Embodiment based Object Recognition for Vision-Based Mobile Agents

Kazunori Terada†, Takayuki Nakamura, Hideaki Takeda and Tsukasa Ogasawara

Dept. of Information Systems, Nara Institute of Science and Technology
8916-5, Takayama, Ikoma, Nara 630-0101, Japan
E-mail: kazuno-t@is.aist-nara.ac.jp†

Abstract

In this paper, we propose a new architecture for recognizing objects based on a concept "embodiment" as one of primitive functions for a cognitive robot. We define the term "embodiment" as the extent of the agent's body, locomotive ability and its sensor. According to embodiment, an object is represented by reaching action paths, which correspond to a set of sequences of movements taken by the agent for reaching the object. Such behavior is acquired by the trial-and-error method based on the visual and tactile information. Visual information is used to obtain sensorimotor mapping which represents the relationship between the change of object's appearance and the movement of the agent. On the other hands, tactile information is utilized to evaluate the change of physical condition of the object caused by such movement. By means of this method, the agent can recognize an object without depending on its position and orientation in the environment. To show the validity of our method, we show an experimental result of computer simulation.

1 Introduction

Recognizing an object based on visual information is essential and useful function for the agent which can behave in the real world. In the computer vision area, so-called model-based approach is popular for recognizing an object based on the visual features. Since human designers always prepare the model of the object from the designer's viewpoint in such approach, it is ambiguous whether such model is suitable or not for the agent having the model-based object recognition system. Therefore, such model should be constructed by the agent itself from the agent's viewpoint.

Recently, there have been many researches regarding to the development of the intelligent agent which can automatically obtain the model of its environment or objects [6][4]. In their researches, the model is acquired by the agent itself through the interaction between the agent and its environment. In this sense, their approach is called "embodiment-based approach."

Embodiment-based approach has attracted interest in the field of cognitive science and artificial intelligence [5] [1]. However, there is no clear definition of "embodiment" in their researches. They regard "embodiment" just as body of the agent and what its body may constrain cognitive internal architecture of the agent. In most of these studies, proximity sensors, such as bumpers and sonar, are used to sense the environment. Therefore, the agent with such sensors has to move around an object or its environment in order that such agent may recognize the object or the environment. As a result, the tasks in such cases are limited to local, reflexive, and simple tasks. In order that the agent may accomplish more complicated tasks in the real world, the use of visual information seems to be indispensable for such agent. However, the use of vision in the conventional embodiment-based approach is very rare due to the high costs of sensing and processing or because the visual information is not considered to be its importance.

To deal with this problem, we propose an embodiment-based visual object recognition method for a visually guided agent. According to embodiment, an object is represented by reaching action paths, which correspond to a set of sequences of movements taken by the agent for reaching the object. Such behavior is acquired by the trial-and-error method based on the visual and tactile information. In our method, such visual information is used to obtain sensorimotor map-

ping which represents the relationship between the change of object's appearance and the movement of the agent. On the other hands, tactile information is utilized to evaluate the change of physical condition of the object caused by such movement. By means of our method, the agent can recognize an object without depending on its position and orientation in the environment. To show the validity of our method, we show an experimental result of computer simulation.

2 Embodiment and Vision

We define the term "embodiment" as the extent of the agent's body, locomotive ability and its sensor. The extent means how and how much the agent's body occupy in the physical space, that is the shape and size of agent's body. In this sense, although locomotive ability is commonly used to model the environment [6][4], we argues that the extent of an agent also plays very important role in modeling it. It is because the representation of its environment depends on the extent of the agent's body. For example, a chair is an instrument to sit for a human, but it does not have any meaning of tool to sit for an ant which has different embodiment. In other words, the significance of existence of the object depends on the embodiment of the agent. Therefore, the embodiment of the agent can be used to represents the relationship among its body, behavior and environment.

Most of researchers on visually guided agents believe that vision is the most essential modality to recognize environment, because vision is useful for the agent to recognize its environment. However, vision is not primary information for cognition of the environment, because an agent can not perceive any information about physical world directly by visual sensors. Philosophical and clinical medicine's findings support this property of vision. Berkeley [2], an English philosopher, suggested (1)that the object of sight have nothing in common with the object of touch, and (2)that the connection of sight and touch is arbitrary and learned by only experience not by calculation. These suggestion are supported by the clinical medicine's findings, which are case studies for behavior patterns and visual rehabilitation after successful operations for congenital blind patients. For example, let us consider a patient whose eyes functions well at the beginning of his life but not after that. Such patient can not recognize anything in the world with his eyes as if the function of eyes are completely cured. It is because there is no correspondence between visual and tactile inputs. These findings suggest that vision is not useful for cognition of the object until the relationship between visual and tactile features is acquired through its experience. After such correspondence is acquired, vision can be used to predict the change of physical condition of the object caused by the movement of the agent.

Based on these suggestions, we propose an method of embodiment-based visual object recognition method. The details of our method is described in the next section.

3 Embodiment-based Visual Object Recognition

The purpose of this system is to acquire knowledge to interpret visual image, i.e., knowledge to discriminate objects by visual information. To realize it, we provide three assumptions as agent's features. The first assumption is that the agent can recognize objects by locomotive experience. An object is recognized as an area where the agent can not enter, and it is achieved by the agent's ability of moving and perception of touching. The second assumption is that the agent can understand the event of contact both physically and visually, that is, the agent can associate tactile information and feature of image locally. The third assumption is that the agent can associate its movement and change of image. We restrict that the latter two are the only abilities for the agents to interpret images. We introduce a reaching action path as locomotive experience. A reaching action path means an optimal path, which moves a point on agent's body to a point on boundary of physical objects. The first assumption tells that collection of reaching action paths can be motional representation of objects. The second assumption assures that the agent can generate a reaching action path. The third assumption provides how to generate reaching action paths by visual information. As a result the agent can understand objects by visual information.

3.1 Overview of Our Method

Objects are considered as an area in which an agent can not exist. In other words, objects are represented by paths each of which terminal is perceived by tactile information. We call these paths as reaching action paths. The agent can discriminate objects by means of reaching action paths because they reflect the shape of objects. In order to represent an object by reaching action paths, we extract several features from them. We call these features as a shape characterizing vector.

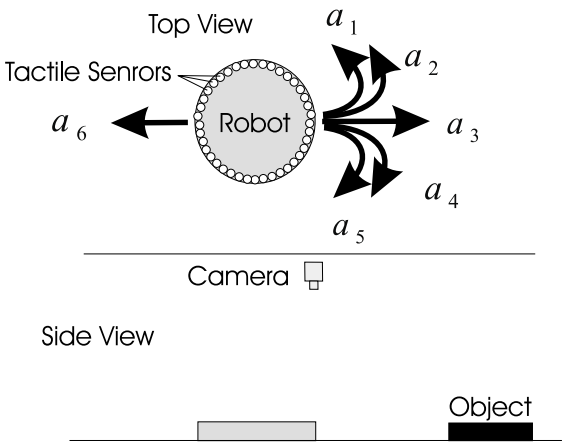


Figure 1: Assumed agent and environment.

On the other hand the agent can not receive any information directly about an environment by vision. In order to recognize the physical world by vision, the visual information should be concerned to tactile information, which is an only modality to receive physical information directly. If the agent knows the visual features which associates with tactile information, it can generate reaching action paths mentally. In our method the relation is learned by means of dynamic programming.

3.2 Assumed Agent and Environment

We assume that the dimension of an agent and an environment is 2, that is, the agent can move only on the 2D plane. Figure 1 shows the agent and the environment assumed in our work. The shape of the agent is circle and the agent is equipped with tactile sensors around its body. We put a camera over the environment as its eye so that the agent can see both its body, object and contact state. We also assume that the agent can generate 6 action commands. We define an action unit as a segment of the agent's movement until a change of state is observed. The agent also has a gyro sensor so that it can perceive rotation of body.

4 Reaching Action Path

A reaching action path means an optimal path, which starts at a point on agent's body and ends at a point on boundary of physical object.

Suppose a point $p_l^b \in \boldsymbol{R^2}$ on the surface of an agent's

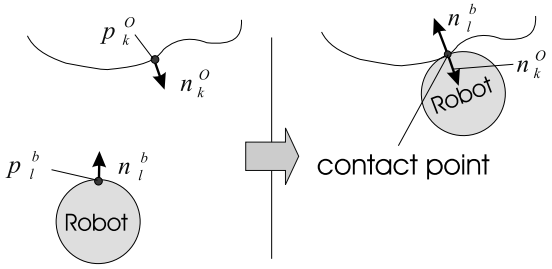


Figure 2: Contact condition.

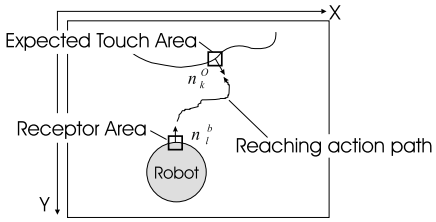


Figure 3: An input image and small areas.

body and a point $p_k^o \in \boldsymbol{R^2}$ on the surface of a physical object (Figure 2). When the agent's body is touching the object, a tangential line of the agent's body coincides with that of the object at a contact point. In other words, the point on agent's body coincides with that of the object, and a normal vector of the agent's body and object have the same size and the opposite direction. This relation is expressed as following equation:

$$p_l^b = p_k^o \tag{1}$$

$$\boldsymbol{n}_l^b = -\boldsymbol{n}_k^o \tag{2}$$

where $n_l^b \in \boldsymbol{R^2}$ is a normal vector on the agent's body at the point p_l^b and $n_k^o \in \boldsymbol{R^2}$ is a normal vector on the object at the point p_k^b.

Next, we explain a method for learning the reaching action path using visual input. Consider an input image like Figure 3 which includes both the agent's body and an object. We treat an input image as a set of several small areas. The small area may include the surface of agent's body and/or the object. We call an area which includes the point of agent's body as Receptor Area (RA), and an area which includes the point of object as Expected Touch Area (ETA). An expected touch area indicates an area in which physical contact will be observed after certain action series, i.e., reaching action. We define the coordinates of expected touch area as (x^{ETA}, y^{ETA}) and an angle of

a normal vector as θ^{ETA}, the coordinates of receptor area as (x^{RA}, y^{RA}) and, a normal vector as θ^{RA}. In a goal state of reaching action path, the state of each area should be as follows;

$$x^{ETA} = x^{RA}, y^{ETA} = y^{RA}, \theta^{ETA} - \theta^{RA} = \pi \quad (3)$$

We use dynamic programming as a learning method for the optimal policy to generate an optimal reaching path. Given an utility function U and if a state transition holds Markov property, optimal policy for Markov decision problem is calculated as follows;

$$f(i) = \arg\max_a \sum_j M_{ij}^a U(j) \quad (4)$$

where M_{ij}^a is the probability of reaching state j if action a is taken in state i, and $\arg\max_a f(a)$ returns the value of a with the highest value for $f(a)$. The utility of a state can be expressed in terms of the utility of its successors:

$$U(i) = R(i) + \max_a \sum_j M_{ij}^a U(j) \quad (5)$$

where $R(i)$ is a reward function which returns a value of reward in state i. In our work, a reward is given when the point of agent's body touches to an object, and M_{ij}^a indicates the transition probability of receptor area and it is obtained through experiences of random action.

Although, as mentioned above, a reaching action path is defined such a path between a receptor area and an expected touch area, we can define multiple reaching action paths. If there are m expected touch areas on boundary of the object and n receptor areas on boundary of the agent's body, $m \times n$ reaching action paths are defined.

5 Characterizing Objects by Embodiment

In order to represent objects by agent's embodiment we use reaching action paths. Reaching action paths are calculated depending on the variety of a size and shape of agent's body. In this section we explain methods to represent the physical properties of object by means of reaching action paths. Physical properties mean pose, i.e., position and orientation, and shape of an object. Before explaining these methods precisely, we explain a method to represent the reaching action path.

Table 1: Example of Relation between action and code.

action	a_1	a_2	a_3	a_4	a_5	a_6
code	-2	-1	0	1	2	0

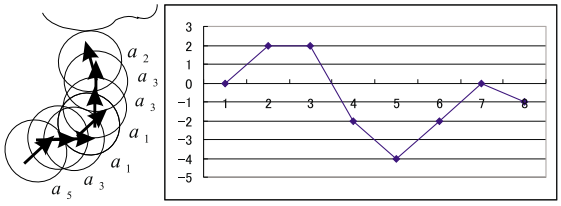


Figure 4: Chain coding.

5.1 Representation of a Reaching Action Path

In order to represent a reaching action path, we utilize a chain coding which is a popular coding technique in the fields of image processing and shape analysis [7]. In our method, a code indicates an angle of rotation of a motor command. Table 1 shows an example of relation between action and code. The ratio between code values is similar to that of rotation angle. However, the moved distance and the angle of rotation corresponding to the action are indeterminate, since one action terminates when change of state is observed. In order to overcome this problem, we adopt an average value of rotating angle to calculate the ratio. Figure 4 shows a summation of code value in each time step corresponding to an action series of reaching action. The summation of the code value indicates a relative angle to the starting point, and the length of the chain code indicates the moved distance of the agent. Consider an action series of a reaching action $a_1, a_2, \ldots a_u$ and a chain code corresponding to the action series $\boldsymbol{c}= \{c_1, c_2, \ldots, c_u\}$, the summation of chain code C is represented as follows:

$$C = \sum_{i=1}^{u} c_i \quad (6)$$

and the length of the chain code L is:

$$L = u \quad (7)$$

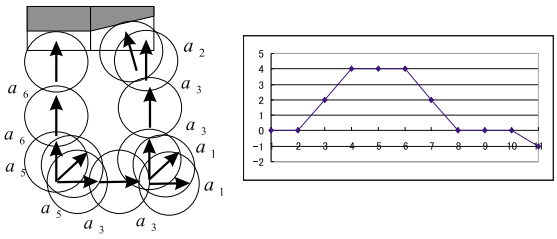


Figure 5: Local haptic motion.

5.2 Representation of Position and Orientation

A position and orientation of a part of an object relative to an agent's body are represented by distance and angle between both surfaces respectively. The distance is represented as the length of the chain code L approximately if an optimality of reaching action path is guaranteed. The angle is the relative one of the agent's surface between the starting point and the goal point (touching the object), and it is calculated as a code value.

If there are m expected touch areas on boundary of object and n receptor areas on boundary of the agent's body, the orientation of object is represented by the following matrix:

$$Orientation = \begin{bmatrix} C_{0,0} & C_{0,1} & \cdots & C_{0,n} \\ C_{1,0} & C_{1,1} & \cdots & C_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{m,0} & C_{m,1} & \cdots & C_{m,n} \end{bmatrix} \quad (8)$$

5.3 Representation of Shape

Shape of an object indicates how the boundary contour of the object varies. The variation of the boundary contour can be perceived by a haptic motion. A local haptic motion is defined as an action series satisfying the following conditions.

1. The agent's body touches an object both at the starting and at the goal point.
2. The goal point is adjacent to the starting point.

We can calculate a reaching action path between starting and goal points, and a chain code of its action series. Figure 5 shows an example of a local haptic motion and a summation of code value $C = -1$ which represents a relative angle between the adjacent points on the object's boundary. The value of $|C|$ becomes larger in proportion to the relative angle of adjacent points.

In the example of Figure 5, $|C|$ can represent the relative angle directly because the same point of the agent's body touches the object both at the starting and at the goal point. If the different point of the agent's body touches the object at the goal point, $|C|$ does not represent the relative angle. In order to cope with this problem, we introduce C' which represents an action series that (1)the goal point of the physical object is similar to the starting point, and that (2)different points of the agent's body touch the object at the starting and at the goal point. As a result, in such situation, the relative angle is represented by $C + C'$.

If there are m expected touch areas on boundary of an object, the shape of the object is represented by $\boldsymbol{c}= \{c_1, c_2, \ldots, c_m\}$. We call this vector as a shape vector. The shape vector can represent the characteristics of the object's shape. For example, an object which consists of only straight lines and right angles, namely, rectangle, $\boldsymbol{C}$ can be $\{0, 0, 4, 0, 0, 4, 0, 0, 4, 0, 0, 4\}$, and in the case that the variation of the object boundary contour is regular like an circle, $\boldsymbol{C}$ can be $= \{1, 1, 1, 1, 1, 1, 1, 1, 1, 1\}$. Note that the shape vector does not change in spite of the change of position and orientation. Because the shape vector includes various information about shape, we should select appropriate features so that it can represent the object adequately. The primary feature is the length of $\boldsymbol{c}$ which corresponds to neighbor the circumference of an object, i.e., $f_1 = |\boldsymbol{C}|$. The feature of characterizing an object is represented by frequency analysis such as Fourier transformation. Consequently, $f_2, f_3, \ldots$ is the Fourier coefficient. We call $\boldsymbol{F}= \{f_1, f_2, \ldots\}$ as a shape characterizing vector.

5.4 Object Recognition

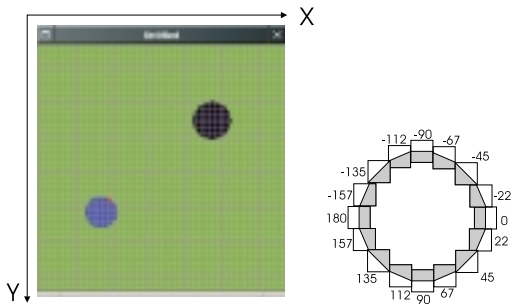
Object recognition is mainly divided into two procedures; (1)learning of the representation of objects, and (2)object recognition by this representation. The learning procedure is as follows:

- Estimation of transition probability.
- Collect examples of various shapes of objects from experience, and store shape characterizing vectors. Note that the agent can generate reaching action paths mentally, i.e., without actual motion, after the agent knows the transition probability.

Object recognition is accomplished by means of NN (nearest neighbor classification). Sample data of NN is collected examples of shape characterizing vectors.

Table 2: Objects used in our experiment.

name	shape	size
obj1	circle	radius 2.1
obj2	circle	radius 2.8
obj3	circle	radius 3.0
obj4	rectangle	3×5
obj5	rectangle	1.5×4
obj6	rectangle	3×3



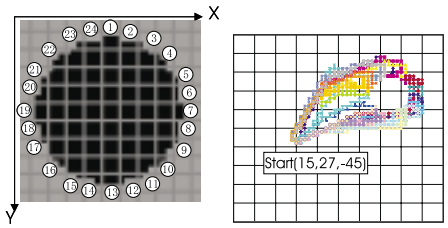
(a) (b)

Figure 6: (a)An input image and (b)angular discretization rule.

6 Experimental Results

In order to show the validity of our method, we had an experiment by computer simulation. In this experiment, the aim of the agent is to recognize objects from any position and orientation, and categorize them correctly. Table 2 shows objects used in our experiment. Note that the size of each object represents the size in the simulated world not in the input image. Figure 6-(a) shows an input image used in computer simulation. The size of image is 160×160 pixels and the size of expected touch area and receptor area is 3×3 pixels, therefore there are 53×53 possibilities for them. A circle on the left lower side in the image shows the body of an agent and another one on the right upper side shows an object (obj3). Figure 6-(b) shows the angular discritization rule applied to both the body of the agent and the object. The angle of a normal vector of the surface is discritized into 16 steps. As a result, the number of state is $53 \times 53 \times 16$.

In this experiment, we assume that the agent has only one contact point in front of the surface of its body.



(a) (b)

Figure 7: (a)Detected ETA and (b)calculated reaching action path

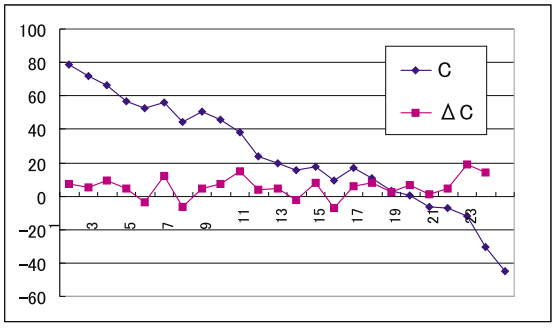


Figure 8: C and ΔC value.

6.1 Learning of the Representation of Objects

6.1.1 Transition Probability Estimation

Consider a state in an image $S_i(x_i, y_i, \theta_i)$, the possible number of the next state $S_{i+1}(x_{i+1}, y_{i+1}, \theta_{i+1})$ is $a \times m \times n$, where a is a number of actions, m is a number of adjacent cell in the image, and n is a number of angular discritization. We assume that transition probability from any (x, y) in the image is equal.

6.1.2 Calculation of Reaching Action Path

Next we calculate reaching action paths by dynamic programming using transition probability calculated above. Figure 7-(a) shows the detected 24 expected touch areas from Figure 6-(a). We put an agent on the starting point where detected receptor area is $(15, 27, -45)$. Figure 7-(b) shows the calculated 24 reaching action paths. We also show the C and ΔC value of each reaching action paths on Figure 8. ΔC means differ-

Table 3: Average of shape characterizing value.

name	f_1	f_2	f_3	f_4	f_5
obj1	16	-1.2	-1.2	0.9	0.7
obj2	22	-0.4	-0.4	-0.9	0.1
obj3	24	-0.6	0.2	0.3	0.1
obj4	24	-0.8	1.9	0	-1.4
obj5	16	-1.0	2.9	0	-2.3
obj6	16	-1.0	-1.1	-0.2	-1.4

Table 4: Object recognition rate.

	obj1	obj2	obj3	obj4	obj5	obj6	average
rate	0.75	0.91	0.75	0.91	0.83	1.0	0.85

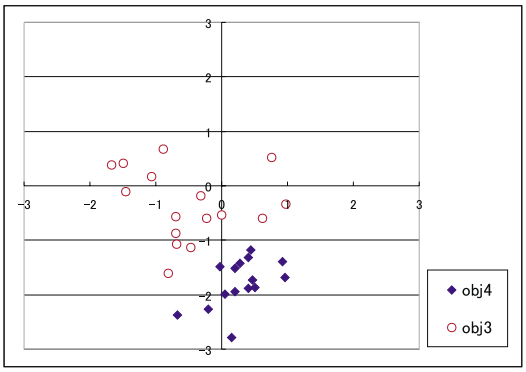


Figure 9: Shape characterizing value of f3-f4.

ence value of C between the adjacent reaching action paths. Although the agent should do haptic motion in order to obtain ΔC value exactly, we use the ΔC value relatively obtained from the same starting point instead.

6.1.3 Extraction of a Shape Characterizing Vector

We extract shape characterizing vectors from ΔC. Extracted features in this experiment are described as follows. f_1 is the length of ΔC i.e. $|\Delta C|$. $f_2 \dots f_5$ are Fourier coefficient extracted by discrete Fourier transform on ΔC;

$$\Delta C(n) = \sum X_k e^{-i\frac{k\pi n}{|\Delta C|}} \tag{9}$$

f_2 is a real part of X_1, f_3 is an imaginary part of X_1, f_4 is a real part of X_2, and f_5 is an imaginary part of X_2.

We provided four cases by changing the starting point, and calculated reaching action paths and shape characterizing vectors. Table 3 shows the average value of shape characterizing vectors for each object after 20 trials for each case.

6.2 Discrimination of Objects

We performed object discrimination tests using the extracted shape characterizing vectors above. We use NN(nearest neighbor classification) as discrimination method. We put each object of $obj1 \dots obj6$ on 12 points and extract shape characterizing vectors and performed object discrimination tests. Table 4 shows the average recognition rate of each object. The highest value is 1.0, the lowest 0.75, and the average 0.85. One of the reasons to fail recognition is fail in path generation. Since the agent may fail to reach objects because of the scattering of transition probability and as a result, a correct shape characterizing vector is not calculated.

Figure 9 shows the shape characterizing values of obj2 and obj4 used in the training and test. From this figure, we can see that divergence of an object is well formed, and that there are clear discrimination boundary. This implies that the agent can recognize objects from any position and orientation, and categorize it to the correct class.

6.3 Discrimination of Spatial Structure

We performed one more experiment for verifying the ability of our method. In this experiment, we check whether our method can discriminate spatial structure or not.

We put two rectangles with same shape (two obj4s) in the environment, where such two rectangles are located in two different ways: Under one situation, the distance between the two rectangles is narrow (see Figure 10). Hereafter, this situation is called "*narrow case.*"Under the other, such distance is wide (see Figure 11). Hereafter, this situation is called "*wide case.*" Then, we calculate the shape characterizing vector for each object (see Figure 12). The shape characterizing vectors of *wide case* coincide with that of one rectangle (one obj4). On the other hand, the shape characterizing vector of *narrow case* differs from that of one obj4. In *narrow case*, the agent would regards the two rectangles as another objects.

In this way, our method can recognize the difference between two arrangements. By using this capability of our ebodiment-based method, the agent might discriminate a spatial structure from another in the same

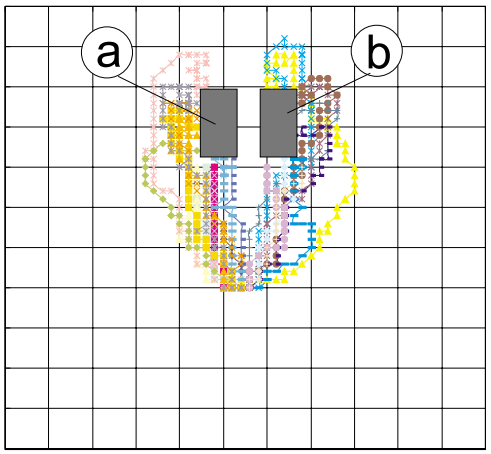


Figure 10: Two objects (narrow case).

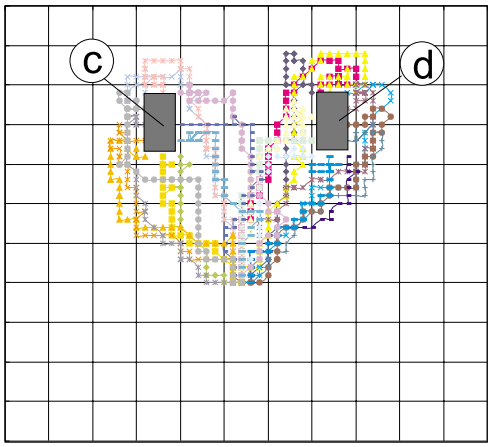


Figure 11: Two objects (wide case).

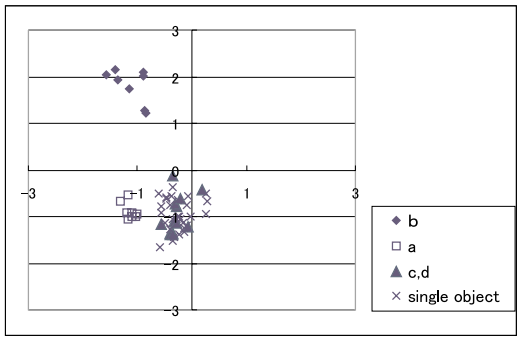


Figure 12: Shape characterizing value of f3-f4 of each object.

way.

7 Discussion and Conclusion

In this paper, we propose a method of embodiment-based visual object recognition. We define the embodiment as agent's own extent of body and locomotive ability. In order to represent an object by embodiment, we employ a reaching action path which represents the relation between surfaces of the agent body and the object. Then, agent can acquire the relation between vision and embodiment through learning of the path with visual input. By means of this method, the agent can recognize objects independently from its position and orientation without prior knowledge. In other words, acquired representation implies invariant advocated by Gibson [3].

In the future work, we will try to develop methods of situation recognition and behavior generation based on the same architecture.

References

[1] Dana H. Ballard, Mary M. Hayhoe, Polly K. Pook, and Rajesh P. N. Rao. Deictic codes for the embodiment of cognition. *Behavioral and Brain Sciences*, 20(4), 1997.

[2] George Berkeley. *Essay Towards a New Theory of Vision*. Everyman's Library, No. 483. Dent., 1709.

[3] James J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin Company, 1979.

[4] U. Nehmzow and T. Smithers. Using motor actions for location recognition. In *Proc. of the First European Conference on Artificial Life*, pages 96–104, 1991.

[5] Erich Prem. The implications of embodiment for cognitive theories. Technical report, Oesterreichisches Forschungsinstitut fuer Artificial Intelligence, Wien, Austria, 1997. TR-97-11.

[6] W. D. Smart and Jhon Hallam. Location recognition in rats and robots. In *Proc. of the Third Int. Conf. on Simulation of Adaptive Behavior*, pages 174–178, 1994.

[7] P. Zingaretti, M. Gasparroni, and L. Vecci. Fast chain coding of region boundaries. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 20(4):407–415, 1998.