Arabian Journal for Science and Engineering
https://doi.org/10.1007/s13369-021-06148-8

RESEARCH ARTICLE-COMPUTER ENGINEERING AND COMPUTER SCIENCE

COLCONF: Collaborative ConvNet Features-based Robust Visual Place Recognition for Varying Environments

A. H. Abdul Hafez[1] · Ammar Tello[1] · Saed Alqaraleh[1]

Received: 3 April 2021 / Accepted: 22 August 2021

Abstract

Several deep learning features were recently proposed for visual place recognition (VPR) purpose. Some of them use the information laid in the image sequences, while others utilize the regions of interest (ROIs) that reside in the feature maps produced by the CNN models. It was shown in the literature that features produced from a single layer cannot meet multiple visual challenges. In this work, we present a new collaborative VPR approach, taking the advantage of ROIs feature maps gathered and combined from two different layers in order to improve the recognition performance. An extensive analysis is made on extracting ROIs and the way the performance can differ from one layer to another. *Our approach* was evaluated over several benchmark datasets including those with viewpoint and appearance challenges. Results have confirmed the robustness of the proposed method compared to the state-of-the-art methods. The area under curve (AUC) and the mean average precision (mAP) measures achieve an average of 91% in comparison with 86% for *Max Flow* and 72% for *CAMAL*.

Keywords Visual place recognition · Deep learning · Regions of interest

1 Introduction

Visual place recognition (VPR) systems aim to make a decision on whether the currently observed place is visited previously or not. This is important for robots to be able to localize themselves with respect to the environment, which in turn is important to successfully complete the navigation task. This problem in fact is challenging when the robot is required to work in an everyday environment and for a long period of time.

The VPR problem is a challenging one due to different visual effects. The first one is the changes in appearance over days or seasons [1], such changes may be periodic as well. The same environment could be visited in any day/night time, or summer/winter time. The second effect is the variations in the observation viewpoint [2]. The robot may look to a different direction when it visits the same place again. The third is the observed dynamic elements in the scene like pedestrians, vehicles, etc. [3].

The VPR research community has been approached the problem through two major techniques. They are image-to-sequence matching-based VPR [4] and sequence-to-sequence matching-based VPR [5]. Even though the former technique compares a test image to the stored sequence, most of them use the sequence information in the localization process. They assume particularly that the stored images are recorded in order based on the robot's previous motion. On the other hand, the sequence-to-sequence matching techniques have the advantage of matching the full sequence and get optimal matches, but they face the problem of finding a solution when the robot deviates from the visited route.

Earlier works on VPR used handcrafted features to represent the images during the matching process. FabMap work presented in [2] used the SURF features [6], while *SeqSLAM* [1] has used the downsized and contrast normalized images as features for matching purpose. The SIFT features [7] are used in [8] to represent the images for visual localization in a highly crowded environment. HOG features are used in the Max Flow network [5], and a binary descriptor called LDB was used in the ABLE algorithm [9]. The deep convolutional neural networks (CNNs)-based features have started attracting the attention of the VPR research community due to its promising successful applications in VPR and other computer vision areas [4,10,11].

✉ A. H. Abdul Hafez
abdul.hafez@hku.edu.tr

[1] Department of Computer Engineering, Hasan Kalyoncu University, Şahinbey/Gaziantep, Turkey

Published online: 02 October 2021

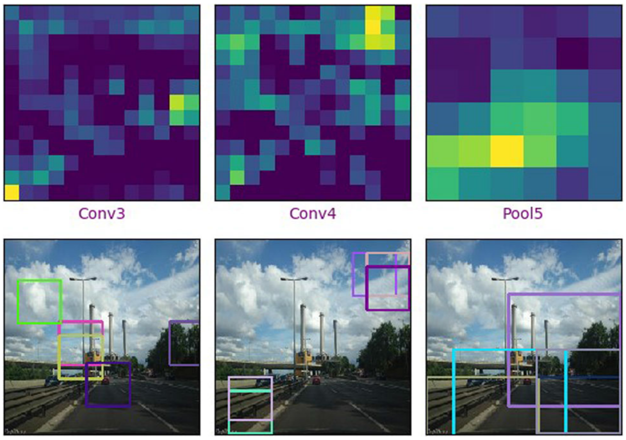

Fig. 1 It can be observed that Conv3 and Conv4 have shown higher activation responses to primitive features (non-semantic) like corners and edges. Pool5 has more meaningful ROIs, detecting the semantic and highly responding to objects with considerable variations in pose and viewpoint. It is clear that no single layer can produce enough features for robust VPR. Our proposal presents collaborative ConvNet features that combine ROIs features from two different layers, Conv3 and Pool5, to meet the different visual challenges

Research community works on using CNN features for VPR have been following three major schemes in order to represent the image using data abstracted from the CNN's layers. These schemas depend on the portions of the image used for the representation. They are (i) global-in and global-out, (ii) local-in and local-out and (iii) global-in and local-out. In the global-in and global-out, the entire image is provided as an input to the CNN. The features are selected as the output of certain middle and end layers [12,13]. Such global features have shown performance degradation with a highly changing point of view and appearance resulted due to occlusions. In the local-in and local-out, pre-identified regions of interest (ROI) are extracted and supplied as an input to the CNN. The output of the corresponding active fields of neurons is used to represent the image [14] and [15]. These representations are usually able to deal with occlusions and viewpoints but on the account of computational cost. In the global-in and local-out, like the first scheme, the entire image is supplied to the input of the CNN, but discriminative regions from the middle and last layers are used to represent the image [3,4,16]. The results reported by works that follow this scheme have shown larger robustness to occlusions and changes in the point of view. We follow the third scheme in this paper, and propose to extract regions from two different layers, then compactly represent them using a visual codebook. This was also adopted in [4,16,17] but using very close layers without addressing multiple challenges as we propose here.

It was shown in the computer vision literature that CNN layers with different depths respond differently to the visual input. As reported by Zeiler and Fergus [18], the activations of the layers exhibit a semantic hierarchy of the image features. Experiments in the VPR literature, particularly those presented in [11,15] have shown that features from the middle layers in the CNN architectures are better to deal with the appearance changes including illumination and dynamic objects. Besides, later layers in the CNN architecture have shown more robustness to the changes in the viewpoint than earlier layers. See Fig. 1 for an example of ROIs produced by the Conv3, Conv4 and Pool5 layers from AlexNet365. In the first row, the ROIs are represented in different colors. In the second row, the projection of these activations on the input image is depicted, where the boxes are referred to the receptive fields of ROI extracted from the feature maps. According to this, we claim that combining features from two layers, one from the middle and another later layer from the CNN, will provide collaborative features scheme with robust representation against the multi-challenges datasets which contain more than one VPR challenge, for example (viewpoint and illumination, dynamic objects and viewpoint).

We present in this paper a solution to both the viewpoint and appearance changes. Since information from a single layer cannot meet both of these challenges as mentioned above, this work proposes a collaborative features that combine the information from one middle layer with information from one more deeper layer. The middle layer is able to recognize changes in the appearance while the deeper layer is able to recognize the changes in the point of view. Regions of interest features are extracted from both the middle layer "Conv3" and the deeper layer "Pool5" from the light Alex365 architecture. PCA is used to reduce the dimensionality of ROIs vector extracted from the "Conv3" layer and concatenate it with the one from the "Pool5" layer. The resulted ROI vector is encoded using Fisher vector (FV) [19] to obtain a compact representation in order to improve the efficiency and speed of the matching process.

Our proposal adopted FV encoding methods inspired by the several works, following the global-in/local-out scheme, that have reported enhanced performance using an encoding method to represent the regions in more compact vectors. This usually improves the efficiency and the speed of the search, and leads to more stable representation since it learns the internal clusters of the data [20,21]. The visual bag of words (vBoW) codebook [22] is used for encoding the selected regions in [4]. The work presented in [3] has introduced the locally aggregated descriptors (VLAD) [23] instead of the (vBoW) claiming that it has less complexity and uses smaller visual codebook. In this work, the resulted ROI vector is encoded using FV to obtain a compact representation in order to improve the efficiency and speed of the matching process.

The contributions of this paper are stated as follows.

1. Novel collaborative ROIs features that are extracted from two different layers. Each layer is able to identify different

visual changes in the test image. Combining the features from these two different layers improves the overall performance when the system concurrently meets different visual challenges.

2. FV representation is used for the first time to compactly represent the extracted features. Due to its less quantization error and longer descriptors, our proposal has shown superior performance through different datasets. Although FV produces longer descriptors comparing to other visual code-books like BoW and VLAD, the time efficiency of the system is not degraded since a small number of clusters is enough.

The rest of the paper is organized as follows. Section 2 reviews the most recent and relevant works in the VPR literature. It focuses on the research gaps left behind them and how our proposal bridges these gaps. Section 3 presents our proposed method, it shows a detailed description of the different stages of our proposal. The Experimental evaluation is presented in Sect. 4. A discussion and conclusion remarks are presented in Sect. 5.

2 Literature Review

We review in this section a few works from the general image classification problem to introduce the idea of feature extraction and classification using deep CNN architectures. After that, we review the most state-of-the-art works that discuss the specific VPR problem. Currently, CNN based has the main portion of researchers in image content systems such as image classification, content-based retrieval and VPR systems. This is due to the fact that CNN models are achieving state-of-the-art performance and almost human-level performance. This achievement was obtained through intensive investigations starting from (i) introducing a CNN and train it for the desired job, then, moving to (ii) adopt a more efficient classifier to the used CNN, here multiple approaches were investigated such as using the output of one layer, or using multiple layers, or using an encoder to empower the obtained features and get more discriminative features.

Later, more advanced approaches such as extracting some specific region from the image using a CNN which is called “silent regions” or “landmark” show superior performance in almost all mentioned fields. Examples of these studies are: In [24], the features extracted from a proposed CNN model that is called “DL-CNN” were passed through a principal component analysis (PCA) to represent the image’s salient features. Then, Euclidean distance was used for calculating the similarity between the query image and reference images. Overall, results showed that this system outperforms CNN when used alone.

Another interesting approach was proposed in [25], where multiple CNN were integrated to build an ensemble system. Here, two architecture were investigated, i.e., (1) using multiple copies of the same model but train each one using a different set of images, or (2) using different models that are trained on the same dataset. Hence, each image is represented by combining the extracted class probability vectors from the proposed ensemble. Overall, both architectures significantly outperformed the performance of individual CNN models. The researchers of Guo et al. [26], showed that combining a feature fusion approach with CNN can improve the classification performance, where multiple spatial–spectral feature fusion was tested and as a result, all were able to outperform the performance of CNN. On the other hand, some impressive improvement was presented in [27] where the input image is represented by combining the features extracted from three descriptors, i.e., improved version of AlexNet, Histogram of Oriented Gradients (HOG) and Local Binary Pattern (LBP). Finally, the Principal Component Analysis (PCA) algorithm was used for reducing the dimension of the obtained features.

In recent years several researches and solutions were introduced in order to solve the VPR problem, overcome its challenging issues and improving its performance. Using the handcrafted features for building efficient VPR systems was intensively investigated [1,2,5,28], where approaches such as the FAB-MAP [2] and *SeqSLAM* [1] can be considered as the state-of-the-art VPR handcrafted-based systems. For instance, FAB-MAP extracts the image features using the SURF descriptor. Then, the BoW is used to encode the extracted features, while *SeqSLAM* searches for all possible matches in the visual map in order to solve the localization problem. In addition, unlike other approaches that use the image similarity, *SeqSLAM* uses image differences for matching.

Another approach that aims to deal with changes in the environment was proposed in [5]. This approach creates and constructs an association graph that relates images in different conditions that is used to compute the network flows in order to generate multiple vehicle route hypotheses. In addition, the image HOG descriptors have been adapted into the approach of Naseer et al. [5]. In [28], an approach that works on learning some useful features using multiple experiences in crowded urban environments was developed.

However, with the advances, particularly in terms of performance, of the CNN models in image classification [29,30]. The CNN-based VPR approaches [3,4,10,13] were able to outperform the existing handcrafted-based VPR ones. In general, such a system works on extracting the output of a specific layer, and a classifier such as the Softmax, K-NN or SVM is used to find the most similar reference image. Such approaches that extract features from a single CNN layer are not able to deal with multiple challenges like changes in appearance and viewpoint simultaneously. As shown in the previous section and experimentally proved in the computer vision and the place recognition literatures [11,15,18,31,32], a single CNN layer responds to a certain visual attribute.

To overcome the aforementioned problems, many methods try to integrate features from multiple layers. The approach proposed in [16] extracts the features from the Max-pooling layer for the first five convolution blocks of the VGG-16. These extracted feature vectors are first padded to the same size. Then, the input image is represented by one vector whose elements are produced by summing up the values of the corresponding indexes in the extracted five vectors. In this approach, the similarity between the reference and the test images is obtained using three fully connected layers that were trained using the Cosine similarity. A multi-process fusion approach was proposed in [17]. It represents each image by four vectors. The first two are produced using the sum of absolute differences (SAD) and HOG descriptors. The last two are extracted from the Conv-5 of HybridNet, one using pyramid spatial pooling, while the second using spatial coordinates of maximum activations. The test image is finally classified using the most similar one from the reference images.

The second set, which is aligned with the global-in and local-out, works on detecting and using the image's most important representative information (landmarks/salient regions). One of the state-of-the-art VPR model is known as *NetVLAD* [10], which integrates the VLAD as a CNN layer. A weakly supervised ranking loss was used in the training procedure for obtaining the values of the architecture's parameters in an end-to-end manner. Overall, the NetVLAD achieved quite good performance as compared with non-learned image representations and off-the-shelf CNN descriptors.

Only Look Once is an approach that aims to overcome the viewpoint challenge was presented in [4]. This approach constructs some landmarks from the output of the last convolutional layer of a pre-trained VGG16, maps the detected landmarks activations of the used convolutional layer using the previous convolutional layer, and uses the BOW technique to encode the image's landmarks. Then, the cosine similarity is used to find the mutual matching of the test image's landmarks against the reference images. This method, i.e., *Only Look Once*, uses very deep CNN models to improve the performance that increases the computational cost of such systems, and as shown in our experimental investigation, its performance degraded when the used dataset has multiple challenging problems.

A similar approach was developed recently in [3] and called *Holistic* visual place recognition. It works on obtaining the feature maps and identifies the salient regions (ROI) from the third convolution layer of the AlexNet model. Every two or more neighbors in that layer are considered as ROI if they have approximately similar responses. Then, it uses the VLAD to encode the extracted regions. The *Holistic* approach uses the cosine distance to compare the similarity of the test image with all the reference images and consequently finding the best-matched reference image.

More recent methods that aim to eliminate the need for training, or only speed up the training of the used CNN model were recently proposed. For instance, an approach named *CoHOG* was presented in [33]. It uses the image entropy and "HOG" descriptor to extract and represent the regions of interest in the image. The image entropy is used to identify the most promising regions of interest, where each of the detected regions is represented by the HOG descriptor. Then, the best-matched region in the reference images is found for each of the query's regions using the Softmax. Finally, the most similar reference image is the one that has the highest mean of its regions similarity with the query's regions. Another approach proposed in [34] uses the *AlexNet365* model. The approach is called *CAMAL*, and it extracts the salient regions that may be changed due to the weather conditions and/or viewpoints while ignoring the salient regions of what they called "confusing instances" such as the sky and moving objects. Then, the VLAD descriptor is obtained for the extracted regions that will be fed into the fully layers to find the most similar reference image.

Our proposal is built on the idea that integrating more than one CNN layer can solve the problem of multiple challenges. Recalling the approaches reviewed above, most of them try using a single CNN layer to improve the performance of the VPR system [11,15,18,31,32]. More recent works such as [16,17] use two neighbor layers in order to facilitate the ROI feature extraction process. Other works use features from very deep CNN layers like [3–5,16,17]. Recently, a model called *CAMAL* [34] that uses very few layers and is able to perform comparatively to the mentioned very deep models against certain visual challenges was developed. However, it has been reported that all mentioned approaches are not able to deal with multiple challenges like changes in appearance and viewpoint simultaneously.

The proposed approach ensembles the advantages of using a light CNN model, uses a few layers, by reducing the training and testing complexity, with the ability to merge two different layers, one from the middle and another from the last layers, to meet multiple visual challenges, viewpoint and appearance changes in particular. The model AlexNet365 is adopted, the layers ("Conv3") and ("Pool5") are selected to provide the region of interest features ROIs. As these layers are of different sizes, PCA is used to unify the dimensions of the ROI features. FV encoding is used to provide a compact representation of the features and improve computation efficiency. The proposed approach has the ability to efficiently handle multiple challenging problems when occurring concurrently. A DTW matching stage is added at the end to improve the final decision precision. The overall performance of the pro-

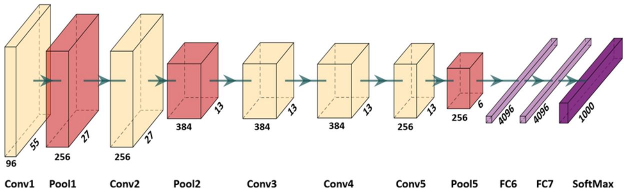

Fig. 2 The architecture of the AlexNet365 CNN model

posal is experimentally found to be superior to the ones of the state-of-the-art methods as shown in Section 4.

3 The Proposed VPR Technique

The proposed approach uses the "AlexNet365" CNN model as it contains very few convolutional layers. It is lighter (less number of layers) compared to other very deep models such as VGG-16 and ResNet while having the ability to achieve better performance and outperform the mentioned models when used for VPR. The "AlexNet365" model consists of five convolutional layers, see Fig. 2. The first, second and fifth layers are followed by max pool layers. Each of the convolutional and pooling layers is followed by activation layers, and three fully connected layers at the end of this structure. In addition, the first convolutional layer consists of 96 feature maps, where each of the first and third pool, in addition to the second and fifth convolutional layers, consisted of 256 feature maps. Whereas each of the rest layers consists of 384 feature maps. Furthermore, each of the first and second fully connected layers has 4096 nodes, and the last layer (SoftMax) has 1000 nodes.

The functionality of our proposed approach, as shown in Fig. 3, starts flowing when the input image is applied to the network. Then, the layer computation is propagated forward, where the activities of the layers "Conv3" and "Pool5" are calculated. After that, the ROIs are extracted from the selected two layers. The activations of these two layers are stacked in a single vector descriptor for each image. These local descriptor vectors are encoded and represented using FV-based visual code-book. This representation will be used later for making the recognition decision. The major stages of the proposed method are presented in the following subsection.

3.1 ROI Identification and Collaborative Layers

The regions of interest (ROIs) are the discriminative regions extracted from the output of a selected layer(s). Extracting ROIs is categorized as Global-in and Local-out, where the entire image is supplied as an input to the CNN, and several regions (ROIs) are detected and extracted for each feature maps (sub-layer). ROIs are extracted based on the similarity of the activation's value a_i^p between the neighbors, i.e., each two or more neighbors are considered as ROI if they have approximately similar responses. Assume a CNN layer contains Z feature maps, then every ROI is represented using a Z-vector R. Consequently, the ith region extracted from the Lth layer is represented using the descriptor vector $R_{L,i}$ where $i \in \{1, \ldots, N_L\}$. Here, N_L is the number of ROI extracted from the Lth layer. The set of ROI descriptors are compactly represented using the array $\mathcal{R}$ whose rows are $R_{L,i}^T$.

In this work, we have adapted the "AlexNet365" architecture, depicted in Fig. 2, and we are using two layers to extract the ROIs. The first one is the "Conv3" which is the third convolutional layer. Its size is equal to 13 × 13 × 384, let here the number of maps is $Z_{C3} = 384$. The second layer is "Pool5," which is the last layer before the fully connected layers. Its size is 6 × 6 × 256, here $Z_{P5} = 256$. The number of ROIs extracted from the "Conv3" and "Pool5" are N_{C3} and N_{P5}, respectively, both values are selected to be $N_{C3} = N_{P5} = 400$.

Readers who are interested in more details about the process of extracting the ROIs, may refer to the object recognition works in computer vision literature like [35–37], and to the VPR literature like [3,16,17,34]. We have investigated the performance when extracting the ROIs and found experimentally that $N_{C3} = N_{P5} = 400$. By this selection, each ROI is represented by a vector of size 384 for the "Conv3" and of size 256 for the "Pool5." Then, the results are aggregated to formulate one descriptor array $\mathcal{R}_{C3}$ whose dimensions are 400 × 384 for the "Conv3," and $\mathcal{R}_{P5}$ of size 400 × 256 for the "Pool5."

As the numbers of columns are different in each descriptor due to the different sizes of the layers, we apply a PCA-based encoding method to reduce the dimensions of the $\mathcal{R}_{C3}$ descriptor from 384 to 256 to become equal to the number of columns of $\mathcal{R}_{P5}$. Similar PCA process is applied to $\mathcal{R}_{P5}$ to represent it in the eigenspace as well. The new dimensions are kept same as 256 without reduction. It is worth noting that we have built the PCA spaces using a separated dataset consisting of 2.6K images collected from Query247 [38], St. Lucia [39] and Mapillary [4,40] as detailed in [3]. This dataset contains images captured under different conditions during the day, night and many appearance changes.

Finally, the resulted descriptors of the two combined layers were concatenated to formulate a descriptor array D_t with a size of 800 × 256 for the tth image, where each row in this array represents a descriptor $R_{t,j}$, and $j \in \{1, \ldots, N_t\}$. Here, N_t is the total number of descriptors for each image, and it is selected here as $N_t = 800$. In other words, in this step, we perform a simple combination between the features extracted from the "Conv3" and "Pool5" layers. It is done by simply concatenating the descriptors $\mathcal{R}_{C3}$ and $\mathcal{R}_{P5}$. Finally, each image in the test and reference sequences is represented by

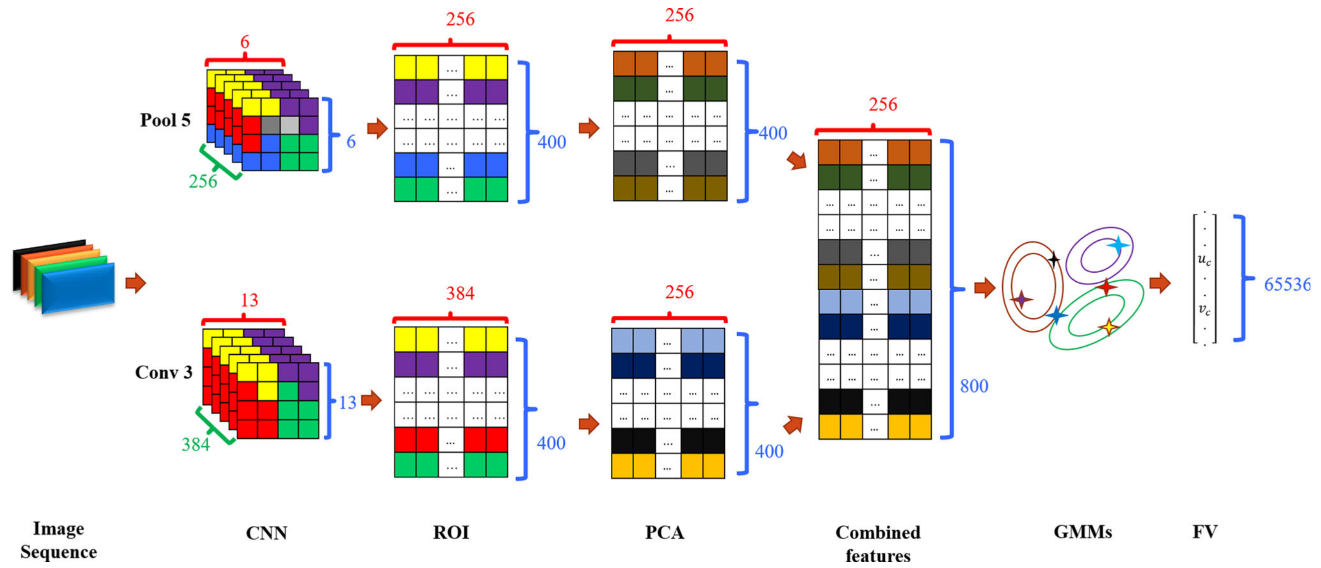

Fig. 3 The proposed visual place recognition. In this proposal, features are extracted from the layers "Conv3" and "Pool5." Regions of interest are extracted from these two layers. Following, the PCA is used to unify the dimensionality of the selected region, which will be further encoded and represented using the FV-based visual codebook. This representation is used for making the recognition decision

a similar descriptor array D_t. As shown in Fig. 1, the ROI extracted from the layers Conv3 and Pool5 refer to different kinds of features in the images, where Conv3 is about finding general features in the target image, so it can detect small or parts of the objects shown in the image like trees or roads. However, for Pool5, it is clear that this layer can detect more semantic features like a whole tree, a bigger part of the road or the whole factory shown in the middle of the image. Consequently, the two layers focus on a different level of features depends on their depth in the network. That is what makes the combination of ROIs extracted from those two layers able to bridge the gaps found using each layer separately. For example, the Conv4 layer, which comes just after the layer Conv3, has a similar level of extracted ROI features to those extracted from Conv3, as shown in Fig. 1. So, using a combination of these two layers will not enhance the recognition capability any more. Sometimes it would produce worse performance as presented in the experiments section.

3.2 FV-based Features Encoding

As mentioned in the computer vision and robotic literature, encoding the extracted deep features can empower its capability to handle the recognition challenges. To the best of our knowledge, this work is the first to investigate using the FV encoding for VPR approaches. FV has the advantage of providing smoother and less quantization errors than VLAD and Bag of Words. To do so, the descriptor generated from the previous step (PCA), i.e., D_t is fed into the GMM model which was built based on the same dataset that was used to build the PCA space. This step provides us the main components of the GMM, i.e., the weight (ω_c), mean (μ_c) and covariance (Σ_c) for each of the GMM's cluster (c). These components can be described as $\lambda = \{\omega_c, \mu_c, \Sigma_c\}$, where $c = 1, \ldots, N_c$, and N_c is the number of clusters which is set to 128 in this work.

Both of the training and testing datasets are represented using FV. For each element in the feature vector $R_{t,j}$, the following two components are calculated:

$$u_{cd} = \frac{1}{N_t\sqrt{\pi_c}} \sum_{j=1}^{N_t} P_r(c|R_{t,j}, \lambda) \frac{R_{t,jd} - \mu_{cd}}{\sigma_{cd}}, \tag{1}$$

$$v_{cd} = \frac{1}{N_t\sqrt{2\pi_c}} \sum_{j=1}^{N_t} P_r(c|R_{t,j}, \lambda) \left[\left(\frac{R_{t,jd} - \mu_{cd}}{\sigma_{cd}} \right)^2 - 1 \right], \tag{2}$$

where $d \in \{1, \ldots, N\}$ represents the elements of the vector, and N is equal to the number of the PCA components, i.e., 256. The posterior probability $P_r(c|R_{t,j}, \lambda)$ of each cluster is given as

$$P_r(c|R_{t,j}, \lambda) = \frac{\omega_c g(R_{t,j}|\mu_c, \Sigma_c)}{\sum_{n=1}^{N_c} \omega_n g(R_{t,j}|\mu_n, \Sigma_n)}; \tag{3}$$

Here, $g(R_{t,j}|\mu, \Sigma)$ is the Gaussian density function. As a result, for each image, the calculated components are concatenated to formulate the final fisher vector illustrated as

$$\Phi(I) = [\ldots, u_c, \ldots, v_c, \ldots]^T. \tag{4}$$

The length of this vector equals $N_c \times N_t \times 2$, where, as mentioned previously, N_c is 128 and N_t is the number of the extracted ROIs, which is in our case equal to 800. Then, the improved version of FV's vector is generated using the square root normalization followed by L_2 normalization applied on $\Phi(t)$.

As soon as the input image is represented using the FV representation, it is supplied to the last stage (classifier) to find out its label or matching place. We proposed to use the DTW as the system classifier (or matching algorithm).

3.3 Descriptor Matching and Recognition Decision

Finally, the system has a decision on whether it is a prior visited place or a new place. In this work, we use the Dynamic Time Warping (DTW) algorithm for aligning the reference with the test image sequences and then make the decision. The decision process is done using the Dijkstra algorithm in the *Max Flow* network algorithm [5]. Sum of absolute differences (SAD) is used along with the Nearest Neighbor Algorithm (NNA) in *SeqSLAM*. Matching using NNA is also used by the *Only Look Once*, *Holistic* and *CAMAL*, they employ the same cosine similarity adopted by our proposal, which is illustrated in Eq. (5).

In the proposed approach, the similarity matrix S in Eq. (5) is filled by the cosine similarity of each of the test images with each of the reference images using their FV representation $\Phi(i)$ and $\Phi(j)$, respectively.

$$S(i,j) = \frac{\Phi(i)^T \cdot \Phi(j)}{\|\Phi(i)\| \|\Phi(j)\|}. \tag{5}$$

Next is to fill the accumulating similarity matrix A_{cc} which represents the sum of the similarity between current matching two images and the maximum of the cumulative similarity of the neighboring images. Finally, DTW works on detecting an optimal path of matches, which is the result of backward tracing in the matrix A_{cc} choosing the previous elements with the highest cumulative similarity. Readers may refer to Hafez et al. [13,41] for more information about the sequence matching using DTW algorithm.

4 Experimental Evaluation and Analysis

We present in this section our experimental work and other implementation and evaluation details. In the conducted experiments, we compare our proposed method with the state-of-the-art VPR methods. This comparison is conducted over multiple datasets including "Garden Point" and "Berlin_A100." These datasets are detailed in Sect. 4.1, followed by the evaluation measures in Sect. 4.2 that were used to state the obtained results in Sects. 4.3–4.5. The FV implementation is adopted from the VLfeat library [42], which is built based on the FV work published in [19].

4.1 Benchmark Datasets

The main challenges and benchmarks in VPR systems can be stated as the illumination, viewpoint and dynamic objects. In order to investigate the performance of the proposed approach against these challenges, four different datasets were used. These datasets are *Garden Point* (the three possible combinations) and *Berlin_A100* datasets.

The *Garden point* dataset consists of three different sequences, each of which consists of 200 images. The first sequence is called "day_left" which was collected by walking on the left side of a route inside the "QUT" campus. The second one is called "day_right" that has been gathered from the right side of the same mentioned route. The third one was also collected on the right side but at night, and it is called "night_right." Hence, this dataset covers both "Illuminations and Viewpoint" challenges. In particular, the combination of "day_left vs day_right" sequences covers the viewpoint challenge, while the "day_right vs night_right" covers the illumination challenge, and finally both mentioned challenges are covered by the combination of "day_left Vs night_right."

The fourth dataset used in this paper is "Berlin_A100." This dataset was collected by different drivers for the same route at different times which gives a variety in terms of appearance, dynamic objects (clouds, cars, trees, etc.) and viewpoint.

4.2 Evaluation Measures

The precision–recall curve (PRC), area under curve (AUC), $F1_{\text{Score}}$, mean average precision (mAP) and mean error are used in this work to evaluate our proposal. The PRC shows the *precision* P and *recall* R values for certain values of a selected setting parameter. They are given as $P = \text{TP}/(\text{TP}+\text{FP})$ and $R = \text{TP}/(\text{TP}+\text{FN})$. Here, TP stands for the true positives, i.e., the number of matched images, FP equal is the false positives that refers to the number of queries matched with the wrong reference images, and FN is the false negative that represents the images classified as non-matched despite the fact they have corresponding images in the reference set. The number of the frames used in our experiments to consider the match as a TP is equal to 4 frames.

The AUC is calculated from the PRC using the following formula

$$\text{AUC} = \sum_{i=1}^{n-1} \frac{P_i^{min} + P_{i+1}^{\max}}{2} (R_{i+1} - R_i). \tag{6}$$

$F1_{\text{Score}}$ which is a weighted average of precision and recall, which takes into account both false positives and false negatives, was also used and can be calculated as

$$F1_{\text{Score}} = 2 \times (P \times R)/(P + R).$$

Mean average precision (mAP) represents the mean of average precision (AP), which is calculated by using the area under the precision–recall curve. The mAP measure can be calculated using the following formula

$$\text{mAP} = \sum_{i=1}^{I} \text{AP}(q_i)/I.$$

where I is the number of images in the testing set and AP (q_i) is the average precision for the ith query (image).

The mean error (the mean of the differences between the test images and the corresponding ground truth) given in frames are also used in the third set of experiments to show the superiority of *Our Approach* over other VPR methods. For all mentioned measures, a higher value means a better performance, except for the mean error where the situation is the opposite.

Implementation details are summarized in Table 1. The information of the CNN structure and its main hyperparameters for our approach and the other encoding-based algorithms are illustrated in the table. Note that all models are pre-trained ones. Also, both MaxFlow and SeqSlam were not added to the table as they are not encoding-based algorithms, i.e., they have no CNN model.

4.3 Exploring the ROIs from Different CNN Layers

In this experiment, we compare the performance of three types of features in order to show the superiority of our proposed features. These types are FV-based encoded ROIs deep features extracted from the Conv3 and Pool5 layers individually, the combinations of them in one descriptor denoted as *Our Approach*, and FV-based encoded features extracted directly from these two layers but without using any ROI.

Table 2 shows the AUC score resulted from testing the above mentioned methods over different datasets. It shows that combining the features from two layers, one from the middle and the another from the last layers, improves the system ability to handle multiple visual challenges. The Pool5 layer has achieved better performance when ROI are extracted from the feature maps of this layer. This is due to the semantic information included in the last layers which were detected and extracted by the ROI method. On the other hand, extracting ROIs from Conv3 improves the performance when there are changes in the dynamic objects like (Berlin_A100). But for the other datasets, extracting ROIs from Conv3 layer has degraded the performance. This is due to the fact that the middle layers have more general features, and extracting ROIs will not give meaningful (semantic) information, but at the same time, it will ignore the dynamic objects, which explains the improvement in the performance for (Berlin_A100).

However, testing over datasets that contain both viewpoint and illumination changes like "day_left vs night_right" and "Berlin_A100" has shown that none of the selected layers individually is able to outperform *Our Approach*. Testing over the "day_left vs night_right" dataset is shown in Fig. 4, while testing over the "Berlin_A100" dataset is shown in Fig. 5 . For the "day_left vs day_right" experiment, *Our Approach* has outperformed other methods. For "Berlin_A100" experiment, Conv3 with FV and ROI has achieved a little bit better performance. This is due to the fact that Pool5 is more likely to give attention to the dynamic objects. However, it can be noticed that *Our Approach* has achieved generally better and more stable performance for all used datasets.

4.4 Comparison with Previous VPR Approaches

In this set of experiments, we evaluated our approach by comparing its performance with some state-of-the-art VPR approaches, particularly with *Holistic*, *NetVLAD*, *CAMAL* and *Only Look Once*. These are using different CNN architectures and pre-trained using different datasets as shown in Table 1. They are classified as "Image to Sequence" approaches. Also, they use ROI extracted from deep feature maps, and apply an encoding method for the representation of the feature extracted as ROIs. Besides, we compared *Our Approach* against *SeqSLAM* [1], and the *Max Flow*-based algorithm which proposed in [5], where both are classified as "Sequence to Sequence" approaches. The comparison results are shown in Tables 3 and 4 and in Figs. 6, 7, 8, 9 and 10. In addition, Table 3 illustrates the AUC results for all mentioned approaches, and Table 4 represents their F1-score.

Note that in this set of experiments, the *Only Look Once* and *Holistic* implementation and evaluation were achieved through using the code published by the respective authors without any changes. For *CAMAL*, we use the same ROI extraction method used by *Holistic*, as mentioned by the authors. We implemented the rest of the *CAMAL* based on the information given in their published paper where the ROI extracted from Conv3 and Conv4 of the HybridNet CNN architecture were combined. In addition, *SeqSLAM* was evaluated based on the OpenSeqSLAM Matlab implementation [43] with the default parameters where the number of the sequential frames is equal to five, and the best trajectory is the one with the minimum score throughout the detected trajectories. The *Max Flow* algorithm was built by us using the "NetworkX" Python Library [44] for building the graph,

Table 1 Implementation details of the used CNN

Approach	CNN	Pre-training dataset	#Layers	#Employed layers	Codebook size	#ROI
Only look once	VGG16	ImageNet	16	Conv5_2 + Conv5_3	10,000	200
Holistic	AlexNet365	Places365	8	Conv3	256	400
CAMAL	HybridNet	ImageNet + SPED	9	Conv3 + Conv4	128	600
Our Approach	AlexNet365	Places365	8	Conv3 + Pool5	128	800

Table 2 The AUC resulted from testing the various feature selection methods over different datasets

Dataset	AUC – Threshold = 4 Frames				
	Conv3_FV	Conv3_ROI_FV	Pool5_FV	Pool5_ROI_FV	Our approach
Day_left vs day_right	0.998	0.986	0.989	0.990	**1.000**
Day_right vs night_right	0.933	0.844	0.914	**0.985**	0.947
Day_left vs night_right	0.824	0.816	0.847	0.718	**0.888**
Berlin_A100	0.792	**0.908**	0.706	0.793	0.813

Bold values indicate a higher value of the performance measure

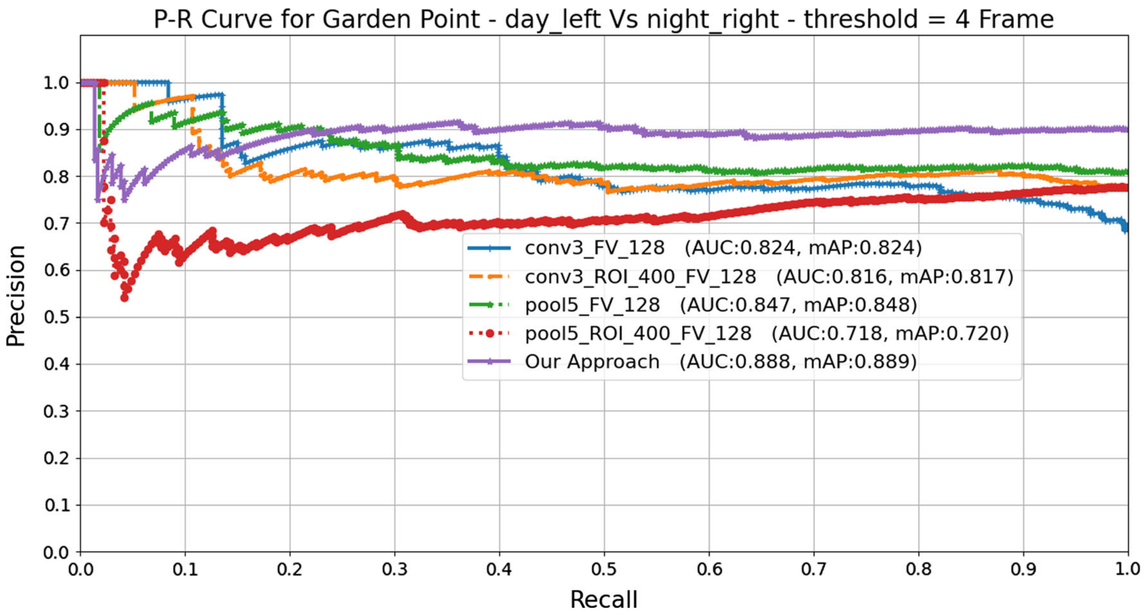

Fig. 4 PR curve resulted from testing the five different cases mentioned in Sect. 4.3 over the Garden Point dataset (day left vs night right). In all cases, descriptors are encoded with FV

Table 3 The AUC resulted from testing the various VPR methods over different datasets

Dataset	AUC – Threshold = 4 Frames						
	Only look once	Holistic	NetVLAD	CAMAL	SeqSLAM	MaxFlow-Hog	Our approach
Day_left vs night_right	0.774	0.711	0.831	0.490	0.089	0.479	**0.888**
Berlin_A100	0.740	0.654	0.651	0.664	0.265	0.669	**0.813**

Bold values indicate a higher value of the performance measure

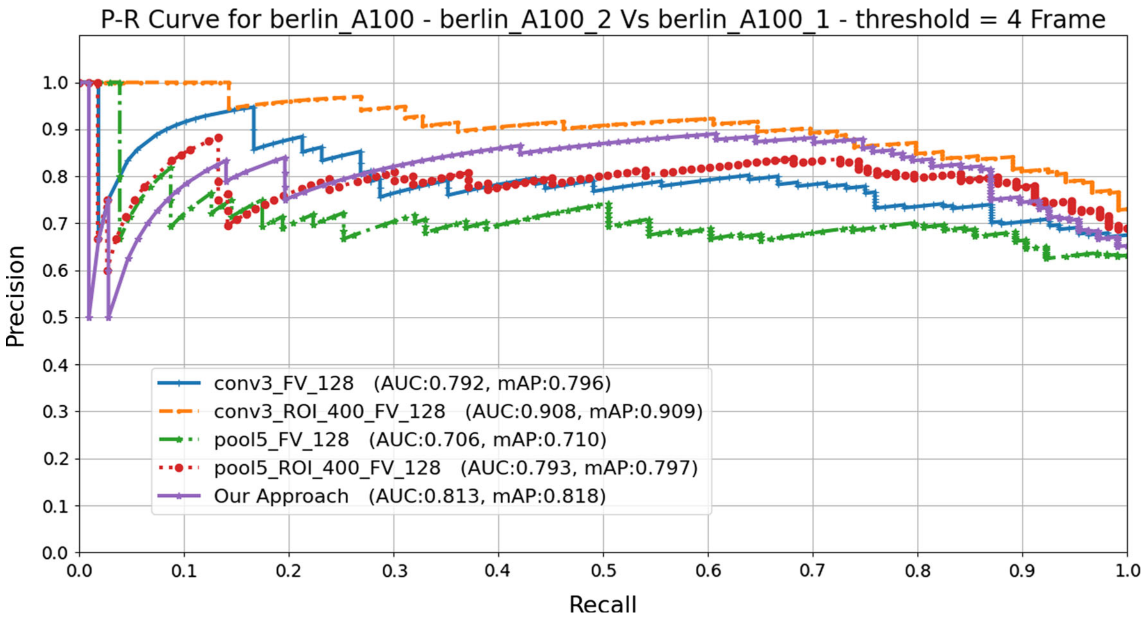

Fig. 5 PR curve resulted from testing the five different cases mentioned in Sect. 4.3 over the Berlin_A100 dataset. In all cases, descriptors are encoded with FV

Table 4 The F1-score resulted from testing the various VPR methods over different datasets.

Dataset	F1-score − Threshold = 4 Frames						
	Only look once	Holistic	NetVLAD	CAMAL	SeqSLAM	MaxFlow-Hog	Our approach
Day_left vs night_right	0.714	0.718	0.799	0.587	0.183	0.719	**0.947**
Berlin_A100	0.724	0.694	0.773	0.754	0.353	0.745	**0.787**

Bold values indicate a higher value of the performance measure

Table 5 The mean error resulted from testing the various VPR methods over different datasets

Dataset	Mean error (frame)						
	Only look once	Holistic	NetVLAD	CAMAL	SeqSLAM	MaxFlow-Hog	Our approach
Day_left vs night_right	21.015	22.745	18.545	33.875	57.000	7.058	**1.832**
Berlin_A100	8.610	7.790	6.518	6.086	20.629	3.980	**3.491**

Bold values indicate a higher value of the performance measure

and the “Dijkstra” algorithm existed in the same mentioned library is used to find the best path.

Based on the results depicted in Tables 3 and 4, *Our Approach* has outperformed the other methods. The experiments shown in Figs. 6 and 7 show also this, it is clear that all other approaches including *Only Look Once* failed to achieve higher or comparable results to *Our Approach*. This is due to the fact that *Only Look Once*, *NetVLAD* and *Holistic* use features extracted from one convolutional layer which make them immutable for the multi-challenge datasets. Even though *CAMAL* uses two convolutional layers, it is still not able to show comparable results to other approaches including ours. Moreover, *CAMAL* has performed worse than *Holistic* and *Only Look Once* as demonstrated in Fig. 6, this is due to the fact that *CAMAL* combines features from two successive layers, which makes the extracted ROI from these two layers being very similar. If, for example, a meaningless ROI were extracted from the first layer, it may also be found and extracted from the next one, which may make the final combination worse in several cases.

Also, *Our Approach* was compared against the *SeqSLAM* and *Max Flow* VPR methods where *Max Flow* was combined with handcrafted features (HOG). *Our Approach* has outperformed the *SeqSLAM* and *Max Flow*-based Hog features for all datasets, see Figs. 8 and 9 and Tables 3 and 4. *SeqSLAM* fails in such scenarios where there are differences between the reference and test sequences either in the viewpoint or the appearance. A further novel finding is that *Our*

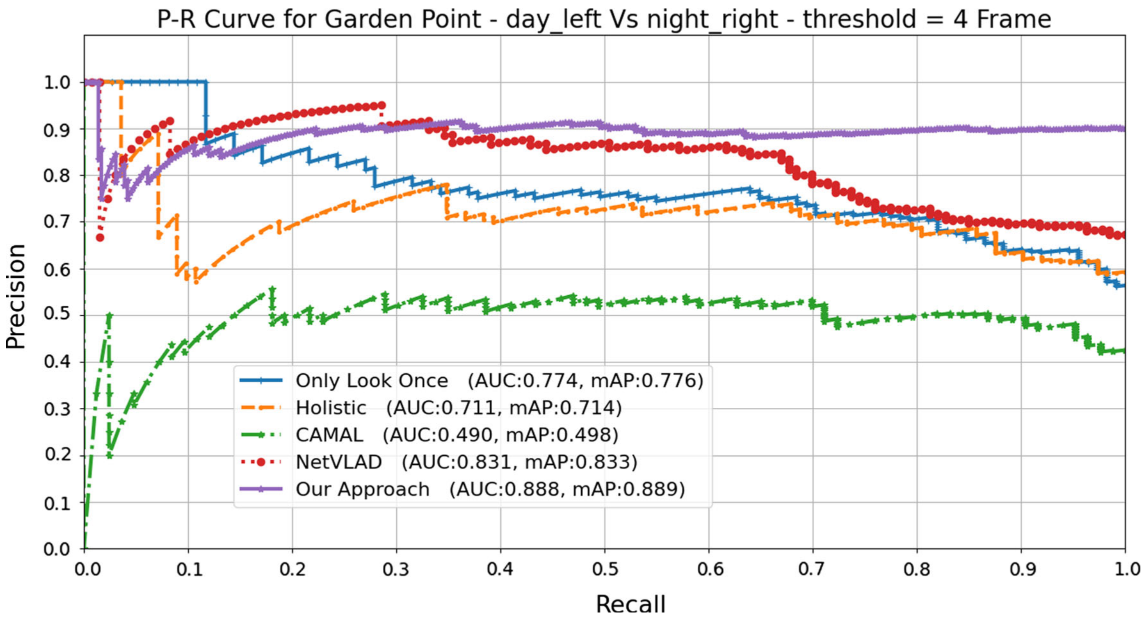

Fig. 6 The PRC resulted from comparing *Our Approach* with *Holistic*, *NetVLAD*, *CAMAL* and *Only Look once* VPR approaches over the Garden Point dataset (Day left vs Night right)

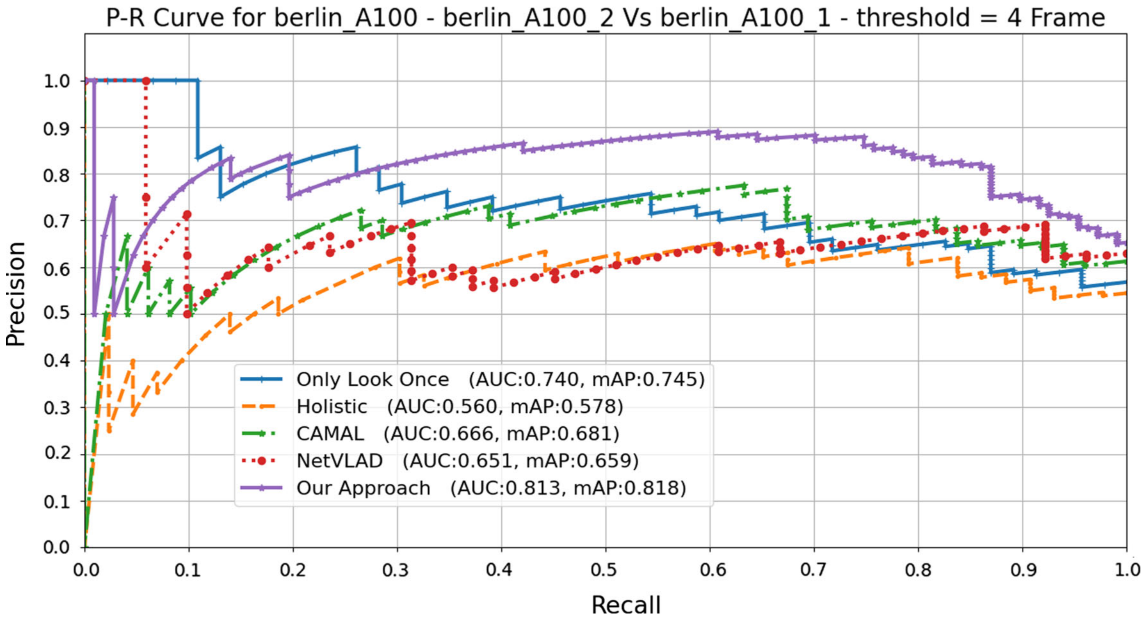

Fig. 7 The PRC resulted from comparing *Our Approach* with *Holistic*, *NetVLAD*, *CAMAL* and *Only Look once* VPR approaches over the Berlin_A100 dataset

Approach, as shown in Table 3, is more robust when multiple challenges are shown in the same dataset, while the other approaches dropped rapidly when multiple challenges are fused in the dataset, so, even though that *Max Flow*-based algorithm combined with the features extracted from the VGG16 has achieved a slightly higher performance than *Our Approach* in two of four experiments, but for Berlin_A100 it drops quiet below, see Fig. 10. This is due to the fact that there is no ROI extraction phase in this algorithm which makes the dynamic objects existing in Berlin_A100 dataset affect the performance as explained earlier in Sect. 4.3. However, the overall PR-Curve of *Our Approach*, even for high recall values, is still higher than *Max Flow*.

4.5 Error Analysis in VPR Approaches

In the VPR approaches, it is very important to take into account the error occurring when retrieving the wrong refer-

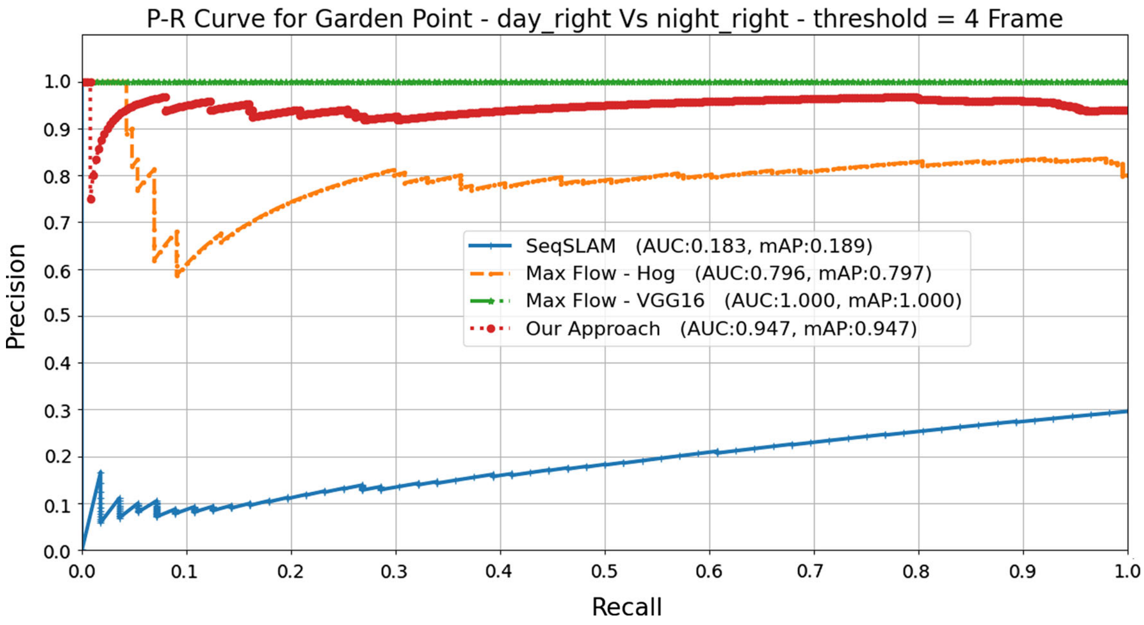

Fig. 8 The PRC resulted from comparing *Our Approach* with *Max Flow*, which uses Deep (VGG16) and (HoG) features, and with the *SeqSLAM* VPR method over the Garden Point dataset (Day right vs Night right)

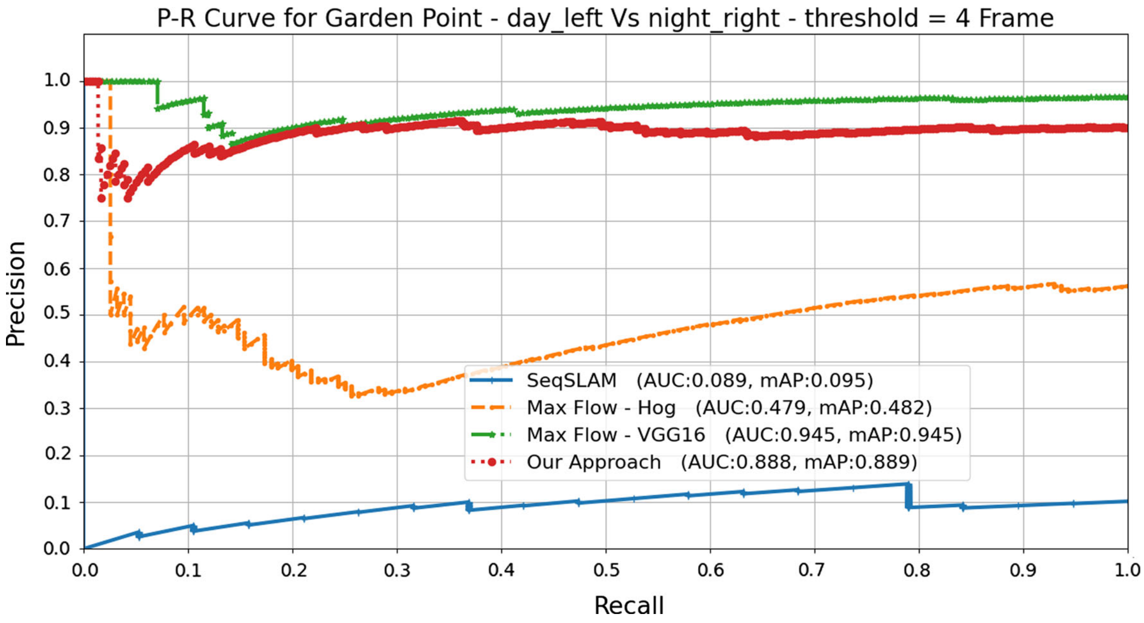

Fig. 9 The PRC resulted from comparing *Our Approach* with *Max Flow*, which uses Deep (VGG16) and (HoG) features, and with the *SeqSLAM* VPR method over the Garden Point dataset (Day left vs Night right)

ence image, where a lower error means that even if the used method retrieved a false reference, it may not lead to bad consequences in terms of the localization task. This is due to the fact that the localization component still has the ability to detect a very close location to the ground truth.

As shown in Figs. 11 and 12, multiple queries (Test images) from the Garden Point (Night right) and Berlin_A100 (Test) sequences are shown, respectively, in addition to the ground truth for those queries (Day left and Reference Sequences for the Garden Point and Berlin_A100, respectively), and the retrieved images by the algorithms mentioned on the left side of each row in the figures. The error, given in frames, between the retrieved image and the ground truth is shown in red below each of the retrieved images. For these examples, it is clear that the sequence-to-sequence methods including *Our Approach* and *Max Flow* achieved lower error than the image-to-image methods. For *SeqSLAM*, the features were extracted directly from the images using a simple

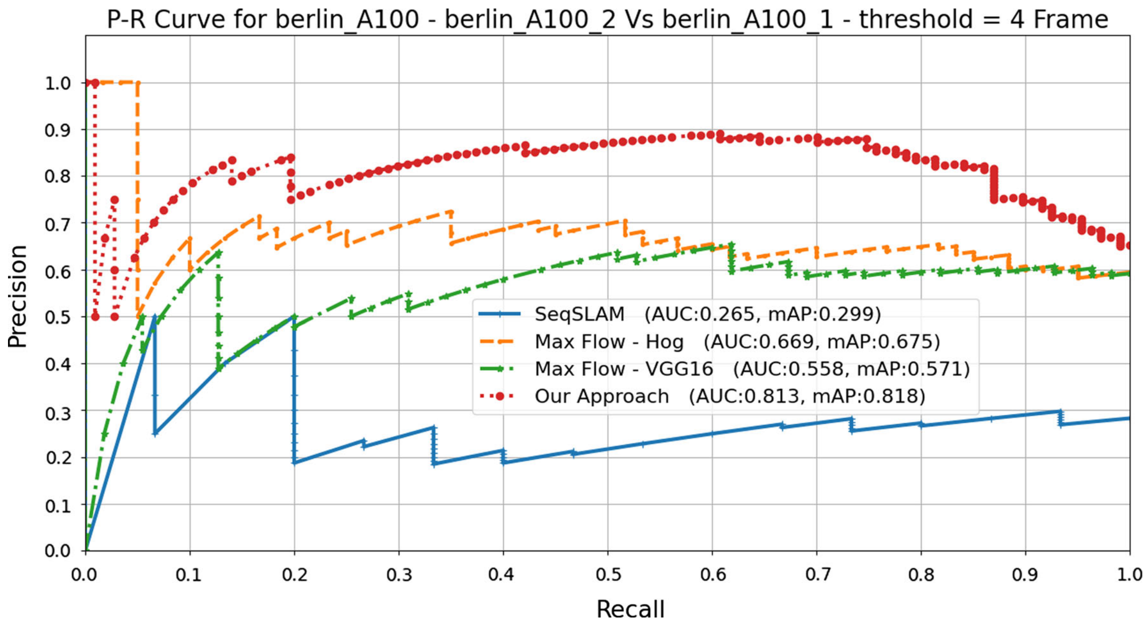

Fig. 10 The PRC resulted from comparing *Our Approach* with *Max Flow*, which uses Deep (VGG16) and (HoG) features, and with the *SeqSLAM* VPR method over the Berlin_A100 dataset

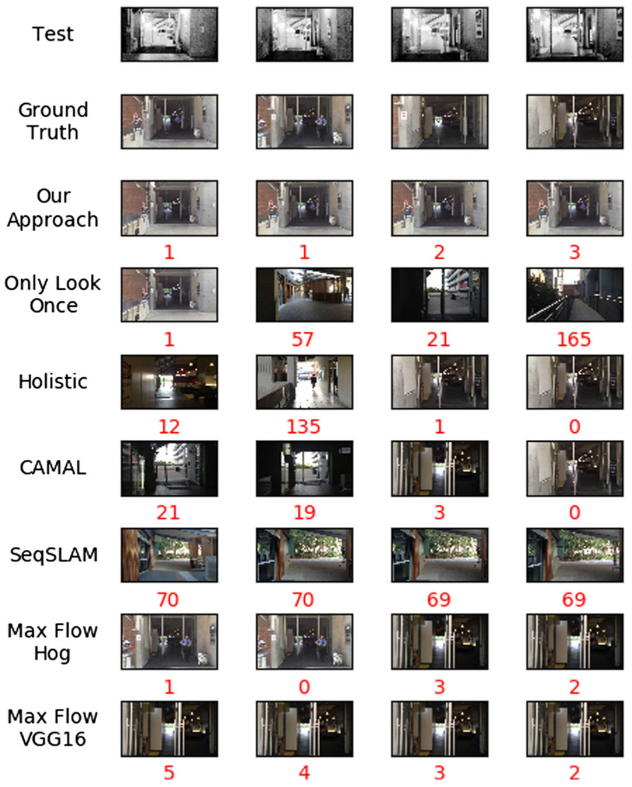

Fig. 11 The retrieved images by the different VPR methods including ours with the Garden Point dataset (day left vs night right). The first row represents the test images, the second row represents the ground truth, while the other rows of images represent the retrieved images according to the algorithms mentioned on the left side. The values in red shown below the images are equal to the difference in frames between the retrieved image and the ground truth value

down-scale sampling and contrast normalization, this makes it vulnerable for strong changes in conditions existed in the used datasets. Also, the overall mean error, as depicted in Table 5, leads to similar conclusion where the *Max Flow* and *Our Approach* gave much lower error than the other methods, and this is due to the fact that the information laid in the sequence is very important in the VPR problem [1].

5 Conclusion Remarks

Despite the VPR work made over the last decade with CNN models, there are still much more to figure out about these models. In this work, we took a step toward understanding the behavior of the different CNN layers according to their depth. A novel VPR approach is presented in this paper. It extracts ROI from two CNN layers laid in the mid and the end of the AlexNet365 model. Those regions were combined to collaborate and take the advantage of both mentioned layers after a PCA step in order to solve the multi-challenge datasets problem. Then, it is encoded with the FV encoding scheme to be fed later into the DTW algorithm. Experimental works using two challenging datasets, i.e., “Garden point” and “Berlin A100” showed that this approach has achieved the most robust results in terms of AUC, F1-score and mean error among the other VPR methods. For instance, it achieves using the area under curve (AUC) measures an average of 91% in comparison with 86% for *Max Flow* and 72% for *CAMAL*. A future research may investigate the possibility of using a lower number of PCA components which can

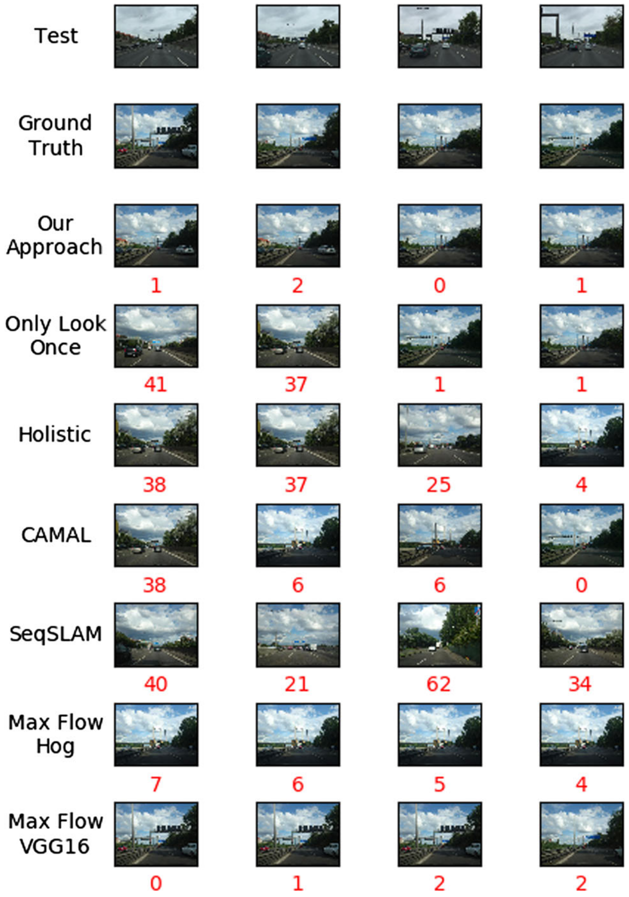

Fig. 12 The retrieved images by the different VPR methods including ours with the Berlin_A100 dataset. The first row represents the test images, the second row represents the ground truth, while the other rows of images represent the retrieved images according to the algorithms mentioned on the left side. The values in red shown below the images are equal to the difference in frames between the retrieved image and the ground truth value

give the same or better performance than the current one. Also, exploring the differences among the CNN architectures in terms of combining features from two layers through each CNN model and noticing the differences between them, which would be desirable for future work.

Acknowledgements The Titan Xp used for this research was donated by the NVIDIA Corporation. This work is supported by TUBITAK under project number 117E173.

References

1. Milford, M.J.; Wyeth, G.F.: Seqslam: visual route-based navigation for sunny summer days and stormy winter nights. In: 2012 IEEE International Conference on Robotics and Automation, pp. 1643–1649. IEEE (2012)
2. Cummins, M.; Newman, P.: Fab-map: probabilistic localization and mapping in the space of appearance. Int. J. Robot. Res. **27**(6), 647–665 (2008)
3. Khaliq, A.; Ehsan, S.; Chen, Z.; Milford, M.; McDonald-Maier, K.: A holistic visual place recognition approach using lightweight CNNs for significant viewpoint and appearance changes. IEEE Trans. Rob. **36**(2), 561–569 (2020)
4. Chen, Z.; Maffra, F.; Sa, I.; Chli, M.: Only look once, mining distinctive landmarks from convnet for visual place recognition. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 9–16. IEEE (2017)
5. Naseer, T.; Burgard, W.; Stachniss, C.: Robust visual localization across seasons. IEEE Trans. Rob. **34**(2), 289–302 (2018)
6. Bay, H.; Tuytelaars, T.; Van Gool, L.: Surf: speeded up robust features. In: Leonardis, A.; Bischof, H.; Pinz, A. (eds.) Computer Vision—ECCV 2006, pp. 404–417. Springer, Berlin, Heidelberg (2006)
7. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
8. Hafez, A.H.A., Singh, M., Krishna, K.M., Jawahar, C.V.: Visual localization in highly crowded urban environments. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2778–2783 (2013)
9. Arroyo, R.; Alcantarilla, P.F.; Bergasa, L.M.; Yebes, J.J.; Gámez, S.: Bidirectional loop closure detection on panoramas for visual navigation. In: 2014 IEEE Intelligent Vehicles Symposium Proceedings, pp. 1378–1383. IEEE (2014)
10. Arandjelovic, R.; Gronat, P.; Torii, A.; Pajdla, T.; Sivic, J.: Netvlad: CNN architecture for weakly supervised place recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5297–5307 (2016)
11. Sünderhauf, N.; Shirazi, S.; Dayoub, F.; Upcroft, B.; Milford, M.: On the performance of convnet features for place recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4297–4304 (2015)
12. Chen, Z.; Lam, O.; Jacobson, A.; Milford, M.: Convolutional neural network-based place recognition. CoRR (2014). ArXiv:1411.1509
13. Hafez, A.A., Alqaraleh, S., Tello, A.: Encoded deep features for visual place recognition. In: 2020 28th Signal Processing and Communications Applications Conference (SIU), pp. 1 – 4. IEEE (2020)
14. Kanji, T.: Self-localization from images with small overlap. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4497–4504. IEEE (2016)
15. Suenderhauf, N.; Shirazi, S.; Jacobson, A.; Dayoub, F.; Pepperell, E.; Upcroft, B.; Milford, M.: Place recognition with convnet landmarks: viewpoint-robust, condition-robust, training-free. In: Hsu, D. (ed.) Robotics: Science and Systems. Robotics: Science and Systems Conference, vol. XI, pp. 1–10 (2015)
16. Li, Z.; Zhou, A.; Wang, M.; Shen, Y.: Deep fusion of multi-layers salient CNN features and similarity network for robust visual place recognition. In: 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 22–29. IEEE (2019)
17. Hausler, S.; Jacobson, A.; Milford, M.: Multi-process fusion: visual place recognition using multiple image processing methods. IEEE Robot. Autom. Lett. **4**(2), 1924–1931 (2019)
18. Zeiler, M.D.; Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T. (eds.) Computer Vision—ECCV 2014, pp. 818–833 (2014)
19. Perronnin, F.; Dance, C.: Fisher kernels on visual vocabularies for image categorization. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007). https://doi.org/10.1109/CVPR.2007.383266
20. Jégou, H.; Perronnin, F.; Douze, M.; Sànchez, J.; Pérez, P.; Schmid, C.: Aggregating local image descriptors into compact codes. IEEE Trans. Pattern Anal. Mach. Intell. **34**(9), 1704–1716 (2012)
21. Sánchez, J.; Perronnin, F.; Mensink, T.; Verbeek, J.: Image classification with the fisher vector: theory and practice. Int. J. Comput. Vis. **105**(3), 222–245 (2013)

22. Sivic, Z.: Video google: a text retrieval approach to object matching in videos. In: Proceedings 9th IEEE International Conference on Computer Vision, vol. 2, pp. 1470–1477 (2003)
23. Arandjelovic, R.; Zisserman, A.: All about VLAD. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1578–1585 (2013)
24. Jammula, M.: Content based image retrieval system using integrated ML and DL-CNN. Ann. Roman. Soc. Cell Biol. 9656–9666 (2021)
25. Hamreras, S.; Boucheham, B.; Molina-Cabello, M.A.; Benitez-Rochel, R.; Lopez-Rubio, E.: Content based image retrieval by ensembles of deep learning object classifiers. Integrated Comput.-Aided Eng. **27**(3), 317–331 (2020)
26. Guo, H.; Liu, J.; Xiao, Z.; Xiao, L.: Deep CNN-based hyperspectral image classification using discriminative multiple spatial-spectral feature fusion. Remote Sens. Lett. **11**(9), 827–836 (2020)
27. Shakarami, A.; Tarrah, H.: An efficient image descriptor for image classification and CBIR. Optik **214**, 164833 (2020)
28. Abdul Hafez, A.H.; Arora, M.; Krishna, K.M.; Jawahar, C.: Learning multiple experiences useful visual features for active maps localization in crowded environments. Adv. Robot. **30**(1), 50–67 (2016)
29. Du, K.; Cai, K.Y.: Comparison research on IOT oriented image classification algorithms. In: ITM Web of Conferences, vol. 7, p. 02006. EDP Sciences (2016)
30. Wang, P.; Liu, L.; Shen, C.; Huang, Z.; van den Hengel, A.; Tao Shen, H.: Multi-attention network for one shot learning. In: Proceedings of the IEEE CVPR, pp. 2721–2729 (2017)
31. Chu, B.; Yang, D.; Tadinada, R.: Visualizing residual networks. CoRR (2017). arXiv:1701.02362
32. Yu, W.; Yang, K.; Bai, Y.; Xiao, T.; Yao, H.; Rui, Y.: Visualizing and comparing alexnet and VGG using deconvolutional layers. In: Proceedings of the 33rd International Conference on Machine Learning (2016)
33. Zaffar, M.; Ehsan, S.; Milford, M.; McDonald-Maier, K.: Cohog: a light-weight, compute-efficient, and training-free visual place recognition technique for changing environments. IEEE Robot. Autom. Lett. **5**(2), 1835–1842 (2020)
34. Khaliq, A.; Ehsan, S.; Milford, M.; McDonald-Maier, K.: Camal: context-aware multi-scale attention framework for lightweight visual place recognition. ArXiv preprint (2019). arXiv:1909.08153
35. Ding, J.; Xue, N.; Long, Y.; Xia, G.S.; Lu, Q.: Learning ROI transformer for oriented object detection in aerial images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2849–2858 (2019)
36. Kim, K.H.; Hong, S.; Roh, B.; Cheon, Y.; Park, M.: Pvanet: deep but lightweight neural networks for real-time object detection. ArXiv preprint (2016). arXiv:1608.08021
37. Liu, B.; Zhao, W.; Sun, Q.: Study of object detection based on faster r-CNN. In: 2017 Chinese Automation Congress (CAC), pp. 6233–6236. IEEE (2017)
38. Torii, A.; Arandjelovic, R.; Sivic, J.; Okutomi, M.; Pajdla, T.: 24/7 place recognition by view synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1808–1817 (2015)
39. Chen, Z.; Jacobson, A.; Sünderhauf, N.; Upcroft, B.; Liu, L.; Shen, C.; Reid, I.; Milford, M.: Deep learning features at scale for visual place recognition. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 3223–3230. IEEE (2017)
40. Sünderhauf, N.; Shirazi, S.; Jacobson, A.; Dayoub, F.; Pepperell, E.; Upcroft, B.; Milford, M.: Place recognition with convnet landmarks: viewpoint-robust, condition-robust, training-free. Robotics: Science and Systems, vol. XI, pp. 1–10 (2015)
41. Hafez, A.H.A., Tello, A., Alqaraleh, S.: Visual place recognition by dtw-based sequence alignment. In: 2019 27th Signal Processing and Communications Applications Conference (SIU), pp. 1–4 (2019)
42. Vedaldi, A.; Fulkerson, B.: VLFeat: an open and portable library of computer vision algorithms (2008). http://www.vlfeat.org/
43. Suenderhauf, N.: Openseqslam code (2013). https://openslam.org/openseqslam.html
44. Hagberg, A.A.; Schult, D.A.; Swart, P.J.: Exploring network structure, dynamics, and function using network. In: Varoquaux, G.; Vaught, T.; Millman, J. (eds.) Proceedings of the 7th Python in Science Conference, pp. 11 – 15. Pasadena, CA USA (2008)