Light-weight Visual Place Recognition Using Convolutional Neural Network for Mobile Robots

Chanjong Park, Junik Jang, Lei Zhang, and Jae-Il Jung*.
Software R&D Center, Samsung Electronics Co., Ltd., Seoul, Republic of Korea
{cj710.park, ji.jang, lei527.zhang, and ji0130.jung}@samsung.com

***Abstract*-- Place recognition is one of the essential components for mobile robot navigation. In this work, we present a light-weight convolutional neural network (CNN) approach for visual place recognition. Our proposed approach specifically targets embedded systems. To reduce the computational complexity in the network, we design a fully convolutional network architecture with fewer layers and filters. The proposed network directly learns a vector space where its distance corresponds to place similarity by metric learning. For effective training, we adopt triplet embedding with image dataset captured at various viewpoints. Such approach allows the network to embed scenes directly without any further training process on robots. The experimental results show that the proposed method significantly outperforms conventional algorithms including other CNN-based approaches in terms of accuracy and computational time.**

I. Introduction

With the development of artificial intelligence and robotics technology mobile robots have received a significant amount of attention. Various types of robots including cleaning robots, social robots, and drones have already been commercialized. In order to provide consumers with conveniences, these mobile robots are required to autonomously navigate in various environments.

One of the most difficult problems in autonomous navigation is a place recognition which recognizes previously visited locations. In many situations external environment is unpredictable for mobile robots, various sensors such as ultrasound and infrared sensors are used to localize mobile robots [1]. However, these active sensors have a risk of malfunction in extreme environments, and can be a factor of rising product prices.

Visual place recognition is an alternative and a supplement to the active sensor-based place recognition [2-4]. The major challenge of visual place recognition is to achieve robustness to diversity in scene appearance [5]. Changes of illumination or viewpoint can have significant impacts on place recognition results. Also, different places with similar appearance may be falsely recognized as the same scene.

Recently proposed approaches for visual place recognition, especially deep neural networks, usually have complex procedures, which make them impractical to be applied to embedded systems with limited computation power. Although mobile robot products with server-client model can solve this problem, it may be inconvenient for customers to construct a wireless network, and bring additional concerns of potential privacy exposure problem.

In this paper, we propose a light-weight visual place recognition technique based on a convolutional neural network (CNN). Our proposed architecture is specifically designed for mobile robots equipped with embedded systems which have limited computation power. Experimental results show that our proposed architecture is robust to various conditions, while providing place recognition results within reasonable time.

II. Related Work

The visual place recognition task has been traditionally regarded as an image retrieval task. The location of the query image is estimated using the locations of the most visually analogous images from geo-tagged database.

The main technical goal is to describe scenes with robust forms to variability in scene appearance. Recent visual place recognition approaches can be categorized into two types; those that are based on collections of local regions in an image and those that directly describe the entire scene.

The local regions-based approaches are preceded by detecting key-points which have striking patterns such as edges and corners, and produce a single vector representation from these patterns [6-8]. These methods are robust to both appearance and viewpoint change because they benefit by local descriptors such as Speed Up Robust Feature (SURF) and oriented FAST/rotated BRIEF (ORB) [9, 10].

Images which lack sufficient textures may have difficulty in extracting key-points. Global descriptors can be an alternative for these images. Descriptors for the entire scene are generated from predefined regions using a grid-based pattern. Histogram of oriented gradients (HOG) [11] and Gist are widely used for place recognition [12], which use gradients and Gabor filters for representing scenes, respectively.

Recently, deep neural networks have emerged as a dominant image representation framework. The CNN that Krizhevsky *et al.* proposed led breakthrough for image feature embedding [13] in 2013, and the researches have been continued to reduce the complexity while improving the performance of the network. The recently proposed networks, GoogLeNet and SqueezeNet, use 12-50 times fewer parameters [14, 15]. However, such networks used in general still require a large amount of computation; They are not suitable for mobile robots equipped with embedded systems.

978-1-5386-3025-9/18/$31.00 ©2018 IEEE

III. Light-weight Visual Place Recognition

In this paper, we propose a visual place recognition approach which specifically targets embedded systems. The main contribution of this paper is that we design a light-weight CNN for describing scenes and employ metric learning to avoid additional training process on mobile robots. We also capture thousands of indoor images at different viewpoints, and train the network with these images to be invariant to changes in viewpoint.

A. *Network Architecture*

The conventional CNNs have too many layers and filters to be calculated on embedded systems. The proposed network is composed of a stack of only five convolutional modules instead of the commonly used fully-connected layers to reduce memory usage and speed up execution. Each module consists of a convolutional layer followed by pooling and activation layers, as shown in Fig. 1.

A pooling layer summarizes the responses of the convolutional layers to reduce the dimensionality. We use a max-pooling algorithm which selects the maximum output within a rectangular neighborhood. This layer makes the output become invariant to translation of the input caused by viewpoint change. The *tanh* function is employed as activation function because it has strong gradients and can avoid bias in the gradients. The output of the last convolutional module is fed to the reshaping layer which produces a normalized 1536-dimensional descriptor.

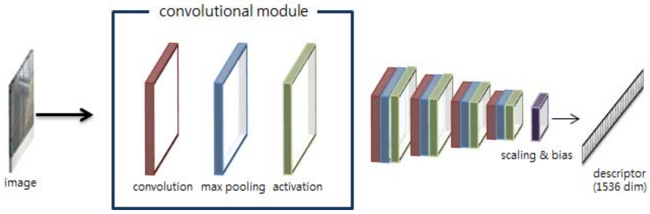

Fig. 1. An architecture of the proposed light-weight CNN

The first convolutional layer filters the 320×240×3 input image with 8 kernels of size 5×5×3. The second convolutional layer has 16 kernels of size 5×5×8. The third convolutional layer has 32 kernels of size 3×3×16. The fourth convolutional layer has 64 kernels of size 3×3×32 and the fifth convolutional layers has 32 kernels of size 3×3×64, respectively. The weights in the network are trained to learn a vector space where its distance corresponds to scene similarity

B. *Training by Triplet Embedding*

It is inefficient to train the neural network model on a mobile robot due to its limited resources; we employ triplet embedding based metric learning to train the network off-line**.** The similarities between scenes are encoded pairwise with the triplet loss which allows the network to embed scenes directly without any further training process on robots. [16].

For this off-line training to work, first we define two images as a positive pair if they are photographed from different view angles at the same scene with same distance. Also, we define two images as a negative pair if they are photographed at different scenes or with different distances at the same scene (described in experimental results section).

Triplet embedding method is a very effective way to train similarity between each image pairs. Unlike contrastive embedding, which minimizes the distance between a pair of data, we randomly pick one positive image (p) and three negative images (n_k) per each anchor image (a) and group them into one positive pair and three negative pairs. Then our model trains the similarity using the difference of vector distances between the positive pairs and negative pairs.

For each iteration in the training process, the parameters in our proposed network are updated to produce a vector space that the distance between positive pair of data is smaller than the distances between three negative pairs of data. Our network uses the loss function below. The vector obtained by embedding an image through our proposed network is represented by *f(x)*. The loss function is to compute the difference of $D\ (a,\ p)$ and $D\ (a,\ n)$, which are L2-Norm between the anchor image vector $f(x_a)$ and the positive image vector $f(x_p)$, and L2-Norm between the anchor image vector $f(x_a)$ and negative image vector $f(x_n)$, respectively.

$$Loss = \frac{\sum_{k=1}^{3}\{\max(0, D(a,p) - D(a,n_k) + C)\}}{2} \qquad (1)$$

$$where\ D(a,p) = \left\|f(x_a) - f(x_p)\right\|_2$$

Moreover, to better distinguish this, we set the marginal constant *C* so that the difference between $D\ (a,\ p)$ and $D\ (a,\ n)$ is at least C. When the distance between $D\ (a,\ p)$ and $D\ (a,\ n)$ is sufficiently large, the max function is used to prevent the loss value from becoming negative.

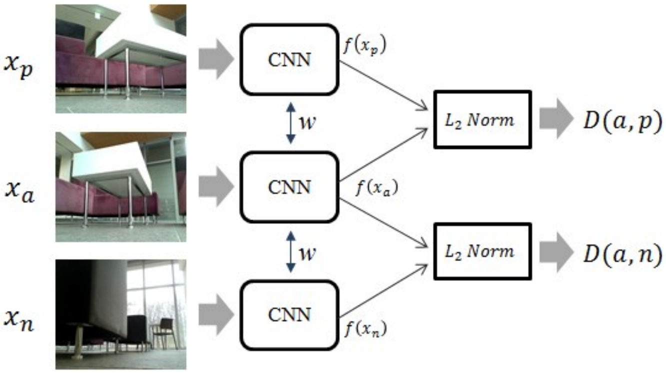

Fig. 2. Training procedure by triplet embedding

We have trained a neural network model that can effectively obtain the similarity between various images through the described method above. Using our trained model, an image is vectorized by forwarding the network and the vector similarity is calculated to obtain the final matching image.

IV. Experimental Results

A. *Dataset and Training Methodology*

The KTH-IDOL2 database [17] is often used for evaluation of visual place recognition tasks, but it lacks variety in angular deviation. If robots deviate from their desired trajectories or navigate without a pre-defined trajectory, there can be angular deviation on captured images even if the robots are the same distance away from a scene.

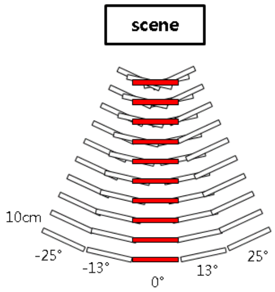

Fig. 3. The configuration of the custom dataset

Therefore, we created the additional custom dataset with the configuration shown in Fig. 3. We captured one scene at five different angles: -25°, -13°, 0°, 13°, and 25°, and each set of the same angle contains ten images with ten-centimeter intervals. With this configuration, we captured 40 scenes for training and 60 scenes for evaluation in a total of 100 scenes. Fig. 4 is an example of a set of images for the same scene.

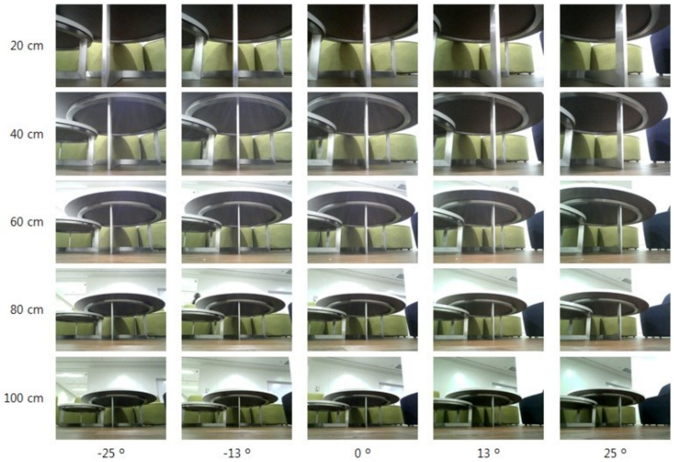

Fig. 4. An example of the custom dataset captured at the different viewpoints

We compared the proposed method to the conventional algorithm, BoW, and CNNs, GoogleNet and SqueezeNet, on both KTH-IDOL2 and custom dataset. For fair comparison, the conventional CNNs and our network were trained in the same way with the same dataset.

B. *Experimental Environment*

We used an embedded system with quad-core ARMv7 CPU@1.0GHz, and installed the deep learning platform, *caffe2*, on the system [18]. Both *NEON*, a single instruction multiple data accelerator, and *VFP*, a floating point accelerator, are enabled. To test in practical conditions that other processes are simultaneously running on the system, we only use a single-core for algorithm execution.

C. *Feature Space*

At first, we examine the embedding of the proposed CNN to understand what the proposed network learns through metric learning. Fig. 5 depicts t-SNE visualization of embedding vectors of the different scenes in the custom dataset [19]. It shows that the images captured at different places are projected separately from each other on the trained vector space. The digits in the enlarged regions stand for the distances from the scene as shown in Fig. 3. It indicates that the model successfully learns to project neighbor scenes into the feature space while preserving their relative spatial configuration.

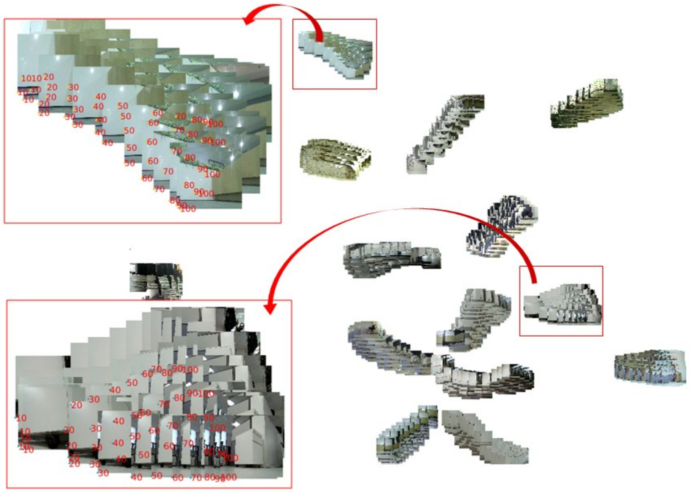

Fig. 5. t-SNE visualization of embedding vectors

D. *Evaluation and Comparison in Accuracy*

1) KTH-IDOL2 dataset

We evaluated the performance of the proposed model in the IDOL Dataset which is the public dataset of place recognition. The dataset consists of 24 image sequences. These sequences were taken under various illumination changes (e.g. sunny, cloudy and night) and different time, and each image sequence has slightly different path.

The query image is considered correctly localized if at least one of top *N* retrieved images is with in threshold around ground truth position. The accuracy of the dataset is plotted for variation of *N*.

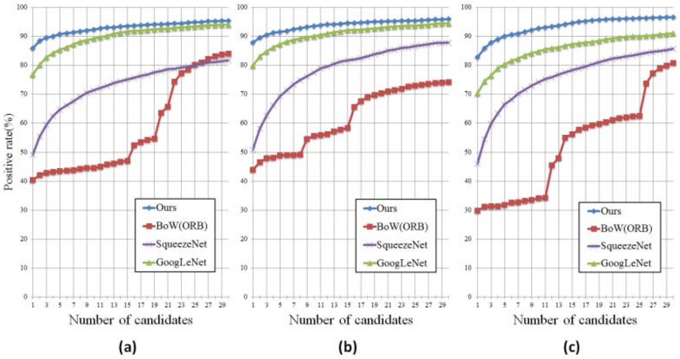

Fig. 6. Comparison of the proposed network versus the conventional approaches on KTH-IDOL2 dataset: (a) sunny, (b) cloudy, and (c) night conditions

Traditional approach of scene recognition is BoW with various descriptors. As mentioned earlier, BoW with ORB descriptor are evaluated on IDOL dataset. Also the other well-known architectures of deep learning area such as GoogLeNet and SqueezeNet were tested on IDOL dataset with same training procedure of proposed algorithm. As shown in Fig. 6, the proposed method has higher performance than traditional approach and the other architecture of deep learning.

2) Custom dataset

Images at 0° for each scene defined as anchor images and the other images which has different angles used as query images. The evaluation on custom dataset was carried out in such a way that input image from query image set feed to the algorithm and then the anchor image with the same distance of the query image would be returned as output.

Also, Using the custom dataset, proposed method compared with traditional algorithm and the other deep learning architecture. As can be seen in Fig. 7a, the proposed method significantly outperforms the BoW algorithm under all conditions. Moreover, it has better performance than the other deep learning architectures. The results on case of ±25° scenes indicate that the other approaches degrade accuracy, but the proposed network maintains reasonable performance. Our network and GoogLeNet has comparable performance which shown in Fig. 7b. But depending on our network has more robustness to view-point variation than GoogLeNet.

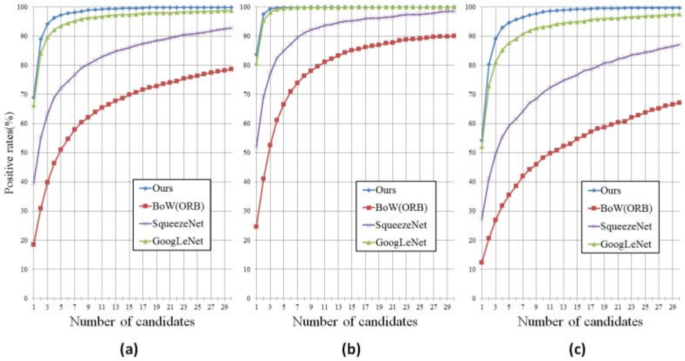

Fig. 7. Comparison of the proposed network versus the conventional approach on the custom dataset: (a) average, (b) ±13°, and (c) ±25° scenes

E. Evaluation of Time Cost

The execution time consumed for each image is described in TABLE. I. Even if ORB algorithm which known as fast local descriptor used for BoW, it spend most of execution time to detection and description for local descriptor. Although the proposed network is slightly slower than BoW, it is 2.1x and 9.1x faster than SqueezeNet and GoogLeNet, respectively. Furthermore, proposed network train faster than the other networks due to it has much fewer parameters than the others.

TABLE I

Comparison of execution time

methods	BoW	SqueezeNet	GoogLeNet	Ours
ms/frame	352	765	3223	355

V. Conclusions

We present a light-weight CNN and metric learning for visual place recognition. Although the proposed CNN has only five convolutional layers with fewer filters, it successfully maps an image to the vector space where its distance corresponds to place similarity. The results on the KTH-IDOL2 and custom datasets demonstrate that the proposed method significantly outperforms the conventional approaches. The proposed technique is expected to be applicable to a variety of electronic products with low computing power.

References

[1] P. Veelaert and W. Bogaerts “Ultrasonic potential field sensor for obstacle avoidance,” in IEEE Transactions on Robotics and Automation, 15(4), 774-779. 1999.

[2] C. H. Lee, Y. C. Su and L. G. Chen “An intelligent depth-based obstacle detection for mobile applications,” in IEEE International Conference on Consumer Electronics-Berlin (ICCE-Berlin), 2012.

[3] I. Ulrich and I. Nourbakhsh “Appearance-based place recognition for topological localization,” in IEEE International Conference on Robotics and Automation (ICRA), 2000.

[4] M. M. Ullah, A. Pronobis, B. Caputo, J. Luo, P. Jensfelt and H. I. Christensen “Towards robust place recognition for robot localization,” in IEEE International Conference on Robotics and Automation (ICRA), 2008.

[5] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford “Visual place recognition: A survey,” in IEEE Transactions on Robotics, 32(1), 1-19. 2016.

[6] D. Gálvez-López, and J. D. “Tardos Bags of binary words for fast place recognition in image sequences,” in IEEE Transactions on Robotics, 28(5), 1188-1197. 2012.

[7] H. Jégou, M. Douze, C. Schmid and P. Pérez “Aggregating local descriptors into a compact image representation,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010.

[8] F. Perronnin, Y. Liu, J. Sánchez and H. Poirier “Large-scale image retrieval with compressed fisher vectors,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010.

[9] H. Bay, T. Tuytelaars, and L. Van Gool. “Surf: Speeded up robust features,” In European Conference on Computer Vision (ECCV), 2006.

[10] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski “ORB: An efficient alternative to SIFT or SURF,” in IEEE international conference on Computer Vision (ICCV), 2011.

[11] N. Dalal and B. Triggs “Histograms of oriented gradients for human detection,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005.

[12] A. Oliva and A. Torralba “Modeling the shape of the scene: A holistic representation of the spatial envelope,” in International journal of computer vision, 42(3), 145-175. 2001.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton “Imagenet classification with deep convolutional neural networks,” in Advances in neural information processing systems (pp. 1097-1105). 2012.

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D.Erhan, and A. Rabinovich “Going deeper with convolutions,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[15] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally and K. Keutzer “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size,” in arXiv:1602.07360, 2016.

[16] F. Schroff, D. Kalenichenko, and J. Philbin. “FaceNet: A unified embedding for face recognition and clustering,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[17] J. Luo, A. Pronobis, B. Caputo, and P. Jensfelt, “The KTH-IDOL2 database,” Technical Report CVAP304, Kungliga Tekniska Hoegskolan, CVAP/CAS, Oct. 2006.

[18] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in Proceedings of the 22nd ACM international conference on Multimedia (pp. 675-678). 2014

[19] L. Van Der Maaten, “Accelerating t-SNE using tree-based algorithms,” in Journal of machine learning research 15(1), 3221-3245, 2014.