Bulletin of the JSME Vol.14, No.4, 2020
Journal of Advanced Mechanical Design, Systems, and Manufacturing

Real-time pose tracking of 3D targets by photo-model-based stereo-vision

Hongzhi TIAN*, Yejun KOU*, Xiang LI** and Mamoru MINAMI*
* Division of Industrial Innovation Sciences, Graduate School of Natural Science and Technology, Okayama University
3-1-1 Tsushimanaka, Kita-ku, Okayama, 700-8530, Japan
E-mail: tianhongzhi9@163.com
** Zhuhai College of Jilin University
Caotang Bay, Jinwan District, Zhuhai, Guangdong, 519041, China

Received: 3 June 2019; Revised: 11 March 2020; Accepted: 22 April 2020

Abstract
Nowadays, robots have been utilized to perform a wide variety of tasks instead of humans, in which those equipped with hand-eye cameras have been proved to be useful in factories for handling metal parts. However, there have been difficulties for robots with vision sensors like cameras to detect a target objects' 3D pose accurately, especially in the case that the target object is unique and the shape is arbitrary, and further, the object is moving. Visual servoing to moving target may enable the robot to pursue an animal to catch it. Aiming at achieving identification ability of a unique object, this paper utilized our previous study concerning photo-model-based object recognition. Based on the study, this paper applies the photo-model-based pose estimation method to video-rate stereo-vision position and orientation (pose) tracking, and it has been applied to an arbitrary-shaped moving target. To confirm the abilities of the proposed stereo-vision-based method to pursue an arbitrary target, real-time 3D pose estimation experiment to track the relative pose between a moving target and hand-eye robot have been conducted. The real-time experimental results show the proposed system can recognize the designated arbitrary-shaped target and track its pose in real-time.

Keywords : Photo-model-based recognition method, Pose estimation, Moving arbitrary-shaped target, Real-time Multi-step Genetic Algorithm (RM-GA), Stereo-vison, Dual-eye cameras

1. Introduction

Humans have mobilized robots to perform repetitive and dangerous tasks that are required to be conducted in exceptional environments such as outer space (the universe beyond the earth's atmosphere) or the bottom of the sea. Since robots have higher reliability and accuracy than humans, they have been used extensively in production factories to perform a wide variety of tasks instead of human workers. However, until now, robots cannot entirely replace humans. While human beings can conduct intended tasks in situated circumstances, an automated robot is not adept enough as adaptabilities of humans. Therefore, the researchers have been trying to improve the adaptive abilities of automated robots.

Concerning automated robots, a robot control technology using visual information obtained from cameras in the feedback loop named as visual servoing, is expected to be able to allow the robot to adapt to changing or unknown environments (Agin et al., 1979; Allen et al., 1993; Hutchinson et al., 1996; Malis et al., 1999 and Oh and Allen, 2001). From the view point of how the cameras are used, there are two main configurations. The first is an eye-in-hand configuration (Song et al., 2007), that cameras are mounted on the robot's end-effector. The second one is the eye-to-hand configuration. It has the camera(s) fixed in the workspace to see the robot's hand (Dune et al., 2007). These methods can obtain multiple different views to observe an object by increasing the number of cameras. The eye-in-hand has a partial but precise sight of the scene since the camera can be placed near targets by a robot hand, whereas the eye-to-hand camera has less accurate but global view of the robot and the targets. However, in eye-to-hand configuration, fixed camera position in the workspace reduces the adaptability of a system for a changing environment since it is fixed. After considering those factors, an eye-in-hand configuration is adopted in our approach.

Paper No.19-00284
[DOI: 10.1299/jamdsm.2020jamdsm0057]

Among methodologies of visual servoing, feature-based and model-based methods have been mainly researched, and their performances have been discussed (Marchand and Chaumette, 2005). The feature-based approach focuses on tracking 2D features such as geometrical primitives (points, segments, circles) or an object's contours, regions of interest. The model-based approach explicitly uses a model of the target objects, which helps the robot estimate target's pose precisely. The pose includes position and orientation. About feature-based methods, there are some studies concerning picking operation based on visual recognition. That is, after 2D object segmentation and classification with a deep learning technology are conducted, then picking or grasping with point cloud gotten from an RGB-D (Red, Green, Blue, and Distance) camera are demonstrated (Morrison et al., 2017; Zeng et al., 2017; Schwarz et al., 2018). However, it needs to wait some time before completing the estimation of the target's pose. Compared with feature-based methods, model-based methods have more information about the target object and usually provides a robust solution. For example, it can cope with partial occlusion of the objects. And all points of a solid 3D model as a group are projected onto 2D camera image planes without the Corresponding Points Identification Problem (Matsuyama et al., 1998) that has been pointed out as the difficulty existing 3D image reconstruction from 2D images input from stereo-vision (2D-3D method).

In contrast, the authors have employed a "Forward Projection," i.e., a 3D model has been projected into stereo-vision image planes, and the projected 3D models are compared with the actual target that is also projected naturally onto the stereo-vision image planes (3D-2D method). The merit of this method is that it can avoid the Corresponding Points Identification Problem, since the points on a 2D model projected to left and right camera images from points defined on a 3D model have no irregularities in the correspondence of the points in left and right images. In the past, our researches are based on the fixed 3D model-based recognition method (Tian et al., 2017a), where the process of model generation of the 3D model-based approach is inevitable to be complex (Tamadazte et al., 2010; Tian et al., 2017a). Based on the merits and demerits of 2D-3D and 3D-2D methods, the model-based matching method using 3D-2D projection has been utilized in our past researches on visual servo system. They are online pose tracking of 3D marker (Song and Minami, 2008), and recently a vision-based docking system for Automated Underwater Vehicles (Myint et al., 2017).

As mentioned above, the 3D model generation process is complicated. To simplify the process and establish a more general and practical recognition approach to arbitrary-shaped and -colored object, a photo-model-based recognition method has been devised (Phyu et al., 2016), since the photo-model can be made at once by taking a photo of an arbitrary object. This recognition method was used for arbitrary clothes handling system (pick and place). Since clothes are deformable, the deformation can also be seen as a disturbance hindering pose estimation. By making the detecting and handling system adaptive and tolerable against the deformation and lighting condition varieties, the photo-model-based recognition method has been proved to be practical and credible (Phyu et al., 2017).

We hope that this technology is not limited only for clothes handling but also for a broader range of applications, e.g., pursuing and catching animals that move to escape from the visual-servoing underwater robot. Compared with static clothes, aquatic animals or other creatures can move with their shapes changing for moving. To achieve catching designated animals, the photo-model-based recognition method needs to be improved to be applicable for real-time pose tracking. Therefore, in response to the above requirement, this paper expands the photo-model methodology to enable the stereo-vision robot to conduct visual servoing to moving targets. And this paper introduces how the photo-model is used to detect the arbitrary target and how the on-line pose tracking could be conducted in real-time.

This pose tracking methodology needs to detect 6 DoF pose of the object in real-time and will enable a robot to conduct 3D visual servoing (Tian et al., 2017b). The authors have proved dual-eye stereo-vision cameras can also work well for tasks of 3D visual servoing (Kou et al., 2018), however it needs to predefine the 3D model consisting of checking dots set on the target 3D-marker. With a photo taken at a known distance, the photo-model-based recognition method can start to work immediately, and this method does not need 3D model definition that should be predefined and the defining task is a burden for the researchers. Even though data-driven method with deep learning technique also uses pictures to detect 3D pose of a target object, it requires a large number of pictures and pre-training time (Tsai et al., 2018; Zeng et al., 2017). The training photos should include many varieties of photos with different position and orientation, which should be prepared for detecting the target's pose precisely. Then it is difficult to be applied to real-time pose estimation. There is a research for real-time pose estimation with photo data and deep neural networks (Bateux et al., 2017). However, its target is just a 2D planar object, not a 3D object.

In this paper, 3D sea animal toys are chosen as targets that exemplify arbitrary targets. Frequency response experiments have been conducted to verify the ability of the real-time pose estimation of 3D solid target and the pose tracking of the dual-eye stereo-vision system.

The rest of the present paper is organized as follows: Section 2 is a brief introduction of the development process

[DOI: 10.1299/jamdsm.2020jamdsm0057]

of the photo-model-based recognition method and contribution points of this paper. Section 3 presents the stereo-vision photo-model-based recognition, which converts pose estimation to optimization problem. Section 4 introduces the experimental environment and experimental hardware. The proposed method has been evaluated in section 5. Section 6 describes the real-time 3D pose estimation experimental results, followed by discussion and conclusion in sections 7 and 8.

2. Contributions

In this section, we review works related to the contents in this paper, conductive to 3D-pose visual servoing to a moving target with arbitrary shape. The foundation comes from photo-model-based clothes handling robot.

Since the photo-model can be made at once by taking a photo of the target object, the target can be recognized instantly, and the pose can be estimated immediately. Figure 1 shows the photo-model-based handling robot (pick and place) introduced by our previous researches (Funakubo et al., 2016 and 2017; Phyu et al., 2016 and 2017). The proposed system aims at picking up clothes after a robot recognizes it and classifies the clothes into a collection box.

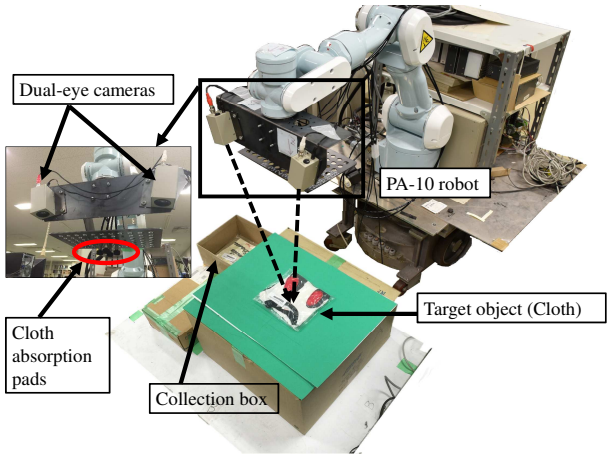


Fig. 1 A photo of the clothes handling robot system with dual-eye cameras: PA-10 robot is equipped with a vacuum unit and two cameras used as stereo-vision, where four pads connected with the air compressor made the robot possible to perform the pick (absorption) and place of the clothes. In the test, the robot picked up 12 kinds of deformable clothes and classified them, and set them into the collection box.

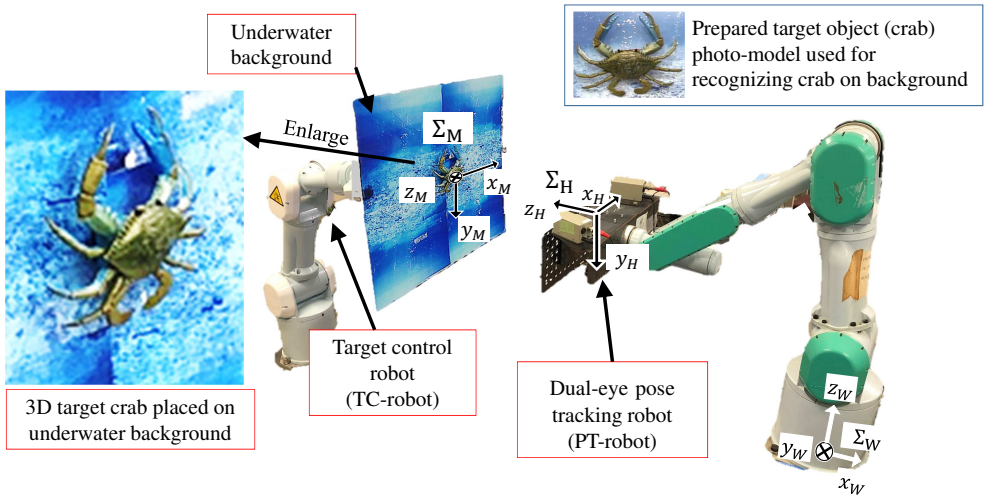


Fig. 2 The motion of the target animal, crab, is given by TC-robot, and the PT-robot moves to keep desired relative pose of the PT-robot against the crab attached on a panel with sea bottom backdrop whose motion is given by TC-robot. World coordinate system Σ_W, hand coordinate system Σ_H, and target coordinate system Σ_M are depicted in the figure.

The robot shown in Fig. 1 has been confirmed to be able to identify 12 different deformable clothes, estimate the pose and handle them under illumination varieties (Phyu et al., 2018a and 2018b). Previous studies (Phyu et al., 2017 and

[DOI: 10.1299/jamdsm.2020jamdsm0057]

2018c) have also verified that the photo-model-based recognition method can detect the partly hidden cloth.

As motioned in the introduction, we hope that this recognition method is not only limited for clothes handling but also applied to a wider range of applications. Therefore, in (Tian et al., 2019), the method is improved and used for the position visual servoing task of a moving target object. However, since orientation tracking is more difficult than position tracking, the object's orientation was fixed at that time, and only the position 3DoF was changed.

Based on these studies, this paper focuses on the 6DoF detection ability of the photo-model-based pose estimation method. Its recognition ability with respect to the deforming objects is not discussed.

In this paper, a photo-model has been used for visual tracking so that the stereo vision can follow a 3D-shaped sea animal model whose motion is given by another robot (TC-robot) as shown in Fig. 2, where the motion is shown to the pose tracking robot (PT-robot) without any communications between the two robots. Regarding visual tracking, this is an important task in the field of computer vision, also known as object tracking (Yilmaz et al., 2006, Li et al., 2013). It can be applied to many domains, such as visual surveillance (Joshi and Thakore, 2012) and reconstruction of the environment (Newcombe et al., 2011). In its simplest form, tracking can be defined as the problem of estimating the trajectory of an object either in the 2D image plane or in the 3D object space as it moves around a scene (Leibe et al., 2008). In the robot control field, visual tracking of an object involves the estimation of the object's position and orientation (pose) (Tamadazte et al., 2011). The pose estimation of the target object is very important for robot movement. Although visual tracking does not involve robot control, it is still a necessary and basic research in visual servoing, autonomous navigation, and other robot functional research (Pan et al., 2015).

The future purpose of the pose tracking and the visual servoing to sea animals with photo-model is to establish control strategy of an underwater robot that can catch fishes or dispose of detrimental sea animals, e.g., crown-of-thorns starfish. Since the model for pose estimation can be given by a photograph, the target animal of visual-tracking could be set arbitrarily and instantaneously.

The RM-GA (Myint et al., 2017) for real-time pose estimation and tracking has been achieved based on a prerequisite of 3D target model being given and predefined precisely to track the target's pose.

However, defining the 3D target model is time-consuming. Therefore, in this paper, we have simplified the 3D target definition to 2D photo-model. Even though the photo-model is 2D image model, the 3D target object moving in 3D space could be tracked by the 2D photo-model since the 3D target object can be represented approximately by the tangential 2D model. This is the reason that the 3D target pose can be estimated by the 2D photo-model.

This paper examines the feasibility about whether 3D pose of target could be estimated through 2D photograph model or not. More precisely, the contributions of this paper are as follows.

- This paper proposes a method to estimate 3D target pose by using stereo-vision and 2D photo-model.
- Based on the above prerequisite, the target recognition and pose estimation problem have been converted to the optimization problem.
- The real-time abilities to track 3D targets have been confirmed through real tracking experiments, which has been enhanced by improving real-time nature of RM-GA.

All above points have helped achieve photo-model-based real-time visual tracking.

3. Photo-model-based recognition

3.1. Kinematics of stereo-vision

The proposed system is an eye-in-hand system with dual-eye stereo-vison cameras. Camera model is pinhole model. Figure 3 shows a perspective projection of the dual-eye vision system. The coordinate systems of dual-eye cameras and the target object consist of world coordinate system Σ_W, j-th model coordinate system Σ_{Mj}, hand portion coordinate system Σ_H, left and right camera coordinate systems Σ_{CL} and Σ_{CR}, and image coordinate systems Σ_{IL} and Σ_{IR}, and they are being shown in Fig. 3. The position vectors of an arbitrary i-th point of the j-th 3D model coordinate Σ_{Mj} based on each coordinate system are as follows:

- ${}^{W}\boldsymbol{r}_i^j$: 3D position of an arbitrary i-th point on j-th 3D model based on Σ_W,
- ${}^{M}\boldsymbol{r}_i^j$: 3D position of an arbitrary i-th point on j-th 3D model in Σ_{Mj},
- ${}^{CR}\boldsymbol{r}_i^j$ and ${}^{CL}\boldsymbol{r}_i^j$: 3D position of an arbitrary i-th point on j-th 3D model based on Σ_{CR} and Σ_{CL},
- ${}^{IL}\boldsymbol{r}_i^j$ and ${}^{IR}\boldsymbol{r}_i^j$: 2D projected position on Σ_{IL} and Σ_{IR} of an arbitrary i-th point on j-th 3D model.

The homogeneous transformation matrix from the right camera coordinate system Σ_{CR} to the target object coordinate system Σ_M is defined as ${}^{CR}\boldsymbol{T}_M({}^{H}\boldsymbol{\phi}_M^j, \boldsymbol{q})$, where ${}^{H}\boldsymbol{\phi}_M^j$ is j-th model's pose based on the robot hand coordinate system Σ_H and $\boldsymbol{q}$ means robot's joint angle vector. The pose of the j-th 3D model, including three position variables and three orientation

[DOI: 10.1299/jamdsm.2020jamdsm0057]

variables in quaternion based on Σ_H, are represented as

$$ {}^{H}\boldsymbol{\phi}_M^j = [{}^{H}x_M^j, {}^{H}y_M^j, {}^{H}z_M^j, {}^{H}\varepsilon_{1M}^j, {}^{H}\varepsilon_{2M}^j, {}^{H}\varepsilon_{3M}^j]^T. \tag{1} $$

For simplicity, the ${}^{H}\boldsymbol{\phi}_M^j$ is written as $\boldsymbol{\phi}_M^j$ hereafter.

${}^{CL}\boldsymbol{r}_i^j$ can be calculated by using Eq. (2),

$$ {}^{CL}\boldsymbol{r}_i^j = {}^{CL}\boldsymbol{T}_M(\boldsymbol{\phi}_M^j, \boldsymbol{q})\ {}^{M}\boldsymbol{r}_i^j, \tag{2} $$

where ${}^{M}\boldsymbol{r}_i^j$ is predetermined as fixed vectors since Σ_{Mj} is fixed on the j-th model. ${}^{CR}\boldsymbol{r}_i^j$ that represents the same i-th point on j-th model based on Σ_{CR} is also calculated by using ${}^{CR}\boldsymbol{T}_M(\boldsymbol{\phi}_M^j, \boldsymbol{q})$. Since $\boldsymbol{q}$ can be measured by robot's joint sensors, it could be thought to have been known, then $\boldsymbol{q}$ is omitted hereafter. Equation. (3) represents the projective transformation matrix $\boldsymbol{P}_k$,

$$ \boldsymbol{P}_k = \frac{1}{{}^{k}z_i}\begin{bmatrix} f/\eta_x & 0 & {}^{I}x_0 & 0 \\ 0 & f/\eta_y & {}^{I}y_0 & 0 \end{bmatrix}, \tag{3} $$

where,

- $k = CL, CR$,
- ${}^{k}z_i$: z-axis position of the i-th point in the camera sight direction in Σ_{CR} and Σ_{CL},
- f: focal length,
- η_x, η_y: [mm/pixel] in x-axis, and y-axis,
- ${}^{I}x_0, {}^{I}y_0$: [pixel] offset of origin of Σ_I.

The 2D position vector of the i-th point in the left camera image coordinates ${}^{IL}\boldsymbol{r}_i^j$ [pixel] can be described by using $\boldsymbol{P}_{CL}$ as,

$$ {}^{IL}\boldsymbol{r}_i^j = \boldsymbol{P}_{CL}\,{}^{CL}\boldsymbol{r}_i^j = \boldsymbol{P}_{CL}\,{}^{CL}\boldsymbol{T}_M(\boldsymbol{\phi}_M^j)\,{}^{M}\boldsymbol{r}_i^j. \tag{4} $$

Then, ${}^{IL}\boldsymbol{r}_i^j$ can be conceptually described by function $\boldsymbol{f}_L$ as,

$$ {}^{IL}\boldsymbol{r}_i^j(\boldsymbol{\phi}_M^j) = \boldsymbol{f}_L(\boldsymbol{\phi}_M^j, {}^{M}\boldsymbol{r}_i^j). \tag{5} $$

Like the description of ${}^{IL}\boldsymbol{r}_i^j$, ${}^{IR}\boldsymbol{r}_i^j$ can also be calculated as the same manner.

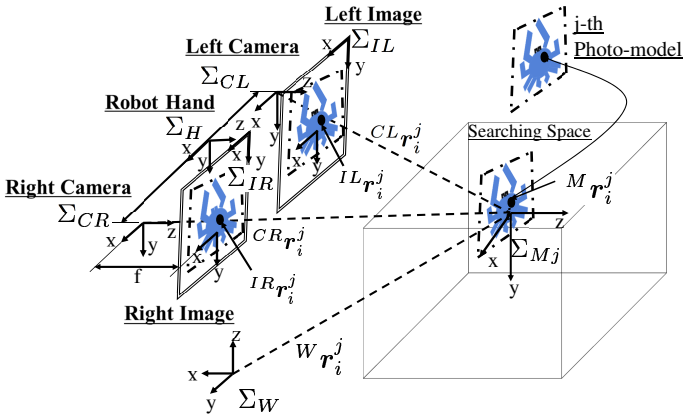


Fig. 3 Perspective projection of dual-eye vision-system. In the searching space, a j-th 3D solid model is represented by the picture of crab, which is defined by j-th model coordinate system Σ_{Mj}. The distance between Σ_{CL} and Σ_{CR}, i.e. baseline, is 323[mm].

3.2. Model generation

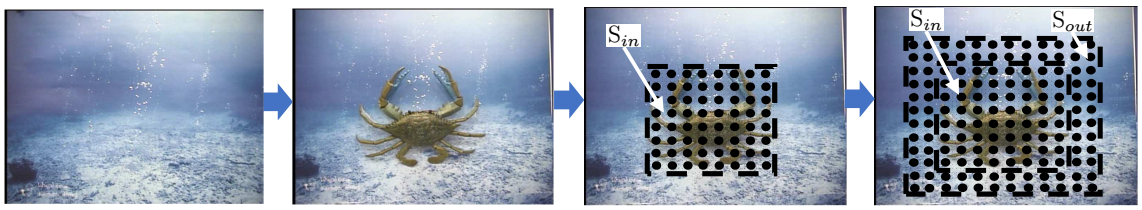
There are two main portions of the proposed system. The first portion is 2D model generation and the latter is relative pose estimation using the generated 2D model. This subsection is for a description of the first portion before an explanation of a matching method.

The hue value in HSV color representation is used for the extraction of the target color. The advantage of HSV is that each of its attributes corresponds directly to the basic color concepts, which makes it conceptually simple. Therefore,

[DOI: 10.1299/jamdsm.2020jamdsm0057]

it is easy to understand program for image matching process. And hue of HSV color system has a good robustness against the illumination intensity changing.

The model generation process is represented in Fig. 4. Firstly, a background image is captured and the averaged hue value of the background image is calculated as shown in Fig. 4 (a). Then, the 3D target crab model is put on the background as shown in Fig. 4 (b). As shown in Fig. 4 (c), the hue value of each point in the image constitutes the surface space S_{in} of the model. Finally, the outside space S_{out} of the model is generated by enveloping S_{in} as shown in Fig. 4 (d).



(a) Background (b) Target object in background (c) S_{in} model is set by black points group on the target (d) Strip area of S_{out} is set as it envelops S_{in}

Fig. 4 Model generation processes are described as (a)~(d): (a) shows a photograph of background image, (b) shows a photograph with a target object (the crab) in the background, (c) represents a surface space model S_{in} constituted by inner points group and (d) represents an outside space of model S_{out} that envelops S_{in}.

3.3. Orientation recognition method using quaternion

The methods widely used to represent the orientation of a 3D object are Euler angles, angle-axis representation, and rotation quaternion. Because the orientation singularities exist in the Euler angles and angle-axis representation methods, quaternion representation has been adopted in our previous research (Song et al., 2008). By axis-angle representation, a unit vector $\boldsymbol{l}$ indicates direction, and an angle θ describes the magnitude of rotation around $\boldsymbol{l}$. By using $\boldsymbol{l}$ and θ, quaternion $\boldsymbol{q} = \{\eta, \boldsymbol{\varepsilon}\}$ is defined as follows,

$$\boldsymbol{\varepsilon} = sin\frac{\theta}{2}\boldsymbol{l}, \tag{6}$$

where, $\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2, \varepsilon_3]^T, \boldsymbol{l} = [l_x, l_y, l_z]^T$. η is the scalar part of the quaternion, and $\boldsymbol{\varepsilon}$ is the vector part of it. This system uses a unit quaternion, then η needs to satisfy the following relationship: $\eta^2 + \boldsymbol{\varepsilon}^T\boldsymbol{\varepsilon} = 1$.

The quaternion in the system is based on Σ_H. Here, $\boldsymbol{\varepsilon}$ is short for ${}^H\boldsymbol{\varepsilon}_M$. And the homogeneous transformation of the frame Σ_H towards an object Σ_M is represented using $[\eta, \varepsilon]$ as

$${}^H\boldsymbol{T}_M = \begin{bmatrix} 1-2\varepsilon_2^2-2\varepsilon_3^2 & 2(\varepsilon_1\varepsilon_2-\eta\varepsilon_3) & 2(\varepsilon_1\varepsilon_3+\eta\varepsilon_2) & {}^Hx_M \\ 2(\varepsilon_1\varepsilon_2+\eta\varepsilon_3) & 1-2\varepsilon_1^2-2\varepsilon_3^2 & 2(\varepsilon_2\varepsilon_3-\eta\varepsilon_1) & {}^Hy_M \\ 2(\varepsilon_1\varepsilon_3-\eta\varepsilon_2) & 2(\varepsilon_2\varepsilon_3+\eta\varepsilon_1) & 1-2\varepsilon_1^2-2\varepsilon_2^2 & {}^Hz_M \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{7}$$

Then because of ${}^{CL}\boldsymbol{T}_M = {}^{CL}\boldsymbol{T}_H{}^H\boldsymbol{T}_M$, projective transformation of Eq. (4) can be calculated, since ${}^{CL}\boldsymbol{T}_H$ is known constant matrix from geometry of hand-eye mechanism. Then ${}^{IL}\boldsymbol{r}_i^j$ given by Eq. 4 can be calculated, and ${}^{IR}\boldsymbol{r}_i^j$ is also. About how is the quaternion used for controlling the motion of a manipulator, readers can refer to (Song et al., 2008).

3.4. 3D model-based matching

After the model generation process be completed, the model is used for recognizing the target object through model-based matching.

Figure 5 shows a generated photo model placed in the 3D searching space with assumed pose of $\boldsymbol{\phi}_M^j$ (sub figure on the top of Fig. 5) and the left and right 2D searching models that are projected from photo model with the pose being assumed to be $\boldsymbol{\phi}_M^j$ (sub figures on the left and right bottom of Fig. 5) respectively. In Fig. 5, a generated 2D photo-model is projected from the 3D space onto the left and right 2D searching planes. The sub figures on the top of Fig. 5 shows a generated 3D solid photo model $S_(\boldsymbol{\phi}_M^j)$ composed of $S_{in}(\boldsymbol{\phi}_M^j)$ (inner dotted points) and the outside space enveloping $S_{in}(\boldsymbol{\phi}_M^j)$ denoted as outer dotted line $S_{out}(\boldsymbol{\phi}_M^j)$. The sub figures on the left/right bottom of Fig. 5 show the left/right projected 2D

[DOI: 10.1299/jamdsm.2020jamdsm0057]

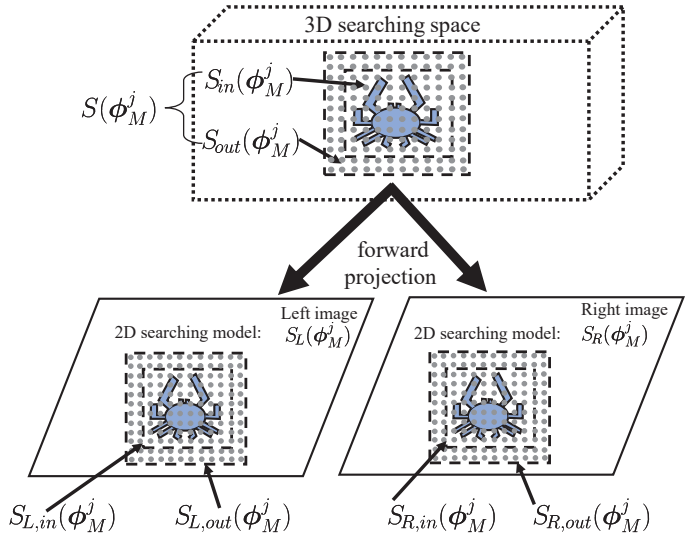


Fig. 5 A photo model $S(\boldsymbol{\phi}_M^j)$ in the 3D searching space on the top of this figure is a 2D model but it has 3D pose information $\boldsymbol{\phi}_M^j$. The left and right 2D searching models represented as $S_L(\boldsymbol{\phi}_M^j)$ and $S_R(\boldsymbol{\phi}_M^j)$ on the left/right bottom, are calculated by forward projection from the 2D photo-model $S(\boldsymbol{\phi}_M^j)$.

searching models $S_L(\boldsymbol{\phi}_M^j)$ and $S_R(\boldsymbol{\phi}_M^j)$ respectively. Both $S_L(\boldsymbol{\phi}_M^j)$ and $S_R(\boldsymbol{\phi}_M^j)$ consist of inner and outer portions $S_{L,in}(\boldsymbol{\phi}_M^j)$, $S_{L,out}(\boldsymbol{\phi}_M^j)$ and $S_{R,in}(\boldsymbol{\phi}_M^j)$, $S_{R,out}(\boldsymbol{\phi}_M^j)$.

3.5. Definition of the fitness function

An overlap degree, that means correlation degree, between a projected model and the target in images captured by the dual-eye cameras is used as a fitness (Minami et al., 2003). The highest fitness value represents the best pose of the model $\widehat{\boldsymbol{\phi}}$ among $\boldsymbol{\phi}_M^j$ that coincides with the crab's pose in 3D space as depicted at top of Fig. 5.

A model is composed of some sampling points. The number of them is "n." After forward projection, as shown in Fig. 3, each point coordinate in left image Σ_{IL} is ${}^{IL}\boldsymbol{r}_i^j$. And evaluation value of each point ${}^{IL}\boldsymbol{r}_i^j$ in inner portion of the model (${}^{IL}\boldsymbol{r}_i^j \in S_{R,in}(\boldsymbol{\phi}_M^j)$) is $p_{L,in}({}^{IL}\boldsymbol{r}_i^j)$ calculated by Eq. (8). The one of outer portion (${}^{IL}\boldsymbol{r}_i^j \in S_{L,out}(\boldsymbol{\phi}_M^j)$) is $p_{L,out}({}^{IL}\boldsymbol{r}_i^j)$ calculated by Eq. (9)

$$p_{L,in}({}^{IL}\boldsymbol{r}_i^j) = \begin{cases} 2, \text{ if}(|H_{IL}({}^{IL}\boldsymbol{r}_i^j) - H_{ML}({}^{IL}\boldsymbol{r}_i^j)| \le 30); \\ -0.005, \quad \text{else if}(|\bar{H}_B - H_{IL}({}^{IL}\boldsymbol{r}_i^j)| \le 30); \\ 0, \qquad\quad \text{otherwise}; \end{cases} \tag{8}$$

$$p_{L,out}({}^{IL}\boldsymbol{r}_i^j) = \begin{cases} 0.1, & \text{if}(|\bar{H}_B - H_{IL}({}^{IL}\boldsymbol{r}_i^j)| \le 20); \\ -0.5, & \text{otherwise}; \end{cases} \tag{9}$$

where

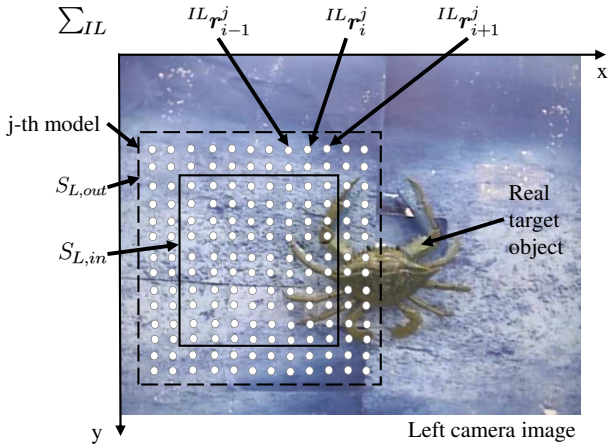
- $H_{IL}({}^{IL}\boldsymbol{r}_i^j)$: the hue value of the left camera image at the point ${}^{IL}\boldsymbol{r}_i^j$ (i-th point in j-th photo model, lying in $S_{L,in}$),
- $H_{ML}({}^{IL}\boldsymbol{r}_i^j)$: the hue value of photo model at the point ${}^{IL}\boldsymbol{r}_i^j$ (i-th point in $S_{L,in}$),
- $\bar{H}_B$: the average hue value of the background image.

The $p_{R,in}({}^{IR}\boldsymbol{r}_i^j)$ and $p_{R,out}({}^{IR}\boldsymbol{r}_i^j)$ are defined as the same above manner. The fitness $F(\boldsymbol{\phi}_M^j)$ of a model is calculated as Eq. (10), and its abbreviated form is Eq. (11),

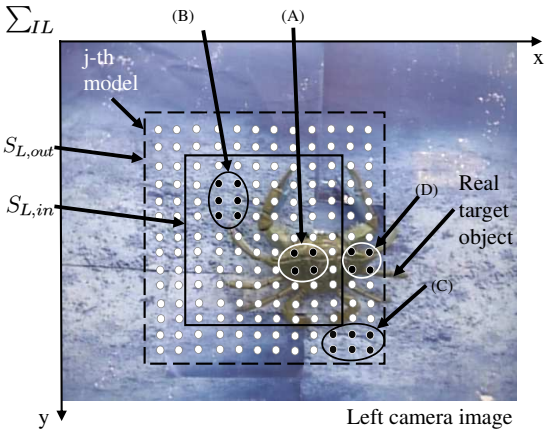
[DOI: 10.1299/jamdsm.2020jamdsm0057]

$$F(\boldsymbol{\phi}_M^j)=\left\{\left(\sum_{\substack{{}^{IR}\boldsymbol{r}_i^j\in\\ S_{R,in}(\boldsymbol{\phi}_M^j)}} p_{R,in}({}^{IR}\boldsymbol{r}_i^j)+\sum_{\substack{{}^{IR}\boldsymbol{r}_i^j\in\\ S_{R,out}(\boldsymbol{\phi}_M^j)}} p_{R,out}({}^{IR}\boldsymbol{r}_i^j)\right)/n+\left(\sum_{\substack{{}^{IL}\boldsymbol{r}_i^j\in\\ S_{L,in}(\boldsymbol{\phi}_M^j)}} p_{L,in}({}^{IL}\boldsymbol{r}_i^j)+\sum_{\substack{{}^{IL}\boldsymbol{r}_i^j\in\\ S_{L,out}(\boldsymbol{\phi}_M^j)}} p_{L,out}({}^{IL}\boldsymbol{r}_i^j)\right)/n\right\}/2 \tag{10}$$

$$=\left\{F_R(\boldsymbol{\phi}_M^j)+F_L(\boldsymbol{\phi}_M^j)\right\}/2. \tag{11}$$



(a) Evaluation position ${}^{IL}\boldsymbol{r}_i^j$, that is i-th point of j-th model, which is projected on left image whose pose $\boldsymbol{\phi}_M^j$ is given by evolutionary process of GA.



(b) Classification of evaluation points (A)~(D) on the photo model is explained. (A) represents points that satisfy the first case of Eq. (8), $|H_{IL}({}^{IL}\boldsymbol{r}_i^j)-H_{ML}({}^{IL}\boldsymbol{r}_i^j)|\leq 30$, representing that inner model $S_{L,in}$ overlaps with the real target. (B) does $|\bar{H}_B-H_{IL}({}^{IL}\boldsymbol{r}_i^j)|\leq 30$, representing that inner model $S_{L,in}$ overlaps with background. (C) does $|\bar{H}_B-H_{IL}({}^{IL}\boldsymbol{r}_i^j)|\leq 20$, meaning that the outer model $S_{L,out}$ overlaps with background, and (D) shows $S_{L,out}$ overlaps with the real target.

Fig. 6 Calculation of the matched degree of each point in model space ($S_{L,in}$ and $S_{L,out}$).

The fixed values in Eqs. (8) and (9) have been tuned experimentally to provide a peak in the fitness value distribution at the true pose. Figure 6 (a) shows j-th model, the evaluation points of hue value, $\cdots {}^{IL}\boldsymbol{r}_{i-1}^j$, ${}^{IL}\boldsymbol{r}_i^j$, ${}^{IL}\boldsymbol{r}_{i+1}^j\cdots$, are indicated by white dots in inside area $S_{L,in}$, and those in outside strip $S_{L,out}$. Figure 6 (b) shows another situation that the overlapping area of real crab and the model increased than the one depicted in (a). The hue value of the left camera input image at the point ${}^{IL}\boldsymbol{r}_i^j$ is represented by $H_{IL}({}^{IL}\boldsymbol{r}_i^j)$. The i-th point of j-th model in $S_{L,in}$ and $S_{L,out}$ and the hue value of the same point ${}^{IL}\boldsymbol{r}_i^j$ on the model is defined as $H_{ML}({}^{IL}\boldsymbol{r}_i^j)$. The average hue value of background calculated from Fig. 4 (a) is defined as $\bar{H}_B$.

In Eq. (8), if the hue value of each point on 3D target in left images, $H_{IL}({}^{IL}\boldsymbol{r}_i^j)$, which lies inside the surface model frame $S_{L,in}$, and the hue value of corresponding same point in a model, $H_{ML}({}^{IL}\boldsymbol{r}_i^j)$, have similar values with a tolerance less than 30, that is $|H_{IL}({}^{IL}\boldsymbol{r}_i^j)-H_{ML}({}^{IL}\boldsymbol{r}_i^j)|\leq 30$ then this means that model's hue value and input target crab's hue value have close hue distance at the same checking point of ${}^{IL}\boldsymbol{r}_i^j$. This represents photo model overlaps to the real crab projected in left camera image in S_{in}, which are represented by dots designated by (A) in Fig. 6 (b). In this case the fitness value would be increased with the voting value of "+2." The fitness value will decrease with the value of "-0.005" for every point ${}^{IL}\boldsymbol{r}_i^j$ in S_{in} by the condition, $|\bar{H}_B-H_{IL}({}^{IL}\boldsymbol{r}_i^j)|\leq 30$ in Eq. (8), when model's crab area overlaps with blue background. This represents that the model does not overlap precisely the target in the input image, which are represented by (B) in Fig. 6 (b). In this case, "-0.005" is given as a penalty to decrease F. Otherwise, the fitness value will be "0."

Similarly, in Eq. (9), if the hue value of each point in the left camera image lying in $S_{L,out}$ has similar value to the average hue value of background $\bar{H}_B$ calculated from Fig. 4 (a) with the tolerance of 20, the fitness value will be increased with the value of "+0.1." This means $S_{L,out}$ strip area surrounding $S_{L,in}$ overlaps the background, expressing the model and the crab overlap rather correctly as (C) in Fig. 6 (b). Since this situation means that the model's position and orientation matches to the real crab, plus points "0.1" is given to the function $p_{L,out}$, which is described in Eq. (9). Otherwise, the fitness value will decrease with the penalty value of "-0.5." This represents points on $S_{L,out}$ overlaps with the real crab as (D) in Fig. 6 (b).

[DOI: 10.1299/jamdsm.2020jamdsm0057]

3.6. Real-time Multi-step Genetic Algorithm (RM-GA)

The main problem of identifying the pose of the object can be converted into an optimization problem if the fitness function has been designed to give the maximum value only in the case that the model whose pose coincides with the target object in the 3D space. Several optimization methods can search the maximum value of the evaluation function. For real-time recognition in dynamic images input with frame rate 30[fps], we have proposed a Real-time Multi-step Genetic Algorithm (RM-GA) (Myint et al., 2017). RM-GA evaluation process is applied to find the maximum value as an optimal solution because of its simplicity and effectiveness. The 20 individuals of RM-GA are used in this experiment, where the chromosome of an individual consists of 72[bit] with six variables. Each variable is coded by 12[bit] as shown in Eq. (12), the first three variables of a model (${}^H x_M^j$, ${}^H y_M^j$, ${}^H z_M^j$) represents the position in 3D space and the last three variables (${}^H \varepsilon_{1M}^j$, ${}^H \varepsilon_{2M}^j$, ${}^H \varepsilon_{3M}^j$) represents the orientation. The genes of RM-GA representing possible pose solution is defined as below;

$$\overbrace{\underbrace{01\cdots01}_{12bits}}^{{}^H x_M^j}\overbrace{\underbrace{00\cdots01}_{12bits}}^{{}^H y_M^j}\overbrace{\underbrace{11\cdots01}_{12bits}}^{{}^H z_M^j}\overbrace{\underbrace{01\cdots01}_{12bits}}^{{}^H \varepsilon_{1M}^j}\overbrace{\underbrace{01\cdots11}_{12bits}}^{{}^H \varepsilon_{2M}^j}\overbrace{\underbrace{01\cdots10}_{12bits}}^{{}^H \varepsilon_{3M}^j}. \tag{12}$$

As the searching result of RM-GA, the output best individual is defined as,

$${}^H \boldsymbol{\phi}_{\widehat{M}} = [{}^H x_{\widehat{M}}, {}^H y_{\widehat{M}}, {}^H z_{\widehat{M}}, {}^H \varepsilon_{\widehat{1M}}, {}^H \varepsilon_{\widehat{2M}}, {}^H \varepsilon_{\widehat{3M}}]^T. \tag{13}$$

Figure 7 (a) shows the process flow in the RM-GA in which 3D models converge into the real 3D solid target object. In Fig. 7 (a), a target object is a crab, and each 3D model is depicted as a rectangle with dotted lines including the same shape and same color information of the target. But each model has different poses $\boldsymbol{\phi}_M^j$(j=1, 2, $\cdots$, 20) as shown at the top of Fig. 7 (a) whose pose has been defined by the above chromosome, Eq. (12). Note that the system performs the evaluation process in the left and right 2D image planes. And the convergence of searching models occurs in 3D searching space. The fitness function value evaluates the overlap degree between an individual and the target object. The fitter ones are selected to regenerate the next genes. Thus, the genes converges to the real target after some transient period of evolutions. Then, the gene that gives the highest fitness value stands for the most trustful pose as shown in the bottom part in Fig. 7 (a).

Figure 7 (b) shows a flowchart for the RM-GA evolution process for recognition and pose estimation:

(1)Firstly, the individuals are randomly generated in the 3D searching area as the first generation.

(2)New images captured by dual-eye cameras are input.

(3)The fitness value of every individual is calculated.

(4)Every individual's fitness value is sorted by the calculated fitness value.

(5)The best individual is selected from the current population, and the weak individuals are removed.

(6)Then, the individuals for the next generation is reproduced by making crossover and mutation between the selected individuals.

(7)Only new individuals in the next generation are evaluated by the fitness function, shown by "Evaluation (2)" block, because the right and left images do not change and top individuals with highest fitness do not need to calculate fitness again since the image is constant during 33[ms].

(8)And then, the above procedures are repeated within 33 [ms]. Because the time needed for transferring one frame of video from image input board to the memory of main CPU is 9[ms] , the remaining time within the video rate 33[ms] is 33 − 9 = 24[ms]. Then 24[ms] remains for RM-GA to evolve.

(9)Finally, the RM-GA output the best individual, then repeat the above (8) until input a new image.

Using Lyapunov analysis in (Song et al., 2010), the time convergence performance of the RM-GA in successively input dynamic images has been confirmed experimentally. Real-time 3D pose estimation using 3D-model-based recognition and the RM-GA has been presented in detail in a previous paper (Myint et al., 2016). Real-time pose estimation using RM-GA and docking performance of ROV is discussed in an earlier study (Myint et al., 2017).

[DOI: 10.1299/jamdsm.2020jamdsm0057]

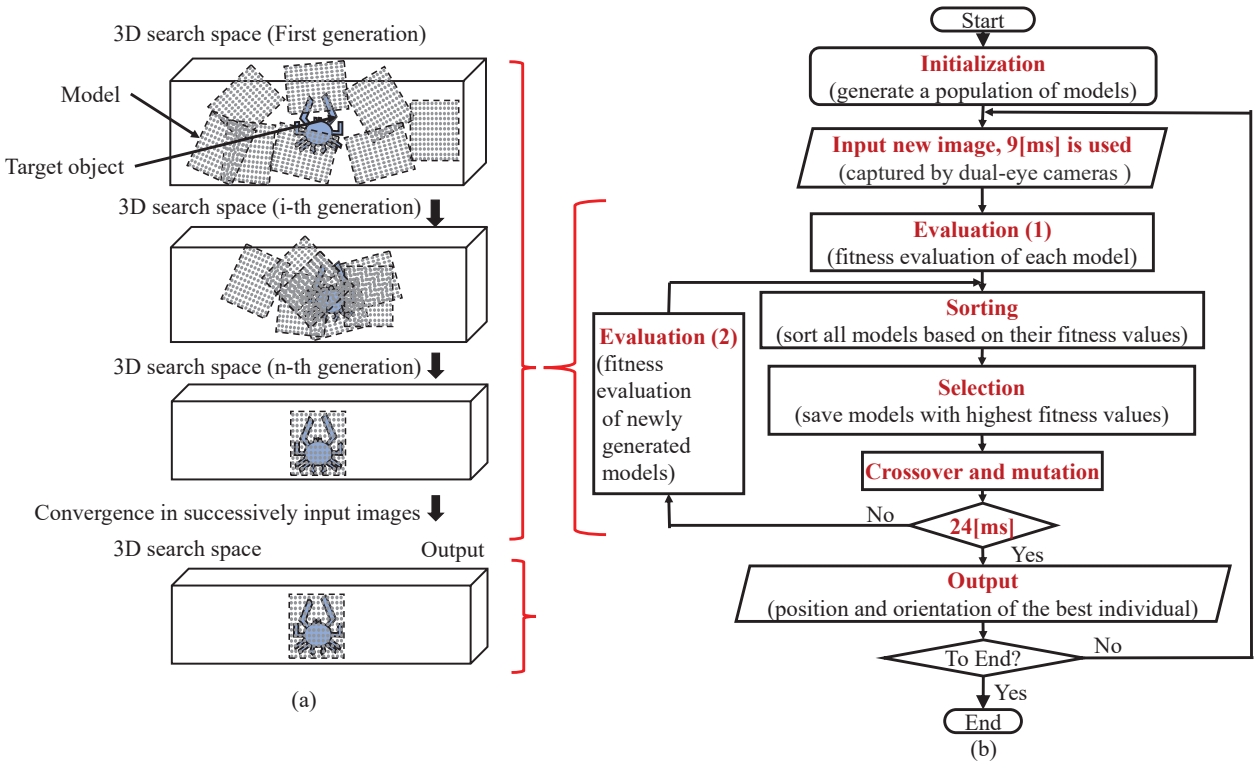


Fig. 7 RM-GA evolution process in which 3D models with random poses converge to the real 3D solid target object in 3D space. The pose of the model with the highest fitness value represents the estimated pose of the target object at that instant: (a) schematic diagram of the evolutionary process and (b) flowchart of RM-GA process during each 33[ms] control period, from "Input new image" to "Output."

4. Experimental environment

The utilized manipulator in the system is a PA-10 robot arm manufactured by Mitsubishi Heavy Industries. And two CCD cameras mounted on the end-effector are FCB-1X11A manufactured by Sony Corporation. The resolution of dynamic images is 640×480 [pixel]. The frame frequency of stereo cameras is set as 30[fps]. The image processing board receiving the image from the CCD camera is connected to the host computer (CPU: Intel Core i7-3770, 3.40[GHz]).

The pose of target 3D model's pose based on Σ_H is set as

$$ {}^H\boldsymbol{\phi}_M = [{}^H\boldsymbol{r}_M^T, {}^H\boldsymbol{\varepsilon}_M^T]^T = [{}^Hx_M, {}^Hy_M, {}^Hz_M, {}^H\varepsilon_{1M}, {}^H\varepsilon_{2M}, {}^H\varepsilon_{3M}]^T = [0, 0, 500[\mathrm{mm}], 0, 0, 0]^T, \tag{14} $$

meaning PT-robot in Fig. 2 is set in front of TC-robot with the relative pose of above ${}^H\boldsymbol{\phi}_M$.

As shown in Fig. 8, 12 different sea creature toys are prepared as 3D target objects whose code names are from C01 to C12. The table includes the English name and the size of each 3D toy. Figure 9 shows photo-models with blue sea background. The size of each picture is 640 × 480 [pixel]. Each dashed line rectangle indicates a photo-model used in pose estimation of 3D toy targets. The photo-model is only part of a picture including a target shape as shown by the rectangles in Fig. 9.

5. Fitness distribution experiments

Fitness function Eq. (10) converts the target recognition and pose estimation problem into an optimization problem if variables to give the maximum peak represents the target's pose. To ensure whether this problem conversion about Eq. (10) is feasible, a way is a brute-force search or an exhaustive search. Using still pictures at an instant moment, the fitness value $F(\boldsymbol{\phi}_M^j)$ is calculated with its pose varied as parameters. We call it "fitness distribution." It is also a way to verify whether the RM-GA can detect the true pose of a target object at that moment. Even though the fitness distribution is made by an exhaustive search method, it is impossible to calculate all possibilities. This time, the position incremental distance of fitness value is set at 1.0[mm], and the orientation increment is 0.01[] (quaternion does not have the unit). Search

[DOI: 10.1299/jamdsm.2020jamdsm0057]

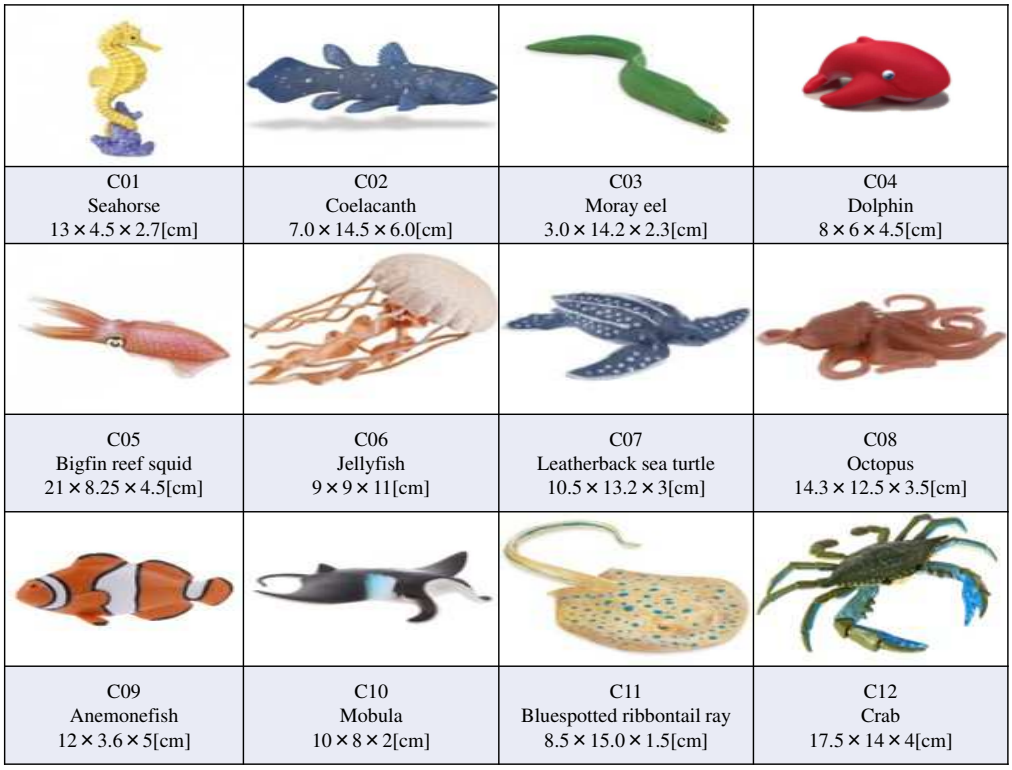


Fig. 8 Twelve marine biological creature models. The code name is from C01 to C12. The second line of each frame shows the English name. And the last line shows the size of each 3D toy (unit: [cm]).

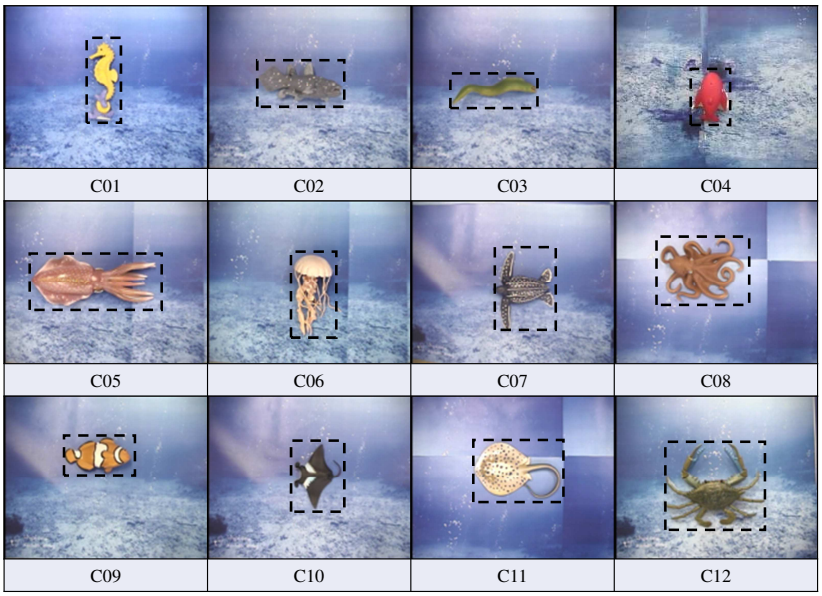


Fig. 9 Twelve pictures of marine biological creature models are shown with blue sea background corresponding to Fig. 8. The code name is from C01 to C12. The size of each picture is 640 × 480 [pixel]. Each dashed line rectangle indicates a photo-model.

[DOI: 10.1299/jamdsm.2020jamdsm0057]

ranges of fitness distribution are set as position: ${}^{H}x_M$ and ${}^{H}y_M \in [-180, 180]$[mm], ${}^{H}z_M \in [320, 680]$[mm]; orientation: ${}^{H}\varepsilon_{1M}$, ${}^{H}\varepsilon_{2M}$, and ${}^{H}\varepsilon_{3M} \in [-0.37, 0.37]$.

Figures 10 and 11 show left and right images captured by the stereo-vision system and the fitness distribution of C04 dolphin and C12 crab in detail. Figure 10 (a) shows the left and right camera images of C04 dolphin, and (b), (c) show the $x-y$ and $y-z$ position fitness distribution respectively, and (d), (e) show the orientation fitness distribution. All the fitness distribution (b)~(e) have peaks. For example, in Fig. 10 (b), the $x-y$ position that gives maximum peak is $({}^{H}x_M, {}^{H}y_M) = (-3, 7)$[mm] and this result shows it is near the true position $(0, 0)$[mm] given by Eq. (14). About another object C12 crab, Fig. 11 (b) and (c) shows the position fitness distribution, and (d) and (e) show the orientation fitness distribution. All the fitness distribution (b)~(e) also have peaks near the true value. The results of other target objects except of C04 and C12 are similar to Figs. 10 and 11, then they are not listed in this paper. Each subfigure of the results has a main peak near the true value ${}^{H}\boldsymbol{\phi}_M$ given by Eq. (14). Therefore, it has been confirmed that fitness function Eq. (10) can convert the target recognition and pose estimation problem into an optimization problem. Furthermore, it has been confirmed that the proposed method can estimate 3D target pose by using stereo-vision and 2D photo-model. But the gentle shapes of peaks given by (d) and (e) in Figs. 10 and 11 mean that the estimated orientations tend to include estimation errors than the positions whose fitness distributions have sharp peaks as shown in Figs. 10 and 11.

RM-GA searching experiments have been also conducted to compare with the fitness distribution. The results show that RM-GA can find the pose of all target objects from C01 to C12 in less than 10[s] by using the left and right still images. In this experiment, the optimization procedure is conducted by static still photographs not dynamic images, then the RM-GA process means usual GA process practically. For example, the left and right camera images shown at Fig. 10 (a) are used for the RM-GA searching experiment concerning C04 dolphin. And the detected pose by RM-GA ${}^{H}\boldsymbol{\phi}_{\widehat{M}} = [{}^{H}x_{\widehat{M}}, {}^{H}y_{\widehat{M}}, {}^{H}z_{\widehat{M}}, {}^{H}\varepsilon_{\widehat{1M}}, {}^{H}\varepsilon_{\widehat{2M}}, {}^{H}\varepsilon_{\widehat{3M}}]^T$ is shown at the row of C04 in Table 1, which includes also results of other 3D toys shown in Fig. 8. The real pose that gives maximum peak is represented by ${}^{H}\boldsymbol{\phi}_M = [{}^{H}x_M, {}^{H}y_M, {}^{H}z_M, {}^{H}\varepsilon_{1M}, {}^{H}\varepsilon_{2M}, {}^{H}\varepsilon_{3M}]^T$, and is shown in the same row. The detection errors $\Delta\boldsymbol{\phi} = {}^{H}\boldsymbol{\phi}_M - {}^{H}\boldsymbol{\phi}_{\widehat{M}} = [\Delta x, \Delta y, \Delta z, \Delta\varepsilon_1, \Delta\varepsilon_2, \Delta\varepsilon_3]^T$ is also listed in the table. From the error values of C01~C12, it has been confirmed that the poses of all 3D toys could be estimated by GA evolutional procedures with the position error being less than 10[mm] and orientation error being less than 0.15 in quaternion.

In this section, by the fitness distribution experiment, it is verified that the fitness function Eq. (10) can transform the target recognition and orientation estimation problems into optimization problems. It is also confirmed that the proposed method can estimate the 3D target pose by using stereo-vision and 2D photo-model. Since the estimated value of RM-GA is close to the peak result in the fitness distribution experiment, RM-GA can be used practically as a solution to detect the pose of the 3D target objects by using 2D photo-model.

6. Real-time 3D pose estimation experiment with RM-GA

6.1. Experimental content

The initial condition of the pose real-time estimation experiment is shown at top subfigure (Step 0) in Fig. 12. The pose of the target object represented by Σ_M based on the end-effector Σ_H is set as Eq. (14). In Fig. 12, each subfigure, i.e., each (Step), is a state of the target object at a special time point in the experiment. For example, subfigure (Step 0) shows the state of the target object at the beginning time point $t = 0$[s] of the experiment. And the target's pose of (Step 0) ${}^{H}\boldsymbol{\phi}_M = [{}^{H}x_M, {}^{H}y_M, {}^{H}z_M, {}^{H}\varepsilon_{1M}, {}^{H}\varepsilon_{2M}, {}^{H}\varepsilon_{3M}]^T = [0, 0, 500\text{[mm]}, 0, 0, 0]^T$ is also shown in Table 2. In this pose estimation experiment, the PT-robot in Fig. 2 does not move. The TC-robot controls the target object to move, with one of the elements of target pose of $({}^{H}x_M, {}^{H}y_M, {}^{H}z_M, {}^{H}\varepsilon_{1M}, {}^{H}\varepsilon_{2M}, {}^{H}\varepsilon_{3M})$ being changed and others being kept to be constant as shown in Table 2. The table lists the pose of TC-robot and the transition of the pose when the target pose is changed from (Step 0) to (Step 19).

For example, as shown in Fig. 12, from (Step 0) to (Step 1), ${}^{H}x_M$ that is x-coordinate of target pose, is changed from 0[mm] to -50[mm] by TC-robot based on the Σ_M. ${}^{H}y_M$, ${}^{H}z_M$, and orientation parameters ${}^{H}\boldsymbol{\varepsilon}_M$ are constant. In subfigure (Step 1) of Fig. 12, the arrow shows the moving direction along the x-axis from former (Step 0) to this (Step 1). And as shown in Table 2, the arrow between rows (Step 0) and (Step 1) in the column of ${}^{H}x_M$ has the same meaning and indicates only ${}^{H}x_M$ is changed from 0[mm] at (Step 0) to -50[mm] at (Step 1). In Fig. 12, from (Step 1) to (Step 2), ${}^{H}y_M$ that is y-coordinate of target pose, is changed from 0[mm] to -50[mm] by TC-robot. ${}^{H}x_M$, ${}^{H}z_M$, and orientation parameters ${}^{H}\boldsymbol{\varepsilon}_M$ at (Step 2) are the same with the parameters at (Step 1). And the arrow in subfigure (Step 2) in Fig. 12 shows the moving direction along the y-axis. And as shown in Table 2, the arrow between rows (Step 1) and (Step 2) in the column of ${}^{H}y_M$ also represents that only ${}^{H}y_M$ is changed from 0[mm] at (Step 1) to -50[mm] at (Step 2) by TC-robot. The position of

[DOI: 10.1299/jamdsm.2020jamdsm0057]

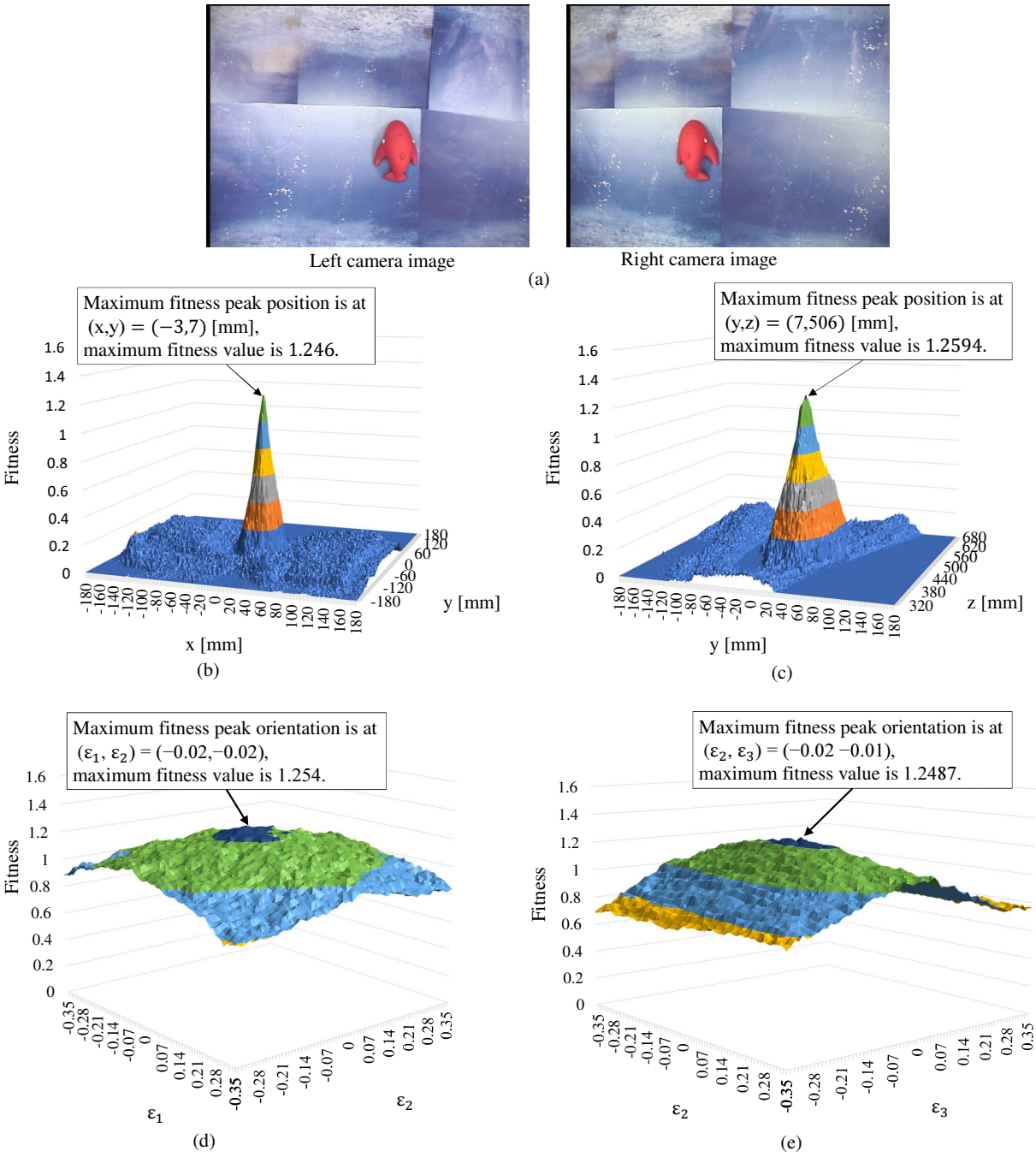


Fig. 10 Fitness distribution of C04 dolphin. (a) Left and right camera images, (b) fitness distribution in the x-y plane, (c) fitness distribution in the y-z plane, (d) fitness distribution of orientation in ε_1-ε_2, and (e) fitness distribution of orientation in ε_2-ε_3. In each subfigure of (b)~(e), the maximum fitness and corresponding coordinate are shown in a text box.

[DOI: 10.1299/jamdsm.2020jamdsm0057]

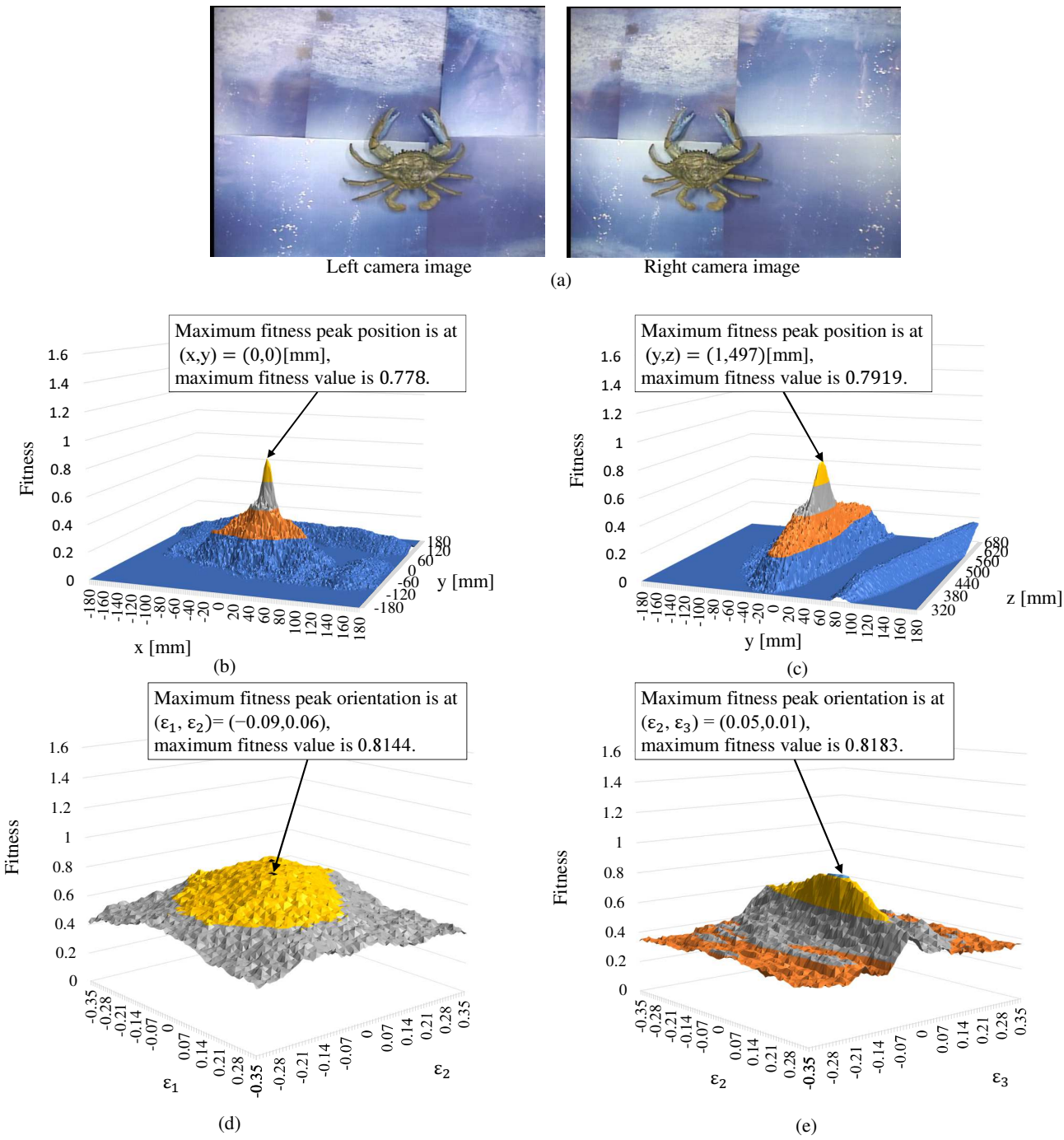


Fig. 11 Fitness distribution of C12 crab. (a) Left and right camera images, (b) fitness distribution in the x-y plane, (c) fitness distribution in the y-z plane, (d) fitness distribution of orientation in ε_1-ε_2, and (e) fitness distribution of orientation in ε_2-ε_3. In each subfigure of (b)~(e), the maximum fitness and corresponding coordinate are shown in a text box.

[DOI: 10.1299/jamdsm.2020jamdsm0057]

Table 1 Peak coordinates ${}^H\boldsymbol{\phi}_M = [{}^Hx_M, {}^Hy_M, {}^Hz_M, {}^H\varepsilon_{1M}, {}^H\varepsilon_{2M}, {}^H\varepsilon_{3M}]^T$ of 12 target objects in the fitness distribution, RM-GA detection results ${}^H\boldsymbol{\phi}_{\widehat{M}} = [{}^Hx_{\widehat{M}}, {}^Hy_{\widehat{M}}, {}^Hz_{\widehat{M}}, {}^H\varepsilon_{\widehat{1M}}, {}^H\varepsilon_{\widehat{2M}}, {}^H\varepsilon_{\widehat{3M}}]^T$ and errors $\Delta\boldsymbol{\phi}_M = {}^H\boldsymbol{\phi}_M - {}^H\boldsymbol{\phi}_{\widehat{M}} = [\Delta x, \Delta y, \Delta z, \Delta\varepsilon_1, \Delta\varepsilon_2, \Delta\varepsilon_3]^T$ are listed. Search range of fitness distribution, position: $x \in [-180, 180]$[mm], $y \in [-180, 180]$[mm], $z \in [320, 680]$[mm]; orientation: $\varepsilon_1, \varepsilon_2$, and $\varepsilon_3 \in [-0.37, 0.37]$. Search interval of fitness are 1.0[mm] in position; orientation: 0.01[]. True values given by TC-robot shown in Fig. 2 are ${}^H\boldsymbol{\phi}_M = [{}^Hx_M, {}^Hy_M, {}^Hz_M, {}^H\varepsilon_{1M}, {}^H\varepsilon_{2M}, {}^H\varepsilon_{3M}]^T = [0, 0, 500\text{[mm]}, 0, 0, 0]^T$.

Pose / Target Number	Real pose that gives maximum peak						Detected pose by RM-GA						Error values					
	Position [mm]			Orientation (quaternion[])			Position [mm]			Orientation (quaternion[])			Position [mm]			Orientation (quaternion[])		
	${}^Hx_M$	${}^Hy_M$	${}^Hz_M$	${}^H\varepsilon_{1M}$	${}^H\varepsilon_{2M}$	${}^H\varepsilon_{3M}$	${}^Hx_{\widehat{M}}$	${}^Hy_{\widehat{M}}$	${}^Hz_{\widehat{M}}$	${}^H\varepsilon_{\widehat{1M}}$	${}^H\varepsilon_{\widehat{2M}}$	${}^H\varepsilon_{\widehat{3M}}$	Δx	Δy	Δz	$\Delta\varepsilon_1$	$\Delta\varepsilon_2$	$\Delta\varepsilon_3$
C01	-2.0	0.0	501.0	-0.03	-0.09	0.04	-2.25	0.39	497.62	-0.03	-0.15	0.04	0.25	-0.39	3.38	0.00	0.06	0.00
C02	5.0	-3.0	494.0	0.11	-0.18	-0.06	12.11	-2.64	495.57	0.10	-0.17	-0.05	-7.11	-0.36	-1.57	0.01	-0.01	-0.01
C03	-2.0	5.0	508.0	0.01	-0.1	0.03	-2.54	4.98	509.24	0.03	-0.07	0.02	0.54	0.02	-1.24	-0.02	-0.03	0.01
C04	-3.0	7.0	506.0	-0.02	-0.02	-0.01	-3.13	7.13	506.41	-0.03	-0.04	-0.04	0.13	-0.13	-0.41	0.01	0.02	0.03
C05	-1.0	-11.0	500.0	0.02	0.02	0.03	-0.39	-10.45	500.55	0.02	0.02	0.04	-0.61	-0.55	-0.55	0.00	0.00	-0.01
C06	-2.0	5.0	493.0	0.11	0.11	0.07	-3.13	5.66	494.30	0.16	0.04	0.02	1.13	-0.66	-1.30	-0.05	0.07	0.05
C07	12.0	4.0	517.0	0.03	-0.01	-0.01	9.38	4.79	514.22	-0.03	-0.08	-0.01	2.63	-0.79	2.78	0.06	0.07	0.00
C08	-6.0	-1.0	504.0	0.02	-0.02	-0.07	-6.05	-2.25	502.11	0.08	-0.07	-0.08	0.05	1.25	1.89	-0.06	0.05	0.01
C09	6.0	-6.0	502.0	-0.1	0.09	-0.04	3.81	-6.05	498.20	-0.06	0.09	-0.04	2.19	0.05	3.80	-0.04	0.00	0.00
C10	6.0	6.0	507.0	0.01	0.04	-0.04	6.35	4.00	504.45	-0.06	-0.10	-0.04	-0.35	2.00	2.55	0.07	0.14	0.00
C11	9.0	1.0	513.0	0.04	0.16	-0.06	7.62	1.27	509.14	0.05	0.09	-0.05	1.38	-0.27	3.86	-0.01	0.07	-0.01
C12	0.0	1.0	497.0	-0.09	0.06	0.01	0.29	1.76	498.50	0.06	-0.04	0.03	-0.29	-0.76	-1.50	-0.15	0.10	-0.02

TC-robot is changed with the same manner from (Step 2) to (Step 10), which represents the same pose as (Step 0).

From (Step 11) to (Step 19), the position of the TC-robot is kept to be constant, but the orientation is changed. The TC-robot rotates the target to $\varepsilon_{1M} = 0.174$ around x_M axis at (Step 11) and then to $\varepsilon_{1M} = -0.174$ around x_M axis at (Step 12). And at (Step 13) the target is rotated back to the initial pose of (Step 10). From (Step 14) to (Step 16), the target object rotates only around y-axis and from (Step 17) to (Step 19) it does around z-axis. The poses of Σ_M at (Step 10), (Step 13), (Step 16), and (Step 19) are the same with the initial state (Step 0).

Table 2 shows the real pose of the target object at each step. The position trajectories of the target are shown at the subfigures from (Step 0) to (Step 10) in Fig. 12 and the time profiles of target pose given by TC-robot are depicted as (a) to (f) at the center of Fig. 12. Target's pose time profiles (a)~(f) are enlarged and shown as dashed lines in Fig. 13. The solid lines in Fig. 13 shows the pose estimation results. The 3D pose estimation error is shown as Fig. 14.

6.2. Results and discussion of pose estimation experiment

Figure 13 (a)~(f) shows the pose estimation results $[{}^Hx_{\widehat{M}}, {}^Hy_{\widehat{M}}, {}^Hz_{\widehat{M}}, {}^H\varepsilon_{\widehat{1M}}, {}^H\varepsilon_{\widehat{3M}}, {}^H\varepsilon_{\widehat{3M}}]^T$ depicted with solid lines. (a)~(c) are position recognition results. (d)~(f) are orientation recognition results. The true values ${}^H\boldsymbol{\phi}_M = [{}^Hx_M, {}^Hy_M, {}^Hz_M, {}^H\varepsilon_{1M}, {}^H\varepsilon_{2M}, {}^H\varepsilon_{3M}]^T$ are shown as dashed lines, which is from Fig. 12. The descriptions of (Step 0)~(Step 19) in Fig. 13, where "Step" has been eliminated to save space, are the time points corresponding to those in Fig. 12. In the beginning period of recognition time $t = 0 \sim 6$[s], the detection results of RM-GA gradually converge to the true pose ${}^H\boldsymbol{\phi}_M$. Then, the estimation results are almost similar to the real pose. Even though the detection result ${}^Hz_{\widehat{M}}$ in (c) have some fluctuations when the target moves along the x- or y-axis at (Step 1), (Step 4), (Step 6), and (Step 8), RM-GA can quickly converge to the true pose in the later. The position estimation results in (a)~(c) shows that the proposed method can track the position of the moving target object. The orientation estimation results in the period of (Step 11)~(Step 13) in (d), (Step 14)~(Step 16) in (e), and (Step 17)~(Step 19) in (f) show that this method can also track the changing orientation of the target in real-time.

Figure 14 shows the errors of the pose tracking results. And the detection errors of x and y coordinates in (a) and (b) are in the range of ±20[mm] except at time (Step 3), (Step 6), and (Step 8), which represents the time that the target's motion includes accelerations. And about distance estimation in z coordinate, Fig. 14 (c) shows that the error is in the

[DOI: 10.1299/jamdsm.2020jamdsm0057]

Table 2 The target pose value ${}^H\boldsymbol{\phi}_M = [{}^Hx_M, {}^Hy_M, {}^Hz_M, {}^H\varepsilon_{1M}, {}^H\varepsilon_{2M}, {}^H\varepsilon_{3M}]^T$ of each motion step is listed with names of (Step 0) to (Step 19), corresponding to the target's motion trajectory in Fig. 12. Similar to Fig. 12, the arrows in this table show the changing parameters from the previous step to the next. For example, in this table, since from (Step 0) to (Step 1) ${}^Hx_M$ is only changed, there is an arrow between row (Step 0) and (Step 1) in the column of ${}^Hx_M$. And the arrow of subfigure (Step 1) in Fig. 12 also shows that the target moves along the x-axis.

Pose / Step	Position[mm]			Orientation(quaternion[])		
	${}^Hx_M$	${}^Hy_M$	${}^Hz_M$	${}^H\varepsilon_{1M}$	${}^H\varepsilon_{2M}$	${}^H\varepsilon_{3M}$
(Step 0)	0	0	500	0	0	0
(Step 1)	-50	0	500	0	0	0
(Step 2)	-50	-50	500	0	0	0
(Step 3)	0	-50	500	0	0	0
(Step 4)	0	0	500	0	0	0
(Step 5)	0	0	550	0	0	0
(Step 6)	-50	0	550	0	0	0
(Step 7)	-50	-50	550	0	0	0
(Step 8)	0	-50	550	0	0	0
(Step 9)	0	0	550	0	0	0
(Step 10)	0	0	500	0	0	0
(Step 11)	0	0	500	0.174	0	0
(Step 12)	0	0	500	-0.174	0	0
(Step 13)	0	0	500	0	0	0
(Step 14)	0	0	500	0	0.174	0
(Step 15)	0	0	500	0	-0.174	0
(Step 16)	0	0	500	0	0	0
(Step 17)	0	0	500	0	0	-0.174
(Step 18)	0	0	500	0	0	0.174
(Step 19)	0	0	500	0	0	0

range of ±30[mm] roughly. Some large fluctuations in (a)~(c) show the time delay of position coordinate detection, e.g., (Step 1), (Step 6), and (Step 8) in (a). About the orientation estimation, the error of ε_3 in (f) is small and less than those of ε_1 in (d) and ε_2 in (e). In the period of (Step 1)~(Step 10), it can be confirmed that the change of position of the target object interferes with tracking errors of the orientation estimation. And in the period of (Step 11) to (Step 19), even though the orientation of the target object has been changed, the position detection errors in (a)~(c) are kept to be small.

Through the above analyses and discussions of experimental results, it has been confirmed that the proposed photo-model-based recognition method can detect an object's pose in real-time by using RM-GA.

7. Discussion

In section 5, the fitness distribution experiments were conducted to verify the feasibility of the proposed photo-model-based recognition method. That distributions of the fitness function in Figs. 10 and 11 have maximum peaks at the true pose of targets, have shown that the problem to detect the sea animal's pose has been confirmed to be converted to an optimization problem. And through the experimental results, it is confirmed that

(1)the proposed photo-model-based recognition method can estimate a 3D target object's pose by using stereo-vision and 2D photo-model;

(2)the fitness function Eq. (10) can transform the target recognition and orientation estimation problems into optimization problems.

And in the above section 6, the real-time pose tracking experiment was conducted to clarify that

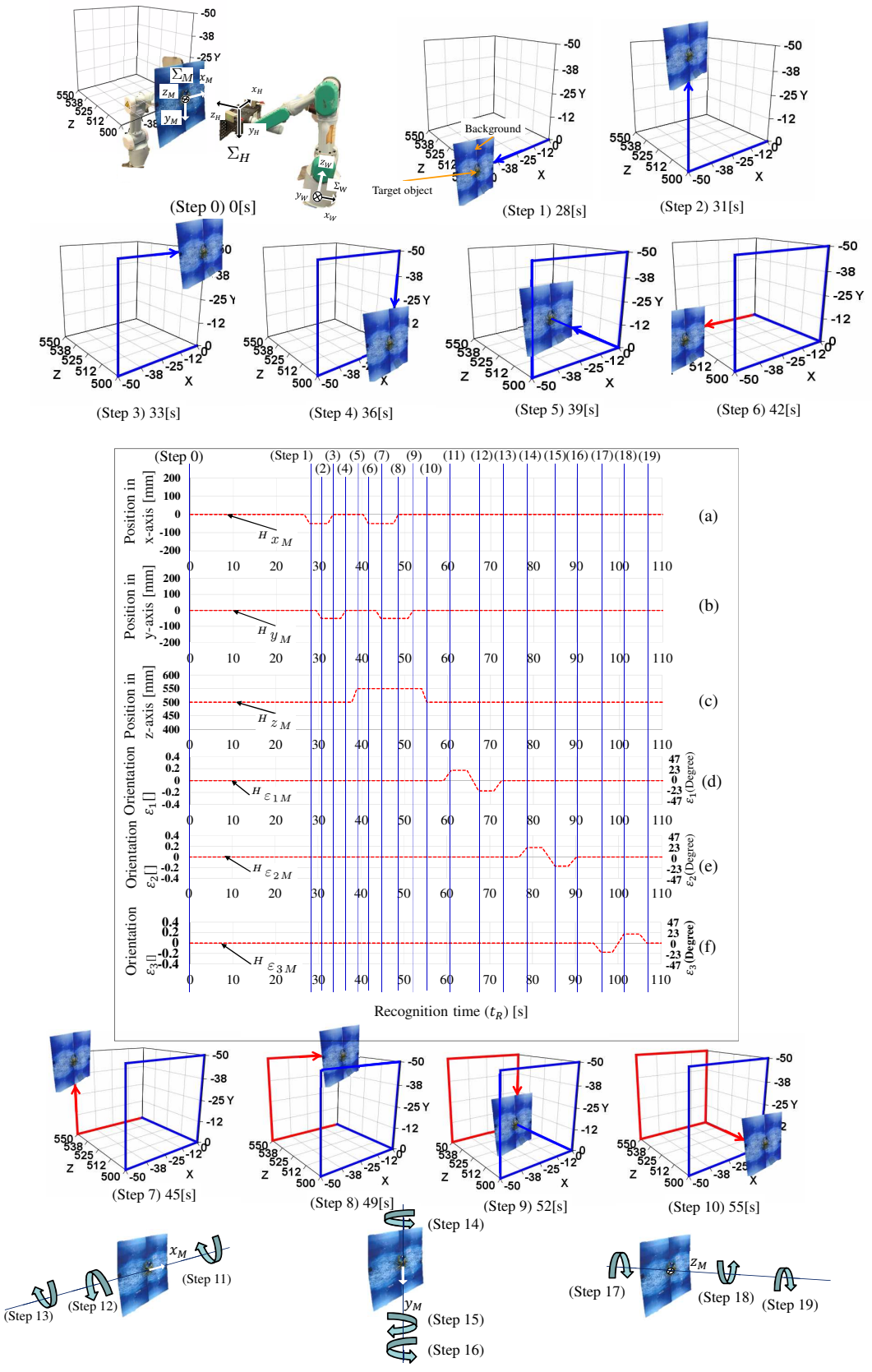
(3)this photo-model-based stereo-vision system can track a moving object's pose in real-time with RM-GA.

The above three points are the contributions of this paper introduced in section 2 and verified by the fitness distribution and real-time pose tracking experiments.

8. Conclusion

In this paper, the photo-model-based recognition method and real-time pose estimation method with photo-model are presented. According to the results of the fitness distribution and the real-time 3D pose estimation experiment, it is

[DOI: 10.1299/jamdsm.2020jamdsm0057]



(Step 11) 61[s] $\varepsilon_{1M} = 0.174$
(Step 12) 67[s] $\varepsilon_{1M} = -0.174$
(Step 13) 73[s] $\varepsilon_{1M} = 0$

(Step 14) 78[s] $\varepsilon_{2M} = 0.174$
(Step 15) 86[s] $\varepsilon_{2M} = -0.174$
(Step 16) 90[s] $\varepsilon_{2M} = 0$

(Step 17) 96[s] $\varepsilon_{3M} = -0.174$
(Step 18) 102[s] $\varepsilon_{3M} = 0.174$
(Step 19) 107[s] $\varepsilon_{3M} = 0$

Fig. 12 The target in this pose tracking experiment is C12 crab shown in Fig. 8 and Fig. 9. The crab's position time profile is shown by (a), (b), and (c) based on the end-effector Σ_H. Orientation motion is shown by (d), (e), and (f). The subfigures (Step 0)~(Step 19) shows the target motion schematically. The subfigures of (a)~(f) show all the poses time profile of the target Σ_M based on Σ_H. The arrows in (Step 0)~(Step 19) show the target's moving direction. Motion curves (a)~(f) are enlarged and shown as dashed lines in Fig. 13. The poses of Σ_M at (Step 10), (Step 13), (Step 16), and (Step 19) are the same with the initial state (Step 0).

[DOI: 10.1299/jamdsm.2020jamdsm0057]

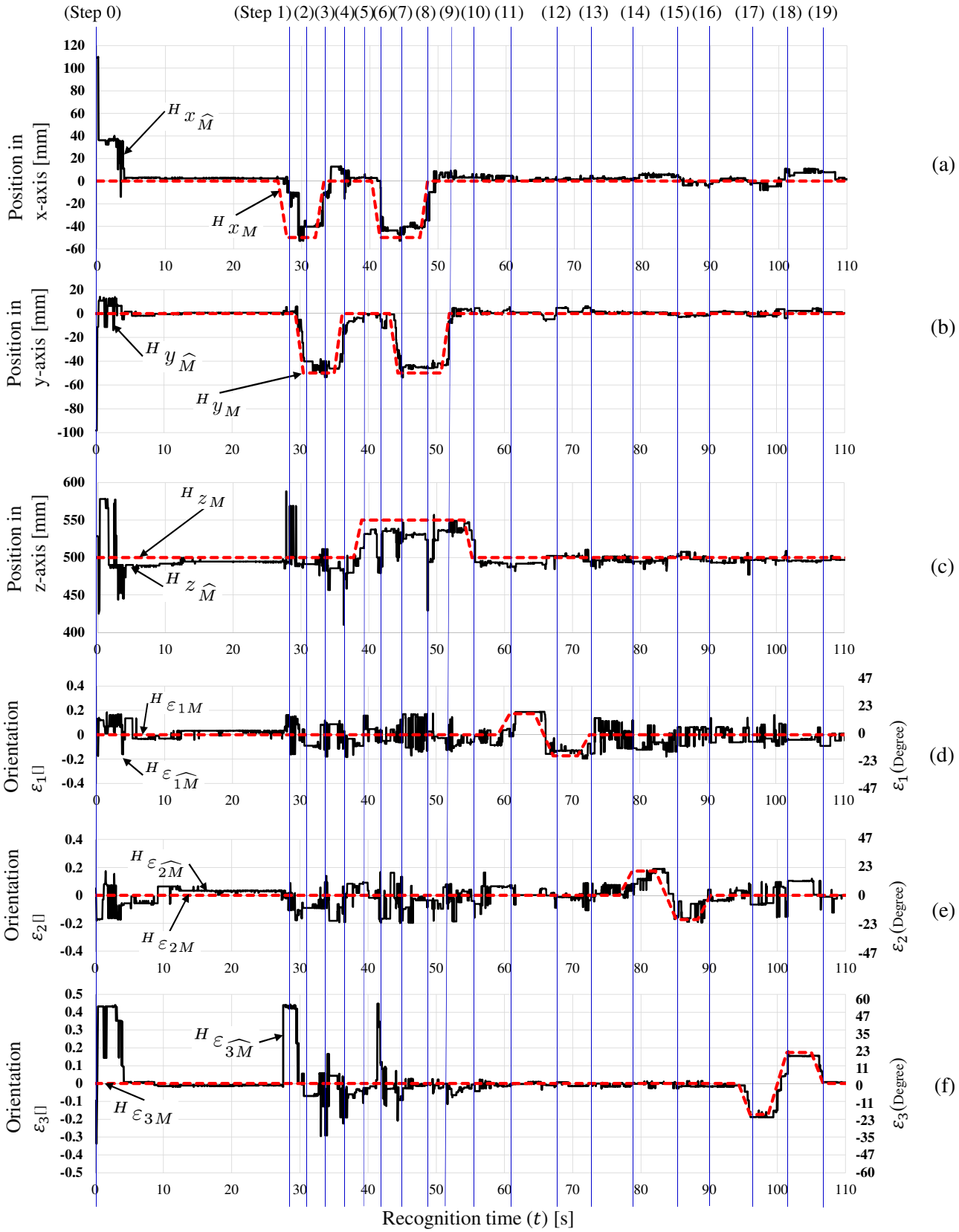


Fig. 13 The 3D pose estimation results that tracks the pose of the target whose motions are displayed in Fig. 12. The target is C12 crab shown in Fig. 8 and 9. The crab's position detection results are shown in above (a), (b), and (c) as solid lines. Orientation detection results are shown in (d), (e), and (f) as solid lines. The dashed lines are enlarged from Fig. 12 and show the true pose of the target object. (Step 0)~(Step 19) that are written at the top of this figure show the specific time points which are corresponding to the subfigures in Fig. 12. And from (Step 2) to (Step 19), "Step" has been eliminated to save space. The right side axes of (d)~(f) indicate angles that are calculated from quaternion to degree.

[DOI: 10.1299/jamdsm.2020jamdsm0057]

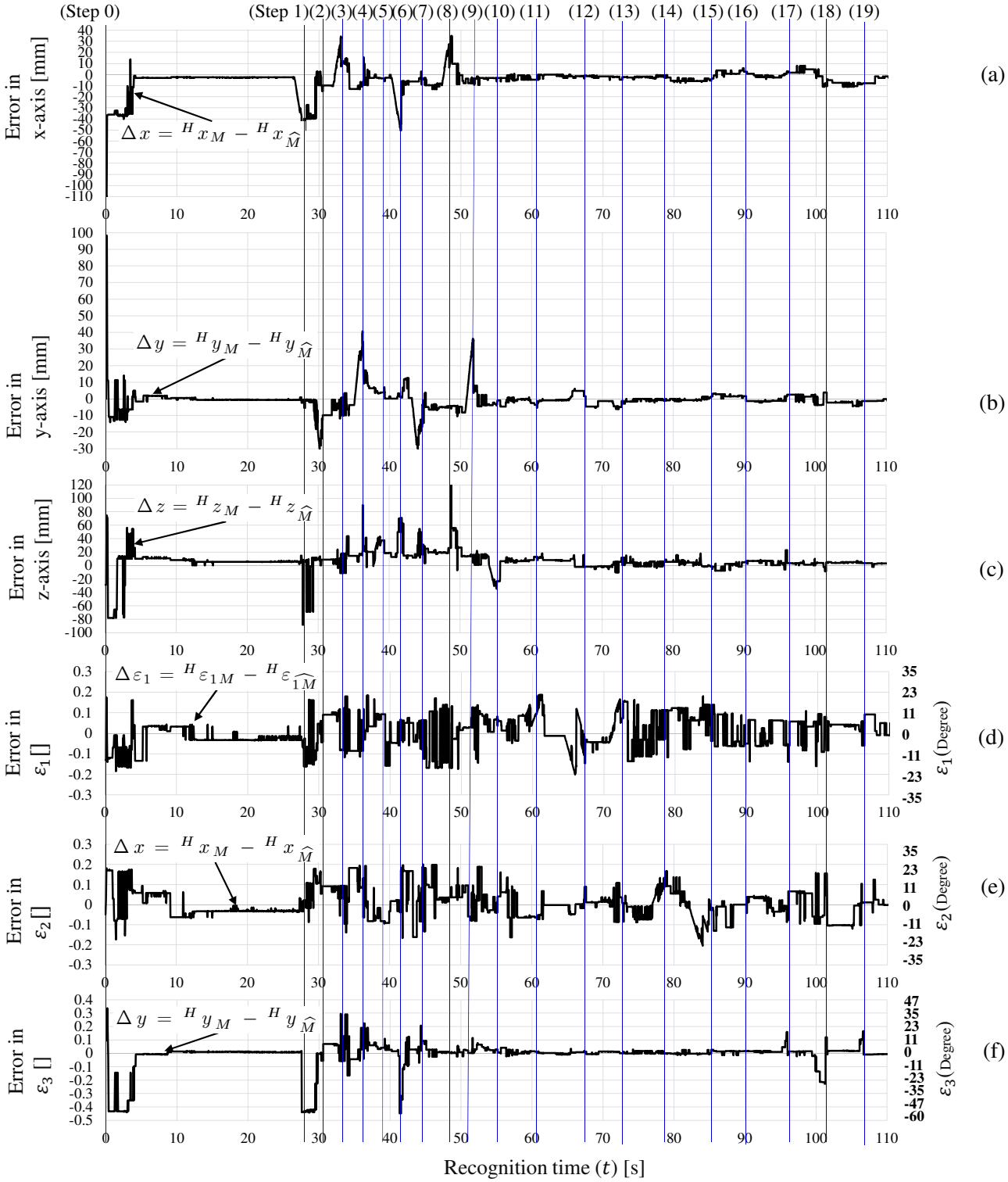


Fig. 14 The 3D pose estimation errors corresponding to Fig. 13. The crab's position detection errors are shown in (a), (b), and (c). Orientation detection errors are shown in (d), (e), and (f). (Step 0)~(Step 19) at the top of this figure show the specific time points which are corresponding to the subfigures in Fig. 12. And from (Step 2) to (Step 19), "Step" has been eliminated to save space.

[DOI: 10.1299/jamdsm.2020jamdsm0057]

confirmed that the proposed photo-model-based stereo-vision system can recognize the target object and detect the pose of sea animal target object with the prepared pictures. Then it has been confirmed that the proposed system can track the object and detect 3D pose in real-time with stereo-vision.

References

Agin, G.J., Real time control of a robot with a mobile camera (1979), SRI International. Artificial Intelligence Center.

Allen, P. K., Timcenko, A., Yoshimi, B. and Michelman, P., Automated tracking and grasping of a moving object with a robotic hand-eye system, IEEE Transactions on Robotics and Automation, Vol. 9, No.2 (1993), pp. 152-165.

Bateux, Q., Marchand, E., Leitner, J., Chaumette, F. and Corke, P., Visual Servoing from Deep Neural Networks, arXiv preprint (2017), arXiv:1705.08940.

Chaumette, F. and Hutchinson, S., Visual servo control. II, Advanced approaches (Tutorial), IEEE Robotics and Automation Magazine, Vol.14, No.1 (2007), pp. 109-118.

Dune, C., Marchand, E. and Leroux, C., One click focus with eye-in-hand/eye-to-hand cooperation, In Robotics and Automation, 2007 IEEE International Conference (2007), pp. 2471-2476.

Funakubo, R., Phyu, K.W., Tian, H. and Minami, M., Recognition and handling of clothes with different pattern by dual hand-eyes robotic system, IEEE/SICE International Symposium on System Integration (SII) (2016), pp. 742-747.

Funakubo, R., Phyu, K.W., Hagiwara, R., Tian, H. and Minami, M., Verification of illumination tolerance for clothes recognition, The Twenty-Second International Symposium on Artificial Life and Robotics (AROB) (2017), pp. 807-812.

Hutchinson, S., Hager, G.D. and Corke, P.I., A tutorial on visual servo control, IEEE transactions on robotics and automation Vol.12, No.5 (1996), pp. 651-670.

Kou, Y., Tian, H., Minami, M. and Matsuno, T., Improved eye-vergence visual servoing system in longitudinal direction with RM-GA, Artificial Life and Robotics, Vol.23, No.1 (2018), pp. 131-139.

Joshi, K.A. and Thakore, D.G., A Survey on Moving Object Detection and Tracking in Video Surveillance System, International Journal of Soft Computing and Engineering, Vol. 2, No. 3 (2012), pp. 44–48.

Leibe, B., Schindler, K., Cornelis, N. and Van Gool, L., Coupled Object Detection and Tracking From Static Cameras and Moving Vehicles, IEEE transactions on pattern analysis and machine intelligence, Vol. 30, No. 10 (2008), pp. 1683–1698.

Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A. and Hengel, A.V.D., A Survey of Appearance Models in Visual Object Tracking, ACM transactions on Intelligent Systems and Technology (TIST), Vol. 4, No. 4 (2013), pp. 1-48.

Malis, E., Chaumette, F. and Boudet, S., 2 1/2 D visual servoing, IEEE Transactions on Robotics and Automation, Vol.15, No.2 (1999), pp. 238-250.

Matsuyama, T., Kuno, Y. and Imiya, A., Computer vision: technical review and future view, New Technology Communications (in Japanese) (1998).

Minami, M., Suzuki, H., Agbanhan, J. and Asakura, T., Visual servoing to fish and catching using global/local GA search, IEEE/ASME International Conference on Advanced Intelligent Mechatronics. Proceedings (Cat. No. 01TH8556) Vol. 1 (2001), pp. 183-188.

Minami, M., Agbanhan, J. and Asakura, T., Evolutionary scene recognition and simultaneous position/orientation detection. In Soft Computing in Measurement and Information Acquisition, Springer Berlin Heidelberg, (2003), pp. 178-207.

Marchand, É. and Chaumette, F., Feature tracking for visual servoing purposes. Robotics and Autonomous Systems, Vol.52, No.1 (2005), pp.53-70.

Morrison, D., Tow, A.W., McTaggart, M., Smith, R., Kelly-Boxall, N., Wade-McCue, S., Erskine, J., Grinover, R., Gurman, A., Hunn, T. and Lee, D., Cartman: The low-cost cartesian manipulator that won the amazon robotics challenge, (2017), arXiv preprint arXiv:1709.06283.

Myint, M., Yonemori, K., Yanou, A., Lwin, K.N., Minami, M. and Ishiyama, S., 2016. Visual Servoing for Underwater Vehicle Using Dual-Eyes Evolutionary Real-Time Pose Tracking. JRM, Vol.28, No.4 (2016), pp.543-558.

Myint, M., Yonemori, K., Lwin, K.N., Yanou, A. and Minami, M., Dual-eyes vision-based docking system for autonomous underwater vehicle: an approach and experiments. Journal of Intelligent & Robotic Systems (2017), pp.1-28.

Newcombe, R.A., Lovegrove, S.J. and Davison, A.J., DTAM: Dense tracking and mapping in real-time, in 2011 interna-

[DOI: 10.1299/jamdsm.2020jamdsm0057]

tional conference on computer vision, IEEE (2011), pp. 2320–2327.
Oh, P.Y. and Allen, K., Visual servoing by partitioning degrees of freedom. IEEE Transactions on Robotics and Automation, Vol.17, No. 1, (2001), pp.1-17.
Pan, S., Shi, L. and Guo, S., A Kinect-Based Real-Time Compressive Tracking Prototype System for Amphibious Spherical Robots, Sensors, Vol. 15, No. 4 (2015), pp. 8232–8252.
Phyu, K.W., Cui, Y., Tian, H., Hagiwara, R., Funakubo, R., Yanou, A. and Minami, M., Accuracy on photo-model-based clothes recognition. In SICE Annual Conference, Tsukuba, Japan, (2016), pp. 20-23.
Phyu, K.W., Funakubo, R., Fumiya, I., Shinichiro, Y. and Minami, M., Verification of recognition performance of cloth handling robot with photo-model-based matching. In IEEE International Conference on Mechatronics and Automation (ICMA) (2017), pp. 1750-1756.
Phyu, K.W., Funakubo, R., Hagiwara, R., Tian, H. and Minami, M., Verification of illumination tolerance for photo-model-based cloth recognition. Artificial Life and Robotics, Vol.23, No.1 (2018a), pp.118-130.
Phyu, K.W., Funakubo, R., Hagiwara, R., Tian, H. and Minami, M., Verification of photo-model-based pose estimation and handling of unique clothes under illumination varieties, Journal of Advanced Mechanical Design, Systems, and Manufacturing, Vol.12, No.2 (2018b), DOI: 10.1299/jamdsm.2018jamdsm0047.
Phyu, K. W., Funakubo, R., Ikegawa, F., and Minami, M., Verification of unique cloth handling performance based on 3D recognition accuracy of cloth by dual-eyes cameras with photo-model-based matching, International Journal of Mechatronics and Automation, Vol.6, No.2-3 (2018c), 55-62.
Song, W., Minami, M., Mae, Y. and Aoyagi, S., On-line evolutionary head pose measurement by feedforward stereo model matching, IEEE International Conference on Robotics and Automation (2007), pp. 4394-4400.
Song, W. and Minami, M., Position-based Visual Servoing to 3D Pose with Feedforward Compensation, 2nd International Symposium on Test Automation and Instrumentation (2008), pp. 702-705.
Song, W., Minami, M. and Aoyagi, S., On-line stable evolutionary recognition based on unit quaternion representation by motion-feedforward compensation. International Journal of Intelligent Computing in Medical Sciences & Image Processing, Vol.2, No.2 (2008), pp. 127-139.
Song, W., Yu, F. and Minami, M., 3D visual servoing by feedforward evolutionary recognition. Journal of Advanced Mechanical Design, Systems, and Manufacturing, Vol.4, No.4 (2010), pp.739-755.
Schwarz, M., Lenz, C., Garc*i*a, G.M., Koo, S., Periyasamy, A.S., Schreiber, M. and Behnke, S., Fast Object Learning and Dual-arm Coordination for Cluttered Stowing, Picking, and Packing. In 2018 IEEE International Conference on Robotics and Automation (ICRA) (2018), pp. 3347-3354.
Tamadazte, B., Marchand, E., Dembélé, S. and Le Fort-Piat, N., Cad model-based tracking and 3d visual-based control for mems microassembly. The International Journal of Robotics Research, Vol.29, No.1 (2010), pp.1416-1434.
Tamadazte, B., Duceux, G., Piat, N.L.F. and Marchand, E., Highly Precise Micropositioning Task Using a Direct Visual Servoing Scheme, in 2011 IEEE International Conference on Robotics and Automation, IEEE (2011), pp. 5689–5694.
Tian, H., Cui, Y., Minami, M. and Yanou, A., Frequency response experiments of eye-vergence visual servoing in lateral motion with 3D evolutionary pose tracking. Artificial Life and Robotics, Vol.22, No.1 (2017a), pp.36-43.
Tian, H., Kou, Y., Phyu, K.W., Yamada, D. and Minami, M., Robust translational/rotational eye-vergence visual servoing under illumination varieties, In IEEE International Conference on Robotics and Biomimetics (ROBIO) (2017b), pp. 2032-2037.
Tian, H., Kou, Y., and Minami, M., Visual servoing to arbitrary target with photo-model-based recognition method, 24th International Symposium on Artificial Life and Robotics (AROB) (2019), pp.950-955.
Tsai, C.Y., Huang, C.C. and Chou, Y.S., Data-Driven Visual Picking Control of a 6-DoF Manipulator Using End-to-End Imitation Learning, in 2018 International Automatic Control Conference (CACS), IEEE (2018), pp. 1-6.
Yilmaz, A., Javed, O. and Shah, M., Object Tracking: A Survey, Acm computing surveys (CSUR), Vol. 38, No. 4 (2006), DOI = 10.1145/1177352.1177355.
Zeng, A., Song, S., Yu, K.T., Donlon, E., Hogan, F.R., Bauza, M., Ma, D., Taylor, O., Liu, M., Romo, E. and Fazeli, N., Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching, (2017), arXiv preprint arXiv:1710.01330.

[DOI: 10.1299/jamdsm.2020jamdsm0057]