Sequence-based Multimodal Apprenticeship Learning For Robot Perception and Decision Making

Fei Han[1], Xue Yang[1], Yu Zhang[2], and Hao Zhang[1]

***Abstract*— Apprenticeship learning has recently attracted a wide attention due to its capability of allowing robots to learn physical tasks directly from demonstrations provided by human experts. Most previous techniques assumed that the state space is known a priori or employed simple state representations that usually suffer from perceptual aliasing. Different from previous research, we propose a novel approach named Sequence-based Multimodal Apprenticeship Learning (SMAL), which is capable to simultaneously fusing temporal information and multimodal data, and to integrate robot perception with decision making. To evaluate the SMAL approach, experiments are performed using both simulations and real-world robots in the challenging search and rescue scenarios. The empirical study has validated that our SMAL approach can effectively learn plans for robots to make decisions using sequence of multimodal observations. Experimental results have also showed that SMAL outperforms the baseline methods using individual images.**

I. Introduction

Apprenticeship learning (AL) has become an active research area in robotics over the past years, which enables a robot to learn physical tasks from expert demonstrations, without the requirement to engineer accurate task execution models. AL has been widely applied in a variety of practical applications, including object grasping [1], robotic assembly [2], helicopter control [3], navigation and obstacle avoidance [4], among others [5], [6], [7], [8]. AL methods automatically learn a mapping from world states to robot actions based on optimal or near optimal demonstrations. These methods can also quantify the trade-off among task constraints, which can be difficult or even impossible for manual task modeling.

Given the advantage of AL, however, most previous techniques focused only on either perception or decision making without good integration between these two key components [9], [6]. It limits the capability of AL methods to address real-world problems when a robot needs to make decisions based upon online observations, especially in cases when the perception data consist of multiple modalities obtained from a variety of equipped sensors. To address this issue, several methods were proposed to integrate perception and planning within the same AL formulation. A promising direction is to utilize images perceived by robot's onboard cameras as a representation of the current state, and then use supervised learning or reinforcement learning for decision making [10], [11]. However, state representation and recognition based on individual images often suffer from the issue of perceptual aliasing (i.e., multiple distinct states of the world give rise to the same percept), due to their incapability to incorporate temporal information or multimodal observations. Unreliable perception will result in wrong planning and decision making, and possibly fail the tasks.

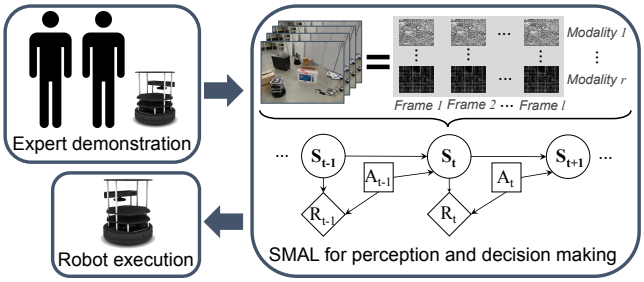


Fig. 1. Overview of the proposed SMAL method to achieve robot apprenticeship learning. Our SMAL approach is able to simultaneously integrating temporal information and multimodal observations to generate a multimodal sequence-based representation of world states. In addition, SMAL integrates perception and decision making for robots to learn physical tasks directly from sequences of multimodal observations. Our SMAL approach has been validated in search and rescue applications to find victims.

In this paper, we develop a novel *Sequence-based Multimodal Apprenticeship Learning* (SMAL) method to integrate spatio-temporal multimodal perception and decision making in the AL scenario. Instead of using individual images, we propose to represent a world state directly as a sequence of multimodal observations. Then, state recognition is achieved by our new multimodal sequence-based scene matching that integrates multimodal features obtained from each individual frame and fuses temporal information contained in the whole sequence. Then, we introduce a framework to integrate the sequence-based multimodal state perception with a reinforcement learning method to achieve apprenticeship learning. We evaluate the proposed SMAL approach in challenging search and rescue applications, as we believe our new AL paradigm has potential to address several critical tasks such as victim search and path planning.

The main contributions of this paper are twofold. First, we propose a novel representation of world states, and introduce an approach to recognize the states by simultaneously fusing temporal information and multimodal data. Second, we develop the SMAL approach that integrates multisensory robot perception and decision making to learn tasks from human experts in challenging environments with perceptual aliasing (e.g., disaster scenarios).

[1]Fei Han, Xue Yang and Hao Zhang are with the Department of Computer Science, Colorado School of Mines, 1500 Illinois Street, Golden, CO 80401, USA fhan@mines.edu, edyxueyx@gmail.com, hzhang@mines.edu

[2]Yu Zhang with the Department of Computer Science and Engineering, Arizona State University, 699 S Mill Ave, Tempe, AZ 85281, USA yzhan442@asu.edu

The rest of this paper is organized as follows. We describe related publications in Section II. In Section III, we propose the sequence-based multimodal state learning. In Section IV, we discuss perception and decision making integration. After presenting experimental results in Section V, we conclude our paper in Section VI.

II. Related Work

In this section, we provide a review of AL techniques, and state representation and recognition methods.

A. Apprenticeship Learning

Apprenticeship learning [6], also known as learning from demonstration (LfD) or imitation learning (IL) has attracted numerous attention in recent decades [12], [13], [14], which allows robots to accomplish tasks autonomously by learning from expert demonstrations without being told explicitly.

Many AL methods were reported in various applications, which fall into two categories: Direct and indirect approaches [12]. *Direct approaches* directly imitate experts by applying supervised learning to learn policy as a direct mapping from states to motion primitives. In problems with discrete action space, classification methods are used as mapping functions [13], [15], [16], [17]. For example, interactive policy learning was proposed to control a car from demonstrations based on Gaussian mixture models [16]. AL techniques based on k-Nearest Neighbors (kNN) classifiers were implemented to learn obstacle avoidance and navigation [17]. In problems with continuous action space, regression-based methods are typically used as state-action mapping functions [10], [18], [19], [20]. For example, driving actions were learned through mapping input images to actions using neural networks [10]. Robot control policy was also estimated in soccer scenarios using sparse online gaussian processes [20].

Indirect approaches models the interaction between agent and environment as a Markovian decision problem, which select the optimal policy to maximize certain reward. Most methods manually defined the reward function. For example, hand-crafted sparse reward functions was applied for policy synthesis in the task of corridor following in the reinforcement learning framework. Reward functions depending on the swing angle were implemented in a ball-in-a-cup game [14], in which optimal actions were chosen to maximize the accumulated reward. Due to the great challenge to define an effective reward function [5]. Inverse reinforcement learning was proposed to learn optimal reward functions given expert demonstrations [21], [22], [23]. For example, three methods were demonstrated in grid world and mountain-car tasks [9]. An inverse reinforcement learning method was proposed to recover unknown reward functions under MDP framework, which was able to output policy with performance close to that of the expert [6].

However, most previous studies assume the state space is known a priori, which still require at least partially manual construction of state space. To address this issue, we propose a state learning method to automatically construct state space from multimodal sequential observations provided in expert demonstrations.

B. State Representation and Recognition

As our objective is to integrate decision making and robot perception that applies onboard sensors to perceive the world state, this review will focus on methods that represent world states based on raw data directly acquired by optical cameras, which have become a standard sensor in modern robots.

Representation: Many techniques have been implemented to characterize and represent world states from image data based on features. Local and global features are two main categories for visual state representation [24]. Local features describe local information in a part of an image, including SIFT [25], ORB [26], etc. Such techniques apply a detector to identify interest points in an image and extract a feature vector by applying a descriptor around each interest point. Unlike local features, state representations based on global features describe the whole image, which encode its global color, shape, and texture signatures [27]. Examples of global features include LDB to encode intensity and gradient differences of image grid cells, GIST [28] to encode dominant spatial structures, and the recent deep feature to learn image statistics [29].

Recognition: Most of the previous state recognition methods (e.g., scene recognition) are based on individual-image matching, using pairwise similarity scoring [30], [31], nearest neighbor search [32], [33], [34], and sparse optimization [35], [36]. However, it has been demonstrated that state (or scene) recognition based on individual images cannot work well in challenging environments (e.g., with strong perceptual aliasing) [37], [38], [30], [31] and fusing information from a sequence of images is critical to match between states [30].

Different from previous techniques, we propose a unified formulation to simultaneously fuse multiple types of features to represent states and match sequence of multimodal observations for state recognition.

III. Sparse Multimodal State Learning

We propose a novel SMAL approach to (1) represent and recognize states based on multimodal observation sequences, and (2) integrate state learning with decision making to guide robot actions (e.g., performing victim search and rescue in disaster areas). This section focuses on contribution (1), and contribution (2) will be detailed in the Section IV.

Notation. In this paper, we represent vectors as boldface lowercase letters, and matrices using boldface, capital letters. Given a matrix $\mathbf{M} = \{m_{ij}\} \in \mathbb{R}^{n \times m}$, we refer to its i-th row and j-th column as $\mathbf{m}^i$ and $\mathbf{m}_j$, respectively. The ℓ_1-norm of a vector $\mathbf{v} \in \mathbb{R}^n$ is defined as $\|\mathbf{v}\|_1 = \sum_{i=1}^{n} |v_i|$, and the ℓ_2-norm of $\mathbf{v}$ is defined as $\|\mathbf{v}\|_2 = \sqrt{\mathbf{v}^\top \mathbf{v}}$. The $\ell_{2,1}$-norm of the matrix $\mathbf{M}$ is defined as:

$$\|\mathbf{M}\|_{2,1} = \sum_{i=1}^{n} \sqrt{\sum_{j=1}^{m} m_{ij}^2} = \sum_{i=1}^{n} \|\mathbf{m}^i\|_2 \tag{1}$$

A. Sequence-based Multimodal State Matching

To solve the problem of state identification in challenging real-world environments (e.g., disaster scenarios in search and rescue operations), we propose to incorporate a temporal sequence of observations (e.g., images) for state recognition and fuse multiple heterogenous sensing modalities to capture comprehensive environmental information to address perceptual aliasing.

Assume a set of templates encoding the states (e.g., scenes in victim search) from a target area $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$, and each template contains a set of r heterogenous feature modalities $\mathbf{x}_i = \left[(\mathbf{x}_i^1)^\top, (\mathbf{x}_i^2)^\top, \cdots, (\mathbf{x}_i^r)^\top\right]^\top \in \mathbb{R}^m$, where $\mathbf{x}_i^j \in \mathbb{R}^{m_j}, j = 1, \cdots, r$ represents the feature vector of length m_j extracted from the j-the feature modality and $m = \sum_{j=1}^{r} m_j$. Because our method focuses on sequence-based state learning, we group adjacent observations (e.g., camera frames) together as a temporal sequence to encode each state, resulting in the set of sequence-based templates $\mathbf{X} = [\mathbf{X}^1, \mathbf{X}^2, \cdots, \mathbf{X}^k]$, where $\mathbf{X}^j, 1 \leq j \leq k$ denotes the j-th sequence that contains l images acquired in a short time period, and k is the number of sequences in the set satisfying $k = \lfloor n/l \rfloor$. Then, given a new query sequence containing a set of l multimodal observations $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_l] \in \mathbb{R}^{m \times l}$, we formulate state identification as a learning task to estimate the weight matrix, $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_l]$:

$$\mathbf{W} = \begin{bmatrix} \mathbf{w}_1^1 & \mathbf{w}_2^1 & \dots & \mathbf{w}_l^1 \\ \mathbf{w}_1^2 & \mathbf{w}_2^2 & \dots & \mathbf{w}_l^1 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{w}_1^k & \mathbf{w}_2^k & \dots & \mathbf{w}_l^k \end{bmatrix} \in \mathbb{R}^{n \times l}, \tag{2}$$

where $\mathbf{w}_p^q \in \mathbb{R}^l$ denotes the weights of the templates in the q-th sequence $\mathbf{X}^q$ with respect to the p-th query observation $\mathbf{y}_p$ in the sequence $\mathbf{Y}$.

Since individual observations in template and query sequences can be noisy or contain missing values, we propose to constrain each observation $\mathbf{y}$ in the sequence $\mathbf{Y}$ to only rely on a small number of representative template sequences for state recognition, leading to the regularized sparse optimization problem as follows:

$$\min_{\mathbf{W}} \sum_{i=1}^{l} \left(\|\mathbf{X}\mathbf{w}_i - \mathbf{y}_i\|_2 + \lambda \|\mathbf{w}_i\|_1 \right), \tag{3}$$

where the ℓ_1-norm regularization of $\mathbf{w}_i$ forces the sparsity of the scene templates used to represent the query scene. Eq. (3) can be rewritten as a more compact matrix expression:

$$\min_{\mathbf{W}} \|(\mathbf{X}\mathbf{W} - \mathbf{Y})^\top\|_{2,1} + \lambda \|\mathbf{W}\|_1, \tag{4}$$

where $\|\mathbf{W}\|_1 = \sum_{i=1}^{l} \|\mathbf{w}_i\|_1$.

However, the regularizer of weight matrix $\mathbf{W}$ in Eq. (4) is an element-wise ℓ_1-norm, which ignores the interrelationship among individual feature modalities within each observation $\mathbf{y}$. To encode this interrelationship among individual modalities within an individual observation $\mathbf{y}$, we use the $\ell_{2,1}$-norm as a new regularization:

$$\min_{\mathbf{W}} \|(\mathbf{X}\mathbf{W} - \mathbf{Y})^\top\|_{2,1} + \lambda \|\mathbf{W}\|_{2,1}. \tag{5}$$

The $\ell_{2,1}$-norm regularization applies an ℓ_2-norm to enforce group effects of all individual modalities in the same individual observation, and uses an ℓ_1-norm to enforce the sparsity among individual observations.

To enable sequence-based state recognition, we propose a new regularization to model the group structure among all sequences. We name it the S_1-norm, because it is a structured ℓ_1-norm encoding the group structure of $\mathbf{W}$, as follows:

$$\|\mathbf{W}\|_{S_1} = \sum_{i=1}^{l} \sum_{j=1}^{k} \|\mathbf{w}_i^j\|_2. \tag{6}$$

The S_1-norm applies the ℓ_2-norm on individual observations within each sequence, and the ℓ_1-norm among sequences. That is, the new S_1-norm not only enforces the observations within the same sequence to have similar weights, but also enforces the sparsity between sequences. For example, if a template sequence $\mathbf{X}^i$ is not representative for a query observation $\mathbf{Y}$, the weights of the individual observations in $\mathbf{X}^i$ have small values; otherwise, their weights are large.

Thus, the final optimization problem becomes

$$\min_{\mathbf{W}} \|(\mathbf{X}\mathbf{W} - \mathbf{Y})^\top\|_{2,1} + \lambda_1 \|\mathbf{W}\|_{2,1} + \lambda_2 \|\mathbf{W}\|_{S_1}. \tag{7}$$

B. State Space Learning and State Identification

Our previous discussion is based upon the assumption that the state space has been provided during the training phase using expert demonstrations. However, the critical problems of how to construct state space has not been discussed.

To address this problem in the training phase, we introduce a new approach in Algorithm 1 to automatically construct the state space $\mathcal{S}$ for our sequence-based state recognition. Intuitively, if a query sequence does not match any template sequences within the database, it will be inserted into the database. Formally, after obtaining the optimal weight matrix $\mathbf{W}$, given a new sequence $\mathbf{Y}$ during training, we identify its state by matching $\mathbf{Y}$ with all existing template sequences $\mathbf{X}$. If the weight of a template sequence $\mathbf{X}^j$ satisfies:

$$\sum_{i=1}^{l} \|\mathbf{w}_i^j\|_1 \leq \tau, \tag{8}$$

where τ is a threshold with a small value, then we conclude that $\mathbf{Y}$ does not match the sequence $\mathbf{X}^j$. If $\mathbf{Y}$ does not have any matches in $\mathbf{X}$, we add $\mathbf{Y}$ into the template database. This approach ensures that there exists only one representative sequence in the template database to encode the same state (e.g., the same scene with similar viewpoints). If duplicated sequences are provided, our algorithm will ignore them, and the state space $\mathcal{S}$ will remain the same.

During the execution phase, given the query sequence of multimodal observations $\mathbf{Y}$ obtained by the robot, our SMAL method recognizes its state by solving the following problem:

$$s = \arg\max_{j} \sum_{i=1}^{l} \|\mathbf{w}_i^j\|_1, j = 1, 2, \cdots, k, \tag{9}$$

where $\mathbf{w}_i^j$ is computed by Algorithm 2.

Algorithm 1: State space learning

Input : Observations recorded during demonstrations
Output: $\mathcal{S}$ (state space), $\mathbf{X}$ (state template database, or STD), and $s\text{-}stream$ (state stream).

Initialize: $\mathbf{X}, \mathcal{S}, s\text{-}stream = \varnothing$.
while *there exist unprocessed observations* **do**
  Calculate the optimal weight matrix $\mathbf{W}$ according to Algorithm 2 with respect to $\mathbf{X}$ and the current sequence of observations $\mathbf{Y}$;
  if *no match is found by Eq. (8)* **then**
    $\mathbf{X} \leftarrow [\mathbf{X}, \mathbf{Y}]$;
    Add the new state to the state space $\mathcal{S}$;
  else
    Find the matched state by Eq. (9);
  end
  Append the current state to $s\text{-}stream$;
  Go to the next sequence of observations;
end
return $\mathcal{S}, \mathbf{X}, s\text{-}stream$.

Algorithm 2: An iterative algorithm to solve the sparse optimization problem in Eq. (7).

Input : The scene templates $\mathbf{X} \in \mathbb{R}^{m \times n}$, the query sequence of frames $\mathbf{Y} \in \mathbb{R}^{m \times l}$.
Output: The weight matrix $\mathbf{W} \in \mathbb{R}^{n \times l}$.

Initialize $\mathbf{W} \in \mathbb{R}^{n \times l}$;
while *not converge* **do**
  Calculate the diagonal matrix $\mathbf{P}$ with the i-th diagonal element as $p_{ii} = \frac{1}{2\|\mathbf{y}_i - \mathbf{X}\mathbf{w}_i\|_2}$;
  Calculate the diagonal matrix $\mathbf{Q}$ with the i-th diagonal element as $\frac{1}{2\|\mathbf{w}^i\|_2}$;
  Calculate the block diagonal matrix $\mathbf{R}^i$ $(1 \le i \le l)$ with the j-th diagonal block as $\frac{1}{2\|\mathbf{w}_i^j\|_2}\mathbf{I}_j$;
  For each $\mathbf{w}_i$ $(1 \le i \le s)$, calculate $\mathbf{w}_i = p_{ii}\left(p_{ii}\mathbf{X}^\top\mathbf{X} + \lambda_1\mathbf{Q} + \lambda_2\mathbf{R}^i\right)^{-1}\mathbf{X}^\top\mathbf{y}_i$;
end
return $\mathbf{W} \in \mathbb{R}^{n \times l}$.

C. Optimization Algorithm

Although the optimization problem in Eq. (7) is convex, it is challenging to solve it since there are non-smooth terms in the objective function. Here we provide an efficient algorithm to solve this problem that grantees theoretical convergence.

After taking the derivative of Eq. (7) with respect to $\mathbf{W}$ and setting it to 0, we have

$$\mathbf{X}^\top\mathbf{X}\mathbf{W}\mathbf{P} - \mathbf{X}^\top\mathbf{Y}\mathbf{P} + \lambda_1\mathbf{Q}\mathbf{W} + \lambda_2\mathbf{R}^i\mathbf{W} = \mathbf{0}, \quad (10)$$

where $\mathbf{P}$ is a diagonal matrix with the i-th diagonal element equals $p_{ii} = \frac{1}{2\|\mathbf{y}_i - \mathbf{X}\mathbf{w}_i\|_2}$, $\mathbf{Q}$ is a diagonal matrix with the i-th element as $\frac{1}{2\|\mathbf{w}^i\|_2}$, and $\mathbf{R}^i$ is a block diagonal matrix with the i-th diagonal block as $\frac{1}{2\|\mathbf{w}_i^j\|_2}\mathbf{I}$, where $\mathbf{I}$ denotes an l dimensional identity matrix. For each i, we obtain

$$p_{ii}\mathbf{X}^\top\mathbf{X}\mathbf{w}_i - p_{ii}\mathbf{X}^\top\mathbf{y}_i + \lambda_1\mathbf{Q}\mathbf{w}_i + \lambda_2\mathbf{R}^i\mathbf{w}_i = \mathbf{0}. \quad (11)$$

Therefore, $\mathbf{w}_i$ can be calculated by

$$\mathbf{w}_i = p_{ii}\left(p_{ii}\mathbf{X}^\top\mathbf{X} + \lambda_1\mathbf{Q} + \lambda_2\mathbf{R}^i\right)^{-1}\mathbf{X}^\top\mathbf{y}_i. \quad (12)$$

We can observe that the matrices $\mathbf{P}, \mathbf{Q}$, and $\mathbf{R}$ in Eq. (12) all depend on the weight matrix $\mathbf{W}$, which are unknown. To solve this regularized optimization problem, we propose an iterative solver as presented in Algorithm 2. We can prove that Algorithm 2 guarantees the theoretical convergence to the global optimal solution. Detailed analysis and mathematical proof is provided in Appendix.

IV. Integration of State Perception and Decision Making

Beyond the ability to automatically learn states, our SMAL method is also able to integrate state perception and decision making. This integration allows a robot to directly utilize raw multisensory observation sequences to make decisions and take actions, without assuming perfect perception or handcrafted states that are not practical in complicated real-world environments (e.g., search and rescue scenarios).

We propose to achieve the integration of our state perception with the general Markov decision process (MDP) model, which has been widely employed for robot decision making, to show the generalization of our SMAL method that has the potential to impact various robotics applications using MDP. From the viewpoint of real-world online robot execution, the input data into our integrated model is the raw multimodal observation sequences obtained by sensors equipped on the robot, and the output of our SMAL method is an optimal action learned in response to the state identified by our perception method.

Formally, the integrated perception and decision making model of our SMAL method is represented as a tuple $\Omega = (\mathcal{S}, \mathcal{A}, T, R, \gamma)$, where $\mathcal{S} = \{s_0, s_1, \cdots, s_{N_s}\}$ denotes a finite set of discrete states; $\mathcal{A} = \{a_0, a_1, \cdots, a_{N_a}\}$ represents a finite set of discrete actions that human/robot can perform to activate state transitions; $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ denotes a discrete transition function representing the probability of a state transition resulted from an action; $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ denotes a mapping from the state-action pair to a scalar, representing the immediate reward received when the robot takes action $a \in \mathcal{A}$ in state $s \in \mathcal{S}$; and $\gamma \in [0, 1]$ is a reward discount factor. Different from previous MDP-based methods whose states are typically computed at a specific time point and represented by a single modality, our integrated model represents a state based on a sequence of observations with multimodal modalities. This integration is realized using our sequence-based multimodal state recognition method that transfers a multimodal observation sequence $\mathbf{Y} \in \mathbb{R}^{m \times l}$ into a discrete value $s = s(\mathbf{Y}) \in \mathbb{Z}$, as defined in Eq. (9).

Same as all MDP-based decision making, our integrated model aims to learn a policy π that is defined as a mapping from the learned state space $\mathcal{S}$ to the action space $\mathcal{A}$. The

value of a policy π is given by $V^{\pi} = \sum_{t=0}^{\infty} \gamma^t R(s_t, \pi(s_t))$. Then, the objective of decision making is to find an optimal policy $\pi^{\star}$ to maximize the value function V^{π}:

$$\pi^{\star} = \arg\max_{\pi} \sum_{t=0}^{\infty} \gamma^t R(s_t, \pi(s_t)) \tag{13}$$

In the following, we describe our implemented methods to learn other components of the MDP model used in integrated SMAL method, as follows:

Learning Action Space. The action space $\mathcal{A}$ can be learned based on the kinematic data collected during expert demonstrations. In our experiments, teleoperation command streams provided by humans are recorded and used to learn the action space $\mathcal{A}$, where each action $a \in \mathcal{A}$ consists of a sequence of l atom movements. Such atom movements include moving forward, moving backward, turning left, and turning right. Ideally, actions are continuous, but robots perform actions in discrete-time during the execution phase, since the specific optimal action is selected based on the current state, which is discrete and recognized at every l frames. The action space $\mathcal{A}$ is learned by Algorithm 3.

Algorithm 3: Algorithm to learn action space $\mathcal{A}$

Input : Recorded kinematic stream $k\text{-}stream$, and state stream $s\text{-}stream$ learned by Algorithm 1
Output: The action space $\mathcal{A}$, and action stream $a\text{-}stream$.

Initialize: $a\text{-}stream, \mathcal{A} = \varnothing$.
while *there exists unprocessed kinematic data* **do**
  Get a sequence of l atom movements am from the kinematic stream
  Append am to the action stream $a\text{-}stream$;
  if am *is not contained in* $\mathcal{A}$ **then**
    Insert am to $\mathcal{A}$;
  end
end
return $\mathcal{A}, a\text{-}stream$.

Learning State Transition. The state transition $T(s, a, s')$ represents the probability that the system will end up in state s' after taking action a in state s. The state transition T is learned using the state and action streams obtained in Algorithms 1 and 3, respectively. In our implementation, the state transition is learned by Algorithm 4.

Learning Immediate Reward. After the MDP model $\Omega = (\mathcal{S}, \mathcal{A}, T) \backslash R$ is learned, we are able to learn the immediate reward $R(s, a)$ provided by the human demonstrations and a predefined γ. A widely used technique is inverse reinforcement learning. We directly employed the technique in [9], in which reward learning is formulated as a sparse optimization problem since the maximum reward (i.e., finding victims) in our application is achieved at the end state.

V. Experiments

To evaluate the performance of our SMAL approach, we performed two sets of experiments in different scenarios to address the application of robot-assisted search and rescue, including (1) urban search and rescue in simulation, and (2) indoor search tasks using real robots. The mission objective for the robot is to find a victim within the environment, who are not directly viewable by the robot.

Algorithm 4: Algorithm to learn state transition.

Input : State stream $s\text{-}stream$ and action stream $a\text{-}stream$
Output: The state transitions T

Initialize: State transition map $STM = \varnothing$,
for $i = 1 :$ *length of* $s\text{-}stream$ **do**
  Append the value of key $s(i)$ with $(a(i), s(i+1))$.
end
for *key* s *in* STM **do**
  $T(s, a, s') = \dfrac{\text{Number of } (a, s') \text{ in } STM[s]}{\text{Number of } (a) \text{ in } STM[s]}$
end
return T.

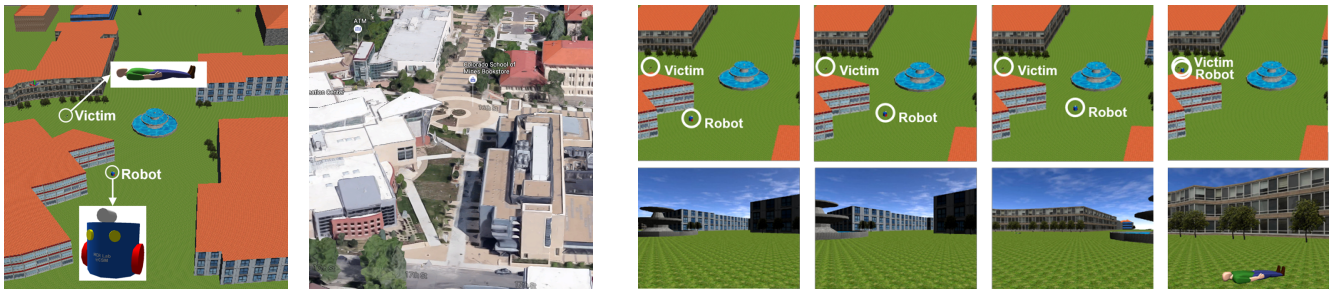
In our experiments, the (simulated and real) robots employ a camera to perceive the surrounding world; multiple feature modalities are applied to extract information to represent the world. To enable real-time performance, we intentionally use feature modalities that can be extracted efficiently, including low-resolution color features on 24×32 downsampled images and histogram of oriented gradients features on 240×320 downsampled images. The visual feature modalities are normalized and concatenated as a multimodal representation of individual observations.

A. Urban Search and Rescue Simulation

In this set of experiments, we apply the Webots simulator[1] [39] to evaluate our SMAL approach in an urban search and rescue application. The objective is to let a robot learn how to find victims in large urban areas from expert demonstrations. We chose the campus of the Colorado School of Mines as our urban environments. The Google satellite map of this area is shown in Fig. 2(b). We imported the OpenStreetMap[2] of this area into the Webots platform, as illustrated in Fig. 2(a). The robot and victim models we built in the Webots platform are shown in Fig. 2(a). The two-wheel mobile robot, named *Rescuebot*, equips with a color camera with a 1024×768 resolution. In addition, we are able to obtain the accurate Rescuebot's location and rotation information from the simulator, which is used as the ground truth to evaluate state recognition. The victim is lying on the ground without any movement during the entire simulation period, waiting for a robot to find him.

During the training process, we teleoperated the *Rescuebot* to approach the target victim using keyboards as the expert demonstration. The image sequences obtained by the *Rescuebot* and the keyboard teleoperation commands were

[1]Webots: `https://www.cyberbotics.com`.
[2]OpenStreetMap: `http://www.openstreetmap.org/#map=18/39.74966/-105.22212`.

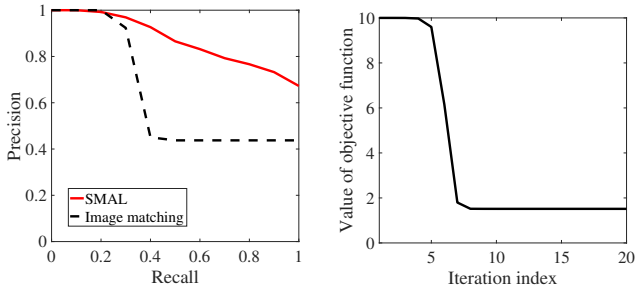


(a) Simulated environments (b) Google map of real scenes (c) Rescuebot moving route and acquired observations

Fig. 2. Experiment setups and qualitative results in robot-assisted urban search and rescue scenarios. Fig. 2(a) illustrates the simulated environment. Fig. 2(a) shows the Google satellite map of the real campus environment of the Colorado School of Mines. Fig. 2(c) illustrates qualitative results with the top row showing the robot moving route and the bottom row showing the observations obtained by the robot camera.

recorded to train our SMAL method. After training was completed, the *Rescuebot* was able to automatically execute search operations using the learned model in the testing phase.

To qualitatively evaluate the experimental results, an example route that the *Rescuebot* successfully finds the victim in the execution phase is presented in Fig. 2(c). It demonstrates that, although the *Rescuebot* cannot see the victim directly, the robot is still able to move and search around to locate the victim. This qualitative result demonstrates that our SMAL method enables robots to learn how to autonomously search for victims in urban search and rescue scenarios.



(a) Precision-recall curves (b) Objective value

Fig. 3. Quantitative evaluation of our SMAL approach in simulated urban search and rescue scenarios.

In addition, we perform quantitative validation using the precision-recall curve as a metric to evaluate the performance of state recognition, as shown in Fig. 3(a) (curves closer to the top right corner indicating a better performance). We also compared the SMAL approach to the baseline method based on individual images with the same modalities, which is demonstrated in Fig. 3(a). It is observed that our SMAL method for sequence-based state recognition outperforms the baseline method using individual images.

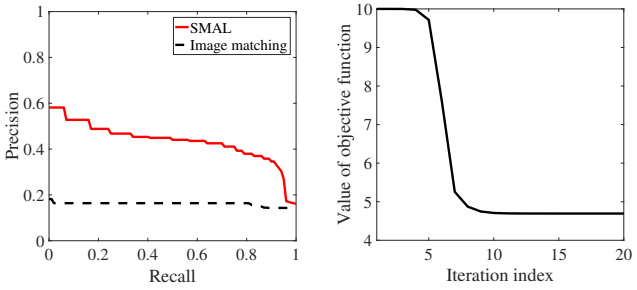
We also evaluate the efficiency of our methods for state recognition through studying the value of objective function iteratively updated by Algorithm 2. The result, presented in Fig. 3(b), indicates the algorithm converges in 9 iterations (in general, it converges within 20 iterations with the value below 10^{-4}, which demonstrates the algorithm efficiency to solve the formulated regularized optimization problem.

B. Indoor Search and Rescue using Real TurtleBot

In this set of experiments, we evaluate our SMAL method to teach robots to perform victim search in indoor scenarios. A real TurtleBot II robot is used to evaluate the performance of our system. The objective is to teach the TurtleBot about how to find victims (in this experiment, a NAO humanoid robot) in the room using expert demonstrations. The setup of the indoor search area is presented in Fig. 4(a). We also install an overhead camera above this area to collect the ground truth of robot location and orientation for evaluation only by tracking the ARTag attached on top of the Turtlebot.

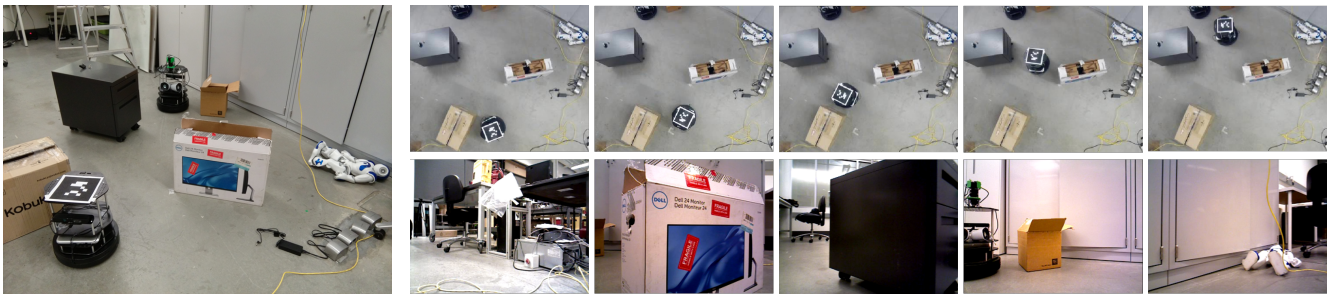
In the training phase, we teleoperated the TurtleBot using keyboards as demonstrations to let it approach the Nao robot. The observation obtained by the TurtleBot and the keyboard teleoperation commands were recorded to train our SMAL model. After that, during the execution phase, the TurtleBot executed the search task based on the learned model to find the NAO robot. A challenge of this real-world experiment in comparison to simulation is that the TurtleBot often shook when moving, making the captured observations unstable, which can decrease the accuracy of state recognition.

The qualitative experimental results are illustrated in Fig. 4(b), which indicates even the Turtlebot cannot directly see the victim (i.e., the NAO robot in this set of experiments), but it can still navigate around multiple obstacles to find the victim. This demonstrates the effectiveness of our SMAL approach to teach robots about how to search victims in



(a) Precision-recall curve (b) Objective value

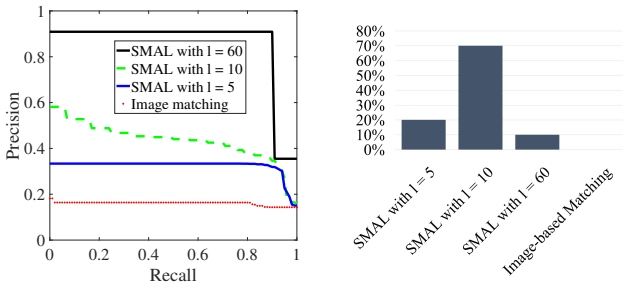
Fig. 5. Quantitative evaluation of our SMAL approach in real-world indoor search and rescue scenarios.



(a) Experiment setup (b) Moving route and observations of the TurtleBot during execution.

Fig. 4. Experiment setups of robot-assisted search and rescue in indoor environments and qualitative results. Fig. 4(a) shows the indoor environment used in this set of experiments for robots to search the victim (i.e., the NAO robot). Qualitative experimental results are presented in Fig. 4(b), with the top row showing the moving route of the TurtleBot from the viewpoint of an overhead camera, and the bottom row showing the observations acquired by the TurtleBot during the execution.

a real indoor environment. We also quantitatively evaluate our method's performance using precession-recall curves and compare SMAL with the baseline method using image matching. The results are presented in Fig. 5(a), which shows our approach significantly outperforms the baseline method. The efficiency of our SMAL approach is proved in Fig. 5(b), which shows the algorithm converges after 12 iterations.



(a) Precision-recall curves (b) Success rates

Fig. 6. Performance evaluation of SMAL using different parameter values.

C. Parameter Analysis

We analyze the effects of various parameter values on our SMAL approach in real-world indoor search tasks using real TurtleBots.

The sequence length l for state recognition is the most important parameter. The precision-recall curves in Fig. 6(a) indicate that better performance can be obtained when we increase the sequence length. That is because long sequences can provide more comprehensive information than short sequences. When $l = 1$, a sequence becomes a single image. In addition, we use success rate as a metric to evaluate the percentage that the robot can successfully find victims without hitting obstacles. The results are demonstrated in Fig. 6(b), where 10 executions are used in each case to calculate the success rate. It is observed that when the used sequence is short, the poor perception result negatively affects decision making, resulting in the low success rate. However, longer sequences do not necessarily result in higher success rates. This is because as we increase the sequence length, although each sequence can contain more information, the frequency in which the robot receives observations decreases. This can dramatically decrease the succuss rate, since the information does not come in time for robot control.

VI. Conclusion

We propose a novel *sequence-based multimodal apprenticeship learning* approach that can automatically learn and identify world states, and integrates perception and decision making. The SMAL approach represents each state as a sequence of multimodal observations by simultaneously fusing temporal information and multimodal data. The SMAL approach also integrates robot perception and decision making to learn tasks from human demonstrations to enable effective robot actions in challenging environments with perceptual aliasing. To evaluate the performance of the SMAL method, experiments using both simulations and real-world robots are performed in the challenging search and rescue applications. Qualitative results have validated that our method is able to guide autonomous robots to successfully finish the search and rescue task. In addition, quantitative evaluation results have demonstrated that our SMAL method outperforms baseline methods based on individual images to find victims in the challenging search and rescue applications.

Appendix I
Convergence Analysis of Algorithm 2

Theorem 1: Algorithm 2 decreases the objective value of the problem in Eq. (7) in each iteration.

The following lemma [40] is used to prove Theorem 1.

Lemma 1: For any nonzero vector $\tilde{\mathbf{a}}$ and $\mathbf{a}$, the following inequality holds: $\|\tilde{\mathbf{a}}\|_2 - \frac{\|\tilde{\mathbf{a}}\|_2^2}{2\|\mathbf{a}\|_2} \leq \|\mathbf{a}\|_2 - \frac{\|\mathbf{a}\|_2^2}{2\|\mathbf{a}\|_2}$.

Then we are ready to prove the convergence of Algorithm 2, which is represented by Theorem 1.

Proof: We denote the update of $\mathbf{W}$ is $\tilde{\mathbf{W}}$. According to Step 6 in Algorithm 2, we have:

$$\tilde{\mathbf{W}} = \arg\min_{\mathbf{W}} Tr((\mathbf{XW} - \mathbf{Y})\mathbf{P}(\mathbf{XW} - \mathbf{Y})^\top) + \lambda_1 Tr(\mathbf{W}^\top \mathbf{QW}) + \lambda_2 \sum_{i=1}^{l} \mathbf{w}_i^\top \mathbf{R}^i \mathbf{w}_i. \quad (14)$$

Thus, we can obtain

$$\begin{aligned} & Tr((\mathbf{X}\tilde{\mathbf{W}}-\mathbf{Y})\mathbf{P}(\mathbf{X}\tilde{\mathbf{W}}-\mathbf{Y})^\top) \\ & \quad + \lambda_1 Tr(\tilde{\mathbf{W}}^\top \mathbf{Q}\tilde{\mathbf{W}}) + \lambda_2 \sum_{i=1}^{l} \tilde{\mathbf{w}}_i^\top \mathbf{R}^i \tilde{\mathbf{w}}_i \\ & \leq Tr((\mathbf{X}\mathbf{W}-\mathbf{Y})\mathbf{P}(\mathbf{X}\mathbf{W}-\mathbf{Y})^\top) \\ & \quad + \lambda_1 Tr(\mathbf{W}^\top \mathbf{Q}\mathbf{W}) + \lambda_2 \sum_{i=1}^{l} \mathbf{w}_i^\top \mathbf{R}^i \mathbf{w}_i \end{aligned} \tag{15}$$

We are able to derive the following inequalities according to the definition of $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$:

$$\begin{aligned} & \sum_{i=1}^{l} \left(\frac{\|\mathbf{X}\tilde{\mathbf{w}}_i - \mathbf{y}_i\|_2^2}{2\|\mathbf{X}\mathbf{w}_i - \mathbf{y}_i\|_2} + \lambda_1 \frac{\|\tilde{\mathbf{w}}\|_2^2}{2\|\mathbf{w}\|_2} + \lambda_2 \sum_{j=1}^{k} \frac{\|\tilde{\mathbf{w}}_i^j\|_2^2}{2\|\mathbf{w}_i^j\|_2} \right) \\ & \leq \sum_{i=1}^{l} \left(\frac{\|\mathbf{X}\mathbf{w}_i - \mathbf{y}_i\|_2^2}{2\|\mathbf{X}\mathbf{w}_i - \mathbf{y}_i\|_2} + \lambda_1 \frac{\|\mathbf{w}\|_2^2}{2\|\mathbf{w}\|_2} + \lambda_2 \sum_{j=1}^{k} \frac{\|\mathbf{w}_i^j\|_2^2}{2\|\mathbf{w}_i^j\|_2} \right) \end{aligned}$$

According to Lemma 1, we obtain the inequalities:

$$\begin{aligned} & \sum_{i=1}^{l} \left(\|\mathbf{X}\tilde{\mathbf{w}}_i - \mathbf{y}_i\|_2 - \frac{\|\mathbf{X}\tilde{\mathbf{w}}_i - \mathbf{y}_i\|_2^2}{2\|\mathbf{X}\mathbf{w}_i - \mathbf{y}_i\|_2} \right) \\ & \leq \sum_{i=1}^{l} \left(\|\mathbf{X}\mathbf{w}_i - \mathbf{y}_i\|_2 - \frac{\|\mathbf{X}\mathbf{w}_i - \mathbf{y}_i\|_2^2}{2\|\mathbf{X}\mathbf{w}_i - \mathbf{y}_i\|_2} \right) \end{aligned} \tag{16}$$

$$\sum_{i=1}^{l} \left(\|\tilde{\mathbf{w}}\|_2 - \lambda_1 \frac{\|\tilde{\mathbf{w}}\|_2^2}{2\|\mathbf{w}\|_2} \right) \leq \sum_{i=1}^{l} \left(\|\mathbf{w}\|_2 - \lambda_1 \frac{\|\mathbf{w}\|_2^2}{2\|\mathbf{w}\|_2} \right)$$

$$\sum_{i=1}^{l}\sum_{j=1}^{k} \left(\|\tilde{\mathbf{w}}_i^j\|_2 - \frac{\|\tilde{\mathbf{w}}_i^j\|_2^2}{2\|\mathbf{w}_i^j\|_2} \right) \leq \sum_{i=1}^{l}\sum_{j=1}^{k} \left(\|\mathbf{w}_i^j\|_2 - \frac{\|\mathbf{w}_i^j\|_2^2}{2\|\mathbf{w}_i^j\|_2} \right)$$

After computing the summation of the three equations in Eq. (16) on both sides (weighted by λs), we obtain:

$$\begin{aligned} & \sum_{i=1}^{l} \|(\mathbf{X}\tilde{\mathbf{w}}_i - \mathbf{y}_i)^\top\|_2 + \lambda_1 \|\tilde{\mathbf{w}}\|_2 + \lambda_2 \|\tilde{\mathbf{w}}\|_2 \\ & \leq \sum_{i=1}^{l} \|(\mathbf{X}\mathbf{w}_i - \mathbf{y}_i)^\top\|_2 + \lambda_1 \|\mathbf{w}\|_2 + \lambda_2 \|\mathbf{w}\|_2 \end{aligned} \tag{17}$$

Thus, we conclude that Algorithm 2 decreases the objective value monotonically during each iteration. Because Eq. (7) is a convex optimization function, Algorithm 2 converges to the global optimal solution. ■

REFERENCES

[1] J. D. Sweeney and R. Grupen, "A model of shared grasp affordances from demonstration," in *Humanoid*, 2007.

[2] J. Chen and A. Zelinsky, "Programing by demonstration: Coping with suboptimal teaching actions," *IJRR*, vol. 22, no. 5, pp. 299–319, 2003.

[3] P. Abbeel, A. Coates, and A. Y. Ng, "Autonomous helicopter aerobatics through apprenticeship learning," *IJRR*, 2010.

[4] W. D. Smart, "Making reinforcement learning work on real robots," Ph.D. dissertation, Brown University, 2002.

[5] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *RAS*, vol. 57, pp. 469–483, 2009.

[6] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *ICML*, 2004.

[7] J. A. Bagnell and J. G. Schneider, "Autonomous helicopter control using reinforcement learning policy search methods," in *ICRA*, 2001.

[8] R. Amit and M. Matari, "Learning movement sequences from demonstration," in *ICDL*, 2002.

[9] A. Y. Ng, S. J. Russell, *et al.*, "Algorithms for inverse reinforcement learning," in *ICML*, 2000.

[10] D. A. Pomerleau, "Efficient training of artificial neural networks for autonomous navigation," *NC*, vol. 3, no. 1, pp. 88–97, 1991.

[11] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, *et al.*, "End to end learning for self-driving cars," *arXiv*, 2016.

[12] G. Neu and C. Szepesvári, "Apprenticeship learning using inverse reinforcement learning and gradient methods," in *UAI*, 2007.

[13] A. Lockerd and C. Breazeal, "Tutelage and socially guided robot learning," in *IROS*, 2004.

[14] B. Nemec, M. Zorko, and L. Žlajpah, "Learning of a ball-in-a-cup playing robot," in *IWRAR*, 2010.

[15] K. R. Dixon and P. K. Khosla, "Learning by observation with mobile robots: A computational approach," in *ICRA*, 2004.

[16] S. Chernova and M. Veloso, "Confidence-based policy learning from demonstration using gaussian mixture models," in *IJCAAMS*, 2007.

[17] J. Saunders, C. L. Nehaniv, and K. Dautenhahn, "Teaching robots by moulding behavior and scaffolding the environment," in *HRI*, 2006.

[18] S. Vijayakumar and S. Schaal, "Locally weighted projection regression: An O(n) algorithm for incremental real time learning in high dimensional space," in *ICML*, 2000.

[19] W. D. Smart and L. P. Kaelbling, "Effective reinforcement learning for mobile robots," in *ICRA*, 2002.

[20] D. H. Grollman and O. C. Jenkins, "Sparse incremental learning for interactive robot control policy estimation," in *ICRA*, 2008.

[21] S. Russell, "Learning agents for uncertain environments," in *ACCLT*, 1998.

[22] P. Ranchod, B. Rosman, and G. Konidaris, "Nonparametric bayesian reward segmentation for skill discovery using inverse reinforcement learning," in *IROS*, 2015.

[23] H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard, "Socially compliant mobile robot navigation via inverse reinforcement learning," *IJRR*, vol. 35, pp. 1352–1370, 2016.

[24] H. Zhang, F. Han, and H. Wang, "Robust multimodal sequence-based loop closure detection via structured sparsity," in *RSS*, 2016.

[25] A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer, "Fast and incremental method for loop-closure detection using bags of visual words," *TRO*, vol. 24, no. 5, pp. 1027–1037, 2008.

[26] R. Mur-Artal and J. D. Tardós, "Fast relocalisation and loop closing in keyframe-based SLAM," in *ICRA*, 2014.

[27] F. Han, X. Yang, Y. Deng, M. Rentschler, D. Yang, and H. Zhang, "SRAL: Shared representative appearance learning for long-term visual place recognition," *RA-L*, 2017, to appear.

[28] M. Dymczyk, S. Lynen, T. Cieslewski, M. Bosse, R. Siegwart, and P. Furgale, "The gist of maps-summarizing experience for lifelong localization," in *ICRA*, 2015.

[29] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *NIPS*, 2014.

[30] M. J. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *ICRA*, 2012.

[31] M. J. Milford, G. F. Wyeth, and D. Prasser, "RatSLAM: a hippocampal model for simultaneous localization and mapping," in *ICRA*, 2004.

[32] C. Chen and H. Wang, "Appearance-based topological Bayesian inference for loop-closing detection in a cross-country environment," *IJRR*, vol. 25, no. 10, pp. 953–983, 2006.

[33] M. Cummins and P. Newman, "FAB-MAP: Probabilistic localization and mapping in the space of appearance," *IJRR*, pp. 647–665, 2008.

[34] M. Labbe and F. Michaud, "Appearance-based loop closure detection for online large-scale and long-term operation," *TRO*, vol. 29, no. 3, pp. 734–745, 2013.

[35] Y. Latif, G. Huang, J. Leonard, and J. Neira, "An online sparsitycognizant loop-closure algorithm for visual navigation," in *RSS*, 2014.

[36] X. Yang, F. Han, H. Wang, and H. Zhang, "Enforcing template representability and temporal consistency for adaptive sparse tracking," in *IJCAI*, 2016.

[37] 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 *ICRA*, 2015.

[38] E. Johns and G.-Z. Yang, "Feature co-occurrence maps: Appearance-based localisation throughout the day," in *ICRA*, 2013.

[39] O. Michel, "Webots™: Professional mobile robot simulation," *arXiv*, 2004.

[40] F. Nie, H. Huang, X. Cai, and C. H. Ding, "Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization," in *NIPS*, 2010.