Attention-Aware Age-Agnostic Visual Place Recognition

Jiahui Li
Attention-Aware Age-Agnostic Visual Place Recognition

by

Jiahui Li

to obtain the degree of Master of Science at the Delft University of Technology, to be defended publicly on Wednesday August 28, 2019 at 13:00

Student number: 4734769
Project duration: November 2018 - August 2019
Thesis committee: Prof. dr. M. J. T. Reinders, TU Delft, chair
Dr. J. C. van Gemert, TU Delft, supervisor
Dr. L. Nan, TU Delft, committee member

An electronic version of this dissertation is available at http://repository.tudelft.nl/.
The thesis report documents the results of my master’s thesis project. First we will show the main content, the scientific paper, including motivation, related work, methodology, experiments and results of this research project. The latter chapters introduce some related knowledge needed to understand my thesis work.

Throughout the thesis work I have received a great deal of support and guidance. I would like to first thank my supervisor Dr. J. C. van Gemert, whose formulate the research topic and steered me in the right the direction during my research through each stage of the process.

I would like to acknowledge my daily supervisors, Dr. S. Khademi and Z. Wang for their wonderful help. Dr. S. Khademi first helped me with formulating the basic methodology for the research problem, and was always keen to know what I was doing and how I was proceeding. Z. Wang always supported me in verifying my own ideas and provided valuable insight. The scientific paper would not have been completed without them.

I would also like to thank my family and friends, whose encouragement and support have made my accomplishments possible.

Jiahui Li
Delft, August 2019
# Contents

1 **Scientific Paper** ......................................................... 1

2 **Introduction** .......................... 12
   2.1 Motivation .......................................................... 12
   2.2 Research objectives ............................................. 13
   2.3 Outline .............................................................. 13
   References .............................................................. 13

3 **Background on Deep Learning** ........................................ 15
   3.1 Introduction to Deep Learning .................................. 15
   3.2 Convolutional Neural Network ................................... 17
   3.3 Triplet Network .................................................... 19
      3.3.1 Triplet mining ............................................... 20
   References .............................................................. 20

4 **Domain Adaptation** ..................................................... 23
   4.1 Introduction to Domain Adaptation ............................... 23
   4.2 Deep Domain Adaptation .......................................... 24
   References .............................................................. 25

5 **VLAD: vector of locally aggregated descriptors** ....................... 27
   5.1 Image vector representation ...................................... 27
   5.2 Deep architecture with VLAD ..................................... 28
   References .............................................................. 29

6 **Attention Module** ....................................................... 31
   6.1 Introduction to Attention Module ................................. 31
   6.2 Attention-aware VLAD Layer ....................................... 32
   6.3 Geometric Interpretation of Attention ............................ 32
   References .............................................................. 33
Scientific Paper
Attention-Aware Age-Agnostic Visual Place Recognition

Jiahui Li
Delft University of Technology
Delft, The Netherlands
jiahuili67210@gmail.com

Abstract

A cross-domain visual place recognition (VPR) task is proposed in this work, i.e., matching images of the same architectures depicted in different domains. VPR is commonly treated as an image retrieval task, where a query image from an unknown location is matched with relevant instances from geo-tagged gallery database. Different from conventional VPR settings where the query images and gallery images come from the same domain, we propose a more common but challenging setup where the query images are collected under a new unseen condition. The two domains involved in this work are contemporary street view images of Amsterdam from the Mapillary dataset (source domain) and historical images of the same city from Beeldbank dataset (target domain). We tailored an age-invariant feature learning CNN that can focus on domain invariant objects and learn to match images based on a weakly supervised ranking loss. We propose an attention aggregation module that is robust to domain discrepancy between the train and the test data. Further, a multi-kernel maximum mean Discrepancy (MK-MMD) domain adaptation loss is adopted to improve the cross-domain ranking performance. Both attention and adaptation modules are unsupervised while the ranking loss uses weak supervision. Visual inspection shows that the attention module focuses on built forms while the dramatically changing environment are less weighed. Our proposed CNN achieves state of the art results (99% accuracy) on the single-domain VPR task and 20% accuracy at its best on the cross-domain VPR task, revealing the difficulty of age-invariant VPR.

1. Introduction

Recently, there has been interest among the computer vision researchers to solve the visual place recognition (VPR) task in the form of image retrieval [3, 12, 20, 24, 27, 41, 48]. In [38], the discriminative visual cues learned for visual place classification task are investigated. Interestingly, CNN filters learn human-like discriminative visual cues to recognize a place, including built forms, signs or vegetation. Among these discriminative attributes, buildings are the most robust that remain, more or less, invariant during the changes in day and night lighting, different seasons and even years. However, CNNs are still influenced by irrelevant objects like roads and the sky. In this work, we introduce a CNN model with attention aggregation module to focus on domain invariant features, i.e. buildings, for the cross-domain VPR task. We will demonstrate that our work can be further combined with multi-kernel Maximum Mean Discrepancy (MK-MMD) loss to obtain better domain adaptation results. The images from the two domains with a large time lag are depicted in Fig.1, being historical images (queries) and current street view images (gallery) of Amsterdam.

The VPR task is commonly formulated as content based image retrieval (CBIR), i.e., sorting the geo-tagged gallery images by their distances to the unknown query image. The query is then labeled based on its best matching image in the gallery. Deep image representation learning is currently state of the art for almost all CBIR settings. Among the deep feature learning methods, distance learning CNNs are the most popular ones [13, 19]. Nevertheless, supervised deep distance learning requires similar and dissimilar image pairs for training. In this work, image pair labels are not available and we only have access to geo-tagged images from the Mapillary street view imagery and thus a weakly supervised deep feature learning is used, similar to the work of NetVLAD[3].

Different from [3], our queries are historical images which are not geo-tagged and exhibit a domain discrepancy between training data and test data. Age-agnostic place recognition that is addressed in this paper is a more challenging problem firstly due to the lack of image pair labels for training, secondly due to the domain shift between the
Figure 1: Correctly retrieved images with our proposed method. The top row illustrate the general place recognition in the same domain: both the query (left) and gallery image (right) are from the same dataset. The bottom row shows the cross-domain place recognition task where the query (left) is from the Beeldbank dataset and the gallery image (right) is from the Mapillary dataset.

gallery and query images caused by the change of scenery over a large time gap and thirdly due to the outliers in target domain. Different technologies of photography, equipment and processes used in the production of photos in the past also contribute to this domain shift. Fig1 shows the general and the age-agnostic place recognition task.

We are inspired by [31], which introduces an attention module into NetVLAD for the classification task to address the unequal importance of local features in VLAD feature aggregation layer. In our work, a new attention aggregation technique is proposed to weigh both global VLAD descriptors and local descriptors. A domain adaptation loss based on MK-MMD is additionally introduced to achieve better cross-domain performance. Note that both the attention and the domain adaption modules are unsupervised and thus no labels are required.

Our attention-aware architecture is depicted in Fig.2 which consists of three modules and a shared convolutional neural network for feature extraction (AlexNet cropped before conv5). The attention module is a single convolutional layer followed by softplus activation function, transforming the feature map to a heatmap. This heatmap contains attention scores for the deep features. The VLAD module aggregates deep features in the attention-aware scheme by assigning attention scores to both local and global descriptors. The unsupervised domain adaptation module is additionally used to learn domain-invariant features. Our oblation studies show that both modules are important to reach state of the art results. Our speculation is that MK-MMD loss aligns the photo styles while attention module focuses on domain invariant contents.

Our contributions are summarized as:

- To the best of our knowledge, this is the first large scale (40k) image database for age-invariant visual place recognition task. We manually annotated 104 matching pairs between historical images and current street view images only for evaluation purpose.

- A new attention aggregation scheme is proposed to combine both the local and global image descriptors (Section 3.2).

- We combined the MK-MMD domain adaptation loss with the ranking loss to learn domain-invariant features for cross-domain VPR task. (Section 3.3).

We tested our proposed model on conventional VPR task and our experiments show the state of the art results on Mapillary dataset compared to other competitors. Detailed results and ablation studies will be presented in Section 4.5. The comparison of single-domain and cross-domain results reveals the difficulty of age-agnostic place recognition task.

2. Related Work

The performance of VPR as an image retrieval problem depends on the ranking accuracy w.r.t. a similarity metric. The query location is suggested based on the top M similar images (annotated with geo-tags). To extract good features for indexing, traditional works focus on handcrafted features such as SIFT[28]) and SURF[8]. Some other efficient methods are based on the aggregation of local gradient-based descriptors like Fisher Vectors [34] and VLAD[21]. [24] is a SURF based model which improves the performance by detecting and removing ‘confusing objects’. [41] uses SIFT to detect the repetitive patterns in the image which is representative for buildings. [30] focuses on matching images that have large view point changes by generating artificial views of a scene for the training process.

Recent works suggest that a CNN trained on a large scale dataset as a feature extractor outperforms hand crafted features on various tasks [11, 12, 32, 36]. In turn, [6] shows that features in the early layers of a CNN trained for image classification can be effectively used as visual descriptors for image retrieval. LIFT [47] is a learning pipeline for feature extraction which introduces an end-to-end unified network for detection, orientation estimation, and feature description. [40] proposes a global image representation by the regional maximum activation of convolutional layers (R-MAC) well-suited for place recognition. [12] proposes
novel CNN-based features designed for place recognition by detecting salient regions and extracting regional representations as descriptors. NetVLAD [4] introduces a novel triplet ranking loss together with a VLAD aggregation layer that can learn powerful representations for the VPR task in an end-to-end manner. A known disadvantage of NetVLAD lies in its global feature aggregation. [16] proposes a region proposal network to learn which regions should be pooled to form the final global descriptor. Similar to [4], we use current geo-location tags for weakly supervised feature learning using triplet distance learning network. However, we do not have access to matched image pairs from the two domains for supervised training, i.e., matched historical and contemporary images. To address this domain mismatch between the test and train data, we need to promote domain-invariant feature learning.

We tailor an attention aggregation model that can boost the cross-domain performance for our specific task, age-agnostic urban scene matching. Attention model is broadly used in natural language processing [14, 43] and computer vision tasks [18, 26, 32, 37, 39, 48, 45]. [23] shows that attention model can also be adopted to benefit metric learning. [32] proposes an attention mechanism to select key points for matching. Attention model is considered to be effective for domain adaptation as well [22, 44]. Our attention model is implemented in an unsupervised way which means no ground truth score maps are available for training. The learning process of the attention module is guided by the image retrieval ranking loss.

Given two different domains, unsupervised deep domain adaptation schemes [25, 42] are mostly used to enhance the performance of CNNs on target domain by using labels only from the source domain. Among the vast amount of literature on deep domain adaption for classification tasks, the Maximum Mean Discrepancy (MMD) loss is introduced by [10] to minimize the domain discrepancy by projecting data into a kernel space. Later [17] proposed multi-kernel MMD (MK-MMD) which uses linear combination of multiple kernels. We adopt MK-MMD loss as an additional domain adaptation module for our attention aggregation model. Similarly, we feed untagged historical images of Amsterdam to the adaptation layer in an unsupervised manner.

3. Method

Our proposed model consists of three modules for feature extraction, namely a weakly supervised image retrieval module with a triplet ranking loss (Section 3.1), an attention aggregation module(Section 3.2) and an unsupervised domain adaptation module with MK-MMD loss (Section 3.3). MK-MMD loss constrains the feature maps after the last convolution layer (conv5). The final loss function for training, $L_u$, can be expressed as:
\[ L_a = L_r + \alpha M(D_s, D_t) \quad (1) \]

where \( M(D_s, D_t) \) is the MK-MMD loss term, \( D_s \) and \( D_t \) denote the source domain and target domain, \( L_r \) is the triplet ranking loss used in NetVLAD [3], \( \alpha \) is the weight that trades off the image retrieval loss and the domain adaptation loss.

### 3.1. Image retrieval with weak supervision

We use NetVLAD [3] as our baseline model which tackles the weakly supervised image retrieval task with a triplet ranking loss. NetVLAD considers the generated \( H \times W \times D \) feature maps as a set of \( \mathbb{N}(H \times W) \times D \) local descriptors where \( \mathbb{N} \) is the number of local descriptors and \( D \) is the dimension. Latter, a soft clustering is used to store the residual information contained in the descriptors to form \( K \times D \) final descriptors denoted as \( V \) where \( K \) is the number of cluster centers. \( V(j, k) \) can be expressed as:

\[ V(j, k) = \sum_{i=1}^{N} a_k(x_i)(x_i(j) - c_k(j)), \quad (2) \]

where \( j \in \{1, \ldots, D\} \) is the \( j \)-th dimension of a descriptor \( \{x_i\} \), \( k \in \{1, \ldots, K\} \) is the \( k \)-th cluster center, and \( a_k(x_i) \) is the soft assignment of the descriptor \( x_i \) to \( k \)-th cluster center \( c_k \). In Eq.1, A weakly supervised triplet ranking loss \( L_r \) is used to govern the learning process of descriptors that ensures the Euclidean distance between the query image and the best potential positive images are smaller than the Euclidean distance between the query image and all the negative pairs (based on geo-tags).

\[ L_r = \sum_{j} l(\min_i d^2_g(q, p^j_i) + m - d^2_g(q, n^j_i)), \quad (3) \]

where, \( q \) denotes the query image and \( p^j_i \) are potential positive images. \( \min_i d^2_g(q, p^j_i) \) denotes the best matching pairing with shortest distance \( d_g \). In turn, \( n^j_i \) are all negative image pairs and \( m \) is the distance margin to be maintained. The function \( l \) is the hinge loss which penalizes the pairs that violate the margin.

### 3.2. Attention module

The triplet network for image retrieval task produces feature maps with the dimension of \( H \times W \times D \). The inserted attention module consists of a \( 1 \times 1 \) convolutional layer with coefficients \( w_a \in \mathbb{R}^{D \times 1} \) and a softplus activation function. This convolutional layer will produce an attention score map \( H_a \) with spatial size \( H \times W \), which could be interpreted as the weight \( \{w_i\} \) for each descriptor \( \{x_i\} \). [31] proposed an attention aware aggregation scheme \( A_1 \) as:

\[ V(j, k)_{A_1} = \sum_{i=1}^{N} w_i a_k(x_i)(x_i(j) - c_k(j)), \quad (4) \]

where \( w_i \in w_a \). Note that the VLAD module first assigns the local descriptors \( \{x_i\} \) to \( K \) cluster centers \( \{c_k\} \), then computes the residuals of each descriptor \( x_i - c_k \) to its cluster center and assigns the weight \( a_k \) of descriptor \( x_i \) to cluster \( c_k \) proportional to their proximity.

In Eq.4, the global descriptors (residuals) are weighed after clustering. However, the VLAD descriptor is very sensitive to cluster centers [5] since it defines the origin of coordinates system to a cluster. Under this circumstance, we propose to weigh the local descriptors according to attention scores before performing clustering. The soft-assignment term \( a_k \) is re-calculated based on the newly weighed descriptors. Our proposed aggregation scheme \( A_2 \) can be formulated as

\[ V(j, k)_{A_2} = \sum_{i=1}^{N} w_i a_k(x_i)(x_i(j)w_i - c_k(j)). \quad (5) \]

The difference between \( A_1 \) and \( A_2 \) is that \( A_1 \) assigns the attention scores after clustering the descriptors to multiple centers so the attention scores are only used to weight the residuals but \( A_2 \) first uses the attention scores to filter out uninteresting regions in the individual local descriptors and then performs the same step as \( A_1 \). Considering that the reweighting of individual descriptors may remove information that are useful for global descriptor generation, we aggregate the two attention schemes linearly:

\[ V(j, k)_{\text{our}} = V(j, k)_{A_1} + V(j, k)_{A_2}. \quad (6) \]

### 3.3. Domain adaptation module

We use MK-MMD loss [17] with five Gaussian kernels of different bandwidths for unsupervised domain adaptation. The loss minimizes the distance between the expectation of the kernel mappings \( \phi(.) \) of the descriptors in the source domain \( x^s_i \) and the target domain \( x^t_i \):

\[ M(D_s, D_t) = \sum_{i=1}^{N} ||\mathbb{E}(\phi(x^s_i)) - \mathbb{E}(\phi(x^t_i))||_2. \quad (7) \]

The MK-MMD loss guides the CNN to learn a latent space where the two domains are not distinctive, i.e., the gap between the statistical means of these two domains are closed in the reproducing kernel Hilbert space (RKHS).

### 4. Experiment

#### 4.1. Dataset

We construct a cross-domain dataset with two sources of data to evaluate our proposed method, namely the street view panorama images of Amsterdam city from the Mapillary dataset[2] and the Beeldbank dataset[1] containing historical images from Amsterdam city archives.
**Table 1: Mapillary40k→Beeldbank dataset**, the source domain is Mapillary40k and the target domain is Beeldbank denoted by M and B, respectively. Beeldbank-train is only used for unsupervised domain adaptation. Cross-domain VPR requires matching query images from Beeldbank to gallery images from Mapillary40k.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Gallery</th>
<th>Query</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Source</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mapillary40k-train</td>
<td>20,884</td>
<td>2,320</td>
</tr>
<tr>
<td>single-domain-test</td>
<td>18,980 (M)</td>
<td>2,108 (M)</td>
</tr>
<tr>
<td><strong>Target</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Beeldbank-train</td>
<td>29,726</td>
<td>-</td>
</tr>
<tr>
<td>cross-domain-test</td>
<td>2,469 (M)</td>
<td>104 (B)</td>
</tr>
</tbody>
</table>

**Baseline work** We compare our attention-aware framework with ‘off-the-shelf’ CNNs for both single-domain VPR and cross-domain VPR tasks. The baseline work uses AlexNet pretrained on ImageNet cropped before conv5 as feature extractor. Features are then sub-sampled by either max pooling ($f_{max}$), average pooling ($f_{avg}$), vanilla VLAD pooling without attention($f_{VLAD}$) and VLAD with attention-aware $A_1$ method ($f_{A1-VLAD}$) [31].

**4.3. Evaluation metrics**

We follow the standard place recognition evaluation metric in [3] where the query image is considered as correctly matched if at least one of the retrieved top $N$ images is located within 25 meters away from the ground truth query location. The Recall@$N$ evaluates the percentage of correctly localized queries at different $N$ matching levels. For cross-domain place recognition, since the Beeldbank dataset contains labeled positive pairs, the Recall@$N$ will be directly calculated using these labels.

**4.4. Implementation Details**

The attention module starts with a ReLU activation, followed by a $1 \times 1$ convolutional layer and softplus activation to produce attention scores. In the VLAD layer, the number of cluster centers used is $K = 64$. Mapillary images were cropped with a random proportion to the original size between (0.3 1.0) for data augmentation before training.

We froze the layers before conv4 and fine-tuned the weights of all the other layers afterwards with the optimizer ADAM. We used the following hyperparameters: learning rate $lr = 1e-5$, batch size $= 2$ tuples (each tuple contains 24 images, including query, positive and negative pairs), epochs $= 25$. The hard negatives mining uses the same technique as NetVLAD[3]: it first caches all the training queries and gallery images for a time and then randomly selects 1000 negatives (image away from 25 meters). It keeps
Figure 3: Correctly retrieved top1 image from our framework trained with unsupervised domain adaptation, (top) queries are from the Beeldbank dataset, (bottom) retrieved images are from the Mapillary dataset. Our model can retrieve images not only depicting a similarly scene (a.), but also images from a different perspective (c.) and images captured further away from the query (b., d.). (b.) is correctly retrieved by matching the features of the building like the window and the unique shape of the door on the right side.

Figure 4: Visualization of attention score maps for source and target images. The top row shows the input images. The middle row is the heatmaps obtained by using [31], defined in Eq.4. The bottom row presents the results from our proposed method defined in Eq.6. It shows that our proposed attention module can generate accurate attention score maps with higher density on domain invariant objects for both source images and target images.
the top 10 hardest negatives from the cached gallery image features. The cache is updated every 1000 training queries.

We center cropped and reshaped all target Beeldbank images to 512 × 512 pixels in the cross-domain VPR experiment. The MK-MMD loss is calculated after conv5. The weight $\alpha$ in Eq.1 is 0.99. The margin $m$ in Eq.3 is set as 0.1.

4.5. Results

This section presents the results of the experiments with a detailed ablation study for the attention module (Section 4.5.1) and the domain adaptation module (Section 4.5.2) separately on both single-domain and cross-domain VPR tasks. Visual inspection of retrieval results and attention heatmaps are shown in Fig.3 and Fig.4.

4.5.1 Attention module

To evaluate the performance of our attention aggregation module on both single and cross-domain VPR tasks, we first trained the model on the source domain (Mapillary40k) and directly tested it on the source test set and the target test set without MK-MMD loss. Tab.2 shows the retrieval results where our attention aggregation method consistently outperforms the model without attention on both $S \rightarrow T$ and $S \rightarrow S$ tasks. A possible explanation could be that the VLAD descriptors are easily affected by the irrelevant objects. By not focusing on representative details that describe unique features of each building, it may retrieve an image that has a similar road or sky etc.

To inspect whether our attention module can produce reasonable attention scores for each descriptor, we visualize the attention maps of different attention-aware schemes in Fig.4. Our attention aggregation method generates heatmaps with higher densities on representative features and better robustness against irrelevant objects. Most attention is assigned to the architectures and less attention is assigned to non-representative regions such as road and sky as expected. Note that in Tab.2, the performance of $f_{AVLAD}$ is worse than $f_{VLAD}$ and $f_{our-VLAD}$ achieves the best results. We conclude that an insufficient attention map will deteriorate the performance.

4.5.2 Domain adaptation module

The additional domain adaptation loss (MK-MMD) is added to our model and all baseline works in this section. The MK-MMD loss is adopted to further minimize the domain discrepancy in this experiment. We applied it on the vanilla NetVLAD ($f_{VLAD-DA}$), $A_1$ attention model ($f_{AVLAD-DA}$) and our attention aggregation model ($f_{our-VLAD-DA}$). The performance of different models with and without MK-MMD loss are examined on both source and target test test. The results are visualized at different recall rates in Fig.5.

When trained with the MK-MMD loss for the $S \rightarrow T$ cross-domain VPR task, both $f_{VLAD-DA}$ and $f_{our-VLAD-DA}$ benefit from domain adaptation, while no significant improvement of $f_{AVLAD-DA}$ is observed. Detailed results are presented in Tab.3.

In addition, we also examined the performance of the model trained for the cross-domain VPR task on the source domain due to the reason that extra data from the target domain may also help with retrieval in source domain if the model is robust to the outliers in the Beeldbank dataset. $f_{VLAD-DA}$ does not show much power in the original source domain compared to $f_{VLAD}$. The retrieval accuracy of $f_{AVLAD-DA}$ in the source domain decreases after domain adaptation. Our proposed model gets better retrieval result even on the source domain as shown in Fig.5. This experiment proves that the domain specific features and outliers are reduced while more domain invariant features are captured by our proposed attention aggregation model which further facilitates the domain adaptation procedure.

Overall, we show that our attention aggregation model can achieve more accurate retrieval results on both single-domain $S \rightarrow S$ and cross-domain $S \rightarrow T$ tasks. DA denotes that the MK-MMD loss is added during training.

5. Discussion

Usually we assume that the training data and test data are sampled from an identical distribution which is violated in our cross-domain setting. We designed an attention-aware...
Table 2: The + denotes that the ‘off-the-shelf’ model is pretrained on ImageNet[15] for classification task. The others are trained on Mapillary40k for place recognition from scratch, and directly tested on the cross-domain dataset.

Table 3: Comparison of different models’ performance on the cross-domain $S \rightarrow T$ VPR task under two conditions: with or without domain adaptation using the MK-MMD loss. DA stands for domain adaptation using MK-MMD loss.

6. Conclusion

We proposed a specially-designed CNN for automatic annotation of historical images with their location. This is helpful specifically for museum curators and historians to retrieve the location information of a historical urban scene or architecture. This task is more challenging than single-domain (conventional) location retrieval due to the domain discrepancy caused by the large time lag between depicted scenes. A cross-domain dataset is collected accordingly with Mapillary40k used as source domain and Beeldbank, as target domain. To tackle this challenge, an attention aggregation module with a domain adaptation layer is designed, the performance of which is demonstrated by detailed experiments and ablation studies. Our attention aggregation model achieves state of the art results on both single and cross-domain VPR tasks by focusing more on domain invariant objects. It can be further combined with an extra domain adaptation module using the MK-MMD loss to achieve higher retrieval accuracy not only on the target domain but also on the source domain. Moreover, we believe our methods can achieve promising results on open-set domain adaptation tasks where unseen classes or outliers are not involved during training.

Besides the large domain shift, our Beeldbank target dataset contains various classes of images like people, indoor scenes, sketches and ground plans of buildings. These outliers are not contained in source dataset Mapillary40k rendering the task more difficult. Domain adaptation with more classes or outliers in the target domain compared to the source domain can be considered as open-set domain adaptation problem [7, 9, 33, 35]. Some other works refer to this as outlier detection problem [46, 29]. We speculate that the attention module can filter out the outliers by weighing them less with the heatmaps.
References

[1] https://beeldbank.amsterdam.nl/beeldbank. 4


[38] X. Shi, S. Khademi, and J. van Gemert. Deep visual city recognition visualization, 2019. 1


[40] G. Tolias, R. Sicre, and H. Jgou. Particular object retrieval with integral max-pooling of cnn activations, 2015. 2


The ability to find the place depicted in an image has a wide range of exciting applications such as: (i) automatic registration of uploaded consumer photos with maps (ii) transferring landmark information to the query image, or (iii) accurate visual localization for robotics. Visual place recognition (VPR) task often cast as image retrieval problem [1–7], the standard procedure mainly consists of two steps: First, extracting and storing feature vectors of gallery images. Then, extracting the feature vector of the query and computing the similarity between the query and all the gallery images. Finally, ranking all the gallery images based on the similarities and get the top \( K \) similar images (annotated with geo-tag) suggesting the location of the query. Conventional place recognition methods mostly based on local hand crafted features such as SIFT [8] and SURF [9], but recent works suggest that Convolutional Neural Networks (CNNs) trained on large scale dataset as feature extractor outperforms hand crafted features on various tasks. Motivated by these achievements, CNN based place recognition has emerged [1, 2, 10–12] and reached the state-of-the-art performance.

The VPR in single or similar domain has been widely studied, however, a more challenging problem, cross domain place recognition, has been rarely studied. The cross domain aims to match images of the same architectures taken in a wide time range, e.g. historical images and current street view images.

### 2.1. Motivation

A cross-domain VPR task is proposed in this work. VPR is commonly treated as an image retrieval task, where a query image from an unknown location is matched with relevant instances from geo-tagged gallery database. Different from conventional VPR settings where the query images and gallery images come from the same domain, we propose a more common but challenging setup where the query images are collected under a new unseen condition. The two domains involved in this work are contemporary street view images of Amsterdam (source domain) and histori-
2.2. Research objectives

The goal of this paper is to find the place depicted in a historical image (target domain) in a current street view image gallery (source domain) based on visual similarity. The main challenges include, general image matching problem, changes of viewpoint and illumination, another challenge is the domain shift between the source and target. To overcome these challenges, our contributions are:

1. A new attention aggregation scheme is proposed to CNN to combine both the local and global image descriptors hence reduce the effect of ‘confusing objects’.

2. We combined the MK-MMD domain adaptation loss with the triplet ranking loss to learn domain-invariant features for cross-domain VPR task.

2.3. Outline

The rest of the thesis report is organized as follows: Chapter 3 gives an background introduction to general convolutional neural networks. Chapter 4 introduces the concept of domain adaptation. Chapter 5 explains the feature aggregation method (VLAD) for image retrieval. Chapter 6 introduces the attention mechanism used in visual tasks.

References

References


3.1. Introduction to Deep Learning

deep learning[1] also referred as Neural Network (NNs) is a subset of machine learning, has achieved great achievements in image, video and audio related fields. Below we will introduce some basic knowledge of NNs.

**Neuron** The neuron is the basic unit of neural network. To describe the neurons statistically, let $\mathbf{x} = \{x_1, ..., x_n\}^T$ be the input, $y$ be the output, Fig. 3.1 shows a mathematical model of the neuron. The final output of the neuron is the sum of weighted input $\mathbf{wx}$ with bias $b$

$$u = \sum_{i=1}^{n} w_i x_i + b,$$

and then pass it through a nonlinear activation function $f$, so the final output is

$$y = f(u),$$

where the weights $\mathbf{w}$ and bias $b$ are trainable parameters.

**Neural Networks** Neural Networks consist of a collection of neurons that are connected in an acyclic graph. Fig. 3.2 illustrates a simple neural networks consisting input layer, hidden layer and output layer(bias dose not show in this figure). Neurons in two neighbor layers are fully pairwise connected, but neurons within a
single layer share no connections. The output $\hat{y}$ is calculated as:

$$\hat{a} = W^{(1)} \hat{x} + b^{(1)}$$  \hspace{1cm} (3.3)$$
$$\hat{y} = W^{(2)} \hat{a} + b^{(2)}$$  \hspace{1cm} (3.4)

which can be easily implemented through matrix operations.

**Activation function** activation function is an nonlinear mapping from the weighted input to output. And it is usually but not necessary monotonic and differentiable, we need it to be this way so as to perform backpropagation optimization strategy, Fig. 3.3 shows some commonly used activation functions. The activation function introduces nonlinear properties to neural networks, and enable the network to learn and model other complicated, high dimensional types of data such as image, video, audio etc..

**Training** The weights and bias in neural networks are the trainable parameters, are usually randomly initialized and updated during training by minimizing the loss function. The loss is usually defined as the sum of the squared difference between the target values and the network output. This is a classic optimization problem which could be solved by gradient decent method, hence we can optimize all parameters based on the gradient. But when the networks gets ‘deeper’, due to the complexity of its structure and huge amount of parameters, it is impractical to calculate all gradients. Backpropagation[2] makes it possible to solve this problem. In this way we can propagate the total loss back into the neural network to know how much of the loss every neuron is responsible for, and subsequently update the weights in such a way that minimizes the loss.
### 3.2. Convolutional Neural Network

Convolutional Neural Networks (ConvNets) is built upon neural networks, it assumes the inputs are images and builds an efficient architecture to extract certain visual features. The previous ‘hidden layer’ is replaced with three main types of layers, including ‘convolutional layer’, ‘pooling layer’ and ‘fully connected layer’ (the same layer type as ‘hidden layer’). Networks built by these types of layers are more efficient and the amount of parameters can be reduced.

**Convolutional layer** Given a high-dimensional input, such as image, it is impractical to connect all the neurons in the way as in Fig. 3.2. The convolutional (CONV) layers consist of a series of filters with size $W \times H \times D$, these filters connect each neuron to only a local region of the input, the $D$ is always the same with the $D$ of input volume, Fig. 3.4 is an example of the CONV layer operation. When sliding
the input window, the regional input changes, while the parameters of the filter is the same, this is the parameter sharing scheme used in CONV layers to reduce the number of parameters. The basic idea behind it is that a useful feature extracted at a certain spatial position may also be useful to extract at another position.

**Pooling layer** is used to reduce the number of parameters also to prevent overfitting. It downsamples the input spatially at each depth slice with max operation, and the depth of output remains unchanged. Fig. 3.5 illustrates the process of max pooling operation.

**Fully-connected layer** connects each neuron to all neurons in the previous layer, as seen in Fig. 3.2. In AlexNet[4], CONV layers and pooling layers serve as feature extractor, and fully-connected layers is followed to combined these features together. Finally, an activation function will classify the output as different classes.

**AlexNet** is used as base network in our work. AlexNet won the championship in the 2012 ImageNet LSVRC competition, also popularized CNN in computer vision. Fig. 3.6 illustrates the architecture of AlexNet, it consist of 5 convolutional
3.3. Triplet Network

CNN architecture has gained great achievements in classification tasks, however, the number of classes may be variant or very large, for example, we want to compare two unknown faces and say whether they are from the same person or not. To solve this, triplet network is a good method by learning deep face embeddings through distance comparisons, has been widely used for face recognition.\[5\]. Triplet network is inspired from Siamese Network\[6\], which contains three feedforward networks with shared parameters, and three inputs: an anchor $a$, a positive from the same class as anchor $p$, a negative from different class $n$, like in Fig 3.7. To bring the positive pair $(a, p)$ together, the triplet loss is introduced, the learning objects are (i) for samples with the same labels, their distance in the embedding space should be close and form separable clusters, and (ii) otherwise should be far away. Formally, for the distance metric $d$ in the embedding space, the loss $\mathcal{L}$ for an input tuple $(a, p, n)$ is:

$$\mathcal{L} = \max(d(a, p) - d(a, n) + \text{margin}, 0).$$ \hspace{1cm} (3.5)

We minimize the loss by forcing the distance of negative pair $d(a, n)$ to be greater by a margin than the distance of positive pair $d(a, p)$. The loss is zero when
the distance of positive pair is less than the distance of negative pair by a certain margin.

3.3.1. Triplet mining

Based on the definition of triplets, here are three categories:

1. easy triplet: \( d(a, p) + m < d(a, n) \), the loss is zero.

2. hard triplet: \( d(a, p) > d(a, n) \)

3. semi-hard triplet: \( d(a, p) + m > d(a, n) > d(a, p) \)

The choice of triplet has great impact on the result. Paper[7] shows that selecting the hardest positive(biggest \( d(a, p) \)), and hardest negative(smallest \( d(a, n) \)), yields the best performance.

References


This chapter aims to introduce basic definition of domain adaptation, and deep domain adaptation method that has been investigated in our research.

4.1. Introduction to Domain Adaptation

Machine learning aims to learn knowledge from data relying on the features and inference. The successful applications of many machine learning algorithms are built on the premise that the training and future data are in the same feature space and follow the same distribution. However, this assumption may not hold for real world problem. For example, when have an interested domain with insufficient training data and another domain with sufficient training data, but two domains are in different feature spaces or have different data distribution. In such cases, if knowledge learned from the latter domain could be flexibly adapt to the interested domain, it would greatly improve the performance in the interested domain without expensive data labeling efforts.

In recent years, domain adaptation as a subproblem of transfer learning, has emerged as a new learning framework to address the domain shift problem, the key of domain adaptation is learning the shared feature in both domains. Specifically, it applies the distribution adaptation on the source domain to the target domain by making use of the similarity of the data.

Formally, a domain $D$ can be described by a feature space $\mathcal{X}$ and a marginal distribution $P(X)$, where $X = \{x_i, \ldots, x_n\} \in \mathcal{X}$ is a set of samples from the same feature space $\mathcal{X}$. Given a domain $\mathcal{D} = \{\mathcal{X}, P(X)\}$, a task $\mathcal{T}$ is consist of label $Y = \{y_1, \ldots, y_n \in \mathcal{Y}\}$ from space $\mathcal{Y}$ and an objective predictive function $f(\cdot)$, which from statistic prospective, can be viewed as a conditional probability distribution $P(Y|X)$. So we can learn the predictive function $f(\cdot)$ by making use of labeled data $\{x_i, y_i\}$, where $x_i \in \mathcal{X}, y_i \in \mathcal{Y}$.

Given a labeled source domain $D_s = \{x_i, y_i\}_{i=1}^n$, and an unlabeled target domain $D_t = \{x_j\}_{j=n+1}^{n+m}$. Traditional machine learning assumes $D_s = D_t$ and $\mathcal{T}_s = \mathcal{T}_t$,
Domain Adaptation paper[1] summarized that the domain shift can be caused by domain divergence $D_s \neq D_t$, including distribution shift $P(X)$ or feature space difference $X$ and task difference $T_s \neq T_t$, like conditional distribution $P(Y|X)$ shift or label space $Y$ difference. In our case, we consider the first situation, domain divergence, so our goal is to use labeled source domain data $D_s = \{x_i, y_i\}_{i=1}^n$ and unlabeled target sample $D_t = \{x_i, y_i\}_{i=1}^n$ to learn an objective predictive function $f : x_t \rightarrow y_t$ to predict target domain label $y_t \in Y_t$.

4.2. Deep Domain Adaptation

Compared with traditional domain adaptation methods, deep domain adaptation based methods boost the performance on many tasks[2, 3]. It directly learns from origin data, automatically extracts more representative features in an end-to-end manner.

Many deep approaches [3, 4] introduced the adaptation layer to minimize the domain disparity. The basic idea of deep adaptation network is to (i) find out which layer is adaptable, then (ii) choose a metric to perform adaptation within these layers. Compared to traditional domain adaptation, the deep one defines loss as:

$$l = l_c(D_s, Y_s) + \lambda l_A(D_s, D_t),$$

where $l_c(D_s, Y_s)$ is the task loss (e.g., classification loss) on labeled data (usually the source domain), $l_A(D_s, D_t)$ is the adaptation loss, and $\lambda > 0$ determines how strongly we would like to confuse the domains. With this loss, we can simultaneously minimize the classification loss and maximize the domain confusion, so that the domain invariant and discriminative representations can be learned, Figure 4.1 gives an explicit image interpretation.

One classic deep domain adaptation method[4] is DDC(Deep Domain Confusion). It uses the AlexNet[5] pretrained on ImageNet, Fig. 4.2 shows how the DDC
works In this architecture, there are a source and a target network which share the weights to ensure domain invariant features. Layers before \( fc8 \) all stay intact, serve as the feature extractor, and followed with an adaptation layer which evaluates the domain disparity with MMD (Maximum mean discrepancy) metric. MMD is defined by the distances between mean embeddings of the features. It maps the features to the reproducing kernel Hilbert space \( \mathcal{H} \), denoted as \( \mathcal{H} \).

\[
\text{MMD}^2(X, Y) = \left\| \sum_{i=1}^{n_1} \phi(x_i) - \sum_{j=1}^{n_2} \phi(y_j) \right\|_\mathcal{H}^2,
\]

where \( \phi \) maps the feature to \( \mathcal{H} \). so the MMD is the distance between the means of the two distributions in space \( \mathcal{H} \).

Different from MMD based on single kernel transformation, multiple-kernel MMD\(^7\) (MK-MMD) assumes the optimum kernel can be described by the linear combination of multiple kernels and has been widely used in latter research\(^3\), which introduces multiple adaptation layers with MK-MMD metric.

References


5

VLAD: vector of locally aggregated descriptors

In this section, some image representations for image retrieval will be introduced. The way to embed the VLAD in the CNN architecture will be shown as well.

5.1. Image vector representation

The task large scale image search/retrieval has been traditional addressed with bag-of-visual-words (BOV)\textsuperscript{[1]} which assigns the image descriptor (e.g. SIFT descriptor) to its closest visual word in a ‘visual vocabulary’: a codebook $C = \{c_1, ..., c_k\}$ of $k$ visual word which learned with k-means, and this will yield a fixed size high-dimensional histogram representation.

Given the descriptors of an image $\{x_i\}_{i=1}^N$ of $D$ dimension, the idea of VLAD\textsuperscript{[2]} is to aggregate, for each cluster center $c_k (k = 1, ..., K)$, the residuals $x - c_i$ of the descriptors. The size of the final VLAD representation $V$ will be $D \times K$, so one element of $V$ is computed as:

$$V(j,k) = \sum_{i=1}^{N} a_k(x_i)(x_i(j) - c_k(j)),$$ (5.1)

where $j, k$ respectively denote the $j^{th}$ ($j = 1, ..., D$) component of the descriptor and the index of the clusters. $x_i(j)$ and $c_k(j)$ are the $j^{th}$ component of the $i^{th}$ descriptor and $k^{th}$ cluster. The $a_k(x_i)$ denotes the membership of $x_i$ to the $k^{th}$ cluster with hard assignment, that is, for every $x_i$ whose closest center is $c_k$, the corresponding $a_k(j)$ is 1, otherwise is 0.

Inspired from VLAD, paper \textsuperscript{[3]} proposed a new architecture named NetVLAD by plugging the VLAD layer into CNN architecture. The non differentiable part of VLAD comes from the hard assignment $a_k(x_i)$, to make this operation differentiable,
Paper[3] replaces the assignment of the descriptor $x_i$ to $k$ cluster center $c_k$ with a soft one:

$$\bar{a}_k(x_i) = \frac{e^{-a\|x_i-c_k\|^2}}{\sum_k e^{-a\|x_i-c_k\|^2}}, \tag{5.2}$$

which assigns the descriptor $x_i$ to cluster $c_k$ with a weight according to the distance between them.

By expanding the squares in (5.2), it will result in the following form:

$$\bar{a}_k(x_i) = \frac{e^{w_k^T x_i + b_k}}{\sum_k e^{w_k^T x_i + b_k}}, \tag{5.3}$$

where vector $w_k = 2\alpha c_k$, scalar $b_k = -\alpha \|c_k\|^2$, so the final VLAD representation which is able to train via backpropagation will be:

$$V(j,k) = \sum_{i=1}^{N} \frac{e^{w_k^T x_i + b_k}}{\sum_{k'} w_{k'}^T x_i + b_{k'}} (x_i(j) - c_k(j)), \tag{5.4}$$

where the $\{w_k\}, \{c_k\}, \{b_k\}$ are trainable parameters, and $K$ as the hyper-parameter, so compared to the origin VLAD only trains $\{c_k\}$, this enables more flexibility and adaptability when facing more complex examples.

### 5.2. Deep architecture with VLAD

This section describes how the CNN architecture is combined with the VLAD layer. Image retrieval tasks usually follow this pipeline: (i) extract local image descriptors, then (ii) perform spatial pooling over the feature map. To learn the representation end-to-end, paper[3] introduced a CNN architecture called NetVLAD, it follows the same image retrieval pipeline: (i) crop the CNN at the last convolutional layer, and treat it as the local descriptor extractor, so the $H \times W \times D$ output can be viewed as $D$ dimensional descriptors at $H \times W (H \times W = N)$ spatial locations. (ii) use the differential VLAD layer mentioned in Section 5.1 to pool the descriptors into a fixed size $D \times K$ image representation. The parameters from VLAD layer are trainable via backpropagation, this new architecture is called ‘NetVLAD’. Figure 5.1 shows the whole pipeline and its inner operation of NetVLAD.
Figure 5.1: CNN architecture with VLAD layer[3]. The VLAD layer can be naturally implemented through normal CNN operations: convolution, soft-max, L2-normalization. And the aggregation layer (‘VLAD core’) is also very easy to implement by following equation (5.4).

References


This chapter will introduce the general attention module for visual recognition, and how the attention module combines with NetVLAD, also will show the geometric interpretation of attention for VLAD.

6.1. Introduction to Attention Module

The attention module has been widely studied in visual recognition domain. CNN has been proven to be able to localize the image areas related to the label [1, 2]. The general idea of the attention module is inspired from how human eye works: Our retina is capable of receiving a wide view of the scene, but our visual processing mechanism is prone to "glance" the overview and only focus on a particular region of the view, while the rest is ‘blurred out’. The visual attention model is trying to leverage on this idea, by paying more attention on a selected informative region, and less attention elsewhere. Fig. 6.1 shows some examples of visual attention mechanism.

Figure 6.1: [3] The soft attention serves as a mask to filter out the uninterested region(sky) and highlights the parts that are important for classification.
6.2. Attention-aware VLAD Layer

From chapter 5 we show that the VLAD representation is highly effective for complex visual recognition tasks by aggregating local features from the entire image. Paper [4] addresses that while it is effective, it ignores the fact that some local descriptors may not be equally important, e.g., for descriptors locate at the sky region are less important than those locate at building region. Based on this, [4] introduces an attentional structured representation learning framework that incorporates an image-specific attention mechanism within the aggregation process to ignore non-relevant background information.

Formally, let $W \times H \times D$ be the final convolution layer output. The attention module usually consists of one $1 \times 1$ convolutional layer which will produce one heatmap with size $W \times H$. To incorporate the attention map (heatmap) into the VLAD layer, we can use the heatmap $w(x_i)$ to reweight the descriptors, since they share the same spatial size. Specifically, we can rewrite Eq 5.1 into

$$V(j, k) = \sum_{i=1}^{N} w(x_i) a_k(x_i)(x_i(j) - c_k(j)). \quad (6.1)$$

6.3. Geometric Interpretation of Attention

Given the descriptors extracted from two images from the same class but have different background, and the descriptors are assigned to the same cluster center, illustrated in Fig. 6.2 the original aggregation scheme would yield resid-

![Figure 6.2: Geometric interpretation of attention](image)

The Voronoi cell shows the clustering used in VLAD descriptor construction, the green dot represents the cluster center, and the dots grouped into two ellipses represent local descriptors extracted from two different images, the descriptors with high attention are show in blue, and those with low attention are show in red and orange, while red arrows correspond to the sum of their residuals, the blue arrows correspond to the weighted sum of residuals.
attention weighted aggregation vectors (blue arrows) are with high cosine similarity. Attention allows us to ignore confusing objects and improve the discriminative power of VLAD.

References


