

## Zero-Shot Detection of Buildings in Mobile LiDAR using Language Vision Model

June Moh Goo<sup>1</sup>, Zichao Zeng<sup>1</sup>, Jan Boehm<sup>1</sup> \*

<sup>1</sup> Department of Civil, Environmental and Geomatic Engineering, University College London, Gower Street, London, WC1E 6BT UK  
– {june.goo.21, zichao.zeng.21, j.boehm}@ucl.ac.uk

**KEY WORDS:** LiDAR, Point Cloud, Building, Scene Understanding, Deep Learning, Foundation Model, Multi-Modal Model

### ABSTRACT:

Recent advances have demonstrated that Language Vision Models (LVMs) surpass the existing State-of-the-Art (SOTA) in two-dimensional (2D) computer vision tasks, motivating attempts to apply LVMs to three-dimensional (3D) data. While LVMs are efficient and effective in addressing various downstream 2D vision tasks without training, they face significant challenges when it comes to point clouds, a representative format for representing 3D data. It is more difficult to extract features from 3D data and there are challenges due to large data sizes and the cost of the collection and labelling, resulting in a notably limited availability of datasets. Moreover, constructing LVMs for point clouds is even more challenging due to the requirements for large amounts of data and training time. To address these issues, our research aims to 1) apply the Grounded SAM through Spherical Projection to transfer 3D to 2D, and 2) experiment with synthetic data to evaluate its effectiveness in bridging the gap between synthetic and real-world data domains. Our approach exhibited high performance with an accuracy of 0.96, an IoU of 0.85, precision of 0.92, recall of 0.91, and an F1 score of 0.92, confirming its potential. However, challenges such as occlusion problems and pixel-level overlaps of multi-label points during spherical image generation remain to be addressed in future studies.

### 1. INTRODUCTION

With the advancement of 3D acquisition technologies such as LiDAR and other 3D sensors, along with reductions in price, the accessibility of point cloud data has become more affordable compared to the past. This has led to the emergence of multiple deep-learning techniques that can extract meaningful information from point clouds. (Qi et al., 2017a,b; Thomas et al., 2019; Guo et al., 2021). In the field of remote sensing and photogrammetry, building detection is one of the most fundamental problems for urban scenes as it forms the basis of 3D urban city model construction. Despite the remarkable progress in the field of deep learning, there are still significant obstacles to overcome when it comes to training deep learning models from scratch. One of the biggest challenges is the amount of data required for training, which can be particularly challenging in the case of point cloud datasets. Moreover, even when the dataset is available, training models on such vast amounts of data requires considerable computing power. In particular, this issue is a significant difficulty for the automation of 3D building detection, where data collection, processing, and time-consuming training are significant obstacles.

In order to address these limitations, several previous studies have proposed techniques such as Sim-to-Real domain transferable learning (Griffiths and Boehm, 2019b; Xiao et al., 2022; Jin et al., 2022; Zhang et al., 2022c). This approach involves training models on synthetic datasets and effectively applying them to real-world datasets. Collecting and annotating real-world data is laborious and time-consuming, and training on synthetic dataset approach helps to avoid that. Moreover, synthesising data allows the generation of extensive data under various conditions such as different lighting and urban layouts. However, due to domain discrepancy, the models trained on

synthetic datasets might struggle to generalise to real-world data due to differences in noise, density, and distribution of point clouds. Therefore, when training on synthetic datasets, caution must be taken to avoid overfitting. It is crucial to build models that possess robustness to accommodate the variations and imperfections in real-world data.

Alternatively, zero-shot transfer can be utilised for various computer vision tasks. Thanks to the advancement of Language Vision Models (LVMs), solving these tasks without the need for explicit training is now possible. LVMs, pre-trained on extensive datasets, consistently demonstrate high performance across different tasks. For instance, CLIP (Radford et al., 2021), a prominent LVM, simultaneously trains an image encoder and a text encoder using an image-text paired dataset. Since the inception of the CLIP model, newer, more powerful models have emerged, such as OpenCLIP (Cherti et al., 2022), ALIGN (Jia et al., 2021), and Flamingo (Alayrac et al., 2022). Subsequently, LVM models have enhanced their robustness and adaptability through knowledge distillation for specific computer vision tasks like object detection (Liu et al., 2023; Du et al., 2022; Lin et al., 2022) and semantic segmentation (Zhou et al., 2023; Liang et al., 2023).

While Language Vision Models (LVMs) have demonstrated impressive performance in visual understanding without fine-tuning and with an open vocabulary, their reliance on extensive training data and computational resources remains a challenge, particularly in the context of 3D point clouds. To address these challenges, researchers have explored techniques such as leveraging pre-trained LVMs for point cloud understanding using multi-view approaches (Takmaz et al., 2023; Peng et al., 2023), knowledge distillation (Zhang et al., 2022b; Zhu et al., 2023), or projections (Zhang et al., 2022b) instead of training LVMs directly on point cloud data from scratch.

\* Corresponding author

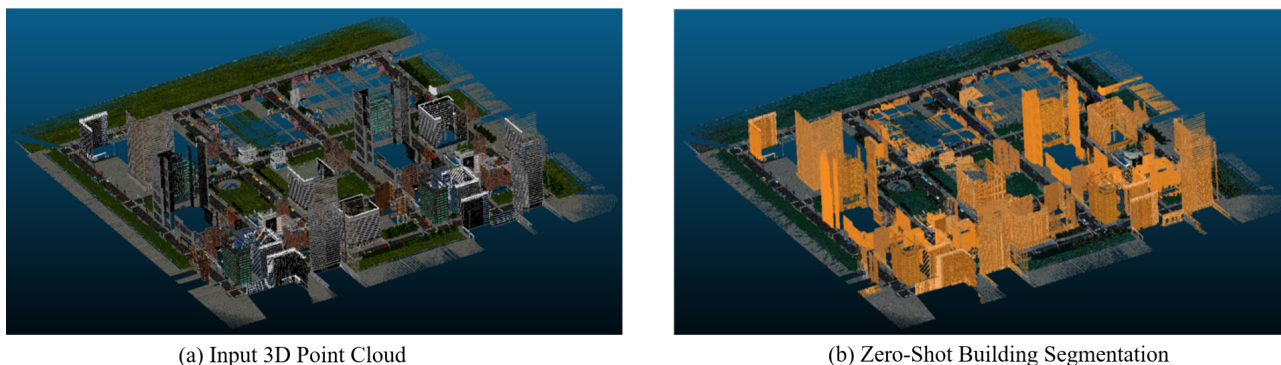


Figure 1. This figure illustrates the zero-shot building segmentation of the "SynthCity" dataset. (a) displays the original full point clouds used as input, while (b) shows the building detection results highlighted in yellow.

In this study, we aim to achieve two main goals. Firstly, we seek to perform 3D point cloud segmentation by adapting an LVM initially designed for 2D computer vision. This adaptation is facilitated through a 3D-to-2D projection method, eliminating the need for pre-training or fine-tuning. Secondly, we analyse the usefulness of this approach by conducting experiments on synthetic data to assess its ability to bridge the domain gap between synthetic and real-world datasets.

## 2. BACKGROUNDS AND PREVIOUS WORKS

### 2.1 Language Vision Models for Point Clouds

Thanks to the high impact of LVMs on the field of computer vision, many research efforts have been made to incorporate LVMs into 3D data processing. However, due to the significant amount of training time and data required to construct LVMs, most progress has been made only in 2D data. These drawbacks are more emphasised when it comes to 3D data, which are much larger in size and require longer training times. Therefore, many works have started investigating ways to utilise existing 2D LVMs for 3D data understanding rather than building LVMs specifically for 3D data, such as point clouds, meshes and voxels.

One method to enable 3D recognition with CLIP-based models is by projecting point clouds onto 2D images. PointCLIP (Zhang et al., 2022b) enables cross-modality zero-shot recognition on point clouds without prior 3D training by leveraging multi-view simple projection to transfer pre-trained 2D knowledge from CLIP to the 3D domain. Under a lightweight interview adapter under few-shot settings, PointCLIP enhances classification performance. However, it still exhibits low performance in zero-shot transfer scenarios. Nevertheless, PointCLIP-V2 (Zhu et al., 2023) addresses these shortcomings by replacing the multi-view simple projection with a realistic projection and employing prompt engineering with GPT-3 (Brown et al., 2020), enhancing performance not limited to classification but also part segmentation and object detection.

An alternative method involves aligning features from point cloud encoders with CLIP representations. For instance, ULIP (Xue et al., 2023) introduces a method that trains triplets consisting point clouds, images, and texts using a limited set of synthesised triplets to align with CLIP image-text space. Liu et al. (2024) enhance 3D representations by aggregating multiple 3D datasets and refine noisy text descriptions through the utilisation of a powerful large language model, GPT-4 (Achiam et al., 2023).

Success in point cloud classification leads to more complex tasks such as object detection and segmentation. OV-3DET (Lu et al., 2023) introduces a novel de-biased triplet cross-modal contrastive learning to connect image, point-cloud, and text modalities for improved performance with LVMs. OpenMask3D (Takmaz et al., 2023) and OpenScene (Peng et al., 2023) models enable Open-Vocabulary 3D scene understanding utilising the pre-trained Image-Text Embedding model, CLIP. These models, based on multi-view scene understanding, are more complex and error-prone as they require precise camera calibration in data integration. Moreover, they are limited to indoor scenes, raising uncertainty about their robustness for outdoor scenes.

### 2.2 Building Segmentation from Point Clouds

Building segmentation tasks in recent studies are often separated based on the method used to collect point clouds. Aerial point clouds are used for building detection and segmentation, while point clouds collected from terrestrial LiDAR or Mobile Laser Scanning (MLS) are more likely to be used for building part segmentation of the facade. Building detection from point clouds is an area that has received limited research attention. Even when studied, most models are trained on aerial point clouds rather than using MLS.

Several previous studies have used classical geometric approaches or non-deep learning methods for building detection. For example, Hrutka et al. (2022) proposed a method that uses RANSAC and DBSCAN algorithms to detect building footprints. Additionally, Mahphood and Arefi (2023) introduced a new approach that involves removing trees and vegetation and analysing the density within a cube in both 2D and 3D dimensions to identify buildings from the LiDAR point cloud.

Nowadays, deep learning models have enabled the extraction of features and solving specific tasks in point cloud understanding. Zhang et al. (2022a) proposed a methodology for building extraction through the fusion of LiDAR and photogrammetric point clouds, applying U-Net deep learning model segmentation.

A recent study, published by Gamal et al. (2021), utilised PointNet (Qi et al., 2017a) and Dynamic Graph Convolutional Neural Network (DGCNN) (Wang et al., 2019) to detect buildings in cities of Indonesia. They collected the necessary LiDAR data using unmanned aerial vehicles, resulting in a promising approach to building detection. The Damage-Sensitive Network (DS-Net)(Xiu et al., 2023) is introduced as a specialised method for identifying collapsed buildings with the Laplacian Unit (LU).

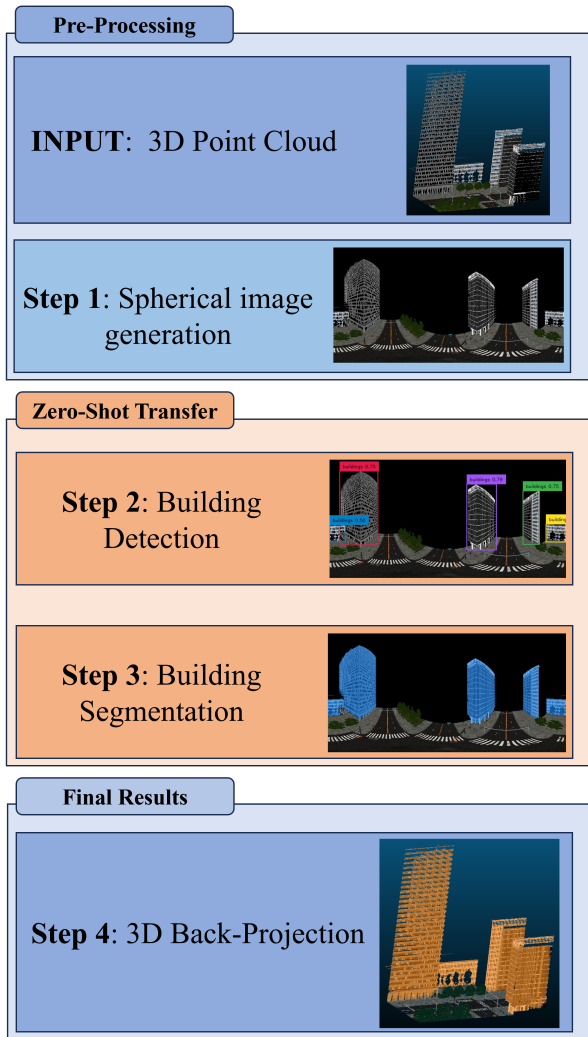


Figure 2. The diagram illustrates the steps of the methodological workflow. The figures on the right side are the examples of each step.

### 3. METHODOLOGY

In our work we use 2D rendered views of the 3D point clouds to perform scene understanding (as reviewed in Griffiths and Boehm (2019a)). This projection and back-projection process follows work by Sanchez Castillo et al. (2021); Karara et al. (2021); Tabkha et al. (2019). Building on the 2D projection, we approach our task of building segmentation using the combination of two 2D foundation models: Segment-Anything Model (SAM) (Kirillov et al., 2023) and Grounding DINO (Liu et al., 2023). Combining these two models was introduced by IDEA research as Grounded Segment-Anything (or Grounded SAM)(Ren et al., 2024).

Our indirect building segmentation method consists of 4 steps as shown in Figure 2. Each step will be introduced in the following sections.

#### 3.1 Spherical Image Generation

In order to generate a spherical image from the point clouds, we first normalise the point  $P(x, y, z)$  by the reference point  $O(x_0, y_0, z_0)$  as the following equation:

$$x' = x - x_0, \quad y' = y - y_0, \quad z' = z - z_0 \quad (1)$$

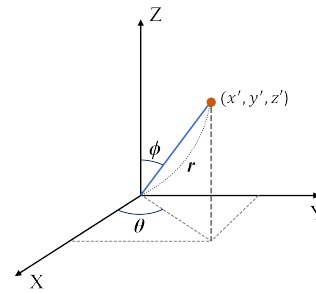


Figure 3. Principles of Spherical Coordinate Generation

We set the reference coordinate in the middle of the road or at an intersection.

In the next step, we compute the spherical coordinates  $(\theta, \phi)$  for all normalised point clouds. The Figure 3 shows the visual explanation of spherical coordinates, where  $r$  is the radius from the origin to the point,  $\theta$  is the azimuthal angle, and  $\phi$  is the polar angle. Each coordinate is derived by the following equations:

$$r = \sqrt{x'^2 + y'^2 + z'^2} \quad (2)$$

$$\theta = \arctan2(y', x') \quad (3)$$

$$\phi = \arccos(z'/r) \quad (4)$$

where  $x', y', z'$  are the normalised coordinates from equation 1

Lastly, we project these spherical coordinates to the 2D plane (image), which is also called an Equirectangular Projection. Each  $x$  and  $y$  coordinate in the image is determined by the following equations:

$$x = \frac{\theta + \pi}{2\pi} \times W \quad (5)$$

$$y = \frac{\phi}{\pi} \times H \quad (6)$$

where  $W$  and  $H$  represent the width and height of the image to project respectively.

Step 1 in Figure 2 shows the result of the spherical image generation procedure of the input 3D point clouds. This figure represents a spherical image generated from a reference point located at the centre of an intersection on the road.

#### 3.2 Building Bounding-Box Detection

For building detection, we use an open-set object detection model, Grounding DINO. Grounding DINO is a model that merges the image feature encoder DINO with grounded pre-training to detect a wide range of objects using language inputs (Liu et al., 2023). This model integrates language for open-set detection, dividing the detection process into feature enhancer, language-guided query selection, and a cross-modality decoder (Liu et al., 2023). We use the model for our task, by using the text prompt "buildings" to obtain bounding boxes of the buildings in the generated image. Step 2 of Figure 2 shows an example of the results of building detection using Grounding DINO.

#### 3.3 Building Segmentation

We proceed to use SAM for the segmentation step. SAM is a foundation model for image segmentation, using input prompts such as points or masks (Kirillov et al., 2023). It also explores

zero-shot segmentation from free-form text (Kirillov et al., 2023). However, we specifically focus on using the segmentation capabilities of SAM with bounding box inputs. For each bounding box detected in the previous step, we run SAM to segment all the pixels that belong to the building category. In Step 3 of Figure 2, blue-coloured areas represent the segmented pixels corresponding to the buildings.

### 3.4 Back-Projection

After segmenting the building image, we perform a back-projection of the segmented image onto the 3D point cloud. To achieve this, we save the mapping details during the generation of the spherical image. Later, we load these mapping details and replace the RGB data of the initial point clouds with the points that correspond to the segmented pixels.

## 4. EXPERIMENTS

We use an NVIDIA GeForce RTX 3070 with 8 GB for running Grounding DINO and SAM.

### 4.1 Performance Metrics

In order to measure model performance, we evaluate a range of commonly used metrics including Accuracy, Precision, Recall, F-1 Score, and Intersection over Union (IoU). IoU is typically used specifically for segmentation tasks. For detailed formulas refer to e.g. Goodfellow et al. (2016); Terven et al. (2023)

### 4.2 Synthcity Dataset

SynthCity (Griffiths and Boehm, 2019b) is an open-source dataset that represents an entire city in the form of a large-scale synthetic point cloud. The dataset includes 9 sub-areas and 9 label categories. SynthCity is specifically designed for pre-training deep learning models, enabling generalisation and expansion of their usage to real-world data. The dataset comprises 367 million completely labelled points with RGB features, making it a valuable resource for researchers and practitioners in this field. All points are labelled with one of 9 categories: Building, Car, Natural Ground, Ground, Pole Like, Street Furniture, Tree, and Pavement.

To avoid reaching computational resource limits, each sub-area in SynthCity is further divided into 4 sub-sub areas. Our dataset excludes 2 sub-sub areas of Area 7 and 1 sub-sub area of Area 9 as they contain neither roads nor buildings. As mentioned in Section 3 earlier, we arbitrarily selected centre points to generate spherical images in each sub-sub area. When making these selections, we imposed a condition that the centre point must first align with the roads. Additionally, if the area includes intersections, we designated the centre point to coincide with the centre of the intersection.

## 5. EXPERIMENTAL RESULTS

We present the comprehensive calculations of the error statistics for all sub-sub areas from Area 1 to 9.

| Area         | Accuracy    | IoU         | Precision   | Recall      | F-1         |
|--------------|-------------|-------------|-------------|-------------|-------------|
| Area 1       | 0.86        | 0.67        | 0.72        | 0.92        | 0.8         |
| Area 2       | 0.98        | 0.90        | 0.97        | 0.93        | 0.95        |
| Area 3       | 0.97        | 0.91        | 0.98        | 0.93        | 0.96        |
| Area 4       | 0.93        | 0.79        | 0.98        | 0.8         | 0.88        |
| Area 5       | 0.97        | 0.88        | 0.95        | 0.92        | 0.94        |
| Area 6       | 0.98        | 0.94        | 0.98        | 0.96        | 0.97        |
| Area 7       | 0.97        | 0.76        | 0.97        | 0.78        | 0.86        |
| Area 8       | 0.98        | 0.86        | 0.96        | 0.89        | 0.92        |
| Area 9       | 0.99        | 0.96        | 1.00        | 0.96        | 0.98        |
| <b>Total</b> | <b>0.96</b> | <b>0.85</b> | <b>0.92</b> | <b>0.91</b> | <b>0.92</b> |

Table 1. Quantitative Analysis of 9 Sub-Areas

|              | Car   | Natural Ground | Ground | Road  | Street Furniture | Tree  | Pavement |
|--------------|-------|----------------|--------|-------|------------------|-------|----------|
| Area 1       | 0.024 | 0.011          | 0.012  | 0.846 | 0.016            | 0.009 | 0.076    |
| Area 2       | 0     | 0.108          | 0.586  | 0     | 0.015            | 0.277 | 0.01     |
| Area 3       | 0     | 0.146          | 0.509  | 0     | 0.033            | 0.298 | 0.014    |
| Area 4       | 0     | 0.037          | 0.577  | 0     | 0.051            | 0.065 | 0.268    |
| Area 5       | 0     | 0.133          | 0.287  | 0     | 0.011            | 0.397 | 0.192    |
| Area 6       | 0     | 0              | 0.362  | 0.012 | 0.049            | 0.126 | 0.441    |
| Area 7       | 0     | 0              | 0.445  | 0     | 0                | 0.528 | 0.027    |
| Area 8       | 0     | 0.032          | 0.853  | 0     | 0.003            | 0.098 | 0.014    |
| Area 9       | 0     | 0              | 1      | 0     | 0                | 0     | 0        |
| <b>Total</b> | 0.018 | 0.031          | 0.137  | 0.617 | 0.019            | 0.076 | 0.098    |

Table 2. False Positive ratio for each category

### 5.1 Quantitative Evaluation

For the proper assessment of our approach, we compute five metrics. Table 1 displays an overall Accuracy of 0.96, IoU of 0.85, Precision of 0.92, Recall of 0.91, and F1-Score of 0.92 across all areas. Since our method has not been benchmarked against other approaches, we cannot claim it to be the superior choice. Nevertheless, given its highly encouraging outcomes, we wish to emphasise the versatility and resilience of LVMs when applied to 3D synthetic datasets using spherical images.

### 5.2 Qualitative Evaluation

The left image in Figure 1 illustrates the complete raw SynthCity dataset, while the right image represents the entire building prediction of the dataset. Our method shows remarkable performance not only in terms of quantitative analysis but also in qualitative analysis, as seen in the full dataset view.

The Figure 4 displays a detailed analysis of the different types of buildings. (a) and (b) highlight the detection of high and low-rise modern buildings, respectively. The results obtained demonstrate the high accuracy of our method in detecting buildings with different heights. Furthermore, (c) illustrates that our method can detect not only modern-style buildings but also old European-style ones, indicating the robustness and versatility of our approach.

### 5.3 False Positive Analysis

We have conducted a False Positive analysis of our method and identified a pattern of misclassified points. Analysis from Table 2 revealed that flat surfaces such as roads, grounds, and pavements often get incorrectly classified as buildings because some points classified as buildings pass through during the back-projection process. This misclassification is not limited to flat surfaces. For instance, tall and large-scale objects, such as trees,

| Building Types       | Raw Data | Prediction |
|----------------------|----------|------------|
| (a) Modern Low-Rise  |          |            |
| (b) Modern High-Rise |          |            |
| (c) Old Low-Rise     |          |            |

Figure 4. The figure shows the results of the segmentation of buildings using zero-shot prediction across different types of buildings. The results indicate that this method is highly effective in achieving accurate building segmentation regardless of building type

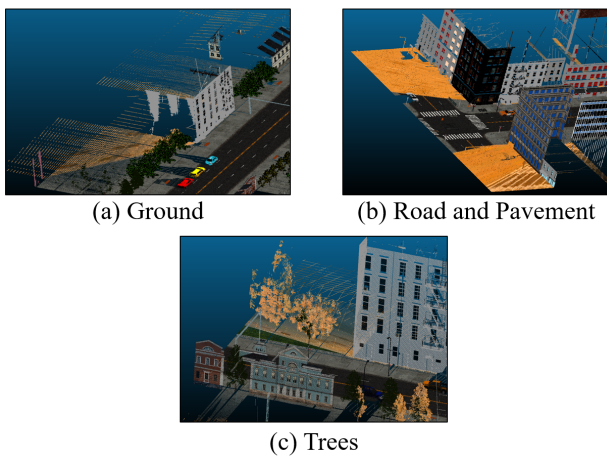


Figure 5. The figure shows in orange the false positive building points for different object categories. For clarity true positives are not highlighted.

can also be labelled as buildings. Because trees typically stand close together, some points predicted as buildings might be projected close to points belonging to the tree category.

From the perspective of visualization, as shown in the Figure 5, false positives occur when we back-project from the mapping of the spherical image. In (a), the segmented area is incorrectly back-projected to the ground behind the buildings, while in (b) and (c), it is incorrectly back-projected to roads, pavements, and trees respectively. They all have a common phenomenon:

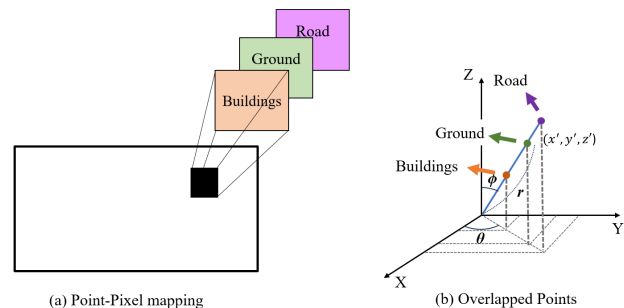


Figure 6. The figure displays the process of mapping that leads to the occurrence of false positives.

all false positives are generated behind the buildings. These results are generated because, for each pixel of the spherical image, our method saves more than one point, each with different labels. As in Figure 6 (a), while we generate Point-Pixel mapping, multiple points with labels could be saved in one pixel. This happens due to several reasons: First, when we generate spherical coordinates, the points with the same  $\theta$  (azimuthal angle) and  $\phi$  (polar angle) but different  $r$  (distances from the origin) will be mapped to the same pixel. This is because the equirectangular projection only considers the angular components of the spherical coordinates and not the radial distance as in the Figure 6 (b). Secondly, the SAM model segments buildings, however, some contour pixels are mistakenly classified as building labels. The pixels that were incorrectly segmented will be projected as buildings, including any points that were within those pixels.

## 6. CONCLUSION

In our study, we experimented with a training-free zero-shot transfer approach by adopting spherical image generation using the 2D-3D projection method for building segmentation tasks, applying a LVM. We integrated the powerful Open-vocabulary segmentation model, Grounded SAM, which combines the Open-vocabulary object detection model called Grounding DINO with the segmentation-enhanced Segment-Anything Model (SAM). Instead of the commonly used Multi-View projection, we utilised a 360-degree spherical projection, which alleviates the complexities of considering various orientations associated with multi-view projection, offering a relatively simple and intuitive approach. Despite the shape distortion challenges of spherical projection, Grounded SAM overcame building shape distortions and confirmed the effectiveness of LVM in accurately detecting buildings. Qualitative analysis revealed the successful detection of buildings regardless of their construction era or height. Quantitatively, the model exhibited excellent performance across all metrics, showing the robustness and versatility of LVM. However, a limitation of spherical image generation is the potential for false positives behind buildings due to the ability to store multiple labels for points in a single pixel. Nevertheless, despite these limitations, the use of LVM for zero-shot transfer allowed computational freedom and demonstrated effective operation even on synthesised datasets, confirming its domain transferability. These promising results could potentially address the lack of point cloud training datasets and compute-intensive training.

## 7. ACKNOWLEDGMENT

J. M. Goo and Z. Zeng are supported by the Engineering and Physical Sciences Research Council through an industrial CASE studentship with Ordnance Survey (Grant number EP/X524840/1).

## References

- Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altschmidt, J., Altman, S., Anadkat, S. et al., 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samingo, S., Monteiro, M., Menick, J., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., Simonyan, K., 2022. Flamingo: a Visual Language Model for Few-Shot Learning. <http://arxiv.org/abs/2204.14198>. arXiv:2204.14198 [cs].
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D., 2020. Language models are few-shot learners.
- Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., Jitsev, J., 2022. Reproducible scaling laws for contrastive language-image learning. <http://arxiv.org/abs/2212.07143>. arXiv:2212.07143 [cs].
- Du, Y., Wei, F., Zhang, Z., Shi, M., Gao, Y., Li, G., 2022. Learning to Prompt for Open-Vocabulary Object Detection with Vision-Language Model. <http://arxiv.org/abs/2203.14940>. arXiv:2203.14940 [cs].
- Gamal, A., Gamal, A., Husodo, A. Y., Husodo, A. Y., Jati, G., Jati, G., Alhamidi, M. R., Alhamidi, M. R., Ma'sum, M. A., Ma'sum, M. A., Ardhianto, R., Ardhianto, R., Jatmiko, W., Jatmiko, W., 2021. Outdoor LiDAR Point Cloud Building Segmentation: Progress and Challenge. *International Conference on Advanced Computer Science and Information System*. DOI: 10.1109/ic-acsis53237.2021.9631345 MAG ID: 4200127030 S2ID: 40f1aa6ffe0e2cb6df8ad61443d3de077e8246e0.
- Goodfellow, I., Bengio, Y., Courville, A., 2016. *Deep learning*. MIT press.
- Griffiths, D., Boehm, J., 2019a. A Review on Deep Learning Techniques for 3D Sensed Data Classification. *Remote Sensing*, 11(12). <https://www.mdpi.com/2072-4292/11/12/1499>.
- Griffiths, D., Boehm, J., 2019b. SynthCity: A large scale synthetic point cloud. <http://arxiv.org/abs/1907.04758>. arXiv:1907.04758 [cs].
- Guo, M.-H., Cai, J.-X., Liu, Z.-N., Mu, T.-J., Martin, R. R., Hu, S.-M., 2021. Pct: Point cloud transformer. *Computational Visual Media*, 7, 187–199.
- Hrutka, B. P., Hrutka, B. P., Siki, Z., Siki, Z., Takács, B., Takács, B., 2022. VOXEL-BASED POINT CLOUD SEGMENTATION AND BUILDING DETECTION. *The international archives of the photogrammetry, remote sensing and spatial information sciences*, XLVIII-4/W1-2022, 209–215. DOI: 10.5194/isprs-archives-xxviii-4-w1-2022-209-2022 MAG ID: 4290098680 S2ID: 35c8c74284d119ee40a66713f3614ce20d075c2d.
- Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q. V., Sung, Y., Li, Z., Duerig, T., 2021. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. <http://arxiv.org/abs/2102.05918>. arXiv:2102.05918 [cs].
- Jin, Z., Lei, Y., Akhtar, N., Li, H., Hayat, M., 2022. Deformation and correspondence aware unsupervised synthetic-to-real scene flow estimation for point clouds. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7233–7243.
- Karara, G., Hajji, R., Poux, F., 2021. 3D Point Cloud Semantic Augmentation: Instance Segmentation of 360° Panoramas by Deep Learning Techniques. *Remote Sensing*, 13(18). <https://www.mdpi.com/2072-4292/13/18/3647>.
- Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., Girshick, R., 2023. Segment anything. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, Paris, France, 3992–4003.
- Liang, F., Wu, B., Dai, X., Li, K., Zhao, Y., Zhang, H., Zhang, P., Vajda, P., Marculescu, D., 2023. Open-Vocabulary Semantic Segmentation with Mask-adapted CLIP. <http://arxiv.org/abs/2210.04150>. arXiv:2210.04150 [cs].

- Lin, C., Sun, P., Jiang, Y., Luo, P., Qu, L., Haf-fari, G., Yuan, Z., Cai, J., 2022. Learning Object-Language Alignments for Open-Vocabulary Object Detection. <http://arxiv.org/abs/2211.14843>. arXiv:2211.14843 [cs].
- Liu, M., Shi, R., Kuang, K., Zhu, Y., Li, X., Han, S., Cai, H., Porikli, F., Su, H., 2024. Openshape: Scaling up 3d shape representation towards open-world understanding. *Advances in Neural Information Processing Systems*, 36.
- Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., Zhang, L., 2023. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. <http://arxiv.org/abs/2303.05499>. arXiv:2303.05499 [cs].
- Lu, Y., Xu, C., Wei, X., Xie, X., Tomizuka, M., Keutzer, K., Zhang, S., 2023.
- Mahphood, A., Arefi, H., 2023. DENSITY-BASED METHOD FOR BUILDING DETECTION FROM LIDAR POINT CLOUD. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-4/W1-2022, 423–428. DOI: 10.5194/isprs-annals-x-4-w1-2022-423-2023 MAG ID: 4316127354 S2ID: c787e17b98bd07755b193689f552cb6ca2e6a536.
- Peng, S., Genova, K., Jiang, C., Tagliasacchi, A., Pollefeys, M., Funkhouser, T. et al., 2023. Openscene: 3d scene understanding with open vocabularies. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 815–824.
- Qi, C. R., Su, H., Mo, K., Guibas, L. J., 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 652–660.
- Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I., 2021. Learning transferable visual models from natural language supervision.
- Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., Zeng, Z., Zhang, H., Li, F., Yang, J., Li, H., Jiang, Q., Zhang, L., 2024. Grounded sam: Assembling open-world models for diverse visual tasks.
- Sanchez Castillo, E., Griffiths, D., Boehm, J., 2021. SEMANTIC SEGMENTATION OF TERRESTRIAL LIDAR DATA USING CO-REGISTERED RGB DATA. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2021, 223–229.
- Tabkha, A., Hajji, R., Billen, R., Poux, F., 2019. SEMANTIC ENRICHMENT OF POINT CLOUD BY AUTOMATIC EXTRACTION AND ENHANCEMENT OF 360° PANORAMAS. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W17, 355–362.
- Takmaz, A., Fedele, E., Sumner, R. W., Pollefeys, M., Tombari, F., Engelmann, F., 2023. OpenMask3D: Open-Vocabulary 3D Instance Segmentation. *arXiv preprint arXiv:2306.13631*.
- Terven, J., Cordova-Esparza, D. M., Ramirez-Pedraza, A., Chavez-Urbiola, E. A., 2023. Loss functions and metrics in deep learning.
- Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L. J., 2019. Kpconv: Flexible and deformable convolution for point clouds. *Proceedings of the IEEE/CVF international conference on computer vision*, 6411–6420.
- Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., Solomon, J. M., 2019. Dynamic Graph CNN for Learning on Point Clouds. <http://arxiv.org/abs/1801.07829>. arXiv:1801.07829 [cs].
- Xiao, A., Huang, J., Guan, D., Zhan, F., Lu, S., 2022. Transfer learning from synthetic to real lidar point cloud for semantic segmentation. *Proceedings of the AAAI conference on artificial intelligence*, 36number 3, 2795–2803.
- Xiu, H., Liu, X., Wang, W., Kim, K.-S., Shinohara, T., Chang, Q., Matsuoka, M., 2023. DS-Net: A dedicated approach for collapsed building detection from post-event airborne point clouds. *International Journal of Applied Earth Observation and Geoinformation*, 116, 103150–103150. DOI: 10.1016/j.jag.2022.103150 MAG ID: 4313200263 S2ID: aecfebf4a652115bf6b0fbee153e7b893d90cd9b.
- Xue, L., Gao, M., Xing, C., Martín-Martín, R., Wu, J., Xiong, C., Xu, R., Niebles, J. C., Savarese, S., 2023. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1179–1189.
- Zhang, P., Zhang, P., He, H., He, H., Wang, Y., Wang, Y., Liu, Y., Yang, L., Lin, H., Lin, H., Guo, L., Guo, L., Yang, W., Yang, W., 2022a. 3D Urban buildings extraction based on airborne LiDAR and photogrammetric point cloud fusion according to U-Net deep learning model segmentation. *IEEE Access*, 1–1. DOI: 10.1109/access.2022.3152744 MAG ID: 4213319905 S2ID: e7f78dcbd244fc5064c74f308152191353617f45.
- Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., Li, H., 2022b. Pointclip: Point cloud understanding by clip. 8552–8562.
- Zhang, Z., Da, F., Yu, Y., 2022c. Learning directly from synthetic point clouds for “in-the-wild” 3D face recognition. *Pattern Recognition*, 123, 108394.
- Zhou, C., Li, Q., Li, C., Yu, J., Liu, Y., Wang, G., Zhang, K., Ji, C., Yan, Q., He, L., Peng, H., Li, J., Wu, J., Liu, Z., Xie, P., Xiong, C., Pei, J., Yu, P. S., Sun, L., 2023. A Comprehensive Survey on Pretrained Foundation Models: A History from BERT to ChatGPT. <http://arxiv.org/abs/2302.09419>. arXiv:2302.09419 [cs].
- Zhu, X., Zhang, R., He, B., Guo, Z., Zeng, Z., Qin, Z., Zhang, S., Gao, P., 2023. PointCLIP V2: Prompting CLIP and GPT for Powerful 3D Open-world Learning. <http://arxiv.org/abs/2211.11682>. arXiv:2211.11682 [cs].