switches and exit signs are identified well. But from Table 3, these classes also have higher false positive counts, where the S3DIS-only trained network wrongly identified objects resembling safety-related assets in HPS dataset, like objects shown in Figure 7. In addition, entirely unfamiliar or different-looking assets, like cylindrical fire extinguishers, are not identified in this experiment. But, with domain adaptation, the S3DIS-only trained network learned and adapted to the features of assets in HPS dataset, resulting in improved predictions and reduced false positives and negatives, as shown in Table 3.

5.3 Case-study: iPhone

As a model generalization experiment, we used the S3DIS-only trained network with all the scans from iPhone data for this case study. Since the iPhone data is unlabeled, evaluation metrics are not calculated and results are only evaluated qualitatively. Though the assets in S3DIS and iPhone look different (see Figure 3) and these datasets have different point densities, the results in Figure 5d show that assets like doors, fire switches, and exit signs are identified successfully by the network.

6. DISCUSSION

Despite our limited training datasets, when tested on new buildings, the network efficiently identified assets with slight visual variations, like fire switches in S3DIS Area-5 and HPS scans. However, it failed to identify entirely different-looking assets, in particular the fire extinguishers. But, through domain adaptation using HPS scans, we demonstrated that the network effectively learns and adapts to the new fire extinguishers achieving improved recall rates.

Though we were able to identify the safety-related assets, relying entirely on deep learning in the context of safety can be debatable as the behavior of networks used can be unpredictable. In addition, indoor scenes from buildings may contain various objects that may have visual resemblances to some of the safety-related assets like in Figure 7. As deep learning networks learn features using geometry and color, they could falsely identify such objects

as safety-related assets, which is not ideal if the results are used during emergencies.

Figure 7: Examples of objects resembling safety-related assets (exit sign and fire switch) in the HPS dataset, for visualization purposes. The first column shows the actual asset, while the second and third column shows objects resembling them.

Detecting assets with a pathological placement is an expected limitation of the proposed method, e.g., detecting fire extinguishers placed inside casings that are not clearly visible. Further, every building can have different-looking safety-related assets resulting in a high intra-class variability. For example, the lights can be of different shapes or be present on the ceiling or wall. The fire extinguishers can be in the form of rolled pipes or cylindrical and wall-mounted or placed on the floor. Such asset variations affect the performance of the network and, thus, affect overall asset identification rates, as seen in the model generalization experiments.

One solution for large-scale buildings with new-looking assets could be that the user can re-train the network with a part of the dataset and use the resulting network on the remaining dataset to identify the safety-related assets accurately. This is feasible with commercial-off-the-shelf low-cost scanning techniques, as we demonstrated by acquiring data of such a use case in the domain adaptation experiment.

7. CONCLUSION

Detection of safety-related assets could make a highly beneficial by-product whenever point clouds for buildings are captured. If done regularly, this process can facilitate maintaining up-to-date records of the assets or detect changes over time. Therefore, we focused on studying the feasibility of using deep learning methods to automate asset identification. In this regard, KP-FCNN was our choice of DL network due to its ability to handle data-level setbacks of safety-related assets and its robustness to variations in the point cloud density, which is common in indoor point clouds generated with mobile scanning techniques.

The safety assets are small and need to be detected in large point clouds, making the problem essentially a *needle-in-the-haystack* problem. The obtained results in Tables 2 and 3 show that the designed method is feasible to identify small-sized assets like fire switches and exit signs in all the experiments. Also, the proposed workflow proved rather robust and invariant with respect to varying point cloud quality for the data from different 3D sensors.

Even though the proposed method could detect certain assets with 100% recall rates in some cases, like with S3DIS Test Area 6, these methods can not solely be reliable for autonomous operation due to the false positives seen in model generalization experiments (Tables 2 and 3). In other words, the method's precision drops significantly when used with data from another environment or a new building. Regardless of this limitation, the proposed method could serve as partial automation, which could be used as an assistive tool for human operators to check, verify, and correct (if necessary) the identified assets. This would lessen the burden for the operators to navigate through the buildings manually or sift through the complete data of buildings, both images or point clouds.

REFERENCES

Anjanappa, G., 2022. Deep learning on 3d point clouds for safety-related asset management in buildings. Master's thesis, University of Twente.

Apple, 2020. iphone 12. `https://www.apple.com/iphone-12/key-features/`. Accessed: 2023-03-04.

Armeni, I., Sax, S., Zamir, A. R. and Savarese, S., 2017. Joint 2d-3d-semantic data for indoor scene understanding. arXiv preprint arXiv:1702.01105.

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M. and Savarese, S., 2016. 3d semantic parsing of large-scale indoor spaces. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1534–1543.

Bello, S. A., Yu, S., Wang, C., Adam, J. M. and Li, J., 2020. Review: Deep learning on 3d point clouds. Remote Sensing.

Chen, C., 2019. Ogc indoor mapping and navigation pilot engineering report. Technical report, Open Geospatial Consortium.

Goodfellow, I., Bengio, Y. and Courville, A., 2016. Deep Learning. MIT Press. `http://www.deeplearningbook.org`.

GPL software, 2022. Cloudcompare (version: 2.12.4). `http://www.cloudcompare.org/`. Accessed: 2022-07-14.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L. and Bennamoun, M., 2020. Deep learning for 3d point clouds: A survey. IEEE transactions on pattern analysis and machine intelligence 43(12), pp. 4338–4364.

Guzov, V., Mir, A., Sattler, T. and Pons-Moll, G., 2021. Human poseitioning system (hps): 3d human pose estimation and self-localization in large scenes from body-mounted sensors. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4316–4327.

Hossain., M., Ma., T., Watson., T., Simmers., B., Khan., J. A., Jacobs., E. and Wang., L., 2021. Building indoor point cloud datasets with object annotation for public safety. In: Proceedings of the 10th International Conference on Smart Cities and Green ICT Systems - SMARTGREENS,, INSTICC, SciTePress, pp. 45–56.

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N. and Markham, A., 2020. Randla-net: Efficient semantic segmentation of large-scale point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11108–11117.

Jaritz, M., Gu, J. and Su, H., 2019. Multi-view pointnet for 3d scene understanding. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3995–4003.

Johnson, J. M. and Khoshgoftaar, T. M., 2019. Survey on deep learning with class imbalance. Journal of Big Data 6, pp. 27.

Kostoeva, R., Upadhyay, R., Sapar, Y. and Zakhor, A., 2019. Indoor 3d interactive asset detection using a smartphone. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-2/W13, pp. 811–817.

Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J. and Beijbom, O., 2019. Pointpillars: Fast encoders for object detection from point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12697–12705.

Lehtola, V. V., Kaartinen, H., Nüchter, A., Kaijaluoto, R., Kukko, A., Litkey, P., Honkavaara, E., Rosnell, T., Vaaja, M. T., Virtanen, J.-P. et al., 2017. Comparison of the selected state-of-the-art 3d indoor scanning and point cloud generation methods. Remote sensing 9(8), pp. 796.

Liu, W., Sun, J., Li, W., Hu, T. and Wang, P., 2019. Deep learning on point clouds and its application: A survey. Sensors.

Nan, L. and Wonka, P., 2017. Polyfit: Polygonal surface reconstruction from point clouds. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2353–2361.

NAPSG, 2020. Best practices: Guide to indoor mapping, tracking, and navigation.

Nikoohemat, S., Godoy, P., Valkhoff, N., Wouters-van Leeuwen, M., Voûte, R. and Lehtola, V., 2021. Point cloud based 3d models for agent based simulations in social distancing and evacuation. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences 4, pp. 113–120.

Polycam, 2022. Polycam: 3d scanner and editor. `https://poly.cam/`. Accessed: 2023-03-26.

Qi, C. R., Su, H., Mo, K. and Guibas, L. J., 2016. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. arXiv:1612.00593 [cs]. arXiv: 1612.00593.

Qi, C. R., Yi, L., Su, H. and Guibas, L. J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan and R. Garnett (eds), Advances in Neural Information Processing Systems, Vol. 30, Curran Associates, Inc.

Shi, S., Wang, X. and Li, H., 2019. Pointrcnn: 3d object proposal generation and detection from point cloud. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–779.

Tchapmi, L., Choy, C., Armeni, I., Gwak, J. and Savarese, S., 2017. Segcloud: Semantic segmentation of 3d point clouds. In: 2017 International Conference on 3D Vision (3DV), pp. 537–547.

Thomas, H., 2020. Kernel point convolution implemented in pytorch. `https://github.com/HuguesTHOMAS/KPConv-PyTorch`. Accessed: 2022-05-25.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F. and Guibas, L. J., 2019. Kpconv: Flexible and deformable convolution for point clouds. Proceedings of the IEEE International Conference on Computer Vision.

Toolbox AI, I., 2022. Scaniverse. `https://scaniverse.com/`. Accessed: 2023-03-26.

Vu, T., Kim, K., Luu, T. M., Nguyen, X. T. and Yoo, C. D., 2022. SoftGroup for 3D Instance Segmentation on Point Clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2708–2717.

Wang, W., Yu, R., Huang, Q. and Neumann, U., 2018. Sgpn: Similarity group proposal network for 3d point cloud instance segmentation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2569–2578.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M. and Solomon, J. M., 2019. Dynamic graph cnn for learning on point clouds. ACM Trans. Graph.

Warsop, T. and Singh, S., 2010. A survey of object recognition methods for automatic asset detection in high-definition video. In: 2010 IEEE 9th International Conference on Cyberntic Intelligent Systems, pp. 1–6.

Yang, B., Wang, J., Clark, R., Hu, Q., Wang, S., Markham, A. and Trigoni, N., 2019. Learning object bounding boxes for 3d instance segmentation on point clouds. In: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox and R. Garnett (eds), Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc.

Yi, L., Kim, V. G., Ceylan, D., Shen, I.-C., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A. and Guibas, L., 2016. A scalable active framework for region annotation in 3d shape collections. ACM Trans. Graph.