



# MTGL40-5: A Multi-Temporal Dataset for Remote Sensing Image Geo-Localization

Jingjing Ma, Shiji Pei, Yuqun Yang, Xu Tang \* and Xiangrong Zhang

School of Artificial Intelligence, Xidian University, Xi'an 710071, China; jjma@xidian.edu.cn (J.M.); 21171213974@stu.xidian.edu.cn (S.P.); 21171110176@stu.xidian.edu.cn (Y.Y.); xrzhang@mail.xidian.edu.cn (X.Z.)

\* Correspondence: tangxu128@xidian.edu.cn

**Abstract:** Image-based geo-localization focuses on predicting the geographic information of query images by matching them with annotated images in a database. To facilitate relevant studies, researchers collect numerous images to build the datasets, which explore many challenges faced in real-world geo-localization applications, significantly improving their practicability. However, a crucial challenge that often arises is overlooked, named the cross-time challenge in this paper, i.e., if query and database images are taken from the same landmark but at different time periods, the significant difference in their image content caused by the time gap will notably increase the difficulty of image matching, consequently reducing geo-localization accuracy. The cross-time challenge has a greater negative influence on non-real-time geo-localization applications, particularly those involving a long time span between query and database images, such as satellite-view geo-localization. Furthermore, the rough geographic information (e.g., names) instead of precise coordinates provided by most existing datasets limits the geo-localization accuracy. Therefore, to solve these problems, we propose a dataset, MTGL40-5, which contains remote sensing (RS) satellite images captured from 40 large-scale geographic locations spanning five different years. These large-scale images are split to create query images and a database with landmark labels for geo-localization. By observing images from the same landmark but at different time periods, the cross-time challenge becomes more evident. Thus, MTGL40-5 supports researchers in tackling this challenge and further improving the practicability of geo-localization. Moreover, it provides additional geographic coordinate information, enabling the study of high-accuracy geo-localization. Based on the proposed MTGL40-5 dataset, many existing geo-localization methods, including state-of-the-art approaches, struggle to produce satisfactory results when facing the cross-time challenge. This highlights the importance of proposing MTGL40-5 to address the limitations of current methods in effectively solving the cross-time challenge.

**Keywords:** geo-localization; remote sensing satellite images; geographic coordinate information



**Citation:** Ma, J.; Pei, S.; Yang, Y.; Tang, X.; Zhang, X. MTGL40-5: A Multi-Temporal Dataset for Remote Sensing Image Geo-Localization. *Remote Sens.* **2023**, *15*, 4229. <https://doi.org/10.3390/rs15174229>

Academic Editors: Francesco Nex and Pablo Rodríguez-González

Received: 16 July 2023

Revised: 18 August 2023

Accepted: 25 August 2023

Published: 28 August 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Image-based geo-localization is an important topic in the field of remote sensing (RS) image processing, which focuses on using image matching technology to predict the landmark information of a given query image. In recent years, it has been applied in various real-world fields, such as agriculture [1], aircraft navigation [2], event detection [3], drone delivery [4], and so on. Generally, image-based geo-localization is regarded as a subtask of image retrieval [5–7], where similar images are retrieved from a database annotated with landmark information (e.g., name or coordinates), providing the corresponding information for the query image [8–14]. Therefore, the difficulty lies in the significant difference between query images and the annotated images within a database, caused by various perspectives, objects, environments, etc. Global positioning system (GPS), as an alternative localization technology, also finds extensive applications across various domains. In general, image-based geo-localization and GPS-based localization are complementary for providing more robust localization performance, as GPS-based localization is sometimes unstable or even

unfeasible. For example, the GPS signal may be weakened or even lost due to many factors, including signal obstruction, the multipath effect, adverse weather conditions, etc. Furthermore, since GPS-based localization technologies require an active online connection, they cannot work in offline scenarios without access to real-time GPS data. Therefore, the study of image-based geo-localization tasks is crucial in addressing these limitations and advancing the related fields.

As a fundamental component of studying image-based geo-localization, datasets play a critical role in exploring the challenges faced in real-world applications. To construct datasets, researchers collect abundant images to create query images and a database, which is mainly captured by three views from distinct altitudes: ground [15–17], unmanned aerial vehicle (UAV) [18,19], and satellite [20]. Each view corresponds to specific applications of geo-localization, such as ground for mobile robots, UAV for drone navigation, and satellite for Earth observation. At first, ground-view geo-localization is popular due to its simplicity and efficiency in capturing relevant images. Since both query and database images are captured from the same view, it can also be named ground-single-view geo-localization. To construct a dataset, numerous ground-view images annotated with landmark information are collected from many platforms [21–23], such as Google Street View photos [24]. Subsequently, the given query images can be geo-located by finding the most similar image from the database and extracting its landmark information. Therefore, the main challenge that ground-single-view datasets need to be concerned with, is that the captured images are often filled with many dynamic and changeable elements, increasing the difficulty of image matching, such as people walking and cars moving [25]. However, due to the limited variety of images involved, ground-single-view geo-localization alone is insufficient in many application scenarios. For instance, in autonomous driving that involves complex urban environments, relying solely on ground-single-view geo-localization to localize a ground-view query image may result in unsatisfactory accuracy.

With the advancements in aircraft technology, acquiring high-quality UAV and satellite RS images has become increasingly easier. Therefore, cross-view geo-localization has attracted significant attention from researchers [26,27], which focuses on the query and database images captured from different views, such as UAV ~ satellite [19,28], ground ~ satellite [29,30], and UAV ~ ground [9]. In addition to the challenges posed by dynamic elements in ground images, the use of UAVs for image capture provides more freedom, resulting in diverse angles of images. Consequently, even images captured from the same landmark can exhibit significant content variation, providing a primary challenge for UAV-related cross-view datasets, such as University-1652 [19]. On the other hand, in satellite-related cross-view datasets, the challenges faced in the image matching process mainly arise from the complex characteristics of RS images, including variations in illumination, cloud cover, resolution, and angle, among others [31,32]. In addition, in all cross-view datasets, image differences arising from the view's difference are a shared challenge, which is the primary focus of existing cross-view datasets [33,34].

As mentioned before, several public datasets have been developed to support the study of initially ground-single-view and subsequently cross-view geo-localization, and successfully explore their respective challenges. However, a common and crucial challenge faced in real-world applications is ignored, i.e., the variation in images of the same landmark due to a significant time gap, named the cross-time challenge in this paper. This notably impacts the accuracy of image matching between query images and the database, consequently decreasing the performance of geo-localization. This challenge becomes even more significant in scenarios marked by long time spans and an absence of real-time data, such as satellite-related datasets. In addition, it is worth noting that most existing geo-localization datasets only offer rough landmark information (e.g., names) instead of precise landmark coordinates, limiting the development of high-accuracy geo-localization techniques.

To address these issues, we propose a multi-temporal geo-localization dataset in satellite view, named MTGL40-5. It consists of query and database images captured by

satellite, from 40 broad geographic locations spanning five different years. By analyzing the difference between query and database images captured from the same landmark but in different years, the cross-time challenge can be explored clearly. Furthermore, MTGL40-5 also provides the latitude and longitude coordinates necessary for conducting accurate geo-localization studies. More details will be introduced in Section 3.

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

1. We propose a MTGL40-5 dataset for conducting a satellite-view geo-localization task, which consists of query images and a database featuring RS images acquired from satellites.
2. MTGL40-5 not only explores the cross-time challenge by providing images from 40 geographic locations spanning five different years, but also provides precise latitude and longitude coordinates for studying high-accuracy geo-localization tasks.
3. A simple yet effective method is proposed to measure the bias of the image center between the query image and its corresponding database image, enhancing the accuracy of the geographic coordinates.
4. Extensive experiments demonstrate that current geo-localization methods, including state-of-the-art techniques, find it difficult to generate satisfactory localization results when encountering the cross-time challenge, proving the necessity of studying it.

## 2. Related Work

In this section, we will provide a brief review about existing geo-localization studies focusing on two aspects: datasets and methods.

### 2.1. Existing Datasets for Geo-Localization

At first, ground-view-based studies gained popularity, resulting in the creation of numerous datasets that included both query images and a corresponding database. For example, Jan et al. [25] presented a dataset consisting of 17K images from Google Street View, capturing the central area of Paris, to facilitate landmark location recognition. In general, the larger the dataset volume, the higher the geo-localization performance. Therefore, a larger dataset [21] was introduced, which comprises 1.7M images annotated with landmark names, thereby providing a robust benchmark for geo-localization. In addition to the dataset's volume, diversity also plays a crucial role in ensuring its robustness. In the literature [35], a dataset was proposed that specifically aims to enhance diversity by expanding the number of landmarks. It contained more than 2.3M images of 30K landmarks. This dataset also was expanded to a larger version [36] that includes 5M images of 200K landmarks. However, the pursuit of data volume and diversity inadvertently leads to numerous noisy labels, negatively impacting the geo-localization performance. Thus, to improve the label quality, Filip et al. [37] re-labeled two datasets, Oxford5K [23] and Paris6k [22], increased the number of queries, and proposed a challenging interference set R1M. These images taken from the ground view consider the characteristics of landmarks from different angles, seasons, and weather conditions, thereby further enriching the ground-view geo-localization datasets.

Recently, the practicality of cross-view geo-localization has attracted more and more attention, so many relevant datasets have been proposed. For those involving UAVs, due to the high freedom of UAVs, it is important to capture diverse UAV images when constructing a dataset. For instance, a comprehensive dataset, called Eichenau [38], comprised satellite images and UAV images for cross-view geo-localization, where UAV images were collected from an open terrain at a high flight altitude of 100 m, with a resolution of  $573 \times 794$  pixels. Furthermore, Li et al. collected a total of 2K images using DJI Phantom 4, a UAV device, covering an extensive area of  $1.8M \text{ m}^2$  [18]. During the collection process, the flight altitude also was intentionally increased to ensure broader coverage and superior image quality. However, while the higher collection altitude ensures greater stability of the image content, it limits the angle diversity of the images. This is because the increased camera altitude leads to a longer flight distance for the UAV to capture multi-angle images of the same landmark,

consequently reducing the flexibility of adjusting the camera angle. Therefore, in the Earth Observation Center (EOC) dataset [38], multi-angle UAV images were collected from a low altitude of 340 m above sea level, which includes not only the typical top-down angles but also various side angles of buildings. To further enhance angle diversity, Zheng et al. presented an extensive collection of UAV images known as the University-1652 dataset [19], which encompasses 1652 buildings across 72 universities. A spiral UAV flight pattern was employed for each building to collect 54 multi-angle images, with flight altitudes ranging from 256 m to 121.5 m. In addition, two large datasets involving ground images, CVACT [29] and CVUSA [30], were proposed for cross-view geo-localization. The training sets of both CVACT [29] and CVUSA [30] contained 35.5K image pairs from ground and satellite views. To further enlarge data volume, Vo and Hays [39] performed a similar data collection procedure on 11 different cities in the United States and produced more than 1M image pairs, which have been widely used in cross-view geo-localization.

## 2.2. Existing Methods for Geo-Localization

For geo-localization methods, the representative features extracted from images play a crucial role in matching query and database images. Initially, traditional hand-crafted features, such as scale-invariant feature transform (SIFT) [40], local binary pattern (LBP) [41], and histogram of oriented gradients (HOGs) [42], were popular due to their easy implementation and high efficiency. For instance, in ground-view geo-localization, Torii et al. [10] first extracted SIFT features from images and further computed the vector of locally aggregated descriptors (VLADs) feature. Subsequently, the similarities between query and database images were calculated using the normalized dot product. However, due to the limited capacity of these features to accurately represent images, they were hardly satisfactory when handling poor-quality (e.g., heavily occluded or obscured) images. To alleviate this problem, an efficient bag of visual words (BOVWs), which has a large vocabulary and fast spatial matching capacity [25], was employed to detect and remove these bad images from databases. By removing poor-quality images, the accuracy of image matching was improved significantly, thereby enhancing the performance of ground-view geo-localization. Similarly, in UAV-view geo-localization, Zhuo et al. [38] found that using SIFT features alone for image matching cannot achieve a satisfactory result, especially when facing the rotational invariance challenges posed by UAV images. Therefore, building upon the generated SIFT features, the simple linear iterative clustering (SLIC) algorithm was employed to further generate dense and compact superpixels. In this way, more keypoints were provided to improve the accuracy of image matching.

While the traditional hand-crafted features have achieved some success, their limited capacity to represent images hinders their effectiveness in conducting geo-localization of complex application scenarios, since they mainly rely on low-level information such as texture and edge information. In recent years, with the advancements in deep learning, particularly convolutional neural networks (CNNs), deep features have emerged as powerful representations for images [43–45] by capturing both low-level and high-level information simultaneously. As a result, deep features have gained significant attention in the field of geo-localization. For instance, in ground-view geo-localization, Gordo et al. [46] proposed a three-stream Siamese network that explicitly optimized the weights of regional maximum activations of convolution (R-MAC) representations using a triplet ranking loss. However, this method overlooks the geometric relationship between ground images, thereby limiting its performance. To solve this problem, CVNET [47] established this geometric relationship and learned different geometric matching patterns from many image pairs. In addition, deep features also were used in cross-view geo-localization, yielding promising results. In this task, the main challenge arises from the variations in views, which leads to significant content diversity within the same landmark. To alleviate this issue, researchers employed CNNs to establish spatial hierarchical relationships between different views [48,49], such as ground ~ satellite [50] and UAV ~ satellite [27]. In this way, a comprehensive understanding of the relationship between different views is constructed, which alleviates the



challenges posed by view diversity. Additionally, as a frequently used technology for improving feature discrimination, the attention mechanism is often employed to selectively extract crucial features from images [4,51], thereby enhancing the performance of cross-view geo-localization models.

### 3. MTGL40-5 Dataset

#### 3.1. Dataset Description

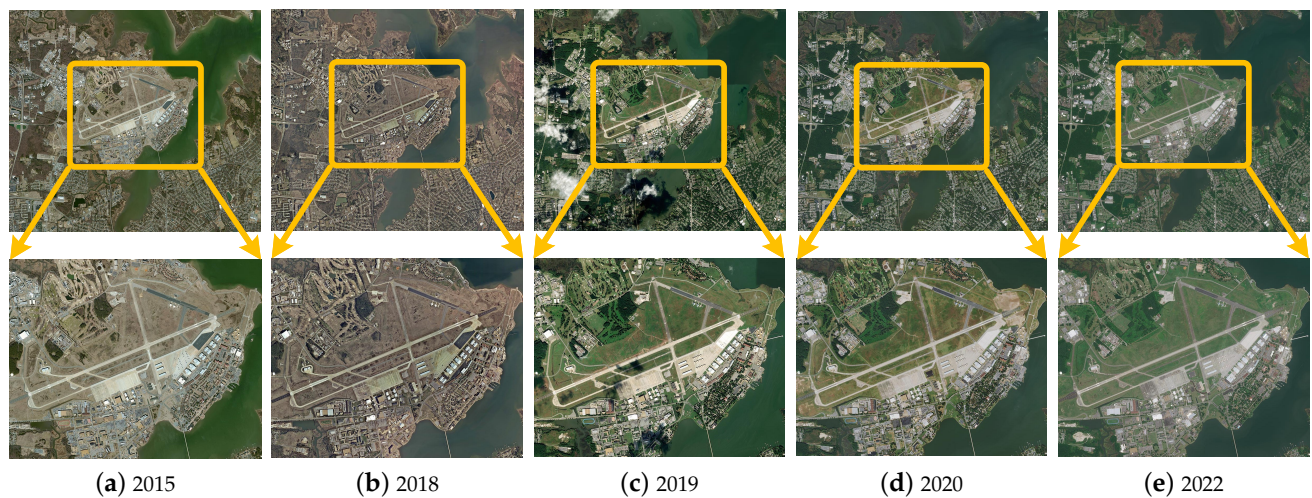
Based on Google Earth (Google Inc., Mountain View, CA, USA), we create a comprehensive dataset for studying the cross-time satellite-view geo-localization, where images are collected from 40 broad geographic locations spanning five different years. The selected locations primarily include large ports and airports due to their broad coverage areas, distinctive structures, and terrain features making them ideal for geo-localization applications. For each location, we collect large-scale original and key images, both labeled with coordinates, which have a spatial resolution of 0.5 m and span over five different years. Note that, the selected five years are not consistently continuous, but are within the time span ranging from 2014 to 2022. The original image captures the complete content of the location, while the key image focuses on the key region (as depicted in the enlarged part of Figure 1). Each original large-scale image contains approximately  $20K \times 20K$  pixels and covers an area of roughly  $80 \text{ km}^2$ , while the key image occupies approximately  $1/4$  of the original image. Subsequently, the original/key images are used to construct the database/query images annotated with geographic coordinates.

The detailed process for constructing this dataset involves the following three steps:

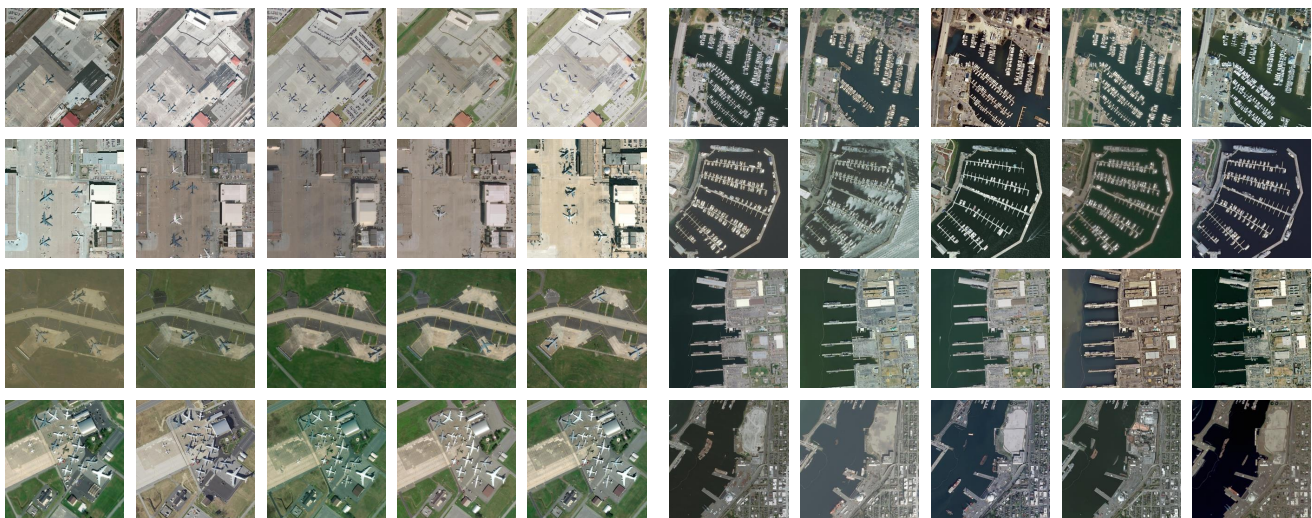
1. Collecting large-scale images: Large-scale images are collected by first searching Wikipedia for numerous ports and airports located in different countries, and their corresponding latitude and longitude coordinates are recorded. Secondly, the historical image data of the selected locations on Google Maps are reviewed to ensure that images spanning at least five years can be obtained. Thirdly, 40 target locations are selected which satisfy this criterion. Finally, based on Google Maps, the images of these locations and their coordinates are collected.
2. Splitting large-scale images: Due to the limited memory space of the computation device, each of the collected large-scale original and key images needs to be split into numerous  $N \times N$  patch images ( $N$  being a hyper-parameter specified in Section 4), which also significantly improves the efficiency and availability of geo-localization. Subsequently, all patch images are further resized to the image size of  $256 \times 256$  pixels to meet the input size of the model.
3. Dividing query images and database: The split patch images from key image are used as query images, while images from original image are used to create the database.

We randomly select a location as the sample and show its five images captured from different years in Figure 1, and display some split patch images in Figure 2.

To further show the characteristics of our dataset, we compare it with the existing six datasets University-1652 [19], CVUSA [30], CVACT [29], Lin et al. [14], Tian et al. [9], and Vo et al. [39] from seven aspects: “training set size”, “target”, “multi-temporal”, “large-scale”, “GPS coordinates”, “multi-angle”, and “evaluation” in Table 1. Here, “training set size” is the total number of training images, “target” denotes the main target of interest, “multi-temporal” means whether multiple temporal images of the same region are collected, “large-scale” represents whether large-scale images and their labels are included, “GPS coordinates” notes whether GPS coordinates are offered, “multi-angle” indicates whether multiple angle images of the same region are provided, and “evaluation” is the assessment metric employed within the dataset. Compared to University-1652 [19] and Tian et al. [9], our dataset does not include multi-angle images. This is because capturing multi-angle images in satellite view poses significant challenges, primarily due to the very long distance between the satellite and the targets within the landmarks. Despite this limitation, the proposed MTGL40-5 dataset offers three distinct advantages as follows:



**Figure 1.** Satellite-view RS image display of airports, where original images and key images are shown in the first and second rows, respectively. The original images were collected by Google Maps, and are available from five distinct years. These images contain the complete content of the location, with an image size of  $18,735 \times 16,523$ . The key images contain the key region of the original images, with an image size of  $9500 \times 7500$ . The GPS coordinates of the center point of the original images are  $(37.0806352, -76.3578087)$ .



**Figure 2.** Split patch image display. Each group consists of five RS image patches, displaying a small area's images from five different years. We can find that the position of the aircraft in the patch, the color features of the landform, and the position of the ships in the same port also change with time, which means that in a multi-temporal dataset, capturing the features of different phases of the same location has a significant impact on the geographic localization research.

- **Multi-temporal:** MTGL40-5 is a dataset consisting of multiple geographic locations, where each location includes images captured in different years, enabling the observation of dynamic changes in various landmarks over time. When training models use this dataset, many image pairs taken from the same landmark but in different years allow models to learn about the time-invariant features.
- **Large-scale:** MTGL40-5 comprises RS images acquired at a spatial resolution of 0.5 m, with each original image covering an area of around  $80 \text{ km}^2$  and containing approximately  $20K \times 20K$  pixels. Large-scale RS images can provide a wider field of view and can cover larger geographic areas. This is valuable for geo-localization research.

- Accurate labels: MTGL40-5 records the latitude and longitude coordinates for the four vertices and center point of each RS image in the database, which is important for performing accurate geo-localization.

**Table 1.** Comparison between MTGL40-5 and other geo-localization datasets.

Datasets	MTGL40-5	University-1652 [19]	CVUSA [30]	CVACT [29]	Lin et al. [14]	Tian et al. [9]	Vo et al. [39]
Training set size	66k/17k/4.7k	50k	71k	71k	75k	31.4k	1800k
Target	Ports and airports	Building	User	User	Building	Building	User
Multi-temporal	✓	×	×	×	×	×	×
Large-scale	✓	×	×	×	×	×	×
GPS coordinates	✓	×	×	×	×	✓	×
Multi-angle	×	✓	×	×	×	✓	×
Evaluation	Recall@K	Recall@K and AP	Recall@K	Recall@K	PR curves and AP	PR curves and AP	Recall@K

Finally, an important usage notice of MTGL40-5 needs to be explained: training the model with cross-time data (the query and database images are from different years) rather than same-time data. To train a model for satellite-view geo-localization, it is important to ensure that the feature distance between the query and database images from the same landmark is reduced, but not to the extent of reaching zero (i.e., overfitting). However, when the query and database images come from the same time, they may have identical pixels, meaning that all pixel values are exactly equal. This can potentially lead to overfitting of the model. Therefore, in this paper, we select the cross-time data of query and database images to optimize the feature distance during model training. Once the model is trained, the database image with the minimum feature distance from the given query image is used to produce the predicted location result.

### 3.2. Additional Tasks

In this section, we introduce three additional tasks for MTGL40-5: (1) correcting the predicted coordinates of query images; (2) predicting the geographic coordinates of key images; and (3) determining which original image the key image belongs to.

(1) Existing geo-localization models can utilize image matching technology to predict the geographic coordinates of query images. However, while the corresponding database image can be successfully matched, it may not perfectly align with the query image and could exhibit some slight position offset (see Figure 3), providing imprecise location information. Therefore, it is essential to correct this result by computing the positional offset, which is a crucial step in accurately determining the center coordinates of the query image. In addition, since the query images are split from the key image, the geographic coordinates of the corresponding key image also can be predicted by analyzing the coordinates of all query images. The total of six steps, including correcting the predicted coordinates of query images and predicting key image's coordinates, are shown in Figure 4. After obtaining the matched image pairs (① and ② in Figure 3) through the geo-localization models, the SIFT features are extracted to generate numerous keypoint pairs, each of which can produce an offset. Then, the final offset between the query and database images can be determined by voting for the keypoint pair offsets and selecting the most voted one, as shown in ③ of Figure 4. In this way, the geographic coordinates of the query images can be corrected by adding the final offset.

(2) For each query image's coordinate, the coordinate of the key image can be generated by matching it to the corresponding original position on the key image. Therefore, if a key image is split into  $N$  query images,  $N$  coordinates of the key image can be generated, as shown in ④ of Figure 4. We map these coordinates to a two-dimensional coordinate system, and analyze the corresponding density (⑤ and ⑥ of Figure 4). Averaging  $N$  coordinate points could be susceptible to the influence of incorrect data points among them. Consequently, utilizing the set of  $N$  coordinate points, we select the point with the highest density as the final coordinate point for the key image.



(3) Since the database images are extracted from the original image, each database image can be assigned an index corresponding to its original image. Therefore, when we obtain image pairs of query and database images,  $N$  query images extracted from a key image will correspond to  $N$  original image indexes. Finally, the key image can be assigned an original image index through a voting mechanism, determining which original image it belongs to.

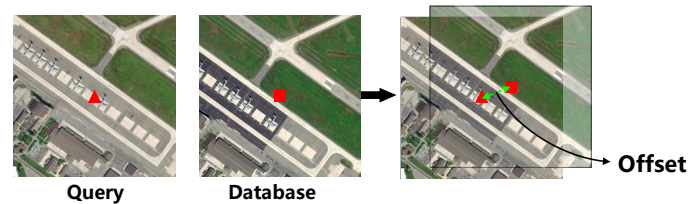


Figure 3. “Offset” represents the center point offset between query and database images.

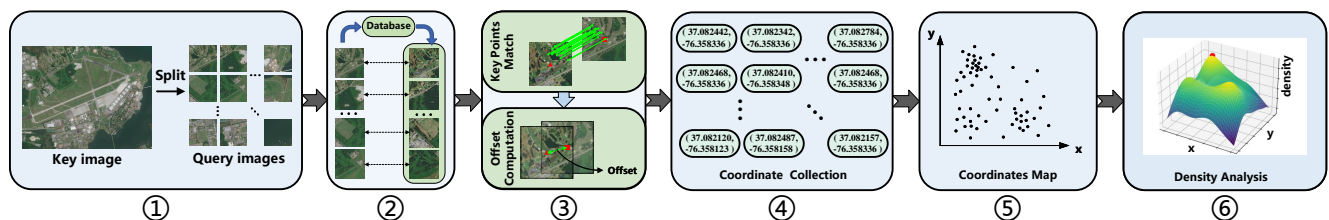


Figure 4. Calculating the coordinates of the key image's center point. ①: Splitting key image to query images. ②: Matching query images with database images. ③: Computing image center offset to correct the predicted coordinates of query images. ④: Collecting the key image's coordinate predictions based on query images. ⑤: Mapping the collected coordinates to the two-dimensional coordinate system. ⑥: Analyzing the density information, and selecting the point of highest density (red dot) as the key image's coordinates.

## 4. Experiment

### 4.1. Experimental Setting

To construct the MTGL40-5 benchmark, we select eight methods, including four widely used feature-extraction methods, VGG16 [52], GoogLeNet [53], ResNet50 [54], and DenseNet121 [55], as well as four geo-localization methods, LCM [56], LPN [49], RK-Net [4], and FSRA [1]. For a fair comparison, the data processing and experimental settings of all geo-localization methods are consistent with those in the original papers. Additionally, all feature-extraction methods adopt the following common settings: (1) model weights pre-trained on the ImageNet dataset [57] are adopted; (2) the Adam [58] optimizer with a learning rate of 0.0001 is employed; (3) the image size for inputting to the models is adjusted to  $256 \times 256$  pixels; (4) the training process consists of 200 epochs, with a batch size of 16; (5) simple data augmentation techniques are applied during training, such as random horizontal flipping and random rotation; and (6) during the testing stage, we utilize the trained models to extract image features, compute their similarity scores using the cosine distance, and generate the final results by sorting the similarity scores. Furthermore, the patch size  $N$  is set to a high value of 2048 for preserving more stable terrain information, which is crucial for accurate geo-localization.

As mentioned in Section 3, to avoid overfitting the query and database images must be selected from different years. For instance, we provide some query images captured in 2014, and then match them with database images collected in 2015. Therefore, we construct the dataset split of training, validation, and testing sets by matching different years. To better explain this, the five years of each location are uniformly re-labeled as 0–4 in ascending order. The corresponding data split is listed in Table 2. This allows us to accurately evaluate the performance change of existing methods when facing the cross-time challenge.



**Table 2.** Split of dataset.

Data Partitioning	Temporal Partitioning
Training set	query:0→database:1 query:1→database:2 query:2→database:0
Validation set	query:4→database:3
Testing set	query:3→database:4

#### 4.2. Evaluation Protocol

(1) Image matching accuracy: In this paper, we rank the image matching process of geo-localization models using the recall at the top- $k$  accuracy of query images as a metric, named Recall@ $k$ -Query (R@ $k$ -Q). Similarly, the accuracy of matching the key and original images is also evaluated through recall (see the third task of Section 3.2), named Recall@ $k$ -Key (R@ $k$ -K) to ensure the accuracy of geo-localization, where  $k$  is set to 1.

(2) GPS coordinates average error: For results involving coordinates, we use the average error between the calculated GPS coordinates and label information as the standard to measure whether the results are accurate or not.

#### 4.3. Comparison Algorithms

Below we will briefly introduce the relevant comparison algorithms:

VGG16 [52]: A deep convolutional neural network with 16 convolutional layers extracts image features with a smaller convolution kernel size and deeper network structure.

GoogLeNet [53]: A convolutional neural network using the inception module, which performs multi-scale convolution operations at the same time, and introduces an auxiliary classifier to strengthen gradient propagation and prevent gradient disappearance.

ResNet50 [54]: A deep residual network by introducing residual connections and skip connections, which solves the problem of gradient disappearance and gradient explosion in deep networks.

DenseNet121 [55]: A densely connected network that enables global sharing of features and information flow by cascade-connecting the output of the previous layer with the input of the current layer at each level.

LCM [56]: A deep network that takes into account the imbalance in the number of input samples, and considers the effect of feature vector size on matching accuracy.

LPN [49]: A deep network with a square-ring feature partition strategy, which provides attention based on the distance from the center of the image, enabling part-wise representation learning.

RK-Net [4]: A deep network that introduces a unit subtractive attention module to automatically learn representative keypoints from images.

FSRA [1]: A feature segmentation and region alignment are introduced in the transformer to enhance the model to understand the context information of the image.

#### 4.4. Experimental Results and Comparisons

##### 4.4.1. Results on MTGL40-5

Table 3 presents the overall experimental results of eight baseline models on the testing set (query:3→database:4). To our surprise, DenseNet121 outperforms other methods, including those specifically designed for geo-localization, achieving query/key accuracies of 55.76%/87.50%. Furthermore, the widely used methods demonstrate the more promising accuracies in R@1-K, more than 85%. In contrast, the specifically designed methods Rk-Net and FSRA exhibit the lowest accuracy for query and key image matching, respectively. This means that existing geo-localization methods may not be suitable when facing the cross-time challenge and may even perform worse.

**Table 3.** Comparison studies on MTGL40-5.

Method	Testing Set	
	R@1-Q (%)	R@1-K (%)
VGG16	46.92	85.00
GoogLeNet	51.41	85.00
ResNet50	54.94	85.00
DenseNet121	<b>55.76</b>	<b>87.50</b>
LCM	53.96	80.00
LPN	55.50	85.00
Rk-Net	46.91	80.00
FSRA	49.92	77.50

Although the widely used methods have certain advantages over the specifically designed methods, their low accuracy (less than 60%) in the R@1-Q metric indicates that there is still significant room for improvement in accurately extracting multi-temporal features of the same landmark. This highlights the necessity of proposing MTGL40-5 to explore the cross-time challenge.

#### 4.4.2. Results on Facing Cross-Time Challenge

To further analyze the model performance when facing the cross-time challenge, we select the top-performing model, DenseNet121, as a sample and list the image matching accuracy of different phases in Table 4, including training, validation, and testing sets.

For the training set, all three phases exhibit excellent accuracy in both R@1-Q and R@1-K, with particular results in R@1-K reaching close to 100%. It is worth noting that the R@1-Q and R@1-K scores of phase “query:2→database:0” (referred to as 2→0) are the lowest. This is because the 2→0 phase has a larger time gap between the query and database images compared to the 0→1 and 1→2 phases, which introduces more serious cross-time challenges. Although the accuracy on the training set is promising, the actual performance on the validation set shows a significant decline. Compared to the three phases of the training set, the R@1-Q/R@1-K of the validation set decrease 23.34%/7.5% (over 0→1), 23.46%/5% (over 1→2), and 20.70%/2.5% (over 2→0). This proves that it is hard for DenseNet121 to accurately capture time-invariant features to alleviate the cross-time challenge, resulting in overfitting on the training set. In the testing set, the lowest results are achieved among all phases, with only 55.76% accuracy in R@1-Q and 87.50% accuracy in R@1-K, which indicates that DenseNet121 faces difficulties in obtaining satisfactory results in practical geo-localization applications when encountering cross-time challenges.

**Table 4.** Cross-time results on DenseNet121.

DenseNet121		R@1-Q (%)	R@1-K (%)
Training set	query:0→database:1	82.33	100
	query:1→database:2	82.45	97.50
	query:2→database:0	79.69	95.00
Validation set	query:4→database:3	58.99	92.50
Testing set	query:3→database:4	55.76	87.50

#### 4.4.3. Accuracy by Location

In addition to the overall accuracy of the testing set, we also provide the accuracy of each location for all eight methods in Table 5. The accuracy of each location is calculated by averaging the matching accuracy of query images from the corresponding location. According to the results, for clarity, we use the symbols ▲ and ★ to represent simple and hard locations, respectively. It is observed that the majority of methods achieve promising accuracies in the simple locations, denoted by ★, with some even reaching a maximum

value of over 95%. In contrast, for hard locations (▲), most methods cannot accurately conduct image matching between database and corresponding query images. The changes in accuracy between hard and simple locations may potentially be influenced by the severity of the cross-time challenge, which refers to the degree of land cover changes over long periods of time. Although the rest of the general locations have a moderate accuracy, they have significant room for accuracy improvement.

**Table 5.** Accuracy by location on the MTGL40-5 dataset. ▲ and ★ denote the hard and simple locations, respectively.

Location Indexes	Testing Set							
	VGG16	GoogLeNet	ResNet50	DenseNet121	LCM	LPN	Rk-Net	FSRA
0	28.00	28.00	48.00	44.00	<b>56.00</b>	<b>56.00</b>	32.00	36.00
1 ▲	32.65	2.04	2.04	<b>55.10</b>	36.73	2.04	2.04	22.45
2 ▲	23.33	3.33	3.33	3.33	<b>56.67</b>	40.00	3.33	3.33
3 ★	95.83	<b>100.00</b>	95.83	87.50	83.33	95.83	91.67	95.83
4 ★	91.84	<b>97.96</b>	89.80	<b>97.96</b>	87.76	89.80	87.76	95.92
5	51.79	53.57	58.93	66.07	<b>71.43</b>	66.07	64.29	58.93
6	75.00	83.33	<b>87.50</b>	85.42	<b>87.50</b>	<b>87.50</b>	70.83	85.42
7	66.67	83.33	78.57	78.57	61.90	69.05	83.33	<b>88.10</b>
8	47.22	52.78	<b>63.89</b>	58.33	36.11	52.78	58.33	<b>63.89</b>
9	27.78	36.11	38.89	36.11	27.78	41.67	30.56	<b>50.00</b>
10	62.50	81.25	<b>83.33</b>	77.08	66.67	79.17	64.58	64.58
11	36.00	36.00	56.00	48.00	52.00	48.00	32.00	<b>72.00</b>
12 ▲	4.00	24.00	4.00	4.00	<b>52.00</b>	28.00	4.00	24.00
13	23.33	36.67	53.33	40.00	<b>66.67</b>	46.67	40.00	43.33
14	2.78	63.89	61.11	<b>72.22</b>	63.89	66.67	50.00	66.67
15 ★	69.44	77.78	<b>91.67</b>	83.33	83.33	<b>91.67</b>	69.44	88.89
16	50.00	75.00	<b>87.50</b>	50.00	75.00	75.00	68.75	68.75
17	50.00	50.00	50.00	40.00	5.00	<b>55.00</b>	45.00	35.00
18 ▲	<b>25.40</b>	1.59	1.59	1.59	1.59	1.59	22.22	1.59
19 ▲	28.57	2.38	40.48	38.10	2.38	<b>42.86</b>	30.95	2.38
20 ▲	2.08	2.08	2.08	2.08	2.08	<b>27.08</b>	2.08	2.08
21	43.75	50.00	52.08	75.00	52.08	<b>81.25</b>	60.42	70.83
22	<b>90.00</b>	70.00	70.00	75.00	50.00	80.00	45.00	10.00
23	4.00	<b>60.00</b>	4.00	40.00	<b>60.00</b>	24.00	28.00	4.00
24 ★	80.56	91.67	91.67	<b>94.44</b>	83.33	88.89	83.33	<b>94.44</b>
25 ▲	30.61	2.04	18.37	26.53	<b>34.69</b>	2.04	2.04	2.04
26 ★	80.00	<b>97.14</b>	85.71	80.00	88.57	88.57	85.71	85.71
27 ★	83.33	93.33	93.33	86.67	90.00	<b>96.67</b>	73.33	63.33
28	4.00	36.00	44.00	48.00	<b>76.00</b>	60.00	28.00	4.00
29 ★	<b>100.00</b>	<b>100.00</b>	<b>100.00</b>	97.96	97.96	97.96	97.96	<b>100.00</b>
30 ★	60.71	87.50	64.29	76.79	<b>96.43</b>	85.71	26.79	85.71
31	48.00	<b>88.00</b>	48.00	48.00	4.00	76.00	72.00	52.00
32	40.00	23.33	33.33	43.33	30.00	16.67	36.67	<b>46.67</b>
33	<b>68.75</b>	68.75	70.83	66.67	66.67	64.58	41.67	56.25
34 ★	83.33	75.00	77.78	75.00	<b>97.22</b>	94.44	69.44	66.67
35	50.00	62.50	50.00	62.50	37.50	50.00	50.00	<b>68.75</b>
36 ▲	28.00	4.00	<b>36.00</b>	28.00	4.00	4.00	4.00	<b>36.00</b>
37 ▲	5.00	5.00	5.00	5.00	<b>45.00</b>	5.00	5.00	5.00
38 ★	62.86	91.43	94.29	94.29	77.14	<b>97.14</b>	91.43	82.86
39 ▲	20.00	3.33	3.33	3.33	<b>30.00</b>	3.33	3.33	3.33
Mean	46.92	51.41	54.94	<b>55.76</b>	53.96	55.50	46.91	49.92

#### 4.5. Parametric Analysis of $N$ and GPS Coordinates Calculation




As mentioned above, patch size  $N$  is a hyper-parameter that determines the balance between global and local information in the query and database images. A larger  $N$

captures more global information, such as the overall terrain features such as rivers and roads. On the other hand, a smaller  $N$  focuses on capturing local information, including more details of land cover. Therefore, to study the impact of patch size  $N$  on geo-localization performance, we change  $N$  from 1024 to 4096 (specifically, 1024, 2048, and 4096). Based on different  $N$ , the image numbers of training, validation, and testing sets are listed in Table 6. Furthermore, three metrics of eight comparison methods in Table 7, including R@1-Q, R@1-K, and average error in predicting coordinates. Based on the results, as the value of  $N$  increases, there is a significant improvement in R@1-Q/K for most methods, but at the same time, the average error also increases. This means that increasing the utilization of global information may lead to higher accuracy in image matching, while incorporating more local information enhances the accuracy of coordinate prediction. Therefore, the value of  $N$  can be adjusted to meet different actual demands in various application scenarios. In this paper, we have chosen to set  $N$  to 2048 for balancing both image matching and coordinate prediction accuracies.

**Table 6.** The number of images generated by different patch sizes  $N$ .

Dataset	Number of Patch-Wise Images		
	$N = 1024$	$N = 2048$	$N = 4096$
For training	65,991	17,283	4713
For validation	27,257	7177	1987
For testing	27,257	7177	1987

**Table 7.** Parameter analysis of patch size  $N$ . For each average error, the first and second lines denote the longitude and latitude errors, respectively.

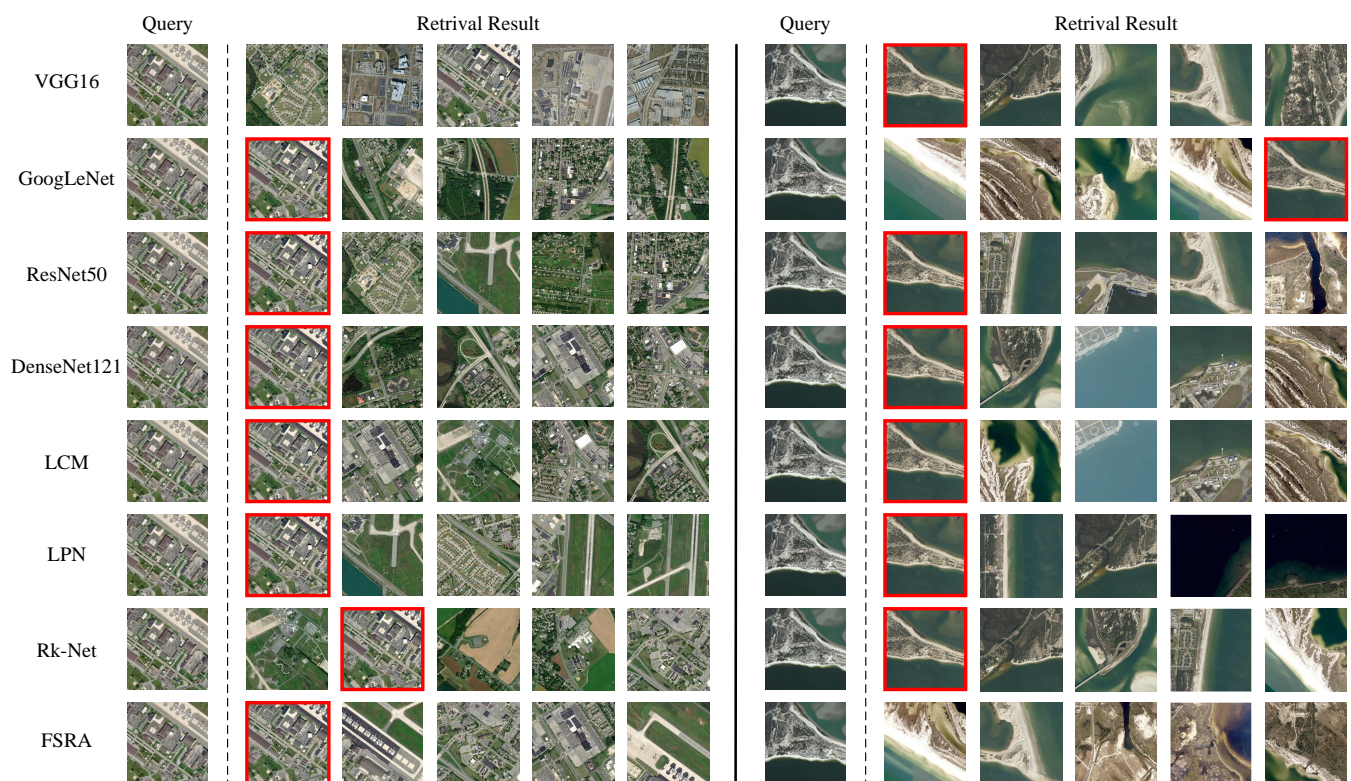
$N$		1024			2048			4096		
Two Samples										
Methods	Evaluation	R@1-Q	R@1-K	Average Error ( $10^{-3}$ ) ↓	R@1-Q	R@1-K	Average Error ( $10^{-3}$ ) ↓	R@1-Q	R@1-K	Average Error ( $10^{-3}$ ) ↓
VGG16		34.94	60.00	3.2991 3.6624	46.92	85.00	1.9368 4.5134	43.58	67.50	7.0392 6.8489
GoogLeNet		44.58	77.50	2.8284 4.0246	51.41	85.00	<b>0.4801</b> 4.5509	59.86	87.50	6.4907 5.7409
ResNet50		38.38	65.00	2.5326 <b>3.2292</b>	54.94	85.00	5.3444 <b>4.2823</b>	65.4	<b>97.50</b>	6.2648 7.4274
DenseNet121		<b>45.05</b>	<b>77.50</b>	1.5646 4.3762	<b>55.76</b>	<b>87.50</b>	2.1003 4.4092	63.27	90.00	9.6459 5.6408
LCM		37.84	65.00	1.9011 4.2125	53.96	80.00	1.9416 4.9594	64.99	92.50	9.7468 5.8561
LPN		41.11	67.50	4.3029 4.5358	55.50	85.00	4.0913 4.5134	<b>66.82</b>	92.50	6.9637 6.8025
Rk-Net		30.05	62.50	<b>0.1688</b> 3.4346	46.91	80.00	2.1611 4.9053	52.07	82.50	11.1828 5.6381
FSRA		26.80	55.00	6.2576 5.3527	49.92	77.50	3.4635 4.7893	63.72	90.00	<b>6.1434</b> <b>5.3352</b>

#### 4.6. Visual Analysis

To visually illustrate the results of the eight compared methods on the MTGL40-5 dataset, we select two distinct query images as samples. For each query image, we sort the matching scores of the corresponding database images, and choose the top five images to save space. The results are shown in Figure 5. Based on the results, all methods demonstrate a certain capacity for image matching through model training. This is evident



as the left/right query images contain buildings/water, and most of the matched database images also contain buildings/water. Furthermore, most of the methods successfully match the correct top database image. However, certain methods find it difficult to match the correct database image when some database images exhibit high similarity with the query image, such as VGG16 with the left query image and FSRA with the right query image. This demonstrates the limited capacity of these methods in accurately conducting the geo-localization task. In summary, when faced with the cross-time challenge, existing methods struggle to obtain satisfactory results, even if the standards are relaxed to the top five images. Therefore, the introduction of the MTGL40-5 dataset is necessary, since it enables the exploration of the cross-time challenge and can aid in developing improved methods.



**Figure 5.** The top five images in ranking the query image matching results on testing set. Here, the database image with a red box represents the correct matched result.

## 5. Conclusions

In this paper, a novel dataset for cross-time geo-localization is proposed, named MTGL40-5. This dataset comprises 40 diverse geographic locations spanning five different years. For each location, the large-scale original and key images are captured, which contain the complete and key content of the corresponding location. Subsequently, the key/original images are split into query/database images for conducting geo-localization. By analyzing the content difference of the same landmark in different years, we aim to address the cross-time challenge and enhance the practicality of geo-localization. In addition to the traditional image matching task, we also perform three supplementary tasks, including correcting the coordinates of the query image, predicting the coordinates of the key image, and matching the key image with the original image. Finally, we conduct extensive experiments and construct a strong benchmark with the MTGL40-5 dataset. These findings serve as evidence for the importance and value of this dataset in advancing the field of cross-time geo-localization.

**Author Contributions:** Conceptualization, J.M.; Methodology, S.P., Y.Y. and X.T.; Writing—original draft, S.P. and Y.Y.; Writing—review & editing, X.T. and X.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was funded in part by the National Natural Science Foundation of China (Nos. 62171332, 62276197), and in part by the Fund of National Key Laboratory of Science and Technology on Remote Sensing Information and Imagery Analysis, Beijing Research Institute of Uranium Geology (No. 6142A010301).

**Data Availability Statement:** The data presented in this study are available on [https://github.com/TangXu-Group/Geo\\_localization\\_dataset\\_MTGL40-5](https://github.com/TangXu-Group/Geo_localization_dataset_MTGL40-5).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Dai, M.; Hu, J.; Zhuang, J.; Zheng, E. A Transformer-Based Feature Segmentation and Region Alignment Method For UAV-View Geo-Localization. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 4376–4389. [CrossRef]
2. Zhu, P.; Zheng, J.; Du, D.; Wen, L.; Sun, Y.; Hu, Q. Multi-drone-based single object tracking with agent sharing network. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 4058–4070. [CrossRef]
3. Deng, S.; Li, S.; Xie, K.; Song, W.; Liao, X.; Hao, A.; Qin, H. A global-local self-adaptive network for drone-view object detection. *IEEE Trans. Image Process.* **2020**, *30*, 1556–1569. [CrossRef]
4. Lin, J.; Zheng, Z.; Zhong, Z.; Luo, Z.; Li, S.; Yang, Y.; Sebe, N. Joint Representation Learning and Keypoint Detection for Cross-view Geo-localization. *IEEE Trans. Image Process.* **2022**, *31*, 3780–3792. [CrossRef] [PubMed]
5. Liu, C.; Ma, J.; Tang, X.; Liu, F.; Zhang, X.; Jiao, L. Deep hash learning for remote sensing image retrieval. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 3420–3443. [CrossRef]
6. Tang, X.; Yang, Y.; Ma, J.; Cheung, Y.M.; Liu, C.; Liu, F.; Zhang, X.; Jiao, L. Meta-hashing for remote sensing image retrieval. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5615419. [CrossRef]
7. Tang, X.; Ma, Q.; Zhang, X.; Liu, F.; Ma, J.; Jiao, L. Attention consistent network for remote sensing scene classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2030–2045. [CrossRef]
8. Arandjelovic, R.; Gronat, P.; Torii, A.; Pajdla, T.; Sivic, J. NetVLAD: CNN architecture for weakly supervised place recognition. *arXiv* **2015**, arXiv:1511.07247.
9. Tian, Y.; Chen, C.; Shah, M. Cross-View Image Matching for Geo-localization in Urban Environments. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
10. Torii, A.; Arandjelovic, R.; Sivic, J.; Okutomi, M.; Pajdla, T. 24/7 place recognition by view synthesis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1808–1817.
11. Hu, S.; Feng, M.; Nguyen, R.M.H.; Lee, G.H. CVM-Net: Cross-View Matching Network for Image-Based Ground-to-Aerial Geo-Localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
12. Rodrigues, R.; Tani, M. Global assists local: Effective aerial representations for field of view constrained image geo-localization. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2022; pp. 3871–3879.
13. Hu, W.; Zhang, Y.; Liang, Y.; Yin, Y.; Georgescu, A.; Tran, A.; Kruppa, H.; Ng, S.K.; Zimmermann, R. Beyond Geo-localization: Fine-grained Orientation of Street-view Images by Cross-view Matching with Satellite Imagery. In Proceedings of the 30th ACM International Conference on Multimedia, Lisboa, Portugal, 10–14 October 2022; pp. 6155–6164.
14. Lin, T.Y.; Cui, Y.; Belongie, S.; Hays, J. Learning deep representations for ground-to-aerial geolocalization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5007–5015.
15. Zhu, S.; Yang, T.; Chen, C. VIGOR: Cross-View Image Geo-Localization Beyond One-to-One Retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 3640–3649.
16. Lu, Z.; Pu, T.; Chen, T.; Lin, L. Content-Aware Hierarchical Representation Selection for Cross-View Geo-Localization. In Proceedings of the Asian Conference on Computer Vision (ACCV), Macao, China, 4–8 December 2022; pp. 4211–4224.
17. Toker, A.; Zhou, Q.; Maximov, M.; Leal-Taixe, L. Coming Down to Earth: Satellite-to-Street View Synthesis for Geo-Localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 6488–6497.
18. Mughal, M.H.; Khokhar, M.J.; Shahzad, M. Assisting UAV localization via deep contextual image matching. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2445–2457. [CrossRef]
19. Zheng, Z.; Wei, Y.; Yang, Y. University-1652: A multi-view multi-source benchmark for drone-based geo-localization. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 1395–1403.
20. Lu, X.; Luo, S.; Zhu, Y. It's Okay to Be Wrong: Cross-View Geo-Localization with Step-Adaptive Iterative Refinement. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 4709313. [CrossRef]

21. Chen, D.M.; Baatz, G.; Köser, K.; Tsai, S.S.; Vedantham, R.; Pylvänäinen, T.; Roimela, K.; Chen, X.; Bach, J.; Pollefeys, M.; et al. City-scale landmark identification on mobile devices. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 737–744.
22. Philbin, J.; Chum, O.; Isard, M.; Sivic, J.; Zisserman, A. Lost in quantization: Improving particular object retrieval in large scale image databases. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
23. Philbin, J.; Chum, O.; Isard, M.; Sivic, J.; Zisserman, A. Object retrieval with large vocabularies and fast spatial matching. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8.
24. Weyand, T.; Leibe, B. Visual landmark recognition from internet photo collections: A large-scale evaluation. *Comput. Vis. Image Underst.* **2015**, *135*, 1–15. [\[CrossRef\]](#)
25. Knopp, J.; Sivic, J.; Pajdla, T. Avoiding confusing features in place recognition. In Proceedings of the European Conference on Computer Vision, Heraklion, Greece, 5–11 September 2010; Springer: Berlin/Heidelberg, Germany, 2010; pp. 748–761.
26. Yang, H.; Lu, X.; Zhu, Y. Cross-view geo-localization with layer-to-layer transformer. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 29009–29020.
27. Tian, X.; Shao, J.; Ouyang, D.; Shen, H.T. Uav-satellite view synthesis for cross-view geo-localization. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 4804–4815. [\[CrossRef\]](#)
28. Dai, M.; Huang, J.; Zhuang, J.; Lan, W.; Cai, Y.; Zheng, E. Vision-Based UAV Localization System in Denial Environments. *arXiv* **2022**, arXiv:2201.09201.
29. Liu, L.; Li, H. Lending orientation to neural networks for cross-view geo-localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5624–5633.
30. Workman, S.; Souvenir, R.; Jacobs, N. Wide-area image geolocalization with aerial reference imagery. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3961–3969.
31. Xia, G.S.; Hu, J.; Hu, F.; Shi, B.; Bai, X.; Zhong, Y.; Zhang, L.; Lu, X. AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3965–3981. [\[CrossRef\]](#)
32. Cheng, G.; Han, J.; Lu, X. Remote sensing image scene classification: Benchmark and state of the art. *Proc. IEEE* **2017**, *105*, 1865–1883. [\[CrossRef\]](#)
33. Zhuang, J.; Dai, M.; Chen, X.; Zheng, E. A Faster and More Effective Cross-View Matching Method of UAV and Satellite Images for UAV Geolocalization. *Remote Sens.* **2021**, *13*, 3979. [\[CrossRef\]](#)
34. Guo, Y.; Choi, M.; Li, K.; Boussaid, F.; Bennamoun, M. Soft Exemplar Highlighting for Cross-View Image-Based Geo-Localization. *IEEE Trans. Image Process.* **2022**, *31*, 2094–2105. [\[CrossRef\]](#)
35. Noh, H.; Araujo, A.; Sim, J.; Weyand, T.; Han, B. Large-scale image retrieval with attentive deep local features. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3456–3465.
36. Weyand, T.; Araujo, A.; Cao, B.; Sim, J. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2575–2584.
37. Radenović, F.; Iscen, A.; Tolias, G.; Avrithis, Y.; Chum, O. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5706–5715.
38. Zhuo, X.; Koch, T.; Kurz, F.; Fraundorfer, F.; Reinartz, P. Automatic UAV image geo-registration by matching UAV images to georeferenced image data. *Remote Sens.* **2017**, *9*, 376. [\[CrossRef\]](#)
39. Vo, N.N.; Hays, J. Localizing and orienting street views using overhead imagery. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 494–509.
40. Yang, Y.; Newsam, S. Comparing SIFT descriptors and Gabor texture features for classification of remote sensed imagery. In Proceedings of the 2008 15th IEEE International Conference on Image Processing, San Diego, CA, USA, 12–15 October 2008; pp. 1852–1855.
41. Ren, J.; Jiang, X.; Yuan, J. Learning LBP structure by maximizing the conditional mutual information. *Pattern Recognit.* **2015**, *48*, 3180–3190. [\[CrossRef\]](#)
42. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), San Diego, CA, USA, 20–26 June 2005; Volume 1, pp. 886–893.
43. Yang, Y.; Tang, X.; Cheung, Y.m.; Zhang, X.; Jiao, L. SAGN: Semantic-Aware Graph Network for Remote Sensing Scene Classification. *IEEE Trans. Image Process.* **2023**, *32*, 1011–1025. [\[CrossRef\]](#) [\[PubMed\]](#)
44. Yang, Y.; Tang, X.; Zhang, X.; Ma, J.; Liu, F.; Jia, X.; Jiao, L. Semi-Supervised Multiscale Dynamic Graph Convolution Network for Hyperspectral Image Classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**. [\[CrossRef\]](#)
45. Tang, X.; Lin, W.; Ma, J.; Zhang, X.; Liu, F.; Jiao, L. Class-level prototype guided multiscale feature learning for remote sensing scene classification with limited labels. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–15. [\[CrossRef\]](#)



46. Gordo, A.; Almazán, J.; Revaud, J.; Larlus, D. Deep image retrieval: Learning global representations for image search. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 241–257.
47. Lee, S.; Seong, H.; Lee, S.; Kim, E. Correlation verification for image retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5374–5384.
48. Zhu, Y.; Sun, B.; Lu, X.; Jia, S. Geographic Semantic Network for Cross-View Image Geo-Localization. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–15. [[CrossRef](#)]
49. Wang, T.; Zheng, Z.; Yan, C.; Zhang, J.; Sun, Y.; Zheng, B.; Yang, Y. Each part matters: Local patterns facilitate cross-view geo-localization. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 867–879. [[CrossRef](#)]
50. Shi, Y.; Liu, L.; Yu, X.; Li, H. Spatial-aware feature aggregation for image based cross-view geo-localization. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 10090–10100.
51. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, arXiv:1706.03762.
52. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
53. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
54. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
55. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
56. Ding, L.; Zhou, J.; Meng, L.; Long, Z. A practical cross-view image matching method between UAV and satellite for UAV-based geo-localization. *Remote Sens.* **2020**, *13*, 47. [[CrossRef](#)]
57. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 248–255.
58. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.

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