

Visual Servo Kinematic Control of Delta Robot using YOLOv5 Algorithm

Kawin Yamtuan¹, Trirat Radomngam², Pradya Prempraneerach^{3*}

^{1,2,3} Department of Mechanical Engineering, Faculty of Engineering, Thammasat University (Rangsit), Pathumthani, Thailand

Email: ¹kawinyamtuan14@gmail.com, ²triratrad@gmail.com, ³ppradya@enr.tu.ac.th

*Corresponding Author

Abstract—Industrial delta robots with motion-and vision-base controllers, providing high-precision and fast/flexible motion, normally come with high cost and complex software. Low-cost delta robots with vision solution could be beneficial to vast industries to increase productivity and to reduce labor cost. A delta robot using a low-cost motion controller and an open-source vision system is developed to accomplish real-time visual servoing with high motion accuracy. In the low-cost motion controller, three parallel links' upper-arm angles, computed from inverse kinematics for a given desired target position by a high-level computer, are used as reference position commands for three AC-motor drives. A low-level Arduino microcontroller is employed to convert these links' angles to high-frequency pulses and on-off signals for synchronously controlling three motor angles and direction. Experimental results of a point-to-point motion tracking exhibit high-precision repeatability. Synchronous pulse generation from Arduino microcontroller and structural misalignments of parallel links are major challenges for achieving high motion accuracy. For the vision-based system, the YOLOv5 algorithm is implemented along with a Python GUI Application. Then, the visual-servo performance is evaluated on localization accuracy and recognition rate of 3-color objects. However, a partial object occlusion can reduce the visual classification rate. A sorting task of 4-category medicine boxes demonstrate a high-speed pick-and-place operation using the low-cost visual-servo system of this delta robot. Therefore, integration of low-cost visual servoing with this delta robot can revolutionize various industries, like automobile, pharmaceutical, and food sectors, in separating, sorting and packing applications.

Keywords—Delta Robot; Inverse Kinematics; Pick-and-Place; Microcontroller; Visual Servoing; YOLOv5 Algorithm.

I. INTRODUCTION

Robotic and automation systems have become increasingly important and employed widely in today's various industrial applications. Industrial robots have been successfully used to handle many types of materials in production line or to replace human workers for performing repetitive task or working in hazardous environments in past few decades. Many industrial robot arms have been designed and developed to operate in a fixed position within a limited workspace. Generally, they can be programmed to perform repetitive and compound motions [1] with different payloads [2] and variable speeds [3] according to various task specifications. Industrial robots; such as serial-link manipulator [3], delta robots [4], hybrid robot [5], [6], flexible manipulator [2], [7] have been continuously researched and improved on different aspects of their

capabilities, including wider working areas, high-speed movement, and flexible manipulations.

The delta robot, known as a parallel or closed-loop manipulator, is usually composed of three parallel links in a combination of either revolute-revolute joints [8], [9] or prismatic-revolute joints [10], [11]. Several advantages of most delta robots over the conventional serial manipulators are 1) fast motion of its end effector, 2) motion accuracy and repeatability, and 3) large payload-capacity to body-weight ratio since driven actuators are fixed on the base plate and only light-weight linkages are moved. However, the delta robot generally yields a small reachable workspace of its end effector and requires complex kinematics [12] and dynamic motion controls [13]. To extend the end-effector workspace, a hybrid parallel robot was constructed by stacking 2 closed-loop linkages [14]. Several challenges in the delta-robot design phase were must be overcome. For example, analysis of delta-robot workspace and singularity [14], [15] were more complex to identify than those of the serial manipulator, thus the inverse kinematics could assist in computation of the delta robot workspace [16]. A trade-off among performance constraints, prescribed workspace, structural kinematics, and joint torque requirement, et. al. could be considered as a multi-level optimization problem [17]. Moreover, solving the inverse kinematics of the delta robot could employ the artificial neural networks as a self-learning process [18]. Owing to flexibility of the delta-robot structure [19], a vibration suppression could be optimized in the trajectory planning by minimizing an elastic deformation of its moving platform with constrains on three-motor torques [20]. The fractional order PID controller, based on an inverse dynamic method to linearized and to uncoupled the delta-robot dynamic model [21], and a robust PD adaptive nonsingular finite-time synergetic control [22] were applied for trajectory tracking and pick-and-place tasks. Furthermore, a model-free iterative learning control strategy combining with Lyapunov approach [23] could precisely control the delta robot for nonrepetitive-trajectory applications. Several design and analysis of parallel manipulators have been incorporated in numerous medical applications: a bone-drilling simulation using a prismatic-revolute parallel manipulator [24], a singularity-free haptic device, constructed by linking a pair of two delta robots to a handle [25], finger-mounted haptic-device prototypes [26], [27], a rehabilitation-massage device using the unified force-position control [28], and a prototype of robot-assisted surgery [29]. On the other hand, the serial-manipulators can cover large workspace when their joints are



controlled independently, resulting in simplified kinematics computation and trajectory control of their end effector. However, the workspace and singularity of the serial manipulator must be consequently determined [30] and avoided [31] to obtain a safe and stable operation of the serial-link robot. In addition, low payload-to-weight ratio due to high inertia of successive linkages and low precision because of cumulative motion errors of connecting joints [32] must be taken into a consideration for specified industrial applications.

Autonomous object grasping by the robot arm is one of challenging tasks that require in various industrial processes. Object detection/localization using the visual feedback as reference positions for multiple objects can be combined with kinematic and dynamic-motion analyses of robotic manipulator such that real-time robotic grasping is achievable. Several hurdles in robotic grasping must be prevailed to improve robot-motion speed, object position identification, robotic/automation dynamic constraint and accurate camera-to-robot pose estimation. A 6-DOF serial robot arm could grasp textureless planar parts using an improved moment-based visual servoing by combining image moments with 3D rotation [33]. For pick-and-place operation, vision system [34], [35], [36] could estimate the position and shape so that a motion control of the delta robot could be coordinated with the conveyer motion to perform a real-time object picking. Also, the delta-robot visual-tracking platforms have been used to provide feedback signal for 1) dynamically visual-servo tracking, employing template matching method on FPGA [37], 2) tracking the position trajectory and controlling desired end-effector force simultaneously [38], 3) optimizing controller designs with respect to both camera and robot-geometric parameters [39] and 4) computing position and velocity of tracked object and automatically camera calibration [40]. Furthermore, kinematic parameters identification of a delta robotic system, consisted of three delta manipulators acting as an individual finger, could be accomplished using a pose estimation with marker board detection from a vision system [41].

The visual servo using YOLO (You Only Look Once) algorithms could enhance accuracy of both object position estimation and object classification to accomplish flexible robotic grasping in real time. The YOLO algorithm, which is one of the most popular one-stage object-detection algorithms, simultaneously performs both object classification and localization [42]-[44] with a smaller inference time than several conventional two-stage object detections, like R-CNN, fast R-CNN and Faster R-CNN [46]-[47], which exhibit comparable accuracy with the YOLO algorithm. Based on Convolutional Neural Network (CNN) architecture, the YOLO can predict objects' class with associated probability across many categories and parallelly detect their positions within bounding boxes. Due to its fast and accurate detection capability, the YOLO algorithm has been used in various applications, for instance, precision livestock monitoring [48]-[49], face-expression identification [50] face detection in complex scene [51], real-time driver assistance system [47], [52]-[53], real-time multiple-object surveillance system of traffic scene [54] and tracking/detection of petrochemical equipment for

static/moving scenes [55]-[56]. Even though, several vision-based systems using the YOLO-v4 and YOLO-v5 algorithms for object detection and recognition, incorporated with a 5-DOF serial-link arm [57], a cobot arm [58] or a YuMi dual-arm collaborative robot [59], can detect and classify multiple objects with a small inference time as well as the robot gripper was able to reach the target or catch-point position of the specified object, but they did not impose any constraint on pick-and-place operational time or accuracy of grasping-position of the robot gripper. On the other hand, the YOLOv4 unified with a particle filter algorithm could guide a UR3 robotic arm to track and to grasp a moving object at high speed with a success rate of 88% [60]. However, most robotic grasp-task researches were employed the industrial collaborative robotic arms that come with a pre-built industrial grade motion controller.

Industrial delta robots can also achieve high motion precision even with varying dynamic load. Nevertheless, cost of the industrial delta robot, sold on the market, is very high up to several thousands of USD. Typical applications of industrial parallel manipulator, such as "FlexPicker" from ABB with fast cycle times [61], were employed in food and electronic industries for sorting and packaging various food products [15], [62] since their linkages can be easily wash and clean. Moreover, a commercial 3D-printed delta robot [63] with a precise motion controller became more attractive for industries and researchers due to lower price around a few thousand of USD, comparing with the industrial-grade type. A few researches have attempted to construct low-cost delta robots with small motion error of their end-effector position. Low-cost revolute-revolute delta robot was built with stepper motors [64] or with servo motors [4], [65] and controlled by the Arduino microcontroller, such that a pick-and-place operation of small light-weight objects can be performed. Another low-cost prismatic-revolute delta robot [66], driven by stepping motors and controlled by a single control chip 8051, could accurately perform pick-and-place (or point-to-point) motion. In a combination with a vision system using CMOS camera, this delta robots could pick up a desired object, classified by color, and then place it in the same color slots and avoid obstacles at the same time. Additionally, apart from the Arduino microcontroller, as a single standalone robot controller, low-cost control firmware and hardware have been implemented on 1) an embedded system using an ESP32 microcontroller for signal generation along with an Arduino microcontroller for position generation for controlling 4-DOF planar manipulator [67] and 2) an embedded system uniting a microcontroller with the PCA9685 chip as a core controller [68] that was potentially applicable for multiple-servo control. Nevertheless, most of the low-cost delta robots still were lacked a synchronous motion controller of three parallel links that can perform fast pick-and-place motion as well as an integration of fast visual-servo system for the real-time pick-and-place operation.

Main contributions of our research are 1) a construction of a three parallel-link, low-cost delta robot, 2) a development of low-cost motion controller using a single Arduino-Due microcontroller for synchronously controlling three upper-arm of this delta robot to accurately move its vacuum gripper to a desired location, and 3) an incorporation of the low-cost

visual feedback system, implemented by the YOLOv5 algorithm, for real-time multiple-object localization with high objects' classification rate. As a result, this low-cost delta robot development with the visual-servo capability can provide fast and flexible multiple-object localization and classification for real-time pick-and-place operation. Thus, this can open up new possibilities for automation in various industries, while lowering investment and labor costs at the same time. However, a workspace of this low-cost delta-robot end-effector was limited within ± 50 cm and 30-60 cm in the horizontal plane and the vertical direction, respectively. Maximum speed limit of this delta robot is less than 22 cm/s with a maximum acceleration of 0.9 m/s², which exhibit comparable speed with that of the commercial 3D-printed SMLT PDR-20 delta robot [63]. Also, the low-cost vision system can detect object with a maximum frame rate of 30-fps due to a limitation of the webcam. This paper is organized as follows: Section 2 describes a structure of the developed delta robot along with robot forward- and inverse-kinematics relations for simulating and verifying a pick-and-place operation of the delta-robot. Also, delta-robot hardware instruments and its motion-control construction using the single low-cost Arduino microcontroller are explained in detail. Besides, the object detection using the YOLOv5 algorithm along with the hand-eye calibration of the developed vision system are presented. Performances of the constructed delta robot are investigated in Section 3 according to motion accuracy and repeatability of the end-effector position. In Section 4, localization accuracy and object classification rate with-object and without-object occlusion of visual-servo system, based on the YOLOv5 algorithm, are experimentally examined using 3-class planar color objects. Finally, a real-time sorting task of 4-class medicine boxes is demonstrated using the visual feedback for this delta robot.

II. DEVELOPMENT OF LOW-COST DELTA ROBOT AND YOLOV5 OBJECT-DETECTION ALGORITHM

A. Delta Robot Structure and Kinematics

Fig. 1 provides a description of the low-cost delta-robot structure and components: 1) a base plate at a top end, 2) three AC motors, located on a base plate, 3) three identical and symmetrical parallel links, composing of upper-arm and lower-arm linkages, 4) a travelling plate, coupled to the lower-arm links using rod ends, at its lower ends of parallel links. In each parallel links, a double rod end, connecting between the upper-arms and lower-arms, allows 2-DOF motion of the lower arm relative to the upper arm. Motion of the delta robot can be synchronously controlled by three motors to rotate its upper arms with prescribed fore-arm's angles θ_{i1} , which can be considered as revolute joints. These coordinated motions allow for a precise movement of the travelling plate.

Fig. 2 and Fig. 3, display reference frames and parameters of the developed delta robot, described in Table I. The base plate and travelling plate have their own inertial frames, denoted by (O_b, X_b, Y_b, Z_b) and (O_t, X_t, Y_t, Z_t) , respectively. Each of the three parallel i -links has an associated angle $\alpha_i = \pi/6, 5\pi/6, \text{ and } 3\pi/2$ for $i = 1, 2, 3$, respectively, each α_i angle is measured relative to the inertial frame O_b . The fore-

arm's angles (θ_{i2} and θ_{i3}) are influenced by a motion constraint of the delta parallel mechanism according to the upper-arm motion coordination through spherical-plain-bearing connections. All variables, defined in this section, are later used in the kinematics formulation [16] of the delta robot.

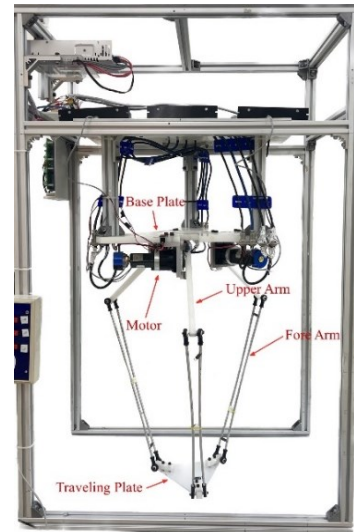


Fig. 1. Overall structure of the developed delta robot

TABLE I. PARAMETERS OF THE DEVELOPED DELTA ROBOT [7]

Delta Robot's Parameters	Symbol	Value (mm)
Upper-arm length	$L_{1i} = L_1$	200
Fore-arm length	$L_{2i} = L_2$	500
Base-plate radius from the O_b center	$R_i = R$	118
Travelling-plate radius from the O_t center	$r_i = r$	112

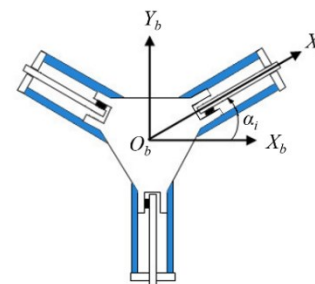


Fig. 2. Inertial reference frame O_b , attached to a center of the delta-robot base plate, and the x -axis of each parallel links from top view

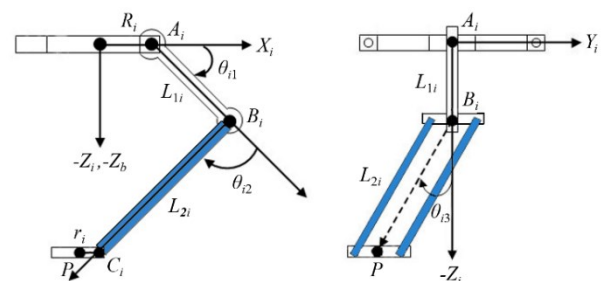


Fig. 3. Reference frames and parameters of the delta-robot parallel links from front view (left) and from side view (right)

The point-to-point movement of the robot end effector can be simulated within MATLAB to validate upper-arm motor angles before deploying to actual hardware of the low-cost delta robot. For a given target point in the point-to-point

motion, rotation angles of each AC servo motors can be computed from the inverse kinematics formulation [16]. Then, a motion simulation of the delta-robot traveling plate, moving from a specified initial position to the target position, can be visualized by solving the forward kinematics to verify the pick-and-place process.

1) Inverse Kinematics of Delta Robot

Inverse kinematics relation is used for computing rotation angles (θ_{i1}) of each i upper-arm link for a given end-effector position at point $P = [X_t, Y_t, Z_t]$. Individual rotation angle of each i servo motor (θ_{i1}) could be solved from the nonlinear inverse-kinematics equation in (1).

$$\tan\left(\frac{\theta_{i1}}{2}\right) = \frac{-2Z_t \pm \sqrt{4Z_t^2 + 4R_1^2 - S_i^2 + Q_i^2 \left(1 - \frac{R_1^2}{L_1^2}\right) - Q_i \left(4R_1 + \frac{2R_1 S_i}{L_1}\right)}}{-2R_1 - Q_i \left(\frac{R_1}{L_1} - 1\right) - S_i} \quad (1)$$

For $i = 1, 2,$ and 3 , $R_1 = R - r$, $Q_i = 2X_t \cos \alpha_i + 2Y_t \sin \alpha_i$, and $S_i = \frac{(-X_t^2 - Y_t^2 - Z_t^2 + L_2^2 - L_1^2 - R_1^2)}{L_1}$. However, for a certain end-effector position at point P , rotation angles (θ_{i1}) can attain two possible solutions, thus continuous values of all θ_{i1} must be cross-checked for a continuous motion of end-effector point P . Furthermore, to obtain real and practical solution of each θ_{i1} , a condition in (2) must be satisfied.

$$4Z_t^2 + 4R_1^2 - S_i^2 + Q_i^2 \left(1 - \frac{R_1^2}{L_1^2}\right) - Q_i \left(4R_1 + \frac{2R_1 S_i}{L_1}\right) \geq 0 \quad (2)$$

2) Forward Kinematics of Delta Robot

The forward kinematics relation can be used for the delta-robot motion visualization. In forward kinematics, the end-effector position point $P = [X_t, Y_t, Z_t]$ can be calculated from an intersection point of three spheres with a shifted center, located at $B_i = [X_i, Y_i, Z_i]$ for $i = 1, 2,$ and 3 , for given three upper-arm rotation angles (θ_{i1}). The point B_i is an end position of the i upper-arm link, which can be calculated from the relation in (3).

$$\begin{aligned} X_i &= (R_1 + L_1 \cos \theta_{i1} - r) \cos \alpha_i, \\ Y_i &= (R_1 + L_1 \cos \theta_{i1} - r) \sin \alpha_i, \\ Z_i &= -L_1 \sin \theta_{i1} \end{aligned} \quad (3)$$

Then, the end-effector position ($[X_t, Y_t, Z_t]$) or the intersection point of three spheres with a radius of L_2 can be solved by (4). The delta-robot motion simulation of a movement of the traveling-plate center (or point P), from an initial position $[X_{t0}, Y_{t0}, Z_{t0}] = (0, 0, -425)$ mm to the first positions $[X_{t1}, Y_{t1}, Z_{t1}] = (130, 120, -540)$ mm and to the second position $[X_{t2}, Y_{t2}, Z_{t2}] = (-130, 120, -540)$ mm is illustrated in Fig. 4.

$$(X_t - X_i)^2 + (Y_t - Y_i)^2 + (Z_t - Z_i)^2 = L_2^2 \quad (4)$$

B. Motion Control Instruments and Software Developments

This section describes both hardware and software of the low-cost motion control for this developed delta robot.

Motion control instruments are composed of 1) three AC servo motors from Yaskawa model SGM-01B-12 [69], directly coupling with the robot upper arm and supplying a maximum rated torque of 0.318 N-m, 2) three AC servo-motor drive from Copley Controls model ACP-180-30 [70], shown in Fig. 5, along with CME2 software for setting up the motor drive, 3) Incremental rotary encoder from Yaskawa model SER48S8-P8G2500-5L with a resolution of 2,500 pulses/revolution such that each upper-arm angle (θ_{i1}) can be feedbacked through each motor drive with high precision. 4) a low-level Arduino Due microcontroller for generating motor-drive commands: variable frequency square-pulse signals from interrupt for position control and digital on/off signals for governing rotational direction as well as transmitting an on/off signal to control a solenoid valve of the vacuum gripper, 5) six limit switches for setting initial or zero position as well as emergently stopping each upper-arm links so that accidental position command, leading to motion outside prescribed workspace, can be prevented, 6) a vacuum ejector from STNC model ZK-08 and the solenoid valve from AirTAG model 3V210-08 for controlling the vacuum gripper operation, and 7) a high-level computer for acquiring desired position command from user through a developed Graphical User Interface (GUI), performing kinematic computation, and communicating with the Arduino Due microcontroller through a USB-3.0 port. Lastly, a DC Switching Power supply model S-350-24 uses to convert 220-VAC voltage to 24-VDC voltage for both AC motor drives and motors of this delta robot. Each motor output shaft is directly coupled with harmonic planetary gear with a gear ratio of 21:1, thus a motion precision of the upper-arm links could be achieved with a maximum static payload upto 50 kg. The Accelnet AC servo-motor drives can perform an angular position control of all 3 AC servo motors, connecting with individual parallel link i of the robot. In summary, Fig. 6 illustrates an overview instrument-connection diagram of this low-cost delta robot.

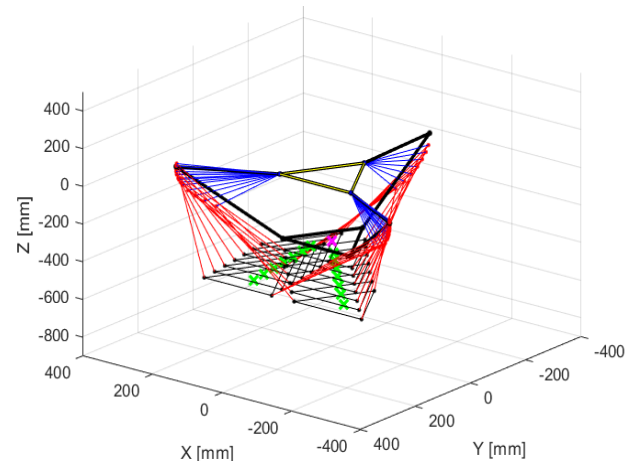


Fig. 4. Motion simulation of delta robot from an initial position at $(0, 0, -425)$ mm to the first-position at $(130, 120, -540)$ mm and to the second position at $(-130, 120, -540)$ mm in isometric-view visualization



Fig. 5. Yaskawa AC Servo-motor: SGM-01B-12 (left) and Accelnet Motor Drive from Copley Controls model ACP-180-30 [70] (right)

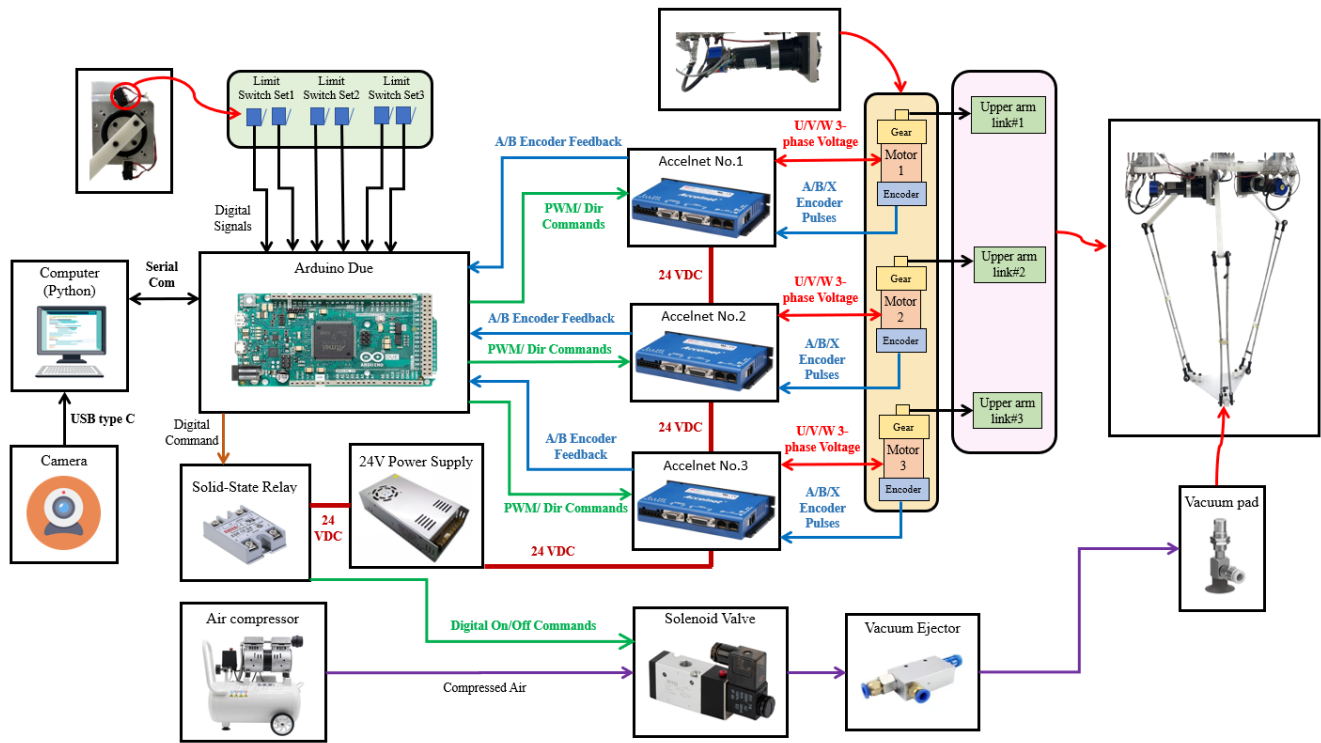


Fig. 6. The instrument-connection diagram of the low-cost delta robot is based on the Arduino-Due microcontroller for the low-cost motion controller and the webcam image analyzing with YOLOv5/Python program for the low-cost vision syste

An implementation of the low-cost motion controller for this delta-robot can be subdivided into 4 major stages. In the first phase, a feedback position control of three AC servo motors for each of the i parallel links must be implemented and verified, thus desired angular position/direction of the upper-arm link can be precisely controlled. Three feedback loops: 1) current, 2) angular speed, 3) angular position of the Accelnet motor drive employ 3-phase motor current for the current feedback and use increment-encoder signals for the angular speed and position feedbacks. Two position commands of the Accelnet motor drive in a position mode: 1) number of square pulse (or PWM waveform) for controlling rotation angle and 2) on/off digital signal of 5VDC for regulating rotation direction can be generated by the low-level Arduino Due microcontroller.

In the second phase, a relationship between number of encoder count and rotation angle of each AC servo motors, connecting to the upper-arm links, must be theoretically established and experimentally validated so that all upper-arm links can be precisely rotated to an initial starting position, called a reset position, after turning on the delta robot for the first time. In theory, one degree rotation of each AC motor leads to encoder counts of 584 pulses. Before starting the delta-robot operation for the first time, the reset or zero angular position can be set by moving three upper-arm links upward to touch the upper limit switches and rotating them downward to stay in the horizontal plane along the x-axis, specified as $\theta_{i1} = 0$ for $i=1,2,3$. This reset position is a very important step for attaining high-precision robot motion in a consequential pick-and-place operation. To calibrate the reset rotation angles of each link, a pitch angle from a microstrain 3DM-GX3 sensor is used to compare with the derived angle from encoder counts, measured by the

CME2 software. Table II summarizes motors' rotation angles of each upper-arm link in the reset-calibration process. Encoder counts from CME2 software as well as measured rotation angles are different for all AC servo motors or i parallel links because the upper limit-switch of all parallel links were manually installed. A ratio of encoder count to rotation angle gives an average value of 585.37 count/degree, which is closely matched with the theoretical value of 584 count/degree. Thus, this conversion relation is verified by the experimental reset procedure. Also, the calculated upper-arm angles from the inverse kinematics can be transformed to the amount of encoder count that would be used in the third stage for generating motion command by the low-cost Arduino microcontroller.

TABLE II. ROTATION ANGLE AND ASSOCIATED ENCODER COUNT OF EACH SERVO MOTORS DURING THE DELTA-ROBOT RESET OPERATION

Motor No.	Encoder Count, measured by CME2 (count)	Rotation Angle, measured by microstrain 3DM-GX3 (degree)
1	36,477	62.34
2	35,063	59.87
3	38,274	65.39

To generate motor-drive position command from the Arduino Due microcontroller, a relation among 1) amount of encoder count, 2) frequency of square-pulse (PWM) signal, and 3) time duration of square-pulse (PWM) signal must be formulated in the third phase. For a synchronous movement of 3 parallel links within a desired time duration, the low-level microcontroller needs to generate different numbers of the square pulse, corresponding to the desired rotation angle of each upper-arm link, by suitably adjusting frequency of the square-pulse (PWM) signal using the microcontroller timer interrupt. The mapping relation of a specified encoder count

must be recorded and tabulated as a function of 1) square-pulse frequency between 20 and 260 kHz with an incremental frequency of 40 kHz and 2) square-pulse duration between 0.5 and 3.5 second with an incremental duration of 0.5 second so that the microcontroller can synchronously and properly generate three square pulses as the position command for all AC-motor drives. Nonlinear characteristics of the square-pulse position command (or number of encoder count), relating to the square-pulse frequency and time duration, can be illustrated by a 3D hyper plane in Fig. 7. However, the encoder count cannot be expressed as an explicit function of the square-pulse frequency and duration, but Python interpolation function can be employed to calculate the square-pulse frequency for a given encoder count with fixed time duration. As a result, desired rotated angle of each parallel link can be simultaneously controlled by the low-level Arduino microcontroller during the same time period.

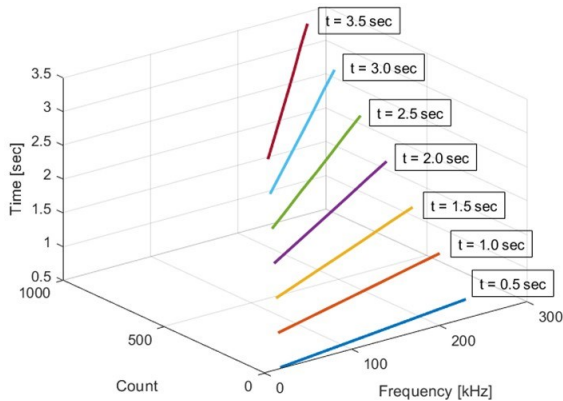


Fig. 7. Nonlinear characteristics of the square-pulse number (or count) as a function of square-pulse frequency and time duration in a 3D hyper plane can be used to generate position commands of each upper-arm link by the low-level Arduino Due microcontroller

To perform the pick-and-place operation, the vacuum gripper needed to be controlled by pneumatic devices using the Arduino microcontroller in the fourth phase. A digital on/off signal from the microcontroller is sent to a solid-state relay to turn on/off the solenoid valve, regulating airflow from the air-pump. Then, a vacuum ejector can generate a vacuum pressure such that a vacuum pad could pick up or release objects due to suction pressure.

C. YOLOv5 Object Detection Algorithm for Visual Servoing

Using the YOLOv5 algorithm, multiple objects' class probability could be predicted as well as objects' center positions could be located in a streaming image, acquiring from a 30-frame/sec webcam with a resolution of 640×480 pixel, consequently the delta robot could pick up the designated object using a vacuum pad and place it at specified position in real-time. Firstly, a relation of hand-eye coordination transformation must be realized. An origin of the vision coordinate system, denoted by $[X_v, Y_v, Z_v]^T$, is set to coincide with that of the inertial reference frame of the delta-robot on the base plate. However, the vision-system x -axis is rotated by a β angle of 150° in the counter-clockwise direction from the delta-robot base frame in Fig. 2, which could be computed in (5), such that the vision-system y -axis is aligned with a conveyor's moving direction. Secondly, using a linear camera calibration method, one of the explicit

calibration techniques [71], a relation between a pixel coordinate and robot inertial-frame or world coordinate is derived to be 0.834 mm/pixel for the 2D-plane based calibration.

$$\begin{bmatrix} X_v \\ Y_v \end{bmatrix} = \begin{bmatrix} \cos(\beta) & -\sin(\beta) \\ \sin(\beta) & \cos(\beta) \end{bmatrix} \cdot \begin{bmatrix} X_b \\ Y_b \end{bmatrix} \quad (5)$$

Thirdly, the YOLOv5 algorithm is implemented and trained for multiple-class objects in Python Google Colab, then the trained model with OpenCV-Python, running on the high-level computer, can detect multiple objects in real time. According to Fig. 8, a high-level network structure of the YOLOv5 model [72], [73] is composed of 1) a backbone, built from a combination of Cross Stage Partial (CSP) and Darknet53 for a feature extraction, 2) a neck, consisted of Spatial Pyramid Pooling (SPP) and PANet for a feature fusion, and 3) a model head, assembled of 3 convolutional layers for handling a multi-scale prediction. The CSP-Darknet can overcome problems of vanishing gradient and redundant gradient by partitioning the feature map into two parts and then merging results according to a cross-stage hierarchy, as a result a large amount of hyperparameters are not required, leading to a shorten inference time. The SPP could enhance a segregation of context features owing to its pyramid formation such that an improvement of relevant information flow increases, resulting in a better localization task. Main advantages of the YOLOv5 model are great object-detection accuracy and small inference time. However, the performance of YOLOv5 algorithm would be considerably reduced when objects were partially occluded [74], [75]. In summary, a flow chart in Fig. 9 describes the pick-and-place procedure using the visual-servo system along with the low-cost motion control of this delta robot, explained from the beginning of this section.

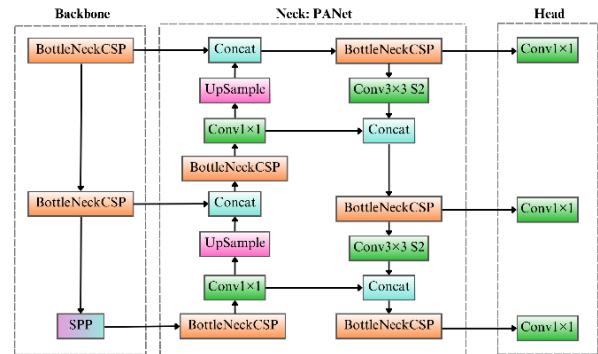


Fig. 8. Network structure of the YOLOv5 algorithm consists of 1) CSPDarknet as the backbone, 2) SPP combining with PANet as the model neck, and 3) Yolo layer built from 3 Convolution layers as its head for the multi-scale prediction

III. PERFORMANCE OF ROBOT MOTION CONTROL USING ARDUINO MICROCONTROLLER

To experimentally evaluate a performance of this developed delta robot, a point-to-point tracking accuracy and a repeatability of simulated pick-and-place motion were examined in the first and the second experiments, respectively. The horizontal position of the travelling-plate center could be measured by a laser-pointer position on the conveyor belt, shown in Fig. 10, such that the position error/accuracy could be assessed. Before performing these

experiments, both reference X -axes and Y -axes must be drawn on the horizontal belt plane below the travelling plate of the delta robot for validating both motion accuracy and repeatability. After resetting all upper-arm links, an initial position of point P was located at $[X_{t_0}, Y_{t_0}, Z_{t_0}] = (0, 0, -528)$ mm.

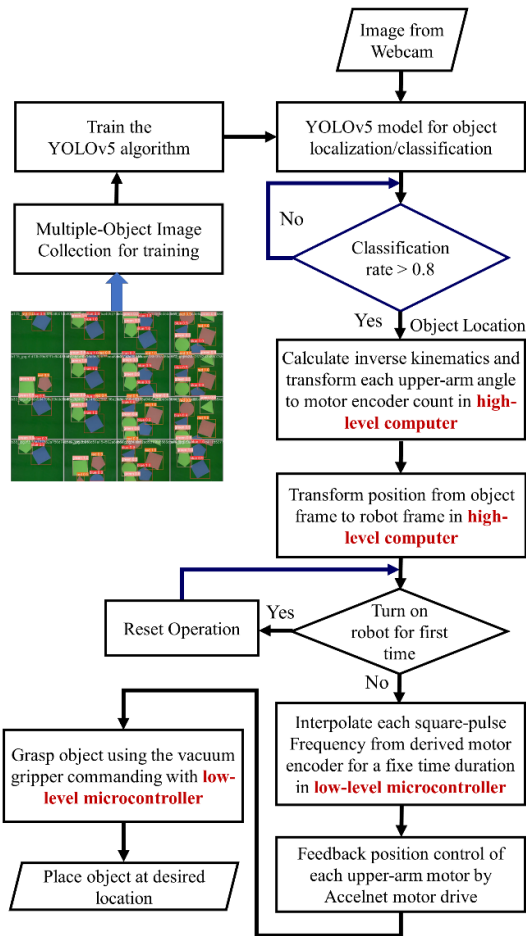


Fig. 9. Flow chart of the pick-and-place procedure by this delta robot using the low-cost motion controller with visual-servo system

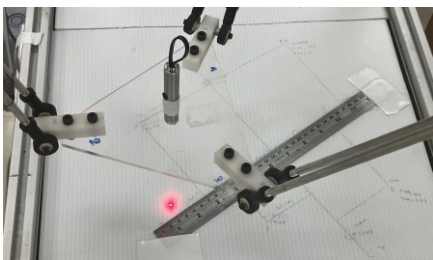


Fig. 10. Using laser-pointer location, motion repeatability and accuracy of the robot end-effector position could be experimentally investigated

A. Point-to-Point Tracking Accuracy

In the first experiment, the motion accuracy of this delta robot was evaluated from a point-to-point movement among eight different positions on the conveyor-belt surface, where the z -direction was fixed at a distance -528 mm from the base plate. An end-effector position discrepancy between the position command and the measured position using the laser pointer is defined in (6).

$$P_error_j = P_{j,command} - P_{j,measured} \quad (6)$$

where $j = x$ -axes or y -axis. To empirically validate the motion along x axis, the delta robot was controlled to move from an initial position at $(0, 0, -528)$ mm to the first position at $(150, 0, -528)$ mm and to the second position at $(-150, 0, -528)$ mm, and then the delta-robot end effector was controlled to move to the third- and fourth-position at $(0, 150, -528)$ and $(0, -150, -528)$ mm, respectively, along the y direction. Fig. 11 and Fig. 12 demonstrate the end-effector motion along x -axes and y -axes. The last four positions were movements both along diagonal and off-diagonal directions in all four quadrants. All end-effector positions and their associated position errors from (6) are summarized in Table III.

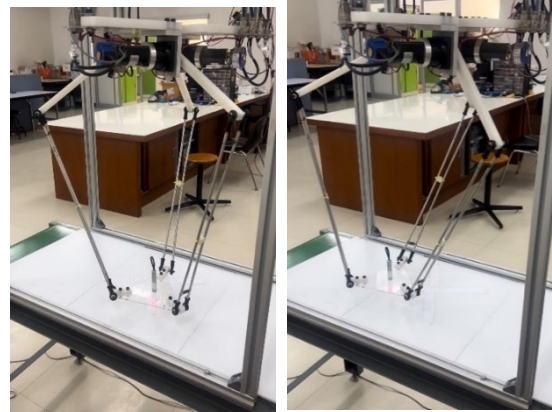


Fig. 11. In motion accuracy experiment, point P located at position $[X_{t_1}, Y_{t_1}, Z_{t_1}] = (120, 0, -528)$ mm (left) and at position $[X_{t_4}, Y_{t_4}, Z_{t_4}] = (0, -120, -528)$ mm (right) illustrate the actual movements of this delta robot

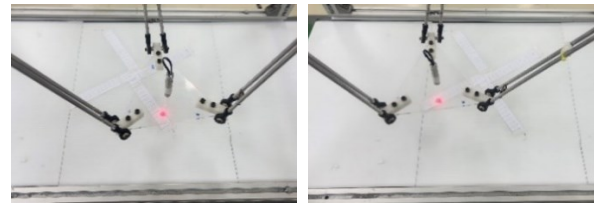


Fig. 12. In motion accuracy experiment, laser-pointer locations measures accuracy when the point P are at position $[X_{t_1}, Y_{t_1}, Z_{t_1}] = (120, 0, -528)$ mm (left) and at position $[X_{t_4}, Y_{t_4}, Z_{t_4}] = (0, -120, -528)$ mm (right)

TABLE III. POINT-TO-POINT MOTION ACCURACY OF ROBOT'S END EFFECTOR, EVALUATED AT 8 POSITIONS

Position No.	Position Command (mm)	Position Error (mm)	
		X axis	Y axis
1	(120, 0, -528)	-2.0	-5.5
2	(-120, 0, -528)	-2.5	-7.0
3	(0, 120, -528)	0	-9.0
4	(0, -120, -528)	0	-3.0
5	(60, 60, -528)	0	-7.5
6	(-80, 100, -528)	-5.0	-8.0
7	(100, -120, -528)	0.5	-5.0
8	(-120, -120, -528)	-4.0	-3.0

In order to evaluate the accuracy of the delta robot motion, command- and measured-positions of all 8 end-effector positions are plotted in Fig. 13, thus results reveal that the actual motion of the delta-robot end-effector position is shifted along the positive y -axis with a maximum error of 9 mm. Nevertheless, motion errors of end-effector positions along the x -axis are much less than those along the y -axis. The shifted end-effector position along the positive y direction could be caused by accumulated errors due to a misalignment of the three parallel-link structure, specifically

from an installation of all rod-end spherical plain bearings, attached with three lower-arm linkages.

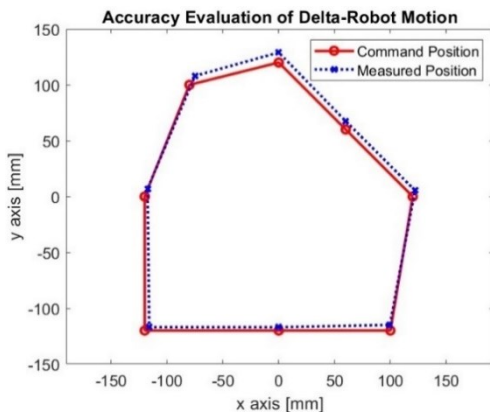


Fig. 13. Eight command- and measurement-positions of the delta-robot end-effector on the 2D level plane in the point-to-point motion accuracy evaluation

B. Point-toPoint Tracking Repeatability

Two main purposes of the robot motion repeatability test are to examine if this delta robot could 1) accurately move back-and-forth between same sets of two points multiple times as well as 2) achieve a desired speed of about 20 cm/s in the repeated pick-and-place motions. In these experiments, the delta-robot end effector were commanded to repeatedly move from the initial position of (0, 0, -528) mm to 4 prescribed positions: 1) pos1 = (0, 120, -528) mm, 2) pos2 = (0, -120, -528) mm, 3) pos3 = (-120, 0, -528) mm, and 4) pos4 = (120, 0, -528) mm for 6 consecutive times.

To quantified precision of the repeatability motion, encoder counts from all 3 motors were recorded, instead of the 2D end-effector position, for 6 repeated times because horizontal position errors of point P were measured to be less than 1 mm in all experiments, which were smaller than a diameter of the laser-pointer location. Fig. 14 illustrates three-motor encoder counts when the delta-robot travelling plate moved between the initial position and the pos1 for 6 repetitions. Moreover, error bars within ± 1 standard deviation (σ) range are plotted overlaying the distribution of three-motor encoder counts at the pos1.

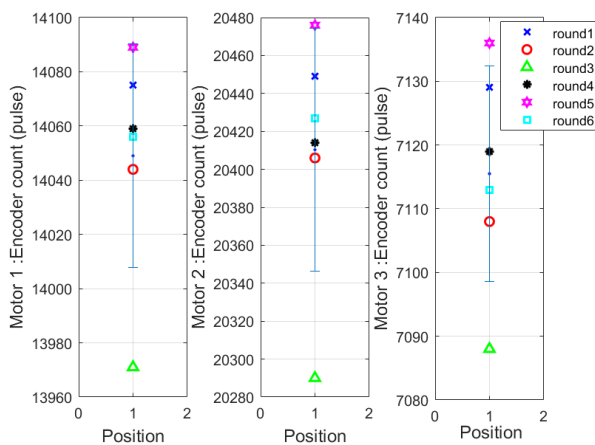


Fig. 14. For motions for 6 consecutive repetitions between pos1 (or pos. no. 1) and initial position, encoder counts of three motors, driving the delta-robot upper-arm links, are narrowly distributed with ± 1 standard deviation, represented by the error bar, around their corresponding mean values, denoted by a solid dot

In Table IV, both mean (μ_i) and standard deviation (σ_i) of motor-encoder counts for each of the i specified position are analyzed; as a result, angular-position errors of the upper-arm links could also be maintained within 0.02° to 0.11° . These results confirmed that three Accelnet motor drives could precisely controlled all three revolute joints of upper-arm links.

TABLE IV. REPEATABILITY EVALUATION FROM MEAN (μ_i) AND STANDARD DEVIATION (σ_i) OF ALL 3 MOTOR ENCODERS FOR 6 REPETITIONS FROM INITIAL POSITION TO 4 DIFFERENT POSITIONS

Pos. No.	Motor 1 (pulse)		Motor 2 (pulse)		Motor 3 (pulse)	
	μ_{i1}	σ_{i1}	μ_{i2}	σ_{i2}	μ_{i3}	σ_{i3}
1	14,049	41.31	20,410	64.18	7,116	16.93
2	14,058	11.36	7,131	33.52	20,406	44.58
3	21,278	30.23	10,064	11.74	10,080	36.08
4	6,044	40.10	17,828	31.28	17,825	30.99

Due to the limitation of position-measurement instrument, the repeatability performance of this delta robot can be assessed by plotting the estimated end-effector position from the forward kinematics. The estimated end-effector position (or estimated point P) at 4 specified positions can be computed from (4) and (5) using three upper-arm angles as inputs, which are converted from all 3-motor encoder counts. Fig. 15 demonstrates scattering plots of 6 repetitive, estimated positions along with 2D bounding ellipses with a horizontal radius of σ_{ix} and a vertical radius of σ_{iy} and an ellipse center, representing the estimated mean positions (μ_{ix}, μ_{iy}).

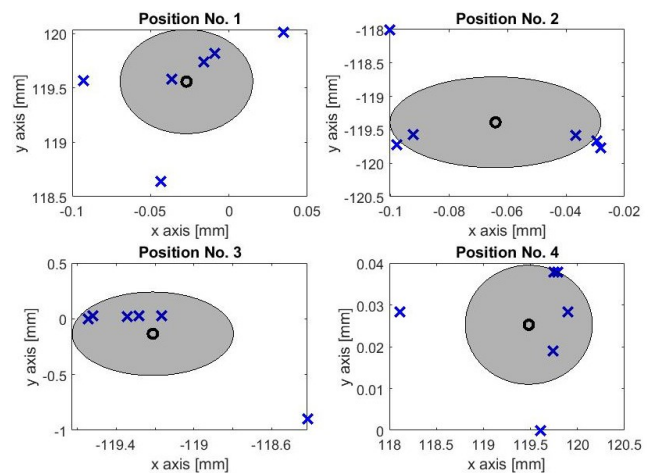


Fig. 15. In repeatability experiments, the end-effector position distribution of 6 repetitive motions is overlaid with an error-bound gray ellipse on the horizontal x-y plane at 4 different specified positions. Centers of ellipses, denoted by thick circles, are means of the position distribution, while the standard deviations in x-axes and y-axes are respectively represented by horizontal- and vertical-radii of each ellipse

Table V sums up both mean (μ_i) and standard deviation (σ_i) of the estimated-position distribution in a 2D plane, computed from 6 repetitive motions, when the robot moves from the initial position to 4 prescribed positions. The horizontal position-error distribution is approximately contained within the maximum range of 1.36 mm. While, the vertical position errors are distributed within a smaller range of 0.3 mm. Thus, these repeatability results confirm a high-precision control using the low-cost motion controller for this delta robot.

TABLE V. REPEATABILITY EVALUATION FROM MEAN (μ_i) AND STANDARD DEVIATION (σ_i) OF POINT- P POSITION FOR 6 REPETITIONS FROM INITIAL POSITION TO 4 DIFFERENT POSITIONS

Pos. No.	X-axis Position (mm)		Y-axis Position (mm)		Z-axis Position (mm)	
	μ_{ix}	σ_{ix}	μ_{iy}	σ_{iy}	μ_{iz}	σ_{iz}
1	-0.03	0.04	119.56	0.48	-528.87	0.15
2	-0.06	0.04	-119.39	0.68	-528.95	0.13
3	-119.21	0.41	-0.13	0.38	-528.83	0.11
4	119.48	0.68	0.03	0.01	-528.94	0.13

In these repeatability experiments, the delta robot was commanded to move to each of the 4 specified positions along a straight path of 120-mm distance within a fixed period of 0.55 second. Therefore, a velocity of this delta-robot movement can be approximated to be 214.55 mm/sec. Even though, estimated position distributions at these 4 locations are contained inside tight elliptical bound, representing the repetition accuracy of $\pm 680 \mu\text{m}$. This fast speed is comparable with the maximum speed of 350 mm/s and the repetition accuracy of $\pm 30 \mu\text{m}$ for the 3D-printed Óscar low-cost delta robot in [63], on the other hand, maximum speed and repetitive accuracy of the other low-cost delta robots [64]-[66] are not stated and analyzed. To quantify a source of the end-effector position error, each encoder count of all three motors was measured and converted to each upper-arm angle (θ_{i1}) for $i = 1, 2, 3$, so that a comparison of measured and simulated upper-arm angles based on inverse-kinematics computation for 5 repetitive motions can be investigated in Fig. 16. Also, high-precision motion control of all three upper-arm links with the travelling speed of 21.5 cm/s can be noticed and confirmed from Fig. 16. However, discrepancies between measured and calculated upper-arm angles are somewhat large, especially for motor 2 and 3. These discrepancies could be caused by structural misalignments of all rod-end bearings as well as imprecise manufacturing tolerances of robot components, leading to the inaccurate motion of the robot end-effector position. To sum up the main advantages of the low-cost motion controller, the repetitive point-to-point motion of the delta robot can achieve high precision with small standard deviation less than 0.68 mm in spite of its fast movement, but its vacuum gripper can pick up only smooth-surface objects larger than 2 cm due to the accuracy limitation of its end-effector position.

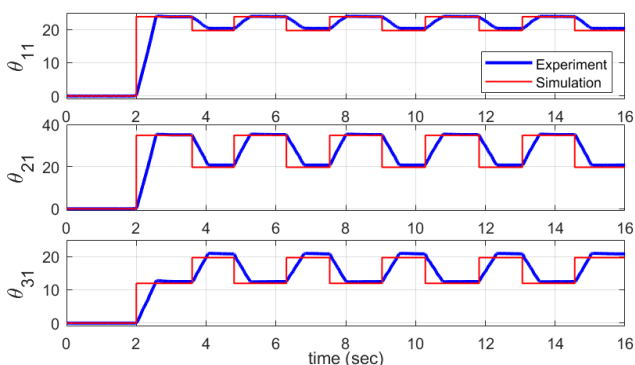


Fig. 16. For 5 repetitive motions between the initial position and pos1 with speed of 214.55 mm/sec, a comparison between 3 measured upper-arm angles, derived from encoder counts, and 3 simulated upper-arm angles, calculated from the inverse kinematics can assist in identifying source of end-effector position error

IV. VISUAL SERVOING FOR PICK-AND-PLACE USING YOLOV5 ALGORITHM

To detect multiple objects at randomize locations and to perform a pick-and-place process, the vision system is built upon the YOLOv5 algorithm, explained in Section II-C, and a user-friendly Graphical User Interface (GUI) is developed for controlling the delta robot and for monitoring its operation. In this research, 2 kinds of objects: 1) 3-class square, color, planar objects and 2) 4-class medicine boxes are correspondingly employed to determine a performance of YOLOv5 algorithm in terms of classification accuracy and object's center localization as well as to demonstrate the real-time sorting task using this low-cost delta robot under a non-controlled interior lighting condition. This developed vision system is limited to operate only in 2D plane, where a depth information of the conveyor belt must be provided beforehand. The GUI, developed by the Python Custom Tkinter, can operate this delta robot in 1) a manual-control mode and 2) a visual-feedback control mode using the YOLOv5 algorithm. For operating in the visual-feedback mode, both center positions and classes' probabilities of all detected objects within a camera's field of view are displayed on the GUI in real time, as illustrated in Fig. 17.

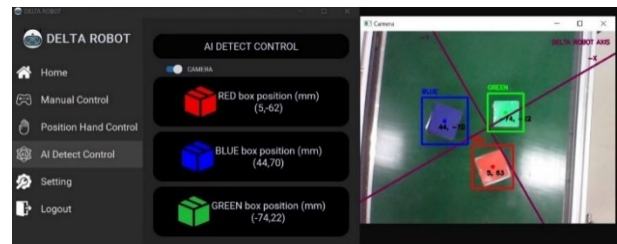


Fig. 17. Graphical User Interface (GUI), developed by Python Custom Tkinter, for localizing object-center positions and recognizing various class objects simultaneously

A. Detection/Classification Accuracy of 3-Class Objects

Detecting and classifying 3-class color papers in a presence of partial occlusion due to the delta-robot lower-arm and travelling-plate was tested for an efficiency of the developed vision system. Firstly, 515 images of red-, green-, and blue-color planar objects of size 5.5×5.5 cm on the green conveyor belt, were collected from the webcam. Then, these images, including partly occluded and non-occluded objects, with a resolution of 480×480 pixels were divided into three sets: 1) 450 images for a training set, 2) 43 images for a validation set, and 3) the remaining images for a testing set. Secondly, the learning process of the YOLOv5 model with its default hyperparameters for these 3-class color-object images was carried out in Google Colab. According to Fig. 18, the training and validating accuracies of the YOLOv5 model with 3-class color-object images can reach 100% and 95%, respectively, after 140 learning epochs.

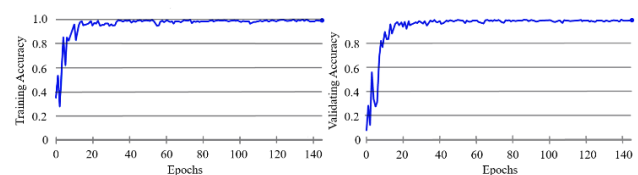


Fig. 18. Training and validating accuracies of the YOLOv5 model using 3-class planar-object images after 140 epochs

After obtaining YOLOv5 model in the training stage, the real-time multiple object detection/classification was tested with 3-color object images without and with partial occlusion due to the movement of the delta-robot travelling plate, vacuum gripper, and lower-arm links, when multiple 3-color objects were randomly placed on the conveyor belt, as demonstrated in Fig. 19. Even though, some parts of these planar objects are obstructed from the camera field of view in several occasions, the trained YOLOv5 model still could predict each of the objects' class with high probability and locate multiple-object positions inside appropriate bounding boxes. Webcam images under partial occlusion from robot structure is one of the major limitations for this visual-feedback system using the YOLOv5. Similarly, if objects are partially occluded, the YOLO performance would be considerably reduced. This problem could be improved by combining with the multi-scale prediction layer [74] and the augmentation policy optimization [75]. Furthermore, to overcome structural-occlusion problem, the visibility maximization controller [76], based on a quadratic-programming path optimization, could be further incorporated to enhance the object visibility during robot movement.

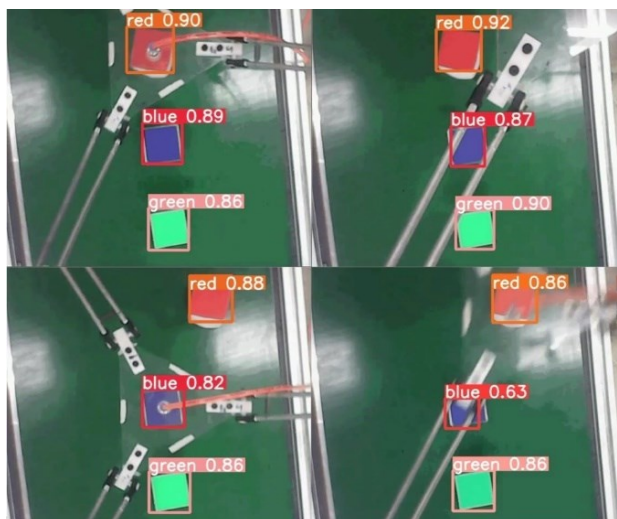


Fig. 19. Demonstrations of object classification using the YOLOv5 model during the validation process illustrates with four sample images from webcam. Three-color planar objects can be correctly classified in real time even with partially-occluded objects due to the delta-robot travelling-plate movements

To determine the object-localization accuracy of the YOLOv5 model in our vision system, centroids of each detected bounding boxes within webcam image were compare against those of actual object centers, marked on each object in advance. These position discrepancies, defined by a difference between bounding-box centroid and actual object center, were measured from snapshot image as pixel variances and then converted to shifted distances. According to Fig. 20, localization-accuracy tests were conducted by randomly placing six-color planar objects on the conveyor belt for four different instances. The larger partially occluded area of object is, the larger the position discrepancy becomes, as summarized in Table VI. Subsequently, these centroids of bounding boxes, localized by the trained YOLOv5 model, can be used as the reference end-effector position command

to perform the visual-feedback motion control of three delta-robot links.

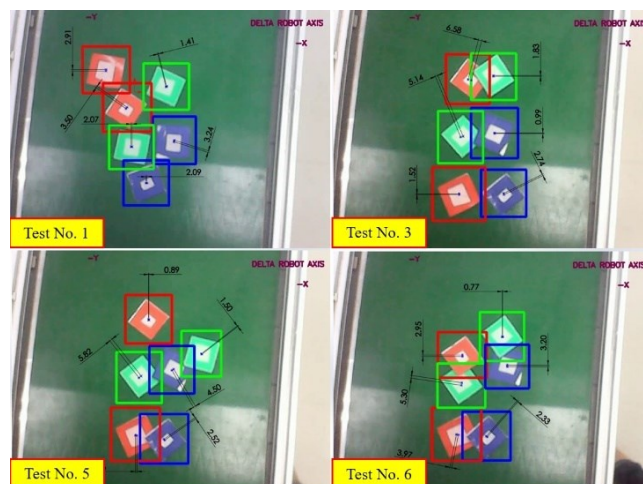


Fig. 20. Localization accuracy of the trained YOLOv5 model in our vision system could be evaluated when six-color planar objects were arbitrarily placed on the green conveyor belt

TABLE VI. LOCALIZATION ACCURACY OF THE YOLOV5 MODEL MEASURED BY POSITION DISCREPANCY BETWEEN BOUNDING-BOX CENTROIDS AND ACTUAL OBJECT CENTERS OF 6 OBJECTS. THE GRAY COLOR DENOTES SITUATION WITH PARTIALLY-OCCLUDED OBJECTS

Test No.	Position Discrepancy (mm)					
	Red1	Red2	Green1	Green2	Blue1	Blue2
1	2.91	3.50	1.41	2.07	3.24	2.09
2	2.06	2.14	2.62	2.25	2.90	3.20
3	6.58	1.52	1.83	5.14	0.99	2.74
4	1.22	1.52	2.67	2.93	2.49	2.31
5	0.89	3.06	5.82	1.50	4.50	2.52
6	2.95	3.97	0.77	5.30	3.20	2.33

B. Pick-and-Place Demonstration by Delta Robot

Firstly, to demonstrate functionality of the pick-and-place operation, 4 different types of medicine boxes must be localized and classified by our vision system, based on the trained YOLOv5 model. Secondly, the vacuum pad of the delta robot can be coordinately moved to pick a desired medicine box and then the grasped box must be placed at its corresponding class location, specified on the conveyor belt by number 1 to 4, as shown in Fig. 21. After training the YOLOv5 model with 61 images of these 4-type medication boxes in a real-world condition, the training and validating accuracies in Fig. 22 can reach 85% and 81%, respectively, after 300 epochs. Several factors contribute to lowering accuracies in training and validating processes of the YOLOv5 algorithm using four-type medicine boxes than those using three-color planar objects. Color and shape similarities of among these 4 types of medicine boxes as well as the smaller set of training images are two of the main reasons for low model accuracy.

Fig. 23 demonstrates sequential images from the pick-and-place operation using the visual-servoing system, implemented for this low-cost delta robot, such that 4 different types of medication boxes were correctly sorted and placed at their specified label locations. This real-time sorting tasks of all 8 medicine boxes were completed within 46 seconds.

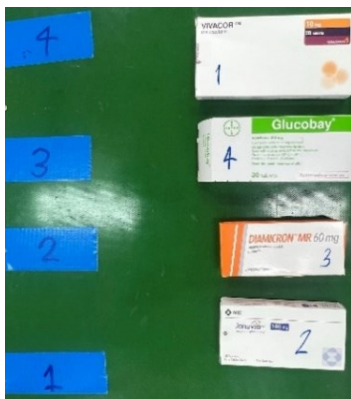


Fig. 21. Four-types medicine boxes and specified label positions are respectively located on the right and left of the green conveyor belt for the sorting-task demonstration using this low-cost delta robot

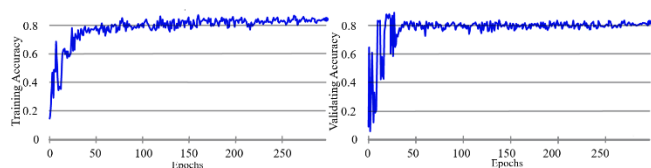


Fig. 22. Training and validating accuracies of the YOLOv5 model using four kinds of medicine-box images after learning for 300 epochs

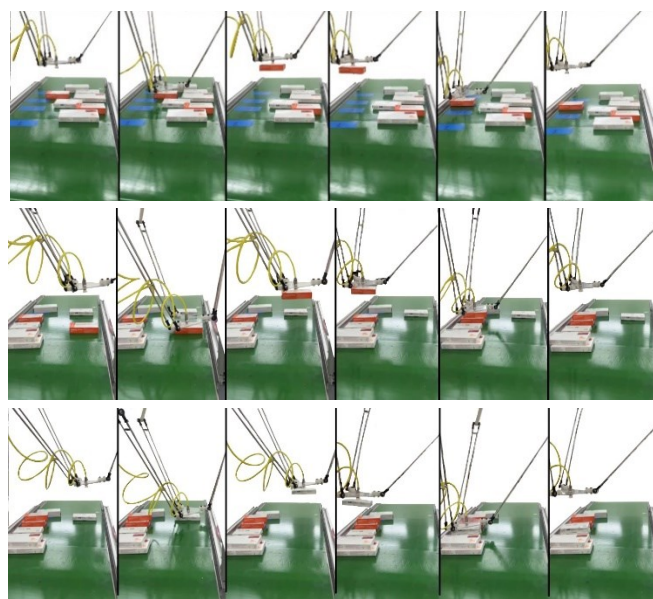


Fig. 23. Pick-and-place sequential operation of this low-cost delta robot is demonstrated for sorting four different types of medication boxes using the visual servoing system. Initially, all 8 medicine boxes, consisted of 2 boxes of each type, were randomly placed on the green conveyor belt

V. CONCLUSION AND FUTURE WORKS

The low-cost delta robot with visual feedback system, developed in this research, contributes to a future development direction of industrial robots that could be beneficial to packaging or material handling applications in various industries in terms of investment- and operation-cost reductions. The low-cost delta robot and kinematic point-to-point control are first developed and simulated to achieve high motion accuracy and high-precision motion repetition. Second, the Arduino Due microcontroller, used as the low-cost motion controller, can synchronously control the movement of the robot upper-arm links as well as coordinate

the grasping operation using the vacuum gripper at the same time. The low-cost visual servoing system, developed base on the YOLOv5 algorithm, can localize and classify multiple objects such that the real-time pick-and-place operation could be automatically performed by this delta robot.

The structure of the developed delta robot are made up of 6 major parts: 1) three parallel upper-arm and fore-arm linkages, connected with a pair of rod-end spherical plain bearings, 2) three AC servo motors and drives for governing angular position of upper-arm links, 3) motor encoders for feeding back individual angular position of all motors to motor drives, 4) the Arduino Due microcontroller as the low-level and low-cost motion controller for synchronously generating position/direction commands for three AC motor drives as well as simultaneously regulating the vacuum-gripper operation, 5) high-level computer for receiving position command from user and for transmitting three computed upper-arm angles to the Arduino microcontroller, and 6) the vacuum gripper for performing the pick-and-place task. The inverse-kinematics computation in the high-level computer and the generation of synchronous motor-drive command in the low-level microcontroller comprise the low-cost motion controller. Inverse kinematics, implemented in Python, can transform the desired end-effector position to three upper-arm angles (θ_{i1}) for $i = 1, 2, 3$. After the Arduino Due microcontroller obtains the computed θ_{i1} , three square-pulse waveforms with appropriate frequency can be created from timer interrupts within the given time duration, deriving from the interpolated tabulation, to control the delta-robot end-effector position. Also, three digital signals as direction commands of three parallel links must be sent to each AC motor drive along with three square-pulse signals. The developed Python GUI can operate this low-cost delta robot in both the manual position control as well as the visual-feedback control modes.

Both motion accuracy and repeatability are generally significant in the majorities of robotic applications such that the right robot can be matched with the right tasks. To evaluate the point-to-point tracking performance of this delta robot, motion accuracy and repeatability were examined from the end-effector position errors in all four quadrants and from 6 repetitive motions along 4 directions, respectively. Results of the end-effector motion-accuracy test revealed that there was a horizontal-position shift along the $+y$ direction, leading to the maximum position error of 9.43 mm. In the repeatability experiment, the encoder counts of three servo motors were measured instead of the end-effector position due to a precision limitation of position measuring instrument. The statistical analysis of encoder counts in 6 repetitive motions unveiled that the feedback control of three upper-arm angular positions using the Accelnet motor drives could achieve high motion precision with small angular error between 0.02 and 0.11 degree. Moreover, the distributions of 6-repetition, end-effector positions, derived from encoder pulses using the forward kinematics, exhibit a maximum horizontal position error of 1.36 mm and a maximum vertical position error of 0.3 mm. Structural and motion-control performances of this delta robot, evaluated in terms of accuracy and repetitive precision, can provide both robot operative specification and its limitation capabilities.

Integration of the low-cost vision system using the YOLOv5 deep-learning algorithm with this delta robot can localize multiple-object positions as well as classify various object types in real time. As a result, the autonomous pick-and-place operation can be performed by the visual servoing of this low-cost delta robot in the real-world condition. The robot guidance using this low-cost vision system can provide several advantages for revolutionizing vast industrial applications, like separating, sorting, and packaging finished products or manufacturing parts. To evaluate the object-localization accuracy of this low-cost vision system, position discrepancy between the bounding-box centroid and the actual object center of the square, planar objects with partial occlusion was greater than distance of 3.5-mm. In case of non-occlusion, the position discrepancy would be smaller than 3.5 mm. Using the trained YOLOv5 model, the object classification rate of 3-class color objects was generally higher than 85%. However, the object classification rate dropped below 60% when objects were severely occluded by the robot structure during its movement. Lastly, the sorting-task demonstration of 4-class medicine boxes could be accomplished with the visual servoing of this low-cost delta robot with its motion speed of 21.5 cm/s. Efficient multi-object sorting tasks using this low-cost visual servoing with the delta robot can directly revolutionize flexible packing and manufacturing applications across numerous industries.

In the near future, both structure and controller improvements should be done for this low-cost delta robot to obtain better motion accuracy and to increase speed in the pick-and-place operation. The robot alignment and calibration technique [77] can be assisted in identifying a source of misalignment of this low-cost delta robot. Furthermore, to achieve faster pick-and-place operation, a nonlinear controller [78]-[79] incorporating with the delta-robot dynamics can be implemented. The reduction of object-detection performance of the YOLOv5 algorithm owing to object occlusion could also be improved by incorporating the multi-scale prediction layer [74]. For a safety operation of this delta robot, particularly at its high-speed operation, infrared line-scan sensors should be installed above the delta-robot base plate in the vertical direction, as an invisible fence. Therefore, the robot operation can immediately stop, if the human worker accidentally come inside this invisible fence area.

ACKNOWLEDGMENT

This work is supported by Mechanical Engineering Department, Thammasat School of Engineering, Thammasat University (Rangsit). Authors would like to thank C. Choosui for constructing several components of this low-cost delta robot.

REFERENCES

- [1] A. Filipescu, D. Ionescu, A. Filipescu, E. Mincă, and G. Simion, "Multifunctional Technology of Flexible Manufacturing on a Mechatronics Line with IRM and CAS, Ready for Industry 4.0," *Processes*, vol. 9, no. 5, p. 864, 2021, doi: 10.3390/pr9050864.
- [2] A. S. Phung, J. Malzahn, F. Hoffmann, and T. Bertram, "Data based kinematic model of a multi-flexible-link robot arm for varying payloads," *2011 IEEE International Conference on Robotics and Biomimetics*, pp. 1255-1260, 2012, doi: 10.1109/ROBIO.2011.6181460.
- [3] M. Abdelaal, "A Study of Robot Control Programing for an Industrial Robotic Arm," *2019 6th International Conference on Advanced Control Circuits and Systems (ACCS) & 2019 5th International Conference on New Paradigms in Electronics & Information Technology (PEIT)*, pp. 23-28, 2019, doi: 10.1109/ACCS-PEIT48329.2019.9062878.
- [4] S. Ahangar, M. V. Mehrabani, A. P. Shorijeh, and M. T. Masouleh, "Design a 3-DOF Delta Parallel Robot by One Degree Redundancy along the Conveyor Axis, A Novel Automation Approach," *2019 5th Conference on Knowledge-Based Engineering and Innovation*, pp. 413-418, 2019, doi:10.1109/KBEL.2019.8734975.
- [5] H. Rakhodaei, "Design and Analysis of 9 DOF Hybrid Parallel Robot," *Doctor of Philosophy Thesis, School of Mechanical Engineering, University of Birmingham, Birmingham, United Kingdom*, 2013.
- [6] M. Michelin, P. -E. Hervé, O. Tempier, J. -B. Izard, and M. Gouttefardé, "Path Following Demonstration of a Hybrid Cable-Driven Parallel Robot," *CableCon 2021 – 5th International Conference on Cable-Driven Parallel Robots*, pp. 323-335, 2021, doi: 10.1007/978-3-030-75789_26.
- [7] H. Park and M. Hwang, "Design of a Multi-bending Flexible Manipulator for Gastrointestinal Surgery," *2022 22nd International Conference on Control, Automation and Systems (ICCAS)*, pp. 1733-1735, 2022, doi: 10.23919/ICCAS55662.2022.10003802.
- [8] C. Choosui and P. Prempraneerach, "Prototype of Delta Robot for Industrial Application," *RSU National Research Conference 2016*, pp. 237-246, 2016.
- [9] S. Patil, S. C. Alvares, P. Mannam, O. Kroemer, and F. Z. Temel, "DeltaZ: An Accessible Compliant Delta Robot Manipulator for Research and Education," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 13213-13219, 2022, doi: 10.1109/IROS47612.2022.9981257.
- [10] M. Ratiu, A. Rus, D. Melentie Anton, and R. Vladimir Ghincu, "Command and Control System for a Parallel Robot," *2021 16th International Conference on Engineering of Modern Electric Systems (EMES)*, pp. 1-4, 2021, doi: 10.1109/EMES52337.2021.9484151.
- [11] B. Mehrafrooz, M. Mohammadi, and M. T. Masouleh, "Kinematic Sensitivity Evaluation of Revolute and Prismatic 3-DOF Delta Robots," *2017 5th RSI International Conference on Robotics and Mechatronics*, pp. 225-231, 2017, doi: 10.1109/ICRoM.2017.8466159.
- [12] Susanto, M. Rohim, and R. Analia, "The Implementation of a Modest Kinematic Solving for Delta Robot," *2019 2nd International Conference on Applied Engineering (ICAE)*, pp. 1-5, 2019, doi: 10.1109/ICAE47758.2019.9221720.
- [13] F. A. Azad, S. A. Rad, and M. Arashpour, "Back-stepping control of delta parallel robots with smart dynamic model selection for construction applications," *Automation in Construction*, vol. 137, p. 104211, 2022, doi: 10.1016/j.autcon.2022.104211.
- [14] D. Malyshev, L. Rybak, G. Carbone, T. Semenenko, and A. Nozdracheva, "Optimal Design of a Parallel Manipulator for Aliquoting of Biomaterials Considering Workspace and Singularity Zones," *Appl. Sci.*, vol. 12, no. 4, p. 2070, 2022, doi: 10.3390/app12042070.
- [15] Y. D. Patel and P. M. George, "Parallel Manipulators Applications – A Survey," *Modern Mechanical Engineering*, vol. 2, no. 3, pp. 57-64, 2012, doi: 10.4236/mme.2012.23008.
- [16] P. Prempraneerach, "Workspace and Dynamic Trajectory Tracking of Delta Parallel Robot," *2014 International Computer science and Engineering Conference*, pp. 1-6, 2014, doi: 10.1109/ICSEC.2014.6978242.
- [17] R. D. J. Portillo-Vélez, I. A. Burgos-Castro, J. A. Vásquez-Santacruz, and L. F. Marín-Urías, "Integrated conceptual mechatronic design of a delta robot," *Machines*, vol. 10, no. 3, p. 186, 2022, doi: 10.3390/machines10030186.
- [18] Z. Iklima, M.I. Muthahhar, A. Khan, and A. Zody, "Self-learning of delta robot using inverse kinematics and artificial neural network," *Sinergi*, vol. 25, no. 3, pp. 237-244, 2021, doi: 10.2244/sinergi.2021.3.001.
- [19] Y.-L. Kuo, "Mathematical modeling and analysis of the Delta robot with flexible links," *Computers and Mathematics with Applications*, vol. 17, no. 10, pp. 1973-1989, 2016, doi: 10.1016/j.camwa.2016.03.018.

- [20] M. Wu, J. Mei, Y. Zhao, and W. Niu, "Vibration reduction of delta robot based on trajectory planning," *Mechanism and Machine Theory*, vol. 153, p. 104004, 2020, doi: 10.1016/j.mechmachtheory.2020.104004.
- [21] J. Viola and L. Angel, "Delta Parallel Robotic Manipulator Tracking Control using Fractional Order Controllers," *IEEE Latin America Transactions*, vol. 17, no. 3, pp. 393-400, 2019, doi: 10.1109/TLA.2019.8863309.
- [22] P.-C. Pham and Y.-L. Kuo, "Robust Adaptive Finite-Time Synergetic Tracking Control of Delta Robot Based on Radial Basis Function Neural Networks," *Appl. Sci.*, vol. 12, no. 21, p. 10861, 2022, doi: 10.3390/app122110861.
- [23] C. E. Boudjedir, M. Bouri, and D. Boukhetala, "Model-Free Iterative Learning Control With Nonrepetitive Trajectories for Second-Order MIMO Nonlinear Systems- Application to a Delta Robot," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 8, pp. 7433-7443, 2021, doi: 10.1109/TIE.2020.3007091.
- [24] J. J. Fernandes, "Kinematic and dynamic analysis of 3PUU parallel manipulator for medical applications," *Procedia computer science*, vol. 133, pp. 604-611, 2018, doi: 10.1016/j.procs.2018.07.091.
- [25] M. Vulliez, S. Zeghloul, and O. Khatib, "Design strategy and issues of the Delthaptic, a new 6-DOF parallel haptic device," *Mechanism and Machine Theory*, vol. 128, pp. 395-411, 2018, doi: 10.1016/j.mechmachtheory.2018.06.015.
- [26] E. M. Young and K. J. Kuchenbecker, "Implementation of a 6-DOF Parallel Continuum Manipulator for Delivering Fingertip Tactile Cues," *IEEE Transactions on Haptics*, vol. 12, no. 3, pp. 295-306, 2019, doi: 10.1109/TOH.2019.2920928.
- [27] S. R. Williams, J. M. Suchoski, Z. Chua, and A. M. Okamura, "A 4-Degree-of-Freedom Parallel Origami Haptic Device for Normal, Shear, and Torsion Feedback," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3310-3317, 2022, doi: 10.1109/LRA.2022.3144798.
- [28] H. Dong, Y. Feng, C. Qiu, and I.-M. Chen, "Construction of Interaction Parallel Manipulator: Towards Rehabilitation Massage," *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 1, pp. 372-384, 2023, doi: 10.1109/TMECH.2022.3202694.
- [29] C. Li, X. Gu, X. Xiao, C. M. Lim, and H. Ren, "A Robotic System With Multichannel Flexible Parallel Manipulators for Single Port Access Surgery," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1678-1687, 2019, doi: 10.1109/TII.2018.2856108.
- [30] M. E. Corominas, "Workspace and singularity determination of a 7-DoF wrist-partitioned serial manipulator," *Bachelor's degree Thesis, Department of Mathematics and Industrial Technology Engineering, Georgia Institute of Technology*, 2020.
- [31] H. Wang, Z. Zhou, X. Zhong, and Q. Chen, "Singular Configuration Analysis and Singularity Avoidance with Application in an Intelligent Robotic Manipulator," *Sensors*, vol. 22, no. 3, p. 1239, 2022, doi: 10.3390/s22031239.
- [32] M. Alexović, P. L. Urban, H. Tabani, and J. Sabo, "Recent advances in robotic protein sample preparation for clinical analysis and other biomedical applications," *Clinica Chimica Acta*, vol. 507, pp. 104-116, 2020.
- [33] Z. He, C. Wu, S. Zhang, and X. Zhao, "Moment-Based 2.5-D Visual Servoing for Textureless Planar Part Grasping," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 10, pp. 7821-7830, Oct. 2019, doi: 10.1109/TIE.2018.2886783.
- [34] W. Zhang, J. Mei, and Y. Ding, "Design and Development of a High Speed Sorting System Based on Machine Vision Guiding," *Physics Procedia*, vol. 25, pp. 1955-1965, 2012, doi: 10.1016/j.phpro.2012.03.335.
- [35] H. Wu, "Research on Control System of Delta Robot Based on Visual Tracking Technology," *Journal of Physics: Conference Series*, vol. 1883, p. 012110, 2021, doi: 10.1088/1742-6596/1883/1/012110.
- [36] Y. Zhang, Z. Wang, M. Li, and P. Gao, "RP-YOLOX-DL: A Deep Learning Hybrid Method for Parallel Robots Target Positioning," *Measurement Science and Technology*, vol. 34, no. 10, 2023, doi: 10.1088/1361-6501/ace20a.
- [37] Z. Chen, S. Li, N. Zhang, Y. Hao, and X. Zhang, "Eye-to-Hand Robotic Visual Tracking Based on Template Matching on FPGAs," *IEEE Access*, vol. 7, pp. 88870-88880, 2019, doi: 10.1109/ACCESS.2019.2926807.
- [38] M. Zhu, C. Huang, Z. Qiu, W. Zheng, and D. Gong, "Parallel Image-Based Visual Servoing/Force Control of a Collaborative Delta Robot," *Front. Neurobot.*, vol. 16, pp. 1-14, 2022, doi: 10.3389/fnbot.2022.922704.
- [39] M. Zhu, S. Briot, and A. Chriette, "Sensor-based design of a Delta parallel robot," *Mechatronics*, vol. 87, p. 102893, 2022, doi: 10.1016/j.mechatronics.2022.102893.
- [40] A. Sharida and I. Hashlamon, "Real Time Vision Based Controller for Delta Robots," *International Journal of Intelligent Systems Technologies and Applications*, vol. 20, no. 4, pp. 271-295, 2021, doi: 10.1504/ijista.2021.121321.
- [41] S. Kansal and S. Mukherjee, "Vision-based kinematic analysis of the Delta robot for object catching," *Robotica*, vol. 40, pp. 2010-2030, 2022, doi: 10.1017/S0263574721001491.
- [42] S. Geethapriya, N. Duraimurugan, and S. P. Chokkalingam, "Real-Time Object Detection with Yolo," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 8, no. 3S, pp. 578-581, 2019.
- [43] Y. H. Lee and Y. Kim, "Comparison of CNN and YOLO for object detection," *Journal of the Semiconductor & Display Technology*, vol. 19, no. 1, pp. 85-92, 2020.
- [44] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A Review of Yolo Algorithm Developments," *Procedia Computer Science*, vol. 199, pp. 1066-1073, 2022, doi: 10.1016/j.procs.2022.01.135.
- [45] T. Diwan, G. Anirudh, and J. V. Tembhrane, "Object detection using YOLO: challenges, architectural, successors, datasets and applications," *Multimed Tools and Appl.*, vol. 82, no. 6, pp. 9243-9275, 2023, doi: 10.1007/s11042-022-13644-y.
- [46] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: optimal speed and accuracy of object detection," *ArXiv, abs/2004.10934*, 2020, doi: 10.48550/arXiv.2004.10934.
- [47] J. S. Murthy, G. M. Siddesh, W.-C. Lai, B. D. Parameshachari, S. N. Patil, and K. L. Hemalatha, "ObjectDetect: A Real-Time Object Detection Framework for Advanced Driver Assistant Systems Using YOLOv5," *Wireless Communications and Mobile Computing*, vol. 2022, 2022, doi: 10.1155/2022/9444360.
- [48] D. Shao, Z. He, H. Fan, and K. Sun, "Detection of Cattle Key Parts Based on the Improved YOLOv5 algorithm," *Agriculture*, vol. 13, no. 6, p. 1110, 2023, doi: 10.3390/agriculture13061110.
- [49] W. Yang, J. Wu, J. Zhang, K. Gao, R. Du, Z. Wu, E. Firkat, and D. Li, "Deformable convolution and coordinate attention for fast cattle detection," *Computers and Electronics in Agriculture*, vol. 211, p. 108006, 2023, doi: 10.1016/j.compag.2023.108006.
- [50] H. Zhong, T. Han, W. Xia, Y. Tian, and L. Wu, "Research on real-time teachers' facial expression recognition based on YOLOv5 and attention mechanisms," *EURASIP J. Adv. Signal Process.*, vol. 2023, no.1, p. 55, 2023, doi: 10.1186/s13634-023-01019-w.
- [51] Q. Xu, Z. Zhu, H. Ge, Z. Zhang, and X. Zang, "Effective Face Detector Based on YOLOv5 and Superresolution Reconstruction," *Computational and Mathematical Methods in Medicine*, vol. 2021, pp. 1-9, 2021, doi: 10.1155/2021/7748350.
- [52] M. A. Farooq, P. Corcoran, C. Rotariu, and W. Shariff, "Object Detection in Thermal Spectrum for Advanced Driver-Assistance Systems (ADAS)," *IEEE Access*, vol. 9, pp. 156465-156481, 2021, doi: 10.1109/ACCESS.2021.3129150.
- [53] A. Kosuge, S. Suehiro, M. Hamada, and T. Kuroda, "mmWave-YOLO: A mmWave Imaging Radar-Based Real-Time Multiclass Object Recognition System for ADAS Applications," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-10, 2022, doi: 10.1109/TIM.2022.3176014.
- [54] I. Ahmed, S. Din, G. Jeon, F. Piccialli, and G. Fortino, "Towards Collaborative Robotics in Top View Surveillance: A Framework for Multiple Object Tracking by Detection Using Deep Learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 7, pp. 1253-1270, 2021, doi: 10.1109/JAS.2020.1003453.
- [55] Z. Wei, S. Dong, and X. Wang, "Petrochemical Equipment Detection by Improved Yolov5 with Multiscale Deep Feature Fusion and Attention Mechanism," *Journal of Electrical and Computer Engineering*, vol. 2022, 2022, doi: 10.1155/2022/8612174.
- [56] Z. Wei, S. Dong, and X. Wang, "Petrochemical Equipment Tracking by Improved Yolov7 Network and Hybrid Matching in Moving

- Scenes,” *Sensors*, vol. 23, no. 9, p. 4546, 2023, doi: 10.3390/s23094546.
- [57] H. Sekkat, S. Tigani, R. Saadane, and A. Chehri, “Vision-Based Robotic Arm Control Algorithm Using Deep Reinforcement Learning for Autonomous Object Grasping,” *Appl. Sci.*, vol. 11, no. 17, p. 7917, 2021, doi: 10.3390/app11177917.
- [58] X. Yue, H. Li, M. Shimizu, S. Kawamura, and L. Meng, “YOLO-GD: A Deep Learning-Based Object Detection Algorithm for Empty-Dish Recycling Robots,” *Machines*, vol. 10, no. 5, p. 294, 2022, doi: 10.3390/machines10050294.
- [59] D. Kijdech and S. Vongbunyon, “Pick-and-place application using a dual arm collaborative robot and an RGB-D camera with YOLOv5,” *IAES International Journal of Robotics and Automation (IJRA)*, vol. 12, no. 2, p. 197, 2023, doi: 10.11591/ijra.v12i2.
- [60] M. Gao, Q. Cai, B. Zheng, J. Shi, Z. Ni, J. Wang, and H. Lin, “A Hybrid YOLO v4 and Particle Filter Based Robotic Arm Grabbing System in Nonlinear and Non-Gaussian Environment,” *Electronics*, vol. 10, no. 10, p. 1140, 2021, doi: 10.3390/electronics10101140.
- [61] M. Mueller, B. Kuhlenkoetter, and H. Geiger, “Cycle Time Estimation for a Delta-Type Robot,” *ISR/Robotik 2014; 41st International Symposium on Robotics*, pp. 1-7, 2014.
- [62] G. -N. Zhu, Y. Zeng, Y. S. Teoh, E. Toh, C. Y. Wong, and I. -M. Chen, “A Bin-Picking Benchmark for Systematic Evaluation of Robotic-Assisted Food Handling for Line Production,” *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 3, pp. 1778-1788, 2023, doi: 10.1109/TMECH.2022.3227038.
- [63] C. M. A. Vasques and F. A. V. Figueiredo, “The 3D-Printed Low-Cost Delta Robot *Óscar*: Technology Overview and Benchmarking,” *Eng. Proc.*, vol. 11, no. 1, p. 43, 2021, doi: 10.3390/ASEC2021-11173.
- [64] L. H. Phuong, V. D. Cong, and H. T. Thanh, “Design a Low-cost Delta Robot Arm for Pick and Place Applications Based on Computer Vision,” *FME Transactions*, vol. 5, no. 1, pp. 99-108, 2023, doi: 10.5937/fme2301099P.
- [65] J. Daniel Martinez Reyes, G. G. Badillo, V. E. E. López, and G. G. Mora, “Objects color classification and repositioning operated by a delta robot,” *2016 XVIII Congreso Mexicano de Robotica*, pp. 1-5, 2016, doi: 10.1109/COMROB.2016.7955157.
- [66] J. Lin, C.-H. Luo, and K.-H. Lin, “Design and Implementation of a New DELTA Parallel Robot in Robotics Competitions,” *International Journal of Advanced Robotic Systems*, vol. 12, no. 10, pp. 1-10, 2015, doi: 10.5772/61744.
- [67] K. Rahul, H. Raheman, and V. Paradkar, “Design of a 4 DOF parallel robot arm and the firmware implementation on embedded system to transplant pot seedlings,” *Artificial Intelligence in Agriculture*, vol. 4, pp. 172-183, 2020, doi: 10.1016/j.aiaa.2020.09.003.
- [68] R. Hu, “Application of Microcontroller-Based Multipath Servos in Industrial Robot Control System,” *Journal of Robotics*, vol. 2023, 2023, doi: 10.1155/2023/7235120.
- [69] Yaskawa, “Sigma series SGM/SGMP/SGDA USER’S MANUAL”, 2023. [Online]. Available: <https://www.yaskawa.com>.
- [70] Copley Controls, April 2023, Accelnet ADP 180-30. [Online]. Available: <https://copleycontrols.com/en/products/adp-180-30/>
- [71] W. Ding, J. Gu, S. Tang, Z. Shang, E. A. Duodu, and C. Zheng, “Development of a calibrating algorithm for Delta Robot’s visual positioning based on artificial neural network,” *Optik*, vol. 127, no. 20, pp. 9095-9104, 2016, doi: 10.1016/j.ijleo.2016.06.126.
- [72] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, “A Forest Fire Detection System Based on Ensemble Learning,” *Forests*, vol. 12, no. 2, p. 217, 2021, doi: 10.3390/f12020217.
- [73] I. Katsamenis, E. E. Karolou, A. Davradou, E. Protopapadakis, A. Doulamis, N. Doulamis, and D. Kalogeras, “TraCon: A novel dataset for real-time traffic cones detection using deep learning,” *Novel & Intelligent Digital Systems: Proceedings of the 2nd International Conference (NiDS 2022)*, pp. 382-391, 2022, doi: 10.1007/978-3-031-17601-2_37.
- [74] K. -J. Kim, P. -K. Kim, Y. -S. Chung, and D. -H. Choi, “Multi-Scale Detector for Accurate Vehicle Detection in Traffic Surveillance Data,” *IEEE Access*, vol. 7, pp. 78311-78319, 2019, doi: 10.1109/ACCESS.2019.2922479.
- [75] S. -E. Ryu and K. -Y. Chung, “Detection Model of Occluded Object Based on YOLO Using Hard-Example Mining and Augmentation Policy Optimization,” *Appl. Sci.*, vol. 11, no. 15, p. 7093, 2021, doi: 10.3390/app11157093.
- [76] K. He *et al.*, “Visibility Maximization Controller for Robotic Manipulation,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8479-8486, 2022, doi: 10.1109/LRA.2022.3188430.
- [77] J. Millington, R. P. Monfared, and D. Vera, “Innovative mechanism to identify robot alignment in an automation system,” *Robotics and Autonomous Systems*, vol. 114, pp. 144-154, 2019, doi: 10.1016/j.robot.2019.01.015.
- [78] R. Zhao, L. Wu, and Y.-H. Chen, “Parallel Robot with Uncertainty: An Online Estimation Approach,” *IEEE Access*, vol. 8, pp. 97604-97617, 2020, doi: 10.1109/ACCESS.2020.2997093.
- [79] C. E. Boudjedir, D. Boukhetala, and M. Bouri, “Nonlinear PD plus sliding mode control with application to a parallel delta robot,” *Journal of Electrical Engineering*, vol. 69, no. 5, pp. 329-336, 2018, doi: 10.2478/jee-2018-0048.