GeoFormer: An Effective Transformer-Based Siamese Network for UAV Geolocalization

Qingge Li, Xiaogang Yang, Jiwei Fan, Ruitao Lu, Bin Tang, Siyu Wang, and Shuang Su

***Abstract*—Cross-view geolocalization of unmanned aerial vehicles (UAVs) is a challenging task due to the positional discrepancies and uncertainties in scale and distance between UAVs and satellite views. Existing transformer-based geolocalization methods mainly use encoders to mine image contextual information. However, these methods have some limitations when dealing with scale changes in cross-view images. Therefore, we present an effective transformer-based Siamese network tailored for UAV geolocalization, called GeoFormer. First, an efficient transformer feature extraction network was designed, which utilizes linear attention to reduce the computational complexity and improve the computational efficiency of the network. Among them, we designed an efficient separable perceptron module based on depthwise separable convolution, which can effectively reduce the computational cost while improving the feature representation of the network. Second, we proposed a multiscale feature aggregation module, which deeply fuses salient features at different scales through a feedforward neural network to generate global feature representations with rich semantics, which improves the model's ability to capture image details and represent robust features. Additionally, we designed a semantic-guided region segmentation module, which utilizes a *k*-modes clustering algorithm to divide the feature map into multiple regions with semantic consistency and performs feature recognition within each semantic region to improve the accuracy of image matching. Finally, we designed a hierarchical reinforcement rotation matching strategy to achieve accurate UAV geolocalization based on the retrieval results of UAV view query satellite images using SuperPoint keypoints extraction and LightGlue rotation matching. According to the experimental results, our method effectively achieves UAV geolocalization.**

***Index Terms*—Cross-view image retrieval, heterologous scene matching, linear attention, Siamese network, transformer, unmanned aerial vehicle (UAV) geolocalization.**

I. Introduction

UNMANNED aerial vehicles (UAVs) have emerged as versatile and efficient tools in various application domains, such as aerial surveillance [1], target tracking [2], [3], and disaster response [4], [5]. A pivotal task within UAV systems is geolocalization, which estimates the geographic coordinates of drones in real time. Geolocalization is achieved by matching UAV and satellite imagery. Its applications can be broadly divided into two categories: UAV target localization (UAV→satellite) and UAV navigation (satellite→UAV). Accurate geolocalization is paramount for ensuring the effectiveness of UAV missions. However, since UAV and satellite views are acquired under different conditions of light, weather, and seasonal variations, there are large differences in the visual detail features of the same scene in the images. In addition, there are positional differences as well as scale and distance uncertainties between objects in UAV images and satellite views due to variations in UAV flight altitude and shooting angles. These increase the difficulty of accurate geolocalization between UAV and satellite views. To address this challenge, it is necessary to efficiently extract salient features in the images, and identify and correct the deviations through invariant features obtained by multiscale feature fusion to ensure that UAVs can achieve accurate geolocalization under various environmental conditions.

The development of deep learning has provided important data, model, and algorithmic support for remote sensing image analysis and applications [6], [7], [8], [9], [10], [11], and significant advancements have been achieved regarding cross-view geolocalization methods. Most deep-learning geolocalization methods [12], [13] utilize convolutional neural networks (CNNs) to extract image features and subsequently estimate the position of the UAV by matching and comparing the visual features between drone and satellite images. However, there are some shortcomings to CNN-based methods. The relatively weak capacity of CNNs in capturing contextual information may lead to inadequate modeling of global relationships in cross-view geolocalization tasks. Simultaneously, operations, such as pooling and convolution, in the CNN may diminish the resolution of images and destroy the recognizable fine-grained information within the images.

Over the past few years, the transformer [14] has been successfully used for various computer version (CV) tasks. The remarkable contextual modeling capability of the transformer compensates for the limitations of CNNs. At present, transformer-based cross-view geolocalization technology mainly utilizes transformer encoders as the backbone of feature extraction, improving the ability of contextual feature extraction [15], [16], [17], [18]. Some methods use ViT [19] as the backbone for extracting context-sensitive information [20], [21] to better adapt to image data. Although these methods have strong geolocalization performance, ViT divides images into fixed-size blocks and then treats the relationships of all image blocks equally on a global scale, without distinguishing whether these image blocks are from adjacent regions. Swin

Manuscript received 8 January 2024; revised 2 March 2024 and 2 April 2024; accepted 21 April 2024. Date of publication 23 April 2024; date of current version 8 May 2024. *(Corresponding author: Xiaogang Yang.)*

The authors are with the College of Missile Engineering, Rocket Force University of Engineering, Xi'an 710038, China (e-mail: lqg19950105@163.com; doctoryxg@163.com; fjw19900619@163.com; lrt19880220@163.com; springgoneautumn@163.com; wsy960328@163.com; 18843226755@163.com).

Digital Object Identifier 10.1109/JSTARS.2024.3392812

transformer [22], [23] effectively reduces both information loss and optimizes computational complexity by introducing variable windows and cross-window connection mechanisms.

Inspired by the Swin transformer, we proposed an effective transformer-based Siamese network for UAV geolocalization, named GeoFormer. We proposed an efficient transformer feature extraction network, where our designed efficient separable perceptron (ESP) module reduces network computational complexity while ensuring effective extraction of image features. In addition, we designed a multiscale feature aggregation module (MFAM), which improves the network's representation of global features by deeply fusing salient features from different scales and different receptive fields. In addition, we design a semantic-guided region segmentation module (SRSM), which clusters the feature map by k-modes algorithm to obtain multiple nonoverlapping semantic consistent regions, and then performs feature recognition within the regions separately. Finally, we design the hierarchical structure enhanced heterogeneous image rotation matching strategy. Based on the results of UAV image query satellite images, SuperPoint is used to extract keypoints and then combined with LightGlue rotation matching to achieve accurate UAV geolocalization.

To summarize, this article makes the following primary contributions.

1) We construct an effective transformer-based Siamese network for UAV geolocalization called GeoFormer. We design the ESP module to replace the multilayer perceptron (MLP) in the original feature extraction network to capture the spatial relevance and contextual information of the image with lower computational complexity. In addition, we utilize linear attention to replace the original dot-product attention to improve the context awareness and computational efficiency of the feature representation.
2) We propose an MFAM module for the deep fusion of salient features at different scales and different receptive fields to generate a global feature representation with rich semantic information. The module is simple and effective and improves the model's ability to capture image details and robust feature representation.
3) We construct an SRSM module that utilizes the k-modes clustering algorithm to segment the feature map into multiple nonoverlapping semantic consistent regions and then recognize them separately within each subregion, making full use of the semantic features of the image to improve the accuracy of feature matching.
4) We designed a hierarchical reinforcement heterogeneous image rotation matching strategy, which utilizes the SuperPoint keypoints extraction algorithm combined with LightGlue secondary rotation matching to improve the rotation matching localization accuracy between heterogeneous images. We also construct a cognition dataset. The experimental results on University-1652 and cognition datasets showed that GeoFormer effectively achieved UAV geolocalization.

The rest of this article is organized as follows. We introduce some related work in Section II. We provide a detailed introduction to the proposed GeoFormer approach in Section III. Section IV gives the experimental results. Finally, Section V concludes this article.

II. Related Work

This section provides a brief overview of relevant prior article, including cross-view geolocalization, transformer in CV, and heterologous scene matching.

A. CNNs in Geolocalization

In recent years, with the development of deep learning, CNN-based geolocalization techniques have achieved remarkable results [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36]. These approaches can be classified into two types: Feature-matching-based and image-retrieval-based methods.

Initially, some feature-matching-based cross-view geolocalization algorithms were developed. The design of the feature extraction and matching technique is the emphasis of these methods. The SeLF [37] method integrates semantic information into the L2-Net [38] feature extraction network, encoding pixel semantics into their feature mappings to obtain better key points and descriptors, thereby improving the robustness of local feature matching. This approach effectively enhanced the localization accuracy on some popular benchmarks [39], [40]. DSM-Net [41] utilizes dynamic similarity to align image directions within a limited field of view. In another article [42], the robustness was further enhanced by considering the local and global properties of aerial images based on DSM-Net. Similarly, the coarse positioning performance was significantly improved by considering the geometric correspondence of feature points based on DSM-Net [43]. However, this approach can only process one image in each view at a time and is unable to simultaneously learn features of multiple images at the same position.

Another type of research is based on the concept of image retrieval [44], [45]. First, views from the same geographical location are considered as a class and, then, based on the image features, the category of images with unknown geographical locations is retrieved within the class set. The generative adversarial networks were utilized to perform cross-view image style conversion to a similar style, followed by image retrieval [46]. LCM [47] simplifies the retrieval problem into a classification problem, achieving bidirectional matching between UAV and satellite images, and attaining satisfactory accuracy on University-1652 [48]. RK-Net [13] utilizes a unit subtraction attention module to detect representative key points, and achieved good geolocalization results on typical benchmarks [48], [49], [50] by comparing salient regions. LPN [51] adopts a square-ring feature partition approach to learn contextual information by utilizing data from the environment around target buildings. Based on the LPN module, multiscale block attention [52] effectively achieves geolocalization by capturing the relationships between regions, enabling each region to attend to different features.

For cross-view geolocalization tasks, due to the significant scale differences between images of different views, a part of the research enhances the image feature representation through

multiscale feature fusion. Li et al. [53] proposed a new multiscale attention encoder aiming at overcoming the challenge of perspective and appearance differences by transforming from street-view images to aerial-view images. PaSS-KD [54] self-enhances the extraction and representation of cross-view image features by using local and multiscale knowledge as fine-grained location-dependent supervision, which effectively handles the large differences in scene context and object scale and significantly improves image retrieval performance. For the problem of significant differences in visual detail features between cross-view images, part of the research improves feature discrimination through semantic information. Rodrigues and Tani [55] solved the problem of scene changes due to temporal differences in cross-view geolocalization through a semantics-driven data augmentation technique and a multiscale attention network. Xue et al. [56] proposed the extraction of global reliable features by embedding high-level semantics to extract global reliable features to improve the visual localization task in large-scale environments, which improves the accuracy of matching by detecting keypoints from reliable regions and reduces the number of unreliable features.

B. Transformer in Geolocalization

Transformer [14] was first applied to machine translation tasks in the field of natural language processing, which is a sequence-to-sequence autoregressive model. The ViT [19] model utilizes the classic transformer encoder structure to achieve image classification tasks, marking the beginning of the transformer's application in the field of vision and gradually playing a role in cross-view geolocalization [57]. Following the architecture of NetVLAD [58], TransVLAD [15] utilizes a sparse transformer encoder to obtain global descriptors. It was further combined with DFM [59] to obtain more dense and accurate matching results. L2LTR [16] employs a transformer encoder as a backbone, utilizing self- and cross-attention mechanisms to emulate global dependency relationships between adjacent layers, thereby enhancing the quality of the learned representations. GeoDTR [17] utilizes a transformer encoder to separate geometric information from the original features and, through a novel geometric contextual extraction module, it can learn the spatial correlations between visual features in satellite and ground images. TransGeo [18] fully leverages the advantages of transformer encoder global information modeling and explicit positional encoding, reducing computational costs and enhancing performance. TransLocator [20] can simultaneously complete tasks of geographic graphic localization and scene recognition using a Siamese network with a ViT backbone. The FSRA [21] employs ViT to extract features from input images. Subsequently, the feature maps undergo spatial segmentation and alignment, demonstrating strong performance in UAV target localization and UAV navigation tasks.

However, since ViT mainly divides the input image into fixed-size blocks and then applies the self-attention mechanism for feature extraction by considering these blocks as elements in a sequence. This approach uniformly considers the relationship between all blocks in each attention layer, but it does not distinguish whether these blocks come from neighboring regions of the image. As a result, ViT's processing is globally homogeneous in space. In contrast, Swin transformer [22] introduces a hierarchical structure and a shift window mechanism, which enhances the model's ability to capture spatial relationships between neighboring regions through window sliding and offsetting. As a result, Swin transformer is able to capture local features in an image more effectively, while ensuring the integration of global information through cross-window connections [60]. In addition, Swin transformer improves computational efficiency by performing self-attention computation independently within each window, enabling the model to process windows in parallel.

C. Heterologous Scene Matching

Finding the matching relationship between two heterogeneous images is the foundation of UAV geolocalization. The matching methods in UAV localization are mainly divided into three categories: region-based, feature-point-based, and dense matching methods. The region-based matching method first divides the image into different region blocks, and matches and locates them by comparing the similarity [61], [62], [63]. Although this method has a simple principle, it requires preconstruction of reference images with known geographic information locations. The dense matching method without detectors matches by directly comparing the features of pixels or image blocks. The COTR [64], LoFTR [65], and ASpanFormer [66] algorithms achieve dense matching of local features based on the self-attention and cross-attention mechanisms. The DFM [59] algorithm adopts a matching strategy from coarse to fine, utilizing geometric transformations and twisted secondary matching to optimize the initial matching results, thereby achieving higher matching accuracy (MA). The dense matching method can better adapt to images without obvious features or targets. However, such methods are sensitive to image noise and difficult to match under complex texture and rotation conditions. Moreover, it requires a large amount of computation and has a slow matching speed, making it unsuitable for UAV localization.

The feature-point-based matching and localization method extracts keypoints in images and compares the similarity for matching and localization. It usually includes steps, such as keypoints detection, feature description, and matching. The matching methods based on feature points are mainly divided into traditional methods and deep-learning-based methods. Traditional methods rely on manually designed local invariant features and descriptors [67], [68], and typical algorithms include SIFT [69], SURF [70], ORB [71]. The manually designed feature points and descriptors are affected by factors, such as image quality, scale changes, and perspective changes, and are not robust enough for complex scenes and large-scale data. In recent years, with the use of convolution instead of SIFT feature extraction in LIFT [72], deep-learning-based feature point matching methods have become mainstream [73], [74], [75], [76], [77]. The most successful one among them is SuperPoint [78], which uses CNNs to detect keypoints in images with good accuracy and

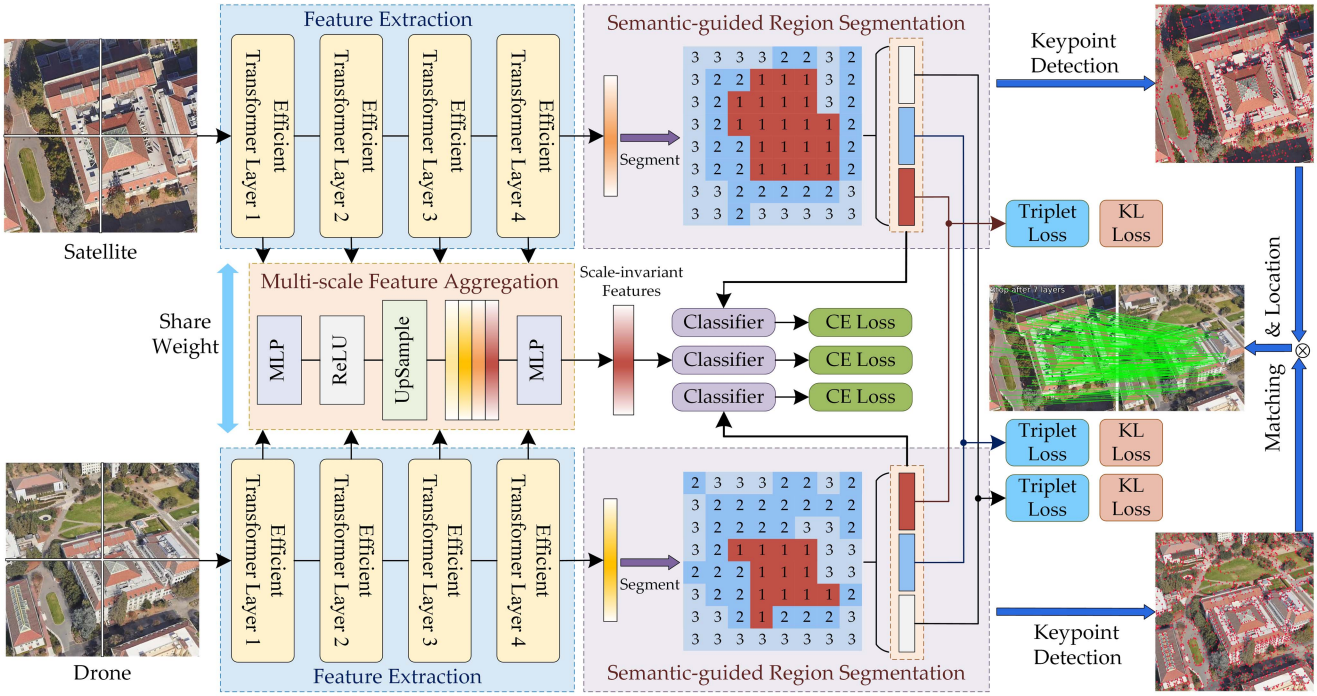


Fig. 1. Architecture of the GeoFormer framework.

robustness and generates corresponding descriptors. On this basis, SuperGlue [79] takes the detected features and descriptors as inputs, uses graph neural networks to identify the cross-attention and self-attention between features, and then uses the Sinkhorn algorithm for optimal matching. SuperGlue can achieve effective and robust matching by learning prior knowledge of scene geometry, but it also inherits the limitations of transformer training, and its computational complexity increases twice with the number of feature points. To address this issue, LightGlue [80] designed a confidence classifier to adaptively adjust the depth and width of the network, thereby reducing the computational complexity of the model and improving its matching speed. The combination of SuperPoint and LightGlue can achieve good matching results, but there is still significant room for improvement in the rotation matching task of heterogeneous scenes.

III. Method

This section provides a detailed description of the proposed transformer-based Siamese network for UAV geolocalization. Fig. 1 depicts the architecture of GeoFormer, which is divided into four sections: The efficient transformer feature extraction network, the MFAM, the SRSM, and the rotation matching positioning module. GeoFormer is composed of two branches: a UAV-view branch and a satellite-view branch. These two branches concurrently process two input data streams while sharing network weights. The input of GeoFormer is 224 × 224-pixel paired images. Each image is divided into four patches and input into a feature extraction network. The feature extraction network consists of four effective transformer layers, and through the designed ESP module, contextual information can be extracted with low computational complexity. We subsequently constructed the MFAM module to integrate multiscale details and global information from different levels of feature maps to improve the robust feature representation capability. Then, we designed the SRSM module to divide the feature map into multiple semantic consistency regions and then match them separately, improving the accuracy of feature matching. In addition, based on the retrieval results of satellite images by drones, a two-stage heterogeneous image rotation matching module was constructed, and precise UAV geolocalization was achieved using homography transformation. Finally, we established loss functions that can effectively train GeoFormer.

A. *Efficient Transformer Feature Extraction*

We designed an efficient transformer feature extraction network as backbone in order to better extract the spatial correlation and contextual information of images, as shown in Fig. 2. The feature extraction network has four layers, each with [2], [2], [6], [2] E-Swin transformer blocks. The hierarchical structure of the network can better capture information at different granularity levels, that is, lower level layers focus on local details and fine-grained features, while higher level layers capture more global and abstract representations. The visualization feature maps output by each layer of the feature extraction network are shown in Fig. 3. It can be seen that the lower levels primarily attend to the fine-textured features, while the upper layers place greater emphasis on the deeper semantic features of the image, ultimately leading to the segmentation of buildings, roadways, and vegetation within the heatmap. Due to the distinct semantic information present in the feature maps output by each level,

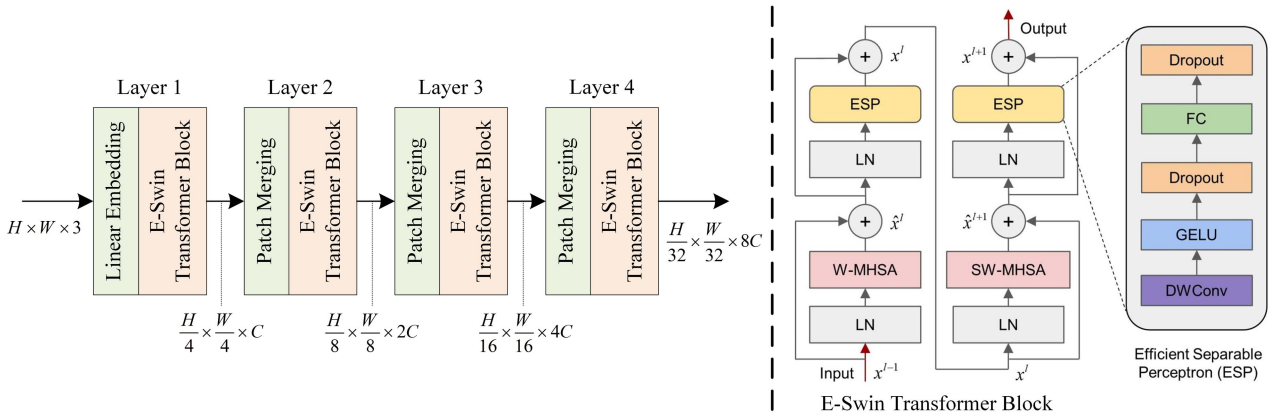


Fig. 2. Structure of the efficient transformer feature extraction network.

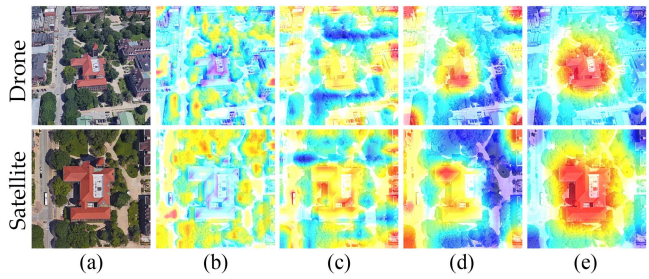


Fig. 3. Visualization of the feature maps: (a) shows input images; (b)–(e) show the feature maps from layers 1, 2, 3, and 4, respectively.

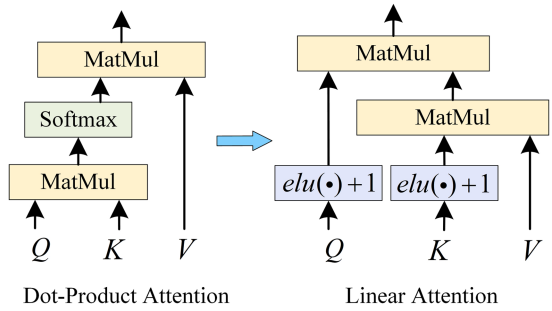


Fig. 4. Attention layer in the feature extraction network.

utilizing MFAM (detailed in Section III-B) and SRSM (detailed in Section III-C) to establish connections and integrate information between different levels can successfully enhance the feature expression ability and matching performance.

We created the E-Swin transformer block to capture the contextual data and the dependencies between elements with low computational burden, as shown in Fig. 2. The fundamental architecture of the E-Swin transformer block consists of feedforward networks and multihead self-attention (MHSA) mechanisms. The crucial components are window-based MHSA (W-MHSA) and shifted window-based MHSA (SW-MHSA). W-MHSA segments the feature map into a series of windows and then independently carries out MHSA computations within each window. This window design enables the model to process the windows in parallel, significantly enhancing the computational efficiency. Building upon W-MHSA, SW-MHSA introduces a mechanism for shifted windows, enhancing the ability to capture spatial relationships between adjacent regions. This enables the model to effectively capture the global context and dependencies between distant regions. Specifically, W-MHSA and SW-MHSA not only possess the inductive bias characteristic of CNNs, but can also capture long-range dependencies and spatial relationships. Hence, they exhibit significant advantages when processing image data. In addition, we replaced the original dot-product attention with linear attention, as shown in Fig. 4.

In dot-product attention, using dot-product operations to calculate attention weights can lead to weight decay or explosion issues, especially when dealing with long-distance dependencies. Linear attention reduces this problem by using linear transformations to calculate attention weights, making it more suitable for handling tasks with long-distance dependencies. Moreover, compared to dot-product operations, linear transformations have lower computational complexity and are more efficient in computation. The following details the computation of the E-Swin transformer blocks:

$$\hat{x}^{l} = \mathrm{W-MHSA}(\mathrm{Norm}(x^{l-1})) + x^{l-1} \tag{1}$$

$$x^{l} = \mathrm{ESP}(\mathrm{Norm}(\hat{x}^{l})) + \hat{x}^{l} \tag{2}$$

$$\hat{x}^{l+1} = \mathrm{SW-MHSA}(\mathrm{Norm}(x^{l})) + x^{l} \tag{3}$$

$$x^{l+1} = \mathrm{ESP}(\mathrm{Norm}(\hat{x}^{l+1})) + \hat{x}^{l+1} \tag{4}$$

where $\hat{x}^{l}$ and x^{l} denote the output features of layer l and $\mathrm{Norm}(*)$ denotes layer normalization. $\mathrm{W-MHSA}(*)$ and $\mathrm{SW-MHSA}(*)$ represent the W-MHSA and SW-MHSA, respectively.

We designed the ESP as an important component of the E-Swin transformer block to extract contextual information with lower computational complexity. The ESP structure is shown in Fig. 2. Due to the high computational cost of W-MSA and SW-MSA, as well as the need for self-attention calculation of the entire input feature map in the E-Swin transformer block, it may occupy a large amount of memory when processing large-sized images. In order to mitigate the memory consumption and computational complexity of the E-Swin transformer block, we

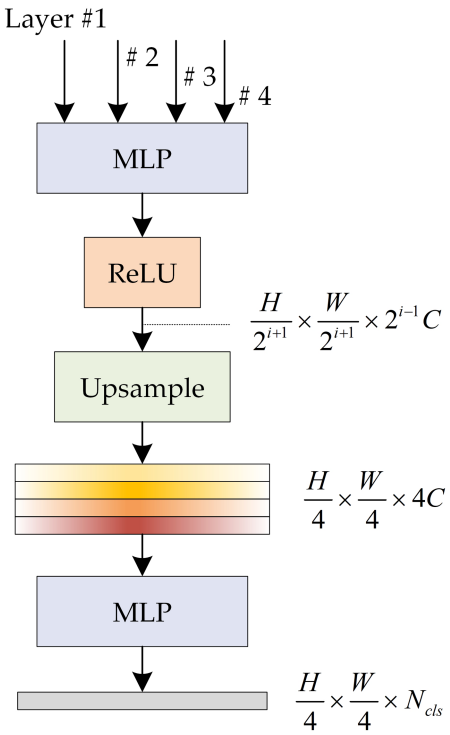


Fig. 5. Structure of the MFAM.

devised ESP. It replaces the standard convolution operation with depthwise separable convolution, including depthwise convolution and pointwise convolutions. This separation significantly decreases the computational complexity while still capturing essential feature interactions. By introducing depthwise separable convolutions, the E-Swin transformer block achieves higher efficiency without sacrificing performance. The ESP layer effectively captures spatial and interchannel dependencies in a more efficient manner, enabling the network to process large-scale data with fewer computational resources. The ESP is formulated as follows:

$$x_{\text{out}} = \text{FC}(\text{GELU}(\text{DWConv}(x_{\text{in}}))) \tag{5}$$

where x_{in} is the feature from the layer normalization, x_{out} is the output feature of ESP, FC means fully connected layer, DWConv means depthwise separable convolution, and GELU is the Gaussian error linear unit activation function.

B. Multiscale Feature Aggregation

We proposed the MFAM, which aims to fully utilize the salient features and semantic information of the feature maps extracted by backbone at different scales and different receptive fields to enhance the global feature representation of the network. MFAM is a simple and effective feedforward neural network, which is a lightweight decoder mainly composed of MLP, as shown in Fig. 5. Initially, MFAM extracts the output feature maps of each layer of the Geoformer backbone, and unifies the channel dimensions of multilevel features F_i through the MLP layer to ensure that the features have the same dimensions before fusion, so as to improve the efficiency and effectiveness of feature fusion. Different scale features represent different receptive fields and semantic information. Subsequently, after the ReLU nonlinear activation and upsampling operations, the features become a quarter of the original and concatenated together. This process increases the model's full utilization of salient features at different scales and in different receptive fields, thus enriching the semantic information captured. Then, utilizing another MLP layer for feature deep fusion, the global feature F is obtained. This design allows the model to better integrate information from different levels to generate global feature representations with rich semantics, which is critical for improving model performance in cross-view geolocalization tasks. Finally, the model is able to utilize more comprehensive and integrated information for the final prediction of global features F, which improves the accuracy of the prediction. The computational process of MFAM is as follows:

$$\widehat{F}_i = \text{ReLU}[\text{Linear}(C_i, C)(F_i)] \; \forall i \tag{6}$$

$$\widehat{F}_i = \text{Upsample}\left(\frac{W}{4} \times \frac{H}{4}\right)(\widehat{F}_i) \; \forall i \tag{7}$$

$$F = \text{Linear}(4C, C)\left(\text{Concat}(\widehat{F}_i)\right) \; \forall i \tag{8}$$

$$M = \text{Linear}(C, N_{\text{cls}})(F) \tag{9}$$

where $\text{Linear}(C_{\text{in}}, C_{\text{out}})(*)$ represents a linear transformation layer, C_{in} and C_{out} represent the input and output vector dimensions, respectively, $\text{ReLU}(*)$ represents the ReLU nonlinear activation function, N_{cls} indicates the total amount of categories, and M denotes the final predicted feature vector.

MFAM enhances the feature representation capability by fusing the details and global features of feature maps extracted by the backbone at different scales. By fusing multilevel features into a single vector before classification, as opposed to separately classifying the outputs at each level, the computational complexity of the model is reduced. Moreover, MFAM exhibits a certain degree of flexibility, allowing for the independent adjustment of feature extraction levels and scales according to the requirements of a given task. MFAM can fully harness semantic information from different scales and levels, thus enhancing its perceptual and expressive capabilities for targets of various scales. Compared with Swin transformer, although Swin transformer is able to generate feature maps at different scales through its hierarchical design, which represents different image details and high-level semantic information, the interaction and fusion between features at different scales are not sufficient. Our MFAM ensures that salient features with different semantics can be combined more effectively by fusing these multiscale features more explicitly, thus improving the model's ability to capture image details and represent robust features.

C. Semantic-Guided Region Segmentation

Inspired by the heatmap segmentation module (HSM) in FSRA [21], we designed SRSM to segment feature maps into multiple regions based on semantic information, and then perform feature matching within each semantic region to improve

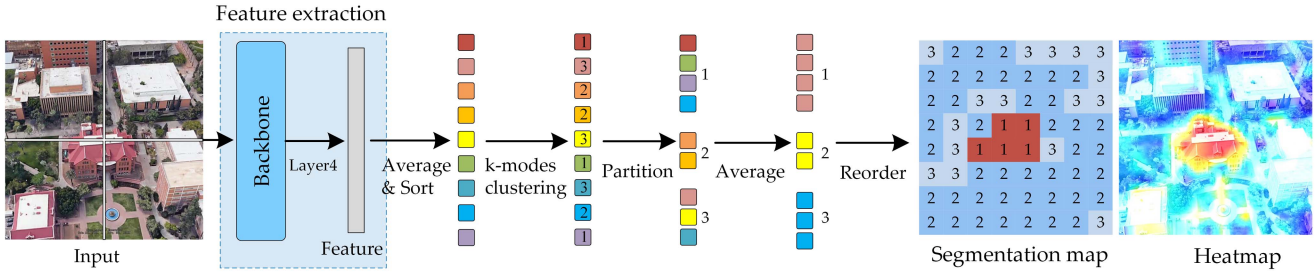


Fig. 6. Structure of SRSM.

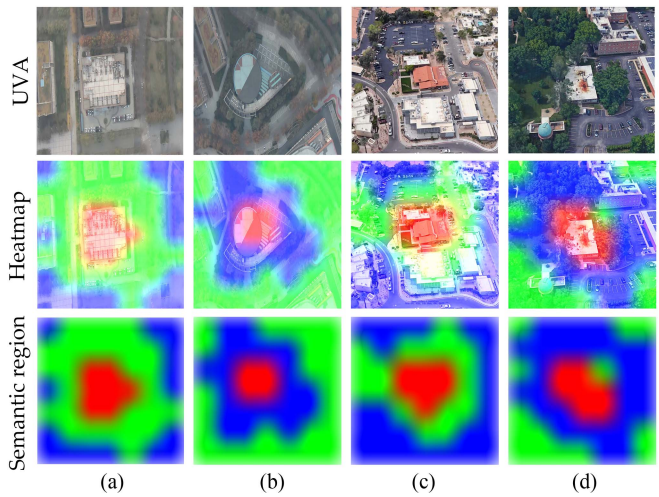


Fig. 7. Semantic region segmentation results of SRSM.

the accuracy of feature matching. The main steps of SRSM are mainly divided into feature sorting, region segmentation, region feature averaging, and feature reordering, as shown in Fig. 6. The input of SRSM is the output of the last layer of the feature extraction network. First, the average value of each slice in the feature vector is computed and then arranged in decreasing order to obtain an ordered sequence of features. We believe that slices with similar feature values often reflect similar semantic information. Therefore, we further cluster the ordered feature sequence using the *k*-modes algorithm, and the ordered feature sequence can speed up the clustering. In this way, we classify the feature sequences into several classes, and the number of slices in each class may be the same or different. Then, based on the classification results, the feature sequence is partitioned into several different region feature sequences, and the slices within each region feature sequence represent a set of visually and semantically similar and related feature information. Therefore, we consider that each region feature sequence corresponds to a class of semantic information. Then, the average value of each region feature sequence is computed and used to replace the feature value of the corresponding region feature sequence. Finally, each feature value of three semantic regions is mapped back to the original feature map in the initial order to obtain the segmentation result, as shown in Fig. 7.

SRSM improves the perception of local targets or details and enhances matching performance by partitioning feature maps into multiple semantic regions. SRSM performs local feature matching within the semantic regions corresponding to two images, which can obtain more accurate matching results. Compare with the HSM in FSRA [21], which arranges the feature sequences in decreasing order and then divide them directly into multiple subsequences of the same length, and then calculate the average of each subsequence as a representative. This method directly divides the sequences into equal lengths based on the order of the sequences, which is a uniform division based on the position, and the lengths of the obtained subsequences are fixed. Our proposed SRSM performs clustering using *k*-modes algorithm after arranging the feature sequences in decreasing order and divides the sequences into several classes based on the similarity of feature values. This method classifies the features adaptively based on the similarity between feature values instead of enforcing equal division, and the number of slices of each region feature sequence is unknown. The SRSM clusters the features by the *k*-modes algorithm, which makes each cluster better reflect the similar feature information, thus improving the semantic sensitivity and accuracy of feature matching. Compared with the uniform segmentation of HSM, SRSM allows different region feature sequences to have different lengths, and this flexibility allows the model to fit the actual semantic distribution better. Ordering the feature sequences first in SRSM can speed up the clustering of the *k*-modes algorithm because ordered feature sequences reduce the number of iterations of the algorithm in the initial stage, making the clustering process more efficient.

SRSM independently classifies the feature vectors of each region through a classifier, obtaining classification results for each semantic region. The classifier first performs a linear transformation on the input data. Next, normalization is introduced to accelerate the convergence speed of model training while improving the robustness and generalization ability of the model. Finally, Dropout is utilized to randomly deactivate some neurons during the training process to solve the overfitting problem. In addition, SRSM can flexibly adjust the number of region segmentation according to the image type and task requirements, endowing SRSM with flexibility and adaptability to different scenarios and objectives.

D. Rotation Matching Positioning

After obtaining satellite images retrieved according to the UAV query image, keypoint matching is utilized to obtain

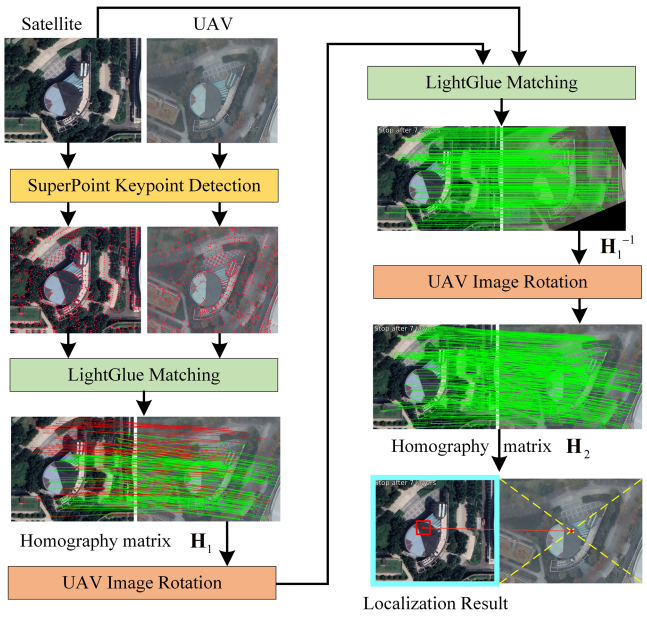


Fig. 8. Structure of keypoint matching and localization.

accurate UAV geolocalization results. The UAV images are viewed directly downwards, the center point of the UAV image represents the position of the UAV, and the position of the UAV center point in the satellite view needs to be found based on the keypoint matching relationship. A hierarchical reinforcement matching and localization method were designed to address the issue of low MA caused by the prevalent angular differences between UAVs and satellite images. The specific process is shown in Fig. 8.

First, SuperPoint is utilized to detect the keypoints in the satellite and UAV images. Then LightGlue is utilized for primary matching to obtain the homography transformation matrix $\mathbf{H_1}$. After that, the UAV view image is rotated to eliminate the angular difference with the satellite view according to $\mathbf{H_1}$. Then, LightGlue secondary matching is performed and the keypoint matching result is rotated to the original image by $\mathbf{H}_1^{-1}$, and the corresponding homography transformation matrix $\mathbf{H}_2$ is calculated. Finally, the position of the UAV is obtained in the satellite view based on the keypoint transform matrix $\mathbf{H}_2$.

E. Image Sequence Consistency Voting Strategy

When the top retrieved target is a false image, the keypoint matching localization fails to achieve UAV geolocalization directly. To address this challenge, we adopt an image sequence consistency voting strategy to optimize the algorithm and improve its robustness. We use a sequence of UAV aerial images rather than a single image to predict the location of the UAV. To reduce the computational burden, we select at least three UAV aerial image sequences for retrieval and keypoint matching. When the keypoint matching result reaches the threshold criterion, it is considered as a successful match. When the number of successful matches is not less than 2, we predict the location of the UAV based on the consistent matching results. Otherwise, it is recognized that the UAV localization fails. At this time, another set of UAV aerial image sequences is taken to relocate the UAV based on the image sequence consistency voting strategy to ensure that accurate UAV geolocalization is achieved. The specific process is shown in Fig. 9.

F. Loss Function

For the classification loss, the cross-entropy loss function is used. The optimization process aims to ensure that feature vectors from the same geographical location are closer together. The following is the classification loss formula:

$$L_{\text{cls}} = -\sum_{i=1}^{C} y_i \log(p_i) \tag{10}$$

where C is the total number of categories, p represents the prediction result, and y_i represents the ground-truth label using one-hot encoding. If the ith category is the correct category, then $y_i = 1$, otherwise $y_i = 0$. Therefore, only items of the correct category will be calculated in the total loss.

In addition, we compute the Kullback–Leibler (KL) loss based on the KL divergence to evaluate the disparity between the ground-truth and the predicted results, using the following formula:

$$\text{KLDiv}(p_1 \parallel p_2) = \sum_{i=1}^{N} p_1^i \log \frac{p_1^i}{p_2^i} \tag{11}$$

$$L_{\text{KL}} = \text{KLDiv}(p_1 \parallel p_2) + \text{KLDiv}(p_2 \parallel p_1) \tag{12}$$

where p_1 and p_2 represent the predicted results for the drone and satellite views, respectively.

Additionally, we utilize the triplet loss [81], [82] to minimize the distance between samples from different geographical locations. The triplet loss algorithm leverages the distance relationships among anchor, positive, and negative samples for training purposes. Here, the anchor and the positive samples belong to the same category, while the negative sample belongs to a different category. The formula for the triplet loss is as follows:

$$L_{\text{Triplet}} = \max\left[(d(s_1, s_2) - d(s_1, s_3) + m), 0\right] \tag{13}$$

$$d(s_1, x) = \|s_1 - x\|_2 \tag{14}$$

where $\| * \|_2$ denotes the 2-norm; s_1, s_2, and s_3 represent the anchor sample, the positive sample, and the negative sample, respectively; and m is a predefined hyperparameter.

The total loss is the sum of the classification loss, KL loss, and triplet loss

$$L_{\text{total}} = L_{\text{cls}} + L_{\text{KL}} + L_{\text{Triplet}}. \tag{15}$$

IV. Experimental Results and Discussion

A. Datasets and Evaluation Metrics

We conducted experiments based on University-1652 and cognition datasets. In this article, we train GeoFormer using University-1652 and perform ablation experiments. The GeoFormer performance is evaluated on the cognition test dataset. First, the satellite image is retrieved using the real-time image

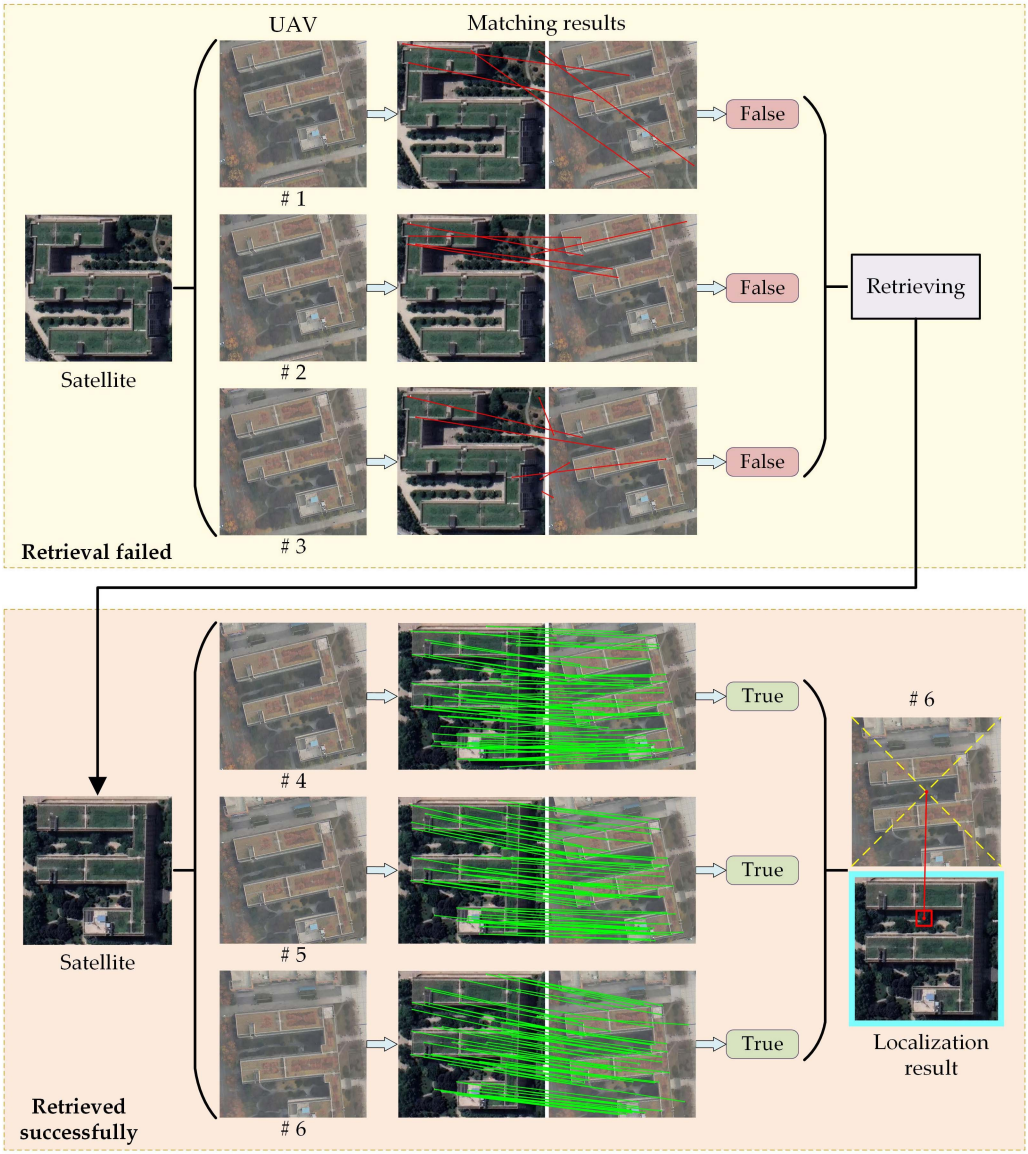


Fig. 9. Process of image sequence consistency voting strategy.

of the UAV view to get the approximate location of the UAV. Then, the precise geographic location of the UAV is obtained using homography transformation based on the matching results between the UAV and the satellite images.

University-1652 includes 1652 buildings from 72 global universities. Each building is associated with an average of 1 satellite view, 54 drone views, and 3.38 ground views. Here, we focused on satellite and drone views. The data structure of University-1652 is detailed in Table I. In the training dataset, there were 701 buildings from 33 universities, and a total of 38 555 images, comprising 37 854 drone views and 701 satellite views, were available for training purposes. The test dataset comprised 951 buildings from 39 universities. The training and test datasets did not overlap. This dataset was used to evaluate two tasks: UAV target localization (UAV→satellite) and UAV navigation (satellite→UAV). For the UAV→satellite task, only one satellite image authentically matched the drone query image.

TABLE I
DATA STRUCTURE OF UNIVERSITY-1652 DATASET

Views	Training		Test			
	Images	Classes	Drone→Satellite		Satellite→Drone	
			Images	Classes	Images	Classes
Drone	37,854	701	37,854	701	51,355	951
Satellite	701	701	951	951	701	701

In addition, we constructed a dataset named cognition, whose training and test datasets are independent of each other. In this article, only the cognition test dataset is utilized to test the effectiveness of the proposed method, and an example of sample images is shown in Fig. 10. Cognition is a multiview multisource dataset, and the data structure is shown in Table II. Cognition dataset includes eight UAV flight scenarios with no overlapping areas in Xi'an, Shaanxi Province, and Fengyang,

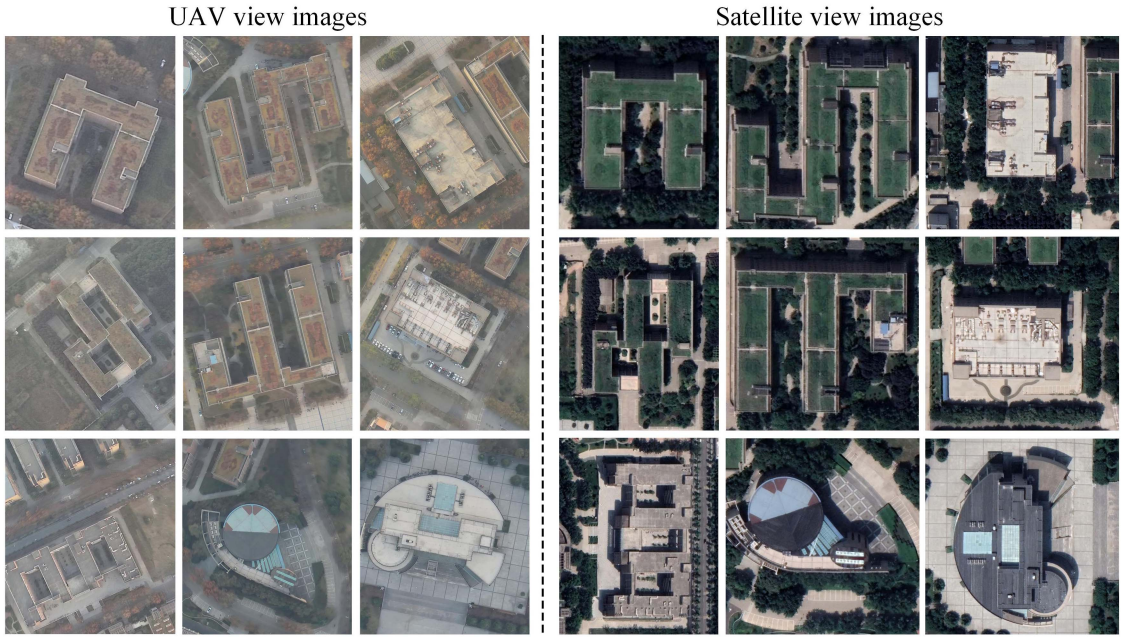


Fig. 10. Sample images of cognition test dataset.

TABLE II
DATA STRUCTURE OF COGNITION DATASET

Views	Training			Test		
	Images	Classes	Scene	Images	Classes	Scene
Drone	4095	25	5	2484	15	Sc 3
Satellite	25	25	5	15	15	3

Anhui Province, suburban areas, and five navigation landmarks are selected in each scenario. The cognition training dataset has a total of 25 navigation landmarks, each with an average of 1 satellite view and 163.8 UAV view images. Among them, the UAV view images for each navigation landmarks contain 108.2 visible and 55.6 infrared images on average.

The cognition test dataset used in this article includes 15 navigation landmarks under three scenarios, each navigation landmarks has 1 satellite view and 165.6 UAV view images on average, where the geographic information location of the satellite view images is known, and the UAV view images are the real-time image sequences captured by the UAVs during their flights according to the predetermined routes.

We employed recall@K (R@K) [49], [50], [83] and average precision (AP) [84], [85] as evaluation metrics. R@K focuses on the ability of the model to find the correct location in the top K retrieval results. The value of K in R@K depends on the requirements in practical applications, with smaller values of K corresponding to more stringent evaluation criteria. Given a query image, if the correctly matched image appears in t the top K images in the sorted list of retrieval results, this query is considered successful and the value of R@K is set to 1. Otherwise, R@K is set to 0, as shown in the following equation:

$$\text{Recall@}K = \begin{cases} 1, & \text{if order}_{\text{true}} \leq K \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

where $\text{order}_{\text{true}}$ is the sequence number of the first correctly matched image in the ranked list.

AP evaluates the performance of the model over the entire retrieval list, in particular for the sorting accuracy of the correct matches. AP calculates the area under the precision–recall curve. Specifically, whenever a correct match is retrieved, the current precision is computed and the average of these precision values is subsequently computed as the AP. Thus, the AP is able to synthesize the accuracy and completeness of the model during the retrieval process. The computation of the AP is as follows:

$$\text{AP} = \sum_{k=1}^{n} P(k) \times \Delta r(k) \tag{17}$$

$$\Delta r(k) = R(k) - R(k-1) \tag{18}$$

where $P(k)$ and $R(k)$ represent the accuracy and recall of the top-K matching results, and $R(0) = 0$.

In order to quantitatively analyze the results of the matching experiments, the matching algorithm is evaluated using correct matching points (CMPs), MA, and matching error (ME). CMPs are pairs of matching points that satisfy the following equation:

$$\text{CMP}(x, y) : \sqrt{(x_i - x_i')^2 + (y_i - y_i')^2} \leq \varepsilon \tag{19}$$

where (x_i, y_i) is the position of the matched feature point and (x_i', y_i') is the position of the true corresponding feature point. If the distance between them is less than the accuracy threshold ε, the feature point is the CMP.

MA is the percentage of feature points correctly matched to all feature points, calculated as follows:

$$\text{MA} = \frac{N_{\text{CMP}}}{N_{\text{all}}} \tag{20}$$

where N_{all} is the number of all feature point pairs matched and N_{CMP} is the number of feature point pairs correctly matched.

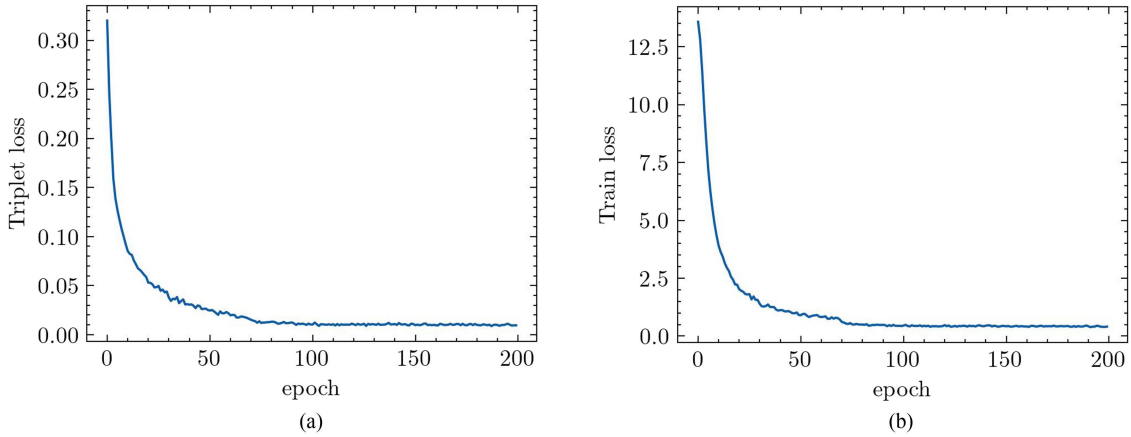


Fig. 11. Loss function variation curve: (a) shows the triplet loss and (b) shows the total loss.

ME refers to the accuracy of feature point matching, calculated as follows:

$$\mathrm{ME} = \frac{1}{\mathrm{CMP}} \sum_{i} \sqrt{\left(\mathbf{H}_2(x_i, y_i) - (x'_i, y'_i)\right)^2} \tag{21}$$

where $\mathbf{H}_2$ represents the true transformation model between the two images obtained after rotation matching. ME reflects the positional offset error at the pixel level of the matching point. Average matching error (AME) is the average of the ME obtained from each matching when performing multiple image matching operations. AME can be regarded as a quantitative measure of the average performance of an image matching algorithm, which effectively reflects the comprehensive performance and stability of the algorithms in different situations.

B. Implementation Details

We utilized image augmentation techniques to alleviate the issue of imbalanced samples. As there was only one satellite image for each category, k images were generated through random drifting, filling, cropping, coloring, and other enhancement techniques, where k represents the sampling rate. Simultaneously, k images were randomly selected from different perspectives, corresponding to the respective satellite view category. The ablation experiment provides a detailed study of the sampling rate k, and the experimental results indicated that GeoFormer performs best when $k = 2$.

In the training period, the input image was adjusted from 512 $\times$ 512 pixels to 224 $\times$ 224 pixels. The backbone was Kaiming-initialized [86] based on the pretrained weights for ImageNet1K. We employed the SGD optimizer with Nesterov momentum, with a weight decay of 5×10^{-4} and a momentum value of 0.9. The model was trained for 200 epochs, and the batch size was set to 8. The learning rate was reduced to one-tenth of its initial value when the training steps reached 70 or 110 epochs. Fig. 11 shows the loss function used during training. The training and testing processes are conducted using the PyTorch [87] platform and an Nvidia 3060 GPU.

C. Comparison to SOTA Methods

We compared GeoFormer with SOTA methods, including those employing different loss functions (soft-margin triplet loss [50], University-1652 [48], instance loss [13], [88], [89], LCM [47], LPN [51], SGM [60], and FSRA [21]). The comparative results are reported in Table III. SGM [60] and FSRA [21] are transformer-based approaches. Our proposed method achieved 89.08% R@1 and 90.83% AP for UAV→satellite, and 92.30% R@1 and 88.54% AP for satellite→UAV. Furthermore, as the number of GeoFormer parameters increased, the performance continuously improved.

GeoFormer outperforms existing state-of-the-art methods in R@1 and AP metrics while reducing computational costs. For instance, compared to the FSRA, GeoFormer improved by 1.71% R@1 and 1.57% AP in the UAV→satellite mission, while reducing parameter count by 26.55M. In the satellite→UAV mission, an increase of 0.85% R@1 and 2.1% AP were achieved, and the model accuracy was improved while significantly reducing the number of parameters. Compared with SGM, GeoFormer improved by 5.07% R@1 and 4.38% AP in the UAV→satellite mission, while reducing parameter count by 6.32M. In the satellite→UAV mission, an increase of 2.42% R@1 and 5.23% AP were achieved, significantly improving the accuracy of the model while reducing the number of parameters.

D. Ablation Study

1) Model Structure Ablation: The experimental results for GeoFormer-T, GeoFormer-S, GeoFormer-B, and GeoFormer-L on University-1652 are shown in Table IV. The test results of the trained GeoFormer-T, GeoFormer-S, GeoFormer-B, and GeoFormer-L on the cognition test dataset are shown in Table V. The backbone of GeoFormer-T, GeoFormer-S, GeoFormer-B, and GeoFormer-L are E-Swin-T, E-Swin-S, E-Swin-B, and E-Swin-L, respectively. During the experiment, the triplet loss ($M = 0.3$) and KL loss were added, with a sampling rate of $k = 2$ and a region number of $n = 3$. The input image sizes are all 224 $\times$ 224. The experimental results demonstrated that, as

TABLE III
COMPARISON WITH SOTA METHODS ON UNIVERSITY-1652

Method	Backbone	Resolution	Params	FLOPs	Drone→UAV		Satellite→UAV	
					R@1 (%)	AP (%)	R@1 (%)	AP (%)
Soft-margin triplet Loss[50]	VGG-16	256 × 256	138M	16G	53.21	58.03	65.62	54.47
University-1652[48]	ResNet50	256 × 256	26M	3.5G	58.49	63.13	71.18	58.74
Instance loss [88]	ResNet50	256 × 256	26M	3.5G	58.23	62.91	74.47	59.45
Instance loss + GeM pooling[89]	ResNet50	256 × 256	26M	3.5G	65.32	69.61	79.03	65.35
Instance loss + USAM[13]	ResNet50	256 × 256	48M	24G	65.63	69.68	78.32	64.87
LCM[47]	ResNet50	256 × 256	26M	3.5G	66.65	70.82	79.89	65.38
LPN[51]	ResNet50	256 × 256	26M	3.5G	74.16	77.39	85.16	73.68
SGM[60]	Swin-T	256 × 256	28M	5.9G	82.14	84.72	88.16	81.81
FSRA[21]	Vit-S	256 × 256	48.23M	18.84G	85.50	87.53	89.73	84.94
Ours ($k = 2$, $n = 3$)	E-Swin-T	224 × 224	21.68M	6.69G	87.21	89.10	90.58	87.04
	E-Swin-S	224 × 224	34.90M	11.72G	88.09	89.88	91.44	87.71
	E-Swin-B	224 × 224	61.01M	19.46G	88.16	90.03	91.87	87.92
	E-Swin-L	224 × 224	157M	37.7G	**89.08**	**90.83**	**92.30**	**88.54**

TABLE IV
BACKBONE COMPARISON ON UNIVERSITY-1652

Method	Backbone	Drone→ satellite				Satellite→ drone			
		R@1 (%)	R@5 (%)	R@10 (%)	AP (%)	R@1 (%)	R@5 (%)	R@10 (%)	AP (%)
GeoFormer-T	E-Swin-T	87.21	95.48	96.91	89.10	90.58	93.72	95.15	87.04
GeoFormer-S	E-Swin-S	88.09	95.90	97.14	89.88	91.44	94.72	95.58	87.71
GeoFormer-B	E-Swin-B	88.16	96.46	97.90	90.03	91.87	95.15	96.01	87.92
GeoFormer-L	E-Swin-L	**89.08**	**96.83**	**98.09**	**90.83**	**92.30**	**95.29**	**96.29**	**88.54**

TABLE V
BACKBONE COMPARISON ON COGNITION DATASET

Method	Drone→ satellite		Satellite→ drone	
	R@1 (%)	AP (%)	R@1 (%)	AP (%)
GeoFormer-T	65.45	71.49	73.33	60.71
GeoFormer-S	68.11	72.09	86.67	67.23
GeoFormer-B	68.77	73.31	86.67	65.19
GeoFormer-L	**76.08**	**79.76**	**93.33**	**73.83**

the model parameter quantity increased, the values of the metrics also increased.

On University-1652, GeoFormer-L compared to the smallest GeoFormer-T, GeoFormer-L demonstrated improvements in the R@1 value (from 87.21% to 89.08%), the R@5 value (from 95.48% to 96.83%), the R@10 value (from 96.91% to 98.96%), and the AP value (from 89.10% to 90.83%) in the drone→satellite task. In the satellite→drone task, the R@1 value was improved from 90.58% to 92.30%, the R@5 value improved from 93.72% to 95.29%, the R@10 value improved from 95.15% to 96.29%, and the AP value improved from 87.04% to 88.54%, demonstrating overall improved performance with this model.

2) Key Components Ablation: To explore the effectiveness of ESP, MHSA, MFAM, and SRSM, we performed ablation experiments of key components on GeoFormer-T. The input image size in the experiment was 224 × 224. Triplet loss ($M = 0.3$) and KL loss are added during the experiments, with the sampling rate k set to 2 and the number of regions $n = 3$. We kept the original design of the GeoFormer-T unchanged as a baseline, and the results of the experiments are shown in Group 5 in Table VI.

For the ESP module, by comparing the experiments in Group 1 and Group 5, we find that after utilizing the ESP module instead of the original MLP in the backbone, the number of parameters is reduced from 29.07M to 21.68M, the computation amount is decreased from 9.36G to 6.69G, and the FPS is improved from 48.89 to 53.00. This indicates that the ESP module can effectively reduce the computational complexity of the model and the computational cost, thus improving computational efficiency. Meanwhile, the retrieval performance of the model is significantly improved on the satellite→UAV and UAV→satellite tasks. This shows that the ESP module can effectively improve the computational efficiency and retrieval accuracy of the model.

For the linear attention-based MHSA module, by comparing the experiments in Groups 2 and 5, it can be found that compared with the dot-product attention-based MHSA, the linear attention

TABLE VI
KEY COMPONENTS ABLATION STUDY ON UNIVERSITY-1652

Group	ESP	MHSA (linear)	MFAM	SRSM	Params	FLOPs	FPS	Drone→satellite		Satellite→drone	
								R@1 (%)	AP (%)	R@1 (%)	AP (%)
1	—	√	√	√	29.07M	9.36G	48.89	82.27	84.82	88.87	82.34
2	√	—	√	√	21.68M	7.61G	47.94	86.62	**89.75**	**90.79**	**88.43**
3	√	√	—	√	**20.61**M	**6.07**G	53.48	86.17	88.21	90.44	85.39
4	√	√	√	—	21.68M	6.69G	**58.52**	85.45	87.49	89.02	84.06
5	√	√	√	√	21.68M	6.69G	53.00	**87.21**	89.10	90.58	87.04

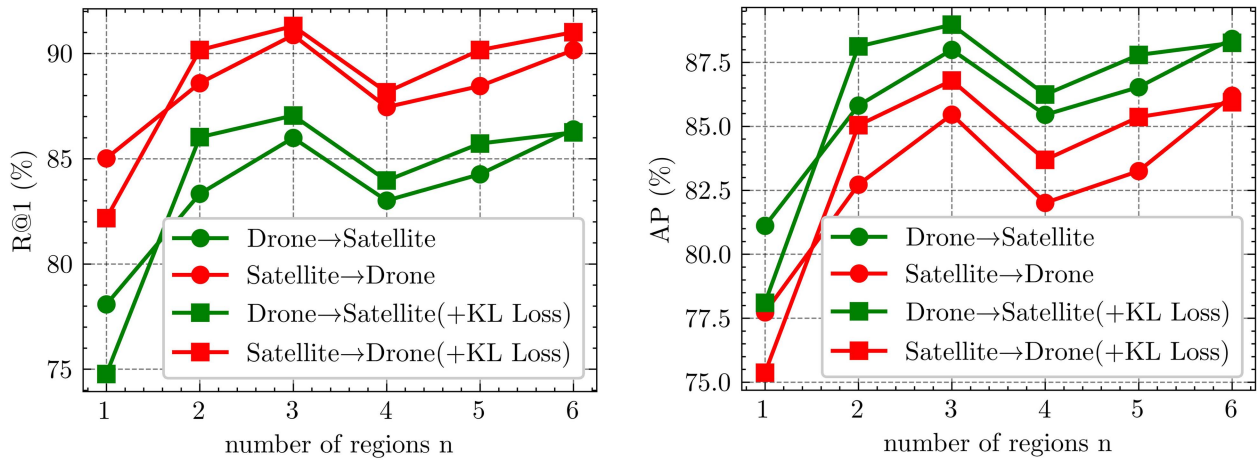


Fig. 12. Variation curve of the number of semantic regions n in the drone→satellite and satellite→drone tasks.

we adopt does not change the number of model parameters, but the computation amount of the model is reduced from 7.61G to 6.69G, and the FPS of the model is improved from 47.94 to 53.00. The computational amount of the model is reduced and computational efficiency is improved. Meanwhile, the R@1 on the UAV→satellite task is improved with the adoption of linear attention in MHSA, which is crucial for the practical task of retrieving satellite images based on UAV aerial images.

After adding MFAM, the R@1 and AP increased by 1.04% and 0.89%, respectively, on the UAV→satellite task, and by 1.65% and 0.14%, respectively, on the satellite→UAV task. The MFAM enhances the efficiency of the model in utilizing multi-scale features, thereby improving the accuracy and robustness of geolocalization.

Since the SRSM does not contain a neural network, the number of parameters and computational complexity calculated in the Table VI are for the neural network, so the number of parameters and computational complexity after removing the SRSM are the same as in the baseline, but the FPS is increased due to the reduction in computation of the entire code. In addition, the geolocalization performance can be effectively improved by adding the SRSM.

3) Semantic Region Quantity Ablation: The number of semantic region segmentations is an important metric for GeoFormer. Experiments were performed to determine whether the number of semantic regions affected R@1 and AP, and the results are shown in Fig. 12. We perform experiments on GeoFormer-T based on the triplet loss ($M = 0.3$), with a sampling rate of $k = 1$ and input images size of 224×224. The green line represents UAV target localization tasks (drone→satellite), while the red line represents UAV navigation tasks (satellite→drone). The optimal performance for R@1 and AP was observed when the number of regions was three. Furthermore, it was found that the performance of R@1 and AP was optimal when n is a multiple of 3. To decrease the parameters of the network and computational complexity, we thus used $n = 3$ as the default setting in subsequent experiments.

4) Sampling Rate Ablation: For every satellite image in University-1652, there are 54 UAV-view images. This significant disparity in sample quantities may lead to the model assigning a higher weight to the more numerous classes during prediction, thereby affecting its performance. We employed a synthetic sampling method to address the problem of unbalanced dataset samples.

The sampling rate, k, can be considered as a hyperparameter. Under the conditions of the triplet loss ($M = 0.3$) and $n = 3$, we conducted sampling rate ablation experiments using GeoFormer-T, as depicted in Fig. 13. It was observed that the AP and R@1 indicators exhibited a trend of increasing followed by decreasing, reaching an overall optimum at $k = 2$. Furthermore, we trained the model by adding the KL divergence loss. The AP and R@1 indicators similarly displayed a pattern of increase followed by decrease, reaching their best levels at $k = 2$. The value of k influenced the training time but had no effect on inference. Additionally, under the conditions of adding the KL loss, triplet loss ($M = 0.3$), and $n = 3$, we compared k across different

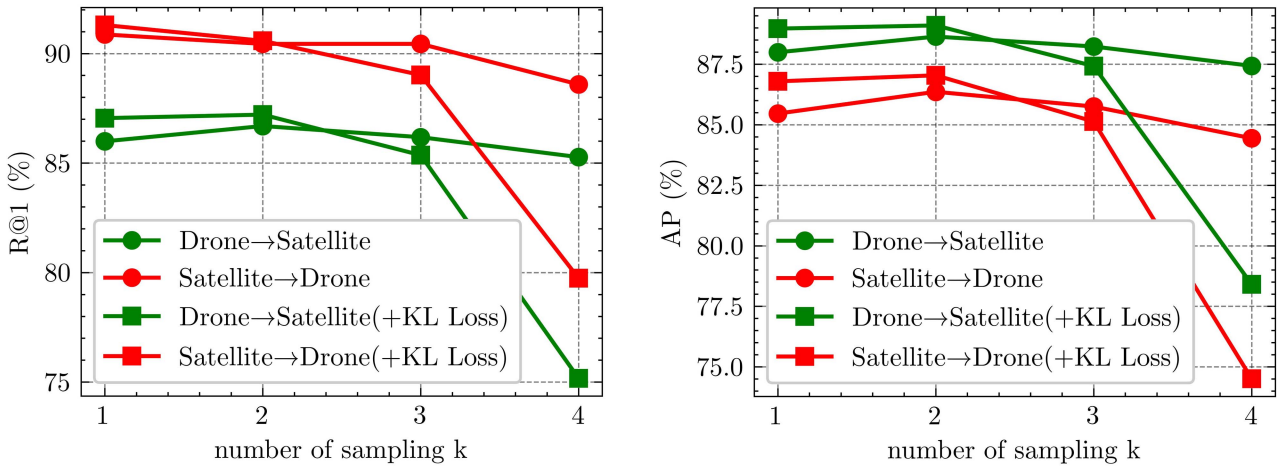


Fig. 13. Variation curve of the sampling rate k in the drone→satellite and satellite→drone tasks.

TABLE VII
COMPARISON OF SAMPLING RATES FOR DIFFERENT MODEL STRUCTURES

Method	K	Drone→satellite		Satellite→drone	
		R@1 (%)	AP (%)	R@1 (%)	AP (%)
GeoFormer-T	1	87.05	88.97	91.30	86.79
	2	87.21	89.10	90.58	87.04
GeoFormer-S	1	87.19	89.08	90.87	86.47
	2	88.09	89.88	91.44	87.71
GeoFormer-B	1	87.21	89.20	91.73	86.58
	2	88.16	90.03	91.87	87.92
GeoFormer-L	1	88.39	90.34	91.87	87.03
	2	**89.08**	**90.83**	**92.30**	**88.54**

TABLE VIII
COMPARISON OF KL LOSS UNDER DIFFERENT MODEL STRUCTURES

Method	Kl_loss	UAV→satellite		Satellite→UAV	
		R@1 (%)	AP (%)	R@1 (%)	AP (%)
GeoFormer-T	—	85.99	87.99	90.87	85.46
	√	87.05	88.97	91.30	86.79
GeoFormer-S	—	86.09	88.19	91.44	86.00
	√	87.19	89.08	90.87	86.47
GeoFormer-B	—	87.30	89.26	91.30	86.89
	√	87.21	89.20	91.73	86.58
GeoFormer-L	—	87.63	89.54	91.30	86.69
	√	**88.39**	**90.34**	**91.87**	**87.03**

model structures, as presented in Table VII. We observed that the performance in terms of R@1 and AP was optimal when $k = 2$. For instance, in the drone→satellite task, GeoFormer-L achieved 89.08% R@1 and 90.83% AP at $k = 2$, representing an improvement of 0.69% in R@1 and 0.49% in AP compared to $k = 1$. In the satellite→drone task, GeoFormer-L attained 92.30% R@1 and 88.54% AP at $k = 2$, marking a 0.43% increase in R@1 and 1.51% increase in AP compared to $k = 1$, thus yielding higher precision.

5) Loss Function Ablation: To examine how the loss function affects the various model architectures, we first compared the presence and absence of the KL loss under the conditions of $k = 1$, triplet loss ($M = 0.3$), and $n = 3$, as shown in Table VIII.

It was observed that the performance in terms of R@1 and AP was optimal when the KL loss was added. For instance, in the UAV→satellite task, GeoFormer-L achieved 88.39% R@1 and 90.34% AP with the addition of the KL loss, leading to an improvement of 0.79% in R@1 and 0.80% in AP compared to when the KL loss was not used. Similarly, in the satellite→UAV task, GeoFormer-L attained 91.87% R@1 and 87.03% AP with the addition of the KL loss, an improvement of 0.57% in R@1 and 0.34% in AP compared to when the KL loss was not added, thus achieving higher accuracy. We employed three strategies to improve the UAV and satellite-view picture matching task performance: The KL loss, the triplet loss ($M = 0.3$), and multisampling when $n = 3$. Table IX displays the results of the ablation experiment.

TABLE IX
EXPERIMENTAL RESULTS OF LOSS FUNCTION ABLATION UNDER DIFFERENT SAMPLING RATES

k	Loss function			UAV→satellite		Satellite→UAV	
	CE	KL	Triplet	R@1 (%)	AP (%)	R@1 (%)	AP (%)
1	√	—	—	81.92	84.53	86.45	80.84
	√	√	—	85.85	87.97	91.73	85.32
	√	—	√	85.99	87.99	90.87	85.46
	√	√	√	87.05	88.97	91.30	86.79
2	√	—	—	79.97	82.79	86.02	79.87
	√	√	—	83.99	86.33	87.73	82.71
	√	—	√	86.69	88.64	90.44	86.36
	√	√	√	87.21	89.10	90.58	87.04

In the UAV→satellite task, when $k = 1$, using only the KL loss increased R@1 by 3.93% and AP by 3.44%, while using only the triplet loss increased R@1 by 4.07% and AP by 3.46%. When both the KL loss and triplet loss were used simultaneously,

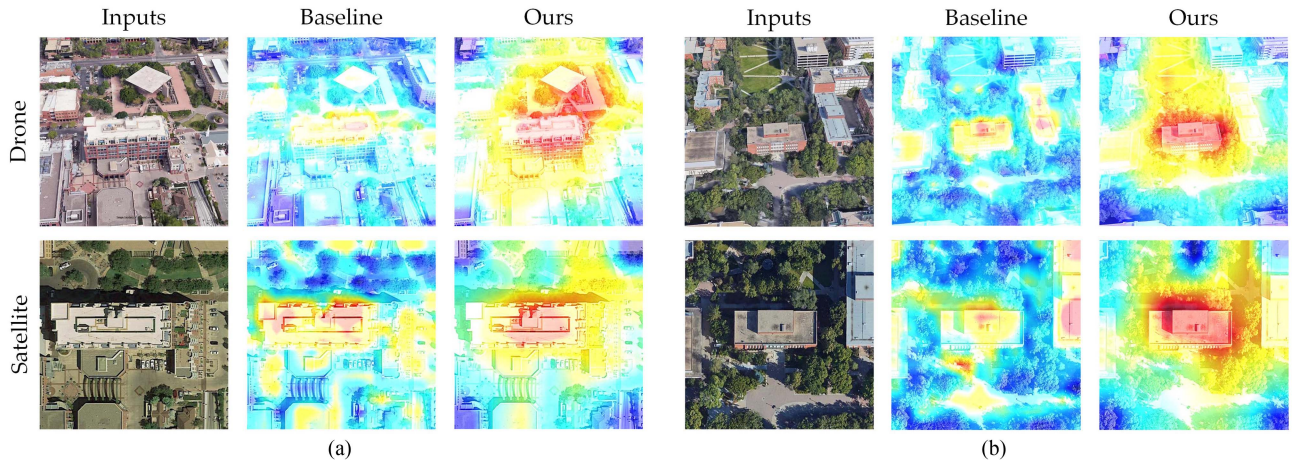


Fig. 14. Comparison of baseline and proposed method for visualization of heat maps. (a) Scenario 1. (b) Scenario 2.

TABLE X
PERFORMANCE COMPARISON OF MATCHING METHODS ON COGNITION DATASET

Methods	N_{all}	N_{CMP}	MA/%	AME/pixel	Time/s
SIFT [69]	201.4	15.8	7.67	12.45	0.14
SURF [70]	191.8	13	7.19	9.36	0.17
ORB [71]	347	14.2	3.95	13.22	0.14
LoFTR [65]	572	320	37.79	11.22	0.21
COTR [64]	93.4	63.8	66.26	13.26	47.15
SuperGlue [79]	181	162.2	**88.01**	6.35	0.18
LightGlue [80]	258	211	55.68	8.83	**0.07**
Ours	**382**	**315.8**	81.67	**4.02**	0.14

there was a 5.13% increase in R@1 and 4.44% increase in AP. In the satellite→UAV task, when $k = 2$, using only the KL loss increased R@1 by 1.71% and AP by 2.84, while using only the triplet loss led to a 4.42% increase in R@1 and 6.49% increase in AP. When both the KL loss and triplet loss were used simultaneously, there was an 4.56% increase in R@1 and 7.17% increase in AP. Therefore, the KL loss and triplet loss were concurrently incorporated into GeoFormer.

E. Matching Methods Comparison

In order to evaluate the matching performance of the proposed matching algorithm, it is compared with several popular feature point matching methods and transformer-based matching algorithms on the cognition dataset, including SIFT [69], SURF [70], ORB [71], LoFTR [65], COTR [64], SuperGlue [79], and LightGlue [80]. In particular, the SuperGlue, LightGlue, and the proposed method are further matched on the basis of SuperPoint feature point extraction. The CMP, MA, AME, and matching time of each matching algorithm are counted, where the accuracy threshold ε of CMP is taken as 30 pixels, and the AME is the average of the ME obtained from 10 experiments conducted in each scenario (5 scenarios in total), and the results are shown in Table X.

Comparing and analyzing the data in the Table X, it can be observed that under five different scenarios, our proposed method matches the highest number of feature points among all the compared algorithms, with an MA of 81.67%. This reflects that it is not only able to recognize a large number of feature points, but also able to match these points efficiently and accurately. Although the matching time is not the fastest among all the algorithms, it is a comprehensive performance algorithm that ensures the accuracy of the matching while maintaining a fast-matching speed. Therefore, it is very suitable for practical applications that need to consider both speed and accuracy. In addition, the proposed method has a high MA with an AME of 4.02 pixel, which is especially important for the subsequent UAV precise localization tasks.

V. DISCUSSION

To further substantiate the reliability of the proposed method, we visualized the heat maps of both GeoFormer and the baseline method, as depicted in Fig. 14. By comparing the heat maps generated by the baseline and GeoFormer models for drone and satellite views, it was observed that our method exhibited a higher focus on critical areas, particularly those pertaining to geographically referenced target structures. Simultaneously, GeoFormer activates the regions where geographic targets are situated, as well as their adjacent areas, thus emphasizing global information. GeoFormer aligns more closely with the perceptual processes of the human visual system; for the image to be recognized, preliminary discrimination is made by paying attention to the salient features, following which contextual information is used for further perception.

The visualization of the experimental result of GeoFormer on the University-1652 dataset is depicted in Fig. 15. A proper matching image is indicated by a green box, while a wrong one is shown by a red box. For the drone→satellite task, three drone images were chosen at random from the test dataset. After that, comparable satellite photos were selected from the satellite gallery dataset and ordered according to their similarity. We took the five most similar drone images out of the retrieval

Fig. 15. Visualization of top-five results of image retrieval on University-1652 dataset. (a) Drone→satellite task. (b) Satellite→drone task. : correctly retrieved images, : misretrieved images.

results for each one. As each drone image category corresponds to only one satellite image, the localization result was entirely accurate, as demonstrated in Fig. 15(a). For the satellite→drone task, we selected three satellite-view images at random from the test dataset. We collected similar drone-view images from the drone gallery dataset for each satellite-view image and ranked them based on their similarity. We then selected the top-five retrieval results for each satellite-view image, as illustrated in Fig. 15(b). The proposed GeoFormer still yielded entirely accurate results. The experimental results demonstrate that this method exhibits high top-five accuracy in UAV target localization and UAV navigation tasks, thus confirming the reliability of the proposed approach.

We perform UAV→satellite and satellite→UAV image retrieval on the cognition dataset, and the results are shown in Fig. 16. For the UAV→satellite task, the center point of the UAV-view image represents the position of the UAV, since the UAV-view image is obtained from a front down view. For each UAV-view image, there is only one corresponding satellite image. We retrieve the top-five satellite images in terms of similarity and get exactly the right localization results. We observe that the localization ability of the model is affected when the

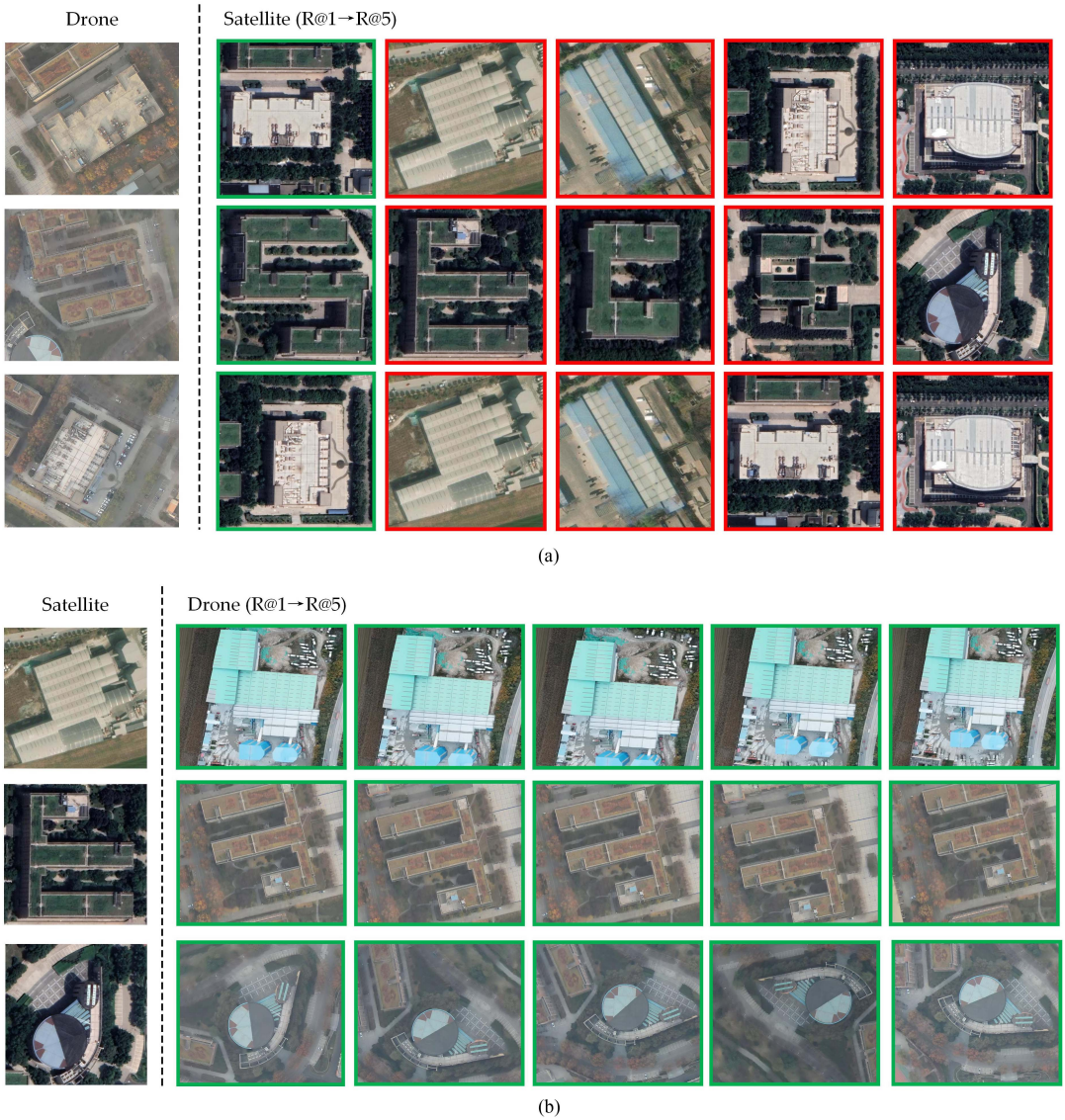


Fig. 16. Visualization of top-five results of image retrieval on cognition dataset. (a) Drone→satellite task. (b) Satellite→drone task. : correctly retrieved images, : misretrieved images.

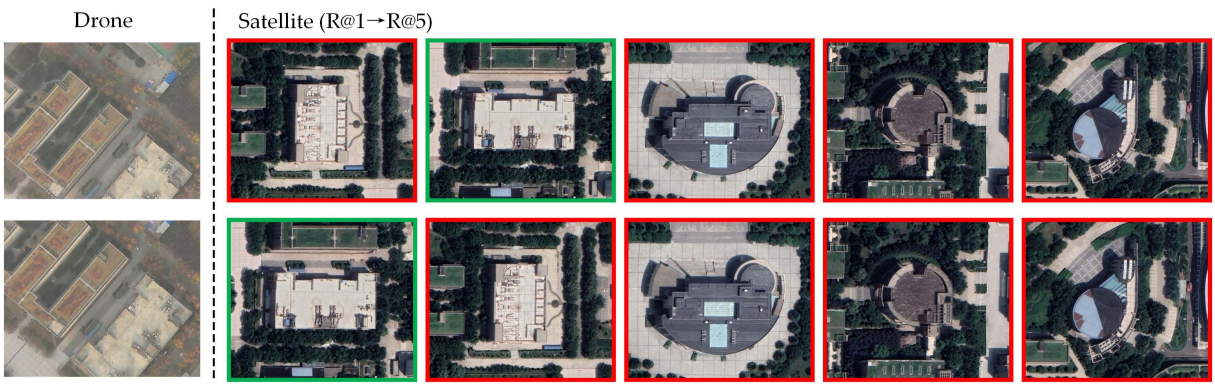


Fig. 17. Difficult and incorrect examples on cognition dataset. : correctly retrieved images, : misretrieved images.

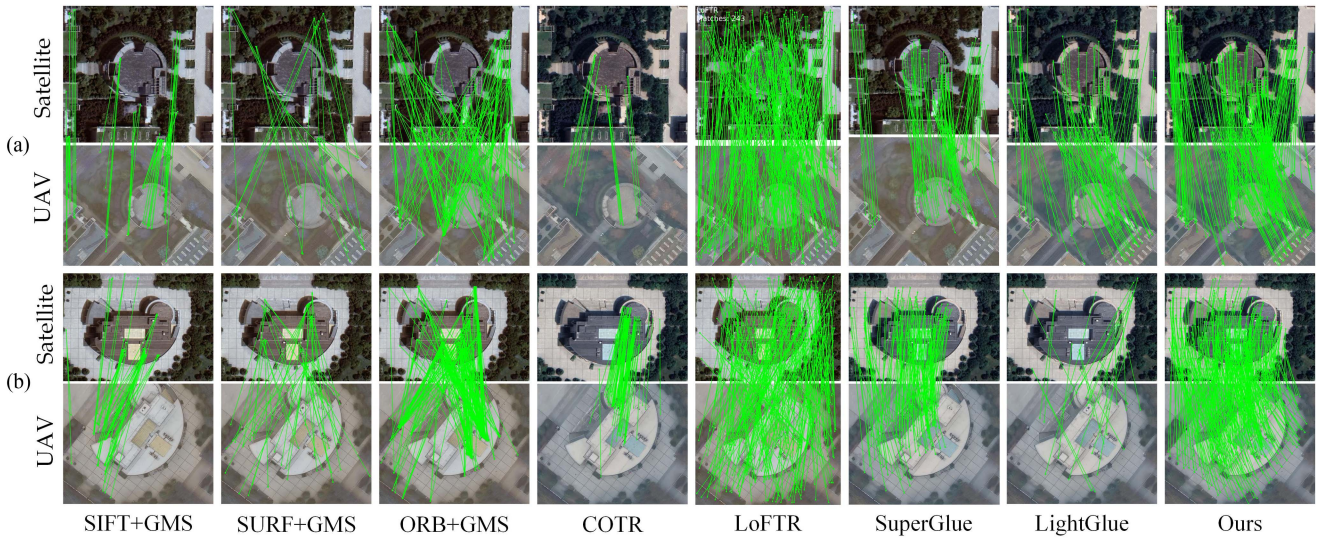


Fig. 18. Comparison of matching results. (a) Scenario 1. (b) scenario 2.

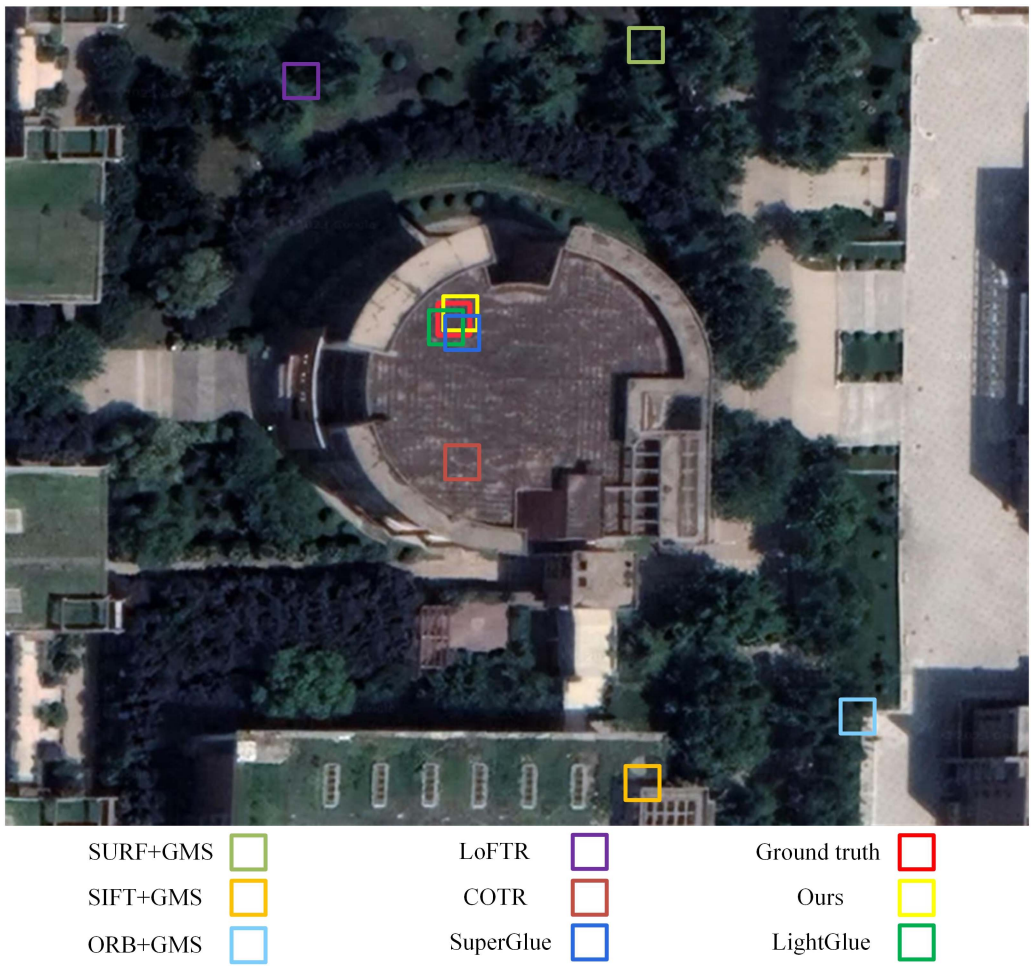


Fig. 19. Visualization of positioning results.

navigation landmarks are shown in only a small part of the image or when large areas are occluded. The failure cases are shown in Fig. 17. In real-world scenarios, target buildings may often be occluded by foreground objects, such as trees or other buildings, resulting in the loss of key feature information. Differences in capturing angles and distances may also cause key features of the target to be only partially displayed in the image. When the key features are incomplete due to occlusion or image capture angle, it does not provide enough information for feature extraction and recognition. In addition, there are a large number of similar buildings and structures in urban environments, and these similar navigation landmarks may have similar visual features to the target building, increasing the difficulty of matching. Therefore, the methods may have limitations in distinguishing highly similar objects and may not be able to effectively distinguish subtle differences, especially when the feature extraction algorithms are unable to capture sufficiently rich feature information.

Based on the retrieval results, the approximate location of the UAV can be obtained, and then the feature point matching algorithm can be used to precisely locate the UAV. The comparison results of the matching algorithms are shown in Fig. 18, which shows that the proposed matching algorithm has the least number of mismatches and the densest number of CMPs. According to the matching results, the homography transformation matrix can be obtained to determine the corresponding position of the center point of the UAV view in the satellite view, and obtain the precise positioning result of the UAV, as shown in Fig. 19. It can be seen that the localization results of the matching method proposed are closest to the ground truth, proving the effectiveness of our method.

During the imaging process of UAV aerial and satellite imagery, the data are inevitably affected by various variabilities, such as illumination changes, imaging angles, atmospheric perturbations, sensor noise, and spatial and temporal variations in the features themselves [90]. These variabilities will undoubtedly have a negative impact on the accuracy of UAV cross-view geolocalization tasks. To address these variabilities, GeoFormer effectively fuses features at different scales through an MFAM, which helps the model capture semantic feature information from detail to global, enhancing the ability to capture image details and represent robust features. The designed SRSM helps to improve the accuracy of feature matching when the imaging conditions and observation angles change, and reduces the possibility of mismatching by matching features within nonoverlapping semantic regions. In addition, the designed hierarchical reinforcement rotation matching method can effectively improve the matching localization accuracy in the case of large differences in imaging angles. Future article can further explore adaptive strategies for various variabilities to enhance the performance of the model in more widely used scenarios.

VI. Conclusion

We proposed an effective transformer-based Siamese network, called GeoFormer, specifically designed for UAV cross-view geolocalization. We designed the ESP module in the efficient transformer feature extraction network to reduce the computational complexity while effectively extracting global features and contextual information using linear attention. Furthermore, our proposed MFAM can effectively fuse multiscale features and improve the robust feature representation ability. Additionally, our designed SRSM improves the accuracy of feature matching by dividing the feature map into nonoverlapping semantic regions and performing feature matching within each semantic region. By designing loss functions and multiple sampling strategies, GeoFormer was adjusted to a better state. Experiments on the University-1652 dataset indicated that GeoFormer exhibits state-of-the-art performance. Experiments on the cognition dataset validate the effectiveness of the proposed UAV geolocalization method. While the proposed GeoFormer demonstrated a high retrieval accuracy and strong robustness, there is still room for further improvement; for example, the feature extraction network of GeoFormer could be further simplified to reduce computation time. Although the proposed GeoFormer has high retrieval accuracy and strong robustness, there is still room for further improvement. In the future, more efficient and lightweight model structures will be the focus of research in order to better adapt to the needs of edge computing and mobile devices. Future article needs to focus more on model inference speed and energy efficiency while maintaining model accuracy. This includes exploring new hardware acceleration techniques, optimizing algorithms to reduce unnecessary computations, and investigating energy efficiency optimization strategies. Algorithm optimization includes further simplifying the feature extraction network, exploring new model compression techniques and knowledge distillation methods to maintain high accuracy while significantly reducing model parameters and inference time.

References

[1] J. W. Fan, X. G. Yang, R. T. Lu, X. L. Xie, and W. P. Li, "Design and implementation of intelligent inspection and alarm flight system for epidemic prevention," *Drones*, vol. 5, no. 3, Sep. 2021, Art. no. 68.

[2] R. Lu, X. Yang, W. Li, J. Fan, D. Li, and X. Jing, "Robust infrared small target detection via multidirectional derivative-based weighted contrast measure," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 7000105.

[3] R. Lu et al., "Infrared small target detection based on local hypergraph dissimilarity measure," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 7000405.

[4] J. Fan, R. Lu, X. Yang, F. Gao, Q. Li, and J. Zeng, "Design and implementation of intelligent EOD system based on six-rotor UAV," *Drones*, vol. 5, no. 4, 2021, Art. no. 146.

[5] Q. Li, X. Yang, R. Lu, J. Fan, S. Wang, and Z. Qin, "VisionICE: Air-ground integrated intelligent cognition visual enhancement system based on a UAV," *Drones*, vol. 7, no. 4, Apr. 2023, Art. no. 268.

[6] D. Hong et al., "Cross-city matters: A multimodal remote sensing benchmark dataset for cross-city semantic segmentation using high-resolution domain adaptation networks," *Remote Sens. Environ.*, vol. 299, Dec. 2023, Art. no. 113856.

[7] D. Hong et al., "SpectralGPT: Spectral foundation model," 2023, *arXiv:2311.07113*.

[8] D. Hong, J. Yao, C. Li, D. Meng, N. Yokoya, and J. Chanussot, "Decoupled-and-coupled networks: Self-supervised hyperspectral image super-resolution with subpixel fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5527812.

[9] J. Wang, W. Li, Y. Gao, M. Zhang, R. Tao, and Q. Du, "Hyperspectral and SAR image classification via multiscale interactive fusion network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 12, pp. 10823–10837, Dec. 2023.

[10] J. Wang, W. Li, M. Zhang, R. Tao, and J. Chanussot, "Remote-sensing scene classification via multistage self-guided separation network," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5615312.

[11] M. Zhang, W. Li, X. Zhao, H. Liu, R. Tao, and Q. Du, "Morphological transformation and spatial-logical aggregation for tree species classification using hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5501212.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[13] 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.

[14] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

[15] Y. Xu, P. Shamsolmoali, E. Granger, C. Nicodeme, L. Gardes, and J. Yang, "TransVLAD: Multi-scale attention-based global descriptors for visual geo-localization," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2023, pp. 2839–2848.

[16] H. Yang, X. Lu, and Y. Zhu, "Cross-view geo-localization with layer-to-layer transformer," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021.

[17] X. Zhang, X. Li, W. Sultani, Y. Zhou, and S. Wshah, "Cross-view geo-localization via learning disentangled geometric layout correspondence," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 3480–3488.

[18] S. Zhu, M. Shah, and C. Chen, "TransGeo: Transformer is all you need for cross-view image geo-localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 1152–1161.

[19] A. Dosovitskiy et al., "An image is worth 16 × 16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021.

[20] S. Pramanick, E. M. Nowara, J. Gleason, C. D. Castillo, and R. Chellappa, "Where in the world is this image? Transformer-based geo-localization in the wild," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 196–215.

[21] 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.

[22] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9992–10002.

[23] Z. Liu et al., "Swin transformer V2: Scaling up capacity and resolution," 2021, *arXiv:2111.09883*.

[24] F. Deuser, K. Habel, and N. Oswald, "Sample4Geo: Hard negative sampling for cross-view geo-localisation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 16801–16810.

[25] Y. Guo, M. Choi, K. Li, F. Boussaid, and M. Bennamoun, "Soft exemplar highlighting for cross-view image-based geo-localization," *IEEE Trans. Image Process.*, vol. 31, pp. 2094–2105, 2022.

[26] X. Zhang et al., "SSA-net: Spatial scale attention network for image-based geo-localization," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 8022905.

[27] 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, 2022, Art. no. 4704315.

[28] Z. Zeng, Z. Wang, F. Yang, and S. Satoh, "Geo-localization via ground-to-satellite cross-view image retrieval," *IEEE Trans. Multimedia*, vol. 25, pp. 2176–2188, 2023.

[29] S. Zhu, T. Yang, and C. Chen, "Revisiting street-to-aerial view image geo-localization and orientation estimation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 756–765.

[30] G. Berton, C. Masone, and B. Caputo, "Rethinking visual geo-localization for large-scale applications," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 4868–4878.

[31] 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.

[32] S. Zhu, T. Yang, and C. Chen, "VIGOR: Cross-view image geo-localization beyond one-to-one retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 5316–5325.

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

[34] G. Berton et al., "Deep visual geo-localization benchmark," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 5386–5397.

[35] 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.*, 2018, pp. 7258–7267.

[36] J. Li, C. Yang, B. Qi, M. Zhu, and N. Wu, "4SCIG: A four-branch framework to reduce the interference of sky area in cross-view image geo-localization," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2024.3379376.

[37] B. Fan et al., "Learning semantic-aware local features for long term visual localization," *IEEE Trans. Image Process.*, vol. 31, pp. 4842–4855, 2022.

[38] Y. Tian, B. Fan, and F. Wu, "L2-net: Deep learning of discriminative patch descriptor in Euclidean space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6128–6136.

[39] Z. Zhang, T. Sattler, and D. Scaramuzza, "Reference pose generation for long-term visual localization via learned features and view synthesis," *Int. J. Comput. Vis.*, vol. 129, pp. 821–844, 2021.

[40] H. Taira et al., "InLoc: Indoor visual localization with dense matching and view synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7199–7209.

[41] Y. Shi, X. Yu, D. Campbell, and H. Li, "Where am I looking at? Joint location and orientation estimation by cross-view matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4063–4071.

[42] R. Rodrigues and M. Tani, "Global assists local: Effective aerial representations for field of view constrained image geo-localization," in *Proc. 22nd IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2022, pp. 2694–2702.

[43] Y. Shi, X. Yu, L. Liu, D. Campbell, P. Koniusz, and H. Li, "Accurate 3-DoF camera geo-localization via ground-to-satellite image matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 2682–2697, Mar. 2023.

[44] T. Y. Chu, Y. M. Chen, H. Su, Z. Z. Xu, G. D. Chen, and A. N. Zhou, "A news picture geo-localization pipeline based on deep learning and street view images," *Int. J. Digit. Earth*, vol. 15, no. 1, pp. 1485–1505, Dec. 2022.

[45] X. Tian, J. Shao, D. Ouyang, and H. T. Shen, "UAV-satellite view synthesis for cross-view geo-localization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 7, pp. 4804–4815, Jul. 2022.

[46] A. Toker, Q. Zhou, M. Maximov, and L. Leal-Taixé, "Coming down to earth: Satellite-to-street view synthesis for geo-localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 6484–6493.

[47] L. R. Ding, J. Zhou, L. X. Meng, and Z. Y. Long, "A practical cross-view image matching method between UAV and satellite for UAV-based geo-localization," *Remote Sens.*, vol. 13, no. 1, Jan. 2021, Art. no. 47.

[48] 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*, 2020, pp. 1395–1403.

[49] 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.*, 2017, pp. 4132–4140.

[50] L. Liu and H. Li, "Lending orientation to neural networks for cross-view geo-localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5617–5626.

[51] 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.

[52] J. D. Zhuang, M. Dai, X. R. Y. Chen, and E. H. Zheng, "A faster and more effective cross-view matching method of UAV and satellite images for UAV geolocalization," *Remote Sens.*, vol. 13, no. 19, Oct. 2021, Art. no. 3979.

[53] S. Li, Z. Tu, Y. Chen, and T. Yu, "Multi-scale attention encoder for street-to-aerial image geo-localization," *CAAI Trans. Intell. Technol.*, vol. 8, pp. 166–176, 2023.

[54] S. Li, M. Hu, X. Xiao, and Z. Tu, "Patch similarity self-knowledge distillation for cross-view geo-localization," *IEEE Trans. Circuits Syst. Video Technol.*, to be published, doi: 10.1109/TCSVT.2023.3336844.

[55] R. Rodrigues and M. Tani, "Are these from the same place? Seeing the unseen in cross-view image geo-localization," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 3752–3760.

[56] F. Xue, I. Budvytis, and R. Cipolla, "SFD2: Semantic-guided feature detection and description," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 5206–5216.

[57] Z. Cui et al., "A novel geo-localization method for UAV and satellite images using cross-view consistent attention," *Remote Sens.*, vol. 15, no. 19, 2023, Art. no. 4667.

[58] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5297–5307.

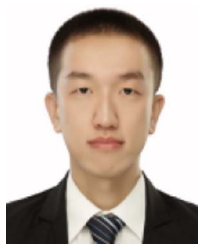
[59] U. Efe, K. G. Ince, and A. Aydin Alatan, "DFM: A performance baseline for deep feature matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4279–4288.
[60] J. Zhuang, X. Chen, M. Dai, W. Lan, Y. Cai, and E. Zheng, "A semantic guidance and transformer-based matching method for UAVs and satellite images for UAV geo-localization," *IEEE Access*, vol. 10, pp. 34277–34287, 2022.
[61] S. Suri and P. Reinartz, "Mutual-information-based registration of TerraSAR-X and Ikonos imagery in urban areas," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 2, pp. 939–949, Feb. 2010.
[62] Y. Fang, J. Hu, C. Du, Z. Liu, and L. Zhang, "SAR-optical image matching by integrating Siamese U-Net with FFT correlation," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4016505.
[63] O. Kwon, *Similarity Measures for Object Matching in Computer Vision*. Bolton, U.K.: Univ. Bolton, 2016.
[64] W. Jiang, E. Trulls, J. Hosang, A. Tagliasacchi, and K. M. Yi, "Cotr: Correspondence transformer for matching across images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 6187–6197.
[65] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "LoFTR: Detector-free local feature matching with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8918–8927.
[66] H. Chen et al., "ASpanFormer: Detector-free image matching with adaptive span transformer," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 20–36.
[67] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. Alvey Vis. Conf.*, 1988.
[68] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 430–443.
[69] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[70] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 407–417.
[71] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.
[72] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "LIFT: Learned invariant feature transform," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 467–483.
[73] M. Dusmanu et al., "D2-net: A trainable CNN for joint description and detection of local features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8084–8093.
[74] J. Revaud, P. Weinzaepfel, C. De Souza, and M. Humenberger, "R2D2: Repeatable and reliable detector and descriptor," in *Proc. 33rd Conf. Neural Inf. Process. Syst.*, 2019, pp. 1–12.
[75] Y. Liu, Z. Shen, Z. Lin, S. Peng, H. Bao, and X. Zhou, "GIFT: Learning transformation-invariant dense visual descriptors via group CNNs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019.
[76] M. Tyszkiewicz, P. Fua, and E. Trulls, "DISK: Learning local features with policy gradient," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 14254–14265.
[77] Z. Luo et al., "ASLFeat: Learning local features of accurate shape and localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6588–6597.
[78] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 337–33712.
[79] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4937–4946.
[80] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "LightGlue: Local feature matching at light speed," 2023, *arXiv:2306.13643*.
[81] 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.*, 2019, pp. 8390–8399.
[82] H. Liu, J. Feng, M. Qi, J. Jiang, and S. Yan, "End-to-end comparative attention networks for person re-identification," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3492–3506, Jul. 2017.
[83] N. N. Vo and J. Hays, "Localizing and orienting street views using overhead imagery," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 494–509.
[84] 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.*, 2015, pp. 5007–5015.
[85] Y. Tian, C. Chen, and M. Shah, "Cross-view image matching for geo-localization in urban environments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1998–2006.
[86] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.
[87] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.
[88] Z. Zheng, L. Zheng, M. Garrett, Y. Yang, M. Xu, and Y.-D. Shen, "Dual-path convolutional image-text embeddings with instance loss," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 16, no. 2, pp. 1–23, 2020.
[89] F. Radenović, G. Tolias, and O. Chum, "Fine-tuning CNN image retrieval with no human annotation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1655–1668, Jul. 2019.
[90] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1923–1938, Apr. 2019.

Qingge Li received the M.S. degree from Xi'an Shiyou University, China, in 2020. She is currently working toward the Ph.D. degree with the College of Missile Engineering, Rocket Force University of Engineering, Xi'an, China.

Her research interests include visual navigation, object detection, and image processing.

Xiaogang Yang was born in Xi'an, Shaanxi, China, in 1978. He received the Ph.D. degree in control science from the Rocket Force University of Engineering, Xi'an, China, in 2006.

He is currently a Faculty Member with the Department of Control Engineering, Rocket Force University of Engineering. He has authored 90 articles and 25 inventions. His research interests include precision guidance and image processing.



Jiwei Fan was born in Shulan, Jilin, China, in 1990. He received the Ph.D. degree from the PLA Rocket Force University of Engineering, Xi'an, China.

His research interests include image processing, machine learning, precision guidance, UAV target detection, and tracking.

Ruitao Lu received the Ph.D. degree in control science from the National University of Defense Technology, Changsha, China, in 2016.

He is currently a Faculty Member with the Department of Control Engineering, Rocket Force University of Engineering, Xi'an, China. His research interests include pattern recognition, image processing, and machine learning.

Bin Tang was born in Zhangjiajie, Hunan, China, in 2000. He is currently working toward the M.S. degree with the PLA Rocket Force University of Engineering, Xi'an, China.

His research interests include precision guidance and image processing.

Siyu Wang received the M.S. degree from Xi'an Shiyou University, Xi'an, China, in 2021. He is currently working toward the Ph.D. degree with the PLA Rocket Force University of Engineering, Xi'an, China.

His research interests include pattern recognition, image processing, and machine learning.

Shuang Su received the M.S. degree from the Northeast Electric Power University, Jilin, China, in 2020. She is currently working toward the Ph.D. degree with the PLA Rocket Force University of Engineering, Xi'an, China.

Her research interests include image processing, machine learning, precision guidance, and visual navigation.