2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

Visual Place Recognition in Long-term and Large-scale Environment based on CNN Feature

Jianliang Zhu, Yunfeng Ai, Bin Tian, Dongpu Cao and Sebastian Scherer

***Abstract*—With the universal application of camera in intelligent vehicles, visual place recognition has become a major problem in intelligent vehicle localization. The traditional solution is to make visual description of place images using hand-crafted feature for matching places, but this description method is not very good for extreme variability, especially for seasonal transformation. In this paper, we propose a new method based on convolutional neural network (CNN), by putting images into the pre-trained network model to get automatically learned image descriptors, and through some operations of pooling, fusion and binarization to optimize them, then the similarity result of place recognition is presented with the Hamming distance of the place sequence. In the experimental part, we compare our method with some state-of-the-art algorithms, FABMAP, ABLE-M and SeqSLAM, to illustrate its advantages. The experimental results show that our method based on CNN achieves better performance than other methods on the representative public datasets.**

I. INTRODUCTION

Long-term navigation [2] in changing environments is one of the major challenges in robotics today, so one of the main problems in visual localization is the place recognition in a long-term and large-scale environment. Regrettably, this is a difficult challenge due to the appearance of places have to cope with significant changes at different times of day, even along different weeks, different months and different seasons. These condition changes are caused in external environment, such as illumination, weather, and season. Approaches such as Fast Appearance-Based Mapping (FAB-MAP) [1] have been demonstrated mapping large, challenging environments. Recently, algorithms named SeqSLAM [3] and ABLE-M [4] defined the methods of matching sequence images to improve a few robustness of condition changes. These place recognition techniques rely on hand-crafted features, such as SIFT, or LDB [4], are very unsuitable for dealing with violent visual changes, such as occurs when moving from daytime to nighttime, season to season, or from clear weather to rain. Fig. 1 gives an example about different seasons.

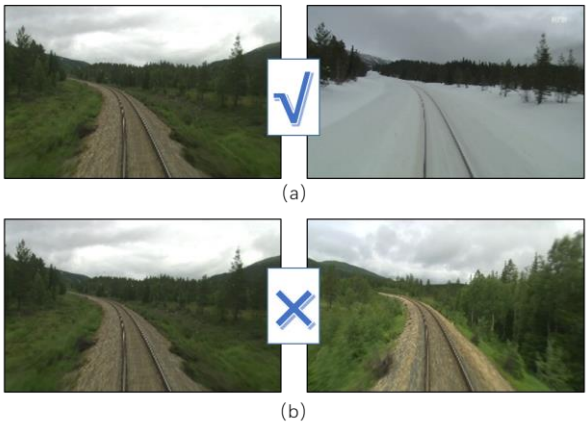

Figure 1. Visual place recognition systems must be able to (a) successfully match very perceptually different images of a same place while (b) also rejecting incorrect matches between similar image pairs of different places.

The recent development of the deep learning technology and convolutional neural networks has provided an alternative method for understanding the place recognition problem. AlexNet [5] shows that features extracted from CNNs trained fully and effectively get a better result than hand-crafted features on classification tasks. [7] proposes an effective deep learning framework to generate binary hash codes for fast image retrieval. Considering that place recognition [8] is similar in spirit to image retrieval, it is reasonable to expect that the power of CNN-based features can be leveraged in devising a solution to the problem of place recognition. However, in visual localization, deep learning is not fully applied except for a few work about object recognition [9].

In this paper, we present a simple but efficient approach that employs a modified CNN model to extract image descriptors and boosted matching of image sequences for visual place recognition. The proposed method is shown in Fig. 2 and is presented in detail in later sections. Our method is with the following characteristics:

- First, we present an improved CNN architecture based on VGG16-places365 [10]. Our model is adapted to the requirements of extracting image features by adding, deleting and fusing layers.
- Second, we transform the features obtained by the CNN layer into binary representation that reduces the computational complexity. One of the main benefits is that they can use Hamming distance to match the places.

*This work was supported in part by the National Natural Science Foundation of China under Grant 61503380 and in part by the Natural Science Foundation of Guangdong Province, China under Grant 2015A030310187.

Jiangliang Zhu and Yunfeng Ai (corresponding author) are with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100190, China (e-mail: zhujianliang15@mails.ucas.edu.cn, aiyunfeng@ucas.ac.cn).

Bin Tian is with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the Cloud Computing Center, Chinese Academy of Sciences, Dongguan 523808, China (e-mail: bin.tian@ia.ac.cn).

Dongpu Cao is with the Department of Mechanical & Mechatronics Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada, (e-mail: dongpu.cao@uwaterloo.ca).

Sebastian Scherer is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A, (e-mail: basti@cs.cmu.edu).

- Third, we present an algorithm that calculates the best candidate matching the location based on a sequence of images following some of the ideas of SeqSLAM and ABLE-M.

The rest of the paper is organized as follows. We briefly review the related work of place recognition algorithms and CNN models in Section 2. The details of our method is then presented in Section 3. Section 4 shows experimental results on three datasets to compare the performance of the presented method. Finally, we conclude the paper in Section 5 and discuss the future work.

II. Related Work

A. Convolutional Neural Networks(CNNs)

Convolutional neural networks can learn image features from training database. In the last five years, with CNNs becoming more of a commodity in the computer vision field, a number of attempts have been made to improve the original architecture of AlexNet [5] in a bid to achieve better accuracy, such as VGG [11], GoogLeNet [6], ResNet [10] and so on. Considering that place recognition is similar in spirit to image retrieval. It is separate but relevant to the place recognition problem. K. Lin, et al. [7] proposes an effective deep learning framework to generate binary hash codes for fast image retrieval.

Scene recognition is another area which has great similarities to visual place recognition, even though it's a task of classification in deep learning. Places [10], which contains more than 10 million images comprising 365 unique scene categories, is a dataset for training scene recognition CNNs models. Based on Places dataset and state-of-the-art CNNs, many researchers train some CNNs models and expose them for other researchers to use. Place recognition can be regarded as a task of image similarity matching, and some researchers achieve it with a pre-trained CNNs model. Inspiring from the advancement of deep learning, we raise a question that can we take the advantage of deep CNN to achieve visual place recognition?

B. Visual Place Recognition

Compared with other sensors, visual sensors have many advantages, such as low price and small volume, which are becoming the most popular robot sensors now. One of the popular approaches to loop closure detection is Fast Appearance-based Mapping (FAB-MAP) [1]. The proposed FAB-MAP uses a single key point descriptor, namely Scale-Invariant Feature Transform (SIFT), and also an offline bag-of-word descriptor (BoW) for landmark description and a Bayes filter for predicting the loop closure candidates. However, FAB-MAP has some inadequacies that need to be trained off-line in advance, and has poor robustness in a scene with intense environmental changes.

Nowadays, the main challenges in visual localization is the place recognition in a long-term and large-scale environment. In order to become more efficient for topology positioning in a long-term environment, other technologies are proposed. In this regard, a successful method is SeqSLAM, which was evaluated place recognition under the same route as challenging conditions. It introduces the idea of using sequence images to determine location, rather than a single image, to improve the performance of a long-term scheme.

Using sequences instead of single images utilize the temporal coherence of visual data acquired by mobile cameras, thereby reducing the number of false positives in self similar environment recognition, and increasing tolerance to local scene changes. This idea is used by the ABLE-M [4] algorithm, which basically reduces the processed images and compares the global binary descriptors with the fast calculation of Hamming distance.

The recent development of the deep learning technology and convolutional neural networks has provided an alternative method for understanding the place recognition problem. Z. Chen, et al. [8] present a visual place recognition method based on Overfeat, by combining the effective features extracted by CNNs. X. Gao, et al. [14] propose a novel method that employs a modified stacked denoising auto-encoder (SDA) to solve the loop closure detection problem for visual SLAM systems. D. Bai, et al. [13] gives a method that fuse AlexNet and SeqSLAM to detect loop closure. Recent recommendations have inspired our current work, aiming to provide a more robust and effective location recognition algorithm based on improved and simplified CNN characteristics.

III. Our Approach

In this section, we describe the main features of our proposed method: CNN model, extracting image descriptor and similarity matching. Fig. 2 shows the proposed framework.

A. CNN Model

In Section 2.A, we talked about the Places dataset, which was proposed by MIT computer science and artificial intelligence laboratory in this year, and the laboratory is holding the Places Challenge 2017 [15] on the dataset now. They want more researchers to use their dataset to train CNNs for Scene recognition tasks, and provide Places365-CNNs which is based on CNN models such as AlexNet, VGG, GoogLeNet, ResNet trained on their dataset. According to the experiment results in their paper, we choose VGG16-places365 as our basic model for place recognition, which has the best performance on multiple datasets.

Starting with LeNet-5 [16], convolutional neural networks have typically had a standard structure-stacked convolutional layers (optionally followed by batch normalization and max-pooling) are followed by one or more fully-connected layers. VGG16-places365 is the same structure with VGG which has 16 weight layers including 13 convolutional layers and 3 fully-connected layers. Places dataset contains more than 10 million images comprising 365 unique scene categories, so the dimensions of the last fully-connected layer should be revised to 365. These 13 convolutional layers are divided into 5 parts, and each layer of one part has the same data dimension. Each part is followed by a max-pooling layer which is carried out over a 2x2 pixel window, with stride 2. A stack of convolutional layers is followed by three fully-connected (FC) layers: the first two have 4096 channels

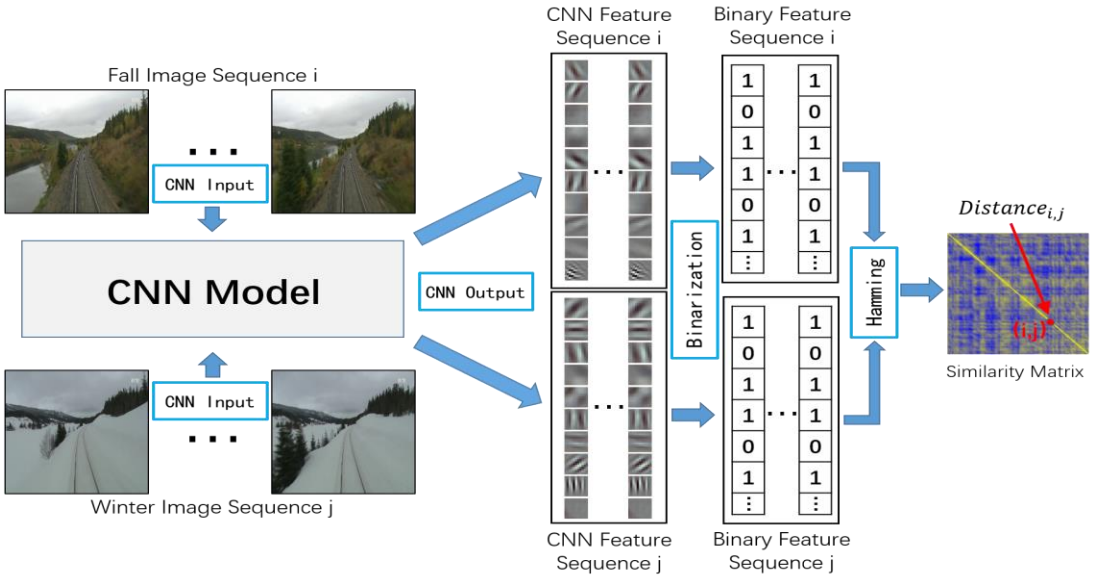

Figure 2. Global system architecture of our method for visual place recognition. *(The images are captured from two reasons of the Nordland dataset and we defined the sequences as i and j. In the last, the picture is a visualization of the Similarity Matrix.)*

each, the third performs 365-way Places classification and thus contains 365 channels (one for each class). Except for these layers, the final layer is the soft-max layer, and all hidden layers are equipped with the rectification (ReLU) non-linearity. Table 1 gives the output dimensions of the weight layers and pooling layers of the VGG16-Places365 network.

With a deep architecture, CNN is able to learn high-level semantic features at various levels of abstraction. However, spatial information of an image is lost through a full-connected layer, and this may not be desirable in applications such as visual places recognition. The experimental results in [8][13] show that the CNN-based deep features generated at the convolutional layers achieve much better performance than full-connected layers features in loop closure detection. According to these, we chosen three layers, 'conv3_3', 'conv4_3'and 'conv5_3' of VGG16-Places365 to extract image features for our task. Also we have made a lot of modifications of the CNN model, including adding several pooling layers and deleting the fully-connected layers, to reduce feature dimensions and save image processing time. Then we fuse the features of the three layers using operations of concatenating [17] after resizing them to be one-dimensional. We did a lot of experiments to adjust the network parameters of the increased pooling layers, and you can see the experiment details in Section 4. The final model structure is shown in Fig. 3.

B. Feature Descriptor for Place Recognition

Visual features are one of the most important factors that affect the accuracy of the image matching. Our method use CNN features extracted from CNN model given above instead of traditional hand-crafted features to calculate the similarities between images. Floating-point is the type of CNN features that we final get from the module. We named the feature as F_{cnn}, and it has a dimension of 1×100352. A practical way to reduce the computational cost for image matching is to convert the feature vectors to binary codes, which can be quickly compared using Hamming distance. We first normalize each of its element to 8 bits integer (0~255), and get the integer feature F_{cnn}^{int}, as shown in (2). Then, F_{cnn}^{int} can be easily converted to binary feature F_{cnn}^{bin}.

$$F_{cnn}^{int} = \frac{F_{cnn} - \min(F_{cnn})}{\max(F_{cnn}) - min(F_{cnn})} \times 255. \quad (2)$$

C. Efficient Matching of Binarization

Matching Hamming distance with binary descriptors is faster and more effective than matching descriptors with the L_2-norm, and used here to calculate the distance between images. In a lot of research, we note that they calculate the similarity of two frames by matching single image. If we defined feature descriptors of two images as $F_{cnn(i)}^{bin}$ and $F_{cnn(j)}^{bin}$, we can calculate their Hamming distance HmD_{ij} to express the similarity. The calculation process is shown in (3).

$$HmD_{ij} = HmD_{ji} = bitsum\left(F_{cnn(i)}^{bin} \oplus F_{cnn(j)}^{bin}\right). \quad (3)$$

Because of the better performance in a long-term and large-scale environment, places are considered as image sequences instead of single images, as introduced in works such as [3][4]. In our method, we defined S_{length} as the length of image sequences for matching current frame. So the image sequence of the i-th frame consists of continuous images in range $(i - S_{length} + 1, i)$, and we concatenate $F_{cnn(i-k+1)}^{bin}$, $F_{cnn(i-k+2)}^{bin}$, …, $F_{cnn(i)}^{bin}$ as the final feature F_i for matching. In this case, we can get the distance between images with sequence information using (4). This distance is the similarity score of the different places and we keep it in the similarity matrix (M). The places are recognized successfully if we find the distance of two frames is less than a given threshold.

$$Dist_{ij} = Dist_{ji} = \frac{\sum_{k=0}^{S_{length}-1} HmD_{i-k,j-k}}{S_{length}}. \quad (4)$$

TABLE I. OUTPUT DIMENSIONS OF EACH LAYERS OF THE VGG16-PLACES365 NETWORK

Conv1_1	*Conv1_2*	*Pool1*	*Conv2_1*	*Conv2_2*	*Pool2*	*Conv3_1*
3211264	3211264	802816	1605632	1605632	401408	802816
Conv3_2	*Conv3_3*	*Pool3*	*Conv4_1*	*Conv4_2*	*Conv4_3*	*Pool4*
802816	802816	200704	401408	401408	401408	100352
Conv5_1	*Conv5_2*	*Conv5_3*	*Pool5*	*Fc6*	*Fc7*	*Fc8*
100352	100352	100352	25088	4096	4096	365

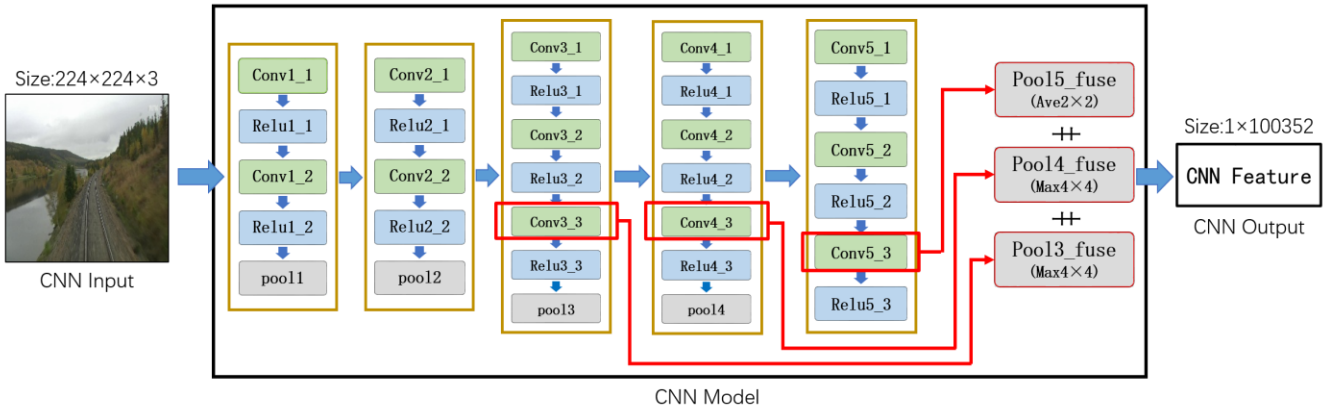

Figure 3. CNN model based on VGG16-Places365 for visual place recognition. *(All the fully-connected layers are removed, and three pooling layers named pool3_fuse, pool4_fuse, pool5_fuse, are added to the back of Conv3_3, Conv4_3 and Conv5_3, respectively. The outputs of three polling layers are fused as the final CNN features.)*

IV. PERFORMANCE EVALUATION

In this section, a set of offline experiments is demonstrated to evaluate the performance of our method. Our implementation is a python program based on Caffe [18], which is an open-source deep learning framework. We start with introducing the datasets and evaluation metrics, and then compare with performance to several well-known algorithms on the public datasets.

A. Datasets and Evaluation Metrics

The first dataset used for the experiments is the City Centre [1] dataset originally used by FAB-MAP. It is a basic dataset, and is widely used in loop closure detection and place recognition research experiments, so we use it to adjust and optimize the network model. Then we have performed tests using the Nordland [2] datasets, which is recorded in long-term conditions using monocular cameras. According to parameter settings of our CNN model, we will do a new size pretreatment of 224×224 for every image before they enter the network.

The most popular evaluation approach of place recognition algorithm is to draw a Precision-Recall(PR) curve, which gives a more informative picture of the algorithm performance. Its main elements are defined as follows: Precision is defined as the number of true positive places to the total number of detections; Recall is defined as the number of true positive places to the number of ground truth places. (5) shows the compute process. We obtain the PR curves by scanning the different distance threshold θ as shown in (6).

$$Precision = \frac{TP}{TP+FP}, \quad Recall = \frac{TP}{TP+FN}. \tag{5}$$

$$Place\ Recognition = \begin{cases} True & if\ M_{i,j}<\theta \\ false & otherwise \end{cases}. \tag{6}$$

B. Results in City Center Dataset

The first dataset, City Center [1], was collected along public roads near the city center by Cummins and Newman. It contains 1237 pairs of images with a size of 640 × 480 taken by two cameras (left and right) on the robot as it was driven through the environment at a frequency of one image every 1.5 m. The images included dynamic objects, additionally, it was collected on a windy day with bright sunshine, which makes the abundant foliage and shadow features unstable, as shown in Fig. 4(b) and 4(c). The dataset GPS information and the ground truth were provided. The robot travels twice around a loop with total path length of 2 km, and we can achieve place recognitions at these locations when the robot running around the second loop which marked with red curve in Fig. 4(a).

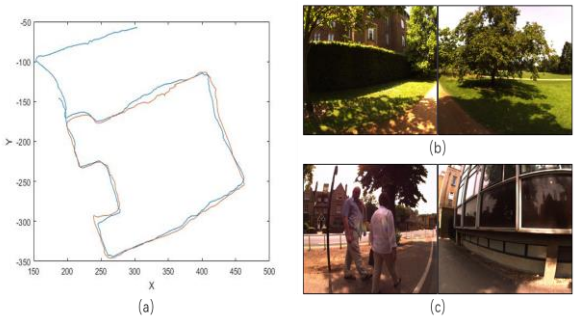

Figure 4. City Center Dataset. ((a) is the visualization of GPS information, the red curve is the second loop which the robot runs around, and we should get place recognations at these positions. (b) and (c) are two pairs of *representative* pictures of this dataset.)

Because the input of the network can only be one image, we only take the images of left camera as our test set. Also we modify some values of ground truth after the robot comes out of the loop, because the image of one camera is completely different when the binocular robot moves backward in the same position. In this case, it is impossible to realize the place recognition.

First of all, we do some experiments on every layer of VGG16-Places365, and the PR curves are shown in Fig. 5(a). As expected, the results prove that the best effect is obtained extracting convolution features, and the performance of every layer is better than OpenFABMAP [20], which is an open toolbox of FABMAP algorithm. The other parts of Fig. 5 give experimental results of the added pooling layers, "pool3_fuse", "pool4_fuse" and "pool5_fuse", respectively. By adjusting the type of these layers, MAX or AVE, and the size of the filters, 2×2, 4×4, 7×7, 8×8 or 14×14, we get the best parameter settings of every layer with a comprehensive consideration of real time and accuracy. When the feature dimension is less than a certain value, the effect of the algorithm becomes worse sharply, and the maximum filter is better than the average filter when the size of filter becomes larger. The pool5_fuse layer uses an average filter with the size of 2×2, the pool3_fuse layer and pool4_fuse layer all use a maximum filter with the size of 4×4. Also, we give the experimental results obtained by the method of fusion of multi-layer features with these settings, and they are showed in Fig. 5 too. It's easy to see that the fusion approach has achieved better results than the single layer, and we think the reason is that feature fusion of multiple layers contains more spatial information. We use this as our final CNN model like Section 3.A describing at last.

C. *Results in Nordland Dataset*

The Nordland dataset [2] records the 728 km train ride in northern Norway from the same view in the front of a train in four different seasons. Therefore, the dataset can be considered to contain a cycle and traverse four times. As illustrated in Fig. 6, the landscape has changed dramatically from the snow-cover in winter over the fresh vegetation and green vegetation in spring and summer, and to the fall colored leaves. Most of the journeys are through natural scenery, but trains also pass through the urban areas, occasionally stopping at railway stations or signal stops. It may be the longest and the most challenging dataset can be used for long-term visual place recognition evaluation at present. After processing, the data is determined as 25fps and a size of 1920 × 1080, and the image sequence is synchronized, that is, each sequence at the same time point data represents the same location.

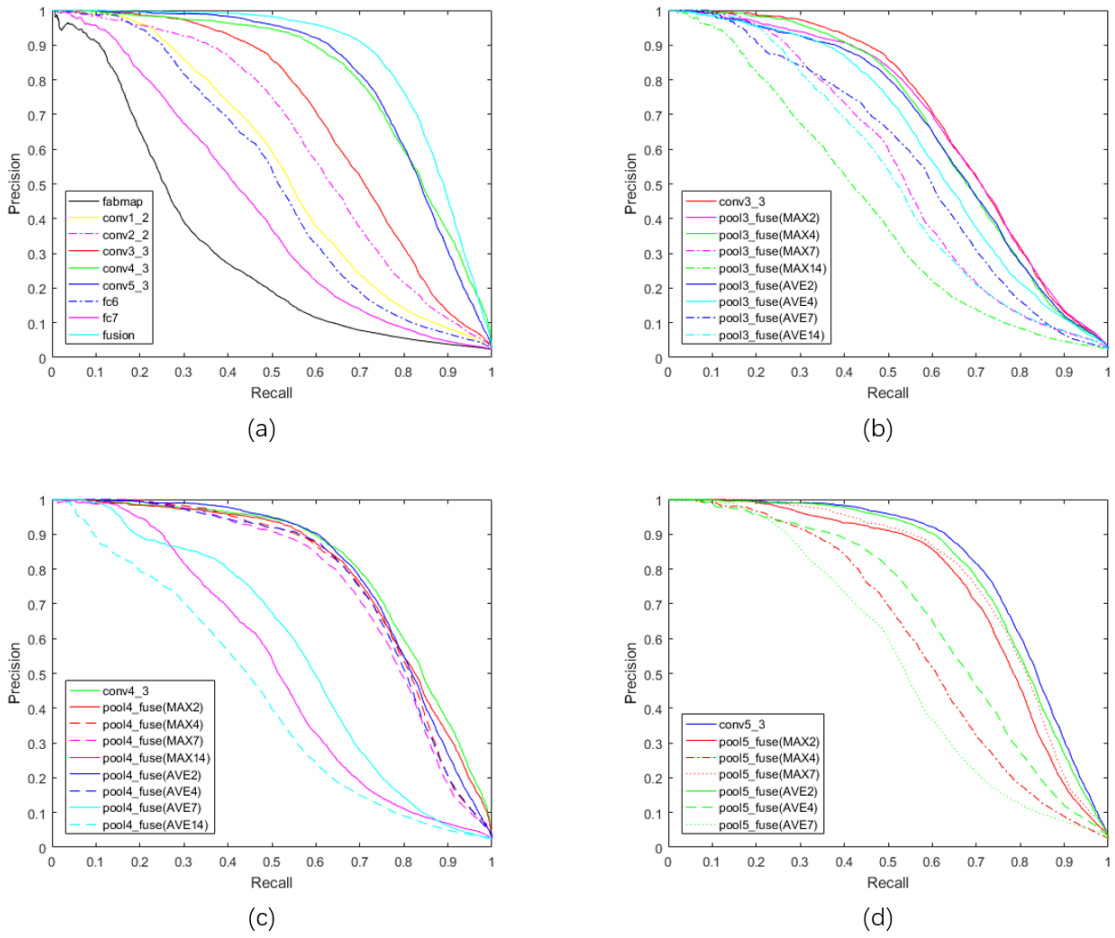

Figure 5. Experimental results on City Center Dataset. *(The curves about different layers of VGG16-Places365 is shown in (a). (b), (c), (d) show the results about the added pooling layers with different settings)*

In our method, we follow the issue of matching image sequences instead of single image for identifying places. On Nordland dataset, we first do experiments with different lengths of image sequences by comparing sequences between spring and fall seasons. This method can achieve better results in long-term and large-scale visual place recognition, as shown in Fig. 7. Attending to the PR curves advanced in the picture, we can find that the effect of the algorithm become better and better with the increase of S_{length}, which proves the correctness of our idea. But when the S_{length} is above 200, the effect of the algorithm starts to get a limit. We analyze that this happens when the S_{length} is sufficiently long to contain some particular positions that cannot be matched, which can be treated as noise. When the accuracy and complexity are taken into consideration, the optimal configuration of sequence length for 25fps data is 200. For this reason, we apply $S_{\text{length}} = 200$ in In other experiments.

Then, we compare the performance of our method with the main state-of-the-art works, include FAB-MAP, SeqSLAM and ABLE-M algorithms. The evaluations are realized thanks to the source codes developed by authors of OpenFABMAP [20], OpenSeqSLAM [2] and OpenABLE [21]. If we don't specify any of the parameters, we are using the default settings in the open source codes.

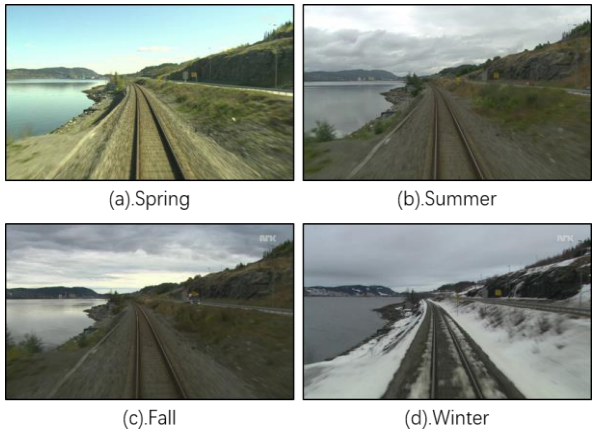

Figure 6. Example images of Nordland Dataset for every season.

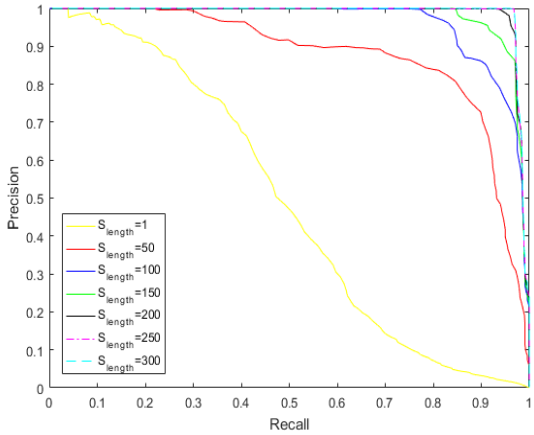

Figure 7. PR curves for different S_{length}.

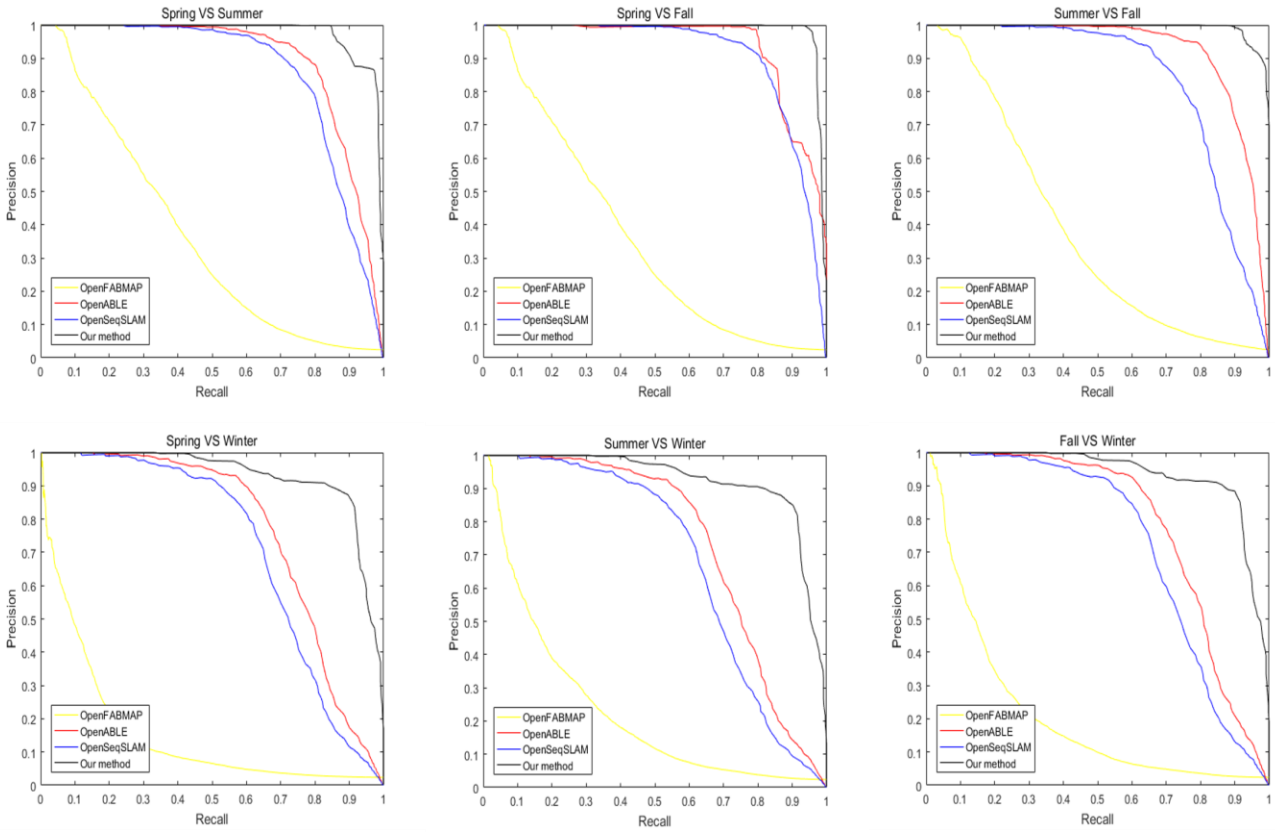

Figure 8. Experimental results on Nordland Dataset.

Now, we process results in the six combinations, spring contrast summer, spring contrast fall, spring contrast winter, summer contrast fall, summer contrast winter, fall contrast winter, for sequences corresponding. These evaluations are described by the PR curves shown in Fig. 8, where can we observe the influence of different seasons on the performance of place recognition. It is worth noting that OpenFABMAP without matching image sequences has achieved worse results than other approaches. Besides for place recognition, the recall with precision of 100% is also a good performance indicator. At 100% precision, our approach achieves better recall than other methods. It should be noticed that influenced by the sequences at the beginning were not complete, it is a limit to achieve 100% recall for the sequence-based approach. Under the same conditions, the experimental performance on winter is poor because of snow covered increases the difficulty of recognition.

D. Discussions

From the above three parts of the experiment, we can see that our method based on CNN in location recognition tasks have great advantages by comparing with traditional methods, which describing images using hand-crafted feature. We give the reasons as follows: (1) the image descriptors through learning on a large number of data by CNN can more accurately describe the difference between images, (2) feature by fusing three best CNN layers retained more spatial information of image, (3) recognition based on image sequence removes the effect of noise places. Also, our method does not achieve good recognition results at some places in above experiments. For example, in continuous multi frame images, most of the fields are covered by moving objects, or occupied by the sky and ground snow. At present, the performance of all recognition algorithms is poor, and it is the most difficult problem to be solved for visual place recognition in long-term and large-scale environment.

V. CONCLUSION

In this work we present a simple and effective CNN framework based on VGG to extract the image descriptors for place recognition. We add three pooling layers with suitable filters behind convolutional layers "conv3_3", "conv4_3" and "conv5_3", and fuse their outputs to be descriptors combine binarization. Also, the final binary strings for describing places are extracted from image sequences instead of single images, and matched for recognition by Hamming distance. The idea of this article comes from our previous work [19] about large-scale traffic scene. Our method has proved that it can successfully accomplish a long-term and large-scale visual place recognition by comparing with other state-of-the-art methods, such as FABMAP, ABLE-M or SeqSLAM, on the representative public datasets which have extreme changes of season, environment or viewpoint.

ACKNOWLEDGMENT

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

[1] M. Cummins and P. Newman, "FAB-MAP: Probabilistic localization and mapping in the space of appearance," International Journal of Robotics Research (IJRR), vol. 27(6), pp. 647-665, Jun. 2008.

[2] N Sünderhauf, P. Neubert, and P. Protzel, "Are we there yet? Challenging SeqSLAM on a 3000 km journey across all four seasons," in Workshop on Long-Term Autonomy at the IEEE International Conference on Robotics and Automation (W-ICRA), Karlsruhe, 2013.

[3] M. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in IEEE International Conference on Robotics and Automation (ICRA), St. Paul, 2012, pp. 1643-1649.

[4] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Towards life-long visual localization using an efficient matching of binary sequences from images," in IEEE International Conference on Robotics and Automation (ICRA), Seattle, 2015, pp. 6328-6335.

[5] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," In the 25th International Conference on Neural Information Processing Systems, Lake Tahoe , 2012, pp.1097-1105.

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. "Going deeper with convolutions," In IEEE Conference on Computer Vision and Pattern Recognition, Boston, 2015, pp. 1-9.

[7] K. Lin, H. Yang, J. Hsiao, and C. Chen, "Deep learning of binary hash codes for fast image retrieval," In IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, 2015, pp. 27-35.

[8] Z. Chen, O. Lam, A. Jacobson, and M. Milford, "Convolutional neural network-based place recognition," arXiv: 1411.1509, 2014.

[9] L. Bo, X. Ren, and D. Fox, "Learning hierarchical sparse features for rgb-d object recognition," International Journal of Robotics Research, vol. 33(4), pp. 581-599, April 2014.

[10] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million Image Database for Scene Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. pp(89), pp. 1-1, July 2017.

[11] K. Simonyan and A. Zisserman. "Very deep convolutional networks for large-scale image recognition," arXiv: 1409.1556v2, 2015.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," arXiv: 1512.03385, 2015.

[13] D. Bai, C. Wang, B. Zhang, X. Yi and X. Yang, "CNN Feature boosted SeqSLAM for Real-Time Loop Closure Detection," arXiv: 1704.05016v1, 2017.

[14] X. Gao and T. Zhang, "Loop closure detection for visual slam systems using deep neural networks," In the 34th Chinese control conference, Hangzhou, July 2015.

[15] http://placeschallenge.csail.mit.edu/.

[16] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, Vol. 86(11), Nov. 1998, pp. 2278-2324.

[17] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Fusion and binarization of CNN features for robust topological localization across seasons," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, 2016.

[18] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," arXiv:1408.5093, 2014.

[19] Jianliang Zhu, Yunfeng Ai, Bin Tian, "Real-time vehicle queue estimation of large-scale traffic scene," in IEEE International Conference on Image, Vision and Computing, 2017, pp. 1160-1165.

[20] A. J. Glover, W. Maddern, M. Warren, S. Reid, M. Milford, and G. F. Wyeth, "Open FABMAP: An open source toolbox for appearance based loop closure detection," in IEEE International Conference on Robotics and Automation (ICRA), Saint Paul, 2012, pp. 4730-4735.

[21] R. Arroyo, L. M. Bergasa, and E. Romera, "Open ABLE: An Open-source Toolbox for Application in Life-Long Visual Localization of Autonomous Vehicles," In IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Nov. 2016.