ALGORITHM FOR HUMAN-ROBOT COLLABORATIVE TASKS

Zee Him Phoon, Tee Tiong Tay
Dept. of Mechanical Engineering, Faculty of Engineering, Tunku Abdul Rahman University College, Kuala Lumpur, MALAYSIA

Abstract*:* The most crucial thing that influenced the collaborative robot performance is the program algorithm that is to determine the robot behaviour to perform in the different type of the applications. In this project, the number of the alternative program algorithms have been developed that is collaborative robot algorithm and the speed and separation monitoring algorithm. The collaborative robot algorithm is mainly executed picking and placed task and using the special action to ask for help from the human if necessary. The speed and separation monitoring algorithm is a safety algorithm to fulfil the ISO standard (ISO 10218-1, 5.10.4, ISO/TS 15066) that the operator's position and robot's position and speed are being supervised to prevent the physical contact between human and moving robot. Also, the vision algorithm such as colour tracking algorithm, pixel to centimeter (cm) conversion algorithm, and more. Furthermore, the inverse kinematics algorithm based on the trigonometric mathematical principal. The 5 degrees-of-freedom (DOF) miniature articulated robotic arm is designed and built up for this project. The series of the experiments had been conducted in this project to verify the robot performance based on the developed program algorithm that mentioned above.

Keywords: Collaborative Robot, Human, Video Processing, Algorithm, Robot Performance

1 INTRODUCTION

Robots able to perform high precision and repeatability and handle heavy loads and operate without performance deterioration even in difficult or dangerous environments (Wang, et al., 2017). But they cannot cope with unexpected situations encountered in unstructured, changing "real-world" conditions (Bechar and Edan, 2003). The robot quickly reached its limits in recognizing and handling unexpected situations, as due to rigid plans and robot programs widespread in today's automated systems. In another side, humans tackle unexpected situations better, aware of a much larger part of the environment than formally declared, and show more dexterity in complex or sensitive task, however, humans are more easy to make error, stress or fatigue (Arai , et al., 2010), and their employment underlies strict health and safety regulations (Wang, et al., 2017). Besides, most of the existing industrial robots all over the world require safety fences, because it is not safe to human to walk too close to these industrial robots (Gustavsson, et al., 2017). Thus, the Human-Robot Collaboration (HRC) technology will maximize the advantage of both robot and human to minimize the disadvantage of both of them.

2 PROJECT OBJECTIVE

This project involves the development of a collaborative robot algorithm that robot able to show the flexibility and capability to execute different appropriate tasks when cooperating with a human in the close distance without safety fences. Therefore, the 5 degrees-of-freedom miniature articulated robotic arm prototype will be designed and developed with a built-in embedded controller and associated circuits. In order to test and verify the capabilities of the robot algorithm, the robot will be required to complete a series of experiments to measure its performance based on the program algorithm. For example that robot able to adapt to an undefined situation, unexpected human behaviour and more.

The objectives are:-

1. Develop the collaborative robot algorithm for the robot to human interaction.

2. Design and develop the miniature articulated 5 degrees-of-freedom robotic arm and appropriate end-effector prototype.

3. Investigate and analyse the robot performance base on the program algorithm in a series of experiments.

3 ARTICULATED ROBOT COLLABORATION ALGORITHM

3.1 Robot Behaviour Flow Chart

In Figure 3-1, the robot behaviour consist of the collaborative robot algorithm and the speed and separation monitoring algorithm. The bottom zone is belongs to the collaborative robot algorithm and the top zone is belongs to the speed and separation monitoring algorithm. The collaborative robot algorithm is mainly to execute picking and placed task and using special motion to ask for help from a human if necessary. The speed and separation monitoring algorithm is a safety algorithm that the operator's position and robot's position and speed are being supervised to prevent the physical contact between human and moving robot. The robot has six if-else decision block that deciding the next action should be executed while the robot is working with a human on the collaborative workspace (CWS). The detail of decision blocks will be explained below.

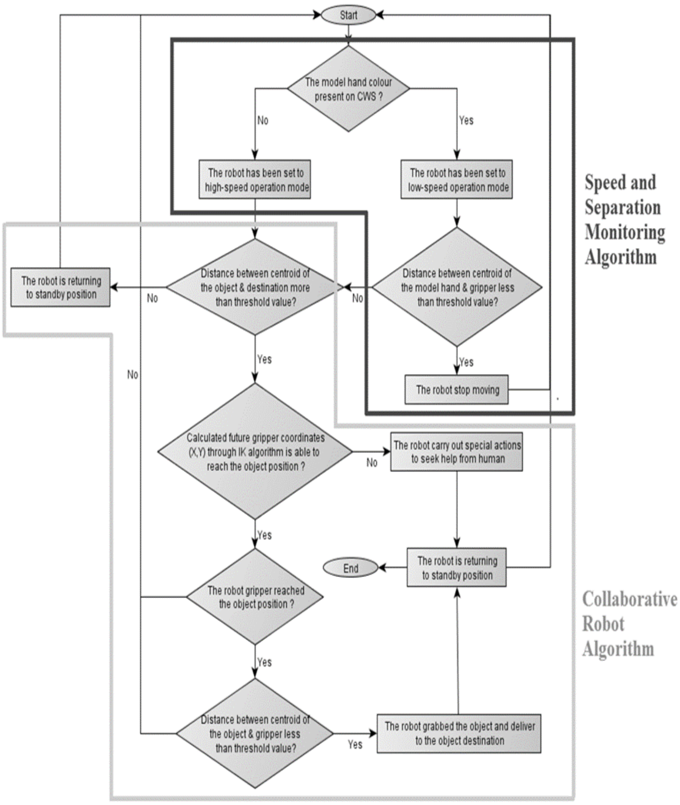

Figure 3-1. Robot Behaviour Flow Chart.

3.2 Collaborative Robot Algorithm Flow Chart

In Figure 3-2, If the separation distance between the centroid of the object and destination (distance square value) above the threshold value that means the targeted object is out of its destination region, then the targeted object coordinates (X, Y) will be sending to microcontroller and processed by the inverse kinematics algorithm. Else if it is below the threshold value that means the targeted object is inside the destination region, the robot will be returning to standby position, then, wait for next video frame. Next, if the separation distance between the centroid of the object and robot gripper (distance square value) below the threshold value that means the robot gripper is close enough to the targeted object location, then the robot will be picking up the targeted object and deliver back to object destination. After the targeted object reached its destination zone, the robot will be returning to standby position, then, wait for next video frame. Else if the separation distance between centroid of the object and robot gripper (distance square value) above the threshold value that means the targeted object is far away from robot gripper, then the robot system will obtain the next video frame to track the targeted object changed position and robot obtained the axis angles needed after object coordinates (X, Y) again processed by inverse kinematic algorithm, the robot gripper will approaching the targeted object again. If the robot gripper is close to the targeted object, the robot pick and place task will be executed. The robot will be returning to the standby position after the task completed, and wait for next video frame.

$$ODDS = (OX - DX)^2 + (OY - DY)^2 \quad (1)$$

Where *ODDS* is the sum square error between the targeted object and object destination, *OX* is the x value of the targeted object, *OY* is the y value of the targeted object, *DX* is the x value of the object destination, and *DY* is the y value of the object destination. The unit is in centimeter.

$$OGDS = (OX - GX)^2 + (OY - GY)^2 \quad (2)$$

Where *OGDS* is the sum square error between the targeted object and robot gripper, *OX* is the x value of the targeted object, *OY* is the y value of the targeted object, *GX* is the x value of the robot gripper, and *GY* is the y value of the robot gripper. The unit is in centimeter.

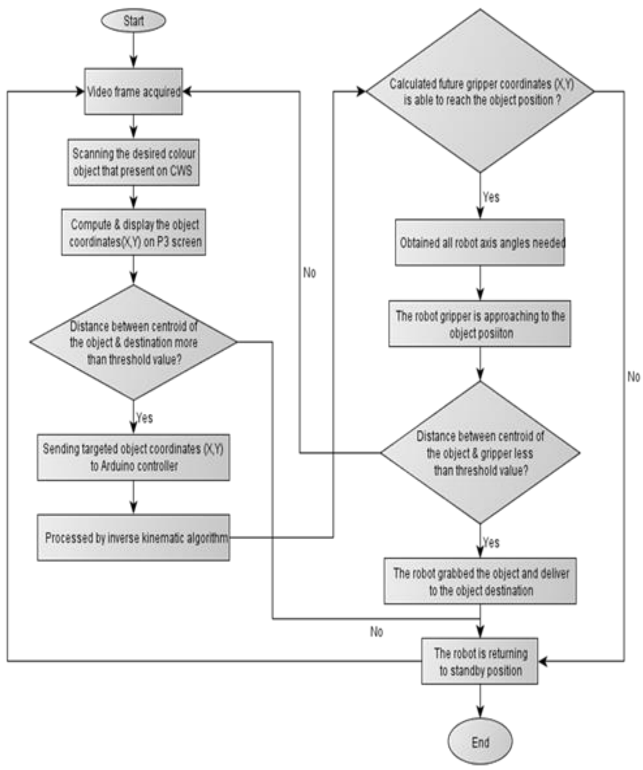

Figure 3-2. Collaborative Robot Algorithm Flow Chart.

3.3 Speed and Separation Monitoring Algorithm Flow Chart

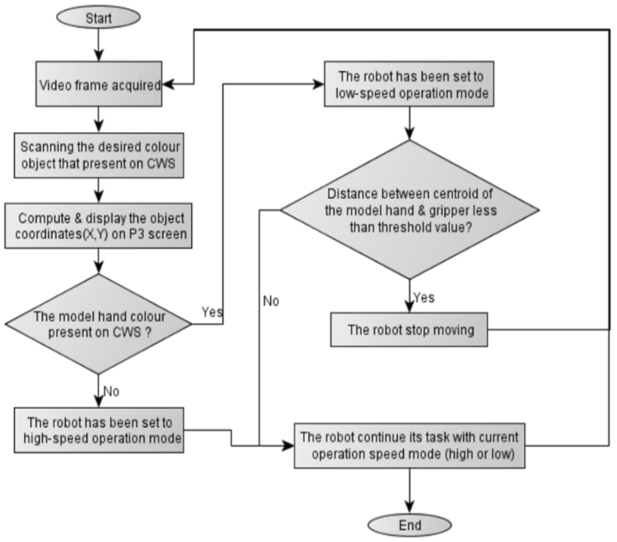

Figure 3-3. Speed and Separation Monitoring Algorithm Flow Chart.

In Figure 3-3, if the model hand is entered the CWS, its colour will be detected by the robot system and the robot operation speed mode will be set to low speed. Then, if the separation distance between the centroid of the model hand and robot gripper (distance square value) below the threshold value that means the model hand is too close to the robot gripper, the robot will stop moving although the drives still energized. Else if the separation distance between the centroid of the model hand and robot gripper (distance square value) above the threshold value that means the model hand is far away from the robot gripper, the robot will continue its task with the low-speed mode. For another scenario, if the model hand is out of the CWS, its colour is not detectable by the robot system, and then the robot operation speed mode will be set to high speed and robot continue its task. The video frame will keep on acquired that updating the status on CWS.

$$HGDS = (MX - GX)^2 + (MY - GY)^2 \qquad (3)$$

Where *HGDS* is the sum square error between the model hand and robot gripper, *MX* is the x value of the model hand, *MY* is the y value of the model hand, *GX* is the x value of the gripper, and *GY* is the y value of the gripper. The unit is in centimeter.

3.4 Vision Algorithm

3.4.1 Colour Tracking Algorithm

To extract tracing object colour properties (RGB) initially. Then, extract the HSB from the object RGB. Compare it with colour (HSB) template by using sum squared error mathematic equation to determine their similarity.

$$HSBDS = (h2 - h1)^2 + (s2 - s1)^2 + (b2 - b1)^2 \qquad (4)$$

Where *HSBDS* is the sum squared error between present HSB (e.g. *h1*, *s1*, and *b1*) and HSB template (e.g. *h2*, *s2*, and *b2*), *h* is hue, *s* is saturation, and *b* is brightness. The unit is in pixel.

3.4.2 Outermost Pixels Filtering Algorithm

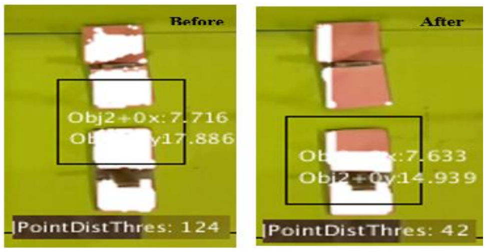

Figure 3-4. Effect of applied the outermost pixels filtering algorithm.

For instance, if the separation distance between the pixel A (current) and pixel B (next) is below the threshold that means they are close enough, both will be stored in an array and pixel B (current) will be comparing with pixel C (next). Else if the separation distance between the pixel A (current) and pixel B (next) is above the threshold that means they are far away from each other, pixel A (current) will be comparing with pixel C (next) and pixel B is ignored. In Figure 3-4, before applied this algorithm, the robot system targeted two objects at the same time that may affect the robot picking process. After the implementation of this algorithm, only one object is targeted instead of two, it removes the unnecessary pixel when the two same color object getting too near.

$$PDS = (NPX - CPX)^2 + (NPY - CPY)^2 \qquad (5)$$

Where *PDS* is the sum square error between the next pixel and current pixel, *NPX* is the x value of the next pixel, *NPY* is the y value of the next pixel, *CPX* is the x value of the current pixel, and *CPY* is the y value of the current pixel. The unit is in pixel.

3.4.3 Centroid of the Colour Pixel Algorithm

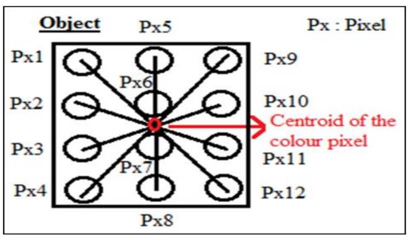

Figure 3-5. Concept of centroid of the colour pixel algorithm.

In Figure 3-5, the vectors have been created that Px1 connects to Px12, Px2 connects to Px11, Px3 connects to Px10, and so on. Then, sum up all the half-way of the vector values and divide by the total number of vectors created to obtain the average value that is the centroid of the colour pixel.

3.4.4 Pixel to Centimeter (cm) Conversion Algorithm

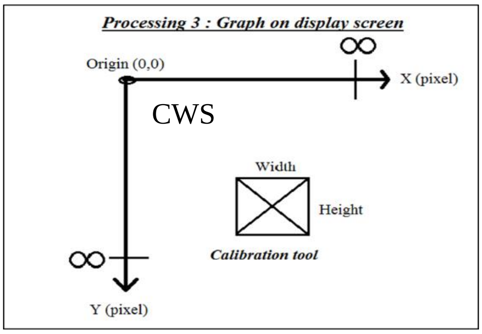

Figure 3-6. Graph on P3's display screen with calibration tool.

In Figure 3-6, this is the default setting of Processing 3 software that X-axis points to the right side direction with positive value and Y- axis points to bottom direction with positive value. The video frame is present with the unit of pixel, hence, the calibration tool is used as reference to convert the unit pixel to the centimeter for later use. Only the value in unit centimeter can be processed by the inverse kinematics algorithm. The coordinates (X, Y) of the targeted object's colour (e.g. model hand, robot gripper, and colourise objects) in CWS are generated by using the equation below:

$$OPX = \frac{TX \times (MXCP - CWSXP)}{CWSWP} \quad (6)$$

Where *OPX* is the x value of object's position in centimeter, *TX* is the width of the CWS in centimeter, *MXCP* is the x value of the centroid of the colour pixel, *CWSXP* is the origin x value of the collaborative workspace (CWS) in pixel, and *CWSWP* is the width of the CWS in pixel.

$$OPY = TY - \frac{TY \times (MYCP - CWSYP)}{CWSHP} \quad (7)$$

Where *OPY* is the y value of object's position in centimeter, *TY* is the height of the CWS in centimeter, *MYCP* is the y value of the centroid of the colour pixel, *CWSYP* is the origin y value of the collaborative workspace (CWS) in pixel, and *CWSHP* is the height of the CWS in pixel.

3.4.5 Object Undetectable Region Algorithm

This algorithm to ensure that once the objects reached their destination, they are undetectable to save the processing power and reduce the potential errors. The undetectable region equation is shown below:

$$\mathrm{TL} = \frac{(\mathrm{STL} - \mathrm{TY}) \times \mathrm{CWSHP}}{(-\ \mathrm{TY})} + \mathrm{CWSYP} \quad (8)$$

Where *TL* is the top line in pixel, *STL* is the set top limit in centimeter, *TY* is the height of the CWS in centimeter, *CWSYP* is the origin y value of the collaborative workspace (CWS) in pixel, and *CWSHP* is the height of the CWS in pixel.

$$\mathrm{BL} = \frac{(\mathrm{SBL} - \mathrm{TY}) \times \mathrm{CWSHP}}{(-\ \mathrm{TY})} + \mathrm{CWSYP} \quad (9)$$

Where *BL* is the bottom line in pixel, *SBL* is the set bottom limit in centimeter, *TY* is the height of the CWS in centimeter, *CWSYP* is the origin y value of the collaborative workspace (CWS) in pixel, and *CWSHP* is the height of the CWS in pixel.

The "STL" and "SBL" are pre-determined by the user. If the destination of an object is on top of the CWS, if it above the top line, it will be unable to be detected. It able to be detected if it stay below the top line. Moreover, if the destination of an object is on the bottom of the CWS, if it below the bottom line, it will be unable to be detected. It able to be detected if it stays above the bottom line. The example shown in Figure 3-7.

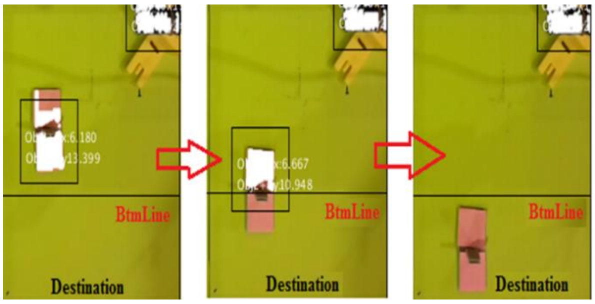

Figure 3-7. Example of "BtmLine" undetectable region.

3.4.6 Vision Algorithm Flow Chart

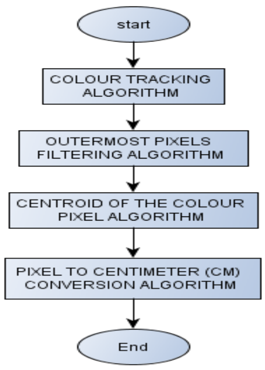

Figure 3-8. Vision Algorithm Flow Chart.

In Figure 3-8, the vision algorithm work flow is from colour tracking algorithm to the pixel to centimeter (cm) conversion algorithm, the final output is ready to be processed by the inverse kinematic algorithm.

3.5 Inverse Kinematic Algorithm

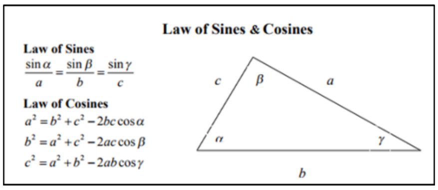

Figure 3-9. Law of Sines and Cosine formula.

This algorithm is created by using the trigonometric mathematical principal including the law of cosine and sine (Figure 3-9) to find each of the robot axis angle needed to approach to the object position after received the input data from pixel to centimeter (cm) conversion algorithm. During the process of the inverse kinematic algorithm, the final robot angle axis to reach object position is not obtained immediately, there is mathematical simulation needed to calculate each of the robot axis angles with some iteration until the coordinates (X,Z,Y) of the robot gripper is meet the required coordinates (X,Z,Y). For instance, the robot axis 4 and 1 will be the first to rotate and this is one literation. The second literation is rotate the robot axis 3, then, the third literation is rotating robot axis 2 and repeat back to rotate robot axis 4 and 1 again. The process keeps repeating until the all robot axis angle meet the required coordinates. The process above is just using mathematical model to calculate the future angle of each robot axis can be reach to. After all the robot axis angles meet the required coordinates, the all final robot axis angles data will be sent to servo driver to drive the servo motor to desired angles.

4 EXPERIMENTAL SETUP

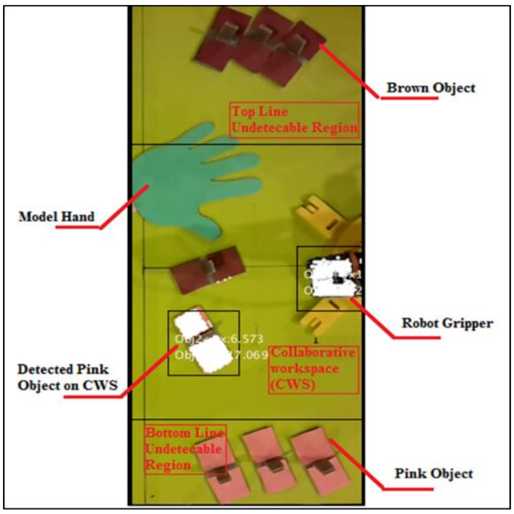

Figure 4-1.Camera view in Processing 3 software.

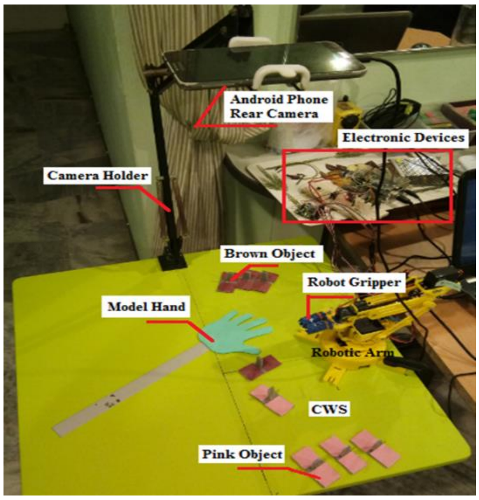

Figure 4-2. Real world view of experimental setup.

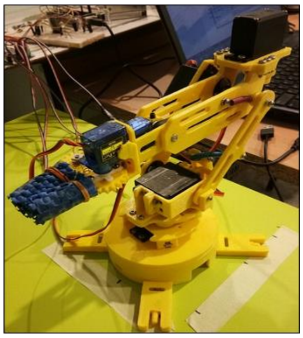

Figure 4-3. 3D printed 5 degrees-of-freedom miniature articulated robotic arm.

In Figure 4-1, the bottom line undetectable region is a pre- determined destination for pink object and for the top line undetectable region, it is a destination belongs to brown object. The collaborative workspace (CWS) is the region that robot system able to trace the objects if the objects are not stay in their destination zone such as the pink object is detected on CWS by the robot system. The robot gripper is keep tracing constantly under robot system. The model hand is represented as a human hand. In Figure 4-2, the Android phone rear camera is used to capture all the objects that present on the CWS. The electronic devices including the Arduino Mega 2560 microcontroller, 12V power supply and SC08A servo controller. In Figure 4-3, 3D printed 5 degrees-of-freedom miniature articulated robotic arm used in this project.

5 RESULT AND ANALYSIS

5.1 Experiment 1: Pick and Place Random Objects

In experiment 1, if the tracing object colour is far away from its own destination zone and exceeds the threshold limit. The robot system will received the object position coordinates (X, Y), and obtained the robot axis angles needed through an inverse kinematic algorithm, the robot gripper will reached to the object position. Then, the object is picked up and placed to predetermined destination. If the object is not detectable and still on collaborative workspace (CWS), the robot will perform special motions to get attention from the human to seek help.

5.2 Experiment 2: Attention Seeking

In experiment 2, if the tracing object colour is invisible or loses detection under the robot system, the robot will be conduct on these actions that returning to the robot standby position and approaching to the object position and repeat the actions until the human noticed, this special robot motion is to deliver a message to the human that the robot needs a help from the human.

5.3 Experiment 3: Random Object Tracking

In experiment 3, the human purposely and randomly changing the object position inside CWS causes the separation distance between centroid of the robot gripper and detected object is above the threshold value, the robot system will be receive the latest object coordinates (X, Y) and send again to Arduino Mega 2560 microcontroller board, triggered the inverse kinematic algorithm to obtain the robot axis angles needed to allow robot gripper to reach closer to the object position. The robot will keep approaching the object based on coordinates (X, Y) until the object is delivered back to its destination, even though the object position is changing frequently by the human.

5.4 Experiment 4: Speed and Separation Monitoring

In experiment 4, if model hand entered CWS, the robot move at low speed. Else the robot moves at high speed. Moreover, if the separation distance between the centroid of the robot gripper and model hand is below the threshold value that means they are too close to each other, the robot will stop moving immediately. Else, the robot can move at low speed since the model hand still remaining in CWS.

6 LIMITATION

There are some limitation for this robot system, the robot is unable to reach if the tracking object is out of the CWS because of the mechanical structure limit of the robotic arm. Since only 2-dimensional coordinates (X, Y) data is available, the Z coordinates data is pre-determined by the user, this causes some small area on CWS that robot is unreachable after calculated by the inverse kinematic algorithm.

7 FUTURE WORK

The projected camera need to be captured not only the X-Y plane information, also the Z plane information. because of the high complexity to obtain XYZ plane information to feedback three-dimensional coordinates (X, Y, Z) to the robot system, hence, this project only used 2-dimensional coordinates (X, Y) data. The three-dimensional coordinates allows robot moves in X-Y-Z direction easily after processed by the inverse kinematics algorithm to increase robot movement flexibility. Next, the robotic arm need to build in bigger size and upgraded from 5 to 6 degrees-of-freedom as a standard robotic arm in the market and replace current servo motors with higher torque drives to enhance the robot capability and payload. Moreover, the artificial intelligence technologies will be applied in the robot system that the threshold values can be adjusted automatically based on the input data received.

8 CONCLUSION

The crucial program algorithms have been developed that are collaborative robot algorithm, the speed and separation monitoring algorithm and vision algorithm. The 5 degrees-of-freedom miniature articulated robotic arm had been designed by utilized the 3D CAD software and used additive manufacturing to print out the real robotic arm. The robot able to work with the human in close distance because of the speed and separation monitoring algorithm. Also, the robot can be adapting the object position changed on CWS that human purposely random placing the object on CWS, still, the robot able to keep tracing and approach to the desired object based on object colour with the help of vision algorithm and inverse kinematic algorithm. Then, the robot can pick up the object and place in its destination accurately and conduct the special motion to ask for help if necessary.

ACKNOWLEDGMENTS

I feeling grateful that my supervisor Tay Tee Tiong giving useful comments, idea, and encouragement through the learning process of this Bachelor of Engineering (Honours) Mechatronic final year degree thesis. Besides, I would like to express gratitude toward Tay Tee Tiong for introducing me to the project topic that I interested in. I would like to thank my loved family and friends who have supported me throughout the entire process, they bring positive energy to me. Thus, I have motivated to complete this degree thesis and overcome incoming problems with brave.

REFERENCE

Arai T, Kato R, Fujita M (2010) Assessment of Operator Stress Induced by Robot Collaboration in

Assembly. CIRP Annals—Manufacturing Technology 59(1).

Bechar A. and Edan Y. (2003). Human-Robot Collaboration for improved target recognition of agricultural robots. Industrial Robot, 30(5).

Gustavsson, P., Syberfeldt, A., Brewster, R., & Wang, L. (2017). Human-Robot Collaboration Demonstrator Combining Speech Recognition and Haptic Control. Procedia CIRP, 63.

Wang, X. V., Kemény, Z., Váncza, J., & Wang, L. (2017). Human–robot collaborative assembly in cyber-physical production: Classification framework and implementation. CIRP Annals-Manufacturing Technology.