GENERATED IMAGE FEATURE BASED SELECTIVE ATTENTION MECHANISM BY VISUO-MOTOR LEARNING

T. Minato[1] and M. Asada[1]

[1] Department of Adaptive Machine Systems, Graduate School of Engineering, Osaka University, 2-1 Yamada-oka, Suita, Osaka 565-0871 Japan

ABSTRACT

Visual attention is an essential mechanism of an intelligent robot. Existing research typically specifies in advance the attention control scheme required for a given robot to perform a specific task. However, a robot should be able to adapt its own attention control to varied tasks. In our previous work, we proposed a method of generating a filter to extract an image feature by visuo-motor learning. The generated image feature extractor is considered to be generalized knowledge to accomplish a task of a certain class. We propose an attention mechanism, by which the robot selects the generated feature extractors based on its task-oriented criterion.

KEYWORDS

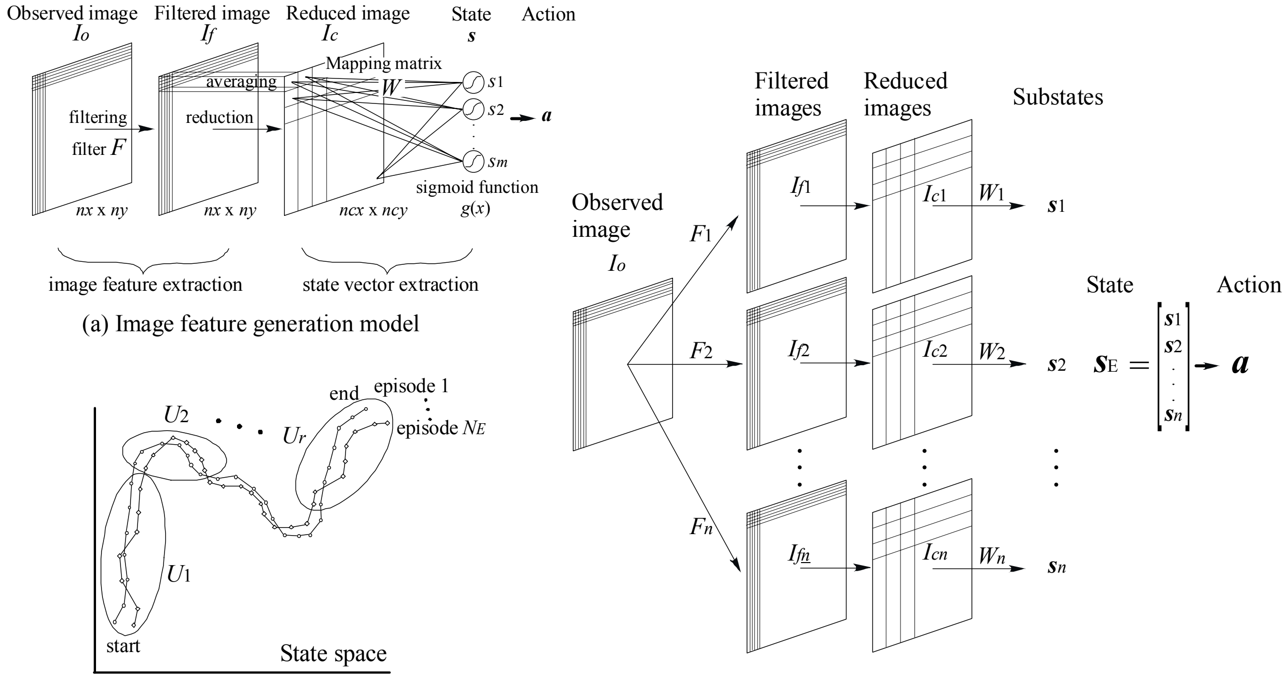
Mobile Robot, Selective attention, Image feature generation, Image feature selection, Task-oriented

INTRODUCTION

Attention control is an essential mechanism for an intelligent robot to avoid processing enormous amounts of data. It is a data reduction process to facilitate decision making. With regard to visual attention control, it involves selection of focus, image features, and so on. Existing research typically specifies in advance the attention control scheme required for a given robot to perform a specific task. However, a robot should be able to adapt its own attention control to varied tasks and environments.

We have focused on visual attention control related to a robot’s actions to accomplish a given task and proposed a method in which a robot generates an image feature extractor (i.e., image filter) which is necessary for the selection of actions through visuo-motor map learning (Minato & Asada, 2003). The robot's learning depends on the experience gathered while performing a task. In this method, the robot uses only one feature extractor for a given task. For more complex tasks, however, multiple feature extractors are necessary to accomplish the tasks and a method of selecting them should be addressed.

Some research has focused on a method of feature selection based on task-relevant criteria. McCallum (1996) proposed a method in which a robot learns not only its action but feature selection using



(a) Image feature generation model

(b) Segmentation of supervised data

(c) Image feature selection model

Figure 1: Image feature generation and selection models

reinforcement learning. Mitsunaga and Asada (2000) proposed a method to select a landmark according to the information gain on action selection. In these methods, however, the image features to detect the landmarks from the observed image are given a priori. It is desirable that the image feature adapts to environmental changes.

This paper proposes a method in which a robot learns to select image feature extractors generated by itself according to a task-relevant criterion. The generated feature extractors are not always suitable for new tasks. The robot must learn to select them to accomplish the task. The criterion of selection is the information gain calculated from given task instances (supervised data). Furthermore, a part of supervised data which gives the local information of the task makes the selective mechanism more effective. The method is applied to indoor navigation.

THE BASIC IDEA

In the proposed method, a robot generates an image feature extractor that is necessary for the action selection through visuo-motor map learning (Minato & Asada, 2003). The state calculation process is decomposed into feature extraction and state extraction (Figure 1(a)). A robot learns the effective feature extractor and state mapping matrix for a given task through a mapping from observed images to supervised actions. During feature extraction, the interactions between raw data are limited to local areas, while the connections between the filtered image and the state spread over the entire space to represent non-local interactions. It is, therefore, expected that the feature extractors are more general and could be generalized knowledge to accomplish a task of a certain class.

The robot calculates the filtered image I_f from the observed image I_o using the feature extractor *F*. The state $\boldsymbol{s} \in \Re^m$ is calculated from a compressed image I_c by the sum of weighted pixel values. The robot decides the appropriate action for the current state $\boldsymbol{s}$. The function model of the feature extractor is given, and the robot learns its parameters and the mapping matrix *W* by maximizing the information gain of $\boldsymbol{s}$ with respect to action $\boldsymbol{a}$.

The robot, which generates one feature extractor for a given task, obviously needs multiple feature extractors for more complex tasks. It is unnecessary to learn a feature extractor for every given task. The generated feature extractor must be generalized to make the robot more adaptable.

In this method, the robot reuses a number of generated feature extractors from past experiences and selects effective ones for action decision. The system is shown in Figure 1(c). The robot is given a number of different feature extractors, but must select those which are appropriate for the given task. The robot, therefore, learns the state mapping matrix using the supervised data and evaluates which feature extractor is appropriate from the distribution of supervised data. If the robot uses all of the supervised data in the evaluation, optimality in a local part of the task is lost. To evaluate the effectiveness in the local task, the robot estimates which local task it is performing from the history of observations and selects the feature extractor using a portion of the supervised data corresponding to the local task.

SELECTIVE ATTENTION MECHANISM BASED ON GENERATED IMAGE FEATURE EXTRACTORS

The System Overview

The robot is given n different feature extractors $(F_i, i = 1,...,n)$ and calculates the substate $\boldsymbol{s}_i \in \Re^m$ using the mapping matrix W_i corresponding to F_i. Each mapping matrix is learned by maximizing the information gain of $\boldsymbol{s}_E$ (direct product of $\boldsymbol{s}_1,...,\boldsymbol{s}_n$) with respect to the supervised action $\boldsymbol{a} \in A$.

The robot selects the feature extractor which has a maximum expected information gain and decides the appropriate action for the substate calculated using the selected feature extractor. It cannot always decide the appropriate action using one feature extractor. It, therefore, estimates the reliability of selected feature extractors and selects repeatedly until the reliability exceeds a given threshold.

For evaluation in the local task, the supervised data is segmented by temporal order. The robot selects a sub-supervised data according to the history of observation and selects feature extractors to decide an action using the selected one.

State learning

First, the robot collects supervised successful instances of the given task for N_E episodes. An episode ends when the robot accomplishes the task. An instance u consists of an observed image I_o^u and a given action $\boldsymbol{a}^u$. Next, the robot learns the mapping matrices. The state $\boldsymbol{s}_E^u$ consists of substates $\boldsymbol{s}_i^u$ which are calculated from I_o^u using F_i and W_i (the superscript denotes the corresponding instance). The evaluation function used to learn W_i is to maximize the information gain of $\boldsymbol{s}_E$ with respect to $\boldsymbol{a}$. It is equivalent to minimizing the following risk function R (see Vlassis, Bunschoten, and Kröse (2001)).

$$R = -\frac{1}{N}\sum_{u \in U} \log p(\boldsymbol{a}^u \mid \boldsymbol{s}_E^u) \,. \tag{1}$$

In Eqn. 1 U denotes a set of all instances and N denotes the number of instances. The probability density functions are computed using kernel smoothing. Using the gradient method, the mapping matrices W_i, which minimize *R*, are obtained.

Feature Extractor Selection

The set of instances *U* is divided into *r* subsets $U_j, j = 1,...,r$ before performing the task (Figure 1(b)). The subsets are arranged by temporal order. The choice of *r* includes a trade-off between the locality of the evaluation and the reliability of the action decision. To evaluate it, *U* is divided so that instances of similar state and action are included into the same subset. The vector $\boldsymbol{c}^u = (\boldsymbol{s}_E^u, \boldsymbol{a}^u, \tau^u / L)$ is defined from the instance *u*, and *U* is divided by applying the ISODATA algorithm for the set $\{\boldsymbol{c}^u\}$. Here, *L* is the time taken to accomplish the task and τ is the time when the instance *u* is observed. The value of each component is normalized to the range [0,1]. To avoid aliasing problems, the robot always uses two neighbouring subsets to evaluate the effectiveness of a feature extractor.

The robot executes the following process at every interval.

1) Selecting subsets of instances: Select subsets of instances $\mathcal{V}$ according to a procedure shown in the next section. $k = 0$.
2) Calculating a reliability of action decision: Calculate substate $\boldsymbol{s}_{ok}$ corresponding to the *k*-th selected feature extractors F_{ok} and the entropy $H_{\mathcal{V}}(A \mid S_o)$ using the instances in $\mathcal{V}$.

$$H_{\mathcal{V}}(A \mid S_o) = -\sum_{u \in \mathcal{V}} P_{\mathcal{V}}(\boldsymbol{a}^u \mid S_o) \log P_{\mathcal{V}}(\boldsymbol{a}^u \mid S_o), \tag{2}$$

where $S_o = \{\boldsymbol{s}_{o1}, ..., \boldsymbol{s}_{ok}\}$ ($S_o = \phi$, if $k = 0$) and $P_{\mathcal{V}}$ denotes a probability calculated on the set $\mathcal{V}$. $H_{\mathcal{V}}(A \mid S_o)$ means an uncertainty of the action decision. Evaluate the uncertainty using a threshold H_{th}.

- If $H_{\mathcal{V}}(A \mid S_o) \le H_{th}$, then go to 4.
- Otherwise, $k = n$ and $\mathcal{V} = U$, then go to 4.
- Otherwise, $k = n$ and $\mathcal{V} \ne U$, then go to 2 with $\mathcal{V} = U$ and $k = 0$.
- Otherwise, go to 3.

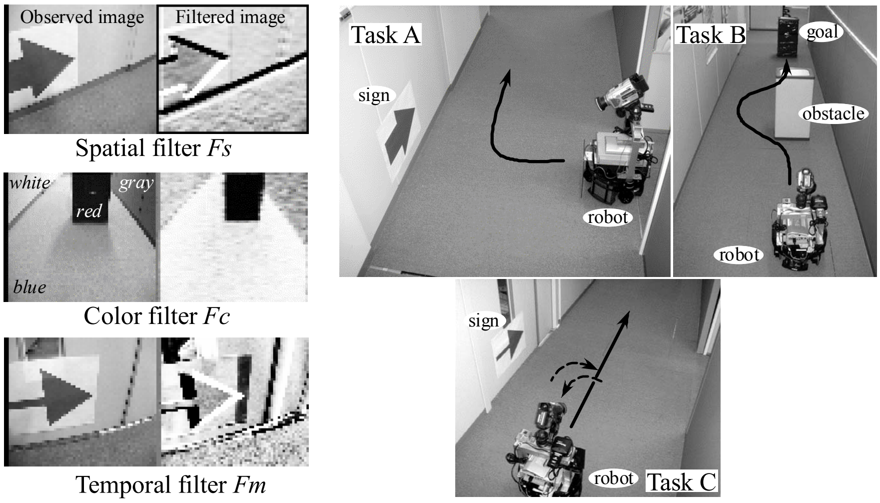
3) Selecting a feature extractor: Let the set of unselected feature extractors be $\mathcal{F}$. Calculate an expected entropy for each unselected feature extractor $F_z \in \mathcal{F}$. The expected entropy is:

$$\sum_{u \in \mathcal{V}} P_{\mathcal{V}}(\boldsymbol{s}_z^u) H_{\mathcal{V}}(A \mid S_o, \boldsymbol{s}_z^u), \tag{3}$$

where $\boldsymbol{s}_z$ is a substate corresponding to F_z. Select the feature extractor F_{ok+1} which has the minimum entropy, that is, has the maximum information gain. $k \leftarrow k+1$. go to 2.
4) Deciding an action: Execute the action $\boldsymbol{a}$ which maximizes $P_{\mathcal{V}}(\boldsymbol{a} \mid S_o)$.

Selecting Subsets of Instance

The robot selects subsets of instances $\mathcal{V}$ in order to calculate a probability and an entropy according to the states $S_o(\tau-1), ..., S_o(\tau-h)$ observed in the past *h* steps. For each subset U_j the robot counts the number of substates which satisfy $P_{U_j}(S_o(\cdot)) > 0$ in *h* substates. If the count C_j is greater than a threshold C_{th}, U_j and U_{j+1} are added to $\mathcal{V}$. If $C_j = 0$, the robot uses all instances ($\mathcal{V} = U$).



(a) Feature extractors (b) Tasks in which the robot learned feature extractors

Figure 2: Feature extractors which are generated in the past tasks

EXPERIMENT

Experimental Setting

We used a small mobile robot which is about 40 cm high and has a camera with a fixed orientation to look ahead at the floor. The task is to move along a given path to a destination. The size of I_o and I_f in pixels is 64 x 54 and that of I_c is 8 x 6. We defined the dimension of a substate as $m = 1$. The robot can move at a translational speed v and a steering speed ω independently. To reduce the computation cost, we discretized the state and action space and calculated the probabilities. We set the history length to $h = 10$. The thresholds are set to $H_{th} = 0.4$ and $C_{th} = 0.8h$.

The robot was given three feature extractors shown in Figure 2. Fs, Fc, and Fm are generated in tasks A, B, and C, respectively. The feature extractors have the following characteristics.

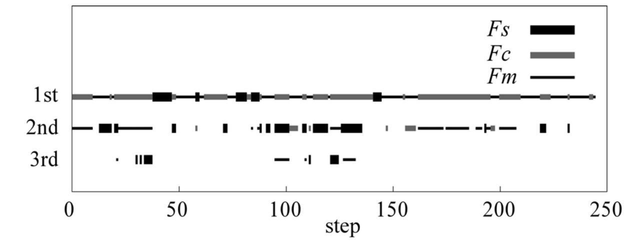
- 3 x 3 spatial filter Fs: This type of filter calculates sum of weighted brightness values of nine neighbouring pixels. The generated filter emphasizes and inhibits horizontal edge.
- Color filter Fc: This type of filter calculates sum of weighted red, green, and blue. The generated filter inhibits red.
- Spatial filter Fm: This type of filter calculates sum of weighted past five images. The generated filter emphasizes the current image and inhibits the past image.

Feature Extractor Selection

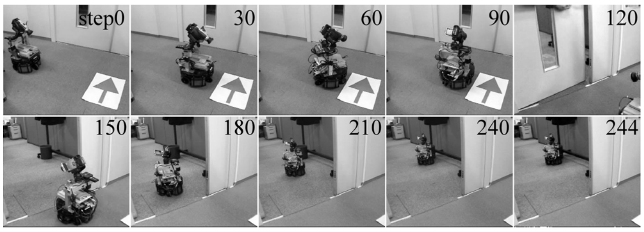
The task given to the robot is shown in Figure 3(a). The robot moves to the front of the door and waits for it to open. It moves to the destination after the door opens. The environment is the same as that of tasks A, B, and C. We gave three episodes of successful instances ($L = 234, 254, 233$). After learning, the robot divided all instances into 13 subsets using ISODATA algorithm.

Figure 3(b) shows the learned behavior and Figure 3(c) shows the selected feature extractors at each time step. The selected feature extractors differ depending on the situation. The average number of selected feature extractors per step is 1.57. Figure 3(d) shows the selected subsets of instances at each step. When the robot could not choose an action from the selected subsets because of low reliability, it used all instances to decide again. $\mathcal{V}$ in the figure shows the step when the robot could choose an action from the selected subsets. It is verified that the robot accomplishes the task while selecting

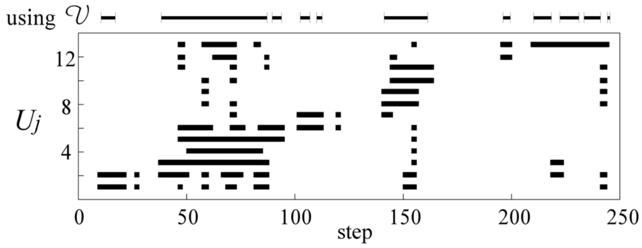
(a) Navigation task



(c) Selected feature extractors



(b) Resultant behavior



(d) Selected subsets of instance

Figure 3: Task and experimental results

effective feature extractors.

To verify the subset of instances, we performed the same experiment except for the procedure to select the subsets. In this experiment, the robot always selected all instances. In the result, the average number of selected feature extractors per step is 1.97, which is larger than the result of Figure 3. This means that the robot spent much more time for action decision at each step. Hence, the robot effectively decides the action using a portion of the instances.

CONCLUSION

This paper has proposed a method in which a robot learns to select image feature extractors generated by itself according to a task-relevant criterion. A portion of supervised data which gives the local information of the task makes the selection of feature extractors more effective. In the proposed method, a robot can accomplish more complicated tasks using multiple feature extractors. Our future work is to verify the extent of effectiveness of the proposed method.

References

McCallum, A. (1996). Learning to Use Selective Attention and Short-Term Memory in Sequential Tasks. *Proceedings of International Conference on Simulation of Adaptive Behavior*, 315-324.

Minato, T. and Asada, M. (2003). Towards Selective Attention: Generating Image Features by Learning a Visuo-Motor Map. *Robotics and Autonomous Systems* **45**, 211-221.

Mitsunaga, N. and Asada, M. (2000). Observation Strategy for Decision Making based on Information Criterion. *Proceedings of International Conference on Intelligent Robots and Systems*, 1038-1043.

Vlassis, N., Bunschoten, R., and Kröse, B. (2001). Learning Task-Relevant Features from Robot Data. *Proceedings of International Conference on Robotics and Automation*, 499-504.