Jurnal Kejuruteraan 34(4) 2022: 729-739
https://doi.org/10.17576/jkukm-2022-34(4)-20

Visual Servo Algorithm of Robot Arm Simulation for Dynamic Tracking and Grasping Application

Rishi Arran A/L Suppramaniam, Mohd Hairi Mohd Zaman*, Mohd Faisal Ibrahim, Seri Mastura Mustaza & Asraf Mohamed Moubark

Department of Electrical, Electronic and Systems Engineering, Faculty of Engineering & Built Environment, Universiti Kebangsaan Malaysia, Malaysia

**Corresponding author: hairizaman@ukm.edu.my*

Received 3 September 2021, Received in revised form 25 November 2021
Accepted 25 December 2021, Available online 30 July 2022

ABSTRACT

Health pandemics such as Covid-19 have drastically shifted the world economics and boosted the development of automation technologies in the industries for continuous operation without human intervention. This paper elaborates on an approach to dynamically track and grasp moving objects using a robot arm. The robot arm has an eye-in-hand (EIH) configuration, where a camera is installed on the robot arm's end effector. The working principle of the robot arm in this paper is mainly dependent on the recognition of augmented reality markers, i.e., Aruco markers, placed on the dynamically moving target object with continuous tracking. Then, the proposed system updates the predicted location for the markers using the Kalman filter for performing grasping. The proposed approach identifies the Aruco marker on the target object and estimates the object's location using previous states, and performs grasping at the exact predicted location. When extracted information is updated, the vision system also implements a feedback control system for stability and reliability. The proposed approach is tested using simulation of the dynamic moving object with different speeds and directions. The robot arm with the Kalman filter can track and grasp the dynamic object at a speed of 0.2m/s with a 100% success rate while obtaining an 80% success rate at a speed of 0.3m/s. In conclusion, the moving object's speed is directly proportional to the grasping time until it reaches the threshold speed for the camera in identifying the Aruco markers. Future works are required to improve the dynamic visual servo algorithm with motion planning when obstacles are present in the path of robot grasping.

Keywords: Eye-in-hand; Aruco Markers; Kalman Filter; Feedback Controller; Velocity Threshold

INTRODUCTION

Industrial Revolution 4.0 (IR4.0) has shifted the industrial sector focus from automation with digital technology to interconnected industrial processes through Internet-of-things (IoT), which has greatly influenced technological development worldwide. The focus lies in integrating humans with autonomous robots in manufacturing processes for continuous operations. Autonomous robots can carry out tasks with ease, safely and precisely, and carry out tasks that are challenging for humans (Vaidya et al. 2018). Health pandemics such as Covid-19 have accelerated the autonomous robot industry. In this case, factories are required to operate with less human intervene with robots taking charge of the situation for continuous operations (Yamani et al. 2020). Latest control methods are being developed rapidly to advance automation to ensure the continuous investment and usage of automation technology in the industry. Covid-19 also demanded developing a visual servo system for robots for dynamic grasping and sorting applications. The servo mechanism achieves the desired location by using the error feedback between the actual and intended position. On the other hand, visual servo utilizes visual data, image processing, and a control system for robot control. Visual servo is being developed as it is low cost to develop visual-aided robots rather than complex joint controlled robots (Chaumette & Hutchinson 2009; T. Chen & Lin 2020; Peter Corke 2017).

Visual servos have two camera configurations: (a) Eye-in-hand and (b) Eye-on-world. Visual servo techniques are divided into two main parts: (a) Image-based visual servo (IBVS) and (b) Position-based visual servo (PBVS) (Peter Corke, 2017). IBVS utilizes the image data point feature of 2D objects is used to identify the target object, whereas, PBVS on the other hand, uses a 3D image feature related to the view of the camera and environment to predict the relative orientation dan movement related to the camera position. IBVS systems using the eye-in-hand method gain their ground as they possess advantage and reliability during their operations, and they reduce the computation complexity and unwanted environmental computations (Thotakuri et al. 2017). However, IBVS need to solve complex control system problem because the image features are a nonlinear function of camera pose (Peter Corke, 2017). Thus, PBVS systems are widely used.

Much research is being conducted based on PBVS systems such as autonomous vehicles and drones, where the research has achieved the specified hypothesis. However, PBVS has its disadvantage where the calibration of the robot arm with the robot arm. IBVS, on the other hand, does not require the calibration process as the coordinate computation is made on the image plane. IBVS also has its own disadvantages when the image obtained was not complete or with noise compared to the actual image. Dynamic IBVS is a servo system using a basic IBVS concept with improvements in identifying objects using dynamic techniques. Dynamic can be referred to as moving target object, unknown target object, or hybrid image processing techniques (Bae et al. 2018; Mohebbi et al. 2014; Wang et al. 2017)

Shaw and Chi developed a dynamic IBVS system for object classification for objects at unknown speeds on the conveyer belt (Shaw & Chi 2018). Zhang also uses a system with the same concept in industrial implementation (Zhang et al. 2018). Visual servo algorithm developed if often related with some specific sets of features from the target objects. The most popular target identification method is using an RGB-D camera for color-coded or varying depth object grasping (Durovic & Cupec 2018; Haviland et al. 2020)techniques for tracking and grasping moving objects with an unknown speed on a conveyor using an eye-in-hand robot arm are presented, which are useful in a production line for automatic object classification. First of all, the CAMshift (Continuously Adaptive Meanshift.

Besides that, a 3D model of the object in grasping tasks is also a common type of IBVS system used in the industry. Finally, some research also developed more complex approaches, where neural networking and deep learning are used in artificial intelligence for continuous learning and grasping the object in the camera field-of-view (FoV). Thus, a visual servo algorithm which not complex is being developed for easier implementation in the industry and reduce the development cost by using the classic method of IBVS (Y. K. Chen et al. 2019; Guérin et al. 2018; Shinde et al. 2019; Zapotezny-Anderson & Lehnert, 2019).

Digital image processing is a signal processing technique where the input for the system is an image or a video frame. The output is a parameter or feature related to an image processing for the computer algorithm's usage. Typically, the acquired digital image contains errors resulting from the unstable camera control; thus image algorithm is required to improve the quality of the image obtained (Aqif et al. 2020). This technology is gaining ground as implemented in the robotics industry for the visual servo to avoid any mishaps related to robots. A popular image processing technique in the industry is the CAMshift algorithm (continuously adaptive mean shift). It recognizes an object in the camera FoV by colors and the minimum rectangle method. This CAMshift algorithm is based on the mean shift algorithm, which searches the object to be tracked in an adjustable search window. The downfall of this algorithm is when the object has bad lighting conditions and rounded corners (Peter Corke 2017; Shaw & Chi 2018).

Besides, Bidirectional Extreme Learning Machine (B-ELM) with Smooth Variable Structure Filter (SVSF) were used in the hybrid system by Ren (2020) for optimal target positioning and overcome the problem related to the depth. This system proves its reliability over Kalman filter implementation, but it is limited to the camera's FoV (Ren et al. 2020). Remodeling CAD models of the target object using 3D cameras for Multiview and RANSAC filter also used in hybrid for obtaining optimal grasp planning for an object. This remodeling process complicates the execution process and increases the cost of implementation (Y. K. Chen et al. 2019). Recent research utilizes augmented reality (AR) tags as the primary tracker for object identification coupled with the minimum rectangle method. This method proves its reliability in many implementations and costs significantly less. Thus, this study used AR markers for object identification coupled with a Kalman filter for object target tracking and identification (Mazlan et al. 2020).

Artemciukas et al. (2016) utilize the Kalman filter as a hybrid tracker with some sensors for identifying objects with AR. In this implementation, the Kalman filter ensures the continuity of tracking the AR marker. Kalman filter can be used for robot arm localization and grasping in dynamic target object situations. Kalman filter eases the computation, and an extended Kalman filter is not required if the object moves with linear velocity (Artemciukas et al. 2016; Mohebbi et al. 2014).

Therefore, the main objective of this study is to develop an algorithm to identify and track the dynamic target objects with AR markers by utilizing the Kalman filter. Thus, AR markers are used as it is common in robot arm implementation. Durovic and Cupec use Aruco marker to identify the robot arm location in the camera FoV and grasp the target object in the FoV where the camera is placed in the environment (Durovic & Cupec 2018). Many types of markers are available in the market where they are known as fiducial markers. Sagitov studied the efficiency of every marker, and CALTag contains improvements in every aspect, such as occlusion and rotational rate (Sagitov et al. 2017). The Aruco marker is used in this study over CALTag as a higher rate of occlusion of CALTag might introduce froing materials to the system.

This study focuses on developing IBVS based visual servo system with eye-in-hand configurations with low cost and robust operation techniques for dynamic tracking and grasping target objects. The study's primary objective is to develop a robot arm system for dynamic object tracking and grasping applications. In addition, the study focuses on developing an algorithm for dynamic grasping and sorting where the target object is identified using image processing and augmented reality. Besides, this study also tracks and estimates the motion and distance of moving objects and evaluates the system performance to grasp moving objects in the simulation environment.

RESEARCH METHODOLOGY

Figure 1 shows the visual servo flowchart based on IBVS for dynamic object tracking and grasping.

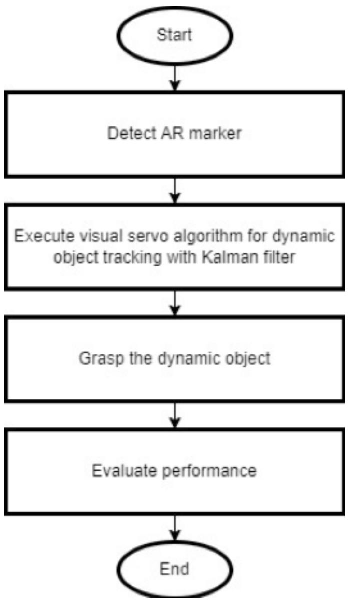
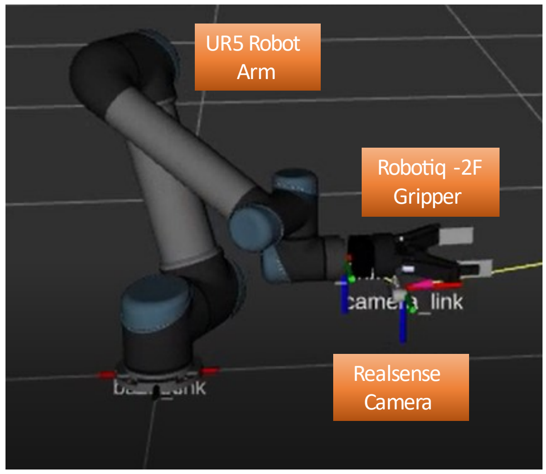


FIGURE 1. IBVS Flowchart

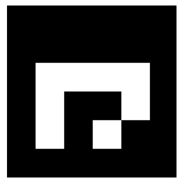
HARDWARE ARCHITECTURE

The system, as displayed in Figure 2, in this study consists of:

1. UR5 (Universal Robots):
   6 DOF with a maximum payload of 5kg. Its joints have ± 360º rotations in all joints, with a speed of 180º/s in rotation and 1m/s in reachability. Maximum reachability is capped at 850mm, and wrist 3 contains the actuator for grasping the object.
2. Intel Realsense D435i Camera:
   Depth mapping camera with depth, RGB and infrared sensor for sweeping and detecting the Aruco marker in FoV. Placed on top of wrist 3 and utilized for obtaining the vector of the gripper to the target object.
3. Robotiq 2F Gripper:
   Two-finger grippers with 140mm off opening for the initial stage of testing development for detecting and tracking application.
4. Suction Gripper:
   Used in overall simulation for implementation in industrial manufacturing as the picking is not limited by 140mm constraint.
5. Aruco Markers:
   Used for object identification and localization by the camera for robot arm for performing detection, tracking, and grasping.



(a) Hardware architecture



(b) Example of Aruco marker

FIGURE 2. Hardware Architecture and Aruco Markers

SIMULATION

As depicted in Figure 3, simulation is done to enable fast recreation of the exact model of the robot arm and defy any negative impacts on the real robot arm when undergoing the testing phase. The software used in this study are as follows:

1. Robot Operating System (ROS):
   An ecosystem for development and testing robot control system which consists of node which communicates through topics and services. The topic is a system bus with no information of node which publishes it. While service acts one by one, where one node offers service and another node asks for executed operation.
2. GAZEBO:
   A robot simulator for real robot simulation, which obtained by URDF file of the robot arm. This can be used to accurately identify collision between objects in the environment, simulate a camera or depth sensor for real-time viewing, and manipulate the joints and robot nodes in the ROS environment.
3. RVIZ:
   Simulation environment in ROS for 3D visualization of the object. This environment is used for camera calibration and visualization in the simulation accurate to the real world.

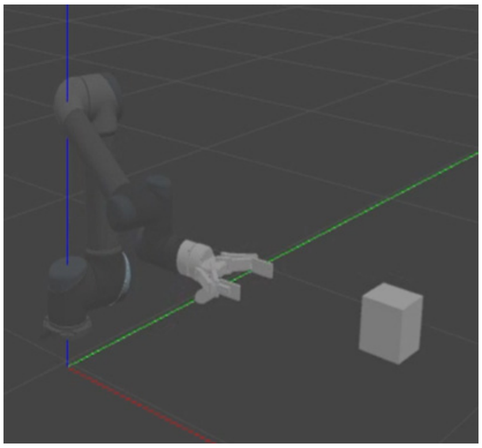


FIGURE 3. Simulation Environment

Software architecture for the recognition and tracking is developed where the Aruco marker must be recognized and tracked by tracking algorithms. Next, the ID data is obtained with its location mapping in the environment. Further movement is predicted, and grasping is taking place.

VISUAL SERVO ALGORITHM DEVELOPMENT

The overall IBVS control system used is shown in Figure 4. The required image is pre-updated to the system when the camera detects the image.

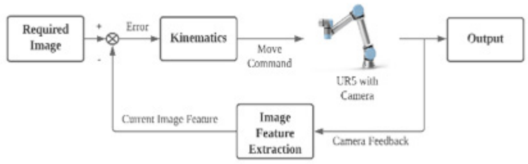


FIGURE 4. IBVS Control System

Overall Visual Servo Algorithm

As displayed in Figure 5, the basic visual servo algorithm mainly works as a detection and movement system, where when the Aruco marker is detected in the camera's FoV, the robot arm moves to the updated location. The tracking is voided when the Aruco marker moves away from the camera's FoV before the location is updated and the robot arm moved. The threshold speed for tracking with this method can be identified using this method. When the tracked marker was not in the arm's reach, this situation caused the robot to move more than 80% or 68cm, and the robot moved back to the home pose.

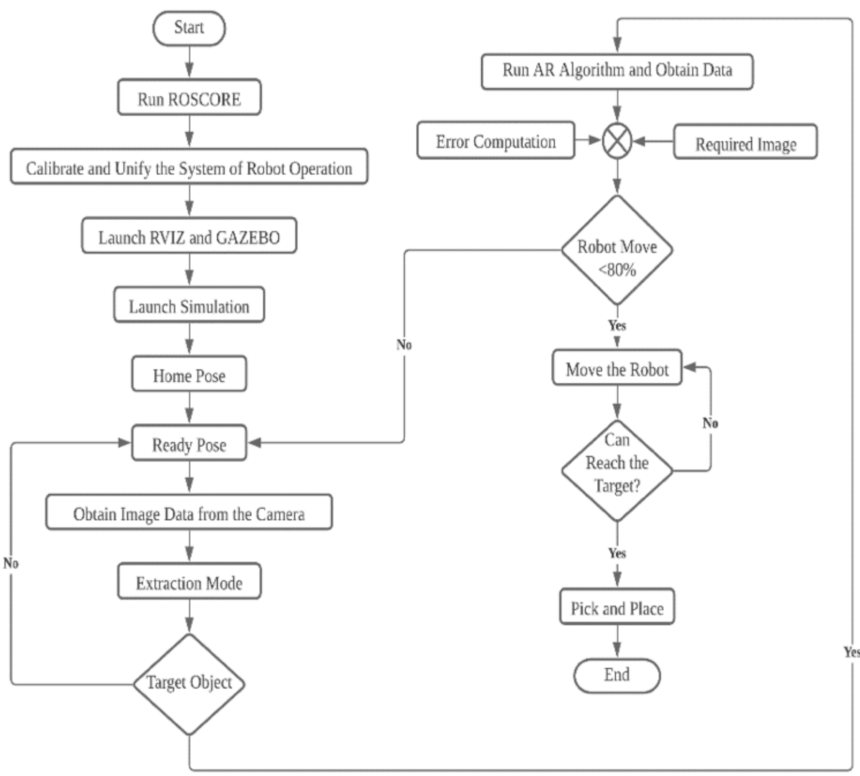


FIGURE 5. Basic Visual Servo Flowchart

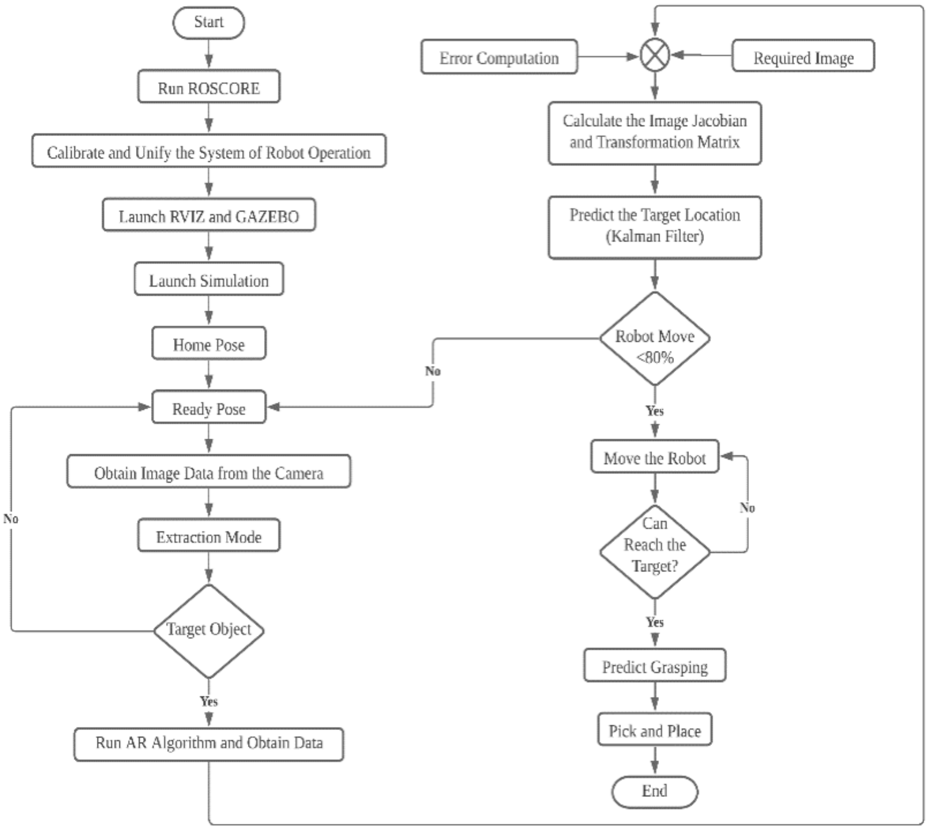


FIGURE 6. Advanced Visual Servo Flowchart

Figure 6 depicts the advanced visual servo algorithm. The algorithm mainly works as a detection, prediction, and movement system. When the Aruco marker is detected in the camera's FoV, the location is predicted with a moving marker, and the robot arm moves to the updated location. This tracking method is considered the advanced tracking method for tracking Aruco markers. Kalman filter implementation provides more reliability than the algorithm without the Kalman filter if the marker moves faster or away from the camera's FoV. The prediction algorithm using the Kalman filter helps track or keep the marker in FoV of the camera. This method makes the tracking possible even on the threshold velocity for basic tracking. When the tracked marker is not in the reach of the arm, the robot moves more than 80% or 68cm, and the robot moves back to the home pose.

Kalman Filter

Kalman filter is used for linear quadratic estimation where statistical error and measurement inaccuracy are utilized in the algorithm. The probability distribution of a variable for each period can be estimated using this filter. Kalman filter is utilized to estimate a parameter to be evaluated in an application. The Kalman filter is usually used for the dynamic visual servo to estimate moving target object location with camera velocity computations. Figures 7 and 8 describe the computations algorithm with great for computing Kalman filter. Noise covariance matrix and sensor covariance matrix can be increased or raised to improve the Kalman filter prediction.

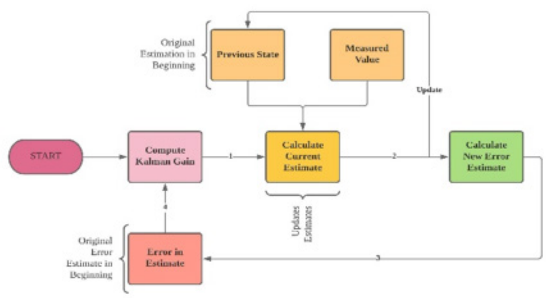


FIGURE 7. Kalman Filter Computation Overview

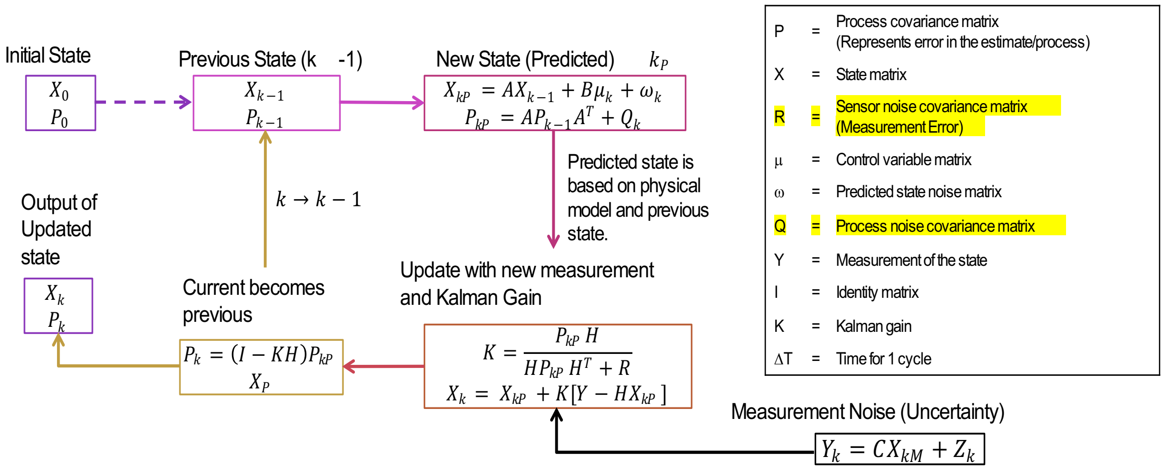


FIGURE 8. Overall Kalman Filter Computation

AR Detection Algorithm

Figure 9 shows the AR (Aruco) detection algorithm. The detection algorithm detected the Aruco marker in the FoV of the camera when it was detected. The depth, orientation, and ED value were obtained from the marker detection and identification and tracking. After done identification and tracking, the algorithm ended. Identification and tracking were considered when the Aruco marker was not in the FoV of the camera for some time or after grasping.

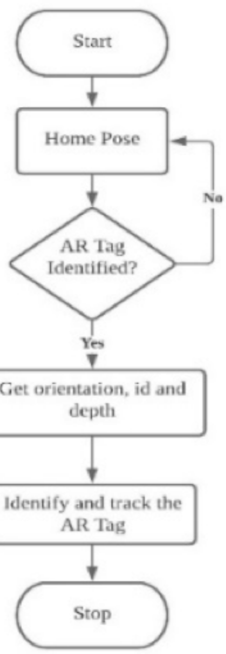


FIGURE 9. Aruco Detection Algorithm

RESULTS AND DISCUSSION

DETECTION

The detection environment in this study is created as follows:

1. EIH configuration robot arm with a camera mounted in wrist 3 of UR5.
2. The Aruco marker will move in FoV of a still camera frame with x, y, and z directions.

Figure 10 portrayed the configuration of the robot arm used for simulation in the detection and tracking part.

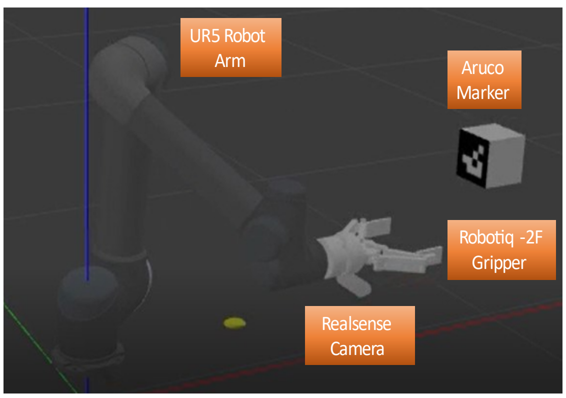


FIGURE 10. Robot Arm Configuration for Detection and Tracking

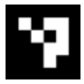
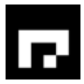
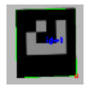
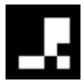
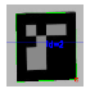
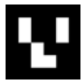
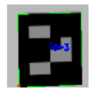
Markers were detected in all orientations, with a maximum distance of 7 meters from the camera, as listed in Table 1. For simplicity of the implementations, pitch, yaw, and roll settings are made constant. Thus, the detection of all the IDs is accurate as it is in simulations, but in real-world implementations, some errors or delays are present in the system. In this case, the detection distance was 7 meters, which is the maximum coded distance in the source library for the camera, but in real-world implementation, the detection may fail when the image processing error occurs.

TRACKING

The same environment as in Figure 5 is utilized in tracking the Aruco marker. First, tracking of the Aruco marker was made for markers approaching from x,y, and z directions with different orientations than the actual marker id. Then the tracking was further developed to enable the following feature for the robot arm to constantly make the marker in the center of the FoV until its maximum reachability.

Next, the threshold speed for tracking the moving marker is determined for the basic method by increasing the marker's movement speed, as shown in Figure 11. The threshold velocity is determined to be 0.4m/s for the basic detect and move method. After that, the manipulator struggles to keep track, and the marker sometimes goes out of the camera frame. Marker position for tracking is kept at 0.2 meters from the end-effector for final tracking using Kalman Filter, as depicted in Figure 12. Tracking is valid at the speed of 0.4m/s, making tracking possible even though the tracking is not very smooth and sometimes lacks behind the object. But having a Kalman filter overshadows the threshold speed tracking. The flat line in tracking using Kalman Filter shows that the marker is not in the camera's FoV.

TABLE 1. Table of Aruco Marker Identification

Marker ID	Initial	Detected	Direction	Distance (m)
0			x, y, z	7
1			x, y, z	7
2			x, y, z	7
3			x, y, z	7

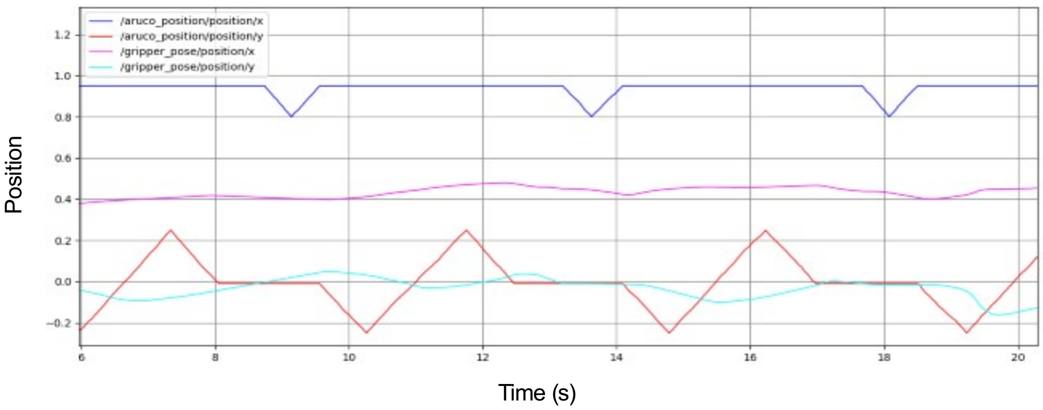


FIGURE 11. Robot Arm Behaviour for Threshold Speed

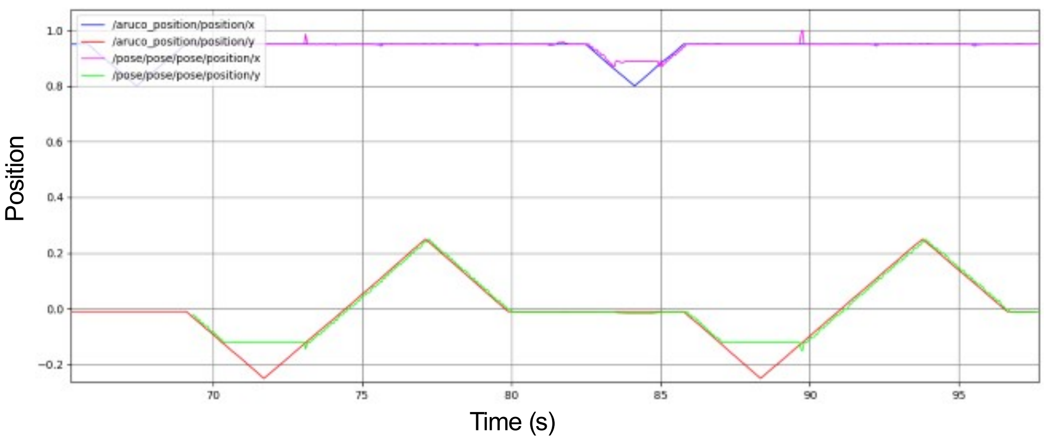


FIGURE 12. Robot Arm Behaviour when using Kalman Filter

GRASPING

The grasping results shown in Figure 13 are built based on the simulation of the robot arm with an EIH configuration camera with a conveyer belt in the environment. First, the target object approaches the FoV of the camera, and the process of grasping takes place. In this phase, the grasping succession of the target was recorded with multiple targets on the conveyer belt approaching the camera's FoV.

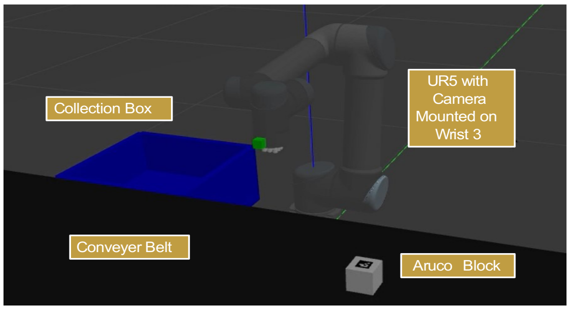


FIGURE 13. Robot Arm Configuration in Simulation Environment

For simplicity, two scenarios are tested marker tracking with Kalman filter and without Kalman filter. Figures 14 and 15 show the process flowchart of the basic grasping and with Kalman filter implementation. Grasping performances for both implementations are recorded by evaluating the time taken to perform the grasp, as shown in Figures 16 to 20. These outcomes are apparent when the motion of the end-effector detects the target object in FoV and moves until the marker and end-effector move together, showing a successful grasping.

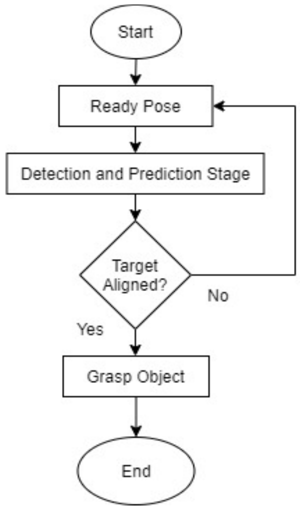


FIGURE 14. Flowchart of Basic Grasping

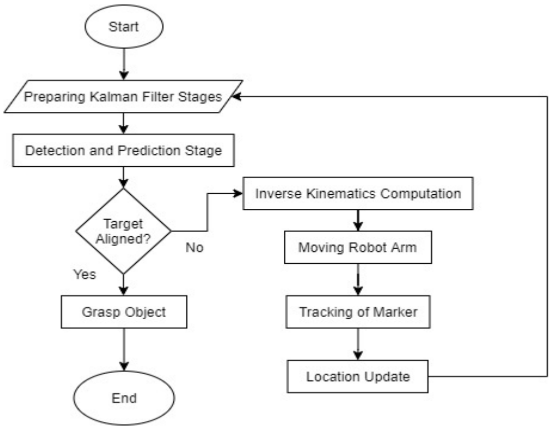


FIGURE 15. Flowchart of Grasping with Kalman

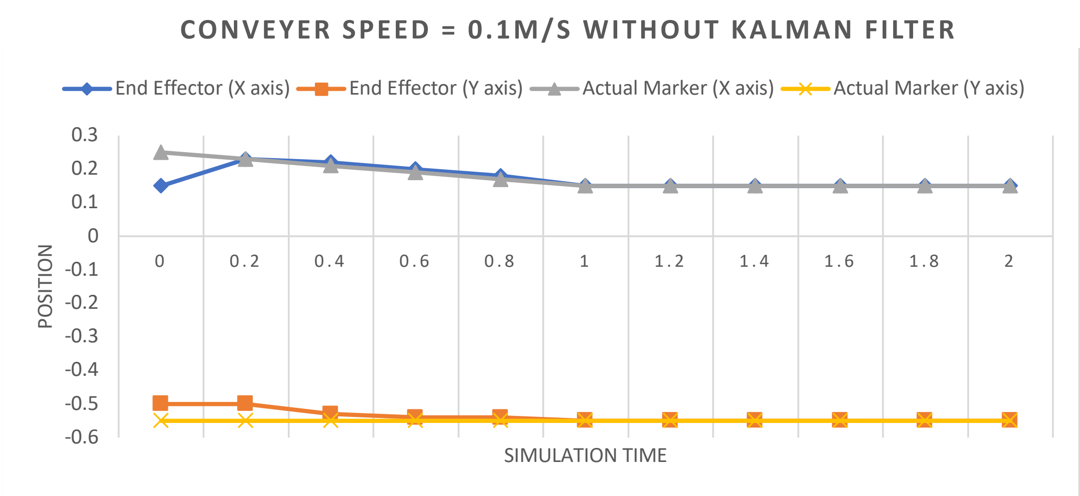


FIGURE 16. Estimated Simulation Time with a Conveyor Speed of 0.1m/s without Kalman Filter

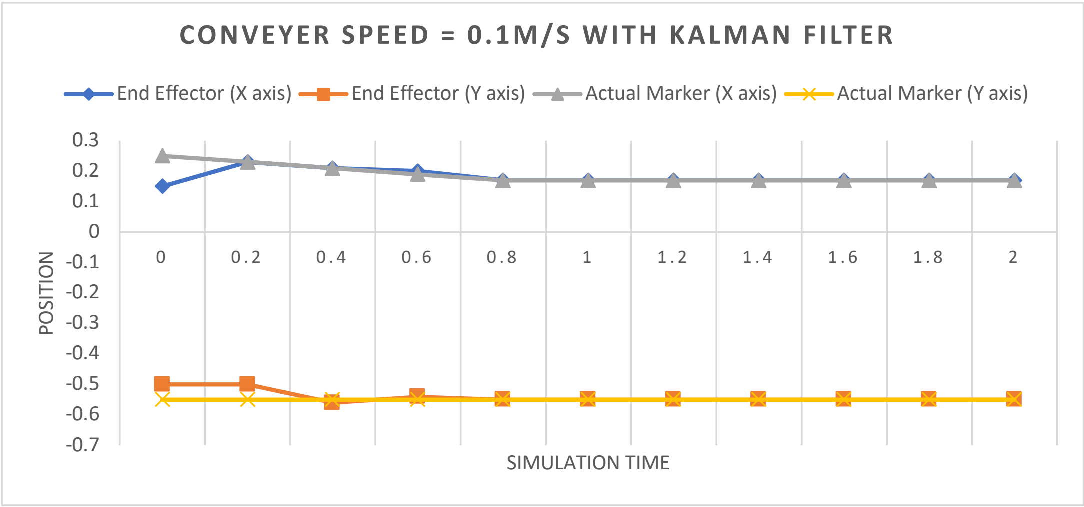


FIGURE 17. Estimated Simulation Time with a Conveyor Speed of 0.1m/s with Kalman Filter

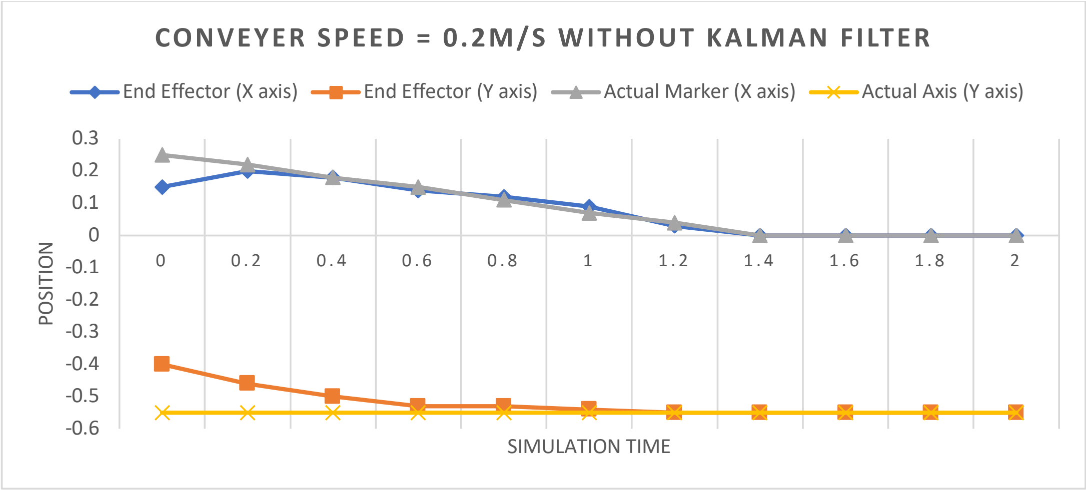


FIGURE 18. Estimated Simulation Time with a Conveyor Speed of 0.2m/s without Kalman Filter

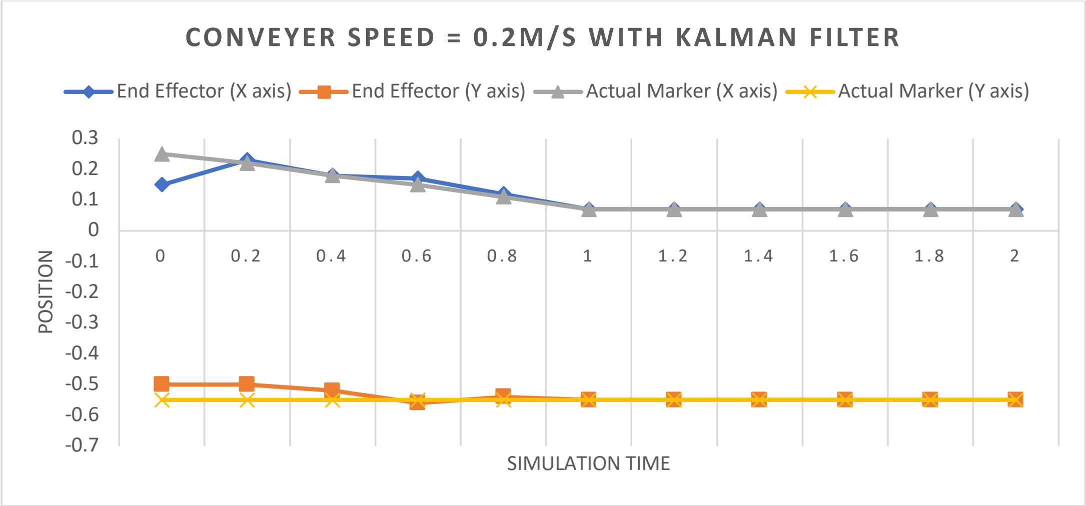


FIGURE 19. Estimated Simulation Time with a Conveyor Speed of 0.2m/s with Kalman Filter

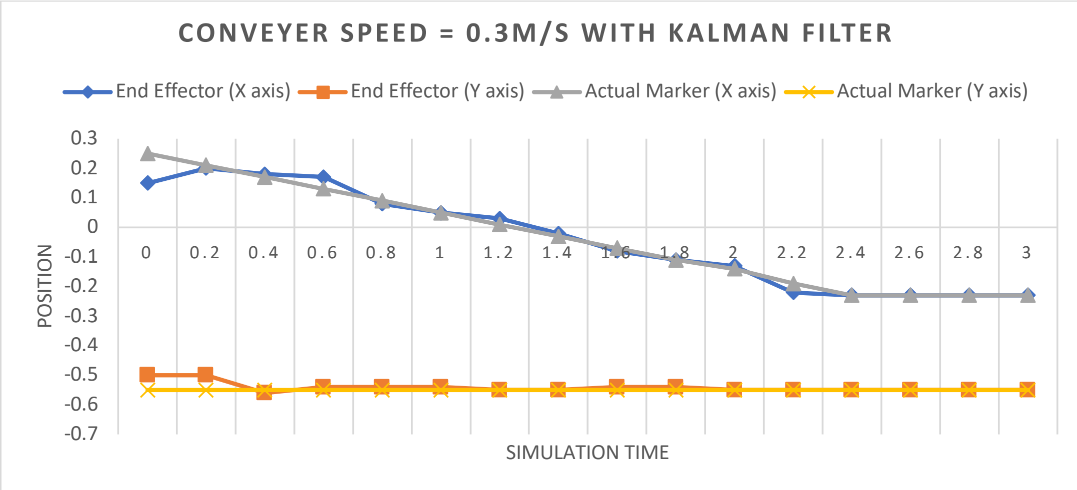


FIGURE 20. Estimated Simulation Time with a Conveyor Speed of 0.3m/s with Kalman Filter

Table 2 shows the success rate when tested with 10 markers continuously in the conveyor system. With basic tracking, the robot arm could not grasp the marker when the object moves at 0.3m/s and even failed to grasp some markers at a speed of 0.2m.s. But with the Kalman filter, a 100% success rate was obtained at a speed of 0.2m/s and an 80% success rate of 0.3m/s. This outcome proves that Kalman filter implementation can be more beneficial for faster moving objects in the real world. The obtained results are also backed up by recent studies showing that grasping time increases correspondence to the speed of moving objects and the image processing controllers (Ishak & Mahmood, 2019; Shaw & Chi, 2018).

TABLE 2. Table of Aruco Marker Grasping Performance

SETUP	CONVEYER SPEED (m/s)	TIME TAKEN (s)	SUCCESS RATE
Basic	0.1	1.2	100
	0.2	2.0	70
	0.3	Keeps Failing	0
With Kalman Filter	0.1	0.8	100
	0.2	1	100
	0.3	2.4	80

CONCLUSION

This paper studies the dynamic object tracking and grasping technique based on the IBVS method with eye-in-hand robot arm configurations. The developed algorithm detected and tracked the Aruco marker using the basic image processing and Kalman filter methods. The grasping is greatly influenced by the object's speed, where it can be said it is directly proportional to the time taken to perform the grasping. Grasping with the Kalman filter proves more effective compared with the basic tracking and updating method in EIH. Basic tracking and grasping can perform the best speed up to 0.2m/s, but some grasping is still missed when the update is slower. Kalman filter implementation proves its superiority by grasping enabled for speed up to 0.3m/s, as compared without Kalman filter.

Even though the tracking threshold speed is defined at 0.4m/s when implemented in the real-world environment, the marker only passes through FoV once, and higher speed caused the marker to append from reachability very fast before the prediction occurs.

The study can be further improved by implementing motion planning for moving objects for obstacle detection and avoidance. Further studies can also be made on the image processing pipeline for faster image processing and updating controller for commending the robot arm joints.

ACKNOWLEDGEMENT

The authors would like to thank Universiti Kebangsaan Malaysia for their financial support under GGPM-2019-051 and CRIM-PIP-SH-2020-06.

DECLARATION OF COMPETING INTEREST

None

REFERENCES

Mazlan, A. M., Zaman, M. H. M., Nasir, M. F., & Ibrahim, M. F. 2020. Robot arm system based on augmented reality approach. *International Journal of Emerging Trends in Engineering Research* 8(10): 6879-6884.

Artemciukas, E., Sakalauskas, L., & Zulkas, E. 2016. Kalman filter for hybrid tracking technique in augmented reality. *Elektronika Ir Elektrotechnika* 22(6): 73-79.

Bae, S. H., Kim, E. J., Yang, S. J., Park, J. K., & Kuc, T. Y. 2018. A dynamic visual servoing of robot manipulator with eye-in-hand camera. *Proceedings of the International Conference on Electronics, Information and Communication (ICEIC 2018)*, 1-4.

Chaumette, F., & Hutchinson, S. 2009. *Visual servo control, Part I : Basic Approaches* 13(4): 82-90.

Chen, T., & Lin, C. W. 2020. Smart and automation technologies for ensuring the long-term operation of a factory amid the COVID-19 pandemic: An evolving fuzzy assessment approach. *International Journal of Advanced Manufacturing Technology* 111(11–12): 3545-3558.

Chen, Y. K., Sun, G. J., Lin, H. Y., & Chen, S. L. 2019. Random bin picking with multi-view image acquisition and CAD-based pose estimation. *Proceedings of the 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2018*) 2218-2223.

Durovic, P., & Cupec, R. 2018. Active vision for low cost SCARA robots using RGB-D camera. *Proceedings of the 2018 Zooming Innovation in Consumer Technologies Conference (ZINC 2018)* 84-87.

Guérin, J., Thiery, S., Nyiri, E., & Gibaru, O. (2018). Unsupervised robotic sorting: Towards autonomous decision making robots. *ArXiv*.

Haviland, J., Dayoub, F., & Corke, P. 2020. Control of the final-phase of closed-loop visual grasping using image-based visual servoing. *arXiv*:2001.05650 [cs.RO].

Ishak, A. J., & Mahmood, S. N. 2019. Eye in hand robot arm based automated object grasping system. *Periodicals of Engineering and Natural Sciences* 7(2): 555–-66.

Mohebbi, A., Keshmiri, M., & Xie, W. F. 2014. An eye-in-hand stereo visual servoing for tracking and catching moving objects. *Proceedings of the 33rd Chinese Control Conference (CCC 2014)* 8570-8577.

Peter Corke. 2017. *Robotics, Vision and Control*. Springer Nature.

Ren, X., Li, H., & Li, Y. 2020. Image-based visual servoing control of robot manipulators using hybrid algorithm with feature constraints. *IEEE Access* 8: 223495-223508.

Sagitov, A., Shabalina, K., Sabirova, L., Li, H., & Magid, E. 2017. ARTag, AprilTag and CALTag fiducial marker systems: Comparison in a presence of partial marker occlusion and rotation. *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2017)* 182-191.

Shaw, J., & Chi, W. L. 2018. Automatic classification of moving objects on an unknown speed production line with an eye-in-hand robot manipulator. *Journal of Marine Science and Technology (Taiwan)* 26(3): 387-396.

Shinde, C., Lima, R., & Das, K. 2019. Deep reinforcement learning based dynamic object detection and tracking from a moving platform. *Proceedings of the 2019 6th Indian Control Conference (ICC 2019)* 244-249.

Thotakuri, A., Kalyani, T., Vucha, M., Chinnaaiah, M. C., & Nagarjuna, T. 2017. Survey on robot vision: Techniques, tools and methodologies. *International Journal of Applied Engineering Research* 12(17): 6887-6896.

Vaidya, S., Ambad, P., & Bhosle, S. 2018. Industry 4.0 - A Glimpse. *Procedia Manufacturing* 20: 233–238.

Wang, K., Ding, N., & Dai, F. 2017. Dynamic visual servoing based regulation of quadrotors on a predefined marker. *Proceedings of the 2017 IEEE International Conference on Information and Automation (ICIA 2017)* 65-72.

Yamani, Y., Long, S. K., & Itoh, M. 2020. Human-automation trust to technologies for naïve users amidst and following the COVID-19 pandemic. *Human Factors* 62(7): 1087-1094.

Zapotezny-Anderson, P., & Lehnert, C. 2019. Towards active robotic vision in agriculture: A deep learning approach to visual servoing in occluded and unstructured protected cropping environments. *IFAC-PapersOnLine* 52(30): 120-125.

Zhang, Y., Li, L., Ripperger, M., Nicho, J., Veeraraghavan, M., & Fumagalli, A. 2018. Gilbreth : A conveyor-belt based pick-and-sort industrial robotics application. *Proceedings of the 2018 Second IEEE International Conference on Robotic Computing (IRC)* 17-24.