Journal of Intelligent & Robotic Systems (2019) 96:245–266
https://doi.org/10.1007/s10846-018-0970-x

Sea Docking by Dual-eye Pose Estimation with Optimized Genetic Algorithm Parameters

Khin Nwe Lwin[1] · **Myo Myint**[1] · **Naoki Mukada**[1] · **Daiki Yamada**[1] · **Takayuki Matsuno**[1] · **Kazuhiro Saitou**[1] · **Waichiro Godou**[1] · **Tatsuya Sakamoto**[1] · **Mamoru Minami**[1]

Received: 1 December 2017 / Accepted: 4 December 2018 / Published online: 3 January 2019

Abstract

Three-dimensional (3D) estimation of position and orientation (pose) using dynamic (successive) images input at video rates needs to be performed rapidly when the estimated pose is used for real-time feedback control. Single-camera 3D pose estimation has been studied thoroughly, but the estimated position accuracy in the camera depth of field has proven insufficient. Thus, docking systems for underwater vehicles with single-eye cameras have not reached practical application. The authors have proposed a new 3D pose estimation method with dual cameras that exploits the parallactic nature of stereoscopic vision to enable reliable 3D pose estimation in real time. We call this method the “real-time multi-step genetic algorithm (RM-GA).” However, optimization of the pose tracking performance has been left unchallenged despite the fact that improved tracking performance in the time domain would help improve performance and stability of the closed-loop feedback system, such as visual servoing of an underwater vehicle. This study focused on improving the dynamic performance of dual-eye real-time pose tracking by tuning RM-GA parameters and confirming optimization of the dynamical performance to estimate a target marker’s pose in real time. Then, the effectiveness and practicality of the real-time 3D pose estimation system was confirmed by conducting a sea docking experiment using the optimum RM-GA parameters in an actual marine environment with turbidity.

Keywords Real-time multi-step GA · Visual servoing · Pose estimation · Dual-eye tracking · Underwater docking

1 Introduction

Controlling robots with visual information has been studied mainly for ground-based robots. For example, target object detection and recognition for mobile robots using a monocular camera were proposed in [1] and [2]. In [1], object detection and recognition was performed by an indoor mobile robot using template matching with scale-invariant feature transform (SIFT). In that approach, a robot was localized relative to the floor using floor marks. The objects in images input from a single camera were extracted and recognized, and a three-dimensional (3D) solid model of the indoor environment was mapped onto the floor. In [2], 3D models were created by finding the edge points in captured images of the actual environment. In [3], an image acquisition method was used to reconstruct the 3D shape of objects for an underwater vehicle. In that approach, the SIFT method and random sample consensus algorithm were used to extract and match some features from a pair of cameras to measure the pose of 3D objects.

Several studies have looked at visual-servoing-based underwater vehicles in recent years. Most of them have used a single camera to estimate the position and orientation (pose) of the target object [4–6]. Binocular vision was used in some studies to detect the target position of the vehicle [7, 8]. Even though two cameras were used in [7], one was facing downward for shooting sea-floor images and the second camera was pointed forward for the purpose of obstacle avoidance. In [8], the vehicle’s position was estimated using two cameras and a sonar system. In that approach, the position of the vehicle was calculated by combining data from a Doppler sonar and CCD cameras, but the system was not used to control the orientation of the vehicle. This means that 3D pose estimation using the parallactic character of dual cameras has not been realized.

✉ Khin Nwe Lwin
pdoj8yez@s.okayama-u.ac.jp

[1] Graduate School of Natural Science and Technology, Okayama University, Okayama, Japan

Realizing autonomous docking of underwater vehicles has been a hot topic in the field of marine science and technology [9, 10]. Conventional methodologies used so far are multi-sensor feedback to control the vehicle pose and follow a desired trajectory based on the estimated vehicle pose, but this conventional method has a shortcoming in that it is difficult to estimate orientation correctly, especially if a single camera and known marker are used for the pose estimation.

Real-time pose estimation has been identified as a vital technology for underwater vehicle control and docking. Since recharging batteries at the bottom of the sea enables a vehicle to conduct tasks for a long time, such as 1 year or longer, and since the recharging function requires the vehicle to dock to a charging station, docking technology has become more important in recent years. In this context, real-time pose estimation that is effective for docking control has become attractive. Based on this motivation, we developed a 3D-perception-based move-on-sensing (3D-MoS) system using dual cameras and a 3D marker for real-time pose tracking by means of visual servoing, as shown in Fig. 1. Thus, visual servoing of an underwater vehicle using the parallactic character of stereoscopic vision and 3D-model-based recognition and real-time multi-step genetic algorithm (RM-GA) was initiated by our research group [11].

When researchers want to improve the dynamical behavior of robots that are controlled by feedback information — closed-loop dynamical performance — they need to carefully examine whether the feedback signal includes a time delay. In devices controlled by sensor signals, the time delay of the signals seldom becomes problematic. However, when the estimated pose is used in the feedback loop of the vehicle, improvement of real-time pose estimation performance becomes much more important, and improvement of real-time pose estimation is required for stable pose control of the vehicle,.

In other optimization-based systems, optimization methods, especially GA, were used for tuning the parameters of the controllers [12, 13]. In our approach, optimization was used in the feedback loop. Moreover, most of the studies on GA-optimization-based systems based their evaluation on iteration number rather than the time domain [20–23] (Detailed discussion is given in Section 2.) Therefore, in comparison to the previous studies discussed above, (1) improvement of the 3D-MoS system by optimizing RM-GA for real-time pose estimation is one of the main contributions of this paper. In addition, our system performed 3D pose estimation using dynamic images with video rate of 30 frames per second (fps). Therefore, (2) verification against dynamic image input highlighted the merits of the proposed approach in comparison to studies [20–23] based on evaluation of iteration number. Finally, other vision-based docking approaches [14, 15] have not discussed performance in turbid conditions, which demonstrates dynamic fitness distribution. Thus, (3) the confirmation of the improved system by real sea docking experiments conducted in a turbid sea environment to examine RM-GA's optimizing behavior is the final contribution of this paper. A detailed discussion of the problem statement is provided in Section 3. According to recognition experiments in a test pool, it was confirmed that optimization improved the time response performance to track a moving 3D marker relative to the vehicle — even though the marker was stationary in space, the video images inputted successively from a camera included fluctuation — improving the stability of the vehicle when controlled by visual servoing. Sea docking experimental results confirmed the effectiveness and the practicality of our proposed 3D-MoS with optimized RM-GA.

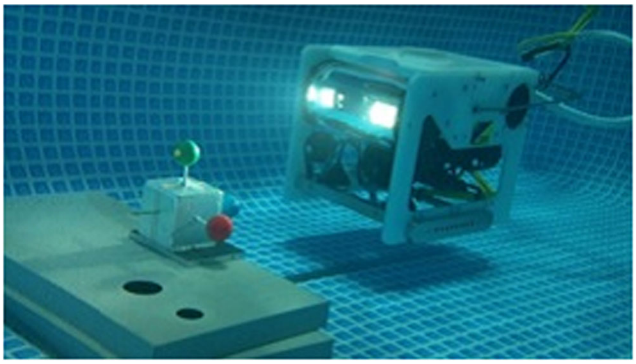


Fig. 1 Underwater vehicle and 3D marker

The remainder of this paper is organized as follows: Section 2 describes the optimization techniques in related work. Section 3 explains the problem. Section 4 describes the method for pose estimation. Experiments for optimizing real-time multi-step GA parameters are discussed in Section 5. A sea docking experiment using the optimum parameters of RM-GA is analyzed and discussed in Section 6. Conclusions and areas for future research are presented in Section 7.

2 Optimization Techniques in Related Work

Since the real-time pose tracking that is used for docking feedback control has been converted into an optimization problem of a time-varying multivariate function to estimate the pose of a remotely operated vehicle (ROV), some aspects of the optimization problem are discussed in this section.

Traditional optimization techniques, such as steepest decent, linear programming, and dynamic programming, have been used for solving multivariate functions to give the lowest function value, and they tend to require the assumption that the slope angle of the inclination of the optimization function is calculable. This assumption sometimes is an obstacle when these techniques are applied to general problems where the inclination angle is not calculable. In such cases, the inclination is calculated as an approximation, leading to optimization errors. Moreover, the traditional optimization techniques often fail to provide local optima for optimization problems with multiple peaks.

Powerful optimization techniques were developed to overcome this problem, including GA, ant colony optimization (ACO), and particle swarm optimization (PSO). These new methods do not need a calculable inclination of the optimizing function distribution. The GA is a global optimization technique for searching solutions in a large space. Its evaluation function works on the principle of the Darwinian theory of the survival of the fittest. The ACO approach is a multi-agent system in which the behavior of artificial ants is inspired by the behavior of real ants seeking food. The PSO algorithm is also a global optimization algorithm that works on the principle of the foraging behavior of swarms of birds. These advanced optimization techniques are utilized in engineering optimization problems and have proved effective for specific kinds of problems. A detailed discussion of these advanced algorithms is provided in [16].

All the above-mentioned algorithms are nature-inspired, population-based search techniques, and they have some limitations. Therefore, the suitability of these algorithms for specific problems needs to be tested, and they have been modified for different problems [17–19]. In [17], the optimization ability of the GA was enhanced and improved using sampling techniques, such as Latin hypercube sampling, Faure sequence sampling, and Hammersley sequence sampling. The performance and effectiveness of modified GA and the traditional algorithm were described and illustrated with a number of test problems. In [18], the authors introduced a new hybrid co-evolution-based multi-objective PSO method. The method in [18] is a hybrid technique that combines the features of co-evolution and Nash equilibrium with an ϵ-disturbance technique to handle multiple-objective optimization problems for ship design. The algorithm in [18] was tested and validated utilizing the most widely used 2D and 3D test functions and applied to an actual ship design. In [19], a modified ACO algorithm for localization of odor sources by multiple robots was proposed. The performance of the system in [19] was improved by adding local traversal search and global random search, and it effectiveness was shown by simulations.

3 Problem Statement

Researchers have discussed optimization of GA parameters for specific problems [20–23]. Rexhepi et al. [20] applied GA with different parameter settings to the travelling salesman problem (TSP) for Kosovo municipalities. In that approach, they studied route solution for the TSP by analyzing GA parameters based on the number of generations. Boyabatli and Sabuncuoglu [21] analyzed the effect of numerical parameters on GA performance and reported that high mutation rates give better performance. Tabassum and Mathew [22] utilized GA for image optimization and demonstrated the capability of solving the knapsack problem. Specifically, they studied how GA parameters affected reproduction of the original images. In [23], the influence of one key GA parameter, chromosome population, was studied for bacteria cultivation. Various population sizes from 5 to 200 chromosomes were explored with a fixed number of 200 generations. The results showed that the larger populations did not improve the solution accuracy while consuming more computing resources.

These studies have been based on the number of iterations to achieve a stable fitness distribution, not on the length of time needed for the optimization. Relative pose estimation by an ROV using dynamic images should be optimized against a time-dependent multi-peak fitness distribution as fast as possible for closed-loop stability. Since the speed of the estimation is related to the time needed for the optimization calculation, not the iteration number, the performance should be evaluated using the convergence response measured in the time domain. However, most optimization methodologies have focused on accuracy and iteration number rather than calculation time, i.e. best time response in the pose estimation against successive video image input, aiming at decreasing time delay of visual feedback to enhance stability of closed loop dynamical motion.

To the best of the authors' knowledge, no existing study has analyzed GA parameters based on real-time performance, especially for pose estimation. In this paper, the relation optimization performance for pose estimation is based on a criterion that the parameters used in RM-GA should achieve the shortest convergence time. The difference between our previous study [24] and this study is that this study provides more detail concerning the choice of optimum parameters based on convergence behavior in the algorithm. In [24], the practicality of the proposed system was presented by showing the experimental results of docking in an indoor pool. The present paper is intended to explain extensively how the proposed system performs real-time pose estimation using optimized GA parameters and how this optimization improves performance. Moreover, this study included the performance of docking in an actual sea environment with turbidity using the optimized RM-GA system; this had not been done previously [24]. The experiments confirmed that the proposed RM-GA system can perform real-time recognition and feedback control.

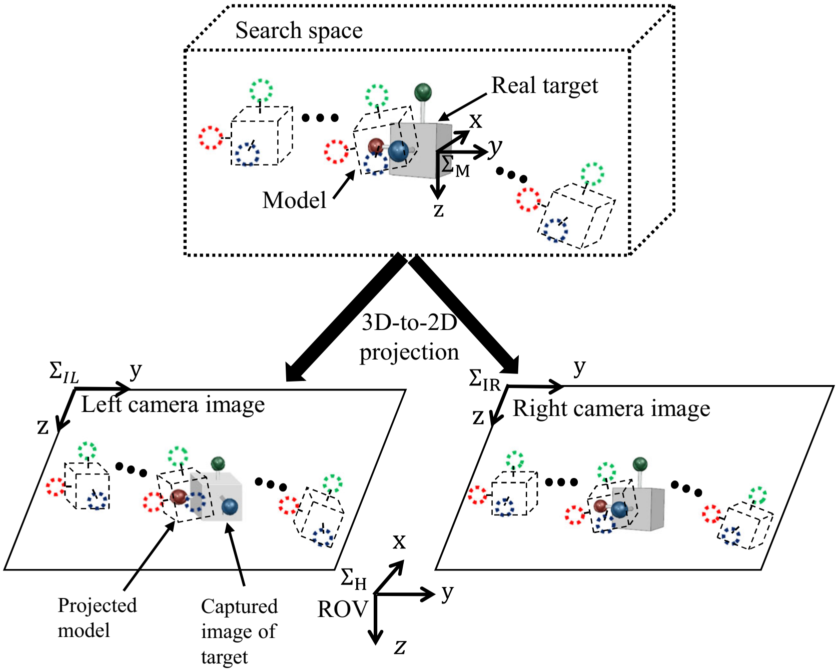


Fig. 2 3D-model-based matching method using dual-eye vision. A fitness function evaluates the degree of matching between the projected 2D model and the real 3D marker as captured in both camera images

4 Pose Estimation Method

Instead of localizing the vehicle and target pose using global coordinates, such as by GPS tracking, our system localizes the vehicle relative to the target through recognition of a known target using dual cameras set on the ROV and a 3D marker that represents the pose of the docking station.

4.1 3D Model-based Matching Method Using Dual-eye Vision System

In this section, we discuss the 3D pose estimation method briefly for the reader's convenience. Please refer to [11] for detailed explanation.

Template matching using 2D models and 2D images has been used in other model-based approaches. This technique cannot reliably provide estimation for 3D poses, particularly for orientation. In contrast, the proposed pose estimation method is based on the idea of recognition using a 3D model to evaluate 2D images with stereo vision. A previous study [25] developed a 3D-model-based matching method that can estimate the relative 3D pose of a human head. This strategy has been validated as realistic and practical for 3D recognition by an underwater ROV.

Based on the known shape, color, and size of the 3D marker, models with assumed poses (since pose is the unknown parameter in this study) are predefined and distributed in the 3D search space in front of the cameras, as shown in Figs. 2 and 3. Then, each model is projected onto the two camera images. A fitness function evaluates the match between the projected model with assumed pose and the real 3D marker, as captured by the cameras. When the projected model and real 3D marker coincide in the 2D images and maximize the fitness value, then the pose of the model that gives the maximum fitness value is the output representing the actual pose of the real 3D marker. Therefore, the main task in the pose estimation method is to search for the model with the highest fitness value. However, it is time-consuming to evaluate all possible models in the search space in a real-time application. Therefore, our approach converts the search problem into an optimization problem. Based on this concept, we implemented a RM-GA in which a number of models with random poses are initially generated and those models are converged to the real target for one image input

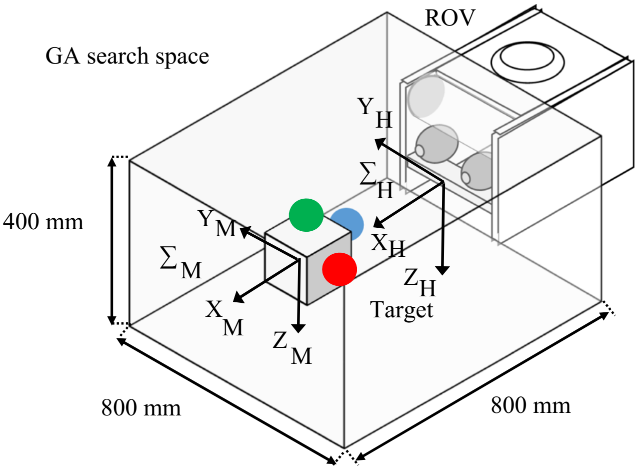


Fig. 3 GA search space

during a 33-ms control period (note that camera video rate is 30 fps). Every 33 ms, the RM-GA outputs the best-fit pose and inputs the next successive image for next calculation of pose. Selection of RM-GA parameters and verification of the proposed system for real-time pose tracking are the main contributions of this paper. The framework and the system architecture of RM-GA are discussed in Section 4.3.1.

The 3D-model-based pose estimation using dual-eye vision is shown in Fig. 2. In the figure, $\sum_{IL}$ and $\sum_{IR}$ are the reference frames of the left and right camera images, respectively; $\sum_{H}$ is the reference frame of the ROV; and $\sum_{M}$ is the reference frame of the real target. The search space of the vision system is defined as shown in Fig. 3. A model of the 3D marker with three balls colored red, green, and blue is predefined in the computer system as the target object. Many 3D models with the same 3D information, including shape, color, and size, but different poses, are allocated randomly in the search area. The real target object in space is captured by a dual-eye camera, and the poses of each model, which is given by a chromosome created by the RM-GA and shown as dotted lines in Fig. 2, are projected onto the 2D images. The degree of matching between the projected solid model and dotted line models is evaluated in 3D space by applying a fitness function to the left and right 2D images. Finally, the best model of the target object that represents the true pose is obtained based on the highest fitness value.

The 3D marker balls are used because they have a perceptible diameter. As the ROV approaches a ball, the circle in the 2D projection becomes proportionately larger. Other shapes revealed the difficulty of estimating the position in the camera depth of field (x) direction. The 3D marker has no characteristic symmetry, such as point symmetry, linear symmetry, plane symmetry, or rotational symmetry. Thus, the 3D marker cannot assume spatial orientations that would restrict the RM-GA. In other words, the marker always presents the ROV with a unique image that cannot also be obtained from another viewing angle.

4.2 Fitness Function

The fitness function is used to measure the degree of matching between the captured image and the projected models. In other words, the fitness function correlates the 3D pose of each individual model with the real 3D target in 3D space. A good fitness function enhances the ability of the GA to explore the search space with a more effective and efficient convergence speed. In this system, hue value is used for recognition of the 3D marker because it is less sensitive to environmental conditions. Figure 4 shows the real 3D marker and a model of the target 3D marker. The 3D marker was constructed using red, green, and blue balls (diameter of 40 mm). The dimensions of the real marker are shown in Fig. 4a, the model of each sphere is shown in Fig. 4b, and c is an enlarged view of one ball model. Each ball model consists of two portions, an inner portion that is the same size as the target and an outer portion that is the background area as shown in Fig. 4. The captured image (pixel) is detected in a 2D image as green, blue, or red in hue space.

The following fitness function is used in this study. Each ball of the model comprises an inner sphere $S_{L,in}$ and an enveloping sphere $S_{L,out}$, as shown in (1). The inner sphere of the left camera model $S_{L,in}$ is intended to evaluate the ball area of the real target — corresponding to inner area in Fig. 4c —, and the enveloping sphere $S_{L,out}$ — corresponding to outer area in Fig. 4c — is for the background area. The fitness function for the model $F(\phi_i)$ with assumed pose ϕ_i is calculated by averaging the fitness functions of both the left camera image $F_L(\phi_i)$ and right camera image $F_R(\phi_i)$. The summation in (1) is concerning

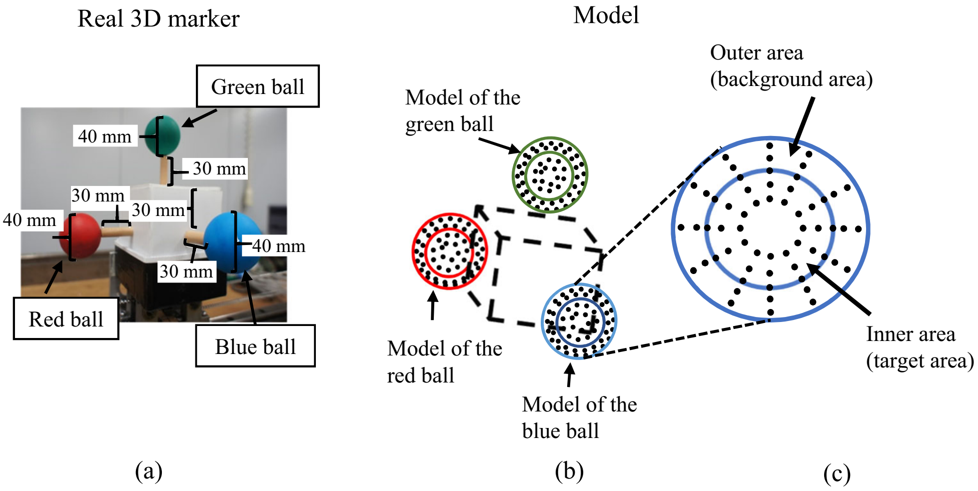


Fig. 4 Real 3D marker and model: (**a**) real 3D marker, (**b**) model, and (**c**) enlarged view of the blue ball model, where the inner area is the same size as the real target object and the outer area is the background area. The dots in (**c**) are points for calculating the degree of overlap between the actual ball and the inner and outer areas of the model

the j-th point ${}^{IL}r_j(\phi_i)$ in left camera image, defined on the i-th 3D model, whose pose is ϕ_i. In (1), N represents the number of points to be evaluated, where j = 1, 2, $\cdots$, N. The allocations of those points are shown in Fig. 4c. If the j-th point of i-th model defined by ϕ_i in $S_{L,in}$ overlaps with real 3D marker and the model's hue color value coincides with the projected real 3D marker's hue value, in this case the fitness function has a value of $p({}^{IL}r_j(\phi_i)) = +1$ in (1) then the fitness value increases. If otherwise, $p({}^{IL}r_j(\phi_i)) = -1$, then the fitness value decreases. Therefore, the fitness value will be maximum when the model and the real target exactly coincide, and the system will have obtained the true pose of the real target 3D marker that gives maximum point in the fitness function distribution. Although RM-GA may not track the true pose in realtime, the pose given by the gene with highest fitness value can exist around the true pose within some pose estimation error ranges. The estimated pose is represented by $\hat{\phi}$ hereafter.

$$
\begin{aligned}
F(\phi_i) = &\frac{1}{N}\{F_L(\phi_i) + F_R(\phi_i\}/2 \\
= \frac{1}{N}&\left\{\left(\sum_{{}^{IL}r_j(\phi_i)\in S_{L,in}(\phi_i)} p({}^{IL}r_j(\phi_i)) + \sum_{{}^{IL}r_j(\phi_i)\in S_{L,out}(\phi_i)} p({}^{IL}r_j(\phi_i))\right)\right. \\
&\left.+\left(\sum_{{}^{IR}r_j(\phi_i)\in S_{R,in}(\phi_i)} p({}^{IR}r_j(\phi_i)) + \sum_{{}^{IR}r_j(\phi_i)\in S_{R,out}(\phi_i)} p({}^{IR}r_j(\phi_i))\right)\right\}/2 \quad (1)
\end{aligned}
$$

The specification of RM-GA is that the number of genes is 60, selection rate 60%, mutation rate 10%, crossover is two-point, evolutionary strategy is elitism preservation. A more detailed description of the fitness function can be found in our previous study [11, 26].

4.3 Evolution in RM-GA

As explained in Section 4.1, the search problem is converted into an optimization problem. To the best of the authors' knowledge, few studies have applied advanced optimization techniques when the real-time performance is dominant. To overcome the computational burden, especially for systems with fast dynamics, fast and efficient algorithms based on a real-time iteration scheme were discussed in [27]. In [28], an ACO algorithm to optimize the parameters of the controller for a real-time process for a highly nonlinear conical tank system was presented. In [29], an identification-based robust motion control was implemented for an autonomous underwater vehicle using a PSO algorithm. Although simulation results were presented to demonstrate the performance of the system, the methodology in [29] has not been used in a real application. Instead of using other methods that could provide powerful accuracy but also increase the computational burden, RM-GA (Note that the former name of RM-GA is 1-step GA) was selected for the proposed system because it has following advantages:

(1) Adaptable to stationary environments that provide a differential distribution with multiple peaks.
(2) Repeatable within the video frame rate to deal with time-varying distribution for newly input images.
(3) Recognition accuracy in terms of

a. Sensitivity: each of six parameters of the pose are coded by 12-bit strings and recognition accuracy is at the millimeter level,
b. Reliability: design of fitness function and selected GA parameters can provide robustness against disturbance, and
c. Convergence speed: the evolutionary algorithm in RM-GA evolves the chromosomes using as many generations as possible within the video frame duration of each image.

A more detailed explanation of why RM-GA was used and how it works is provided in our previous work [11].

4.3.1 Framework and System Architecture of RM-GA

The pose of the model estimated by RM-GA in 4.2, $\hat{\phi}$, is expressed as a 72-bit string to represent six pose parameters (x, y, z, ϵ_1, ϵ_2, and ϵ_3), as shown in Fig. 5. The first 36 bits (12 bits each for x, y, and z) represent the position coordinates of the 3D model. The last 36 bits (12 bits each for ϵ_1, ϵ_2, and ϵ_3) describe the orientation defined by a quaternion. Supposing the desired pose is given as ϕ_d, the visual servoing error can be defined as $e = \phi_d - \hat{\phi}$. The controller used for the visual servoing experiment in this paper is a simple proportional controller, since damping effect seems to be enough through the preparatory experiments.

A flowchart for the RM-GA is shown in Fig. 6b, and a shows the GA process steps for converging on the best solution from the first to the final generation. Note that evaluation is performed in 2D and convergence occurs in 3D. The search space of the GA is defined as shown in Fig. 3. First, a random population of models with different poses are generated within the search space, and a new pair of left and right images captured by the dual cameras is input. The appearance of the real target and the first generation of models in 3D search space and the corresponding 2D images of the target and models is shown at the top of Fig. 6a. Each model in the population is evaluated using the fitness function. After the results are ranked, the selection operator selects the models from the current generation that have the best fitness values for reproduction in the next generation. A two-point crossover operator creates the next generation of models by randomly

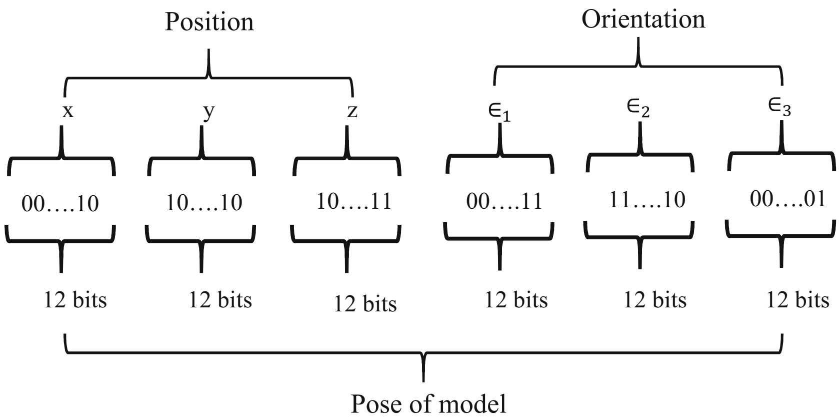


Fig. 5 GA chromosome for model pose: 12 bits each for x, y, and z represent the position coordinates of the 3D model and 12 bits each for ϵ_1, ϵ_2, $and\epsilon_3$ describe the orientation defined by a quaternion

exchanging data between chromosomes at two positions. In spite of the above reproducing processes, the best gene would be preserved as it is into the next generation, called elitist preservation strategy. This is intended that the best pose may not be changed in 33 ms. Then, the mutation operator randomly changes additional bits in the crossover progeny. At this time, the next generation has a better fitness value for position and orientation that more closely matches the real target's pose than did the previous generation. The genetic information is transferred to the next generation

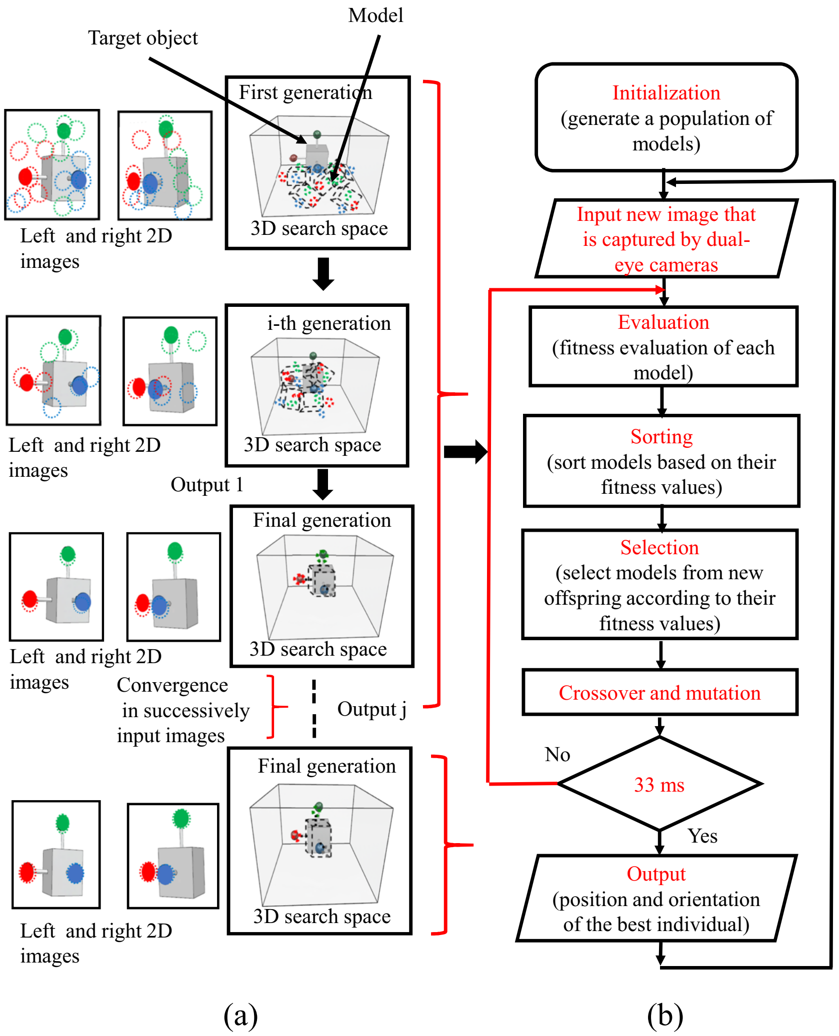


Fig. 6 Process flow in the RM-GA: (**a**) graphical representation of solution evaluation and chromosome evolution during each 33-ms control period and (**b**) flowchart of GA process steps during convergence performance from the first to final generation

to optimize the genes for each input image. Again, the convergence of the genes to a better solution with higher fitness value than the previous generation is achieved within 33 ms. By performing this procedure repeatedly, RM-GA searches the optimum solution that represents the real target's true pose, as shown at the bottom of Fig. 6a.

Regarding the RM-GA system hardware, the two cameras used for 3D imaging had the following specifications: CCD imaging elements, 380,000 pixels, NTSC video system, minimum illumination of 0.8 lx, and no zoom. The specifications of the PC used for 3D pose estimation were Intel Core(TM) i7-3517UE CPU operating at 1.70 GHz with RAM of 4,096 MB and a 64-bit operating system. Two PCI-5523 interface boards were installed in the PC to receive images from the two cameras. The convergence performance of the GA as a dynamic evaluation function was proved mathematically by a Lyapunov analysis in a previous study [30]. Also, the effectiveness of the GA search was demonstrated in a previous study on visual servoing for catching fish [31].

Optimization of the number of generations, population size, selection rate, and mutation rate is discussed in detail in the next section.

5 Experiment for Optimizing RM-GA Parameters

5.1 Experiment Environment

A static environment is defined as an environment that does not contain any moving objects, while a dynamic environment has moving objects (i.e., moving targets and robots). In the dynamic environment, square and fair comparisons are very difficult for convergence time because the vehicle or marker motion can disturb the GA convergence process. Therefore, the dynamic performance of GA image fitting convergence was analyzed in a static environment with the cameras and marker in fixed positions. The layout of the experiment environment is shown in Fig. 7. The frame was designed to hold the experimental devices firmly. The experiment was done in a tank (2,870 mm long × 2,010 mm wide × 1,000 mm deep) filled with clear water. The horizontal distance between the cameras was 178 mm, which is the same distance between the ROV's cameras. For the search area, as shown in Fig. 3, the distance between the 3D marker and the two cameras was set at 415 mm. The distance between the center of the 3D marker and the bottom of the tank was 120 mm, and the distance between from the center of the camera to the bottom of the tank was 130 mm. Power and control signals from the PC were transmitted through a tether cable.

5.2 Number of Generations Based on Size of Chromosome Population

The use of correct population size is an important factor for successful GA application, so we first analyzed how many generations could be created within 33 ms based on different chromosome population sizes. Figure 8 shows the number of generations formed when the chromosome population size was varied from 10 to 500. According to the graph, as the population size increased, the number of generations formed within 33 ms decreased. The maximum number of generations was 37 for a population of 10, and the minimum number was 1 for populations of 360-500 chromosomes. The experimental results for the number of generations informed the population sizes used in further experiments, such as the convergence performance of dynamic image analysis described in the next section.

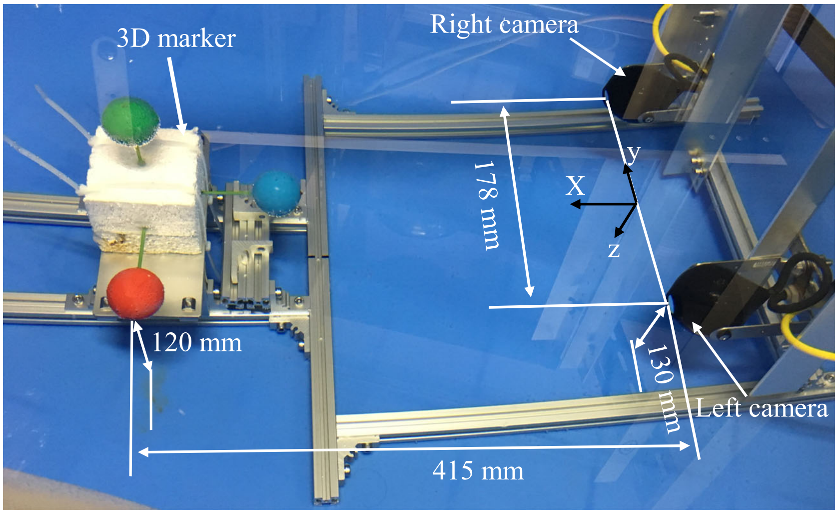
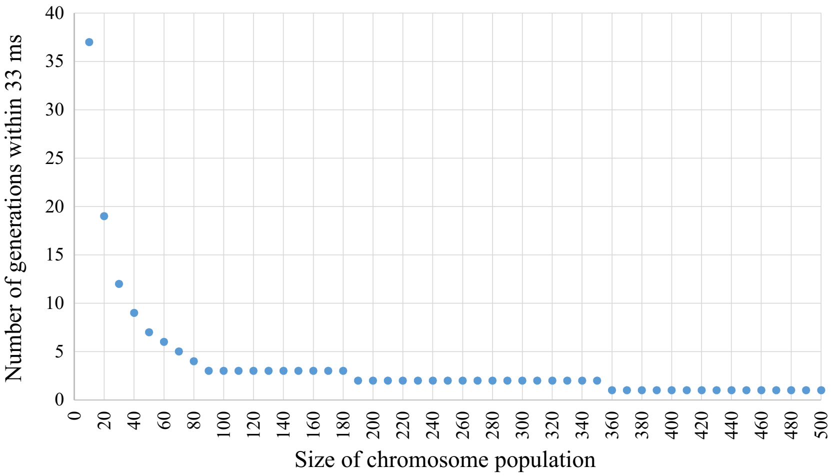


Fig. 7 Layout of experiment using 3D marker and two cameras

Fig. 8 Number of generations created by RM-GA within 33 ms based on different chromosome population sizes



5.3 Convergence Performance of RM-GA Using Dynamic Images

In the RM-GA process, selection, crossover, and mutation operators act on every generation to evolve the best pose chromosome toward the true pose of the target object (x, y, z, $\epsilon_1, \epsilon_2, and \epsilon_3$) within 33 ms. The primary parameters used to optimize the accuracy of the system were selection rate, mutation rate, and population size. The selection rate indicates the percentage of chromosomes that will be reproduced to form the next generation, and the mutation rate indicates the percentage of chromosomes that are subject to random bit changes at the end of each generation. The image convergence performance of RM-GA was analyzed with different chromosome population sizes, selection rates, and mutation rates while holding the fitness value and crossover technique constant.

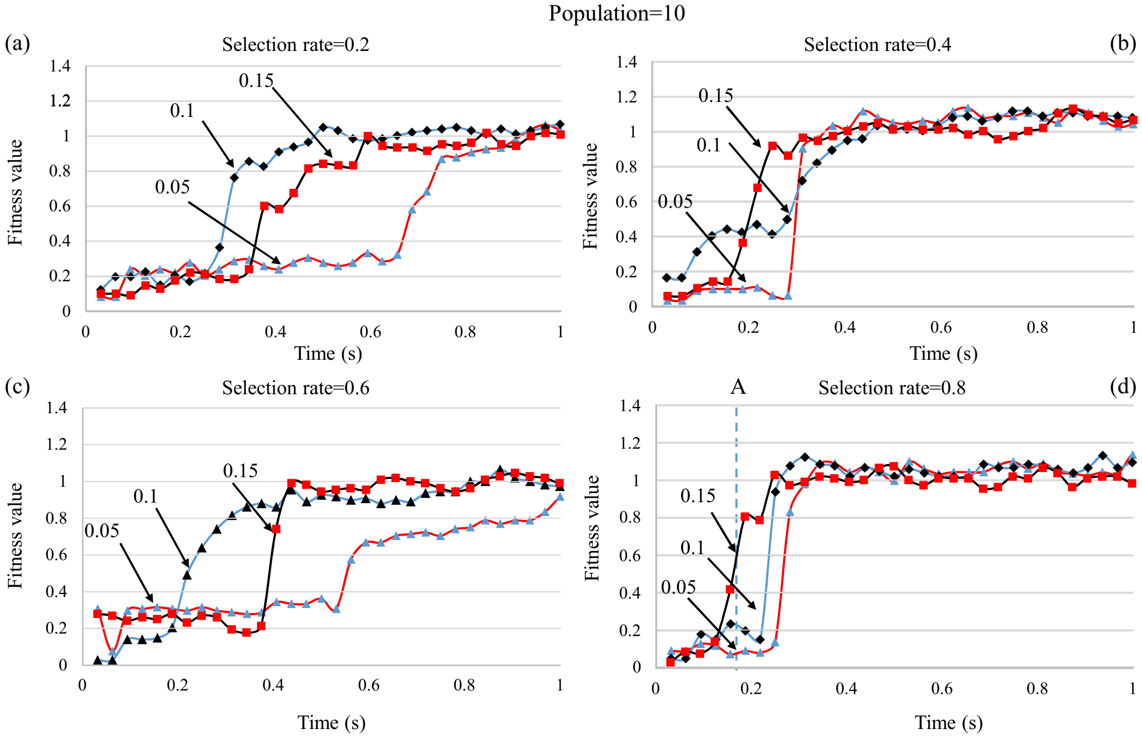


Fig. 9 Comparison of convergence time for fitness value of 0.6 and population size of 10 with different selection and mutation rates: selection rates = 0.2, 0.4, 0.6, and 0.8 and mutation rates = 0.05, 0.1, and 0.15. The vertical dotted line “A” shows the convergence time of the best combination of RM-GA parameters for a population of 10. The parameters of “A” are given in Table 1

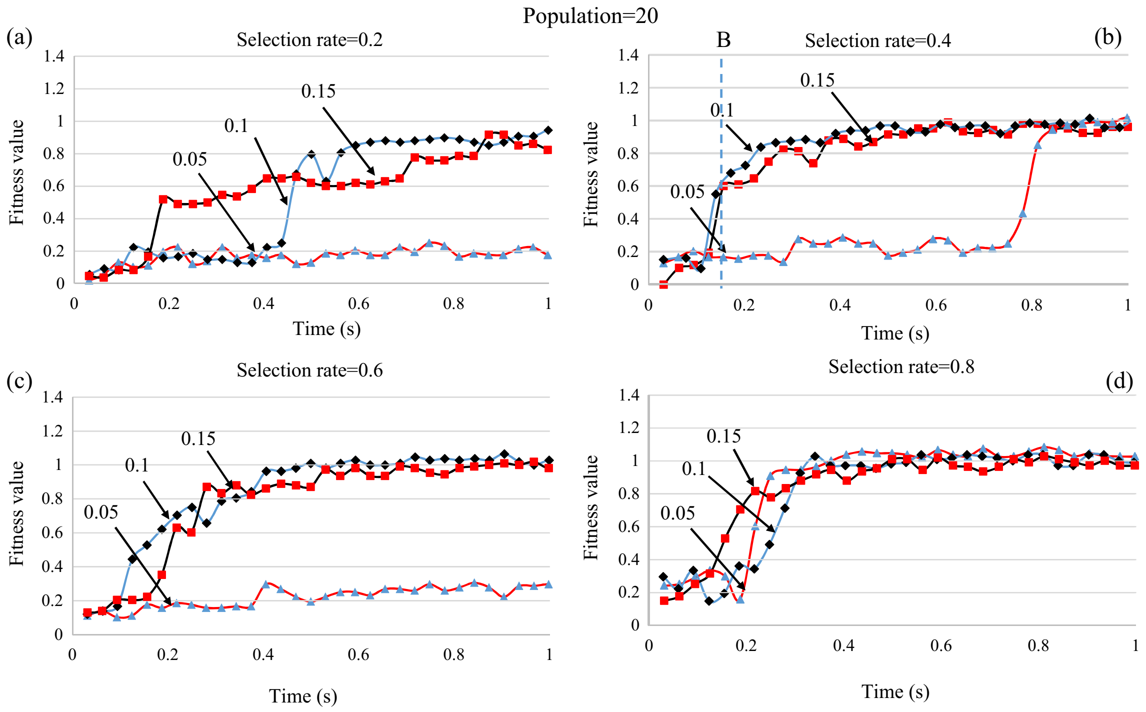


Fig. 10 Comparison of convergence time for fitness value of 0.6 and population size of 20 with different selection and mutation rates: selection rates = 0.2, 0.4, 0.6, and 0.8 and mutation rates = 0.05, 0.1, and 0.15. The vertical dotted line "B" shows the convergence time of the best combination of RM-GA parameters for a population of 20. The parameters of "B" are given in Table 1

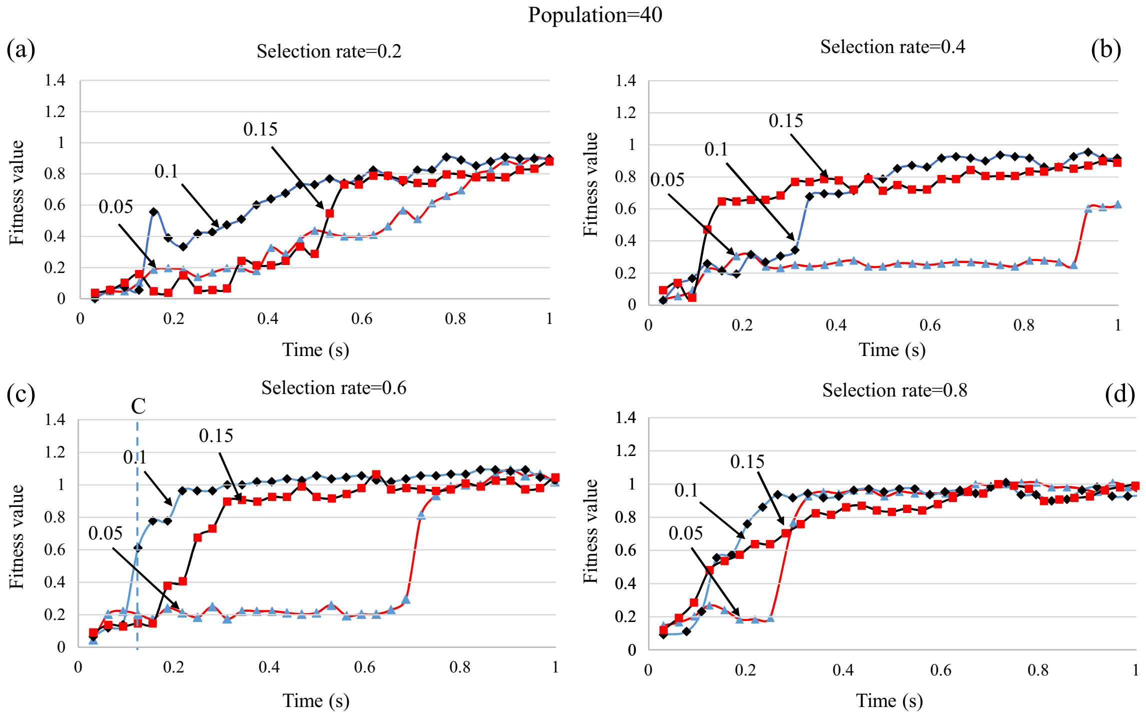


Fig. 11 Comparison of convergence time for fitness value of 0.6 and population size of 40 with different selection and mutation rates: selection rates = 0.2, 0.4, 0.6, and 0.8 and mutation rates = 0.05, 0.1, and 0.15. The vertical dotted line "C" shows the convergence time of the best combination of RM-GA parameters for a population of 40. The parameters of "C" are given in Table 1

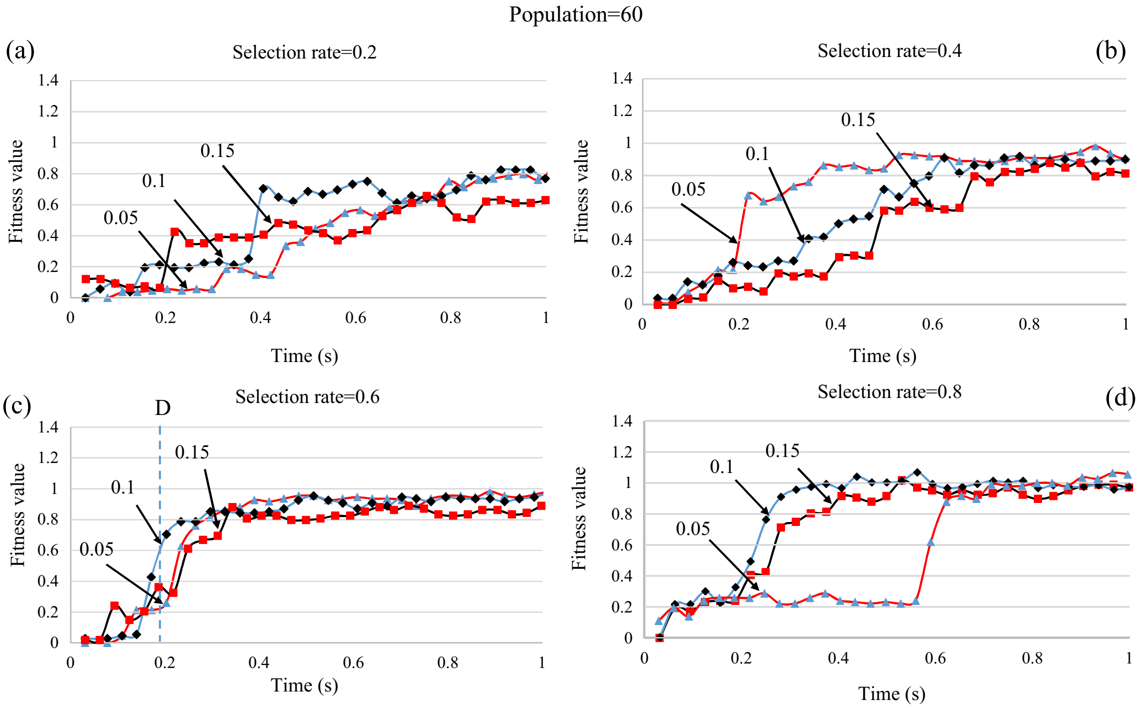


Fig. 12 Comparison of convergence time for fitness value of 0.6 and population size of 60 with different selection and mutation rates: selection rates = 0.2, 0.4, 0.6, and 0.8 and mutation rates = 0.05, 0.1, and 0.15. The vertical dotted line "D" shows the convergence time of the best combination of RM-GA parameters for a population of 60. The parameters of "D" are given in Table 1

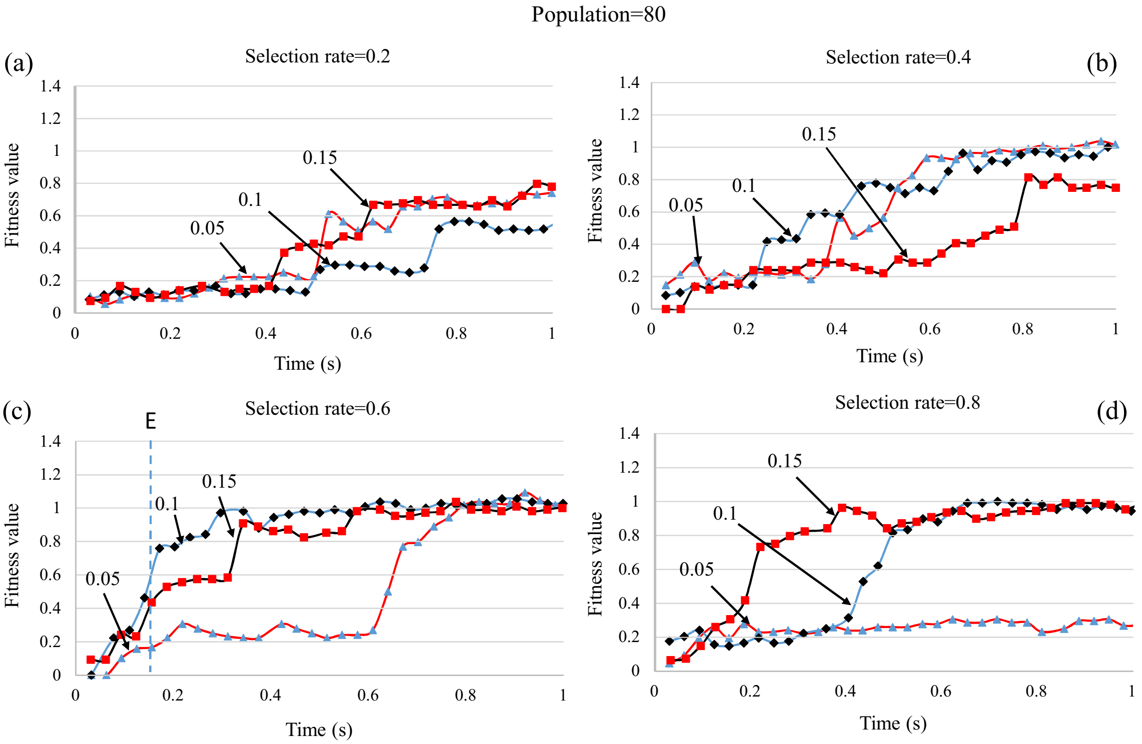


Fig. 13 Comparison of convergence time for fitness value of 0.6 and population size of 80 with different selection and mutation rates: selection rates = 0.2, 0.4, 0.6, and 0.8 and mutation rates = 0.05, 0.1, and 0.15. The vertical dotted line "E" shows the convergence time of the best combination of RM-GA parameters for population 80. The parameters of "E" are given in Table 1

Table 1 Primary parameter values that produced the fastest convergence time for each population size with a fitness value of 0.6

Case	Population size	Selection rate (%)	Mutation rate (%)	Convergence time (s)
A	10	0.8	0.15	0.218
B	20	0.4	0.1	0.156
C	40	0.6	0.1	0.125
D	60	0.6	0.1	0.203
E	80	0.6	0.1	0.172

The convergence performance of RM-GA was analyzed using a fitness value of 0.6 and various population sizes (10, 20, 40, 60, and 80), selection rates (0.2, 0.4, 0.6, and 0.8), and mutation rates (0.05, 0.1, and 0.15). The values for these primary parameters, as well as the fitness value, were chosen based on experiments. Also, a fitness value of 0.6 was deemed as representing good performance. Each different population value produced the number of generations shown in Fig. 8. The experimental results are shown in Figs. 9-13. In total, 60 combinations of different population sizes, selection rates, and mutation rates were analyzed.

The experimental results of the convergence time for a population size of 10 and the various values for selection and mutation rates are shown in Fig. 9. The fastest time for convergence of GA image recognition was obtained with a selection rate of 0.8 and mutation rate of 0.15, which produced a convergence in 0.187 s, as shown by dotted line “A” in Fig. 9d.

The other results population sizes of 20, 40, 60, and 80 and different selection and mutation rates are shown in Figs. 10–13. In Fig. 10b, the combination of selection rate of 0.4 and mutation rate of 0.1 had the fastest convergence time for a population of 20. The rapid convergence behavior of GA can be seen in Fig. 11c for a population of 40, selection rate of 0.6, and mutation rate of 0.1, where convergence was achieved in 0.125 s. In Fig. 12c, the convergence performance of GA with a population of 60 was fastest for a selection rate of 0.6 and mutation rate of 0.1; convergence was reached in 0.203 s. In Fig. 13c, the fastest convergence time for a population of 80 can be seen for a selection rate of 0.6 and mutation rate of 0.1. Table 1 summarizes the primary parameter values that obtained the best convergence times with a fitness value of 0.6 for each population size. Figure 14 compares the fastest convergence times, as indicated by the lines labeled “A”-“E” in Figs. 9-13.

Considering the effect of changing the selection and mutation rates in the population sizes evaluated, the effect of the mutation rate seems more important because it supports the generation of new solutions in the search space. The effect of the mutation rate was noticed at a value of 0.05 in different population sizes, as the system could not converge to the solution in the available calculation time. This slow convergence time means that the small mutation rate did not generate enough new and better solutions. For most population sizes, a mutation rate of 0.1 provided better

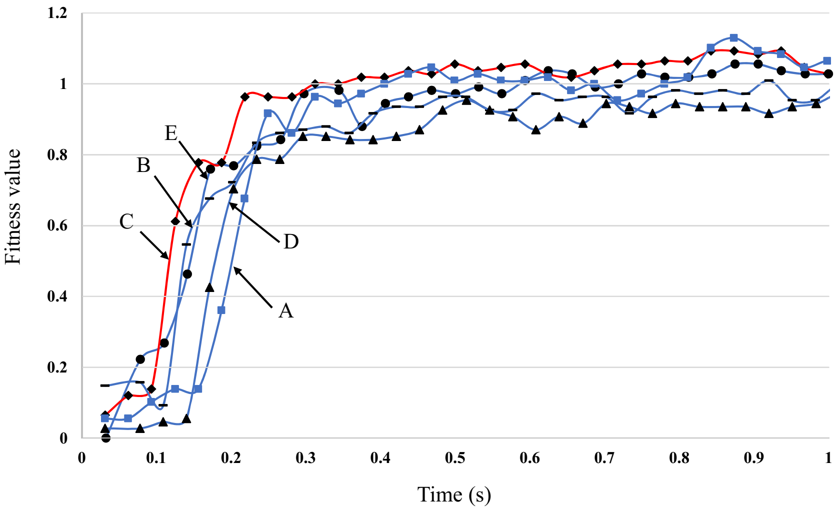


Fig. 14 Comparison of effect of optimal primary parameters on fitness value for population sizes of 10, 20, 40, 60, and 80 and increased calculation time. Values for selection and mutation rates are given in Table 1

Table 2 Optimum parameters for RM-GA

Chromosome population size	40
Search area (mm)	$[x, y, z] = \pm 400 \pm 200 \pm 400$
Selection rate (%)	60
Crossover technique	Two points
Mutation rate (%)	10
Control period (ms)	33
Number of generations (per 33-ms period)	9
Evolution strategy	Elitist preservation

results, although the higher rate of 0.15 was better for a population of 10.

Overall, the fastest convergence of 0.125 s was obtained with a population size of 40, selection rate of 0.6, and mutation rate of 0.1 The optimum RM-GA parameters used in the sea docking experiments are given in Table 2.

6 Sea Docking Experiment using the Optimum Parameters of RM-GA

We conducted the docking experiment in the sea to verify the effectiveness of the real-time 3D pose estimation system using the optimum parameters obtained in tank experiments. Even though the effectiveness of the proposed system in docking experiments was confirmed under different environmental disturbances in previous work [32–37], this was the first time we checked the system's tolerance against turbidity. Real sea environments may degrade the visual ability of the system beyond that produced in a simulated environment with a pool due to turbidity, sunlight, and waves. Since the proposed system was confirmed to be effective against other disturbances, we inferred that it would also be effective against turbidity. To ensure the presence of turbidity, the docking experiment was conducted in a turbid coastal environment rather than clear oceanic water.

6.1 Experimental System and Conditions

6.1.1 Structure of Docking Station

The layout of the sea docking experiment is shown in Fig. 15. The docking station was a cubic prism 600 mm long × 450 mm wide × 3,000 mm tall, oriented with the long sides parallel to a pier. The 3D marker and docking hole were fixed in the docking station. The diameter of the docking hole was 130 mm, and the center distance between the marker and docking hole was 145 mm. Two underwater cameras were attached to the docking station for monitoring the behavior of the ROV during the experiments and recording it for later analysis. The ROV was tethered

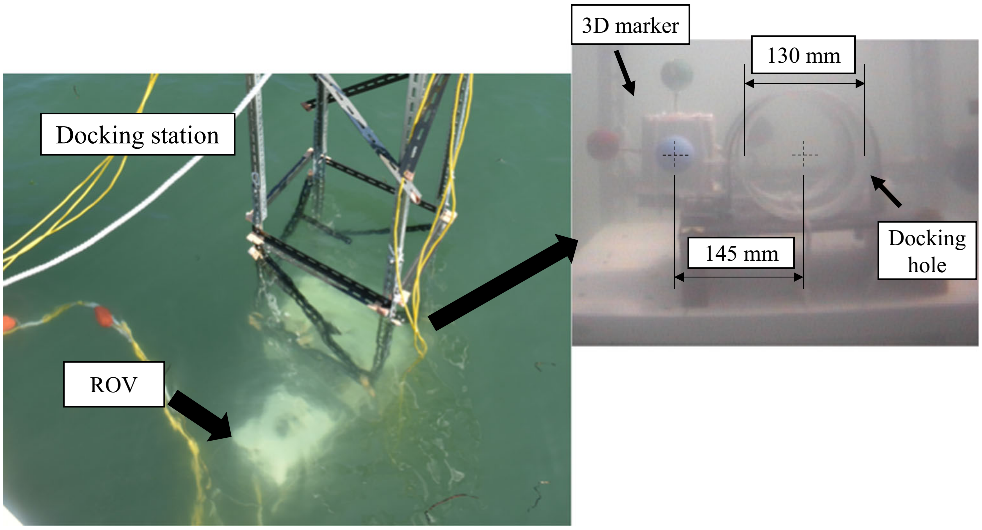


Fig. 15 Layout of the station for sea docking

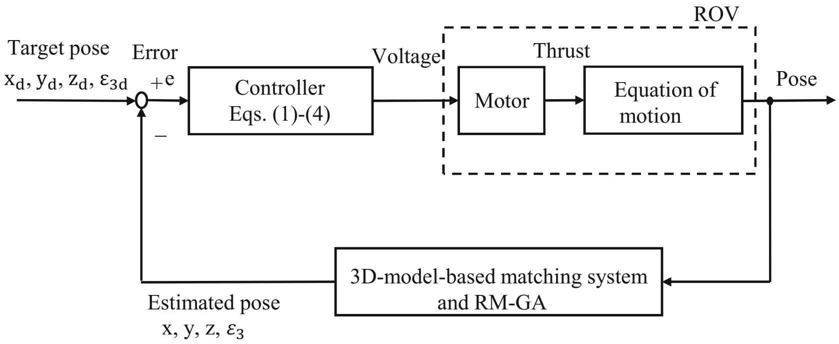


Fig. 16 Block diagram of servo control using visual feedback

and connected by a 200-mm cable to the RM-GA PC (controller and 3D pose estimator) on the pier.

6.1.2 Environmental Conditions

The sea docking experiment was conducted on the coast of Okayama Prefecture, Japan. The environmental conditions during the experiment were as follows: the time of day was about 15:45 p.m., the water depth from the surface to the sea bottom was 2.1 m, the turbidity level was 10 Formazin turbidity units, the illumination at the sea surface was 80,000 lx, the illumination underwater at a depth of 1 m was 3,000 lx, and sea wave action was continuous during the experiment. The turbidity level was measured using a portable turbidity monitor (TD-M500, OPTEX).

6.1.3 3D Motion Controller

The control system is depicted as a block diagram in Fig. 16. The four thrusters mounted on the ROV were controlled by sending a command voltage based on feedback regarding the difference between the current pose $(x,\ y,\ z,\ \epsilon_3)$ of the ROV, represented by Σ_H in Fig. 17, and the desired pose $(x_d,\ y_d,\ z_d,\ \epsilon_{3d})$. In this study, the control of rotation around the x-axis and y-axis in Σ_H was neglected because the orientation of the ROV (center of buoyancy located above the center of gravity) was autonomously maintained in a stable horizontal state. The control voltages of the four thrusters were calculated as follows:

$$\begin{aligned}&\text{Surge direction}: v_1 = k_{p1}(x_d - x) + 2.5\\ &\qquad (v_1 = 0\text{V for thrust of } 9.8\text{ N in } X_H \text{of } \Sigma_H \text{in Fig. 17}\\ &\qquad v_1 = 5\text{ V for } -9.8\text{N})\end{aligned} \tag{2}$$

$$\begin{aligned}&\text{Sway direction}: v_2 = k_{p1}(y_d - y) + 2.5\\ &\qquad (v_2 = 0\text{ V for thrust of } 4.9\text{ N in } X_H \text{of } \Sigma_H \text{in Fig. 17}\\ &\qquad v_2 = 5\text{ V for } -4.9\text{N})\end{aligned} \tag{3}$$

$$\begin{aligned}&\text{Orientation around } Z_H \text{ axis}: v_3 = k_{p2}(\epsilon_{3d} - \epsilon_3) + 2.5\\ &\qquad (v_3 = 0\text{V for } 0.882\text{N in } Z_H \text{ of } \Sigma_H \text{ in Fig. 17}\\ &\qquad v_3 = 5\text{ V for } -0.882\text{ N})\end{aligned} \tag{4}$$

$$\begin{aligned}&\text{Heave direction}: v_4 = k_{p3}(z_d - z) + 2.5\\ &\qquad (v_4 = 0\text{ V for } -4.9\text{ N in } Z_H \text{ of } \Sigma_H \text{in Fig. 17}\\ &\qquad v_4 = 5\text{ V for } 4.9\text{ N})\end{aligned} \tag{5}$$

In Eqs. 1–4, v_1, v_2, and v_4 are the control voltages of the four thrusters for the movement of the ROV in the x, y, and z directions, respectively. Here, x_d, y_d, z_d, and ϵ_{3d} specify the desired relative pose between the vehicle and the target, and ϵ_{3d} is the orientation around the z-axis, which is controlled by the value of v_3. A proportional controller was used for all thrusters. We were prepared to add integration and derivative terms in the controller if such adjustments became necessary, but they were unnecessary for these experiments. The gain coefficients were adjusted to achieve the best condition for visual servoing in the docking process based on preliminary experiments.

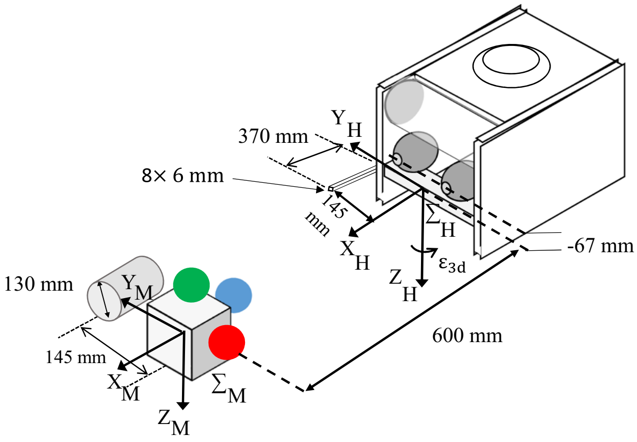


Fig. 17 Layout of docking experiment. The ROV had a rod ($8mm \times 6$ mm) on its right-hand side, and the target had a round hole with a diameter of 130 mm on its left-hand side

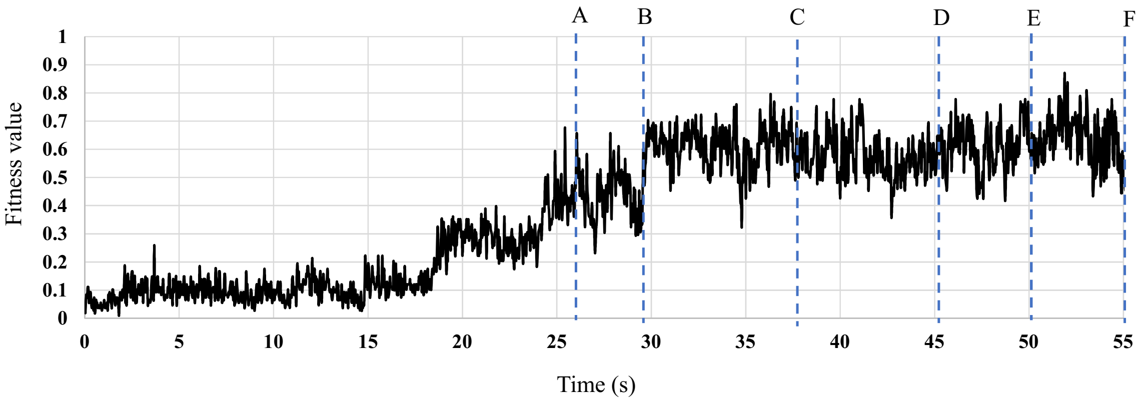


Fig. 18 Fitness value results for sea docking experiment. Detailed positioning results for the times corresponding to lines “A”-“F” are given in Fig. 19

6.1.4 Docking Strategy

Here, we provide a brief explanation of the docking strategy for the reader’s convenience. A more detailed discussion is provided in [32]. The goal of docking was to insert a rod attached to the vehicle into a docking hole. The layout of the docking experiment is shown in Fig. 17. The homing stage was not considered in this experiment. In the docking strategy, the docking operation has four stages: (1) manual operation, (2) visual servoing, (3) docking, and (4) completion of docking. The ROV approached the docking station by manual operation until the object was in the

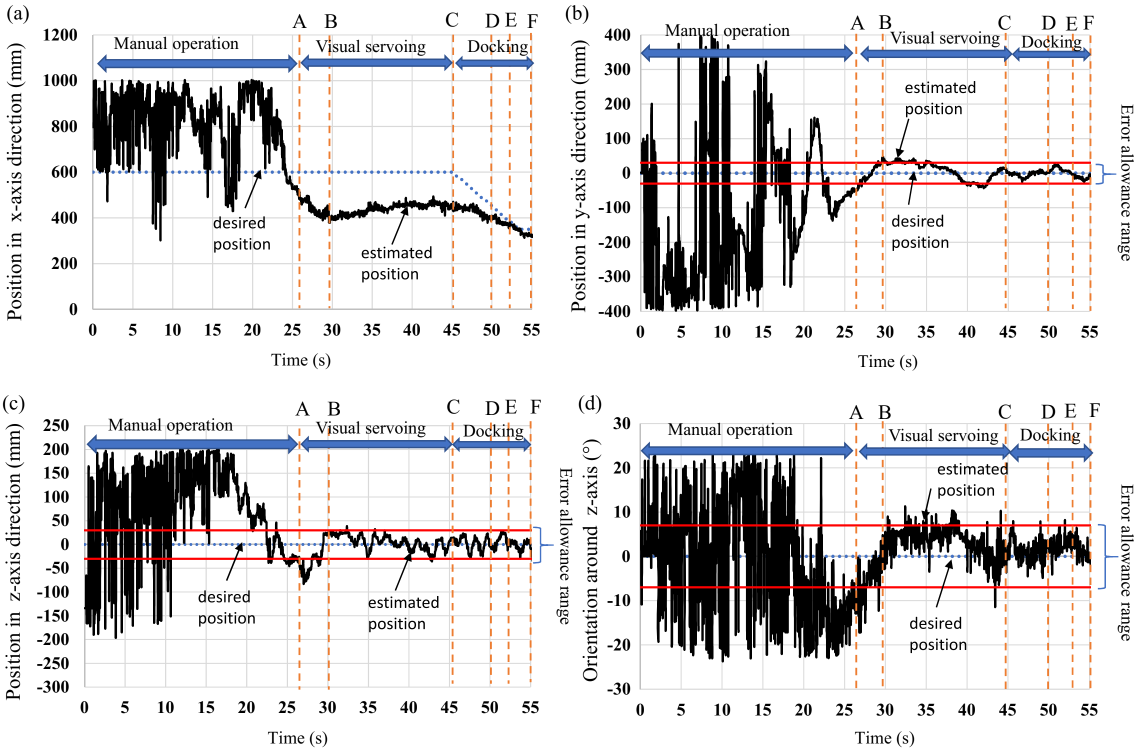


Fig. 19 Docking experiment in actual sea environment with turbidity: (**a**) position in x-axis direction, (**b**) position in y-axis direction, (**c**) position in z-axis direction, and (**d**) orientation around z-axis. The dotted lines “A”-“F” in each panel correspond to docking stages as follows: (**A**) end of manual operation, (**B**) transition to visual servoing, (**C**) start of docking, and (D-F) completion of docking. Photographs taken at these times are shown in Fig. 20

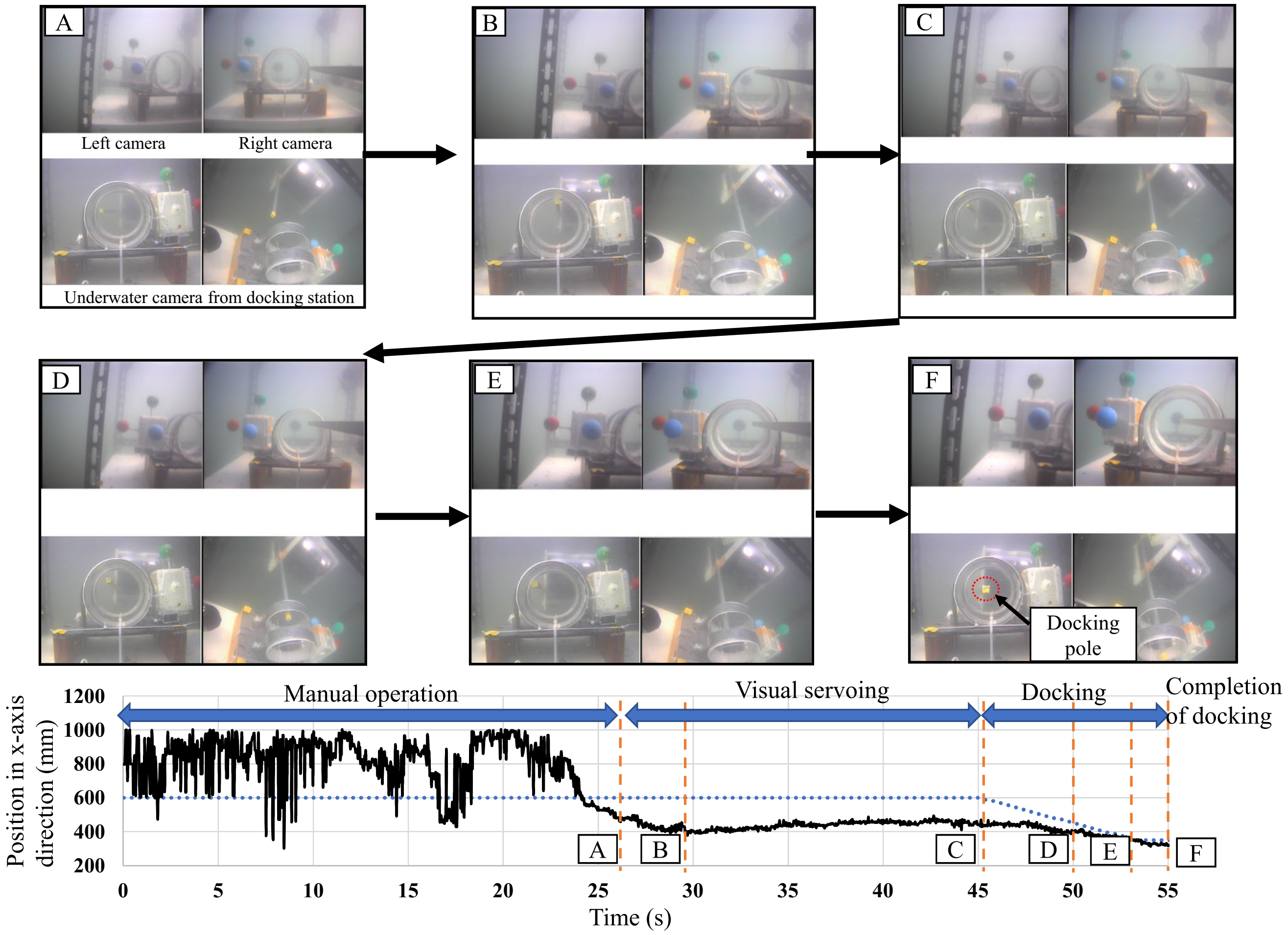


Fig. 20 Photographs of docking experiment in actual sea environment with turbidity and position in x-axis direction: Each panel corresponds to the time denoted dotted lines “A”-“F” in Fig. 19. Note the placement of the docking pole in the docking hole in photograph ”F.” In each panel, the upper photographs are the left and right camera images acquired by the ROV and the lower photographs were taken by two underwater monitoring cameras (back view and top view of docking hole from cameras inside the docking station)

camera field of view. Visual servoing started when the 3D marker was detected, which corresponded to a fitness value of 0.2. This means that if a pose had a fitness value that was less than 0.2, this pose was not used in the feedback system, signaling that the target may not be in the field of view. In other words, when the fitness value reached 0.2, the visual servoing stage started. When the pose of the vehicle was stable within the error allowance range of ±30 mm, and the orientation around the z-axis was controlled to within 7° for the desired period (165 ms, which is equal to five times the control loop period), docking started by decreasing the distance between the ROV and the 3D marker from 600 mm to 350 mm. Finally, docking was completed by closing the distance from 350 mm. Then, the vehicle stopped visual servoing for a few seconds after docking completion to store the experimental data in memory for later analysis.

6.2 Analysis and Discussion

In this section, the results of the sea docking experiment using the optimum RM-GA parameters are analyzed and discussed. First, a detailed discussion of docking performance is given in Section 6.2.1 in terms of practical performance. Second, RM-GA performance in terms of genetic distribution is analyzed and discussed in detail in Section 6.2.2 to verify the optimization performance of RM-GA in the actual sea environment.

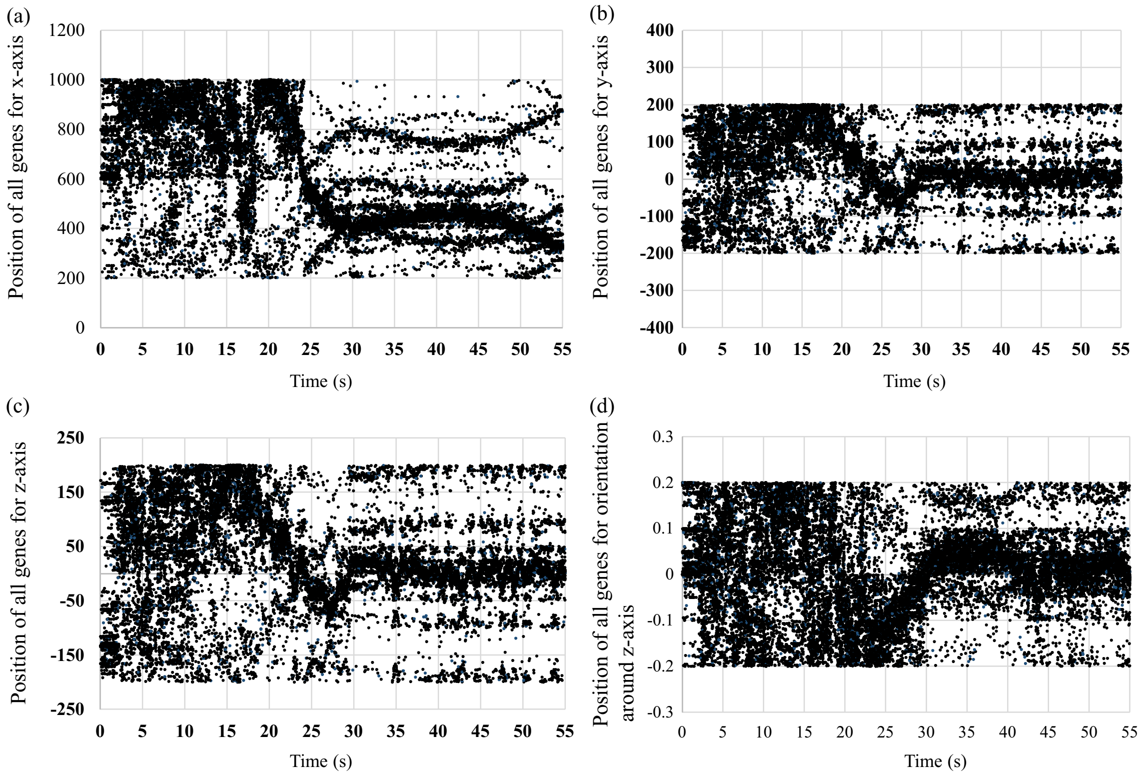


Fig. 21 All genetic data analyses: (**a**) position of all genes for x-axis, (**b**) position of all genes for y-axis, (**c**) position of all genes for z-axis, and (**d**) position of all genes for orientation around z-axis

6.2.1 Verification of Control Function

The ROV is controlled to maintain a predefined relative pose ($x_d = 600$ mm , $y_d = 0$ mm, $z_d = 0$ mm, and $\epsilon_{3d} = 0°$) between the target and the ROV so that the ROV performs the regulating through visual servoing. The fitness value of the sea docking experiment is shown in Fig. 18. The results of the sea docking experiment are shown in Fig. 19, which shows the positions along the x, y, and z axes and the orientation around the z-axis. The vertical dotted lines in each panel, denoted by "A"-"F," indicate the docking stage, as explained in the caption for Fig. 19. The pairs of red horizontal lines in Fig. 19b, c and d are the error allowance range for the docking start position. Photographs corresponding to the results of Fig. 19 are shown in Fig. 20. In this figure, each photograph corresponds to one of the times denoted by the vertical lines "A"-"F."

At the start of the experiment, the vehicle moved from the starting point and manually approached the 3D marker until it was in the field of view of the camera, as can be seen in each panel of Fig. 19. The ROV did not correctly recognize the marker at the initial stage of the experiment. Note that pose estimation during the manual operation is not accurate and not used for feedback control, since the fitness value was below 0.2, as shown in Fig. 18. During the long manual navigation in the first stage, significant continuous oscillation occurred for the first 25 s.

After completing the manual stage, the ROV moved forward and transitioned to visual servoing by switching to automatic control after 25 s (dotted lines "A"-"C" in Fig. 19). At this time, the fitness value was above 0.6, and the vehicle easily recognized the 3D marker. Then, the underwater robot servoed to the desired relative target pose. The dotted line "A" in each subfigure marks the start of the visual servoing stage. At time "B," there was a gap between the desired position and the estimated position in the x-axis direction because the error allowance for the docking operation was defined for only the y- and z-axes and rotation about the z-axis. At that time, the position in y-axis exceeded the error allowance range, as shown in Fig. 19b. The vehicle rotated around the z-axis to its left (based on Σ_H – see Fig. 17).

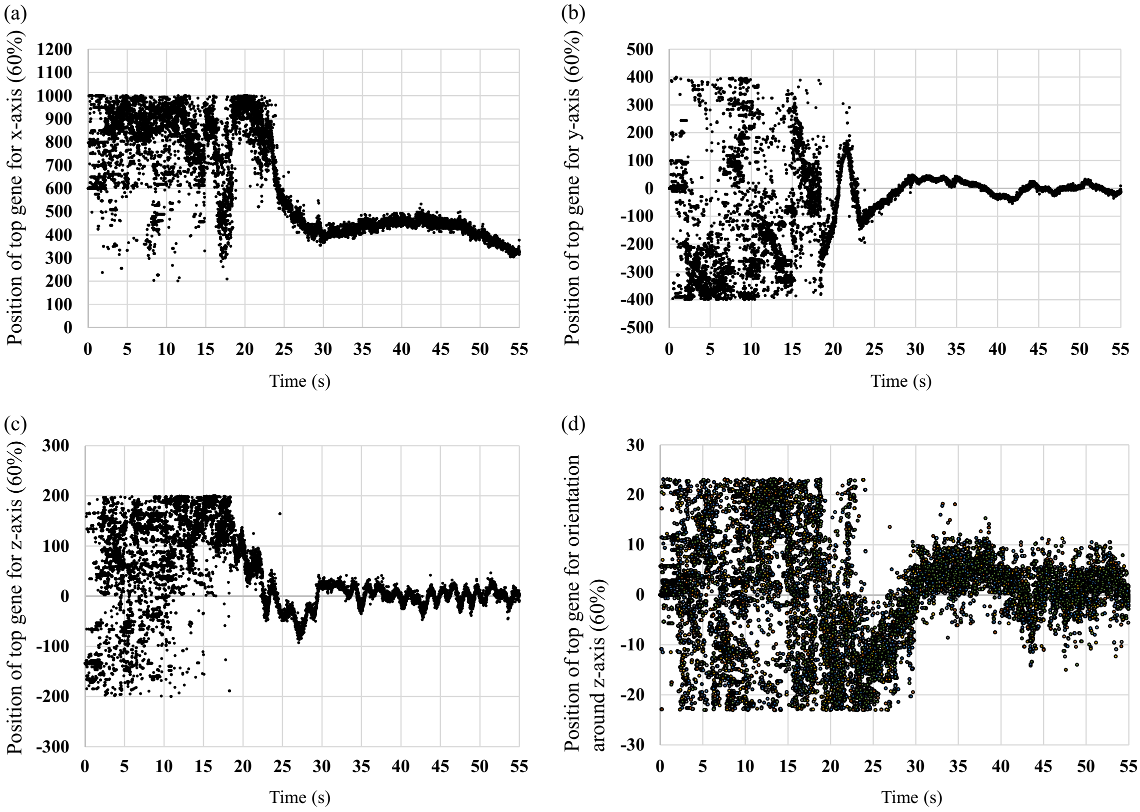


Fig. 22 Top genes (60 %) data analyses: (**a**) position of top genes for x-axis, (**b**) position of top genes for y-axis, (**c**) position of top genes for z-axis, and (d) position of top genes for orientation around z-axis. The top genes are ranked and selected to represent the true pose

In the docking stage, which started at line "C," the pose of the vehicle was stable within the error allowance range, and the docking step was performed by reducing the distance between the vehicle and the marker in the x-axis direction from 600 mm to 350 mm. In the x-axis direction in Fig. 19, it can be confirmed that the robot performed the docking operation after 45 s (line "C"). At that time, the y-axis, z-axis, and orientation around in z-axis positions were within the error allowance range, as shown in Fig. 19, panels (b)-(d). As the docking stage continued, the pose of the vehicle remained stable within the error allowance range during the docking operation, as can be seen at times "D" and "E." At some points, the orientation around the z-axis exceeded the error allowance range. However, the ROV could perform the docking operation by adjusting the orientation around the z-axis to compensate for the error. Finally, the docking operation was completed smoothly at about 55 s (see dotted line "F"). In photograph "F" of Fig. 20, the part surrounded by the dotted circle is the docking pole tip and it is correctly fitted in the docking hole. The vehicle stopped visual servoing for a few seconds after docking completion to store the experimental data. The experimental results confirmed the robustness of the 3D pose estimation and the docking experiment in an actual sea environment with turbidity.

6.2.2 Genetic Evolution during Sea Docking Experiment

In this section, the performance of the RM-GA was confirmed by analyzing all genetic data of the position and orientation along the x, y, and z axes and around the z-axis during the docking process, as shown in Fig. 21. Each black dot in Fig. 21 represents one pose of the target object created by RM-GA. Dark areas indicate convergence of the top genes and the other areas represent the distribution of bottom genes. We confirmed that the distribution of all genes was within the search area (see Fig. 3).

The top genes – that is, the top 60% of genes after ranking by the fitness function – are searched and selected to represent the true pose of the target object. However, the bottom genes – the bottom 40% after ranking – are

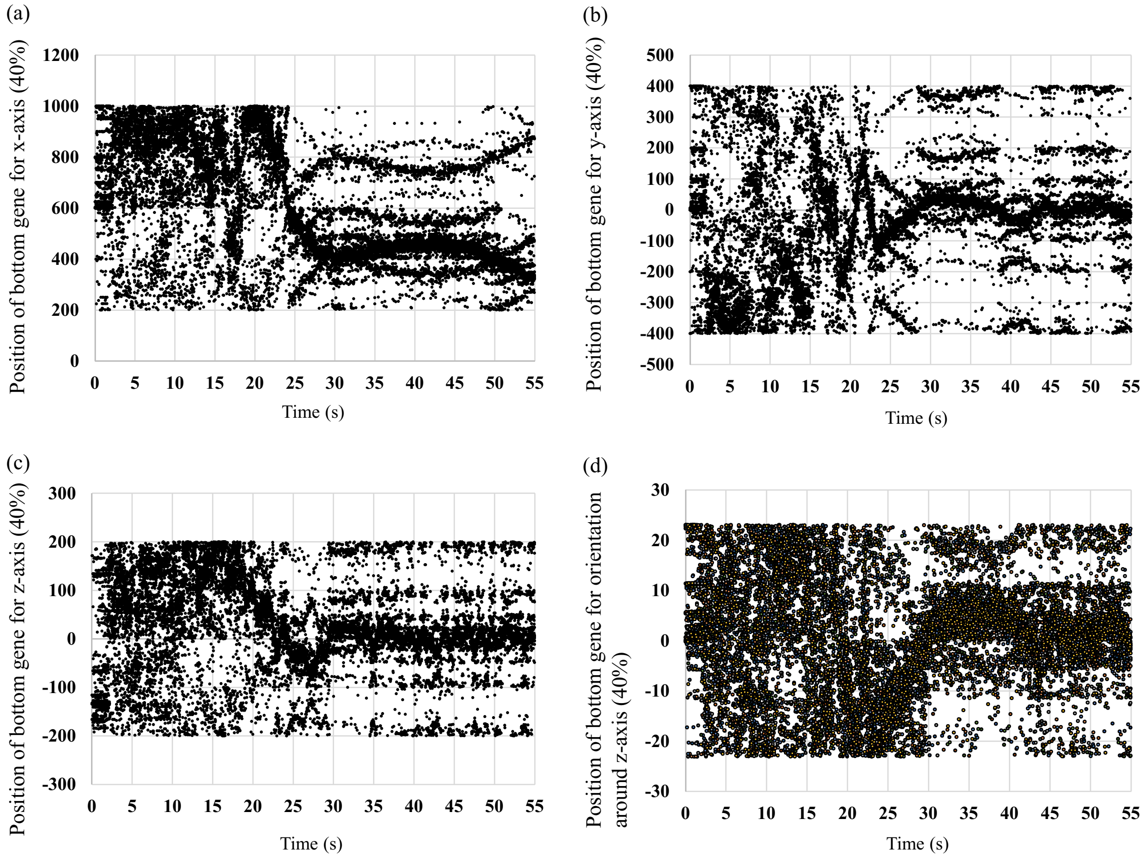


Fig. 23 Bottom genes (40 %) data analyses: (**a**) position of bottom genes for x-axis, (**b**) position of bottom genes for y-axis, (**c**) position of bottom genes for z-axis, and (**d**) position of bottom genes for orientation around z-axis

also analyzed to check that they are distributed well, which allows the top genes to converge to the solution quickly, and that they are diverged enough to cover the search area. The top 60% of genes in the x, y, and z axes and orientation around in z-axis converged to the real solution, as shown in Fig. 22, and the other 40% of genes were divergent, as shown in Fig. 23. At the start of the experiment, all genes were distributed randomly along the position axes, as shown in Figs. 22 and 23. After 25 s, the top genes (60%) converged to the real solution and the bottom genes (40%) remained well distributed within the search area. However, the bottom genes diverged enough not to miss the target when the ROV moved quickly. The result, shown in Fig. 21, confirmed that the top genes converged to the real solution with a selection rate of 0.6. The mean values and the standard deviation of position along the position axes are shown in Figs. 24 and 25. The standard deviation of all genes with respect to position along the x-axis was within 250 mm, position along the y-axis within 300 mm, and position along the z-axis within 140 mm, and the orientation around the z-axis was within 10°. Since the shape of the mean value in Fig. 24 is similar to that of the top gene's pose in Fig. 19, it can be concluded that all genes are uniformly distributed in the search space. The range of standard deviation in Fig. 25 also confirmed that the space searched by RM-GA had coverage sufficient to not miss the 3D marker.

According to the analyses and discussions for the sea docking experiment using the optimum parameters of RM-GA explained here, we can reach the following conclusions: (1) The functionality and practicality of the proposed system was confirmed according to successful sea docking experiments in which real-time poses estimated by RM-GA provided direct feedback for position control. (2) Real-time pose estimation using optimum RM-GA parameters was verified for the intended application, that is, docking for underwater battery recharging. Convergence to the solution within 33 ms using optimum parameters was proved by analysis of RM-GA performance.

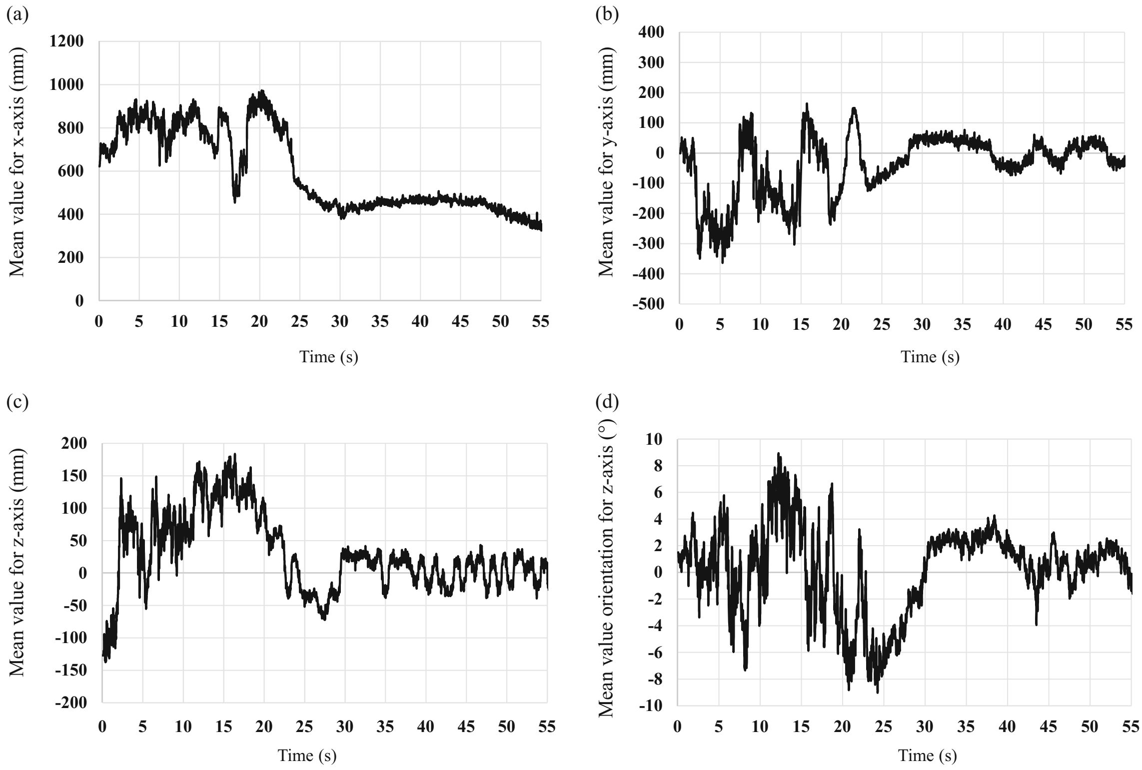


Fig. 24 Mean value of all genes analyses: (**a**) mean value for x-axis, (**b**) mean value for y-axis, (**c**) mean value for z-axis, and (**d**) mean value for orientation around z-axis. The shape of the mean value is similar to the shape of top gene's pose in Fig. 22

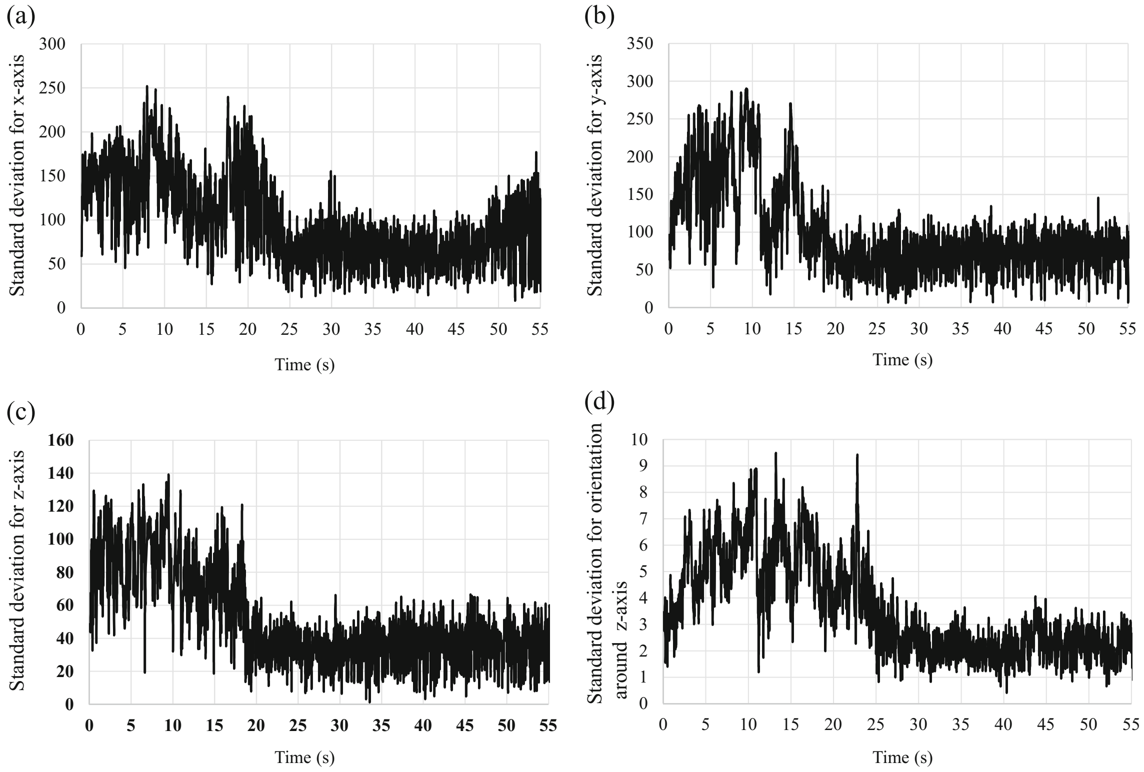


Fig. 25 Standard deviation of all genes analyses: (**a**) standard deviation for x-axis, (**b**) standard deviation for y-axis, (**c**) standard deviation for z-axis, and (**d**) standard deviation for orientation around z-axis

7 Conclusion

This paper presents performance analyses and optimization of real-time pose estimation for an underwater ROV using a 3D marker and dual-eye cameras. The optimum parameters for the RM-GA were selected based on the time for convergence using dynamic images in a static environment. We confirmed and analyzed the data from the sea docking experiment using the optimum RM-GA parameters. According to the experimental results, the proposed system can converge to the real solution in the RM-GA evolution process using dynamic images. Experimental results showed that the recognition accuracy of the system was optimized to errors on the order of millimeters for real-time pose estimation. In the future, a recharging experiment using an actual battery will be conducted with an autonomous underwater vehicle in the sea.

Acknowledgements This work supported by JSPS KAKENHI Grant Number JP16K06183. The authors would like to thank Mitsui E&S Shipbuilding Co., Ltd.; and Kowa Corporation for their collaboration and support for this study. The authors would also like to express their thanks to Dr. Yuya Nishida, Prof. Kazuo Ishii, Prof. Toshihiko Maki, and Prof. Tamaki Ura for their helps and supports.

References

1. Aoyagi, S., Hattori, N., Kohama, A., Komai, S., Suzuki, M., Takano, M., Fukui, E.: Object detection and recognition using template matching with SIFT features assisted by invisible floor marks. J. Robot. Mechatron. **21**(6), 689 (2009)
2. Tomono, M.: 3D object modeling and segmentation using image edge points in cluttered environments. J. Robot. Mechatron. **21**(6), 672 (2009)
3. Brandou, V., Allais, A.G., Perrier, M., Malis, E., Rives, P., Sarrazin, J., Sarradin, P.M.: 3D reconstruction of natural underwater scenes using the stereovision system iris. In: OCEANS 2007-Europe, pp. 1–6. IEEE (2007)
4. Manikandan, G., Sridevi, S., Dhanasekar, J.: Vision based autonomous underwater vehicle for pipeline tracking, International Journal of Innovative Research in Science, Engineering and Technology, vol. 1, pp. 2347–6710 (2015)
5. Eustice, R.M., Pizarro, O., Singh, H.: Visually augmented navigation for autonomous underwater vehicles. IEEE J. Ocean. Eng. **33**(2), 103–122 (2008)
6. Ghosh, S., Ray, R., Vadali, S.R., Shome, S.N., Nandy, S.: Reliable pose estimation of underwater dock using single camera: a scene invariant approach. Mach. Vis. Appl. **27**(2), 221–36 (2016)
7. Kume, A., Maki, T., Sakamaki, T., Ura, T.: A method for obtaining high-coverage 3D images of rough seafloor using AUV-real-time quality evaluation and path-planning. J. Robot. Mechatron. **25**(2), 364–374 (2013)
8. Guo, J.: Mooring cable tracking using active vision for a biomimetic autonomous underwater vehicle. J. Mar. Sci. Technol. **13**(2), 147–15 (2008)
9. Vallicrosa, G., Bosch, J., Palomeras, N., Ridao, P., Carreras, M., Gracias, N.: Autonomous homing and docking for AUVs using Range-Only Localization and Light Beacons. IFAC-PapersOnLine **49**(23), 54–60 (2016)
10. Palomeras Rovira, N., Peñalver, A., Campos Massot, M., Black, P.L., Fernández, J.J., Ridao Rodríguez, P., Oliver Codina, G.: i-AUV Docking Panel and Intervention at Sea. Sensors **16**(10), 1673 (2016)
11. Myint, M., Yonemori, K., Lwin, KN., Yanou, A., Minami, M.: Dual-eyes Vision-based Docking System for Autonomous Underwater Vehicle: an Approach and Experiments, J Intell Robot Syst. https://doi.org/10.1007/s10846-017-0703-6 (2017)
12. Uddin, M., Abido, M.A., Rahman, M.A.: Nasir Real-Time implementation of a genetic algorithm based fuzzy logic controller for interior permanent magnet synchronous motor drive proceeding of ICECE (2002)
13. Nguyen, V.B., Morris, A.S.: Genetic algorithm tuned fuzzy logic controller for a robot arm with two-link flexibility and two-joint elasticity. J. Intell. Robot. Syst. **49**(1), 3–18 (2007)
14. Park, J.-Y., Jun, B.-H., Lee, P.-M., Oh, J.: Experiments on vision guided docking of an autonomous underwater vehicle using one camera. IEEE J. Ocean. Eng. **36**(1), 48–61 (2009)
15. Palomeras, N., Ridao, P., Ribas, D., Vallicrosa, G.: Autonomous i-AUV docking for fixed-base manipulation, in Proc. Int. Fed. Autom. Control **47**(3), 12160–12165 (2014)
16. Rao, R.V., Savsani, V.J.: Advance Optimization Techniques. Mechanical design optimization using advanced optimization techniques, pp. 5–34. Springer Science & Business Media (2012)
17. Preechakul, C., Kheawhom, S.: Modified genetic algorithm with sampling techniques for chemical engineering optimization. J. Ind. Eng. Chem. **15**(1), 110–118 (2009)
18. Cui, H., Turan, O.: Application of a new multi-agent hybrid co-evolution based particle swarm optimization methodology in ship design. Comput.-Aided Des. **2**, 1013–1027 (2010)
19. Zou, Y., Luo, D.: A modified ant colony algorithm used for multi-robot odor source localization. In: International Conference on Intelligent Computing, pp. 502–509. Springer, Berlin (2008)
20. Rexhepi, A., Maxhuni, A., Dika, A.: Analysis of the impact of parameters values on the Genetic Algorithm for TSP. Int. J. Comput. Sci. Issues **10**(1), 158–164 (2013)
21. Boyabatli, O., Sabuncuoglu, I.: Parameter selection in genetic algorithms, Journal of Systemics. Cybern. Inf. **4**(2), 78 (2004)
22. Tabassum, M., Mathew, K.: A genetic algorithm analysis towards optimization solutions. Int. J. Digit. Inf. Wirel. Commun. (IJDIWC) **4**(1), 124–142 (2014)
23. Roeva, O., Fidanova, S., Paprzycki, M.: Influence of the population size on the genetic algorithm performance in case of cultivation process modelling. In: 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 371–376. IEEE (2013)
24. Lwin, K.N., Yonemori, K., Myint, M., Mukada, N., Minami, M., Yanou, A., Matsuno, T.: Performance analyses and optimization of real-time multi-step GA for visual-servoing based underwater vehicle, Techno-Ocean 2016. IEEE (2016)
25. Yu, F., Minami, M., Song, W., Zhu, J., Yanou, A.: On-line head pose estimation with binocular hand-eye robot based on evolutionary model-based matching. J. Comput. Inf. Technol. **2**(1), 43–54 (2012)
26. Minami, M., Agbanhan, J., Asakura, T.: Evolutionary Scene Recognition and Simultaneous Position Orientation Detection. In: Soft Computing in Measurement and Information Acquisition, pp. 178–207. Springer, Berlin (2003)

27. Mehrez, M.W., Mann, G.K., Gosine, R.G.: An optimization based approach for relative localization and relative tracking control in multi-robot systems. J. Intell. Robot. Syst. **85**(2), 385–408 (2017)
28. GiriRajkumar, S.M., Ramkumar, K., Sanjay, S.O.V.: Real time application of ants colony optimization. Int. J. Comput. Appl. **3**(8), 0975–8887 (2010)
29. Mousavian, S.H., Koofigar, H.R.: Identification-based robust motion control of an AUV: optimized by particle swarm optimization algorithm. J. Intell. Robot. Syst. **85**(2), 331–352 (2017)
30. Song, W., Fujia, Y., Minami, M.: 3D visual servoing by feedforward evolutionary recognition. J. Adv. Mech. Des. Syst. Manuf. **4**(4), 739–755 (2010)
31. Suzuki, H., Minami, M.: Visual servoing to catch fish using global/local GA search. IEEE/ASME Trans. Mechatron. **10**(3), 352–357 (2005)
32. Myint, M., Yonemori, K., Yanou, A., Minami, M., Ishiyama, S.: Visual-servo-based autonomous docking system for underwater vehicle using dual-eyes camera 3D-pose tracking. In: 2015 IEEE/SICE International Symposium on System Integration(SII), pp. 989–994 (2015)
33. Myint, M., Yonemori, K., Yanou, A., Lwin, K.N., Minami, M., Ishiyama, S.: Visual-based deep sea docking simulation of underwater vehicle using dual-eyes cameras with lighting adaptation. In: Proceedings of OCEAN 2016-Shanghai, pp. 1–8 (2016)
34. Myint, M., Yonemori, K., Yanou, A., Lwin, K.N., Mukada, N., Minami, M.: Dual-eyes visual-based sea docking for sea Bottom battery recharging. OCEAN (2016)
35. Lwin, K.N., Yonemori, K., Myint, M., Yanou, A., Minami, M.: Autonomous docking experiment in the sea for visual-servo type underwater vehicle using three-dimensional marker and dual-eyes cameras. In: Society of Instrument and Control Engineers of Japan (SICE), 2016 55th Annual Conference, pp. 1359–1365. IEEE (2016)
36. Myint, M., Yonemori, K., Yanou, A., Lwin, K.N., Minami, M., Ishiyama, S.: Visual servoing for underwater vehicle using dual-eyes evolutionary real-time pose tracking. J. Robot. Mechatron. **28**(4), 543–558 (2016)
37. Myint, M., Yonemori, K., Yanou, A., Ishiyama, S., Minami, M.: Robustness of visual-servo against air bubble disturbance of underwater vehicle system using three-dimensional marker and dual-eye cameras. In: OCEANS 2015-MTS/IEEE, pp. 1–8, Washington (2015)

Khin Nwe Lwin received her B.E. degree of Mechatronic engineering from Technological University (Kyaukse), Myanmar in 2006. She has obtained her M.E. degree of Mechatronic engineering in 2011 from Mandalay Technological University, Mandalay, Myanmar. She received Ph.D. degree in Division of Industrial Innovation Sciences, Graduate School of Natural Science and Technology, Okayama University, Japan in 2018. Currently, she is a Lecturer on Faculty of Precision Engineering in University of Technology (Yatanarpon Cyber City), Myanmar. Her research interests include visual servoing, control of autonomous underwater vehicle, robotics, and genetic algorithm.

Myo Myint received his M.E. and Ph.D. degrees in Electronic Engineering from Yangon Technological University, Mandalay Technological University, Myanmar, in 2006 and 2009, respectively. He worked as an Associate Professor in Yangon Technological University from 2011 to 2015. He received Ph.D. degree in Division of Industrial Innovation Sciences, Graduate School of Natural Science and Technology, Okayama University, Japan in 2018. Currently, he is an Associate Professor of Electronic Engineering in Yangon Technological University (Thanlyin), Myanmar. His research interests include autonomous underwater vehicles, robotics, electronics, and renewable energy.

Naoki Mukada received his B.E degree from Okayama University in 2016. He obtained his M.E degree in robotics from Okayama University in 2018. His research interests include 3D pose estimation, robotics and optimization.

Daiki Yamada received his B.E. degree form Okayama University in 2017. His research interests include 3D pose estimation, robotics and optimization.

Takayuki Matsuno received M.E. degree from Nagoya University in 2000. He has obtained his doctorated degree from Nagoya University in 2005. He worked for Nagoya University in 2004. He worked for Toyama University in 2006. In 2011, he worked in Okayama University. His research interests include manipulation of deformable linear objects.

Kazuhiro Saitou He is a technical staff at the Ushimado Marine Institute (UMI), Faculty of Science, Graduate School of Natural Science and Technology, Okayama University.

Waichiro Godou He is a technical staff at the Ushimado Marine Institute (UMI), Faculty of Science, Graduate School of Natural Science and Technology, Okayama University.

Tatsuya Sakamoto He has received Ph.D. University of Tokyo in 1992. From 2003, he is Professor & director at the Ushimado Marine Institute (UMI), Faculty of Science; Professor at the Graduate School of Natural Science and Technology, Okayama University, and from 2014 Chairperson of the Director-Council of Japanese National Marine and Inland Water Biological Stations.

Mamoru Minami He has received B.S. and M.S. degrees in Aeronautical engineering from the University of Osaka Prefecture in 1978 and 1980, respectively, and the Ph.D. degree in Faculty of Engineering from the Kanazawa University in 1992. Since 1994 he has been an Associate Professor of Mechanical Engineering at University of Fukui, he moved to a newly established department of Human and Artificial Intelligence Systems in the same University in 1999. He has been a professor there from 2002. He moved to Faculty of Engineering in Okayama University at April in 2010 as a professor. His current research interests are in the field of robotics and automation in dynamics, kinematics and control of robots including mobile manipulators, and intelligent motion control. And to research intelligent robot, he has been engaged in machine vision for real-time adaptive operation, stereo-vision, eye-vergence visual servoing, autonomous underwater vehicle and underwater docking system.