

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter

Implementation of an Artificial Vision System for Welding in the Retrofitting Process of a Robotic Arm Industrial

Yomin Estiven Jaramillo Munera, Jhon Edison Goetz Mora, Juan Camilo Londoño Lopera and Edgar Mario Rico Mesa

Abstract

An industrial approach to the use of artificial vision is worked, and we are searching for the improvement of the welding process using a robotic arm. These kinds of robots in the last years have been associated to high accuracy tasks like classification, welding, object manipulation, assembly, and so on. Generally, the artificial vision is not used in works which use manipulator arm; this is normally due to the robot programmer who plans the robot task, which is executed cyclically, however, there are some approaches where different tasks using artificial vision are implemented. In this chapter, we present a retrofitting process of a manipulator welder arm Miller MR-2000, and the development of an artificial vision system, which could be used in the positioning of the machine. The developed system is able to look for areas suitable for the welding task between two pieces of material within a workspace; this process is possible using techniques of computational vision and image processing. Subsequently, the algorithm calculates the number of welding points based on the area identified previously, and finally, it sends the respective coordinates by means of G code to the robot for welding the pieces.

Keywords: robotic arm, welding system, palletizing robot, kinematics

1. Introduction

Many works related with manipulators arms have been developed since the first patent of anthropomorphic industrial robots was presented in 1973 [1]. These kinds of robot were thought to carry out a complex task in industries which represented great physical loads, or some kinds of risks for the workers. In addition, they used to be related with the search of the improvement of quality and production [2–4]. Currently, these robots have been associated to high accuracy tasks like classification, welding, object manipulation, assembly, and so on.

Artificial vision is a topic which is getting strength in the last years, which can be seen in several works where the artificial vision is used in different applications of accuracy and classification as is presented in [5, 6]. Nevertheless, the artificial vision is not constantly used in works which make developments with manipulator

arms; this is normally due to the robot programmer who plans the given robot task, which is executed cyclically [7]. However, some authors present approaches where they implemented different tasks using artificial vision; an example is presented in [8], where the artificial vision is shown as an alternative to guarantee the security of the manipulator arms against collisions, instead of using sensors. On the other hand, in [9], it is shown how an object manipulator robot, which is able to distinguish between different pieces of cutlery, was designed; they divided their work in two stages: the first stage includes the recognition of the objects through computer vision algorithms and finding the coordinates of the objects; and the second stage is the realization of the movement of the robot to the found coordinates using the forward kinematic, and in addition, by means of descent gradient method. A similar approach was shown in [10], where the authors presented a prototype of a system that classifies the rocks that feed the grinding process in a concentration plant; in this case, the researchers use a 2D artificial vision system that estimates the size of the rocks based on their area and distinguishes them; nevertheless, their results are ambiguous because they did the images acquisition statically in a dynamic system.

In the case of the welding robot arms, when we can integrate technologies like artificial vision and computational intelligence with industrial robot, it is possible to adapt us to multiple environments and different types of processes, which are very attractive. The principal reason to do this implementation is to search a reduction in production times and an increase in the efficiency of the welding process. Developments as discussed earlier in this chapter take strength because most of the technologies currently available are very expensive for medium and small companies.

The aim of this chapter is to include an artificial vision system in the retrofitting robotic arm process. For this purpose, an old robot manipulator supplied by a company was used, and its behavior was checked. In the general review of the system, it was observed that it cannot make complex moves due to failures in the system, associated mainly to encoders of the servomotors. A new panel control was designed, and in addition, an open source software was implemented to integrate the welding process with artificial vision, in order to locate spatially the working pieces. In this section, the kind of robot arm, camera, and the welding system are described in a deeper way. The methodology of each stage of the project is explained, based on the robot arm kinematic and the generation of a G code with the intersection points of the pieces. A similar work as proposed in this chapter is presented in [1], where the retrofitting of an industrial robot is developed; this work is specially related with ours because the authors showed a modernization of an old robot arm, and besides, they did a comparative study related with the numerical control machines controllers integration using LinuxCNC controller and MatLab/Mach3, where they concluded that LinuxCNC is a better solution because it needs few hardware resources in comparison with Mach3/MatLab alternative; in addition, the license does not have any cost, however this work does not use artificial vision in the positioning of the tool.

2. Materials and methods

Before starting with the implementation of the artificial vision system, it was necessary to carry out a general adjustment of the robot. This process has several stages which involve hardware conditioning and considerable work in software development. The methodology addressed in this chapter is presented in **Figure 1**.

In this section, the kind of robot arm, camera, and the welding system are described in a deeper way. In addition, the methodology of the development of each stage of the project is explained.

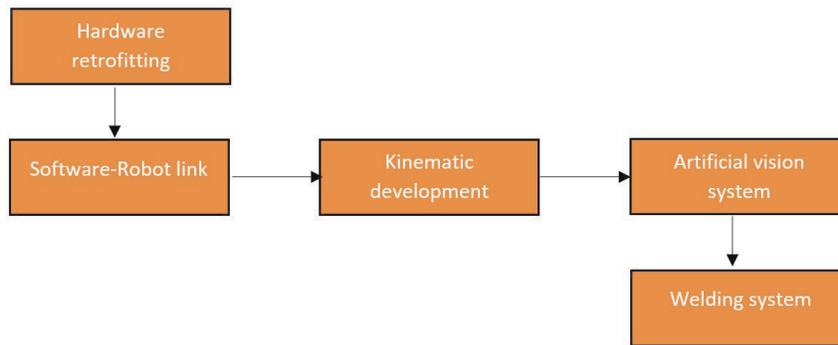


Figure 1.
Methodology description.

2.1 Robot arm

An MR-2000 welding robot Miller manufactured in 1995 is used for the development project, as shown in **Figure 2**. The robot spent several years deteriorating, and at the time of the evaluation, several errors in the motherboard and the encoders of the servomotors were found. This made impossible to use the original hardware and software.

The arm counts with a morphology of a palletizing robot, 6 freedom grades, and an approximated weight of 353 lb (160 kg). Due to the bad condition of the robot, a total rebuild of the control panel was made; for these modifications, several components were required and the most representative of them are exposed in **Table 1**.

Taking as reference the architecture of palletizing robot presented in [11], the control of the heaviest parts of the robot (base, big forward arm, and big back arm) is designed so that three servomotors of great power (1.0 kW) are responsible for global robot movements. Another three servomotors of 0.12 kW control the remaining pieces which are more focused in the final tool orientation (forearm, wrist, and a grabber). Also, the control includes computational hardware that contains the OS Linux and the vision system for the location and control of the arm.

2.2 Software-robot link

An intensive search was made for the selection of the most appropriate software for arm control; taking that into count, we realized that some of the programs in the middle are made by people who do not find a good option for this application;



Figure 2.
Welding robot Miller.

Description	Quantity
Servomotor 880-DST-A6HKB	3
Servomotor 11A-DST-A6HKB	3
DYN4-H01A2-00	3
DYN2-TLA6S-00	3
Computational Hardware	1
I25 Superport FPGA based PCI	1
Artificial vision system	1

Table 1.
Materials.

therefore, they choose to design and build their own software. These approaches have an open architecture but with much work to develop and some are unavailable.

The software used to control the robot arm is designed to CNC type machines, however, it can be implemented the kinematics of different kind of robots and all the instructions and protocols to move until a 9-Axis robot. LinuxCNC is a software widely known in robotic applications and has several advantages in respect to others [1]. It is integrated with an interface to control and monitor each move of the servomotor in a join or world mode. The software was configured using the MESA i525; this device is a general purpose FPGA based programmable I/O card for the PCI bus, and it uses standard parallel port pinouts and connectors for compatibility with most parallel port interfaced motion control/CNC breakout cards/multi-axes step motor drives; for this interface, a new parallel breakout board is designed by means of a digital optocoupled connection between the mesa 5i25 and the servodrivers; the capacitive sensor which are considered as the home switch of the robot also receives signals; this sensor is the physical mean to enable all the arm controls and has the connection of the stop control that disables the robot's hardware and software. Two of these boards are implemented, the first one for the axes 0, 1, and 2 and the other one for the axes 3, 4, and 5 distributed by the type of servodriver that drives each axis [12]. They have the compatibility to receive a numerical program (G code) for the movement of the robot, which makes possible the integration between the control of the robot and the algorithm that uses computer vision to locate the working pieces, and it generates the spacial coordinates XYZ of the welding points through a G code loaded in LinuxCNC. In order to set the MESA 5i25 for working with our electronic boards that were previously made, a custom firmware is installed in the board, where the pin inputs and pin outputs are set based on the pin distribution of our control system.

It is necessary to make the initial configuration of the machine in PNCconf of LinuxCNC, that is a helped interface which leads to user in the set up of machines which use MESA boards; however, PNCconf has several limitations from its interface, and usually, it is necessary to make modifications in the principal configuration files created by this software; those files have the *.ini* and *.hal* extensions, respectively. Normally, the modifications in the configuration files are related to the number of joint of the machine, because the interface is not possible to set more than four of them.

2.3 Kinematics development

Kinematics is a common problem in the development of software for the manipulation of the robotics arm due to math complexity. Typically, robotic arms

have a setup serial where each articulation depends on the previous articulation. For the case of the robots type serial, the system can be defined by Denavit-Hartenberg parameters [13]. This approach uses homogeneous transformations by means of matrices, as this method is frequently used. Our robot is not a serial robot. The MR-2000 has a palletized [11, 14] extension that belongs to the joint 3. By this motive, it was necessary to develop a model kinematics which is able to work with that geometry.

The movements of a robotic arm are described by two types of kinematics, forward kinematics and inverse kinematics.

- *Forward kinematics*: this is a procedure to get the coordinates in our reference plane $[X_T, Y_T, Z_T, A, B]$ taking as reference the angular positions of each joint of the system $[\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]$. This kind of kinematics is worked in LinuxCNC from joint mode.
- *Inverse kinematics*: this principally seeks to find the angular positions $[\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]$ of each joint based on spatial coordinates $[X_T, Y_T, Z_T, A, B]$ given. In this case, multiple solutions are possible. Therefore, it is quite important to define the restrictions of the robot. This kind of kinematics is worked in LinuxCNC from world mode.

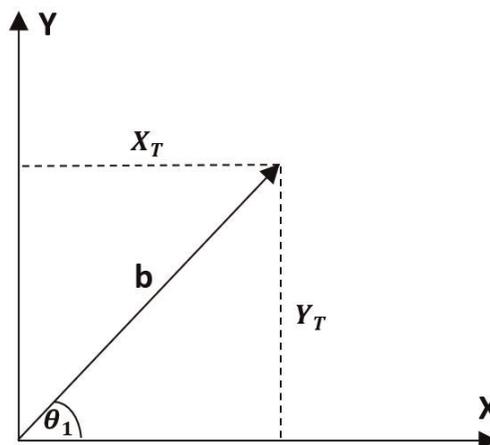


Figure 3.
 Top view of the robot.

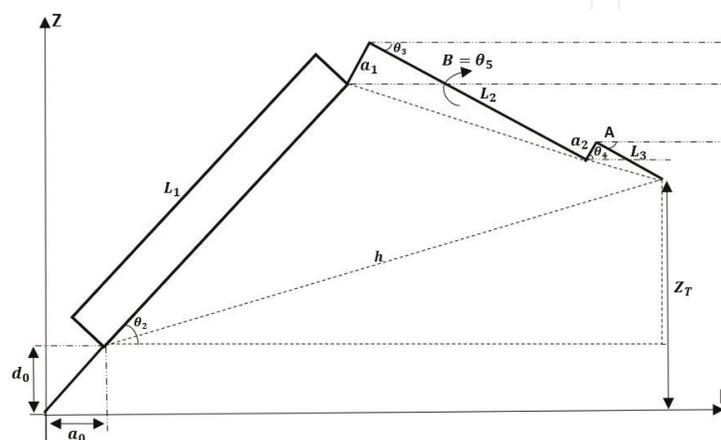


Figure 4.
 Side view of the robot.

An approximation to the structure of the robot is presented in **Figures 3** and **4**, where it is possible to observe the different parameters involved in the kinematics of the robot.

LinuxCNC has a module of kinematics that allows loading a file of code C++ previously compiled with the model of the kinematics of the robot. In this file, both kinematics should be defined for the complete functioning of the robot.

2.4 Artificial vision system

The camera used is the Blackfly of 1.3 MP that has the following characteristics: resolution of 1288×964 px, type of CCD sensor, pixel size of $3.75 \mu\text{m}$. This type of camera is manufactured for applications of artificial vision. It has robust features and captures precise images to contribute to the image processing stage.

3. Results

3.1 Kinematics development

In order of setting the kinematics of the robot in LinuxCNC, it is necessary to define the forward kinematics and the inverse kinematics. Therefore, it is necessary to describe mathematically both kinematics. This work is done by linking the input parameters with the output parameters with the help of geometry. The geometrical parameters (GP) of MR-2000 are shown in **Table 2**.

3.1.1 Forward kinematics

As described earlier in Section 2.3, the forward kinematics allows to find our coordinates $[X_T, Y_T, Z_T, A, B]$ from the values of the angles of each joint, taking into count the value of the parameters $[\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]$ (See **Figures 3** and **4**).

Knowing this, the coordinates are calculated as follows:

$$b = -a_1 \sin(\theta_3) + L_2 \cos(\theta_3) - a_2 \sin(A) + L_3 \cos(A) + L_1 \cos(\theta_2) + a_0 \quad (1)$$

$$X_T = b * \cos(\theta_1) \quad (2)$$

$$Y_T = b * \sin(\theta_1) \quad (3)$$

$$Z_T = L_1 \sin(\theta_2) + a_1 \cos(\theta_3) + L_2 \sin(\theta_3) + a_2 \cos(A) + L_3 \sin(A) + d_0 \quad (4)$$

$$A = \theta_3 + \theta_4 \quad (5)$$

GP of the arm	Value
a_0	300 mm
a_1	100 mm
a_2	0 mm
d_0	300 mm
L_1	500 mm
L_2	700 mm
L_3	500 mm

Table 2.
Parameters of robot.

$$B = \theta_5 \quad (6)$$

X_T and Y_T are located in **Figure 3**, and Z_T , A, and B in **Figure 4**. Based on the above, the LinuxCNC system can find the final position of the robot in the system of coordinates $[X_T, Y_T, Z_T, A, B]$ in the joint mode.

3.1.2 Inverse kinematics

In order to find the values of angular positions based on the spatial coordinates, the following equations are developed:

$$b = \sqrt{X_T^2 + Y_T^2} \quad (7)$$

$$h = \sqrt{\left(Z_T - d_0 + \sqrt{a_2^2 + L_3^2} \sin \left(A - \arccos \left(\frac{L_3}{\sqrt{a_2^2 + L_3^2}} \right) \right) \right)^2 + (b - a_0)^2} \quad (8)$$

$$\theta_1 = \arctan \left(\frac{X_T}{Y_T} \right) \quad (9)$$

$$\theta_2 = \arccos \left(\frac{h^2 + L_1^2 - \sqrt{a_1^2 + L_2^2}}{2hL_1} \right) + \arcsin \left(\frac{Z_T - d_0 + \sqrt{a_2^2 + L_3^2} \sin \left(A - \arccos \left(\frac{L_3}{\sqrt{a_2^2 + L_3^2}} \right) \right)}{h} \right) \quad (10)$$

$$\theta_3 = -\theta_2 + \arccos \left(\frac{h^2 + L_1^2 - \sqrt{a_1^2 + L_2^2}}{2hL_1} \right) - \arcsin \left(\frac{a_1}{\sqrt{a_1^2 + L_2^2}} \right) - \arccos \left(\frac{\sqrt{a_1^2 + L_2^2} + h^2 - L_1^2}{2h\sqrt{a_1^2 + L_2^2}} \right) \quad (11)$$

$$\theta_4 = A - \theta_3 \quad (12)$$

$$\theta_5 = B \quad (13)$$

For the execution of a welding process, it is necessary that the robotic arm meets a series of requirements related to the precision and straightness in the movement of the axes. These parameters are necessary for taking the robot to work under the welding standards of parts. To verify the precision in the displacement of the axes, a dial comparator is placed as shown in **Figure 5** and the moving each axis is measured, so that the following results are obtained.

Commands by MDI (Command Line Interface) was sent to test the move of the robotic arm and the kinematics. The result of these moves is shown in the **Table 3**, where the precision of the arm is measured with a digital dial comparator with a resolution of 0.01 mm. The results show a maximum difference of 0.16 mm being permissible for the application in which it will be employed. Each axis is assembled as in **Figure 5**, in which the indicator is located parallel and centered to the axis to be measured.



Figure 5.
Digital indicator.

Displacement	Axis (X)	Axis (Y)	Axis (Z)
To 1 (mm)	0.98	0.96	0.99
To 1.5 (mm)	1.48	1.46	1.47
To 2.5 (mm)	2.46	2.45	2.47
To 5 (mm)	4.88	4.84	4.90

Table 3.
Movements.

3.2 Artificial vision system

An algorithm of computational vision is implemented in the welding process to locate spatially the pieces respect to the robot coordinates. The camera is set perpendicular to the working area at a fixed coordinate XYZAB with respect to the center robot position; the CCD makes the capture of the image that contains the pieces to be weld. OpenCV is used for the segmentation process and it is performed applying filters and morphological operation to determine the area of contact to be weld; this area becomes a straight line and the points that will be used for the creation of G code will be loaded in the numerical control.

For the processing of the images, the capture of the pieces to be weld is made, as can be seen in **Figure 6**. The working pieces are located on a surface that

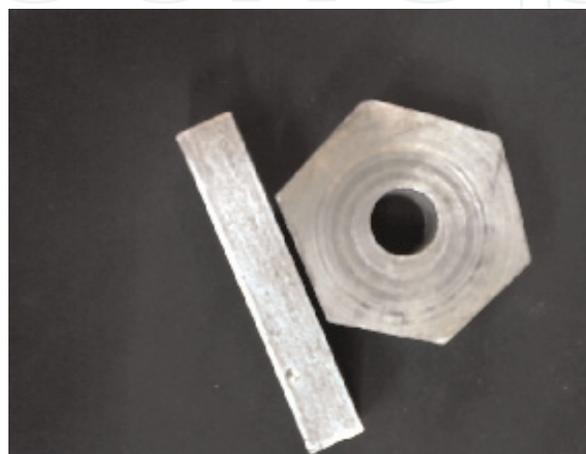


Figure 6.
Original image.

contrasts with the color of the pieces to facilitate the segmentation and identification of the edges.

Subsequently, the image is converted to a gray scale, and a Sobel filter is applied to create an image emphasizing edges. The resulting image can be observed in **Figure 7**; this process was made with the aim of being able to identify the possible closeness of the working pieces.

Before the pieces are segmented to highlight the edges, a morphological operation of closing is made to eliminate small gaps (filling them) and join components connected nearby. The kernel used is the structuring element of a rectangle of 1×8 pixels. The result is presented in **Figure 8**, where the space that separates the two pieces was totally filled.

In the last process, it is possible to highlight the suitable area for the welding process; the algorithm is able to do it due to a Gaussian filter that is applied to eliminate the rest of the edges of the piece and leave only the highlighted union of both. With the segmentation process done, the line that traces the union of the pieces is transformed into points that will be used to coordinate to which the arm must reach to carry out the welding process.

To verify the precision of the algorithm, several probes are made to check the real coordinates and the coordinates calculated from the computer vision. In this process, the values of the location of the camera and the relationship of the pixels and the location XYZAB change according to the precision of each servomotor.

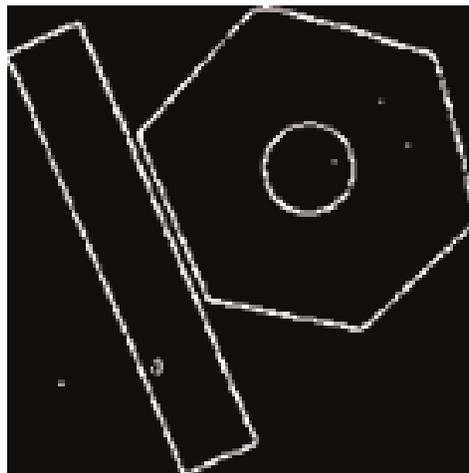


Figure 7.
Edge detection.

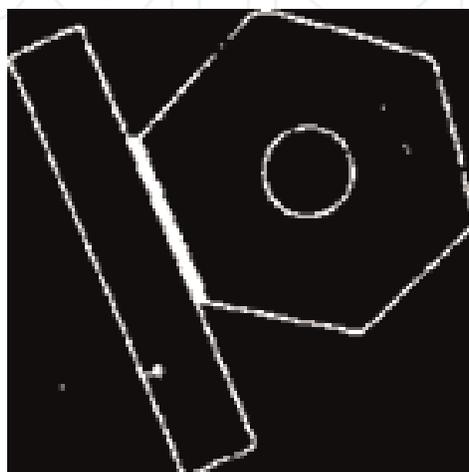


Figure 8.
Welding trajectory.

4. Conclusions

The trials show that the kinematics is the main factor that is needed for producing the natural movements of the robotic arm. Respect to the welding process, we can observe that the system achieves to be in the right standards for the development of the process. Finally, based on the results of the artificial vision phase, it is correct to say that the segmentation and edges identification process are successful, and therefore, it is possible to perform this process within the typical ranges of separation of pieces for a welding process.

Author details

Yomin Estiven Jaramillo Munera*†, Jhon Edison Goez Mora†,
Juan Camilo Londoño Lopera† and Edgar Mario Rico Mesa†
Servicio Nacional de Aprendizaje, Medellín, Colombia

*Address all correspondence to: yejaramillom@sena.edu.co

† These authors contributed equally.

IntechOpen

© 2020 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Alberto A, Eduardo J, Lima I, Marcelo H, Souza B. Retrofitting of ASEA IRB2-S6 industrial robot using numeric control technologies based on linuxcnc and MACH3-MATLAB. In: IEEE International Conference on Robotics and Biomimetics (ROBIO); Macau, China; 2017
- [2] Brett PN, Khodabandehloo K. Analysis of bi-arm robots for applications in manufacturing industry. Proceedings of the Institution of Mechanical Engineers. Part B: Journal of Engineering Manufacture. 1991;**205**: 43-50. DOI: 10.1243/PIME
- [3] Wehe DK, Lee JC, Martin WR. Intelligent robotics and remote systems for the nuclear industry. Nuclear Engineering and Design. Department of Nuclear Engineering, University of Michigan. 1989;**113**:259-267
- [4] Neda A, Ali L. Socialization of industrial robots: An innovative solution to improve productivity. In: 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON); Vancouver, Canada. IEEE; 2018. pp. 832-837
- [5] Londono Lopera JC, Goetz Mora JE, Rico Mesa EM. Positioning of the Cutting Tool of a CNC Type Milling Machine by Means of Digital Image Processing Advances in Computing. CCC 2018. Communications in Computer and Information Science. Vol. 885. Cham: Springer;
- [6] Goetz Mora JE, Londono Lopera JC, Patino Cortes DA. Automatic Visual Classification of Parking Lot Spaces: A Comparison Between BoF and CNN Approaches. Applied Computer Sciences in Engineering. WEA 2018. Communications in Computer and Information Science. Vol. 915. Cham: Springer;
- [7] Frank C. Programming vision-guided industrial robot operations. Journal of Engineering Technology;**3009**(26):10-15
- [8] Fevery B, Wyns B, Boullart L. Industrial robot manipulator guarding using artificial vision. In: Robot Vision. Vol. 28. Rijeka, Croatia: IntechOpen; 2006. DOI: 10.5772/9294
- [9] Bilal I, Huseyin A, Yakup K, Serdar Y, Yildirim E. Smart robot arm motion using computer vision. Elektronika Ir Elektotehnika. 2015;**21**(6):3-7
- [10] Rene V, Aldo C. System for classifying rocks by using artificial vision and a robot arm. In: ISIE '97 Proceeding of the IEEE International Symposium on Industrial Electronics; July 7-11, 1997. Guimaraes, Portugal: IEEE; 1997
- [11] Yong T, Fang C, Hegen X. Kinematics and workspace of a 4-DOF hybrid palletizing robot. Advances in Mechanical Engineering. 2014;**6**:1-6
- [12] 5I25 Picture, MESA Electronics [Internet]. Available from: http://store.mesane.com/index.php?route=product/product&product_id=55 [Accessed: 27 March 2019]
- [13] Corke P. A simple and systematic approach to assigning a simple and systematic approach to assigning. IEEE Transactions on Robotics. 2007;**23**: 590-594
- [14] Rui Z, Zhou B, Liu J. Dynamic simulation of palletizing robots based on ADAMS. In: 2nd International Conference on Electronic & Mechanical Engineering and Information Technology; Lanzhou, China; 2012. pp. 1446-1449