https://doi.org/10.20965/jrm.2022.p1063

Paper:

Real-Time Marker-Based Tracking and Pose Estimation for a Rotating Object Using High-Speed Vision

Xiao Liang[*], **Masahiro Hirano**[**], **and Yuji Yamakawa**[***]

[*]Department of Mechanical Engineering, The University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan
E-mail: liangxiao010@outlook.com
[**]Institute of Industrial Science, The University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan
E-mail: mhirano@iis.u-tokyo.ac.jp
[***]Interfaculty Initiative in Information Studies, The University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan
E-mail: y-ymkw@iis.u-tokyo.ac.jp

[Received March 28, 2022; accepted July 22, 2022]

Object tracking and pose estimation have always been challenging tasks in robotics, particularly for rotating objects. Rotating objects move quickly and with complex pose variations. In this study, we introduce a marker-based tracking and pose estimation method for rotating objects using a high-speed vision system. The method can obtain pose information at frequencies greater than 500 Hz, and can still estimate the pose when parts of the markers are lost during tracking. A robot catching experiment shows that the accuracy and frequency of this system are capable of high-speed tracking tasks.

Keywords: high-speed vision, pose estimation, marker, stereo vision, rotating object

1. Introduction

In a robotic system, the component providing the perception, processing, and interpretation of visual signals is very important. Object tracking and pose estimation are essential and challenging tasks for this component. In view of the combination of these two problems, most robotics applications must estimate the 3D trajectories of objects.

Rotating objects [1] such as diabolos and in-flight rings have more complex motion trajectories with both translations and high-frequency rotations. The high-speed rotation increases the linear velocity of the tracking points on the rotating objects. Moreover, the corresponding objects (such as gyroscopes) rotate along multiple axes, leading to more complex orientation variations.

Yashima et al. [2] implemented robot catching for an in-flight ring using a high-speed visual tracking system. Both the position and orientation of the ring were estimated during the in-flight trajectory, and the robot adjusted the catching pose until the ring was successfully obtained. Murooka et al. [3] used a depth camera to measure the position and orientation of a diabolo based on its geometric features and led a PR2 robot to stabilize the orientation of a diabolo.

For these robotic applications, easy usage, high-frequency tracking, and accurate pose estimation are considered as critical elements. Recently, many advanced pose estimation methods [4,5] have applied deep learning to predict the 3D bounding boxes of objects in an image, and to calculate the corresponding 6D poses in the real world. These methods are usually relatively slow, owing to their high computational costs. Meanwhile, existing motion capture systems [6–8] such as OptiTrack [a] and Xsens [b] require a large working space, multiple advanced cameras, and/or manual marker labeling, making expensive system non-portable and inconvenient to use.

Several high-speed vision systems have been developed for providing high-frequency visual feedback [9,10]. In [11], the authors proposed a 1 kHz target tracking system applying a simple strategy using image moments to track a target. However, it was designed to estimate the 3D positions without orientation information of the moving objects. In our previous works [12, 13], we used reflective markers in a high speed vision system; thus, the tracking required an initialization step to manually label the starting locations of the markers. In [14], a stereo vision algorithm was designed to track the positions and orientations of spheroidal objects. To accurately measure the 3D positions and orientations, a stereo vision system equipped with two high-speed cameras was required. The stereo vision system had a higher pose estimation accuracy than the monocular system, and did not require a fixed distance between markers. However, for stereo vision, marker-based methods also require initialization labeling, implying that the target object must be inside the camera view space from the beginning.

In this study, we designed a high-speed object tracking and pose estimation algorithm with an easy setup system for handling fast-moving objects, particularly fast-

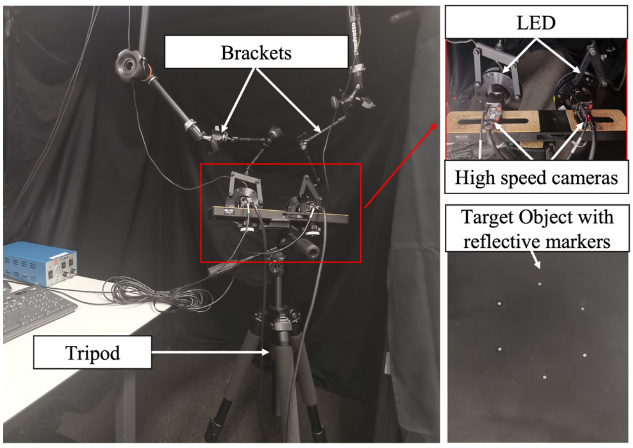


Fig. 1. System setup.

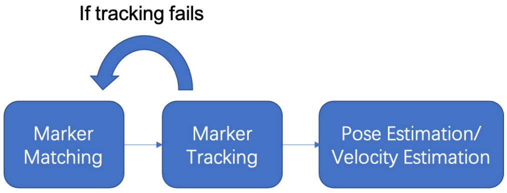


Fig. 2. Overall pipeline.

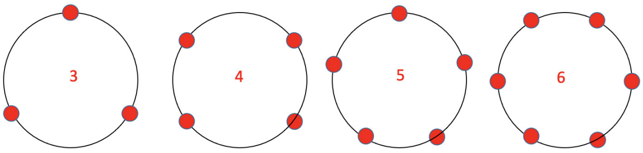


Fig. 3. Marker pattern.

rotating objects. The target objects should be planar or have a visible planar face that is not occluded by other faces. A robot-catching application using the proposed vision system with three markers was published in [15], and focused on the robot system and control strategy. In this study, we focus on the vision system and pose estimation algorithm. Moreover, we design new marker-matching and circular fitting methods to extend the marker number to over three. Compared with [15], the methodology proposed herein provides more robust tracking and more accurate pose estimation. The main contributions of our study are as follows.

- To avoid the manual labeling of markers, we propose a marker-matching algorithm for registering markers in left and right views when markers appear for the first time, or when they appear again after being temporarily lost.
- We design a regular-polygon (circular) marker pattern to make it more robust to marker occlusion. Regular polygon (circular) fitting could be achieved with parts of the markers under tracking.
- An adaptive local window tracking method is applied to improve the image processing speed.
- We implement a robot catching experiment to demonstrate the accuracy and high frequency of this system. The vision system can be mounted on a robot to serve as a body vision system, owing to its easy setup and small volume.

2. System Setup

Figure 1 shows the entire system setup, including high-speed cameras (Ximea MQ013MG-ON, 1,000 fps at an image resolution of 640 × 400) and LED lights.

The two high-speed cameras are fixed on the tripod platform with the bases on the same plane. Two brackets are used to fix two LED lights around the two high-speed cameras respectively. The three OptiTrack (PrimeX 22) cameras are installed in the bottom of the experiment space. The data measured by an OptiTrack camera system served as ground-truth for evaluating the accuracy of the high-speed camera system. An HP Z840 workstation with Intel Xeon E5 CPUs is used for the image processing in the proposed high-speed vision system. In addition, an EOL UNI-XW-E5HS computer with Intel Xeon E5 CPUs serves the OptiTrack system.

3. Methodology

3.1. Overview

Figure 2 shows the overall pipeline for the object tracking and pose estimation. The algorithm first matches the marker pairs in the left and right views. Then, it tracks each marker in the local window with the registered label. When the tracking fails, it performs marker matching and then tracks all of the markers again. A stereo linear triangulation method is applied to calculate the 3D position of each marker. A circular-fitting-based method is applied to calculate the position and orientation of the target object using the detected markers' positions. Compared with a monocular positioning method using "Perspective-n-Point," the stereo positioning with epipolar geometry optimization provides more accurate results. In addition, it can offset the circular radius error using circular fitting. In this way, there is no need to strictly fix the distances among the markers.

3.2. Sorting Based Stereo Matching Algorithm

As shown in **Fig. 3**, the markers are aligned in regular polygon patterns. Unlike fixing the marker number as three in [15], in this algorithm, there should be at least three markers (i.e., a greater number of markers can be used). The position of the circle center and orientation of the circular plane are estimated. **Fig. 4** shows a target object with three, four, and six markers.

The polar coordinate sorting method requires at most three steps, as shown in **Fig. 5**.

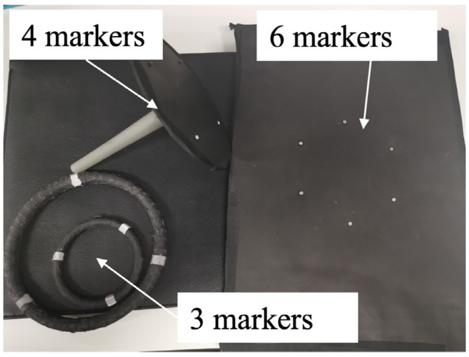


Fig. 4. Target object.

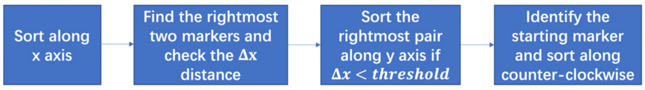

Sort along x axis
Find the rightmost two markers and check the Δx distance
Sort the rightmost pair along y axis if $\Delta x < threshold$
Identify the starting marker and sort along counter-clockwise

Fig. 5. Polar sorting flowchart.

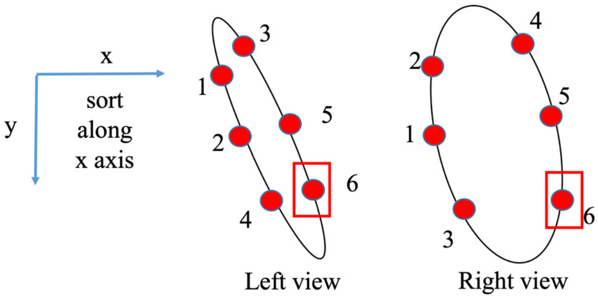


Fig. 6. Sort markers along *x*-axis.

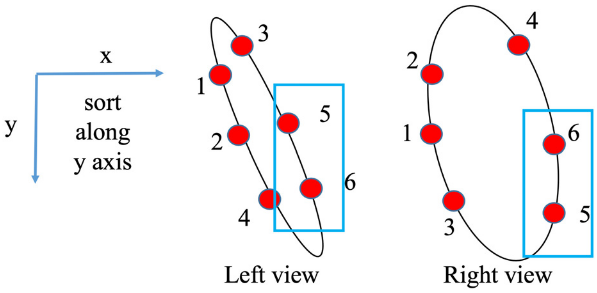


Fig. 7. Sort rightmost markers along *y*-axis.

First, we sort the markers in the left and right views along the *x*-axis. As shown in **Fig. 6**, the markers are labeled from 1 to 6 from left to right. Based on this sorting result, we can identify the rightmost marker with the ID 6.

However, when the rightmost two markers are very close along the *x*-axis as shown in **Fig. 7**, the makers 6 in the left view and right views are not matched with each other. For the right views, the marker 5 is on the left side of marker 6 in **Fig. 6**. However, the opposite is true for the case in **Fig. 7**.

The two markers in the blue box are sorted along the *y*-axis. The markers in the red box are then matched with each other, as shown in **Fig. 8**.

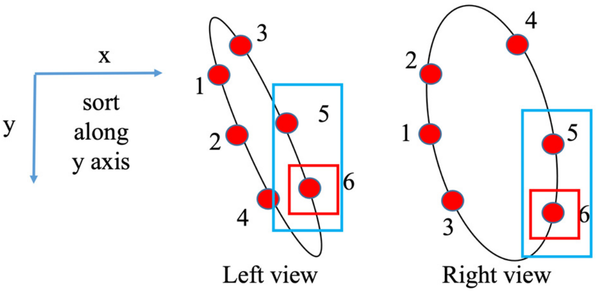


Fig. 8. Sort markers along *y*-axis and get starting marker in red box.

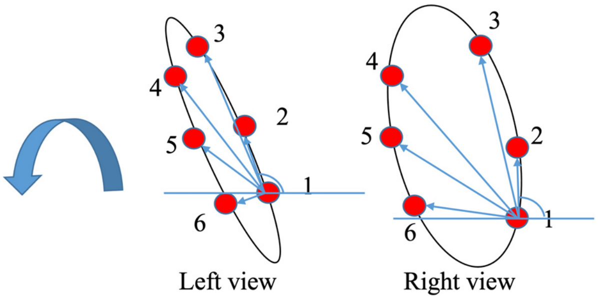


Fig. 9. Counter-clockwise sorting result.

Finally, the marker in the red box is selected as the starting marker. As shown in **Fig. 9**, the remaining markers are sorted in a counterclockwise order. Specifically, we obtain the angles of the vectors from the starting marker to the remaining markers. In this way, all markers are correctly matched in the left and right views.

As a summary, the matching procedures are as shown in **Algorithm 1**.

Figure 10 shows examples of the marker-matching results based on sorting along the polar coordinates. In the left and right views, the corresponding marker pairs are represented using different colors. The three examples show the matching results for different poses. In particular, for the first pose, the object is extremely tilted, and some of the markers are almost coincident. All of the matching results under the various poses demonstrate the robustness of the matching algorithm.

3.3. High-Speed Stereo Marker Tracking

We propose a dual-mode tracking algorithm for high-speed and accurate marker tracking. **Fig. 11** shows the overall tracking pipeline.

In the first frame, we have no prior knowledge of the pixel position of the marker. The global mode searches the entire image to find all of the marker contours, and records their centroid positions. If all of the markers are detected, the marker-matching algorithm is applied to match the left–right marker pairs. Then, the marker position is updated and the algorithm turns to local mode to track each marker in the subsequent frames.

Based on the prior marker position, the local mode only searches for markers in the local window around the cor-

Algorithm 1: Marker matching algorithm for sorting along polar coordinates.

```
$N$: number of markers; $\{L^i | i = 1, \ldots, N\}$, $\{R^i | i = 1, \ldots, N\}$: the marker centroid point sets for left and right views separately;
sort $\{L^i\}$ and $\{R^i\}$ along $x$-axis;
compute the gap distance along $x$-axis for the rightmost two markers: $\delta L^{N-1} = L_x^N - L_x^{N-1}$, $\delta R^{N-1} = R_x^N - R_x^{N-1}$;
compute the threshold $\delta x = (L_x^N - L_x^1 + R_x^N - R_x^1)/(4N-4)$;
if $\delta L^{N-1} \leq \delta x$ or $\delta R^{N-1} \leq \delta x$ then
    sort $\{L^{N-1}, L^N\}$ and $\{R^{N-1}, R^N\}$ along $y$-axis.
end if
initialize $L^N$ and $R^N$ as starting points;
compute the angles from $x$-axis
$\{\Theta_L^i = \arctan(L_x^i - L_x^N, L_y^i - L_y^N) | i = 1, \ldots, N-1\}$
$\{\Theta_R^i = \arctan(R_x^i - R_x^N, R_y^i - R_y^N) | i = 1, \ldots, N-1\}$
computed sorted $\{L^i | i = 1, \ldots, N-1\}$ according to the sorting result of $\{\Theta_L^i | i = 1, \ldots, N-1\}$;
computed sorted $\{R^i | i = 1, \ldots, N-1\}$ according to the sorting result of $\{\Theta_R^i | i = 1, \ldots, N-1\}$.
```

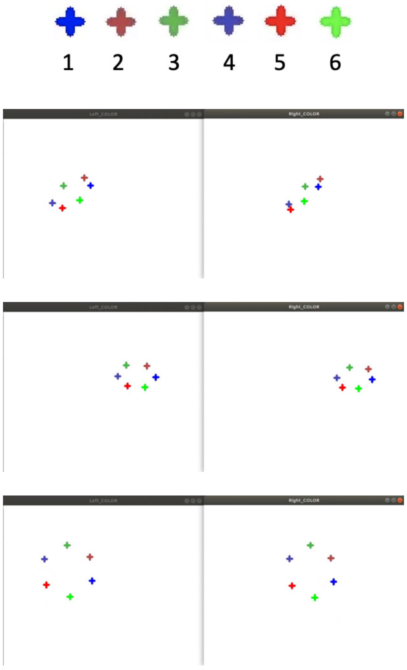


Fig. 10. Marker-matching results.

responding prior position. If the under-tracking marker number is larger than the minimum threshold, the tracked marker position is updated and the algorithm continues in local mode for subsequent frames. However, if many markers are lost during tracking and the minimum marker number is not reached, the algorithm will continue working in global mode until achieving success. The adaptive region of interest (ROI) tracking in the local mode is explained in detail below.

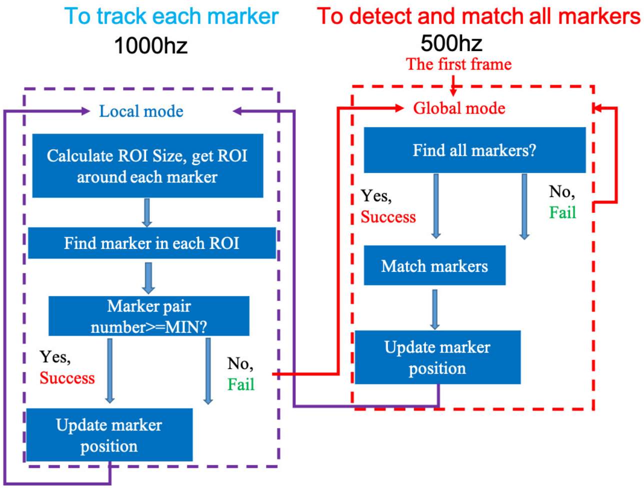
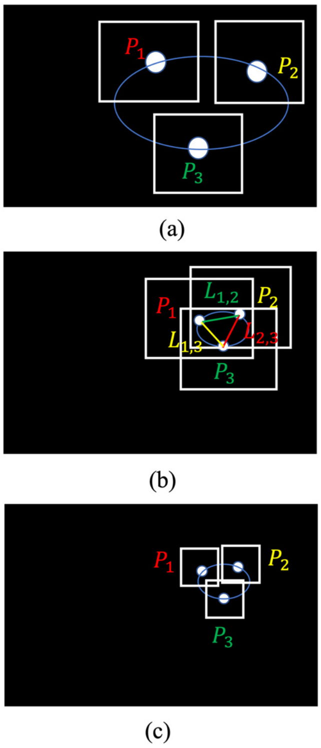


Fig. 11. Pipeline for marker tracking.



(a)

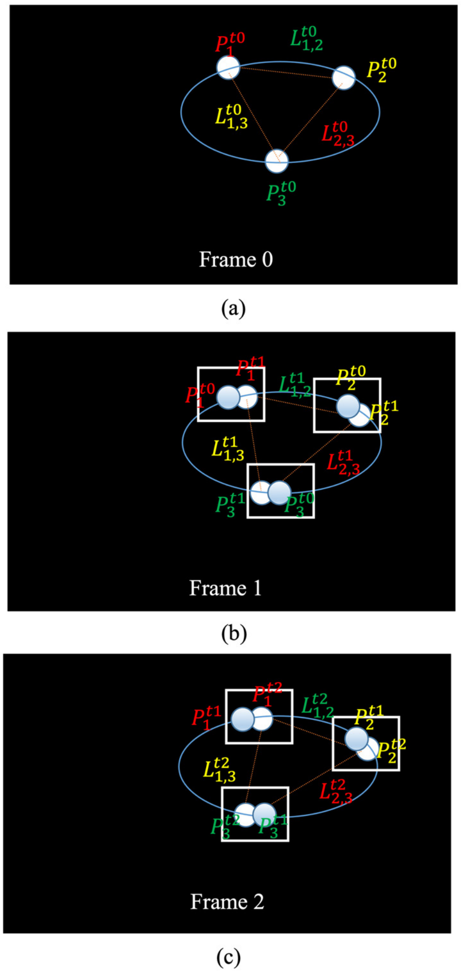
(b)

(c)

Fig. 12. Adaptive region of interest (ROI) box size.

3.3.1. Region of Interest (ROI) Box Size

In the local mode using ROI tracking, the previous study set a fixed ROI box size [16] for each tracked region, as shown in **Fig. 12(a)**. However, when a target object is far from the cameras, the ROI boxes can overlap with each other. As shown in **Fig. 12(b)**, we can choose the same marker in different ROI boxes if an overlap occurs. To solve this problem, for a marker number N, we



(a)

(b)

(c)

Fig. 13. Adaptive ROI box center.

choose the box size as the minimum polygon side length, as follows: $L = \min\{L_{i,j} | i, j = 1, \ldots, N; i \neq j\}$.

Figure 12(c) shows the calculated ROI box size. In this case, the different ROI boxes will never cover the same marker in our high-speed visual tracking.

3.3.2. ROI Box Center

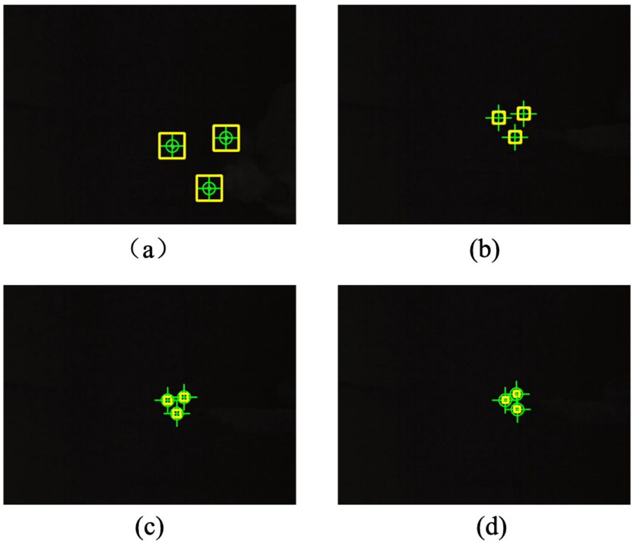
Instead of centering the ROI box (local window) at the recorded marker position in the last frame, we introduce a method for predicting the marker position in the new frame, and center of the ROI box at the predicted position. In the first frame (frame 0 in **Fig. 13(a)**), we record each marker pixel position as P_i^t. In the next frame (**Fig. 13(b)**), the position is updated to P_i^{t+1}. Then, in the new frame (**Fig. 13(c)**), we can predict the new position as follows:

$$P_i^{t+2} = P_i^{t+1} + \left(P_i^{t+1} - P_i^t\right) \quad \ldots\ldots\ldots \quad (1)$$

In this manner, we can avoid the marker moving outside the ROI box in a high-speed rotation case.

3.3.3. ROI Tracking Demonstration

Figure 14 shows the ROI tracking results in the local mode. As the object moves away from the camera, the box size (yellow box) and center are adjusted accordingly.



（a）

(b)

(c)

(d)

Fig. 14. Demonstration of ROI tracking.

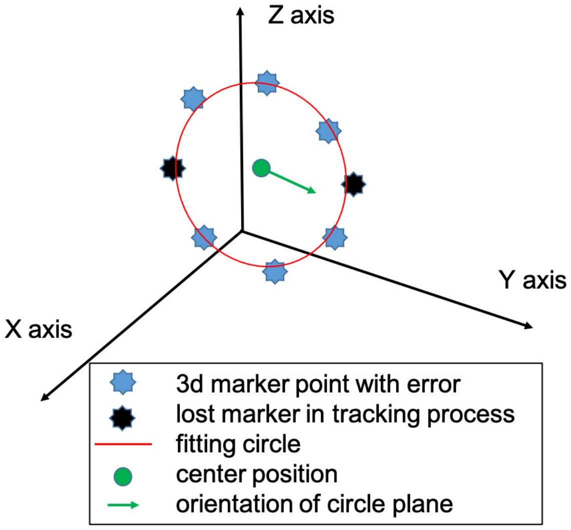

Z axis
Y axis
X axis
3d marker point with error
lost marker in tracking process
fitting circle
center position
orientation of circle plane

Fig. 15. Fitting a circle for the markers.

3.4. Pose Estimation

As shown in **Fig. 15**, we can fit a circle along the tracked markers. Even when some of the markers (such as the black markers) are lost, we can still fit the circle using the remaining markers (blue markers) and estimate the circular center and orientation.

To fit the circle, all of the scatter points should be aligned along a plane. The plane function is as follows:

$$a_1x + a_2y + a_3z - 1 = 0 \quad \ldots\ldots\ldots\ldots \quad (2)$$

In matrix format, the above plane function for n coplanar points is as follows:

$$M = \begin{bmatrix} x_1 & x_2 & \ldots & x_n \\ y_1 & y_2 & \ldots & y_n \\ z_1 & z_2 & \ldots & z_n \end{bmatrix}^T \quad \ldots\ldots\ldots \quad (3)$$

$$\mathbf{a} = (a_1, a_2, a_3)^T, \quad \mathbf{h}_1 = (1, 1, \ldots, 1)^T \quad \ldots \quad (4)$$

According to the least-squares method, we can obtain the normal vector of the plane as follows:

$$\mathbf{a} = \left(M^T M\right)^{-1} M^T \mathbf{h}_1 \quad \text{. . . . .} \quad (5)$$

Let us assume that the center of the circle is $\mathbf{c}$ (x_0, y_0, z_0), and any point on the circle is $\mathbf{p}_i$ (x_i, y_i, z_i). Based on the chord mid-perpendicular theorem, we obtain an equation for the n points on the circle as follows:

$$B = \begin{bmatrix} \Delta x_{12} & \Delta y_{12} & \Delta z_{12} \\ \Delta x_{13} & \Delta y_{13} & \Delta z_{13} \\ \vdots & \vdots & \vdots \\ \Delta x_{1n} & \Delta y_{1n} & \Delta z_{1n} \\ \Delta x_{23} & \Delta y_{23} & \Delta z_{23} \\ \Delta x_{24} & \Delta y_{24} & \Delta z_{24} \\ \vdots & \vdots & \vdots \\ \Delta x_{2n} & \Delta y_{2n} & \Delta z_{2n} \\ \vdots & \vdots & \vdots \\ \Delta x_{(n-1)n} & \Delta y_{(n-1)n} & \Delta z_{(n-1)n} \end{bmatrix} \quad \text{. . .} \quad (6)$$

$$\mathbf{h}_2 = \begin{bmatrix} l_{12} \\ l_{13} \\ \vdots \\ l_{(n-1)n} \end{bmatrix} \quad \text{. . . . .} \quad (7)$$

$$B\mathbf{c}^T = \mathbf{h}_2 \quad \text{. . . . .} \quad (8)$$

In the above, $\Delta x_{ij} = x_j - x_i$, $\Delta y_{ij} = y_j - y_i$, $\Delta z_{ij} = z_j - z_i$, $l_{ij} = (x_j^2 + y_j^2 + z_j^2 - x_i^2 - y_i^2 - z_i^2)/2$.

Equation (8) can be solved by constructing an optimization problem under the constraints $\mathbf{a} \cdot \mathbf{c}^T = 1$, as follows:

$$f(\mathbf{c}) = \left\| B\mathbf{c}^T - \mathbf{h}_2 \right\|^2 + \lambda \left(\mathbf{a} \cdot \mathbf{c}^T - 1\right) \quad \text{. . . .} \quad (9)$$

Here, λ denotes the Lagrangian multiplier. For the derivative of $f(\mathbf{c})$ with respect to $\mathbf{c}$ and λ, we set the derivative value to be 0, and the transformation yields as follows:

$$\begin{bmatrix} B^T B & \mathbf{a} \\ \mathbf{a}^T & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{c}^T \\ \lambda \end{bmatrix} = \begin{bmatrix} B^T \mathbf{h}_2 \\ 1 \end{bmatrix} \quad \text{. . . . .} \quad (10)$$

Finally the circle center position $\mathbf{c}$ is calculated as follows:

$$\begin{bmatrix} \mathbf{c}^T \\ \lambda \end{bmatrix} = \begin{bmatrix} B^T B & \mathbf{a} \\ \mathbf{a}^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} B^T \mathbf{h}_2 \\ 1 \end{bmatrix} \quad \text{. . . .} \quad (11)$$

4. Evaluation of Tracking and Pose Estimation

To evaluate the pose estimation accuracy of this high-speed vision system, we used the data from the OptiTrack system as the ground truth. This data have a position error of less than 0.2 mm and rotational error of less than 0.1°. The system setup was introduced in Section 2, and the target objects were circular plates with markers attached as shown in the pattern in **Fig. 3**. We moved the target objects inside the co-visible space between the proposed high-speed vision system and OptiTrack system.

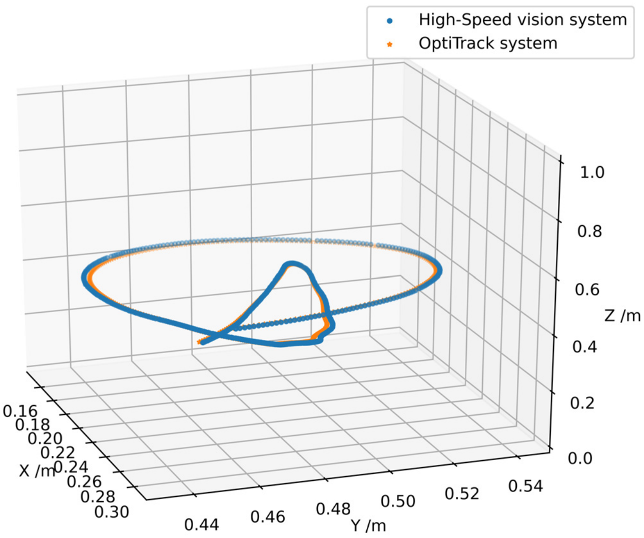


Fig. 16. 3D position measurement of high-speed vision system and OptiTrack system (marker number = 6).

4.1. Static Accuracy

First, we evaluated the static accuracy of the high-speed vision system using the OptiTrack system as the ground truth. To measure the static position at different locations, we fixed the object on the camera tripod and adjusted the height and location of the tripod. For the target with four markers, the average difference from the OptiTrack result was 0.5 cm, with the largest error being less than 1 cm. Similarly, we determined the average orientation difference as 0.57°, with the largest error being less than 2.1°.

4.2. Dynamic Accuracy

4.2.1. Dynamic Accuracy of Position Measurement

We defined the 3D position error as the distance between the calculated position using the proposed system and that measured using OptiTrack. Both the high-speed vision system and the OptiTrack system tracked the same reflective markers. **Fig. 16** shows the measured position data from the high-speed vision and OptiTrack systems for an object with six markers.

Figures 17 and **18** show the 3D positions and 3D position errors in the time series, respectively. The 3D position error varies at different timestamps because the target object has different poses and velocities. In the measured trajectory, we always change the distance from the camera and motion velocity. The farther the target object is from the vision system, the larger the 3D position error. The faster the target object moves, the larger the 3D position error.

The average dynamic error for the position measurement δp_{avg} with four markers is 4.4 mm with the largest error δp_{max} of approximately 7.4 mm. For an object with six markers, δp_{avg} is 3.1 mm and δp_{max} is 5.7 mm.

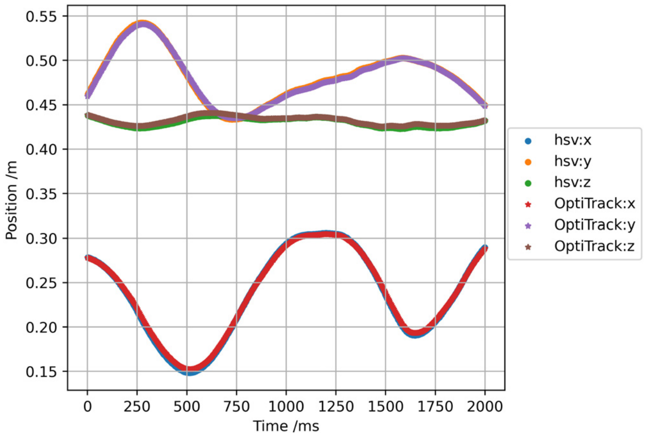


Fig. 17. 3D position measurement of high-speed vision system and OptiTrack system in time series (marker number = 6). The "hsv" denotes the proposed high-speed vision system and "OptiTrack" demotes the motion capture system. "*x*," "*y*," and "*z*" are the three-axis components of the 3D position, respectively.

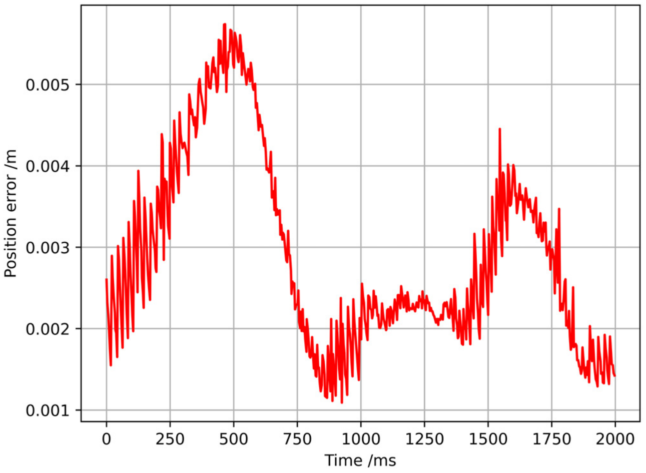


Fig. 18. 3D position error of high-speed vision system in time series (marker number = 6).

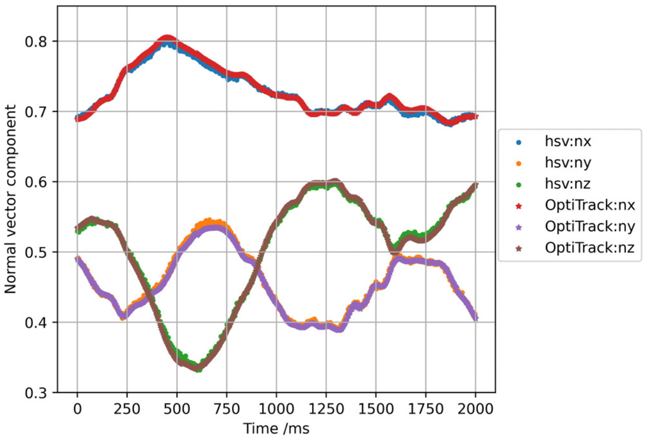


Fig. 19. 3D orientation measurement of high-speed vision system and OptiTrack system in time series (marker number = 6). The "hsv" denotes the proposed high-speed vision system and "OptiTrack" denotes the motion capture system. "*nx*," "*ny*," and "*nz*" are the three-axis components of the 3D orientation vector, respectively.

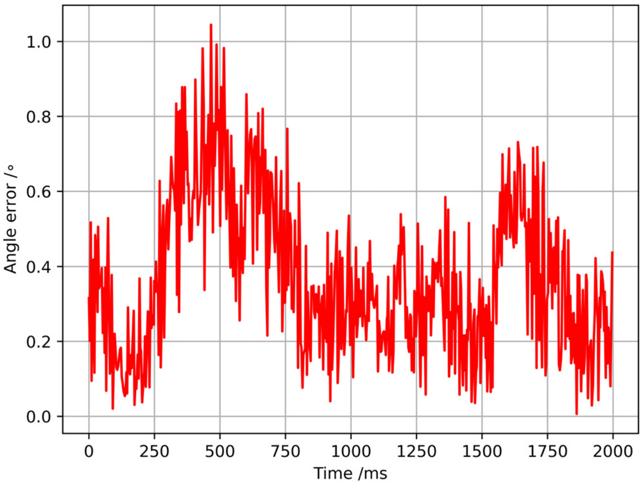


Fig. 20. 3D orientation error of high-speed vision system in time series (marker number = 6).

4.2.2. Dynamic Accuracy of Orientation Measurement

We defined the 3D orientation error as the angle between the calculated orientation using the proposed system and the orientation measured using OptiTrack.

Figures 19 and **20** show the orientation data and 3D orientation error in the time series for an object with six markers, respectively.

The average dynamic error for the orientation measurement $\delta\theta_{avg}$ with four markers is 0.60°, with the largest error $\delta\theta_{max}$ of approximately 2.26°. For an object with six markers, $\delta\theta_{avg}$ is 0.36° and $\delta\theta_{max}$ is 1.04°.

Table 1 lists the frequency and pose estimation errors for different numbers of markers.

All of the results shown in **Table 1** are achieved from a 1280 × 1024 resolution image input with down sampling to a 640 × 512 resolution. The high speed vision system can obtain a processing frequency larger than 500 Hz. By reducing the field of view of the camera to the ROI, we can improve the processing speed to approximately 1000 Hz when using three markers for 640 × 400 resolution image input without down sampling. The accuracy is the same as that shown in **Table 1**. Here, the frequency is the number of processed frames per second. The success frequency represents the frames under tracking in which at least three pairs of markers are successfully detected.

From **Table 1**, it can be seen that the dynamic pose estimation error does not deviate significantly from the static error, indicating that the high-speed vision system can stably acquire an accurate marker position during rotation and translation. The accuracy of the pose estimation increases with the number of markers because the circular fitting accuracy depends on the number of fitting points. Moreover, the rate of successful frames (suc-

Table 1. Evaluation result of pose estimation (static and dynamic).

Marker number	Frequency [Hz]	Success frequency [Hz]	Static position error [mm]	Static orientation error [°]	Dynamic position error [mm]	Dynamic orientation error [°]
3	761	744	7.0	0.75	6.5	0.78
4	701	697	5.0	0.57	4.4	0.60
5	629	623	3.7	0.39	3.8	0.47
6	585	579	2.9	0.32	3.1	0.36
8	562	559	2.5	0.25	2.7	0.29

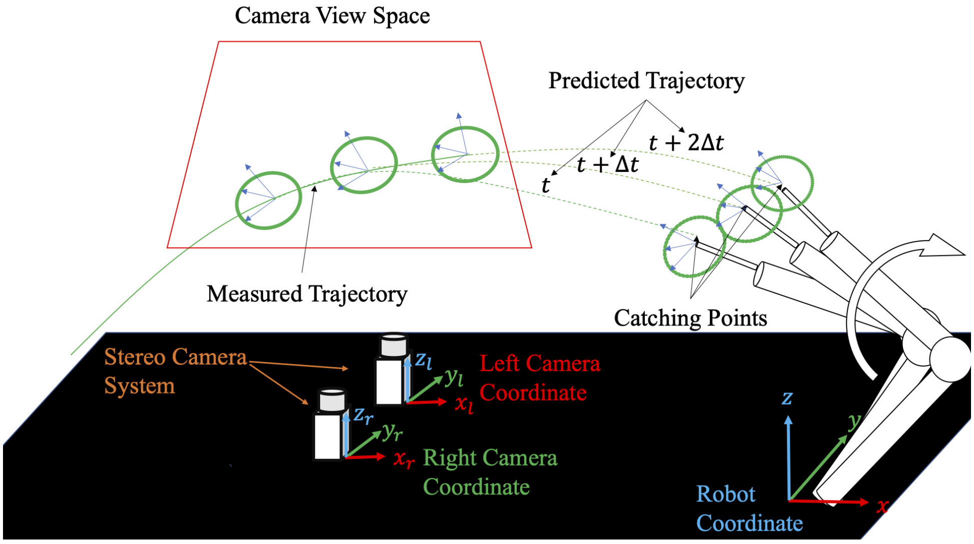


Fig. 21. System setup for robot catching experiment.

cess frequency / frequency) increases with the number of markers. The existing method [2] uses three markers to localize a target object. When one of the markers is lost during tracking, pose estimation cannot continue. As shown in **Table 1**, the rate of successful frames (success frequency / frequency) increases with marker number. With over three markers, the circular fitting method and marker matching can make the tracking more robust to occlusion, leading to a higher rate of successful frames.

5. Robot Catching Experiment

For a robot, the catching of in-flight objects is a challenging task, requiring a high-frequency sequence of pose estimations, trajectory predictions, catching point determinations, and motion planning. Considering the small working space and visual occlusion in the natural environment, robot catching was investigated in this study using a short-distance and partially observable trajectory. The proposed marker-based high-speed visual tracking method was used to collect sufficient data from the limited trajectory. Based on the proposed method and a dynamics analysis of a thrown ring, we obtained a success rate of 90% in experiments for catching in-flight rings using a collaborative robot arm [15]. For the 10% of cases ending in failure, one reason is the prediction error of the catching pose. This means that the predicted catching position and orientation have a non-negligible deviation from the real flying trajectory. As another possible reason, the robot arm cannot reach the target pose in a timely manner owing to a joint velocity limitation; this indicates that the planned trajectory for the robot arm is not the optimum.

As shown in **Fig. 21**, the target object was a ring with three markers wrapped at the trisection points. We chose three markers for the highest perception frequency so that the trajectory prediction could be established as quickly as possible. A high-speed camera system with two cameras was set up. The robot arm (UR5e) with a stick end-effector was 1.5 m away from the cameras and 2.5 m away from the starting point of the ring's trajectory. **Fig. 22** shows the second half of the catching process (also the second half of the in-flight trajectory), where the robot arm moves a short distance to catch the rotating ring.

We conducted several test experiments to continuously catch rings with diameters of 15, 13, 10, and 6 cm. Our method had a success rate of 90%, and boosted the ability of the industrial robot to perform high-speed manipulations. A video of the experimental results is available on our website [c].

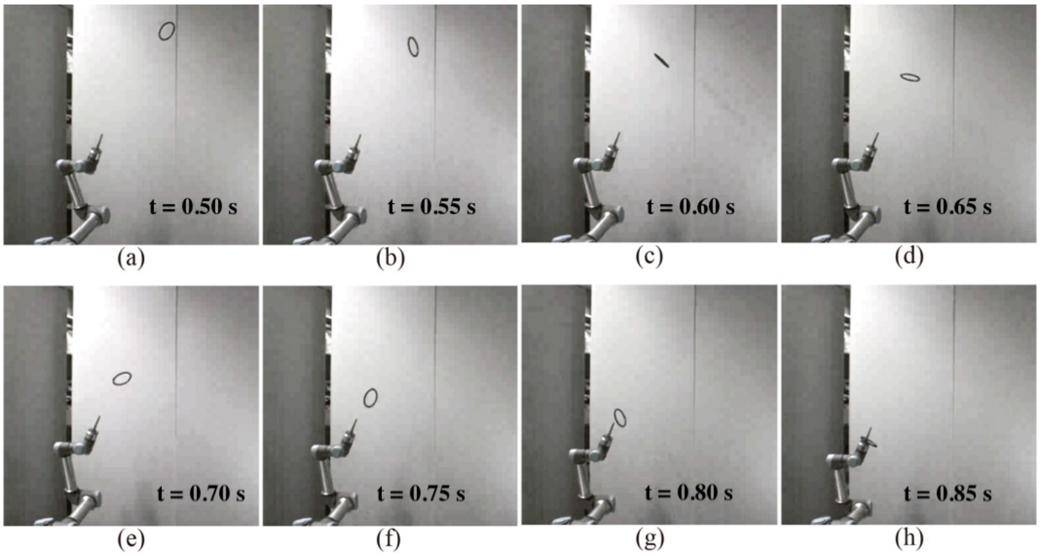


Fig. 22. States of the robot arm in the second half of a catching process.

6. Conclusion

In this study, we introduced a marker-based tracking and pose-estimation method using a high-speed vision system. The marker tracking has two modes: global and local. To avoid an initial marker matching requiring manual work, we designed a stereo marker-matching algorithm to match the marker pairs in left and right views. When the markers are temporarily lost in the camera view and appear again, the marker-matching method can provide a quick response to start a new round of tracking. To increase the tracking stability and velocity, we applied an adaptive ROI tracking method to dynamically adjust the center position and box size of the local window. Based on the detected 2D pixel positions of the markers, we used a circular plane fitting algorithm to find the circle center as the target position, and obtained the plane orientation as the target orientation. The algorithm can achieve more than 500 Hz in tracking and pose estimation, making it powerful for tracking high-speed rotating objects and robust to partial occlusion problems. We also applied this tracking system to a robot catching experiment and verified its stability and accuracy through a high success rate. The tracking system is a low-cost and easy setup and can be easily attached to a robot as an on-body vision system. Further work will be conducted to install such a system on a robot arm and to perform manipulation tasks.

References:

[1] H. Kim, Y. Yamakawa, T. Senoo, and M. Ishikawa, "Robotic manipulation of rotating object via twisted thread using high-speed visual sensing and feedback," 2015 IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI), pp. 265-270, 2015.

[2] M. Yashima and T. Yamawaki, "Robotic nonprehensile catching: Initial experiments," 2014 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 2461-2467, 2014.

[3] T. Murooka, K. Okada, and M. Inaba, "Diabolo orientation stabilization by learning predictive model for unstable unknown-dynamics juggling manipulation," 2020 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 9174-9181, 2020.

[4] B. Tekin, S. N. Sinha, and P. Fua, "Real-Time Seamless Single Shot 6D Object Pose Prediction," 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2018.

[5] S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao, "PVNet: Pixel-wise voting network for 6dof pose estimation," 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2019.

[6] X. Zhu and K. F. Li, "Real-time motion capture: An overview," 2016 10th Int. Conf. on Complex, Intelligent, and Software Intensive Systems (CISIS), pp. 522-525, 2016.

[7] H. Sunada, K. Yokoyama, T. Hirata, T. Matsukawa, and H. Takada, "Visualization system for the information measured by motion capture system," 2016 11th Int. Conf. on Computer Science Education (ICCSE), pp. 21-26, 2016.

[8] R. Sakata, F. Kobayashi, and H. Nakamoto, "Development of motion capture system using multiple depth sensors," 2017 Int. Symposium on Micro-NanoMechatronics and Human Science (MHS), pp. 1-7, 2017.

[9] T. Senoo, Y. Yamakawa, S. Huang, K. Koyama, M. Shimojo, Y. Watanabe, L. Miyashita, M. Hirano, T. Sueishi, and M. Ishikawa, "Dynamic intelligent systems based on high-speed vision," J. Robot. Mechatron., Vol.31, No.1, pp. 45-56, 2019.

[10] S. Hu, M. Jiang, T. Takaki, and I. Ishii, "Real-time monocular three-dimensional motion tracking using a multithread active vision system," J. Robot. Mechatron., Vol.30, No.3, pp. 453-466, 2018.

[11] M. Ishikawa, A. Namiki, T. Senoo, and Y. Yamakawa, "Ultra high-speed robot based on 1 khz vision system," 2012 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 5460-5461, 2012.

[12] Y. Yamakawa, A. Namiki, and M. Ishikawa, "Motion planning for dynamic folding of a cloth with two high-speed robot hands and two high-speed sliders," 2011 IEEE Int. Conf. on Robotics and Automation, pp. 5486-5491, 2011.

[13] Y. Yamakawa, Y. Matsui, and M. Ishikawa, "Development and analysis of a high-speed human-robot collaborative system and its application," 2018 IEEE Int. Conf. on Robotics and Biomimetics (ROBIO), pp. 2415-2420, 2018.

[14] Y. Nakabo, I. Ishi, and M. Ishikawa, "3d tracking using two high-speed vision systems," IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Vol.1, pp. 360-365, 2002.

[15] X. Liang, H. Zhu, Y. Chen, and Y. Yamakawa, "Tracking and catching of an in-flight ring using a high-speed vision system and a robot arm," 47th Annual Conf. of the IEEE Industrial Electronics Society (IECON 2021), pp. 1-7, 2021.

[16] S. Kagami, "Utilizing opencv for high-speed vision processing," J. of the Robotics Society of Japan, Vol.31, No.3, pp. 244-248, 2013.

Supporting Online Materials:

[a] Motion Capture for Robotics, Jun. 2021. https://optitrack.com/applications/robotics [Accessed June 29, 2021]

[b] Xsens, Home – Xsens 3D motion tracking, Aug. 2021. https://www.xsens.com/ [Accessed August 6, 2021]

[c] Yamakawa Laboratory – High-speed Flexible Robotics, Dec. 2020. http://www.hfr.iis.u-tokyo.ac.jp/research/RingCatching/index-e.html [Accessed July 15, 2021]

Name:
Xiao Liang

Affiliation:
Graduated Student, Department of Mechanical Engineering, The University of Tokyo

Address:
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan

Brief Biographical History:

2019 Received Bachelor degree (Mechanical Engineering) from Shanghai Jiao Tong University

2021 Received Master degree (Mechanical Engineering) from The University of Tokyo

Main Works:

- X. Liang, H. Zhu, Y. Chen, and Y. Yamakawa, "Tracking and catching of an in-flight ring using a high-speed vision system and a robot arm," 47th Annual Conf. of the IEEE Industrial Electronics Society (IECON 2021), pp. 1-7, 2021.
- S. Quan, X. Liang, H. Zhu, M. Hirano, and Y. Yamakawa, "HiVTac: A High-Speed Vision-Based Tactile Sensor for Precise and Real-Time Force Reconstruction with Fewer Markers," Sensors, Vol.22, Issue 11, Article No.4196, 2022.

Name:
Masahiro Hirano

Affiliation:
Assistant Professor, Institute of Industrial Science, The University of Tokyo

Address:
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan

Brief Biographical History:

2018 Received Ph.D. (Information Science and Technology) from The University of Tokyo

2018- Project Assistant Professor, The University of Tokyo

2020- Assistant Professor, The University of Tokyo

Main Works:

- M. Hirano, Y. Watanabe, and M. Ishikawa, "Rapid blending of closed curves based on curvature flow," Computer Aided Geometric Design, Vol.52-53, pp. 217-230, 2017.
- M. Hirano, Y. Yamakawa, T. Senoo, N. Kishi, and M. Ishikawa, "Multiple Scale Aggregation with Patch Multiplexing for High-speed Inter-vehicle Distance Estimation," IEEE Intelligent Vehicles Symposium, pp. 1436-1443, 2021.
- M. Hirano, Y. Yamakawa, T. Senoo, and M. Ishikawa, "An acceleration method for correlation-based high-speed object tracking," Measurement: Sensors, Vol.18, Article No.100258, 2021.

Membership in Academic Societies:

- Institute of Electrical and Electronics Engineers (IEEE)
- Information Processing Society of Japan (IPSJ)
- The Society of Instrument and Control Engineers (SICE)

Name:
Yuji Yamakawa

Affiliation:
Associate Professor, Interfaculty Initiative in Information Studies, The University of Tokyo

Address:
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan

Brief Biographical History:

2011 Received Ph.D. (Information Science and Technology) from The University of Tokyo

2011- Project Assistant Professor, The University of Tokyo

2014- Assistant Professor, The University of Tokyo

2017- Lecturer, The University of Tokyo

2020- Associate Professor, The University of Tokyo

Main Works:

- Y. Yamakawa, Y. Matsui, and M. Ishikawa, "Development of a Real-Time Human-Robot Collaborative System Based on 1 kHz Visual Feedback Control and Its Application to a Peg-in-Hole Task," Sensors, Vol.21, Issue 2, Article No.663, 2021.
- T. Wang and Y. Yamakawa, "Real-Time Occlusion-Robust Deformable Linear Object Tracking With Model-Based Gaussian Mixture Model," Frontiers in Neurorobotics, Vol.16, Issue ISS, Article No.886068, 2022.
- Y. Yamakawa and K. Yoshida, "Teleoperation of High-Speed Robot Hand with High-Speed Finger Position Recognition and High-Accuracy Grasp Type Estimation," Sensors, Vol.22, Issue 10, Article No.3777, 2022.
- S. Quan, X. Liang, H. Zhu, M. Hirano, and Y. Yamakawa, "HiVTac: A High-Speed Vision-Based Tactile Sensor for Precise and Real-Time Force Reconstruction with Fewer Markers," Sensors, Vol.22, Issue 11, Article No.4196, 2022.

Membership in Academic Societies:

- The Japan Society of Mechanical Engineers (JSME)
- The Robotics Society of Japan (RSJ)
- The Society of Instrument and Control Engineers (SICE)