CAMP: A Cross-View Geo-Localization Method Using Contrastive Attributes Mining and Position-Aware Partitioning

Qiong Wu, *Graduate Student Member, IEEE*, Yi Wan, *Member, IEEE*, Zhi Zheng, *Member, IEEE*, Yongjun Zhang, *Member, IEEE*, Guangshuai Wang, and Zhenyang Zhao

***Abstract*— Cross-view geo-localization (CVGL) task aims to utilize geographic data, such as maps or high-resolution satellite images, as reference to estimate the positions of a ground- or near-ground- captured query image. This task is particularly challenging due to the significant changes in visual appearance resulting from the extreme viewpoint variations. To address this challenge, a range of innovative methods have been proposed. However, intra-scene geometric information and inter-scene discriminative representation are not fully explored. In this article, we propose a novel CVGL method using contrastive attributes mining and position-aware partitioning (CAMP), which incorporates a position-aware partition branch (PPB) and a contrastive attributes mining (CAM) strategy. PPB learns fine-grained local features of different parts and captures their spatial information, providing a comprehensive understanding of scenes from both textual and spatial perspectives. CAM establishes supervision of the negative samples based on the images from the same platform, empowering the model to better discern differences between distinct scenes without extra memory cost. The proposed CAMP surpasses existing methods, achieving state-of-the-art results on the satellite-drone CVGL datasets University-1652 and SUES-200. Additionally, our method also outperforms existing methods in cross-dataset generalization, achieving an 8.85% increase in R@1 when trained on the University-1652 dataset and tested on the SUES-200 dataset at a height of 150 m. Our code and model are available at https://github.com/Mabel0403/CAMP.**

***Index Terms*— Cross-view geo-localization (CVGL), image retrieval, remote sensing, satellite image, unmanned aerial vehicles (UAVs).**

I. Introduction

CROSS-VIEW geo-localization (CVGL) aims to determine the geographical location of query data based on the reference data, despite the perspective differences between them. Image-level CVGL is to retrieve images captured from the same scene but on different platforms, with applications spanning various domains [1], [2]. For example, when presented with a drone-view image, the system intends to seek corresponding images within a database of satellite images [3]. These satellite images come pre-annotated with geo-tags, facilitating the process of accurately pinpointing the location of targets seen in the drone image. Image-level CVGL not only facilitates the identification of corresponding scene images for structure from motion (SfM) processing in aerial photogrammetry but also enables the georeferencing of visual intelligence data (such as images and videos), allowing for location-based intelligence gathering. Moreover, it provides initial position information for more precise geo-localization. However, due to the significant changes in visual appearance resulting from extreme viewpoint variations, CVGL remains a highly challenging task.

Previous researchers have devoted considerable effort to image-level CVGL [4], [5], [6], with a primary focus on two tasks: 1) using ground panoramic images as queries and aerial images (drone or satellite images) as references and 2) employing both drone and satellite images interchangeably as queries and references. Specially, matching drone and satellite images can be broadly categorized into two main applications: drone-view target localization and drone navigation.

Recent years have seen significant advancements in CVGL, largely driven by progress in deep learning [7], [8], [9]. Scholars have combined emerging technologies (like convolutional neural networks [CCNs], attention mechanisms, and contrastive learning) to converge on a feature space, which effectively brings matched image pairs closer while pushing unmatched pairs further apart. For instance, Shi et al. [1] utilize polar transformation to convert images from satellite perspective to ground perspective. LPN [10] employs a square-ring partitioning method to extract features from the background of the images. FSRA [11] segments images based on semantic information and retrieval matched images with implicit semantic local features. Sample4Geo [12] introduces two hard negative sample mining strategies, enabling the model to focus on distinguishing between similar but distinct scenes. On the basis of the work so far available, we summarize that the key of CVGL hinges on the model's ability for scene identification, which depends on two aspects: 1) the model's ability to extract invariant features across different perspectives

Manuscript received 18 March 2024; revised 7 July 2024 and 4 August 2024; accepted 19 August 2024. Date of publication 23 August 2024; date of current version 3 September 2024. This work was supported in part by China Railway Group Laboratory Basic Research Project under Grant L2023G014, in part by the National Natural Science Foundation of China under Grant 42030102, in part by the Major Special Projects of Guizhou under Grant [2022]001, and in part by Tianjin Key Laboratory of Rail Transit Navigation Positioning and Spatio-Temporal Big Data Technology under Grant TKL2023B09. *(Corresponding author: Yi Wan.)*

Qiong Wu, Yi Wan, and Yongjun Zhang are with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: mabel_wq@whu.edu.cn; yi.wan@whu.edu.cn; zhangyj@whu.edu.cn).

Zhi Zheng is with the Department of Geography and Resource Management, The Chinese University of Hong Kong, Shatin, Hong Kong, China (e-mail: zhizheng@cuhk.edu.hk).

Guangshuai Wang and Zhenyang Zhao are with China Railway Design Group Company Ltd., Tianjin 300308, China (e-mail: gswang0806@126.com; zhaozhenyang@crdc.com).

Digital Object Identifier 10.1109/TGRS.2024.3448499

of the scene and 2) the model's ability to distinguish between distinct scenes. However, most methods fail to simultaneously integrate scene texture and spatial information when extracting invariant features of the scene. Additionally, when comparing the differences between scenes, they overlook the contrastive constraints between images from the distinct scenes but the same platform.

Integrating these two aspects, we introduce a CVGL method using contrastive attributes mining and position-aware partitioning, named CAMP. The position-aware partition branch (PPB) in CAMP is designed to perceive scenes from both texture and spatial perspectives, which conducts part-based feature extraction and emphasizes the positional information before partitioning the scene features. PPB yields fine-grained, position-aware features, enhancing model's ability to extract invariant features across different perspectives of the scene. Based on the previous methods that applied contrastive learning architectures to the field of CVGL, we proposed a contrastive attributes mining (CAM) strategy to add contrastive constraints between images from the same platform, which has been overlooked before. CAM strategy significantly augments the number of negative samples, enhancing the model's ability to distinguish between distinct scenes without increasing additional memory cost.

In short, the main contributions of this article are as follows.

1) A PPB is utilized in the contrastive learning stage, extracting and aligning the fine-grained, position-aware features of each scene, to enhance the model's perception of geographical scenes.
2) A CAM strategy is proposed, adding contrastive constraints between images from the same platform. It empowers the discrimination of the extracted scene feature without increasing additional memory cost during the training stage.
3) The experimental results on and across the University-1652 and SUES-200 datasets demonstrate that our CAMP achieves state-of-the-art performance on both tasks of drone-view target localization and drone navigation and has excellent generalization.

The rest of this article is organized as follows. In Section II, we briefly introduce some of the relevant works. Section III presents our proposed CAMP in detail. Experimental results are presented in Section IV, followed by the conclusion in Section V.

II. Related Work

In this section, we briefly review related previous works, including image-based CVGL and contrastive visual representation learning.

A. Image-Level CVGL

CVGL has attracted widespread attention in recent years due to its extensive and promising applications. Image-based CVGL has been approached as an image retrieval task, with early researches relying on hand-crafted operators to extract and align features from images captured at different viewpoints [13], [14], [15].

With the rapid development of deep learning, CNNs have made remarkable progress in extracting image features. Workman and Jacobs [9] pioneered using a pre-trained CNN, specifically AlexNet [16], for extracting scene features in CVGL. Subsequently, Workman et al. [17] fine-tuned the pre-trained feature extractor using information from image pairs, leading to improved performance. This sparked a series of works aimed at leveraging constraints between scenes to train models and enhance their discriminative capability. Lin et al. [18] adopted methods from face recognition, utilizing contrastive loss to train a Siamese network. Vo and Hays [19] analyzed limitations of Siamese networks in CVGL tasks and proposed soft-margin triplet loss to improve localization accuracy. Hu et al. [20] introduced a weighted soft-margin ranking loss, which enhanced both convergence speed and localization accuracy. Cai et al. [21] mined hard samples in training batches to strengthen the penalty of soft-margin triplet loss. Zheng et al. [4] grouped images from the same scene into the same category and introduced instance loss to learn discriminative features. Wang et al. [22] refined the widely used Barlow Twins method in contrastive learning by introducing dynamic weighted decorrelation regularization, motivating models to learn discriminative embeddings by removing feature redundancy. Deuser et al. [12] proposed a simplified yet effective architecture based on contrastive learning, incorporating two novel sampling strategies to mine hard negatives, significantly improving the model's ability to distinguish different scenarios.

Another line of work focused on addressing spatial misalignment issues caused by extreme viewpoint variations and extracting common features from cross-view image pairs. Hu et al. [20] used NetVLAD to extract local features to reduce the visual gap between images from different viewpoints. Liu and Li [5] encoded orientation information into feature maps to better align features of images from the same scene. Shi et al. [23] first attempted to use optimal transport theory to close the spatial layout information of high-level features. Subsequently, Shi et al. [1] directly applied polar transformation to align satellite images to ground view, achieving pixel-level alignment. Wang et al. [10] proposed a square-ring partition strategy to utilize background information in images. Lin et al. [24] designed a method for automatically detecting salient keypoints, improving the model's robustness to appearance changes. Dai et al. [11] performed patch-level segmentation of images, followed by region-level alignment. Shen et al. [25] achieved cross-dimension feature interaction for feature alignment from both spatial and channel perspectives. Zhao et al. [26] implicitly learned salient features of scenes and dynamically aggregated contextual information.

B. Contrastive Learning

Contrastive learning has been widely employed in both supervised and self-supervised visual representation learning. The fundamental concept of contrastive learning involves minimizing the distance between anchor and positive pairs while maximizing the distance between anchor and negative pairs through a contrastive loss function, aiming to derive a

discriminative feature space [27]. Self-supervised contrastive learning guides models to learn more general features and achieve superior performance in downstream tasks [28], [29]. In supervised learning, contrastive learning is frequently utilized in deep metric learning, including tasks such as image classification, face recognition, person re-identification, and image retrieval.

In earlier works, InstDisc [30] contended treating each image as a class, leading to a series of contrastive representation learning methods based on the discriminative proxy task. These methods use the augmented results of anchor as positive samples and augmented results of other data as negative samples. CMC [31] suggested defining scenes as the unit of samples, where images from the same scene but with different modalities or perspectives act as positive samples for each other, while data from other scenes act as negative samples. CPC [32] introduced a contrastive learning method based on generative models, taking multimodal data as input and utilizing autoregression to predict future inputs, thus generating positive and negative samples through generative model-based data augmentation.

Inspired by prior work, contrastive learning has become a prominent research focus. He et al. [33] proposed MoCo, incorporating a momentum encoder module to ensure consistency in sample encodings during training, surpassing the performance of supervised methods in downstream tasks. Chen et al. [34] introduced SimCLR, enhancing model performance by eliminating domain differences between branches through a projector added to the network's backend. Subsequently, MoCo v2 [35] and SimCLR v2 [36] were proposed, further optimizing data augmentation methods, learning rate adjustment strategies, and the momentum encoder to achieve improved results.

However, contrastive learning based on discriminative models requires careful balancing of positive and negative sample constraints to avoid model collapse. To address this, Grill et al. [37] proposed a contrastive representation learning method based on generative proxy tasks, which eliminates dependence on negative samples by establishing mappings between branches, thereby enhancing training efficiency and stability. This structure has since become one of the mainstream contrastive learning approaches. MoCo v3 [38] adopts a vision transformer as the backbone network for contrastive learning, improving the tokenization part of the model's frontend. DINO [39] introduces centering operations to prevent model collapse effectively.

Due to the similarity in input data formats, scholars naturally apply contrastive learning mechanisms to cross-view geolocation [12], [22]. However, these approaches neglect the intrinsic relationship between images from the same platform. To effectively harness this relationship, we propose the CAM strategy. By integrating homologous image pairs into the contrastive learning process, CAM enhances the model's capability to discern differences between distinct scenes. By reinforcing contrastive constraints, CAM optimizes information utilization during training without necessitating additional memory cost.

III. Proposed Method

In this section, we introduce our proposed method CAMP. The complete network structure is shown in Fig. 1. Similar to the general contrastive learning framework, we first encode cross-view images as corresponding features. In CAMP, global and local features are extracted by ConvNeXt network and PPB. Then, we use a contrastive optimization procedure to train the features extracting module so that the feature extractor can distinguish the same or different scenes from cross-view images. The CAM strategy and symmetric InfoNCE loss are used to strengthen the constraints between distinct scenes, supervising the model to narrow the distance between images from the same scene and enlarge the gap between images from different scenes.

Problem Formulation: Given a geo-localization dataset, denote the drone image as x_{i_t} and the satellite image as y_j. $i, j \in [1, R]$, where R indicates the number of scenes. i denotes the scene id of the image x_{i_t}, and j denotes the scene id of the image y_j. $i = j$ means that x_{i_t} and y_j are collected from the same scene. $t \in [1, N]$, where N indicates the number of drone images in each scene. X denotes a batch of drone images x_i^t, while Y denotes a batch of satellite images y_j. The task of CVGL can be described as follows: Given a query image x_{i_t}, and reference images $\{y_1, y_2, \ldots, y_R\}$, find the reference y_i that best matches x_{i_t} on the feature space $\mathcal{F}$ and the similarity metric $\mathrm{sim}(x, y)$. Formally, find

$$y_i = \mathrm{argmax}_{y_j} \mathrm{sim}(\mathcal{F}(x_{i_t}), \mathcal{F}(y_j)). \tag{1}$$

In the following, the contrastive learning pipeline of CAMP is described in Section III-A, followed by the details of PPB in Section III-B. Then the CAM strategy to strengthen the contrast constraints between distinct scenes is introduced in Section III-C.

A. Contrastive Attributes Mining and Position-Aware Partitioning

Our method adopts a contrastive learning architecture. Similar to other methods utilizing contrastive learning [40], our pipeline comprises two main components: 1) feature extraction and 2) contrastive optimization.

1) Feature Extraction: To extract rich contextual information, CAMP first adopts ConvNeXt-B as its backbone and then PPB to capture features. Following [4], we share weights between the satellite-view branch and drone-view branch due to the shared patterns observed in aerial views from both sources. ConvNeXt [41] is a standard CNN-based network known for its comparable performance to the Vision Transformer network [42], [43] in terms of both processing speed and accuracy, albeit with a simpler design.

Given an input $X \in \mathbb{R}^{B \times H \times W \times C_0}$ for drone branch, where B, H, W, C_0 represent its batch size, height, width, and channels. X undergoes a series of transformations starting with a convolution layer followed by a layer normalization, generating features with dimensions 1/4 of the input size. Subsequently, it passes through four stages, each composed

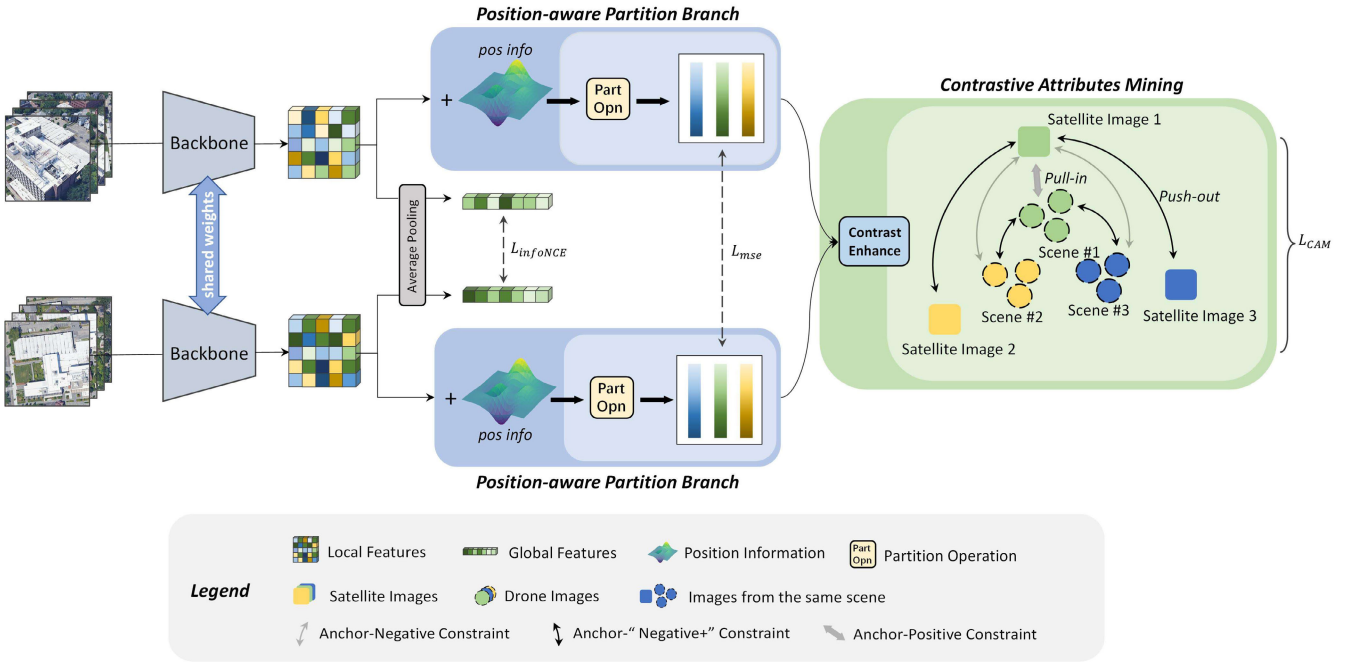


Fig. 1. Pipeline of our proposed CAMP. The blue-highlighted region in the figure represents PPB, extracting local features. This module injects positional information into the image feature map, followed by partitioning to obtain features from different blocks. The green-highlighted region represents the CAM strategy, enhancing the model's ability to distinguish between distinct scenes without increasing additional memory cost. The contrastive constraints for Scene #1 (green elements) are depicted. The gray arrows represent the original constraints between the anchor sample and the positive/negative samples, while the black arrows represent the constraints between the anchor sample and the newly added "Negative+" samples introduced by the CAM strategy.

of a downsample layer and ConvNeXt blocks B_c, where $c = \{1, 2, 3, 4\}$. The resulting output $L^{(X)} \in \mathbb{R}^{B\times N\times C}$, where N denotes the number of feature map elements, and C denotes the number of channels of the ConvNeXt Layer $\mathcal{F}_{\text{ConvNeXtLayer}}$, can be represented as follows:

$$L^{(X)} = \mathcal{F}_{\text{ConvNeXtLayer}}(X). \tag{2}$$

To extract comprehensive feature representations, we input feature $L^{(X)}$ into two branches. In the global branch, the global average pooling operation is applied to $L^{(X)}$ and it is transformed into a $B \times 1 \times C$-dim feature vector $L^{(X)}_{\text{global}}$, which is denoted as the global feature vector of the corresponding input X.

Meanwhile, in the local branch, feature $L^{(X)}$ serves as input to the PPB. PPB partitions the feature map into K different part features based on the feature value and location, resulting in the local features $L^{(X)}_{\text{local}_k}$, where $k \in [1, K]$. This process is detailed in Section III-B. Combining the outputs of these two branches, we obtain both global and local features corresponding to the input images.

2) Contrastive Optimization: CAMP employs symmetric InfoNCE loss, mean-square error (MSE) loss, and CAM loss to optimize the feature extractor [32], ensuring effective learning of the differences and similarities among various scene features. Given an input $Y \in \mathbb{R}^{B\times H\times W\times C_0}$ for the satellite branch, where B, H, W, C_0 represent its batch size, height, width, and channels. In CAMP, the treatment of Y parallels that of X, resulting in the final feature extractor extracting both the global feature $L^{(Y)}_{\text{global}}$ and local features $L^{(Y)}_{\text{local}_k}$, where $k \in [1, K]$. Following feature extraction, we proceed with the computation of loss to supervise the feature extractor. This procedure can be represented as follows:

$$\text{Loss} = L_{\text{Sin}fN} + m * L_{\text{MSE}} + n * L_{\text{CAM}} \tag{3}$$

where

$$L_{\text{Sin}fN} = \mathcal{L}\left(L^{(X)}_{\text{global}}, L^{(Y)}_{\text{global}}\right)_{\text{Sin}fN}$$

$$L_{\text{MSE}} = \frac{1}{K}\sum_{k=1}^{K}\mathcal{L}\left(L^{(X)}_{\text{local}_k}, L^{(Y)}_{\text{local}_k}\right)_{\text{MSE}}$$

$$L_{\text{CAM}} = \frac{1}{K}\sum_{k=1}^{K}\mathcal{L}\left(L^{(X)}_{\text{local}_k}, L^{(Y)}_{\text{local}_k}\right)_{\text{CAM}}.$$

The Loss represents the final loss directly used for supervising the model. $\mathcal{L}_{\text{Sin}fN}$, $\mathcal{L}_{\text{MSE}}$, and $\mathcal{L}_{\text{CAM}}$ represent the symmetric InfoNCE loss, MSE loss, and CAM loss, respectively. MSE loss is a classic loss function in deep learning, which is widely favored for its mathematical interpretability and ease of computation in various regression tasks. It is a measure used to assess the performance of models by computing the average squared difference between two features, with lower MSE indicating better model fit to the data. InfoNCE loss is a standard loss function in metric learning, while Symmetric InfoNCE loss is a refined version tailored for the specific demands of image representation learning [44]. CAM loss, derived from our proposed CAM strategy, is based on InfoNCE loss and serves to augment the number of negative samples, thereby fortifying the feature contrast constraints between distinct scenes. The specific definitions of InfoNCE loss function and CAM loss function are elucidated in Section III-C. Besides, the scaling coefficients m and n in the formula are set to 1.0.

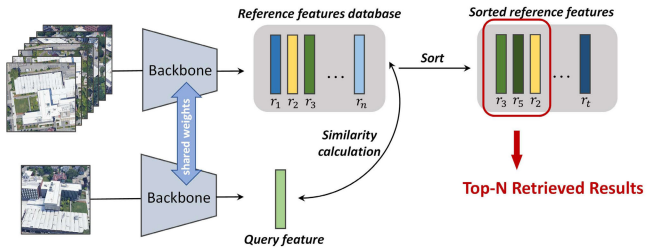


Fig. 2. Retrieval process of our method. We use the backbone network to encode images into features and obtain retrieval results based on the similarity between the query feature and reference features.

3) Retrieval Process: CVGL tasks require high processing speed and low storage usage. Therefore, the features for both the reference and query images in the retrieval phase are obtained from the last layer of the backbone. The dimension of these extracted features is 1×1024, effectively balancing expressiveness and simplicity. They contain sufficient information while being compact, ensuring quick computations during retrieval without occupying excessive space. Specifically, we demonstrate the retrieval process in Fig. 2. First, all reference images are encoded into 1×1024 feature vectors using the backbone, which construct a reference features database. For each query image, its feature vector is extracted using the same model, resulting in a query feature vector. Second, the distances between the query feature vector and each of the reference feature vectors are calculated. Finally, the reference images are sorted based on their calculated distances from the query image. The top N closest reference images are then selected based on this sorted order. This process ensures that the most similar images to the query are retrieved efficiently.

B. Position-Aware Partition Branch

While extracting global features that are robust and contextually connected is essential, numerous prior studies have demonstrated the superior effectiveness of part-based methods for image retrieval. Therefore, to direct the model's attention to the details of scenes, we introduce PPB, which generates local features with fine-grained information. Unlike the mainstream part-based geographic scene retrieval methods [10], [11] (based on manual spatial partitioning and automatic partitioning based on feature values), our proposed PPB prioritizes spatial positional information before adaptive partitioning. This ensures that each local feature has distinct positional attributes, enabling the feature extractor to capture fine-grained, position-aware information that assists the model in discerning images from different scenes.

In Fig. 1, the blue-highlighted region illustrates the process of the PPB, which incorporates positional information for the input $L^{(X)} \in \mathbb{R}^{B\times N\times C}$ and divide it into K parts. Although the ConvNeXt network inherently retains positional information of features, direct partitioning of final layer output features based solely on their values neglects the positions of individual elements. Therefore, we emphasize positional information after the backbone, followed by partitioning operation. Specifically, for each element in the feature map, we add positional values and compute the mean of its feature value across all channels. Subsequently, we sort all elements in descending order based on their channel-wise mean feature values and partition them equally according to the number of partitions K. Denote $P \in \mathbb{R}^{B\times N\times C}$ and represents positional encoding information, which is initialized randomly and automatically adjusted during the training process. Following [11], the number of partitions K is set to 3. In Section IV-E, we validated the effectiveness of the PPB.

C. Mining Contrastive Attributes

Contrastive learning can be viewed as the task of querying a dynamic dictionary. It is desirable to build dictionaries that are: 1) large and 2) evolves consistently during the training process. Intuitively, a larger dictionary facilitates better sampling of continuous, high-dimensional visual spaces, and the keys in dictionary should be represented by similar or identical encoders so that their comparison with queries is consistent [33]. Significantly increasing the batch size is evidently the most straightforward method to build the ideal dictionary mentioned above. However, this demands computational resources to an impractical extent. Our CAM strategy, using contrastive constraints between images from same platform, improves feature extractor's ability to distinguish between distinct scenes. It augments the number of negative samples without extra memory cost, offering a partial solution to the problem aforementioned.

In the green-highlighted region of Fig. 1, the CAM strategy utilizes the contrast between images from the same platform but distinct scenes, tripling the number of negative samples in each training iteration. The straight arrow in the figure represents the constraint that supervise the model to pull the anchor closer to the sample, while the curved arrows represent the constraints that supervise the model to push the anchor further away from the samples. The gray arrows denote common constraints in previous methods, while the black arrows represent the additional negative sample constraints introduced by the CAM strategy. We refer to the newly introduced negative samples as "Negative+" samples, which are captured from the same platform. CAM implements constraints in loss calculation, based on infoNCE loss function. Given a batch of drone images $x_1, x_2, \ldots, x_B$ and their corresponding satellite images $y_1, y_2, \ldots, y_B$, the feature extractor output features $L^{(X)}(L_1^{(X)}, L_2^{(X)}, \ldots, L_B^{(X)})$ and $L^{(Y)}(L_1^{(Y)}, L_2^{(Y)}, \ldots, L_B^{(Y)})$, where B represents the batch size. Consistent subscripts indicate images (or features) from the same geographical scene and serve as positive samples for each other. The process of computing the loss can be represented by the following equations:

$$\mathcal{L}(L^{(X)}, L^{(Y)})_{\text{CAM}} = \mathcal{L}(L^{(X)}, L^{(Y)})_{\text{Sin}fN} + \lambda_1 \mathcal{L}(L^{(X)}, L^{(X)})_{\text{in}fN} + \lambda_2 \mathcal{L}(L^{(Y)}, L^{(Y)})_{\text{in}fN} \tag{4}$$

where

$$\mathcal{L}(L^{(X)}, L^{(Y)})_{\text{in}fN} = -\frac{1}{B}\sum_{j=0}^{B}\left(\log \frac{\exp\left(L_j^{(X)} \cdot \frac{L_j^{(Y)}}{\tau}\right)}{\sum_{i=0}^{B}\left(\exp\left(L_j^{(X)} \cdot \frac{L_i^{(Y)}}{\tau}\right)\right)}\right)$$

$$\mathcal{L}(L^{(X)}, L^{(Y)})_{\text{Sin}fN} = \mathcal{L}(L^{(X)}, L^{(Y)})_{\text{in}fN} + \mathcal{L}(L^{(Y)}, L^{(X)})_{\text{in}fN}.$$

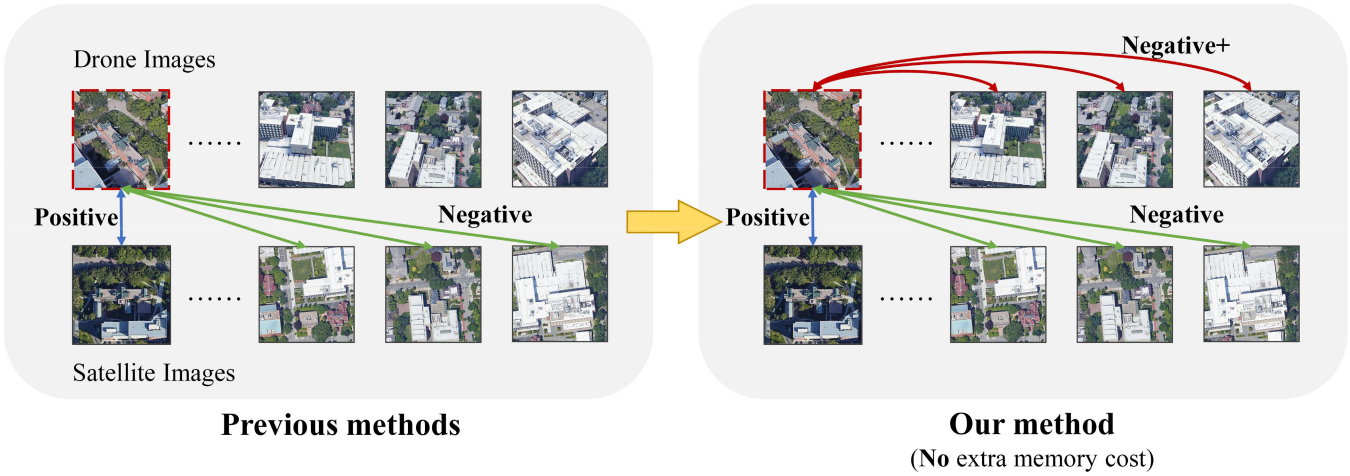


Fig. 3. Comparison of contrastive attributes between previous methods and our CAM. In previous methods where contrastive learning was applied to CVGL tasks, images from the same scene but distinct platforms were considered positive samples (as shown by the blue arrows), while images from distinct scenes and platforms were considered as negative samples (as shown by the green arrows). Our approach leverages the relationships between images from the same source, adding a considerable number of newly negative samples named "Negative+ samples" (as indicated by the red arrows). Furthermore, this process incurs no extra memory cost.

$\mathcal{L}_{\text{in}fN}$ represents the InfoNCE loss, quantifying the similarity between query and reference images using dot-product. It yields low values when the query and the positive match are similar, and high values when the negative references are dissimilar to the query. The cross-entropy is computed as the loss function for measuring the similarity between views. The temperature parameter τ, which can either be learned or set to a fixed value, adjusts the scale of the loss function. InfoNCE loss primarily used for unsupervised image representation learning in an asymmetric manner, but symmetric formulations are proven useful to bridge the gap between different domains. Therefore, we use the same symmetric approach $L_{\text{Sin}fN}$ to leverage the supervision in both directions. In (4), $\mathcal{L}(L^{(X)}, L^{(Y)})_{\text{Sin}fN}$ calculates the InfoNCE loss between drone and satellite images, a commonly used approach in contrastive learning research to align features from different domains. $\mathcal{L}(L^{(X)}, L^{(X)})_{\text{in}fN}$ computes the InfoNCE loss between drone images from distinct scenes, effectively mining negative samples from the original drone images. Similarly, $\mathcal{L}(L^{(Y)}, L^{(Y)})_{\text{in}fN}$ calculates the InfoNCE loss between satellite images from distinct scenes, enhancing feature extractor's ability to distinguish between distinct scenes. In summary, our CAM loss is composed of the contrastive learning loss between samples from different platforms (drone and satellite) and within the same platform (intra-drone and intra-satellite). This approach increases the number of negative samples within the same platform without requiring additional total samples, significantly improving the model's ability to distinguish between scenes while maintaining the same memory consumption. Additionally, the scaling factors λ_1 and λ_2 in (4) both control the magnitude of loss between anchors and Negative+ samples in the drone and satellite branches, respectively. The losses in both branches share a common objective: to facilitate the model in better learning the distinctions between scenes. Therefore, we constrain λ_1 and λ_2 according to the following equation and treat them as hyper parameters to be learned during the training process:

$$\lambda_1 \cdot \lambda_2 = 1. \tag{5}$$

Moreover, since Negative+ samples originate from the existing samples within the batch, the CAM strategy does not require additional memory cost to introduce new negative samples, thereby simplifying its implementation. In Section IV-E, we validate the effectiveness of the CAM strategy.

IV. Experiment

We first introduce two CVGL datasets and the evaluation protocols. Then, Section IV-B describes the implementation details. We provide the comparison with the state-of-the-arts in Section IV-C, followed by the ablation studies in Section IV-D.

A. Datasets and Evaluation Protocols

CAMP is proposed to solve the CVGL between satellite-view images and drone-view images. Therefore, we train and evaluate our method on two mainstream CVGL datasets: University-1652 [4] and SUES-200 [45].

1) University-1652 is a large-scale multiview multisource dataset containing synthetic drone-view images, satellite-view images, and ground-view images. It pioneered the integration of drone images for CVGL, proposing two new tasks, i.e., drone navigation (Satellite→Drone) and drone-view target localization (Drone→Satellite). University-1652 collects 50 218 training images in total, and has 71.46 images per class on average, captured from 1652 buildings of 72 universities. Moreover, the buildings in the training set and the test set have no overlap in University-1652.
2) SUES-200 is the latest cross-view matching dataset, which contains images from two views, drone-view, and satellite-view. It provides diverse scenes and height

TABLE I
COMPARISON WITH STATE-OF-THE-ART RESULTS ON UNIVERSITY-1652

Method	Publication	Drone→Satellite		Satellite→Drone	
		R@1	AP	R@1	AP
LPN [10]	TCSVT'22	75.93	79.14	86.45	74.79
FSRA [11]	TCSVT'22	82.25	84.82	87.87	81.53
MSBA [46]	RemoteSensing'21	82.33	84.78	90.58	81.61
TransFG [26]	TGRS'24	84.01	86.31	90.16	84.61
Swin-B+DWDR [22]	arXiv'22	86.41	88.41	91.30	86.02
MBF [47]	Sensors'23	89.05	90.61	93.15	88.17
SeGCN [48]	JSTARS'24	89.18	90.89	94.29	89.65
MCCG [25]	TCSVT'23	89.64	91.32	94.30	89.39
Sample4Geo [12]	ICCV'23	92.65	93.81	95.14	91.39
CAMP (Ours)	**-**	**94.46**	**95.38**	**96.15**	**92.72**

views for each scene. Images in SUES-200 are from four height views (150, 200, 250, and 300 m), acquired in real environments of multiple types of scenes, i.e., parks, schools, lakes, and public buildings near the Shanghai University of Engineering Science. There are 50 drone-view images per height view and one corresponding satellite-view image for each location. Therefore, experiments on this dataset, especially images with low flight height, are more challenging.

3) *Evaluation protocol:* In our experimental evaluations, we employ Recall@K (R@K) and average precision (AP) metrics to assess the effectiveness of our model. R@K quantifies the percentage of correctly matched images within the top-K of the ranking list, where a higher recall score indicates superior model performance. The formula for R@K is

$$\text{Recall@K} = \frac{n_K}{N_q} \tag{6}$$

where n_K represents the count of query images for which the correct location is found within the top K results and N_q represents the total number of query images tested. Additionally, we compute the area under the precision–recall curve, denoted as AP, whose formula is

$$\text{AP}(q) = \sum_{i_q=1}^{n_q} (R(i_q) - R(i_q - 1)) \cdot P(i_q) \tag{7}$$

where q represents the query item. $P(i)$ represents the precision at position i in the list of retrieved results of q and $R(i)$ represents the recall at position i, showing the proportion of correctly localized queries up to position i. n_q represents the total number of correct reference images of q. AP evaluates the retrieval performance by considering the precision at various recall levels, which provides insight into the precision–recall trade-off in retrieval performance.

B. Implementation Details

Our method was implemented using the PyTorch platform, and all experiments were conducted on a desktop computer running Ubuntu 22.04 with an NVIDIA GeForce RTX 4090 GPU.

1) In Terms of Network Structure: In our experiments, the ConvNeXt-B with 88M parameters is used as backbone. The backbone networks of the satellite-view branch and the drone-view branch share weights.

2) In Terms of Training Strategy: Each experiment is conducted with a batch size of 24 using the AdamW optimizer. We set the initial learning rate of 0.001 and the cosine learning rate scheduler with a one-epoch warmup period. The position information used truncated normal initialization. Data augmentation included resizing to 384 × 384, horizontal flipping, random padding, rotation, grid dropout, cropping, and color jitter. Moreover, we use a custom sample strategy to prevent the occurrence of multiple images from the same class within the batch.

3) In Terms of Loss Function: To mitigate overfitting during training, we incorporate label smoothing with a value of 0.1 into the InfoNCE loss function and consider the temperature parameter τ as a trainable parameter. Furthermore, we treat the scaling factors λ_1 and λ_2 in the CAM loss function as hyperparameters subject to learning.

4) In Terms of Testing: We utilize the Euclidean distance metric to compute the similarity between the query image and the candidate images in the satellite gallery.

C. Comparison With the State-of-the-Arts

1) Results on University-1652: As shown in Table I, we compare our method with other competitive approaches on University-1652. The proposed CAMP method has achieved 94.46% Recall@1 accuracy and 95.38% AP on Drone→Satellite and 96.15% Recall@1 accuracy and 92.72% AP on Satellite→Drone. All experiments only use drone and satellite views for training. The performance of CAMP has surpassed the reported result of other competitive methods.

Specially, the performance has surpassed the SOTA method [12] of about 2% R@1 improvement in the drone-view target localization task (Drone→Satellite), establishing a new state-of-the-art. The notable performance gain can be attributed to CAMP's advanced ability to effectively capture and differentiate geographic features from both drone and satellite perspectives. This capability significantly enhances precision in cross-view localization tasks by ensuring that even subtle differences in geographic characteristics are accurately

TABLE II
COMPARISON WITH STATE-OF-THE-ART RESULTS ON SUES-200

Drone→Satellite									
Method	Publication	150m		200m		250m		300m	
		R@1	AP	R@1	AP	R@1	AP	R@1	AP
SUES-200 Baseline [45]	TCSVT'23	55.65	61.92	66.78	71.55	72.00	76.43	74.05	78.26
LPN [10]	TCSVT'22	61.58	67.23	70.85	75.96	80.38	83.80	81.47	84.53
FSRA [11]	TCSVT'22	68.25	73.45	83.00	85.99	90.68	92.27	91.95	91.95
MCCG [25]	TCSVT'23	82.22	85.47	89.38	91.41	93.82	95.04	95.07	96.20
SeGCN [48]	JSTARS'24	90.80	92.32	91.93	93.41	92.53	93.90	93.33	94.61
Sample4Geo [12]	ICCV'23	92.60	94.00	97.38	97.81	**98.28**	**98.64**	99.18	99.36
CAMP (Ours)	-	**95.40**	**96.38**	**97.63**	**98.16**	98.05	98.45	**99.33**	**99.46**
Satellite→Drone									
Method	Publication	150m		200m		250m		300m	
		R@1	AP	R@1	AP	R@1	AP	R@1	AP
SUES-200 Baseline	TCSVT'23	75.00	55.46	85.00	66.05	86.25	69.94	88.75	74.46
LPN	TCSVT'22	83.75	66.78	88.75	75.01	92.50	81.34	92.50	85.72
FSRA	TCSVT'22	83.75	76.67	90.00	85.34	93.75	90.17	95.00	92.03
MCCG	TCSVT'23	93.75	89.72	93.75	92.21	96.25	96.14	98.75	96.64
SeGCN	JSTARS'24	93.75	92.45	95.00	93.65	96.25	94.39	97.50	94.55
Sample4Geo	ICCV'23	**97.50**	93.63	**98.75**	96.70	**98.75**	**98.28**	98.75	98.05
CAMP (Ours)	-	96.25	**93.69**	97.50	**96.76**	**98.75**	98.10	**100.00**	**98.85**

identified and matched. As a result, CAMP provides a more accurate solution for applications requiring precise localization across various aerial and satellite views.

2) Results on SUES-200: We also test CAMP on SUES-200. As shown in Table II, in the drone-view target localization task (Drone→Satellite), the proposed CAMP has achieved 95.40%, 97.63%, 98.05%, 99.33% Recall@1 and 96.38%, 98.16%, 98.45%, 99.46% AP at four heights. In the drone navigation task (Satellite→Drone), CAMP achieves 96.25%, 97.50%, 98.75%, 100.00% Recall@1 and 93.69%, 96.76%, 98.10%, 98.85% AP at four heights. This consistent performance highlights the robustness of CAMP in adapting to altitude variations, which is crucial for real-world applications. The applications of CVGL often involve diverse drone platform capabilities, where maintaining stable performance despite changes in altitude is essential for reliable and effective operation.

D. Cross-Dataset Generalization Results

The generalization of geo-localization methods serves as a crucial measure in practice. The generalization of a model refers to its performance when trained on data from specific geographical region or type of scenes and subsequently tested on data from other regions or different types of scenes. To evaluate the generalization of our proposed method for geo-localization, we conducted experiments using the University-1652 dataset for training and the SUES-200 dataset for testing. In our experiments, we compared our proposed method, CAMP, with the state-of-the-art methods as MCCG and Sample4Geo for satellite-drone CVGL. All methods were trained under identical conditions, utilizing ConvNeXt-B as the backbone, an image input size of 384, and a training batch size of 48. The results presented in Table III demonstrate that although our CAMP may not achieve the same level of geo-localization performance as methods trained specifically on the target scene, it still exhibits a notably high success generalization in satellite-drone CVGL. Furthermore, compared to the current state-of-the-art methods, CAMP outperforms by a large margin in geo-localization performance. Our method shows an average improvement of 4.56% in the R@1 and 4.38% in the AP. Particularly when there is a significant viewpoint difference between query and reference images (height at 150 m), CAMP, compared to Sample4Geo, achieves an 8.85% increase in R@1 and 7.45% in AP for the drone-view target localization task (Drone→Satellite) and a 3.75% increase in R@1 and 5.15% in AP for the drone navigation task (Satellite→Drone) at the 150-m height. These results demonstrate that our approach exhibits excellent generalization and robustness across different datasets.

E. Comparison of Feature Heatmaps

Understanding how different networks extract and emphasize features is crucial for evaluating their performance and interpretability. Feature heatmaps provide a visual representation of the areas in an input that a network considers important for its predictions. Therefore, we compare the feature heatmaps generated by our network with other method. For this comparison, we chose to contrast our method, CAMP, with the MCCG [25] network. Both MCCG and CAMP use ConvNeXt as the backbone network and apply specialized processing to the features outputted by the penultimate layer of ConvNeXt. Therefore, we extracted the features from the penultimate layer of ConvNeXt in both networks to generate the heatmaps. The visualized results are shown in Fig. 4.

TABLE III
GENERALIZATION FROM UNIVERSITY-1652 TO SUES-200

Drone→Satellite								
Method	150m		200m		250m		300m	
	R@1	AP	R@1	AP	R@1	AP	R@1	AP
MCCG [25]	57.62	62.80	66.83	71.60	74.25	78.35	82.55	85.27
Sample4Geo [12]	70.05	74.93	80.68	83.90	87.35	89.72	90.03	91.91
CAMP (Ours)	**78.90**	**82.38**	**86.83**	**89.28**	**91.95**	**93.63**	**95.68**	**96.65**
Satellite→Drone								
Method	150m		200m		250m		300m	
	R@1	AP	R@1	AP	R@1	AP	R@1	AP
MCCG	61.25	53.51	82.50	67.06	81.25	74.99	87.50	80.20
Sample4Geo	83.75	73.83	91.25	83.42	93.75	89.07	93.75	90.66
CAMP (Ours)	**87.50**	**78.98**	**95.00**	**87.05**	**95.00**	**91.05**	**96.25**	**93.44**

TABLE IV
COMPARISON OF MODEL PARAMETERS AND INFERENCE TIME ON UNIVERSITY-1652

Method	Backbone	Model Parameters	Inference Time
FSRA [11]	Vit-S	52,014,556	0.277s
MCCG [25]	ConvNeXt-B	91,247,975	1.078s
Sample4Geo [12]	ConvNeXt-B	87,566,465	0.321s
CAMP (Ours)	ConvNeXt-B	91,395,441	0.325s

The heatmap results reveal that the MCCG's features tend to focus on the center of the scene. In contrast, CAMP's features naturally highlight prominent areas within different scenes. This difference is particularly noticeable in scenes with irregular or multiple buildings. Our network is able to precisely locate and gather areas of interest, avoiding a mechanical focus on the image center. This suggests that our method is more adapt at identifying and concentrating on significant regions within various contexts.

F. Comparison of Model Parameters and Inference Time

In this section, we compare our proposed CAMP with several other state-of-the-art networks in terms of the number of parameters and inference time required to inference one-step images in "test" part of University-1652 dataset. These metrics are crucial for evaluating the efficiency and feasibility of deploying models in real-world applications. A model with fewer parameters and faster inference time is generally more desirable for practical use, particularly in resource-constrained environments.

We have selected the MCCG [25] and Sample4Geo [12] models for comparison, as they use the same backbone network as our proposed CAMP. Additionally, we included the FSRA [11] model, which utilizes a transformer-based backbone. As shown in Table IV, the FSRA model has fewer parameters and shorter inference time due to its smaller backbone network. While both MCCG and our CAMP model incorporate additional network structures on top of the backbone, leading to a higher number of parameters compared to Sample4Geo, our CAMP model has a distinct advantage. The additional network structure in CAMP is only employed during training. For inference, CAMP directly utilizes the highly summarized feature vectors produced by the backbone network. This design choice ensures that despite having more parameters overall, the inference time for CAMP is similar to that of Sample4Geo, demonstrating the efficiency of our approach.

G. Ablation Studies

In the ablation studies, we first investigated the effects of the PPB and CAM in our proposed CAMP. Considering that the motivation behind designing the CAM strategy was to enhance the model's discernment without increasing the batch size, we purposefully designed experiments to investigate the adaptability of the CAM strategy when reducing the batch size.

1) Effect of the PPB: As depicted in Table V, ablation experiments were conducted on University-1652. Compared to the model without PPB, our proposed method achieves a performance improvement of 1.20% in R@1 and 1.17% in AP for the drone-view target localization task (Drone→Satellite). For the drone navigation task (Satellite→Drone), there is a performance gain of 0.57% in R@1 and 0.78% in AP. The experimental results demonstrate that enhancing the model's perception of scenes through the PPB effectively improves the results of cross-view geolocation.

2) Effect of the CAM Strategy: The CAM strategy is designed to increase the number of negative samples in contrastive learning, which strengthens the contrast between different scenes. As shown in Table V, "w/ CAM" means that our CAM strategy was adopted, which increases the

TABLE V
ABLATION STUDY TO VERIFY THE EFFECT OF PPB AND CAM

Method	Drone→Satellite		Satellite→Drone	
	R@1	AP	R@1	AP
w/o CAM, PPB	91.69	93.10	95.29	91.16
w/ CAM, w/o PPB	92.56 ↑ 0.87	93.71 ↑ 0.61	95.86 ↑ 0.57	91.75 ↑ 0.59
w/o CAM, w/ PPB	93.16 ↑ 1.47	94.27 ↑ 1.17	95.86 ↑ 0.57	91.94 ↑ 0.78
Ours (w/CAM, PPB)	**94.46** ↑ 2.77	**95.38** ↑ 2.28	**96.15** ↑ 0.86	**92.72** ↑ 1.56

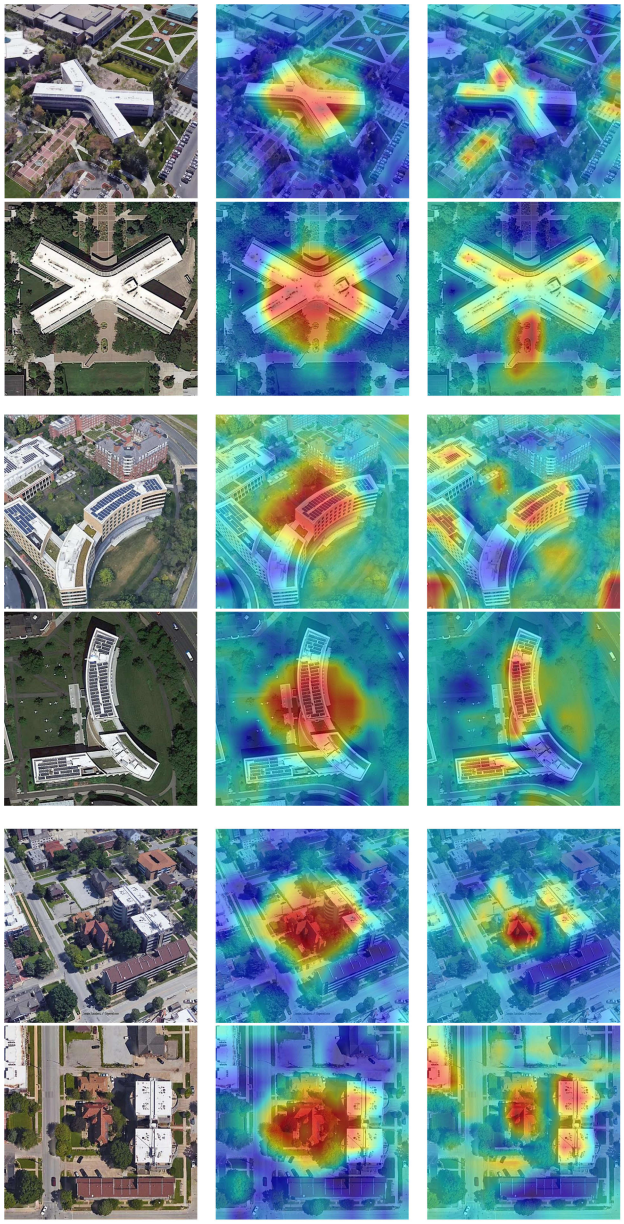


Fig. 4. Comparison of heatmaps between MCCG [25] and CAMP. The first column is the original image pair. The second and third columns are the heat maps generated by the features extracted by the MCCG and CAMP network, respectively.

number of negative samples from the same platform. "w/o CAM" refers to the use of the traditional positive and negative sample construction strategy in contrastive learning. Incorporating CAM strategy in the geo-localization model, compared to not using it, results in a 0.87% increase in R@1 and a 0.61% increase in AP for the drone-view target localization task (Drone→Satellite). In the drone navigation task (Satellite→Drone), there is a performance gain of 0.57% in R@1 and 0.59% in AP. Although the sole integration of the CAM strategy may not lead to a substantial improvement in results, augmenting the CAM strategy on top of the PPB achieves a further enhancement in the drone-view target localization task performance, with an increase of 1.30% in R@1 and 1.11% in AP metrics.

Evidently, the combined usage of both PPB and CAM strategy for localization demonstrates a significantly superior performance compared to their individual application. The experimental results demonstrate that the CAM strategy can significantly enhance retrieval performance regardless of the utilization of PPB.

3) Adaptability of CAM When Reducing Batch Size: As shown in Fig. 5, we designed ablation studies to explore the adaptability of the CAM strategy when reducing batch size. We conducted two sets of experiments: one set employed the CAM strategy while the other did not. We compared the results of the two models across batch sizes ranging from 8 to 24. Considering that the metrics for the drone navigation task (Satellite→Drone) were relatively high and approaching saturation, we chose the Recall@1 metric for the drone-view target localization task (Drone→Satellite) as the evaluation protocol. Under each batch size condition, the model with the CAM strategy consistently outperformed the model without the CAM strategy. Specifically, when the batch size decreased from 24 to 6, the Recall@1 of the model with the CAM strategy decreased from 94.46% to 87.36%, a decrease of 7.1%. In contrast, the Recall@1 of the model without the CAM strategy decreased from 93.31% to 84.96%, a decrease of 8.35%. These results demonstrate that our CAM strategy not only can improve the identification ability of the model but also alleviate the loss in geo-localization performance caused by insufficiently large batch sizes.

4) Effect of K in PPB: The number of partitions K is a crucial parameter in the PPB. In our experiments, we set $K = 3$ as the default. To analyze the impact of different K values, we conducted an ablation study examining how varying Kinfluences the accuracy metrics R@1 and AP. As shown in Fig. 6, we explored the retrieval results for the drone-view target localization and drone navigation tasks as K increased from 1 to 5. Specifically, when $K = 0$, the PPB is inactive.

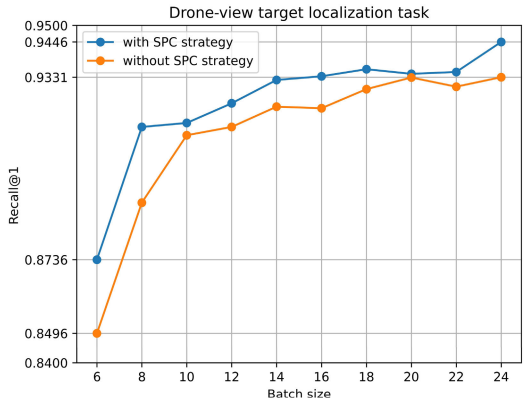


Fig. 5. Ablation study to explore the adaptability of CAM when reducing batch size.

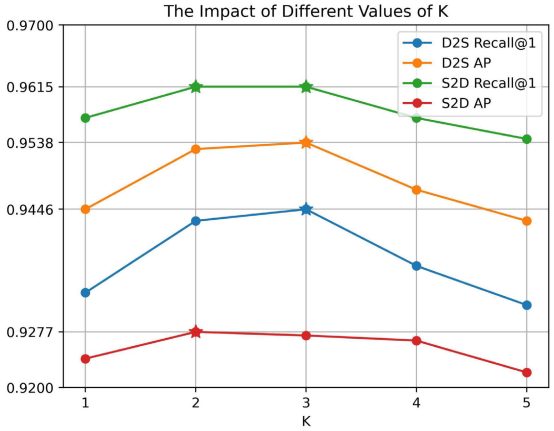


Fig. 6. Ablation study to explore the effect of the number of partitions in PPB.

When $K = 1$, the PPB does not partition the feature map, directly treating it as a whole. Our results indicate that the best performance was achieved with $K = 3$, followed by $K = 2$. This suggests that three partitions provide the optimal balance between granularity and representational ability in our model.

5) Choice of Loss Function for PPB: Our CAMP employs a combination of symmetric InfoNCE loss, MSE loss, and CAM loss. The symmetric infoNCE loss supports the overall contrastive learning framework and is commonly used in contrastive learning research. The CAM loss, introduced through our CAM strategy, increases the number of negative samples without additional memory cost. The MSE loss is utilized for PPB. We chose the MSE loss because our PPB is designed to partition and align features, allowing MSE loss to achieve fine-grained alignment. However, MSE loss is not irreplaceable compared to the symmetric InfoNCE loss and CAM loss. Therefore, we added a simple ablation study to detect the influence of MSE loss by replacing it with other loss functions. The results, presented in Table VI, indicate that the best performance was achieved with MSE loss regardless of the utilization of CAM. Upon analysis, we found that after processing through the PPB, features are less suitable for contrastive loss functions like InfoNCE loss or triplet loss, which

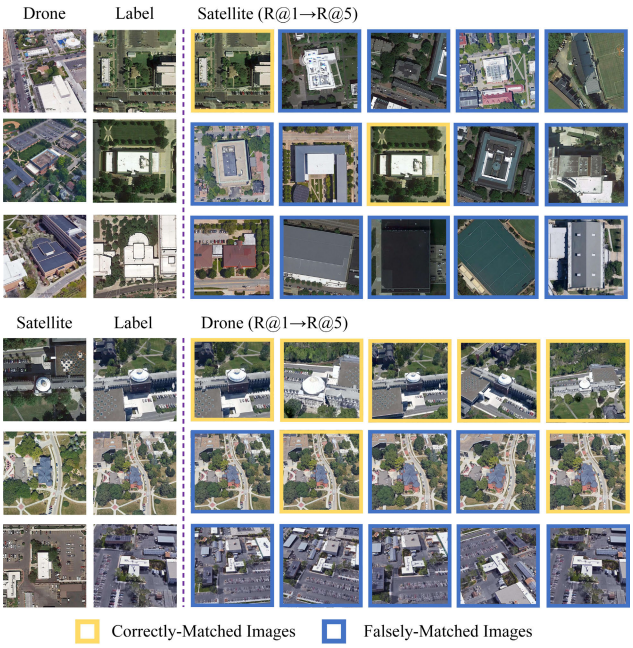


Fig. 7. Qualitative image retrieval results on University-1652. (Top) Top-5 retrieval results of drone-view target localization. (Bottom) Top-5 retrieval results of drone navigation.

are typically used to compare differences between features from different regions. Instead, the features are better suited for directly learning the similarities between features from the same region but across different platforms. The MSE loss effectively captures these fine-grained alignments within the same region, leading to superior performance. This suggests that MSE loss achieves the best fine-grained alignment for the features partitioned by PPB, as it directly addresses the small domain gap and enhances feature consistency across platforms.

H. Visualization of Qualitative Results

As an additional qualitative assessment, we provide visualizations of retrieval outcomes for various tasks using the University-1652 dataset in Fig. 7. For the drone-view target localization task and the drone navigation task, we both selected three scenes and showcased the top five retrieval outcomes generated by the model. Correctly matched results between query and reference images are highlighted in yellow boxes, while incorrect results are indicated with blue boxes.

We specifically analyzed cases that do not achieve Top-1 matching and categorized the results. We defined "Mediocre Results" as cases that do not achieve Top-1 matching but do achieve Top-5 matching (as shown in the second and fifth rows of Fig. 7). "Bad Results" are defined as cases that do not achieve Top-10 matching (as shown in the third and sixth rows of Fig. 7). We analyzed the main reasons for these poor results.

1) *High similarity between geographic scenes:* Many geographic scenes in the University-1652 dataset share a high degree of similarity, which poses a significant challenge for accurate image retrieval. For example, urban environments often feature similar patterns such

TABLE VI
REPLACE MSE LOSS WITH OTHER LOSS FUNCTIONS IN PPB ON "DRONE TO SATELLITE" TASK

Loss Function in PPB	w/ CAM		w/o CAM	
	R@1	AP	R@1	AP
InfoNCE Loss	90.30	91.90	92.27	93.64
Triplet Loss	92.70	93.93	91.36	92.79
MSE Loss	**94.46**	**95.38**	**93.16**	**94.27**

as grids of streets, uniform building styles, and similar vegetation layouts. This high similarity can cause the model to confuse one area with another, leading to incorrect matches.

2) *Changes in geological objects:* Aerial and satellite images are often not collected simultaneously, resulting in significant changes in the main objects within the scene, such as building demolition, reconstruction, or repainting. These temporal discrepancies make it difficult for the model to correctly identify the same geographical area in both aerial and satellite imagery, resulting in erroneous retrievals.
3) *Dense region partitioning:* In the University-1652 dataset, some of the data are collected in a very dense manner. Therefore, aerial images with smaller elevation angles often capture objects of adjacent scenes, leading to confusion and poor retrieval results. This is why, in some cases, the "label" and "retrieval results" appear to be the same scene but are evaluated as different due to misidentification of the correct scene.

V. CONCLUSION

In this article, we analyze the CVGL task and propose a novel method CAMP with the PPB and the CAM strategy. The PPB is proposed to learn fine-grained features of different parts and captures their spatial information, providing a comprehensive understanding of scenes from both textual and spatial perspectives. To reinforce the constraints between distinct scenes, we introduce the CAM strategy, which effectively leverages the constraints between same-platform images without extra memory cost. Our method achieves competitive accuracy on the University-1652 and SUES-200 datasets and demonstrates robust generalization capability. The results of ablation studies confirm the viewpoint advocated throughout this article: enhancing the model's identification ability for scenes is crucial and hinges on two aspects: improving its perceptual understanding of each scene and enhancing its ability to differentiate between distinct scenes. Moving forward, we aim to explore a module for enhancing the model's discriminative abilities toward scenes by integrating images from diverse perspectives.

REFERENCES

[1] Y. Shi, L. Liu, X. Yu, and H. Li, "Spatial-aware feature aggregation for image based cross-view geo-localization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–11.

[2] Y. Long, Y. Gong, Z. Xiao, and Q. Liu, "Accurate object localization in remote sensing images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2486–2498, May 2017.

[3] X. Lu, S. Luo, and Y. Zhu, "It's okay to be wrong: Cross-view geo-localization with step-adaptive iterative refinement," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4709313, doi: 10.1109/TGRS.2022.3210195.

[4] Z. Zheng, Y. Wei, and Y. Yang, "University-1652: A multi-view multi-source benchmark for drone-based geo-localization," in *Proc. 28th ACM Int. Conf. Multimedia*. Seattle, WA, USA: ACM, Oct. 2020, pp. 1395–1403, doi: 10.1145/3394171.3413896.

[5] L. Liu and H. Li, "Lending orientation to neural networks for cross-view geo-localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 5617–5626, doi: 10.1109/CVPR.2019.00577.

[6] M. Zhai, Z. Bessinger, S. Workman, and N. Jacobs, "Predicting ground-level scene layout from aerial imagery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 4132–4140, doi: 10.1109/CVPR.2017.440.

[7] Y. Zhu, B. Sun, X. Lu, and S. Jia, "Geographic semantic network for cross-view image geo-localization," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2021.

[8] B. Sun, G. Liu, and Y. Yuan, "F3-Net: Multiview scene matching for drone-based geo-localization," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5610611.

[9] S. Workman and N. Jacobs, "On the location dependence of convolutional neural network features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2015, pp. 70–78.

[10] T. Wang et al., "Each part matters: Local patterns facilitate cross-view geo-localization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 2, pp. 867–879, Feb. 2022, doi: 10.1109/TCSVT.2021.3061265.

[11] M. Dai, J. Hu, J. Zhuang, and E. Zheng, "A transformer-based feature segmentation and region alignment method for UAV-view geo-localization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 7, pp. 4376–4389, Jul. 2022, doi: 10.1109/TCSVT.2021.3135013.

[12] F. Deuser, K. Habel, and N. Oswald, "Sample4Geo: Hard negative sampling for cross-view geo-localisation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 16847–16856.

[13] F. Castaldo, A. Zamir, R. Angst, F. Palmieri, and S. Savarese, "Semantic cross-view matching," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW)*, Dec. 2015, pp. 9–17.

[14] M. Bansal, H. S. Sawhney, H. Cheng, and K. Daniilidis, "Geo-localization of street views with aerial image databases," in *Proc. 19th ACM Int. Conf. Multimedia*, Nov. 2011, pp. 1125–1128.

[15] T.-Y. Lin, S. Belongie, and J. Hays, "Cross-view image geolocalization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 891–898.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25. Red Hook, NY, USA: Curran Associates, 2012, pp. 1–9.

[17] S. Workman, R. Souvenir, and N. Jacobs, "Wide-area image geolocalization with aerial reference imagery," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3961–3969.

[18] T.-Y. Lin, Y. Cui, S. Belongie, and J. Hays, "Learning deep representations for ground-to-aerial geolocalization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 5007–5015, doi: 10.1109/CVPR.2015.7299135.

[19] N. N. Vo and J. Hays, "Localizing and orienting street views using overhead imagery," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*. Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 494–509.

[20] S. Hu, M. Feng, R. M. H. Nguyen, and G. H. Lee, "CVM-Net: Cross-view matching network for image-based ground-to-aerial geo-localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7258–7267.

[21] S. Cai, Y. Guo, S. Khan, J. Hu, and G. Wen, "Ground-to-aerial image geo-localization with a hard exemplar reweighting triplet loss," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8390–8399.

[22] T. Wang, Z. Zheng, Z. Zhu, Y. Gao, Y. Yang, and C. Yan, "Learning cross-view geo-localization embeddings via dynamic weighted decorrelation regularization," 2022, *arXiv:2211.05296*.

[23] Y. Shi, X. Yu, L. Liu, T. Zhang, and H. Li, "Optimal feature transport for cross-view image geo-localization," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 11990–11997.

[24] J. Lin et al., "Joint representation learning and keypoint detection for cross-view geo-localization," *IEEE Trans. Image Process.*, vol. 31, pp. 3780–3792, 2022, doi: 10.1109/TIP.2022.3175601.

[25] T. Shen, Y. Wei, L. Kang, S. Wan, and Y.-H. Yang, "MCCG: A ConvNeXt-based multiple-classifier method for cross-view geo-localization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 3, pp. 1456–1468, Mar. 2024, doi: 10.1109/tcsvt.2023.3296074.

[26] H. Zhao, K. Ren, T. Yue, C. Zhang, and S. Yuan, "TransFG: A cross-view geo-localization of satellite and UAVs imagery pipeline using transformer-based feature aggregation and gradient guidance," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 4700912, doi: 10.1109/TGRS.2024.3352418.

[27] H. Li et al., "Global and local contrastive self-supervised learning for semantic segmentation of HR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5618014.

[28] M. Huang, L. Dong, W. Dong, and G. Shi, "Supervised contrastive learning based on fusion of global and local features for remote sensing image retrieval," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5208513.

[29] S. Hou, H. Shi, X. Cao, X. Zhang, and L. Jiao, "Hyperspectral imagery classification based on contrastive learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5521213.

[30] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3733–3742.

[31] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*. Glasgow, U.K.: Springer, Aug. 2020, pp. 776–794.

[32] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[33] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9726–9735.

[34] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[35] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," 2020, *arXiv:2003.04297*.

[36] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton, "Big self-supervised models are strong semi-supervised learners," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 22243–22255.

[37] J.-B. Grill et al., "Bootstrap your own latent—A new approach to self-supervised learning," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 21271–21284.

[38] X. Chen, S. Xie, and K. He, "An empirical study of training self-supervised vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9620–9629.

[39] M. Caron et al., "Emerging properties in self-supervised vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9630–9640.

[40] M. Liang, J. Dong, L. Yu, X. Yu, Z. Meng, and L. Jiao, "Self-supervised learning with learnable sparse contrastive sampling for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5530713.

[41] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11966–11976.

[42] A. Dosovitskiy et al., "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[43] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 9992–10002, doi: 10.1109/ICCV48922.2021.00986.

[44] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[45] R. Zhu, L. Yin, M. Yang, F. Wu, Y. Yang, and W. Hu, "SUES-200: A multi-height multi-scene cross-view image benchmark across drone and satellite," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 43, no. 9, pp. 4825–4839, Sep. 2023, doi: 10.1109/TCSVT.2023.3249204.

[46] J. Zhuang, M. Dai, X. Chen, and E. Zheng, "A faster and more effective cross-view matching method of UAV and satellite images for UAV geolocalization," *Remote Sens.*, vol. 13, no. 19, p. 3979, Oct. 2021, doi: 10.3390/rs13193979.

[47] R. Zhu, M. Yang, L. Yin, F. Wu, and Y. Yang, "UAV's status is worth considering: A fusion representations matching method for geo-localization," *Sensors*, vol. 23, no. 2, p. 720, Jan. 2023, doi: 10.3390/s23020720.

[48] X. Liu, Z. Wang, Y. Wu, and Q. Miao, "SeGCN: A semantic-aware graph convolutional network for UAV geo-localization," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 6055–6066, 2024, doi: 10.1109/jstars.2024.3370612.

Qiong Wu (Graduate Student Member, IEEE) was born in 2001. She received the B.S. degree from Wuhan University, Wuhan, China, in 2022, where she is currently pursuing the Ph.D. degree with the School of Remote Sensing and Information Engineering.

Her research interests include deep learning for image retrieval and supervised contrastive learning.

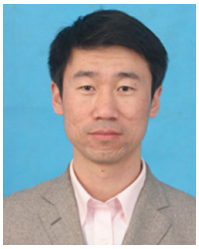
Yi Wan (Member, IEEE) was born in 1991. He received the B.S. and Ph.D. degrees from Wuhan University, Wuhan, China, in 2013 and 2018, respectively.

He is currently an Associate Professor with Wuhan University. His research interests include digital photogrammetry, computer vision, 3-D reconstruction, and change detection in remote sensing imagery.

Zhi Zheng (Member, IEEE) received the B.S. degree in remote sensing and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2017 and 2023, respectively.

He is currently a Post-Doctoral Fellow with the Department of Geography and Resource Management, The Chinese University of Hong Kong, Hong Kong, China. He has authored or co-authored more than ten research articles. His research interests include satellite remote sensing, stereo matching, change detection, and geohazard monitoring using deep learning technology.

Dr. Zheng was awarded the Research Fellowship Scheme by The Chinese University of Hong Kong, in January 2024. In recent years, he has frequently served as a referee for several international journals.



Yongjun Zhang (Member, IEEE) received the B.S. degree in geodesy, the M.S. degree in geodesy and surveying engineering, and the Ph.D. degree in geodesy and photography from Wuhan University, Wuhan, China, in 1997, 2000, and 2002, respectively.

He is currently the Dean with the School of Remote Sensing and Information Engineering, Wuhan University. He has published more than 180 research articles and three books. His research interests include aerospace and low-attitude photogrammetry, image matching, combined block adjustment with multisource datasets, object information extraction and modeling with artificial intelligence, integration of light detection and ranging (LiDAR) point clouds and images, and 3-D city model reconstruction.

Dr. Zhang is the Coeditor-in-Chief of The Photogrammetric Record.

Guangshuai Wang was born in 1994. He received the B.S. degree from Wuhan University, Wuhan, China, in 2020.

He is currently an Engineer with China Railway Design Group Company Ltd., Tianjin, China. His research interests include railway aerial photogrammetry, multisource remote sensing data fusion, and 3-D reconstruction.

Zhenyang Zhao received the master's degree in surveying and mapping engineering from Wuhan University, Wuhan, China, in 2017.

He is currently an Engineer with China Railway Design Group Company Ltd., Tianjin, China. His research interests are railway geographic information big data and geographic information system (GIS) development.