Deep SIMBAD:
Active Landmark-based Self-localization Using Similarity-based Scene Descriptor

Tanaka Kanji

***Abstract*— Landmark-based robot self-localization has attracted recent research interest as an efficient maintenance-free approach to visual place recognition (VPR) across domains (e.g., times of the day, weathers, seasons). However, landmark-based self-localization can be an ill-posed problem for a passive observer (e.g., manual robot control), as many viewpoints may not provide effective landmark view. Here, we consider active self-localization task by an active observer, and present a novel reinforcement-learning (RL) -based next-best-view (NBV) planner. Our contributions are summarized as follows. (1) SIMBAD-based VPR: We present a landmark ranking -based compact scene descriptor by introducing a deep-learning extension of similarity-based pattern recognition (SIMBAD). (2) VPR-to-NBV knowledge transfer: We tackle the challenge of RL under uncertainty (i.e., active self-localization) by transferring the VPR's state recognition ability to NBV. (3) NNQL-based NBV: We view the available VPR as the experience database by adapting a nearest-neighbor -based approximation of Q-learning (NNQL). The result is an extremely compact data structure that compresses both the VPR and NBV modules into a single incremental inverted index. Experiments using public NCLT dataset validate the effectiveness of the proposed approach.**

Fig. 1. Motivation. Unlike existing approaches that maintain the model of entire environment (left panel), we aim to maintain only a small fraction (i.e., the landmark regions) of the robot workspace (right panel). As a key advantage, its per-domain cost for map maintenance (change detection, map updating) is inherently low. On the other side, self-localization with such spatially sparse landmarks can be an ill-posed problem for a passive observer, as many viewpoints may not provide effective landmark view. Therefore, we consider an active observer and present an active landmark-based self-localization framework.

I. Introduction

Landmark-based robot self-localization has attracted recent research interest as an efficient maintenance-free approach to visual place recognition (VPR) across domains (e.g., times of the day, weathers, seasons). In long-term cross-domain navigation [1]–[3], a robot vision has to be able to recognize where it is (i.e., self-localization) and what the main objects [4] (i.e., landmarks) in the scene are. Such landmark-based self-localization problem has two unique challenges. (1) Landmark selection: In offline training stage, the robot must learn the main landmarks that represent the robot workspace, in either self-supervised or unsupervised manner [5]. (2) Next-Best-View (NBV): In online self-localization stage, the robot must determine NBVs to re-identify such spatially sparse landmarks as many as possible [6]. This paper focuses on this NBV problem.

In landmark-based self-localization, the only available measurement (Fig. 1) in each map/live scene x is which landmark $p_i (i \in [1,r])$ is observed at which signal strength $d(x,p_i)$ (i.e., landmark ID + intensity). This is in contrast to many existing self-localization frameworks that assume the availability of vectorial features for each image. We formulate landmark-based self-localization as SIMBAD (similarity-based pattern recognition) [7] and further present its deep-learning extension. In pattern recognition, SIMBAD is a method to describe an object as its dissimilarities from r prototypes (e.g., $r = 500$), which is particularly effective when traditional vectorial description of objects are difficult to obtain or inefficient for learning purposes. Our motivations behind using SIMBAD are two folds. First, the landmark selection problem is analogous to and can be informed by the well-studied problem of prototype selection or prototype optimization [8]. Second, many recent deep learning techniques are good at measuring of (dis)similarity between an object and prototypes (e.g., training images, reference images) [9], rather than at describing an object. However, deep learning extension of SIMBAD techniques have not been sufficiently explored yet, and is a main focus of our study.

Thus far, most of existing self-localization techniques suppose a passive observer (e.g., manual robot control), and do not take into account the issue of viewpoint planning or controlling the observer. However, landmark-based self-localization can be an ill-posed problem for a passive observer, as many viewpoints may not provide any effective landmark view. Therefore, we aim to develop an active observer that can adapt its viewpoint trajectory, avoiding non-salient scenes that provide no effective landmark view, or moving efficiently towards places which are most informative, in the sense of reducing the sensing/computation costs. This is most closely related to the NBV problem studied in machine vision literature [10]. However, in our cross-domain scenario, a difficulty arises from the fact that the

Our work has been supported in part by JSPS KAKENHI Grant-in-Aid for Scientific Research (C) 17K00361 and 20K12008.

The authors are with Graduate School of Engineering, University of Fukui, Japan. tnkknj@u-fukui.ac.jp

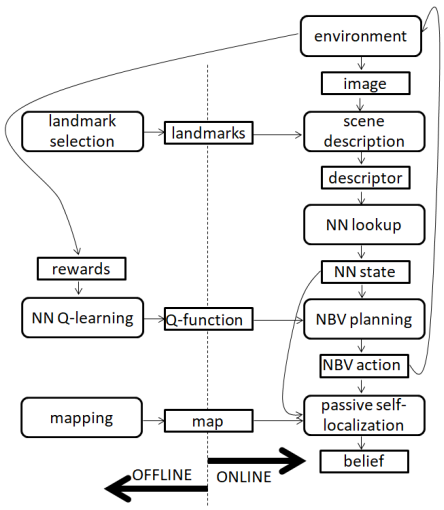

Fig. 2. The system architecture. The online processing consists of three main stages: scene description, passive self-localization, and NBV planning. The offline processing consists of landmark selection, mapping, and NN Q-learning.

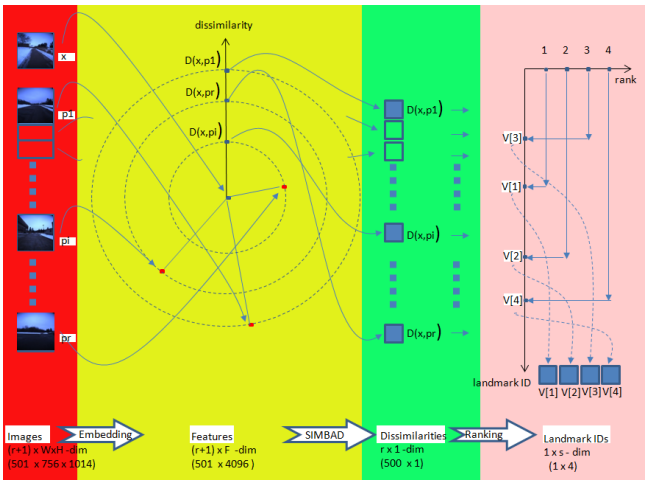

Fig. 3. Scene description module. First, the input query/landmark images are translated into vectorial features. Then, the query feature is described by its dissimilarities from the landmark features. Then, the dissimilarity values are ranked to obtain a ranked list of top-s most similar landmarks. The result is the proposed landmark ranking -based scene descriptor.

NBV planner is trained and tested in different domains. The mainenance cost for retraining such an NBV planner that does not take into account domain shifts would be significant in cross-domain scenarios, which we aim to address in the current work.

In this study, we explore a reinforcement-learning (RL) -based NBV planner from the novel perspective of SIMBAD. Our primary contributions are three folds. (1) SIMBAD-based VPR: First, we present a landmark ranking -based compact scene descriptor by introducing a deep-learning extension of similarity-based pattern recognition (SIMBAD). Our strategy is to describe each map/live scene x with dissimilarities $\{d(x,p_i)\}$ from the prototypes (i.e., landmarks) $\{p_i\}_{i=1}^{r}$ and further compress it into a lengh s ($\ll r$) list of IDs of most similar landmarks v_1, $\cdots$, v_s ($d(x,p_{v_1})\leq\cdots\leq d(x,p_{v_s})$), which can then be efficiently indexed by inverted index. (2) VPR-to-NBV knowledge transfer: Second, we tackle the challenge of RL under uncertainty (i.e., active self-localization) by transferring the VPR's state recognition ability to NBV. Our scheme is based on recently-developed method of reciprocal rank feature (RRF) [11], which is effective for transfer learning [12] and information retrieval [13]. (3) NNQL-based NBV: Third, we view the available VPR as the experience database by adapting a nearest-neighbor -based approximation of Q-learning (NNQL) [14]. The result is an extremely compact data structure that compresses VPR and NBV into a single incremental inverted index. Experiments using public NCLT dataset validate effectiveness of the proposed approach.

II. Approach

Figure 2 depicts the architecture of the active self-localization system. The system consists of three main modules: scene description, (passive) self-localization, and NBV planning. The scene description module is responsible for describing an input map/live image as its dissimilarities from the pre-defined landmarks. The self-localization module is responsible for incorporating each perceptual/action measurement into the belief of the robot viewpoint. The NBV planning module is responsible for determining the NBV action. In addition, there are three additional modules, landmark selection, mapping, and nearest-neighbor Q-learning, which run offline to learn the prior knowledge ("landmarks", "Q-function", "map") needed for the above three main modules. These individual modules are detailed in the following subsections.

A. Landmark Model

The proposed landmark model is explained as follows. Let $R=\{p_1,p_2,\cdots,p_r\}$ be a set of r pre-defined prototypes (i.e., landmarks). Given a dissimilarity measure d, which measures dissimilarity between a prototype p and an input object x, one may consider a new description based on the proximities to the set R, as $D(x,R)=[d(x,p_1),\ d(x,p_2),\ \cdots,\ d(x,p_r)]$. Here, $D(x,R)$ is a data dependent mapping $D(\cdot,R)$: $X\rightarrow\mathbb{R}^r$ from the representation X to dissimilarity space, defined by the set R. This is a vector space, in which each dimension corresponds to a dissimilarity $D(\cdot,p_i)$ to a prototype from R. A vector $D(\cdot,p_i)$ of dissimilarities to the prototype p_i can be interpreted as a feature. It is noteworthy that even when no landmark is present in the input image, the proposed model still can describe the image as dissimilarities to the landmarks.

In our view, the advantage of this representation is that any methods including recent deep learning techniques in dissimilarity spaces can now be used. For example, d could be (1) a pairwise comparison network (i.e., deep Siamese) that predicts dissimilarity $d(x,p)$ between p and x, (2) a deep anomaly detection network (e.g., deep autoencoder) that predicts the deviation $d(x|p)$ of x from learned prototypes p, (3) an object proposal network (e.g., feature pyramid

network) that predicts object bounding-boxes and class label $d(p|x)$ (e.g., [15]), (4) L2 distance $|f(x)-f(p)|$ of deep feature $f(\cdot)$ between p and x, as well as (5) any methods for dissimilarities $d(p,x)$ including those for non-visual modality (e.g., natural language). In the current paper, the experimental system is based on the above (4) using NetVLAD [16] as the feature extraction network $f(\cdot)$. We believe that this is one effective way to exploit the discriminative power of a deep neural network within SIMBAD.

B. Landmark Selection

The landmark selection module aims to select a set of landmarks that represent the robot workspace. Landmarks should be selected so that they are dissimilar to each other. This is because, if the dissimilarity $d(p_i,p_j)$ is small for a prototype pair, $d(x,p_i) \simeq d(x,p_j)$ for other object x meaning that either of p_i or p_j is a redundant prototype. Intuitively, an ideal clustering algorithm may find good landmarks as cluster representatives. However, this clustering is NP hard. In practice, heuristics such as k-means algorithms are often used as an alternative. Moreover, the issue of landmark utility complicates the landmark selection task. That is, landmark visibility and other observation conditions (e.g., occlusions, field-of-views) should be taken into consideration to find landmarks that are actually useful. For such landmark selection, no analytical solution but only heuristic approximate solution exists [17].

In our study, we do not focus on the landmark selection method but we present a simple effective method. Our solution is a three-step procedure. (1) First, high-dimensional vectorial features $X=\{x\}$ (i.e., NetVLAD) are extracted from individual map images. (2) Then, each map image $x \in X$ is scored by the dissimilarity $\min_{x' \in X \setminus \{x\}} |x-x'|$ from its nearest-neighbor over the other features. (3) Then, all the map images are sorted in descending order of the scores and top-r images ($r=500$) are selected as the prototype landmarks. This approach has two merits. First, the selected landmarks can be dissimilar to each other. In addition, the dissimilarity features $D(\cdot,R)$ are expected to become accurate when the robot's viewpoint comes close to the landmark objects.

C. Self-localization

The (offline) mapping module extracts a 4,096 dim NetVLAD image feature from each map image x and then translates it to an r-dim dissimilarity descriptor $D(x,\cdot)$. Then, it sorts the r elements of the descriptor in ascending order, and returns an ordered list of top-s ranked prototype IDs. This returned list is our proposed s-dim (e.g., s=4) SIMBAD descriptor (Fig. 3). Such a descriptor can be directly indexed by an inverted index, in which each pairing of image ID and the rank w.r.t. the ordered list is indexed by its prototype ID.

The (online) self-localization module also translates an input query image x to the dissimilarity descriptor. Then, the inverted index is looked up by using each prototype ID in the ordered list as query. As a result, a short list of map images that have common prototype ID are obtained. Then, relevance of a map image is evaluated as the inner product $\langle v_{RRF}(q), v_{RRF}(m) \rangle$ of reciprocal rank features $v_{RRF}(\cdot)$ between the query image q and each map image m in the short list. An RRF is an r-dim s-hot vector whose i-th element is the reciprocal $v[i]^{-1}$ of the rank value $v[i]$ of i-th landmark if $v[i] \geq s^{-1}$ or 0 otherwise. See [11] for details.

Active self-localization is a multi-view self-localization task. The Bayes filter is one of most widely used framework for such multi-view self-localization. In particular, the particle filter is a computationally efficient implementation of the Bayes filter that can deal with multi-modal belief distribution [18]. In the particle filter -based inference system, the state vector is chosen as $x_k=(l,\theta)$, where l and θ are the location and orientation IDs of the robot viewpoint. The inference system is initialized at the robot's start viewpoint. In the prediction stage, the particles are propagated through the dynamic model using the robot's latest action. In the update stage, the likelihood models for the landmark observation is approximated by the reciprocal rank vector and the weight of each k-th particle is updated: $w_k \leftarrow w_k + f[x_k]$, where $f[x_k]$ is the reciprocal rank feature's element that corresponds to the hypthesized viewpoint x_k. Other details follows the particle filter -based self-localization framework [18].

D. NBV Planning

The NBV planning task is formulated as an RL problem, in which a learning agent interacts with a stochastic environment. The interaction is modeled as a discrete-time discounted Markov decision process (MDP). A discounted MDP is a quintuple (S,A,P,R,γ), where S and A are the set of states and actions, P is the state transition distribution, R is the reward function, and $\gamma \in (0,1)$ is a discount factor. We denote by $P(\cdot|s,a)$ and $r(s,a)$ the probability distribution over the next state and the immediate reward of taking action a at state s, respectively.

A stationary Markov policy $\pi(\cdot|s)$ is the distribution over the control actions given the current state s. It is deterministic if this distribution concentrates over a single action. The action-value functions of a policy π, denoted by $Q: S \times A \rightarrow \mathbb{R}$, is defined as the expected sum of discounted rewards that are encountered when the policy π is executed. Given an MDP, the goal is to find a policy that attains the best possible values, $Q^*(s,a) = \sup_\pi Q^\pi(s,a), \forall (s,a) \in S \times A$. The optimal action-value function Q^* is the unique fixed-point of the Bellman optimality operator J defined as $(JQ)(x,a) = r(x,a) + \gamma \sum_{y \in S} P(y|s,a) \max_{b \in A} Q(y,b), \forall (x,a) \in S \times A$. It is important to note that J is a contraction with factor γ, for any pair of action-value functions Q and Q', we have $|JQ - JQ'| \leq \gamma |Q - Q'|$.

A key issue is how to implement the "Q-function" $Q(\cdot,\cdot)$ as a computationally tractable function. A naive implementation of the function $Q(\cdot,\cdot)$ is to employ a two-dimensional table, which is indexed by state-action-pair (s,a) and whose element is Q-value [19]. However, this simple implementation has several limitations. Especially, it requires a large amount of memory space, in proportion to the number and dimensionality of the state vectors, which is intractable in many

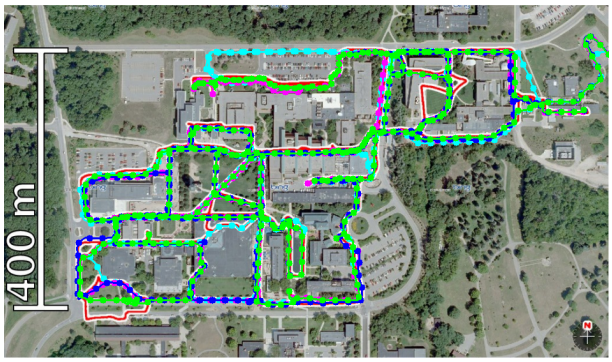

Fig. 4. Experimental environments. The trajectories of the four datasets, "2012/1/22", "2012/3/31", "2012/8/4", and "2012/11/17", used in our experiments are visualized in green, purple, blue, and light-blue curves, respectively.

TABLE I

ANR PERFORMANCE.

Top-(10%) ANR [%]

(Test, Train)	(SP, WI)	(SU, SP)	(AU, SU)	(WI, AU)
Proposed	0.147	0.140	1.688	1.006
Baseline	0.642	0.840	1.746	2.711

Top-(20%) ANR [%]

(Test, Train)	(SP, WI)	(SU, SP)	(AU, SU)	(WI, AU)
Proposed	0.445	1.030	4.140	1.923
Baseline	1.459	1.439	6.058	3.974

Top-(30%) ANR [%]

(Test, Train)	(SP, WI)	(SU, SP)	(AU, SU)	(WI, AU)
Proposed	0.818	2.140	5.913	3.196
Baseline	2.693	2.742	9.586	5.780

Top-(40%) ANR [%]

(Test, Train)	(SP, WI)	(SU, SP)	(AU, SU)	(WI, AU)
Proposed	1.576	3.580	8.311	4.537
Baseline	4.226	4.834	12.374	7.826

applications including ours. An alternative approach would be a deep neural network based approximation of the Q-table (e.g., DQN [20]), which has been employed by recent works including ours (e.g., [6]). However, this requires to maintain the DQN with large amount of space/time overhead. In the current paper, we present a new highly efficient nearest-neighbor based approximation of the Q-learning (NNQL) that reuses the existing inverted index (Subsection II-C) to approximate $Q(\cdot,\cdot)$ with a small additional space cost.

Our approach is inspired by the recently-developed nearest-neighbor approximation of the Q-function [14]. Specifically, the inverted index which is originally developed for the VPR purpose (Subsection II-C) is viewed as a compressed representation of visual experiences in the training domain. Recall that the inverted index aims to index map image ID by prototype ID. We now introduce a supplementary 2D table that aims to index action-specific Q-values $Q(s,\cdot)$ by map image ID. We then approximate the Q-function by the NNQL [14]. A key difference of NNQL [14] from the standard Q-learning is that the Q-value of an input state-action-pair (s,a) is approximated by a collection of Q-values that are associated with its nearest neighbors $N(s,a)$:

$$Q(s,a) = \frac{1}{|N(s,a)|} \sum_{(s',a') \in N(s,a)} Q(s',a'). \qquad (1)$$

Furthermore, we employ the supplementary table to store the action-specific Q-values $Q(s,\cdot)$. Thus, the Q-function is approximated by:

$$Q(s,a) = \arg\max_{a} \frac{1}{|N(s|a)|} \sum_{(s',a) \in N(s|a)} Q(s',a), \qquad (2)$$

where $N(s|a)$ is the nearest neighbors of (s,a) conditioned on a given action a. Such an action-specific NNQL also can be viewed as an instance of RL. See details of NNQL for [14].

III. EXPERIEMENTS

We evaluated the effectiveness of the proposed algorithm via active cross-domain landmark-based self-localization in a cross-season scenario. We used the publicly available NCLT dataset [21]. The NCLT dataset is a large-scale, long-term autonomy dataset available for robotics research that was collected at the University of Michigan's North Campus by a Segway vehicle robotic platform. The data we used includes view image sequences along vehicle trajectories acquired by the front-facing camera of the Ladybug3, as well as the ground-truth GPS viewpoint information. Both indoor and outdoor change objects such as cars, pedestrians, construction machines, posters, and furniture are present during seamless indoor and outdoor navigation by the Segway robot. In experiments, we used four different datasets, "2012/1/22 (WI)", "2012/3/31 (SP)", "2012/8/4 (SU)", and "2012/11/17 (AU)", which respectively consist of 26,208, 26,364, 24,138, and 26,923 images (Fig. 4). They are used to create four different landmark-training-test triplets: WI-SP-SU, SP-SU-AU, SU-AU-WI, and AU-WI-SP. The number of landmarks is set 500 for every triplets, which is consistent with the settings in our previous studies that use the same dataset (e.g., [22]). For NNQL, the learning rate $\alpha = 0.1$, the discount rate $\gamma = 0.9$, and the number of nearest neighbors $|N(s|a)| = 4$. A positive reward of 100 is provided when the belief value of the ground-truth viewpoint is top-10% ranked, while the default Q-value for a state-action-pair is set 0.0001. The number of steps per episode was 10. The number of training spisodes was 10,000.

The VPR performance at each viewpoint in the above context is measured by averaged normalized rank (ANR). In ANR, a subjective VPR is modeled as a ranking function, which takes an input query image and assigns a rank value to each map image. Then, the rank values for the ground-truth map images are computed, normalized by the rank list length, and averaged over all the test queries, which yields ANR. In our experiments, ANR is computed at top-10, 20, 30, and 40 [%] in a list of query images that is sorted in the ascending order of the normalized rank values. As a performance index, ANR can measure how stable the ranking is in the presence of uncertainty in the ranked images. In other words, a VPR system with better ANR performance can be considered to have higher retrieval robustness [23].

Figure 5 shows examples of landmark based scene de-

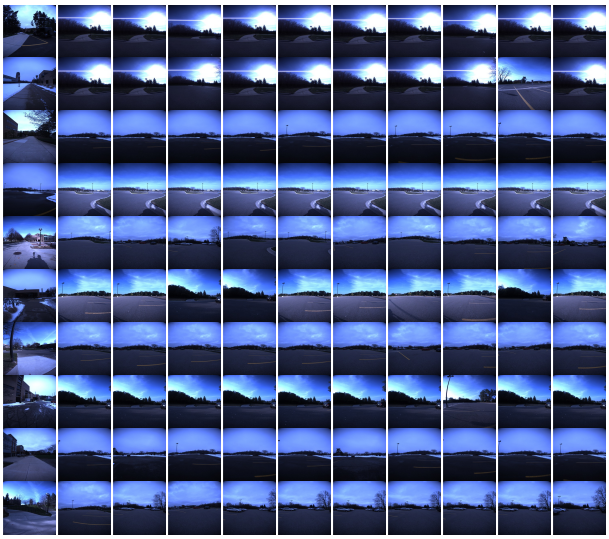

Fig. 5. Examples of similarity-based scene description. 1st column: input image. $(i+1)$-th column $(i \geq 1)$: i-th most similar landmark.

scriptions. As shown, each input image is explained by spatially distant but very similar landmark images, owing to the descriptive power of the deep feature extraction network.

Our proposed method ("Proposed") is compared against the baseline method ("Baseline"). For both methods, the action set A is defined as a size 10 set consisting of 10 different forward moves $\{1, 2, \cdots, 10\}$ [m], along the vehicle's viewpoint trajectories that are defined in the NCLT dataset. The baseline is the algorithm that randomly chooses the action. Table I shows the ANR performance for the both methods. It is clear that the proposed approach significantly outperforms the baseline approach considered here. It is noteworthy that a non-trivial good policy was learned with a model-free RL using the proposed approach despite the fact that the landmark ranking -based VPR and NNQL-based NBV are compressed into a single incremental inverted index.

IV. Conclusions

In this paper, a novel framework for active landmark-based self-localization using SIMBAD was proposed. Unlike previous approaches to active self-localization, we assumed the main landmark objects as the only available model of the robot's workspace ("map"), and proposed to describe each map/live scene as dissimilarities to these landmarks. Based on this idea, a novel reciprocal rank feature -based knowledge transfer from VPR to NBV was introduced to address the challenge of RL under uncertainty. Further, an extremely compact data structure that compresses VPR and NBV into a single incremental inverted index was presented. The proposed framework has been experimentally verified using the publicly available NCLT dataset.

References

[1] M. J. Milford and G. F. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 1643–1649.

[2] W. Churchill and P. Newman, "Experience-based navigation for long-term localisation," *The International Journal of Robotics Research*, vol. 32, no. 14, pp. 1645–1661, 2013.

[3] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Fusion and binarization of cnn features for robust topological localization across seasons," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 4656–4663.

[4] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *Computer Vision, IEEE International Conference on*, vol. 2. IEEE Computer Society, 2003, pp. 273–273.

[5] K. Tanaka, "Mining minimal map-segments for visual place classifiers," *arXiv preprint arXiv:1909.09594*, 2019.

[6] ——, "Domain-invariant nbv planner for active cross-domain self-localization," *arXiv preprint arXiv:2102.11530*, 2021.

[7] A. Feragen, M. Pelillo, and M. Loog, *Similarity-Based Pattern Recognition: Third International Workshop, SIMBAD 2015, Copenhagen, Denmark, October 12-14, 2015. Proceedings*. Springer, 2015, vol. 9370.

[8] K. S. Gurumoorthy, P. Jawanpuria, and B. Mishra, "Spot: A framework for selection of prototypes using optimal transport," *arXiv preprint arXiv:2103.10159*, 2021.

[9] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "Siamrpn++: Evolution of siamese visual tracking with very deep networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4282–4291.

[10] M. Mendoza, J. I. Vasquez-Gomez, H. Taud, L. E. Sucar, and C. Reta, "Supervised learning of the next-best-view for 3d object reconstruction," *Pattern Recognition Letters*, vol. 133, pp. 224–231, 2020.

[11] K. Takeda and K. Tanaka, "Dark reciprocal-rank: Teacher-to-student knowledge transfer from self-localization model to graph-convolutional neural network," in *2021 International Conference on Robotics and Automation (ICRA)*, 2021, pp. 4348–4355.

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

[13] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli, "Multimodal fusion for multimedia analysis: a survey," *Multimedia systems*, vol. 16, no. 6, pp. 345–379, 2010.

[14] D. Shah and Q. Xie, "Q-learning with nearest neighbors," *arXiv preprint arXiv:1802.03900*, 2018.

[15] S. Hanada and K. Tanaka, "Partslam: Unsupervised part-based scene modeling for fast succinct map matching," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1582–1588.

[16] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," pp. 5297–5307, 2016.

[17] M. Bürki, C. Cadena, I. Gilitschenski, R. Siegwart, and J. Nieto, "Appearance-based landmark selection for visual localization," *Journal of Field Robotics*, pp. 4137–4143, 2016.

[18] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *1999 International Conference on Robotics and Automation (ICRA)*, vol. 2, 1999, pp. 1322–1328.

[19] R. S. Sutton, A. G. Barto, *et al.*, *Introduction to reinforcement learning*. MIT press Cambridge, 1998, vol. 135.

[20] J. Fan, Z. Wang, Y. Xie, and Z. Yang, "A theoretical analysis of deep q-learning," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 486–489.

[21] 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*, vol. 35, no. 9, pp. 1023–1035, 2016.

[22] T. Hiroki and K. Tanaka, "Long-term knowledge distillation of visual place classifiers," in *2019 22st International Conference on Intelligent Transportation Systems (ITSC)*, 2019.

[23] P. Hiremath and J. Pujari, "Content based image retrieval using color, texture and shape features," in *15th International Conference on Advanced Computing and Communications (ADCOM 2007)*. IEEE, 2007, pp. 780–784.