Each Part Matters: Local Patterns Facilitate Cross-view Geo-localization

Tingyu Wang, Zhedong Zheng, Chenggang Yan, Jiyong Zhang, Yaoqi Sun, Bolun Zheng, and Yi Yang, *Senior Member, IEEE*

***Abstract*—Cross-view geo-localization is to spot images of the same geographic target from different platforms, *e.g.*, drone-view cameras and satellites. It is challenging in the large visual appearance changes caused by extreme viewpoint variations. Existing methods usually concentrate on mining the fine-grained feature of the geographic target in the image center, but underestimate the contextual information in neighbor areas. In this work, we argue that neighbor areas can be leveraged as auxiliary information, enriching discriminative clues for geo-localization. Specifically, we introduce a simple and effective deep neural network, called Local Pattern Network (LPN), to take advantage of contextual information in an end-to-end manner. Without using extra part estimators, LPN adopts a square-ring feature partition strategy, which provides the attention according to the distance to the image center. It eases the part matching and enables the part-wise representation learning. Owing to the square-ring partition design, the proposed LPN has good scalability to rotation variations and achieves competitive results on three prevailing benchmarks, *i.e.*, University-1652, CVUSA and CVACT. Besides, we also show the proposed LPN can be easily embedded into other frameworks to further boost performance.**

***Index Terms*—Geo-localization, Image Retrieval, Agriculture, Deep Learning.**

I. Introduction

CROSS-VIEW geo-localization is to retrieve the most relevant images from different platforms, which could be applied to many fields, such as accurate delivery, autonomous driving, robot navigation, event detection, and so on [1], [2], [3], [4]. For instance, given a drone-view image, the system intends to search images of the same location in the candidate images of the satellite. The satellite-view images are automatically annotated with geo-tags. Obtaining the true-match satellite-view image, we could localize the building in the drone-view image. Besides, the image-based cross-view geo-localization can facilitate the positioning devices, *e.g.*, GPS, to provide a more robust and accurate result.

In recent years, cross-view geo-localization has obtained a significant development due to the advance in deep learning.

Tingyu Wang, Chenggang Yan, Jiyong Zhang, Yaoqi Sun and Bolun Zheng are with the Intelligent Information Processing Lab, Hangzhou Dianzi University, Hangzhou 310018, China (e-mail: wongtyu@hdu.edu.cn; cgyan@hdu.edu.cn; jzhang@hdu.edu.cn; syq@hdu.edu.cn; blzheng@hdu.edu.cn). Chenggang Yan is the Corresponding Author.

Zhedong Zheng, Yi Yang are with the Australian Artificial Intelligence Institute, Uinversity of Technology Sydney, NSW 2007, Australia (e-mail: Zhedong.Zheng@student.uts.edu.au; Yi.Yang@uts.edu.au).

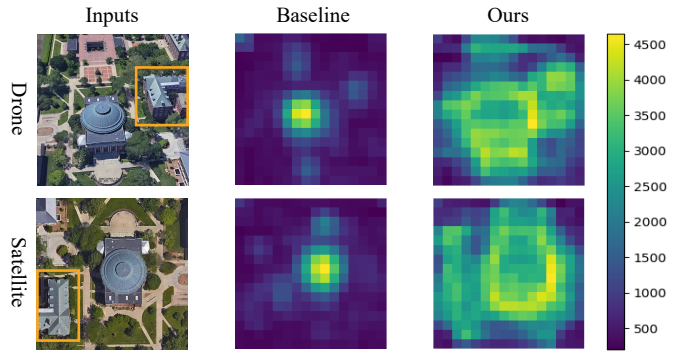

Fig. 1. Difference of the activation maps generated by the baseline method [2] and our method. The first column shows two input images from different platforms, *i.e.*, satellite and drone, with the same geo-tag. We observe that the contextual information, such as the neighbor building in the yellow box, can be used as an auxiliary clue to facilitate the cross-view image-based geographic localization. In the second column, we visualize the activation map of the baseline model [2]. We could observe that the baseline method [2] activates only the patterns at the center geographic target, while our method activates more contextual information around the center geographic target (see the third column). †: **The baseline method is a three-branch network with ResNet-50 [5] as the backbone, and the model is optimized by the instance loss [6].**

Most works [1], [4], [7], [8], [9] explore the deep neural network with the metric learning to learn the discriminative feature. Specifically, the network is to learn one feature space that brings matched image pairs closer and pushes non-matched pairs far apart [10], [11], [12]. The attention mechanism and orientation information are also widely used in the network design [1], [13], [4]. However, most existing methods only consider the global information via pooling functions, ignoring the contextual information (see Figure 1).

Generally, the aerial-view platform, *e.g.*, drone or satellite, captures the scene image with a wide angle. When the platform acquires a geographic target, the contextual information around the target is also captured as a by-product. When existing works usually ignore such information, we argue that the contextual information provides a key clue for cross-view geo-localization. For instance, when there is no apparent difference between two geographic targets, such as two straight roads, the human visual system is challenging to identify the true-match target. However, the task is much easier with the help of contextual information, *e.g.*, neighbor houses. Mining and utilizing the contextual information in the image can improve the accuracy of the cross-view geo-localization.

Our work is inspired by the procedure that the human visual

system interprets and matches the same scene of different viewpoints [14], [15], [16]. When recognizing a geography scene of two different platforms, the human visual system generally adopts a hierarchical processing manner to improve the accuracy of judgement. Specifically, the human visual system first pays attention to whether the same geographic target is contained in different viewpoint scenes. Then, the human visual system will check the contextual information around the geographic target to verify the correctness of the match. When there is no remarkable landmark, people usually resort to the map to find discriminative neighbor areas. Imitating the process mentioned above, we design a Local Pattern Network (LPN), which is an effective way to explicitly explore the contextual information in an end-to-end learning manner. Specifically, we divide the high-level feature into several parts in a square-ring partition, as shown in Figure 2. Because the geographic target is generally located in the center of the image with the contextual information surrounded. Our partition method can obtain not only the geographic target information (the region of A) but also several contextual-information parts (the region of B and C) with different distances from the geographic target. Therefore, we can explicitly exploit contextual information to optimize LPN. We also observe that our partition strategy is robust to the image rotation in nature. For instance, when rotating the left image in Figure 2 as the right image, the three regions (A, B, and C) still contain the same semantic information as corresponding regions of the left image. Therefore, the network designed according to the square-ring manner has good scalability to image rotation. To verify the effectiveness of the proposed method, we conduct experiments on three public datasets, *i.e.*, University-1652 [2], CVUSA [17] and CVACT [4]. LPN achieves the Recall@1 accuracy of 75.93% for the task of drone-view target localization (**Drone** $\rightarrow$ **Satellite**) and Recall@1 accuracy of 86.45% for the task of drone navigation (**Satellite** $\rightarrow$ **Drone**), which is higher than the baseline work [2] by 17.44% and 15.27% respectively. Similar results are also observed on CVUSA and CVACT. Compared with the baseline model [2], the Recall@1 accuracy increases from 43.91% to 79.69% (+35.78%) on CVUSA and 31.20% to 73.85% (+42.65%) on CVACT. Besides, the proposed method is complementary to most previous works. The proposed method could be easily fused with the state-of-art methods, *i.e.*, SAFA [1], and boost the performance from 89.84% Recall@1 accuracy to 92.83% (+2.99%) Recall@1 accuracy on CVUSA and 81.03% Recall@1 accuracy to 83.66% (+2.63%) Recall@1 accuracy on CVACT.

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

- We propose a simple and effective model, called Local Pattern Network (**LPN**). Different from existing works, LPN explicitly takes contextual patterns into consideration and leverages the surrounding environment around the target building. Specifically, the model deploys the square-ring partition strategy and learns contextual information in an end-to-end manner.
- We demonstrate the effectiveness of our method on three prevailing cross-view geo-localization datasets, *i.e.*, University-1652 [2], CVUSA [17] and CVACT [4]. Our method outperforms the strong baseline on both benchmarks by a large margin. Furthermore, we show that the proposed method is complementary to existing works, and can be fused with the state-the-art approaches to further boost the performance.

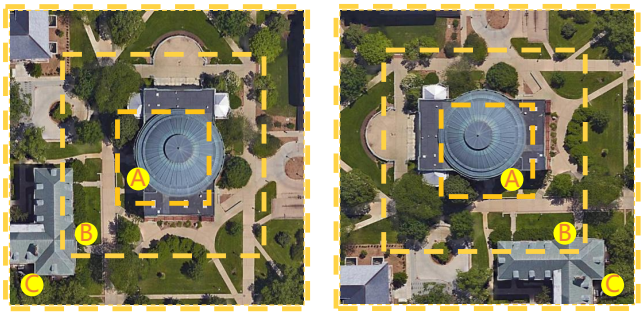

Fig. 2. The simplified diagram of our partition strategy, which is invariant to the rotation. The region of part A represents the geographic target in the center. According to the distance from the geographic target, the region of part B can be viewed as the first hierarchical contextual information, and the region of part C is the second hierarchical contextual information.

We organize the rest of this paper as follows. In Section II, we briefly introduce some of the relevant works. Section III presents our designed LPN in detail. Experimental results are presented in Section IV and followed by the conclusion in Section V.

II. Related Work

In this section, we briefly review related previous works, including deep cross-view geo-localization and part-based representation learning.

A. Deep Cross-view Geo-localization

Cross-view geo-localization has been attracting more attention in recent years due to a large number of potential applications. Some pioneering approaches [18], [19], [20], [21] focus on extracting hand-crafted features. Inspired by the great success of the deep convolutional neural networks (CNNs) on ImageNet, researchers resort to the deeply-learned feature in recent years. Workman *et al.* [3] are among the first attempts to utilize a pre-trained CNN to extract features for the cross-view localization task. They demonstrate that features from the high-level layer of CNN contain semantic information about the geographic location. To take one step further, Workman *et al.* [22] fine-tune the pre-trained network by reducing the feature distance between pairs of ground-level images and aerial images, yielding better performance. Inspired by the face verification approaches, Lin *et al.* [23] adopt a modified Siamese Network [24], which optimizes network parameters by the contrastive loss [10], [11]. Zhai *et al.* [9] plug the NetVLAD [25] into a Siamese-like architecture, making image descriptors robust against large viewpoint changes. Liu *et al.* [4] stress the importance of orientation information and

encode corresponding coordinate information into the network to boost the discrimination of the feature. In a recent work, Shi *et al.* [7] use the spatial layout information to make up the shortcoming of the global aggregation step in feature extraction. Furthermore, Shi *et al.* [1] improve the performance of cross-view geo-localization through domain alignment and spatial attention mechanism. Besides, DSM [8] considers a limited Field of View setting and adopts a dynamic similarity matching module to align the orientation of cross-view images. Another line of works considers the metric learning and designs different training objectives to learn the discriminative representation. Vo *et al.* [26] design an orientation regression loss, yielding the performance improvement. Hu *et al.* [9] employ a weighted soft margin ranking loss, which not only speeds up the training convergence but also improves the retrieval accuracy. Different from adopting metric learning loss (*i.e.*, contrastive loss [10], [11] and triplet loss [12], [27]), Zheng *et al.* [2] regard the cross-view image retrieval as a classification task. They apply the instance loss [28], [6] to optimize the network and has achieved a competitive result. However, these methods usually concentrate on exploring the global information but ignore the contextual information as shown in Figure 1. Different from existing works, the proposed method intends to take advantage of the neighbor areas. We deploy the feature-level partition strategy, which facilitates the end-to-end learning on the contextual information.

B. Part-based Representation Learning

The local feature has been widely studied in the design of hand-crafted algorithms [29], [30], [31], [32], [33]. Ojala *et al.* [34] propose a local binary pattern (LBP) descriptor to extract the rotation-invariant feature. Lowe *et al.* [35] develop a Scale Invariant Feature Transform (SIFT) descriptor for the image-based match. SIFT is invariant to translations, rotation, and scaling transformations by summarizing description of the local image structures in a local neighborhood around each interest point. In the spirit of the conventional part-based descriptor, some researchers also explore the local pattern learning in the deep-leaned models. One line of works divides the features based on an extra estimator, such as landmark detection, human pose estimated, and human parsing. Spindle Net [36] leverages the landmark points of the human body to obtain semantic features from different body regions. Xu *et al.* [37] propose a pose-guide part attention module to learn a confidence map. Guo *et al.* [38] acquire the accurate human part-aligned representation by the human parsing model to enhance the robustness of the feature. Another line of works does not need an extra pose estimator and deploys a coarse part alignment, such as horizontal matching. Li *et al.* [39] capture the three parts information corresponding to the head-shoulder, upper body, and lower body by Spatial Transformer Network (STN) [40], [41]. Zhao *et al.* [42] utilize the attention mechanism to learn aligned part information from the input image automatically. A strong Part-based Convolutional Baseline (PCB) [43] shows a uniform partition strategy to extract high-level features. Then, by correcting within-part inconsistency of all the column vectors according to their similarities to each part, the performance of this work becomes better. Currently, some state-of-the-art works [44], [45], [46], [47] extend the PCB with more partitions or optimization losses. Our work also studies a part-based representation learning on the convolutional layer, but is different in two aspects: Different from works of the first line [37], [36], [48], [49], [50], the proposed method does not need an extra part estimator. Different from works of the second line [40], [43], [44], [45], our partition method makes the network have good scalability to image rotation (see Figure 2).

III. Proposed Method

In this section, we introduce the Local Pattern Network (LPN) (see Figure 3). We first illustrate the network architecture for feature extraction, followed by the partition strategy for feature maps and the optimization objective. Finally, we provide a discussion on our intuition and special cases for different datasets.

Problem formulation. Given one geo-localization dataset, we denote the input image as x, and y represents the corresponding label. We apply the subscript j to denote the platform where the data x_j is collected, and $j \in \{1, 2, 3\}$. In particular, x_1 denotes the sample from the satellite view, x_2 denotes the drone-view data, and x_3 denotes the ground-view image. The label $y \in [1, C]$, where C indicates the number of categories. For instance, a dataset includes 701 buildings and each building contains multiple images. We number 701 buildings into 701 different indexes. Each index represents a category, *i.e.*, the label $y \in [1, 701]$. For cross-view geo-localization, we intend to learn one mapping function, which could project images from different platforms to one shared semantic space. The images of the same location are close, while the images from different location are apart from each other.

A. Local Pattern Network

Feature extraction. The proposed model, *i.e.*, Local Pattern Network (LPN), contains three branches, which extends from the Siamese network [24]. From top to bottom in Figure 3, the three branches are the satellite-view branch, the drone-view branch and the ground-view branch respectively. LPN can deploy various network architectures as backbones to extract features, such as VGG [51], and ResNet [5]. For illustration, we choose ResNet-50 [5] as the network architecture of each branch if not specified. ResNet-50 contains five blocks named conv1, conv2, conv3, conv4, conv5, one average pooling layer, and one fully connected layer. We remove the final average pooling layer and the fully connected layer, and obtain intermediate feature maps for subsequent partition processing. Following [2], we share weights between the satellite-view branch and the drone-view branch, since input images of both branches are from the aerial viewpoint. Three branches have the same feature extraction manner. Specifically, given an input image of size 256×256, we can acquire feature maps with the shape of $16 \times 16 \times 2048$ in each branch. We denote this

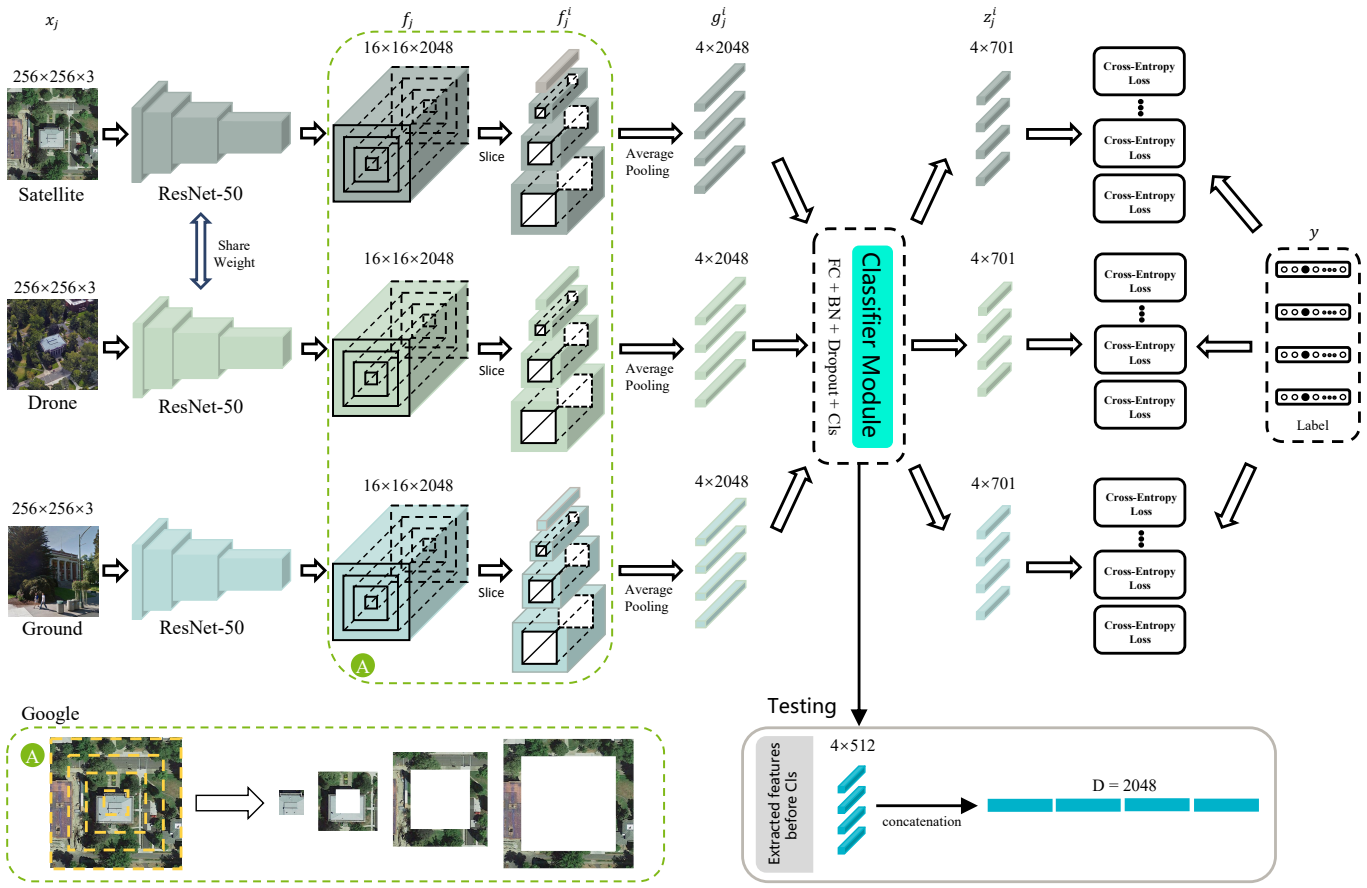

Fig. 3. Overview of the proposed LPN framework. Given one input image, we first extract feature maps. Since we study the cross-view geo-localization, the input image can be from different platforms. The proposed LPN contains three branches, *i.e.*, the satellite-view branch, the drone-view branch and the ground-view branch, respectively, to deal with different kinds of inputs. The satellite-view branch and the drone-view branch share weights since images from the satellite view and the drone view have similar patterns. Then, the output feature maps from each branch are sliced according to the square-ring partition strategy. Next, the average pooling layer is used to transform each part-level feature maps into a column feature descriptor. Finally, all these feature descriptors are fed into a classifier module to get prediction vectors. In addition to the classification layer (Cls), the classifier module also contains other three type layers, which are the fully-connected layer (FC), the batch normalization layer (BN), and the dropout layer (Dropout). During training, we leverage the classifier module to predict the geo-tag of each part. The network is optimized by minimizing the sum of the cross-entropy losses over all parts. When testing, we obtain the part-level image representation before the classification layer in the classifier module. Then we concatenate part-level features as the final visual descriptor of the input image, and the dimension of the feature is 2048. In (A) (a green dotted line), we show the square-ring partition strategy. Note that here we display the framework for the University-1652 dataset of input data from three platforms. For two-view datasets, *e.g.*, CVUSA, we use two CNN branches.

function as $\mathcal{F}_{backbone}$, and the process of feature extraction can be formulated as:

$$f_j = \mathcal{F}_{backbone}(x_j), \tag{1}$$

where f_j stands for the extracted feature map of the input image x_j.

Feature partition strategy. To explicitly take advantage of contextual information, we apply the square-ring partition strategy to divide feature maps. We observe that the geographic target is generally distributed in the center of the image, and the contextual information is radiantly distributed around. Based on this assumption of semantic information distribution, the center of the square-ring partition can be approximately aligned at the center of the feature maps. As shown in Figure 3 (A) (green box), we separate images into four parts according to the distance to the image center. In practice, we separate the global feature maps f_j to four feature parts $f_j^i (i \in \{1, 2, 3, 4\})$. The superscript i represents the i-th part from the center. Then we apply the average pooling layer to transform each part f_j^i with different shapes into a 2048-dim part feature g_j^i. The process can be formulated as:

$$f_j^i = \mathcal{F}_{slice}(f_j, i), \tag{2}$$

$$g_j^i = \mathcal{A}vgpool(f_j^i), \tag{3}$$

where $\mathcal{F}_{slice}$ indicates the square-ring partition, and $\mathcal{A}vgpool$ represents the average pooling operation.

Optimization objective. Now we have obtained part features from different sources. Since the features are extracted from different branches, they may have different distribution, which could not be directly used for matching. To solve this limitation, we set up a mapping function that maps features of all sources into one shared feature space. In this shared space, features of the same geo-tag will have a closer distance, while features of different geo-tags are apart from each others.

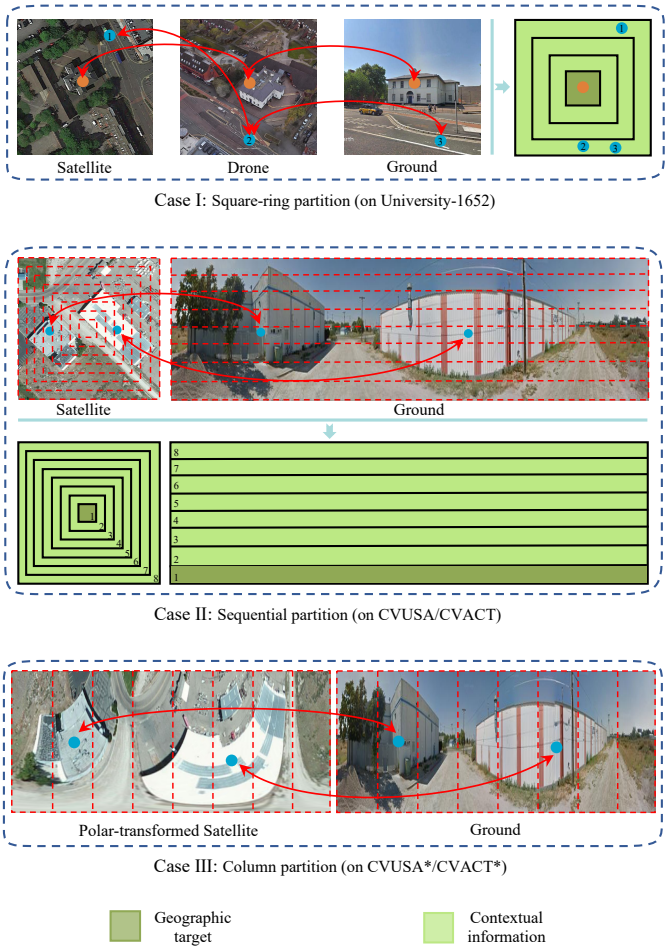

Fig. 4. Illustration of the square-ring partition (Case I), the sequential partition (Case II) and the column partition (Case III). **The sequential partition strategy and the column partition strategy are two special cases of the square-ring partition strategy.** The sequential partition considers the geometric correspondence for matching satellite-and-ground panorama image pair [1], [8]. The column partition directly splits the feature maps vertically. All three partition strategies exploit the contextual information and achieve the spatial alignment of each part. The square-ring partition strategy is suitable for processing images that the contextual information is distributed around the geographic target, such as University-1652. When localizing panoramic image and the orientation of different view images is aligned, the sequential partition has a higher priority, such as CVUSA and CVACT. The column partition enables a fine-grained spatial alignment for pre-processed image pairs on CVUSA* and CVACT* [1], [7] whose orientation and semantic information distribution are roughly aligned.

This classifier module consists of following layers: a fully connected layer (FC), a batch normalization layer (BN), a dropout layer (Dropout), and a classification layer (Cls), which is a fully-connected layer. The classifier module is deployed to predict the geo-tag of each image based on part features. Given the part features g_j^i as the input, the classifier module outputs a column vector z_j^i. The dimension of z_j^i equals the number of geo-tag categories C.

$$z_j^i = \mathcal{F}_{classifier}(g_j^i). \tag{4}$$

The cross-entropy loss could be formulated as:

$$\hat{p}(y|x_j^i) = \frac{exp(z_j^i(y))}{\sum_{c=1}^{C} exp(z_j^i(c))}, \tag{5}$$

$$Loss = \sum_{i,j} -log(\hat{p}(y|x_j^i))), \tag{6}$$

where $z_j^i(y)$ is the logit score of the ground-truth geo-tag y. We apply the softmax function (Equation 5) to obtain the normalized probability score $\hat{p}(y|x_j^i)$ in $[0, 1]$. $\hat{p}(y|x_j^i)$ is the predicted probability that x_j^i belongs to the geo-tag y. In Equation 6, we accumulate the losses on the image of different parts and different platforms to optimize the whole network.

B. Discussion

Our method is inspired by the mechanism of the human visual system on matching images of different viewpoints. In the ancient time, people compare the map and the surrounding environment to know where they are. The contextual information plays an important role. Nowadays, cameras on different platforms typically use wide-angle lenses to obtain complete geographic targets, and the contextual information around the geographic target is also collected in the image. We argue that the contextual information, as a by-product, can facilitate the discriminative representation learning. For example, the neighbor building also could help to predict the target location. Instead of dividing images in the pixel level, we split the feature maps in practice, which could not only improve the efficiency but also enable the larger receptive fields as well as the part alignment. The square-ring partition strategy is also robust to the rotation variants. Case I (see Figure 4) shows the application of the square-ring partition strategy on three images of different views in University-1652, *i.e.*, satellite view, drone view and ground view. The orientation of these three-view images is not aligned. However, we can observe that the geographic target (orange point) is generally located in the image center. Because of the random orientation of three-view images, the contextual information with the same semantics (blue point) may not be distributed in the same orientation but can be located in the same part. We note that the sequential partition strategy is a special case of the square-ring partition strategy. The sequential partition strategy takes into account the geometric correspondence [1], [8] for a north aligned satellite-and-ground panorama image pair, such as data on CVUSA [17]/CVACT [4]. **Specifically, we apply the square-ring partition to satellite images and the row partition to ground panoramas.** As shown in Case II of Figure 4, image pairs on CVUSA/CVACT have different visual appearances. But the same semantic information (*e.g.*, the blue point pair) from the true-matched image pair can still be roughly located in the same part of the divided feature maps. Besides, the column partition is also a variation of our method. The column partition can be adapted when the orientation and the spatial semantics of the matching image pair are roughly aligned. For example, images have been pre-processed by Optimal Transport theory [7] or the polar transform [1], [8]. Case III (see Figure 4) provides an example

that the column partition is applied to a polar-transformed satellite image and a ground panorama.

TABLE I
STATISTICS OF THREE DIFFERENT TEST SETS, INCLUDING THE IMAGE NUMBER OF QUERY SET AND GALLERY SET FOR DIFFERENT GEO-LOCALIZATION TASKS.

Dataset	Task			
	Drone → Satellite		Satellite → Drone	
	Query	Gallery	Query	Gallery
University-1652 [2]	37,855	951	701	51,355
	Ground → Satellite		Satellite → Ground	
	Query	Gallery	Query	Gallery
CVUSA [17]	8884	8884	8884	8884
CVACT_val [4]	8884	8884	8884	8884

IV. EXPERIMENT

We first introduce three large-scale cross-view geo-localization datasets, two small-scale landmark retrieval datasets and the evaluation protocol. Then Section IV-B describes the implementation detail. We provide the comparison with the state of the arts in Section IV-C, followed by the ablation study in Section IV-D.

A. Datasets and Evaluation Protocol

We mainly train and evaluate our method on three large-scale geo-localization datasets, *i.e.*, University-1652 [2], CVUSA [17] and CVACT [4]. Table I shows the image number of query and gallery sets for testing different tasks using these three datasets.

University-1652 [2] is a multi-view multi-source dataset containing satellite-view data, drone-view data and ground-view data. It collects 1652 buildings of 72 universities around the world. The training set includes 701 buildings of 33 universities, and the testing set includes the other 951 buildings of the rest 39 universities. **There are no overlapping universities in the training and test set.** Since some buildings do not have enough ground-view images to cover different aspects of these buildings, the dataset also provides an additional training set. Images in the additional training set are collected from the Google Image, and they have a similar view as the ground-view images. Therefore, the additional training set can be used as a supplement of the ground-view images. The dataset is employed to study two new tasks, *i.e.*, drone-view target localization (Drone → Satellite) and drone navigation (Satellite → Drone). There are 701 buildings with 50,218 images for training. In the drone-view target localization task (Drone → Satellite), there are 37,855 drone-view images in the query set and 701 true-matched satellite-view images and 250 satellite-view distractors in the gallery. There is only one true-matched satellite-view image under this setting. In the drone navigation task (Satellite → Drone), there are 701 satellite-view query images, and 37,855 true-matched drone-view images and 13,500 drone-view distractors in the gallery. There are multiple true-matched drone-view images under this setting.

CVUSA [17] provides the data collected from two views, *i.e.*, the ground view and the satellite view. Specifically, it contains 35,532 ground-and-satellite image pairs for training and 8884 image pairs for testing. All ground-view panoramic images are collected from Google Street View. Meanwhile, corresponding satellite-view images are downloaded from Microsoft Bing Maps.

CVACT [4] is a large-scale cross-view dataset. Same as CVUSA, CVACT provides 35,532 ground-and-satellite image pairs for training, and ground-view images are panoramas. Besides, CVACT provides a validation set with 8884 image pairs named CVACT_val and a testing set with 92,802 image pairs denoted as CVACT_test. A query image only has one true-matched image in the gallery for CVACT_val, while for CVACT_test, a query image may correspond to several true-matched images in the gallery.

Oxford5k [52] & Paris6k [53] are two prevailing landmark retrieval datasets collected from Flickr. Oxford5k consists of 5062 images that belong to 11 different Oxford landmarks, and Paris6k contains 6412 images of 12 particular Paris buildings. There are 55 query images in Oxford5k and 12 queries in Paris6k.

Evaluation protocol. In our experiments, we use the Recall@K (**R@K**) and the average precision (**AP**) to evaluate the performance of our model. R@K represents the proportion of correctly matched images in the top-K of the ranking list. A higher recall score shows a better performance of the network. We also calculate the area under the Precision-Recall curve, which is known as the average precision (AP), which reflects the precision and recall rate of the retrieval performance.

B. Implementation Details

We employ the ResNet-50 [5] with pre-trained weights on ImageNet [54] to extract visual features. Following [2], we modify the stride of the second convolutional layer and the last down-sample layer in conv5_1 of the ResNet-50 from 2 to 1. The newly-added layers in LPN, *i.e.*, the classifier module, are initialized with *kaiming initialization* [55]. We resize each input image to a fixed size of 256 × 256 pixels during training and testing. In training, we employ random cropping and flipping to augment the input data. For the optimizer, we adopt stochastic gradient descent (SGD) with momentum 0.9 and weight decay 0.0005 with a mini-batch of 32. The initial learning rate is 0.001 for backbone layers and 0.01 for the new layers. We train our model for 120 epochs, and the learning rate is decayed by 0.1 after 80 epochs. During testing, we utilize the Euclidean distance to measure the similarity between the query image and candidate images in the gallery. Our model is implemented on Pytorch [56], and all experiments are conducted on one NVIDIA RTX 2080Ti GPU.

C. Comparison with the State-of-the-arts

Results on University-1652. As shown in Table II, we compare the proposed method with other competitive approaches on University-1652. The proposed LPN has achieved 74.18% Recall@1 accuracy and 77.39% AP on Drone →

Satellite and 85.16% Recall@1 accuracy and 73.68% AP on Satellite → Drone without using the additional Google training data. The performance has surpassed the reported result of other competitive methods such as [23], [22], [57], [58], [9], [4], and the proposed method outperforms the best method, *i.e.*, instance loss [2] by a large margin, *i.e.*, about 14% AP improvement. If the extra training data, *i.e.*, noisy data collected from Google Image, is added into the training set [2], we could further boost the retrieval performance. In the drone-view target localization task (Drone → Satellite), the accuracy of Recall@1 increases from 74.18% to 75.93% and the value of AP goes up from 77.39% to 79.14%; in the drone navigation task (Satellite → Drone), the accuracy of Recall@1 increases from 85.16% to 86.45% and the value of AP raises from 73.68% to 74.79%. For the drone-view target localization task, there are 951 satellite-view images in the gallery. To make this retrieval task more challenging, we add 8884 satellite-view images collected from the testing set of CVUSA into the gallery of University-1652 as the distractors. Although the distractors would decrease the overall performance, indicated by Rank@1 and AP accuracy, the results are still competitive. This demonstrates the robustness of our proposed method against distractors.

TABLE II
COMPARISON WITH THE STATE-OF-THE-ART RESULTS REPORTED ON UNIVERSITY-1652. M STANDS FOR THE MARGIN OF THE TRIPLET LOSS. (W/O GOOGLE) INDICATES THAT THE EXTRA TRAINING SET COLLECTED FROM GOOGLE IMAGE IS NOT DEPLOYED IN TRAINING PHASE. (W/ CVUSA DISTRACTORS) DENOTES THAT ALL SATELLITE-VIEW IMAGES COLLECTED FROM THE TESTING SET OF CVUSA ARE ADDED INTO THE SATELLITE-VIEW GALLERY OF UNIVERSITY-1652 AS THE DISTRACTORS.

Method	University-1652			
	Drone → Satellite		Satellite → Drone	
	R@1	AP	R@1	AP
Instance Loss [2]	58.49	63.31	71.18	58.74
Contrastive Loss [23]	52.39	57.44	63.91	52.24
Triplet Loss ($M = 0.3$) [57]	55.18	59.97	63.62	53.85
Triplet Loss ($M = 0.5$) [57]	53.58	58.60	64.48	53.15
Soft Margin Triplet Loss [9]	53.21	58.03	65.62	54.47
Ours (w/o Google)	74.18	77.39	85.16	73.68
Ours	**75.93**	**79.14**	**86.45**	**74.79**
Ours (w/ CVUSA distractors)	70.61	73.53	-	-

Results on CVUSA. The comparison with other competitive methods on CVUSA is detailed in Table III. Ground-view images on CVUSA are panoramas, in which, the contextual information is generally distributed on both sides of the geographic target. Basing on the discussion in III-B, we deploy the sequential partition strategy to explicitly mine the contextual information on CVUSA (see Figure 4). The sequential partition strategy is a specific case of the square-ring partition strategy. As shown in Table III, we could observe two points. First, we deploy category recognition as the pretext task to conduct geo-localization on CVUSA. In particular, we regard 35,532 pairs as 35,532 location categories to train the model. The proposed method, whether using VGG16 [51] or ResNet-50 [5] as the backbone, surpasses most existing methods. Specifically, when using VGG16 as the backbone, our method achieves 79.69% R@1, 91.70% R@5, 94.55% R@10 and 98.50% R@Top1% on CVUSA. Since the feature expression capability of ResNet-50 is powerful than VGG16, our method with ResNet-50 backbone obtains 85.79% R@1, 95.38% R@5, 96.98% R@10 and 99.41% R@Top1% on CVUSA. Second, the proposed method is complementary to existing methods. For instance, our method can combine with the CVFT [7] and the SAFA [1] orthogonally. We re-implement CVFT and SAFA. Specifically, we keep the VGG16 as the backbone on both models unchanged and divide feature maps basing on Case III. For CVFT, we divide the final aligned feature maps into 8 parts and compute the loss of each corresponding part. For SAFA, the polar transform is retained. We first divide the feature maps into 8 parts, and then we use one SPE in SAFA to deal with each part separately before computing the loss. Combined with our partition strategy, CVFT+Ours boosts the R@1 accuracy from 61.43% to 68.20% (+6.77%) and the R@Top1% accuracy from 99.02% to 99.30% (+0.28%). Similarly, SAFA+Ours can further improve the R@1 accuracy from 89.84% to 92.83% (+2.99%) and the R@Top1% accuracy from 99.64% to 99.78% (+0.14%).

Significant performance improvements suggest that our method helps to mine more contextual information, yielding discriminative features.

Results on CVACT. CVACT has a similar data pattern with CVUSA. Our method with ResNet-50 [5] backbone achieves 79.99% R@1, 90.63% R@5, 92.56% R@10 and 97.03% R@Top1% on CVACT. For a fair comparison, our method using VGG16 as the backbone also acquires competitive results. CVFT [7]+Ours obtains the improvement with the R@1 accuracy from 61.05% to 62.90% (+1.85%) and the R@Top1% accuracy from 95.93% to 97.22% (+1.29%). SAFA [1]+Ours increases the R@1 accuracy from 81.03% to 83.66% (+2.63%) and the R@Top1% accuracy from 98.17% to 98.41% (+0.24%). The experimental results demonstrate that our method is still effective on CVACT.

D. Ablation Studies

To verify the effectiveness of components in our model, we design several ablation studies.

Effect of the number of parts. The number of parts n is one of the key parameters in our network. By default, we deploy $n = 4$. When n = 1, the model is employed to global average pooling. At this time, the model equals the baseline with global attention [2]. As shown in Figure 5, with the increment of n, both the Recall@1 and AP values have a significant improvement, since more contextual information has been taken into consideration. Intuitively, concatenating more contextual information parts can improve the discriminability of the final feature descriptor. However, we note that, as n increases, each part contains less receptive fields with limited semantic information. As a result, a higher value of n compromises the discriminability of the image representation. When n = 6 or 8, the performance gains slowly or even slightly degrades. Therefore, we use n = 4 as the default choice for our network, which balances the mining of the contextual information and the appropriate size of the receptive field.

TABLE III

RESULTS ON CVUSA, LIST SHOWS COMPARISONS OF VARIOUS METHODS. THERE ARE TWO SCHEMES TO OPTIMIZE THE NETWORK, *i.e.*, INSTANCE LOSS AND DEEP METRIC LEARNING. ZHENG *et al.* [2] GET THE BEST RESULT IN THE SCHEME OF INSTANCE LOSS, WHILE, IN THE DEEP METRIC LEARNING SCHEME, SAFA [1] IS A STATE-OF-THE-ART WORK. WE OBSERVE THAT THROUGH COMBINING OUR METHOD TO THESE TWO METHODS, THE OFF-THE-SHELF NETWORK CAN ACHIEVE A SIGNIFICANT PERFORMANCE BOOST. †: THE METHOD UTILIZES EXTRA ORIENTATION INFORMATION AS INPUT.

Method	Publication	Backbone	CVUSA				CVACT_val			
			R@1	R@5	R@10	R@Top1%	R@1	R@5	R@10	R@Top1%
MCVPlaces [22]	ICCV'15	AlexNet	-	-	-	34.40	-	-	-	-
Zhai [17]	CVPR'17	VGG16	-	-	-	43.20	-	-	-	-
Vo [26]	ECCV'16	AlexNet	-	-	-	63.70	-	-	-	-
CVM-Net [9]	CVPR'18	VGG16	18.80	44.42	57.47	91.54	20.15	45.00	56.87	87.57
Orientation† [4]	CVPR'19	VGG16	27.15	54.66	67.54	93.91	46.96	68.28	75.48	92.04
Zheng [2]	MM'20	VGG16	43.91	66.38	74.58	91.78	31.20	53.64	63.00	85.27
Regmi [59]	ICCV'19	X-Fork	48.75	-	81.27	95.98	-	-	-	-
Siam-FCANet [60]	ICCV'19	ResNet-34	-	-	-	98.30	-	-	-	-
CVFT [7]	AAAI'20	VGG16	61.43	84.69	90.94	99.02	61.05	81.33	86.52	95.93
SAFA [1]	NIPS'19	VGG16	89.84	96.93	98.14	99.64	81.03	92.80	94.84	98.17
Ours	-	VGG16	79.69	91.70	94.55	98.50	73.85	87.54	90.66	95.87
Ours	-	ResNet-50	**85.79**	**95.38**	**96.98**	**99.41**	**79.99**	**90.63**	**92.56**	**97.03**
CVFT [7] + Ours	-	VGG16	68.20	88.00	92.69	99.30	62.90	84.14	89.11	97.22
SAFA [1] + Ours	-	VGG16	**92.83**	**98.00**	**98.85**	**99.78**	**83.66**	**94.14**	**95.92**	**98.41**

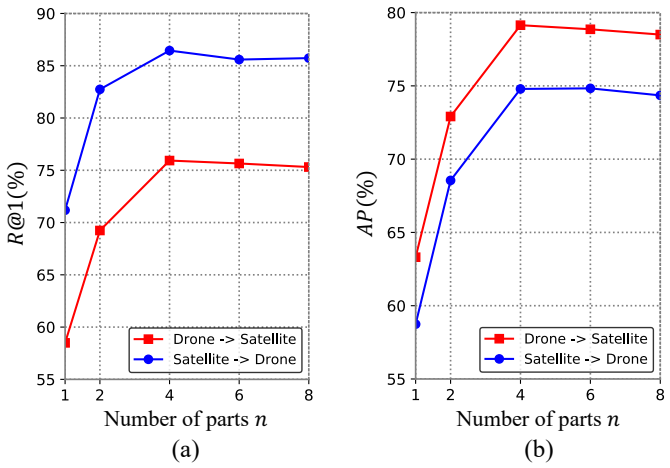

Fig. 5. The effect of the number of parts n on two tasks of the University-1652 dataset, *i.e.*, Drone → Satellite and Satellite → Drone. The red line refers to the task of drone-view target localization (Drone → Satellite). The blue line shows the task of drone navigation (Satellite → Drone). We show the effect of the number of parts for R@1 accuracy (a), and AP accuracy (b). We observe that LPN achieves the best performance when the number of parts n =4.

TABLE IV

ABLATION STUDY ON THE EFFECT OF DIFFERENT INPUT SIZES ON UNIVERSITY-1652.

Image Size	Drone → Satellite		Satellite → Drone	
	R@1	AP	R@1	AP
224	69.28	72.98	82.45	68.92
256	75.93	79.14	86.45	74.79
320	77.65	80.56	85.31	75.36
384	78.02	80.99	86.16	76.56
512	77.71	80.80	90.30	78.78

TABLE V

ABLATION STUDY ON ROTATING IMAGES DURING INFERENCE ON UNIVERSITY-1652.

Rotation Angle		Drone → Satellite		Satellite → Drone	
Query	Gallery	R@1	AP	R@1	AP
0°	0°	75.93	79.14	86.45	74.79
16°	0°	75.64	78.86	85.16	72.78
45°	0°	72.04	75.62	85.16	72.27
67°	0°	70.39	74.09	85.73	73.06
90°	0°	68.80	72.67	86.31	75.31
180°	0°	70.76	74.47	85.45	74.03
204°	0°	69.92	73.68	84.45	72.22
270°	0°	69.06	72.49	86.73	75.12
317°	0°	72.29	75.87	84.17	71.85
32°	75°	73.19	76.69	83.17	66.41
216°	87°	69.54	73.27	83.45	65.29

Effect of the input image size. A small training size compresses the fine-grained information of the input image, which compromises the discriminative representation learning. In contrast, a larger input size introduces more memory costs during training. To balance the input image size with the memory usage, we study the effect of the input image size. We just change the image size and the covered region of the image is not changed in the experiment. As shown in Table IV, in both tasks, *i.e.*, (Drone → Satellite) and (Satellite → Drone), as the input image size from 224 to 384, we observe that the performance gradually improves. When we continue to enlarge the input size to 512, the improvement is not clear on the Drone → Satellite task. We hope this study could help the real-world application in selecting the appropriate input size, when computation sources are limited.

Is LPN robust to rotation variants? Satellite-view images in University-1652 are north aligned and the orientation of drone-view images is random. In the training phase, the rotation augmentation is applied in the satellite-view branch but not in other branches. To verify the scalability of LPN for

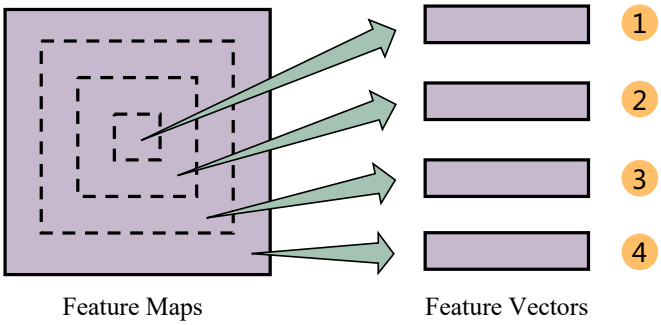

Fig. 6. The feature maps are first divided into four parts in LPN. Then, an average pooling layer transforms these four parts into four column vectors which are treated as subsequent feature descriptors. For each part, we use the numbers $1, 2, 3, 4$ to represent.

TABLE VI
ABLATION STUDY OF USING ONE PART OR A COMBINATION OF DIFFERENT PARTS DURING INFERENCE. 1, 2, 3, 4 INDICATE FOUR AVERAGED PARTS WHICH ARE SLICED FROM FEATURE MAPS.

Part Combination		Drone → Satellite		Satellite → Drone	
Query	Gallery	R@1	AP	R@1	AP
1	1	71.95	75.50	84.02	70.24
2	2	71.94	75.49	83.74	71.32
3	3	71.97	75.62	85.45	71.61
4	4	70.75	74.41	82.03	69.50
$1+2$	$1+2$	74.85	78.14	85.59	73.69
$1+2+3$	$1+2+3$	75.74	78.97	86.31	74.76
$1+2+3+4$	$1+2+3+4$	**75.93**	**79.14**	**86.45**	**74.79**
$1+2+3$	$2+3+4$	0.07	0.38	0.14	0.21
$1+2$	$2+3$	0.08	0.39	0.00	0.17
$1+2$	$3+4$	0.13	0.50	0.00	0.21

image rotation, we conduct experiments on rotating the query image to retrieval the true-matched images. We do not rotate gallery images but query images. The experimental results are shown in Table V. The $0°$ denotes the input query image without rotation. For the task of drone navigation (Satellite → Drone), LPN obtains robust features for unseen satellite-view query images against different rotation angles. In contrast, for the task of drone-view target localization (Drone → Satellite), we do not train the model on the rotated drone-view data. The LPN still achieves one competitive performance without a significant performance drop. In addition, we also attempt to rotate different angles on query and gallery images to further test our model. The experimental results suggest that LPN has good scalability to rotation variations.

Does LPN learn complementary part features? The square-ring partition strategy divides the feature maps into four parts in LPN. We use the numbers 1, 2, 3, and 4 to represent the four parts of the feature maps, as shown in Figure 6. Subsequently, we conduct experiments by choosing one or a combination of the four parts. The experimental results demonstrate LPN has learned complementary features (see Table VI). We observe that using only one part has obtained one fairly good result in two tasks, *i.e.*, (Drone → Satellite) and (Satellite → Drone). When we further concatenate two or three parts, the accuracy of Recall@1 and AP gradually increases. When all parts are leveraged, we obtain the best performance in both tasks. It demonstrates that LPN has learned complementary part feature, enriching the model capability. But the Recall@1 accuracy and AP drop dramatically in both tasks if query features and gallery features cover different parts. The results reflect from the side that the learned semantic information between the parts is complementary. Because of the complementary in each part, the semantic information contained in different parts is not overlapping. When there are different parts in the query and gallery feature (*i.e.*, the true-matched image pair covers significant different areas), the different parts become the distractors for the final visual feature, resulting in terrible retrieval performance.

TABLE VII
ABLATION STUDY ON SHIFTING QUERY IMAGES DURING INFERENCE ON UNIVERSITY-1652.

Shifted Pixel	Drone → Satellite		Satellite → Drone	
	R@1	AP	R@1	AP
0	75.93	79.14	86.45	74.79
10	75.26	78.57	85.16	72.84
20	72.37	76.04	83.02	70.67

Fig. 7. The first and last images are the original query and gallery images in the test set. We pad 10 and 20 pixels in the left of the query image in the way of reflection, respectively. Then, we crop the padded image to the original image size in a left-aligned manner. Thus we can obtain the second and third images, which have an offset of the geographic target with the gallery image. The left space of the red dotted line is the extra padded pixels.

Is LPN robust to the shifted query image? In the realistic scenario, there is usually an offset in the geographic target location of the query image and true-matched images in the gallery. To explore whether LPN can cope with the offset of the geographic target location in a true-matched image pair, we carry out experiments on shifting the query image during testing. Specifically, we shift the query image to the right in pixels and keep images in the gallery intact (see Figure 7). Table VII shows the experimental results. 0 indicates that the input query image is not offset. When the input query image is shifted 10 pixels, we can hardly observe a performance drops for drone navigation and drone-view target localization tasks. While the shifted pixels is 20, the performance on both tasks decreases slightly. The experimental results suggest that LPN is robust when there is a small offset of the geographic target location for a true-matched image pair in retrieval.

Geo-localization between satellite-view images and ground-view images. In University-1652, geo-localization between satellite-view images and ground-view images is a challenging task. We can sum up the difficulties in the

TABLE VIII
THE PERFORMANCE OF GEO-LOCALIZATION BETWEEN SATELLITE-VIEW IMAGES AND GROUND-VIEW IMAGES. (W/O DRONE) INDICATES THAT LPN IS TRAINED WITHOUT DRONE-VIEW IMAGES.

Method	Satellite → Ground		Ground → Satellite	
	R@1	AP	R@1	AP
Baseline [2]	1.14	1.14	1.20	2.52
Ours (w/o drone)	1.43	1.31	0.74	1.83
Ours	1.85	1.66	0.81	2.21

TABLE IX
ABLATION STUDY ON USING DRONE-VIEW IMAGES WITH DIFFERENT DISTANCE TO THE GEOGRAPHIC TARGET TO CONDUCT RETRIEVAL. "ALL" INDICATES THAT WE APPLY ALL DRONE-VIEW QUERY IMAGES.

Distance	Drone → Satellite	
	R@1	AP
All	75.93	79.14
Long	60.20	65.01
Short	75.04	78.35
Middle	80.03	82.77

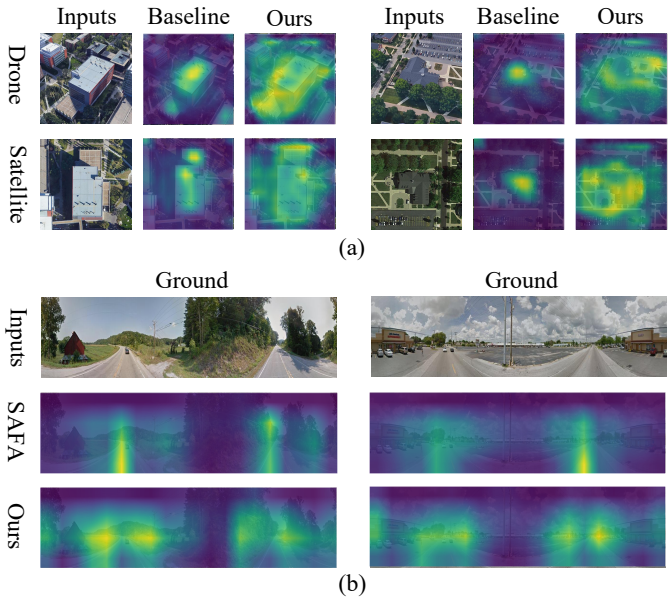

Fig. 8. Visualization of heatmaps. (a) Heatmaps generated by baseline [2] and ours in different platforms of University-1652. (b) Ground-view heatmaps learned from SAFA [1] and ours (SAFA + ours) on CVUSA.

following points. (1) Ground and satellite images that are collected from different viewpoints naturally have a distinct visual appearance. (2) Unlike ground-view images in CVUSA and CVACT, which are panoramas, the ground-view image in University-1652 only covers part of the whole building. (3) Ground-view images in University-1652 contains vast obstacles, *e.g.*, trees and cars. From Table VIII, we can notice that geo-localization between satellite-view and ground-view images do not work well. Using satellite-view images to retrieve the ground-view images, our method achieves the best results in Recall@1 accuracy and AP. Whether employing drone-view images or not, there is a performance decrease in the ground-to-satellite localization compared with baseline [2]. But using drone-view images in LPN obtains better results than without using these.

Effect of the drone distance to the geographic target. The scale of the satellite-view image in University-1652 is fixed, while the scale of the drone-view image changes dynamically with the drone distance to the geographic target. We adopt drone-view images with different distances to the geographic target as queries to study the impact of the changed scale for LPN. As shown in Table IX, when the drone-view image is captured in a middle distance to the geographic target, we obtain the best performance. When the drone distance is short to the geographic target, we can observe that the results are still competitive compared with using all drone-view query images. The scales of these drone-view images are close to satellite-view images. Another reason is that these drone-view images mainly contain the target building without extra trees and other buildings.

Transfer learning from University-1652 to small-scale datasets. To study the generalization ability of LPN trained on University-1652, we evaluate three models on two small-scale landmark retrieval datasets, *i.e.*, Oxford5k [52] and Paris6k [53]. The first model is ResNet-50 [5] trained on ImageNet [54]. The second model is baseline [2] and LPN is the third model. During the evaluation, three models have not been fine-tuned on these two datasets. For baseline and LPN, we choose two different branches, *i.e.*, $\mathcal{F}_s$ and $\mathcal{F}_g$ to extract features, since these two branches focus on different low-level patterns of input images. Weights on $\mathcal{F}_s$ are trained by the satellite-view images, while $\mathcal{F}_g$ is learned on the ground-view images. From Table X, we observe that the extracted feature from LPN shows better performance on both two datasets than features obtained from ResNet-50 and baseline. This result also demonstrates that the square-ring partition strategy can enhance the generalization ability of our model. We also note that in the same model, $\mathcal{F}_g$ has a superior generalization ability than $\mathcal{F}_s$. It is because that images in Oxford5k and Pairs6k are closer to the Google Street View images, which are similar to the ground-view images collected from Google Image. Besides, $\mathcal{F}_s$ is trained by the aerial-view data, which viewpoint is perpendicular to the ground plane. In contrast, the data viewpoint in Oxford5k or Paris6k is parallel to the ground plane.

TABLE X
TRANSFER LEARNING FROM UNIVERSITY-1652 TO SMALL-SCALE DATASETS, *i.e.*, OXFORD5K [52] AND PARIS6K [53]. WE SHOW THE AP (%) ACCURACY ON TWO DATASETS.

Dataset	ResNet-50	baseline [2]		Ours	
		$\mathcal{F}_s$	$\mathcal{F}_g$	$\mathcal{F}_s$	$\mathcal{F}_g$
Oxford5k [52]	8.43	15.62	41.12	27.02	51.71
Paris6k [53]	27.93	38.18	59.00	45.81	67.73

E. Qualitative Result

As an additional qualitative evaluation, we visualize some heatmaps created by our and compared methods. Figure 8 (a) shows heatmaps generated by baseline [2] and LPN in the

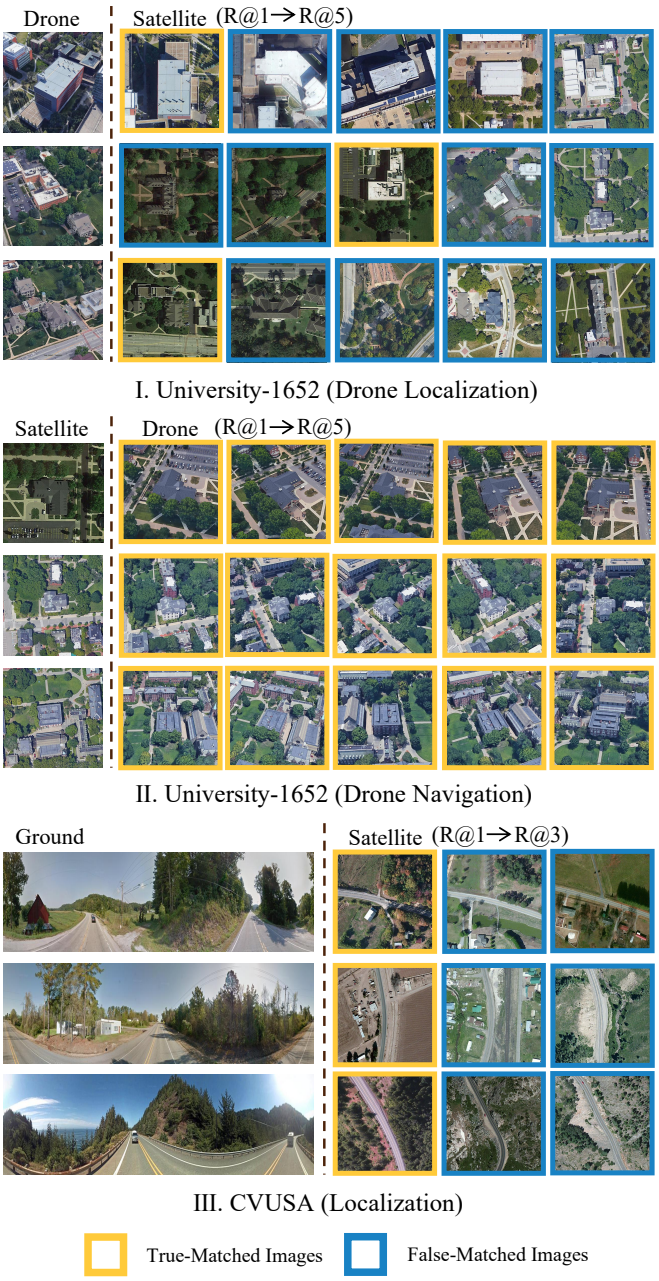

Fig. 9. Qualitative image retrieval results. (I) Top-5 retrieval results of drone-view target localization on University-1652. (II) Top-5 retrieval results of drone navigation on University-1652. (III) Top-3 retrieval results of geographic localization on CVUSA. The true matches are in yellow boxes, while the false matches are displayed in blue boxes.

drone and satellite platforms. Compared with the baseline, our approach activates the region of the geographic target and neighbor areas containing the contextual information. Figure 8 (b) shows the ground-view heatmaps generated by the original SAFA [1] and SAFA fused our partition strategy (Ours). SAFA only activates the position of the road, while our method further places emphasis on contextual information next to the road position. Our method is more consistent with the processing of the human visual system to locate an unfamiliar road.

Moreover, we show some retrieval results for different tasks on University-1652 and CVUSA in Figure 9. On University-1652, we observe that LPN can adapt to retrieve the reasonable images based on the content in both drone-view localization and drone navigation tasks. One failure case is also shown in the second row of Figure 9 (I), in which LPN can not recall the matched image in top-1. We notice that it is challenging in that the recalled top-1 image has a very similar pattern with the query image, especially the appearance of the geographic target in two images. On CVUSA, we observe a similar result. Our method with SAFA [1] has successfully retrieved the relevant satellite-view images.

V. Conclusion

In this paper, we identify the challenge in cross-view geo-localization, and propose a simple and effective deep neural network, called Local Pattern Network (LPN), to explicitly mine the contextual information. Specifically, we introduce a square-ring partition strategy for learning complementary spatial features according to the distance to the image center. The contextual information enhances the discriminability of the image representation with more fine-grained patterns. Our approach has achieved competitive accuracy on three cross-view geo-localization benchmarks, *i.e.*, University-1652, CVUSA and CVACT. Moreover, the proposed LPN has good scalability to rotation variation, which is close to the real-world application. The square-ring partition strategy also can be easily embedded into other frameworks to boost performance. In the future, we will investigate applying a module, such as STN [41], to estimate the scale of the drone-view image.

References

[1] Y. Shi, L. Liu, X. Yu, and H. Li, "Spatial-aware feature aggregation for image based cross-view geo-localization," in *Neural Information Processing Systems*, 2019.

[2] Z. Zheng, Y. Wei, and Y. Yang, "University-1652: A multi-view multi-source benchmark for drone-based geo-localization," in *ACM International Conference on Multimedia*, 2020, doi: 10.1145/3394171.3413896.

[3] S. Workman and N. Jacobs, "On the location dependence of convolutional neural network features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[4] L. Liu and H. Li, "Lending orientation to neural networks for cross-view geo-localization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[6] Z. Zheng, L. Zheng, M. Garrett, Y. Yang, M. Xu, and Y.-D. Shen, "Dual-path convolutional image-text embeddings with instance loss," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 2, pp. 1–23, 2020, doi: 10.1145/3383184.

[7] Y. Shi, X. Yu, L. Liu, T. Zhang, and H. Li, "Optimal feature transport for cross-view image geo-localization," in *AAAI Conference on Artificial Intelligence*, 2020.

[8] Y. Shi, X. Yu, D. Campbell, and H. Li, "Where am i looking at? joint location and orientation estimation by cross-view matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[9] S. Hu, M. Feng, R. M. Nguyen, and G. Hee Lee, "Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[10] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[11] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao, "Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[12] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[13] Y. Fu, X. Wang, Y. Wei, and T. Huang, "Sta: Spatial-temporal attention for large-scale video-based person re-identification," in *AAAI Conference on Artificial Intelligence*, 2019.

[14] R. A. Rensink, "The dynamic representation of scenes," *Visual Cognition*, vol. 7, no. 1-3, pp. 17–42, 2000.

[15] M. Corbetta and G. L. Shulman, "Control of goal-directed and stimulus-driven attention in the brain," *Nature Reviews Neuroscience*, vol. 3, no. 3, pp. 201–215, 2002.

[16] Z. Zheng, T. Ruan, Y. W. Wei, Y. Yang, and M. Tao, "Vehiclenet: Learning robust visual representation for vehicle re-identification," *IEEE Transactions on Multimedia (TMM)*, 2020, doi: 10.1109/TMM.2020.3014488.

[17] M. Zhai, Z. Bessinger, S. Workman, and N. Jacobs, "Predicting ground-level scene layout from aerial imagery," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[18] F. Castaldo, A. R. Zamir, R. Angst, F. A. N. Palmieri, and S. Savarese, "Semantic cross-view matching," in *IEEE International Conference on Computer Vision Workshops*, 2015.

[19] T.-Y. Lin, S. Belongie, and J. Hays, "Cross-view image geolocalization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[20] T. Senlet and A. Elgammal, "A framework for global vehicle localization using stereo images and satellite and road maps," in *IEEE International Conference on Computer Vision Workshops*, 2011.

[21] M. Bansal, H. S. Sawhney, H. Cheng, and K. Daniilidis, "Geo-localization of street views with aerial image databases," in *ACM International Conference on Multimedia*, 2011.

[22] S. Workman, R. Souvenir, and N. Jacobs, "Wide-area image geolocalization with aerial reference imagery," in *IEEE International Conference on Computer Vision*, 2015.

[23] T.-Y. Lin, Y. Cui, S. Belongie, and J. Hays, "Learning deep representations for ground-to-aerial geolocalization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[24] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[25] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[26] N. N. Vo and J. Hays, "Localizing and orienting street views using overhead imagery," in *European Conference on Computer Vision*, 2016.

[27] P. Li, P. Pan, P. Liu, M. Xu, and Y. Yang, "Hierarchical temporal modeling with mutual distance matching for video based person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[28] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," in *IEEE International Conference on Computer Vision*, 2017, doi: 10.1109/ICCV.2017.405.

[29] Y. Amit and A. Trouvé, "Pop: Patchwork of parts models for object recognition," *International Journal of Computer Vision*, vol. 75, no. 2, pp. 267–282, 2007.

[30] D. Crandall, P. Felzenszwalb, and D. Huttenlocher, "Spatial priors for part-based recognition using statistical models," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[31] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[32] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

[33] M. Weber, M. Welling, and P. Perona, "Towards automatic discovery of object categories," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[34] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[35] D. G. Lowe, "Object recognition from local scale-invariant features," in *IEEE International Conference on Computer Vision*, 1999.

[36] H. Zhao, M. Tian, S. Sun, J. Shao, J. Yan, S. Yi, X. Wang, and X. Tang, "Spindle net: Person re-identification with human body region guided feature decomposition and fusion," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[37] J. Xu, R. Zhao, F. Zhu, H. Wang, and W. Ouyang, "Attention-aware compositional network for person re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[38] J. Guo, Y. Yuan, L. Huang, C. Zhang, J.-G. Yao, and K. Han, "Beyond human parts: Dual part-aligned representations for person re-identification," in *IEEE International Conference on Computer Vision*, 2019.

[39] D. Li, X. Chen, Z. Zhang, and K. Huang, "Learning deep context-aware features over body and latent parts for person re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[40] Z. Zheng, L. Zheng, and Y. Yang, "Pedestrian alignment network for large-scale person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 10, pp. 3037–3045, 2018, doi: 10.1109/TCSVT.2018.2873599.

[41] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Neural Information Processing Systems*, 2015.

[42] L. Zhao, X. Li, Y. Zhuang, and J. Wang, "Deeply-learned part-aligned representations for person re-identification," in *IEEE International Conference on Computer Vision*, 2017.

[43] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, "Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline)," in *European Conference on Computer Vision*, 2018.

[44] X. Sun and L. Zheng, "Dissecting person re-identification from the viewpoint of viewpoint," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[45] Z. Zhong, L. Zheng, Z. Luo, S. Li, and Y. Yang, "Invariance matters: Exemplar memory for domain adaptive person re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[46] J. Song, Y. Yang, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Generalizable person re-identification by domain-invariant mapping network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[47] Y. Fu, Y. Wei, Y. Zhou, H. Shi, G. Huang, X. Wang, Z. Yao, and T. Huang, "Horizontal pyramid matching for person re-identification," in *AAAI Conference on Artificial Intelligence*, 2019.

[48] L. Zheng, Y. Huang, H. Lu, and Y. Yang, "Pose-invariant embedding for deep person re-identification," *IEEE Transactions on Image Processing*, vol. 28, no. 9, pp. 4500–4509, 2019.

[49] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian, "Pose-driven deep convolutional model for person re-identification," in *IEEE International Conference on Computer Vision*, 2017.

[50] L. Wei, S. Zhang, H. Yao, W. Gao, and Q. Tian, "Glad: Global–local-alignment descriptor for scalable person re-identification," *IEEE Transactions on Multimedia*, vol. 21, no. 4, pp. 986–999, 2018.

[51] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[52] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[53] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[54] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[55] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *IEEE International Conference on Computer Vision*, 2015.

[56] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Neural Information Processing Systems*, 2019.

[57] G. Chechik, V. Sharma, U. Shalit, and S. Bengio, "Large scale online learning of image similarity through ranking," in *Iberian Conference on Pattern Recognition and Image Analysis*, 2009.

[58] C. Deng, Z. Chen, X. Liu, X. Gao, and D. Tao, "Triplet-based deep hashing network for cross-modal retrieval," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3893–3903, 2018.

[59] K. Regmi and M. Shah, "Bridging the domain gap for ground-to-aerial image matching," in *IEEE International Conference on Computer Vision*, 2019.

[60] 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 *IEEE International Conference on Computer Vision*, 2019.