Acquisition of Body and Object Representation Based on Motion Learning and Planning Framework

Takahiro Asamizu and Yuichi Kobayashi*
Tokyo University of Agriculture and Technology / RIKEN RTC*
Department of Electrical and Electronic Engineering
2-24-16 Nakacho, Koganei, Tokyo, JAPAN
yu-koba@cc.tuat.ac.jp

Abstract

Vision information processing is important for robots that act in human-interactive environments. In this paper, we propose to acquire visual representation of robot body and object that is suitable for motion learning in a bottom-up manner. An advantage of the proposed framework is that it does not require specific hand-coding depending on the visual properties of objects or the robot. A subtraction technique and SOM are used to compose the state space based on the image with extracted robot body and objects. Motion of the robot is planned based on reachable set. The task of moving an object to a target position is divided into two phases, one to reach a position that is suitable for starting pushing motion and the other to push the object to the target. The proposed method is verified by experiment of pushing manipulation of an object with a robot arm.

1. Introduction

Improvement of vision information processing is one of the problems for robots that act in human-interactive environment. One of the difficulties in image processing of those robots is that the applicability of the robot might be restricted when the image processing methods assume specific properties of the objects, such as colors and shapes. In other words, it is important to avoid ‘hand-coded’ processing by human designers to widen the applicability of the robot. One approach to automatic recognition of object is known as generic object recognition [9]. The generic object recognition aims at enabling a computer to recognize objects in images with their category names, which is one of ultimate goals of the computer vision research. Some approaches on autonomous image processing including the generic object recognition are independent of robot motion generation. From the viewpoint of development of robotics, it is important to consider how the results of image processing can be utilized to robot motion generation. Cognitive developmental robotics [1] is a promising approach to consider combination of motion generation and autonomous image processing. Referring the human developmental processes, one can get an idea that autonomous recognition of objects for robots might be realized through concurrent development of robot motion learning and image features acquisition.

There are some approaches which aim body image acquisition. As an approach of cognitive developmental robotics, Fuke et al. proposed an acquisition method of the body image with tactile sensors [4]. The relation between the hand position and the sensors on the robot face is obtained using Self Organizing Map (SOM) [1]. However, how to generate motion based on the body representation was not discussed. Stoytchev realized recognition of robot body in video image using synchronousness of action command and motions of markers equipped with the robot arm [7]. However, the designer gave knowledge of the robot body by using the marker in the robot.

On the other hand, there are approaches to acquire object representation autonomously as well as the robot body. Fitzpatrick et al. proposed to learn how objects move when a robot arm pushes them by extracting the object and the robot arm from the image [3]. This research realized to extract the object autonomously without teaching the shape and size of the object by the designer. Hikita et al. realized learning of relation between image features and a robot arm postures without hand-coded image processing using saliency map [10]. Kato et al. proposed a framework of manipulation learning that can reduce the amount of calculation of the image processing by adjusting the resolution of the image by clustering and principal component analysis [6]. Ridge et al. proposed to learn distinctions of the kinds of objects by pushing with a robot arm [2]. In

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.112

their approach, the affordance of the object was acquired by extracting how the objects can be moved by pushing with the robot arm. Compared with those approaches, we focus more on motion generation including planning, while object representations are autonomously acquired (with single kind of object).

In this paper, a state space construction is proposed with autonomous extraction of robot body and an object. The acquisition of object representation is connected to motion planning and generation framework, which realizes concurrent learning of object manipulation and building image representations of robot body and the object.

The paper is organized as follows. In section 2, problem settings are introduced. The proposed method for motion generation with image representation acquisition is described in section 3. Section 4 describes verification in experiments, followed by conclusion in section 5.

2. Problem settings

The system is composed of a camera and a robot arm, where the camera is fixed. The robot's movements are restricted to the vertical plane. It is assumed that the robot can detect the contact between a certain part of the robot body and an obstacle.

The task for the robot is to move an object to a desired position. The followings are unknown information to the robot, i.e., not utilized by the robot in the proposed framework.

- Colors, shapes and sizes of the components of the robot arm and the object.
- The distinction among objects, obstacles and parts of the robot arm.
- The dynamics of the object caused by the contact with the robot arm in the image.

Instead of using image features of the robot arm and the object, we use synchronousness of motion in image and motor command to the robot to extract them. For detecting the synchronousness, the subtraction technique is used in this research. An advantage of the subtraction technique is simplicity. A limitation of this approach is that the subtraction technique assumes that the object to be extracted has some texture. Therefore it is impossible to apply this method when the robot arm or the object has no texture (uniform surface). It is also assumed that the posture and the shape of the object do not change, i.e., the object does not rotate and it is a rigid body, and the case with multiple objects is omitted

An advantage of the proposed method is that the robot constructs object representation (state space) autonomously without specific knowledge on the properties of the object image and acquires controllers for manipulating the object based only on actual motion experiences.

3. Motion planning with acquisition of image representation

The four procedures of the proposed framework are described in the followings.

1. Initially the robot moves its arm randomly. During random operations, the robot arm is extracted as concurrently moving regions from the image as depicted in Fig.1.
2. The object is extracted by excluding the robot's body in the image, when the robot arm contacts with the object (see Fig.2). Joint angles of the robot arm are memorized, if the robot arm does not move (see Fig.3).
3. When the object moves by contact with the robot arm, the robot memorizes sequences of the joint angles and the images for reachable set estimation.
4. By using the reachable set, the robot performs the motion planning and motion generation to achieve the manipulation task.

The data sets which realize these procedures are described in the followings.

Robot body image
: Images of the robot arm with various postures are memorized.

Object image map
: An object is extracted while robot explores contact area with the object. The result of extraction is stored in SOM.

Obstacle joint angles set
: The joint angles are memorized when the robot arm does not move by contact with the objects.

Object reachable set
: Reachable set is estimated by memorizing the sequences of the object positions expressed by the object image map and the joint angles of the robot arm.

In the proposed approach, the task is divided into two phases. In the first phase the robot plans where to touch the object based on **object reachable set** and generates reaching motion to the target contact position without colliding with obstacles. **Obstacle joint angles set** is used for this collision avoidance. In the second phase, the robot moves its arm so that the object reaches the target. Here joint angles contained in **object reachable set** are utilized to realize the robot arm motion.

The following sections illustrate **object image map**, **obstacle joint angles set**, **object reachable set** and the motion planning in the two phases with those data sets.

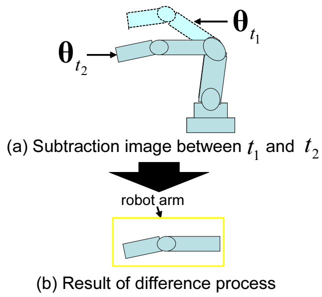

Figure 1. Extraction of the robot arm

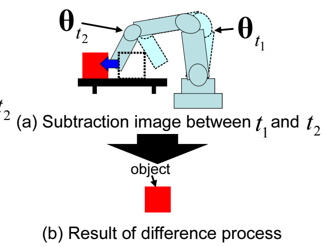

Figure 2. Extraction of the object

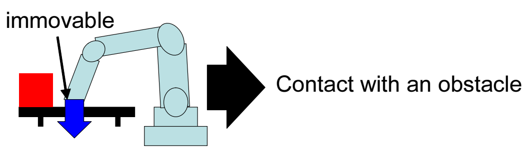

Figure 3. Extraction of the obstacle

3.1. Object image map

The images of the object extracted by the subtraction technique are used to construct a state representation (state space) of the object, which will be used in learning of manipulation. The method to extract objects is shown in the followings.

1. Let $\boldsymbol{\theta}(t)$ and $I(t)$ denote the joint angle and image input at time t, respectively, where $\boldsymbol{\theta}$ denotes the joint angles of the three degree freedom and $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3]^T$. After a contact with the object occurs at t, two subtraction images are generated. One using $I(t-1)$ and $I(t)$ and the other with $I(t+1)$ and $I(t)$. Note that the object is supposed to have moved in the both subtraction images.

2. The object and the robot arm are extracted for time t by taking the logical product of the two subtraction object-arm images obtained by procedure 1.

3. The robot arm is excluded from the object-arm image by taking exclusive-OR between the image obtained by procedure 2 and the **robot body image**.

The robot arm moves at random until contacting with the object while searching the object. When the robot arm comes in contact with the object, the robot stops pure-random motion and sets a target joint angle that is close to the angle where contact has been occurred with a small disturbance. SOM with one dimensional topology is used to represent the object images. Let $\mathbf{W}_i \in \mathbb{R}^N (i = 1, ..., l)$ denote the weight vector of i-th node, where $N = x \times y$ and x and y are the sizes of the image.

3.2. Obstacle joint angles set

The robot arm explores the space randomly and stores images and joint angles when contact with the obstacles such as tables, floors and object[1] occurs under the assumption that it can detect contact with the obstacle. Let $\mathbf{O}$ denote the **obstacle joint angles set**. When the robot detects that the robot arm is contacting with an obstacle at time t, $\mathbf{O}$ is updated as

$$\mathbf{O} \leftarrow \mathbf{O} \cup \{[\boldsymbol{\theta}(t)^T, I(t)^T]^T\}. \tag{1}$$

In motion planning which will be described later, negative reward for contact between the robot arm and the obstacle is generated by $\mathbf{O}$.

3.3. Object reachable set

The **object reachable set** is used to estimate where the object can reach by manipulation of the robot arm. Let k denote the node number which corresponds to the current input image as

$$k(t) = \underset{i}{\operatorname{argmin}} \, ||\mathbf{W}_i - I(t)||. \tag{2}$$

Suppose that contact with an object occurred at time t and $(n-1)$ matrices have been stored in the **object reachable set**, that is, n-th matrix which describes the reachable region will be added. After a sequential movement of the robot arm, suppose that contact has been kept until $(t+m_n)$. The **object reachable set** is defined as $\mathbf{L}$ and updated as

$$\mathbf{L} \leftarrow \mathbf{L} \cup \mathbf{L}_n, \tag{3}$$

where $\mathbf{L}_n$ is defined as

$$\mathbf{L}_n = \left[\begin{array}{cccc} \boldsymbol{\theta}(t) & \boldsymbol{\theta}(t+1) & \cdots & \boldsymbol{\theta}(t+m_n) \\ k(t) & k(t+1) & \cdots & k(t+m_n) \end{array} \right]. \tag{4}$$

Fig.4 shows the idea of the **object reachable set**. Note that the **object reachable set** contains the information of transition. Arrows in the figure indicate the unilateral transitions which corresponds to the order of vectors in matrices $\mathbf{L}_p$.

3.4. Online: motion generation

To achieve the task, the task is divided into two phases. In the first phase, the robot plans a path from the initial position of the robot arm to a position that is suitable for pushing the object and moves the robot arm along the path (see

[1]Note that the situation where the robot arm contacts with the upper side of the object is regarded as collision with an obstacles, because the robot arm can not move downward in such case.

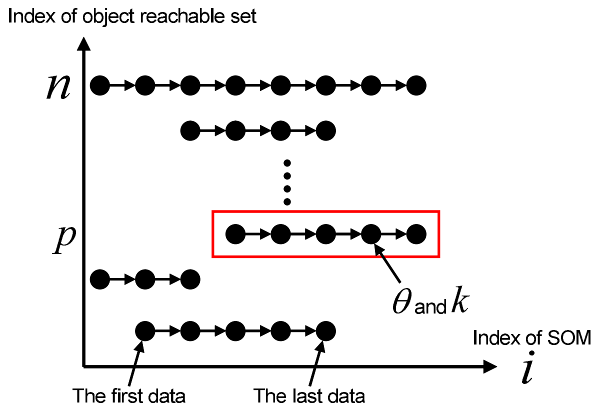

Figure 4. Object reachable set

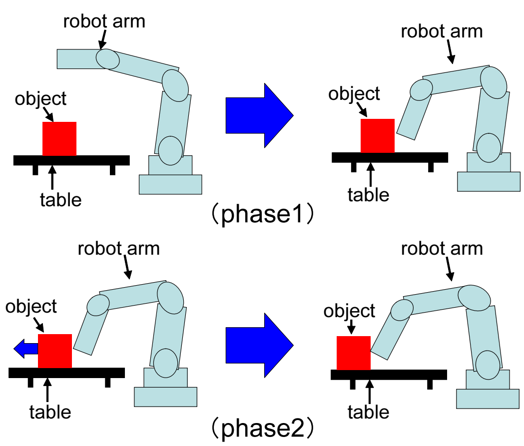

Figure 5. Motion generation of the robot

phase1 of Fig.5). The phase is switched from phase one to two when the robot arm reaches the suitable position for pushing. In the second phase, the robot pushes the object to the target position (see phase2 of Fig.5).

3.4.1 Phase1: path planning of the robot arm

Fig.6 shows the flow of processing in the first phase. The minimum norm node $k(t)$ is decided by (2), which corresponds to recognition of the current position of the object. Similarly k_{goal} is obtained by I_{goal}, where I_{goal} is the image with the object at the target position. A trajectory can be found in the **object reachable set** as

$$\mathbf{L}_{p^*} = \begin{bmatrix} \boldsymbol{\theta}_1 & \boldsymbol{\theta}_2 & \cdots & \boldsymbol{\theta}_{(m_{p^*}+1)} \\ k_1 & k_2 & \cdots & k_{(m_{p^*}+1)} \end{bmatrix}, \quad k_1 = k(t), {}^{\exists}k_j = k_{goal} \tag{5}$$

where p^* denotes index of trajectory that reaches k_{goal} starting from $k(t)$. $\boldsymbol{\theta}_1$ in $\mathbf{L}_{p^*}$ is set as a subgoal to start the pushing operation. The subgoal information is interpreted as a positive reward in the reinforcement learning framework. The positive reward is defined as

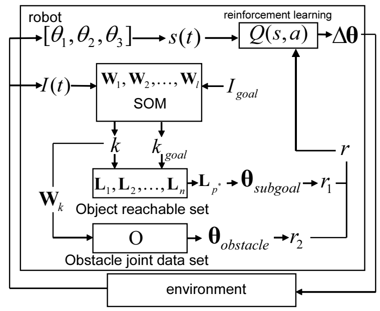

Figure 6. Construction in phase1

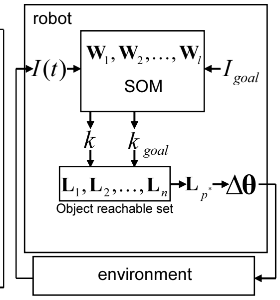

Figure 7. Construction in phase2

$$r_1(\boldsymbol{\theta}) = \begin{cases} R_{goal} \geq 0 & \text{if } ||\boldsymbol{\theta}_1 - \boldsymbol{\theta}|| \leq \beta_1 \\ -1 & \text{otherwise} \end{cases}. \tag{6}$$

where r_1 denotes the positive reward and β_1 is a threshold value. For motion planning for obstacle avoidance, a set of joint angles with collision is defined as

$$\mathbf{O}_{\boldsymbol{\theta}(t)} = \left\{ \boldsymbol{\theta}_i \middle| ||I_i - \mathbf{W}_k(t)|| \leq \beta_I, [\boldsymbol{\theta}_i^T, I_i^T]^T \in \mathbf{O} \right\}, \tag{7}$$

where β_I is a threshold value. The negative reward is defined as

$$r_2(\boldsymbol{\theta}) = \begin{cases} R_{obst} < 0 & \text{if } {}^{\exists}\boldsymbol{\theta}_i \in \mathbf{O}_{\boldsymbol{\theta}(t)}, ||\boldsymbol{\theta}_i - \boldsymbol{\theta}|| \leq \beta_2 \\ 0 & \text{otherwise} \end{cases}, \tag{8}$$

where β_2 is a threshold value. The total reward is given as $r = r_1 + r_2$ and used for reinforcement learning. Dyna-Q [8] is applied for motion planning. The action value function update is the same as Q-learning. State s for reinforcement learning is defined by discretizing $[\theta_1, \theta_2, \theta_3]$. Action a denotes a small displacement of $[\theta_1, \theta_2, \theta_3]$. The update of the action value function in Dyna-Q is given as

$$Q(s, a) = (1 - \alpha)Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a')]. \tag{9}$$

α and γ are learning coefficient and discount factor, respectively. s' is the next state when action a is taken.

3.4.2 Phase2: object pushing

In the second phase, the robot refers $\mathbf{L}_{p^*}$ and follows the sequence of joint angles in $\mathbf{L}_{p^*}$ until $\boldsymbol{\theta}(t) = \boldsymbol{\theta}_j$, where $k_j = k_{goal}$.

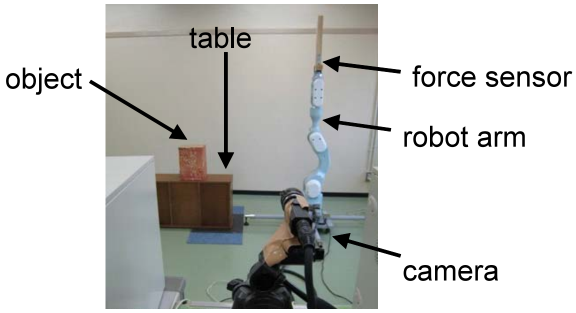

Figure 8. Experimental setup

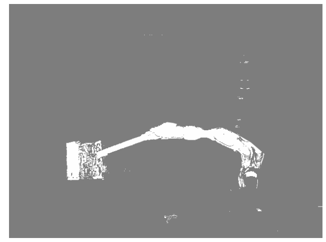

Figure 11. Image of the robot and the object

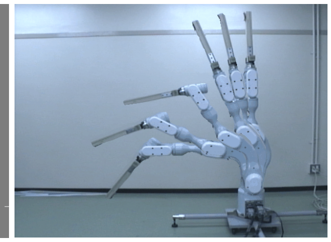

Figure 12. Image of the robot

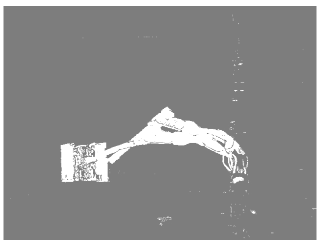

Figure 9. Subtraction image of $(t-1)$ and t

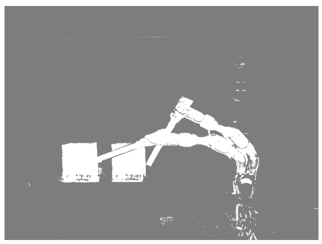

Figure 10. Subtraction image of t and $(t+1)$

Figure 13. Image of the extracted object

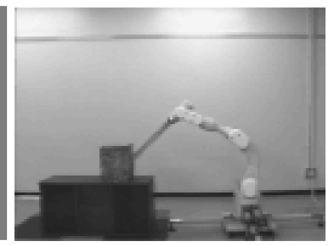

Figure 14. Image of the object and the robot

4. Experiment

The experimental setup for the verification of the proposed learning method is shown in Fig.8. All experiments were performed using the robot arm (Mitsubishi PA-10) and a CCD camera (640×480, gray scale). The robot has six degrees of freedom and controls three joints while fixing three other joints. The robot detects contact with objects or obstacles using the information of the force sensor installed at the wrist of the robot arm. The threshold value is set so that the case when the object notates is detected as collision with an obstacle.

4.1. Extraction of body and object

The images obtained by the subtraction technique with time $(t-1)$ and time $(t+1)$ for the image at t are shown in Fig.9 and Fig.10. Fig.11 shows the result of extraction of the object and the robot arm. Some images of the **robot body image** are shown in Fig.12 with various postures. Fig.13 shows an example of the extracted object by the proposed procedure. An example of the image in the **object reachable set** is shown in Fig.14 .

4.2. Generation of object image map

The number of nodes of SOM is set as $l = 25$. Some weight vectors in SOM are shown in Fig.15. It can be seen that close nodes in SOM have similar image vectors. The maximum displacement of the object in the image is 190[pixel](575[mm]), which implies that one node in SOM corresponds to 7.6[pixel]. Fig.16 shows the minimum norm node for three images inputs. The obtained SOM could identify images with different object positions.

4.3. Planning and motion generation for pushing manipulation

The data number of the **object reachable set** is set as $n = 20$. Fig.17 shows a trajectory of reaching and pushing manipulation realized by the proposed planning and control method. The number of the images that are collected for the experiment are described in the following.

- The number of the images of the **robot body image**

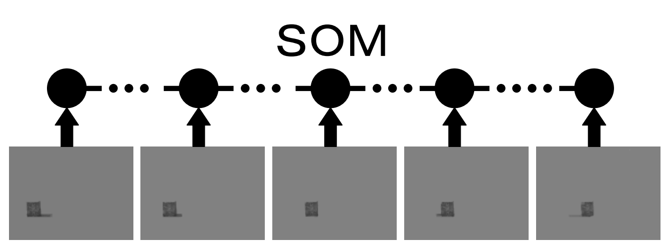

Figure 15. Result of learning of SOM

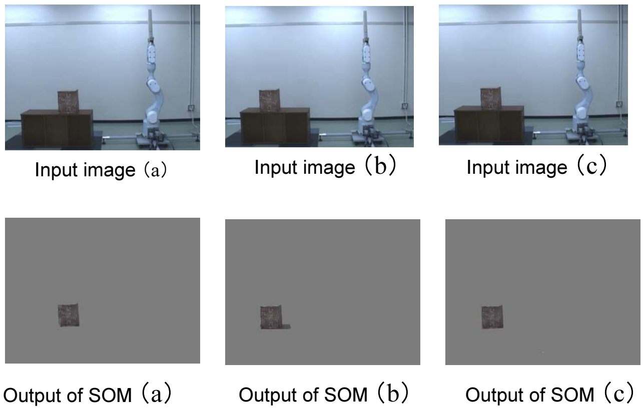

Figure 16. Identification of object position with SOM

are 9788 (with range of $0 \le \theta_1 \le 60$, $0 \le \theta_2 \le 120$, $0 \le \theta_3 \le 150$).

- The number of the images which are collected for generation of the **object image map** are 30.
- The number of the joint angles of the **obstacle joint angles set** are 30.
- The number of the matrices which are collected for generation of the **object reachable set** are 25.

5. Conclusion

This paper proposed a method for motion planning and generation of the object manipulation that does not require any specific hand-coded image processing. The proposed framework integrates state representation by subtraction and SOM, estimation of the object reachable set and motion generation learning with reinforcement learning. The state space is constructed by extracted images of the robot body and the object. In the experiment, the robot realized pushing manipulation of the object toward a target position including appropriate reaching motion toward the object. In the experiment, pushing from right to left was realized. It is possible to learn pushing of the opposite direction, which was not implemented due to the limitation of the work space of the robot arm. One of the future works is that the framework is extended to the case with various objects. In addition, it will be important to effectively explore the space where 'curious' behavior can be observed, that is, effective identification of robot-object interaction dynamics.

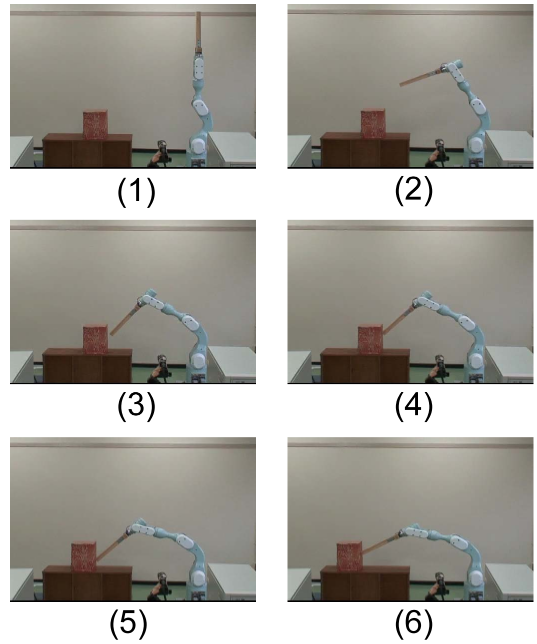

Figure 17. Experimental result

References

[1] M. Asada, K. F. MacDorman, H. Ishiguro, and Y. Kuniyoshi. Cognitive developmental robotics as a new paradigm for the design of humanoid robots. *Robotics and Autonomous System*, 37:185–193, 2001.

[2] B.Ridge, D.Skocaj, and A.Leonardis. A system for learning basic object affordances using a self-organizing map. *International Conference on Cognitive Systems CogSys 2008*, April 2008.

[3] P. Fitzpatrick. Learning about objects through action - initial steps towards artificial cognition. *Proceedings of IEEE International Conference on Robotics and Automation*, pages 3140–3145, 2003.

[4] S. Fuke, M. Ogino, and M. Asada. Body image constructed from motor and tactle images with visual informaiton. *International Journal of Humanoid Robotics*, 4(2):347–364, 2007.

[5] M. Hikita, S. Fuke, M. Ogino, and M. Asada. Cross-modal body representaion based on visual attention by saliency. *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2041–2046, 2008.

[6] M. Kato, Y. Kobayashi, and S. Hosoe. Optimizing resolution for feature extraction in robotic motion learning. *IEEE Int. Conf. on Systems, Man and Cybernetics*, pages 1086–1091, 2005.

[7] A. Stoytchev. Toward video-guided robot behaviors. *In Proceedings of the Seventh International Conference on Epigenetic Robotics (EpiRob)*, pages 165–172, 2007.

[8] R. S. Sutton. *Reinforcement Learning*. MIT press, 1998.

[9] K. Yanai. Current state and future directions on generic object recognition. *IPSJ SIG Technical Reports*, 2006(93):121–134, Novemver 2006.

[10] Y.Ber-Cohen and C.Breazeal. *Biologically Inspired Intelligent Robots*. SPIE Press, 2003.