Building Instance Segmentation for Dense Urban Settlements

Adnan Firoze[1], **Raymond A. Yeh**[1], **Daniel Aliaga**[1]

[1]Department of Computer Science, Purdue University, West Lafayette, Indiana, USA
afiroze@purdue.edu, rayyeh@purdue.edu, aliaga@purdue.edu

Abstract

About 25% of the world's population live in informal urban settlements containing densely packed buildings (approximately 8,000 houses per km^2) which do not lend themselves favorably to state-of-the-art satellite-based building segmentation methods due to, for example, occlusion, vegetation, shadows and low resolution. To address these challenges, we introduce a novel instance segmentation and counting approach for dense buildings. Our system first extracts a conservative set of tentative building center points using a deep network for jumpstarting a Segment Anything Model 2 (SAM2) module to produce an initial over-segmentation. Second, we use a graph neural network to refine the over-segmented regions into polygons representing accurate building masks. Experiments show that our approach achieves higher accuracy in instance segmentation and counting especially in challenging densely packed building areas in Brazil, Mexico, India, Pakistan, and Kenya, for instance.

Appendix, Code, Data —
https://github.com/adnan0819/UDOS

1 Introduction

Segmenting, counting, and monitoring urban settlements globally is crucial for many downstream sustainability and urban planning applications. While the ideal is for cities to be organized and highly structured, more than 25% of the world's urban population lives in dense, compact, and disorganized urban neighborhoods (United Nations Human Settlements Programme 2018). United Nations suggest that unless active sustainability measures are taken, even more of the urban population will live in such settlements by 2050 (United Nations Human Settlements Programme 2024). This makes it crucial to accurately and efficiently segment and count such urban structures at a global scale.

Prior urban segmentation research has focused on identifying informal urban settlement areas as a whole (e.g., (Zhang et al. 2024)) or on instance segmentation of mostly separated objects or buildings. Methods use point-clouds/LiDAR, unmanned aerial vehicles (UAV), and aerial/satellite images (see (Zhang et al. 2021; Gamal et al. 2020; Xiao et al. 2022; Zhang et al. 2022)). General segmentation methods have made good strides e.g., Mask R-CNN (He et al. 2017), Segment Anything Model 2 (SAM2) (Ravi et al. 2025), ParSense (Siddiqui et al. 2024), pyramid-based approaches (PSPNet (Zhao et al. 2017)), Attention-based networks (PSANet (Zhao et al. 2018), DANet (Fu et al. 2019), CARAFE++ (Hu et al. 2021)), and transformers (SWIN (Liu et al. 2021b)), but there remains a salient gap in instance segmentation of images with densely packed objects (Yang, Yang, and Gao 2023).

Prior urban-specific building segmentation methods (e.g., LOGCAN++ (Ma et al. 2025), GlobalMapper (He and Aliaga 2023), UrbanBIS (Yang et al. 2023)) exploit typical urban domain features such as street-side layouts and city block patterns. However, in our targeted informal settlement areas, building density is the highest globally (i.e., one square kilometer can have over 8,000 buildings - similar in color, densely packed, and abutting or partially overlapping) and buildings are not well organized into formal street and city blocks structures which greatly hinders segmentation.

We propose the first satellite-based urban building instance segmentation and counting pipeline focused on informal building settlements where roofs are highly similar, often overlapping, occluding, and/or abutting (Fig. 1). Our approach consists of three main phases: (a) We use a convolutional neural network (CNN) to obtain an initial super-set of building centers. (b) These centers are then used to obtain preliminary building instances. (c) Our methodology converts the preliminary instances into a graph isomorphic network with edge features (GINE) (Hu et al. 2020), and then alters and refines the preliminary building instances by making use of node and edge features transmitted via multiple iterations of message passing. Each tentative building has a set of node features including its shape type and parameters (i.e., one of five parameterized shapes that span 99% of all building outlines), geospatial coordinates, bounding box dimensions, orientation, and references to adjacent buildings. Through multiple iterations of this specialized graph convolution, we learn and optimize the features of the nodes from the initial tentative segmentation. Our method optimizes bounding box size, geospatial coordinates, shape type parameters, and orientation as well as deciding if an initially identified node should remain.

To evaluate model performance, we labeled 23,984 build-

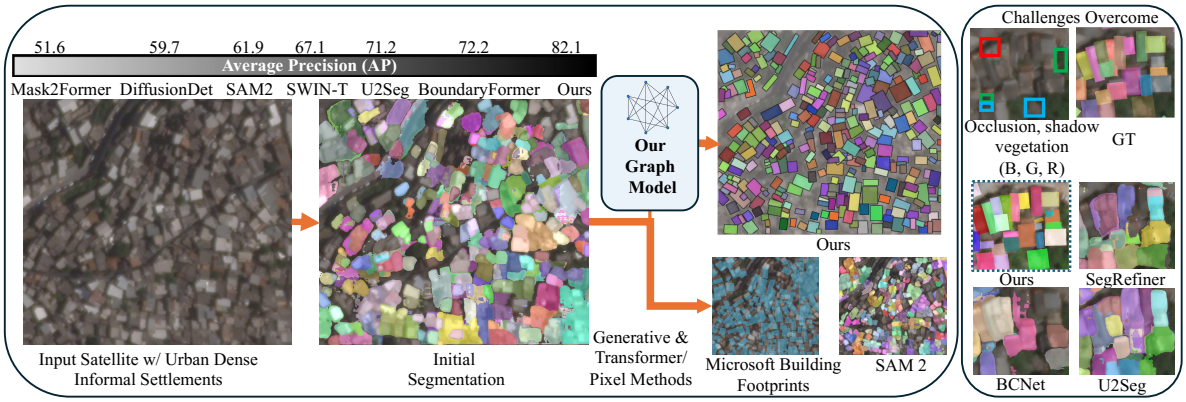

Figure 1: Method Overview. Our graph-based building instance segmentation from satellite images exploits fundamental shape types and building features to segment dense informal urban settlements, outperforming state-of-the-art baseline models.

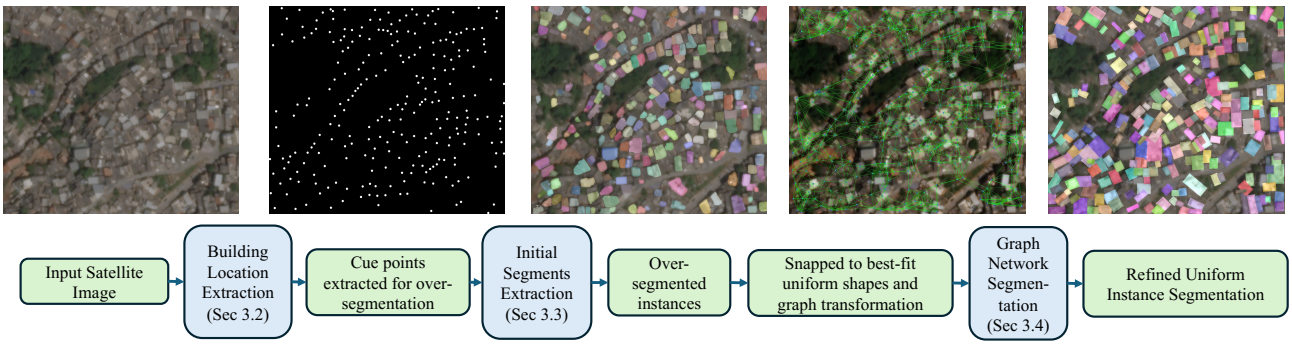

Figure 2: Visual Pipeline. Illustrative workflow of our dense urban settlement segmentation.

ing roofs in satellite images from Planet Skysat Satellite (PlanetLabs 2024b) sampled from the five largest informal urban settlements of the world (for Humanity 2023): Belo Horizonte (Brazil), Ciudad Nezahualcóyotl (Mexico), Orangi Town (Pakistan), Mumbai (India), and Kibera (Kenya). We highlight that state-of-the-art (SOTA) building datasets, such as Microsoft Building Footprints (Microsoft 2018) and Open Street Maps (Contributors 2004), do not have footprints of such settlements – underscoring the importance of our work. *Our labeled datasets will be made public after the anonymized review*. Using these datasets, our method can identify individual building shapes in these challenging urban areas. As we show, our performance in such difficult areas is consistently superior to 10 recent baseline instance segmentation and counting approaches (by 15-30%). Our average segmentation accuracy is 80.61%, average count accuracy is 88.9%, and an average computational throughput is 697 image tiles/minute using a single GPU (each tile is 256 $\times$ 256 pixels).

In addition, our method can be applied to organized urban areas, as we show by also applying our approach to existing satellite-based urban building benchmark datasets, achieving an IoU of 0.96 where SOTA building-specific models (e.g., AFM2Mask (Li et al. 2022), LOGCAN++ (Ma et al. 2025)) only reach an IoU of 0.86 and 0.95, respectively (Appendix). Globally, our approach has the potential of mapping the homes of an additional 130 million people, in addition to improving building instance segmentation in general. See Fig. 2 for an overview of our approach.

Our main contributions include:

- **Instance Segmentation:** single building segmentation supporting dense informal urban settlements that are highly disorganized, similar, abutting and/or partially overlapping,
- **Counting:** a framework to count buildings including in often overlooked communities, globally corresponding to about 130 million people's homes,
- **Efficiency and accuracy:** our method exploits parameterized shape types spanning most buildings (He and Aliaga 2023; Steadman 2006) and features of parallelism, consistency, overlap, and distance, and
- **Dataset:** segmentations of dense informal settlements across five countries and three continents for future work.

2 Related Work

General Instance Segmentation in Dense Urban Areas. Such segmentation is challenging since the resolution is relatively low and structures are irregular, closely spaced, and visually extremely similar. Mask R-CNN (He et al. 2017), Swin Transformers (Liu et al. 2021b), and layer-based segmentation methods (e.g., (Chen, Wu, and Merhof 2022; Yu

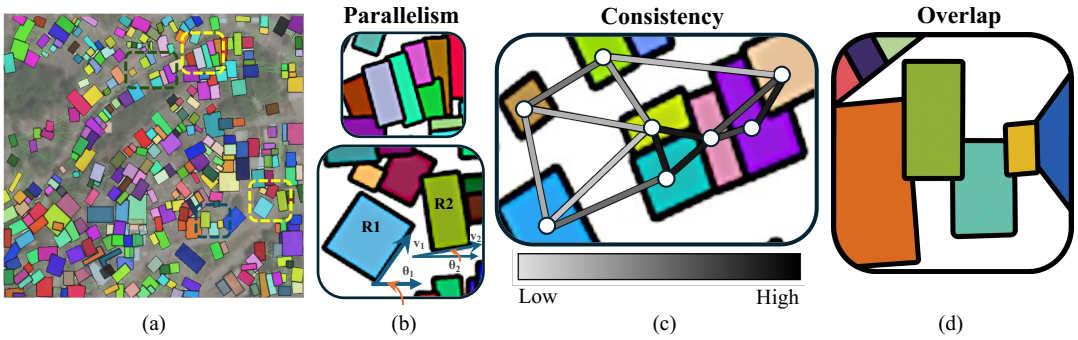

Figure 3: Neighborhood Features Expressed as Edge Attributes. (a) Reference RGB image with overlaid output from our method (dashed Y, G, B = parallelism, consistency, overlap), (b) Illustration of parallelism (mathematically defined in Sec. 3.4) (Low = shape and size uniformity, High = divergent shapes and sizes), (c) Visualization of consistency feature (inside a packed vicinity, tightly packed buildings exhibit high consistency), (d) Overlap tolerance showing occlusion among multiple buildings (modern (sparse) cities = low overlap, dense informal settlements = allowing overlap tolerance outputs accurate segmentation.).

et al. 2022)) struggle in this case. The large number of instances hinders layer-based segmentation methods.

Graph-based and Generative Approaches. Firoze et al. (2023) show promise by converting closed contours of tree crowns into nodes and using graph convolution to extract segments. However, this requires UAV videos. For lower-resolution images, generative deep learning shows promise. Xu et al. (2021) and He, Shan, and Aliaga (2023), for instance, leverage recognizable patterns within organized city blocks; but, dense informal settlements are not organized.

Counting vs. Instance Segmentation. Counting-focused methods, e.g., MCAC-ABC123 (Hobley and Prisacariu 2024; Lian et al. 2019; Liu et al. 2021a), use regression to estimate object counts. However, these approaches cannot differentiate individual instances, which is crucial for applications needing spatial footprint analysis. Nonetheless, we do compare to counting methods.

Urban Segmentation and Datasets. Recent works like UVSAM (Zhang et al. 2024) and LOGCAN++ (Ma et al. 2025) perform urban land segmentation but lack instance-level distinction, while combining them with SAM2 (Ravi et al. 2025) for fine-grained instance segmentation shows limited performance in dense settlements. Urban segmentation datasets such as Microsoft Building Footprints (MSBF) (Microsoft 2018) and OpenStreetMap (OSM) (Contributors 2004) leverage generative learning for extensive building outlines with claimed global coverage, but largely focus on formal, well-mapped urban areas and do not fully generalize to informal settlements with densely packed, irregular structures across our five cities of interest, underscoring the need for tailored approaches capable of handling high-density and complex patterns.

3 Building Instance Segmentation

Our approach has three main phases: First, with an RGB satellite image as input, we use CNNs for the initial detection of building centers. Second, we use the detected centers as cues to SAM2 for obtaining an overestimation of building instances. Lastly and most importantly, we optimize the preliminary segments and output high-accuracy building footprints including for dense informal settlements.

3.1 Representation

Graph Formulation. We formulate building extraction as node optimization on overhead RGB imagery of dense informal housing, initially over-segmented and snapped to parameterized shape types (Fig. 2 and Fig. 6). Building masks are represented as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where nodes in $\mathcal{V}$ correspond to individual buildings and edges in $\mathcal{E}$ encode spatial relationships between neighbors. Each node is defined by geometric properties and one of five parameterized shape types $\{R, U, T, I, L\}$ (Rectangular, U-shaped, T-shaped, I-shaped, L-shaped) formulated by prior works (Castagno and Atkins 2018; Steadman 2006; Zeng et al. 2018), which capture 99% of target buildings based on IoU (histogram in Appendix). These parameterized types improve performance while providing robustness to non-uniform and degraded building masks, with neighborhood contextual information via message passing (Fig. 5).

Node and Edge Attributes. Each node $v_i \in \mathcal{V}$ is associated with attributes, including shape id, shape type (one of R, U, T, I, L), shape geometry parameters, x and y-coordinates, bounding box width and height, orientation angle, and auxiliary features (i.e., a representative building image patch and a validity flag which is set to 1 if the shape appears in the final output and 0 otherwise). The edge features between nodes include a parallelism score, consistency score, overlap amount, and distance, which provide contextual data on spatial relationships between adjacent buildings.

Objective. Thus, given an input RGB image, the task is to produce a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where the predicted node attributes accurately represent each building shape. The optimization objective is formulated as: $\mathcal{O} =$

$$\frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \left(\mathbf{w} \odot (\mathbf{z}_v - \mathbf{x}_v)^2 \right) + \lambda \sum_{(v_i, v_j) \in \mathcal{E}} \varphi(\mathbf{z}_i, \mathbf{z}_j), \quad (1)$$

where $\mathbf{w}$ is the weight associated with each feature attribute, $\mathbf{z}_v$ is the predicted feature vector for node v, $\mathbf{x}_v$ is the

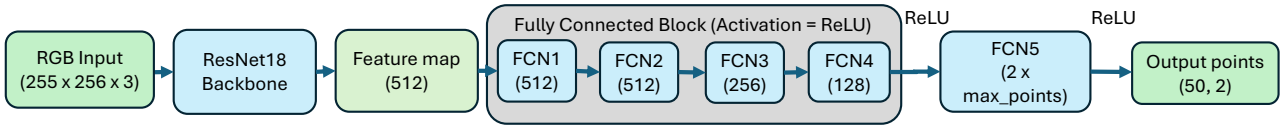

Figure 4: Convolutional Center Point Extraction Network. Resnet-18 based network to conservatively extract tentative roof center points for obtaining an oversegmentation with SAM2.

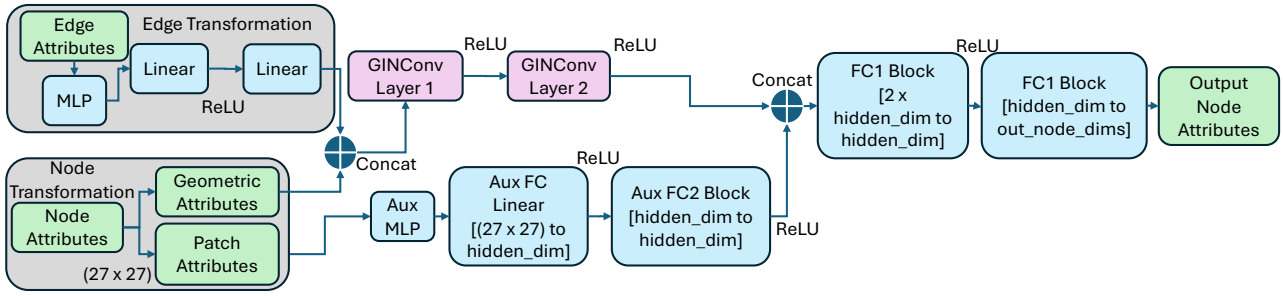

Figure 5: GINConv based Graph Network. Our novel graph isomorphic network leveraging patch and neighborhood features extracts accurate building footprints. Edge attributes include neighborhood information (parallelness, consistency, overlap, and distance). Output node attributes include shape ID, shape type and parameters, coordinates, dimensions, and orientation angle.

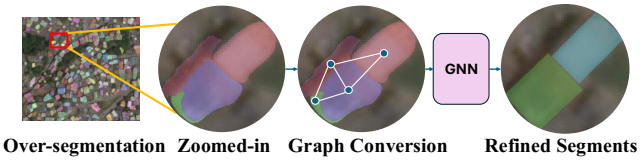

Figure 6: Shape Refinement Workflow. Visual workflow of how initial oversegmentation is refined through our graph-based model. This example shows the removal of two invalid nodes that represent shadow and occlusion due to vegetation.

ground-truth feature vector for node v, and $\varphi(\mathbf{z}_i, \mathbf{z}_j)$ is a regularization term for spatial consistency between neighboring nodes v_i and v_j, based on edge features. The term λ is a weighting factor for the regularization. This objective function optimizes the target geometric properties with spatial coherence, ensuring an accurate representation of building shapes across the output graph with accurate footprints.

3.2 Building Center Extraction

This point extraction module uses a deep learning approach to predict building center points for input into a SAM2 (Ravi et al. 2025) block, generating an initial oversegmentation. We input 256 px $\times$ 256 px RGB tiles at 0.5 meters per pixel (mpp) resolution of a larger image, denoted as $\mathbf{X} \in \mathbb{R}^{256\times256\times3}$. The tile is processed through a ResNet-18 backbone for feature extraction, resulting in a 512-dimensional feature map $\mathbf{F} \in \mathbb{R}^{512}$ (see Fig. 4). The feature map is then passed through fully connected layers, progressively refining the representation. The final layer outputs the coordinates of up to 50 predicted points as $\mathbf{Y} = [(x_1, y_1), (x_2, y_2), \ldots, (x_{50}, y_{50})] \in \mathbb{R}^{50\times2}$. These tiles are subsequently mosaicked to generate a complete point map. Further, we experiment with SOTA tracking approaches (e.g., FCOS (Tian et al. 2019) and Faster RCNN (Ren et al. 2015)) for center extraction but ours perform faster (see Appendix).

3.3 Preliminary Building Segment Extraction

The center points are given to a SAM2 network for obtaining a conservative initial set of freeform masks. Following this over-segmentation, the building shapes are approximated as parameter-optimized R, U, T, I, L types, and are represented as a node in a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Each building $R_i \in \mathbb{R}^{h_i\times w_i\times\theta_i}$ is described by height h_i, width w_i, and rotation θ_i of its enclosing bounding box. Edges are created between nodes i and j if the Euclidean distance between their centroids, $d(c_i, c_j)$, is within a predefined empirically determined radius $r = 80$ meters, defined by:

$$\mathcal{E} = \{(i,j) \mid d(c_i, c_j) \leq 80\}, \quad c_i = \left(\frac{h_i}{2}, \frac{w_i}{2}, \theta_i\right) \quad (2)$$

3.4 Graph-based Shape Optimization and Segmentation Phase

The primary task is now optimization and refinement of the nodes in $\mathcal{G}$. The task of predicting node attributes in such a graph involves accurately estimating essential geometric parameters of shapes, such as their position, size, and orientation. Here, the set of vertices $\mathcal{V}$ corresponds to individual shapes characterized by their attributes, and the edges $\mathcal{E}$ encode relational information between these shapes. Each vertex $v_i \in \mathcal{V}$ is associated with a feature vector $\mathbf{x}_i \in \mathbb{R}^d$, encompassing attributes noted above. The edges have attributes $\mathbf{e}_{ij} \in \mathbb{R}^m$ to help contextualize the interactions between nodes. The objective of this step is to predict node attributes, denoted as $\mathbf{z}_i$, which include the shape ID, shape type, shape parameters, x-y coordinates, bounding box width and height, orientation angle, and validity flag for

each node i. Mathematically, the model processes node features through GINConv (Hu et al. 2020) layers and auxiliary fully connected layers, producing predictions as

$$\mathbf{z}_i = \text{FC}_2\left(\text{ReLU}\left(\text{FC}_1\left([\mathbf{h}_i^{(L)}; \text{Aux}_i]\right)\right)\right), \quad (3)$$

where $\mathbf{h}_i^{(L)}$ represents the final node embedding after message passing through L layers of GINConv (Hu et al. 2020). This formulation encapsulates the integration of both node and edge features, ultimately enabling accurate predictions of the desired geometric attributes for each shape in the graph. The edge attribute of distance is given as the Euclidean distance between the centroids of adjacent nodes i and j, denoted as $c_i = (x_i, y_i)$ and $c_j = (x_j, y_j)$, respectively and $distance = dist(c_i, c_j)$ where $dist$ is Euclidean distance.

We discuss neighborhood features (i.e., parallelism, consistency, and overlap) and salient components below.

• *Parallelism:* Parallelism measures the alignment of two footprints (embedded as edge features between two neighboring buildings) based on cosine similarity derived from the dot product of their orientation vectors (see Fig. 3b) relative to their local origin (bottom left). Each building's orientation angle $(\theta))$ is represented as a unit vector in 2D space: $[cos(\theta), sin(\theta)]$. The parallelism between two orientation vectors v_i and v_j for buildings R_i and R_j is:

$$\text{Parallelism}_{ij} = (v_i \cdot v_j)/(|v_i| \times |v_j|). \quad (4)$$

Perfect parallelism is represented when the value turns 1, whereas theoretically, 0 is perfectly perpendicular.

• *Consistency:* This feature measures how typical a building pair's separation is relative to their neighborhood. For each building R_i, we compute pairwise distances $dist(i, j)$ to all neighbors within 80m and define the median as the neighborhood consistency distance (NCD). Each edge receives a consistency score measuring its deviation from this median, scaled so 0.5 is at the median:

$$\text{Consistency}_{ij} = 1 - (|\text{dist}(i, j) - \text{NCD}|)/0.5 \quad (5)$$

Scores near 1 indicate typical spacing, while scores near 0 indicate outliers, determining how much neighbor R_j influences R_i's features—clustered neighbors have stronger impact than distant outliers (Fig. 3c).

• *Overlap.* In dense informal settlements, significant building overlap and occlusion occur, unlike structured urban areas. This is learned between node pairs as an edge feature with an initial 10% tolerance from heuristics, updated based on samples. In the final output, if edge overlap exceeds tolerance, the smaller building is discarded. Performance gains are quantified in our ablation study (Tab. 3) and visualized by comparing models allowing unbounded overlap (Fig. 7).

Model Architecture. The model architecture (Fig. 5) consists of the GINConv (Hu et al. 2020) layers (Xu et al. 2019) for message passing, augmented by auxiliary fully connected layers for auxiliary node features, and a final multi-layer perceptron (MLP) for predicting the target node attributes. The architecture (Fig. 5) is –decomposed next.

Edge Feature Transformation. The initial edge features are processed through a multi-layer perceptron (MLP) to capture non-linear transformations. Let e_{ij} denote the edge features for an edge between nodes i and j; the edge MLP transforms e_{ij}, $\mathbf{e}_{ij} = \text{MLP}_{\text{edge}}(e_{ij})$, into a hidden dimension for use in GINConv.

GINConv Layers. The primary node features $\mathbf{x}_i$ (shape ID, shape type, x-y coordinates, bounding box width and height, orientation angle, and validity) are input to two GINConv layers for message passing. At each layer l, node features $\mathbf{h}_i^{(l)}$ are updated by aggregating the transformed features from neighboring nodes $\mathcal{N}(i)$ and associated edge features, resulting in the updated node features:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\text{GINConv}^{(l)}\left(\mathbf{h}_i^{(l)}, \{\mathbf{h}_j^{(l)}, \mathbf{e}_{ij} : j \in \mathcal{N}(i)\}\right)\right), \quad (6)$$

where σ denotes the ReLU activation function.

Auxiliary Feature Processing. Auxiliary node attributes (i.e., image patches) are processed separately through two fully connected layers to avoid interference with primary node features in GINConv (Hu et al. 2020) layers. These auxiliary features are transformed into a hidden space and later concatenated with GINConv outputs (see Fig. 5).

Final Prediction. The final node representations from the GINConv layers and auxiliary transformations are concatenated and passed through additional fully connected layers to produce the target node attribute predictions:

$$\mathbf{z}_i = \text{FC}_2(\text{ReLU}(\text{FC}_1([\mathbf{h}_i^{(L)}; \text{Aux}_i]))) \quad (7)$$

where $\mathbf{z}_i$ are predicted shape type, x-y coordinates, bounding box width and height, orientation angle, and validity.

Loss Function and Optimization. We use weighted Mean Squared Error (MSE) loss to emphasize attributes prone to variability:

$$\mathcal{L} = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \left(\mathbf{w} \odot (\mathbf{z}_v - \mathbf{x}_v)^2\right) \quad (8)$$

where $\mathbf{w}$ is a learned weight vector. This single loss function achieves faster convergence by jointly optimizing discrete (one-hot encoded) and continuous features, outperforming cross entropy loss in accuracy (see Appendix).

Training and Evaluation. During training, the model learns to aggregate node and edge features through GINConv layers while auxiliary layers embed the patch features (see Fig. 5). To validate the model, we compare the predicted values against the ground truth (GT) for each target attribute. After training, the model is evaluated on unseen data to assess the accuracy of predicting each node's shape ID, shape type, position, and geometric attributes.

4 Experiments

Datasets. We curated five datasets from the largest informal settlements per UN Department of Economic and Social Affairs (United Nations Human Settlements Programme 2024): Belo Horizonte (Brazil), Ciudad Nezahualcóyotl (Mexico), Dharavi (India), Orangi Town (Pakistan), and Kibera (Kenya). Images are 0.5 mpp resolution from Planet Skysat (PlanetLabs 2024b) with 23,984 labeled buildings. Brazil and Mexico use 80-20 train-test split while we use India, Pakistan, and Kenya datasets to evaluate out-of-domain

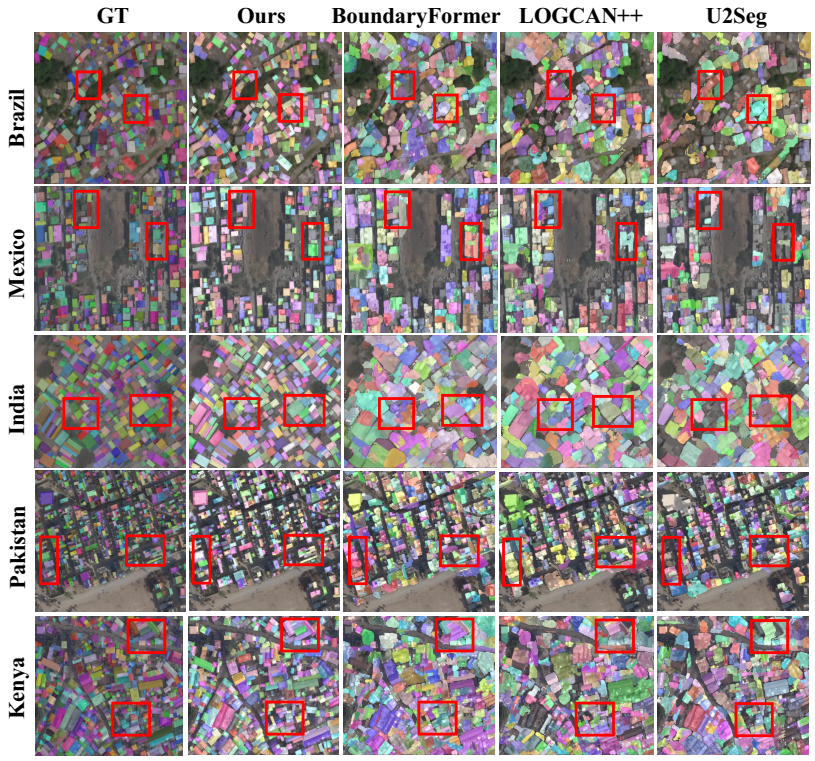

Figure 7: Qualitative Results. Comparisons of our approach vs. BoundaryFormer, LOGCAN++, and U2Seg. Red boxes highlight our closeness to GT. Comparisons to more baselines are in the Appendix.

generalization. Annotations began with World Settlement Footprint (WSF3D, WSF2019) then manual refinement.

Evaluation Metrics. We used the evaluation protocol established by He et al. (2017). For *dense instance segmentation*, we report the Average Precision (AP) over different intersection-over-union (IoU) thresholds of 0.5 and 0.75, as AP_{50} and AP_{75} respectively. For *counting*, we report the count accuracy percentage (*Acc*) with raw counts in the Appendix. To report efficiency, we use throughput represented as the number of $256px \times 256px$ tiles processed in a minute using the same hardware across all models (see Appendix).

Baselines. We compare our method to SOTA instance segmentation methods - MaskRCNN with SWIN-T backbone (Liu et al. 2021b), SAM2 (Ravi et al. 2025), DiffusionDet (Liu, Ren, and Zhao 2023), Mask Scoring RCNN (Huang et al. 2019), Mask2Former (Cheng et al. 2022), BoundaryFormer (Lazarow, Xu, and Tu 2022), U2Seg (Niu et al. 2024), the recent MCAC-ABC123 counting model (Hobley and Prisacariu 2024), and to urban specific segmentation models AFM2Mask (Li et al. 2022) and LOGCAN++ (Ma et al. 2025). In addition, we also show the performance relative to the INRIA (Maggiori et al. 2017) benchmark datasets for cities (see Appendix).

4.1 Segmentation Performance Results

Tab. 1 shows AP comparisons across multiple informal settlements, where our method outperforms all baselines in Brazil (82.1), Mexico (79.1), India (80.4), Pakistan (77.2), and Kenya (79.9), exceeding BoundaryFormer by 12.81 AP on average. Out-of-distribution performance (Tab. 2) demonstrates high accuracy across SkySat 0.5 mpp (PlanetLabs 2024b) and Dove 3 mpp (PlanetLabs 2024a), with minor degradation on NASA LandSat (NASA 2024); multi-resolution training achieves $AP > 77$ across all resolutions. Additional ablations in the Appendix validate robustness to resolution, degradation, and density variations, achieving IoU of 0.96 on INRIA benchmark GT.

4.2 Dense Object Counting Results

In Fig. 8b, we report a counting accuracy comparison to our baselines. The *Acc.* metric is defined as $1 - MAE$ (normalized) as a percentage, where MAE is the mean absolute error. Our method exceeds the performance of all baselines in all datasets. On the Brazil dataset, our model achieves a *Acc.* of 93.6% compared to the next best baseline (LOGCAN++) of 87.3%. On the Mexico dataset, our method achieves 88.2% compared to the next best baseline (MCAC) of 84.8%, which is a counting-specific model.

4.3 Qualitative Study

In Fig. 7, we present qualitative results of our method and three best-performing baselines across five datasets, where our predictions closely match GT. Typical failure cases include BoundaryFormer and LOGCAN++ merging multiple buildings into one (Fig. 7), and a recent model, namely

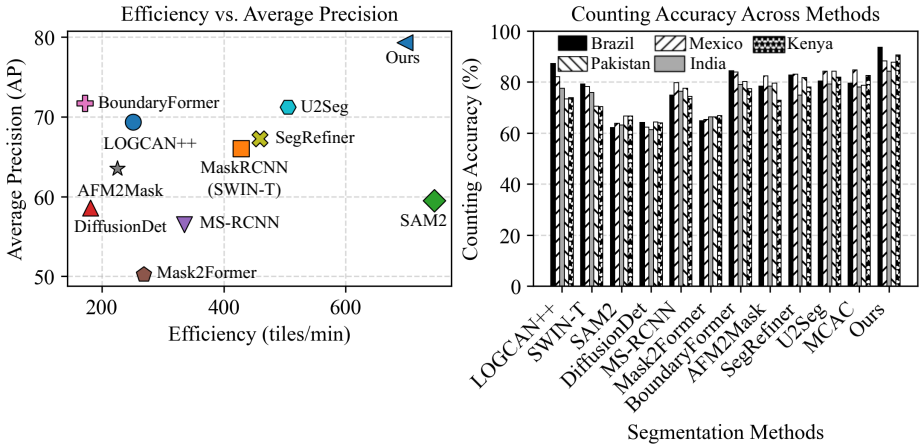

Figure 8: Performance and Efficiency Comparison. (L) AP vs. Efficiency, (R) Counting Acc. vs. Baselines.

Methods	Brazil			Mexico			India			Pakistan			Kenya		
	AP↑	AP_{50}	AP_{75}	AP↑	AP_{50}	AP_{75}	AP↑	AP_{50}	AP_{75}	AP↑	AP_{50}	AP_{75}	AP↑	AP_{50}	AP_{75}
LOGCAN++	71.2	72.6	70.8	69.0	75.1	67.4	68.2	69.2	64.7	68.6	73.4	63.1	69.6	71.8	67.9
MaskRCNN (Swin-T)	67.0	68.9	68.9	69.4	69.9	64.7	65.7	66.6	65.5	64.2	66.3	62.4	63.9	67.7	61.5
SAM2	61.9	64.2	59.4	58.6	58.7	56.8	59.5	63.1	58.6	56.2	57.6	54.3	61.3	61.5	57.4
DiffusionDet	59.7	59.9	55.8	59.9	65.6	58.6	58.7	59.2	53.6	56.3	63.2	54.3	58.4	59.3	54.7
MS-RCNN (ResNet-101)	59.0	59.4	57.6	55.8	60.8	54.2	58.4	59.2	55.9	51.6	58.8	50.4	57.6	58.4	55.3
Mask2Former	51.7	54.9	51.1	50.1	51.5	49.2	51.1	54.1	49.7	47.9	48.9	45.8	50.4	52.4	50.1
BoundaryFormer	74.7	75.4	73.7	72.9	77.0	69.1	71.0	72.3	70.2	68.9	71.4	66.9	70.9	72.4	68.1
AFM2Mask	69.2	70.1	65.3	62.4	64.1	60.8	59.7	64.2	55.9	61.4	65.8	59.4	64.9	68.1	62.3
SegRefiner	68.5	69.4	66.8	66.2	68.5	64.2	64.8	68.1	63.8	68.2	69.9	65.8	68.4	70.1	65.1
U2Seg	74.3	75.8	72.8	70.3	73.8	68.9	70.2	72.5	68.4	69.5	72.3	68.1	71.7	72.9	69.1
Ours	**82.1**	**83.9**	**80.2**	**79.1**	**78.8**	**77.0**	**80.4**	**80.9**	**77.3**	**77.2**	**79.7**	**74.6**	**79.9**	**82.9**	**75.7**

Table 1: Comparison of to SOTA Approaches. Comparisons of segmentation performance to prior state-of-the-art instance segmentation baselines on images with 300-400 building roofs (in a single image).

Tested on→ Trained on↓	Brazil	Mexico	India	Pakistan	Kenya
Brazil	82.6	72.5	79.5	74.8	79.1
Mexico	77.4	82.4	72.8	73.4	77.1
Combined	82.1	79.1	80.4	77.2	79.9

Table 2: Crossvalidated Quantitative Results (AP). Segmentation performance comparisons of cross-validated training and testing domain data.

Features	AP	AP_{50}	AP_{75}	Acc.
All Features (no ablation)	82.1	83.7	79.4	93.6
All − Overlap	78.8	79.5	77.2	88.2
All − Parallelism	70.9	74.4	69.2	84.3
All − Consistency	71.7	75.2	68.8	79.5
All − Patch	55.2	57.6	53.7	65.8
All − Distance	65.5	68.9	63.9	64.2

Table 3: Feature Ablations. Performance on Brazil and Mexico combined dataset by ablating each feature.

U2Seg, omitting notable buildings compared to ours. Additional comparisons are in the Appendix.

4.4 Efficiency Gains & Ablations

Our method achieves over 20% higher AP than SAM2 while being 38% faster than the next-best accurate model (BoundaryFormer) (Fig. 8). This efficiency stems from our graph-based refinement network (Fig. 5), which enables sparse parallel propagation through adjacency matrices rather than dense pixel convolutions used in SOTA models. Tab. 3 shows ablation results on the Brazil dataset, sorted by descending count accuracy, where adjacency distance ('Distance') most impacts AP and count accuracy, while overlap tolerance has minimal effect.

5 Conclusion

In this work, we have introduced a novel approach to accurately and efficiently segment and count dense informal urban settlements, yielding a performance superior to multiple current baselines. In future work, we aim to extend to other challenging high-density and abutting objects, e.g., trees and animals, where counting and segmentation are challenging.

Ethical Statement

This work involves the analysis of satellite imagery of urban areas for segmenting informal dense settlements. None of the satellite data contained personally identifiable information (PII). The released datasets are vector shapefile datasets and therefore also contain no PII, enabling researchers to use our work to perform further research. This work aims to serve as a means to extract critical information and analytics to improve urban planning and resource allocation in underserved communities in unmapped informal settlements in developing regions.

Acknowledgements

This research is partially funded by NSF Grants 2107096 and 2411273, and an Adobe Research Grant Gift.

References

Castagno, J.; and Atkins, E. 2018. Roof Shape Classification from LiDAR and Satellite Image Data Fusion Using Supervised Learning. *Sensors (Basel)*, 18(11): 3960.

Chen, L.; Wu, Y.; and Merhof, D. 2022. Instance Segmentation of Dense and Overlapping Objects via Layering. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, 400. BMVA Press.

Cheng, B.; Misra, I.; Schwing, A. G.; Kirillov, A.; and Girdhar, R. 2022. Masked-attention Mask Transformer for Universal Image Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Contributors, O. 2004. OpenStreetMap. Accessed: 2024-11-08.

Firoze, A.; Wingren, C.; Yeh, R. A.; Benes, B.; and Aliaga, D. G. 2023. Tree Instance Segmentation with Temporal Contour Graph. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2193–2202.

for Humanity, H. 2023. The world's largest slums: Dharavi, kibera, khayelitsha & neza. Accessed: 2025-01-08.

Fu, J.; Liu, J.; Wang, Y.; Li, L.; Zhang, Y.; and Zhang, W. 2019. Dual Attention Network for Scene Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3146–3154. IEEE.

Gamal, A.; et al. 2020. Automatic LIDAR Building Segmentation Based on DGCNN and Euclidean Clustering. *Journal of Big Data*, 7(1): 1–19.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask R-CNN. *IEEE International Conference on Computer Vision (ICCV)*, 2980–2988.

He, L.; and Aliaga, D. 2023. GlobalMapper: Arbitrary-Shaped Urban Layout Generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 454–464.

He, L.; Shan, J.; and Aliaga, D. 2023. Generative Building Feature Estimation From Satellite Images. *IEEE Transactions on Geoscience and Remote Sensing*, 61: 1–13.

Hobley, M. A.; and Prisacariu, V. A. 2024. ABC Easy as 123: A Blind Counter for Exemplar-Free Multi-Class Class-agnostic Counting. *Proceedings of the European Conference on Computer Vision*.

Hu, W.; Liu, B.; Gomes, J.; Zitnik, M.; Liang, P.; Pande, V.; and Leskovec, J. 2020. Strategies for Pre-training Graph Neural Networks. In *International Conference on Learning Representations*.

Hu, Z.; Gu, Y.; Wang, C.; Xu, W.; Yang, S.; Wu, C.; and Zhang, L. 2021. CARAFE++: Content-Aware ReAssembly for Image Segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 1–10. IEEE.

Huang, Z.; Huang, L.; Gong, Y.; Huang, C.; and Wang, X. 2019. Mask Scoring R-CNN. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lazarow, J.; Xu, W.; and Tu, Z. 2022. Instance Segmentation With Mask-Supervised Polygonal Boundary Transformers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Li, Q.; Mou, L.; Hua, Y.; Shi, Y.; and Zhu, X. X. 2022. Building Footprint Generation Through Convolutional Neural Networks With Attraction Field Representation. *IEEE Transactions on Geoscience and Remote Sensing*, 60: 1–17.

Lian, D.; Li, J.; Zheng, J.; Luo, W.; and Gao, S. 2019. Density Map Regression Guided Detection Network for RGB-D Crowd Counting and Localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liu, L.; Chen, J.; Wu, H.; Li, G.; Li, C.; and Lin, L. 2021a. Cross-Modal Collaborative Representation Learning and a Large-Scale RGBT Benchmark for Crowd Counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liu, S.; Ren, T.; and Zhao, Y. 2023. DiffusionDet: Diffusion Model for Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6903–6912.

Liu, Z.; Huang, G.; Zhang, Z.; Wang, M.; Wu, D.; Lin, L.; and Guo, B. 2021b. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 10012–10022. IEEE.

Ma, X.; Lian, R.; Wu, Z.; Guo, H.; Yang, F.; Ma, M.; Wu, S.; Du, Z.; Zhang, W.; and Song, S. 2025. LOGCAN++: Adaptive Local-Global Class-Aware Network For Semantic Segmentation of Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing*.

Maggiori, E.; Tarabalka, Y.; Charpiat, G.; and Alliez, P. 2017. Can Semantic Labeling Methods Generalize to Any City? The Inria Aerial Image Labeling Benchmark. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 3226–3229. Fort Worth, United States.

Microsoft. 2018. Microsoft Building Footprints. Accessed: 2025-01-08.

NASA. 2024. Landsat Program. https://landsat.gsfc.nasa.gov/. Accessed: 2025-01-12.

Niu, D.; Wang, X.; Han, X.; Lian, L.; Herzig, R.; and Darrell, T. 2024. Unsupervised Universal Image Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

PlanetLabs. 2024a. Dove-R Satellite. https://www.planet.com/products/satellites/dove/. Accessed: 2024-11-12.

PlanetLabs. 2024b. Planet Skysat Satellite Imagery. Accessed: 2024-11-08.

Ravi, N.; Gabeur, V.; Hu, Y.-T.; Hu, R.; Ryali, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; Mintun, E.; Pan, J.; Alwala, K. V.; Carion, N.; Wu, C.-Y.; Girshick, R.; Dollár, P.; and Feichtenhofer, C. 2025. SAM 2: Segment Anything in Images and Videos. In *The Thirteenth International Conference on Learning Representations (ICLR)*.

Ren, S.; He, K.; Girshick, R. B.; and Sun, J. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39: 1137–1149.

Siddiqui, M. I.; Sheikh, M. U.; Abid, H.; and Khan, M. H. 2024. PerSense: Personalized Instance Segmentation in Dense Images. *arXiv*.

Steadman, P. 2006. Why are most buildings rectangular? *University College London*.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. FCOS: Fully Convolutional One-Stage Object Detection. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 9626–9635.

United Nations Human Settlements Programme. 2018. SDG Indicator 11.1.1: Proportion of Urban Population Living in Slums. *UNSD SDG Indicators*. Accessed: 2025-01-08.

United Nations Human Settlements Programme. 2024. Share of the Urban Population Living in Slums. Processed by Our World in Data.

Xiao, A.; Huang, J.; Guan, D.; Zhan, F.; and Lu, S. 2022. Transfer Learning from Synthetic to Real LiDAR Point Cloud for Semantic Segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 2795–2803.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations (ICLR)*.

Xu, L.; Xiangli, Y.; Rao, A.; Zhao, N.; Dai, B.; Liu, Z.; and Lin, D. 2021. Block-Planner: City Block Generation with Vectorized Graph Representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 5077–5086.

Yang, B.; Yang, T.; and Gao, M. 2023. Techniques and Challenges of Image Segmentation: A Review. *Electronics*, 12(5): 1199.

Yang, G.; Xue, F.; Zhang, Q.; Xie, K.; Fu, C.-W.; and Huang, H. 2023. UrbanBIS: a Large-scale Benchmark for Fine-grained Urban Building Instance Segmentation. In *ACM SIGGRAPH 2023 Conference Proceedings*, SIGGRAPH '23. New York, NY, USA: Association for Computing Machinery. ISBN 9798400701597.

Yu, X.; Shi, D.; Wei, X.; Ren, Y.; Ye, T.; and Tan, W. 2022. SOIT: Segmenting Objects with Instance-Aware Transformers. In *Proceedings of the Thirty-Sixth AAAI Conference on Artificial Intelligence*, 3188–3196.

Zeng, H.; Benes, B.; Aliaga, D. G.; and Sitthi-Amorn, P. 2018. Neural Procedural Reconstruction for Residential Buildings. In *ECCV*.

Zhang, X.; Liu, Y.; Lin, Y.; Liao, Q.; and Li, Y. 2024. UV-SAM: Adapting Segment Anything Model for Urban Village Identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 22520–22528.

Zhang, Y.; et al. 2021. Deep Learning Based 3D Segmentation: A Survey. *arXiv*.

Zhang, Y.; et al. 2022. Semantic Urban Mesh Segmentation Based on Aerial Oblique Images and LiDAR. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2022: 485–492.

Zhao, H.; Li, J.; Shen, J.; Wang, J.; and Huang, T. S. 2018. PSANet: Point-wise Spatial Attention Network for Scene Parsing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 267–283. Springer.

Zhao, H.; Shi, J.; Qi, X.; Wang, J.; and Huang, T. S. 2017. Pyramid Scene Parsing Network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6230–6239. IEEE.