International Journal of Engineering & Technology, *7 (4.27) (2018) 50-56*

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Robotic Arm System with Computer Vision for Colour Object Sorting

Ong Kok Meng[1*], Ong Pauline, Low Ee Soong[2], Sia Chee Kiong[3]

[1 2] Faculty of Mechanical and Manufacturing Engineering, Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, Malaysia.
**Corresponding author E-mail: ongkokmeng93@hotmail.com*

Abstract

This study presents the development of robotic arm with computer vision functionalities to recognise the objects with different colours, pick up the nearest target object and place it into particular location. In this paper, the overview of the robotic arm system is first presented. Then, the design of five-degrees of freedom (5-DOF) robotic arm is introduced, followed by the explanation of the image processing technique used to recognize the objects with different colours and obstacle detection. Next, the forward kinematic modelling of the robotic arm using Denavit-Hartenberg algorithm and solving the inverse kinematic of the robotic arm using modified flower pollination algorithm (MFPA) are interpreted. The result shows that the robotic arm can pick the target object accurately and place it in its particular place successfully. The concern on user safety is also been taken into consideration where the robotic arm will stop working when the user hand (obstacle) is detected and resume its process when there is no obstacle.

Keywords: *Robotic Arm; Image Processing Technique; Forward Kinematic; Denavit-Hartenberg Algorithm; Modified Flower Pollination Algorithm*

1. Introduction

Robots – machines that are programmed to perform repetitive, complex, dull, dirty and dangerous tasks, have been integrated with success in various forms, for instance, arc welding, pick and place in assembly line, handling toxic chemicals in manufacturing, and carrying radioactive materials in hazardous environments (Sajjad, Talpur, & Shaikh, 2012) (Zhang & Wang, 2004). This can help improve the quality and efficiency of productivity, and address the issue of manpower saving. Besides that, robot can also work in any hostile and harsh environment, which may be not accessible to human (Sajjad et al., 2012).

In industry, the assembly of discrete products is done through Pick and Place (PNP) operation. The robotic arm picks the components and places it in its respective position (Taylor, 2007). In PNP operation, the gripper of the robot is placed in the pickup position, and subsequently, the gripper starts to grab the object. The robot moves the gripper to the placement position, and then releases the object from gripper (Moghaddam & Nof, 2016). For instance, robotic surgery (Dhepekar & Adhav, 2017), packaging (Y. Li, Huang, & Chetwynd, 2018), palletizing (Abdou & Lee, 1991), loading/unloading of machines (Gu et al., 2015), goods sorting (Dharmannagari, 2014), and warehousing (Culler & Long, 2016), are some examples of the utilisation of PNP operation in real-world application.

The commonly used robotic arm in industry, however, applies Point to Point (PTP) trajectory. In PTP trajectory, the point where the arm picks up an object and the point where it releases the object is known. There are very few industrial robots are embedded with intelligent decision-making system (Rai, Rai, & Rai, 2014). In order to embrace the Industry 4.0, where all processes within the industry are fully automated for mass customization, a robot that exhibits intelligent behaviour as human is needed. Therefore, this is the motivation of this study in which a robotic arm that can automatically recognize and sort the objects into different classes is developed for a PNP operation, in conjunction with the integration of computer vision and evolutionary computation.

In robotic arm, the rotating angle of each joint is determined using inverse kinematics so that the end-effector can be located in the targeted position. However, infinite number of feasible joint configurations are possible when applying the inverse kinematics to solve the end-effector position. To solve the inverse kinematics problem more efficiently, Nearchou utilized modified genetic algorithm for unravelling the inverse kinematics problem in this regard (Nearchou, 1998). A biomimetic approach has been proposed by Artemiadis et al. to correctly locate the position of end-effector (Artemiadis et al., 2010). In addition, Li et al. applied recurrent neural network in solving kinematic control problem of the robotic arm (S. Li et al., 2012). Besides that, Starke proposed hybrid genetic swarm algorithm to solve the inverse kinematics problem (Starke, 2016). Results showed that the algorithm is robust in redundant manipulator with 30, 90 and 180 degree of freedom.

In computer vision, extracting meaningful information from an image is important prior to performing a specific task. Due to the low cost with high processing technology of vision sensors, the computer vision field is rapidly grown (Reddy & Nagaraja, 2015). Recently, learning based methods are also commonly used in computer vision based applications, such as controlling processes, navigation, detecting events, modelling objects or environments (Lampert, Nickisch, & Harmeling, 2009; Saxena, Driemeyer, & Ng, 2008; Martinez-De Dios & Ollero, 2015; Brunetti et al., 2018; X. Li et al., 2018).

The similar trend is observed for the integration of computer vision in the applications of robotic arm system. For example, Hayashi et al. developed three types of harvesting robot, namely strawberry-harvesting robot, eggplant-harvesting robot and tomatoharvesting robot in which the computer vision has been applied to determine the ripeness of fruits (Hayashi et al., 2005). Mehta developed a robot based on vision control in harvesting citrus automatically (Mehta, 2007). An autonomous fruit harvester with machine vision built by Almendral et al. has shown the capability to detect and cut an orange from tree (Almendral et al., 2018). Not to mention that, robotic arm with vision system as feedback control can be applied in space applications in future (Sangeetha et al., 2018). The successful implementation of the computer vision in a robotic arm system available in literature has spurred the development of the robotic arm in this study, particularly the computer vision is integrated for object sorting in a PNP operation.

The paper is organized as follows. An overview of the robotic arm system is first presented, followed by the design of the robotic arm. After that, the image processing technique used to recognize the objects with different colours and detect obstacle is explained, followed by the explanation of the Denavit-Hartenberg algorithm used in modelling the forward kinematic of the robotic arm and modified flower pollination algorithm (MFPA) used in solving the inverse kinematic of the robotic arm in PNP operation. Experimental simulations are then presented and lastly, some conclusions are drawn.

2. System Overview

The robotic arm with computer vision functionalities developed in this study is shown in Figure 1. It consists of two major parts, specifically, a 5-degree of freedom (DOF) robotic arm and object recognition based on computer vision. Initially, the robotic arm is triggered and it moves to its initial position, as shown in Figure 2. The webcam mounted on a side pole is used to snapshot the layout of the objects in workspace. From the captured image, the positions of all objects are located, and the colour of each object is identified, either in red, green or blue colour. The robotic arm selects and grips the object with the shortest distance to its body. The MFPA is used to determine the optimal angles of robotic arm's joints such that the gripper is able to grip the selected object precisely. The angles of the robotic arm's joints in initial position are assigned as the starting angles, whereas the ending angles of robotic arm's joints depend on the colour of selected object, *i.e*, the position of the container for red, green and blue object.

From the starting angles, the robotic arm is moving slowly, with every 4° is added continuously to the angle of five servo motors, until the ending angles are reached. In this case, the object is released from the gripper, and placed in its corresponding position according to its colour, as shown in Figure 1. Occasionally, human hand is identified by the webcam. The robotic arm stops operating and enters stopping mode (looping) as long as the human hand is detected. The rotation of the servo motor resumes, when there is no occlusion been detected. When all objects have been picked and sorted, the robotic arm is not in use and enters the sleeping mode. The detailed flowchart of the system is shown in Figure 3 and Figure 4.

3. The Designed 5-DOF Robotic Arm

The 5-DOF robotic arm is designed with five servo motors are attached on the arms and one servo motor is attached on the gripper, as shown in Figure 5. The specifications of servo motors are shown in Table 1. The servo motor at second arm requires high torque as it needs to support the rest of the arms, except the first arm. The servo motor has feedback control loop circuitry with no motor drivers are required. The feedback control loop circuitry allows the servo motor to control the angular position accurately.

The robotic arm can grip the objects with the size up to 40mm x 40mm x 40mm with the weight up to 200 grams. Furthermore, the gripper is designed in such a way that it can be changed and attached to the arm in order to grab objects with different shapes. The arm has reachable area of 40cm, as shown in Figure 6. The 3D drawing of the robotic arm is presented in Figure 7.

The body of robotic arm is fabricated using 3D printing technology, as shown in Figure 8. The material used to print the parts of the robotic arm is polylactic acid (PLA). The 3D printing technology is chosen to build the robotic arm as it is fast and the material is light.

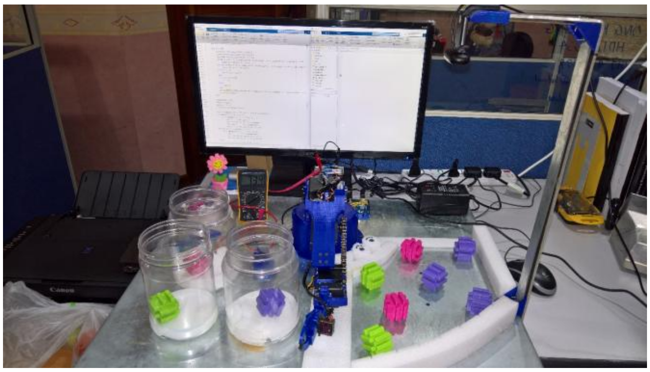

Fig 1: Robotic arm with computer vision functionalities

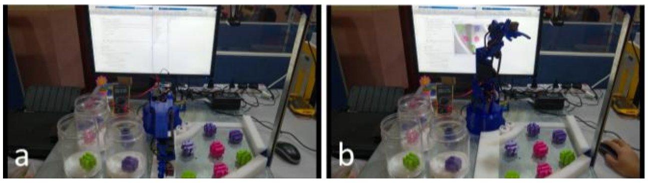

Fig 2: Robotic arm in operation. (a) Sleeping stage of robotic arm. (b) Wake-up stage of robotic arm. It is also known as its initial position.

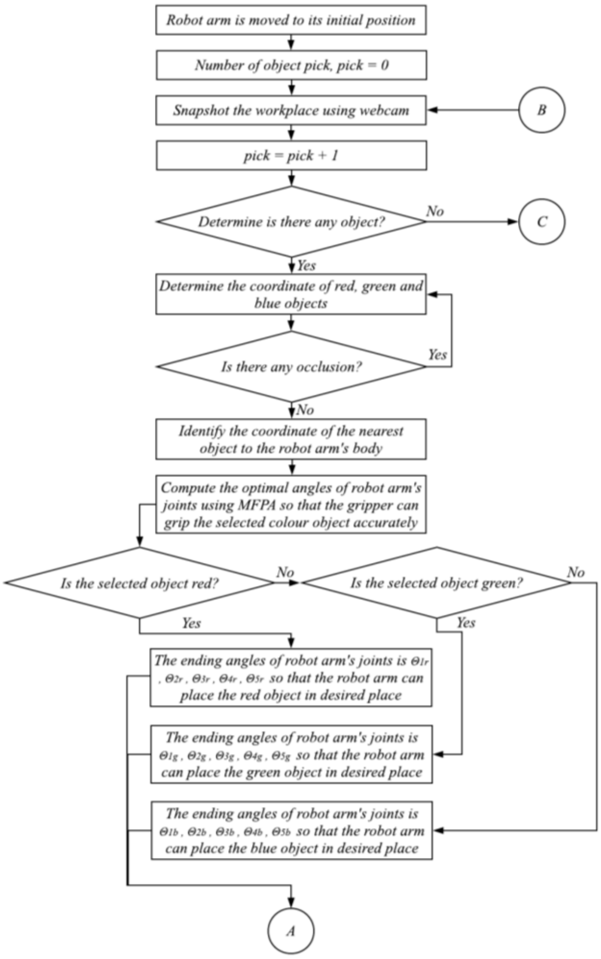
Robot arm is moved to its initial position
Number of object pick, pick = 0
Snapshot the workplace using webcam
B
pick = pick + 1
Determine is there any object?
No
C
Yes
Determine the coordinate of red, green and blue objects
Is there any occlusion?
Yes
No
Identify the coordinate of the nearest object to the robot arm's body
Compute the optimal angles of robot arm's joints using MFPA so that the gripper can grip the selected colour object accurately
Is the selected object red?
No
Is the selected object green?
No
Yes
Yes
The ending angles of robot arm's joints is Θ_{1r}, Θ_{2r}, Θ_{3r}, Θ_{4r}, Θ_{5r} so that the robot arm can place the red object in desired place
The ending angles of robot arm's joints is Θ_{1g}, Θ_{2g}, Θ_{3g}, Θ_{4g}, Θ_{5g} so that the robot arm can place the green object in desired place
The ending angles of robot arm's joints is Θ_{1b}, Θ_{2b}, Θ_{3b}, Θ_{4b}, Θ_{5b} so that the robot arm can place the blue object in desired place
A

Fig. 3: Flow chart of robotic arm with computer vision

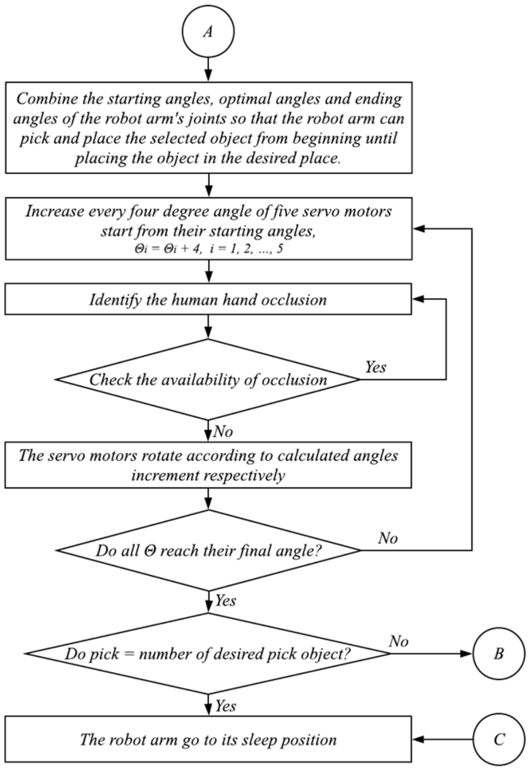
A
Combine the starting angles, optimal angles and ending angles of the robot arm's joints so that the robot arm can pick and place the selected object from beginning until placing the object in the desired place.
Increase every four degree angle of five servo motors start from their starting angles,
Θi = Θi + 4, i = 1, 2, ..., 5
Identify the human hand occlusion
Check the availability of occlusion
Yes
No
The servo motors rotate according to calculated angles increment respectively
Do all Θ reach their final angle?
No
Yes
Do pick = number of desired pick object?
No
B
Yes
The robot arm go to its sleep position
C

Fig. 4: Flow chart of robotic arm with computer vision (Cont.)

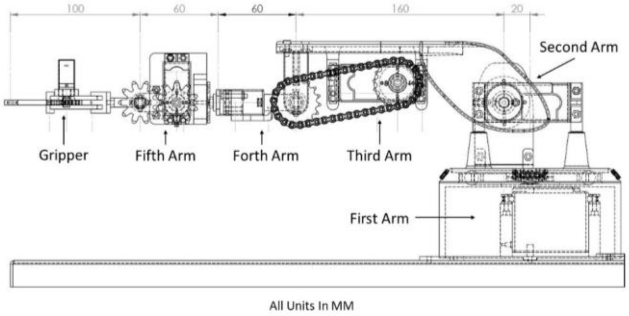

Fig. 5: A 5-DOF robotic arm

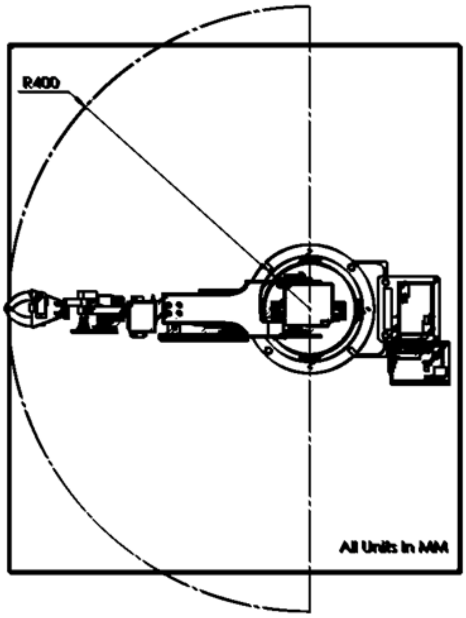

Fig. 6: Reachable Area of Robotic Arm

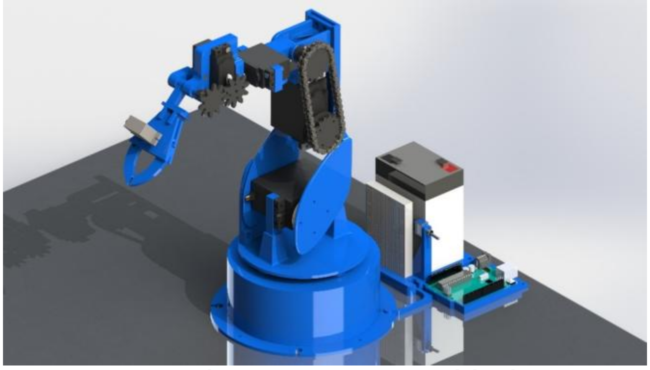

Fig. 7: Robotic Arm 3D Drawing in Solidworks

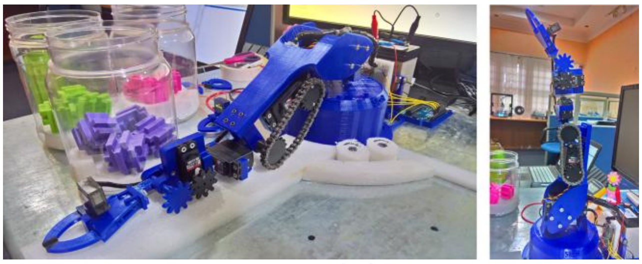

Fig. 8: 3D printed robotic arm

Table 1: Specification of servo motors

Axis Capabilities: Mechanical Assembly	Maximum Angle	Motor Torque, *kg cm*	Voltage Required For Motor operation, *volt*	Motor Weight, *gram*
Servo Motor at First Arm	180°	11.0 ~ 13.2	4.8	110
Servo Motor at Second Arm	180°	19.8 ~ 24.7	4.8	197
Servo Motor at Third Arm	180°	11.0 ~ 13.2	4.8	110
Servo Motor at Forth Arm	180°	3.3 ~ 4.1	4.8	46
Servo Motor at Fifth Arm	180°	3.3 ~ 4.1	4.8	46
Servo Motor at Gripper	180°	2.20	4.8	9

4. Computer Vision for Colour Recognition

Computer vision is important in this system as it allows the robotic arm to locate the position of selected object as well as recognize its colour. Since the height of the object is known, the information about z-coordinate of the object is no longer needed. Therefore, computer vision in this system is used to calculate the x-coordinate and y-coordinate of the objects in the workspace. In this section, the image processing techniques used to differentiate the colour of the object and hand detection are first presented, and continued by the explanation of finding the nearest object to the robotic arm's body.

The image is captured through external webcam which is mounted on the side of the workspace, as illustrated in Figure 1. This image shown in Figure 9 is transferred to computer for image processing in order to identify the position of objects with different colours. For example, to locate the centroid coordinate of red coloured object, the red components in the image are extracted from the grayscale image, as shown in Figure 10 (b). Subsequently, median filter is used to filter out the noise, as shown in Figure 10 (c). The resulting grayscale image is then converted into a binary image as illustrated in Figure 10 (d). Lastly, those pixels less than 300 pixels are removed from the binary image, as shown in Figure 10 (e). Red colour objects are detected and its centroid coordinate is shown in Figure 10 (f).

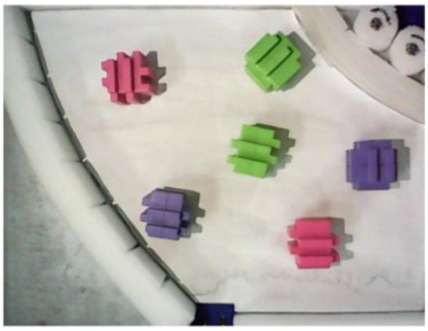

Fig. 9: Color object in workspace

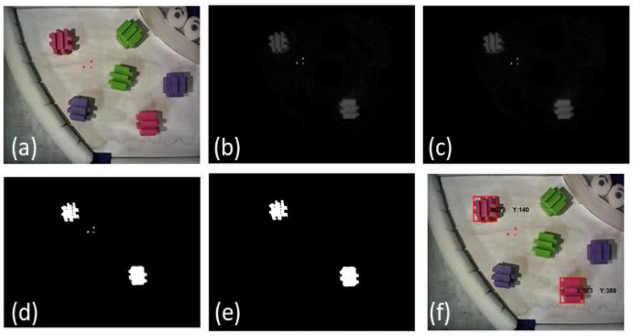

Fig. 10: Red color detection using Image processing technique

If hand is detected, one extra step is performed prior to the extraction of the colour components from the grayscale image. Specifically, each pixel value in RGB image is subtracted from the maximum pixel value, and the difference is used as the pixel value in the output image. In this regard, the dark areas become lighter and light areas become darker in the output image, as presented in Figure 11 (b). The following steps are same as the steps described previously. As shown in Figure 11 (g), the hand is detected in the workspace and the robotic arm will stop functioning until no hand is detected.

After identifying the centroid coordinate and colour of the objects, the Euclidean distance between the centroid coordinate of each object and the robotic arm's body is computed, as illustrated in Figure 12. The object with the shortest distance to robotic arm's body is selected as the target, and will be sorted according to its colour.

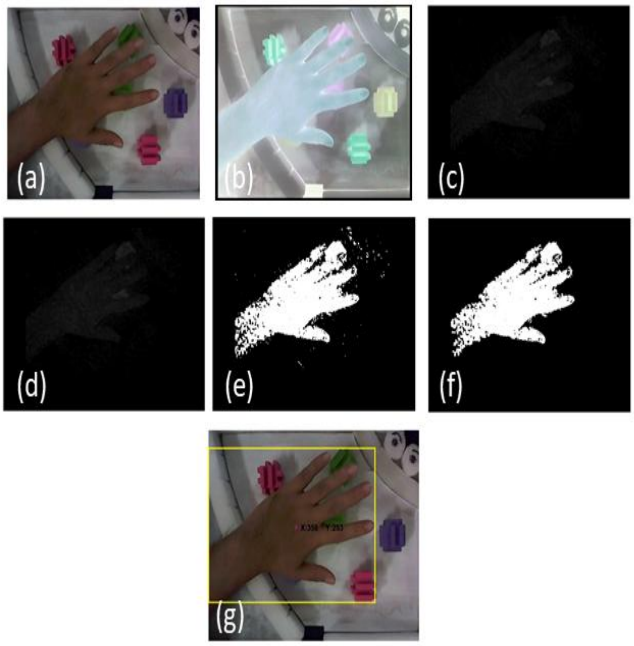

Fig. 11: Hand Detection using image processing technique

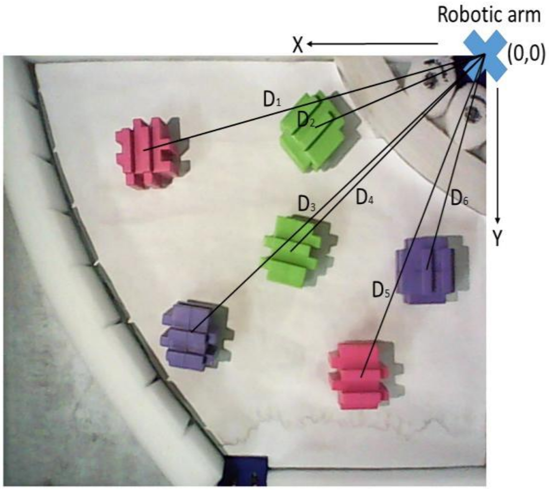

Fig. 12: The distance of each color object to robotic arm's body

5. Control Strategy of Robotic Arm

Due to promising result of MFPA in engineering problems (Ong, Ong, & Sia, 2016), the MFPA is used to find the optimal angle of each joint such that the robotic arm can reach and pick the target object accurately(Ong et al., 2016). Prior to optimizing the joints' angle using the MFPA, a mathematical model is developed through forward kinematic model, particularly, the Denavit-Hartenberg (DH) algorithm is utilized. The mathematical model is used to determine the position of end effector (gripper) from those five specific values of joint angles.

5.1. Forward Kinematic Model

Figure 13 presents a simplified kinematic representation of the 5-DOF robotic arm. The DH parameters are obtained through the simplified kinematic representation of the robotic arm as shown in Table 2. Then, the coordinate transformation matrix H_0^1, H_1^2, H_2^3, H_3^4, H_4^5 for each link is calculated using Equation (1) from the DH parameters. After that, the coordinate transformation matrix H_0^5 between the world coordinate system (x_0, y_0, z_0) and the end-effector coordinate system (x_5, y_5, z_5) are calculated using Equation (2). Finally, Equation (3) is used to determine the coordinate (x, y, z) of the end-effector, when there are inputs θ_1, θ_2, θ_3, θ_4 and θ_5 . Equation (3) is also known as forward kinematic model of this 5-DOF robotic arm.

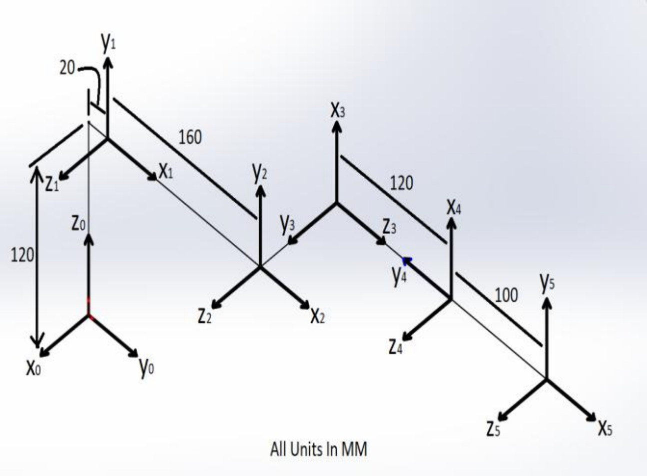

Fig. 13: Arm kinematic representation showing frame assign

Table 2: Denavit-Hartenberg parameters

Link	θi	di	ai	αi
1	θ1+90	120	20	90
2	θ2	0	160	0
3	θ3+90	0	0	90
4	θ4	120	0	-90
5	θ5-90	0	100	0

$$H_{i-1}^{i}=\begin{bmatrix}\cos(\theta_i) & -\cos(\alpha_i)\cdot\sin(\theta_i) & \sin(\alpha_i)\cdot\sin(\theta_i) & a_i\cdot\cos(\theta_i)\\ \sin(\theta_i) & \cos(\alpha_i)\cdot\cos(\theta_i) & -\sin(\alpha_i)\cdot\cos(\theta_i) & a_i\cdot\sin(\theta_i)\\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i\\ 0 & 0 & 0 & 1\end{bmatrix},$$

$$i=1,2,\dots,5 \tag{1}$$

$$H_0^5=H_0^1\cdot H_1^2\cdot H_2^3\cdot H_3^4\cdot H_4^5 \tag{2}$$

$$\begin{bmatrix}x\\ y\\ z\\ 1\end{bmatrix}=H_0^5\cdot\begin{bmatrix}0\\ 0\\ 0\\ 1\end{bmatrix} \tag{3}$$

5.2. Modified Flower Pollination Algorithm

From the previous section, the forward kinematic (FK) model of the robotic arm is developed. With inputs θ_1, θ_2, θ_3, θ_4 and θ_5, the (x, y, z) coordinates of end effector are obtained through FK model, as shown in Figure 14. Since the (x, y, z) coordinates of the target object are known, the optimal values of θ_1, θ_2, θ_3, θ_4 and θ_5 can be computed in order for the robotic arm to grip the target object accurately without collision with obstacle. In this case, the MFPA is used to determine the optimal values of θ_1, θ_2, θ_3, θ_4 and θ_5. Figure 15 presents the flow of using the MFPA in optimizing the joints' angle.

In short, determining a collision-free configuration of 5-DOF robotic arm can be formulated as an optimization problem as:

Minimize

$$\text{Fitness}=\text{Position Error}+\text{coll} \tag{4}$$

Where

$$\text{Position Error}=\sqrt{(x_e-x_g)^2+(y_e-y_g)^2+(z_e-z_g)^2} \tag{5}$$

with the coordinate of end effector is given by (x_e, y_e, z_e), and the coordinate of desired position is specified by (x_g, y_g, z_g).

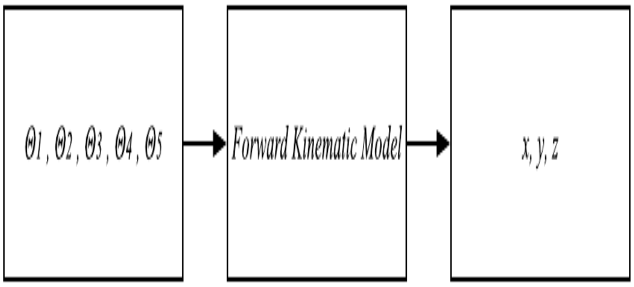

Fig. 14: FK Model

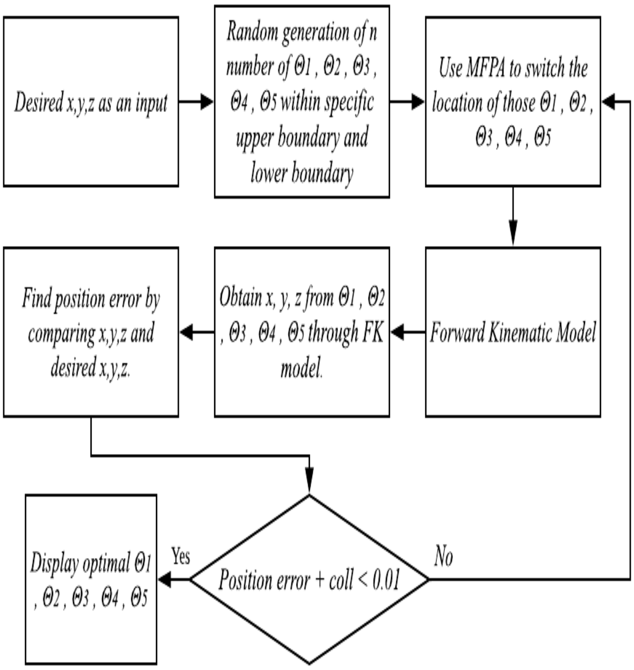
Desired x,y,z as an input
Random generation of n number of Θ1 , Θ2 , Θ3 , Θ4 , Θ5 within specific upper boundary and lower boundary
Use MFPA to switch the location of those Θ1 , Θ2 , Θ3 , Θ4 , Θ5
Forward Kinematic Model
Obtain x, y, z from Θ1 , Θ2 , Θ3 , Θ4 , Θ5 through FK model.
Find position error by comparing x,y,z and desired x,y,z.
Position error + coll < 0.01
Yes
No
Display optimal Θ1 , Θ2 , Θ3 , Θ4 , Θ5

Fig. 15: Flow of using MFPA in finding θ_1, θ_2, θ_3, θ_4 and θ_5

Considering the requirement for the robotic arm to avoid the obstacle (floor of the work place in this case), the safety factor, coll, is introduced and defined as:

$$coll=\begin{cases}0, & \text{safe}\\ 10^{15}, & \text{collide}\end{cases} \tag{6}$$

Next, the detail flow of how MFPA can be applied to solve the optimization problem of the robotic arm is described as follows:

Step 1: Parameters Initialization

First, the parameters of population size, *n*, dimension of search space, *d*, maximum iteration, *max_iter*, switch probability, *p*, range of search space, *[Lb, Ub]*, number of memeplexes, *m* and iterations with each memeplex, *it*, are initialized.

Step 2: Generate Initial Population based on Chaos Theory

After that, the initial population is generated using chaos theory based on circle map with

$$x_{n+1}=(x_n+0.2-(0.5/2\pi)\sin(2\pi x_n))\cdot\mathrm{mod}(1) \tag{7}$$

$$X_{ij}=Lb+(Ub-Lb)x_{ij} \tag{8}$$

where x_{n+1} is the generated solution by circle map, while X_{ij} is the remapped initial population based on lower bound and upper bound of design variables. The best solution g_1 is then identified and its fitness value is evaluated.

Step 3: Information Sharing based on Shuffled Frog Leaping Algorithm

The entire population will undergo the local search using the shuffled frog leaping algorithm (SFLA). The n solutions are divided into m tribal groups, with p solutions in each group. For each tribal group, the location of the worst solution in i tribal group is updated with:

$$x_{wi}{}^{(t+1)}=x_{wi}{}^{(t)}+rand(x_{bi}{}^{(t)}-x_{wi}{}^{(t)})-s_{\max} \tag{9}$$

where rand is a random number chosen between 0 and 1, x_{bi} is best solution in *i* tribal group, x_{wi} is the worst solution in *i* tribal group, and s_{max} is the maximum step length. The location updating process is repeated until termination condition is achieved.

Step 4: Perform Global Search using Lévy Flight with Adaptive Step Size Strategy

Following from the local search using SFLA, if a generated random number is greater than the switching probability, p (rand>p), the global search using

$$x_i^{t+1} = x_i^t + wL(x_i^t - g_*) \tag{10}$$

takes place. L is Lévy distribution while w is the inertia weight determined from

$$w = w_{\min}\left(1+\left(w_{\max}/\sqrt{t}\right)\cdot\tanh\left(g_*/g_1\right)\right) \tag{11}$$

where t denotes the current iteration number and g* is the current fitness value. If rand<=p, the solution will remain in its position. The introduction of inertia weight into Lévy flight can expand the MFPA search space at the early phase so that it will not easily get trapped in local minima. It also increases the convergence rate at the later phase due to the value of w decreases as the iteration increases.

Step 5: Update new solution

For each of the newly generated solution, the corresponding fitness value is calculated from the objective function. If the new solution is found to be better, it replaces the solution from previous generation. Subsequently, the best solution g* is updated.

Step 6: Stopping criterion

The algorithm terminates when it reaches the pre-determined maximum number of iteration.

In a nutshell, the MFPA is used to determine the joint angle of each link joint of a robotic arm so that the end-effector of the robotic arm can reach the desired position accurately, without collide with any obstacle during its movement. Equation (4) is formulated as the fitness function of MFPA.

6. Result Analysis

The developed 5-DOF robotic arm with computer vision functionalities is used to pick and place the objects with different colours in the particular places. As shown in Figure 16 (a), the colour of each object is differentiated accurately, indicated by the colour of the rectangular box. Subsequently, the robotic arm will start to pick the nearest object, and sort it according to its colour. Figure 17 shows the PNP operation performed using the developed robotic arm. It can be observed that with the aid of computer vision and optimization of joints' angle using the MFPA, the robotic arm is able to pick the targeted object precisely and place it to the specific place correctly. The robotic arm picks and places the objects properly through iterative process, regardless of their positions, colour and orientations. In addition, the robotic arm exhibits obstacle avoidance behaviour. It stops automatically when occlusion is detected and resumes working when the occlusion is missing, as shown in Figure 16 (b).

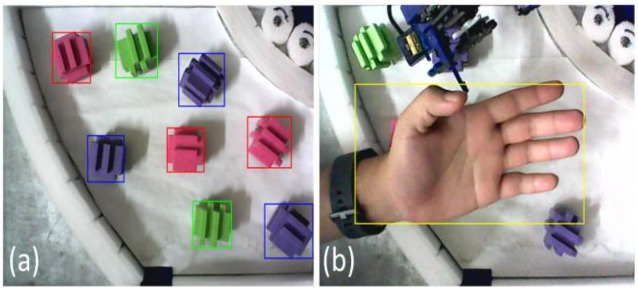

Fig. 16: Image from workspace. (a) Color differentiation by computer vision. (b) Obstacle detection

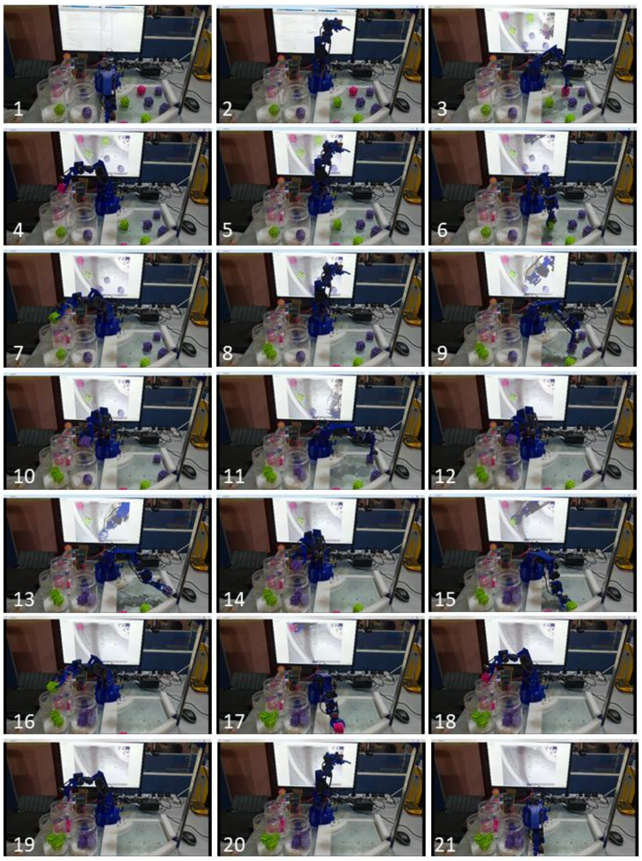

Fig. 17: Pick and place operation of robotic arm

7. Conclusion

By combining image processing technique and evolutionary computation based on MFPA, a 5-DOF robotic arm that can recognize the nearest object's colour, pick it accurately and place it in its particular location has been developed successfully. In addition, the concern on user safety is also been taken into consideration during the design of the robotic arm. The robotic arm stops working when the user hand (obstacle) is detected and continues the PNP operation when there is no obstacle detected.

The performance of the robotic arm can be affected by the lightning of the environment. Fine-tuning the parameters of converting the grayscale image into a binary image and reduce the pixels that need to be removed from binary image may improve the adaptability of the robotic arm to different environments. Integrating the robotic arm with Raspberry Pi instead of using the personal computer is advantageous for ease-of-use. Furthermore, the potential applications of this robotic arm system in fruit/crops ripeness classification, fast picker robotic arm for manufacturing, object tracking system for surveillance monitoring, classification of fish species/sizes in aquaculture industry, operating in hazardous condition or handling hazardous materials and as an assistance tool for teaching and learning, can be explored further.

Acknowledgement

Financial support from Universiti Tun Hussein Onn Malaysia (UTHM) in the form of IGSP Grant Vot U671, RMC, UTHM and GPPS Vot U806 are gratefully acknowledged.

References

[1] Abdou, G., & Lee, E. (1991). "Physical model for robotics palletization. *Computers in Industry"*, *16*(3), 255–266. https://doi.org/10.1016/0166-3615(91)90063-F

[2] Almendral, K. A. M., Babaran, R. M. G., Carzon, B. J. C., Cu, K. P. K., Lalanto, J. M., & Abad, A. C. (2018). "Autonomous Fruit Harvester with Machine Vision. *Journal of Telecommunication", Electronic and Computer Engineering*, *10*(1), 79–86. Retrieved from http://journal.utem.edu.my/index.php/jtec/article/view/3671/2554

[3] Artemiadis, P. K., Katsiaris, P. T., & Kyriakopoulos, K. J. (2010). "A biomimetic approach to inverse kinematics for a redundant robot arm. *Autonomous Robots"*, *29*(3-4), 293–308. https://doi.org/10.1007/s10514-010-9196-x

[4] Brunetti, A., Buongiorno, D., Trotta, G. F., & Bevilacqua, V. (2018). "Computer vision and deep learning techniques for pedestrian detection and tracking: A survey. *Neurocomputing"*, *300*, 17–33. https://doi.org/10.1016/j.neucom.2018.01.092

[5] Culler, D., & Long, J. (2016). "A Prototype Smart Materials Warehouse Application Implemented Using Custom Mobile Robots and Open Source Vision Technology Developed Using EmguCV". *Procedia Manufacturing*, *5*, 1092–1106. https://doi.org/10.1016/j.promfg.2016.08.080

[6] Dharmannagari, V. K. R. (2014). "Sorting of Objects Based on Colour By Pick and Place Robotic Arm and With", *3*(1).

[7] Dhepekar, P., & Adhav, Y. G. (2017). "Wireless robotic hand for remote operations using flex sensor. *International Conference on Automatic Control and Dynamic Optimization Techniques, ICACDOT" 2016*, 114–118. https://doi.org/10.1109/ICACDOT.2016.7877562

[8] Gu, J., Wang, H., Pan, Y., & Wu, Q. (2015). "Neural network based visual servo control for CNC load/unload manipulator". *Optik*, *126*(23), 4489–4492. https://doi.org/10.1016/j.ijleo.2015.07.153

[9] Hayashi, S., Ota, T., Kubota, K., Ganno, K., & Kondo, N. (2005). "Robotic Harvesting Technology for Fruit Vegetables in Protected Horticultural Production. *FRUTIC 05", Information and Technology for Sustainable Fruit and Vegetable Production*, (September), 227–236.

[10] Lampert, C. H., Nickisch, H., & Harmeling, S. (2009)." Learning to detect unseen object classes by between-class attribute transfer". *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, *2009 IEEE* , 951–958. https://doi.org/10.1109/CVPRW.2009.5206594

[11] Li, S., Chen, S., Liu, B., Li, Y., & Liang, Y. (2012). "Decentralized kinematic control of a class of collaborative redundant manipulators via recurrent neural networks". *Neurocomputing*, *91*, 1–10. https://doi.org/10.1016/j.neucom.2012.01.034

[12] Li, X., Qiao, T., Pang, Y., Zhang, H., & Yan, G. (2018). "A new machine vision real-time detection system for liquid impurities based on dynamic morphological characteristic analysis and machine learning. *Measurement": Journal of the International Measurement Confederation*, *124*(October 2017), 130–137. https://doi.org/10.1016/j.measurement.2018.04.015

[13] Li, Y., Huang, T., & Chetwynd, D. G. (2018). "An approach for smooth trajectory planning of high-speed pick-and-place parallel robots using quintic B-splines". *Mechanism and Machine Theory*, *126*, 479–490. https://doi.org/10.1016/j.mechmachtheory.2018.04.026

[14] Martinez-De Dios, J. R., & Ollero, A. (2015). "A learning-based thresholding method customizable to computer vision applications". *Engineering Applications of Artificial Intelligence*, *37*, 71–90. https://doi.org/10.1016/j.engappai.2014.08.015

[15] Mehta, S. S. (2007). "Vision-Based Control for Autonomous Robotic Citrus Harvesting".

[16] Moghaddam, M., & Nof, S. Y. (2016). "Parallelism of Pick-and-Place operations by multi-gripper robotic arms". *Robotics and Computer Integrated Manufacturing*, *42*, 135–146. https://doi.org/10.1016/j.rcim.2016.06.004

[17] Nearchou, A. C. (1998). "Solving the inverse kinematics problem of redundant robots operating in complex environments via a modified genetic algorithm""*Application of Modified Flower Pollination Algorithm on Mechanical Engineering Design Problem*. Batu Pahat, Johor.

[18] Rai, N., Rai, B., & Rai, P. (2014). "Computer Vision Approach for Controlling Educational Robotic Arm based on Object Properties" .

[19] Reddy, R., & Nagaraja, S. R. (2015). "Integration of robotic arm with vision system". *2014 IEEE International Conference on Computational Intelligence and Computing Research*, *IEEE ICCIC 2014*, 373–378. https://doi.org/10.1109/ICCIC.2014.7238302

[20] Sajjad, M., Talpur, H., & Shaikh, M. H. (2012). "Automation of Mobile Pick and Place Robotic System for Small", 522–526.

[21] Sangeetha, G. R., Kumar, N., Hari, P. R., & Sasikumar, S. (2018). "Implementation of a Stereo vision based system for visual feedback control of Robotic Arm for space manipulations". *Procedia Computer Science*, *133*, 1066–1073. https://doi.org/10.1016/j.procs.2018.07.031

[22] Saxena, a., Driemeyer, J., & Ng, a. Y. (2008). "Robotic Grasping of Novel Objects using Vision"". *The International Journal of Robotics Research*, *27*(2), 157–173. https://doi.org/10.1177/0278364907087172

[23] Starke, S. (2016). "A Hybrid Genetic Swarm Algorithm for Interactive Inverse Kinematics". MSc Thesis.

[24] Taylor, P. (2007). "On Optimizing Bin Picking and Insertion Plans for Assembly Robots On Optimizing Bin Picking and Insertion" Plans for Assembly Robots, (December 2013), 37–41.

[25] Zhang, Y., & Wang, J. (2004). "Obstacle Avoidance for Kinematically Redundant Manipulators Using a Dual Neural Network", *34*(1), 752–759.