remote sensing

MDPI

Article

Unifying Building Instance Extraction and Recognition in UAV Images

Xiaofei Hu, Yang Zhou *, Chaozhen Lan, Wenjian Gan, Qunshan Shi and Hanqiang Zhou

School of Geospatial Information, Information Engineering University, Zhengzhou 450001, China; huxiaofeicn@163.com (X.H.); lan_cz@163.com (C.L.); 14737117985@163.com (W.G.); hills1@163.com (Q.S.); 18890485697@163.com (H.Z.)

* Correspondence: zhouyang3d@163.com

Abstract: Building instance extraction and recognition (BEAR) extracts and further recognizes building instances in unmanned aerial vehicle (UAV) images, holds with paramount importance in urban understanding applications. To address this challenge, we propose a unified network, BEAR-Former. Given the difficulty of building instance recognition due to the small area and multiple instances in UAV images, we developed a novel multi-view learning method, Cross-Mixer. This method constructs a cross-regional branch and an intra-regional branch to, respectively, extract the global context dependencies and local spatial structural details of buildings. In the cross-regional branch, we cleverly employed cross-attention and polar coordinate relative position encoding to learn more discriminative features. To solve the BEAR problem end to end, we designed a channel group and fusion module (CGFM) as a shared encoder. The CGFM includes a channel group encoder layer to independently extract features and a channel fusion module to dig out the complementary information for multiple tasks. Additionally, an RoI enhancement strategy was designed to improve model performance. Finally, we introduced a new metric, *Recall@*(*K*, *iou*), to evaluate the performance of the BEAR task. Experimental results demonstrate the effectiveness of our method.

Keywords: building instance extraction; building instance recognition; remote sensing; unmanned aerial vehicle; urban understanding

check for updates

Citation: Hu, X.; Zhou, Y.; Lan, C.; Gan, W.; Shi, Q.; Zhou, H. Unifying Building Instance Extraction and Recognition in UAV Images. *Remote Sens.* **2024**, *16*, 3449. https://doi.org/10.3390/rs16183449

Academic Editors: Massimiliano Pepe and Haigang Sui

Received: 25 July 2024
Revised: 2 September 2024
Accepted: 16 September 2024
Published: 17 September 2024

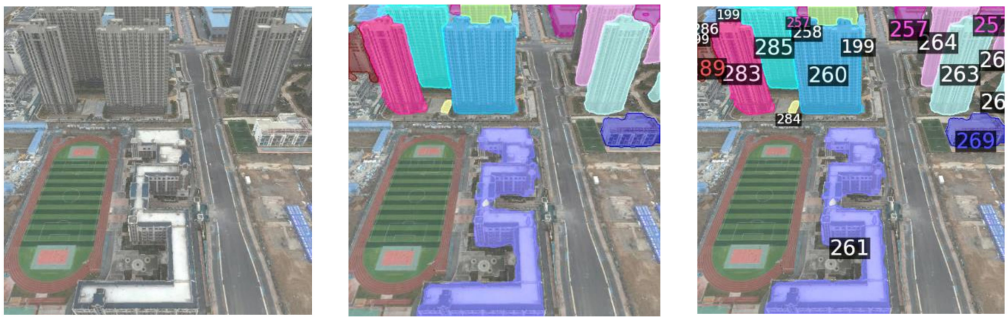
Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Buildings, as a major type of urban feature, use high-resolution remote sensing images (RSI) to acquire building instance information, a crucial task in urban understanding [1–3]. Satellites and unmanned aerial vehicles (UAVs) are currently the two mainstream platforms for acquiring RSI [4,5]. Satellite imagery can achieve large-scale building target detection and statistics [6,7]. Compared with satellites, UAVs fly closer to the ground and can capture higher-resolution building images from different views, providing more detailed information [8], thus prompting demand for more precise interpretation tasks, such as building instance extraction and recognition (BEAR).

The BEAR task extracts and further recognizes building instances from UAV images, as shown in Figure 1. Building instance extraction (BIE) involves detecting and segmenting all buildings in an image [9]. Building instance recognition (BIR) aims to identify the 'identity information' of buildings in an image [10]. Its technical process is similar to face recognition [11], where a descriptor of the query building is first extracted and compared to the known building descriptors in the database to confirm the query building's 'identity'. BEAR, which can be seen as the integration of BIE and BIR, holds significant application value and practical demand in visual positioning and navigation [12–14], accurate delivery [15], building damage assessment [16], and urban planning [17,18]. For example, Xue et al. [13] and Tian et al. [14] have effectively accomplished image location recognition by first segmenting the building instances on an image and then identifying the global

ID of each building. However, these previous works were limited to street view imagery, and separately performed BIE and BIR tasks using different models. Although integrating BIE and BIR is more in line with practical application requirements and can also reduce redundant processing and unnecessary complexity, as far as we know, there is currently no end-to-end BEAR solution for UAV images.



(**a**) Raw Image (**b**) BIE Result (**c**) BEAR Result

Figure 1. Comparison of BEAR and BIE. (**a**) An UAV image with multiple buildings. (**b**) BIE, which can simultaneously detect and segment all building instances in the image. (**c**) We assign IDs to real-world building instances, and BEAR will extract all building instances and recognize the ID of each building in the image. By using IDs, buildings in the image are linked to their real-world counterparts.

The challenges of implementing an end-to-end BEAR solution mainly lies in two aspects. Firstly, buildings in UAV images are characterized by small areas and multiple instances [19]. They tend to contain less information due to their small area, and similar appearances, which makes BIR more difficult in UAV images than that in ground-view images. Secondly, BIE and BIR are two tasks with different paradigms. How to integrate them within a unified model for training and inference remains an issue that needs further exploration.

To solve the abovementioned difficulties, we propose an end-to-end BEAR network: BEAR-Former. First, a novel building feature extraction method, Cross-Mixer, is proposed to address the challenges in BIR for UAV images. Cross-Mixer comprises a cross-regional branch and an intra-regional branch to extract the global context dependencies and local spatial structural details of buildings, respectively. In the cross-regional branch, we utilize cross-attention and polar-coordinate relative position encoding (PRPE) to learn more discriminative features. To integrate the BIE and BIR, we designed a channel group and fusion module (CGFM) as a shared encoder, which includes the channel group encoder layer (CGEL) to independently extract features of multiple tasks, and a channel fusion module (CFM) to dig out the complementarity information between different tasks. In addition, we introduce a region of interest (RoI) enhancement strategy to improve model performance.

The novel contributions of this study can be summarized as follows:

1. We propose a novel BIR method (Cross-Mixer) based on the dual-branch structure, which can capture the global context while preserving spatial-detailed. The method achieves the state-of-the-art performance on the test benchmarks.
2. We discovered the complementarity between BIE and BIR tasks. Integrating them into a unified model effectively can not only reduce computational redundancy but also improves performance in both tasks.
3. We developed an end-to-end BEAR network (BEAR-Former). BIE and BIR are effectively integrated by designing a CGFM module and an RoI enhancement strategy.
4. We developed an evaluation metric for the BEAR task. Additionally, we introduce a virtual matching-based building instance annotation generation method, which reduces the workload of training and evaluation data preparation.

2. Related Work

2.1. Building Instance Recognition

In the BIR task, a query image containing one or more buildings is given to retrieve the same building instance from an established building images database. BIR is also known as landmark recognition or visual place recognition (VPR) [20–22], can be used for various tasks, such as robot localization [23–25], and urban planning [26]. BIR is usually regarded as an image retrieval problem, can be solved by image instance retrieval following the technical process represented in Figure 2b. The process generally involves two modules: the backbone and recognition decoder. The backbone extracts local semantic information from the raw image, and the recognition decoder, also known as the feature aggregation module, aggregates the local semantic information into a high-dimensional vector, called a feature descriptor. The feature descriptor for the query image is searched against the benchmark feature database to find the most similar features for accurate building instance recognition.

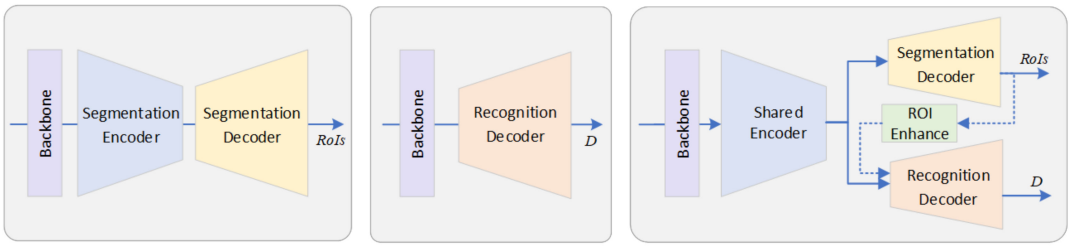


Figure 2. Typical network structures. (**a**) Typical network structure of BIE; (**b**) Typical network structure of BIR; (**c**) The network structure of BEAR-Former.

Early techniques rely on manually determined local features, such as SIFT [27], SURF [28], and ORB [29], which are aggregated into a global descriptor using bag-of-words (BoW) [30] or vector of locally aggregated descriptor (VLAD) [31]. Deep learning methods for BIR, with traditional local features replaced by convolutional neural network (CNN) [32], and various decoders for feature aggregation have been proposed. Deep learning features are robust to variation in lighting, viewpoint, and geometric distortion. With the success of deep learning, almost all recent BIR techniques make use of learned representations. Arandjelovic et al. [33] integrated NetVLAD, a trainable version of VLAD, on the CNN backbone to achieve performance superior to that of traditional methods. On this basis, many variants have been proposed, such as CRN [34] and Patched-NetVLAD [35]. GeM [36] stands out as another significant aggregation method, offering a versatile, learnable adaptation of global pooling. CosPlace [37] enhances GeM by integrating with a linear projection layer, achieving excellent performance for landmark recognition. Vision transformers have gradually been applied in BIR. Wang et al. [38] developed TransVPR, which uses CNNs for feature extraction and transformers for attention fusion to create global descriptors. AnyLoc [39], with features extracted by the foundation model DINOv2 [40], combined with unsupervised aggregation methods, achieves remarkable performance. Currently, the latest method is MixVPR [41], a multilayer perceptron (MLP) aggregation technique that has achieved state-of-the-art performance across multiple benchmark tests. In general, current methods have adopted advanced technologies such as attention mechanisms, striving to maintain good performance in large-scale, complex environments.

It should be noted that the aforementioned methods are typically designed for ground-view images, which generally contain only one building (the entire image can be considered as the region of the building itself). However, UAV images have significant differences from ground-view images. Buildings in UAV images are characterized by small areas and multiple instances, which means less information within the building area itself. Existing methods have explored various advanced technologies to generate compact feature descriptors, which can effectively extract the inherent features of buildings, but they are inadequate for UAV images. Additionally, UAV images usually have large viewpoint

changes, we need to employ view-based 3D retrieval techniques, learning the complete features of a building by studying its multiple views images. Therefore, we propose a multi-view learning method for BIR in UAV images, which significantly improves recognition accuracy by considering cross-regional features of buildings. In practical applications, BIR needs to consider appearance differences caused by different lighting, seasons, weather, scales, and viewpoints. Due to limited space, in this study, we focus on the multi-view learning for buildings, which is one of the unique features of UAV images.

2.2. Building Instance Extraction

BIE involves delineating the pixel regions of each individual building instance in remote sensing images, and is usually regarded as an instance segmentation task. Instance segmentation can be a two-step or single-step process. The two-step process involves detection and segmentation, exploits the object detectors to obtain the regions of RoIs and then predicts the segmentation mask for each RoI. Mask Region-based CNN (Mask R-CNN) [42], Cascade R-CNN [43], and RefineMask [44] are two-step instance segmentation methods widely used in BIE. Wang et al. [45] integrated the path aggregation network, feature pyramid network, and atlas space pyramid pooling into Mask R-CNN to improve the multi-scale feature extraction and fusion capabilities of the traditional village building extraction method. Based on the two-stage framework of Mask R-CNN, Xu et al. [9] proposed a gated spatial memory and centroid-aware network to achieve building extraction. The single-step process starts with pixel-level segmentation and then clusters the pixels into different instances. Examples of single-step methods include SOLO [46] and SOLOv2 [47]. InstanceCut [48] first predicts all instance boundaries and then infers the globally optimal instance segmentation method. Wagner et al. [49] proposed an architecture consisting of three parallel paths for instance segmentation of buildings in high-resolution satellite imagery. TernausNetV2 [50] includes an encoder–decoder network with skipped connections that allows objects to be extracted from high-resolution imagery at the instance level.

The vision transformer (ViT) has become a mainstream framework for BIE. One advantage of ViT is their ability to produce models with a large number of parameters and to be trained on extensive datasets, exhibiting stronger feature extraction capabilities. Based on the ViT architecture, Mask2Former [51] achieving high precision in image segmentation, has been widely applied in BIE and is a commonly used framework for this task. Its basic structure is shown in Figure 2a. Fu et al. [52] designed a fused building extraction method by taking advantage of the complementarity between CNN and transformer. RSPrompter [53] is a prompt-learning method for instance segmentation in remote sensing images, leveraging the SAM [54] foundation model to automatically obtain semantic instance-level masks.

The purpose of this study is not to investigate new methods that can improve the performance of instance segmentation, but to explore approaches that integrate BIE and BIR tasks, constructing an end-to-end BEAR framework. Integrating them not only reduces computational redundancy but also better meets the needs of actual applications. This is because UAV images contain multiple buildings, which need to be extracted before they can be recognized. Since ViT-based method usually has higher precision, this study chooses Mask2Former as the foundational BIE framework. We primarily focus on the improvement of the encoder, designing a shared encoder capable of generating specific semantic information required for both BIE and BIR tasks. It is worth mentioning that we discovered the complementarity between BIE and BIR, which means a cleverly designed shared encoder can enhance the performance of both tasks.

3. Materials and Methods

3.1. Problem Formulation

We formulate the BEAR task in UAV images as a combination of two stages, training and testing, as shown in Figure 3. During the training stage, we need to design the BEAR-Former network architecture, train it on the training dataset, obtain the model weight

M, and share M in the testing stage. The testing stage uses a test dataset containing multi-view database images of all buildings and the query images in region R. BEAR-Former generates a descriptor $\hat{d}_i$ for each building instance in region R, yielding a binary set $D_R = \left\{\left(ID_1, \hat{d}_1\right), \left(ID_2, \hat{d}_2\right), \ldots, \left(ID_i, \hat{d}_i\right)\right\}$, where ID_i and $\hat{d}_i$ represent the ID and descriptor of ith building in region R, respectively.

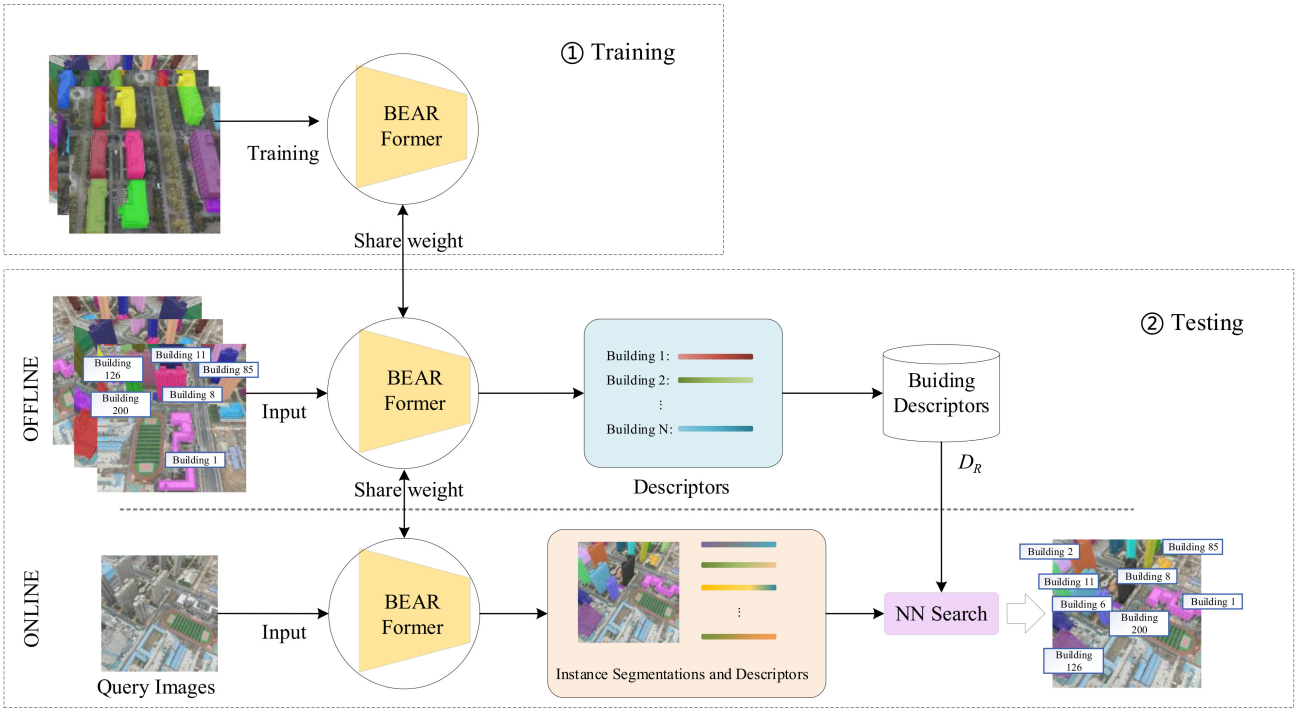


Figure 3. The BEAR task consists of training and testing stages. In the training stage, the parameter weights of BEAR-Former are obtained. The testing stage begins with generate the building descriptor database D_R. Then, all building instances in the query image are extracted, and the ID of each building is recognized by instance retrieval.

The query images are those captured at arbitrary viewpoints in region R. Our goal is to identify the pixel region and ID of each building present in the query images. Similarly, BEAR-Former is employed to extract and generate the descriptor d_i for each building instance in the query image. Subsequently, it queries the dataset D_R to retrieve the most similar building instance $\hat{d}_i$ and its ID. This process achieves the simultaneous extraction and recognition of building instances in the query image.

Our designed BEAR-Former is a combination of BIE and BIR network structures, as shown in Figure 2c. BEAR-Former is composed of a backbone, a shared encoder, an instance segmentation decoder, an instance recognition decoder, and an ROI enhancement module. The backbone is used to generate a feature map of buildings from the input image. The shared encoder is implemented based on a CGFM and are used to obtain high-level semantic features required for the multiple tasks. The segmentation decoder extracts all building instances from the image's semantic features to obtain each building's ROI. The recognition decoder, implemented based on Cross-Mixer, takes the ROI and semantic features of each building as input to recognize its global ID. Additionally, we designed an ROI Enhancement module to enhance the accuracy of building recognition.

3.2. Recognition Decoder Based on Cross-Mixer

We propose the Cross-Mixer method to implement the recognition decoder. The recognition decoder takes the RoI of each building generated by the segmentation decoder as input, with the goal of generate more compact and robust descriptors for all building

instances. BIR for UAV images differs considerably from that for traditional ground view images, as shown in Figure 4. Buildings on UAV images are typically characterized by small areas and multiple instances, leading to less information within the building region. which makes it difficult to achieve accurate recognition without cross regional relationship information. Although local information is essential to preserve rich spatial details, global context dependencies are crucial for building recognition. In this regard, the proposed Cross-Mixer constructs two parallel branches (a cross-regional branch and an intra-regional branch) to extract the global context dependencies and local spatial structural details of buildings, respectively, as shown in Figure 5.

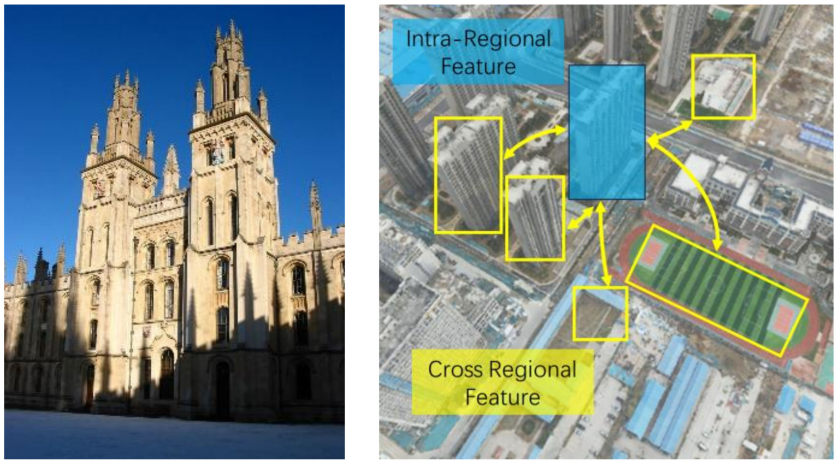


Figure 4. Comparison of ground view and UAV images. Ground view building image (**a**), generally containing a single building; UAV image (**b**), where blue pixel area represents the building to be recognized. The UAV image contains multiple similar buildings, each with a small area.

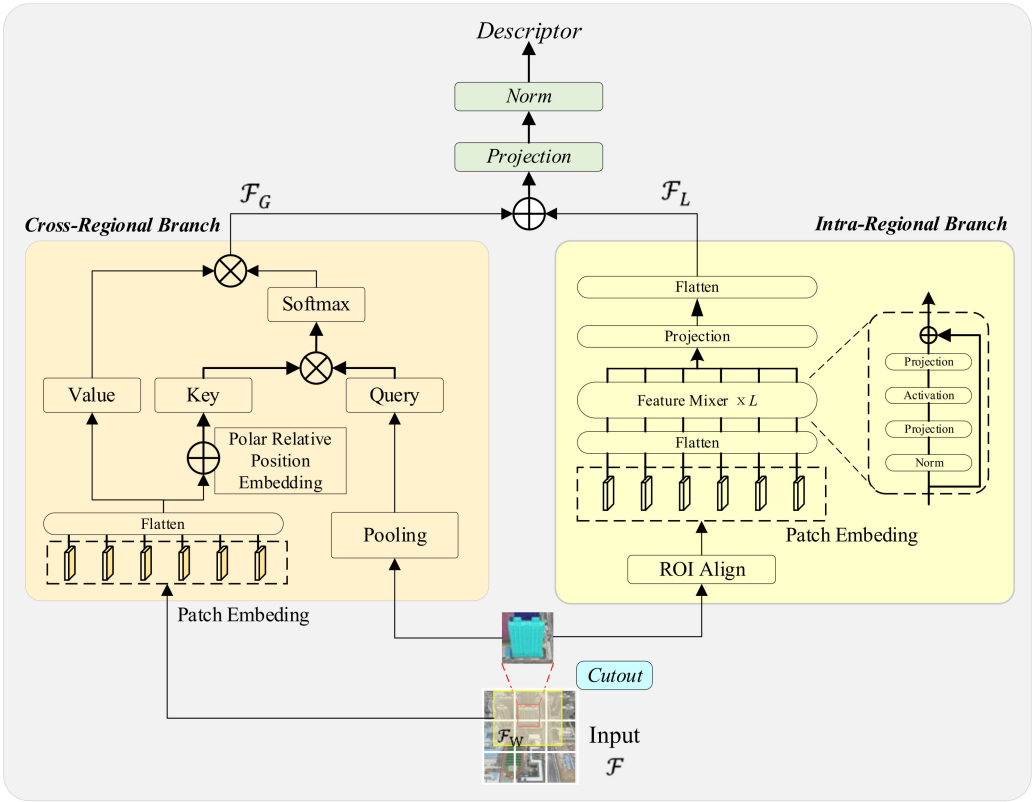


Figure 5. Structure of Cross-Mixer.

3.2.1. Cross-Regional Branch with Cross-Attention

The cross-regional branch will generate a descriptor $\mathcal{F}_G$ that represents the dependency between the query building and other image regions. The branch employs multi-head cross-attention to capture the dependency relationship. We take the building instance that needs identification as Q and a larger window as K and V. Cross-attention can learn the cross-relationship between vectors Q and K. By focusing on relevant information, the model can make connections and capture the dependencies between the query building and the rest of the regions in the image, thereby producing a representation that are more discriminative.

The input feature map $\mathcal{F}$ generates three representations in cross-attention:

$$\begin{aligned} Q_i &= adaptive_avg_pool(R_a(\mathcal{F}, RoI_i)) \\ K &= PatchEmbedding(\mathcal{F}_w) + PositionEmbedding(\mathcal{F}_w) \\ V &= PatchEmbedding(\mathcal{F}_w) \end{aligned} \tag{1}$$

where R_a denotes the ROI Align operation; Q_i is generated by adaptive average pooling of the aligned features of ith building. Suppose that there are M buildings in the image; then, $Q = concatenate(Q_1, Q_2 \ldots, Q_M)$, $Q \in M * B * C$. K and V are generated by $PatchEmbedding$ of the larger window feature $\mathcal{F}_w$. The attention between Q and K is calculated and applied to the producing representation of each building as follows:

$$\mathcal{F}_G = Softmax\left(\frac{QK}{\sqrt{d_k}}\right) * V \tag{2}$$

In addition, we employ a feedforward network (FFN) consisting of a fully connected layer, which helps to further refine the feature representation and extract higher-level semantic information.

3.2.2. Polar Relative Position Embedding (PRPE)

It is very important to embed positional information in Q and K in cross-attention. Typically, a method based on cartesian coordinates is employed to embed the absolute position of each embedded patch. However, in BEAR task, we focus on the relative position of all patches to the query building, because the absolute position is non-discriminative for different buildings in the image. Although we can embed the position of Q, and the model learn implicitly the relative cartesian position between Q and K, it is difficult to evaluate whether the model is robust in learning the relative position between different patches. Therefore, we propose a relative position embedding method based on polar coordinates to directly and explicitly embed the relative position information.

Let O_q be the center point of the ROI for the building to be recognized. Define a polar coordinate system with O_q as the origin, as illustrated in Figure 6, where the positive direction is straight above the image ($\theta = 0$). Thus, we can obtain the normalized relative polar coordinates for each patch.

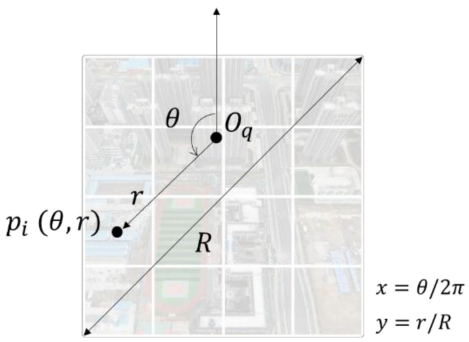


Figure 6. Relative position embedding based on polar coordinates.

Traditional positional embedding can be represented as:

$$P_{(pos,2i)} = sin\left(pos/10,000^{(2i/d)}\right) P_{(pos,2i+1)} = cos\left(pos/10,000^{(2i/d)}\right) \quad (3)$$

where $P_{(pos,2i)}$ is the 2ith element of the d-dimensional vector, and pos is the coordinate of the point. In PRPE, pos is different from the traditional coordinates. We adopt the coordinate embedding of the two-dimensional plane and then take the polar coordinates θ and r as the X and Y coordinates, respectively. The calculation formula is as follows.

$$pos_\theta = \begin{cases} arctan(-\Delta_y/\Delta_x) + \frac{\pi}{2}, & \Delta_x < 0 \\ arctan(-\Delta_y/\Delta_x) + \frac{3\pi}{2}, & \Delta_x > 0 \end{cases} pos_r = \frac{r}{R} * 2\pi \quad (4)$$

where is Δ_x and Δ_y are the cartesian coordinates distance to the origin O_q, $r = \sqrt{{\Delta_x}^2 + {\Delta_y}^2}$, and R is the diagonal length of the image plane. The coordinate range of pos_θ is $0 - 2\pi$. In practice, the coordinate values can be scaled according to actual needs.

3.2.3. Intra-Regional Branch with the Feature Mixer

The intra-regional branch generates a descriptor $\mathcal{F}_L$ that represents the local spatial structural features of the building itself. Since MLP shows excellent performance in building recognition [41], MixVPR (an advanced method in building recognition with an isotropic all-MLP architecture) was used to extract intra-regional features. We align the ROI of the building instance with size $N \times N$, flatten it, and fed it into the Feature Mixer. The composition of the feature mixer is shown in Figure 5. After repeating the operation L times, a feature descriptor is generated through projection and flatten.

Finally, $\mathcal{F}_G$ and $\mathcal{F}_L$ are concatenated and then through a projection and normalization operation to obtain the final feature descriptor.

3.3. Shared Encoder Based on a CGFM

Encoders can learn high-level semantic information of buildings, which can be very useful in instance segmentation and recognition. To effectively integrate BIE and BIR tasks, we designed a CGFM Module as the shared encoder, which consists of CGEL and CFM modules, as shown in Figure 7. CGEL can independently train the network weights, and generating high-level semantic information for specific tasks. CFM takes advantage of the complementarity between the two tasks.

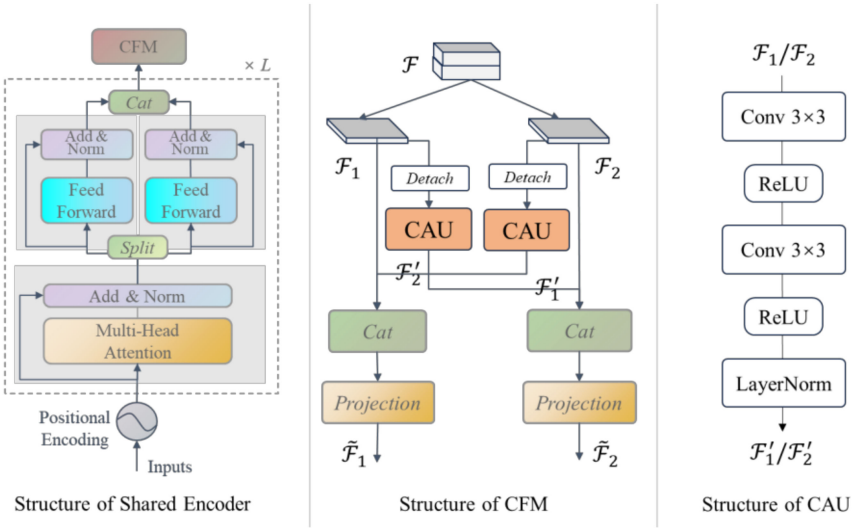


Figure 7. Structure of a CGFM.

(1) CGEL: In a standard ViT, each encoder layer has a standard architecture consisting of a multi-head self-attention module and a FFN module. Such an encoder is designed for

a single task. To train two tasks simultaneously, we modified the design of the standard encoder layer. Specifically, after the multi-head attention processing, the feature channels are divided into two groups, each fed into an FFN module. Following the independent execution of the two FFNs, the two channel groups are concatenated and input into the next CGEL layer.

(2) CFM: We designed the CFM module to dig out and fuse the complementary information from the features of the BIE and BIR tasks. We also divide the features $\mathcal{F}$ output by the CGEL into two groups, $\mathcal{F}_1$ and $\mathcal{F}_2$, perform *Detach* operations on $\mathcal{F}_1$ and $\mathcal{F}_2$, and utilize a commonalities analysis unit (CAU) to mine and extract complementary information, $\mathcal{F}_1'$ and $\mathcal{F}_2'$. *Detach* operation can prevent negative interactions during training. CAU first uses two 3×3 convolutions followed by a ReLU activation to capture building information in the image and suppress non-building information. Next, CAU performs cross-concatenation and projection to obtain high-level semantic features for different tasks. The projection is implemented based on a linear layer, which aims to recombine features from different sources.

3.4. RoI Enhancement

In the BEAR-Former, the segmentation decoder obtains the ROIs of all buildings and then fed them into the recognition decoder to generate their descriptors. To improve the accuracy of BIR, we propose an RoI enhancement strategy. As shown in Figure 8, the operations include training with ground-truth RoI (TGR), RoI random offset (RRO), and RoI enlargement (RE).

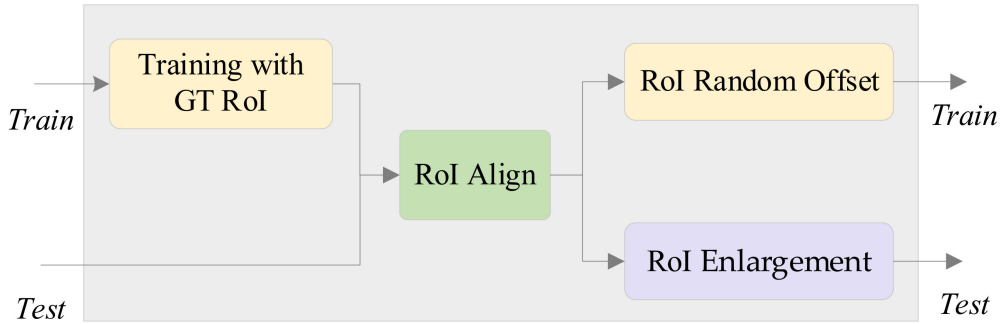


Figure 8. Strategies for RoI enhancement.

TGR: We fed the building's ground-truth RoI into the recognition decoder module, instead of the predicted RoI, in the training process. The predicted RoI by the segmentation decoder is different from the ground-truth. If predicted RoI is directly used to supervise the building recognition, the wrong data will be fed into the module for training. Therefore, we input the ground-truth RoI of the buildings.

RRO: During the training process, we apply an RoI random offset for each building instance to mimic the intersection-over-union (IOU) error of the predicted RoI. Generally, the prediction for larger RoIs is more accurate than those for smaller RoIs. Therefore, the magnitude of the offset should be proportional to the RoI edge length. The calculation equation is as follows:

$$C_x = random\left(-\frac{\sqrt{L_x}}{2}, \frac{\sqrt{L_x}}{2}\right) C_y = random\left(-\frac{\sqrt{L_y}}{2}, \frac{\sqrt{L_y}}{2}\right) \tag{5}$$

where L_x and L_y represent the lengths of the edge of the ROI.

RE: We assume that the receptive field should be appropriately increased to reduce the error of predicted ROI and thus improve the recognition accuracy. If the edge length of the ROI is assumed to be L, then the length of the expanded edge becomes $L' = L + \sqrt{L}$.

In addition, the area of each building instance on the image is different, but the input features are required to have the same size in the subsequent phase. Therefore, each RoI is aligned into a fixed size.

3.5. Loss Function

The loss function used to train the model consists of two parts, namely segmentation loss and recognition loss.

For the recognition loss ($Loss_R$), two requirements must be met: the descriptors must be discriminative across different building, and the descriptors must be invariant to the viewpoint changes in the same building. We employ triplet-center loss (TCL) [55] to address these two requirements simultaneously. TCL enables learning a center for each building instance, penalizes the distance between sample and corresponding central descriptors. This approach results in all central descriptors being distinctively distributed in the embedding space, with sample descriptors from the same instance being closer to their respective central descriptors, while the descriptors from different instances being farther apart. For N samples in a batch, TCL is calculated as:

$$Loss_R = \sum_{i=1}^{N} max(D(f_i, c_i) + m - \min_{j \neq i} D(f_i, c_j), 0) \quad (6)$$

where $D(\cdot)$ represents the distance between two vectors. f_i represents a sample descriptor of *ith* building, and c_i represents its center descriptor.

The segmentation loss ($Loss_S$) is composed of two parts. We follow the commonly used in standard instance segmentation loss, which consists of binary cross-entropy loss and dice loss. Please refer to [51] for details.

Finally, $Loss_R$ and $Loss_S$ are weighted and combined to obtain the final loss function:

$$L = Loss_S + \beta \, Loss_R \quad (7)$$

where β is the proportional adjustment factor. We experimentally analyzed the effect of β.

4. Results and Discussion

4.1. Building Instance Annotation Generation Based on Virtual Matching

Establishing a multi-view building images dataset used in BEAR is a very tedious and costly process, requiring intense manual annotation. To improve the annotation efficiency, we designed a virtual matching-based annotation generation method. The specific process is shown in Figure 9. The UAV is used to collect image set Img_{src}. Our goal is to generate the corresponding building instance annotation masks $Mask_{src}$ for Img_{src}. This process involved three steps. The first step is building modeling. In this step, we employ SFM technology or use existing 3D models to obtain a 3D scene S_1, then each building in S_1 is monomerized and assigned a global ID. The ID was colorized to obtain a masked 3D scene S_2. The second step is simulation. We re-photograph within the 3D scenes S_1 and S_2 according to the pose information of the images in Img_{src}, generating a virtual image set Img_{sim} and its corresponding masks $Mask_{sim}$. The images in Img_{sim} correspond one to one with the images in Img_{src}. The third step is matching. In this step, Img_{src}, Img_{sim}, and $Mask_{sim}$ are the inputs, and output the mask set $Mask_{src}$. The images in Img_{src} are matched with the corresponding images in Img_{sim}, and the geometric transformation (H) is calculated. Then, we apply the transformation H to the corresponding mask in $Mask_{sim}$, followed by cropping, to obtain $Mask_{src}$. In this process, only building monomerization is performed manually; therefore, the approach is highly efficient.

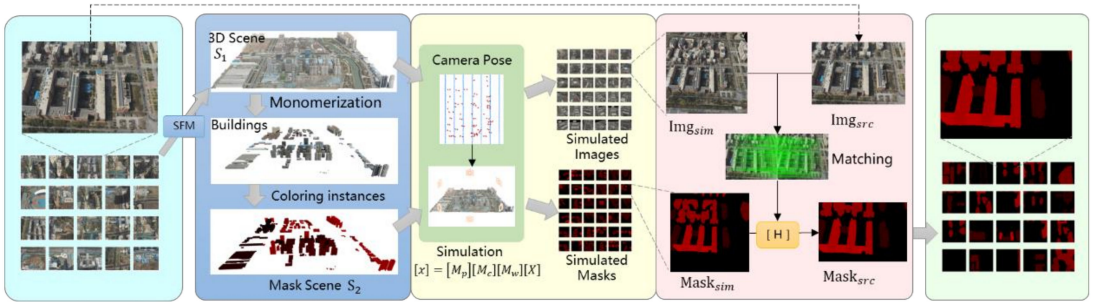


Figure 9. Building instance annotation framework. (**a**) The images dataset Img_{src}, our goal is to generate the corresponding instance masks $Mask_{src}$ of Img_{src}; (**b**) 3D modeling and monomerization, the 3D model scene S_1 of Img_{src}, and masked 3D scene S_2. The monomerization is performed manually, with different buildings annotated by different colors; (**c**) virtual image generation, in which, according to the pose information of the images in Img_{src}, the scenes S_1 and S_2 are re-photograph to obtaining a virtual image set Img_{sim} and a virtual mask image set $Mask_{sim}$; (**d**) matching Img_{sim} and Img_{src} to generate the instance annotation masks $Mask_{src}$ for Img_{src}; (**e**) samples of $Mask_{sim}$.

4.2. Dataset Description

Using the above methodology, we created three building multi-view images datasets to evaluate the BEAR task performance, including STPLS3D [56] dataset and two UAV buildings datasets (Scene 1 and Scene 2). We also used the WHU-Buildings dataset [57] to evaluate the BIE task performance. Details are as follows.

STPLS3D: STPLS3D is a large-scale synthetic point cloud dataset for urban scene understanding. The dataset is a virtually reconstructed 3D scene covering 16 square kilometers containing more than 25 urban scene areas. We re-photograph the 3D scene as described in Section 4.1 to obtain building instance multi-view images for training and testing. We used half of the scenes for training and all scenes for testing. In each scene, we got 9×9 evenly sampled points, and took eight images for each sampled point (camera height of 200 m, pitch angle of $-45°$, yaw angle interval of $45°$). The query image set were captured at random positions in test scenes, including 500 images. In this study, we focus on multi-view learning for buildings; therefore, the query images are collected at the different yaw angles with the training images (camera height of 200 m, pitch angle of $-45°$, random yaw angle).

UAV-Buildings: We collected UAV images in the suburbs of Zhengzhou, China. The dataset consists of two scenes. Scene 1 (1.7×1.5 km) includes 352 building instances. First, the training data were collected at a flight altitude of 335 m, a pitch angle of $-60°$, and a yaw angle interval of $90°$. After the 3D reconstruction, the virtual scene images were collected for data supplementation at a yaw angle interval of $45°$. The training dataset has 4312 images (also used as test database), and 302 images were collected for query (altitude of 335 m, pitch angle of $-60°$, random yaw angle). Scene 2 (2.0×1.5 km) contains 376 building instances. The images for the training data were taken at an altitude of 335 m and a pitch angle of $-90°$. Each image was rotated at a $60°$ interval for data augmentation, and 4744 training images were obtained. The query set contains 390 images with random yaw angle.

WHU-Buildings: We used the WHU-Buildings dataset to evaluate the impact of building instance extraction of our proposed method. The WHU-Buildings dataset contains 8188 images (512×512 pixels) with a ground resolution of 0.3 m. There are 4736 images in the training set, 1036 images in the validation set, and 2416 images in the test set. The ground truth values for each image are generated from the provided building vector maps. The WHU aerial dataset includes approximately 187,000 buildings with different uses, sizes, and colors.

4.3. Evaluation Metrics

To evaluate the BEAR task for UAV images, we created a metric, $Recall@(K, iou)$, which reflects the system's ability to retrieve the correct results. The query building is successfully retrieved if at least one of the top-k retrieved reference images has same global ID and the IOU is greater than threshold.

$$Recall@(K, iou) = \frac{\sum_i^N TP_i}{\sum_i^N TP_i + FN_i} \tag{8}$$

where TP_i denotes the number of buildings that were successfully retrieved in *ith* image, FN_i denotes the number of buildings that were not retrieved, K refers to the maximum number of queries per search, with a connotation the same as that of K used in traditional building recognition methods, *iou* denotes the intersection/union ratio threshold between the predicted ROI and the ground-truth of each building. In accordance with conventions, we set three parameters with different levels of difficulty, $Recall@(1, 0.75)$, $Recall@(5, 0.5)$, and $Recall@(10, 0.5)$.

We use $Recall@K$ to evaluate the performance of the BIR task, which is a standard recall metric for instance retrieval.

For BIE task, we use the average precision (mAP), AP@0.5 and AP@0.75 metrics, which are widely used in the evaluation of instances segmentation [58].

4.4. Implementation Settings

The proposed model is implemented based on Detectron2 [59] framework and trained on four NVIDIA 4090 GPUs (Santa Clara, CA, USA). We use the AdamW [60] optimizer and a stepwise learning rate schedule. An initial learning rate of 0.0001 and a decay parameter of 0.05 were used for all network weights by default. A learning rate multiplier of 0.1 is applied to the building recognition decoder module. We use 80 k iterations with a batch size of 16 for training unless otherwise specified. The default descriptor dimension is 512. We resize the image to a fixed size of 512 × 512. To accelerate convergence, the backbone network parameters are initialized using pre-trained weights on the COCO dataset.

4.5. BEAR Experiment

Table 1 shows the overall performance of the proposed method for the BEAR task on different datasets. The results show that the proposed method achieved good performance on the three datasets, supporting its ability to achieve the task of building instance segmentation and recognition robustly.

Table 1. Results of the BEAR experiment.

Dataset	R@(1, 0.75)	R@(5, 0.5)	R@(10, 0.5)	mAP	AP@0.5	AP@0.75
STPLS3D	79.2	84.6	87.3	65.9	79.5	70.5
UAV-Buildings (Scene 1)	65.4	78.4	83.6	56.8	79.1	61.6
UAV-Buildings (Scene 2)	92.9	96.2	98.6	74.0	90.2	81.8

To further evaluate the performance of the proposed method, Figure 10 shows a representative example of the BEAR results. The proposed method generated a clear building outline and accurately identified the ID of the building. Most of the recognition errors were in buildings that were heavily occluded or had a small area. The visualization results show the effectiveness of the proposed method in the BEAR task.



Figure 10. BEAR examples using different methods. (**a**) Query image. (**b**) BIE result. (**c**) BEAR results for our proposed method. (**d**) Ground truth. Recognition errors are marked with red boxes.

4.6. BIR Experiment

To verify the effectiveness of the proposed Cross-Mixer, comparative analyses were carried out with several algorithms, including NetVlad, GeM, CosPlace, ConvAP [61], DELG [62], Patch-NetVLAD and MixVPR. For comparison, we used the same backbone network, fixed the size of each building to 64 × 64 through the ROI Align operation, and generated descriptors are all 512-dimensional descriptors. Testing is conducted by inputting the ground-truth RoI. The experimental results are shown in Table 2. Our method greatly outperformed all other algorithms across all benchmarks. For example, Cross-Mixer achieved the highest R@1 value in STPLS3D (69.5%), an improvement of 12.3% over that for the recently developed MixVPR technique; In UAV-Buildings (Scene 1), Cross-Mixer achieved the highest R@1 value (60.1%), an improvement of 7.0% over that for CosPlace. In

UAV-Buildings (Scene 2), Cross-Mixer achieved the highest R@1 of 92.7%, an improvement of 1.6% over that for DELG, the second-best performing technique.

Table 2. Experimental comparison of BIR results.

Method	STPLS3D			UAV-Buildings (Scene 1)			UAV-Buildings (Scene 2)		
	R@1	**R@5**	**R@10**	**R@1**	**R@5**	**R@10**	**R@1**	**R@5**	**R@10**
NetVLAD	33.6	51.7	58.3	31.1	64.8	67.5	90.3	92.4	95.5
GeM	4.6	5.7	5.89	16.4	25.7	35.0	46.7	55.0	58.9
CosPlace	42.6	53.9	54.4	53.1	59.3	71.8	75.6	82.3	86.1
ConvAP	21.8	26.0	26.3	36.6	46.0	53.3	62.1	79.3	81.8
DELG	56.3	61.9	63.6	49.5	57.2	64.9	91.1	93.3	96.5
Patch-NetVLAD	35.5	53.3	60.1	48.1	58.3	70.5	90.1	93.3	94.5
MixVPR	57.2	61.7	62.8	49.8	59.7	65.5	88.3	91.2	96.0
Cross-Mixer	**69.5**	**75.3**	**78.6**	**60.1**	**67.1**	**72.8**	**92.7**	**95.6**	**97.6**

The bold values represent the results of our method.

The images in UAV-Buildings (Scene 2) were taken from a nadir view, the appearance of the building changes little. Therefore, there was little variation among used methods. This indicates that BIR in the oblique UAV images is more difficult. However, our method still achieved the best performance (1.6% higher than that of DELG).

Figure 11 shows the experimental results obtained using Cross-Mixer and three recently developed instance retrieval methods (NetVLAD, CosPlace, and MixVPR). The results show that our method has fewer retrieval errors and markedly outperforms the other methods.

Experimental result show that the proposed method is more suitable for BIR in UAV images. Although the advanced building instance recognition algorithm explores the attention mechanism and loss function, the recognition accuracy is poor due to the unfavorable conditions (small area, repetition, and occlusion of buildings) of UAV images and the lack of supervision of global information. Our method constructs dual-branches to, respectively, extract the global context dependencies and local spatial structural information, can generate more robust and discriminative features.

Analyze the failure cases in the BEAR and BIR experiments. We found that the failures mainly consist of two situations. The first is edge recognition errors, where a small number of building instances located at the edges of the image fail to be recognized, as shown in rows 1, 3, and 4 of Figure 11. This occurs because buildings at the image edges are often incomplete, allowing only a portion of them to be observed, which leads to incomplete features. The second situation is adjacent recognition errors, where a building instance is incorrectly identified as its neighboring building instance, as shown in the third row of Figure 10. This happens because the neighboring buildings are in the same environment, resulting in similar cross-regional features. Sometimes, both types of errors occur simultaneously, as illustrated in the fourth row of Figure 10, where multiple small buildings are clustered together and located at the image edge, presenting a particularly challenging situation. Query expansion and spatial verification are potential improvements to the aforementioned problems.

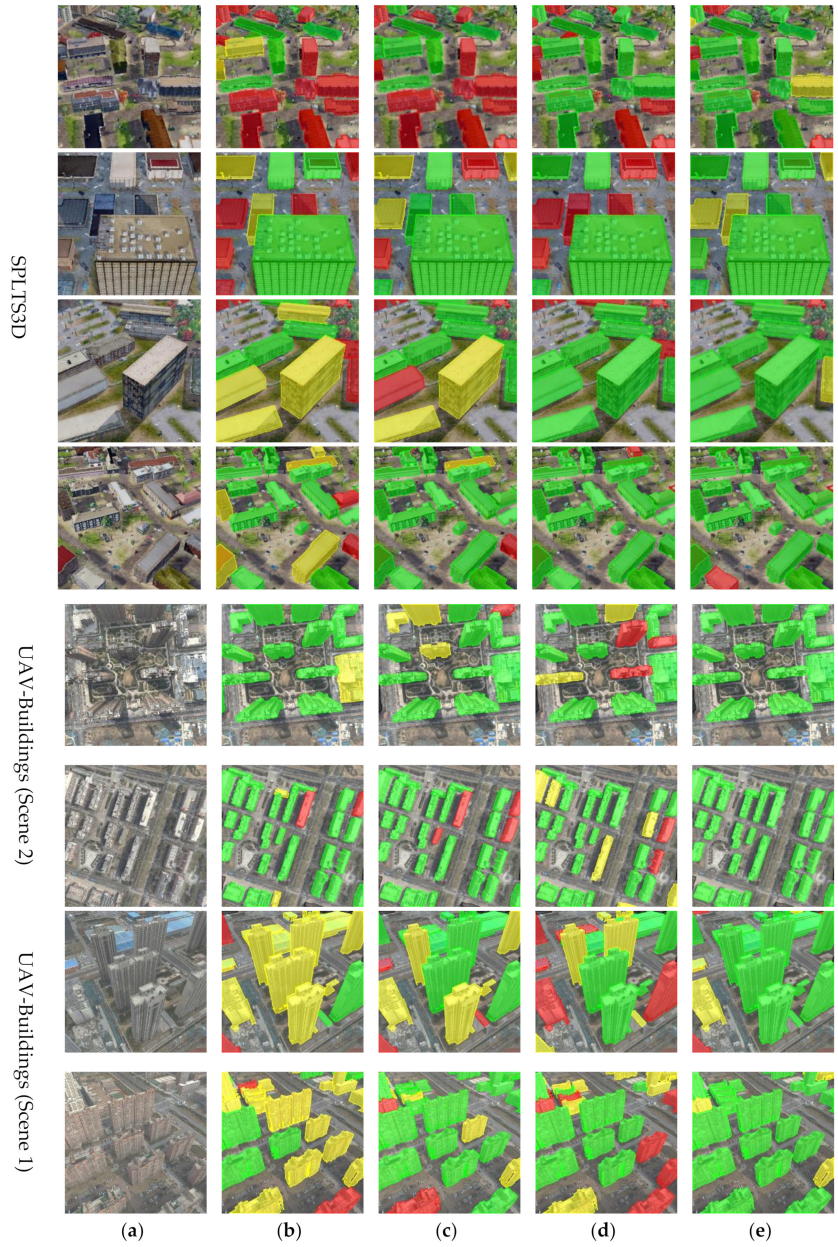


Figure 11. BIR examples for different methods. (**a**) Original query image. (**b**) NetVLAD result. (**c**) CosPlace result. (**d**) MixVPR result. (**e**) Cross-Mixer result. The red mask indicates retrieval failure, the green mask indicates successful retrieval of TOP-1, and the yellow mask indicates successful retrieval of TOP-10.

4.7. Ablation Study

4.7.1. A CGFM

To verify the effectiveness of a CGFM, we designed an experiment for BIR and BIE tasks. The BIR results are shown in Table 3. Baseline indicates that we only train the BIR module. In the STPLS3D and UAV-Buildings benchmarks, a CGFM improved the improves recall@1 performance of the BIR task by 8.9%, 5.8%, and 1.6%, respectively.

We evaluated the effectiveness of a CGFM on BIE task, as shown in Table 4. BIE256/128 indicates that the BIE module is individual trained without BIR module, and the encoder channels is 256/128. F256 represents the features (256 channels) generated by the shared encoder is fully shared by the segmentation decoder and recognition decoder. The CFGM encoder also includes 256 channels, with 128 channels each for segmentation decoder and recognition decoder. The mAP of a CGFM was only 0.5% lower than that of BIE256 in the

WHU buildings benchmark but was 1.5% higher in the STPLS3D benchmark. A comparison of the experimental results for F256 and BIE128 show that when two tasks share the same encoder features without clever design during training, it is struggles to mine the specific and complementary information of different tasks, leading to a decline in instance segmentation performance. In contrast, a CGFM effectively extracts the complementary information for different tasks and significantly enhancing the precision of BIE. Experimental result shows that there is a significant benefit from integration the two tasks by a CGFM, as evidenced by the improved accuracy.

Table 3. Ablation study of building recognition using a CGFM.

Method	STPLS3D			UAV-Buildings Scene 1			UAV-Buildings Scene 2		
	R@1	**R@5**	**R@10**	**R@1**	**R@5**	**R@10**	**R@1**	**R@5**	**R@10**
Baseline	60.6	68.5	71.6	54.3	60.9	65.4	91.1	93.0	95.2
With CGFM	69.5	75.3	78.6	60.1	67.1	72.8	92.7	95.6	97.6

Table 4. Ablation study of BIE using a CGFM.

Dataset	Method	mAP	AP@0.5	AP@0.75
WHU Buildings	BIE256	60.8	84.7	69.3
	BIE128	57.7	80.3	65.5
	F256	57.7	80.9	65.9
	CGFM	**60.3**	**82.8**	**68.7**
STPLS3D	BIE256	64.4	76.1	67.4
	BIE256	61.3	73.2	64.6
	F256	60.6	75.1	64.4
	CGFM	**65.9**	**79.5**	**70.5**

The bold values represent the results of our method.

The experimental results also support the complementarity between the BIE and BIR tasks. The complementarity occurs because both tasks extract the semantic information of building instances but tend to extract different types of semantic. Specifically, the BIE task tends to extract the common features, whereas the BIR task tends to extract the unique features of building instances. Common and unique features are complementary; accordingly, they promote each other. The integration of the feature extraction of the two tasks not only reduces model redundancy but also yields mutually beneficial effects.

4.7.2. Effectiveness of the Cross-Regional Branch

The comparison between MixVPR and the proposed method in Table 2 also illustrates the role of the newly designed cross-regional branch. The proposed building recognition method Cross-Mixer includes cross-regional and intra-regional branch, and the latter adopts the MIixPR method. A comparison of the last two rows in Table 2 shows that the performance of BIR increased significantly after introducing cross-regional attention, and the recall@1 performance on the three benchmarks increased by 12.3%, 10.3%, and 4.4%, respectively, demonstrating the effectiveness of cross-regional attention.

4.7.3. Impact of PRPE

We evaluated the impact of designed PRPE, as shown in Table 5 (where baseline denotes the use of the traditional absolute positional embedding). The proposed method improved the accuracy of BIR, with the recall@1 performance increased by 1.7%. This is

because the relative position is a better indication of the accurate dependency between the building instance and its surroundings.

Table 5. Ablation study of PRPE.

Method	R@(1, 0.75)	R@(5, 0.5)	R@(10, 0.5)	mAP	AP@0.5	AP@0.75
Baseline	77.5	83.2	86.1	65.1	79.3	70.8
Ours	79.2	84.6	87.3	65.9	79.5	70.5

4.7.4. Importance of RoI Enhancement

To illustrate the importance of the RoI enhancement module, we performed a series of ablation experiments using the STPLS3D dataset. First, no RoI enhancement strategy was used (Baseline). Second, only the TGR strategy was used (TGR). Thirdly, in addition to the TGR strategy, the RRO strategy was added; that is, the ground-truth RoI was randomly offset (TGR + RRO). Finally, TGR, RRO, and RE were all used together (TGR + RRO + RE). The experimental results are shown in Table 6. The RoI enhancement module significantly enhanced the recognition accuracy, with the R@(1, 0.75) performance increased by 8.5%. In addition, the performance improved with each additional strategy. The TGR improves R@(1, 0.75) by 5.9%, RRO improves R@(1, 0.75) by 1.4%, and RE improves R@(1, 0.75) by 1.2%.

Table 6. Ablation study results for enhanced prediction.

Methods	R@(1, 0.75)	R@(5, 0.5)	R@(10, 0.5)
Baseline	70.7	75.8	78.1
TGR	76.6	82.1	84.1
TGR + RRO	78.0	83.0	86.2
TGR + RRO + RE	79.2	84.6	87.2

4.7.5. Optimization of the Loss Weight Parameter

In the design of the loss function, the weight β of recognition loss is an important hyperparameter. We designed multiple experiments by varying β. We set the β values in steps of 5, with 5 as the minimum and 60 as the maximum. Figure 12 illustrates the impact of β on BEAR-Former. This parameter influenced the building recognition results substantially. When $\beta < 20$, the $Recall@(K, iou)$ was low; when $\beta > 25$, the $Recall@(K, iou)$ tended to be stable. When adjusting the value of β, the BIE results remained stable, without large fluctuations. But when $\beta > 25$, the BIE and BIR results showed similar trends. When $\beta = 45$, both tasks achieved the best performance, further supporting the complementarity of the two tasks.

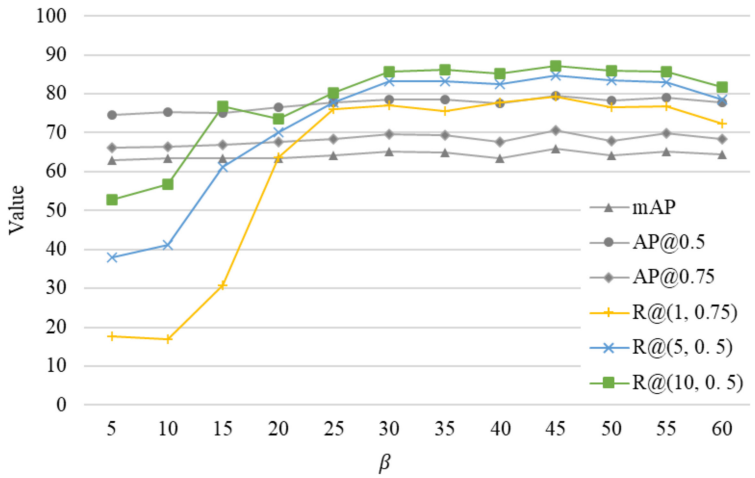
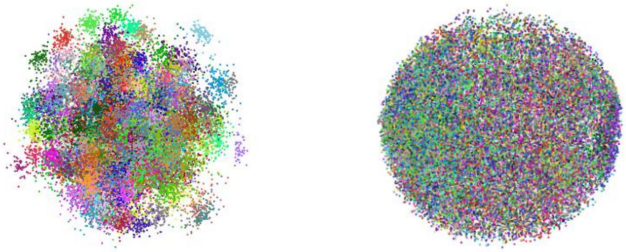


Figure 12. Effects of the parameter β on the BEAR performance.

4.7.6. Loss Function Experiment

To illustrate the effectiveness of TCL loss adopted in this study, we experimentally compared it with multi-similarity loss (MSL) [63], a commonly used loss function in BIR. We visualized the descriptors for all building instances in the query images (Figure 13). While MSL is good at mining difficult samples, it does not learn the common features of each instance. TCL can learn the common features of multi-view images for same building instance, so that the descriptors show obvious clustering.



(**a**) (**b**)

Figure 13. Visualization results for different loss functions. (**a**) TCL visualization results. (**b**) MSL visualization results. Points of the same color represent the same instance.

5. Conclusions

We conducted in-depth research on the BEAR task in UAV images and proposed a novel multi-view learning framework: BEAR-Former. In view of the characteristics of buildings in UAV images (i.e., small areas and multiple instances), we designed a novel BIR method Cross-Mixer, which constructs a cross-regional branch and an intra-regional branch to, respectively, extract the global context dependencies and local spatial structural details of buildings. We demonstrated the effectiveness of the cross-regional feature through ablation studies, and proved that the proposed method achieves the best performance on multiple datasets. We discovered the complementarity between BIE and BIR, and to integrate the two tasks effectively, we designed a CGFM module as shared encoder and an RoI enhancement strategy. The effectiveness of a CGFM and RoI enhancement were experimentally demonstrated.

This work is just preliminary study on BEAR, and more efforts are needed to realize a mature BEAR system. The efficiency and generalizability of the algorithm are worth further research. The proposed method specifically applies to intelligent perception tasks for building understanding. In future work, we will extend the BEAR framework to more general visual perception tasks, such as urban planning, visual positioning, and navigation. It is hoped that this work can contribute to this study of UAV remote sensing and urban understanding.

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

Funding: This research was funded by the Joint Fund of Collaborative Innovation Center of Geo-Information Technology for Smart Central Plains, Henan Province and Key Laboratory of Spatiotemporal Perception and Intelligent processing, Ministry of Natural Resources, No. 232109.

Data Availability Statement: The WHU building dataset can be downloaded from http://gpcv.whu.edu.cn/data/building_dataset.html (accessed on 15 September 2024). The STPLS3D dataset can be downloaded from https://github.com/meidachen/STPLS3D (accessed on 15 September 2024). For the processed multi-view images, please write an email to the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, Q.; Mou, L.; Sun, Y.; Hua, Y.; Shi, Y.; Zhu, X.X. A Review of Building Extraction from Remote Sensing Imagery: Geometrical Structures and Semantic Attributes. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 4702315. [CrossRef]
2. Wang, L.; Fang, S.; Meng, X.; Li, R. Building Extraction with Vision Transformer. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5625711. [CrossRef]
3. Li, J.; Huang, X.; Tu, L.; Zhang, T.; Wang, L. A Review of Building Detection from Very High Resolution Optical Remote Sensing Images. *GIScience Remote Sens.* **2022**, *59*, 1199–1225. [CrossRef]
4. Deng, R.; Guo, Z.; Chen, Q.; Sun, X.; Chen, Q.; Wang, H.; Liu, X. A Dual Spatial-Graph Refinement Network for Building Extraction from Aerial Images. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–20. [CrossRef]
5. Chen, S.; Ogawa, Y.; Zhao, C.; Sekimoto, Y. Large-scale individual building extraction from open-source satellite imagery via super-resolution-based instance segmentation approach. *ISPRS J. Photogramm. Remote Sens.* **2023**, *195*, 129–152.
6. Shao, Z.; Tang, P.; Wang, Z.; Saleem, N.; Yam, S.; Sommai, C. BRRNet: A fully convolutional neural network for automatic building extraction from high-resolution remote sensing images. *Remote Sens.* **2020**, *12*, 1050. [CrossRef]
7. Shakeel, A.; Sultani, W.; Ali, M. Deep built-structure counting in satellite imagery using attention based re-weighting. *ISPRS J. Photogramm. Remote Sens.* **2019**, *151*, 313–321.
8. Lyu, Y.; Vosselman, G.; Xia, G.-S.; Yilmaz, A.; Yang, M.Y. UAVid: A Semantic Segmentation Dataset for UAV Imagery. *ISPRS J. Photogramm. Remote Sens.* **2020**, *165*, 108–119. [CrossRef]
9. Xu, L.; Li, Y.; Xu, J.; Guo, L. Gated Spatial Memory and Centroid-Aware Network for Building Instance Extraction. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 4402214. [CrossRef]
10. Li, J.; Huang, W.; Shao, L.; Allinson, N. Building Recognition in Urban Environments: A Survey of State-of-the-Art and Future Challenges. *Inf. Sci.* **2014**, *277*, 406–420. [CrossRef]
11. Deng, J.; Guo, J.; Xue, N.; Zafeiriou, S. ArcFace: Additive angular margin loss for deep face recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
12. Huang, B.; Lian, D.; Luo, W.; Gao, S. Look before you leap: Learning landmark features for one-stage visual grounding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 16888–16897.
13. Xue, F.; Budvytis, I.; Reino, D.O.; Cipolla, R. Efficient large-scale localization by global instance recognition. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 17327–17336.
14. Tian, Y.; Chen, C.; Shah, M. Cross-View Image Matching for Geo-Localization in Urban Environments. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1998–2006. [CrossRef]
15. Brar, S.; Rabbat, R.; Raithatha, V.; Runcie, G.; Yu, A. *Drones for Deliveries*; Technical Report; Sutardja Center for Entrepreneurship & Technology, University of California: Berkeley, CA, USA, 2015; Volume 8, p. 2015.
16. 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.
17. Yi, S.; Liu, X.; Li, J.; Chen, L. UAVformer: A composite transformer network for urban scene segmentation of UAV images. *Pattern Recognit.* **2023**, *133*, 109019.
18. Muhmad Kamarulzaman, A.M.; Wan Mohd Jaafar, W.S.; Mohd Said, M.N.; Saad, S.N.M.; Mohan, M. UAV implementations in urban planning and related sectors of rapidly developing nations: A review and future perspectives for Malaysia. *Remote Sens.* **2023**, *15*, 2845. [CrossRef]
19. Liu, X.; Chen, Y.; Wang, C.; Tan, K.; Li, J. A lightweight building instance extraction method based on adaptive optimization of mask contour. *Int. J. Appl. Earth Observ. Geoinf.* **2023**, *122*, 103420.
20. Zhang, X.; Wang, L.; Su, Y. Visual place recognition: A survey from deep learning perspective. *Pattern Recognit.* **2021**, *113*, 107760.
21. Zheng, Z.; Wei, Y.; Yang, Y. University-1652: A multi-view multi-source benchmark for drone-based geo-localization. In Proceedings of the 28th ACM International Conference on Multimedia, MM '20, Seattle, WA, USA, 12–16 October 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 1395–1403.
22. Peng, G.; Yue, Y.; Zhang, J.; Wu, Z.; Tang, X.; Wang, D. Semantic reinforced attention learning for visual place recognition. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 13415–13422.
23. Sarlin, P.-E.; Cadena, C.; Siegwart, R.; Dymczyk, M. From coarse to fine: Robust hierarchical localization at large scale. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–17 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 12708–12717.
24. Nie, J.; Feng, J.; Xue, D.; Pan, F.; Liu, W.; Hu, J.; Cheng, S. A training-free, lightweight global image descriptor for long-term visual place recognition toward autonomous vehicles. *IEEE Trans. Intell. Transp. Syst.* **2023**, *25*, 1291–1302.
25. Zhuang, J.; Dai, M.; Chen, X.; Zheng, E. A faster and more effective cross-view matching method of UAV and satellite images for UAV geolocalization. *Remote Sens.* **2021**, *13*, 3979. [CrossRef]

26. Thakur, N.; Nagrath, P.; Jain, R.; Saini, D.; Sharma, N.; Hemanth, D.J. Artificial intelligence techniques in smart cities surveillance using UAVs: A survey. In *Machine Intelligence and Data Analytics for Sustainable Future Smart Cities*; Ghosh, U., Maleh, Y., Alazab, M., Pathan, A.-S.K., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 329–353.
27. Lowe, D.G. Distinctive image features from scaleinvariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110.
28. Bay, H.; Tuytelaars, T.; Van Gool, L. SURF: Speeded up robust features. In *Computer Vision—ECCV 2006, Proceedings of the 9th European Conference on Computer Vision, Graz, Austria, 7–13 May 2006*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 404–417.
29. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to sift or surf. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571.
30. Bampis, L.; Gasteratos, A. Revisiting the bag-of-visual-words model: A hierarchical localization architecture for mobile systems. *Rob. Auton. Syst.* **2019**, *113*, 104–119.
31. Arandjelovic, R.; Zisserman, A. All about VLAD. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1578–1585.
32. Zheng, L.; Yang, Y.; Tian, Q. SIFT meets CNN: A decade survey of instance retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 1224–1244.
33. Arandjelovic, R.; Gronat, P.; Torii, A.; Pajdla, T.; Sivic, J. NetVLAD: CNN architecture for weakly supervised place recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 5297–5307.
34. Kim, H.J.; Dunn, E.; Frahm, J.M. Learned contextual feature reweighting for image geolocalization. In Proceedings of the IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 3251–3260.
35. Hausler, S.; Garg, S.; Xu, M.; Milford, M.; Fischer, T. Patch-NetVLAD: Multi-scale fusion of locally-global descriptors for place recognition. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 14136–14147.
36. Radenovic, F.; Tolias, G.; Chum, O. Finetuning CNN image retrieval with no human annotation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 1655–1668. [PubMed]
37. Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P.; et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 6881–6890.
38. Wang, R.; Shen, Y.; Zuo, W.; Zhou, S.; Zheng, N. TransVPR: Transformer-based place recognition with multi-level attention aggregation. In Proceedings of the IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 13648–13657.
39. Keetha, N.; Mishra, A.; Karhade, J.; Jatavallabhula, K.M.; Scherer, S.; Krishna, M.; Garg, S. Anyloc: Towards universal visual place recognition. *arXiv* **2023**, arXiv:2308.00688.
40. Oquab, M.; Darcet, T.; Moutakanni, T.; Marc Szafraniec, H.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; Assran, M.; et al. Dinov2: Learning robust visual features without supervision. *arXiv* **2023**, arXiv:2304.07193.
41. Ali-Bey, A.; Chaib-Draa, B.; Giguere, P. MixVPR: Feature mixing for visual place recognition. In Proceedings of the 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–7 January 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 2997–3006. [CrossRef]
42. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2980–2988.
43. Cai, Z.; Vasconcelos, N. Cascade R-CNN: High quality object detection and instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 1483–1498.
44. Zhang, G.; Lu, X.; Tan, J.; Li, J.; Zhang, Z.; Li, Q.; Hu, X. RefineMask: Towards high-quality instance segmentation with fine-grained features. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021), Virtual, 19–25 June 2021; pp. 6857–6865.
45. Wang, W.; Shi, Y.; Zhang, J.; Hu, L.; Li, S.; He, D.; Liu, F. Traditional village building extraction based on improved mask R-CNN: A case study of beijing, China. *Remote Sens.* **2023**, *15*, 2616. [CrossRef]
46. Wang, X.; Kong, T.; Shen, C.; Jiang, Y.; Li, L. SOLO: Segmenting objects by locations. In Proceedings of the Computer Vision–ECCV 2020, Glasgow, UK, 23–28 August 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 649–665.
47. Wang, X.; Zhang, R.; Kong, T.; Li, L.; Shen, C. SOLOv2: Dynamic and fast instance segmentation. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, BC, Canada, 6–12 December 2020; pp. 17721–17732.
48. Kirillov, A.; Levinkov, E.; Andres, B.; Savchynskyy, B.; Rother, C. InstanceCut: From edges to instances with MultiCut. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5008–5017.
49. Wagner, F.H.; Dalagnol, R.; Tarabalka, Y.; Segantine, T.Y.F.; Thomé, R.; Hirye, M.C.M. U-Net-Id, an instance segmentation model for building extraction from satellite images—Case study in the Joanópolis City, Brazil. *Remote Sens.* **2020**, *12*, 1544. [CrossRef]

50. Iglovikov, V.; Seferbekov, S.; Buslaev, A.; Shvets, A. TernausNetV2: Fully convolutional network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision, and Pattern Recognition (CVPR) Workshops, Salt Lake City, UT, USA, 18–22 June 2018.
51. Cheng, B.; Misra, I.; Schwing, A.G.; Kirillov, A.; Girdhar, R. Masked-attention mask transformer for universal image segmentation. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1280–1289.
52. Fu, W.; Xie, K.; Fang, L. Complementarity-aware local-global feature fusion network for building extraction in remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5617113.
53. Chen, K.; Liu, C.; Chen, H.; Zhang, H.; Li, W.; Zou, Z.; Shi, Z. RSPrompter: Learning to prompt for remote sensing instance segmentation based on visual foundation model. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 4701117.
54. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.-Y.; et al. Segment anything. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 4015–4026.
55. He, X.; Zhou, Y.; Zhou, Z.; Bai, S.; Bai, X. Triplet-Center Loss for Multi-View 3D Object Retrieval. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1945–1954. [CrossRef]
56. Chen, M.; Hu, Q.; Yu, Z.; Thomas, H.; Feng, A.; Hou, Y.; McCullough, K.; Ren, F.; Soibelman, L. STPLS3D: A large-scale synthetic and real aerial photogrammetry 3D point cloud dataset. *arXiv* **2022**, arXiv:2203.09065.
57. 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.* **2019**, *57*, 574–586.
58. Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common objects in context. In Proceedings of the Computer Vision–ECCV 2014, Zurich, Switzerland, 6–12 September 2014; Springer: Cham, Switzerland, 2014; pp. 740–755.
59. Wu, Y.; Kirillov, A.; Massa, F.; Lo, W.Y.; Girshick, R. Detectron2. 2019. Available online: https://github.com/facebookresearch/detectron2 (accessed on 16 September 2024).
60. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv* **2017**, arXiv:1711.05101.
61. Ali-bey, A.; Chaib-draa, B.; Giguère, P. GSV-Cities: Toward appropriate supervised visual place recognition. *Neurocomputing* **2022**, *513*, 194–203.
62. Cao, B.; Araujo, A.; Sim, J. Unifying deep local and global features for image search. In Proceedings of the 16th European Conference, Computer Vision–ECCV 2020, Glasgow, UK, 23–28 August 2020; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 726–743.
63. Wang, X.; Han, X.; Huang, W.; Dong, D.; Scott, M.R. Multi-similarity loss with general pair weighting for deep metric learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.