Hierarchical Estimation of Multiple Objects from Proximity Relationships Arising from Tool Manipulation

Kotaro Nagahama∗, Kimitoshi Yamazaki†, Kei Okada∗, and Masayuki Inaba∗
∗The University of Tokyo, Tokyo, Japan, †Shinshu University, Nagano, Japan
Email: ∗{nagahama, k-okada, inaba}@jsk.t.u-tokyo.ac.jp, †kyamazaki@shinshu-u.ac.jp

***Abstract*—In this paper, we propose a novel method to estimate a tool's function for a humanoid robot observing a person using a tool. In this method, two types of evaluations are integrated: the relational hierarchy between a tool and objects, and their accompanying movements. This enables to estimate not only moving or conveying function but also cutting or stinging function. To estimate the relational hierarchy, overlapping regions of multiple objects are explicitly tracked based on their viewable regions. We tested this system by basic experiments in which a robot tracked a tool and an object, and estimated functions of the tool in a kitchen environment.**

I. Introduction

Daily assistive humanoid robots are expected to do various domestic work with tools for humans. In recent years, various recognition and motion generation system for these daily operations have been presented [1][2][3][4]. It is necessary for daily assistive robots not only to do fixed work but also to learn new operations that suit the characteristics of each family. For instance, places to be cleaned and tools to be used are different from family to family. However it is hard for many people to program robots for these changes. Therefore the ability to acquire new task knowledge by watching a task of end-users' [6][7] is helpful for these robots.

The purpose of this research is to develop a method for robots to acquire knowledge of everyday tasks based on the observation of a human who handles objects with tools. To pick and to move an object, robots need to know the position on the object and the way to grasp, and the target coordinates to where they have to move it. These pieces of knowledge enabled daily assistive humanoids to pour tea, to wash dishes, and to brush the floor with a broom [5]. Some methods to acquire them by watching human tasks have been developed [7][8].

However, in order to manipulate objects with a tool, more pieces of knowledge are needed:

"Function"
: The way of changing the state of an object.

"Point of action"
: The position or the region on an object and that on a tool where the function occurs.

Fig.1 shows examples of these two pieces of knowledge. Information of function gives a goal state for a robot which is manipulating a tool. It also enables the robot to judge whether it succeeds in a task or not. Information of "point of action" gives constraints for movements of the tool. However,

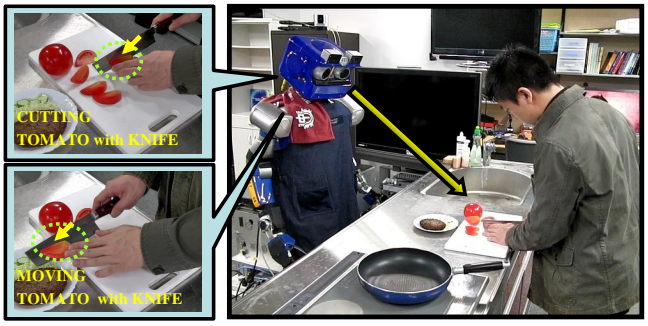


Fig. 1. A humanoid robot observing a person. Not only the orbits of tools and objects but also tools' function and their points of action are needed.

most of the researches on recognizing tasks have dealt with tasks in which a person or a robot moves an object directly with his or its hands. There has been little research with the functions and the points of action. This is because a method to estimate various functions other than "moving" function remains to be developed. In addition, observation of the point of action is difficult because of occlusion when a tool is close to objects.

We present a novel method to estimate functions and a point of action by evaluating I) the accompanying movements and II) the hierarchical relationship between a tool and objects. The accompanying movements means a similar movement of two objects. It is observed when a tool is moving an object. The hierarchical relationship is the superposition between the surface of an object and that of another object. The state of superposition changes when a tool is loading or cutting an object, for example. To achieve the evaluation I) and II), we propose a method to track objects under occlusion by estimating the invisible region of a tool and an object.

This paper is organized as follows. Section II describes related work and our approach. In Section III, the state of superposition is defined. Section IV provides our method of tracking planate objects. In Section V, our integrated system to estimate a state of superposition, to evaluate accompanying movements, and to estimate functions between two objects are described. We evaluate our system in Section VI, and Section VII concludes this paper.

II. Related Work and Our Approach

The purpose of this research is to develop a method for robots to 1) track both the region of a tool and that of an

object , and to 2) estimate the point of action and functions, when the robots are observing human tasks. There are two problems to be solved to achieve this purpose as follows:

- A method to estimate complex functions other than "moving something" while the function is being executed.
- A method to track regions of targets which are overlapping one another.

In order to learn tasks by watching human manipulation, some methods to recognize changes of hand motions and contact relations have been presented. They achieved to recognize moving function and to make robots execute learned tasks such as moving objects, pick-and-place, and assembling [6][7][8]. Griffith et al. [9] used co-movements to check movements dependencies, and classified "containers" and "non-containers." However in daily tasks, functions to change the shape of an object, such as cutting, also often appear. Although methods to recognize changes of targets' shape have also been developed [10], they have to compare the appearance of targets before and after the function is executed. Therefore it was impossible to recognize a function while it is being executed. They can not be used to change templates of tracked targets when their appearance are changing. Duric et al. [11] used tools' motions to classify their functions. Askoy et al. [12] tracked objects' regions and made semantic scene graphs for action recognition and for object categorization. They succeeded in categorizing "Moving Object," "Opening Book," "Making Sandwitch" and "Filling Liquid." However, for daily humanoid robots to learn the motion of a tool, the orbits in the three-dimensional space, including the rotations, of tools and objects are important.

On the other hand, methods to track objects under occlusion have been developed. Ito et al. presented a tracking method which uses divided templates to calculate correlation [13]. Grabner et al. achieved tracking under complete occlusion by tracking "supporters" points around a target [14]. A target with sufficient texture can be detected by feature matching methods [15] if some part of it is visible. However, many daily tools such as knifes and dishes do not have sufficient texture. Ross et al. achieved to track multiple regions with low texture even if they are overlapping one another partially [16]. They used the position of edges and inner color to calculate likelihood in particle filtering. Although our tracking method is similar to theirs, we take into account states of superposition of multiple objects so that they can be tracked when large parts of them are overlapping one another.

In order to solve above issues, we take the following approaches:

- A method of estimating functions of a tool using the state of superposition and the accompanying movements between a tool and an object is presented.
- A method to track a region of a planate target, in which the template of the target is changed using the state of superposition of the target, is implemented.

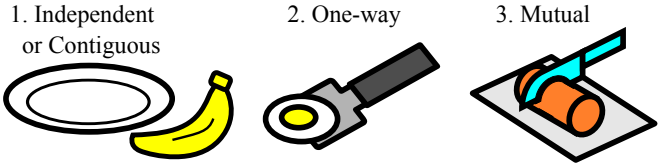


Fig. 2. Species of the state of superposition

III. States of Superposition

A. Classification of States of Superposition

A state of superposition represents how an object's surface is overlapping with another object's surface on the picture plane when they are observed by a certain point. Occlusion occurs when the surface of an object is on the far side of the surface of another object. Therefore occlusion happens 1) when an object exists on the far side of another object, or 2) when an object is running into another object.

We classify the states of superposition into the following three types by how two objects' regions are overlapping one another:

1. Independent or Contiguous
   Two objects' regions are not overlapping.
2. One-way
   Part of an object's region occults part of another object's region.
3. Mutual
   Part of an object's region occults part of another object's region, and another part of the latter also occults another part of the former.

Fig.2 shows examples of the three types of superposition state. At Fig.2,3, some part of a knife region overlaps some part of a vegetable region, and some part of the vegetable's overlaps some part of the knife's at the same time. Therefore this shows the state of "Mutual" superposition.

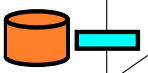
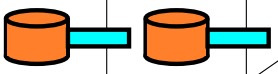
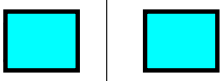
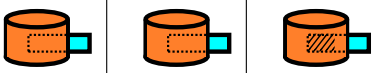
Tab.I shows states of superposition which can be observed when a planate tool and a convex object are located at various positions in the three-dimensional space. The observing point is out of the tool plane. "Independent," "Contiguous," "Partially inserted," and "Fully inserted" are three-dimensional relationships between a tool and an object. "Independent" means that a tool and an object are distant in the three-dimensional space. "Contiguous" means they are in touch. "Partially inserted" means a tool is partially inserted to an object. "Fully inserted" means a tool is fully inserted to an object. Note that the "Mutual" superposition state is observed only when a object is "Partially inserted" by a tool. The "Partially inserted" state happens when an object is cut by a knife, bored by a drill, et al. Therefore recognition of the "Mutual" state of superposition is useful to estimate the function of changing the shape of objects.

B. Function Estimation Using Accompanying Movements and States of Superposition between Two Objects

Functions of a tool is estimated using the state of superposition and accompanying movements as is shown in Tab.II.

TABLE I

SPECIES OF SUPERPOSITION WHICH CAN BE OBSERVED WHEN A TOOL IS IN VARIABLE RELATIONSHIPS WITH AN OPERATIONAL OBJECT

State of Superposition	Tool	Operational Object	Independent	Contiguous	Partially inserted	Fully inserted
Independent	Fully visible	Fully visible				
Independent (Contiguous)	Fully visible	Fully visible				
One-way	Fully invisible	Fully visible				
	Fully visible	Fully invisible				
	Fully visible	Partially visible				
	Partially visible	Fully visible				
Mutual	Partially visible	Partially visible				

Each function in Tab.II represents:

Cut Function of changing the shape of an object. Inserting, cutting, et al.

Move Function of moving an object. Moving, mixing, et al.

Cut & Move
Function of changing the shape of an object and moving it. Sticking and moving an object with a fork et al.

Unknown
Nothing happens or the function can not be estimated.

TABLE II

FUNCTION ESTIMATION USING ACCOMPANYING MOVEMENTS AND STATES OF SUPERPOSITION OF MULTIPLE OBJECTS' REGIONS

		Superposition		
		Independent	Covered	Mutual
Accompanying Movements	Unclear	Unknown	Unknown	Cut
	No	Unknown	Unknown	Cut
	Yes	Move, if contiguous	Move	Cut&Move

IV. A METHOD TO TRACK A PLANATE OBJECT REGION UNDER OCCLUSION

This section explains a method of tracking a planate object with low texture under occlusion. The initial region of the target $R(0) \subset \mathbb{N}^2$, and the coordinate of it ${}^C\boldsymbol{H}_0$ at the initial time $t = 0$ [frame] are given.

A. Planate Object Tracking Using Particle Filtering

A planate object is tracked using particle filtering [17]. $\boldsymbol{x}_t$, the state variable of the particle filter has six degrees of freedom. It describes translation and rotation of the object in the camera coordinate system as follows,

$$\boldsymbol{x}_t = [x_t \quad y_t \quad z_t \quad \alpha_t \quad \beta_t \quad \gamma_t]^T. \tag{1}$$

After the initial coordinate of the target is set at first, initial particles $s_0^{(k)}$ are made with equal weight w_0 as follows,

$$s_0^{(k)} = \{\boldsymbol{x}_0, w_0\}. \tag{2}$$

Let k be the index of a particle. Our motion model $p(X_t|X_{t-1})$ is as follows,

$$\boldsymbol{x}_{t|t-1}^{(k)} = \boldsymbol{x}_{t-1}^{(k)}. \tag{3}$$

After the particles are evaluated by the likelihood function, the weights of particles are normalized and updated.

B. Likelihood Function

The likelihood is calculated as the degree of similarity between the input image $I(t)$ and the image composed only of the target $I_A(t)^{(k)} \subset \mathbb{N}^2 \times [0, 255]^3$. $I_A(t)^{(k)}$ is calculated from the initial coordinate of the target ${}^C\boldsymbol{H}_0$, the coordinate of $\boldsymbol{x}_t^{(k)}$, and the initial image of the target's region $I_A(0) \subset \mathbb{N}^2 \times [0, 255]^3$.

The similarity is calculated by integrating two types of evaluation methods. One is the evaluation of the target's edge information, and the second is the evaluation of the target's inner color information.

1) Evaluation Based on the Boundary Information: The evaluation score based on the target's boundary information is calculated using chamfer distances [18]. To calculate a chamfer distance, edges are abstracted from the input image at t [frame], $I(t)$, at first. Next, a distance map in which each pixel value represents the distance between the pixel and the nearest edge point. A chamfer distance $D_{ch}(P, I)$ is calculated by the distance map and the edges of the target's image $I_A(t)^{(k)}$ as follows,

$$D_{ch}(P, I) = \frac{1}{|P|} \sum_{p \in P} d_{I(t)}(p). \quad (4)$$

Let $P \subset \mathbb{N}^2$ and $|P|$ be the set of boundary pixels of the target image $I_A(t)^{(k)}$ and the number of P, respectively. $d_{I(t)}(p)$ represents the value of the distance map at position p. Therefore the $D_{ch}(P, I)$ becomes lower if the shape of the boundary of the target image $I_A(t)^{(k)}$ is more similar to that of the input image $I(t)$.

The likelihood is calculated by Eq.5.

$$E_e(\boldsymbol{x}_t^{(k)}) = \frac{1}{1 + \exp\{-G_e \cdot (-D_{ch}(P(t)^{(k)}, I(t)) + C_e)\}}, \quad (5)$$

where the G_e and the C_e are constant numbers.

2) Evaluation Based on the Inner Color Information: The evaluation based on the inner color information is useful in case there are some edges of another target near the correct target's boundary. Eq.6 is the evaluation function. E_c is calculated using the number of pixels of the different set of $R_c(t)$ and $R_e(t)^{(k)}$. $R_c(t) \subset \mathbb{N}^2$ is the set of pixels of the target's color which is near the boundary of the target region $R(t)^{(k)}$. $R_e(t)^{(k)} \subset \mathbb{N}^2$ is the set of pixels which are in $R(t)^{(k)}$ and near the boundary of the target region $R(t)^{(k)}$.

$$E_c(\boldsymbol{x}_t^{(k)}) = 1 - \frac{|(R_c(t) - R_e(t)^{(k)}) \cup (R_e(t)^{(k)} - R_c(t))|}{|R_e(t)^{(k)}|}, \quad (6)$$

where "$-$" and "$||$" calculate difference set and the number of items, respectively. $E_c(\boldsymbol{x}_t^{(k)})$ runs from 0 to 1, and it becomes larger if the difference of the abstracted color regions becomes smaller.

3) Integration of Two Evaluations: Two types of evaluations are integrated by Eq.7.

$$E(\boldsymbol{x}_t^{(k)}) = E_c(\boldsymbol{x}_t^{(k)})^{C_c} \cdot E_e(\boldsymbol{x}_t^{(k)}), \quad (7)$$

where the C_c is a positive constant number. $E(\boldsymbol{x}_t^{(k)})$ runs from 0 to 1, and it comes close to 1 when the likelihood is high.

C. Tracking with Non Shielded Region

In order to track a target under occlusion, trackers calculates the likelihood at $T+1$ [frame] using only the target's image of viewable, non shielded region at T [frame] (see right of Fig.4). Therefore the shielded region of the target need to be calculated. This calculation is described in section V.

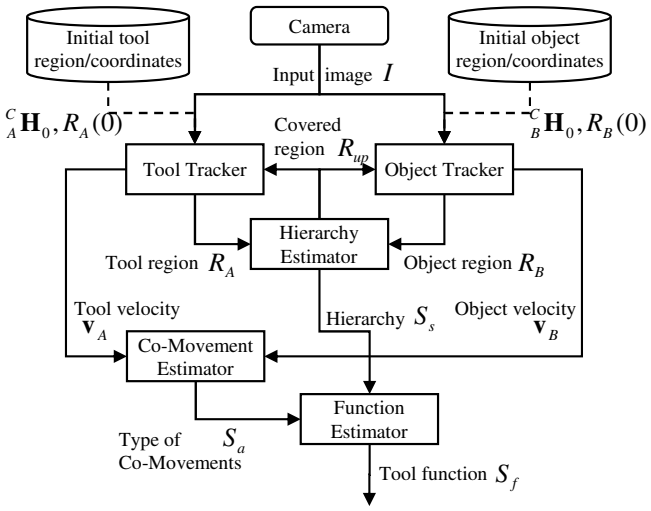


Fig. 3. Flowchart of tracking multiple regions and estimating function

V. Function Estimation Based on State of Superposition and Co-Movements

A. Procedure of Function Estimation

Fig.3 shows the procedure of function estimation based on the state of superposition and the accompanying movements between a tool and an object. At first the coordinates and the regions of a tool and an objects are set. Then both targets are tracked in parallel using single camera images. At each frame, the tool image $I_A(t)$ and the object image $I_B(t)$ is reported to the module which evaluates the state of superposition. At the same time, the velocity of the tool $\boldsymbol{v}_A(t)$ and that of the object $\boldsymbol{v}_B(t)$ is reported to the module which evaluates accompanying movements.

The hierarchy estimator calculates the type of superposition $S_s(t) \in \{$"Independent","One-way","Mutual"$\}$ from the image of the tool $I_A(t)$ and that of the object $I_B(t)$. It also estimates the overlapping region of two targets. The overlapping region is reported to the trackers and it is used to fix the template of the tracker, as described in section IV. The module for accompaniment movements calculates the type of co-movements $S_a(t) \in \{$"Unclear","No","Yes"$\}$ from the velocity of the tool $\boldsymbol{v}_A(t)$ and that of the object $\boldsymbol{v}_B(t)$. The state of superposition and the co-movements become the inputs of the function estimator.

The function estimator classify the function based on Tab.II. Final output of this system is the estimated function, $S_f(t) \in \{$"Cut", "Move", "Cut & Move", "Unknown"$\}$.

B. Calculation of the Type of Superposition

First, the hierarchy estimator in Fig.3 uses the targets' region image $I_A(t), I_B(t) \subset \mathbb{N}^2 \times [0, 255]^3$ at t [frame] to calculate the overlapping region $R_{A,B}(t) = R_A(t) \cap R_B(t) \subset \mathbb{N}^2$ (the region enclosed by dotted line at Fig.4) of two object regions $R_A(t), R_B(t) \subset \mathbb{N}^2$. Second, labeling of the overlapping region is done: the estimator calculates which

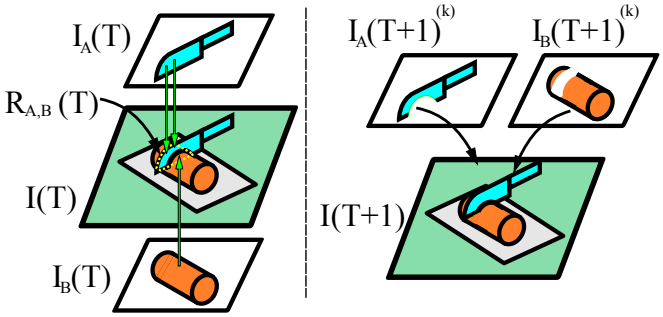


Fig. 4. A method to estimate objects' hierarchy (left) and to track them (right)

of $I_A(t)$ and $I_B(t)$ is closer to the color of the input image $I(t)$ at each pixel in $R_{A,B}$ (the arrows in the left image of Fig.4).

The state of superposition are estimated as follows,

$$S_s(t) = \begin{cases} \text{"Independent", if } R_{A,B} = \emptyset. \\ \text{"One-way", if} \\ (\forall p \in R_{A,B}; s(p,A,B)) \vee (\forall p \in R_{A,B}; s(p,B,A)). \\ \text{"Mutual", if} \\ (\exists p \in R_{A,B}; s(p,A,B)) \wedge (\exists p \in R_{A,B}; s(p,B,A)). \end{cases} \quad (8)$$

$$s(p,X,Y) \triangleq ||\boldsymbol{I}(p) - \boldsymbol{I}_X(p)|| \geq ||\boldsymbol{I}(p) - \boldsymbol{I}_Y(p)||$$

Let $\boldsymbol{I}(p) \in [0,255]^3$ be the intensities of three colors (R, G, B) in the input image I. The sign $||\boldsymbol{x} - \boldsymbol{y}||$ expresses the distance between two colors $\boldsymbol{x}, \boldsymbol{y} \in [0,255]^3$. We use Manhattan distance as this calculation.

As previously explained, object region trackers calculates the likelihood at $T+1$ [frame] using only the target's image of non shielded region at T [frame] (see right of Fig.4). Therefore the result of labeling is transferred not only to the function estimator but to target trackers.

C. Evaluation of the Accompanying Movements

Methods to evaluate co-movements have been developed for robots to abstract the region of handling objects. For instance, canonical correlation between the 2-D movements of an object and joint angles of a robot is used to estimate an object in its hand while the robot is wielding it [19]. In our approach, both the velocity of a tool and that of an object are calculated in the same three-dimensional space. Therefore the difference between two velocities can be calculated and used to evaluate the degree of accompaniment.

Let $\boldsymbol{v}_A(t)$ and $\boldsymbol{v}_B(t)$ be the velocity of a tool, and that of an object at t [frame], respectively. The evaluation of accompanying movements at T [frame] is calculated as the average difference of their velocities from $T-N$ [frame] to T [frame] as follows.

$$E_{acm}(T) = \exp\left\{\frac{-1}{C_{acm}} \sum_{t=T-N+1}^{T} ||\boldsymbol{v}_A(t) - \boldsymbol{v}_B(t)||\right\}, \quad (9)$$

where C_{acm} is a constant positive number. $E_{acm}(t)$ runs from 0 to 1, and it becomes larger if noticeable co-movements are observed.

However, the co-movements are not correctly evaluated with the equation above when both objects are at rest. Therefore the average speed of each object is used to decide whether the evaluation is possible or not. The average speed from $T-N$ [frame] to T [frame] is calculated as follows,

$$\overline{||\boldsymbol{v}_X(T)||} = \frac{1}{N} \sum_{t=T-N+1}^{T} ||\boldsymbol{v}_X(t)||. \quad (10)$$

Finally, the type of accompanying movements of the tool and the object at t [frame], $S_a(t) \in$ {"Unclear", "No", "Yes"}, is calculated by Eq.11.

$$S_a(t) = \begin{cases} \text{"Yes", if } h(A,t) \wedge h(B,t) \wedge a(t). \\ \text{"No", if } (h(A,t) \wedge h(B,t) \wedge \neg a(t)) \\ \vee (h(A,t) \wedge \neg h(B,t)) \\ \vee (\neg h(A,t) \wedge h(B,t)). \\ \text{"Unclear", if } (\neg h(A,t) \wedge \neg h(B,t)). \end{cases} \quad (11)$$

$$h(X,t) \triangleq \overline{||\boldsymbol{v}_X(t)||} \geq v_{thr}$$

$$a(t) \triangleq E_{acm}(t) \geq E_{acm\_thr},$$

where v_{thr} and E_{acm_thr} are constant numbers.

If both of the tool and the object are moving and there is a correlation between them, the type is calculated as "Yes". If both objects are at rest, the type is "Unclear". In other cases the type of the co-movement is calculated as "No".

VI. Experiments

We tested the tracking system and the method to estimate functions by three basic experiments with simulated images and real images. At the first frame of each experiment, the regions of the targets and the coordinates of them are set by a human interactively. To set the initial regions of the targets, we used Graph-cut method [20].

A. Experiment 1

We tested the tracking method described in Section IV. In this experiment, the targets to be tracked were two planate objects which moved and overlapped one another. We compared two methods: A) our tracking method which changes the templates of the targets, B) a method which uses fixed templates and our likelihood functions. Fig.5 illustrates the result of the experiment. The tracking succeeded completely with the method A). On the other hand, the tracking failed when the red object went below the blue one with the method B).

B. Experiment 2

We tested the method to estimate functions, described in Section V. The inputs were simulated images of a tool which was A) cutting an object and B) moving an object. Fig.6 shows the result of the experiment. Although the system failed to estimate the type of co-movement at some frames in B), it mostly succeeded in estimating functions. The failure of the calculation of the co-movement was caused by the noisy result of the tracking. The success rate of estimating functions was 95% in two experiments. Here the success rate

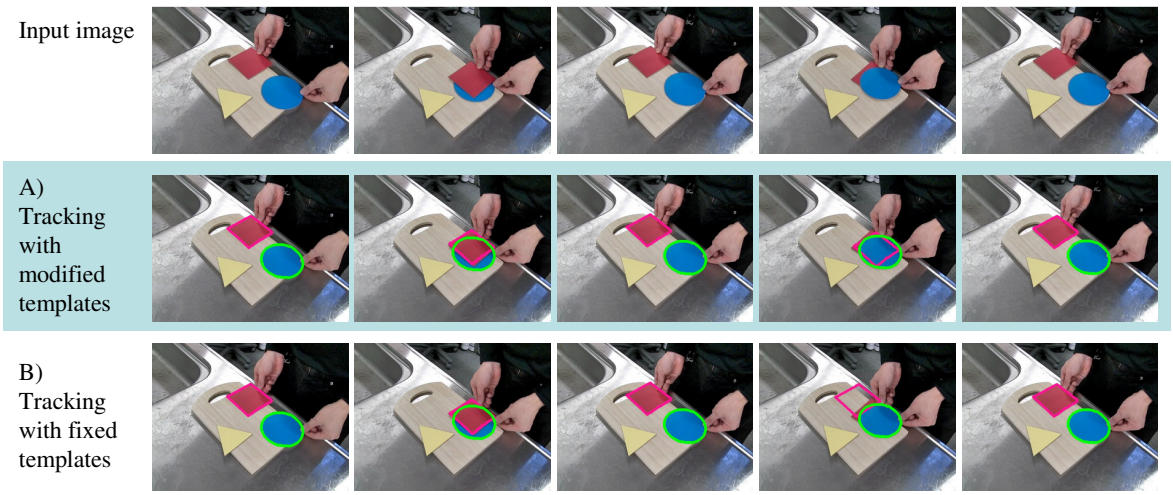


Fig. 5. Result of tracking planate objects' regions which are moving and overlapping one another

was calculated by dividing the number of the frames in which the estimated function was same to the function decided by a human, by the number of all frames.

C. Experiment 3

Finally, we evaluated our system by tracking a tool and estimating functions of it in a kitchen environment.

1) Experimental: For this experiment, the camera captured a knife which was being moved by a person and working on a fruit. The operation was cutting in Experiment 3-1, picking up and moving in Experiment 3-2. In both experiments, regions of the knife and the fruit were tracked, and functions of the knife were estimated.

2) Results: Fig.7 shows the estimated type of co-movement, that of superposition, and function. The dotted lines are correct answers drawn by a human. The state of co-movement "Yes" was not observed in Experiment 3-1. On the other hand in Experiment 3-2, it was observed after 80 frame. The tool and the object were independent at first. In Experiment 3-1, the state of superposition changed from "Independent" to "One-way" and "Mutual," and returned to "Independent." This is a correct transition which is observed when a tool is cutting an object. In Experiment 3-2, the state of superposition changed from "Independent" to "One-way". This is also a correct transition observed when a tool is moving an object. At some frames in the experiment, the state of superposition was estimated as "Mutual". This is because two targets' parts of same color caused an error in estimating the upper and lower object. The success rate of estimating functions was 91 % in Experiment 3-1, and 65 % in Experiment 3-2.

VII. Conclusion

In this paper, we presented a novel method to estimate a function of a tool for a robot observing a person. In this method, a tool and an object are tracked in parallel, and the accompanying movements and the state of superposition of them are evaluated. Finally, two species of information are integrated to estimate a function. This method enables to estimate not only moving function but also cutting function while the function is being executed. This method have restrictions on the observing point because the camera should capture Mutual superposition relationship for cutting function, and One-way superposition relationship for moving function. However, if the robot is life-sized and its camera is mounted on its head, it could capture the relationships in many cases, like humans. In addition, some kinds of view point planning method could be useful to change the observing point of the robot.

For future work, we will try to use this method for robots manipulating tools to recognize whether the task succeeds or not. Moreover, online recognition of functions enables a tracking system to change targets to track after a function occurs. This enables for robots to observe objects continuously when they are divided and their shapes are changed, for instance.

References

[1] T. Asfour, K. Regenstein, P. Azad, J. Schröde and R. Dillmann, "ARMAR-III: A HUMANOID PLATFORM FOR PERCEPTION-ACTION INTEGRATION," *2nd International Workshop on Human-Centered Robotic Systems*, (2006).

[2] M. Beetz, U. Klank, I. Kresse, A. Maldonado, L. Mösenlechner, D. Pangercic, T. Rühr and M. Tenorth, "Robotic Roommates Making Pancakes," *Proceedings of the 11th IEEE-RAS International Conference on Humanoid Robots*, (2011), pp. 529–536.

[3] K. Yamazaki, Y. Watanabe, K. Nagahama, K. Okada and M. Inaba, "Recognition and Manipulation Integration of a Daily Assistive Robot Working on Kitchen Environment," *Proceedings of the 2010 IEEE International Conference on Robotics and Biomimetics*, (2010), pp. 196–201.

[4] K. Okada, M. Kojima, S. Tokutsu, Y. Mori, T. Maki and M. Inaba, "Task Guided Attention Control and Visual Verification in Tea Serving by the Daily Assistive Humanoid HRP2JSK," *Proceedings of the 2008 IEEE International Conference on Intelligent Robots and Systems*, (2008), pp. 1551–1557.

[5] K. Okada, M. Kojima, Y. Sagawa, T. Ichino, K. Sato and M. Inaba, "Vision based behavior verification system of humanoid robot for daily environment tasks," *Proceedings of the 6th IEEE-RAS International Conference on Humanoid Robots*, (2006), pp. 7–12.

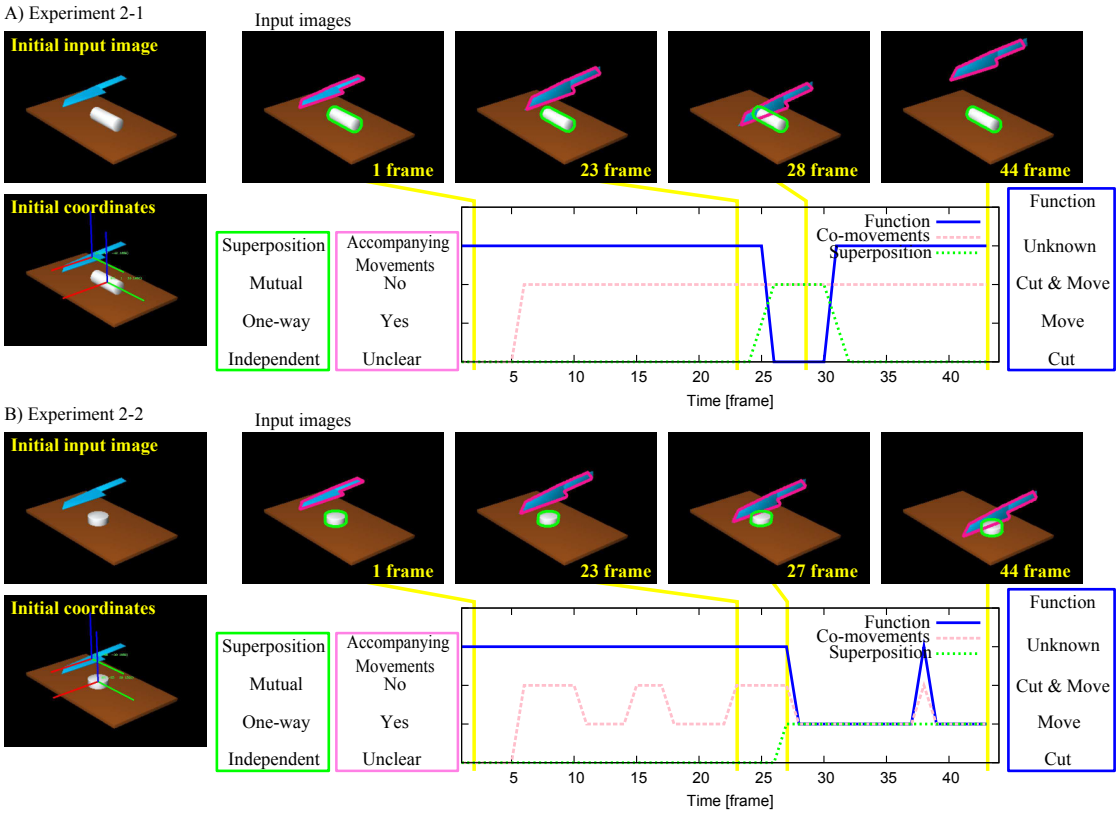


Fig. 6. Result of tracking objects' regions and estimating functions on simulated images

[6] Y. Kuniyoshi, M. Inaba and H. Inoue, "Learning by Watching: Extracting Reusable Task Knowledge from Visual Observation of Human Performance," *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 6, (1994), pp. 799–822.

[7] K. Ogawara, J. Takamatsu, H. Kimura and K. Ikeuchi, "Generation of a task model by integrating multiple observations of human demonstrations," *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, (2002), pp. 1545–1550.

[8] S. Calinon, F. Guenter and A. Billard, "On Learning, Representing, and Generalizing a Task in a Humanoid Robot," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, (2007), No.2, vol.37, pp.286–298.

[9] S. Griffith, V. Sukhoy and A. Stoytchev, "Using sequences of movement dependency graphs to form object categories," *Proceedings of the 11th IEEE-RAS International Conference on Humanoid Robots*, (2011), pp. 715–720.

[10] Y. Shinchi, Y. Sato and T. Nagai. "BAYESIAN NETWORK MODEL FOR OBJECT CONCEPT," *Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing*, (2007), pp. II-473–II-476.

[11] Z. Duric, J. A. Fayman and E. Rivlin, "Function from motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (1996), Vol. 18, Issue 6, pp. 579–591.

[12] E. E. Askoy, A. Abramov, F. Wörgötter and B. Dellen, "Categorizing object-action relations from semantic scene graphs," *Proceedings of the IEEE International Conference on Robotics and Automation*, (2010), pp. 398–405.

[13] K. Ito and S. Sakane, "Robust View-based Visual Tracking with Detection of Occlusions," *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*, (2001), pp. 1207–1213.

[14] H. Grabner, J. Matas, L. Van Gool and P. Cattin, "Tracking the Invisible: Learning Where the Object Might be," *Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition*, (2010), pp. 1285–1292.

[15] D. G. Lowe, "Object Recognition from Local Scale-Invariant Features," *Proceedings of the 1999 IEEE International Conference on Computer Vision*, (1999), pp. 1150–1157.

[16] M. Ross, "MODEL-FREE, STATISTICAL DETECTION AND TRACKING OF MOVING OBJECTS," *Proceedings of the 2006 IEEE International Conference on Image Processing*, (2006), pp. 557–560.

[17] M. Isard and A. Blake, "CONDENSATION – conditional density propagation for visual tracking," *Int. Journal on Computer Vision*, Vol. 28, No. 1, (1998), pp. 5–28.

[18] D M. Gavrila, "Multi-feature Hierarchical Template matching Using Distance Transforms," *Proceedings of the 14th International Conference on Pattern Recognition*, (1998), pp. 439–444.

[19] C. Nabeshima, Y. Kuniyoshi and M. Lungarella, "Towards a Model for Tool-Body Assimilation and Adaptive Tool-Use," *Proceedings of The 6th IEEE International Conference on Development and Learning*, (2007), pp.288–293.

[20] Y. Boykov and V. Kolmogorov, "An Experimental Comparison of Min-Cut/Max-flow Algorithms for Energy Minimization in Vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 9, (2004), pp. 1124–1137.

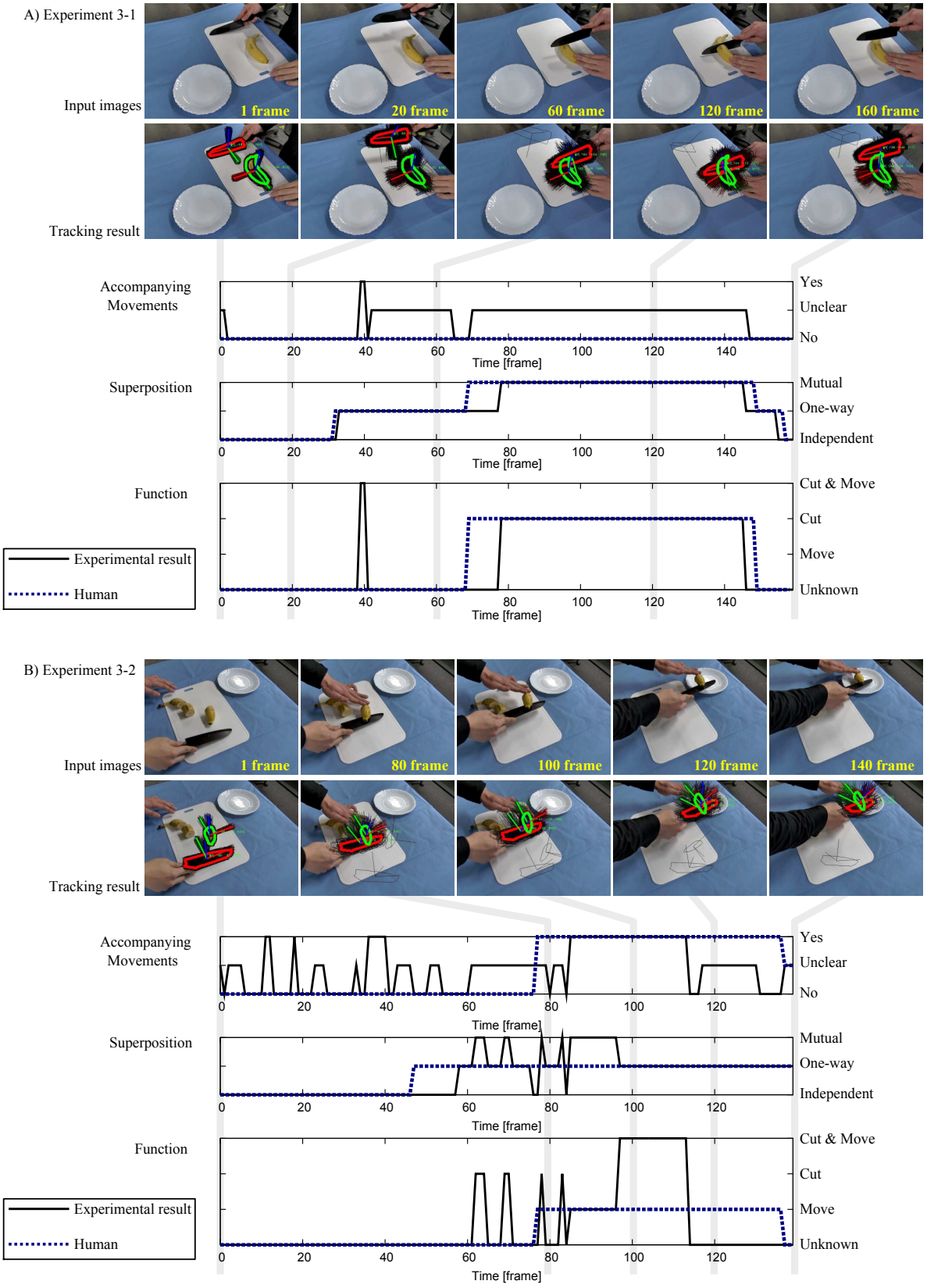


Fig. 7. Result of tracking and estimation of accompaniment, superposition, and function in A) Experiment 3-1 and B) Experiment 3-2