PolyAttractNet: Graph-Based Polygonal Segmentation of Building Footprints Using Attraction Field Maps

Muhammad Kamran, Mohammad Moein Sheikholeslami, Gunho Sohn

Department of Earth and Space Science and Engineering, York University, Canada - (mkamran9, mmoein, gsohn)@yorku.ca

Keywords: Instance segmentation, Satellite images, GCN, Attraction Field Maps, Regularized Boundaries.

Abstract

Since the launch of Landsat-1 in 1972, Earth observation satellites have undergone significant advancements, enabling the collection of vast amounts of high-resolution imagery. These satellites continuously provide critical data for monitoring urban expansion, infrastructure development, and disaster response. In recent years, the number of remote sensing satellites in orbit has increased substantially, generating extensive visual datasets essential for precise spatial mapping across civil, public, and military applications. One of the key challenges in utilizing satellite imagery is the automated reconstruction of building footprints, which demands high precision to account for variations in architectural styles. Traditional methods rely on manual or semi-automated approaches, which are often time-consuming and prone to inaccuracies. To address these limitations, this paper introduces PolyAttractNet, a novel deep learning framework designed to improve building boundary delineation in satellite imagery. Our approach incorporates Attraction Field Maps (AFMs) within a Graph Neural Network (GNN) framework, combined with an enhanced Mask R-CNN backbone. The proposed architecture effectively detects building instances from a single satellite image while minimizing boundary noise by embedding geometric regularity and integrating multi-scale, multi-resolution, and boundary-preserving mask features. AFMs play a crucial role in refining boundary precision by guiding feature extraction toward geometric consistency. As a result, our model achieves a 9.6% improvement in Average Precision (AP) and a 5% increase in Average Recall (AR) compared to the baseline, demonstrating its effectiveness in producing more accurate and regularized building footprints.

1. Introduction

Buildings play a significant role in shaping cities, serving as the backbone of urban infrastructure. Rapid urbanization requires precise monitoring and analysis of urban environments, with remote sensing offering critical data for managing urban growth. Satellite images have played a pivotal role in generating digital maps for Geographic Information Systems (GIS), with building footprint information serving as a critical asset for urban planning, smart city development, and other related fields. Moreover, building footprints with well-regularized boundaries can be represented as vectorized polygons, offering significant advantages in terms of transferability across various GIS platforms, thereby enabling a broad range of applications.

Despite the widespread availability and accessibility of satellite imagery, there remains a persistent demand for higher-quality building footprint data. This demand has yet to be fully met due to several key challenges. First, the creation of highly precise building footprints on GIS maps often requires manual or semi-automated processes, which are both labor-intensive and time-consuming. Additionally, the vast diversity in building roof designs presents further obstacles to large-scale, automated footprint extraction. The geometric potential inherent in satellite imagery has also not been completely utilized.

Traditional methods often struggle with complex building geometries, occlusions, and variations in scale and orientation in aerial and satellite images. In the last decade, deep learning has driven significant advancements in artificial intelligence (AI). This remarkable success of deep learning has prompted exploration into its potential applications within the remote sensing domain. Several deep learning-based approaches (Hu et al., 2023, Xu et al., 2023, Sheikholeslami et al., 2024a) for polygonal building segmentation have emerged. However, they often present challenges such as high training complexity, computational intensity, and issues like inconsistent projections or missing corner points. Motivated by the challenges of regularized building footprint extraction and enhancing our previous baseline model, R-PolyGCN (Zhao et al., 2020), this study will present our novel model, PolyAttractNet. This network improves the baseline model by inculcating the orientation information acquired from the attraction field maps (Xue et al., 2019) and better initializing graphs based on corner prediction. This enhances the feature map for generating the initial polygon, resulting in more regularized building footprints.

2. Literature Review

Deep learning is a specialized subset of machine learning distinguished by its multi-layer neural architectures, which enable it to capture hierarchical data representations at varying levels of abstraction (LeCun et al., 2015). Unlike traditional machine learning approaches that depend on manually crafted features, deep neural networks autonomously learn complex, structured features through a sequence of linear and non-linear transformations. In supervised learning, these networks iteratively adjust their parameters by minimizing the discrepancy between predicted outputs and ground truth labels using an appropriate loss function, thereby improving model accuracy and robustness.

Deep learning-based object detection models are broadly classified into two-stage and one-stage architectures. Two-stage models follow a sequential detection process, with Region-based Convolutional Neural Networks (R-CNN) pioneering this paradigm. R-CNN (Girshick et al., 2014) generates around 2000 candidate regions, extracts features using a convolutional neural network, and subsequently classifies each region using a support vector machine (Cortes, 1995). Fast R-CNN (Girshick,

2015) optimizes this approach by processing the entire image with a single CNN pass while still relying on a separate region proposal step. Faster R-CNN (Ren, 2015) further streamlines the pipeline by integrating a Region Proposal Network (RPN) for an end-to-end learning framework. In contrast, one-stage object detectors such as YOLO (Redmon, 2016) and SSD (Liu et al., 2016) eliminate the need for explicit region proposal generation by directly predicting bounding boxes across densely sampled locations in the image, leading to faster inference. Recent advancements (Zhou et al., 2021) have further improved detection accuracy by leveraging keypoint-based object localization strategies.

Instance segmentation, a crucial task for delineating individual objects within a scene, often involves two primary approaches: semantic segmentation followed by object grouping or direct instance-level segmentation. Early methods, such as SharpMask (Pinheiro et al., 2016), followed the former approach, whereas Mask R-CNN (He et al., 2017) reversed the order by first detecting objects and then segmenting them, leading to more precise contours. Additionally, U-Net (Maggiori et al., 2016) has proven highly effective for building extraction tasks, showcasing the potential of deep learning in instance-level segmentation.

One key challenge in segmentation tasks, particularly for structured objects such as buildings, is boundary regularization. Traditional approaches, including Binary Space Partitioning and Minimum Description Length (Jung and Sohn, 2019), applied geometric heuristics to refine boundaries in point cloud data. More recent deep learning-based methods integrate boundary regularization directly into neural networks. For instance, DSAC (Marcos et al., 2018) incorporates active contour models within CNNs to enhance boundary precision. PolyRNN (Castrejon et al., 2017) and its improved version, PolyRNN++ (Acuna et al., 2018), use recurrent neural networks (RNNs) to sequentially predict polygon vertices for semi-automated annotations. CurveGCN (Ling et al., 2019) introduces graph convolutional networks (GCNs) to generate polygonal representations that are more geometrically efficient. R-PolyGCN (Zhao et al., 2020) further refines this approach by integrating an object detection module to predict building corners in a single pass. However, CNN-GCN frameworks often suffer from redundant vertices due to fixed vertex counts. (Li et al., 2019) addressed this limitation by reframing corner detection as a segmentation task, followed by GCN-based vertex refinement.

Further advancements include PolyWorld (Zorzi et al., 2021), which introduces a permutation matrix to encode vertex connectivity for accurate polygon generation. CornerRegNet (Sheikholeslami et al., 2024a) and OriCornerNet (Sheikholeslami et al., 2024b) enhances R-PolyGCN by incorporating oriented corners as auxiliary representations, refining geometric consistency. Meanwhile, HiSup (Xu et al., 2023) integrates attraction field maps to achieve precise polygon mapping, though it requires post-processing to fully regularize building boundaries. These developments highlight the ongoing evolution of deep learning techniques in structured segmentation, particularly for applications in building footprint extraction and urban mapping.

3. Methodology

Traditional segmentation approaches that classify individual pixels often struggle to capture the geometric structure of objects, as their pixel-wise representation provides limited contextual information about shape. In contrast, graph-based models inherently preserve geometric properties by representing buildings as a network of vertices and edges. By leveraging a Graph Neural Network (GNN) for convolutional operations, we enable feature propagation across vertices, allowing the model to learn and maintain structural coherence more effectively.

Attraction Field Maps (AFMs) further contribute to boundary refinement by directing pixels toward their nearest edges. This vector-based guidance enforces smoother and more consistent delineations that align with building contours. By respecting the underlying geometry, AFMs help maintain the structural integrity of extracted building footprints, ultimately leading to more accurate and regularized polygonal representations.

To harness the geometric advantages, we introduce PolyAttractNet, an end-to-end architecture comprising two essential components: a backbone network and a graph convolutional network. By integrating Attraction Field Maps in the backbone, PolyAttractNet enhances edge detection and achieves more precise building footprint extraction by implicitly learning polygonal shapes, resulting in accurate and complete representations.

3.1 Backbone Network

The backbone network in our model is optimized for feature encoding, building object detection, and localization. We utilize a Residual Network combined with a Feature Pyramid Network to extract deep, multi-scale features critical for accurately detecting objects of varying sizes within large satellite images encompassing diverse building structures. We employ a two-stage object detection model to detect and localize buildings, incorporating a Region Proposal Network (RPN) and a localization layer with bounding box regression and classification. The RPN generates initial bounding box proposals using predefined anchor boxes on extracted features, which are then applied to the feature maps for cropping. Based on box proposal sizes, the corresponding feature maps are selected for cropping, and the cropped features are passed through a RoI pooling layer, yielding standardized RoI features. These features are then processed to refine bounding box coordinates and class scores. The final bounding boxes and the multi-scale feature maps are fed into a RoI-Align layer, generating highly localized RoI features.

We also employ AFMs, which provide explicit pixel-wise guidance toward boundaries, enabling precise localization of building edges by directing the model's focus to the most accurate boundary, even in regions of low contrast or where multiple boundaries are in close proximity. The attraction field maps and their application in our network are explained below.

3.2 Attraction Field Maps

Attraction Field Maps (AFMs) provide a structured representation of spatial attraction forces, guiding pixels toward their nearest object boundaries. Unlike traditional edge detection methods, AFMs enhance boundary localization by capturing geometric relationships and structural consistency, making them particularly useful in complex scenarios with occlusions, weak gradients, or noisy environments.

In our network, AFMs define regions of interest by modeling buildings as attractive objects, aiding in shape reconstruction and precise boundary delineation. By generating a vector field

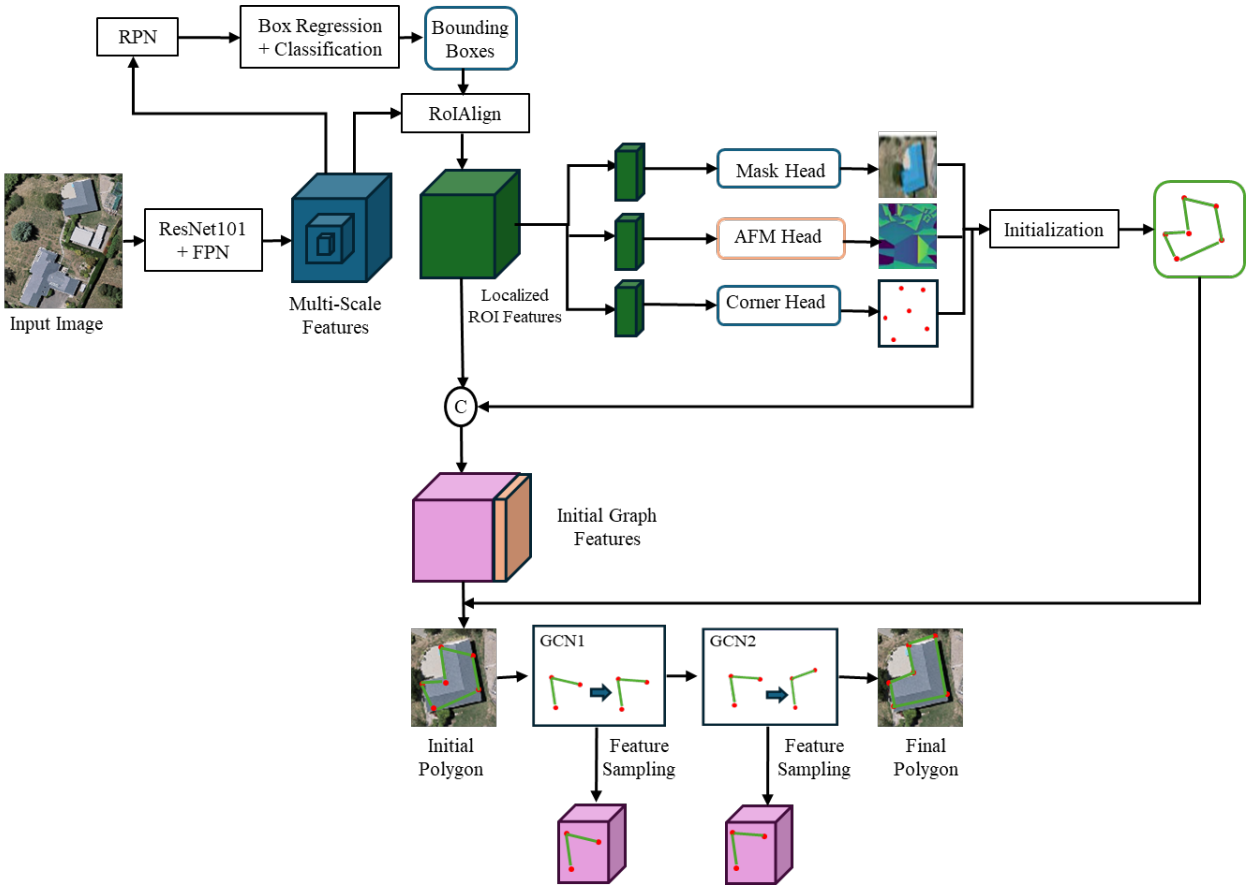

Figure 1. Network Architecture

where each pixel is directed toward the closest boundary point, AFMs refine feature representation and improve object-based image analysis. The Euclidean distance between a pixel p and its nearest boundary point b determines the attraction vector, which can be normalized to retain directional consistency while discarding magnitude.

$$D(P) = \min_{b \in \text{boundary}} \|p - b\|^2 \tag{1}$$

$$a(p) = \frac{(x_b - x, y_b - y)}{\|(x_b - x, y_b - y)\|^2} \tag{2}$$

AFMs are integrated into deep learning frameworks by concatenating them with feature maps at multiple layers, enabling the network to leverage both semantic and geometric cues. Additionally, AFMs can be incorporated into the loss function to encourage alignment between predicted and ground truth attraction vectors, improving model robustness. Similarly, in our network, AFMs provide explicit pixel-wise guidance toward boundaries, enabling precise localization of building edges by directing the model’s focus to the most accurate boundary, even in regions of low contrast or where multiple boundaries are in close proximity. Not restricted to specific building shapes, AFMs allow our model to generalize effectively across diverse structures.

3.3 Graph Convolution Network

In the second stage, we employ a multi-step architecture for coarse-to-fine polygon prediction, progressively refining vertex positions using a Graph Convolution Network (GCN). The process begins by generating an initial polygon based on the predicted masks and corners, with vertices uniformly resampled to match the ground truth vertex count.

GCNs require an input graph, which can either be fixed, as seen with the arbitrary circle used in R-PolyGCN, or dynamically based on prior predictions, such as masks and corners. The latter approach optimally leverages available information, initializing the GCN with a more accurate estimation of the building shape and allowing it to focus on fine-tuning details. Our initialization module constructs this initial polygon by using predicted masks and corners as input. First, the contour of the mask is extracted via the marching squares algorithm (Lorensen et al., 1998) and simplified with the procedure by Douglas Peuker Algorithm(Ramer et al., 1972). Next, a polygon is formed by matching predicted corners to their closest points on the contour, recovering any potentially missing corners from the contour points to ensure a complete shape. This results in a more geometrically accurate initial graph. Finally, 16 vertices are uniformly resampled from the graph to match the vertex count in the ground truth.

The initial graph features are processed through a GCN to compute vertex offsets, which adjust the vertex positions and produce updated graph features. These refined vertices are then fed into a subsequent GCN stage, which predicts additional offsets for further refinement. This iterative approach continues over multiple steps, progressively improving the accuracy of vertex positions and polygon predictions. For this study, we implement a three-step GCN pipeline, where each step incorporates a multi-layer GCN to enable comprehensive feature extraction

and vertex refinement.

3.4 Loss Design

Our network loss function is structured into two primary branches. The first branch focuses on the backbone network, which includes Region Proposal Network (RPN) loss, bounding box regression loss, classification loss, and localization loss. The second branch addresses polygon vertex prediction and geometric regularization, ensuring structural accuracy. Additionally, we define training strategies to optimize the performance of these loss functions.

3.4.1 Backbone Network Losses: Object detection losses in the backbone network are computed in two stages: RPN training and box regression with classification. Both stages employ a multi-task learning approach, combining box regression loss and classification loss to improve detection accuracy.

Box Regression Loss: Instead of directly predicting bounding box coordinates, our model estimates box deltas, which define the transformation required to refine anchor boxes into more precise proposals. These deltas are computed as.

$$t_x = (x - x_a)w_a, t_y = (y - y_a)h_a \tag{3}$$

$$t_w = log(w/w_a), t_h = log(h/h_a) \tag{4}$$

$$t_x^* = (x^* - x_a)w_a, t_y^* = (y^* - y_a)h_a \tag{5}$$

$$t_w^* = log(w^*/w_a), t_h^* = log(h^*/h_a) \tag{6}$$

where x, y = bounding box center
w, h = bounding box width , height
x, x_a, x^* = predicted box, anchor box, actual box

To evaluate box deltas, we use the Smooth L1 loss, which balances sensitivity to minor errors and robustness against outliers.

$$L_{\text{reg}}(t, t^*) = \sum_{i \in \{x,y,w,h\}} \text{Smooth}_{L1}(t_i - t_i^*) \tag{7}$$

$$Smooth_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1, \\ |x| - 0.5 & \text{otherwise.} \end{cases} \tag{8}$$

Classification Loss: To assess the confidence of the predicted class labels, we employ Binary Cross-Entropy (BCE) loss, which measures how well the model distinguishes between building and non-building classes.

$$L_{cls}(p(y)) = -(ylog(p(y)) + (1 - y)log(1p(y))) \tag{9}$$

where y = actual class label (0 or 1)
$p(y)$ = predicted probability

RPN Loss: During RPN training, anchor boxes are assigned objectness scores to classify them as positive or negative. The RPN loss function is formulated as:

$$L_{\text{rpn}}(p^{\text{obj}}, t^{\text{rpn}}) = \frac{1}{N_{\text{cls}}} \sum_{i=1}^{N_{\text{cls}}} L_{\text{cls}}(p_i^{\text{obj}}(y)) + \frac{1}{N_{\text{box}}} \sum_{i=1}^{N_{\text{box}}} L_{\text{reg}}(t_i^{\text{rpn}}, t_i^*) \tag{10}$$

where N_{cls} = number of boxes after NMS
N_{box} = positive boxes

Localization Loss: To refine the final building bounding boxes, we compute localization loss, combining classification and regression.

$$L_{\text{loc}}(p^{\text{class}}, t^{\text{loc}}) = \frac{1}{N_{\text{cls}}} \sum_{i=1}^{N_{\text{cls}}} L_{\text{cls}}(p_i^{\text{class}}(y)) + \frac{1}{N_{\text{box}}} \sum_{i=1}^{N_{\text{box}}} L_{\text{reg}}(t_i^{\text{loc}}, t_i^*) \tag{11}$$

where p_{class} = predicted class probability
t_{loc} = refined bounding box deltas

Attraction Field Map (AFM) Loss: To improve boundary regularization, we introduce an Attraction Field Map (AFM) loss, which minimizes the difference between predicted attraction vectors and ground truth vectors.

$$L_{\text{AFM}} = \frac{1}{N} \sum_{p \in P} \|\tilde{v} - v_p\|^2 \tag{12}$$

where $\tilde{v}$ = predicted attraction vector
v_p = ground truth attraction vector
N = number of pixels in the image

Thus, the total backbone loss is:

$$L_{backbone} = L_{rpn} + L_{loc} + L_{AFM} \tag{13}$$

3.4.2 Polygon-Based Losses: In addition to backbone losses, our second branch focuses on polygon localization and geometric regularization.

Polygon Localization Loss: We define a polygon as a sequence of N-ordered vertices $p = v_i | i = 1, 2, ..., N$. Given K predicted polygons, we measure their alignment with ground truth using the geometric L1 distance.

$$L_1(p^{\text{pre}}, p^{\text{gt}}) = \sum_{i=0}^{N} \left(|x_i^{\text{pre}} - x_i^{\text{gt}}| + |y_i^{\text{pre}} - y_i^{\text{gt}}| \right) \tag{14}$$

Since the starting vertices of the predicted and ground truth polygons may not align, we perform vertex correspondence matching by iterating over all N possible alignments. The final polygon localization loss is:

$$L_{\text{poly}}(p^{\text{pre}}, p^{\text{gt}}) = \frac{1}{K}\sum_{k=1}^{K} \min_{j\in\{0,1,\ldots,N-1\}} \left(L_1(p^{\text{pre}}_{k+j}, p^{\text{gt}}_{k})\right) \quad (15)$$

Orthogonality Loss: To enforce geometric regularity, we introduce an orthogonality loss, which encourages building boundaries to form right angles. This loss penalizes deviations from 0°, 90°, 180°, and 270° angles.

$$L_{ortho} = \frac{1}{N}\sum_{j=1}^{N} L(P_j) \quad (16)$$

$$L(P) = \frac{1}{n}\sum_{i=1}^{n} \min_{\theta_{peak}} |\theta_i - \theta_{peak}| \quad (17)$$

where θ = internal angles of the polygon

By enforcing orthogonality, we ensure that building footprints maintain realistic structural constraints, improving both visual quality and detection accuracy.

3.4.3 Total Loss Function: By integrating polygon localization loss and orthogonality loss into the backbone losses, we establish a robust loss formulation:

$$L_{total} = L_{backbone} + \lambda\frac{1}{N}L_{poly} + \lambda L_{ortho} \quad (18)$$

where λ = Weighting Fator for the losses

The proposed loss function ensures precise object detection while refining polygon-based representations of buildings. AFM loss improves boundary accuracy, polygon localization loss enhances vertex alignment, and orthogonality loss enforces geometric regularity. These components collectively enhance model performance in complex urban environments, ensuring accurate and structured building footprint extraction.

4. Experiments and Results

4.1 Dataset

For training and evaluating our network, we employed the WHU Building Dataset (Ji et al., 2018), which contains approximately 220,000 annotated building footprints extracted from high-resolution aerial imagery. Each image measures 300 × 300 pixels with a fine spatial resolution of 0.075 meters per pixel, covering a total area of 450 square kilometers in Christchurch, New Zealand. This dataset was compiled using remote sensing imagery from various global urban regions, captured by advanced Earth observation satellites such as QuickBird, the WorldView series, IKONOS, and ZY-3. The diversity and scale of the dataset make it a valuable benchmark for assessing and refining building footprint extraction models, ensuring robust performance across different geographic and imaging conditions.

4.2 Performance Metrics

To assess the performance of building extraction models, we utilize Average Precision (AP), Average Recall (AR), and PoLiS (Polygonal Line String Similarity). These metrics provide insights into how accurately the model captures the shape and extent of buildings. Precision measures the proportion of correctly identified building pixels among all pixels classified as buildings. A high precision score indicates fewer false positives, which is crucial in applications where incorrect detections can be problematic. It is defined as:

$$Precision = \frac{TP}{TP + FP} \quad (19)$$

where TP = True Positives (correctly detected pixels)
FP = False Positives (incorrectly classified pixels)

Recall evaluates the model's ability to detect all actual building pixels, reflecting how many relevant structures are correctly identified. A high recall score suggests fewer false negatives, which is essential in applications where missing buildings are critical. It is calculated as:

$$Recall = \frac{TP}{TP + FN} \quad (20)$$

where FN = False Negatives (missed building pixels)

PoLiS (Polygonal Line String Similarity) is a vector-based metric designed to assess the geometric accuracy of predicted building polygons. Unlike pixel-based metrics, PoLiS evaluates boundary alignment and structural similarity by measuring the shortest distance from each vertex in the predicted polygon to the closest point on the ground truth polygon and vice versa. This bidirectional comparison ensures a thorough assessment of shape consistency, capturing discrepancies in spatial arrangement and structural detail.

$$\text{PoLiS}(P,Q) = \frac{1}{2N_P}\sum_{p_j\in P}\min_{q\in Q}\|p_j - q\| + \frac{1}{2N_Q}\sum_{q_k\in Q}\min_{p\in P}\|q_k - p\| \quad (21)$$

By incorporating PoLiS alongside precision and recall, we achieve a more comprehensive evaluation of building extraction models, focusing on both detection accuracy and geometric fidelity.

4.3 Results

Training on the WHU dataset is performed for 35 epochs on a single NVIDIA GeForce RTX 2080 GPU with a batch size of 1. Our neural network models, training, and inference codes were implemented with Python 3.9 on Pytorch 1.10.1

We assess the results through qualitative analysis using both raster-based and vector-based metrics to evaluate model performance. These evaluations highlight the effectiveness of our

approach in overcoming key challenges in building extraction, such as automating the footprint extraction process, handling diverse roof appearances, achieving a balance between recognition accuracy and precise localization, distinguishing closely spaced structures, detecting buildings of varying sizes, and accurately preserving the geometric integrity of building polygons. The robustness of these metrics demonstrates the adaptability and reliability of our method across diverse and complex urban environments.

Networks	AP(%)	AR(%)	PoLiS ↓
Baseline	45.7	57.5	2.58
+Orthogonality Loss	46.1	58.3	1.96
+Augmented Features	53.0	61.4	1.90
+Attraction Field Maps	**55.3**	**62.5**	**1.46**

Table 1. Ablation Study of PolyAttractNet on the WHU Dataset (Lower PoLiS Values Indicate Better Performance)

By integrating Orthogonality, Feature Augmentation, and Attraction Field Maps into the baseline, our model achieves significant improvements, with a 9.6% increase in AP and a 5% increase in AR, as shown in Table 1. Figure 2 illustrates that PolyAttractNet produces predictions that closely match the ground-truth, accurately capturing buildings of various sizes and shapes.

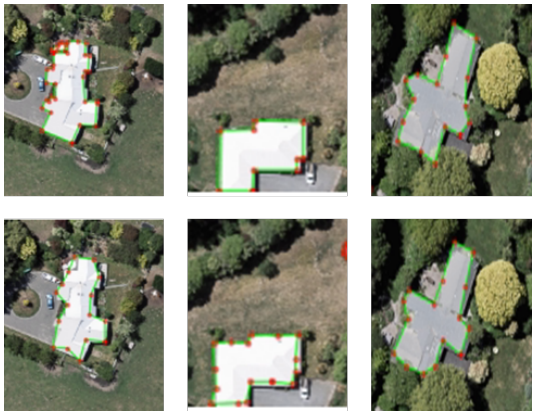

Figure 2. Comparison of Results: The top row shows the ground-truth building footprints, while the bottom row displays the predictions by PolyAttractNet..

5. Conclusion

In this study, we present a deep learning framework for automatic building footprint extraction from satellite imagery, incorporating enhanced boundary regularization. Our approach utilizes a backbone network for multi-scale feature encoding and object detection, generating well-localized Region of Interest (RoI) features. These features are further refined through orientation information from Attraction Field Maps (AFMs) and a Graph Convolutional Network (GCN) to reconstruct building footprints with greater geometric accuracy. By leveraging AFMs and GCNs, the framework enhances geometric learning and improves boundary precision. AFMs are particularly effective in capturing diverse building structures, improving the detectability of smaller buildings while ensuring consistent vector guidance for larger or more complex structures, even in occluded areas. Additionally, AFMs mitigate challenges related to overlapping building edges by maintaining distinct attraction fields, which helps preserve clear and well-separated boundaries. Experimental evaluations demonstrate that our network, PolyAttractNet outperforms both our baseline model and prior approaches, achieving higher accuracy and producing well-regularized building footprints. The results are on par with state-of-the-art methods, confirming the effectiveness of our approach. This advancement represents a crucial step toward fully automated, high-precision building extraction, reducing reliance on manual intervention. The proposed framework offers a scalable and robust solution for applications in urban planning, Geographic Information Systems (GIS) mapping, and spatial data analysis.

References

Acuna, D., Ling, H., Kar, A., Fidler, S., 2018. Efficient interactive annotation of segmentation datasets with polygon-rnn++. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 859–868.

Castrejon, L., Kundu, K., Urtasun, R., Fidler, S., 2017. Annotating object instances with a polygon-rnn. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5230–5238.

Cortes, C., 1995. Support-Vector Networks. *Machine Learning*.

Girshick, R., 2015. Fast r-cnn. *arXiv preprint arXiv:1504.08083*.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn. *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

Hu, Y., Wang, Z., Huang, Z., Liu, Y., 2023. PolyBuilding: Polygon transformer for building extraction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 199, 15–27.

Ji, S., Wei, S., Lu, M., 2018. Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on geoscience and remote sensing*, 57(1), 574–586.

Jung, J., Sohn, G., 2019. A line-based progressive refinement of 3D rooftop models using airborne LiDAR data with single view imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 149, 157–175.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *nature*, 521(7553), 436–444.

Li, Z., Wegner, J. D., Lucchi, A., 2019. Topological map extraction from overhead images. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1715–1724.

Ling, H., Gao, J., Kar, A., Chen, W., Fidler, S., 2019. Fast interactive object annotation with curve-gcn. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5257–5266.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A. C., 2016. Ssd: Single shot multibox detector. *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, Springer, 21–37.

Maggiori, E., Tarabalka, Y., Charpiat, G., Alliez, P., 2016. Convolutional neural networks for large-scale remote-sensing image classification. *IEEE Transactions on geoscience and remote sensing*, 55(2), 645–657.

Marcos, D., Tuia, D., Kellenberger, B., Zhang, L., Bai, M., Liao, R., Urtasun, R., 2018. Learning deep structured active contours end-to-end. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8877–8885.

Pinheiro, P. O., Lin, T.-Y., Collobert, R., Dollár, P., 2016. Learning to refine object segments. *Computer Vision–ECCV 2016: 14th European conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, Springer, 75–91.

Redmon, J., 2016. You only look once: Unified, real-time object detection. *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Ren, S., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*.

Sheikholeslami, M. M., Kamran, M., Wichmann, A., Sohn, G., 2024a. Cornerregnet: Building segmentation from overhead imagery using oriented corners in deep networks. *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*, 4642–4647.

Sheikholeslami, M. M., Kamran, M., Wichmann, A., Sohn, G., 2024b. Enhancing Polygonal Building Segmentation via Oriented Corners. *arXiv preprint arXiv:2407.12256*.

Xu, B., Xu, J., Xue, N., Xia, G.-S., 2023. HiSup: Accurate polygonal mapping of buildings in satellite imagery with hierarchical supervision. *ISPRS Journal of Photogrammetry and Remote Sensing*, 198, 284–296.

Xue, N., Bai, S., Wang, F., Xia, G.-S., Wu, T., Zhang, L., 2019. Learning attraction field representation for robust line segment detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1595–1603.

Zhao, K., Kamran, M., Sohn, G., 2020. Boundary Regularized Building Footprint Extraction from Satellite Images Using Deep Neural Networks. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 5, 617–624.

Zhou, D., Wang, G., He, G., Yin, R., Long, T., Zhang, Z., Chen, S., Luo, B., 2021. A large-scale mapping scheme for urban building from Gaofen-2 images using deep learning and hierarchical approach. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14, 11530–11545.

Zorzi, S., Bittner, K., Fraundorfer, F., 2021. Machine-learned regularization and polygonization of building segmentation masks. *2020 25th International Conference on Pattern Recognition (ICPR)*, IEEE, 3098–3105.