TransVLAD: Multi-Scale Attention-Based Global Descriptors for Visual Geo-Localization

Yifan Xu[1], Pourya Shamsolmoali[1,2], Eric Granger[3], Claire Nicodeme[4], Laurent Gardes[4], Jie Yang[1]
[1]Shanghai Jiao Tong University, [2]East China Normal University, [3]ETS Montreal, [4]SNCF Paris
[1]{yifanxu, pshams, jieyang}@sjtu.edu.cn, [2]Eric.Granger@etsmtl.ca,
[3]{claire.nicodeme, l.gardes}@sncf.fr

Abstract

Visual geo-localization remains a challenging task due to variations in the appearance and perspective among captured images. This paper introduces an efficient TransVLAD module, which aggregates attention-based feature maps into a discriminative and compact global descriptor. Unlike existing methods that generate feature maps using only convolutional neural networks (CNNs), we propose a sparse transformer to encode global dependencies and compute attention-based feature maps, which effectively reduces visual ambiguities that occurs in large-scale geo-localization problems. A positional embedding mechanism is used to learn the corresponding geometric configurations between query and gallery images. A grouped VLAD layer is also introduced to reduce the number of parameters, and thus construct an efficient module. Finally, rather than only learning from the global descriptors on entire images, we propose a self-supervised learning method to further encode more information from multi-scale patches between the query and positive gallery images. Extensive experiments on three challenging large-scale datasets indicate that our model outperforms state-of-the-art models, and has lower computational complexity. The code is available at: `https://github.com/wacv-23/TVLAD`.

1. Introduction

Visual geo-localization is an important task with a broad range of applications in, e.g., automatic driving [14, 18] and robot navigation [20, 26]. Given the rapid development of deep CNNs [11, 33, 35] and vision transformers (ViTs) [7, 25, 38], more discriminant and comprehensive feature representation can be extracted from image data. Therefore, image-based localization has attracted growing attention [1, 12, 17, 23, 31, 41]. In this paper, the visual geo-localization problem is treated as an image retrieval task, which aims to estimate the geospatial location of a query image by matching it with a gallery of geo-tagged images.

The fundamental problem consists in learning discriminative representations from images with variations in appearance and perspective [1, 12]. To address this problem, state-of-the-art methods [1, 17, 23, 41] typically utilize CNNs with NetVLAD [1], inspired by the Vector of Locally Aggregated Descriptors (VLAD) [19], to learn feature representations. However, the locality assumption of the CNNs hinders their performance in complex scenes, where there may be visual complexities such as occlusions and transient objects (*e.g.* trees, cars and pedestrians). In contrast, when visual signals are ambiguous or incomplete, the human visual system uses not only local information, but also the global context for accurate predictions. Recent works [9, 12] therefore, exploit ViTs [7, 25, 38] to solve image retrieval tasks. However, ViTs perform poorly on our task because: (1) transformers lack inductive biases of CNNs, such as translation equivariance and locality [7]; and (2) the geo-localization datasets only have noisy GPS tags, and the current approaches design geo-localization models through weakly-supervised learning [1].

Motivated by the above observations, we consider retaining the CNN architecture with NetVLAD [1]. However, existing methods [1, 17, 23, 41] use VGG-16 [33], which is intractable on compact embedded or mobile devices due to model complexity. Therefore, we adopt a lightweight CNN, like MobileNetV3 [13] as the CNN backbone, and introduce an efficient TransVLAD module. More specifically, our proposed TransVLAD is composed of a transformer module, along with a grouped VLAD layer. The transformer improves global contextual reasoning, and attention-based feature maps can be produced by exploiting its self-attention properties. This can effectively reduce visual ambiguities or incompleteness in large-scale geo-localization tasks. Moreover, the positional embedding mechanism in transformers enables our TransVLAD module to encode corresponding geometric configurations between query and gallery images during training. Considering that ViTs are computationally intensive [7, 38, 25, 12], we propose a sparse transformer

based on the idea of sparse attention [4]. In addition, inspired by [21], the efficiency of our grouped VLAD layer is improved by decomposing high-dimensional feature vectors into groups of relatively low-dimensional vectors before performing VLAD aggregation.

The other key problem lies with the geo-localization datasets [1, 36, 37] that have noisy GPS tags, and can only be used for training in a weakly-supervised setting. Therefore, state-of-the-art methods [1, 17, 23, 41] use the most similar ones in the representation space as the first-ranked positive images. In those works, the models are trained to force feature representations from the queries images to be close to those from the first-ranked positive gallery images. However, these methods are not robust enough on large-scale datasets due to variations in appearance, and limited information in the first-ranked positive images. Therefore, we considered using lower-ranked positive images to provide more diverse and representative information, and train robust models. Lower-ranked positive images may have fewer overlapping patches with the query images, which can result in inaccurate learning of global descriptors if only whole images are used. To address the above issue, we propose a self-supervised learning method to encode more information from multi-scale patches between the query and positive gallery images. Specifically, we split and merge feature maps from the query and positive images into multi-scale patches, and produce corresponding local descriptors through VLAD layers. By matching the local descriptors between the query and positive gallery images, we estimate the multi-scale similarity scores that provide refined self-supervisions for training. We also separate the training into several generations to progressively learn to provide more accurate multi-scale similarity scores.

The key contributions of this paper can be summarized as follows. (1) An efficient TransVLAD module is introduced that aggregates attention-based feature maps into a discriminative and compact global descriptor. We propose a sparse transformer to encode global dependencies, and employ the positional embedding mechanism to encode geometric configurations between query and gallery images. For efficiency, we also introduce a grouped VLAD layer. (2) A self-supervised learning method is proposed to encode more information from multi-scale patches between the query and positive gallery images, rather than only learning from the global descriptors extracted from the whole images. (3) An extensive set of experiments on three challenging large-scale datasets [1, 36, 37] indicating that our proposed TransVLAD model outperforms state-of-the-art models, with lower computational complexity.

2. Related Work

VLAD Layers. The Vector of Locally Aggregated Descriptors (VLAD) [2, 19] is one of the early global image descriptor to aggregate local features based on learnable semantic centers. Based on the dense sampling of an image grid, DenseVLAD [36] aggregates local features into a compact feature representation. NetVLAD [1] was proposed to change the original VLAD layer [19] into a learnable layer, which easily can be added to other network structures. For video classification, NeXtVLAD [21] decomposes the input features into groups to improve efficiency by aggregating temporal information. SPE-VLAD [42] exploits the spatial pyramid structure of the images to enhance the VLAD, which can make the feature representation reflect the structural information of the images. Moreover, Patch-NetVLAD [10] is a new method that combines the advantages of both local and global descriptors by exploiting patch-level features from NetVLAD residuals. Distinct from existing approaches only considering aggregating local features, this paper introduces TransVLAD, which both improves global contextual reasoning and aggregates a discriminative and compact global descriptor.

Vision Transformers (ViTs). The Transformer [39] is first model proposed for machine translation, and has developed into saved variants that provide high performances in various Natural Language Processing (NLP) tasks [3, 6, 29, 30]. For image classification, the ViT [7] applies a Transformer Encoder on non-overlapping medium-sized image patches. DeiT [38] uses knowledge distillation to effectively train the ViT with small datasets. To use ViTs for more general-purposes of computer vision, Swin Transformer [25] proposes a hierarchical transformer which exploits shifted windows to compute feature representations. For cross-view geo-localization, L2LTR [12] proposes a self-cross attention mechanism to interact features between adjacent layers. Given the power of ViTs in global contextual reasoning, we propose a novel TransVLAD module by combining the ViT and VLAD layer. Besides, this paper introduces a sparse transformer since the ViTs are computationally intensive.

Image-Based Localization (IBL). The IBL methods can be subdivided into two directions, the 2D image retrieval methods [1, 12, 17, 23, 41] and 2D-3D matching methods [22, 31, 32]. In this paper, we consider the visual geo-localization problem as a 2D image retrieval task. With the development of deep CNNs [11, 33, 35], features extraction from images has become more dense. Thus, NetVLAD [1] exploits the dense features to effectively produce a global descriptor based on learnable semantic centers for visual geo-localization. Based on the CNNs with NetVLAD, several works have proposed some representation learning models to further improve the geo-localization performance, such as Contextual Reweighting Network [17], SARE loss [23] and HAF networks [41]. Since ViTs [7, 38, 25] have attracted much attentions, recent works [12, 9, 40] exploit ViTs instead of CNNs to deal with image retrieval task, and achieve higher matching accuracy.

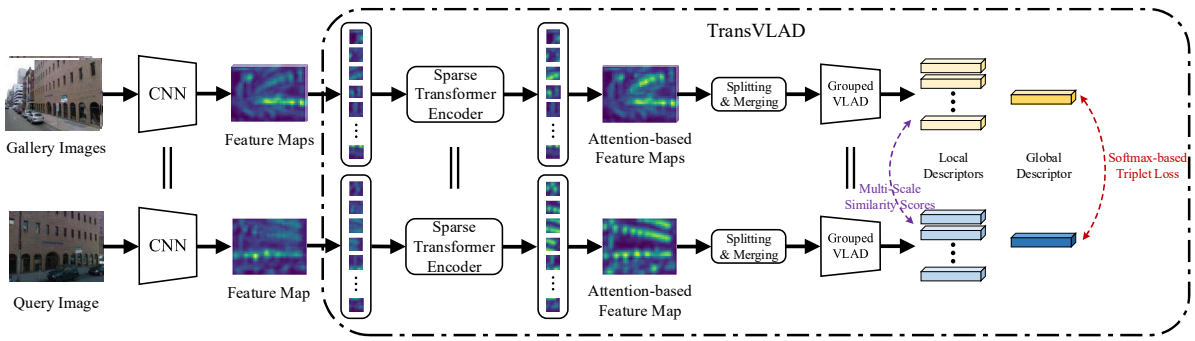

Figure 1. Overview of training architecture. A given feature map extracted using a CNN is input to the TransVLAD module, which aggregates attention-based feature maps into a compact global descriptor, and generates multi-scale similarity scores from local descriptors.

3. Methods

Our model is composed of a CNN backbone and the proposed TransVLAD module (see the architecture in Figure 1). This section provides additional details of our TransVLAD module and self-supervised learning method.

3.1. Sparse Transformer of TransVLAD

Considering that transformers are computation-intensive and lack of the inductive biases inherent in CNNs, such as translation invariance and locality [7], we use a CNN backbone to extract local features, and propose a sparse transformer as shown in Figure 2 to compute attention-based feature maps. In general, the ViT receives a sequence of 1D token embeddings as input [7, 25, 38]. For the 2D feature maps, we split a feature map $\mathbf{m} \in \mathbf{R}^{C\times H\times W}$ into a sequence of 2D patches $\mathbf{m}_p \in \mathbf{R}^{N\times P^2\cdot C}$, and down-sample the patches into single pixels (vectors) $\mathbf{x}_p \in \mathbf{R}^{N\times C}$, where C is the number of channels, (H, W) is the resolution of the feature map, $N = HW/P^2$ is the total number of patches, and P is the down sampling rate. Then, $\mathbf{x}_p$ is employed as the input embeddings.

Similar to ViT [7], we use standard learnable 1D position embeddings $\mathbf{x}_{pos} \in \mathbf{R}^{N\times C}$. The embedding mechanism retains positional information, and lets our TransVLAD encode corresponding geometric configurations between query and gallery images during training. Therefore, we can produce the resulting sequence of embedding vectors $\mathbf{x}_0 = \mathbf{x}_p + \mathbf{x}_{pos}$, which is used as the input for the L-layer transformer encoder.

We adopt the ViT as a background for our L-layer transformer encoder. Each layer consists of a multihead self-attention (MSA) module, a multi-layer perceptron (MLP), and Layernorm (LN) blocks. For the MSA module, which can effectively reduce visual ambiguities or incompleteness in large-scale geo-localization, we use the self-cross attention mechanism in L2LTR [12] to transfer features between adjacent layers. $\mathbf{x}_{l-1}$ represents the input sequence of layer

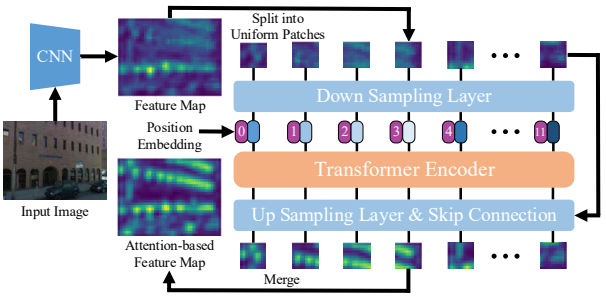

Figure 2. Sparse transformer of the proposed TransVLAD. the feature map using a CNN is split into uniform patches, which are down-sampled into single pixels (vectors). A position embedding is assigned to each vector, and then the sequence of vectors is fed to a transformer encoder. The output sequence of vectors are up-sampled and added with the input patches, and then merged into an attention-based feature map.

l, where $l \in \{1, 2, \cdots, L\}$, and a single self-cross attention head is formulated as follows:

$$\mathbf{z}_l = \mathrm{LN}(\mathbf{x}_{l-1}),\ \mathbf{z}_{l-1} = \mathrm{LN}(\mathbf{x}_{l-2}), \tag{1}$$

$$\mathbf{Q}_l = \mathbf{z}_l\mathbf{W}_l^q,\ \mathbf{K}_l = \mathbf{z}_{l-1}\mathbf{W}_l^k,\ \mathbf{V}_l = \mathbf{z}_l\mathbf{W}_l^v, \tag{2}$$

$$\mathbf{A}_l = \mathrm{softmax}(\frac{\mathbf{Q}_l\mathbf{K}_l^T}{\sqrt{D}})\mathbf{V}_l. \tag{3}$$

where $\mathbf{W}_l^q$, $\mathbf{W}_l^k$ and $\mathbf{W}_l^v$ are linear projection matrices.

Our sparse transformer can be expressed as:

$$\mathbf{x}_p = \mathrm{DSL}(\mathbf{m}_p),\ \mathbf{x}_0 = \mathbf{x}_p + \mathbf{x}_{pos}, \tag{4}$$

$$\mathbf{z}'_l = \mathrm{MSA}(\mathrm{LN}(\mathbf{z}_{l-1}),\ \mathrm{LN}(\mathbf{z}_{l-2})) + \mathbf{z}_{l-1},\ \mathbf{z}_0 = \mathbf{x}_0 \tag{5}$$

$$\mathbf{z}_l = \mathrm{MLP}(\mathrm{LN}(\mathbf{z}'_l)) + \mathbf{z}'_l,\ l = 1, 2, \cdots, L \tag{6}$$

$$\mathbf{x}_{\mathrm{out}} = \mathrm{LN}(\mathbf{z}_L),\ \mathbf{m}_{\mathrm{out}} = \mathbf{m}_p + \mathrm{USL}(\mathbf{x}_{\mathrm{out}}). \tag{7}$$

where DSL refers to the down sampling layer, which is average pooling. USL represents the up sampling layer, which is depthwise separable transposed convolution. The MLP is

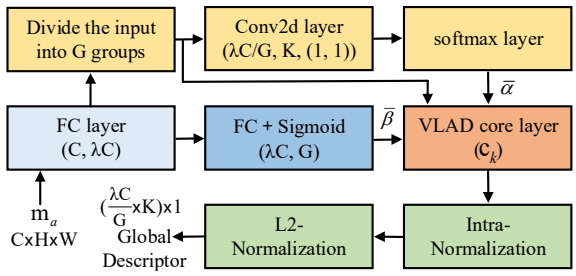

Figure 3. Grouped VLAD layer of our TransVLAD module. It is composed of standard neural networks (fully-connected layer, convolution layer) and a VLAD core layer to perform aggregation as illustrated in Eq. (8). The main parameters for the layers are shown in brackets.

used to alleviate the rank collapse issue. It contains two linear layers, and a GELU non-linearity activation layer between the two layers. LN blocks are utilized before MSA modules, MLP, and residual connections.

Finally the output sequence $\mathbf{m}_{\text{out}}$ is merged back into an attention-based feature map $\mathbf{m}_a \in \mathbf{R}^{C\times H\times W}$.

3.2. Grouped VLAD of TransVLAD

The original NetVLAD [1] utilizes the VLAD layer to generate a compact global descriptor of an image by aggregating the local feature maps extracted from CNNs. However, the dimension of global descriptors is usually high for image retrieval tasks, and a PCA layer with whitening [15] is added to convert it to an appropriate dimension $N^{'}$, which needs hundreds of millions parameters. For instance, a NetVLAD layer with 64 clusters will encode a feature map of 1024 dimensions as a 65536-dimensional vector. A PCA layer with 4096-dimensional outputs will result in about 268M parameters, making the model intractable on compact devices with limited resources. To alleviate this problem, we decompose these high-dimensional input feature vectors into groups of relatively low-dimensional vectors before VLAD aggregation to reduce the output dimension of the VLAD layer. In addition, we introduce a grouped weight $\bar{\beta}$ to maintain the non-linearity of the VLAD layer.

The architecture of the grouped VLAD layer of our TransVLAD module is shown in Figure 3. Specifically, the input attention-based feature map $\mathbf{m}_a \in \mathbf{R}^{C\times H\times W}$ is represented as $D^{'}$ ($D^{'} = HW$) C-dimensional local image descriptors. The descriptor vectors are first expanded to λ times through a fully-connected layer. Then the dimension of high-dimensional vectors $\{\hat{\mathbf{x}}_i\}$ are divided into G groups. The lower-dimensional descriptor vectors are represented as $\{\hat{\mathbf{x}}_{gi}\}, g = 1, \cdots, G, i = 1, \cdots, D^{'}$. Given K cluster centers $\{c_k\}$ as the parameters of the VLAD core layer, the output global descriptor V is $\frac{\lambda C}{G} \times K$-dimensional vector. To simplify the expression, V is converted to an $\frac{\lambda C}{G} \times K$ size matrix, and the element (j, k) of V is calculated as:

$$V(j,k) = \sum_{g=1}^{G} \sum_{i=1}^{D^{'}} \bar{\alpha}_k(\hat{\mathbf{x}}_{gi}) \bar{\beta}_g(\hat{\mathbf{x}}_i)(\hat{\mathbf{x}}_{gi}(j) - c_k(j)), \tag{8}$$

where $\hat{\mathbf{x}}_{gi}(j)$ and $c_k(j)$ are the j-th dimension of descriptor $\hat{\mathbf{x}}_{gi}$ and cluster k respectively. The proximity measurement of the lower-dimensional vector $\hat{\mathbf{x}}_{gi}$ with the cluster centre c_k is composed of two parts:

$$\bar{\alpha}_k(\hat{\mathbf{x}}_{gi}) = \frac{e^{\mathbf{w}_{gk}^T \hat{\mathbf{x}}_{gi} + b_{gk}}}{\sum_{k'=1}^{K} e^{\mathbf{w}_{gk'}^T \hat{\mathbf{x}}_{gi} + b_{gk'}}}, \tag{9}$$

$$\bar{\beta}_g(\hat{\mathbf{x}}_i) = \sigma(\mathbf{w}_g^T \hat{\mathbf{x}}_i + b_g). \tag{10}$$

where $\sigma(\cdot)$ represents a sigmoid function with output from 0 to 1. Similar to the original NetVLAD, our grouped VLAD layer aggregates the first-order statistics of residuals $(\hat{\mathbf{x}}_{gi}(j) - c_k(j))$ in different parts of the feature descriptors weighted by $\bar{\alpha}_k(\hat{\mathbf{x}}_{gi})$ and $\bar{\beta}_g(\hat{\mathbf{x}}_i)$. The first part $\bar{\alpha}_k(\hat{\mathbf{x}}_{gi})$ is the soft-assignment of descriptor $\hat{\mathbf{x}}_{gi}$ to cluster k, while the second part $\bar{\beta}_g(\hat{\mathbf{x}}_i)$ is a weight coefficient on all groups.

Then, the matrix V obtained via Intra-Normalization [2] is converted into a $\frac{\lambda C}{G} \times K$-dimensional vector, and finally L2-normalized as a global descriptor. The dimension of global descriptors is finally converted to an appropriate dimension $N^{'}$ through a PCA layer. The number of parameters in our grouped VLAD layer with a PCA layer is about $\frac{\lambda}{G}$ times smaller than that of vanilla NetVLAD.

3.3. Self-supervised Learning Method

State-of-the-art models [1, 17, 23, 41] only utilize the first-ranked positive images for the most similar ones in representation space as the positive training samples. We propose a self-supervised learning method to further encode more representative information from multi-scale patches between the query and lower-ranked positive gallery images. This is used to train our TransVLAD module with a CNN backbone. Given one query image q and one lower-ranked positive image p^1, we get the corresponding attention-based feature maps $\{\mathbf{m}_q^{\theta_0}, \mathbf{m}_{p^1}^{\theta_0}\}$, where θ_0 is the parameters of the initial network. To achieve multi-scale patches, we split and merge the feature maps $\{\mathbf{m}_q^{\theta_0}, \mathbf{m}_{p^1}^{\theta_0}\}$ as shown in Figure 4. Then, the feature maps $\{\mathbf{m}_q^{\theta_0}, \mathbf{m}_{q_1}^{\theta_0}, \cdots, \mathbf{m}_{q_{20}}^{\theta_0}, \mathbf{m}_{p^1}^{\theta_0}, \mathbf{m}_{p_1^1}^{\theta_0}, \cdots, \mathbf{m}_{p_{20}^1}^{\theta_0}\}$ are fed into the grouped VLAD layer to get corresponding global and local descriptors $\{f_q^{\theta_0}, f_{q_1}^{\theta_0}, \cdots, f_{q_{20}}^{\theta_0}, f_{p^1}^{\theta_0}, f_{p_1^1}^{\theta_0}, \cdots, f_{p_{20}^1}^{\theta_0}\}$. Given one query image q and top-k lower-ranked positive images $\{p^1, \cdots, p^k\}$, the multi-scale similarity scores is estimated as follows:

$$\begin{aligned}\mathcal{R}_{\theta_0}(\tau_0) = \text{softmax}([&\langle f_q^{\theta_0}, f_{p^1}^{\theta_0}\rangle/\tau_0, \langle f_q^{\theta_0}, f_{p_1^1}^{\theta_0}\rangle/\tau_0, \cdots, \\ &\langle f_q^{\theta_0}, f_{p_{20}^1}^{\theta_0}\rangle/\tau_0, \cdots, \langle f_{q_1}^{\theta_0}, f_{p^1}^{\theta_0}\rangle/\tau_0, \cdots, \langle f_{q_1}^{\theta_0}, f_{p_{20}^1}^{\theta_0}\rangle/\tau_0, \\ &\cdots, \langle f_{q_{20}}^{\theta_0}, f_{p^1}^{\theta_0}\rangle/\tau_0, \langle f_{q_{20}}^{\theta_0}, f_{p_1^1}^{\theta_0}\rangle/\tau_0, \cdots, \langle f_{q_{20}}^{\theta_0}, f_{p_{20}^1}^{\theta_0}\rangle/\tau_0]),\end{aligned} \tag{11}$$

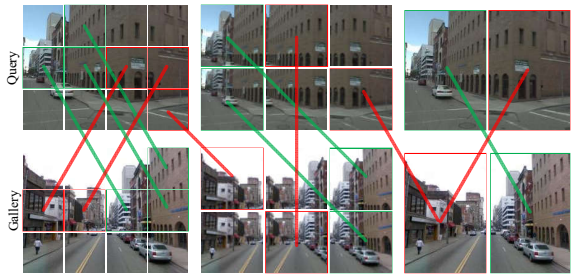

Figure 4. The splitting and merging of feature maps to produce multi-scale patches. The feature maps of these patches are fed to the grouped VLAD layer to generate corresponding descriptors. Cross-matching of these local descriptors are performed across query and gallery images to compute multi-scale similarity scores. We show part of the results in this figure. Green lines indicate high similarity scores, while red lines indicate low ones.

where $\langle \cdot, \cdot \rangle$ represents the inner product between two feature vectors. τ_0 is the temperature hyper-parameter which controls the smoothness of the multi-scale similarity scores $\mathcal{R}_{\theta_0}$ in the initial network.

The multi-scale similarity scores $\mathcal{R}_{\theta_0}$ estimated by the initial network are used to supervise the 1st generation network through a cross-entropy loss. Thus, the multi-scale similarity loss is expressed as:

$$L_s^{\theta_1}(q, p^1, \cdots, p^k) = \ell_{ce}(\mathcal{R}_{\theta_1}(1), \mathcal{R}_{\theta_0}(\tau_0)). \quad (12)$$

where ℓ_{ce} represents the cross-entropy loss, which is given by $\ell_{ce}(y, \hat{y}) = -\sum_i \hat{y}(i) \log(y(i))$. θ_1 and θ_0 are the parameters of the 1st generation and initial networks respectively. Note that only the target similarity vector $\mathcal{R}_{\theta_0}$ adopts the temperature hyper-parameter to control its smoothness.

We separate the training into ω generations to progressively provide more accurate multi-scale similarity scores for training more discriminative global descriptors. The similarity vector $\mathcal{R}_{\theta_\nu}$ is distributed equally with a larger $\tau_\nu, \nu = 0, \cdots, \omega$, which means the multi-scale similarity loss $L_s^{\theta_\nu}$ focuses on a large number of lower-ranked positive images. Therefore, the temperature τ_ν is set to be relatively large in early generations given the lower accuracy of the multi-scale similarity scores. As the similarity scores in later generations will be more accurate, the multi-scale similarity loss $L_s^{\theta_\nu}$ is pushed to focus on a small number of true positive patches by reducing the temperature τ_ν.

Similar to the recent visual geo-localization methods [1, 17, 23], we use a base loss to supervise the network by feeding triplets. Each triplet consists of a single query image q, its first-ranked positive image p^* and several high-ranked negative images $\{n_i|_{i=1}^M\}$. The rank of the positive or negative images is based on the distances between the global descriptor vectors aggregated by our TransVLAD module. Therefore, the vanilla triplet loss written as:

$$L_{t_0}^\theta(q, p^*, n) = \sum_{i=1}^{M} \max(0, ||f_q^\theta - f_{p^*}^\theta||_2^2 - ||f_q^\theta - f_{n_i}^\theta||_2^2 + \epsilon), \quad (13)$$

where ϵ is the minimum offset between distances of similar versus dissimilar pairs. As in [41], p^* is the first-ranked gallery image within 10 meters of the query q. $\{n_i|_{i=1}^M\}$ are randomly selected from the top-500 gallery images more than 25 meters away from the query q.

The above triplet loss can be used to optimize our TransVLAD module with a CNN backbone. However, it is not very robust, and relies greatly in the accurate choice of negative images to improve performance. Therefore, we exploit a softmax-based triplet loss [23] to better maximize the ratio between the most similar pair and several dissimilar pairs. We can write this assumption as follows:

$$L_t^\theta(q, p^*, n) = -\sum_{i=1}^{M} \log \frac{\exp\langle f_q^\theta, f_{p^*}^\theta \rangle}{\exp\langle f_q^\theta, f_{p^*}^\theta \rangle + \exp\langle f_q^\theta, f_{n_i}^\theta \rangle}, \quad (14)$$

Therefore, our TransVLAD module with a CNN backbone is supervised by both softmax-based triplet loss, and the multi-scale similarity loss. The total loss in each generation is expressed as:

$$L_{total}^{\theta_\nu} = L_t^{\theta_\nu}(q, p^*, n) + \lambda_s L_s^{\theta_\nu}(q, p^1, \cdots, p^k). \quad (15)$$

where θ_ν is the parameters of the ν-th generation network, and λ_s is the loss weighting factor.

4. Experiments

In this section, we present and discuss the experimental results obtained when applying our proposed TransVLAD module on challenging geo-localization and image retrieval datasets. Its performance is evaluated and compared against related state-of-the-art methods.

4.1. Datasets

The TransVLAD is evaluted on three challenging geo-localization benchmarks – the Pitts30k/250k-test [37], TokyoTM-val [1], and Tokyo 24/7 [36] – which only have GPS tags. These datasets contain changeable appearances and perspectives of the actual images. Testing is performed on these datasets in their recommended configuration. In addition, the generalization ability of our model is evaluated on three standard image retrieval benchmarks – the Oxford 5K [27], Paris 6K [28] and Holidays [16].

4.2. Implementation Details

The proposed TransVLAD module was implemented in PyTorch. In our experiments, all the networks are trained on the Pitts30k-train [37], the same dataset as the other works

Table 1. Complexity and accuracy of the proposed and state-of-art models on Pitts30k/250k-test, TokyoTM-val, and Tokyo 24/7 datasets.

Method	Model Complexity		Pitts30k-test			Pitts250k-test			TokyoTM-val			Tokyo 24/7		
	Parmas(M)	FLOPs(G)	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
NetVLAD [1]	148.97	94.35	85.6	92.9	94.9	86.0	93.2	95.1	93.9	96.8	97.6	73.3	82.9	86.0
CRN [17]	148.97	94.35	-	-	-	85.5	93.5	95.5	-	-	-	75.2	83.8	87.3
SARE [23]	148.97	94.35	-	-	-	89.0	95.5	96.8	94.5	96.7	97.3	79.7	86.7	90.5
HAF [41]	158.87	1791.27	-	-	-	89.4	95.8	97.1	94.8	97.8	98.2	78.2	84.4	87.8
Patch-NetVLAD [10]	148.97	94.22	88.7	94.5	95.9	89.8	95.9	97.0	95.2	97.8	98.3	**86.0**	88.6	90.5
Ours-VGG16	84.74	95.60	**89.3**	94.5	96.0	**91.1**	**96.3**	**97.5**	**96.0**	**98.3**	**98.8**	85.4	89.8	91.7
Ours-MobileNetV3	**80.14**	**7.91**	89.1	**94.9**	**96.1**	90.7	96.2	97.4	95.4	98.0	98.7	83.5	**90.5**	**92.1**

[1, 17, 23, 41, 10], and tested on several other datasets [37, 1, 36, 27, 28, 16]. For the evaluation and comparison, our TransVLAD was trained with a minor-modified MobileNetV3 [13] CNN backbone. Specifically, we use the ImageNet-pretrained MobileNetV3-Large as the backbone, in which the stride of its last stride-2 convolution layer is changed from 2 to 1, and the final channels from 960 to 1024. We also use the same VGG-16 backbone as the other works for fair comparisons. For the sparse transformer, the down sampling rate is set to 2. For the grouped VLAD layer, we set $\lambda = 2, G = 8$.

Training is separated into 4 generations with 5 epochs each. The stochastic gradient descent (SGD) algorithm is exploited to optimize the total loss function, with learning rate = 0.0001, weight decay = 0.001 and momentum = 0.9. To achieve better hyper-parameters, we perform a grid search to find the training configuration that performs best on the Pitts30k-val dataset. We set the loss weighting factor $\lambda_s = 0.5$ and temperatures $\tau_\nu = 0.04 \sim 0.07$.

4.3. Comparison with the State-of-Art

To assess the performance of our model, we compare it with five state-of-the-art models for the geo-localization task: NetVLAD [1], CRN [17], SARE [23], HAF [41] and Patch-NetVLAD [10]. For fair comparisons, we also set the number of clusters $K = 64$, and use the 4096-dimensional global descriptors to perform image retrieval.

Geo-localization Benchmarks. Table 1 shows experimental results comparing our models with others on both the model complexity and the precision-recall on the three geo-localization datasets. The number of model parameters consists of parameters in the backbone, VLAD layers, and a PCA layer with whitening. The floating-point operations per second (FLOPs) is calculated by the input image size (640, 480) during testing. The top-k recall is the percentage of successfully retrieved query images. Our model outperforms state-of-the-art models on almost all the benchmarks. For example, our model with VGG-16 achieves 91.1% rank-1 recall on Pitts250k-test dataset with an improvement of 1.3% compared to Patch-NetVLAD. On the challenging Tokyo 24/7 dataset, our model with VGG-16 has a robust generalization ability and achieves 85.4% rank-1 recall, up to 5.7% accuracy improvement against SARE. In terms of complexity, our model with VGG-16 has about half the parameters compared to the other models with the same backbone. In addition, our model with MobileNetV3 has lower model complexity, with an accuracy that is comparable to our model with VGG-16, of which the model complexity is as low as 80.14M parameters and 7.91G FLOPs in testing.

Image Retrieval Benchmarks. Table 2 reports our experimental results in terms of mean Average Precision (mAP). To evaluate the generalization performance, our models are tested them on three image retrieval datasets without any fine-tuning and compare the results with other five approaches [1, 17, 23, 41, 10]. For Oxford 5K [27] and Paris 6K [28] datasets, we test on both full and cropped query images. For Holidays [16] dataset, we only use the original query images for testing. "Full" means the whole image is directly used as a query image, while "Crop" means the query image only uses the landmark region of the whole image. Compared to others, our model with VGG-16 shows good generalization ability, and outperforms the second-best Patch-NetVLAD with 0.5% to 1.1% improvements on all three datasets. In addition, our model with MobileNetV3 also has good generalization ability and performs comparably to our model with VGG-16. On the Holidays dataset, our model with VGG-16 performs lower than NetVLAD since Pitts30k-train dataset lacks images with natural scenes like those in Holidays dataset.

4.4. Qualitative Evaluation

Figure 5 shows the the top-1 retrieval gallery images, and the attention maps of query images by our model with VGG-16, SARE, and NetVLAD on challenging Tokyo 24/7 dataset. The results show the performance of our model

Table 2. Mean Average Precision (mAP) of our proposed and state-of-art methods on image retrieval datasets.

Method	Oxford 5K		Paris 6K		Holidays
	full	crop	full	crop	
NetVLAD [1]	69.1	71.6	78.5	79.7	**83.1**
CRN [17]	69.2	-	-	-	-
SARE [23]	71.7	75.5	82.0	81.1	80.7
HAF [41]	71.9	75.8	82.3	81.5	81.4
Patch-NetVLAD [10]	72.1	75.9	82.1	81.8	81.7
Ours-VGG16	**73.2**	**76.4**	**82.8**	**82.3**	82.5
Ours-MobileNetV3	72.8	76.2	82.5	82.1	82.2

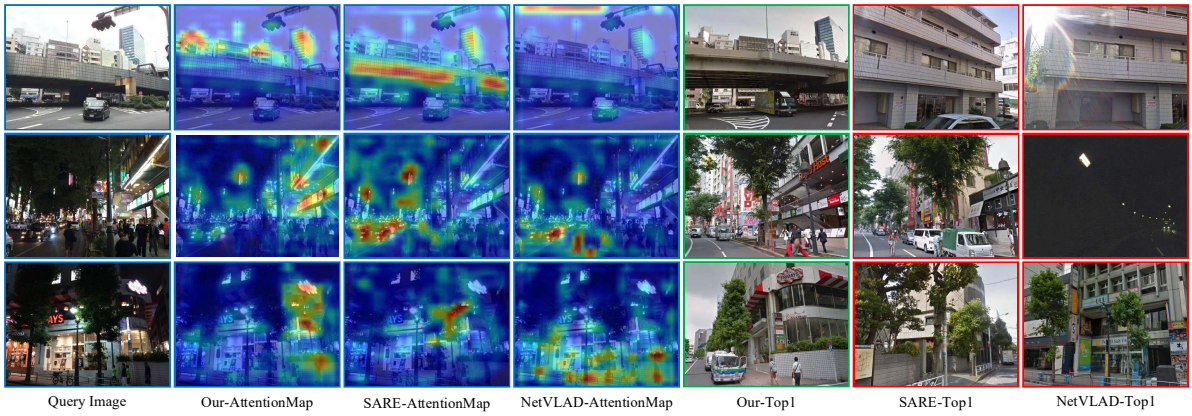

Figure 5. Examples of challenging cases for retrieval results from the Tokyo 24/7 dataset. The attention maps of query images are generated to show the regions where the models focus. We compare the attention maps and the top-1 retrieved gallery images. Green and red borders indicate correct and incorrect retrieval results, respectively. (Best viewed in color.)

Table 3. Ablation studies with our TransVLAD module and self-supervised learning (SSL) method on the Pitts250k-test and Tokyo 24/7 datasets. Note that G-TransVLAD refers to the proposed TransVLAD module, while V-TransVLAD refers to the TransVLAD module with the vanilla NetVLAD. S-Triplet refers to the softmax-base triplet loss while V-Triplet means the vanilla triplet loss.

Method	**Pitts250k-test**			**Tokyo 24/7**		
	R@1	R@5	R@10	R@1	R@5	R@10
Baseline (VGG16 + NetVLAD + V-Triplet)	86.5	93.9	95.8	75.3	83.9	87.4
Ours (VGG16 + V-TransVLAD + V-Triplet)	**87.4**	94.4	96.1	77.2	84.9	88.5
Ours (VGG16 + G-TransVLAD + V-Triplet)	87.3	**94.6**	**96.4**	**77.4**	**85.2**	**88.7**
Baseline (VGG16 + NetVLAD + S-Triplet)	89.0	95.5	96.8	79.7	86.7	90.5
Ours (VGG16 + V-TransVLAD + S-Triplet)	**90.0**	**95.8**	96.8	80.9	87.9	90.9
Ours (VGG16 + G-TransVLAD + S-Triplet)	89.8	95.7	**96.9**	**81.2**	**88.3**	**91.0**
Ours (VGG16 + NetVLAD + SSL)	90.1	95.9	97.1	82.1	89.0	91.5
Ours (VGG16 + V-TransVLAD + SSL)	90.8	96.2	**97.5**	83.4	89.5	91.5
Ours (VGG16 + G-TransVLAD + SSL)	**91.1**	**96.3**	**97.5**	**85.4**	**89.8**	**91.7**
Ours (MobileNetV3 + NetVLAD + S-Triplet)	86.8	94.4	96.0	76.7	84.3	88.3
Ours (MobileNetV3 + G-TransVLAD + S-Triplet)	87.7	94.7	96.5	77.8	85.5	89.1
Ours (MobileNetV3 + NetVLAD + SSL)	89.3	95.4	96.8	81.3	88.1	90.9
Ours (MobileNetV3 + G-TransVLAD + SSL)	**90.7**	**96.2**	**97.4**	**83.5**	**90.5**	**92.1**

with VGG-16 on the geo-localization datasets with complex environments and variable lighting conditions. We use the feature maps before the grouped VLAD layer and adopt the method in [34] to generate the attention maps.

From the results of three difficult query images, our model with VGG-16 is shown to focus on the discriminative regions (*e.g.* buildings, signs), while the other two models incorrectly focus on changeable objects (*e.g.* trees, cars, pedestrians and light). This misdirection by changeable objects results in false retrieval results, since the objects may shift or vanish from the right gallery images, or appear in the incorrect gallery images. Therefore, our model with VGG-16 appears to pay more attention to the discriminative landmarks rather than changeable objects.

4.5. Ablation Studies

To verify the effectiveness of our TransVLAD module and self-supervised learning (SSL) method, we use the networks (VGG-16 + NetVLAD) and the results of the SARE as the baseline, and perform ablation studies by comparing different methods on models with both VGG-16 and MobileNetV3 backbones. For fair comparisons, all models have $K = 64$ clustering centers and use 4096-dimensional global descriptors to perform image retrieval.

The ablation studies on Pitts250k-test and Tokyo 24/7 datasets are reported in Table 3. From the results, we can draw the following conclusions.

Softmax-based Triplet Loss is more robust than the vanilla triplet loss and can better help the model to converge. For example, exploiting the softmax-based triplet loss improves the R@1 performance of the SARE from 86.5% to 89.0% on Pitts250k-test dataset and achieves 4.4% improvement at R@1 on Tokyo 24/7 dataset.

TransVLAD Module can improve global contextual reasoning and enable the model to encode corresponding geometric configurations between query and gallery images. In particular, using TransVLAD module to replace the NetVLAD improves the R@1 performance of our model

from 90.1% to 91.1% on Pitts250k-test dataset and achieves 3.3% improvement at R@1 on Tokyo 24/7 dataset.

Grouped VLAD provides marginally higher accuracy than the vanilla NetVLAD while the total number of parameters is about four times (as mentioned in Section 3.2) smaller than that of NetVLAD.

Self-supervised Learning Method can effectively exploit the potential of lower-ranked positive images, and incite the model to encode more accurate information from multi-scale patches between the query and positive gallery images. Specifically, adopting our SSL method improves the R@1 performance of the SARE from 89.0% to 90.1% on Pitts250k-test dataset and achieves 2.4% improvement at R@1 on Tokyo 24/7 dataset. In addition, our TransVLAD module and SSL method are also effective in the models with MobileNetV3.

Table 4. Ablation studies with different down sampling methods in the sparse transformer module.

Method	Pitts250k-test			Tokyo 24/7		
	R@1	R@5	R@10	R@1	R@5	R@10
Avg Pooling	**91.1**	96.3	**97.5**	**85.4**	**89.8**	**91.7**
Max Pooling	91.0	**96.4**	97.4	85.0	89.4	91.4
Center Pooling	90.7	96.1	97.4	84.5	89.0	91.3

We also perform ablation studies with different down sampling methods in the sparse transformer module. We use our model with VGG-16 and test on Pitts250k-test and Tokyo 24/7 datasets. Table 4 shows that our model with average pooling achieves the highest level of accuracy.

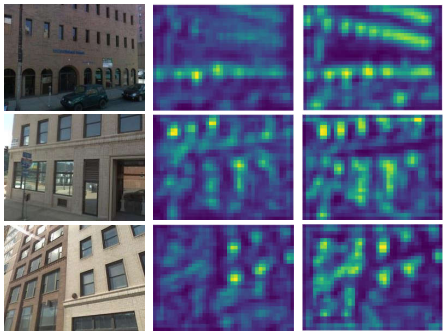

Figure 6. Comparison of feature maps and attention-based feature maps from the query images of Pitts30k/250k-test datasets.

To show the impact of the TransVLAD module on global contextual reasoning, we compare the feature maps versus attention-based feature maps from the query images in Pitts30k/250k-test. As shown in Figure 6, feature maps extracted by the VGG-16, and attention-based feature maps are the output of the sparse transformer of our TransVLAD module. Results show that attention-based feature maps pay more attention to the global information of a landmark.

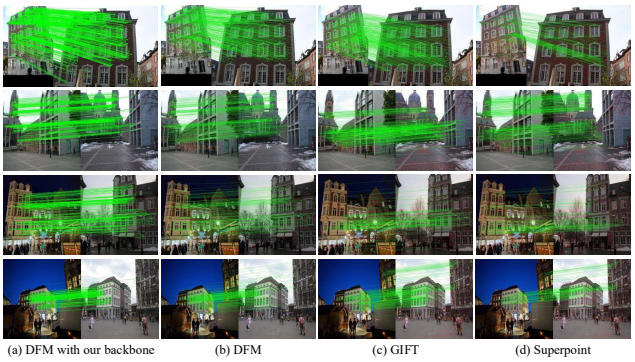

Figure 7. Qualitative matching results of four models. The correct matches are drawn in green lines.

4.6. Generalization on Matching Keypoints

To further assess the performance of our network (VGG-16 + sparse transformer of our TransVLAD module) trained by proposed self-supervised learning method, we adopt the DFM [8] and replace the VGG-16 backbone with our network trained on Pitts30k-train dataset for feature extraction. Apart from the backbone, we do not fine-tuning the DFM to improve matching. As shown in Figure 7, we estimate matching results on both day and night image pairs and compare them with the results of three state-of-the-art models – DFM, GIFT [24], and Superpoint [5]. The results show that DFM with our CNN backbone generates more dense and accurate matching in comparison with the other three models, which can attest to the stronger generalization ability of our network.

5. Conclusions

In this paper, a novel TransVLAD module is introduced for aggregating local features into a discriminative and compact global descriptor. A sparse transformer is proposed, and high-dimensional feature vectors are decomposed into groups of relatively low-dimensional vectors before performing VLAD aggregation to construct an efficient module. Finally, a self-supervised learning method is proposed to further encode more information from multi-scale patches between the query and lower-ranked positive gallery images. We evaluated our model on both geo-localization and image retrieval benchmarks. Experimental results indicate that our model can achieve higher accuracy and efficiency compare to related state-of-art methods.

Acknowledgments

This research was supported in part by Mitacs. The corresponding author is Pourya Shamsolmoali.

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[2] Relja Arandjelovic and Andrew Zisserman. All about vlad. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, and et al. Language models are few-shot learners. In *Adv. Neural Inform. Process. Syst.*, pages 1877–1901, 2020.

[4] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

[5] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, 2018.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Pro. Conf. N. Am. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, pages 4171–4186, 2019.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, and et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. Learn. Represent.*, 2021.

[8] Ufuk Efe, Kutalmis Gokalp Ince, and Aydin Alatan. Dfm: A performance baseline for deep feature matching. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4284–4293, 2021.

[9] Alaaeldin El-Nouby, Natalia Neverova, Ivan Laptev, and Hervé Jégou. Training vision transformers for image retrieval. *arXiv preprint arXiv:2102.05644*, 2021.

[10] Stephen Hausler, Sourav Garg, Ming Xu, Michael Milford, and Tobias Fischer. Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 14141–14152, 2021.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[12] hongji yang, Xiufan Lu, and Yingying Zhu. Cross-view geo-localization with layer-to-layer transformer. In *Adv. Neural Inform. Process. Syst.*, 2021.

[13] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, and et al. Searching for mobilenetv3. In *Int. Conf. Comput. Vis.*, 2019.

[14] Christian Häne, Lionel Heng, Gim Hee Lee, Friedrich Fraundorfer, Paul Furgale, Torsten Sattler, and Marc Pollefeys. 3d visual perception for self-driving cars using a multi-camera system: Calibration, mapping, localization, and obstacle detection. *Image Vis. Comput.*, pages 14–27, 2017.

[15] Hervé Jégou and Ondřej Chum. Negative evidences and cooccurences in image retrieval: The benefit of pca and whitening. In *Eur. Conf. Comput. Vis.*, 2012.

[16] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Eur. Conf. Comput. Vis.*, pages 304–317, 2008.

[17] Hyo Jin Kim, Enrique Dunn, and Jan-Michael Frahm. Learned contextual feature reweighting for image geo-localization. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[18] Youngran Jo, Jinbeum Jang, and Joonki Paik. Camera orientation estimation using motion based vanishing point detection for automatic driving assistance system. In *IEEE Int. Conf. Consum. Electron.*, pages 1–2, 2018.

[19] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3304–3311, 2010.

[20] Ahmad Khaliq, Shoaib Ehsan, Zetao Chen, Michael Milford, and Klaus McDonald-Maier. A holistic visual place recognition approach using lightweight cnns for significant viewpoint and appearance changes. *IEEE Trans. Robot.*, pages 561–569, 2020.

[21] Rongcheng Lin, Jing Xiao, and Jianping Fan. Nextvlad: An efficient neural network to aggregate frame-level features for large-scale video classification. In *Eur. Conf. Comput. Vis. Workshops*, 2018.

[22] Liu Liu, Hongdong Li, and Yuchao Dai. Efficient global 2d-3d matching for camera localization in a large-scale 3d map. In *Int. Conf. Comput. Vis.*, 2017.

[23] Liu Liu, Hongdong Li, and Yuchao Dai. Stochastic attraction-repulsion embedding for large scale image localization. In *Int. Conf. Comput. Vis.*, 2019.

[24] Yuan Liu, Zehong Shen, Zhixuan Lin, Sida Peng, Hujun Bao, and Xiaowei Zhou. Gift: Learning transformation-invariant dense visual descriptors via group cnns. In *Adv. Neural Inform. Process. Syst.*, 2019.

[25] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Int. Conf. Comput. Vis.*, pages 10012–10022, 2021.

[26] Raúl Mur-Artal and Juan D. Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.*, pages 1255–1262, 2017.

[27] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1–8, 2007.

[28] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1–8, 2008.

[29] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *Technical Report*, 2018.

[30] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, page 9, 2019.

[31] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[32] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Fast image-based localization using direct 2d-to-3d matching. In *Int. Conf. Comput. Vis.*, pages 667–674, 2011.

[33] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Int. Conf. Learn. Represent.*, 2015.

[34] Abby Stylianou, Richard Souvenir, and Robert Pless. Visualizing deep similarity networks. In *IEEE Winter Conf. Appl. Comput. Vis.*, pages 2029–2037, 2019.

[35] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, and et al. Going deeper with convolutions. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1–9, 2015.

[36] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015.

[37] Akihiko Torii, Josef Sivic, Tomas Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013.

[38] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers & amp; distillation through attention. In *Proc. Int. Conf. Mach. Learn.*, pages 10347–10357, 2021.

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Adv. Neural Inform. Process. Syst.*, 2017.

[40] Ruotong Wang, Yanqing Shen, Weiliang Zuo, Sanping Zhou, and Nanning Zheng. Transvpr: Transformer-based place recognition with multi-level attention aggregation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13648–13657, 2022.

[41] Liqi Yan, Yiming Cui, Yingjie Chen, and Dongfang Liu. Hierarchical attention fusion for geo-localization. In *IEEE Int. Conf. Acoust. Speech. Signal. Process. Proc.*, pages 2220–2224, 2021.

[42] Jun Yu, Chaoyang Zhu, Jian Zhang, Qingming Huang, and Dacheng Tao. Spatial pyramid-enhanced netvlad with weighted triplet loss for place recognition. *IEEE Trans. Neural Netw. Learn. Syst.*, pages 661–674, 2020.