Research Article

Image Geolocation Method Based on Attention Mechanism Front Loading and Feature Fusion

Huayuan Lu, Chunfang Yang, Baojun Qi, Ma Zhu, and Jingqian Xu

1Henan Provincial Key Laboratory of Cyberspace Situational Awareness, Zhengzhou 450001, China
2Zhengzhou Science and Technology Institute, Zhengzhou 450001, China

Correspondence should be addressed to Chunfang Yang; chunfangyang@126.com

Received 24 January 2022; Accepted 18 May 2022; Published 13 June 2022

Academic Editor: Ernestina Cianca

Copyright © 2022 Huayuan Lu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Image geolocation is an important technique for robotics and autonomous systems. The existing methods mainly extract local features from images directly and use global descriptors, which are aggregated by these local features, to retrieve candidate references from all references. Thus, the training efficiency is affected by the image noises and the accuracy is so limited that the further verification is extremely time consuming. To address these issues, this work proposes an image geolocation framework, which adds the noise filtering layer before local feature extraction. Based on this framework, an image geolocation method based on attention mechanism front loading and feature fusion is designed. In the noise filtering layer, the proposed method uses triplet attention to denoise images thus leading to higher training efficiency. In the feature aggregation layer, an improved SPP (spatial pyramid pooling) is designed to extract the local factors reflected by the position relationships among local features. Then, the local factors are incorporated with the global factors extracted by NetVLAD. The fused descriptors contain not only the statistic of the geometric elements but also the position relationships among them. The experimental results show that the proposed method outperforms NetVLAD in terms of the training convergence round and Recall@N (N = 1, 5, 10, 20); especially, the convergence round of Recall@5 reduces from 25 to 10, the convergence round of Recall@10 reduces from 25 to 7, Recall@1 increases from 79.45% to 84.01%, and Recall@5 increases from 90.10% to 92.81%.

1. Introduction

Image geolocation is a technique for geolocating a scene (in the image) or a camera based on the content of the image or its side information. This technique has been applied to many fields of IoT (Internet of things), such as autonomous driving [1–4], intelligent robotics [5], and virtual reality/augmented reality [6], and has received increasing attention in recent years [7–9].

Existing image geolocation methods can mainly be classified into image geolocation methods based on 3D point cloud [2, 10–17] and image geolocation methods based on image retrieval [18–21]. Image geolocation methods based on 3D point cloud use the SfM (structure from motion) [22] to construct a 3D point cloud from multiple images with different angles near the query (the image to be located) and then compare the constructed 3D point cloud with the local features of the query image point by point to obtain the exact shooting position and the estimation of the camera posture of the query image. Such methods have high positioning accuracy but require stronger conditions and a high computational cost. Image geolocation methods based on image retrieval extract the descriptors of the query and the reference (the image with known geographic location), then find the reference closest to the query according to the similarity between descriptors, and use its geographic location to estimate the location of the query. The geolocation precisions of these methods are greatly affected by the geographic labels of the references, but they are computationally efficient and robust and can be extended to global geolocation, which has received widespread attention [18–21]. Therefore, this paper focuses on image geolocation based on image retrieval.

The key to image geolocation based on image retrieval is the construction of image descriptors. According to different manners of image descriptor construction, the existing image geolocation methods based on image retrieval can be
generally classified into two classes: (1) image geolocation methods based on the heuristic descriptors and (2) image geolocation methods based on deep learning.

Image geolocation methods based on the heuristic descriptor firstly detect key points using algorithms such as Harris [23] and MSER (maximally stable extremal regions) [24], then aggregate or directly use SIFT (scale invariant feature transform) [25], SURF (speeded up robust features) [26], or other local features as image descriptors [10, 13, 27–36], and finally, match the query descriptor with the reference descriptors to geolocate the query. Earlier studies on image geolocation mainly focus on such methods. For example, Johns and Yang [27] clustered the SIFT features of all references, then built a feature tree, scanned the feature tree to find the references closest to the query image, and finally used a geometric verification method to determine the final position. In 2012, Gálvez-López and Tardos [37] constructed the BoVW (bag of visual words) [38] histogram of the BRIEF [39] local features of FAST [40] key points as the descriptor of this image and then represented it in binary form for fast image geolocation. Cao and Snavely [10] constructed the BoVW [38] histogram of SIFT features of an image as its descriptor, then clustered references by descriptors, trained a model for each cluster to determine whether the query belongs to that cluster, and finally geometrically verified it against the references within the cluster to geolocate the query. Zemene et al. [30] proposed the DSC (dominant set clustering) algorithm to dynamically select the N nearest neighbors of the query’s SIFT feature points from the feature point set of references and then determined the query’s geolocation based on the similarity and adjacency relationships between feature points. The abovementioned methods consider the similarity and position relationships between feature points and can achieve high geolocation accuracy, strong interpretability, and high computational efficiency. The construction of their descriptors mainly considers edges and corners in images with drastic texture changes. But in foggy and weakly illuminated environments, the extraction of the edge and corner features in images is easily affected by noises, so these methods perform poorly in such environments [41].

Deep learning-based image geolocation methods use lots of images to train deep networks to extract features as descriptors and then match the descriptors to geolocate the query [20, 42–52]. These methods are easily affected by the training data but have strong robustness, such as high adaptability to the changes of brightness, color, and viewpoint angle [41]. Since Sharif et al. [53] first introduced CNN (convolutional neural networks) to image geolocation in 2014, this kind of method has become the hotspot in the field of image geolocation. For example, Sünderhauf et al. [45] trained a network to extract images’ ROI (region of interest) features from images as region-level descriptors and matched them to geolocate the query. Anoosheh et al. [54] proposed ToDayGAN, which used GAN (generative adversarial network) to convert weakly illuminated images into normally illuminated images and then extracted its dense-VLAD [55] features as descriptors to geolocate the query. In 2016, Arandjelovic et al. [20] proposed NetVLAD, which extracted local features by VGG16 [56], then aggregated them as descriptors by a hot-plugging aggregation layer similar to VLAD (vector of locally aggregated descriptors) [57], and finally matched descriptors to geolocate the query. NetVLAD [20] is a milestone work in the field of image geolocation, since it first constructs a deep hot-plugging layer to aggregate local features as descriptors, making it possible to optimize the aggregation parameters automatically.

Based on NetVLAD [20], a number of improved methods have been proposed [2, 18, 19, 58]. For example, in 2020, Yu et al. [58] proposed SPE-VLAD, which divided an image into multiple nonoverlapping blocks, then concatenated the NetVLAD features of all blocks as descriptors, and matched them to geolocate the query. In 2021, Ge et al. [18] proposed an improved method of NetVLAD based on self-supervised learning, which used the descriptors obtained from NetVLAD to calculate the similarity between images and the similarity between images subregions, and then used the similarities to iteratively train the descriptor extraction network for image geolocation. The descriptors obtained by the abovementioned NetVLAD-based methods can effectively represent the whole image and achieve excellent geolocation performance. However, the global descriptors used by the abovementioned methods aggregate all local features and treat them equally, making task-irrelevant local features interfere with image geolocation. In addition, NetVLAD [20] represents the image with a sum of residuals, a global statistic, ignoring the positional relationships between local features.

To solve the problem that noisy local features tend to affect the accuracy of geolocation, researchers filter local features using attention mechanisms before aggregation and obtain more effective descriptors to improve geolocation accuracy [21, 59–65]. For example, in 2017, Kim et al. [59] proposed the CRN (contextual reweighting network), which upsampled filtered local features to get a weight matrix and reweighed local features, and then aggregated them as descriptors to geolocate the query. In 2018, Chen et al. [21] filtered the output of multiple layers in the feature extraction network and then constructed attention matrices to fuse the filtered results as descriptors for image geolocation. In 2021, Peng et al. [63] proposed SRALNet (semantic reinforced attention learning network), which clustered local features, and then aggregated the double-weighted residuals of local features and corresponding cluster centers as descriptors to geolocate the query. In 2021, Peng et al. [64] proposed APPSVR (attentional pyramid pooling of salient visual residuals for place recognition), which combined SRALNet with SPP (spatial pyramid pooling) [66] to fuse local features as descriptors for image geolocation. The abovementioned methods suppress task-irrelevant features and enhance task-relevant features to obtain more robust descriptors and can achieve better geolocation performance. To solve the problem of NetVLAD ignoring local factors, in 2021, Hausler et al. [19] proposed patch-NetVLAD, which retrieved candidate references using NetVLAD and reranked them based on the similarity between local features of two images to geolocate the query. However, on one hand, because the abovementioned methods directly extract local
features from the images with noisy signals, the feature extraction networks are difficult to converge quickly. On the other hand, because the candidate references are retrieved from all references by the global descriptors, the accuracy is so limited that the further verification in patchNetVLAD is extremely time consuming.

Therefore, this work is decided to improve the training efficiency and accuracy of image geolocation and mainly contains the following three contributions.

An image geolocation framework is proposed by adding the noise filtering layer before local feature extraction. The proposed framework consists of the noise filtering layer, the local feature extraction layer, the feature aggregation layer, and the descriptor matching layer.

According to the proposed framework, an image geolocation method based on attention mechanism front loading and feature fusion is designed. The noise filtering layer uses triplet attention [67] to denoise images thus improving the training efficiency. In the feature aggregation layer, the local factors extracted by an improved SPP are incorporated with the global factors extracted by NetVLAD. The fused descriptors contain not only the statistic of the geometric elements but also the position relationships among them.

SPP is improved by replacing the max grouping with GeM [68] when the number of SPP grids is $1 \times 1$. The improved SPP can extract the local factors reflected by the position relationships among local features.

The experimental results show that the proposed method can efficiently improve the accuracy of the model and the efficiency of training; especially, the convergence round of Recall@5 reduces from 25 to 10, convergence round of Recall@10 reduces from 25 to 7, Recall@1 increases from 79.45% to 84.01%, and Recall@5 increases from 90.10% to 92.81%.

The rest of the paper is organized as follows: Section 2 introduces the works strongly related to this paper, Section 3 details the proposed method, Section 4 shows the experimental results and analysis, and Section 5 concludes this paper.

2. Review of VLAD and NetVLAD

NetVLAD [20] is a milestone work in the field of image geolocation. It is derived from the classical VLAD [57] which extracts the descriptor as follows (see Figure 1):

(1) Extract the local features of the images

(2) Cluster the local features to obtain $K$ clusters, each of which represents a type of feature (e.g., representing the corners of a window)

(3) Calculate the sum of residuals between the features in each cluster and their corresponding cluster center, as shown in equation (1) as follows

$$v_k = \sum_{i=1}^{N} a_k(x_i)(x_i - c_k), \tag{1}$$

where a local feature is denoted by a vector $x_i = (x_{i,1}, x_{i,2}, \cdots, x_{i,D}), D$ denotes the dimension of the local feature, $a_k(x_i)$ is 1 if feature $x_i$ belongs to cluster $k$ and 0 if otherwise, $N$ is the number of local features, $c_k$ is the center of the $k$th cluster, and $v_k$ denotes the sum of residuals in the $k$th cluster

(4) Concatenate all $v_k, k = 1, 2, \cdots, K$ to obtain a single vector $\nu$ as the descriptor

NetVLAD improves the traditional VLAD to a hot-plugging layer of deep networks that automatically learns better parameters and then extracts more robust descriptors to improve geolocation performance.

In NetVLAD, the piecewise function $a_k(x_i)$ is replaced by a derivable form $a_k^*(x_i)$ as shown by equation (2), to preserve the following property of $a_k(x_i)$ as much as possible, which is that when feature $x_i$ is close to the $k$th cluster, the value of $a_k^*(x_i)$ is close to 1; otherwise, it is close to 0.

$$a_k^*(x_i) = \sum_k e^{-\alpha_k^2 - \alpha^2}, \tag{2}$$

where $\alpha$ is a parameter (positive constant) that controls the decay of the response with the magnitude of the distance. $||x_i - c_k||^2$ denotes the square of the L2 norm of $x_i - c_k$, namely, the square of the Euclidean distance between feature $x_i$ and the center of the $k$th cluster. Let $\omega_k = 2\alpha^2_k, b_k = -\alpha^2_k$, and then, equation (2) is transformed into a soft assignment of the following form:

$$\bar{a}_k^*(x_i) = \frac{e^{\omega_k x_i + b_k}}{\sum_k e^{\omega_k x_i + b_k}}. \tag{3}$$

It can be seen that the expression of $a_k^*(x_i)$ is derivable. Essentially, the cluster operation in VLAD is transformed to find proper functions $a_k^*(x_i), k = 1, 2, \cdots, K$, namely, to learn proper values of $\omega_k$ and $b_k$, which are the parameters in $k$ convolution kernels with the size of $1 \times 1$.

The final form of the NetVLAD layer is obtained by plugging the soft assignment (3) into the VLAD descriptor (1) resulting in

$$v_k = \sum_{i=1}^{N} \bar{a}_k^*(x_i)(x_i - c_k). \tag{4}$$

In general, NetVLAD extracts the descriptor as follows (see Figure 2):

(1) Extract local features $X = \{\cdots, x_{i-1}, x_i, x_{i+1}, \cdots\}$ using CNN

(2) Cluster local features to $K$ cluster centers using $K$ convolutions whose kernel size is $1 \times 1$

(3) Calculate the sum of residuals between the features in each cluster and their corresponding cluster center
3. Proposed Image Geolocation Framework

Image geolocation primarily utilizes the content information of images, but some task-irrelevant noises are inevitably introduced during image acquisition and processing. However, the existing deep learning-based image geolocation methods usually directly extract features from the query and references, which slows the learning speed of the model and affects the geolocation accuracy because of the interference of noises to the extracted features. Therefore, this section proposes an image geolocation framework by adding a noise filtering layer before local feature extraction. The proposed framework contains 4 superlayers as shown in Figure 3: noise filtering layer, local feature extraction layer, feature aggregation layer, and descriptor matching layer.

The noise filtering layer uses a filter to suppress the task-irrelevant signals and enhance the task-relevant signals in the image, which can accelerate the learning of the effective model and improve the geolocation accuracy.

The local feature extraction layer extracts local features from query and references. The local features can be traditional key point vector representations (such as SIFT [25] and SURF [26]) or feature maps extracted by the encoder of deep networks (such as VGG [56], ResNet [69], and EfficientNet [70]). In general, local features need to reflect the texture information, such as edges and corners, because this information can effectively distinguish the geographic location of the image.

The feature aggregation layer aggregates extracted local features into descriptors to geolocate the query. The aggregation methods can be traditional methods such as VLAD and BOVW or deep learning-based methods such as NetVLAD and GeM. The generated descriptors are used to retrieve images with similar content and should be robust to changes in a viewing angle and brightness.

The descriptor matching layer calculates the similarity between query and references to retrieve candidate references. Then, the geographic location of the candidate reference is regarded as the geographic location of the query. The mainstream similarity calculation methods include Euclidean distance, Manhattan distance, and cosine similarity.

4. Image Geolocation Based on Attention Mechanism Front Loading and Feature Fusion

In existing methods based on NetVLAD, the model convergence speed and accuracy are interfered with by the task-irrelevant noises in images and the positional relationships between local features are either ignored or used in an extremely time-consuming manner, such as the reranking of patch-NetVLAD. Therefore, under the guidance of the abovementioned framework, this section proposes an image geolocation method based on attention mechanism front loading and feature fusion, as shown in Figure 4.

In the proposed method, the encoder of VGG16 is used in the local feature extraction layer to extract local features and Euclidean distance is used in the descriptor matching...
layer to calculate the similarity between query and references, the same as NetVLAD and its improved versions such as patch-NetVLAD, CRN, and SARE [71]. Unlike existing image geolocation methods, triplet attention is plugged into as a noise filtering layer, to eliminate task-irrelevant noise while retaining task-relevant information of image contents. In the feature aggregation layer, the local features are aggregated by NetVLAD and an improved SPP; then, the aggregated results are concatenated as the descriptor.

In the following 2 subsections, the noise filtering layers and the feature aggregation layer will be described in detail.

4.1. Noise Filtering Layer Based on Triplet Attention. The noise filtering layer uses a filter to suppress the task-irrelevant signals in the image and enhance the task-relevant signals, which can accelerate the learning of the effective model and improve the geolocation accuracy.

The existing image noise filtering methods mainly use various correlations in the image to reduce the noises and keep the image contents. Among the existing methods for capturing various correlations in images, the attention mechanism models the correlation among information in the channel domain, spatial domain, or temporal domain, to effectively filter noises and achieve excellent performance on many tasks such as image classification, object detection, and semantic segmentation [72]. Scholars have proposed many attention mechanisms such as SKNet [73], SENet [74], residual attention network [75], CBAM [76], and triplet attention [67]. Among these methods, the triplet attention [67] can model correlations in both the spatial and channel domain of images with almost no parameter increase and can achieve excellent performance. Therefore, the proposed noise filtering layer uses it to denoise images before extracting local features to eliminate the influence of noise information and improve the feature effectiveness.
Let $x_{G \times H \times W}$ denote the input image, where $G, H, W$ are the number of channels, height, and width of the input image, respectively. The specific steps of the adopted triplet attention are as follows (see Figure 5):

1. Perform the dimensional rotation (permutation) operation on the input image, to obtain 2 tensors $x_{H \times G \times W}$ and $x_{W \times H \times G}$

2. Perform the following operations on $x_{G \times H \times W}$ to obtain the weighted tensors $\tilde{x}_{G \times H \times W}$
   
   (i) Perform max pooling and avg pooling (mean pooling) on $x_{G \times H \times W}$, which can obtain two tensors whose size is $1 \times H \times W$, and then, stack them and get $x_{2 \times H \times W}$
   
   (ii) Set a convolution whose kernel size, step value, and padding value are $7 \times 7, 1,$ and $3$, respectively, and perform it on the $x_{2 \times H \times W}$, which can obtain $x_{1 \times H \times W}'$
   
   (iii) Perform batch normalization and sigmoid on $x_{1 \times H \times W}'$, which can obtain the weight matrix $P_{1 \times H \times W}$, and then, perform pointwise multiplication on it by the tensor $x_{G \times H \times W}$, which can obtain weighted tensor $\tilde{x}_{G \times H \times W}$

3. Perform the same steps in (2) on $x_{H \times G \times W}$ and $x_{W \times H \times G}$, to obtain the other 2 weighted tensors $\tilde{x}_{H \times G \times W}$ and $\tilde{x}_{W \times H \times G}$

4. Inverse permute $\tilde{x}_{H \times G \times W}$ and $\tilde{x}_{W \times H \times G}$ to obtain 2 tensors $\tilde{x}_{G \times H \times W}$ and $\tilde{x}_{G \times H \times W}^\prime$, and then, perform element-wise addition and average operations on the 3 tensors $\tilde{x}_{G \times H \times W}$, $\tilde{x}_{G \times H \times W}^\prime$, and $\tilde{x}_{G \times H \times W}$, to obtain the filtered tensor $\tilde{x}_{G \times H \times W}^\prime$, whose size is the same as the input image

4.2. Feature Aggregation Layer Based on the Fusion of Global and Local Features. In the mainstream NetVLAD, a global statistical vector, containing the sums of residuals, is used as the descriptor, which weakens the role of the local factor. Recent excellent works, such as DOLG [77], DELG [78], and patch-NetVLAD, argue that the introduction of local factors can improve the retrieval and geolocation performance. However, directly using local features as descriptors will bring not only high computational complexity but also poor robustness. Therefore, the proposed feature aggregation layer incorporates the aggregation method in NetVLAD and an improved SPP method to maintain the local factor.

SPP not only can maintain position relationships among local features but can also pool the input feature maps with different sizes, while having high computational efficiency. Its effectiveness has been proven in many retrieval tasks [68, 77–79]. SPP grids a feature map into equally nonoverlapping parts and performs max pooling for every part. But in the original SPP, the importance difference among feature maps is ignored, which is proven useful by GeM for image retrieval and geolocation tasks [68, 77–79]. When the SPP grid number is $1 \times 1$, the pooling operation is similar...
to GeM, which can reflect the importance of different feature maps. Additionally, GeM has a simple and effective calculation method. Thus, in the proposed method, SPP is improved by replacing the max grouping with GeM when the number of SPP grids is $1 \times 1$.

Let $x_{g,h,w}$ denote the input feature maps, where $g, h, w$ denotes the number of feature maps, height, and width of each feature map, respectively. The feature maps play the role of local features. As shown in Figure 6, the outputs of improved SPP and NetVLAD are concatenated as the final descriptor. The detailed steps are described as follows.

1. Use NetVLAD to aggregate the input feature maps as output $f^0$.
2. Perform GeM pooling on all feature maps to obtain $f^1$ by
   \[ f^1 = \left[ f^1_1, \cdots, f^1_k, \cdots, f^1_g \right]_T, \quad f^1_k = \left( \frac{1}{|F_k|} \sum_{x \in F_k} x^p_k \right)^{1/p_k}, \]
   where $p_k$ is the learnable parameter, which could also be adjusted manually, and $f^1_k$ is the pooling result of the $k$th feature map calculated by equation (4).
3. Perform the following operations on $x_{g,h,w}$ to obtain the pooling result $f^L$, $L = 2, 3, 4$
   
   (i) Divide each feature map in $x_{g,h,w}$ into $L \times L$ equally nonoverlapping feature submaps. When the height $h$ (or width $w$) is not an integral multiple of $L$, the feature maps should be 0 padded until the height (or width) is the minimal integral multiple of $L$ not less than $h$ (or $w$).
   (ii) Perform max pooling on each feature submap, arrange $L \times L$ max pooling results of $L \times L$ feature submaps as a column vector, and then, concatenate $g$ column vectors obtained from $g$ feature submaps as a vector $f^L$ with the length of $(g \times L \times L)$.

4. Finally, concatenate $f^0, f^1, f^2, f^3, f^4$ and perform PCA on the concatenated feature to obtain the final descriptor.

5. Experimental Results and Analysis

5.1. Experimental Setup. The proposed method was evaluated in the following setup as shown in Table 1. In the noise filtering layer, the learnable parameters of triplet attention were set as the corresponding line in Table 1. The feature extraction layer retained the NetVLAD setting, which used a partially pretrained VGG16 encoder to extract local features. That is, the last "ReLU & MaxPooling" of VGG16 pretrained on ImageNet was removed and other parts of it were used. Then, only the parameters of the last three convolutional layers were fine-tuned while other parameters were kept unchanged during training. In the feature aggregation layer, the parameters of SPP improved by GeM, $p_k, k = 1, 2, \cdots, 512$ were set as $3.0$. In the descriptor matching layer, Faiss [80] was used to accelerate the feature matching process. The training procedure used query image $q$, positive reference $p$, and 10 negative references $\{n_1, n_2, \cdots, n_{10}\}$ to form 10 triplets $\{q, p, n_1\}, \{q, p, n_2\}, \cdots, \{q, p, n_{10}\}$, then used triplet loss to calculate loss, and used SGD to optimize the model.

The experimental dataset Pittsburgh 30K [81] contains 51840 Google Street View images captured at different times in the same year, which well simulates the real-application scenario. This dataset was roughly equally divided into 3 parts as train, validation, and test sets, and the number of queries and references contained in each part is shown in Table 2.

5.1.1. Evaluation Metrics. The performance of the proposed method was compared with the classical NetVLAD method in terms of Recall@N and the number of convergence rounds.
5.1.2. Recall@N. The Recall@N [81] of M queries is calculated in the manner described by the following formula.

\[
\text{Recall@N} = \frac{\sum_{j=1}^{M} \min \left( \sum_{i=1}^{N} \alpha_j (r_{ij}), 1 \right)}{M},
\]

\[
\alpha_j (r_{ij}) = \begin{cases} 
0, & \text{if } \text{dis}(q_j, r_{ij}) > 25 \text{ m}, \\
1, & \text{if } \text{dis}(q_j, r_{ij}) \leq 25 \text{ m}.
\end{cases}
\]

If the distance between the jth query image \(q_j\) and the ith closest reference \(r_{ij}\) is less than 25 m, the output of \(\alpha_j (r_{ij})\) is 1 and otherwise 0. Taking query \(q_j\) as an example, \(R = \{r_{ij} | i = 1, \ldots, N\}\) denotes top N closest references of \(q_j\), if the distances between the \(R\) and \(q_j\) are all greater than 25 m, then, the output of \(\sum_{i=1}^{N} \alpha_j (r_{ij}), 1\) is 0, and the output of \(\min \left( \sum_{i=1}^{N} \alpha_j (r_{ij}), 1 \right)\) is also 0. Conversely, if the distance between any element of \(R\) and \(q_j\) is less than 25 m, the output of \(\sum_{i=1}^{N} \alpha_j (r_{ij}), 1\) is 1.

Equations (5) and (7) mean that if one of the first N candidate locations of the query is correct, then, the geolocation result is considered correct. And the Recall@N value is the percentage of correctly geolocated queries.

5.1.3. Training Convergence Round. Let \(A_E \in (0, 1)\) denote the Recall@N of the \(E\)th round of training epoch \(E\). If max \((A_{E+1}, A_{E+2}, \ldots, A_{E+10}) - A_E \leq 0.0005\), then, the convergence round is considered as \(E\). The lower the value of \(E\), the higher the training efficiency of the model.

5.2. Effectiveness Test of the Noise Filtering Layer. The 1st and 6th rows of Table 3 show the Recall@1, Recall@5, Recall@10, and Recall@20 of the original NetVLAD and the version improved by adding the noise filtering layer before local feature extraction. It can be seen that compared with the original NetVLAD, the improved version gets better accuracy. If performing PCA on the aggregated feature, the superiority of adding noise filtering layer is more significant.

Table 1: Experimental setup.

<table>
<thead>
<tr>
<th>Layers/others</th>
<th>Setup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noise filtering layer</td>
<td>Triplet attention, conv kernel size 7 × 7, stride 1, padding length 3</td>
</tr>
<tr>
<td>Feature extract layer</td>
<td>The same VGG16 as is in NetVLAD, only fine-tune the last 3 conv layers</td>
</tr>
<tr>
<td>Feature aggregation layer</td>
<td>NetVLAD &amp; SPP improved by GeM, (p_0 = 3.0), SPP’s (L = 1, 2, 3, 4)</td>
</tr>
<tr>
<td>Descriptor matching layer</td>
<td>European distance implemented by Faiss</td>
</tr>
<tr>
<td>Training strategies</td>
<td>Loss function triplet loss, optimizer SGD with momentum = 0.9</td>
</tr>
<tr>
<td>Experiment platform</td>
<td>Ubuntu20.04, CUDA 11.0, PyTorch 1.7.1, Faiss 1.7.1</td>
</tr>
</tbody>
</table>

Table 2: The experimental dataset splits.

<table>
<thead>
<tr>
<th>Parts</th>
<th>Number of queries</th>
<th>Number of references</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>7416</td>
<td>10000</td>
</tr>
<tr>
<td>Valid</td>
<td>7608</td>
<td>10000</td>
</tr>
<tr>
<td>Test</td>
<td>6816</td>
<td>10000</td>
</tr>
</tbody>
</table>
as shown in the 2nd, 3rd, 4th, 5th, 7th, 8th, 9th, and 10th rows of Table 3. The reason should be that the noise filtering layer effectively filters the noises irrelevant to the image geolocation task.

Grad-CAM (gradient-weighted class activation mapping) [82] was used to visualize the comparison of the images before and after the noise filter layer on the R, G, B components of them, as shown in Figure 7. It can be seen that the edges and corners attract more attention after filtering and the focused effective areas are more abundant. The reason should be that in the original NetVLAD, due to the noises, the model may not be able to distinguish the local features in these areas from them in the noisy areas, resulting in the loss of local features important to the image geolocation. But adding the noise filtering layer suppresses the inference of noises in advances, which makes the effective areas attract more attention and the model easier to learn high-quality local features.

Figure 8 shows the training procedure of origin NetVLAD and the version improved by adding the noise filtering layer before local feature extraction. It can be seen that the improved version outperforms the origin NetVLAD in terms of the training convergence round, especially on Recall@10 and Recall@20. The reason may be that image denoising by triplet attention can eliminate the inference of noises, make the model focus on important areas, and then accelerate the learning of features.

The experimental results show that the noise filtering layer is effective in improving the image geolocation performance.

5.3. Effectiveness Test of the Proposed Method. The last 5 rows of Table 3 show the Recall@1, Recall@5, Recall@10, and Recall@20 of the proposed method, which improved by adding the noise filtering layer before local feature extraction and incorporating local factor into the aggregation layer. Experimental results show that the proposed method has a significant improvement in geolocation accuracy, especially, when PCA is used to reduce the dimensionality of the final descriptor. The performance improvement maybe because the proposed feature aggregation layer reduces the task-irrelevant components in the features by combining NetVLAD with SPP (improved by GeM) and performing PCA. The NetVLAD is used to compute the global factors, and the SPP improved by GeM is used to compute the local factors, which reflect the position relationships of local features.

Moreover, the computational efficiency also has been improved by PCA. That is, the computational efficiency and geolocation performance have both been improved by the proposed method.

Figure 9 shows an image geolocation example of the original NetVLAD method and the proposed method. It can be seen that the geometric elements in the references retrieved by the original NetVLAD have stronger similarities to those of the query and the geometric elements in the references retrieved by the proposed method not only have stronger similarity to them of the query but also own similar position relationships to them of the query. And the proposed method gets a more accurate image geolocation result. This is because the original NetVLAD only extracts the geometric elements, but the proposed method extracts both the global factors and the local factors, which reflect the position relationships among the geometric elements.

Furthermore, it can be seen in Figure 8 that the proposed method still remains the advantage of the version improved by adding a noise filtering layer in terms of training convergence rounds.

In a word, the proposed method outperforms classical NetVLAD in both geolocation accuracy and training speed.

### Table 3: The performance comparisons between the proposed method and NetVLAD on the Pitts 30k [81].

<table>
<thead>
<tr>
<th>Noise filtering layer</th>
<th>Feature aggregation layer</th>
<th>PCA</th>
<th>Dimension</th>
<th>R@1</th>
<th>R@5</th>
<th>R@10</th>
<th>R@20</th>
</tr>
</thead>
<tbody>
<tr>
<td>×</td>
<td>NetVLAD</td>
<td>×</td>
<td>32768</td>
<td>79.45</td>
<td>90.10</td>
<td>92.77</td>
<td>95.19</td>
</tr>
<tr>
<td>×</td>
<td>NetVLAD</td>
<td>√</td>
<td>512</td>
<td>77.52</td>
<td>89.41</td>
<td>92.59</td>
<td>95.39</td>
</tr>
<tr>
<td>×</td>
<td>NetVLAD</td>
<td>√</td>
<td>1024</td>
<td>78.92</td>
<td>90.10</td>
<td>92.87</td>
<td>95.48</td>
</tr>
<tr>
<td>×</td>
<td>NetVLAD</td>
<td>√</td>
<td>2048</td>
<td>79.55</td>
<td>90.42</td>
<td>92.97</td>
<td>95.38</td>
</tr>
<tr>
<td>×</td>
<td>NetVLAD</td>
<td>√</td>
<td>4096</td>
<td>79.50</td>
<td>90.23</td>
<td>92.9</td>
<td>95.19</td>
</tr>
<tr>
<td>×</td>
<td>NetVLAD</td>
<td>√</td>
<td>32768</td>
<td>80.99</td>
<td>91.09</td>
<td>93.57</td>
<td>95.48</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD</td>
<td>√</td>
<td>512</td>
<td>80.34</td>
<td>91.37</td>
<td>93.97</td>
<td>95.76</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD</td>
<td>√</td>
<td>1024</td>
<td>81.24</td>
<td>91.61</td>
<td>94.07</td>
<td>95.77</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD</td>
<td>√</td>
<td>2048</td>
<td>81.41</td>
<td>91.62</td>
<td>93.85</td>
<td>95.85</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD</td>
<td>√</td>
<td>4096</td>
<td>81.29</td>
<td>91.37</td>
<td>93.63</td>
<td>95.6</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD</td>
<td>√</td>
<td>48128</td>
<td>83.67</td>
<td>92.36</td>
<td>94.16</td>
<td>95.77</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD + SPP^GeM</td>
<td>×</td>
<td>512</td>
<td>82.88</td>
<td>92.39</td>
<td>94.91</td>
<td>96.24</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD + SPP^GeM</td>
<td>√</td>
<td>1024</td>
<td>83.73</td>
<td>92.81</td>
<td>94.91</td>
<td>96.20</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD + SPP^GeM</td>
<td>√</td>
<td>2048</td>
<td>84.01</td>
<td>92.78</td>
<td>94.81</td>
<td>96.11</td>
</tr>
<tr>
<td>√</td>
<td>NetVLAD + SPP^GeM</td>
<td>√</td>
<td>4096</td>
<td>83.92</td>
<td>92.62</td>
<td>94.5</td>
<td>95.95</td>
</tr>
</tbody>
</table>

"×" means that the operation corresponding to the column name is not applied to the model, and "√" means the opposite. "Dimension" denotes the dimension of the final descriptor; "Recall@N" denotes Recall@N.
Figure 7: The Grad-CAM visualizations for the original NetVLAD and the version adding noise filtering layer. (a) The original image, (b) the R, G, B components of the original image, (c) the heat map of the original NetVLAD, (d) the R, G, B components of the image after the noise filtering layer, and (e) the heat map of the version improved by adding the noise filtering layer.
Recall@1

Figure 8: Continued.
Figure 8: The training performance comparison of classical NetVLAD and the proposed method.
6. Conclusions

In this work, we have proposed a novel image geolocation framework by adding the noise filtering layer before feature extraction. Based on this framework, an image geolocation method based on attention mechanism front loading and feature fusion has been designed. Unlike original NetVLAD, our method uses triplet attention to denoise images and gets more effective descriptors by considering not only global factors but also local factors reflected by the relationships of local features extracted by an improved SPP. Experimental results show that our proposed method outperforms the original NetVLAD in terms of Recall@N(N = 1, 5, 10, 20) and training convergence round.

Research works such as DELG and PatchNetVLAD show that the accuracy can be further improved by geometric verification. However, the verification procedure is extremely time consuming and its time complexity is closely related to the number of references under the same recall rate, viz., the value of N in Recall@N. Therefore, in future works, we will combine the proposed method with geometric verification, to reduce the time complexity and improve the accuracy of methods, such as DELG and patch-NetVLAD. Furthermore, we will try to extend the proposed method to other fields related to image retrieval.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

We thank Hao Li for helpful discussions. This work is supported by the National Natural Science Foundation of China (Grant nos. 61872448, 61772549, and U1804263) and Science and Technology Research Project of Henan Province (no. 222102210075), China.

References


Wireless Communications and Mobile Computing


[43] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, “Particular object retrieval with integral max-pooling of CNN activations,” in In


