# Deep Semantic Segmentation for Building Detection Using Knowledge-Informed Features from LiDAR Point Clouds


Weiye Chen*
weiyec@umd.edu
University of Maryland

Zhihao Wang*
zhwang1@umd.edu
University of Maryland

Zhili Li*
lizhili@umd.edu
University of Maryland

Yiqun Xie
xie@umd.edu
University of Maryland

Xiaowei Jia
xiaowei@pitt.edu
University of Pittsburgh

Anlin Li
anlinl@andrew.cmu.edu
Carnegie Mellon University



## ABSTRACT

Airborne LiDAR point clouds record three-dimensional structures of ground surfaces with high precision, and have been widely used to identify geospatial objects, facilitating the understanding of the distribution and changing dynamics of the environment. Detection can be complicated by the complex structures of ground objects and noises in LiDAR point clouds. Related work has explored the use of deep learning techniques such as YOLO in detecting geospatial objects (e.g., building footprints) on both optical imagery and LiDAR point clouds. However, deep networks are data hungry and there are often limited labeled samples available for many geospatial object mapping tasks, making it difficult for the models to generalize to unseen test regions. This paper describes the framework used in the 11$^{th}$ SIGSPATIAL Cup Competition (GIS CUP 2022), which received the top-3 performance. Our framework incorporates domain knowledge to reduce the difficulty of learning and the model's reliance on large training sets. Specifically, we present knowledge-informed feature generation and filtering based on morphological characteristics to improve the generalizability of learned features. Then, we use a deep segmentation backbone (U-Net) with training- and test-time augmentation to generate preliminary candidates for building footprints. Finally, we utilize domain rules (e.g., geometric properties) to regularize and filter the detections to create the final map of building footprints. Experiment results show that the strategies can effectively improve detection results in different landscapes.



## CCS CONCEPTS

• **Information systems** → **Spatial-temporal systems**; • **Computing methodologies** → **Machine learning**.

## KEYWORDS

LiDAR, object detection, building footprint, deep learning


*These authors contributed equally to this work.



*SIGSPATIAL '22, November 1–4, 2022, Seattle, WA, USA*



ACM ISBN 978-1-4503-9529-8/22/11…$15.00
https://doi.org/10.1145/3557915.3565985

**ACM Reference Format:**
Weiye Chen, Zhihao Wang, Zhili Li, Yiqun Xie, Xiaowei Jia, and Anlin Li. 2022. Deep Semantic Segmentation for Building Detection Using Knowledge-Informed Features from LiDAR Point Clouds. In *The 30th International Conference on Advances in Geographic Information Systems (SIGSPATIAL '22), November 1–4, 2022, Seattle, WA, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3557915.3565985


## 1 INTRODUCTION

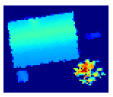
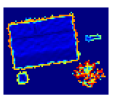
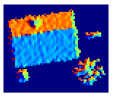
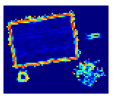
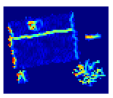
Buildings are among the most common types of objects in satellite imagery and LiDAR point clouds. Identifying building footprints provides important information to understand the structure of urban and rural communities, revealing cues of the congregation of population, supporting the decision making in industries including retailing, advertising, architecture, etc. Building footprints are also essential for various other tasks in urban planning [2], natural disaster management [6], solar energy [8], census, etc.

We aim to automatically map building footprints using LiDAR point cloud data in different landscapes. Airborne LiDAR point clouds have long been used for earth observation, and it provides detailed information of the three-dimensional structures in high resolutions. When compared with optical images, which are affected by various conditions such as lighting, reflectance and camera angles, LiDAR point clouds provide robust information not limited to horizontal features like shape, area and colors, but also vertical features like layers and depth, while the scales of details are variable.

In related work, both optical images and LiDAR point clouds have been extensively studied for building detection with machine learning and deep learning. Locally-constrained YOLO frameworks were proposed to detect small and densely-distributed building footprints in satellite imagery, which also tend to have arbitrary directions [11]. With LiDAR point clouds, methods have been developed to construct 3D building models using differential geometries [13], adaptive clustering [3], auxiliary data [1], deep networks (e.g., convolutional network [7], PointNet [4]), etc. However, existing methods require auxiliary data sources or high-resolution point clouds, or rely on data-driven feature learning, which needs large volumes of training data under diverse conditions to generalize to unseen regions.

We present our approach used in the 11$^{th}$ SIGSPATIAL Cup Competition (GIS CUP 2022), which is selected among the top three results. The approach uses a knowledge-informed feature generation and filtering strategy based on morphological characteristics, which can provide more stable and generalizable representations of buildings in different geographical regions. Moreover, we employ a

**Table 1: Illustrative examples of morphological features.**

| Name | Definition | Example |
|---|---|---|
| Image for reference | The spectral image is not used as a feature and only as a visual reference for buildings and trees. |  |
| Canopy Heights $H$ | Heights of above-ground objects (DSM - DEM). Heights of buildings and trees can be similar in this feature. |  |
| Gradient Magnitude $g_M$ | Local rate of change in surface elevation (using DSM). The change rates are the largest at the boundaries of buildings and trees. |  |
| Gradient Orientation $g_O$ | Direction of the maximum gradient of the surface elevation, which is more contiguous on building roofs but scattered on trees. |  |
| Profile Curvature $g_M^2$ | Rate of change of gradient orientation (second-order derivatives). Building roofs tend to have near-zero values as the first-order gradient magnitudes are often constant on each facade, whereas tree tops show larger variations. |  |
| Planer Curvature $g_O^2$ | Rate of change of gradient orientation (second-order derivatives), which have similar patterns to profile curvature for buildings and trees. |  |

deep segmentation backbone, U-Net, to generate preliminary candidates of building footprints. The backbone uses both training- and test-time augmentation to mitigate challenges related to limited data availability. Finally, we utilize domain rules including geometric properties to regularize and filter the detections to create the final map of building footprints.

We evaluate our framework on a set of USGS captured airborne LiDAR point clouds provided by the the 11$^{\text{th}}$ SIGSPATIAL Cup Competition, using metrics such as intersection-over-union (IoU), F1-score, etc. The results show that the presented method can effectively improve prediction results over the baselines.

## 2 METHOD

### 2.1 Knowledge-Informed Feature Generation

We generate domain knowledge-informed features using surface morphological characteristics derived from LiDAR point clouds. The goal is to construct features that can robustly distinguish buildings from other geospatial objects in LiDAR datasets. Among different types of objects, urban trees are the most common objects that often co-appear with buildings, either in long distance or being adjacent (e.g., tree canopies may overlap with building roofs). In addition, considering that trees often have similar heights as buildings, they alone cause the most prediction errors for building pixel classification. Thus, these challenging situations limit the performance of the canopy height, a traditional feature for LiDAR building detection. After analyzing the three-dimensional properties of buildings and trees, we observe that a key distinction is that the surface of buildings are largely contiguous (e.g., nearby locations normally share the same slope) whereas the surface of trees is more random and scattered (Table 1). Therefore, we use the following common morphological features in topographical studies as the input features for our deep semantic segmentation network: (1) canopy height $H$ (2) gradient magnitude $g_M$ [10] (3) gradient orientation $g_O$ [5] (4) profile curvature $g_M^2$ [9] and (5) planer curvature $g_O^2$ [9]. The features are illustrated in Table 1.

We use the following pipeline to generate the desired features. First, we re-project the point clouds from their local coordinate system to the EPSG 3857 for consistency. Second, we construct digital elevation models (DEMs) and digital surface models (DSMs) using ground points (based on default LiDAR point classification) and the last-returns, respectively. We did not use the last returns for DEM construction as a substantial proportion of locations in the data only have a single return. The spatial resolution of DEMs and DSMs is set to 0.5m, and we use inverse distance weighting (IDW) interpolation to fill empty pixels. The canopy height is computed as the difference between DSMs and DEMs. The gradient magnitude, gradient orientation, profile and planar curvatures are derived from DSMs (since DEMs do not contain above-ground objects).

### 2.2 Deep Semantic Segmentation

We use the U-Net architecture to identify building pixels using the morphological features and their learned representations across multiple scales. U-Net consists of two parts, an encoder and a decoder, as shown in Fig. 1. In the encoder, the input features are passed through several stages of convolutional layers, and the resolution is gradually reduced using strides to learn multi-scale features. In the decoder, coarse-resolution features are up-sampled through deconvolutional layers. To help recover fine-grained signals at higher resolutions, the up-sampled features are concatenated with features from corresponding layers in the encoder (Fig. 1). At the final layer, U-Net outputs pixel-level classifications of building vs. non-building at the resolution of the original input image.

In our implementation, both the encoder and the decoder have 3 blocks, respectively. For the encoder, each block consists of two successive $3 \times 3$ convolutions, where the second convolution uses a stride of 2 to reduce the image size to half. For the decoder, each block consists of a $3 \times 3$ deconvolutional layer and a $3 \times 3$ convolutional layer. After each deconvolutional layer, the feature map is concatenated to the corresponding feature map from the encoder with the same size and is then followed by a $3 \times 3$ convolution. At the final stage, we get a feature map with 2 channels, where one represents the "confidence" on the building class and the other for non-building.

We train the network using the Dice loss, which can alleviate class imbalance issues common for the building detection problem (i.e., the amount of building pixels is much smaller compared to non-building pixels). The dice loss is defined as: $L_{dice} = 1 - 2 \cdot \sum_i y_i \hat{y}_i / \sum_i (y_i + \hat{y}_i)$, where $y_i$ and $\hat{y}_i$ are the prediction and ground truth label, respectively. We use the Adam optimizer with a learning rate of $10^{-4}$ for the training steps.

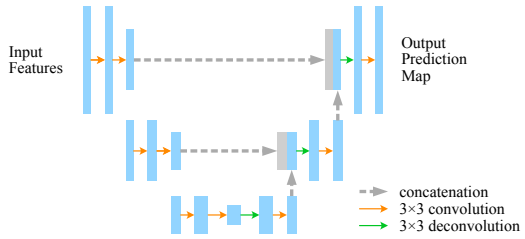



**Figure 1: U-Net based architecture for segmentation.**

In testing, we use test-time augmentation [11] to improve the prediction performance and reduce the generalization error. Specifically, we augment test images with horizontal and vertical flips, as well as rotations by 90, 180, and 270 degrees; we obtain the final result by averaging the softmax probabilities of the original and transformed images.

## 2.3 Post-Processing

The pixel-level predictions are transformed to vector-level footprints as the first step in the post-processing. As noises are commonly presented in LiDAR datasets, leading to undesired artifacts (e.g., irregular boundaries) and inaccuracies in the vector maps, we use rule-based filters to improve the prediction quality as well as the fidelity. First, we use the ground-mask as the filter to remove detections that appear in environments (e.g., river, lakes) where buildings are unexpected. Second, buildings in general occupy an area (the 2D projection on the surface) that is larger than a certain threshold. This characteristic can be easily used to remove small polygons created by tree tops, small yard cabinets, etc. Based on the observations in the datasets, we select 15 square-meters as the threshold to remove unlikely detections. There may be exceptions but we found this filtering is often effective. On top of the filters applied, we use geometric regularization on our prediction, as the results produced by the U-Net are in pixelated formats, which inevitably introduces artifacts. We apply the Douglas–Peucker algorithm to simplify the boundary geometry of the detected objects, so that they are better aligned with the boundaries of the actual footprints.

# 3 EXPERIMENTS

## 3.1 Experimental Settings

**Dataset.** We train and evaluate our model using the LiDAR data from the 3D Evaluation Program at the USGS. Specifically, 17 LiDAR files with building footprints are provided across the U.S. and we selected 15 files for training and 2 files for validation. Fig. 2 shows the selected validation areas with different surface characteristics: buildings and trees co-appear with similar heights in Area-1, whereas buildings and trees are well separated apart in Area-2. Moreover, the buildings in Area-1 are much smaller than those in Area-2 as shown in Fig. 2, leading to an easier semantic segmentation task. Based on the two areas, we expect the knowledge-informed features to have better powers in distinguishing buildings from trees compared to the raw canopy height model.

**Evaluation Metrics.** We evaluate our U-Net results using the F1-score and final post-processed building footprints using Intersection over Union (IoU). F1-score is defined as the harmonic mean of precision and recall: $F1 = 1/(pre^{-1}+rec^{-1})$, where $pre = TP/(TP+FP)$, $rec = TP/(TP+FN)$; and $TP$, $FP$ and $FN$ are true positives, false positives and false negatives, respectively. IoU is defined as: $IoU = A_{int}/A_{uni}$, where $A_{int}$ and $A_{uni}$ are areas of the intersection and union between predicted and true footprints, respectively.

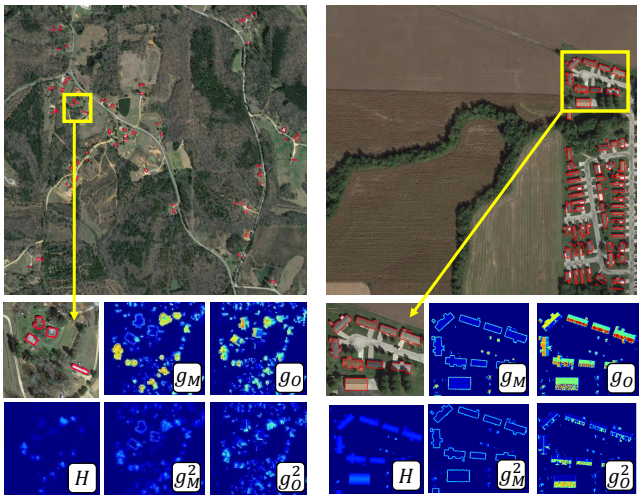



**Figure 2: Examples of morphological features in validation Areas 1 (left) and 2 (right). Spectral images are used only for reference.**

**Table 2: Result evaluation at the pixel level.**

| | | No Augmentation | | | Test-Time Augmentation | | |
|---|---|---|---|---|---|---|---|
| | Test | F1 | Precision | Recall | F1 | Precision | Recall |
| H | Area-1 | 0.841 | 0.775 | 0.919 | **0.846** | **0.781** | **0.921** |
| | Area-2 | 0.921 | **0.934** | 0.908 | **0.923** | **0.934** | **0.912** |
| H+KI | Area-1 | 0.883 | 0.845 | **0.925** | **0.888** | **0.854** | 0.924 |
| | Area-2 | 0.936 | 0.951 | 0.921 | **0.941** | **0.956** | **0.926** |

*H: Height; KI: Knowledge-informed features.

## 3.2 Results and Analysis

**Effects of knowledge-informed features.** Models learned from knowledge-informed features and height consistently achieved the best results compared to those learned from height only. According to Table 2, the model using knowledge-informed features is able to improve the averaged F1-score from 0.883 to 0.911 and 0.886 to 0.916, without and with test-time augmentation, respectively. It is worth noting that using knowledge-informed features, the model improvement in Area-1 (0.841 to 0.883 in F1-score) is much larger than that in Area-2 (from 0.921 to 0.936), especially the precision increasing from 0.775 to 0.845. Consistent with our expectations, Area-1 is more difficult than Area-2 for model classification only using heights, because buildings and trees show roughly similar heights and co-appear frequently in this area. By introducing the knowledge-informed features, the models were able to distinguish buildings from surrounding trees using morphological characteristics.

**Effects of test-time augmentation.** From the results, we can see that test-time augmentation shows the ability to further improve

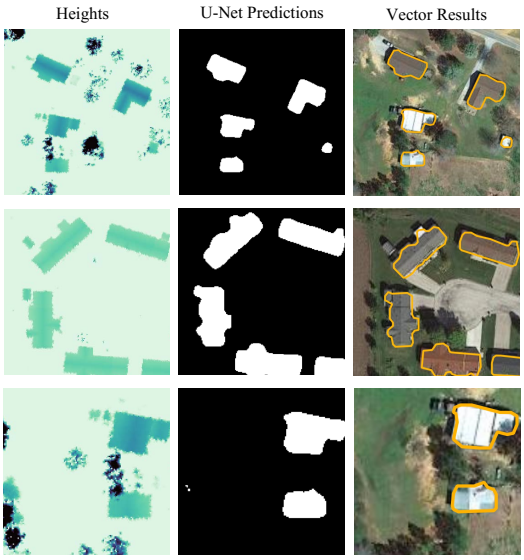



**Figure 3: Examples: Inputs (height as an example) and results.**

model accuracy. By incorporating votes from multiple augmented images, the model is able to reduce the variance in the final predictions. The improvement is relatively small as the suburban and rural areas have simpler and more homogeneous building structures, resulting in a smaller baseline variance. We expect the effect of test-time augmentation to have more effects in regions with greater variance, such as urban and metropolitan areas.

**Effects of post-processing.** As shown in Fig. 3, post-processing strategies make the semantic segmentation results visually more realistic. Converting pixelated building masks to vector polygons inevitably results in jagged or irregular boundaries, and the use of smoothing methods such as boundary simplification reduces the irregularities and improves solution quality (e.g., vector IoU). Moreover, domain rules led to further improvements. For example, applying the ground mask removes false detections located in regions where buildings are in general not expected (e.g., inside lakes or water bodies).

**IoU Results.** Table 3 shows the vector-based IoU scores for Areas 1 and 2. The trends are similar to pixel-based F1-scores, where the full approach with augmentation and knowledge-informed features obtained the best results. We additionally compare the results before and after post-processing. As we can see, the rule-based refinements were able to greatly improve the scores particularly in Area 1.

**Table 3: Result evaluation at the object (vector) level.**

| | | No Augmentation | | Test-Time Augmentation | |
|---|---|---|---|---|---|
| | Test | Raw | Post-processed | Raw | Post-processed |
| H | Area-1 | 0.482 | 0.545 | 0.516 | **0.575** |
| | Area-2 | 0.851 | 0.851 | **0.855** | **0.855** |
| H+KI | Area-1 | 0.600 | 0.661 | 0.632 | **0.673** |
| | Area-2 | 0.882 | 0.882 | **0.885** | **0.885** |

*H: Height; KI: Knowledge-informed features.

## 4 CONCLUSIONS

We presented a knowledge-informed deep segmentation approach for building footprint detection from LiDAR point clouds. Specifically, we used morphological characteristics from LiDAR-derived topographical models that are distinctive between buildings and other objects which tend to be confused as buildings (e.g., trees). U-Net was then used to learn feature representation at multiple scales to generate pixelated predictions with test-time-augmentation, which were converted to building footprints and refined during rule-based post-processing. Experiment results showed that the approach can effectively improve detection results in different landscapes. In future work, we will further improve the generalizability of the model using heterogeneity-aware frameworks [12].

## ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 2105133, 2126474 and 2147195; NASA under Grant No. 80NSSC22K1164 and 80NSSC21K0314; USGS under Grant No. G21AC10207; Google's AI for Social Good Impact Scholars program; the DRI award at the University of Maryland; Pitt Momentum Funds award and CRC at the University of Pittsburgh.

## REFERENCES

[1] Saleh Albeaik, Mohamad Alrished, Salma Aldawood, Sattam Alsubaiee, and Anas Alfaris. 2017. Virtual cities: 3d urban modeling from low resolution lidar data. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 1–4.

[2] Cici Alexander, Sarah Smith-Voysey, Claire Jarvis, and Kevin Tansey. 2009. Integrating building footprints and LiDAR elevation data to classify roof structures and visualise buildings. *Computers, Environment and Urban Systems* 33, 4 (2009), 285–292.

[3] Philip E Brown, Yaron Kanza, and Velin Kounev. 2019. Height and facet extraction from LiDAR point cloud for automatic creation of 3D building models. In *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 596–599.

[4] Zdzisław Kowalczuk and Karol Szymański. 2019. Classification of objects in the LIDAR point clouds using Deep Neural Networks based on the PointNet model. *IFAC-PapersOnLine* 52, 8 (2019), 416–421.

[5] Dae Geon Lee, Young Ha Shin, and Dong-Cheon Lee. 2020. Land cover classification using SegNet with slope, aspect, and multidirectional shaded relief images derived from digital surface model. *Journal of Sensors* 2020 (2020).

[6] Xue Li, Ning Shu, Jian Yang, and Liang Li. 2011. The land-use change detection method using object-based feature consistency analysis. In *2011 19th International Conference on Geoinformatics*. IEEE, 1–6.

[7] Evangelos Maltezos, Anastasios Doulamis, Nikolaos Doulamis, and Charalabos Ioannidis. 2018. Building extraction from LiDAR data applying deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters* 16, 1 (2018), 155–159.

[8] M Rylatt, S Gadsden, and K Lomas. 2001. GIS-based decision support for solar energy planning in urban environments. *Computers, Environment and Urban Systems* 25, 6 (2001), 579–603.

[9] Shaohui Sun and Carl Salvaggio. 2013. Aerial 3D building detection and modeling from airborne LiDAR point clouds. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 6, 3 (2013), 1440–1449.

[10] Melis Uzar. 2014. Automatic building extraction with multi-sensor data using rule-based classification. *European Journal of Remote Sensing* 47, 1 (2014), 1–18.

[11] Yiqun Xie, Jiannan Cai, Rahul Bhojwani, Shashi Shekhar, and Joseph Knight. 2020. A locally-constrained yolo framework for detecting small and densely-distributed building footprints. *International Journal of Geographical Information Science* 34, 4 (2020), 777–801.

[12] Yiqun Xie, Erhu He, Xiaowei Jia, Han Bao, Xun Zhou, Rahul Ghosh, and Praveen Ravirathinam. 2021. A statistically-guided deep network transformation and moderation framework for data with spatial heterogeneity. In *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 767–776.

[13] Qian-Yi Zhou and Ulrich Neumann. 2008. Fast and extensible building modeling from airborne LiDAR data. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*. 1–8.