ETR: An Efficient Transformer for Re-ranking in Visual Place Recognition

Hao Zhang[1], Xin Chen[1], Heming Jing[1], Yingbin Zheng[2], Yuan Wu[1], Cheng Jin[1*]

[1]School of Computer Science, Fudan University, Shanghai, China [2]Videt Technology, Shanghai, China

{zhanghao20,20210240337,20210240185,wuyuan,jc}@fudan.edu.cn zyb@videt.cn

Abstract

Visual place recognition is to estimate the geographical location of a given image, which is usually addressed by recognizing its similar reference images from a database. The reference images are usually retrieved via similarity search using global descriptor, and the local descriptors are used to re-rank the initial retrieved candidates. The local descriptors re-ranking can significantly improve the accuracy of global retrieval but comes at a high computational cost. To achieve a good trade-off between accuracy and efficiency, we propose an Efficient Transformer for Re-ranking (ETR), utilizing both global and local descriptors to re-rank the top candidates in a single shot. In contrast to traditional re-ranking methods, we leverage self-attention to capture relationships between local descriptors in a single image and cross-attention to explore the similarity of the image pairs. We show that the proposed model can be regarded as a general re-ranking algorithm for significantly boosting the performance of other global-only retrieval methods. Extensive experimental results show that our method outperforms state-of-the-arts and is orders of magnitude faster in terms of computational efficiency.

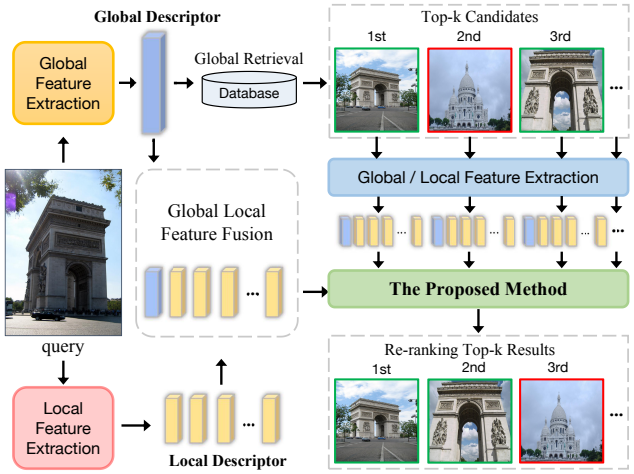

Figure 1. Pipeline of the proposed re-ranking method. For a query image, its global and local descriptors are firstly extracted, and the top-k candidates are retrieved from the database via global descriptor. Then the global/local descriptors of the query and candidates are fused to constitute the input of the model. So that re-ranking the top-k candidates requires just a single shot, which is orders of magnitude faster than the traditional re-ranking methods (such as geometric verification).

1. Introduction

Visual Place Recognition (VPR) is a challenging task in video-based navigation systems such as autonomous driving and mobile robotic localization, which has generally been approached as a special case of image retrieval. Given a query image, the VPR algorithms usually retrieve the candidate images from a database via image representations [47]. The image representations can be further subdivided into two major classes, i.e., global descriptors [6, 45, 1, 37, 28, 36] and local descriptors [24, 6, 10, 16, 5]. Global descriptor describes the whole image by a single feature vector, leading to a compact representation for boosting large-scale search and a discriminative representation for appearance and illumination changes. Local descriptors focus on the interest (*e.g.* landmarks) of an image and highlight patterns that differ from its neighborhood, which are shown to be more important to improve the retrieval precision. To further boost the retrieval accuracy, VPR usually adopts a two-stage process: the global features are firstly applied to retrieve candidates from database, and then the pair-wise local descriptors matching is used to re-rank the initial candidates.

Considering the re-ranking stage, many state-of-art approaches [6, 16] still rely on traditional methods such as geometric verification [27]. Geometric verification usually performs local descriptors matching in a brute-force manner, i.e., exhaustive comparison between two local descrip-

*Corresponding author.

tors sets to find mutual nearest neighbor matches. The image pair similarity is given by the number of inliers when estimating the homography based on the matches with RANSAC [14]. Utilizing local descriptors matching to reorder the initial candidates can significantly improve retrieval performance but comes at a high computational cost which is unfriendly for time-sensitive systems [2, 26].

To make full use of local descriptors to guarantee performance while alleviating the computation costs, we design ETR, an Efficient Transformer for Re-ranking that can directly generate a similarity score for an image pair, as shown in Figure 1. Inspired by the success of Transformers [39] and seminal work such as SuperGlue [31] and LoFTR [33], we use Transformers to process local descriptors extracted from pre-trained CNN models. Thanks to the attention mechanism and global receptive field of the Transformer, we can leverage the self-attention to capture complex spatial relationships encoded in a single image. And the bidirectional cross-attention can replace the relatively expensive process of mutual nearest neighbor search and perform the local feature matching across images more efficiently. Different from conventional re-ranking methods, the proposed model takes the fused descriptors of the query and top-k candidates as input, so that re-ranking top-k neighbors requires a single forward pass. Compared with traditional geometric verification methods that can only process image pairs serially, the proposed model can be easily parallelized and significantly accelerate the process of re-ranking while achieving competitive results on multiple benchmarks.

The contributions of this paper unfolds as follows:

- We propose an efficient Transformer for image re-ranking, by leveraging self- and cross-attention to directly predict the similarity of an image pair. ETR is able to process image pairs in parallel with low computational time and memory requirements.
- We show that ETR can be regarded as a general re-ranking algorithm to improve retrieval performance for those global-only methods, and can be used as an alternate for other re-ranking approaches.
- Experiment results show that ETR can achieve state-of-the-art performance on several VPR benchmarks.

2. Related Work

Image representations play an important role for visual place recognition, which can be further divided into two categories. The **local descriptors** can also be treated as key-point descriptors or regional descriptors, including the traditional hand-crafted local features (*e.g.*, SIFT [23], SURF [4]) and more recent learning-based features (*e.g.*, DELF [24], R2D2 [30]). Local descriptors can either be aggregated to obtain global descriptors or perform cross-matching between image pairs. In order to learn task-specific local features (*e.g.* landmarks) from an image, several attempts [24, 6] have been proposed. To better leverage VPR prior knowledge, Patch-NetVLAD [16] directly extracted multil-scale patch features from global descriptors generated by a VPR-optimized aggregation technique NetVLAD [1], which reverses the traditional local-to-global process of image representation and provides a new perspective for local feature extraction. The **global descriptors** are used to summarize an image into a compact representation for large-scale image retrieval while being robust to appearance, illumination and viewpoint changes. In the age of traditional machine learning, global descriptors were developed mainly by aggregating hand-crafted local descriptors [18, 19, 3]. Nowadays, most high-performing global features are based on deep convolution neural networks [32, 17] or vision Transformers [12, 13]. Many approaches are proposed to optimize operations like pooling (*e.g.*, GeM [28] and R-MAC [36]) or aggregating (*e.g.*, NetVLAD [1] and NetBoW [25]) to create more compact and discriminative global features. To train deep CNN or visual Transformers models, ranking-based losses [8, 42] and classification-based losses [9] are proposed.

Re-ranking in visual place recognition. Re-ranking the initial candidates has been proved as an effective way to improve performance. Geometric verification [16, 24, 6, 20] is a kind of re-ranking method widely used in VPR, which can generate stable and explainable results. The re-ranking process can be divided into two steps: feature matching and consistency check. Feature matching is used to detect feature-to-feature correspondences among a pair of images, which usually adopts brute force search to find local features that are mutual nearest neighbors. Several algorithms such as SuperGlue [31] and LoFTR [33] have been developed to optimize this process. Consistency check is used to analyze the consistency of spatial transformations and verify the reliability of the correspondences, typically implemented by using RANSAC [14]. Some other spatial matching attempts [16, 5] have been proposed to reduce computation complexity. Nonetheless, these methods are still computationally intensive processes and require a large number of local descriptors to guarantee performance.

Transformers in vision tasks. Transformers [39] was first introduced in natural language processing field, has become the *de facto* standard for sequence modeling. Recently, Transformers have attracted more and more attention in pure vision tasks [12, 7]. As the key part of Transformers architecture, self-attention mechanism has also been studied for image retrieval and visual place recognition [13, 41]. The most related to our work is RRT [35], RRT uses the standard Transformer structure to learn the visual relation of an image pair. Different from RRT, we leverage self-

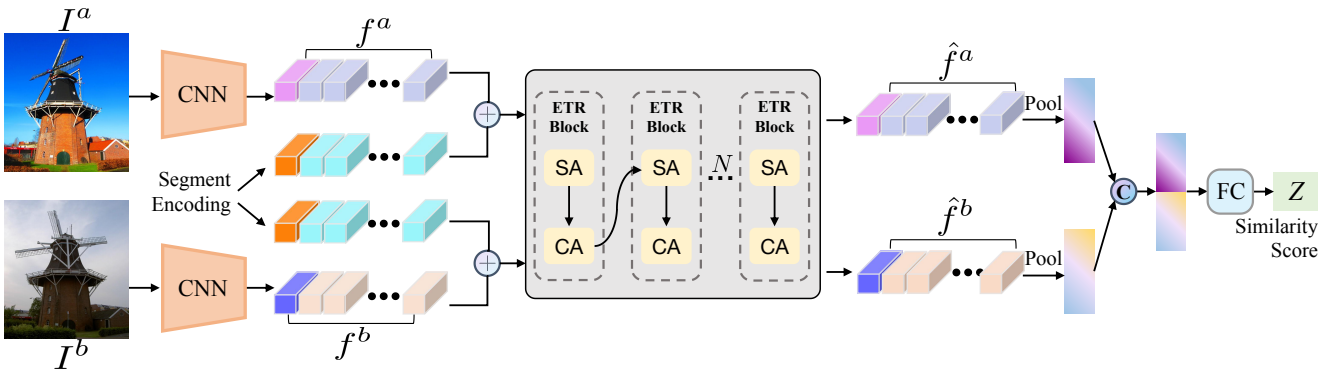

Figure 2. **Overview of the proposed method**. For a given input image pair (I^a, I^b), global and local descriptors extracted from a pre-trained CNN are concatenated to constitute the input of ETR, denoted as (f^a, f^b). They are then added with the segment encoding and processed by N ETR blocks, which consist of a self-attention layer and a cross-attention layer. The model finally produces a similarity score of (I^a, I^b). The model is trained to optimize a binary cross entropy loss.

attention and cross-attention layers for message passing between two sets of descriptors, which is proved to have better performance through our experiments.

3. Method

The overview of the ETR is illustrated in Figure 2. For a given query image, our method first uses its global descriptor to retrieve the top-k candidates. Then for the query image and each image in the candidate set, we construct an image pair and feed them into ETR to obtain a similarity score, which will be used to reorder the initial candidates. Different from previous CNN-based re-ranking methods (*e.g.*, [34, 20, 6, 16]), ours can learn distinct knowledge from global and local descriptors to directly compute the image similarity.

3.1. Feature Extraction

Note that our ETR is designed to focus on image re-ranking based on global and local descriptors. Theoretically, descriptors produced by CNN-based methods can be used as the input of the proposed model. Considering the feature extraction time and descriptors size, we propose two versions of ETR, named ETR-S and ETR-D, using DELG [6] and SuperPoint [10] as feature extractors, respectively. DELG is a unified framework for both global and local feature extraction, while SuperPoint only focuses on local feature extraction. We propose these two model variants to demonstrate the generality of the architecture and provide practical options for time-critical applications.

3.2. ETR Block

Before describing the proposed ETR Block, we first briefly introduce the Transformer architecture here as background. The Transformer contains a Multi-Head Self-Attention (MHA) layer and a fully connected Feed-Forward Network (FFN) layer. A self-attention layer first transforms the input vector into three different matrices, namely, Q, K, and V with the dimension d_q=d_k=d_v=d_{model}. The output of a self-attention layer is computed as:

$$\texttt{Attention}(Q, K, V) = \texttt{softmax}(\frac{Q \cdot K^T}{\sqrt{d_k}})V \quad (1)$$

MHA is a mechanism to boost the performance of the vanilla self-attention layer. The Q, K, V are linearly projected h times to $d_{q'}$, $d_{k'}$ and $d_{v'}$ dimensions respectively. Here, h is the number of heads, $d_{q'}$=$d_{k'}$=$d_{v'}$=d_{model}/h. The MHA takes Q, K, V as input and comprises multiple self-attention modules:

$$\begin{aligned} \texttt{MHA}(Q, K, V) &= \texttt{Concat}(head_1, ..., head_h)W^O \\ head_i &= \texttt{Attention}(Q_i, K_i, V_i) \end{aligned} \quad (2)$$

Here Q (and similarly K and V) is the concatenation of $\{Q_i\}_{i=1}^h$, and $W^O \in \mathbb{R}^{d_{model} \times d_{model}}$ is the linear projection matrix. FFN consists of two linear transformation layers with a nonlinear activation function in between and can be denoted as: $\texttt{FFN}(X) = W_2\sigma(W_1X)$, $W_1 \in \mathbb{R}^{d_{model} \times d_h}$ and $W_2 \in \mathbb{R}^{d_h \times d_{model}}$ are two parameter matrices, d_h is the hidden layer dimension. The σ represents the non-linear activation function. The output is computed as:

$$\begin{aligned} \overline{x} &= \texttt{LN}(x + \texttt{MHA}(Q, K, V)) \\ y &= \texttt{LN}(\overline{x} + \texttt{FFN}(\overline{x})) \end{aligned} \quad (3)$$

where LN represents a layer normalization function.

Now we introduce the proposed ETR Block, as shown in Figure 3. The ETR block interleaves a self-attention (self-attn) layer and a cross-attention (cross-attn) layer. For self-attention layer, the input vector Q, (K, V) come from the same input of an image pair (either f^a or f^b). The self-attention is responsible for capturing relationships between local descriptors of the image itself and enabling long-range

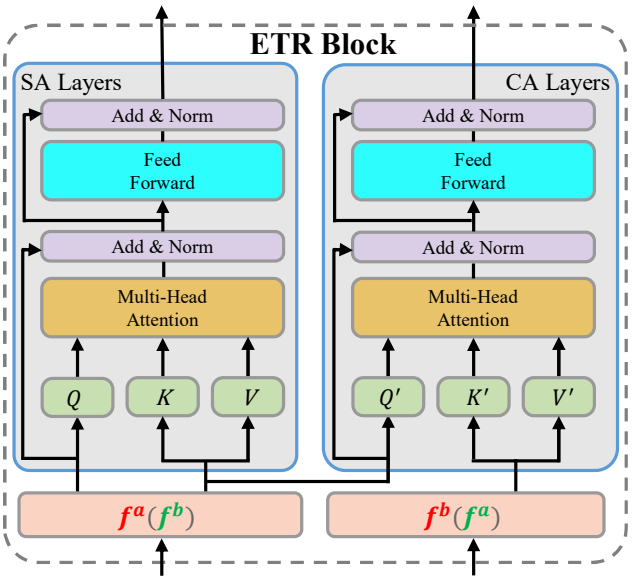

Figure 3. The proposed ETR block which consists of a self-attention layer and a cross-attention layer.

dependencies. For cross-attention layer, Q', (K' and V') come from either (f^a and f^b, marked with red) or (f^b and f^a, marked with green), depending on the direction of cross-attention. The cross-attention focuses on learning the relationships between local descriptors across images and exploring the similarity of the image pairs [40]. Given an input pair (f^a, f^b), the output ($\hat{f}^a$, $\hat{f}^b$) of an ETR block is obtained by:

$$\begin{aligned} y^a &= \texttt{self-attn}(Q_{f^a}, K_{f^a}, V_{f^a}) \\ y^b &= \texttt{self-attn}(Q_{f^b}, K_{f^b}, V_{f^b}) \\ \hat{f}^a &= \texttt{cross-attn}(Q_{y^a}, K_{y^b}, V_{y^b}) \\ \hat{f}^b &= \texttt{cross-attn}(Q_{y^b}, K_{y^a}, V_{y^a}) \end{aligned} \tag{4}$$

where `self-attn` and `cross-attn` are standard Transformer layers.

3.3. Model Architecture

Model input. For an input image I, its global descriptor and local descriptors are denoted as $X_g \in \mathbb{R}^{d_g}$, $X_l = \{X_{l,i} \in \mathbb{R}^{d_l}\}_{i=1}^{L}$ respectively, where L is the number of local descriptors. For ETR-D, which leverages DELG [6] as feature extractor, an additional scale factor $S_{l,i}$ list is required. The list contains a set of pre-defined image scales, each element of them is an integer, representing the scale from which the corresponding local descriptor $X_{l,i}$ is extracted. For ETR-S, which leverages SuperPoint [10] as feature extractor, since only one image scale is used, so the scale factor S_l is not required.

Given image pair (I^a, I^b), the global descriptor and local descriptors are denoted as (X_g^a, X_g^b) and (X_l^a, X_l^b) respectively. To better utilize the attention mechanism of Transformer, we arrange the input sequence (f^a, f^b) as follows:

$$\begin{aligned} f^a &= [\mathcal{H}_g(X_g^a); \mathcal{H}_l(X_{l,1}^a); ...; \mathcal{H}_l(X_{l,L}^a)] \\ f^b &= [\mathcal{H}_g(X_g^b); \mathcal{H}_l(X_{l,1}^b); ...; \mathcal{H}_l(X_{l,L}^b)] \\ \mathcal{H}_g(X_g) &= X_g W_g + \gamma \\ \mathcal{H}_l(X_l) &= X_l + \phi(S_l) + \gamma \end{aligned} \tag{5}$$

$W_g \in \mathbb{R}^{d_g \times d_l}$ is a parameter matrix to project X_g to d_l dimension, and γ is a segment embedding used in BERT [11] to distinguish the global descriptor and local descriptors. ϕ is a linear embedding function taking the scale index $S_{l,i}$ as input to obtain corresponding scale embedding. [;] denotes the concatenation operation. Different from RRT [35], we do not use position embedding and class token. For position embedding, we observe no benefit in performance gain. For class token, ablation studies have been conducted, see more details in Section 4.4.

With the input and ETR block, the complete pipeline can be described as:

$$\begin{aligned} f_{i+1}^a, f_{i+1}^b &= \texttt{ETR-Block}_i(f_i^a, f_i^b) \\ \hat{f}^a, \hat{f}^b &= f_N^a, f_N^b \\ z &= \texttt{Concat}([\texttt{Pool}(\hat{f}^a), \texttt{Pool}(\hat{f}^b)])W_z \end{aligned} \tag{6}$$

where $i = 0, ..., N-1$ (N is the number of ETR block), `Pool` is the pooling method (average pooling is adopted in this paper), and $W_z \in \mathbb{R}^{2d_l \times 1}$ is a linear projection matrix.

Supervision. We treat the image re-ranking as a classification task, the commonly used BCE loss is adopted as the training objective defined as follows:

$$\begin{aligned} \mathcal{L}(z, \hat{y}) &= -\frac{1}{n}\sum_{i=0}^{n}[\hat{y}\log(\sigma(z)) + (1-\hat{y})\log(1-\sigma(z))] \\ \hat{y} &= \mathbb{I}(I^a, I^b) \end{aligned} \tag{7}$$

Where n is the number of training image pairs, $\mathbb{I}(I^a, I^b)$ is an indicator function which equals to one when I^a and I^b represent the same place, or zero otherwise. $\sigma(z)$ is the sigmoid function to convert the output z to a probability.

4. Experiments

Training dataset. The ETR models are trained with the subsets of "v2-clean" split [46] of Google Landmarks v2 (GLDv2, [44]). GLDv2 is a benchmark for large-scale place recognition, which contains more than 4 million images annotated with labels. For ETR-S, we randomly sample 15,000 landmarks from GLDv2-clean where each landmark has at least 10 images and at most 100 images. This results in 450,508 images, which is 30% of the "v2-clean" split. ETR-D uses the same training set as RRT [35] for a fair comparison. This set contains 322,008 images which

Table 1. Comparison with state-of-the-arts on the benchmarks.

Method	MSLS val			Pitts30k			Tokyo 24/7			MSLS challenge		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
SFRS [15]	69.2	80.3	83.1	89.4	94.7	95.9	85.4	91.1	93.3	42.5	53.7	58.0
NetVLAD [1]	58.6	71.2	76.1	81.9	91.2	93.7	67.0	77.8	80.3	35.1	47.4	51.7
AP-GEM [29]	65.0	75.7	78.2	80.7	91.4	94.0	58.4	69.5	74.3	30.2	41.3	47.1
DELG global [6]	72.2	81.4	84.6	78.4	87.4	91.6	73.0	83.5	87.0	39.3	52.6	58.2
RRT [35]	72.4	86.5	89.0	80.7	90.7	93.9	86.7	94.0	94.9	39.1	55.4	63.0
SP-SuperGlue [10, 31]	78.4	82.8	84.2	87.2	94.8	96.4	88.2	90.2	90.2	50.6	56.9	58.3
DELG local [6]	**83.2**	**89.3**	89.5	**89.8**	**95.3**	**96.6**	86.4	92.4	93.0	52.2	61.9	65.4
Patch-NetVLAD-s [16]	77.8	84.3	86.5	87.5	94.5	94.8	70.2	78.7	82.2	48.1	59.4	62.3
Patch-NetVLAD-p [16]	79.5	86.2	87.7	88.7	94.5	95.9	86.0	88.6	90.5	48.1	57.6	60.5
ETR-S (ours)	80.5	86.5	88.9	83.1	91.1	93.8	**90.1**	93.0	94.6	**53.9**	**62.8**	**66.1**
ETR-D (ours)	79.3	88.0	**89.6**	84.2	91.6	93.8	89.2	**94.3**	**95.2**	50.6	62.1	65.8

are randomly sampled from 12,000 landmarks (each landmark has at least 10 images).

Testing dataset. To verify the generalization ability of our proposed model, we directly evaluate our model on several key benchmark datasets: MSLS [43], Pitts30k [38] and Tokyo 24/7 [37]. The Pitts30k contains 6816 query images and 1000 gallery images. The Tokyo 24/7 contains 76k gallery images and 315 query images taken using mobile phone cameras. These two datasets are extremely challenging since the query images were taken under varying conditions, including daytime, sunset and night, while the gallery images were only taken during daytime. The MSLS is a large-scale long-term place recognition dataset that contains 1.6M street-level images, which in particular includes simultaneous variations in all of the following: geographical diversity (30 major cities across the globe), season, time of day, date (over 7 years), viewpoint and weather [16]. We evaluate our model on the MSLS val set and MSLS challenge set which have 1.9k query images and 57k gallery images in total.

Evaluation metrics. We use the same evaluation Recall@N metric following [1], where a query image is determined to be correctly localized if at least one of the top N retrieved reference images is within the ground truth tolerance. For Pitts30k [38] and Tokyo 24/7 [37], the ground truth tolerance is 25m translational error. For MSLS [43], 25m translational and 40° orientation error. The recall is defined as the percentage of correctly localized query images.

4.1. Implementation Details

ETR-D settings. ETR-D uses DELG [6] as the pre-trained feature extractor. The DELG which leverages ResNet-50 [17] as backbone is adopted in our experiment. DELG unifies global and local feature extraction into one single deep model. The global descriptor is extracted at 3 scales $\{\frac{1}{\sqrt{2}}, 1, \sqrt{2}\}$ with dimensionality $d_g = 2048$. The local descriptors are extracted at 7 image scales (range from 0.25 to 2.0) in total, each with dimensionality $d_l = 128$. The original DELG model extracts top 1000 local descriptors with highest attention scores for each image. Following the RRT [35], we only leverage the top $L = 500$ local descriptors. To unify the dimensions, we use an extra linear projection layer $W_g \in R^{d_g \times d_l}$ to project the global descriptors to a dimension of 128. The model has $N = 3$ ETR blocks, The self-attn and cross-attn has $h = 4$ heads, d_q, d_k, d_v and d_{model} are set to 128. The hidden layer dimension in FFN layer is $d_h = 1024$. The model is trained using AdamW [22] optimizer and cosine learning rate decay policy with an initial learning rate of 1×10^{-3}. We train the model with a batch size of 196 for 30 epochs, in which 2 epochs for learning rate warm-up [21].

ETR-S settings. Since the local feature extraction process of DELG is still a high cost for large-scale systems, we propose ETR-S, which uses an efficient feature extractor SuperPoint [10]. In ETR-S, we do not incorporate the global descriptor term $\mathcal{H}_g(X_g)$ in Eq. 5. We also drop the scale embedding $\phi(S_l)$ since SuperPoint only performs local feature extraction at one image scale. We extract the top 1024 local descriptors with the highest attention scores for each image, each with dimensionality $d_l = 256$. The top 500 descriptors are used in our experiment ($L = 500$). For the architecture, we use 2 ETR blocks ($N = 2$) with 4 heads ($h = 4$). d_q, d_k, d_v are set to 256, d_h in FFN is set to 2048. The model is trained using AdamW [22] optimizer for 100 epochs, using a learning rate of 1×10^{-3}.

Model training. During training, for a query image, we first use the global descriptor to retrieve the top 100 candidates, from where the negative images are randomly sampled, which have a different label from the query image. The positive samples are randomly selected from the images sharing the same label as the query. Note that different from other methods [1, 16, 15], our model does not train on any of the training sets of the benchmarks.

Table 2. Compare ETR-S and ETR-D with RRT [35] on Pitts30k and Tokyo 24/7 datasets. All methods re-rank top-k (k=10, 50, 100, 200, 300) images retrieved by DELG global [6]. The original metrics of DELG global on Pitts30k are 78.4% R@1, 87.4% R@5, 91.6% R@10, and 73.0% R@1, 83.5% R@5, 87.0% R@10 on Tokyo 24/7.

Dataset	#reranked image	RRT [35]			ETR-S (ours)			ETR-D (ours)		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Pitts30k	10	81.1	89.5	91.6	82.8	**89.6**	**91.6**	**83.8**	89.6	91.6
	50	80.8	90.5	93.8	82.6	91.0	**93.8**	**84.3**	**91.2**	93.7
	100	80.7	90.7	**93.9**	83.1	91.1	93.8	**84.2**	**91.6**	93.8
	200	80.5	90.6	93.9	82.9	91.0	93.7	**84.2**	**91.7**	**93.9**
	300	80.4	90.7	**93.9**	88.6	93.3	93.7	**84.1**	**91.8**	93.8
Tokyo 24/7	10	83.2	85.7	87.0	**86.0**	**89.5**	**87.0**	84.1	86.0	87.0
	50	85.1	92.4	92.7	**89.8**	**94.9**	**94.9**	87.9	93.3	94.9
	100	86.7	94.0	94.9	**90.1**	93.0	94.6	89.2	**94.3**	**95.2**
	200	87.6	94.0	95.6	88.9	93.0	93.3	**89.2**	**94.9**	**96.5**
	300	87.9	94.3	95.9	88.6	93.3	93.7	**89.5**	**95.6**	**96.5**

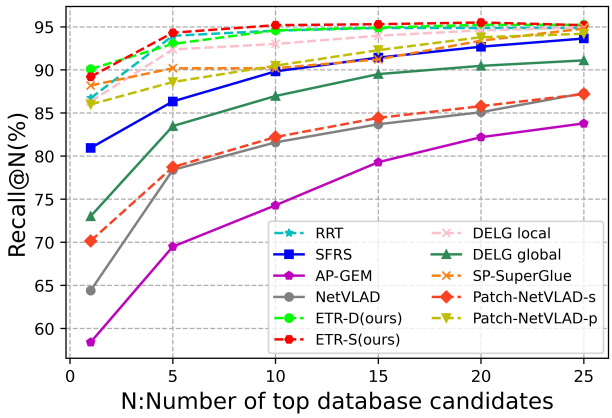

Figure 4. Comparison with state-of-the-art methods on Tokyo 24/7 dataset. The global-only retrieval results are depicted in solid line, while re-ranking results are depicted in dash line. our method outperforms all the global retrieval and re-ranking methods.

4.2. Comparison with State-of-the-arts

We compare ETR with several state-of-art approaches. Among them, the first group contains global-only retrieval methods, including NetVLAD [1], SFRS [15], DELG global [6] and AP-GEM [29]. We also compare the re-ranking methods including Patch-NetVLAD [16], SP-SpuerGlue [31, 6], RRT [35] and DELG local [6]. For Patch-NetVLAD, we test both its speed-oriented and performance-focused configurations, denoted as Patch-NetVLAD-s and Patch-NetVLAD-p respectively. For SP-SuperGlue, which re-ranks candidates by using SuperGlue [31] to identify matches from local descriptors extracted by SuperPoint [10]. Patch-NetVLAD and SP-SuperGlue re-rank top-100 images retrieved by NetVLAD, while RRT, DELG local, ETR-S and ETR-D re-rank top-100 images retrieved by DELG global.

Table 1 shows the quantitative results of our method compared with other approaches. ETR-D outperforms all the global-only retrieval methods, SFRS, NetVLAD, AP-GEM and DELG global on MSLS val, Tokyo 24/7 and MSLS challenge datasets, on average by 7.3%, 19.5%, 21.8%, and 11.5% (all percentages are absolute increase for R@1) respectively. ETR-S achieves similar results with ETR-D on MSLS val, Tokyo 24/7 and Pitts30k test dataset, and outperforms all compared methods on MSLS challenge dataset. For Pitts30k dataset, our method does not perform as well as the strong baseline method SFRS. Note that SFRS is finely trained on Pitts30k dataset using proposed self-supervising fine-grained region similarities, which can greatly improve performance, while our methods are only trained on a subset of GLDv2.

ETR also yields competitive results compared with the two-stage approaches. ETR-D achieves the best performance on almost all four datasets compared to RRT, especially the MSLS val set and MSLS challenge set, with an absolute improvement of 6.9% and 11.5% on R@1 respectively. It is worth noting that ETR-D and RRT are trained under the same dataset for a fair comparison. ETR-D outperforms Patch-NetVLAD in MSLS val set, MSLS challenge set, and Tokyo 24/7 while performing worse in Pitts30k. Note that Patch-NetVLAD was trained on Pitts30k and MSLS dataset, while our method is not fine-tuned on any of these datasets. Besides, Patch-NetVLAD requires huge storage space and extremely large computational cost due to its multi-scale features and high dimensionality, which is unsuitable for resource-constrained and time-sensitive systems. ETR-S achieves competitive results in Pitts30k and is 633 times and 12,183 times faster than SP-SuperGlue and DELG local in terms of re-ranking latency (see Table 3). For a more intuitive comparison of ETR with other methods, see Figure 4.

Comparison with RRT. RRT [35] is also a Transformer-based model for the image re-ranking, therefore we con-

Table 3. Feature extraction and re-ranking (top-100) latency, memory requirements for different re-ranking methods. Latency is measured on an NVIDIA GeForce RTX 3090 GPU. We conduct these experiments on Pitts30k [38] test dataset.

Method	**Extraction latency** (ms)	**Re-ranking latency** (ms)	**Memory** (MB)
DELG local [6]	152	73100	0.9
Patch-NetVLAD-s [16]	21	200	1.9
Patch-NetVLAD-p [16]	487	7700	44.2
SP-SuperGlue [10, 31]	7	3800	0.7
RRT [35]	152	8	0.5
ETR-D (ours)	152	14	0.5
ETR-S (ours)	7	6	0.7

Table 4. Performance of different pooling methods on MSLS val and Tokyo 24/7 datasets for ETR-D.

Pooling Method	**Tokyo 24/7**			**MSLS val**		
	R@1	R@5	R@10	R@1	R@5	R@10
CLS	58.9	70.6	84.7	55.7	71.5	77.5
GeM	84.8	94.3	94.6	76.4	85.5	87.7
Max Pool	66.7	87.9	92.1	66.7	79.1	83.9
Average Pool	**89.5**	**95.6**	**96.5**	**79.3**	**88.0**	**89.6**

sider RRT as the main baseline to compare in more details. For a fair comparison, ETR-D and RRT are trained on the same dataset and use the same number (*e.g.* 500) R50-DELG [6] descriptors. ETR-S uses 500 SuperPoint local descriptors. All models re-rank the top-k (k=10,50,100,200,300) images retrieved by DELG global. Table 2 presents the results on two benchmark datasets Pitts30k [38] and Tokyo 24/7 [37].

ETR-D achieves the best re-ranking performance on almost all conditions across two datasets compared with DELG global and RRT. For example, when re-ranking the top 300 neighbors, our model considerably improves over the DELG global method, with an average improvement of 11.1%, 8.3%, 5.9% on R@1, R@5, R@10 respectively, 2.7%, 1.2%, 0.3% absolute increase when compared with RRT. ETR-S also surpasses the DELG global by noticeable margins and achieves comparable results with ETR-D. Taking the average of all datasets, ETR-S outperforms RRT with absolute gains of 3.3%, 1.5%, 2.2% on R@1, R@5, R@10 score when re-ranking the top 50 images. Such improvements show that our proposed self- and cross-attention layers can better capture underlying relationships encoding in descriptors compared with the original Transformer architecture used in RRT.

4.3. Latency and Memory

In practical VPR applications, latency and storage consumption are important factors that must be taken into account. Table 3 shows the computational time and memory footprint for all compared techniques. The **extraction latency** represents the time to extract features for a single image, while the **memory** is the size of the extracted features.

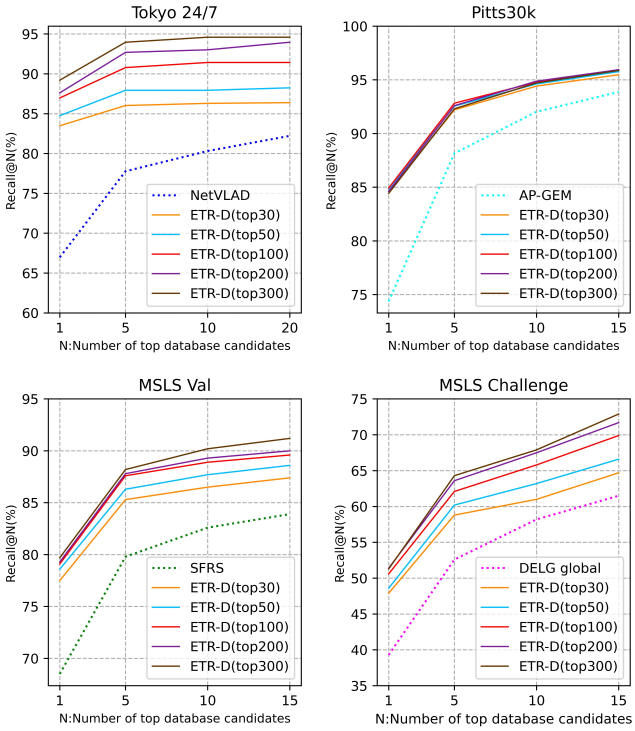

Figure 5. The re-ranking performance of our model on Tokyo 24/7 [37], MSLS val [43], Pitts30k [37] and MSLS challenge [43], base on global retrieval results generated by NetVLAD [1], SFRS [15], AP-GEM [29] and DELG global [6] respectively. The global-only retrieval results are depicted in dot line, while our re-ranking results are depicted in solid line. Our approach can significantly improve the retrieval metrics of different global-only retrieval methods, showing strong generalization ability.

The **re-ranking latency** is the time required to re-rank the top 100 neighbors for a query image. ETR-D is about 550 times and 271 times faster than Patch-NetVLAD-p [16] and SP-SuperGlue [10, 31] in terms of re-ranking latency, and 88.4 times and 1.4 times smaller in memory consumption. The Patch-NetVLAD-p extracts the high dimensionality (dim = 4096) patch features at multi-scale (*e.g.* patch sizes = 2, 5, 8), which leads to an extremely large memory footprint. ETR-S is about 1.3 times and 33.3 times faster than RRT and Patch-NetVLAD-s. Our method is orders of magnitude faster than traditional methods in re-ranking latency, which is more suitable for practical scenarios.

Both ETR-D and RRT [35] use half of the DELG local descriptors, so the memory consumption is nearly half of the DELG local method. RRT [35] is 1.75 times faster than ETR-D, this is because RRT concatenates the two sets of descriptors from the image pair into one single sequence, so the output can be obtained in just one single forward-pass. Our method leverages self- and cross-attention to pass messages between two sets of descriptors, which requires two forward-pass to obtain the output.

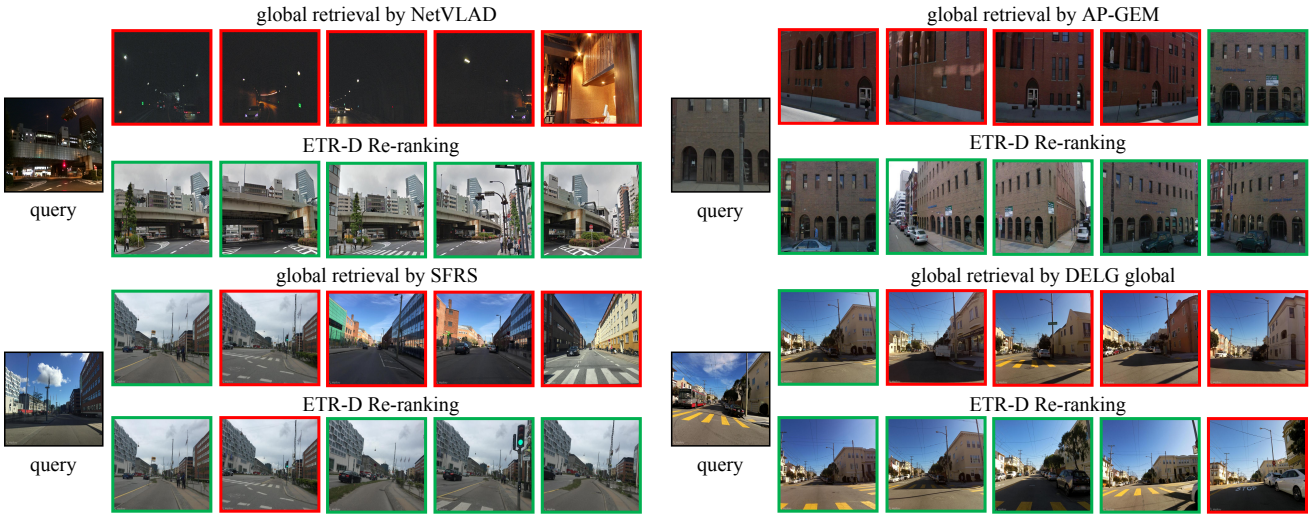

Figure 6. Qualitative examples from Tokyo 24/7 [37], Pitts30k [38], MSLS val [43] and MSLS challenge datasets. For each query, the top-5 neighbors ranked by the global retrieval and re-ranked by ETR-D are presented. Correct/incorrect neighbors are marked with green/red borders. The global-only methods show a tendency to retrieve images with a similar global layout as the query, while our full re-ranking approach can capture a more fine-grained matching between images. The top left one is the most representative example, which contains very severe day-night changes.

4.4. Ablation Study

Re-ranking generalization ability. We show that ETR can be considered as a general re-ranking method. To verify the generalization capability of our method, we use the model to re-rank the global results retrieved by 4 different methods, NetVLAD, SFRS, AP-GEM and DELG global on 4 different datasets includes Tokyo 24/7, MSLS val, Pitts30k and MSLS challenge dataset. The ETR-D and ETR-S achieve similar results in this experiment, we only show the results of ETR-D, see Figure 5 for details. our model can significantly improve the performance of the four global retrieval methods across all four datasets, with an absolute increase of 20.0%, 10.6%, 10.3%, 12.0% on R@1 respectively when re-ranking the top 100 neighbors. Besides that, by re-ranking more neighbors (*e.g.* top-200, top-300), the performance can be further improved. The results demonstrate that our model showing excellent generalization capability and can be used as a plug and play module to replace the traditional time-consuming re-ranking methods. Figure 6 illustrates qualitative examples retrieved by global-only methods and our re-ranking approach. The global-only methods can retrieve images that are similar in general and can not handle severe day-night and viewpoint changes, as shown in the first row. ETR can successfully perform matching and show great robustness to appearance and illumination changes, which can significantly improve the performance over the global-only retrieval.

Choice of pooling methods. In Table 4, we provide a comparison between different feature pooling methods. we observe that utilizing average pooling achieves the best performance on Tokyo 24/7 and MSLS val dataset. There is a slight drop in performance when replacing GeM (we use a pooling exponent value of p = 3) with average pooling. While the performance degrades significantly when using the class token to replace average pooling. This reveals that all output tokens are important for the final feature representation. Note that we conduct these experiments using the same training settings.

5. Conclusion

We proposed ETR, a novel Transformer-based re-ranking mothod that leverages self- and cross-attention layers to directly explore the similarity of an image pair. The ETR model is lightweight and can be easily parallelized such that re-ranking the top-k images requires just a single shot. We show that ETR outperforms state-of-the-arts on several VPR datasets. Moreover, ETR can be regarded as a general re-ranking model to further improve the performance of other global retrieval methods while requiring less computation time and memory consumption. ETR is efficient and well suited to systems emphasis on computational efficiency and real-time execution such as autonomous driving navigation and mobile robot localization.

6. Acknowledgement

This work was supported by National Key R&D Program of China (Grant No. 2019YFB2102800) and Shanghai Archives Research Program (2108).

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *CVPR*, pages 5297–5307, 2016.

[2] Yannis Avrithis and Giorgos Tolias. Hough pyramid matching: Speeded-up geometry re-ranking for large scale image retrieval. *IJCV*, 107(1):1–19, 2014.

[3] Artem Babenko and Victor Lempitsky. Aggregating local deep features for image retrieval. In *ICCV*, pages 1269–1277, 2015.

[4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *ECCV*, pages 404–417, 2006.

[5] Luis G Camara, Carl Gäbert, and Libor Přeučil. Highly robust visual place recognition through spatial matching of cnn features. In *ICRA*, pages 3748–3755, 2020.

[6] Bingyi Cao, Andre Araujo, and Jack Sim. Unifying deep local and global features for image search. In *ECCV*, pages 726–743, 2020.

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020.

[8] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *CVPR*, pages 403–412, 2017.

[9] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, pages 4690–4699, 2019.

[10] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPR Workshops*, pages 224–236, 2018.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2018.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

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

[14] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[15] Yixiao Ge, Haibo Wang, Feng Zhu, Rui Zhao, and Hongsheng Li. Self-supervising fine-grained region similarities for large-scale image localization. In *ECCV*, pages 369–386, 2020.

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

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[18] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, pages 3304–3311, 2010.

[19] Hervé Jégou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Pérez, and Cordelia Schmid. Aggregating local image descriptors into compact codes. *IEEE Trans. PAMI*, 34(9):1704–1716, 2011.

[20] Zakaria Laskar, Iaroslav Melekhov, Hamed Rezazadegan Tavakoli, Juha Ylioinas, and Juho Kannala. Geometric image correspondence verification by dense pixel matching. In *WACV*, pages 2521–2530, 2020.

[21] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv:1608.03983*, 2016.

[22] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv:1711.05101*, 2017.

[23] David G Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[24] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *ICCV*, pages 3456–3465, 2017.

[25] Eng-Jon Ong, Syed Sameed Husain, Mikel Bober-Irizar, and Miroslaw Bober. Deep architectures and ensembles for semantic video classification. *IEEE Trans. CSVT*, 29(12):3568–3582, 2018.

[26] Kohei Ozaki and Shuhei Yokoo. Large-scale landmark retrieval/recognition under a noisy and diverse dataset. *arXiv:1906.04087*, 2019.

[27] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007.

[28] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation. *IEEE Trans. PAMI*, 41(7):1655–1668, 2018.

[29] Jerome Revaud, Jon Almazán, Rafael S Rezende, and Cesar Roberto de Souza. Learning with average precision: Training image retrieval with a listwise loss. In *ICCV*, pages 5107–5116, 2019.

[30] Jerome Revaud, Philippe Weinzaepfel, César De Souza, Noe Pion, Gabriela Csurka, Yohann Cabon, and Martin Humenberger. R2d2: repeatable and reliable detector and descriptor. In *NeurIPS*, 2019.

[31] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *CVPR*, pages 4938–4947, 2020.

[32] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[33] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *CVPR*, pages 8922–8931, 2021.

[34] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense

matching and view synthesis. In *CVPR*, pages 7199–7209, 2018.

[35] Fuwen Tan, Jiangbo Yuan, and Vicente Ordonez. Instance-level image retrieval using reranking transformers. In *ICCV*, pages 12105–12115, 2021.

[36] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. Particular object retrieval with integral max-pooling of cnn activations. In *ICLR*, 2016.

[37] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In *CVPR*, pages 1808–1817, 2015.

[38] Akihiko Torii, Josef Sivic, Tomas Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *CVPR*, pages 883–890, 2013.

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[40] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching. *arXiv:2203.09645*, 2022.

[41] Ruotong Wang, Yanqing Shen, Weiliang Zuo, Sanping Zhou, and Nanning Zhen. Transvpr: Transformer-based place recognition with multi-level attention aggregation. *arXiv:2201.02001*, 2022.

[42] Ruikui Wang, Ruiping Wang, Shishi Qiao, Shiguang Shan, and Xilin Chen. Deep position-aware hashing for semantic continuous image retrieval. In *WACV*, pages 2493–2502, 2020.

[43] Frederik Warburg, Soren Hauberg, Manuel Lopez-Antequera, Pau Gargallo, Yubin Kuang, and Javier Civera. Mapillary street-level sequences: A dataset for lifelong place recognition. In *CVPR*, pages 2626–2635, 2020.

[44] T. Weyand, A. Araujo, B. Cao, and J. Sim. Google Landmarks Dataset v2 - A Large-Scale Benchmark for Instance-Level Recognition and Retrieval. In *CVPR*, 2020.

[45] Min Yang, Dongliang He, Miao Fan, Baorong Shi, Xuetong Xue, Fu Li, Errui Ding, and Jizhou Huang. Dolg: Single-stage image retrieval with deep orthogonal fusion of local and global features. In *ICCV*, pages 11772–11781, 2021.

[46] Shuhei Yokoo, Kohei Ozaki, Edgar Simo-Serra, and Satoshi Iizuka. Two-stage discriminative re-ranking for large-scale landmark retrieval. In *CVPR Workshops*, pages 1012–1013, 2020.

[47] Xiwu Zhang, Lei Wang, and Yan Su. Visual place recognition: A survey from deep learning perspective. *Pattern Recognition*, 113:107760, 2021.