Hindawi
Mobile Information Systems
Volume 2017, Article ID 8104386, 14 pages
https://doi.org/10.1155/2017/8104386

Research Article

Efficient ConvNet Feature Extraction with Multiple RoI Pooling for Landmark-Based Visual Localization of Autonomous Vehicles

Yi Hou,[1] Hong Zhang,[2] Shilin Zhou,[1] and Huanxin Zou[1]

[1]*College of Electronic Science and Engineering, National University of Defense Technology, Changsha, Hunan, China*
[2]*Department of Computing Science, University of Alberta, Edmonton, AB, Canada T6G 2E8*

Correspondence should be addressed to Yi Hou; yihouhowie@gmail.com

Received 12 May 2017; Revised 28 July 2017; Accepted 11 October 2017; Published 9 November 2017

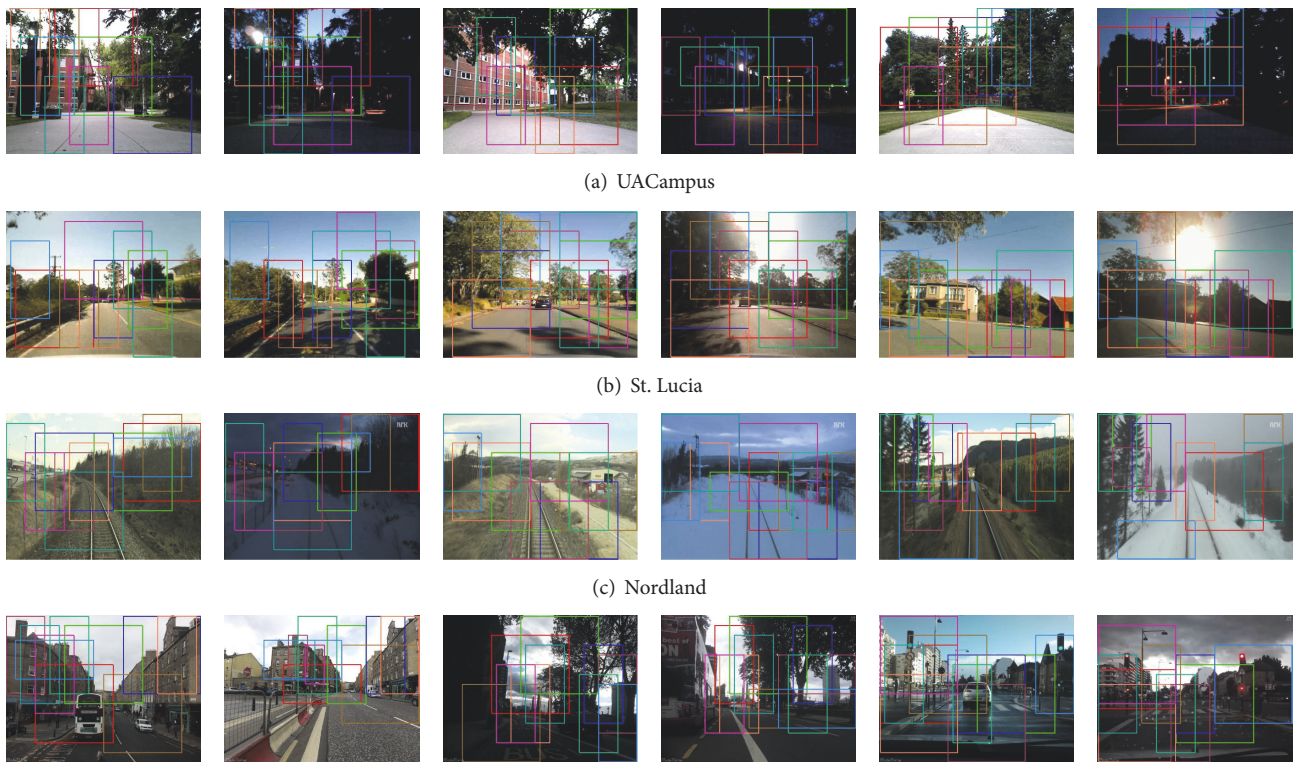
Academic Editor: Paolo Bellavista

Efficient and robust visual localization is important for autonomous vehicles. By achieving impressive localization accuracy under conditions of significant changes, ConvNet landmark-based approach has attracted the attention of people in several research communities including autonomous vehicles. Such an approach relies heavily on the outstanding discrimination power of ConvNet features to match detected landmarks between images. However, a major challenge of this approach is how to extract discriminative ConvNet features efficiently. To address this challenging, inspired by the high efficiency of the region of interest (RoI) pooling layer, we propose a *Multiple RoI (MRoI)* pooling technique, an enhancement of RoI, and a simple yet efficient ConvNet feature extraction method. Our idea is to leverage MRoI pooling to exploit multilevel and multiresolution information from multiple convolutional layers and then fuse them to improve the discrimination capacity of the final ConvNet features. The main advantages of our method are (a) high computational efficiency for real-time applications; (b) GPU memory efficiency for mobile applications; and (c) use of pretrained model without fine-tuning or retraining for easy implementation. Experimental results on four datasets have demonstrated not only the above advantages but also the high discriminating power of the extracted ConvNet features with state-of-the-art localization accuracy.

1. Introduction

Efficient and reliable visual localization is a core requirement for smart transportation applications such as autonomous cars, self-driving public transport vehicles, and mobile robots. Its aim is to use visual sensors such as cameras to solve the problem of "where am I?" and facilitate life-long navigation, by determining whether the current view of the camera corresponds to a location that has been already visited or seen [1]. Compared to the solutions that use other sensors such as LIDAR, visual localization is inherently more flexible and cheaper to use [1]. Therefore, visual localization for transportation systems has become a hot topic. In particular, recent interest in autonomous vehicles has created a strong need for visual localization techniques that can efficiently operate in challenging environments. Although current state-of-the-art approaches have made great strides [2–12], visual localization for long-term navigation of autonomous vehicles still remains an unsolved problem when image appearance experiences significant changes caused by time of the day, season, weather, camera pose, etc. [1].

Recently, a ConvNet landmark-based visual localization approach proposed in [13] has achieved state-of-the-art localization accuracy under conditions of significant environmental and viewpoint changes, raising the interest of the community [1, 14, 15]. Some sample examples of matched image pairs produced by such an approach are illustrated in Figure 1. Its key idea is to leverage the discrimination power of ConvNet features to describe high-level visual landmarks in the image, in order to achieve viewpoint invariance [1, 13]. For this point, such an approach relies heavily on the great descriptive power of ConvNet features to match detected landmarks between images. At the same time, a practical consideration for ConvNet feature extraction is to be efficient.



(a) UACampus

(b) St. Lucia

(c) Nordland

(d) Mapillary

Figure 1: Sample examples of matched image pairs produced by a ConvNet landmark-based visual localization approach, which extracted ConvNet features by one variant of our proposed method, that is, *MRoI-FastRCNN-AlexNet* (see Section 5.1.2 for details). These images come from the testing datasets used in our experiments (see Section 5.1.1 for details). Six images on each row come from one dataset, and the three pairs illustrate images correctly matched by our method. The bounding boxes of the same color in each pair of matched images show the landmarks that have been matched. For clarity, we show only ten matched landmarks in each image. Best viewed in color.

However, to the best of our knowledge, efficient extraction has been largely overlooked in visual localization research, and people rely on existing ConvNet feature extraction methods introduced for computer vision applications such as image classification [16–18] and object detection [19–21], without specializing them for the visual localization application. As we will discuss in detail in Section 3, these existing methods fall into two groups: *original image-based* and *feature map-based*. In general, methods in the first group are accurate enough for localization but time-consuming while those in the second group are fast enough but not accurate as the first group for localization. Therefore, there is an urgent need to develop a method to achieve the speed and accuracy at the same time.

To this end, in this paper we present a simple yet efficient method to extract discriminative ConvNet features for visual localization of autonomous vehicles that is highly efficient both in computation and in GPU memory, using the technique which we refer to as *multiple RoI (MRoI)* pooling. As an enhancement of a special pooling layer called *region of interest (RoI)* [20], MRoI pooling inherits the high efficiency of RoI pooling. Therefore, we are able to use MRoI pooling to efficiently exploit multilevel and multiresolution information from multiple convolutional layers, instead of only one as in previous feature map-based methods. Furthermore, we fuse information across multiple layers to improve the discrimination capacity of the final ConvNet features.

Extensive experimental results on four datasets with various changes in environmental conditions have demonstrated that (a) our proposed method is fast, GPU memory efficient, and based on a pretrained model without fine-tuning or retraining and (b) the discrimination capacity of ConvNet features extracted by our method is higher than those of feature map-based methods on all testing datasets. Moreover, our method is also comparable with those of original image-based methods, with state-of-the-art localization accuracy.

The rest of this paper is organized as follows. Section 2 briefly reviews related literature with respect to visual localization. Section 3 describes and analyzes existing methods for extracting ConvNet features. Section 4 provides the details of our proposed method. Section 5 presents the experiments and results. Finally, we conclude the work in Section 6.

2. Related Work

Prior to the emergence of CNN, visual localization approaches mainly depended on hand-crafted features developed in computer vision, in order to represent the scenes

observed by vehicles or mobile robots during navigation. A popular baseline algorithm among traditional approaches is FAB-MAP [2]. It used local features like SURF to represent an image and achieved efficient image matching with bags-of-words. For SLAM, RTABMap [22, 23] used both SIFT and SURF. On the other hand, some methods used binary local features for the high-efficiency matching. For example, by encoding BRIEF and FAST into a bag of binary words, [24] performed fast localization. Recently, ORB-SLAM [25] showed promising performance by employing ORB features. However, most of local feature-based approaches have demonstrated only a limited degree of environmental invariance, despite displaying a reasonable degree of viewpoint invariance [1]. The reason for this limited success is that local features are usually only partially invariant to environmental changes.

In contrast, global feature-based methods have demonstrated better environmental invariance. For example, Gist features were used to construct a whole descriptor of an image in visual localization applications such as [3, 6]. Besides Gist, BRIEF-Gist [5] further integrated BRIEF to improve the efficiency of image matching by computing the Hamming distance. To handle significant environmental changes due to weather, daylight, and season, SeqSLAM [7] and its variants [8–10] have been developed. They exploited temporal information and consistency of image sequences instead of single images to define places. However, these global feature-based approaches are known to fail easily in the presence of viewpoint changes. In summary, traditional approaches are difficult to satisfy practical requirements in conditions that experience both environmental and viewpoint changes simultaneously [1].

With the outstanding power on various visual tasks, CNN has been popularly applied to visual localization and has achieved promising results [12, 26, 27]. A comprehensive evaluation performed in [26] has demonstrated that the discrimination capacity of ConvNet features is much more than those of the state-of-the-art hand-crafted features such as Gist [28], BoW [29], Fisher vector [30], and VLAD [31]. In addition, the advantages of ConvNet features in environments with various changes have been further confirmed by another evaluation study [27]. Since then, ConvNet features have been widely applied to improve some existing visual localization methods such as SeqSLAM [7] and a season-robust method using network flows [11], where hand-crafted features were replaced by ConvNet features [12, 32].

Instead of directly using pretrained CNN models, some works [33, 34] fine-tuned or redesigned and retrained specialized CNNs on datasets that are specific to visual localization, in order to further improve the discrimination capacity of ConvNet features. Regardless, because ConvNet features were still used as a global image descriptor, all these approaches mentioned above suffer the weakness of viewpoint sensitivity, although their robustness against environmental changes has been improved.

To address this problem, a ConvNet landmark-based approach was proposed in [13]. It has been shown state-of-the-art localization accuracy in challenging environments. This success is attributed to two reasons. First, viewpoint invariance is achieved by combining the benefits of global and local features [1]. Second, compared to previous methods using hand-crafted visual features, this approach improves the description capability of the detected landmarks, by making full use of the discrimination power of ConvNet features [26, 27]. However, the ConvNet feature extraction method used in such an approach lacks time efficiency that is required in a visual localization application of autonomous vehicles. It is the need for producing an efficient solution with excellent invariance properties that motivated our research described in this paper.

3. Existing ConvNet Feature Extraction Methods

In this section, we will describe existing methods for extracting ConvNet features and discuss their advantages and disadvantages in detail. According to the type of subimages from which a ConvNet feature is extracted, existing ConvNet feature extraction methods fall into two groups: *original image-based* and *feature map-based*. Similar to R-CNN [19], original image-based methods usually first crop corresponding subimages from the original input image according to the bounding boxes of detected landmarks as shown in Figure 1 and then resize them to predefined dimensions, before feeding them into a CNN network to extract ConvNet features. As shown in [13], the ConvNet features extracted by such a method are discriminative enough to achieve the state-of-the-art localization accuracy under challenging conditions. However, its computation is usually too time-consuming to meet the real-time requirement. This is because these methods need to not only resize the cropped subimages but also repeatedly evaluate the layers of the CNN network as many times as there are landmarks detected in an image. Even though sending all cropped regions into the network as a batch can reduce the running time, the computational efficiency is still unsatisfactory. Moreover, the batch processing of all cropped images increases the requirement on GPU memory, making its implementation difficult in embedded systems or mobile devices with limited GPU resources, which are popularly equipped in an autonomous vehicle.

On the contrary, feature map-based methods are much more efficient in computation and GPU memory, but their ConvNet features are less discriminative. Similar to Fast R-CNN [20], feature map-based methods directly extract ConvNet features from the feature maps at the last and coarsest convolutional layer. Specifically, they utilize RoI [20] to directly pool the feature of a detected landmark on the feature maps and then generate a fixed-length representation for describing this landmark. In this way, the convolutional layers of a CNN network are needed to be computed only once on the entire image. For this reason, feature map-based methods are much faster than original image-based methods. Despite the computational advantage, the ConvNet features extracted by existing feature map-based methods are less discriminative than those of original image-based methods. This is due to the fact that the feature maps are a downsampled form of the original image, causing a loss in performance. For example, the size of each feature map at

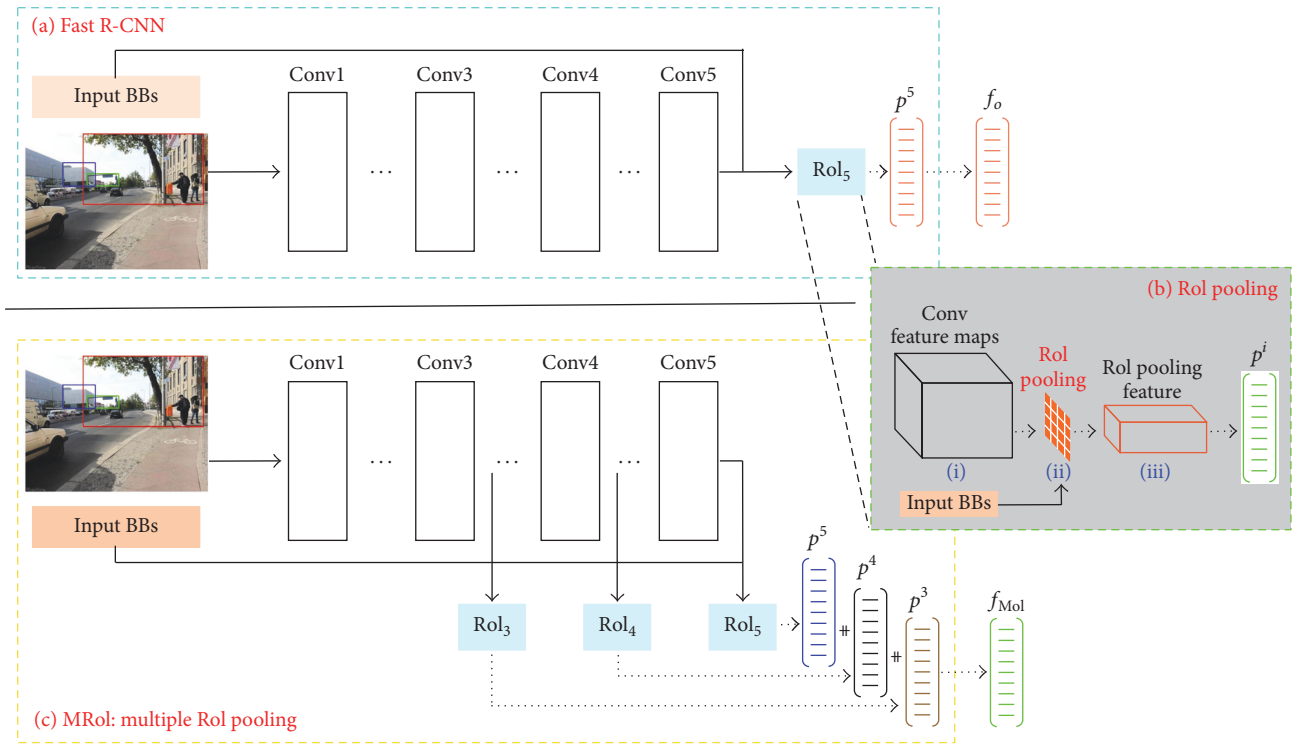


FIGURE 2: Illustration of our proposed ConvNet feature extraction method with multiple RoI pooling (MRoI). For ease of understanding, we show in (a) the existing feature map-based methods which directly use the Fast R-CNN [20] to extract ConvNet features. In addition, we also show the principle of a RoI pooling layer in (b). Our method is illustrated in (c). Obviously, our method is very simple because it only needs to add two extra RoI pooling layers (i.e., RoI_3 and RoI_4) behind the Conv3 and Conv4 layers. Note that "p_i" ($i = 3, 4, 5$) represents the vectorized RoI pooling features from the corresponding RoI pooling layer. For the purpose of feature fusion, p_i is first ℓ_2-*normalized* and then concatenated (⋕). The final output ConvNet features of Fast R-CNN and our method are denoted as "f_o" and "f_{MoI}," respectively. For clarity, we present only three of the bounding boxes (BBs) detected within an image.

the Conv5 layer is as small as 14×14 pixels when sending an original input image of $224 \times 224 \times 3$ pixels into the AlexNet network [16]. Obviously, such a small feature map is fairly coarse. Consequently, the extracted ConvNet feature of a landmark may not contain adequate discriminative information if its bounding box is not large enough. The deficient discrimination power of ConvNet features often reduces the final localization accuracy, especially in the case of significant viewpoint change.

To sum up, existing feature extraction methods have not successfully tackled the issue of achieving high computational and GPU memory efficiency and high discrimination power at the same time, serving as the motivation of our research.

4. Our Proposed Method to Visual Localization

In this section, we present the proposed method for efficient ConvNet landmark-based visual localization. Details of our method are illustrated in Figure 2. As will be seen, our method is simple and straightforward. Here, we first describe how to construct the proposed multiple RoI pooling layer and then present our feature extraction method.

4.1. MRoI: Multiple RoI Pooling. In essence, our proposed MRoI is an enhanced version of the RoI pooling layer, which is a special pooling layer following the Conv5 layer of Fast R-CNN [20].

The principle of an RoI pooling layer is illustrated in Figure 2(b). It takes as input the C (the number of channels) feature maps at the corresponding convolutional layer and the bounding boxes (BBs) of all detected landmarks, as shown in Figure 2(b)-(i). Based on the transform relationship between the sizes of the original input image and the feature maps, these BBs are converted to corresponding regions of interest (RoIs) on the feature maps. For each RoI, its region is divided into 6×6 spatial bins, as shown in Figure 2(b)-(ii) (for clarity we draw only 4×4 spatial bins). Moreover, the max pooling is performed within each spatial bin across all channels. Thus, for each detected landmark, its subimages on the feature maps, which are a multidimensional array of size $6 \times 6 \times C$, can be obtained, as shown in Figure 2(b)-(iii). These subimages are finally used as the RoI pooling feature of the landmark. Therefore, by using the RoI pooling layer, a feature map-based method such as Fast R-CNN computes the feature maps from the entire image only once and then pools features in the arbitrary region of a detected landmark

Table 1: Main properties of four testing datasets used in our experiments. For a dataset with multiple subsets, two subsets with extreme changes are listed below and will be matched in our experiments. "No." indicates the number of images in a subset. "—" means the change is negligible.

Dataset	Subset	No.	Main changes			
			Illumination	Shadow	Season	Viewpoint
UACampus [35]	06:20 22:15	649 647	large	—	—	—
St. Lucia [36]	2009-09-10 08:45 2009-09-11 15:45	1000 1000	moderate	large	—	minor
Nordland [37]	Spring winter	1000 1000	large	—	large	—
Mapillary [38]	mploop1 mploop2	2028 2028	moderate	—	—	large

to generate its representation. Most importantly, RoI pooling avoids repeatedly computing the convolutional layers. This is the main reason why feature map-based methods are usually much faster than original image-based methods.

Despite the speed superiority, the ConvNet features extracted by existing feature map-based methods are less discriminative than desired, because these methods extract features from only one RoI pooling layer after the Conv5 layer, that is, the coarsest convolutional layer. To enhance the discrimination capacity while retaining the high computation efficiency, we propose a MRoI pooling layer, which are made up of three RoI pooling layers, to exploit the finer and richer information from multiple convolutional layers than that available from a single layer.

The construction of MRoI is simple. As illustrated in Figure 2(c), we only need to simply add two extra RoI pooling layers, that is, RoI_3 and RoI_4, behind the Conv3 and Conv4 layers, respectively. Note that the two extra RoI pooling layers are easy to insert in any pretrained CNNs, because we only need to slightly modify the corresponding configuration file, that is, by replicating the setting of RoI_5 behind the Conv3 and Conv4 layers. Therefore, our method can work in a "plug and play" solution.

4.2. ConvNet Feature Extraction with MRoI. For each landmark detected within an image, we extract its ConvNet feature based on MRoI in three steps:

(S1) MRoI pooling at multiple convolutional layers: from each RoI pooling layer of MRoI, the RoI pooling feature corresponding to the detected landmark is first obtained. As described above, this RoI pooling feature is a multidimensional array of size $6 \times 6 \times C$. So it is then vectorized. We denote the vectorized RoI pooling features from a RoI pooling layer as p_i, where $i = 3, 4, 5$.

(S2) ℓ_2-normalizing MRoI features layer by layer. With this normalization, we observed an improvement in localization accuracy in our experiments. Its definition is as follows:

$$p^i_{\ell_2} = \ell_2 \,\text{norm}\left(p^i\right), \quad i = 3, 4, 5. \tag{1}$$

(S3) Fusing normalized features across the MRoI layers. In order to improve the discrimination capacity of the final ConvNet feature, f_{MoI}, we fuse the ℓ_2-*normalized* features across the MRoI layers by concatenation:

$$f_{\text{MoI}} = p^3_{\ell_2} \shuffle p^4_{\ell_2} \shuffle p^5_{\ell_2}, \tag{2}$$

where $\shuffle$ means concatenation [34].

The ConvNet features of all detected landmarks are extracted as described above. Note that Steps 2 and 3 are considered to postprocess the output of MRoI layers obtained by Step 1.

5. Experimental Evaluation

To verify the effectiveness our proposed method, we performed experimental assessments on four datasets. In this section, the experimental setup is first provided from the aspects of testing datasets, compared methods, evaluation prototype, and evaluation metrics. Then, we will present experimental results with respect to the localization accuracy reflecting the discrimination capacity of extracted ConvNet features, the computational cost, and GPU memory efficiency.

5.1. Experimental Setup

5.1.1. Testing Datasets. In this paper, four popular visual localization datasets that exhibit typical variations in real-world visual localization applications were used to evaluate the performance. Main properties of all datasets are listed in Table 1. Sample images are shown in Figure 1.

(a) For the *UACampus* [35] dataset, two subsets captured at 06:20 and 22:15 were used, for the reason that they exhibit the greatest relative illumination change. To generate the ground truth, their images were manually matched.

(b) For the *St. Lucia* [36] dataset, two subsets collected at 08:45 on September 10, 2009, and at 15:45 on September 11, 2009, were used, because they exhibit the

greatest appearance changes caused by illumination and shadow. In our experiments, we used 1000 images uniformly sampled from each of the two subsets. To generate the ground truth, two images within 30 metres of the distance calculated based on GPS are considered to be the same position.

(c) For the *Nordland* [37] dataset, spring and winter subsets were used, because they exhibit the greatest variation in appearance caused by the seasonal changes. In our experiments, we used 1000 images uniformly sampled from each of the two subsets. The fact that these subsets have been time-synchronized was used to create the ground truth. In other words, an image with a given frame number in the spring subset corresponds to the image with the same frame number in the winter subset.

(d) The *Mapillary* [38] dataset was downloaded from Mapillary [39], an alternative service like Google Street View. It is regarded as an ideal platform that provides datasets for visual localization under everyday conditions [13]. To evaluate the performance under a significant viewpoint change as well as some appearance changes due to variations in the weather, we specifically downloaded 2028 image pairs with different viewpoints across several countries in Europe. Considering the fact that the GPS reading attached in each image is quite inaccurate, we first used the GPS readings to create the initial ground truth and then refined the initial ground truth manually.

5.1.2. Compared Methods. To evaluate the performance of our proposed method, we compared our method with the representative methods from the two aforementioned groups in the following experiments. Moreover, in order to examine the capability of our method with respect to generalization to different CNN models, we conducted experiments on AlexNet [16] and VGG-M1024 [17], two basic and popular CNN models. It is worth noting that our strategy to compare performance was to use the best performing single CNN layer (i.e., pool5) as a representative and compare it with the proposed MRoI method. The relative performance of single layer was established in an earlier study of ours [26], to justify the use of pool5 as the representative. For simplicity, in the rest of this paper we adopt the following notations to refer the two kinds of compared methods and two variants of our proposed method:

(i) *AlexNet* and *VGG-M1024* are two typical representatives of existing original image-based methods. They extract ConvNet features at the pool5 layer of CNN models of AlexNet and VGG-M1024, respectively. Note that resized subimages can be fed into the CNN models in one of two ways: (a) one-by-one and (b) in a batch. The two ways produce the same ConvNet features but require different computational costs and GPU memories, as we will discuss in the result section.

(ii) *FastRCNN-AlexNet* and *FastRCNN-VGG-M1024* are two typical representatives of exiting feature map-based methods. We directly ran Fast-RCNN [20] on CNN models of AlexNet and VGG-M1024 to extract ConvNet features at the pool5 layer.

(iii) *Two variants of our proposed method* are essentially the enhanced versions of above representatives of feature map-based methods. Accordingly, they are denoted as *MRoI-FastRCNN-AlexNet* and *MRoI-FastRCNN-VGG-M1024*. Our method extracts ConvNet features at not only the RoI_5 layer but also the added RoI_3 and RoI_4 layers. With respect to the values of C corresponding to the RoI3, RoI4, and RoI5 layer, those of *MRoI-FastRCNN-AlexNet* are 384, 384, and 256, respectively, and those of *MRoI-FastRCNN-VGG-M1024* are 512, 512, and 512, respectively.

5.1.3. Visual Localization Prototype. To verify the effectiveness of our proposed method in ConvNet landmark-based visual localization, we ran visual localization using the state-of-the-art framework proposed in [13]. Here we provide a brief summary of this framework for completeness. For more details regarding this framework, the reader is referred to [13]. Note that our feature extraction method is not specific to this framework and could easily be adapted to other frameworks for ConvNet landmark-based visual localization.

(i) Landmark detection: in [13], 100 landmarks per image were detected. Compared with [13], the difference of our experiments is that we detected landmarks using BING [40] instead of EdgeBoxes [41], which are two object proposal methods developed by the object detection community. We prefer BING for the following three reasons: (a) as has been demonstrated in [42], compared to EdgeBoxes, BING has slightly better repeatability, which is an important property for localization accuracy; (b) our previous experimental evaluation also shows that the localization accuracy achieved by BING is comparable with, or in some cases even better than, EdgeBoxes in the presence of severe environmental changes; and (c) BING has the speed advantage, which is a crucial consideration for real-time visual localization applications. In our test, BING is one order of magnitude faster than EdgeBoxes, with an execution time of 24 ms per image on a desktop PC.

(ii) ConvNet feature extraction and dimensionality reduction: to improve the efficiency in the subsequent image matching and storage, dimensionality reduction with an appropriate method is usually applied to the extracted ConvNet features. Following [13], the dimensions of all extracted ConvNet features in our experiments were reduced to 1024-D using Gaussian Random Projection (GRP) [43, 44]. Note that all extracted ConvNet features were $l2$-normalized before GRP was performed.

(iii) Image matching: the method of [13] uses bidirectional matching based on a linear nearest neighbour

search to find the matched landmarks. This matching strategy is optimized for accuracy and is therefore appropriate for comparing the discrimination capacity of extracted ConvNet features. Therefore, we have implemented the method of [13] for our evaluation. To ensure the validity of our experimental evaluation, our implementation has been verified to reproduce the results in [13].

For each dataset in Table 1, the first subset was considered as the query set of visual localization, and the second subset was used as the database (map) set. For each image in the query set, we find its best-matched image from the database set. Here we focus on finding the correct location without the customary verification using, for example, multiview geometry. Therefore, the corresponding ground truth is utilized to determine the correctness and then evaluate the localization accuracy.

Note that all the experiments in this paper were run on a desktop PC with eight cores CPU@4.00 GHz, 32 GB RAM memory, and a single GeForce GTX TITAN X GPU with 12 GB memory. In all the experiments, we use Caffe [45], which is a popular deep learning framework, to extract ConvNet features.

5.1.4. Evaluation Metrics. To evaluate the discrimination capacity of ConvNet features extracted by the proposed method for visual localization, we compare its localization accuracy with those of compared methods in terms of the following four popular metrics:

(a) *Precision-recall curve* is a standard criterion used to evaluate the localization accuracy for a range of confidence thresholds [1]. It is defined as follows: Precision = TP/(TP + FP), Recall = TP/(TP + FN), where TP, FP, and FN indicate the number of true positives, false positives, and false negatives, respectively. By varying the confidence threshold, we can produce a precision-recall curve. In general, a high precision over all recall values is desirable.

(b) *Maximum precision at 100% recall* is a popular criterion for directly evaluating the localization accuracy without using a confidence threshold. This criterion is useful, especially in environments under changing conditions or cross multiple different regions. In such an environment, an optimal confidence threshold is usually difficult to be predetermined. To avoid missing out the possible correct localization, in this case each query image always finds one best match from the database images without a confidence threshold.

(c) *Maximum recall at 100% precision* is a key metric to evaluate the performance of a method in cases of prioritizing avoidance of false positive localization.

(d) *Average precision (AP)* is useful when a scalar value is required to characterize the overall performance of visual localization [26, 46]. Average precision captures this property by computing the average of the precisions over all recall values of a precision-recall curve.

Besides, the *average running time per image* was measured to evaluate the computational efficiency. Finally, the actual cost of *GPU memory* was recorded to assess the GPU memory efficiency.

5.2. Localization Accuracy. In this section, we compare the localization accuracy of two variants of our method, *MRoI-FastRCNN-AlexNet/VGG-M1024*, with those of compared methods in terms of the first four above metrics. The corresponding results are shown in Figure 3 and Tables 2, 3, and 4. It can be generally observed from these results that, among all methods, the two variants of our method are the best or tying for the best across all of the testing datasets, and original image-based methods are the second or tying for the best, followed by feature map-based methods being the worst. The following observations can be further made.

(a) As can be seen in Figure 3 and Table 2, *FastRCNN-AlexNet/VGG-M1024* are comparable to *AlexNet/VGG-M1024* in environments without significant viewpoint changes, such as the UACampus, St. Lucia, and Nordland datasets. However, they are inferior to *AlexNet/VGG-M1024* in handling significant viewpoint change as exhibited in the Mapillary dataset. This demonstrates the problem we aim to solve in this paper.

(b) It can be clearly seen from Figure 3 and Table 3 that two variants of our method outperform original image-based methods, that is, *AlexNet/VGG-M1024*, in environments without significant viewpoint change (as exhibited in the UACampus, St. Lucia, and Nordland datasets) and are comparable to these original image-based methods in environments with significant viewpoint variation like the Mapillary dataset.

(c) One can clearly observe from Figure 3 and Table 4 that the precision-recall curves and the three numerical metrics produced by our *MRoI-FastRCNN-AlexNet/VGG-M1024* are higher than those of feature map-based methods, that is, *FastRCNN-AlexNet/VGG-M1024*, on all datasets. Moreover, the superiority of two variants of our method becomes more obvious in environments significant viewpoint change such as the Mapillary dataset. Considering the fact that feature map-based methods extract features from only one RoI pooling layer after the coarsest convolutional layer (i.e., the Conv5 layer), these comparison results demonstrate that using our MRoI method to fuse the features extracted from multiple RoI pooling layers is able to enhance the discrimination capacity of the final ConvNet features. As a result, our method improves the localization accuracy of feature map-based methods in environments with different kinds of conditional changes.

To qualitatively evaluate the matched results obtained by our method, Figure 1 shows examples of matched image pairs and corresponding matched landmark pairs produced by our *MRoI-FastRCNN-AlexNet*. Six images on each row

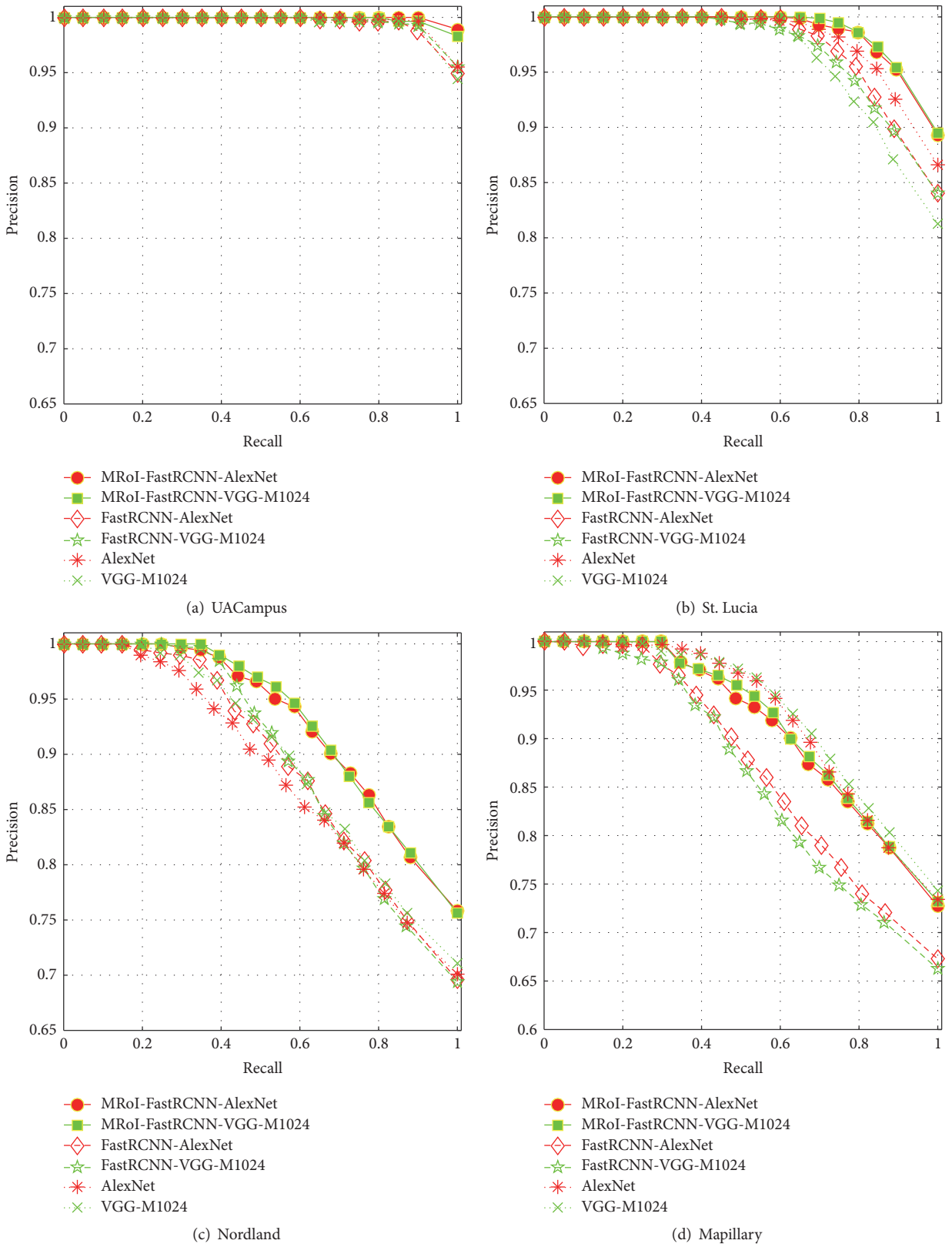


Figure 3: Comparisons of localization accuracy in terms of the precision-recall curve.

Table 2: Localization accuracy comparison of *FastRCNN-AlexNet* versus *AlexNet* and *FastRCNN-VGG-M1024* versus *VGG-M1024* in terms of maximum precision at 100% recall (Pr. at 100% Re.), maximum recall at 100% precision (Re. at 100% Pr.), and average precision (AP). The highest value with respect to each metric on each dataset is highlighted in bold. The middle *italic* values are the difference between *FastRCNN-AlexNet/VGG-M1024* and *AlexNet/VGG-M1024*.

Method	UACampus			St. Lucia			Nordland			Mapillary		
	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP
FastRCNN-AlexNet	94.9%	85.9%	99.2%	84.0%	61.2%	96.8%	69.0%	**23.9**%	**88.4**%	67.2%	16.0%	86.2%
	−0.6%	*−5.4%*	*−0.2%*	*−2.6%*	*−3.9%*	*−0.7%*	*−0.5%*	*+5.3%*	*+0.9%*	*−6.2%*	*−17.4%*	*−5.3%*
AlexNet	**95.5**%	**91.3**%	**99.4**%	**86.6**%	**65.1**%	**97.5**%	**69.5**%	18.6%	87.5%	**73.4**%	**33.4**%	**91.5**%
FastRCNN-VGG-M1024	**95.5**%	84.5%	**99.3**%	**84.0**%	**48.4**%	**96.4**%	68.8%	**35.3**%	**88.7**%	66.2%	14.1%	85.1%
	+1.0%	*−0.5%*	*+0.2%*	*+2.7%*	*+2.1%*	*+0.7%*	*−1.7%*	*+10.8%*	*−0.1%*	*−8.1%*	*−13.8%*	*−6.9%*
VGG-M1024	94.5%	**85.0**%	99.1%	81.3%	46.3%	95.7%	**70.5**%	24.5%	88.8%	**74.3**%	**27.9**%	**92.0**%

Table 3: Localization accuracy comparison of *MRoI-FastRCNN-AlexNet* versus *AlexNet* and *MRoI-FastRCNN-VGG-M1024* versus *VGG-M1024* in terms of maximum precision at 100% recall (Pr. at 100% Re.), maximum recall at 100% precision (Re. at 100% Pr.), and average precision (AP). The highest value with respect to each metric on each dataset is highlighted in bold. The middle *italic* values are the difference between *MRoI-FastRCNN-AlexNet/VGG-M1024* and *AlexNet/VGG-M1024*.

Method	UACampus			St. Lucia			Nordland			Mapillary		
	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP
MRoI-FastRCNN-AlexNet	**98.9**%	**98.6**%	**99.7**%	**89.3**%	**68.2**%	**98.2**%	**75.2**%	**34.0**%	**92.1**%	72.6%	31.2%	90.4%
	+3.4%	*+7.3%*	*+0.3%*	*+2.7%*	*+3.1%*	*+0.7%*	*+5.7%*	*+15.4%*	*+4.6%*	*−0.8%*	*−2.2%*	*−1.1%*
AlexNet	95.5%	91.3%	99.4%	86.6%	65.1%	97.5%	69.5%	18.6%	87.5%	**73.4**%	**33.4**%	**91.5**%
MRoI-FastRCNN-VGG-M1024	**98.3**%	**92.0**%	**99.6**%	**89.5**%	**74.4**%	**98.3**%	**75.0**%	**36.7**%	**92.3**%	73.1%	24.3%	90.7%
	+3.8%	*+7.0%*	*+0.5%*	*+8.2%*	*+28.1%*	*+2.6%*	*+4.5%*	*+12.2%*	*+3.5%*	*−1.2%*	*−3.6%*	*−1.3%*
VGG-M1024	94.5%	85.0%	99.1%	81.3%	46.3%	95.7%	70.5%	24.5%	88.8%	**74.3**%	**27.9**%	**92.0**%

Table 4: Localization accuracy comparison of *MRoI-FastRCNN-AlexNet* versus *FastRCNN-AlexNet* and *MRoI-FastRCNN-VGG-M1024* versus *FastRCNN-VGG-M1024* in terms of maximum precision at 100% recall (Pr. at 100% Re.), maximum recall at 100% precision (Re. at 100% Pr.), and average precision (AP). The highest value with respect to each metric on each dataset is highlighted in bold. The middle *italic* values are the difference between *MRoI-FastRCNN-AlexNet/VGG-M1024* and *FastRCNN-AlexNet/VGG-M1024*.

Method	UACampus			St. Lucia			Nordland			Mapillary		
	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP	Pr. at 100% Re.	Re. at 100% Pr.	AP
MRoI-FastRCNN-AlexNet	**98.9**%	**98.6**%	**99.7**%	**89.3**%	**68.2**%	**98.2**%	**75.2**%	**34.0**%	**92.1**%	**72.6**%	**31.2**%	**90.4**%
	+4.0%	*+12.7%*	*+0.5%*	*+5.3%*	*+7.0%*	*+1.4%*	*+6.2%*	*+10.1%*	*+3.7%*	*+5.4%*	*+15.2%*	*+4.2%*
FastRCNN-AlexNet	94.9%	85.9%	99.2%	84.0%	61.2%	96.8%	69.0%	23.9%	88.4%	67.2%	16.0%	86.2%
MRoI-FastRCNN-VGG-M1024	**98.3**%	**92.0**%	**99.6**%	**89.5**%	**74.4**%	**98.3**%	**75.0**%	**36.7**%	**92.3**%	**73.1**%	**24.3**%	**90.7**%
	+2.8%	*+7.5%*	*+0.3%*	*+5.5%*	*+26.0%*	*+1.9%*	*+6.2%*	*+1.4%*	*+3.6%*	*+6.9%*	*+10.2%*	*+5.6%*
FastRCNN-VGG-M1024	95.5%	84.5%	99.3%	84.0%	48.4%	96.4%	68.8%	35.3%	88.7%	66.2%	14.1%	85.1%

Table 5: Comparisons of average running time per image and the GPU memory cost between two variants of our method and compared methods for extracting ConvNet features. The total running time consists of the computational costs for preprocessing, going through the Caffe and postprocessing. Note that $AlexNet_b$/$VGG\text{-}M1024_b$ refer to the costs of computation and GPU memory when sending 100 detected landmarks into Caffe as a batch of 100. "—" means the computational cost is negligible. We can clearly see that the computation speed and GPU memory consumption of two variants of our method are close to those of *FastRCNN-AlexNet/VGG-M1024* and several times faster and fewer than those of $AlexNet_b$/$VGG\text{-}M1024_b$.

Method	GPU Memory (MB)	Average running time (ms)			
		Pre	Caffe	Post	Total
MRoI-FastRCNN-AlexNet	240	6.9	18.3	3.8	29.0
MRoI-FastRCNN-VGG-M1024	396		33.5	5.8	46.2
FastRCNN-AlexNet	218	6.9	13.3	—	20.2
FastRCNN-VGG-M1024	367		31.0	—	37.9
AlexNet	183	30.3	518.5	—	548.8
VGG-M1024	229		998.8	—	1029.1
$AlexNet_b$	880		115.9	—	146.2
$VGG\text{-}M1024_b$	1965		199.7	—	230.0

come from one dataset, and the three pairs represent correctly matched images by our method. It can be seen that the matched landmarks are correctly identified, even in environments with different changes. These results demonstrate that the ConvNet features extracted by our method have satisfactory discrimination capacity.

5.3. Computational Efficiency. To evaluate the computational efficiency, we report *average running time per image* of our method and compared methods for extracting ConvNet features in Table 5. For *AlexNet/VGG-M1024*, we also report their computational cost when sending 100 detected landmarks into Caffe as a batch of 100. The corresponding cost is denoted as $AlexNet_b$/$VGG\text{-}M1024_b$. Specifically, the table lists the breakdown of average running time per image, that is, the computational costs for preprocessing, going through Caffe and postprocessing. For the preprocessing, the cost of *AlexNet/VGG-M1024* and $AlexNet_b$/$VGG\text{-}M1024_b$ is the highest. The reason is as follows. Before feeding into the Caffe when using an original image-based method, subimages of detected landmarks need to first be cropped from the original input image according to their bounding boxes, and all cropped subimages are then resized to predefined dimensions to meet the requirement of the networks. As a result, the computational cost is as high as 30.3 ms per image. For the postprocessing, only our method needs several milliseconds for $l2$-normalizing the output from the MRoI pooling layer. In addition, some further observations can be made based on results in Table 5, as follows.

(a) *FastRCNN-AlexNet/VGG-M1024* are much faster than *AlexNet/VGG-M1024* (≈27/27 times) and even $AlexNet_b$/$VGG\text{-}M1024_b$ (≈seven/six times). This verifies our motivation, that is, feature map-based methods are greatly superior in computational efficiency to original image-based methods.

(b) Both variants of our method, *MRoI-FastRCNN-AlexNet/VGG-M1024*, are only approximately nine ms per image slower than corresponding feature map-based methods, *FastRCNN-AlexNet/VGG-M1024*. More importantly, the two variants achieve real-time computing efficiency, with average running times of 29.0 and 46.2 ms per image, respectively. Such high efficiency of the two variants is expected because they inherit the characteristic of feature map-based methods.

(c) Two variants of our method are 19 and 22 times faster than *AlexNet/VGG-M1024*, respectively, and both are approximately five times faster than $AlexNet_b$/$VGG\text{-}M1024_b$.

5.4. GPU Memory Efficiency. To evaluate the GPU memory efficiency, we report the *GPU memory cost* of our method and compared methods for extracting ConvNet features in Table 5. It can be seen that *AlexNet/VGG-M1024* require the minimal GPU memory (183/229 MB); however, $AlexNet_b$/$VGG\text{-}M1024_b$ consume the maximal GPU memory (880/1965 MB) because they send 100 detected landmarks into Caffe as a batch of 100 for speed-up. Compared with those of *AlexNet/VGG-M1024*, the GPU memory costs of our methods, *MRoI-FastRCNN-AlexNet/VGG-M1024*, increase by 57 MB and 167 MB, respectively. Nevertheless, our GPU memory consumption is still approximately four and five times less than those of $AlexNet_b$/$VGG\text{-}M1024_b$, respectively. In addition, compared with those of *FastRCNN-AlexNet/VGG-M1024*, our GPU memory consumption only increase by 22 MB and 29 MB, respectively, for the reason of using MRoI pooling layer. Perhaps most importantly, the GPU memory costs of our two variants still retain 240 MB and 396 MB, respectively. This means that our method is able to meet the requirement of visual localization on embedded systems or mobile devices with limited GPU resources.

6. Conclusion

In this paper, we have proposed a simple and efficient method of ConvNet feature extraction with multiple RoI pooling for

landmark-based visual localization of autonomous vehicles. The aim of our method is to deliver excellent localization accuracy comparable to original image-based methods while remaining the high computational efficiency of feature map-based methods. To achieve this, our method exploits the efficiency of RoI pooling and fuses the multilevel and multiresolution information from multiple RoI pooling layers to improve the discrimination capacity of extracted ConvNet features.

Experimental results on four popular visual localization datasets have demonstrated that the ConvNet features extracted by our method are discriminative to allow us to achieve the state-of-the-art localization accuracy and high computational efficiency with an average running time of 29.0 ms per image at the same time. In addition, our method is GPU memory efficient for mobile devices. Moreover, it is based on a pretrained CNN model without fine-tuning or retraining for ease of use, which is important for us to overcome the difficulty caused by the lack of training data in visual localization research. In short, the proposed MRoI method extracts highly discriminating ConvNet features efficiently, and the idea can be potentially extended in solving other vision tasks, such as object detection and object recognition.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

The authors gratefully acknowledge the support from the Hunan Provincial Innovation Foundation for Postgraduate (CX2014B021) and the Hunan Provincial Natural Science Foundation of China (2015JJ3018). This research is also supported in part by the Program of Foshan Innovation Team (Grant no. 2015IT100072) and by NSFC (Grant no. 61673125).

References

[1] S. Lowry, N. Süenderhauf, P. Newman et al., "Visual place recognition: a survey," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, 2016.

[2] M. Cummins and P. Newman, "FAB-MAP: probabilistic localization and mapping in the space of appearance," *International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[3] G. Singh and J. Kosecka, "Visual loop closing using gist descriptors in Manhattan world," in *Proceedings of the in IEEE International Conference on Robotics and Automation (ICRA) Omnidirectional Robot Vision Workshop*, 2010.

[4] H. Zhang, "BoRF: Loop-closure detection with scale invariant visual features," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation, ICRA 2011*, pp. 3125–3130, May 2011.

[5] N. Sünderhauf and P. Protzel, "BRIEF-Gist - Closing the loop by simple means," in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems: Celebrating 50 Years of Robotics, IROS'11*, pp. 1234–1241, September 2011.

[6] Y. Liu and H. Zhang, "Visual loop closure detection with a compact image descriptor," in *Proceedings of the 25th IEEE/RSJ International Conference on Robotics and Intelligent Systems, IROS 2012*, pp. 1051–1056, October 2012.

[7] M. J. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1643–1649, 2012.

[8] Y. Liu and H. Zhang, "Towards improving the efficiency of sequence-based SLAM," in *Proceedings of the 10th IEEE International Conference on Mechatronics and Automation (ICMA '13)*, pp. 1261–1266, Takamatsu, Japan, August 2013.

[9] M. Milford, "Vision-based place recognition: How low can you go?" *International Journal of Robotics Research*, vol. 32, no. 7, pp. 766–789, 2013.

[10] E. Pepperell, P. I. Corke, and M. J. Milford, "All-environment visual place recognition with SMART," in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation, ICRA 2014*, pp. 1612–1618, June 2014.

[11] T. Naseer, L. Spinello, W. Burgard, and C. Stachniss, "Robust visual robot localization across seasons using network flows," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 2564–2570, July 2014.

[12] Z. Chen, O. Lam, A. Jacobson, and M. Milford, "Convolutional neural network-based place recognition," in *Proceedings of the Australasian Conference on Robotics and Automation, ACRA 2014*, pp. 2–4, December 2014.

[13] N. Sünderhauf, S. Shirazi, A. Jacobson et al., "Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free," in *Proceedings of the 2015 Robotics: Science and Systems Conference, RSS 2015*, Rome, Italy, July 2015.

[14] P. Neubert and P. Protzel, "Local region detector + CNN based landmarks for practical place recognition in changing environments," in *Proceedings of the European Conference on Mobile Robots, ECMR 2015*, pp. 1–6, September 2015.

[15] P. Neubert and P. Protzel, "Beyond Holistic Descriptors, Keypoints, and Fixed Patches: Multiscale Superpixel Grids for Place Recognition in Changing Environments," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 484–491, 2016.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12)*, pp. 1097–1105, December 2012.

[17] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: delving deep into convolutional nets," in *Proceedings of the 25th British Machine Vision Conference (BMVC '14)*, Nottingham, UK, September 2014.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the in International Conference on Learning Representations (ICLR)*, 2015.

[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 580–587, Columbus, Ohio, USA, June 2014.

[20] R. Girshick, "Fast R-CNN," in *Proceedings of the 15th IEEE International Conference on Computer Vision (ICCV '15)*, pp. 1440–1448, December 2015.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, pp. 91–99, 2015.

[22] M. Labbe and F. Michaud, "Appearance-based loop closure detection for online large-scale and long-term operation," *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 734–745, 2013.

[23] M. Labbe and F. Michaud, "Online global loop closure detection for large-scale multi-session graph-based SLAM," in *Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2014*, pp. 2661–2666, September 2014.

[24] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[25] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[26] Y. Hou, H. Zhang, and S. Zhou, "Convolutional neural network-based image representation for visual loop closure detection," in *Proceedings of the IEEE International Conference on Information and Automation(ICIA)*, pp. 2238–2245, August 2015.

[27] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of ConvNet features for place recognition," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2015*, pp. 4297–4304, October 2015.

[28] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[29] J. Sivic and A. Zisserman, "Video google: a text retrieval approach to object matching in videos," in *Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV '03)*, pp. 1470–1477, October 2003.

[30] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Proceedings of the 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'07*, pp. 1–8, June 2007.

[31] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 3304–3311, June 2010.

[32] T. Naseer, M. Ruhnke, C. Stachniss, L. Spinello, and W. Burgard, "Robust visual SLAM across seasons," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2015*, pp. 2529–2535, October 2015.

[33] M. Shahid, T. Naseer, and W. Burgard, "DTLC: Deeply trained loop closure detections for lifelong visual SLAM," in *Proceedings of the RSS Workshop on Visual Place Recognition: What is it Good For?* 2016.

[34] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Fusion and binarization of CNN features for robust topological localization across seasons," in *Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2016*, pp. 4656–4663, October 2016.

[35] Y. Liu, R. Feng, and H. Zhang, "Keypoint matching by outlier pruning with consensus constraint," in *Proceedings of the 2015 IEEE International Conference on Robotics and Automation, ICRA 2015*, pp. 5481–5486, May 2015.

[36] A. J. Glover, W. P. Maddern, M. J. Milford, and G. F. Wyeth, "Fab-map + ratslam: appearance-based slam for multiple times of day," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 3507–3512, May 2010.

[37] P. Neubert, N. Sünderhauf, and P. Protzel, "Superpixel-based appearance change prediction for long-term navigation across seasons," *Robotics and Autonomous Systems*, vol. 69, no. 1, pp. 15–27, 2015.

[38] Y. Hou, H. Zhang, and S. Zhou, "Tree-based indexing for real-time ConvNet landmark-based visual place recognition," *International Journal of Advanced Robotic Systems*, vol. 14, no. 1, 2017.

[39] Mapillary, 2016, https://www.mapillary.com.

[40] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "BING: Binarized normed gradients for objectness estimation at 300 fps," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 3286–3293, June 2014.

[41] C. L. Zitnick and P. Dollár, "Edge boxes: locating object proposals from edges," in *Proceedings of the European Conference on Computer Vision (ECCV '14)*, pp. 391–405, Springer, 2014.

[42] J. Hosang, R. Benenson, P. Dollar, and B. Schiele, "What makes for effective detection proposals?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 4, pp. 814–830, 2016.

[43] S. Dasgupta, "Experiments with random projection," in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pp. 143–151, 2000.

[44] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: applications to image and text data," in *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '01)*, pp. 245–250, August 2001.

[45] Y. Jia, E. Shelhamer, J. Donahue et al., "Caffe: convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*, pp. 675–678, ACM, Orlando, Fla, USA, November 2014.

[46] W. Su, Y. Yuan, and M. Zhu, "A relationship between the average precision and the area under the ROC curve," in *Proceedings of the 5th ACM SIGIR International Conference on the Theory of Information Retrieval, ICTIR 2015*, pp. 349–352, September 2015.