



## A Development of Visual-feedback Automatic Control for Robotic Manipulator

Hiroshi Nakahara<sup>1</sup> and Kittikhun Thongpull<sup>2</sup>

### ABSTRACT

In this work, an automatic robotic manipulator control based on visual information for object tracking is presented. The tracking process employs Recurrent Neural Network and Kalman filter techniques to predict the location of the target object which is used to calculate the movement of the robot. A manipulator control process was developed to provide consistent motion and reduce vibration. We also propose an object analysis based on an image processing technique for object size determination. We implemented the proposed system with an actual robotic manipulator with a gripper for experiments. The experiments have shown that object size estimation method results in the maximum accuracy at 98.87%. The proposed object tracking and manipulator control process achieved vibration reduction 3.05 times compared to the simple control method and reduced the time of end-effector reaching the target object by 1.29 sec.

### Article information:

**Keywords:** Automatic Robotic Manipulator, Kalman Filter, Object Tracking, Object Analysis

### Article history:

Received: April 22, 2021

Revised: September 11, 2021

Accepted: December 6, 2021

Published: March 31, 2022

(Online)

**DOI:** 10.37936/ecti-cit.2022161.246998

### 1. INTRODUCTION

Robotic manipulators are becoming a major component in a broad range of applications such as industrial manufacturing, medical, and home use [2]. One of the major purposes of robotic manipulators is to pick up an object by using a gripping tool as an end-effector and move the object from one location to another which is a complicated process due to the multiple dimensions to be controlled consistently and the number of movement joints of the robot. This task poses the difficulty to an operator in case of manual or semi-auto control operations while automatic control can be implemented conventionally using closed loop position control of manipulator [1] with stability improvement whereas visual information can be used to guide the movement of manipulator without the knowledge of manipulator joint position [2,9]. The closed-loop control approach was presented in [3-4] with image-based visual servo control. To realize the visual based manipulator control, the object identification and tracking process are important. CNN was employed for object detection [11] whereas RNN exhibits superior performance for object tracking [12].

In this work we present an automatic robotic manipulator controller method based on a computer vision technique with Recurrent Neural Network based

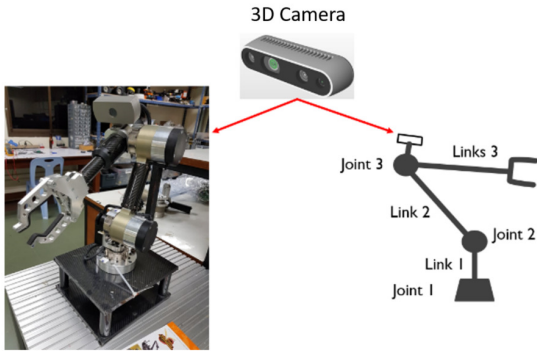
object detection and tracking. The proposed tracking process consists of an image tracking task and a manipulator joint control task. The proposed method has been implemented in an actual robotic manipulator with a 3D-camera installed at the end-effector of the robot. A Recurrent Neural Network based object tracking technique was employed for tracking process. In addition, a Kalman filter is used to enhance the image tracking process. We also present the manipulator control process that uses the information from the tracking process to adjust the movement speed to help reduce the vibration and provide consistent motions. Using the proposed method, the vibration was suppressed 3.05 times compared to simple control method. The Kalman filter reduced the time of end-effector reaching the target object by 1.29 sec in the experimental cases. Moreover, we present an object analysis method based on image processing technique to estimate the size of the object. The experiments show that our object size estimation method results in the highest accuracy at 98.87% and the tracking process can effectively keep the target object in the gripping position.

<sup>1,2</sup> The author are with Department of Electrical Engineering Faculty of Engineering, Prince of Songkla University, Thailand, E-mail: flexcode.fox@aol.com and kittikhun.t@psu.ac.th

<sup>2</sup> The corresponding author: kittikhun.t@psu.ac.th

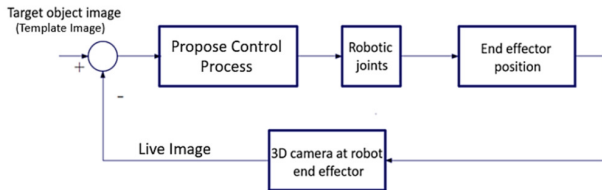
## 2. VISUAL FEEDBACK MANIPULATOR CONTROL

The aim of this work is to develop a closed-loop robotic manipulator control system based solely on visual information. We focus on robotic manipulators without position sensors. A camera is mounted on the robot as shown in **Figure 1**. It provides visual information comprised of a 2D frame from CMOS sensor and a depth frame from Time-of-Flight (ToF) sensor. The main process is to keep the target object image in the center of the camera frame which is the position of the end-effector of the robotic manipulator by moving its joints and moving the end effector to the target object afterward to perform a gripping task.



**Fig.1:** Robotic manipulator and camera setup.

The target image information, such as the cropping box location on the camera frame, is given to the manipulator position control process as a set point. Thus, the controller commands all of its joints based on the deviation of target image information from the center of the camera frame. **Figure 2** illustrates the overall structure of the proposed visual feedback robotic manipulator control system.



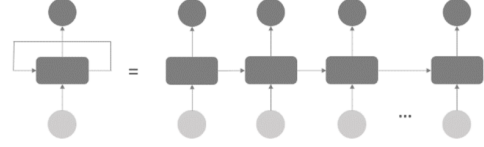
**Fig.2:** Structure of the proposed.

### 2.1 Object Tracking by Recurrent Neural Network

The vital information of the proposed control process is the location of the target object in the camera frame. An object tracking technique based on a Neural network [5-8] is used to provide the coordination with the target object also keep tracking accurate when the object moves. We employ Recurrent

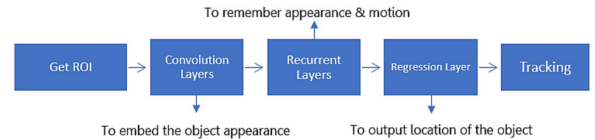
Neural Network (RNN) due to its lower computational complexity compared to CNN [13-14].

RNN performs its update process with a sequential scheme [11-12] as shown in **Figure 3**, providing the ability to cope with fluctuating information which is common in images with moving objects as well as the camera position.



**Fig.3:** General Recurrent Neural Network.

There are multiple layers constructed in a Recurrent Neural Network prediction procedure as illustrated in **Figure 4**. The Region of Interest (ROI) is used to specify the area of the target object image in the camera frame as raw input information for the prediction process. The convolutional layers perform feature extraction tasks to obtain meaningful information from the raw input [10]. The motion behavior and characteristics are learned and will be used to predict upcoming motions. At the regression layer, the prediction results from the recurrent layer are interpreted to find the object in the camera frame. However, RNN tracking performance with uncertainty tends to miss the target and cannot recover due to its sequential update mechanism. Therefore, we employ a Kalman Filter as additional processing to tackle the issue.



**Fig.4:** Recurrent Neural Network for object tracking.

### 2.2 Improved Tracking by Kalman Filter

A Kalman Filter is a mathematical algorithm for state estimation based on a linear filtering prediction model [19]. Kalman Filter is one of the most common approaches for estimating linear states that are assumed to have a Gaussian distribution [15, 17].

A state  $x$  at  $k$  can be estimated by the true state model described in eq.1 [16, 18, 20-21] learned from previous state characteristic  $k - 1$  :

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (1)$$

$$z_k = Hx_k + v_k \quad (2)$$

where:

$A$  is the state transition of the prior state  $k-1$  to the current state  $k$

$B$  is control-input model, matrix applied to the optional control input  $u_{k-1}$

$H$  is transformation matrix that transforms the state into the measurement domain  $w_k$  and  $v_k$  are Zero-mean Gaussian white noise.

State transition  $A$  and control-input model  $B$  can be determined using Kinematic equations. The relation between the position  $x$  and velocity  $\dot{x}$  can be written as follows:

$$\text{Position: } x_k = x_{k-1} + \dot{x}_{k-1}\Delta t + \frac{1}{2}\ddot{x}_{k-1}(\Delta t)^2 \quad (3)$$

$$\text{Velocity: } \dot{x}_k = \dot{x}_{k-1} + \ddot{x}_{k-1}\Delta t \quad (4)$$

where:  $\ddot{x}$  is the acceleration.

From Equations (3) and (4), a state vector model can be derived as:

$$x_k = \begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix} = \begin{bmatrix} x_{k-1} + \dot{x}_{k-1}\Delta t + \frac{1}{2}\ddot{x}_{k-1}\Delta t^2 \\ \dot{x}_{k-1} + \ddot{x}_{k-1}\Delta t \end{bmatrix} \quad (5)$$

Equation (5) can be written in matrix multiplication form as follows:

$$x_k = \begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ \dot{x}_{k-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}(\Delta t)^2 \\ \Delta t \end{bmatrix} \ddot{x}_{k-1} \quad (6)$$

Since our application is in 2-dimensional plane, we reformulate (6) by applying a new vector  $x_k$ .

$$x_k = \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \dot{x}_{k-1} \\ \dot{y}_{k-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}(\Delta t)^2 & 0 \\ 0 & \frac{1}{2}(\Delta t)^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} \ddot{x}_{k-1} \\ \ddot{y}_{k-1} \end{bmatrix} \quad (7)$$

We can simplify (7) as:

$$x_k = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} \frac{1}{2}(\Delta t)^2 & 0 \\ 0 & \frac{1}{2}(\Delta t)^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} a_{k-1} \quad (8)$$

From (8) and (1), we can conclude the state transition matrix  $A$  is as mentioned in (9) and the control-input matrix  $B$  is as mentioned in (10).

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

$$B = \begin{bmatrix} \frac{1}{2}(\Delta t)^2 & 0 \\ 0 & \frac{1}{2}(\Delta t)^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \quad (10)$$

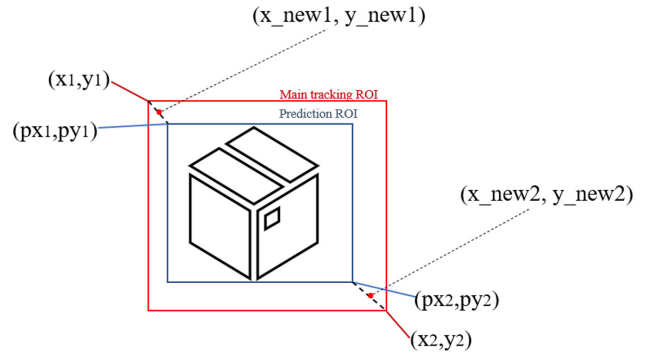
Thus, the measurement model is fulfilled as:

$$z_k = H \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} + v_k, \quad (11)$$

And the transformation matrix  $H$  is  $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$

The object tracking process starts with steady object tracking assuming no movement after the main ROI is given by the user.

After the main ROI is tracked by the RNN, the Kalman tracking process performs simultaneously, outputting a Prediction ROI for every frame. The prediction ROI tends to be smaller than the main ROI (typically 5 - 10%) which reduces unwanted area as illustrated in **Figure 5**. The coordinates are different:  $x\_new$  and  $y\_new$  are given to the main ROI tracking process to improve tracking performance.



**Fig.5:** ROI from RNN tracking and Kalman filter.

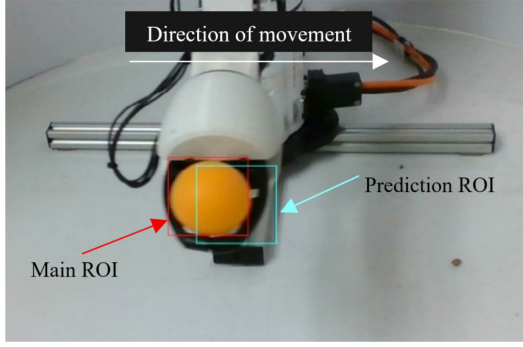
where:

$(x, y)$  is coordinate of Main tracking ROI  
 $(px, py)$  is coordinate of Prediction ROI

$$x\_new = |x - px|$$

$$y\_new = |y - py|$$

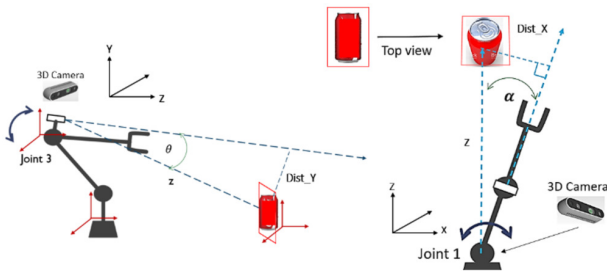
**Figure 6** illustrates an example of the moving object tracking results of Main tracking ROI from RNN and Prediction ROI from Kalman filter showing the consistency of predicted ROI to the actual object trajectory.



**Fig.6:** Prediction ROI 3 frame/times.

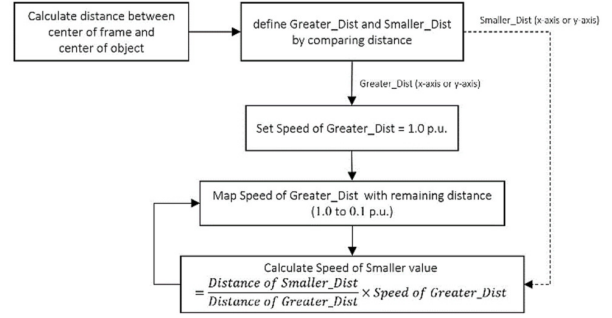
### 2.3 Manipulator Control Process

The main aim of this work is to control the manipulator to approach a target object to act, for example grasping, loading or unloading tools. Thus, the manipulator is attached to the camera where the center of the image frame is set to align with the end effector tip. In this case, the tip is the manipulator gripper. The manipulator is controlled by a process to keep the center of the ROI box from the image tracking process at the center of the image frame. As the ROI box is in the 2-dimensional plane, two parameters, the distance between the center of the ROI box and the center of the image frame in x and y-axis as shown in **Figure 7**, are tracked by the manipulator control process.



**Fig.7:** Axial movement characteristics ( $x$ ,  $y$ ).

The first step of the control process is to determine the greater and smaller distance values used to regulate the manipulator joint movement speed. The joint speed regulation procedure is illustrated in **Figure 8**. The axis with greater distance is assigned with the maximum movement speed (1.0 pu of rated speed). The other axis is moved with the speed scaled by the distance ratio of both axes. The speed of both axes are then updated every new frame as the result the speeds are decreased while the end effector approaches the object.



**Fig.8:** Manipulator control process procedure.

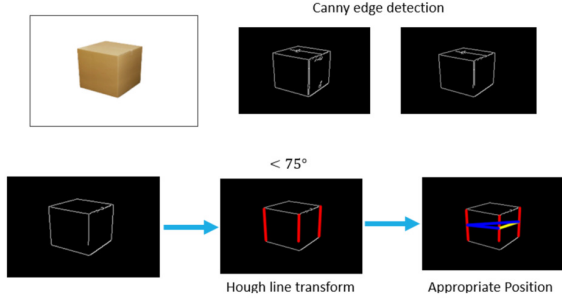
**Figure 9** illustrates the control technique of the gripper without a force sensor which can be performed by attaching an AR marker to measure the opening and closing size of the gripper.



**Fig.9:** Control techniques using the AR marker.

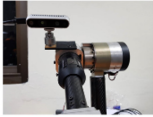
### 2.4 Object Shape Determination

In order to perform the correct action of grasping, object shape information is needed. One of the vital object parameters is the object size which is important for correct grasping action and to avoid a collision of the end-effector and the object. In this work, Hough's transformation-based line detection method is used in the proposed object size estimation process as shown in **Figure 10**. The Canny edge detection algorithm [22] is used to transform the original image into a binary type image which is required for Hough's transformation process [23]. The lines detected from Hough's transformation are then selected by the allowance angle to accept only vertical lines of the object shape. The angle was experimentally set at  $75^\circ$ .



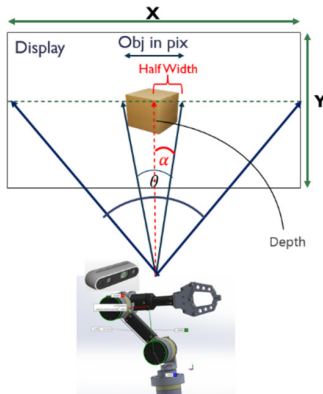
**Fig.10:** Object shape line detection process.

The selected lines are then used to calculate the object width based on trigonometry calculation as explained in **Figures 11-12**. The important information for the calculation is the distance between the end-effector and object. We employ a 3D camera that provides depth information that can be used in the calculation.



Format	D400/D410/D415	D420/D430/D435
Horizontal FOV (HD 16:9) (HFOV)	64	86
Vertical FOV (HD 16:9) (VFOV)	41	57
Diagonal FOV (HD 16:9) (DFOV)	72	94

**Fig.11:** Camera specs.



**Fig.12:** Object width estimation algorithm.

$$\frac{HOV}{X} = \text{Degree per Pix} \quad (12)$$

$$\theta = (X - \text{Obj in pix}) \times \text{Degree per Pix} \quad (13)$$

$$\alpha = \frac{\theta}{2} \quad (14)$$

$$\text{Half Width} = (\tan \alpha) \times \text{Depth} \quad (15)$$

$$\text{Width} = \text{Half Width} \times 2 \quad (16)$$

(In symmetrical object case)

### 3. EXPERIMENTS AND DISCUSSION

We conducted experiments to evaluate the performance of the proposed system in both object tracking and Manipulator Control aspects. In addition, an algorithm which we implemented for object size determination is presented.

#### 3.1 Manipulator Control Process

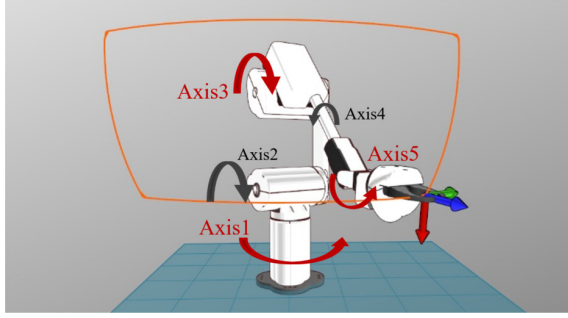
In this experiment, we used the manipulator control process presented in 2.3 to observed the performance of the manipulator movement approaching a target object. Commonplaces Robotics Mover6, as illustrated in **Figure 13, 15**, was used to move the target object in a set of defined paths listed in **Table 1**. The Mover6 was placed 70cm. in front of the tracking manipulator as illustrated in **Figure 14**. In this experiment, only main tracking (RNN) was performed as we focused mainly on the manipulator



**Fig.13:** Mover6 with the sample object.



**Fig.14:** Manipulator control process experiment setup.

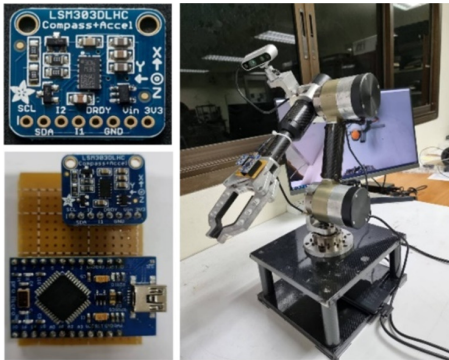


**Fig.15:** Mover6 and the defined trajectories.

**Table 1:** Mover6's joint angles for each test position.

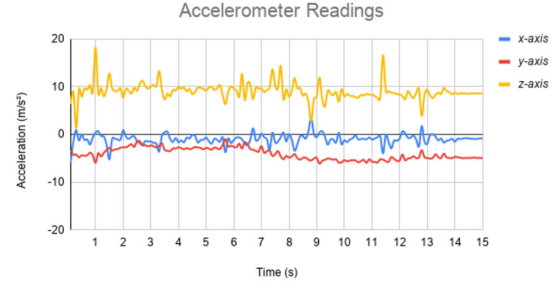
Position	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
1	30	0	50	0	-45
2	30	0	10	0	-15
3	-30	0	10	0	-15
4	-30	0	50	0	-45
5	30	0	50	0	-45

A 6-DoF IMU depicted in **Figure 16** was employed at the end-effector to observe the movement characteristics.



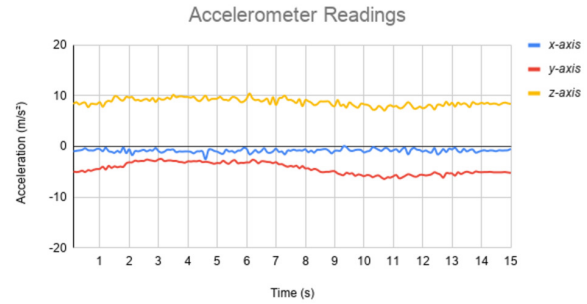
**Fig.16:** The IMU installed at the end-effector.

**Figure 17** shows the acceleration readings at the end-effector of the manipulator while the manipulator moved to follow the object. In this case, the manipulator was controlled by a simple controller. There were strong vibrations which occurred along the path which may have affected the image tracking performance due to the image shaking because the camera was attached to the manipulator.



**Fig.17:** End-effector acceleration readings of the manipulator using simple controller.

Applying the proposed manipulator control method with a similar trajectory, the acceleration shown in **Figure 18** appears to be smoother. The vibrations were reduced resulting in a more stable image while the manipulator was moving.



**Fig.18:** End-effector acceleration readings of the manipulator using proposed manipulator control method.

**Table 2** describes the quantitative comparison of the end-effector vibration of both control methods. The coefficient of variation (C.V.) was used to determine the relative distribution. The proposed method showed its ability to significantly reduce the vibration wherein x-axis the vibration was suppressed 3.05 times compared to the simple control method.

$$C.V. = \frac{S.D.}{\bar{X}} \times 100 \quad (17)$$

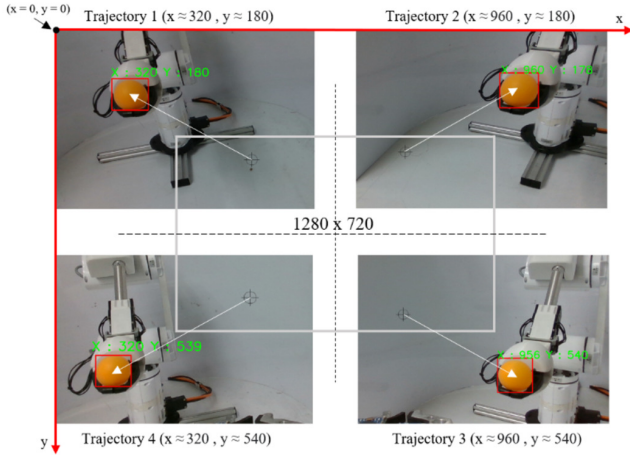
**Table 2:** The coefficient of variation of acceleration from both control methods.

Control method	x	y	y
simple	69.01%	29.02%	12.99%
proposed	22.63%	26.19%	7.16%

In addition to the vibration analysis, we studied the motion characteristic of the proposed control method by conducting the experiment depicted in Figure 19. The manipulator was assigned to track the object in 4 trajectories. The reference movement

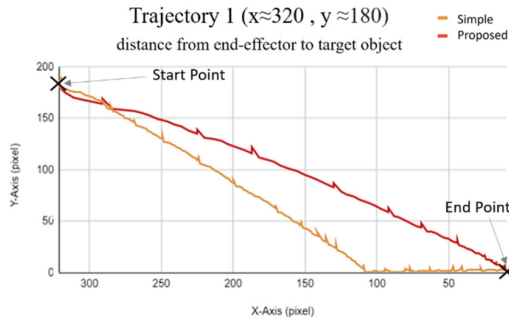


path of each trajectory is a linear line from the end-effector location at the beginning to the target object.

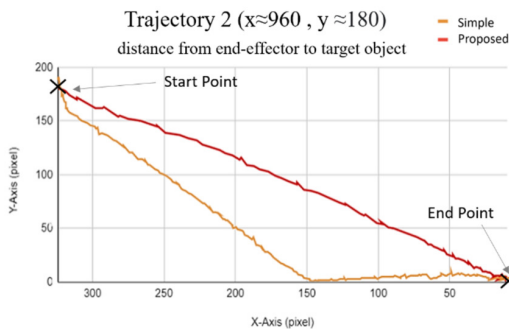


**Fig.19:** Direction and trajectories.

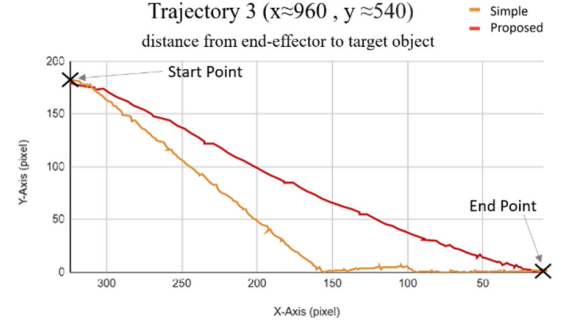
**Figures 20 – 23** describe the distance from end-effector to the target object while the manipulator was approaching the target object. The result shows that the proposed method controlled the end-effector to follow the reference movement path closely in all 4 cases.



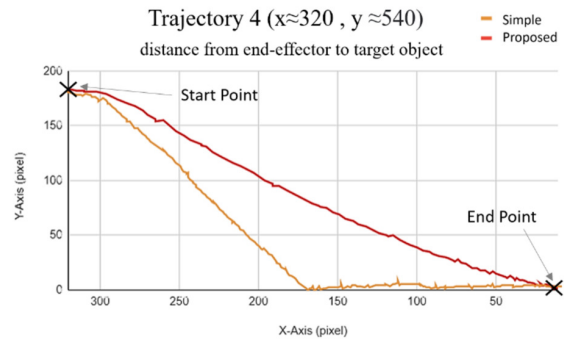
**Fig.20:** Distance from end-effector to target object of trajectory 1.



**Fig.21:** Distance from end-effector to target object of trajectory 2.



**Fig.22:** Distance from end-effector to target object of trajectory 3.



**Fig.23:** Distance from end-effector to target object of trajectory 4.

### 3.2 Improved Image Tracking Experiment

In this experiment we observed the time consumed in the object approaching process comparing two image tracking methods: conventional RNN based and the proposed image tracking method using a Kalman filter. The experimental setup was similar to the experiment in 3.1. Table 3 describes the time duration from the beginning of the tracking process until the end-effector reached the target. The proposed image tracking method resulted in a shorter approach time by 1.29 sec. on average.

**Table 3:** Time used to approach the target object.

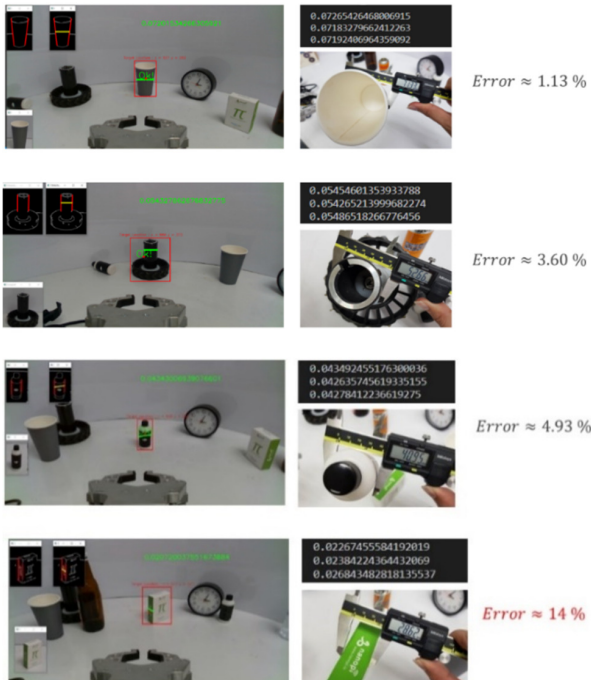
Round	RNN based (s)	RNN + Kalman (s)
1	16.13	14.47
2	15.60	14.20
3	15.33	14.40
4	16.66	14.20
5	15.33	14.13
6	15.23	14.20
7	15.43	14.53
8	15.56	14.30
9	15.46	14.33
10	15.26	14.36
Aaverage	15.60	14.31

### 3.3 Object Size Determination



**Fig.24:** Object size determination experiment setup.

**Figure 24** shows the experiment. A 3D-Camera installed at the gripper is used to obtain an image for providing depth information for object width estimation. Four example objects with different size shapes were used as target objects in each experiment trial. The tracking process shows effective performance as the manipulator can reach object target location (centre of camera frame) in all experiment trails. The object width estimation experimental results are shown in **Figure 25**. The best accuracy is at 1.13% and the error is less than 5% for cylindrical-shape objects, while error increased for the rectangle-shape box due to confusion resulting from multiple detected lines.



**Fig.25:** Object width estimation experiments.

## 4. CONCLUSION

The proposed object tracking and manipulator control process were implemented and evaluated. The

object tracking was enhanced by employing a Kalman filter that reduced the movement duration to approach the target object by 1.29 sec. The manipulator control process reduced the vibration 3.05 times compared to a simple control method. The object size determination could determine the object size with minimum error at 1.13%. The proposed methods show their capability to realize a fully automatic manipulator position control without the need of actual encoder or angular position sensor. Moreover, the proposed methods based on visual information can be applied to other applications requiring the handling of a moving object.

## References

- [1] W. Wolfslag, M. Plooi, R. Babu ska and M. Wisse, "Learning robustly stable open-loop motions for robotic manipulation," *Robotics and Autonomous Systems*, April 2015.
- [2] S. Hutchinson, G. Hager and P. Cork, "A Tutorial on Visual Servo Control," in *IEEE Transactions on Robotics and Automation*, vol. 12, no.5, pp. 651-670, Oct. 1996.
- [3] P. I. Corke, VISUAL CONTROL: High-Performance Visual Servoing, CSIRO Division of Manufacturing Technology, Australia., 1996.
- [4] a. S. H. F. Chaumette, "Visual Servo Control Part I: Basic Approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82-90, 2006.
- [5] J. Wu, Introduction to Convolutional Neural Networks, National Key Lab for Novel Software Technology, May 1,2017.
- [6] R. L. Galvez, A. A. Bandala, E. P. Dadios, R. R. P. Vicerra and J. M. Z. Maningo, "Object Detection Using Convolutional Neural," in *TENCON 2018 - 2018 IEEE*, Jeju, 2018.
- [7] B. Qiang, R. Chen, M. Zhou, Y. Pang, Y. Zhai and M. Yang, "Convolutional Neural Networks-Based Object Detection Algorithm by Jointing Semantic Segmentation for Image," *MDPI Sensors 2020*, 2020.
- [8] J. Fan, W. Xu, Y. Wu and Y. Gong, "Human Tracking Using Convolutional Neural Networks," in *IEEE Transactions on Neural Networks*, vol. 21, no. 10, pp. 1610-1623, Oct. 2010.
- [9] M. Erdmann and M. T. Mason, "An Exploration of Sensorless Manipulation," in *IEEE Journal on Robotics and Automation*, vol. 4, no. 4, pp. 369-379, Aug. 1988.
- [10] W. Zhang, Y. Luo, Z. Chen, Y. Du, D. Zhu and P. Liu, "A Robust Visual Tracking Algorithm Based on Spatial-Temporal Context Hierarchical Response Fusion," *MDPI Article Algorithms*, December 2019.
- [11] F. Lotf, V. Ajallooeian and H. D. Taghirad, "Robust Object Tracking Based on Recurrent Neural Networks," *2018 6th RSI International Confer-*



- ence on Robotics and Mechatronics (IcRoM)*, pp. 507-511, 2018
- [12] D. Gordon, A. Farhadi and D. Fox, "Real-Time Recurrent Regression Networks for Visual Tracking of Generic Objects," in *IEEE ROBOTICS AND AUTOMATION LETTERS. PREPRINT VERSION. DECEMBER*, 2017.
- [13] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Jun 2017.
- [14] K. He, G. Gkioxari, P. Dollar and R. Girshick, "Mask R-CNN," in *IEEE transactions on pattern analysis and machine intelligence*, 2018.
- [15] G. Welch and G. Bishop, An Introduction to the Kalman Filter, Chapel Hill, University of North Carolina at Chapel Hill, 2006.
- [16] G. Salaganata, Munri and Y. Affryenni, "Moving Object Tracking Using Hybrid Method," in *2018 International Conference on Information and Communications Technology (ICOIACT)*, pp. 607-611, 2018.
- [17] Taylor, Liana; Mirdanies, Midriem; Saputra, Roni Permana, "Optimized object tracking technique using Kalman filter," *Mechatronics Electrical Power and Vehicular Technology*, July 2016.
- [18] H. A. Patel and D. G. Thakore, "Moving Object Tracking Using Kalman," *International Journal of Computer Science and Mobile Computing*, vol. 2, no. 4, pp. 326-332, 2013.
- [19] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35-45, 1960.
- [20] K. Saho, "Kalman Filter for Moving Object Tracking: Performance Analysis and Filter Design," 20 December 2017.
- [21] Y. Kim and H. Bang, "Introduction to Kalman Filter and Its Applications," 5 November 2018.
- [22] W. Mokrzycki, "New version of Canny edge detection algorithm," in *ICCVG*, Olsztyn, January 2012, pp. 533-540.
- [23] W. Gongjian and W. Runsheng, "A Survey on Hough Transform, Theory, Techniques and Applications," *IJCSI International Journal of Computer Science Issues*, vol. 12, no. 1, January 2015.



**Hiroshi Nakahara** received the B.Eng. degree and the M.Eng. degree in electrical engineering from Prince of Songkla University, Hat Yai, Songkhla, Thailand, in 2016 and 2022, respectively. His current research interests include machine learning, robotics, computer vision, and embedded systems.



**Kittikhun Thongpull** received the B.Eng. degree and the M.Eng. degree in electrical engineering from Prince of Songkla University, Hat Yai, Songkhla, Thailand, in 2008 and 2010, respectively. And the Ph.D. degree (Electrical and Computer Engineering) University of Kaiserslautern, Germany, in 2015. Since September 2015, he has been a Faculty Member with the Department of Electrical Engineering, Faculty of Engineering, Prince of Songkla University. His research interests include wireless sensor networks, embedded systems, and industrial applications.