Cross-Season Visual Route Classification Using a Domain-Invariant Next-Best-View Planner

Kurauchi Kanya Tanaka Kanji
Robotics Course, Department of Mechanical and System Engineering
School of Engineering, University of Fukui
3-9-1, bunkyo, Fukui, Japan
Email: tnkknj@u-fukui.ac.jp

***Abstract*—This paper addresses the problem of active visual place recognition (VPR) from a novel perspective of long-term autonomy. In our approach, a next-best-view (NBV) planner plans an optimal action-observation-sequence to maximize the expected cost-performance for a visual route classification task. A difficulty arises from the fact that the NBV planner is trained and tested in different domains (times of day, weather conditions, and seasons). Existing NBV methods may be confused and deteriorated by the domain-shifts, and require significant efforts for adapting them to a new domain. We address this issue by a novel deep convolutional neural network (DNN) -based NBV planner that does not require the adaptation. Our main contributions in this paper are summarized as follows: (1) We present a novel domain-invariant NBV planner that is specifically tailored for DNN-based VPR. (2) We formulate the active VPR as a POMDP problem and present a feasible solution to address the inherent intractability. Specifically, the probability distribution vector (PDV) output by the available DNN is used as a domain-invariant observation model without the need to retrain it. (3) We verify efficacy of the proposed approach through challenging cross-season VPR experiments, where it is confirmed that the proposed approach clearly outperforms the previous single-view-based or multi-view-based VPR in terms of VPR accuracy and/or action-observation-cost.**

I. Introduction

Long-term visual place recognition (VPR) is central to mobile robot and intelligent vehicle applications. The task of VPR is to classify egocentric view images into predefined place classes [1]. Long-term VPR is a challenging scenario, in which the VPR system is trained and tested in different domains (e.g., times of day, weather conditions, and seasons) [2]. A key difficulty comes from appearance changes caused by domain shift, such as changes in viewpoints, illumination, weather conditions, and seasons. One of the most successful solutions to address this challenge is to fine-tune a deep convolutional neural network (DNN) as a visual place classifier (VPC) by using action-observation experiences in a past domain as the training data [3]–[5].

Most of the previous works on VPR have focused on a passive setting, in which the robot's action is determined by a predefined control rule, such as a constant speed motion rule. However, such a passive VPR system has several limitations.

Our work has been supported in part by JSPS KAKENHI Grant-in-Aid for Scientific Research (C) 26330297, and (C) 17K00361.

Fig. 1. The robot plans an optimal action-observation-sequence to maximize the expected cost-performance for a visual route classification task. In the figure, the ground-truth route class is c=2. The action set consists of forward (FW) and backward (BW) moves. There is significant difference in appearance between test (blue) and train (red) images due to domain shift.

First, the viewpoints are not necessarily optimized for the VPR task, yielding less informative or non-discriminative images that may lead to suboptimal VPR performance. Second, such a simple control rule may produce an unnecessarily large number of redundant observations, which may result in a significant decrease in the cost performance of VPR.

In this work, we consider an active VPR setting called visual route classification (Fig. 1). The task of active VPR is to classify a view sequence into predefined route classes. To maximize the cost performance of VPR, we pretrain a state-to-action mapping function called next-best-view (NBV) planner in a past domain and use it to select an NBV at each time step during the active VPR task. However, a difficulty comes from the domain shift. Typical planners, such as those based on low-level image features (e.g., color, texture, and shape) [6] or middle-level image features (e.g., objects, landmarks, and GIST) [7], may be confused and deteriorated by domain shifts [8]. Adapting these planners to a new domain requires significant efforts to collect training data and to retrain [9].

To address these issues, we propose a novel domain-invariant NBV planner that does not require the adaptation. Our study was motivated by recent findings in the VPR community that show that the probability distribution vector (PDV) output by the final layer of a DNN provides domain-invariant semantic information for the input scene [10]. Specifically, we were inspired by our recent study on long-term

knowledge distillation (KD) [11], in which the PDVs of DNNs ("teacher") provide abundant information that can be used even for training other DNNs ("student") in a different domain. Our idea in this study is to exploit such PDV information for an observation model to develop a novel domain-invariant NBV planner. Moreover, we formulate the NBV planning task as a partially observable Markov decision process (POMDP) [12] and show that PDVs can actually be used as useful information that realizes a feasible solution.

Our main contributions in this paper are summarized as follows: (1) We address the problem of active VPR from a novel perspective of long-term VPR, in which the effect of domain shift is explicitly addressed by introducing a domain-invariant NBV planner. (2) We formulate the active VPR task as a POMDP problem and provide a feasible solution to address the inherent intractability. Specifically, our domain-invariant planner extracts and exploits useful semantic information from the existing DNN component without the need to retrain it. (3) We verify the efficacy of the proposed approach through challenging cross-season VPR experiments, focusing on outdoor scenes with no distinctive landmark objects, using the publicly available NCLT dataset (Fig. 2). Through the experiments, it was confirmed that the proposed approach clearly outperforms the previous single view-based or multiview-based VPR in terms of VPR accuracy and/or action-observation-cost.

II. Approach

The goal of an NBV planner is to plan an optimal action-observation-sequence that is expected to maximally improve the cost performance of VPR. Our work is based on the following three assumptions. (1) We assume the availability of a DNN-based VPR that is pretrained in a self-supervised manner. Prior to the active VPR task, the robot workspace is partitioned into K disjoint routes, or place classes, by using a place partitioning technique such as that used in [13]. Then, the viewpoint trajectory in the training data is precisely reconstructed by using techniques such as the SfM [14] or the SLAM [15] technique. Then, a DNN is fine-tuned as a K-class VPC by using the route-specific viewpoint-image pairs as the training data. (2) We also assume that, during an active VPR task, the robot is located in one of the routes or places. A robot's action is defined as a move between viewpoints along the route. It should be noted that even though the robot is located in the same route, there remains a non-negligible difference in appearance between the train and the test scenes. (3) Finally, we assume the availability of a scoring function, which takes as input an image sequence produced by an action-observation sequence, applies each image to the DNN, and aggregates the PDV sequence into a final decision in the form of a K-dimensional score vector. In the implementation, we use a simple element-wise multiplication for the aggregation: score[c]=π_t PDV[c].

This section provides a description of the active VPR framework. First, the active VPR task is formulated as a POMDP problem (Section II-A). Then, its main components, namely, the action set (Section II-B), reward function (Section II-C), and observation function (Section II-D), are introduced. Finally, a multiple-hypothesis approach is presented for real-world VPR applications (Section II-E).

A. Active VPR Formulation

Following the definition by Kaelbling [12], we define POMDP as a six-tuple (S,A,T,R,W,O), where S is a finite set of states, A is a finite set of actions, $T : S \times A \times S \to [0,1]$ is the transition function defining the probability of state change caused by an action, $R : S \times A \to \mathbb{R}$ is the reward function that represents the reward granted after having reached the new state with the given action, W is a finite set of observations, and $O : S \times A \times W \to [0,1]$ is the probability distribution of the observations according to the states and the actions.

At each time step, given a current state $s \in S$, the agent receives an observation $o \in W$ with the probability $O(s,o) = Pr(o|s)$. Depending on this observation and the current state, the agent takes an action $a \in A$, which causes a transition to state s' with the probability $T(s,a,s') = Pr(s'|s,a)$. Finally, the agent receives a reward from the reward function $R(s,a)$. The above process repeats until the "terminate" action (see Section II-B) is selected by the planner.

We define the state as a pairing of class c and viewpoint x: $s = (x,c)$. The class c is the hidden class of the place or route. The viewpoint x is represented as a travel distance along the route from the start location on the route (Fig. 1). For a while, we assume that the viewpoint x is measurable. Obviously, in practice, this assumption does not hold in the application of VPR. Therefore, we relax this assumption by introducing a multiple-hypothesis-tracking approach (Section II-E).

B. Action Set

The set of actions involves moving between viewpoints along the class-specific route. In practice, such an action could be implemented by using a vision-based route-following algorithm such as that in [16]. In our experimental system, the action set A consists of seven different possible actions. This includes three forward moves (FWs, i.e., "a1," "a3," and "a5"), three backward moves (BWs, i.e., "a2," "a4," and "a6"), and an additional special action (TN, i.e., "a7") that terminates the active VPR and outputs the most likely place class as the final answer. These actions are defined as follows:

a1 : $x \leftarrow x + L/2$,
a2 : $x \leftarrow x - L/2$,
a3 : $x \leftarrow x + L/4$,
a4 : $x \leftarrow x - L/4$,
a5 : $x \leftarrow x + L/8$,
a6 : $x \leftarrow x - L/8$,
a7 : TN,

where L is the length of the route in terms of the total number of image frames.

The transition function defines the probability of state change given an action. The state transition function can be written as follows: $T(s,a,s') = Pr(s'|s,a)$. In this study, we assumed that a transition causes no motion noise. In practice,

this assumption could be held by introducing a precise visual odometry system [17].

C. Reward Function

The reward function $R(s,a)$ represents the reward for reaching a new state s with a given action a. We define rewards for several different events. When the robot takes an action, e.g., moves forward or backward for a distance $i(L/8)$ $(i \in \{1,2,4\})$ (i.e., "a1"-"a6"), the reward is $i \cdot C_l$, which is proportional to the travel distance.

To ensure that the robot should move within a predefined fixed area $[0, L]$ on the class-specific route, we grant an infinity negative reward -∞ when it attempts to move outside of the area.

When the robot takes the special action "a7," a large positive (C_c) or negative $(-C_f)$ reward is granted depending on whether the answer is correct or not.

Every time the robot observes to take a new image, a fixed predefined reward of C_b is granted.

D. Observation Function

The observation set is represented as Ω. As mentioned previously, we are motivated to use the PDV output by a DNN as a domain-invariant observation model. A PDV is a K-dimensional real-valued vector, and its elements sum up to 1. For the DNN, we use a convolutional neural network with the VGG16 architecture [18] that has been fine-tuned on our K-class VPR task in the training domain. The input images are resized to 256 × 256 before being inputted into the VGG16 network.

We expect that a PDV output by the final layer of a DNN contains a high-dimensional rich semantic information that is invariant against domain shift. Recent findings from the DNN community (e.g., KD [11]) have proven that such a PDV can tell us not only the domain-invariant information about the most likely place class but also how much confidence the DNN has in the prediction. Therefore, a PDV-driven planner is expected to terminate the active VPR task (i.e., the action "a7") as early as possible to avoid unnecessary action-observation costs. Moreover, the PDVs also provide the relative strength of confidence between different place classes. Therefore, a PDV-driven planner is expected to learn K different behaviors, each of which is tailored to one of the K different routes.

E. Multiple Hypothesis Tracking

Now, we relax the unrealistic assumption that the viewpoint x is measurable and consider multiple hypotheses for x. Under the assumption of no-motion noise (Section II-B), this is simply realized by hypothesizing the initial viewpoint. Formally, at the initial time step, a set of H independent POMDP planners are initialized for H different initial viewpoints, and they are run in parallel. At each time step t, the H different actions outputted by the H planners are aggregated into a single action plan, which is the final decision that is executed by the real robot controller. As a result, a single actual observation result is obtained and the observation is used to trigger the next step run of the planners. In this way, the above steps are iterated until the special action "a7" is outputted as the final decision.

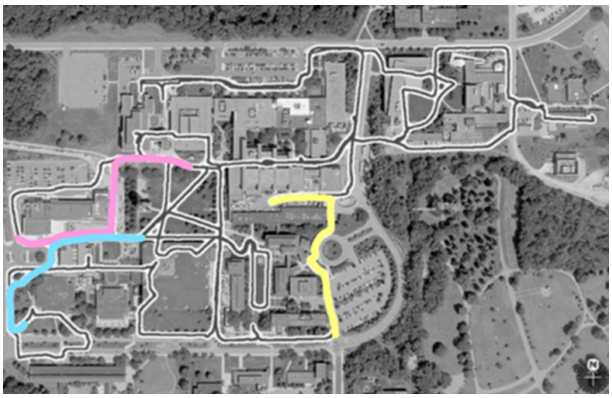

Fig. 2. Illustration of the workspaces. The colored curves, i.e., pink, light blue, and yellow, correspond to workspaces #1, #2, and #3, respectively.

TABLE I
PERFORMANCE RESULTS.

(a) success rate [%]

	WS1	WS2	WS3
single	86.7	64.5	77.1
random	84.5	64.5	77.2
proposed	87.8	66.7	82.8

(b) average number of observations

	WS1	WS2	WS3
single	1.00	1.00	1.00
random	2.57	2.67	2.66
proposed	2.14	1.99	2.51

(c) average travel distance

	WS1	WS2	WS3
single	0.0	0.0	0.0
random	79.2	89.3	78.1
proposed	34.3	34.4	40.0

For the action aggregation, we consider a voting scheme in which every POMDP planner votes for the action plan that it selected. Subsequently, the action plan that received the highest votes is outputted as the final decision. When multiple action plans receive the same number of votes, the plan with the lowest cost (defined in Section II-C) is chosen as the final decision. When there is a combination of FW and BW actions receiving the same lowest cost (e.g., "a3" + "a4"), the FW one is chosen.

III. EXPERIMENTS

We verified the efficacy of the proposed method through active VPR experiments. For the dataset, we considered the NCLT dataset [19]. This dataset is one of the largest datasets for long-term autonomy and contains sequences of image data from the robot's onboard imagery collected during a multi-session exploration across several different seasons. We used three datasets in NCLT: "2012/5/26" for training the VGG16 classifier, "2012/5/11" for training a POMDP planner, and

"2012/3/31" for testing the active VPR system, respectively. We considered three independent workspaces (WSs), as shown in Fig. 2, using the outdoor parts of the datasets that contain no distinctive landmark objects. All the three WSs were almost of the same length in terms of travel distance, and each WS was divided into four equal-length subsequences, each of which defines a route or a place class. In this way, we considered three independent sets of four-class classification problems. The image sequence in each route was downsampled to have the same number of L =225 image frames; thus, all the classes summed up to 900 frames in total. Considering the dependency of the initial viewpoints, we executed 45 independent tests for 45 different initial viewpoints that were spaced by 5 image frames on each train route. The total performance was measured as the average of these 45 $\times$ 4 = 180 independent tests. For the POMDP solver, we employed SARSOP [20] and a script that automatically produces POMDPX data before using them as the input to the solver. For the NCLT dataset, the original image size is 1232 $\times$ 1616, and we resized every image before it was inputted to the VGG16 classifier. The parameters were set C_r=100, C_f=100, C_b=5, and $C_l = 15$.

For performance comparison, we considered three comparison methods. The first method (i.e., "single") is the conventional passive single-view VPR, which uses just a single image at the initial viewpoint as the sole query input. The second method (i.e., "random") is a multi-view VPR, which randomly selects the next action among the several possible actions, namely, "a1," $\cdots$, "a7," defined in Section II-B. The last one (i.e., "proposed") is the proposed multi-view VPR method, which is based on the pretrained POMDP model and the PDV-based observation model.

Table I shows the performance results. It is evident that the proposed method clearly outperformed the other two methods in terms of VPR accuracy for all the WSs. The increase in performance was significant, ranging from 2.2 to 5.6 percentage points. Especially, the performance gain was highest in the case of WS #3.

Table I summarizes the costs in terms of average number of observations and motion costs. Evidently, the "single" method produced the lowest average cost. Comparing the "random" and "proposed" methods, we can see that the proposed method resulted in a significant cost reduction in terms of both the number of observations and the motion costs.

IV. Conclusions

We have presented a DNN-based NBV planner that incorporates the output PDVs as a domain-invariant observation model. An algorithm is developed for planning an optimal action-observation-sequence to maximize the expected cost-performance of a visual route classification task. We also formulate the active VPR as a POMDP problem and present a feasible solution to address the inherent intractability. Our algorithm outperformed the existing results in a challenging dataset that does not contain any distinctive landmark object. Although this paper was focused on solving a cross-season visual route classification task, this idea could be extended to any cross-domain DNN-based VPR in general.

References

[1] S. M. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. D. Cox, P. I. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE Trans. Robotics*, vol. 32, no. 1, pp. 1–19, 2016.

[2] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," vol. 32, no. 1. IEEE, 2015, pp. 1–19.

[3] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2938–2946.

[4] A. Kendall and R. Cipolla, "Geometric loss functions for camera pose regression with deep learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5974–5983.

[5] E. Muller-Budack, K. Pustu-Iren, and R. Ewerth, "Geolocation estimation of photos using a hierarchical model and scene classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 563–579.

[6] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision, ICCV*, 2011, pp. 2564–2571.

[7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, 2005, pp. 886–893.

[8] R. Gomez-Ojeda, M. Lopez-Antequera, N. Petkov, and J. Gonzalez-Jimenez, "Training a convolutional neural network for appearance-invariant place recognition," *arXiv preprint arXiv:1505.07428*, 2015.

[9] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *European conference on computer vision*. Springer, 2016, pp. 443–450.

[10] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[11] T. Hiroki and K. Tanaka, "Long-term knowledge distillation of visual place classifiers," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 541–546.

[12] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[13] O. Erkent and I. Bozma, "Place representation in topological maps based on bubble space," in *2012 IEEE International Conference on Robotics and Automation*, May 2012, pp. 3497–3502.

[14] D. Novotny, N. Ravi, B. Graham, N. Neverova, and A. Vedaldi, "C3dpo: Canonical 3d pose networks for non-rigid structure from motion," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7688–7697.

[15] A. Walcott-Bryant, M. Kaess, H. Johannsson, and J. J. Leonard, "Dynamic pose graph slam: Long-term mapping in low dynamic environments," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 1871–1878.

[16] A. M. Zhang and L. Kleeman, "Robust appearance based visual route following for navigation in large-scale outdoor environments," *The International Journal of Robotics Research*, vol. 28, no. 3, pp. 331–356, 2009.

[17] J. Straub, S. Hilsenbeck, G. Schroth, R. Huitl, A. Möller, and E. Steinbach, "Fast relocalization for visual odometry using binary features," in *2013 IEEE International Conference on Image Processing*, 2013, pp. 2548–2552.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[19] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of michigan north campus long-term vision and lidar dataset," *The International Journal of Robotics Research*, pp. 1023–1035, 2015.

[20] H. Kurniawati, D. Hsu, and W. S. Lee, "SARSOP: efficient point-based POMDP planning by approximating optimally reachable belief spaces," in *Robotics: Science and Systems IV, Eidgenössische Technische Hochschule Zürich, Zurich, Switzerland, June 25-28, 2008*, 2008.