PolyFootNet: Extracting Polygonal Building Footprints in Off-Nadir Remote Sensing Images

Kai Li, *Student Member, IEEE*, Yupeng Deng, Jingbo Chen, *Member, IEEE*, Yu Meng*, Zhihao Xi, Junxian Ma, Chenhao Wang, Maolin Wang, Xiangyu Zhao

***Abstract*—Extracting polygonal building footprints from off-nadir imagery is crucial for diverse applications. Current deep-learning-based extraction approaches predominantly rely on semantic segmentation paradigms and post-processing algorithms, limiting their boundary precision and applicability. However, existing polygonal extraction methodologies are inherently designed for near-nadir imagery and fail under the geometric complexities introduced by off-nadir viewing angles. To address these challenges, this paper introduces Polygonal Footprint Network (PolyFootNet), a novel deep-learning framework that directly outputs polygonal building footprints without requiring external post-processing steps. PolyFootNet employs a High-Quality Mask Prompter to generate precise roof masks, which guide polygonal vertex extraction in a unified model pipeline. A key contribution of PolyFootNet is introducing the Self Offset Attention mechanism, grounded in Nadaraya-Watson regression, to effectively mitigate the accuracy discrepancy observed between low-rise and high-rise buildings. This approach allows low-rise building predictions to leverage angular corrections learned from high-rise building offsets, significantly enhancing overall extraction accuracy. Additionally, motivated by the inherent ambiguity of building footprint extraction tasks, we systematically investigate alternative extraction paradigms and demonstrate that a combined approach of building masks and offsets achieves superior polygonal footprint results. Extensive experiments validate PolyFootNet's effectiveness, illustrating its promising potential as a robust, generalizable, and precise polygonal building footprint extraction method from challenging off-nadir imagery. To facilitate further research, we will release pre-trained weights of our offset prediction module at https://github.com/likaiucas/PolyFootNet.**

***Index Terms*—Building footprint extraction, Building detection, Segment Anything Model (SAM), Off-nadir aerial image, Nadaraya-Watson regression, Oblique monocular images**

I. Introduction

BUILDING Footprint Extraction (BFE) in off-nadir images has been a research subject for over a decade, forming the foundation for critical tasks such as 3D building reconstruction and building change detection [1]–[5]. Off-nadir imagery is particularly attractive as it provides a more efficient and economical alternative to near-nadir images, enabling broader coverage with fewer acquisitions. Early approaches to BFE primarily consisted of two categories: geometric-feature-based algorithms and traditional machine learning algorithms [6]–[10]. Then, the rise of deep convolutional networks revolutionized BFE, leading to various deep learning approaches. Among these innovations, offset-based methods have emerged as a prominent solution in recent years, demonstrating remarkable performance in building feature extraction [11]–[18].

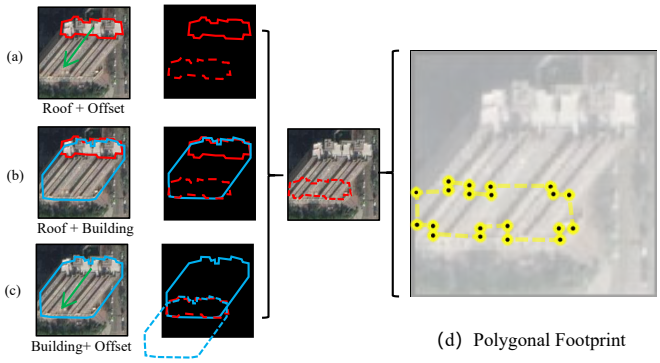


Fig. 1. Previous approaches primarily relied on Paradigm (a) for extracting building footprints. In this study, we explore the extraction of building footprints using task decomposition based on Paradigms (b) and (c). Additionally, this work achieves the first implementation of Paradigm (d), enabling the extraction of polygonal footprints by the model via decoding *offset tokens* and *vertex tokens*. Compared to Paradigms (a), (b), and (c), which depend on post-processing algorithms to extract final results, the proposed method eliminates this dependency.

However, former building footprint methods mainly focus on semantic segmentation following Fig.1 (a), which is often restricted in accurately delineating boundaries and exhibits limited generalizability, which can affect their real-world applicability. Meanwhile, to the best of our knowledge, existing polygonal BFE methods are designed for near-nadir scenarios [19], [20], relying on a crucial hypothesis that the projected polygonal positions of the visible roof and invisible footprint significantly overlap—a condition that no longer holds true in off-nadir imagery.

Results of the latest methods [12], [14], [15] followed Fig.1 (a), which can automatically extract footprints, failed to be applied as polygonal footprints because instance segmentation methods struggled to identify a single building footprint for each building since their processing methods,

* Yu Meng is the corresponding author. This research was funded by the National Key R&D Program of China under Grant number 2021YFB3900504.

Kai Li, Junxian Ma, and Chenhao Wang are with Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100101, China, and also with School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China.

Kai Li, Maolin Wang, and Xiangyu Zhao are also with Applied Machine Learning Lab, School of Data Science, City University of Hong Kong, Kowloon Tong, Hong Kong 999077.

Yupeng Deng, Jingbo Chen, Yu Meng, and Zhihao Xi are with Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100101, China

Region Proposal Network (RPN) and Non-Maximum Suppression (NMS) will assign more than one mask result in pieces for one building [16]. These repeatedly predicted footprints are hard to post-process with other image features since the footprints on off-nadir images are less visible, especially the tall buildings, compared with well-overlapped footprints with roofs in near-nadir images. Meanwhile, the performance gap between predicting bungalows and tall buildings was discovered, especially in the prediction of roof-to-footprint offsets. This made the predicted polygonal footprints of bungalows inaccurately express their footprints.

Therefore, beyond merely establishing a polygonal footprint extraction paradigm tailored for off-nadir imagery, two critical issues warrant further investigation: (1) mitigating the performance discrepancy observed between predictions for low-rise (bungalows) and high-rise buildings and (2) systematically exploring multiple plausible solutions inherent to the BFE problem, as illustrated by examples in Fig.1(b) and (c).

In this paper, we proposed PolyFootNet, which can extract polygonal building footprints from off-nadir images as Fig.1 (d). To the best of our knowledge, PolyFootNet is the first paradigm that achieves polygonal building footprints extraction under off-nadir scenarios. In addition to supporting automated extraction independently, its roof prompt input is the first approach that can be integrated with various existing roof extraction models.

Specifically, as shown in Fig.2 (a), PolyFootNet extracts polygonal building footprints by directly computing the footprint vertices in the coordinates. PolyFootNet is a model built on the Segment Anything Model (SAM) [21], which supports zero-shot segmentation. To avoid misrepresentation caused by repetitive predictions from the RPN, PolyFootNet employs a High-Quality Mask Prompter to generate roof masks as automatic prompts, subsequently enabling its decoder to extract and compute the polygonal footprints. Additionally, PolyFootNet is flexible enough to leverage pre-trained models from existing libraries for roof segmentation prompts or directly utilize human visual prompts to provide roof-to-footprint offsets for each roof, thus facilitating downstream applications and future research.

To bridge the performance gap among different buildings within the image, we specifically designed a Self Offset Attention (SOFA) mechanism, formulated using Nadaraya-Watson Regression [22]–[24], based on earlier observations [16] that low buildings exhibit accurate length predictions but poor angular predictions, whereas high-rise buildings demonstrate accurate angular predictions. From the experimental results, the proposed Self-Offset Attention (SOFA) mechanism enables low-rise buildings to perform self-correction by utilizing angle offsets learned from high-rise building predictions, significantly improving their overall polygon extraction accuracy.

On the other hand, motivated by the inherent ambiguity and multi-solution nature of extracting building footprints, we explored alternative solutions, as illustrated in Fig.1. Our experiments revealed that adopting a prediction scheme based on the combination of *building masks + offsets* effectively improves the accuracy of polygonal footprint extraction.

In summary, the contributions of this paper are as follows:

- We proposed PolyFootNet, the first polygonal footprint extraction network, demonstrating strong generalization capability.
- To bridge the performance gap between different buildings, we designed SOFA based on the Nadaraya-Watson regression. This module can improve the angle quality of low-rise building offsets.
- We explore the multi-solution of BFE and discover that *building mask + offset* maybe more suitable for extract building footprint masks.
- The experiments on three datasets, BONAI, OmniCity-View3 and Huizhou, demonstrate the practical effectiveness of PolyFootNet.

II. Related Work

PolyFootNet is a multi-task network built upon the architecture of the SAM [21], integrating tasks such as prompt segmentation, polygon extraction, and offset learning. The SOFA block is designed using the Nadaraya-Watson regression framework and an attention mechanism. This paper also explores the multiplicity of solutions for BFE by leveraging existing geometric relationships, geographical knowledge, and iterative methods to enhance performance.

In the following subsections, we will review related works in the application of SAM, polygonal building extraction, and building offset methods to highlight the novelty and contributions of PolyFootNet.

A. Segment Anything Model and its application

The SAM [21] is a foundational model for segmentation, supporting tasks using point, bounding box, and semantic prompts. SAM 2 [25] was introduced for promptable image and video segmentation, offering faster inference speeds and improved accuracy on both image and video tasks compared to the original SAM. Point to Prompt (P2P) [26], also based on SAM, transforms point supervision into fine visual prompts through a two-stage iterative refinement process. Additionally, GaussianVTON [27] employed a SAM-based model for post-editing view images after face refinement. SAM-HQ [28] enhanced the quality of SAM's image embeddings by incorporating deconvolution blocks, while OBM [16] introduced offset tokens and the ROAM structure to enable SAM to extract footprint masks.

In this paper, PolyFootNet builds upon the idea of SAM-HQ to generate high-quality semantic prompts for automatic extraction. Furthermore, PolyFootNet introduces a novel concept of shallow vertex tokens to extract key points for footprints. These vertex tokens are related to roof polygons.

B. Polygonal mapping of buildings

Polygonal mapping of buildings involves extracting vectorized building instances that accurately represent building edges. Douglas *et al.* [29] introduced the Douglas-Peucker simplification techniques, but these algorithms often produce rough results that fail to capture the high-quality edges of buildings. Wei *et al.* [30] proposed refinement strategies based

on empirical building shapes, while Girard *et al*. [31] used Frame-Field methods to better align extracted fields with ground truth contours. Zorzi *et al*. [32] described all building polygons in an image as an undirected graph, connecting detected vertices to form the building boundaries. Hisup [20] addressed the challenge of mask reversibility by using deep convolutional neural networks for vertex extraction, followed by boundary tracing of predicted building segmentations to connect the vertices. SAMPolyBuild [19] is a model for solving building layout results in near lidar scenarios, which achieves vectorized result extraction through cropping of pre-selected boxes and editing of weights.

In this paper, PolyFootNet introduces the concept of vertex tokens for extracting vertices and these vertices. Different from SAMPloyBuild, PolyFootNet will not crop each building to ensure each token can reference global information [16]. Meanwhile, to address the consequential imbalance between positive and negative samples, we propose the Dynamic Scope Binary Cross Entropy Loss (DS-BCE Loss). Unlike the aforementioned methods, PolyFootNet is the first model to extract polygonal building footprints from off-nadir images. The footprint polygons are derived through vector operations, offering higher precision.

C. Offset-based footprint extraction

Extracting building footprints through offset-based methods leverages the structural similarity between roofs and footprints. Christie *et al*. [33] proposed a method using a U-Net decoder to predict image-level orientation and pixel-level height values. Building upon this, Li *et al*. [12], [15], [17] introduced multi-task learning approaches for the BFE problem, enabling models to train on datasets with varying labels. LOFT [14] was later developed to extract building footprints as part of an instance segmentation task, representing a building footprint with a roof mask and roof-to-footprint offset. OBM [16] was the first model to tokenize offset representations and extract building footprints using SAM. However, these models all only extract semantic mask results and require a sequence of post-processing operations to derive footprint masks.

In this paper, we expand on traditional offset-based methods, solving the BFE problem by integrating multiple tasks rather than relying solely on offsets. Prior knowledge is applied to explore the potential of using various sources of information. For instance, we investigate how building footprints can be extracted through building segmentation and offsets or building and roof segmentation. Finally, PolyFootNet is integrated with other models to enable fully automatic building footprint extraction.

D. Offset building model

The OBM [16] is a model inherited from the SAM [21]. The primary contribution of OBM is introducing the concept of Offset Tokens and designing associated encoding and decoding methods, which have technically brought the issues related to Off-Nadir imagery into the transformer era. This model supports interactive footprint extraction with visual prompts. OBM proposed a Reference Offset Augmentation Module (ROAM), a module including a base head and adaptive heads, following the idea of Mix-of-Expert [34]. The encoding and decoding methods are included in Base Head and Adaptive Head, consisting of one Feed Forward Network and an Offset Coder. By fine-tuning SAM, OBM achieves precise roof segmentation in interactive mode. Furthermore, integrating the orientation awareness brought by Offset Tokens enables the direct segmentation of building footprints.

Another contribution in this paper is that the authors discovered from the experimental results that longer offsets have better directions than shorter offsets. PolyFootNet designed Self Offset Attention based on this discovery to mitigate the angle performance of shorter offsets.

III. Methodology

This section outlines the methodology of this study. We begin by reintroducing the BFE problem, and then introduce the core designs of PolyFootNet.

A. Problem statement

In an off-nadir remote sensing image *I*, there are N buildings represented as $\hat{B} = \hat{b_1}, \hat{b_2}, \ldots, \hat{b_N}$. The BFE problem identifies the building footprints $F = f_1, f_2, \ldots, f_N$. In OBM, each building $\hat{b_i}$ is represented by a corresponding prompt p_i, which is interactively fed into the model along with the image *I*. The model then predicts the roof segmentation r_i and the roof-to-footprint offset o_i. The footprint f_i is derived by applying the offset to the roof mask in the evaluation stage.

In PolyFootNet, the process is enhanced with the capability for fully automatic footprint prediction, allowing p_i to be empty. PolyFootNet introduces a roof vertex segmentation task to extract vertex points v_i for each building $\hat{b_i}$, and refocuses on building body segmentation b_i. Additionally, in PolyFootNet, prompts are primarily designed for roof-related tasks, differing from OBM by reducing semantic overlap in off-nadir scenarios.

In summary, PolyFootNet introduces two additional prompt-level segmentation tasks along with a global semantic segmentation task for prompting: (1) prompt-level roof vertex segmentation task; (2) prompt-level building segmentation; (3) roof semantic segmentation.

B. Overview of PolyFootNet

Fig.2 (a) illustrates the architecture of the proposed Polygonal Footprint Network (PolyFootNet). For simplicity of exposition, in this paper, we collectively refer to the right side of the figure as the *Decoder*, while all components preceding it are referred to as the *Encoder*.

To begin with, a new encoder was designed as the left side of Fig.2 (a). The core change of this encoder is that it receives visual prompts related to roofs. This allows PolyFootNet to be integrated with other popular near-nadir roof extraction methods with an external proposal model hub. Except for prompts from external models, the encoder includes a lightweight roof extraction module and an HQ mask prompter, which is composed of an HQ encoder and a segmentation decoder.

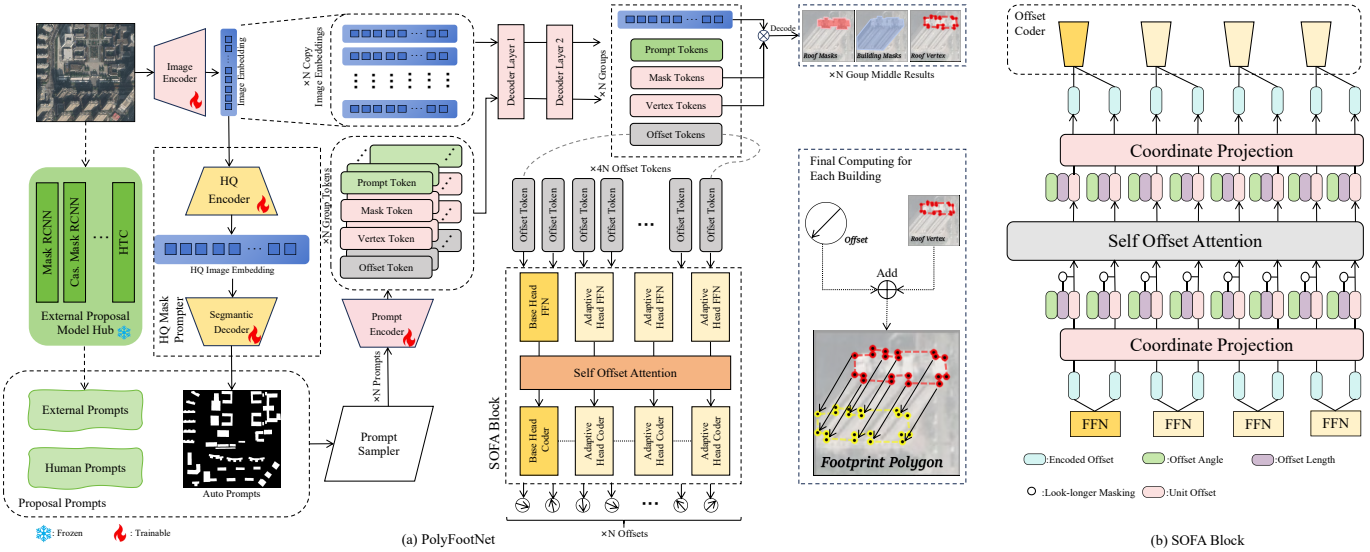


Fig. 2. This figure illustrates the main structures of PolyFootNet and SOFA. In (a), PolyFootNet's newly added Proposal Networks allow the model to extract buildings automatically. In the prompt level, a roof vertex task was added, and the model can directly compute the location of the roof vertex. The footprint polygon is calculated directly from the coordinates. In (b), we provide a detailed SOFA Block. Once outputted from the Feed Forward Network (FFN), the encoded offset will be fed to SOFA. Then, adjusted offsets will be passed to offset coders and compute the final output offsets.

The HQ encoder will enlarge the image embeddings from the image encoder in scale, and the segmentation decoder will decode them as auto roof prompts. The design of the HQ mask prompter referenced the design of Segmenter [35]. Then, the prompt encoder will encode them as prompt tokens. Based on the number of prompts, each prompt token will be allocated mask tokens, a vertex token, and an offset token.

In addition to utilizing the HQ Mask Prompter, PolyFootNet can also leverage pre-trained roof extraction models from the External Proposal Model Hub, which are tailored to specific application scenarios. By integrating the roof predictions generated by these external models, PolyFootNet is able to infer the corresponding building footprint locations. This decoupled design significantly enhances the model's adaptability and applicability in real-world deployments. These models in the hub are **all pre-trained on relevant datasets** prior to integration.

On the right side, after the process of two layers of two-way decoder [21], these tokens will be divided into two streams: vertex tokens will be used to extract key points, and offset tokens will be used to describe their positions to footprints. Via direct coordinate computing, the model can extract footprint key points. From key points to polygons, we adopt a mask-guided connecting strategy similar to the approach used in HiSup [20], leveraging segmentation results to guide the polygon construction process.

To clarify our designs of PolyFootNet, it is broken down into three components: self offset attention, multi-solution of BFE, and training settings.

C. Self offset attention (SOFA)

In prior experiments [16], it was observed that models in prompt mode performed better when extracting footprints of taller buildings with significant offsets compared to lower buildings.

In this paper, we propose a novel, trainable SOFA mechanism based on Nadaraya-Watson Regression [22], [23], and Look-Ahead Masking [36] techniques used in Natural Language Processing (NLP). The SOFA module is designed to address the performance disparity seen in buildings with different offset lengths. The diagram for SOFA is presented in Fig.2.

In machine learning, attention layers can be interpreted as pooling mechanisms. The key role of SOFA is to aggregate offset information, particularly in cases where longer offsets are more reliable.

To introduce SOFA, we begin with Nadaraya-Watson Kernel Regression, a non-parametric regression method. Within SOFA, this technique is applied to accumulate offset knowledge, a vital aspect of the model's improved functionality. The Nadaraya-Watson Kernel Pooling can be described as follows:

$$f(\mathbf{x}) = \sum_{i=1}^{n} \frac{\mathcal{K}(\mathbf{x} - x_i)}{\sum_{j=1}^{n} \mathcal{K}(\mathbf{x} - x_j)} y_i, \tag{1}$$

where f represents the Nadaraya-Watson Kernel Regression, and $\mathcal{K}$ denotes the kernel function. In the context of Transformers, $\mathbf{x}$ refers to the query, $x_1, x_2, \ldots, x_n$ are the keys, and $y_1, y_2, \ldots, y_n$ are the values. This equation can be interpreted as a weighted sum of the values, where the weights are computed based on the similarity between the query $(\mathbf{x})$ and the keys $(x_1, x_2, \ldots, x_n)$.This equation can be interpreted as a weighted sum of the values, where the weights are computed based on the similarity between the query $(\mathbf{x})$ and the keys $(x_1, x_2, \ldots, x_n)$. In essence, Nadaraya-Watson Kernel Regression operates analogously to an attention mechanism, where the kernel function $\mathcal{K}(\mathbf{x} - x_i)$ serves to compute a soft

similarity between the query and each key. These similarities are then normalized to form an attention mask, which is subsequently used to perform a weighted average over the values.

In the BFE problem, our prior knowledge tells us that longer offsets tend to have better direction. Based on the interpretation of Kernel Regression, for determining the offset angle, we compute weights according to the relationship and similarity between each offset. Then, the weights will be used to computing final angles for each offset. Therefore, Eq.1 is translated as Eq.2:

$$\dot{\boldsymbol{\alpha}} = f(\boldsymbol{\rho}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \frac{\mathcal{K}(\boldsymbol{\rho} - \rho_i)}{\sum_{j=1}^{n} \mathcal{K}(\boldsymbol{\rho} - \rho_j)} \alpha_i. \tag{2}$$

In Eq.2, offset queries $\vec{\mathbf{O}} = \{\mathbf{o_1}, \mathbf{o_2}, ..., \mathbf{o_n}\}$ was expressed as polar coordinate $\vec{\mathbf{O}} = \{(\rho_1, \alpha_1), ..., (\rho_n, \alpha_n)\}$. $\boldsymbol{\rho}$ means lengths of the offsets, and $\boldsymbol{\alpha}$ represents angles of the offsets. $\dot{\boldsymbol{\alpha}}$ is the corrected offset angles.

To facilitate computation, $\mathcal{K}$ is defined as Gaussian Kernal:

$$\mathcal{K}(u) = \frac{1}{\sqrt{2\pi}} e^{(-\frac{u^2}{2})}. \tag{3}$$

Consequently, Eq.2 transformed as:

$$f(\boldsymbol{\rho}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \text{softmax}\left(-\frac{1}{2}(\boldsymbol{\rho} - \rho_i)^2\right) \alpha_i. \tag{4}$$

To describe more complicated similarity, we make the whole attention learnable, and a trainable parameter w was added to Eq.4:

$$\begin{aligned} \dot{\boldsymbol{\alpha}} &= \text{SOFA}_a(\boldsymbol{\rho}, \boldsymbol{\alpha}) \\ &= \sum_{i=1}^{n} \text{softmax}\left(-\frac{1}{2}\left(w \times (\boldsymbol{\rho} - \rho_i)\right)^2\right) \alpha_i. \end{aligned} \tag{5}$$

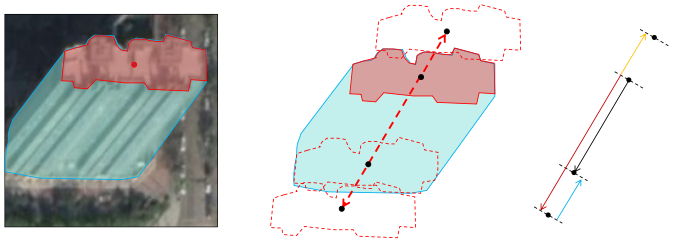
In NLP, look-ahead masking was used to combine the fact that tokens at one stage should only make the current and past knowledge visible for the model. Tokens in subsequent positions were masked via a substantial negative number in softmax$(\cdot)$. In the BFE problem, the longer offsets perform better than the shorter offsets. Based on the aforementioned ideas, we need to design a look-longer masking $\boldsymbol{\mathcal{M}}$ for those shorter offsets. Finally, angle-level SOFA can be expressed as:

$$\begin{aligned} \dot{\boldsymbol{\alpha}} &= \text{SOFA}_a(\boldsymbol{\rho}, \boldsymbol{\alpha}) \\ &= \sum_{i=1}^{n} \text{softmax}\left(-\frac{1}{2}\left(w \times \boldsymbol{\mathcal{M}}(\boldsymbol{\rho} - \rho_i)\right)^2\right) \alpha_i. \end{aligned} \tag{6}$$

Based on similar mathematical reasoning processes, vector-level SOFA can be written as:

$$\begin{aligned} \dot{\mathbf{O}} &= \text{SOFA}_v(\boldsymbol{\rho}, \vec{\mathbf{U}}) \\ &= \boldsymbol{\rho}^\top \sum_{i=1}^{n} \text{softmax}\left(-\frac{1}{2}(w \times \boldsymbol{\mathcal{M}}(\boldsymbol{\rho} - \rho_i))^2\right) \vec{\mathbf{u}_i}, \end{aligned} \tag{7}$$

$\vec{\mathbf{U}} = \{\vec{\mathbf{u}_i}\}$ is the unit offsets. $\dot{\mathbf{O}} = \{\dot{\mathbf{o}_i}\}$ is the output offset. Additionally, the derivation of vector SOFA is provided in Appendix A.



(a) Roof and building segmentation (b) Critical Condition (c) Length Relationship

Fig. 3. (a) describes the predicted roof and building for one building. (b) displays the critical condition of regressing building offset (c) abstracts the situation of (b).

The w among the mentioned operations was set to 0. Based on our experiment, this parameter fluctuated at around 0 while training.

From Eq.6 and Eq.7, SOFA is a portable and plug-and-play block, because (1) the learnable parameter w was light; (2) in one off-nadir image I, the number of buildings N tends to be under 100, making the matrix's spatial operation not consume much GPU memory.

D. Multi-solution of BFE

In the BFE problem, most of the studies are intuitively established based on the mask similarity between roofs and footprints, as shown in Fig.1(a). Inspired by the third law of geography, which posits that information in spatial representation is often redundant; we want to explore different solutions of BFE to determine whether *roof + offset* is the only solution to the BFE problem. Under this setting, we explore the solution of Fig.1(b) and (c).

In (c), the extracting method is also explicit: move the building mask along the roof-to-footprint offset, and the union between that and the building mask is the footprints. Similarly, if the building mask were moved in the opposite direction, the union would be the roof mask.

In (b), the situation was more complicated. The first step was to find a direction to better represent the direction from roof to footprint. In this direction, a search algorithm was applied to detect the location of footprints by valuing different lengths of movement. Of course, we can also use the predicted global offset direction as this direction. In Algorithm 1, we introduced how to extract building footprint with only a roof and building segmentation.

From lines 1 to 11 in Algorithm 1, a linear search was applied to identify the optimal rotation angle for moving the roof segmentation to best overlap with the corresponding building segmentation. Subsequently, given this optimal angle, a binary search was utilized to determine the optimal offset length along this angle precisely.

Specifically, when employing spatial information in Algorithm 1 to extract building footprints, the optimal offset length is determined based on the critical condition illustrated in Fig.3. In the algorithm, the expression $\mathbf{V}_{\theta,l} \cdot r_i$ denotes the spatial translation of the roof mask r_i along the optimal direction θ by the offset length l, where $\mathbf{V}_{\theta,l}$ represents the corresponding homogeneous movement matrix.

Algorithm 1 Footprint Searching

Require: Roof segmentation r_i, building segmentation b_i, maximum iteration N, offset step length $\bar{l}$
Ensure: Related footprint segmentation f_i

Define angle set: $\boldsymbol{\alpha} \leftarrow \{0°, 1°, \ldots, 359°\}$
Initialize optimal angle $\theta \leftarrow 0$, optimal offset length $l \leftarrow 0$, and maximum overlap $S_{\max} \leftarrow 0$
for $\alpha \in \boldsymbol{\alpha}$ **do**
  Construct movement matrix:

$$\mathbf{V}_\alpha = \begin{bmatrix} 1 & 0 & -\bar{l}\cos\alpha \\ 0 & 1 & -\bar{l}\sin\alpha \\ 0 & 0 & 1 \end{bmatrix}$$

  Move roof segmentation: $\dot{R}_\alpha \leftarrow \mathbf{V}_\alpha \cdot r_i$
  Calculate overlap ratio:

$$S_\alpha \leftarrow \frac{\text{Area}(\dot{R}_\alpha \cap b_i)}{\text{Area}(r_i)}$$

  if $S_\alpha > S_{\max}$ **then**
    Update optimal angle: $\theta \leftarrow \alpha$
    Update maximum overlap: $S_{\max} \leftarrow S_\alpha$
  end if
end for
Determine optimal offset length:

$$l \leftarrow \arg\min_{l_c} \left| \frac{\text{Area}(\mathbf{V}_{\theta,l_c} \cdot r_i \cap b_i)}{\text{Area}(r_i)} - 1 \right|$$

where

$$\mathbf{V}_{\theta,l_c} = \begin{bmatrix} 1 & 0 & -l_c\cos\theta \\ 0 & 1 & -l_c\sin\theta \\ 0 & 0 & 1 \end{bmatrix}$$

Compute final footprint segmentation:

$$f_i \leftarrow \mathbf{V}_{\theta,l} \cdot r_i$$

Fig.3(c) demonstrates the method for accurately extracting offsets. As shown, the binary search mentioned previously was conducted twice: once to determine the optimal yellow offset and once for the optimal red offset. Given that the blue offset length equals the yellow offset, the final roof-to-building offset is calculated as the difference between the lengths of the red offset and the yellow offset.

E. Training settings

The OBM's losses were combined with two parts: prompt-level segmentation loss and offset losses in ROAM, which can be expressed as:

$$\mathcal{L}_{OBM} = \mathcal{L}_{ROAM} + \mathcal{L}_{roof} + \mathcal{L}_{building}, \tag{8}$$

where $\mathcal{L}_{ROAM}$ is the loss of ROAM, which applies SmoothL1 Loss [37] for each offset head. $\mathcal{L}_{roof}$ is CrossEntropy Loss [38] of roof segmentation, and $\mathcal{L}_{building}$ is CrossEntropy Loss of building segmentation.

For two new tasks, CrossEntropy Loss is applied for roof semantic segmentation. The model outputs a vertex map for each building in prompt-level vertex segmentation. Because the vertex map contains the whole scope of the inputted image, most pixels will be valued as a negative sample by 0. Sparse positive key points will mislead the model to predict negative samples only. Via experiments, if the model uses a fixed window size to crop the building area, it can lead to severe grid effects. As a result, DS-BCE Loss was designed for prompt-level vertex segmentation:

$$\mathcal{L}_{vetex} = \sum^{p\in Z+\Delta} -y_p \log y'_p - (1-y_p)\log(1-y'_p), \tag{9}$$

y_p and y'_p are pixels on the ground truth and prediction maps at p. Z is the original prompted area, and Δ is a random small neighborhood of this area. This neighborhood enlarges the original prompt area by 1 to 3 pixels randomly. In Appendix B, a clearer definition can be found. Finally, PolyFootNet was trained as:

$$\mathcal{L} = \lambda \times (\mathcal{L}_{OBM} + \beta\mathcal{L}_{vertex}) + \kappa \times \mathcal{L}_{seg}, \tag{10}$$

λ and κ were parameters equal to 1 or 0 to control whether semantic heads would be trained together. β is a scale factor to balance the losses of vertex tasks and other tasks.

IV. Experiment and Analysis

This section describes the data used in this work, justifies their choice, and specifies their sources. Then, the main results of the models are reported and analyzed. Lastly, a generalization test was conducted on the Huizhou test set.

A. Dataset

In our experiments, three datasets were employed to make a comparison:

BONAI [14]: This dataset was launched with benchmark model LOFT. There are 3,000 train images and 300 test images, and the height and width of the images are 1024. Building annotations include roof segmentation, footprint segmentation, offset, and height.

OmniCity-view3 [39]: OmniCity-view3 provided 17,092 train-val images and 4,929 test images in height and width 512. Footprint segmentation, building height, roof segmentation, and roof-to-footprint offset were labeled for each building.

Huizhou test set [16]: This small dataset labeled images from a new city Huizhou, China. All buildings were plotted point-to-point by human annotators. The shape of the images is the same as that of BONAI, and there are over 7,000 buildings with offsets, roofs, and footprint segmentation.

In comparison, OmniCity images have a relatively higher spatial resolution and smaller cropped image sizes than those from BONAI and Huizhou.

B. Metrics

1) Offset metrics: Basically, we measure each offset in three aspects: Vector Error (VE), Length Error (LE), and Angle Error (AE), which defined as:

$$VE = |\vec{p} - \vec{p}_g|_2 \,; LE = \big|\, |\vec{p}|_2 - |\vec{p}_g|_2 \,\big| \,; AE = |\theta - \theta_g|, \tag{11}$$

TABLE I
Main results on BONAI [14].

Model	F1	Precision	Recall	EPE	mVE	mLE	mAE	aVE	aLE	aAE
PANet	58.06	59.26	56.91	-	-	-	-	-	-	-
M RCNN	58.12	59.26	57.03	-	-	-	-	-	-	-
MTBR-Net	63.60	64.34	62.87	5.69	-	-	-	-	-	-
LOFT	64.42	64.43	64.41	4.85	-	-	-	-	-	-
Cas.LOFT*	62.58	63.67	61.52	4.79	-	-	-	-	-	-
MLS-BRN	66.36	65.90	66.83	4.76	-	-	-	-	-	-
p.LOFT	72.98	85.74	64.01	-	15.4	**12.6**	0.18	6.12	4.51	0.32
p.Cas.LOFT*	76.05	**87.20**	67.82	-	15.8	13.5	0.17	5.97	4.48	0.31
OBM	**80.03**	82.57	**77.97**	-	**15.3**	13.9	0.12	**5.12**	**4.05**	0.22
OBM†	78.65	80.41	77.21	-	17.0	15.7	0.12	5.38	4.35	0.22
Ours(mask)	78.74	79.85	77.91	-	17.0	15.8	**0.11**	5.46	4.40	**0.22**
Ours(poly.)	75.31	78.12	73.13							

where $\vec{p}$ and $\vec{p}_g$ represent the predicted and ground truth offset. θ and θ_g represent the predicted and ground truth angle. The $|\cdot|$ and $|\cdot|_2$ represent the 1-Norm and 2-Norm respectively.

Referring to COCO metrics [40], each offset will be grouped based on the length of its related ground truth.

$$m\mathcal{E} = \frac{\mathcal{E}_{(10n,\infty)} + \sum_{i=0}^{n} \mathcal{E}_{(10i,10i+10)}}{n+1} \tag{12}$$

where $\mathcal{E}$ can represent VE, LE and AE. $\mathcal{E}_{(i,j)}$ represents the mean error of the offset group whose length is between i and j. Specifically, $\mathcal{E}_{(0,\infty)}$ is named as average error $a\mathcal{E}$.

For instance, in segmentation-based methods, we compute the object-wise end-point error [14] in pixels to evaluate the Euclidean distance between the endpoints of the predicted and ground truth offset vector.

2) Mask metrics: Precision, Recall, and F1score are used to evaluate the quality of footprint masks and polygons.

C. Baseline introduction

To demonstrate the effectiveness of our method, the results of the following models were chosen as comparative experiments.

- Path Aggregation Network (PANet) [41] is an instance segmentation model.
- Mask RCNN [42] is an ROI-based instance segmentation network.
- MTBR-Net [12] is a semantic segmentation network based on global offset features used for extracting building footprints.
- LOFT [14] is an *instance-segmentation-based* footprint extraction model.
- MLS-BRN [15] is a multi-task model that gives more diverse building-related information, like shooting angles and building height.
- OBM [16] is a specially designed model for promptable footprint extraction.

D. Experimental settings

In the training process of our PolyFootNet was trained in two stages. All images will be reshaped in 1024×1024 pixels. Experiments were conducted on a server with 4 NVIDIA RTX 3090. In the first stage, PolyFootNet was trained in roof prompting mode. Then, proposal networks were trained solely. In the whole training process, we used stochastic gradient descent (SGD) [43] as the optimizer with a batch size of 4 for 48 epochs, an initial step learning rate of 0.0025, a momentum of 0.9, and a weight decay of 10^{-4}. The number of parameters in PolyFootNet is 77.69 M(+32.17 M for HQ prompter).

In this paper, the experiments will be divided into three groups: in the experiments on the datasets BONAI and OmniCity, we will use the author's original training and testing set partitioning criteria, respectively. For the experiments on the Huizhou testing set, the models trained in the BONAI experiment will be directly tested.

E. Main results

In this section, we evaluate our methods on BONAI and OmniCity-view3. The primary comparison will focus on LOFT [14], Cascade LOFT and OBM between PolyFootNet because these models are available for extensive experiments.

In the listed tables, the displayed results were separated into three parts. The first part lists results from end-to-end models. Note that: * model is a model that we reproduce based on the author's intention in paper [14]. OBM† was retrained under the same training setting as PolyFootNet. p. is short for *prompt*. Additionally, EPE, mVE, mLE, aVE, and aLE were in pixels. Precision, Recall, F1score were measured in percentage(%).

All available promptable models were compared with our proposed model. From Tab.I, although PolyFootNet can provide better roofs, the quality of offsets and footprints is not as good as the formerly proposed methods. Additionally, except experiments on OmniCity in Tab.II, clear drops between polygon results and mask results was found on Huizhou and BONAI datasets: their F1score dropped by 3.43% and

4.01%. This may be caused by the resolution of images. As shown in Fig.4, BONAI and Huizhou datasets have lower pixel resolution compared with OmniCity dataset, leading to less detailed edge information of buildings. Consequently, the key point extraction and point connections are hard to do compared with those on OmniCity. Extensive experiments were conducted in ablation studies to examine this hypothesis.

Experiments on OmniCity-view3 demonstrate the advance of our PolyFootNet. In Tab.II, PolyFootNet nearly outperformed all mentioned models. Although vectorized footprint polygons still have lower F1score than their mask footprints by 0.81%, PolyFootNet outperformed all mentioned models. In PolyFootNet, polygonal results even have better Precision than mask results(+0.7%). Moreover, PolyFootNet performed better than any other model in terms of offset prediction. aVE of PolyFootNet is 1.423 pixels lower than that of prompt LOFT and 0.544 pixels lower than the figure for OBM.

In Fig.4, visualized results in the prompt mode were provided. The first and second lines of illustrations were selected from BONAI [14] and Huizhou [16]. The third and fourth lines were from OmniCity-view3 [39]. In comparison, BONAI and Huizhou have a similar spatial resolution and crop size, but OmniCity is different. Our model can provide more editable polygonal results compared with other models.

A generalization test was conducted at the Huizhou test set in Tab.III. All models were pre-trained on BONAI, and there was no extra training. In predicting footprints, results of OBM and PolyFootNet exhibit similar attributes with them on BONAI, *e.g.* the gap between F1scores of footprints predicted by PolyFootNet on BONAI (75.31%) and Huizhou Test (75.35%) is 0.04%, and a similar gap between mask and polygon results were found.

In summary, PolyFootNet can directly predict footprint polygons, the performance of which showcased a certain generalization ability. Moreover, by comparing Tab. I, Tab. II, and Tab. III horizontally, we observed two key patterns: (1) PolyFootNet does not consistently outperform other methods on the BONAI and Huizhou datasets as it does on the OmniCity dataset; and (2) the gap between the predicted mask results and polygonal results is noticeably larger on BONAI and Huizhou compared to OmniCity. We believe this is caused by differences in dataset resolution and cropping strategies, which lead to varying levels of "effective pixel perception" at the model level. To validate this hypothesis, we added an experiment in Section V-A to test this assumption.

F. Multi-solution for BEF problem

Multi-information can be divided into two classes: information extracted by different kinds of building-related models and extracted multi-information by PolyFootNet. Human-plotted prompts will motivate models to reach their ceiling performance. In this part, footprints F1score were selected to measure mask ability; meanwhile, EPE and aVE, which are similar in definition, were used to measure offset ability. Each model will be scattered on coordinates. To facilitate comparison, BONAI was used due to its diverse open-source methods and known experimental results.

In Fig.5, the BFE problem was solved with different information. For promptable models, OBM and PolyFootNet, building segmentation with offsets can predict better building footprint (F1score +1.37 and +0.23 respectively). End-to-end ROI-based can also extract footprint with building segmentation and offsets, which provides similar performance with related models using roof and offset. Additionally, extracting footprints with roof and building segmentation has been proved applicable, although offsets regressed in this version were inaccurate compared with models using roofs. All models can provide better results than Mask RCNN.

The automatic extraction of building footprints commonly relies on proposal regions. In Tab.IV, PolyFootNet was tested with different region proposal functions. Additionally, utilizing different sources of information to extract footprint print was also examined.

r., b., o. and d. represent roof mask, building mask, offset and offset direction.

With the help of HTC, PolyFootNet can automatically extract building footprints, and the Recall of this model is higher (+8.16) than the former SOTA MLS-BRN. By adjusting different prompting modes, PolyFootNet can also reach a similar precision to MLS-BRN (-1.24).

In Fig.6, footprints extracted by auto mode were compared with each other. Our model can still provide polygonal results compared with other models.

V. Ablation Studies

This section will examine the proposed algorithms and modules. Apart from the SOFA module, the aforementioned algorithms proposed for extracting footprints with different information also need ablation.

A. Pixel resolution caused marginal performance

Polygonal footprints are typically generated by connecting the extracted key points based on the orientation of the mask footprint contours. During this process, discrepancies between mask results and polygon results are inevitable. However, compared to the OmniCity dataset, the performance discrepancies observed on the BONAI and Huizhou datasets are significantly greater. As shown in Fig.4, the spatial resolution of most images in OmniCity is considerably higher than that of BONAI and Huizhou datasets. This leads to buildings occupying more effective pixels within images, providing clearer key points and edges, thereby reducing the differences between vectorization and mask results.

Under the PolyFootNet setting, we provide a detailed calculation of the differences between the datasets. First, the model's encoder adopts a ViT architecture, which only accepts fixed-size inputs of 1024×1024 pixels and ultimately encodes them into 256×256 feature maps. The images in the BONAI and Huizhou datasets primarily have a spatial resolution of 0.5 meters, with image sizes of 1024×1024. As a result, each pixel on the corresponding feature map represents a real-world area of approximately 2m × 2m. In contrast, the OmniCity dataset mainly features images with a spatial resolution of about 0.2 meters and a size of 512×512. These images are upsampled

TABLE II
Experimental results on OmniCity-view3.

Model	F1	Precision	Recall	EPE	mVE	mLE	mAE	aVE	aLE	aAE
M RCNN	69.75	69.74	69.76	-	-	-	-	-	-	-
LOFT	70.46	68.77	72.23	6.08	-	-	-	-	-	-
MLS-BRN	72.25	69.57	75.14	5.38	-	-	-	-	-	-
p.LOFT	82.27	90.63	75.81	-	54.3	48.5	0.65	7.57	5.29	0.70
p.Cas.LOFT	83.75	**91.62**	77.54	-	52.9	48.4	**0.62**	7.25	5.12	0.69
OBM	86.03	90.17	82.52	-	56.9	53.7	0.66	6.69	5.15	0.64
Ours(mask)	**88.42**	90.06	**87.01**	-	**51.5**	**47.9**	0.63	**6.15**	**4.65**	**0.59**
Ours(poly.)	87.61	90.76	84.93							

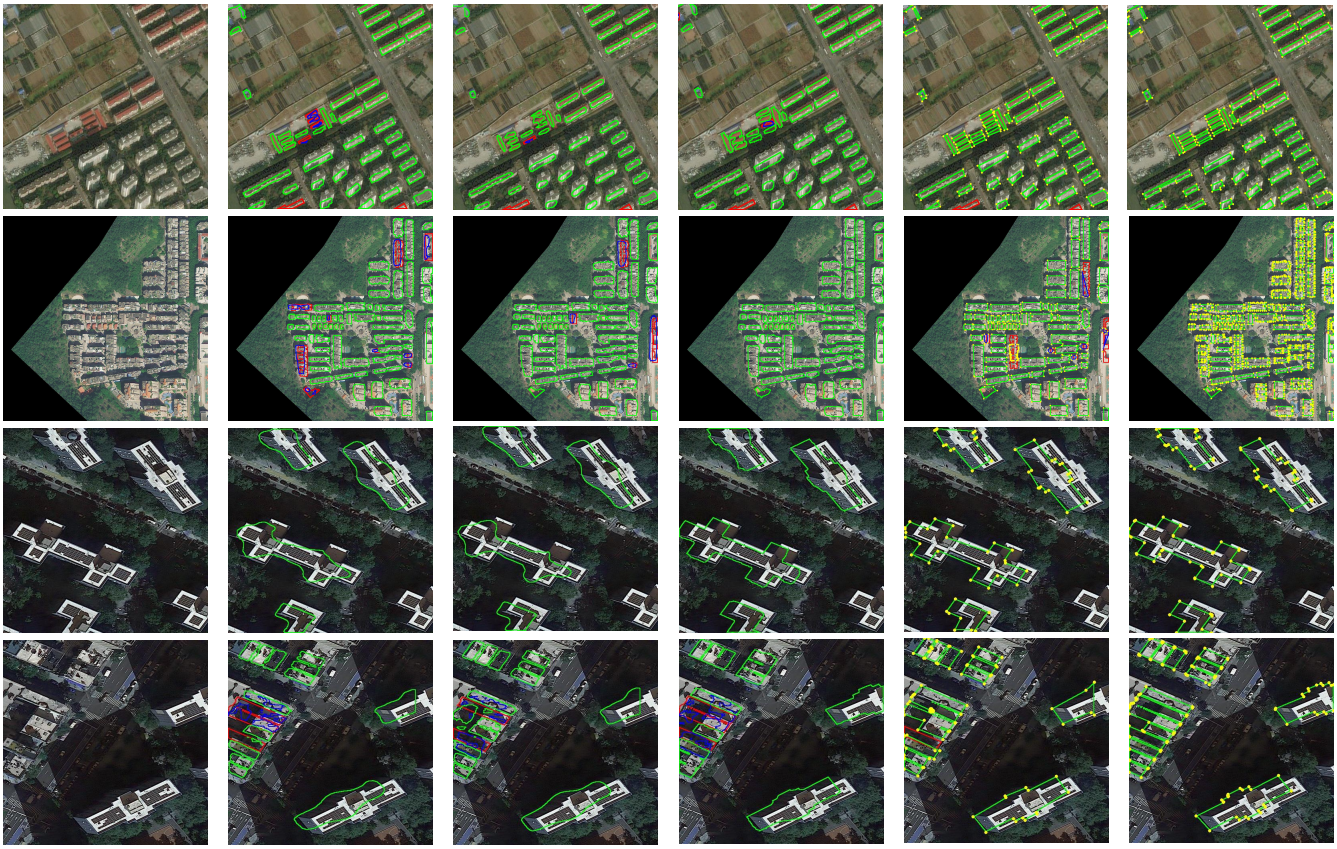


Fig. 4. Main results extracted by prompting mode. The lines represent building footprint boundaries, and the yellow points are key nodes of the building footprints. The green, blue, and red polygons denote the TP, FP, and FN.

to 1024×1024 before being fed into the model, resulting in feature maps where each pixel corresponds to a real-world area of about 0.4m × 0.4m.

To validate this hypothesis, we simulated higher-resolution inputs by upsampling the BONAI and Huizhou images by a factor of two, dividing each upsampled image evenly into four 1024×1024 sub-images, and retraining the model under this new setting. As shown in Tab. V, our model outperforms OBM, and the performance gap between the mask and polygon results is noticeably reduced compared to the original datasets.

B. Ablation of SOFA

Unlike other transformer modules, SOFA was developed to be insensitive to the input length of embedded tokens. Because of this feature, SOFA module can adapt to any offset-based model. After all other modules finished training, the SOFA module was trained in the last stage. In Tab.VI, SOFA module was applied in all open-source models and can reduce both prompt-level and instance-level offset errors. *e.g.* EPE of LOFT on the BONAI dataset declined by 0.33 pixels.

TABLE III
Experimental results on Huizhou test set.

Model	F1	Precision	Recall	aVE	aLE	aAE
p.LOFT	72.56	83.02	65.33	7.935	6.449	0.752
p.Cas.LOFT	75.83	81.71	71.13	7.894	5.938	0.818
OBM	81.53	78.80	84.80	4.898	4.351	0.169
OBM†	80.30	79.04	81.85	4.985	4.412	0.198
Ours(mask)	80.36	78.99	82.18	4.959	4.636	0.144
Ours(poly.)	75.35	77.04	74.25			

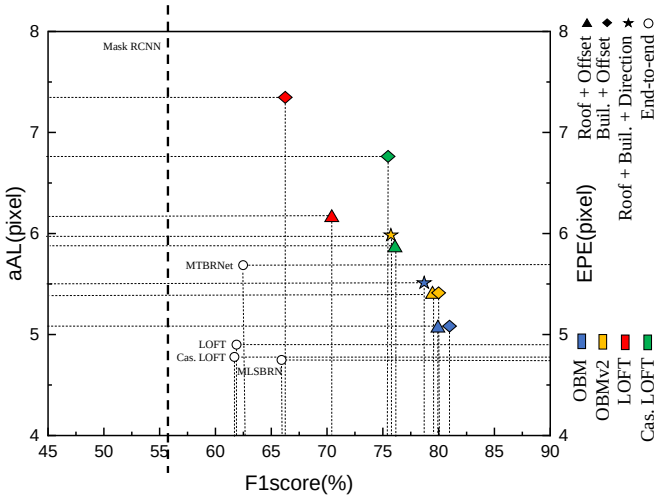


Fig. 5. Extracting footprints with multi-solutions of BFE.

TABLE V
Performance comparison under upsampled BONAI and Huizhou dataset

Model	Dataset	F1	Precision	Recall	mVE	mLE	mAE	aVE	aLE	aAE
OBM		63.18	74.91	55.41	22.87	21.12	0.17	6.91	5.63	0.28
Ours(mask)	BONAI	74.97	77.12	73.56	23.34	21.67	0.18	6.24	5.05	0.25
Ours(poly.)		74.49	76.89	73.00						
OBM		57.99	73.94	48.61	24.88	20.11	0.55	8.24	6.40	0.61
Ours(mask)	Huizhou	72.94	73.96	72.61	14.01	12.42	0.17	6.01	5.32	0.25
Ours(poly.)		72.73	73.15	73.09						

TABLE VI
SOFA module ablation studies on BONAI.

Model	EPE	mVE	mLE	mAE	aVE	aLE	aAE
LOFT	4.85	15.4	12.6	0.18	6.12	4.51	0.32
LOFT+SOFA	**4.52**	**14.6**	**12.6**	**0.13**	**5.62**	**4.49**	**0.22**
Cas.LOFT	4.79	15.8	13.5	0.17	5.97	4.48	0.31
Cas.LOFT+SOFA	**4.43**	**15.3**	**13.4**	**0.12**	**5.64**	**4.44**	**0.22**
OBM	-	15.3	13.9	0.12	5.12	4.05	0.22
OBM+SOFA	-	**15.3**	**13.9**	**0.11**	**5.08**	**4.04**	**0.21**
OBM†	-	17.0	15.7	0.12	5.38	4.35	0.22
OBM†+SOFA	-	16.8	15.7	0.11	5.36	4.35	0.21

C. Multi-solutions of BFE

Ablation studies were conducted with ground truth labels to ensure the proposed algorithms can extract building footprints.

r., b., o. and d. represent roof mask, building mask, offset and offset direction.

TABLE IV
Auto extraction results on BONAI test set.

Model	F1	Precision	Recall	EPE
M RCNN	56.12	57.02	55.26	-
LOFT(r.o.)	64.42	64.43	64.41	4.85
LOFT(b.o.)	61.97	61.41	62.55	5.83
Cas.LOFT(r.o.)	62.58	63.67	61.52	4.79
Cas.LOFT(b.o.)	61.67	64.59	59.00	5.23
MLS-BRN	**66.36**	**65.90**	66.83	4.76
Ours+HTC(r.o.)	61.48	55.19	74.79	**4.59**
Ours+HTC(b.o.)	61.48	55.13	**74.99**	
Ours+seg.	55.20	64.66	50.47	-

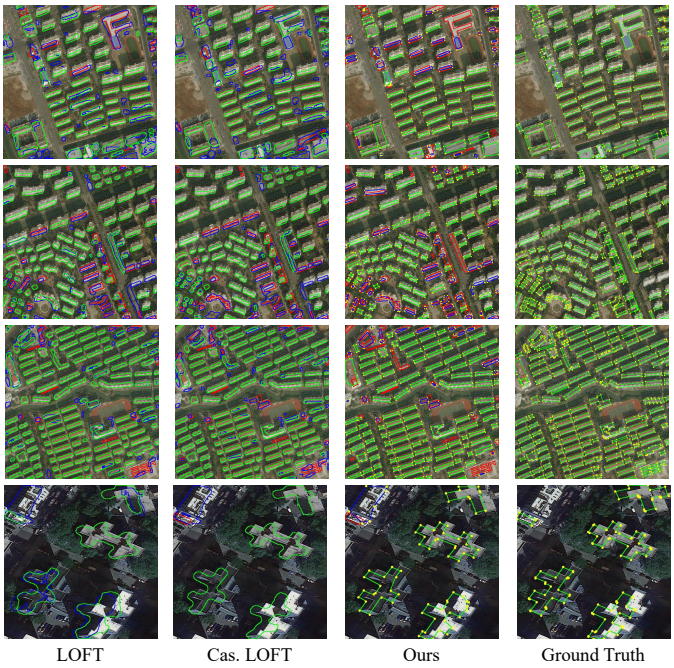


Fig. 6. Main results extracted by auto mode. The green, blue, and red polygons denote the TP, FP, and FN.

From Tab.VIII, our proposed algorithms can accurately extract building footprints via different information. Although our algorithms can extract footprints merely with roof and building segmentations, grid-based digital images are always limited by image continuity. The influence of this problem is more severe, especially on the representations of buildings with short offsets, *e.g.* an offset with length 1 pixel, there are only 4 points nearby that can represent its endpoint. This ambiguity leads to a poor perception of direction. As a result, when the offset direction was given, the aVL dropped by 5.16 pixels and the f1-score of footprints increased by 6.5%.

D. Proposal methods

PolyFootNet can receive almost all kinds of models that can provide bounding boxes related to buildings. Except for an extra segmentation head on PolyFootNet, PolyFootNet was integrated with other models that can provide roof-bounding

boxes or segmentations. These outputs perform as rough extraction results, and PolyFootNet will correct and refine them.

In Tab.IX, PolyFootNet was integrated with its segmentation head, HTC and LOFT. Specifically, HTC and LOFT are matched with different NMS strategies. ♠ represents soft NMS algorithms with a score threshold of 0.05, IoU threshold of 0.5, and a maximum of 2000 instances per image. ♣ represents NMS algorithms with a score threshold of 0.1, an IoU threshold of 0.5, and a maximum of 2000 instances per image. ♡ leverages the result from ♠, but the same instance, which was repeatedly predicted, will be merged as one instance.

Soft NMS algorithm gives almost all bounding boxes with a very low score threshold. This means most output boxes will be selected as final outputs. Consequently, they can provide results with high Recall, but PolyFootNet was trained with annotations that can cover the whole building or roof. As a result, the Precision was not good. *e.g.* Precision of PolyFootNet + HTC♠ is lower than that of PolyFootNet + HTC♡ by 15.96%, although the Recall of PolyFootNet + HTC♠ is higher than that of PolyFootNet + HTC♡ by 10.49%. Finally, the score of PolyFootNet + HTC♠ was adversely influenced, which only reached 51.79%.

E. Inference speed analysis of model components

To evaluate the inference capability of the model, we use frames per second (FPS) as the metric. Specifically, we input 300 images of size 1024×1024 from the BONAI test set, along with a total of 20,589 building location visual prompts. The experiments are conducted on a single RTX 3090 GPU, processing one image and its corresponding prompts at a time. The final FPS is calculated as the average over five repeated runs.

TABLE VII
Inference speed (FPS) under different module configurations on the BONAI test set. *Foot.* represents extracting footprint location. *Poly.* represents vectorized results.

Model	SOFA	Foot.	Poly.	Auto	FPS
PolyFootNet					0.8480
	✓				0.8709
	✓	✓			0.3153
		✓	✓		0.2321
	✓	✓	✓		0.2296
	✓	✓	✓	100 p.	0.1733
	✓	✓	✓	All	0.1601
p. LOFT					3.1423
	✓				3.2358
LOFT				✓	0.4542
	✓			✓	0.4613
MLSBRN		✓		✓	0.1968

As shown in Tab.VII, SOFA is a highly efficient module whose impact on inference speed is negligible—even smaller than the experimental variability observed across repeated runs. Currently, the primary factor affecting the efficiency of PolyFootNet is the number of input prompts, as well as certain CPU processing steps. Under the automated extraction mode, we evaluated the inference speed using 200 prompts, 100 prompts, and all available prompts provided by the HQ Mask Prompter. We also assessed the impact of integrating SOFA with other models.

From Tab.VII, we could know that PolyFootNet may only output 100-200 buildings for each image in BONAI on average, while other models like LOFT and MLSBRN, may predict over 1,000 buildinga with soft NMS. This observation in Tab.IV highlights a key characteristic of our approach when using the HQ Mask Prompter: its Precision is significantly higher than its Recall. This outcome is not incidental but reflects a deliberate design choice within PolyFootNet. The model is intentionally calibrated to favor precision over recall, making a trade-off that prioritizes the reliability of its output for engineering applications. By minimizing false positives, PolyFootNet produces cleaner, more trustworthy footprint data, which is crucial for practical use cases where correctness often outweighs completeness.

VI. Discussion

A. Try to understand SOFA

The design of SOFA is inspired by prior knowledge: when predicting longer offsets, the model can provide relatively accurate directions but less precise lengths; whereas, for shorter offsets, the model tends to predict more accurate lengths but less precise directions. As shown in Eq.7, SOFA introduces the concept of kernel regression in its design, determining the angular mixture weighting ratios by calculating the length relationships among all building offsets: when longer offsets come across shorter offsets, their related similarity will be larger than others. Combined with modern deep learning module design principles, this approach enhances the performance of building offset prediction at the module level.

To demonstrate the effectiveness of the SOFA module, Tab.X presents the improvements in both length and angle prediction across different offset lengths after applying the SOFA vector version. This table indicates that SOFA helps improve angle prediction for shorter offsets, as well as achieve more accurate overall offset predictions.

Initially, we assumed that improving the offset predictions would naturally lead to better footprint accuracy. However, our experiments revealed that this improvement is not always guaranteed. This unexpected finding is particularly intriguing.

Initially, we intuitively assumed that the improvement of offset prediction quality brought by SOFA would naturally lead to better footprint results. However, this was not always the case. We hypothesize the existence of a "dust equilibrium effect" in footprint prediction, where correcting offsets does not necessarily improve the final footprint performance. This phenomenon is discussed further in the following subsection.

B. Dust equilibrium effect in BEF: the role of offset metrics

The *dust equilibrium effect* refers to the phenomenon where cleaning dust from long-used electronics may cause them to

TABLE VIII
Extract footprints with different ground truth labels on BONAI test set.

Model	F1	Precision	Recall	aVE
b.+o.	98.22	98.60	97.88	0
r.+o.	98.55	99.30	97.84	0
b.+r.	87.87	88.67	87.11	5.83
b.+r.+d.	94.37	98.17	91.69	0.67

TABLE IX
Extract footprints with other roof extraction models on BONAI test set.

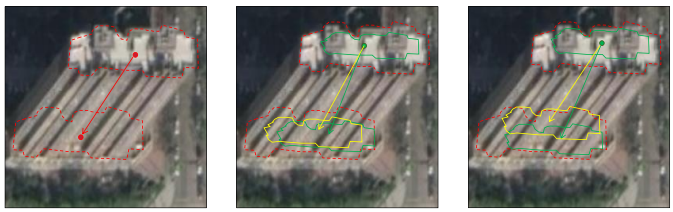
Model	F1	Precision	Recall	EPE
PolyFootNet	78.74	79.85	77.91	-
+eve	59.67	68.66	55.32	5.03
+HTC♡	61.48	55.19	74.80	4.59
+HTC♠	51.79	39.23	85.29	4.92
+HTC♣	60.70	59.18	66.32	4.42
+LOFT♠	56.24	44.01	84.33	5.07
+LOFT♣	60.26	56.32	68.48	4.59

TABLE X
Offset prediction improvement for different length groups with SOFA.

Model	$\mathcal{E}_{(i,j)}$	$(0,10)$	$(10,20)$	$(20,30)$	$(30,40)$	$(40,50)$
p. LOFT	VE	5.49	4.36	6.38	6.85	8.48
(w/o)	LE	4.16	2.77	4.49	5.24	6.59
	AE	0.54	0.20	0.17	0.11	0.11
	F1	**69.68**	75.29	**73.02**	**71.88**	**70.79**
p. LOFT	VE	4.73	4.04	6.15	6.54	8.64
(w)	LE	4.05	2.80	4.50	5.23	6.86
	AE	0.34	0.15	0.14	0.08	0.09
	F1	69.40	**75.51**	72.72	71.56	69.43
Ours	VE	4.46	4.31	6.58	6.98	8.68
(w/o)	LE	3.57	3.10	5.19	5.71	7.09
	AE	0.41	0.17	0.16	0.09	0.10
	F1	78.20	79.78	76.95	73.63	**75.75**
Ours	VE	3.91	4.00	5.92	6.34	7.49
(w)	LE	3.20	2.73	4.46	5.17	6.10
	AE	0.34	0.16	0.15	0.08	0.08
	F1	**78.42**	**79.79**	**76.99**	**73.64**	75.66

TABLE XI
Impact of mask quality improvement accompanying SOFA offset enhancement on the BONAI dataset

Model	SOFA	GT	F1	Precision	Recall	aVE
p. LOFT			72.98	85.74	64.01	6.12
	✓		72.85	85.62	63.88	5.62
		✓	83.61	84.18	83.61	6.12
	✓	✓	84.50	85.10	83.93	5.62
p. Cas.LOFT			76.05	87.20	67.82	5.97
	✓		75.64	86.79	67.42	5.64
		✓	84.27	84.88	83.69	5.97
	✓	✓	84.65	85.29	84.06	5.64



(a) Ground Truth (b) Footprint Metrics Unchange (c) Footprint Metrics Decline

Fig. 7. Possible dust equilibrium effect. In (a), the red area represents the ground truth label. In (b) and (c), the green areas denote the predicted results, while the yellow areas show the outcomes after offset correction.

fail, even if they previously functioned well. This often occurs in systems or tasks that require multi-step coordination. As illustrated in Fig.7, we demonstrate two examples of the *dust equilibrium* in the footprint extraction process. (b) illustrates that when the predicted footprint lies within a valid range, even if the offset is incorrect, correcting it does not change the final mask metrics, as the footprint still overlaps with the ground truth. In (c), the model predicts a roof close to the upper boundary of the ground truth and applies a long offset to shift it towards the lower boundary. Although this is acceptable under mask-based evaluation, correcting the offset directionally might result in the footprint moving outside the valid area, paradoxically leading to a worse evaluation score.

In Tab.XI, we selected prompt LOFT and prompt Cascade LOFT models—both exhibiting relatively poor mask and offset quality—for comparison. We found that when using the predicted mask as the initialization for the roof, even though the offset quality was significantly improved, this did not necessarily lead to an improvement in the footprint results. However, when the roof mask was replaced with the ground truth, the enhancement in offset quality more consistently resulted in improved footprint mask outcomes.

This observation prompts us to reconsider an important question: what role should offset metrics play in off-nadir BFE? Historically, research has predominantly focused on evaluating footprint mask metrics. However, with recent advancements, BFE has entered an era of interactive and controllable extraction methods. For instance, the OBM model simulates manual annotation processes, requiring annotators to manually select an entire building to derive the footprint. Similarly, PolyFootNet not only supports automated extraction of polygonal footprints but also provides explicit offsets between roofs and their corresponding footprints.

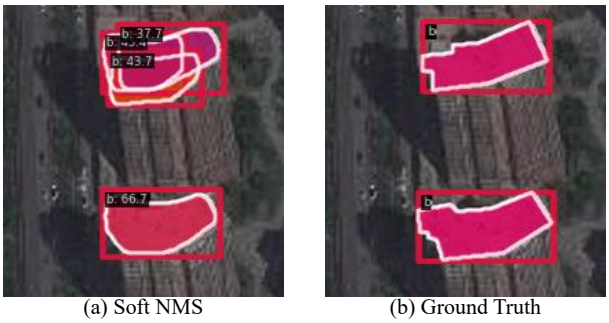
Under these increasingly interactive and controllable scenarios, accurately predicting offsets themselves may become increasingly valuable. This is because offset estimation serves as a critical technical bridge connecting the challenging off-nadir footprint extraction problem with the more mature and widely adopted near-nadir roof extraction approaches. Leveraging continually improving roof extraction methods, such as PolyFootNet, highlights the significance of focusing explicitly on offset metrics, facilitating more reliable and generalizable solutions to off-nadir BFE.

C. External models and multi-solutions of BFE

The use of external prompts for interactive models has been studied in many cases. *e.g.* contrastive Language-Image Pre-training (CLIP) was trained on over 400 million pairs of images and text [44]. When researchers conduct experiments on classifying ImageNet [45]. The researchers found that using a prompt template "A photo of a {*label*}" can directly improve the accuracy by 1.3% compared to using a single category word "{*label*}". In the video recognition zone, FineCLIPER [46] using regenerated video captions for facial activities also improves the quality of the model. In the BFE problem, Li *et al*. [16] discovered that slightly larger building prompts can extract better footprints than entirely fitting box prompts. Another example is the application of the Large Language Model (LLM). RAG in LLM was another key tool to improve the final generated results [47]. *e.g.* Query2doc [48] use pseudo-documents prompting and concatenates them with the original query to improve predicting quality.

BFE problems solved by a promptable model like PolyFootNet must consider similar issues. Researchers who proposed OBM, the former version of PolyFootNet, found that slightly inaccurate prompts can improve the performance of the model [16]. In this paper, PolyFootNet extended the "Prompting Test" to external models and studied three methods to extract footprints.

Except Fig.5 and Tab.IX, more interesting methods must help the model reach and exceed the ceiling performance of using Ground Truth prompts.



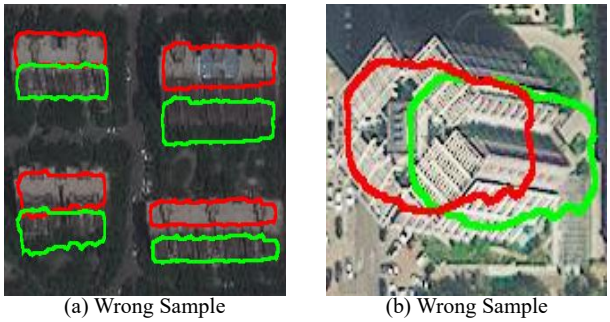
(a) Soft NMS (b) Ground Truth

Fig. 8. Some mistake samples of roof extraction made by HTC and Ground Truth label

D. The benefit of extracting footprints with building segmentation and offset

Considering building-related information was one of the contributions of this paper. For OBM and PolyFootNet, using building segmentation and offset robustly improves the quality of extracted footprints. We carefully analyzed the results of the predicted roof and building segmentation to find an explanation. We found that the edge of a roof and building facade is not that obvious compared with the edge between the building and background in one remote sensing image.

Fig.9 shows some typical samples that predict false roofs due to the "edge problem". Low roof quality but relatively correct offsets finally lead to a poor quality footprint. The situation mentioned above can be improved by using building segmentation as in Fig.1(c).



(a) Wrong Sample (b) Wrong Sample

Fig. 9. Some mistake samples of roof extraction

E. Future work

The contributions of PolyFootNet and the exploration of the multi-solution nature of the BFE problem extend beyond the scope of current experiments. In the future, designing more suitable visual prompts for PolyFootNet may be even more critical than developing new architectures, as effective prompting could significantly enhance model adaptability and performance. Furthermore, the concept of multi-solution reasoning can be applied in reverse to address limitations in earlier datasets. *e.g.*, the BANDON [13] dataset only provides roof and facade masks. By leveraging the multi-solution property, it becomes feasible to infer additional annotation types, such as footprints or building projections. This approach can support the construction of larger-scale off-nadir datasets, thereby facilitating the training of more robust and generalizable models for building footprint extraction.

VII. Conclusion

This paper presents PolyFootNet, a novel model for the BFE problem. Functionally, PolyFootNet can extract polygonal building footprints for off-nadir remote sensing images. Meanwhile, PolyFootNet effectively balances the prediction discrepancies between long and short offsets by introducing the SOFA. In addition, it explores the multi-solution nature of the BFE problem at the model level through combinatorial reasoning of building-related features. The experimental results on three datasets demonstrate the superiority of our method.

References

[1] Q. Li, L. Mou, Y. Sun, Y. Hua, Y. Shi, and X. X. Zhu, "A review of building extraction from remote sensing imagery: Geometrical structures and semantic attributes," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–15, 2024.

[2] J.-Y. Rau, J.-P. Jhan, and Y.-C. Hsu, "Analysis of oblique aerial images for land cover and point cloud classification in an urban environment," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1304–1319, 2015.

[3] B. Li, D. Xie, Y. Wu, L. Zheng, C. Xu, Y. Zhou, Y. Fu, C. Wang, B. Liu, and X. Zuo, "Synthesis and detection algorithms for oblique stripe noise of space-borne remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–14, 2024.

[4] G. Zhou, X. Bao, S. Ye, H. Wang, and H. Yan, "Selection of optimal building facade texture images from uav-based multiple oblique image flows," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1534–1552, 2021.

[5] Y. Zhou, W. Jiang, and B. Wang, "Nesf-net: Building roof and facade segmentation based on neighborhood relationship awareness and scale-frequency modulation network for high-resolution remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 226, pp. 247–266, 2025.

[6] F. Lafarge, X. Descombes, J. Zerubia, and M. Pierrot-Deseilligny, "Structural approach for building reconstruction from a single DSM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 135–147, 2010.

[7] M. Ortner, X. Descombes, and J. Zerubia, "A Marked Point Process of Rectangles and Segments for Automatic Analysis of Digital Elevation Models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 1, pp. 105–119, 2008.

[8] A. Shackelford, C. Davis, and X. Wang, "Automated 2-d building footprint extraction from high-resolution satellite multispectral imagery," in *IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium*, vol. 3, 2004, pp. 1996–1999 vol.3.

[9] H. Sportouche, F. Tupin, and L. Denise, "Building detection by fusion of optical and sar features in metric resolution data," in *2009 IEEE International Geoscience and Remote Sensing Symposium*, vol. 4, 2009, pp. IV–769–IV–772.

[10] J. Inglada, "Automatic recognition of man-made objects in high resolution optical remote sensing images by SVM classification of geometric image features," *ISPRS J. Photogramm. Remote Sens.*, vol. 62, no. 3, pp. 236–248, 2007.

[11] S. Wei, S. Ji, and M. Lu, "Toward automatic building footprint delineation from aerial images using cnn and regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2178–2189, 2020.

[12] W. Li, L. Meng, J. Wang, C. He, G.-S. Xia, and D. Lin, "3d building reconstruction from monocular remote sensing images," in *Int. Conf. Comput. Vis.*, 2021, pp. 12 548–12 557.

[13] C. Pang, J. Wu, J. Ding, C. Song, and G.-S. Xia, "Detecting building changes with off-nadir aerial images," *Science China Information Sciences*, vol. 66, no. 4, p. 140306, 2023.

[14] J. Wang, L. Meng, W. Li, W. Yang, L. Yu, and G.-S. Xia, "Learning to Extract Building Footprints From Off-Nadir Aerial Images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 1294–1301, 2023.

[15] W. Li, H. Yang, Z. Hu, J. Zheng, G.-S. Xia, and C. He, "3d building reconstruction from monocular remote sensing images with multi-level supervisions," *arXiv preprint arXiv:2404.04823*, 2024.

[16] K. Li, Y. Deng, Y. Kong, D. Liu, J. Chen, Y. Meng, J. Ma, and C. Wang, "Prompt-driven building footprint extraction in aerial images with offset-building model," *IEEE Trans. Geosci. Remote Sens.*, pp. 1–1, 2024.

[17] W. Li, Z. Hu, L. Meng, J. Wang, J. Zheng, R. Dong, C. He, G.-S. Xia, H. Fu, and D. Lin, "Weakly supervised 3-d building reconstruction from monocular remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–15, 2024.

[18] Q. Tang, Y. Li, Y. Xu, and B. Du, "Enhancing building footprint extraction with partial occlusion by exploring building integrity," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–14, 2024.

[19] C. Wang, J. Chen, Y. Meng, Y. Deng, K. Li, and Y. Kong, "Sampolybuild: Adapting the segment anything model for polygonal building extraction," *ISPRS J. Photogramm. Remote Sens.*, vol. 218, pp. 707–720, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271624003563

[20] B. Xu, J. Xu, N. Xue, and G.-S. Xia, "Hisup: Accurate polygonal mapping of buildings in satellite imagery with hierarchical supervision," *ISPRS J. Photogramm. Remote Sens.*, vol. 198, pp. 284–296, 2023.

[21] A. Kirillov *et al.*, "Segment anything," in *Int. Conf. Comput. Vis.*, 2023, pp. 4015–4026.

[22] E. A. Nadaraya, "On estimating regression," *Theory Probab. Its Appl.*, vol. 9, no. 1, pp. 141–142, 1964.

[23] G. S. Watson, "Smooth regression analysis," *Sankhyā: The Indian Journal of Statistics, Series A*, pp. 359–372, 1964.

[24] P. R. Srivastava, Y. Wang, G. A. Hanasusanto, and C. P. Ho, "On data-driven prescriptive analytics with side information: A regularized nadaraya-watson approach," *arXiv preprint arXiv:2110.04855*, 2021.

[25] N. Ravi *et al.*, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024.

[26] G. Guo, D. Shao, C. Zhu, S. Meng, X. Wang, and S. Gao, "P2p: Transforming from point supervision to explicit visual prompt for object detection and segmentation," *IJCAI*, 2024.

[27] H. Chen, Y. Huang, H. Huang, X. Ge, and D. Shao, "Gaussianvton: 3d human virtual try-on via multi-stage gaussian splatting editing with image prompting," *arXiv preprint arXiv:2405.07472*, 2024.

[28] L. Ke, M. Ye, M. Danelljan, Y.-W. Tai, C.-K. Tang, F. Yu *et al.*, "Segment anything in high quality," *Adv. Neural Inform. Process. Syst.*, vol. 36, 2024.

[29] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica: the international journal for geographic information and geovisualization*, vol. 10, no. 2, pp. 112–122, 1973.

[30] S. Wei, S. Ji, and M. Lu, "Toward automatic building footprint delineation from aerial images using cnn and regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2178–2189, 2019.

[31] N. Girard, D. Smirnov, J. Solomon, and Y. Tarabalka, "Polygonal building extraction by frame field learning," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 5887–5896.

[32] S. Zorzi, K. Bittner, and F. Fraundorfer, "Machine-learned regularization and polygonization of building segmentation masks," in *Int. Conf. Pattern Recog.* IEEE, 2021, pp. 3098–3105.

[33] G. Christie, R. R. R. M. Abujder, K. Foster, S. Hagstrom, G. D. Hager, and M. Z. Brown, "Learning geocentric object pose in oblique monocular images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 512–14 520.

[34] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural computation*, vol. 3, no. 1, pp. 79–87, 1991.

[35] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," in *Int. Conf. Comput. Vis.*, October 2021, pp. 7262–7272.

[36] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Adv. Neural Inform. Process. Syst.*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

[37] R. Girshick, "Fast r-cnn," in *Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[38] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, 1948.

[39] W. Li, Y. Lai, L. Xu, Y. Xiangli, J. Yu, C. He, G.-S. Xia, and D. Lin, "Omnicity: Omnipotent city understanding with multi-level and multi-view images," *arXiv e-prints*, pp. arXiv–2208, 2022.

[40] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[41] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, June 2018.

[42] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[43] H. Robbins and S. Monro, "A stochastic approximation method," *The annals of mathematical statistics*, pp. 400–407, 1951.

[44] A. Radford *et al.*, "Learning transferable visual models from natural language supervision," in *Int. Conf. Mach. Learn.*, ser. Proc. of Mach. Learn. Res., M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 8748–8763. [Online]. Available: https://proceedings.mlr.press/v139/radford21a.html

[45] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 248–255.

[46] H. Chen, H. Huang, J. Dong, M. Zheng, and D. Shao, "Finecliper: Multi-modal fine-grained clip for dynamic facial expression recognition with adapters," *arXiv preprint arXiv:2407.02157*, 2024.

[47] P. Lewis *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 9459–9474.

[48] L. Wang, N. Yang, and F. Wei, "Query2doc: Query expansion with large language models," in *Conf. Empirical Methods Natural Lang. Process.*, 2023. [Online]. Available: https://openreview.net/forum?id=QH4EMvwF8I

APPENDIX

A. Reasoning of vector-level SOFA

From Eq.1, the Nadaraya-Watson Kernel Regression, we want to derive the vector form of SOFA as Eq.7. The difference between SOFA_v and SOFA_α is the coordinate that is used to describe roof-to-footprint offsets, the Cartesian coordinates for SOFA_v and the polar coordinates for SOFA_α.

In our vector-based derivation, the offset angle serves as the intermediary bridge. Substituting α and α_i in Eq.4 by the unit offsets $\vec{\mathbf{U}}$ and $\vec{\mathbf{u_i}}$, the expression can be rewritten as:

$$\dot{\mathbf{U}} = \sum_{i=1}^{n} \text{softmax}\left(-\frac{1}{2}(\boldsymbol{\rho} - \rho_i)^2\right) \vec{\mathbf{u_i}}, \tag{13}$$

where $\dot{\mathbf{U}}$ represents the fixed unit offset. In Eq.4 and Eq.13, their physical representation is identical: the kernel regression collects a weighted sum for unit offsets, which is conceptually equal to the angles under polar coordinates.

Then, we need to compute the practical offset $\dot{\mathbf{O}}$ by scaling up the united offsets with their original lengths:

$$\dot{\mathbf{O}} = \boldsymbol{\rho}^\top \dot{\mathbf{U}}. \tag{14}$$

For Eq.14, we add a learnable weight w and a look-longer attention mask $\boldsymbol{\mathcal{M}}$ with the same reasons as Eq.5 and Eq.6. From here, we derive the SOFA_v:

$$\begin{aligned} \dot{\mathbf{O}} &= \text{SOFA}_v(\boldsymbol{\rho}, \vec{\mathbf{U}}) \\ &= \boldsymbol{\rho}^\top \sum_{i=1}^{n} \text{softmax}\left(-\frac{1}{2}(w \times \boldsymbol{\mathcal{M}}(\boldsymbol{\rho} - \rho_i))^2\right) \vec{\mathbf{u_i}}. \end{aligned} \tag{15}$$

Since SOFA_v is an attention and summation mechanism specifically designed for unit vectors, the weighted sum of vectors, even when the weights sum to one, does not necessarily yield a resulting vector with unit length. Coincidentally, this slight change in magnitude aligns well with a certain pattern of length errors commonly observed in offset predictions across various models, thereby providing a degree of correction for such errors. This property is not possessed by SOFA_α.

B. How we design DS-BCE loss?

In the initial experiments, we trained the vector map with a standard BCE loss, but the model failed to grasp the notion of keypoints—hardly surprising given the class imbalance. For a single-building prompt, the roof typically contains only 4 – 5 keypoints, and each keypoint corresponds to just a 3 × 3 patch (9 positive pixels) in the ground truth; every other pixel in the image is negative. On a 256 × 256 feature map, the positives therefore make up only about 0.06% of all samples. Faced with such extreme imbalance, the network quickly becomes "lazy," defaulting to predicting every pixel as a non-keypoint.

To alleviate the extreme class imbalance, we limited the loss calculation to the tight bounding rectangle of the user-provided prompt, thereby discarding large areas filled only with negative pixels. As training progressed, however, a new issue surfaced: this simple cropping strategy produced feature maps with a peculiar "periodic grid" artifact. We have not yet conducted an in-depth investigation, but by analogy with other polygon-extraction models that perform instance segmentation (*e.g.*, SAMPolyBuild), we suspect the following mechanism: in those models, the proposal network samples a slightly different bounding box for each instance on every training step, injecting stochasticity. For PolyFootNet, deliberately introducing a similar randomness during training may help suppress the grid-like patterns we observed.

Therefore, the DS-BCE loss was proposed. Based on the descriptions, the design of DS-BCE loss can be divided into three steps. First, for each prompt area, we get 4 random integers ranging from 0 to 3. Each integer represents the enlarged size for the four edges of the prompt area. Then, the new area was used to crop the model output and the ground truth label. Finally, we use the standard BCE loss function to compute the loss between the cropped output and the label.

Kai Li (Graduate Student Member, IEEE) received a bachelor's degree in engineering, spatial information and digital technology from UESTC, Chengdu, China, in 2021. He is currently pursuing a PhD degree with UCAS, Beijing, China, supervised by Zhongming Zhao and Yu Meng. He also joined CityU of Hong Kong as a joint PhD student, and his supervisor is Xiangyu Zhao. His research interests include remote sensing, computer vision, and machine learning.

Yupeng Deng received the Ph.D. degree from the Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, China, in 2023.Now, he is a Post-Doctoral Researcher Supervised by Jianhua Gong. He is specialized in remote sensing and change detection.

Jingbo Chen (Member, IEEE) received the Ph.D.degree in cartography and geographic information systems from the Institute of Remote Sensing Applications, Chinese Academy of Sciences,Beijing, China, in 2011.
He is currently an Associate Professor with the Aerospace Information Research Institute, Chinese Academy of Sciences. His research interests cover intelligent remote sensing analysis, integrated application of communication, navigation, and remote sensing.

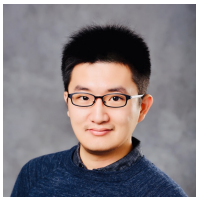
Yu Meng received the Ph.D. degree in signal and information processing from the Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, China, in 2008.
She is currently a professor at the Aerospace Information Research Institute, Chinese Academy of Sciences. Her research interests include intelligent interpretation of remote sensing images,remote sensing time-series signal processing, and big spatial-temporal data application.
Dr Meng serves as an editor and board member of the National Remote Sensing Bulletin, Journal of Image and Graphics.

Zhihao Xi received the B.S. degree from the Wuhan University of Technology, Wuhan, China, in 2019, and the Ph.D. degree from the Aerospace Information Research Institute, Chinese Academy of Sciences (CAS), Beijing, China, in 2024. He is currently an Assistant Professor with the Aerospace Information Research Institute, CAS. His research interests include computer vision, domain adaptation, and remote sensing image interpretation.

Junxian Ma received the bachelor's degree from Peking University, Beijing, China, in 2020. He is currently pursuing the Ph.D. degree with the University of Chinese Academy of Sciences, Beijing.He focuses on image generation and conditional video generation.

Chenhao Wang received the bachelor's degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2022. He is currently pursuing the Ph.D. degree with the University of Chinese Academy of Sciences, Beijing, China.His research focuses on building extraction from remote sensing images.

Maolin Wang Maolin Wang received an M.Phil degree from the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Sichuan, China, 2021. He is currently pursuing a Ph.D. degree in data science at the City University of Hong Kong, HKSAR, China. His research interests include tensor, graph neural networks, and their applications.



Xiangyu Zhao is an assistant professor of the school of data science at City University of Hong Kong (CityU). Prior to CityU, he completed his PhD (2021) at MSU, MS (2017) at USTC and BEng (2014) at UESTC. His current research interests include data mining and machine learning. He has published more than 100 papers in top conferences (e.g., KDD, WWW, AAAI, SIGIR, IJCAI, ICDE, CIKM, ICDM, WSDM, RecSys, ICLR) and journals (e.g., TOIS, SIGKDD, SIGWeb, EPL, APS). His research has been awarded ICDM'22 and ICDM'21 Best-ranked Papers, Global Top 25 Chinese New Stars in AI (Data Mining). He serves as top data science conference (senior) program committee members and session chairs (e.g., KDD, WWW, SIGIR, IJCAI, AAAI, ICML, ICLR), and journal guest editors and reviewers (e.g., TKDE, TKDD, TOIS, TIST, CSUR, Frontiers in Big Data). He serves as the organizers of Tutorial Co-Chair@APWeb-WAIM'24, DRL4KDD@KDD'19/WWW'21, DRL4IR@SIGIR'20-22/CIKM'23, DLP@RecSys'23, RWGM@WWW'24, and a lead tutor at KDD'23, WWW'21/22/23, IJCAI'21/23 and WSDM'23, etc. More information about him can be found at https://zhaoxyai.github.io/.