applied sciences

MDPI

Article

Building Polygon Extraction from High-Resolution Remote Sensing Imagery Using Knowledge Distillation

Haiyan Xu [1,2], Gang Xu [1,3,*], Geng Sun [3], Jie Chen [3] and Jun Hao [1,4]

[1] Zhejiang College of Security Technology, No. 2555 Ouhai Main Road, Wenzhou 325016, China; 20096347@zjcst.edu.cn (H.X.)
[2] Wenzhou Key Laboratory of Natural Disaster Remote Sensing Monitoring and Early Warning, No. 2555 Ouhai Main Road, Wenzhou 325016, China
[3] School of Geosciences and Info-Physics, Central South University, Changsha 410083, China
[4] Wenzhou Collaborative Innovation Center for Space-borne, Airborne and Ground Monitoring Situational Awareness Technology, No. 2555 Ouhai Main Road, Wenzhou 325016, China
* Correspondence: 20096342@zjcst.edu.cn; Tel.: +86-577-88350183

Abstract: Building polygons plays an important role in urban management. Although leveraging deep learning techniques for building polygon extraction offers advantages, the models heavily rely on a large number of training samples to achieve good generalization performance. In scenarios with small training samples, the models struggle to effectively represent diverse building structures and handle the complexity introduced by the background. A common approach to enhance feature representation is fine-tuning a pre-trained model on a large dataset specific to the task. However, the fine-tuning process tends to overfit the model to the task area samples, leading to the loss of generalization knowledge from the large dataset. To address this challenge and enable the model to inherit the generalization knowledge from the large dataset while learning the characteristics of the task area samples, this paper proposes a knowledge distillation-based framework called Building Polygon Distillation Network (BPDNet). The teacher network of BPDNet is trained on a large building polygon dataset containing diverse building samples. The student network was trained on a small number of available samples from the target area to learn the characteristics of the task area samples. The teacher network provides guidance during the training of the student network, enabling it to learn under the supervision of generalization knowledge. Moreover, to improve the extraction of buildings against the backdrop of a complex urban context, characterized by fuzziness, irregularity, and connectivity issues, BPDNet employs the Dice Loss, which focuses attention on building boundaries. The experimental results demonstrated that BPDNet effectively addresses the problem of limited generalization by integrating the generalization knowledge from the large dataset with the characteristics of the task area samples. It accurately identifies building polygons with diverse structures and alleviates boundary fuzziness and connectivity issues.

Keywords: building extraction; knowledge distillation; building vector polygons; high-resolution remote sensing imagery

Citation: Xu, H.; Xu, G.; Sun, G.; Chen, J.; Hao, J. Building Polygon Extraction from High-Resolution Remote Sensing Imagery Using Knowledge Distillation. *Appl. Sci.* **2023**, *13*, 9239. https://doi.org/10.3390/app13169239

Academic Editor: Dimitris Mourtzis

Received: 26 July 2023
Revised: 7 August 2023
Accepted: 9 August 2023
Published: 14 August 2023

1. Introduction

Buildings are essential components of cities and serve as the primary places for residential and commercial activities [1]. Building polygons refer to the vector line information representing the planar outline of buildings when viewed from an overhead perspective. They play a crucial role in various fields such as urban planning [2], smart cities [3], 3D modeling [4], and disaster assessment [5]. Therefore, there is significant interest in extracting building polygon information rapidly and accurately.

In general, the process of extracting the polygon of a building is to first extract the raster mask, then use the texture, shape, and structure to design conversion rules, and finally convert the raster mask to a vector polygon within the constraints of the rules [6]. For

example, firstly, remote sensing imagery is initially interpreted manually or through semi-automatic methods to extract the raster mask [7]. Secondly, mathematical morphology [8], edge detection operators such as Canny and Sobel [9,10] and Hough transform [11] are employed to extract building features, which are used to design rules for the transformation process. Finally, a rule set is used to determine the boundary positions of building raster masks and convert them into vector polygons. However, these methods can only extract accurate building polygons when predefined rules are satisfied. Besides, the methods are complex, resource-intensive, and have high labor costs, resulting in low automation levels [12].

In recent years, deep learning techniques have been widely applied in the field of remote sensing and have achieved remarkable results in building extraction [13]. For instance, using high-resolution remote sensing imagery, accurate building regions can be extracted using techniques such as semantic segmentation [14–16], object detection [17–19], and instance segmentation [20,21]. Building polygon extraction techniques have also gradually shifted towards deep learning-based methods. The basic workflow of such methods involves first obtaining a raster mask for the building class using deep learning-based semantic segmentation techniques and then vectorizing the raster results. Furthermore, researchers have integrated strategies such as multi-scale information [22], attention mechanisms [23], and multi-feature fusion [24] to extract more precise and effective features from high-resolution imagery to obtain more accurate raster masks. In the vectorization process, some studies have incorporated building corner and edge information to optimize building boundaries [25,26] or used Frame-Field to align the building segmentation maps for better edge representation [27,28]. To achieve end-to-end automatic building polygon extraction, some methods adopt the idea of multi-task learning by combining building segmentation with the vectorization process [29]. Certain approaches first detect the bounding boxes of building objects and then use LSTM to predict the building corners [30,31]. Additionally, some researchers treat building polygons as graphs [32], detecting building vertices and computing their adjacent connections to obtain polygonal contours.

However, deep learning methods require a large number of well-annotated labeled samples for support [33], and obtaining building polygon labels through manual annotation incurs significant labeling costs [34]. In practical applications, government agencies can only provide a small number of building polygon-labeled samples due to data confidentiality requirements. Training with a small number of samples leads to insufficient model generalization, making it difficult to handle the challenges of extracting diverse building polygons against complex backgrounds [35]. It is worth noting that some large and high-quality building extraction datasets are publicly available [26,36,37]. These datasets include building samples from different regions, encompassing various building styles, structures, and diverse backgrounds. Therefore, it is possible to extract generalization knowledge from these abundant and diverse building samples. However, directly applying models trained on other datasets to predict the target area may not yield satisfactory results.

To account for the building characteristics in the target area, it is possible to incorporate samples from the target area during the training process. A commonly used approach in current research is to first pre-train the model on publicly available datasets and then fine-tune the model on samples from the target area [38–41]. While this approach allows the model to learn the specific characteristics of buildings in the target area, it does not effectively inherit the generalization knowledge learned from the pretrained model on public datasets. Knowledge distillation techniques [42,43] offer a solution by pretraining a teacher neural network on a large dataset. During the training process, the knowledge learned by the teacher network is utilized to guide the student neural network, thereby improving the accuracy of the student network. For example, Wang et al. addressed the problem of insufficient labels in remote sensing scene classification by employing self-distillation to learn complex scene knowledge in diverse backgrounds [44]. Nabi et al. used knowledge distillation to extract visual knowledge in complex urban scenes [45]. By applying knowledge distillation, the model not only retains the generalization knowledge

learned from the public datasets but also leverages the supervision from the limited samples in the target area to enhance the model's adaptability to diverse types of buildings.

Therefore, this paper proposes the Building Polygon Distillation Net (BPDNet) to accurately extract building polygons in the target area with limited samples. The network consists of two structurally identical sub-networks known as the teacher network and the student network. The teacher network is trained on a large building polygon dataset that contains structurally diverse building samples, while the student network is trained on a small number of available samples from the target area. During the training process of the student network, the teacher network provides guidance, allowing the student network to access generalization knowledge. Furthermore, to address the issue of fuzzy, irregular, and fragmented building extraction caused by complex urban backgrounds, we employed the Dice Loss [46] to encourage the model to focus more on building boundaries. This helps alleviate the problem of building fragmentation caused by blurry boundaries.

The primary contributions of this work can be summarized as follows:

1. We propose BPDNet for extracting building polygons in the area with limited high-resolution remote sensing image samples. This network enables the model to inherit the generalization knowledge from the large dataset via knowledge distillation;
2. BPDNet employs Dice Loss to improve the extraction of buildings against the backdrop of complex urban context, characterized by fuzziness, irregularity, and connectivity issues.

The paper is structured as follows: Section 1 introduces the background to the methodology and the problem addressed in this paper; Section 2 describes the study area, the external dataset used, and the process and methods used to process the data; Section 3 presents the details of the methodology; Section 4 presents the experimental design, parameter settings, and analysis of the results; Section 5 provides a discussion of the methodology; and, finally, the conclusions are presented in Section 6.

2. Study Area and Data

2.1. Remote Sensing Images and Building Vector Data

Wenzhou, located in Zhejiang Province, China, is an important coastal commercial and regional center in southeastern China. In recent years, the urban area of Wenzhou has been continuously expanding. The intelligent and automated extraction of building polygons will effectively enhance the efficiency of building supervision. The remote sensing imagery and building vector data used in this study are derived from six regions within the urban area of Wenzhou, captured by unmanned aerial vehicles. Each region's imagery has a size of 20,480 $\times$ 20,480 pixels with a spatial resolution of 0.2 m, including RGB bands. The study area and the corresponding imagery are shown in Figure 1a. In the high-resolution remote sensing imagery, various buildings with different shapes, arrangements, and heights can be observed in the six regions, as shown in Figure 1b. Moreover, the imagery exhibits challenges in extracting building polygons due to significant issues such as building tilts and shadow occlusions, which can cause confusion between building rooftops, facades, and the background.

In the data pre-processing stage, we cropped six remote sensing images into 2400 patches of 1024 $\times$ 1024 size and randomly divided the training and test sets according to the ratio of 7:3. The building contour label data is the building vector data corresponding to the image range, as shown in the sample in Figure 2.

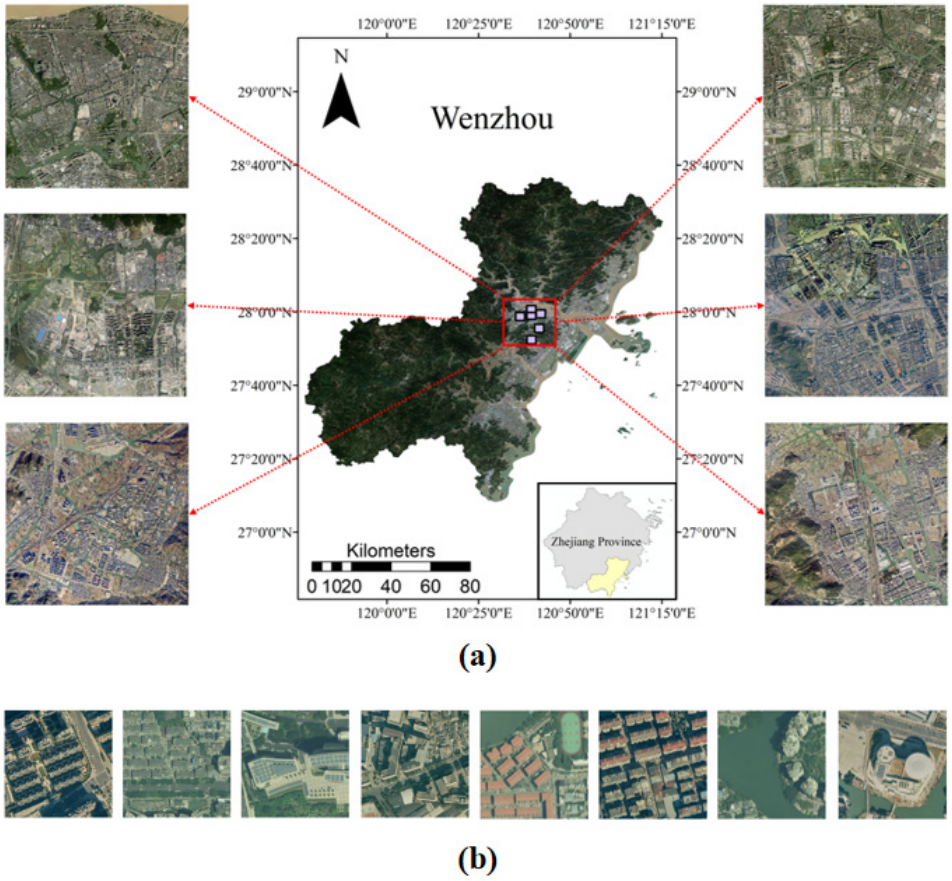

Figure 1. The study area and image data: (**a**) illustrates the study area and the corresponding image data; (**b**) illustrates the diverse types of buildings within the study area.

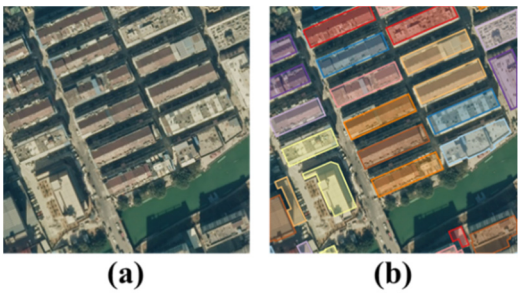

Figure 2. Example of a sample from the study area: (**a**) shows a patch from the remote sensing image; (**b**) shows the corresponding building polygon labels of (**a**).

2.2. WHU-Mix Dataset

In response to the lack of data diversity and poor label quality of current building datasets, the WHU-Mix dataset [47] collects images and polygon vector labels of over 754k buildings from around the world. The training set of this dataset contains 43,727 images, and the test set contains 8402 images from another five cities on five continents. The WHU-Mix dataset integrates data from the WHU dataset [37], Crowd AI [48], Open AI [49], SpaceNet [50], and Inria [36] datasets, with manual corrections for offsets and missing data. Sample data from the dataset are shown in Figure 3. To better capture the diversity of real-world scenarios and consider the wide range of building variations, we trained the teacher model based on the WHU-Mix dataset. This allowed us to transfer the knowledge of generalization to the target study area effectively.

Figure 3. Sample examples from the WHU-Mix Dataset.

3. Methods

3.1. Overall Architecture

Figure 4 illustrates the overall architecture of BPDNet, which consists of two structurally identical networks: the teacher network and the student network. The teacher network is trained on the WHU-Mix dataset for building polygon extraction. Once the teacher network is trained and reaches convergence, its model parameters are frozen. The student network is trained on the training set constructed from six images of the Wenzhou area and distilled using the trained teacher network. Therefore, the student model is capable of integrating the generalization knowledge from the large dataset and the feature distribution knowledge of the task area samples to address the issue of insufficient generalization of the building polygon extraction model caused by a small number of samples in the task area.

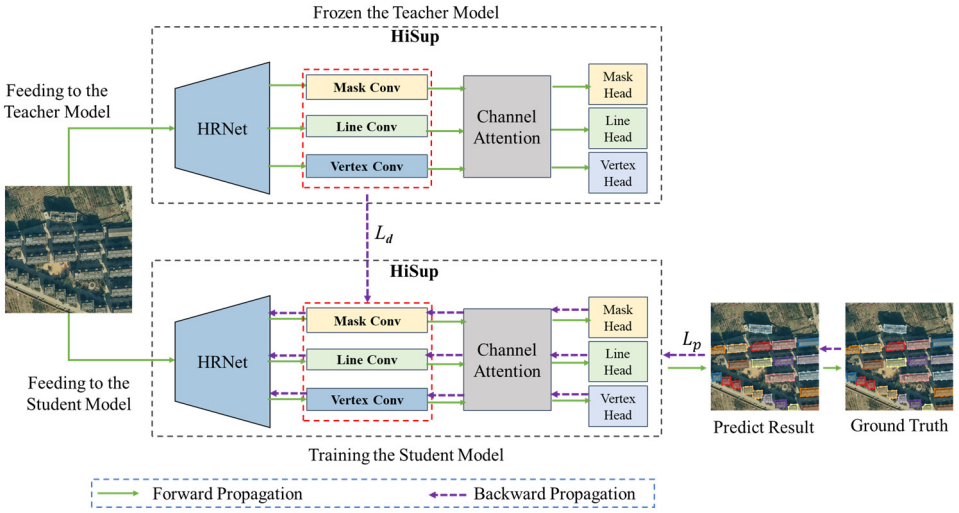

Figure 4. Overall Architecture of BPDNet. L_d denotes the distillation loss and L_p denotes the building polygon prediction loss. Mask Conv, Line Conv, and Vertex Conv represent the convolution operations in the Mask branch, Line branch, and Vertex branch, respectively. Mask head, Line head, and Vertex head represent the prediction head in the mask branch, line branch, and vertex branch, respectively.

The teacher network and the student network are both constructed based on HiSup [49]. HiSup serves as the baseline model for our method, consisting of the HRNet [51] feature encoder, Channel Attention module, and three building representation branches. Details of HiSup and our improvements to it are described in Section 3.2. The teacher network was trained on the WHU-Mix dataset, so during the training of the student network, the parameters of the teacher network are no longer updated to guide the training of the student network. In the training process of the student network, the remote sensing images from the task area are input to both the student network and the teacher network with frozen parameters. The forward propagation data flow of the network is shown by the solid green arrows in Figure 4. The gradients of the network parameters are backpropagated through the purple dashed arrows in Figure 4. The loss of the student network consists of two parts: distillation loss and building polygon prediction loss. The distillation loss represents the difference between the output features of the student network and the

teacher network in the three building representation branches, encouraging the student model to learn generalization knowledge from the teacher model and improve its ability to extract building polygons with diverse structures. The building polygon prediction loss represents the difference between the predicted building polygons by the student network and the ground truth building polygon labels, guiding the student model to learn the feature distribution knowledge of the data in the task area and enhance its accuracy in extracting building polygons in the task area.

3.2. Baseline Model: HiSup

This paper adopts HiSup as the baseline model for extracting building polygons from remote sensing images. The model structure is illustrated in Figure 5. HiSup consists of three feature learning branches, including the Mask Branch, Line Branch, and Vertex Branch. Additionally, HiSup incorporates modules for multi-scale feature extraction, channel attention, boundary enhancement, and polygon construction and simplification. Specifically, for an input remote sensing image $I \in R^{3\times H\times W}$, HRNet is employed as the multi-scale feature extractor to obtain the feature map $F \in R^{3\times H_S\times W_S}$. Here, $H_S = H/S$ represents the length of feature map F, and $W_S = W/S$ represents its width, with S being the downsampling factor. After obtaining the image feature map F through the multi-scale feature extractor, F is fed into three distinct branches to learn the vertex feature F_{ver}, line feature F_{line}, and semantic mask feature F_{seg} of buildings. Each branch contains three consecutive processing units, including 3×3 convolutional layers, a batch normalization layer [52], and a ReLU [53] layer in each processing unit. Specifically, the vertex feature F_{ver} and line feature F_{line} are derived from mid-level image features, while the semantic mask feature F_{seg} is obtained from high-level image features. Subsequently, the channel attention module integrates these three features to generate the vertex prediction heat map M_H, vertex distance offset field M_O, line attraction field [29] prediction map M_A, and building a semantic mask prediction map M_S. Next, by utilizing the line attraction field M_A, and image feature F to impose boundary constraints on M_S, a more regular segmentation result is obtained. Finally, the vertex, line, and semantic prediction results are utilized to construct vector polygons, which are further simplified to obtain the final building polygon extraction results.

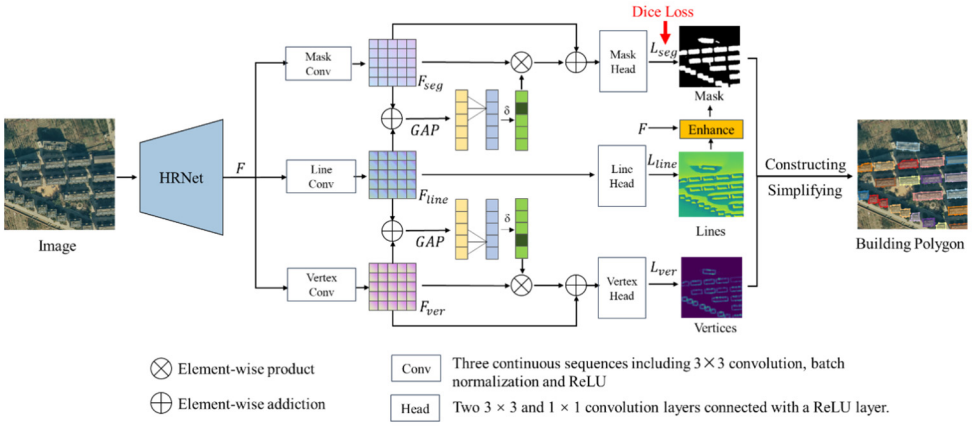

Figure 5. The architecture of the improved HiSup. F denotes the image features from HRNet. F_{seg} denotes the semantic features. F_{line} denotes the line features. F_{ver} denotes the vertex features. δ represents the sigmoid activation function. GAP represents the global average pooling. L_{seg}, L_{line}, and L_{ver} represent the loss function in the Mask branch, Line branch, and Vertex branch, respectively.

To further reduce the influence of complex backgrounds on building boundaries, we improved HiSup. In the boundary enhancement module, Dice Loss was incorporated to alleviate the issue of building adhesion. This modification aims to improve the segmentation performance by effectively separating buildings from their surrounding environment, as shown in Figure 5.

3.2.1. Channel Attention Module

HiSup utilizes the channel attention mechanism [54] to enhance the feature representation of vertex maps F_{ver} and semantic features F_{seg}, thereby improving the accuracy and consistency of building shape prediction. The computation method is as follows:

$$F_{seg}^{e} = \delta\left(C1D\left(GAP\left(F_{line} + F_{seg}\right)\right)\right) \times F_{seg} + F_{seg} \tag{1}$$

$$F_{ver}^{e} = \delta(C1D(GAP(F_{line} + F_{ver}))) \times F_{ver} + F_{ver} \tag{2}$$

where $C1D(\cdot)$ represents one-dimensional convolution, $GAP(\cdot)$ represents global average pooling, and $\delta(\cdot)$ represents the sigmoid activation function.

3.2.2. Boundary Enhancement

Due to the widespread irregularity and fuzzy boundaries of building shapes in the semantic mask results, HiSup employs the boundary enhancement module to utilize the line attraction field results M_A to constrain the semantic mask results M_S, thereby making the building shapes in the semantic mask more regular and the boundaries clearer. Specifically, firstly, M_A is concatenated with the feature map F extracted by the backbone network to obtain the fused feature map M_{AF}, incorporating the line constraint information into F. Then, M_{AF} is fed into the semantic mask classification head to predict the semantic mask image M_{AS}. Subsequently, the semantic mask loss function is employed to calculate the discrepancy between M_{AS} and the semantic mask labels. Finally, through backpropagation, M_A receives additional supervision from the semantic mask labels, enabling it to have a better representation. After further gradient backpropagation, the backbone network enhances the expression of building shapes in F to facilitate the generation of M_A. Consequently, the backbone network learns the correct building shapes guided by M_A.

To further reduce the merging of buildings and confusion with the background, we improved the loss function of HiSup. Dice Loss [46] was added to the binary cross-entropy-based segmentation loss L_{seg} to allow the model to focus more on the boundaries of building targets and reduce the adhesion between building polygons. The segmentation loss L_{seg} of the model can be represented as follows:

$$L_{seg}(\theta) = BCE\left(M_{AS}, M_S^{GT}\right) + BCE\left(M_S, M_S^{GT}\right) + \theta L_{dice} \tag{3}$$

where BCE represents the binary cross-entropy loss, M_{AS} represents the semantic mask result guided by the line features, M_S^{GT} represents the semantic mask label for the building, L_{dice} represents the Dice Loss, θ is the coefficient. Dice Loss can be represented as follows:

$$L_{dice} = 1 - \frac{2TP}{2TP + FP + FN} \tag{4}$$

where TP represents the number of true positive pixels, FP represents the number of false positive pixels, and FN represents the number of false negative pixels.

3.3. Polygon Construction and Simplification

After extracting building features, it is necessary to construct the polygonal outlines of the buildings using post-processing methods. First, the semantic mask image M_S is filtered using a given threshold value $\varepsilon \in (0,1)$(reference HiSup sets $\varepsilon = 0.6$) to obtain the resulting image S, which contains n building polygons. Then, local non-maximum suppression is applied to filter out non-key vertices in the vertex prediction heatmap M_H, and based on S, the vertices located on the boundary pixels of building polygons are connected to construct the initial building outlines. Finally, the building outlines are simplified, with a focus on the vertices of the buildings. The rule for vertex simplification is as follows: if the distance between two vertices of the same building polygon is smaller

than a threshold value τ (reference HiSup sets τ = 5), the midpoint of the line connecting the two points is taken as the simplified vertex, resulting in the final building polygon.

3.4. Knowledge Distillation

In this study, knowledge distillation was employed to learn generalization knowledge from the teacher network trained on a large dataset. The specific process of knowledge distillation is illustrated in Figure 6. The input images are passed through the HRNet feature extractors of both the teacher network and the student network, resulting in image features denoted as F^T and F^S, respectively. For F^T in the teacher network, it is separately inputted into the Mask Branch, Line Branch, and Vertex Branch. The feature F^T in each branch undergoes three consecutive processing units, where each unit consists of a 3×3 convolutional layer, a batch normalization (BN) layer, and a rectified linear unit (ReLU) activation layer. The output of the last processing unit in each branch is, respectively, the semantic mask feature F^T_{seg}, the line feature F^T_{line}, and the vertex feature F^T_{ver}. Similarly in the student network, the feature F^S is separately inputted into the Mask Branch, Line Branch, and Vertex Branch to obtain the semantic mask feature F^S_{seg}, the line feature F^S_{line}, and the vertex feature F^S_{ver} in the student network. Due to the well-trained teacher network on the WHU-Mix dataset, it possesses the ability and knowledge to extract structurally diverse building polygons, including the features of building semantic masks, building lines, and building vertices. Therefore, we employed the CWD (Channel-wise Knowledge Distillation) [43] to distill these capabilities of the teacher network regarding building outlines into the student network. Taking the semantic mask feature as an example, we aligned the feature distributions of each channel in F^S_{seg} from the student network with the corresponding channel in F^T_{seg} from the teacher network. Similarly, both F^S_{line} and F^S_{ver} need to be aligned with F^T_{line} and F^T_{ver} in the same manner as described for F^S_{seg}. CWD, as it distills knowledge in the channel dimension, can effectively utilize the knowledge contained in each channel, making it suitable for dense prediction tasks such as building polygon prediction in this paper.

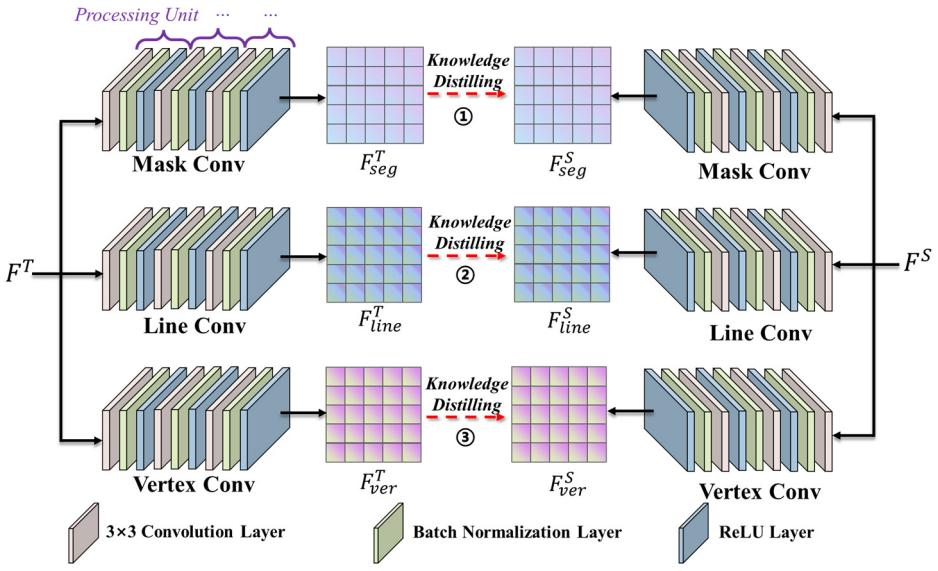

Figure 6. The way of knowledge distillation. F^T and F^S denote the image features from HRNet of the teacher network and the student network, respectively. F^T_{seg}, F^T_{line}, and F^T_{ver} represent the features map from the Mask Branch, Line Branch, and Vertex Branch of the teacher network, respectively. F^S_{seg}, F^S_{line}, and F^S_{ver} represent the features map from the Mask Branch, Line Branch, and Vertex Branch of the student network, respectively.

CWD minimizes the asymmetry Kullback–Leibler (KL) divergence between the channel-wise soft probability maps of the teacher and student networks to align the feature distributions of each channel. Therefore, the distillation loss can be defined as follows:

$$L_d(T) = \frac{T^2}{C}\sum_{c=1}^{C}\sum_{i=1}^{W\cdot H}\phi\left(F_{c,i}^{T}\right)\cdot log\left[\frac{\phi\left(F_{c,i}^{T}\right)}{\phi\left(F_{c,i}^{S}\right)}\right] \tag{5}$$

where $c = 1, 2, \ldots, C$ indexes the channel; and i indexes the spatial location of a channel. W and H are the width and height of the feature map respectively. T is a hyperparameter. F^T denotes the feature map from the teacher network and F^S denotes the feature map from the student network. $\phi(\cdot)$ is used to convert the activation values into a probability distribution as below:

$$\phi(T) = \frac{exp\left(\frac{F_{c,i}}{T}\right)}{\sum_{i=1}^{W\cdot H} exp\left(\frac{F_{c,i}}{T}\right)} \tag{6}$$

where $c = 1, 2, \ldots, C$ indexes the channel; and i indexes the spatial location of a channel. W and H are the width and height of the feature map respectively. T is the temperature hyper-parameter. The probability becomes softer if we use a larger T, meaning that we focus on a wider spatial region for each channel.

3.5. Loss Function

The loss function of BPDNet consists of two parts: the building polygon prediction loss L_p and the distillation loss L_d. The loss function for building polygon contour prediction is composed of three components: the semantic mask branch loss function L_{seg}, the line attraction field prediction branch loss function L_{line}, and the vertex prediction branch loss function L_{ver}. L_{seg} as shown in Equation (3), and L_{line} and L_{ver} can be expressed as follows:

$$L_{line} = l_1\left(M_A, M_A^{GT}\right) \tag{7}$$

where l_1 represents the l-penalized loss, which corresponds to the absolute deviation loss. M_A^{GT} refers to the line attraction field generated from the ground truth labels of the building polygon in the image.

$$L_{ver}(\alpha,\beta) = \alpha BCE\left(M_H, M_H^{GT}\right) + \beta l_1\left(M_O\cdot M_H^{GT}, M_O^{GT}\cdot M_H^{GT}\right) \tag{8}$$

where α and β are coefficients, M_H represents the predicted heat map for building vertices, M_O corresponds to the short-distance offset field for vertices, and M_H^{GT} and M_O^{GT} refer to the ground truth heat map and short-distance offset field for vertices generated from the actual labels of the building polygon in the image.

The loss function for building polygon contour prediction can be expressed as follows:

$$L_p(\lambda_1,\lambda_2,\lambda_3,\alpha,\beta,\theta) = \lambda_1 L_{seg}(\theta) + \lambda_2 L_{line} + \lambda_3 L_{ver}(\alpha,\beta) \tag{9}$$

where λ_1, λ_2, λ_3, α, β, and θ are coefficients.

The overall loss function consists of the building polygon contour prediction loss and the distillation loss and can be represented as follows:

$$L_{total}(\lambda_1,\lambda_2,\lambda_3,\alpha,\beta,\theta,\eta,T) = \lambda_1 L_{seg}(\theta) + \lambda_2 L_{line} + \lambda_3 L_{ver}(\alpha,\beta) + \eta L_d(T) \tag{10}$$

where η represents the coefficient hyperparameter for L_d, and T represents the temperature hyperparameter.

4. Results

4.1. Experimental Design

To validate the effectiveness of the proposed BPDNet model, we conducted the following experiments. By comparing the results of each model on evaluation metrics, we were able to quantitatively analyze the strengths and weaknesses of each model in comparison to BPDNet. Additionally, we performed a visual analysis of the predicted results of each model to qualitatively assess the building extraction performance of each model.

(1) Frame-field [27] is a building polygon extraction method proposed in 2020. It aligns the predicted frame field with the true outline. We implemented the model based on the code of the original paper. In our experiments, the frame-field was trained on a training set composed of samples from Wenzhou and was tested on a test set constructed from samples from Wenzhou;
(2) Model_1 (Baseline) was trained on a training set composed of samples from Wenzhou using the HiSup model. It was then tested on a test set constructed from samples from Wenzhou. This serves as the baseline model for comparison;
(3) Model_2 (w/o Finetune) uses the WHU-Mix dataset to train the HiSup model without using fine-tuning but directly on a test set constructed from the Wenzhou sample.
(4) Model_3 (w/Fine-tune) was trained on the WHU-Mix dataset using the HiSup model. Subsequently, fine-tuning was performed on the training set constructed from samples from Wenzhou. Finally, Model_3 was tested on the test set composed of samples from Wenzhou;
(5) Model_4 represents the proposed BPDNet method. It involves training the HiSup model on the WHU-Mix dataset, which serves as the teacher model. The CWD (Collaborative Weight Distillation) method was then employed to guide the training of the student model on the training set constructed from samples of Wenzhou. Finally, Model_4 is evaluated on the test set composed of samples from Wenzhou.

4.2. Evaluation Metrics

For instance, for segmentation tasks such as building contour extraction, it is important to evaluate not only pixel-level segmentation performance but also instance-level building extraction effectiveness. To quantify the performance of different models, we utilize the following metrics: Overall Accuracy (OA), IoU, Precision for IoU threshold > 0.5 called P_{50}^{IoU}, Precision for Boundary IoU [55] threshold > 0.5 called $P_{50}^{Boundary}$, Recall for IoU threshold > 0.5 called R_{50}^{IoU} and Recall for Boundary IoU threshold > 0.5 called $R_{50}^{Boundary}$. Among the metrics used for evaluation, OA and IoU are employed to assess the pixel-level effectiveness of building extraction. P_{50}^{IoU} is used to evaluate the precision of extracting building polygon instances, while $P_{50}^{Boundary}$ is utilized to evaluate the precision of the boundaries of the extracted building polygon instances. R_{50}^{IoU} is used to evaluate the recall of extracting building polygon instances, while $R_{50}^{Boundary}$ is utilized to evaluate the recall of the boundaries of the extracted building polygon instances. The evaluation metrics are designed such that larger values indicate better performance of the models. The specific calculation methods for these evaluation metrics are as follows:

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where TP indicates the number of pixels predicted to be true positives, TN indicates the number of pixels predicted to be true negatives, FP indicates the number of pixels predicted to be false positives, and FN indicates the number of pixels predicted to be false negatives.

$$IoU(X, Y) = \frac{X \cap Y}{X \cup Y} \tag{12}$$

where X represents the predicted results, and Y represents the ground truth labels. IoU measures the intersection-to-union ratio between the predicted building polygon and the ground truth building polygon. A higher IoU value indicates a better match between the predicted building polygon and the ground truth, signifying more accurate building prediction results.

The Boundary IoU metric provides a reasonable measure to assess the accuracy of polygon boundaries. It helps to reveal the precision of the boundaries of the detected polygon instances in a meaningful way. For two polygon instances X and Y, Boundary IoU calculates the IoU only for pixels that are within a distance d of the boundaries of the two polygon instances. It focuses on evaluating the IoU specifically for the pixels that lie within a specified distance d from the boundaries of the polygon instances X and Y. Define the boundaries of X and Y as X_d and Y_d. Similar to IoU, a larger value of Boundary IoU indicates better extraction performance. The calculation formula for Boundary IoU can be represented as follows:

$$Boundary\ IoU(X, Y) = \frac{|(X_d \cap X) \cap (Y_d \cap Y)|}{|(X_d \cup X) \cup (Y_d \cup Y)|} \tag{13}$$

The value of parameter d in the formula is set based on reference [55], where d = 0.2.

$$P_{50}^{IoU} = \frac{TP_{50}^{IoU}}{TP_{50}^{IoU} + FP_{50}^{IoU}} \tag{14}$$

where TP_{50}^{IoU} represents the number of true positive pixels within instances where the IoU is greater than 0.5. FP_{50}^{IoU} represents the number of false positive pixels within instances where the IoU is greater than 0.5.

$$P_{50}^{Boundary} = \frac{TP_{50}^{Boundary}}{TP_{50}^{Boundary} + FP_{50}^{Boundary}} \tag{15}$$

where $TP_{50}^{Boundary}$ represents the number of true positive pixels within instances where the Boundary IoU is greater than 0.5. $FP_{50}^{Boundary}$ represents the number of false positive pixels within instances where the Boundary IoU is greater than 0.5.

$$R_{50}^{IoU} = \frac{TP_{50}^{IoU}}{TP_{50}^{IoU} + FN_{50}^{IoU}} \tag{16}$$

where TP_{50}^{IoU} represents the number of true positive pixels within instances where the IoU is greater than 0.5. FN_{50}^{IoU} represents the number of false negative pixels within instances where the IoU is greater than 0.5.

$$R_{50}^{Boundary} = \frac{TP_{50}^{Boundary}}{TP_{50}^{Boundary} + FN_{50}^{Boundary}} \tag{17}$$

where $TP_{50}^{Boundary}$ represents the number of true positive pixels within instances where the Boundary IoU is greater than 0.5. $FN_{50}^{Boundary}$ represents the number of false negative pixels within instances where the Boundary IoU is greater than 0.5.

4.3. Experimental Parameter Setting

In this experiment, the HRNet model used for feature extraction is HRNetV2-W48 [51]. As for the variables λ_1, λ_2, λ_3, α, and β in the loss function of BPDNet, we set their values based on the reference HiSup [49]: $\lambda_1 = 1.0$, $\lambda_2 = 0.1$, $\lambda_3 = 1.0$, $\alpha = 8.0$, $\beta = 0.25$. Furthermore, for the variables θ, η, and T in the loss function, we conducted experiments to determine their optimal values for achieving the best results with BPDNet. Based on

the experimental results, we found that setting θ = 1.0, η = 1.0, and T = 1.0 yielded the best performance for BPDNet.

All experiments in this paper were conducted on an NVIDIA RTX3090 GPU. The batch size for the models was set to 2, and the training process consisted of 60 epochs. Random flipping augmentation was applied during the training of all methods. The AdamW optimizer was used for training, with an initial learning rate of 6×10^{-5}, β_1 = 0.9, β_2 = 0.999, and a weight decay coefficient of 0.01. The learning rate decay strategy employed the poly strategy.

4.4. Experimental Results

4.4.1. Quantitative Analysis

The experimental results are presented in Table 1. From the results, we can observe the following findings. Firstly, our proposed BPDNet method achieved improvements compared to the frame-field and HiSup. It was shown that BPDNet is superior to the current state-of-the-art building polygon extraction algorithms when facing the problem of limited training sample sizes. Secondly, compared with HiSup, BPDNet showed an increase of 1.40% in OA, 4.17% in IoU, 3.51% in P_{50}^{IoU}, 8.48% in $P_{50}^{Boundary}$, 3.29% in R_{50}^{IoU} and 8.01% in $R_{50}^{Boundary}$. The significant improvement in $P_{50}^{Boundary}$ and $R_{50}^{Boundary}$ indicated that BPDNet reduces missed and false extraction of building polygons and enhances the accuracy of building polygon extraction, including the precise positioning of building boundaries. This improvement effectively mitigates issues related to incorrect building polygon localization and blurry boundaries caused by diverse building structures and complex backgrounds. Thirdly, Model_2 performed the worst among the four comparative models, as indicated by the lowest evaluation scores, where the OA was 90.34%, IoU was 61.89%, P_{50}^{IoU} was 52.01%, $P_{50}^{Boundary}$ was 45.18%, R_{50}^{IoU} was 56.03% and $R_{50}^{Boundary}$ was 48.17%. This result indicates that the significant difference in the sample distribution between WHU-Mix and the study area greatly affected the performance of Model_2. Directly applying a model trained on WHU-Mix to predict the test set from the study area shows that the generalization capability obtained from a large dataset is still insufficient to overcome the impact of sample distribution differences. To address this issue, it is necessary to enable the model to learn the distribution characteristics of the samples in the task area. Compared to Model_2, Model_3 incorporates fine-tuning on the training set from the study area, enabling the model to learn the distribution characteristics of the samples in the task area. Although Model_3 showed improved performance compared to both Model_1 and Model_2, its evaluation metrics were still lower than those of Model_4 (BPDNet). Model_4 (BPDNet) achieved an OA of 92.28%, IoU of 66.54%, P_{50}^{IoU} of 56.01%, $P_{50}^{Boundary}$ of 54.19%, R_{50}^{IoU} of 60.18%, $R_{50}^{Boundary}$ of 57.35%. This indicates that although fine-tuning allows the model to learn the distribution characteristics of the task-specific dataset and partially overcome the differences in sample distribution, it does not effectively preserve the generalization performance obtained from the large dataset training. Model_4 (BPDNet) effectively distills the building feature representation capability obtained from diverse structured building samples in the large dataset to the student model through the CWD method. This enables the student model to achieve the best prediction results on the test set of the study area. Additionally, the inclusion of Dice Loss in the mask branch of Model_4 (BPDNet) further optimizes the boundaries of buildings, allowing for better differentiation between buildings and complex backgrounds. Therefore, based on the aforementioned analysis, the BPDNet method not only improves the pixel-level extraction results of buildings but also enhances the accuracy of the shape and position of building polygons. It reduces missed and false extraction of building polygons and demonstrates superior generalization ability compared to other methods for extracting building polygons with few samples in the Wenzhou area.

Table 1. Accuracy results on the test set using different methods. "w/Fine-tune " denotes that fine-tune was employed. "w/o Fine-tune" denotes that fine-tune was not employed.

Method	OA	IoU	P_{50}^{IoU}	$P_{50}^{Boundary}$	R_{50}^{IoU}	$R_{50}^{Boundary}$
Frame-field	88.43	59.84	50.17	43.25	54.24	47.11
Model_1 (Baseline)	90.88	62.37	52.50	45.71	56.89	49.34
Model_2 (w/o Fine-tune)	90.34	61.89	52.01	45.18	56.03	48.17
Model_3 (w/Fine-tune)	91.47	63.54	53.22	49.98	57.65	51.29
Model_4 (Ours)	**92.28**	**66.54**	**56.01**	**54.19**	**60.18**	**57.35**

4.4.2. Qualitative Analysis

To facilitate visual comparison of the results, we have provided visualizations of prediction outputs (as shown in Figure 7). From Figure 7, it is evident that Model_4 (BPDNet) exhibited more accurate positioning and boundary delineation of building polygons. BPDNet demonstrated fewer instances of missed detections compared to other methods, which is consistent with the findings of Table 1. In Row (a), Model_4 accurately identified the boundaries of the buildings within the red box, while other methods mistakenly recognized the background as buildings, resulting in inaccurate building polygon results. The Hisup-based methods (Model_1, Model_2, Model_3, Model_4) were able to recognize the complete building polygon in the yellow box, but the Frame-field model mistakenly identified the one building polygon as two parts in the yellow box. In Row (b), based on the imagery and labels in the second column, it can be observed that the building within the red box is a composite structure consisting of three high-rise buildings and a low-rise building. The extracted contour should only represent the entire complex, and only BPDNet and Model_3 successfully identified it as a single entity. Regarding the small building within the yellow box in Row (b), both BPDNet and Model_3 successfully detected it, while Frame-field, Model_1, and Model_2 overlooked this small building. In Row (c), Frame-field, Model_1, Model_2, and Model_3 exhibited varying degrees of issues with building polygon merging within the yellow box, where buildings are incorrectly connected or misclassified as part of the background. However, BPDNet accurately identified the boundaries of the buildings. In Row (d), only BPDNet recognized and precisely extracted the polygon of the small building within the red box, while other models failed to detect it. In the yellow box region, Model_2 mistakenly identified the water body as a building, indicating the impact of the difference in sample distribution between WHU-Mix and the study area on the accuracy of building recognition. In conclusion, the results of building polygon extraction demonstrated that BPDNet effectively tackles the challenges posed by diverse building structures and complex backgrounds. It accurately identifies the positions of building polygon and alleviates issues such as boundary blurring and merging.

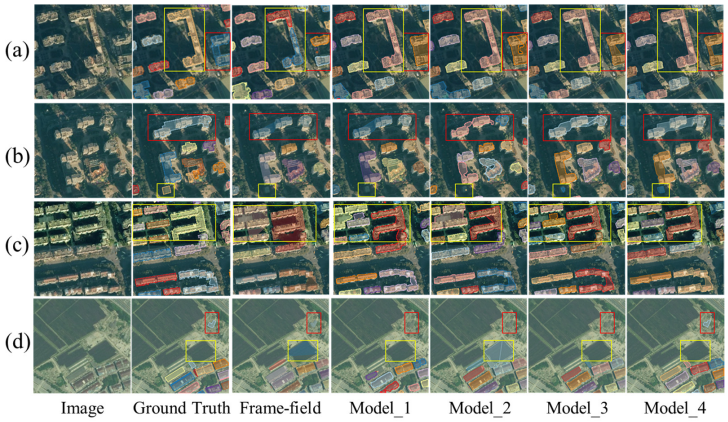

Figure 7. Comparison of visualization results of different methods for buildings with various structures. (**a**–**d**) denote different samples and results of the different methods.

5. Discussion

5.1. Ablation Experiments

To validate the effectiveness of the proposed method, we conducted ablation experiments by removing individual modules used in this study. The experimental results are shown in Table 2 below.

Table 2. Ablation experiments.

Method	Distillation	Dice Loss	OA	IoU	P_{50}^{IoU}	$P_{50}^{Boundary}$	Batch Time	Params
Baseline			90.88	62.37	52.50	45.71	**0.84s**	**74.29M**
Ours	√		91.37	64.63	54.11	51.98	0.88s	**74.29M**
Ours	√	√	**92.28**	**66.54**	**56.01**	**54.19**	0.90s	**74.29M**

From the experimental results, it can be observed that when using knowledge distillation alone, compared to the baseline, the OA increased by 0.49%, IoU increased by 2.26%, P_{50}^{IoU} increased by 1.61%, and $P_{50}^{Boundary}$ increased by 6.27%. All metrics showed improvement, with $P_{50}^{Boundary}$ showing a significant increase. This indicates that by using knowledge distillation, the accuracy of building boundary extraction can be greatly improved. The model effectively learns generalization knowledge from the large dataset, enabling more precise identification of diverse building boundaries. When both knowledge distillation and Dice Loss were used together, the model achieved the best performance. Compared to using knowledge distillation alone, OA increases by 0.91%, IoU increased by 1.91%, P_{50}^{IoU} increased by 1.90%, and $P_{50}^{Boundary}$ increased by 2.21%. All metrics showed stable growth. This indicates that the inclusion of Dice Loss effectively mitigates the problem of building boundary adhesion, resulting in more complete extraction of building polygon instances and more accurate boundaries.

Additionally, we calculated the GPU processing time per batch for each method during training and the number of parameters of each model. From Table 2, the Baseline achieved the least GPU processing time per batch. With the addition of distillation loss and Dice Loss, the GPU processing time for each batch increased. Compared to the baseline, the processing time per batch for the model with distillation loss and Dice Loss increased from 0.84 s to 0.90 s. We believe that this small increase in GPU processing time is acceptable due to the significant improvement in modeling results. Meanwhile, Table 2 shows that the number of parameters in each model is equal and the use of distillation loss and Dice Loss does not increase the complexity of the model.

5.2. Distillation Methods

In this section, we extensively investigate the impact of feature selection and various parameter settings on the experimental results during the knowledge distillation process, aiming to explore effective approaches for knowledge distillation.

5.2.1. Selection of Feature to Distillate

During the training of the student network guided by the teacher network, it is necessary to select distilled features. In this paper, the available features for selection include semantic mask features F_{seg}, line features F_{line}, and vertex features F_{ver}. We aimed to ensure that the student network learns effective feature representations, and thus conducted ablation experiments to investigate different feature combinations.

Table 3 presents the experimental results of different feature combinations during the distillation process. The student network achieved the best learning performance when using the semantic mask feature F_{seg}, line feature F_{line}, and vertex feature F_{ver} simultaneously. The results demonstrated that all three features have a positive impact on building polygon extraction. The combination of these three features outperformed any pairwise

combination, indicating their complementary nature and collective contribution to building polygon extraction.

Table 3. Distillation feature ablation experiments. √ denotes the feature is selected for knowledge distillation.

Distillation Feature			OA	IoU	P_{50}^{IoU}	$P_{50}^{Boundary}$
F_{seg}	F_{line}	F_{ver}				
√			91.33	64.43	54.01	52.93
	√		92.08	66.31	55.35	53.34
		√	92.04	66.26	54.92	53.86
√	√		92.13	66.32	55.62	53.65
√		√	92.16	66.37	55.56	54.12
	√	√	92.21	66.45	55.61	53.74
√	√	√	**92.28**	**66.54**	**56.01**	**54.19**

5.2.2. Distillation Temperature Hyperparameter

In knowledge distillation, the temperature hyperparameter T is used to control the smoothness of the SoftMax function's output during the feature map normalization process. A higher value of T indicates a larger spatial region of interest for each feature channel and leads to smoother outputs. We conducted experiments with different distillation temperatures, aiming to explore the appropriate values for the hyperparameter T and determine the optimal focus scale for feature channels during distillation.

Table 4 presents the experimental results with different hyperparameter T values. The model achieved the best performance when T = 1.00. When T < 1, the probability distribution became sharper as T decreased. On the other hand, when T > 1, the probability distribution became smoother. When T = 1, the SoftMax probability distribution remained consistent with the original distribution. In the results of this section, maintaining the original probability distribution actually yielded better performance. Comparing T = 0.50 with T = 0.02 and T = 0.70, there was a significant decrease in performance, indicating that a smaller focus scale for feature channels made it difficult for the student network to learn effective generalization knowledge from the teacher network.

Table 4. Experimental results for different distillation temperatures T.

Distillation Temperature	OA	IoU	P_{50}^{IoU}	$P_{50}^{Boundary}$
$T = 0.02$	91.25	63.21	53.86	51.84
$T = 0.50$	88.76	60.13	47.31	45.05
$T = 0.70$	92.05	65.97	54.96	52.81
$T = 1.00$	**92.28**	**66.54**	**56.01**	**54.19**
$T = 1.50$	90.85	62.76	52.24	50.96

5.2.3. Weighting of Distillation Loss

The total loss function of our proposed method consists of two components: the building polygon prediction loss and the distillation loss. The weight of the distillation loss η is a hyperparameter that controls the importance of the distillation process. In this study, we conducted experiments with different values of the distillation loss weight η to explore the optimal value for this hyperparameter.

Table 5 presents the experimental results with different values of the distillation loss weight. As the value of the weight η increased, the model's performance metrics steadily improved, indicating the crucial role of knowledge distillation in our proposed method. Particularly, when the weight value η was set to 1, the model achieved the best performance, demonstrating the effectiveness of the knowledge distillation approach.

Table 5. Distillation loss weighting and experimental results.

Weight of Distillation Loss	OA	IoU	P_{50}^{IoU}	$P_{50}^{Boundary}$
$\eta = 0.20$	92.12	66.30	55.13	53.16
$\eta = 0.40$	92.17	66.41	55.25	53.47
$\eta = 0.60$	92.18	66.43	55.24	53.01
$\eta = 0.80$	92.21	66.47	55.08	53.80
$\eta = 1.00$	**92.28**	**66.54**	**56.01**	**54.19**

5.2.4. Application of the Proposed Distillation Method

In order to verify the effectiveness of the proposed distillation method for other models, we applied the present distillation method to the Frame-field multi-task learning building polygon extraction model based on ResNet-101. The experimental results are shown in Table 6.

Table 6. The proposed distillation method's experimental results.

Method	Knowledge Distillation	Backbone	OA	IoU	P_{50}^{IoU}	$P_{50}^{Boundary}$	R_{50}^{IoU}	$R_{50}^{Boundary}$
Frame-field		ResNet-101	88.43	59.84	50.17	43.25	54.24	47.11
Frame-field	√	ResNet-101	90.45	61.74	51.33	44.97	55.15	48.36
Baseline		HRNetV2-W48	90.88	62.37	52.50	45.71	56.89	49.34
Ours	√	HRNetV2-W48	**92.28**	**66.54**	**56.01**	**54.19**	**60.18**	**57.35**

From Table 6, it can be found that although the method framework and backbone of Frame-field and baseline are different, both Frame-field and baseline are multi-task learning models, which are able to apply our proposed multi-feature distillation. After applying the proposed distillation method, the Frame-field was also able to improve significantly in all indicators. Therefore, our proposed distillation method can be applied to other building profile extraction models based on a multi-task learning approach. Due to the limitation of the distillation method, it is difficult to apply the proposed distillation method to the building polygon extraction model that does not adopt the multi-task learning method.

5.3. Weight Setting of Dice Loss

The overall segmentation loss of our model in this paper is composed of both binary cross-entropy (BCE) loss and Dice Loss, with a coefficient θ controlling the contribution of the Dice Loss. The Dice Loss measures the similarity between the predicted mask and the ground truth mask in the semantic mask branch, and the value of θ reflects the importance of the Dice Loss in semantic mask learning. We conducted experiments with different values of θ to explore the optimal weight for the Dice Loss in our proposed method.

Table 7 presents the experimental results with different values of the weight coefficient θ. When θ was set to 1, the model achieved the optimal performance in terms of all evaluation metrics. However, when θ was too small, there was a noticeable decline in model performance, indicating the important role of Dice Loss in the segmentation process. It is worth noting that setting θ too high can also lead to a decrease in model performance, suggesting that a larger value of θ is not necessarily better.

Table 7. Dice Loss weights and experimental results.

Weight of Dice Loss	OA	IoU	P_{50}^{IoU}	$P_{50}^{Boundary}$
$\theta = 0.25$	90.95	62.78	53.01	51.52
$\theta = 0.50$	91.38	64.64	54.13	52.09
$\theta = 0.75$	92.10	66.25	55.05	53.13
$\theta = 1.00$	**92.28**	**66.54**	**56.01**	**54.19**
$\theta = 1.25$	92.11	66.26	55.87	53.62
$\theta = 1.50$	92.16	66.32	55.46	54.01
$\theta = 1.75$	89.57	60.45	51.42	49.47
$\theta = 2.00$	91.44	64.73	54.18	52.23

6. Conclusions

This paper proposed a knowledge distillation method called BPDNet for building polygon extraction. BPDNet utilizes the structural knowledge of buildings distilled from a teacher network trained on a large dataset containing diverse building polygons. By leveraging the generalization capability of the teacher network in extracting structurally diverse building polygons, the proposed method enables the student network to learn the characteristics of building polygons in the target region, thereby improving the student network to represent the features of buildings in the target region. This approach addresses the challenge of limited samples and diverse building structures, which can lead to insufficient generalization performance of the model. Moreover, the incorporation of Dice Loss in the model enhances the accuracy of the boundaries and reduces the blurring of building edges caused by complex backgrounds such as building shadows and impermeable surfaces. The experimental results demonstrated that BPDNet exhibits superior performance in terms of evaluation metrics compared to models trained solely on the target region samples or models fine-tuned using large datasets. The ablation experiments further validated the effectiveness of our knowledge distillation method and the use of Dice Loss. The discussion on hyperparameters also confirmed the rationality of the parameter settings in BPDNet. In addition, the method has the following limitation: the method embeds sample knowledge obtained from a large dataset on the task areas with an insufficient sample size for enhancing the model's feature representation. However, if the sample size of the task areas is sufficient, the model can be trained without the sample knowledge from the large dataset to obtain a better representation of various styles of buildings. Therefore, the method is not applicable when the sample size of the task areas is sufficient. When the sample size is large enough, the extraction results of this method may not exceed the SOTA method by much.

In future work, our method can be further enhanced by incorporating prior knowledge about the structural regularities of buildings, such as the symmetry of building structures and the presence of right angles at building corners. By leveraging this prior knowledge, we can assist the inference process of building polygons and extract more accurate building polygons.

Author Contributions: Conceptualization, G.X. and J.C.; methodology, H.X. and G.S.; software, J.H.; validation, J.H.; formal analysis, H.X.; investigation, H.X.; data curation, J.H.; writing—original draft preparation, H.X. and G.S.; writing—review and editing, G.X. and J.C.; visualization, J.H.; supervision, G.X.; project administration, G.X.; funding acquisition, G.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Wenzhou Basic Scientific Research Project, grant number S20210017, and the Science and Technology Project of the Department of Natural Resources of Zhejiang Province, grant number 2021-38.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: All authors declare no conflict of interest.

References

1. Habitat, U. *Envisaging the Future of Cities*; World Cities Report; Un-Habitat: Nairobi, Kenya, 2022.
2. Deng, W.; Shi, Q.; Li, J. Attention-gate-based encoder–decoder network for automatical building extraction. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2611–2620. [CrossRef]
3. Sampath, A.; Shan, J. Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. *IEEE Trans. Geosci. Remote Sens.* **2009**, *48*, 1554–1567. [CrossRef]
4. Lafarge, F.; Descombes, X.; Zerubia, J.; Pierrot-Deseilligny, M. Automatic Building Extraction from DEMs Using an Object Approach and Application to the 3D-City Modeling. *ISPRS J. Photogramm. Remote Sens.* **2008**, *63*, 365–381. [CrossRef]
5. Ge, J.; Tang, H.; Yang, N.; Hu, Y. Rapid Identification of Damaged Buildings Using Incremental Learning with Transferred Data from Historical Natural Disaster Cases. *ISPRS J. Photogramm. Remote Sens.* **2023**, *195*, 105–128. [CrossRef]
6. Noronha, S.; Nevatia, R. Detection and modeling of buildings from multiple aerial images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 501–518. [CrossRef]
7. Yang, H.; Wu, P.; Yao, X.; Wu, Y.; Wang, B.; Xu, Y. Building Extraction in Very High Resolution Imagery by Dense-Attention Networks. *Remote Sens.* **2018**, *10*, 1768. [CrossRef]
8. Huang, X.; Zhang, L. Morphological building/shadow index for building extraction from high-resolution imagery over urban areas. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2011**, *5*, 161–172. [CrossRef]
9. Ali, M.; Clausi, D. Using the Canny edge detector for feature extraction and enhancement of remote sensing images. In Proceedings of the IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No. 01CH37217), Sydney, NSW, Australia, 9–13 July 2001.
10. Chen, G.; Jiang, Z.; Kamruzzaman, M. Radar remote sensing image retrieval algorithm based on improved Sobel operator. *J. Vis. Commun. Image Represent.* **2020**, *71*, 102720. [CrossRef]
11. San, D.K.; Turker, M. *Building Extraction from High Resolution Satellite Images using Hough Transform*; International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science: Kyoto Japan, 2010; Volume XXXVIII, Part 8.
12. Shunping, J.; Shiqing, W. Building extraction via convolutional neural networks from an open remote sensing building dataset. *Acta Geod. Et Cartogr. Sin.* **2019**, *48*, 448.
13. Yuan, Q.; Shen, H.; Li, T.; Li, Z.; Li, S.; Jiang, Y.; Xu, H.; Tan, W.; Yang, Q.; Wang, J.; et al. Deep Learning in Environmental Remote Sensing: Achievements and Challenges. *Remote Sens. Environ.* **2020**, *241*, 111716. [CrossRef]
14. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef] [PubMed]
15. Guo, Z.; Wu, G.; Song, X.; Yuan, W.; Chen, Q.; Zhang, H.; Shi, X.; Xu, M.; Xu, Y.; Shibasaki, R.; et al. Super-Resolution Integrated Building Semantic Segmentation for Multi-Source Remote Sensing Imagery. *IEEE Access* **2019**, *7*, 99381–99397. [CrossRef]
16. Yi, Y.; Zhang, Z.; Zhang, W.; Zhang, C.; Li, W.; Zhao, T. Semantic Segmentation of Urban Buildings from VHR Remote Sensing Imagery Using a Deep Convolutional Neural Network. *Remote Sens.* **2019**, *11*, 1774. [CrossRef]
17. Cheng, G.; Han, J. A survey on object detection in optical remote sensing images. *ISPRS J. Photogramm. Remote Sens.* **2016**, *117*, 11–28. [CrossRef]
18. Ding, W.; Zhang, L. Building detection in remote sensing image based on improved YOLOv5. In Proceedings of the 2021 17th International Conference on Computational Intelligence and Security (CIS), Chengdu, China, 19–22 November 2021.
19. Han, Q.; Yin, Q.; Zheng, X.; Chen, Z. Remote Sensing Image Building Detection Method Based on Mask R-CNN. *Complex Intell. Syst.* **2022**, *8*, 1847–1855. [CrossRef]
20. Su, H.; Wei, S.; Yan, M.; Wang, C.; Shi, J.; Zhang, X. Object Detection and Instance Segmentation in Remote Sensing Imagery Based on Precise Mask R-CNN. In Proceedings of the IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 1454–1457.
21. Su, H.; Wei, S.; Liu, S.; Liang, J.; Wang, C.; Shi, J.; Zhang, X. HQ-ISNet: High-Quality Instance Segmentation for Remote Sensing Imagery. *Remote Sens.* **2020**, *12*, 989. [CrossRef]
22. Ma, J.; Wu, L.; Tang, X.; Liu, F.; Zhang, X.; Jiao, L. Building Extraction of Aerial Images by a Global and Multi-Scale Encoder-Decoder Network. *Remote Sens.* **2020**, *12*, 2350. [CrossRef]
23. Guo, M.; Liu, H.; Xu, Y.; Huang, Y. Building Extraction Based on U-Net with an Attention Block and Multiple Losses. *Remote Sens.* **2020**, *12*, 1400. [CrossRef]
24. Ran, S.; Gao, X.; Yang, Y.; Li, S.; Zhang, G.; Wang, P. Building Multi-Feature Fusion Refined Network for Building Extraction from High-Resolution Remote Sensing Images. *Remote Sens.* **2021**, *13*, 2794. [CrossRef]
25. Li, W.; Zhao, W.; Zhong, H.; He, C.; Lin, D. Joint Semantic-Geometric Learning for Polygonal Building Segmentation. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 1958–1965. [CrossRef]
26. Wei, S.; Zhang, T.; Ji, S.; Luo, M.; Gong, J. BuildMapper: A Fully Learnable Framework for Vectorized Building Contour Extraction. *ISPRS J. Photogramm. Remote Sens.* **2023**, *197*, 87–104. [CrossRef]
27. Girard, N.; Smirnov, D.; Solomon, J.; Tarabalka, Y. Polygonal Building Segmentation by Frame Field Learning. *arXiv* **2021**, arXiv:2004.14875.

28. Sun, X.; Zhao, W.; Maretto, R.V.; Persello, C. Building Polygon Extraction from Aerial Images and Digital Surface Models with a Frame Field Learning Framework. *Remote Sens.* **2021**, *13*, 4700. [CrossRef]
29. Xue, N.; Bai, S.; Wang, F.; Xia, G.-S.; Wu, T.; Zhang, L. Learning Attraction Field Representation for Robust Line Segment Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1595–1603.
30. Li, Z.; Wegner, J.D.; Lucchi, A. Topological map extraction from overhead images. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
31. Zhao, W.; Persello, C.; Stein, A. Building outline delineation: From aerial images to polygons with an improved end-to-end learning framework. *ISPRS J. Photogramm. Remote Sens.* **2021**, *175*, 119–131. [CrossRef]
32. Zorzi, S.; Bazrafkan, S.; Habenschuss, S.; Fraundorfer, F. PolyWorld: Polygonal Building Extraction With Graph Neural Networks in Satellite Images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1848–1857.
33. Tsai, Y.-H.; Hung, W.-C.; Schulter, S.; Sohn, K.; Yang, M.-H.; Chandraker, M. Learning to Adapt Structured Output Space for Semantic Segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7472–7481.
34. Parvaneh, A.; Abbasnejad, E.; Teney, D.; Haffari, G.R.; van den Hengel, A.; Shi, J.Q. Active Learning by Feature Mixing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 12237–12246.
35. Osco, L.P.; Marcato Junior, J.; Marques Ramos, A.P.; de Castro Jorge, L.A.; Fatholahi, S.N.; de Andrade Silva, J.; Matsubara, E.T.; Pistori, H.; Gonçalves, W.N.; Li, J. A Review on Deep Learning in UAV Remote Sensing. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *102*, 102456. [CrossRef]
36. Maggiori, E.; Tarabalka, Y.; Charpiat, G.; Alliez, P. Can Semantic Labeling Methods Generalize to Any City? The Inria Aerial Image Labeling Benchmark. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 3226–3229.
37. Ji, S.; Wei, S.; Lu, M. Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Trans. Geosci. Remote Sens.* **2018**, *57*, 574–586. [CrossRef]
38. Guo, Y.; Shi, H.; Kumar, A.; Grauman, K.; Rosing, T.; Feris, R. SpotTune: Transfer Learning Through Adaptive Fine-Tuning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4805–4814.
39. Too, E.C.; Yujian, L.; Njuki, S.; Yingchun, L. A Comparative Study of Fine-Tuning Deep Learning Models for Plant Disease Identification. *Comput. Electron. Agric.* **2019**, *161*, 272–279. [CrossRef]
40. Alshalali, T.; Josyula, D. Fine-tuning of pre-trained deep learning models with extreme learning machine. In Proceedings of the 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 12–14 December 2018.
41. Dhande, A.; Malik, R. Design of a Highly Efficient Crop Damage Detection Ensemble Learning Model Using Deep Convolutional Networks. *J. Ambient Intell. Human. Comput.* **2023**, *14*, 10811–10821. [CrossRef]
42. Gou, J.; Yu, B.; Maybank, S.J.; Tao, D. Knowledge Distillation: A Survey. *Int. J. Comput. Vis.* **2021**, *129*, 1789–1819. [CrossRef]
43. Shu, C.; Liu, Y.; Gao, J.; Yan, Z.; Shen, C. Channel-Wise Knowledge Distillation for Dense Prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 5311–5320.
44. Wang, X.; Zhu, J.; Yan, Z.; Zhang, Z.; Zhang, Y.; Chen, Y.; Li, H. LaST: Label-Free Self-Distillation Contrastive Learning With Transformer Architecture for Remote Sensing Image Scene Classification. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [CrossRef]
45. Nabi, M.; Maggiolo, L.; Moser, G.; Serpico, S.B. A CNN-Transformer Knowledge Distillation for Remote Sensing Scene Classification. In Proceedings of the IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium, Kuala Lumpur, Malaysia, 17–22 July 2022; pp. 663–666.
46. Sudre, C.H.; Li, W.; Vercauteren, T.; Ourselin, S.; Jorge Cardoso, M. Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations. In *Proceedings of the Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Cardoso, M.J., Arbel, T., Carneiro, G., Syeda-Mahmood, T., Tavares, J.M.R.S., Moradi, M., Bradley, A., Greenspan, H., Papa, J.P., Madabhushi, A., et al, Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2017; pp. 240–248.
47. Luo, M.; Ji, S.; Wei, S. A diverse large-scale building dataset and a novel plug-and-play domain generalization method for building extraction. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2023**, *16*, 4122–4138. [CrossRef]
48. Mohanty, S. CrowdAI Dataset (2018). 2018. Available online: https://github.com/crowdai/crowdai-mapping-challenge-mask-rcnn (accessed on 1 March 2023).
49. OpenAI, 2018 Open AI Tanzania Building Footprint Segmentation Challenge. 2018. Available online: https://competitions.codalab.org/competitions/20100 (accessed on 1 March 2023).
50. Van Etten, A.; Lindenbaum, D.; Bacastow, T.M. Spacenet: A remote sensing dataset and challenge series. *arXiv* **2018**, arXiv:1807.01232.
51. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep High-Resolution Representation Learning for Human Pose Estimation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5693–5703.

52. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015.
53. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on International Conference on Machine Learning, Omnipress, Madison, WI, USA, 21 June 2010; pp. 807–814.
54. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 11534–11542.
55. Cheng, B.; Girshick, R.; Dollar, P.; Berg, A.C.; Kirillov, A. Boundary IoU: Improving Object-Centric Image Segmentation Evaluation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 15334–15342.