International Journal of Combinatorial Optimization Problems and Informatics, 12(3), Sep-Dec 2021, 20-31. ISSN: 2007-1558.

www.editada.org

Vision assisted pick and place robotic machine

José L. Arévalo-Hernández[1], Elsa Rubio-Espino[1], Víctor H. Ponce-Ponce[1], J. Humberto Sossa-Azuela[1]

[1] Centro de Investigación en Computación del Instituto Politécnico Nacional.
jarevalo.cic.ipn@gmail.com, erubio@cic.ipn.mx, vponce@cic.ipn.mx, hsossa@cic.ipn.mx

Abstract. We present the design and implementation of a vision-assisted pick and place robotic machine for laboratory activities. The robotic device can automatically locate and pick small electronic components placed in containers in scattered order and put them in their final assembly position, indicated by pre-located visual marks on a printed circuit board. The execution of these tasks is supported by the design of two main systems. The first one consists of an image processing algorithm, fastly and precisely, capable of recognizing objects of interest arranged in containers. The second system involves the design and implementation of a Cartesian robot with four degrees of freedom. The robot's electromechanical assembly design involved implementing a PID controller and a power and sense electronics module. The resulting robotic machine solved the surface-mount capacitors manipulation problem, rendering an accuracy with a maximum deviation angle of ±15°, during the components placement tasks.

Keywords: Image processing, PID control, surface-mount technology.

Article Info
Received April 15, 2021.
Accepted June 2, 2021.

1. Introduction

Within Industry 4.0, pick and place robotic machines are widely used for manufacturing of today's Surface-Mount Technology (SMT) to assembly tinny electronic components on printed circuit boards (PCB) [1], [2], [3], and [4]. Using these machines, it is possible to accurately pick and place large numbers of small electronic components quickly and accurately onto large PCB batches for commercial production. However, most of the time, these machines, when used at an academic environment, focus mainly on producing experimental test benches to conduct single circuit prototypes' electrical measurements. Therefore, under these circumstances, high-volume production of PCB is not required. The pick and place workstation MPP-21 [5] is an example of a hand-assembly operation system designed to allow an operator to manually place surface-mount devices (SMD) onto a PCB, manually utilizing a vacuum tip during the component pick-up and placement tasks. This system was also conceived to dispense and deposit solder paste, adhesives, and various potting compounds. The MPP-21 workstation has optional tape and sticks feeders with a feeder rack for component handling. For an operator, working under these conditions is tedious, and it frequently leads to make mistakes during a work session. Taking this into account, we were motivated to use, as starting components, some spare parts of an MPP-21 workstation to design, implement and test an automatic pick and place robotic machine suitable for use in academic and research environments. We designed and implemented a Cartesian robot, jointly with an image analysis module involved with picking, driving, and placing the SMD components onto a PCB. We tested the robotic machine with ceramic capacitors SMD-type, sized (3.6 mm×1.6 mm×0.9 mm).

The organization of the paper is as follows. The general description of the system and the main components used in the project are addressed in Section II. The system's design is discussed in Section III; the experimental test results are addressed in Section IV. The system's performance is discussed in Section V. The conclusions are presented in Section VI and, finally, the future work outlined in Section VII.

2 General Description of the System

Considering that this project's primary motivation was to design and implement a pick and place robotic machine, we begin by describing its general structure and dimensions. A Cartesian robot with 4 degrees of freedom (DoF) was selected; see Figure 1. Each DoF corresponding to the *x*, *y,* and *z*-axis produces linear movements, while the *w* axis produces rotatory movements.

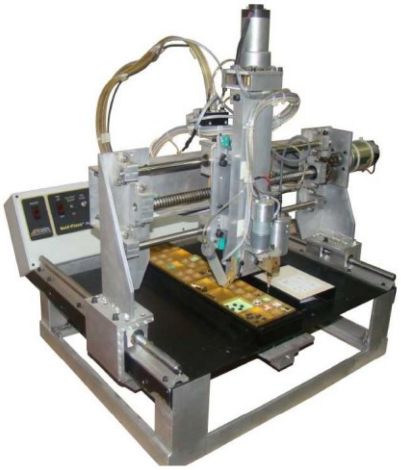

Figure 1. Implementation of a *x,y,z* Cartesian robot in the modified work area of a MPP-21 manual workstation.

The DoF configuration for the Cartesian robot is implemented by a screw-nut and parallel guide system, with a gear reducer and an encoder adapted in each case, which corresponds to the *x* and *y*-axis. The movements performed at each DoF are produced by permanent magnet DC motors (PMDC). At the *z*-axis, there is also an encoder mechanically coupled to the PMDC motor structure. The *x*, *y,* and *z*-axis encoders have a resolution of 512 pulses per revolution (part number HEDS-5645-I13). Meanwhile, the PMDC motor used for the *w*-axis is coupled to the pneumatic suction cup's rotation axis by a gear reduction mechanism. This motor has connected an encoder with a resolution of 16 pulses per revolution. The resulting workspace for the Cartesian robot defines a volume of approximately 25cm×25cm×6cm, see [23].

The control strategy used for the PMDC motors was materialized through designing and implementing a customized power and electronic control module (PECU), which is part of the designed subsystem for the Cartesian robot's automation. The PECU subsystem relies on a general-purpose ATAVRMC300 processor board linked to four microcontroller units (MCU) ATxmega128A1. Figure 2 shows a picture of the PECU casing designed for this project. The MCU ATxmega128A1, was used to generate the necessary command signals to drive each of the four PMDC motors, and at the same time, to execute specific real-time processing tasks to transfer information from and to a personal computer (PC), concerning the angular position and the angular velocity of each PMDC motor, which is used by the position control algorithm.

Regarding the image processing algorithm, the original MPP-21 camera was reused by fixing it to the Cartesian robot on one side of the pneumatic suction cup (Figure 3), this camera is a SENSORAY 2250s model, which allows a 720×480 pixels resolution in MPEG 1/2/4 and JPEG format video, this camera was used as a part of the vision system. Thereby, the distance between the camera and the work area remains constant. As depicted in Figure 3, the camera takes pictures of the SMD components from the top. To the left, we can see the pneumatic suction cup. Also, Figure 3 shows the camera framework of the vision system. An algorithm coded in C++ was developed in a *Qt* Creator environment into Ubuntu Linux to accomplish this task. Concerning the image analysis, the OpenCV library was used [6]. Figure 4 shows the block diagram of the system components of the designed robotic machine showed in Figure 2.

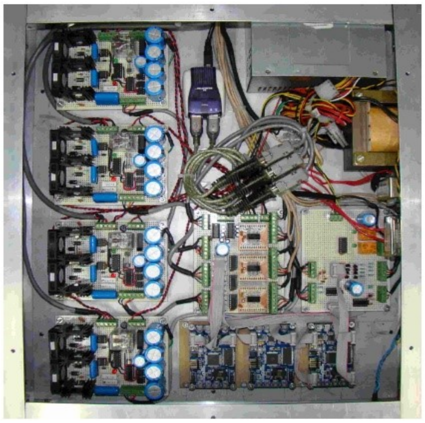

Figure 2. Power and electronic control module (PECU) cabinet, completely designed for the automation of the Cartesian robot.

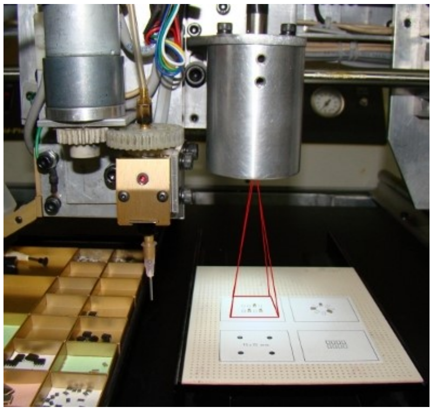

Figure 3. Workspace and camera installed in the Cartesian robot.

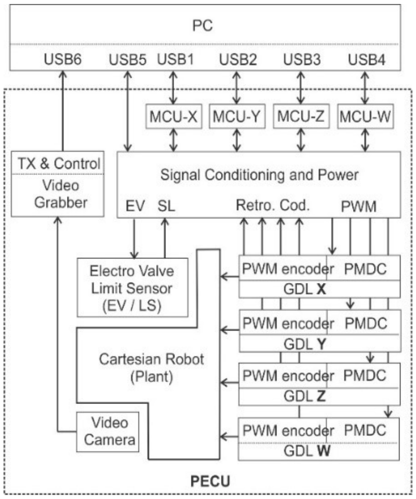

Figure 4. Pick and place robotic machine's blocks diagram.

3 System Design

3.1 PMDC control subsystem design

We propose a distributed control model for the PMDC motor set that combines a PID and a PI for each PMDC motor (Figure 5). According to [7], their design and characterization are developed. Table 1 shows the parameters obtained for each of the four motors, specifying the characteristics of each motor according to [7], see also [23]. The meaning of the listed parameters is as follows: Ka is the motor torque constant, Kb is the constant counter-electromotive, Ra and La are the resistance and inductance associated with the motor winding. Parameter *b* is the viscous friction constant, and *J* is the rotor moment of inertia, being these last parameters characterized by the system load's presence.

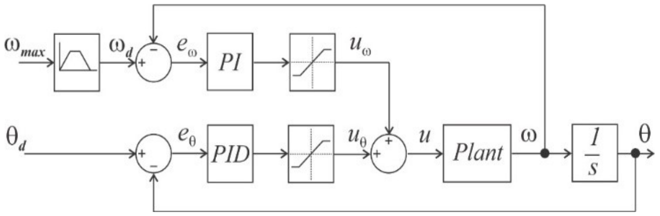

Figure 5. Blocks diagram of the control system for the PMDC motors.

Table 1. Extracted parameters of the Cartesian robot's motors according to [7]

Parameters	**x**	**y**	**z**	**w**
K_a (N·m/A)	0.0935	0.0923	0.0306	0.0054
K_b (V·s/rad)	0.107	0.112	0.0611	0.0059
R_a (Ω)	3.812	3.918	4.481	7.527
L(H)	0.00416	0.00392	0.00413	0.001719
b(N·m·s/rad)	0.00049	0.00082	0.00006	0.000044
J(kg·m^2)	0.00032	0.00039	0.00028	0.000106

The PMDC motor model is represented by the equivalent motor schematic diagram depicted in Figure 6, where e_a is the applied voltage to the armature, the voltage e_b is due to the counter-electromotive force, i_a is the electric current of the mesh, associated with the motor winding, θ represents the angular displacement, T_e the torque developed by the engine and, i_f is the field current. Current i_f is considered to be constant and proportional to the magnetic force produced by the permanent magnet.

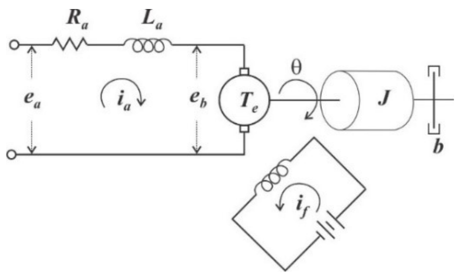

Figure 6. Electrical equivalent schematic diagram of the PMDC.

As it well-known, the following two equations can be derived from the above schematic diagram representing the time domain's engine dynamics.

$$e_a = L_a \frac{di_a}{dt} + R_a i_a + e_b \tag{1}$$

$$K_a i_a = J \frac{d^2\theta}{dt^2} + b \frac{d\theta}{dt} \tag{2}$$

We obtain the plant's transfer function by applying the Laplace transformation to solve the system of differential equations; see Equations (1) and (2). Then, for the system shown in Figure 5, we obtain the following expression:

$$\frac{\theta(s)}{E_a(s)} = \frac{K_a}{L_a J s^3 + s^2(R_a J + L_a b) + s(R_a b + K_a K_b)} \tag{3}$$

A variant of the well-known PID control [8] has been proposed for each motor's positioning control strategy. As illustrated in Figure 5, the controller is split into two parts. The first part evaluates the angular error position e_θ and outputs a control signal u_θ. The second part evaluates the angular velocity error e_ω and outputs a control signal u_ω. The final objective is to reach a specific θ value by modifying the angular velocity ω at each time step. The input parameter θ_d represents the controller setpoint in radians. The input parameter ω_d represents the instantaneous magnitude of the signal generator's angular velocity that precedes it. This parameter will set the following Equation for the controller in the domain of s.

$$u(s) = K_{p\theta} e_\theta(s) + \frac{K_{i\theta}}{s} e_\theta(s) + K_{d\theta} s e_\theta(s) + K_{p\omega} e_\omega(s) + \frac{K_{i\omega}}{s} e_\omega(s) \tag{4}$$

Where $K_{p\theta}$, $K_{i\theta}$, $K_{d\theta}$ are the proportional, integer and derivative constants of the PID controller and $K_{p\omega}$ and $K_{i\omega}$ are the proportional and integral constants of the PI controller. Note that these constants should have the units required for the controllers u_θ and u_ω to give voltage units. Then, they can be added. In this way, $u(s)$ can be substituted in $E_a(s)$ of Equation (3) to simulate the system. System tuning was conducted empirically, based on the system response behavior analysis. The resulting values for the PID and PI constants are summarized in Table 2. Concerning the *w*-axis PMDC motor, we only applied a PI controller.

Table 2. Parameter values of the PID and PI controllers

Parameter	***x***	***Y***	***Z***	***w***
$K_{p\theta}$	5	6.36	9.09	65
$K_{i\theta}$	0.00091	0.0045	0.0045	0.013
$K_{d\theta}$	0.00091	0.00091	0.00045	131.5
$K_{p\omega}$	1.09	1.09	0.91	-
$K_{i\omega}$	0.18	0.18	0.181	-

Some implementation details of the designed control algorithm, embedded in each MCU, are next exposed. For instance, an anti-Windup condition was added [9]. It consisted of eliminating the integration constant only when saturation is present. The Pulse Width Modulation (PWM) technique was used to drive each PMDC motor [10]. In this particular case, the frequency generated by each MCU is established in f_{PWM}= 18KHz. The PWM resolution was 10 bits. Whereas the PWM range is from 0 to 1023; then a value of PWM = 512 is equivalent to a 50% work cycle. It is possible to establish an empirical proportional relationship between the PWM work cycle and the equivalent applied voltage to the motor. To achieve this voltage ratio, speed tests were conducted, and the results are depicted in Table 3. These functions are used to obtain a numerical PWM value to be applied to each PMDC, employing the PECU power circuitry.

Table 3. Functions for the relation between PWM and voltage of each motor

Motor	Function
X	PWM=11.68 E_a+512.49
Y	PWM=11.05 E_a +513.27
Z	PWM=11.15 E_a +518.05
W	PWM=38.21 E_a +510.91

In the following subsection, we explain the image analysis algorithm.

3.2 Image analysis algorithm

Machine vision technology is nowadays widely used instead of human vision to detect interest objects. Some previous works reported visual location technology to pick and place at high speed, and precision electronic components can be found in [1], [4], [11], and [12]. In this work, the primary purpose of implementing an image analysis algorithm is to automatically locate the SMD capacitors placed in containers accurately, in scattered order, as shown in Figure 7(a). To achieve this task, we applied a combination of imaging techniques that allows segmenting interest objects. As a result, the robotic system was able to segment and tag the objects successfully.

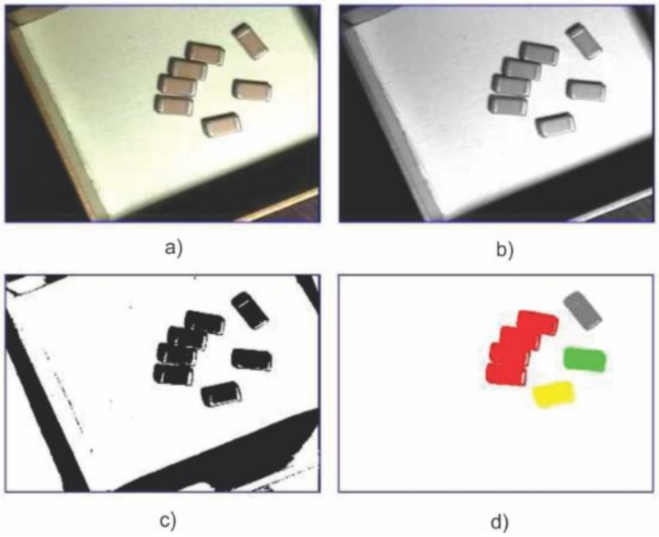

Figure 7. Analysis of the SMD components images inside the containers, a) Colored image, b) Grayscale image, c) Binarized version, and d) Filtered and labeled desired objects.

The images shown in Figure 7, 8, and 9 explain the proposed image analysis algorithm. Figure 7(a) shows the camera's original photograph, Figure 7(b) shows the gray scaled version of this image. Next, Figure 7(c) shows the results of applying the adaptive thresholding method, well described in [13] and [14] to the previous image. If we group the neighboring regions by a connected component algorithm and apply a minimum area filter, together with an elimination filter of the areas touching the image bounds, Figure 7(d) is obtained.

We can summarize the proposed image processing algorithm as follows:

```
Image capture and grayscale conversion
Image adaptive thresholding
Image filtering
Connected component labelling
For each object found in the image
{
   Apply segmented pyramid binarization
}
  Add objects
  Apply connected component labelling
For each object found
{
  Apply filter
  Calculate Hu moments
}
```

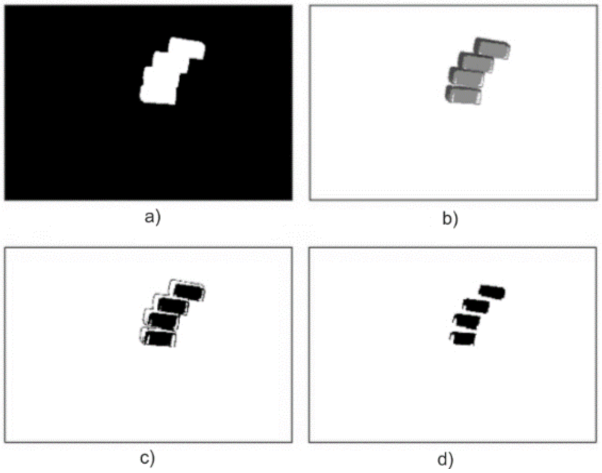

Figure 8. Labelled objects from the SMD components' analysis inside the containers, a) Mask, b) Segmented pyramid, c) Binarized image and d) Eroded regions.

Once the complete list of tagged (labelled) objects is obtained, we proceed to generate a mask for all the objects, as shown in Figure 8(a), to extract each region of interest. After the image is masked with the grayscale image, an operation of pyramid segmentation as well described in [24] and [25] is applied, as shown in (Figure 8(b)). This operation is used to reduce the number of gray image levels, in this case, to three gray levels. Then, the region of the image, whose area is more significant, is selected and binarized (Figure 8(c)). Next, a morphological erosion step is conducted [14] to separate the elements and obtain a tractable image of the capacitor, as illustrated in Figure 8(d). The localized objects are registered, and a second connected component labeling algorithm step is applied next, as illustrated in (Figure 9(a)). Finally, first and second order Hu moments are calculated [14] to obtain the center locations and $\left(\bar{x}, \bar{y}\right)$ orientations, θ, of the capacitors, are obtained, (Figure 9(b)), [14], by means of the very-well known equations:

$$\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}} \tag{5}$$

where, in the binary case: m_{00} is the area of the interest region and m_{10} and m_{01} are its first order moments [14, pages 141-142].

$$\phi = \frac{1}{2} tan^{-1}\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right) \tag{6}$$

where ϕ is the orientation of the object's main axis and the central moments are defined by: $\mu_{11} = m_{11} - \overline{y}m_{10}$, $\mu_{20} = m_{20} - \overline{x}m_{10}$, $\mu_{02} = m_{02} - \overline{y}m_{01}$, see [14, pages 142-143]).

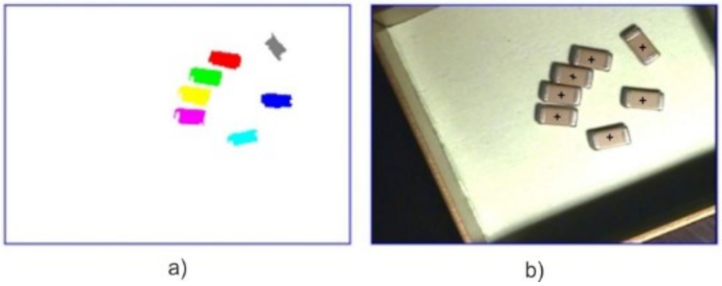

Figure 9. Results from the analysis of the SMD components images inside the containers. a) Labeled capacitors and b) Center locations.

4 Experimental Results

To verify the resulting precision movements of the Cartesian robot supported by the image analysis algorithm, we conducted some experimental tests. We provide a link to appreciate a short video demonstration of one of the test's movements performed by our designed pick and place robotic machines; see [22] and [23].

4.1 PMDC controller movement test results

Figure 10(a) shows the result of one movement test conducted over the *x*-axis. Starting from an initial *x*-position of 100 mm, the Cartesian robot moves to a final position of 130 mm with a maximum linear speed of 35 mm/s. The elapsed time for this test approximates 1.2s. Figure 10(b) depicts the angular PMDC motor speed, *ω*. Finally, Figure 10 (c) illustrates the resulting PWM units, *u*, sent to the PMDC motor. Figures 11(a), (b), and (c) show the result of one movement test conducted over the *w*-axis. The initial position is 0 rad, the final position is π rad. In this case, the elapsed time approximates 1.073s. An overshoot in the dynamical response is observed. The average set of errors are summarized in Table 4. As can be appreciated, these errors are tolerable.

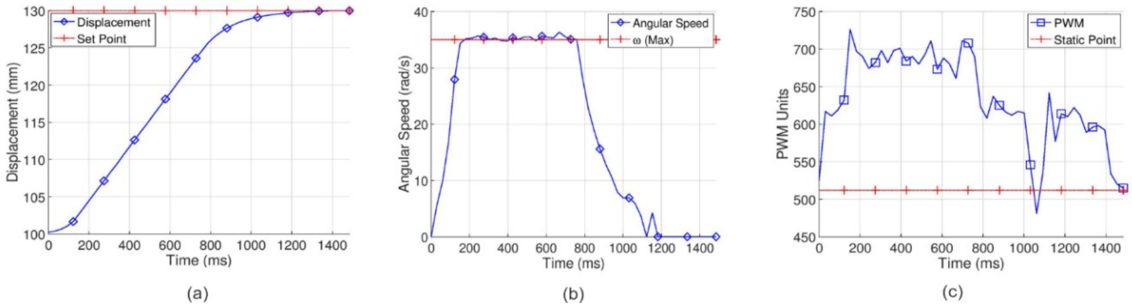

Figure 10. Motor of axis *x* response to a setpoint with a top speed of 35 rad/s.

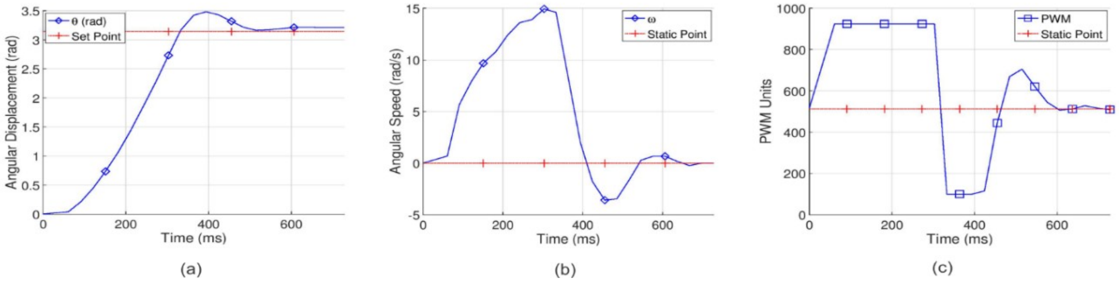

Figure 11. Displacement from 0 to π-rad, angular speed, and controller response in *w*-axis.

Table 4. The resulting average set of errors.

DoF	Error	Units
x	±0.022	mm
y	±0.012	mm
z	±0.004	mm
w	±0.0087	rad

4.2 Image analysis algorithm test results

The results of the application of the image analysis algorithm are next described. We employed a capacitor container with eight randomly dispersed SMD capacitors (no occlusions) for these tests, as depicted in Figure 12(a). The image analysis algorithm sets a cross mark representing the object's geometric center and the container's inclination angle. With this information, each capacitor is chosen, picked up, and placed in a predefined experimental circular pattern, as depicted in Figures 12 (b) and (c). 100 tests were conducted. In 20 of them, the robot failed to place some of the components accurately.

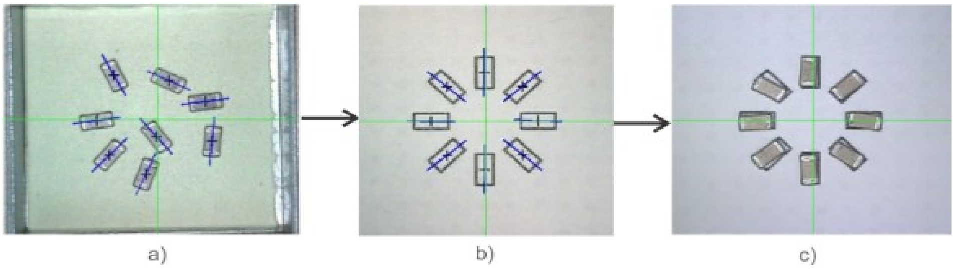

Figure 12. a) Localization, b) Object Tag and c) Placement of component in a circular pattern.

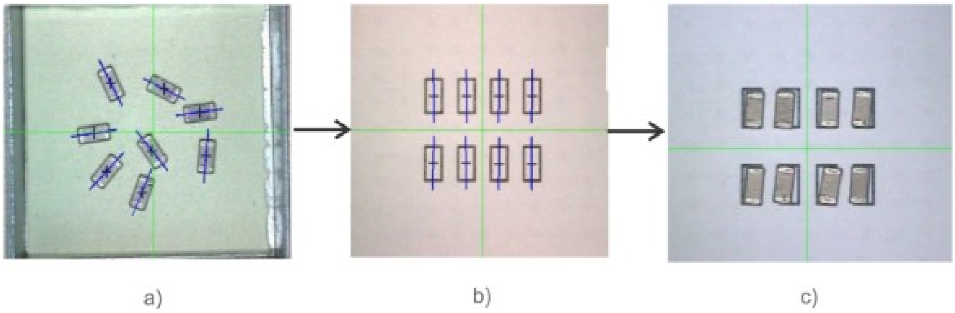

Figure 13. a) Localization, b) Object Tag and c) Placement of component in a vertical pattern.

Despite the system failing in 20 of the experiments to place the 8 capacitors, the robotic machine managed to properly place at least six of the eight elements. These test results provide an effective operation of around 95%. By using another predefined experimental vertical pattern, the second set of tests was conducted. See, Figure 13(a), (b), and (c), where similar results were obtained (also around 95% efficiency). RGB led lamps were employed as the light source to illuminate the working area. Blue light allowed better machine performance in comparison with other colors.

5 Performance Conditions

Here we delimit the performance of the working conditions obtained. As it was mentioned in the Abstract Section, a design and implementation of an algorithm for image analysis was successfully achieved. To identify, locate and manipulate the SMD capacitors used for testing the system, we consider that they are randomly dispersed in a container within the robot work area. The SMD components should not be stacked, so it is possible to recognize them from a top view using a video camera when the Cartesian robot approaches the capacitor container. Under this scenario, the algorithm for image analysis identifies and locates each capacitor in the container, registering its location. Then, the robot takes each capacitor with its pneumatic suction cup. Next, the robot approaches the capacitor to the PCB to locate a rectangular predefined vision mark of 3.2mm × 1.6mm. Lastly, the robot automatically places the capacitor firmly over the mark defined on the PCB. A limit sensor helps to accomplish this last action.

We evaluated the effectiveness of the SMD capacitors placing tasks by measuring the obtained area from the pixels pattern between the rectangular mark in the PCB and the mounted capacitor to be set correctly, with 100% of the pixels matching.

6 Conclusions

The design and implementation of this automatic pick and place robotic machine solved the SMD manipulation problem that originated from using hand-assembly pick and place machines, achieving an overall accuracy of ±0.035 mm in linear positioning, representing an integration error of 2.2%.

The performance related to the rotary movements at the suction cup is not satisfactory enough, providing precision in terms of degrees of the order of ±5°. Despite these deficiencies, it was possible to control the motors' speed according to the different setpoint values without excessive forces that would damage the components.

As for the image analysis algorithm, it was possible to effectively locate the SMD capacitor components' center of mass and orientation. The error that results in the mark orientation calculation reached an overall value was of ±2.5°, approximately. Then, by integrating the error that provides the mechanical precision associated with the Cartesian robot and the error from the image analysis algorithm, this could deviate to a maximum of ±15° the SMD components' placement over the predefined marks in the component destination place.

One of the main contributions of this work, compared to other reported research [2], [3], [4], [12], [15], [16], [17], [18], [19], [20], and [21], is that our system operates with components arranged in containers instead of parts set in supply reels, enabling low volume and cost production of PCB prototypes with isolated SMD components.

7 Future Work

In this work, type 1206 SMD capacitors were used as test elements. Therefore, we propose improving the image analysis algorithms to recognize many SMD components, including diodes, resistors, and integrated circuits with different dimensions as [1]. Additionally, it is necessary to improve the mechanical design, incorporating a planetary reduction system to the motor that drives the rotary movement for the *w*-axis to achieve a finer positioning control.

It is intended to continue this work but now adding artificial intelligence (AI) to recognize the PCB components, with a large number of experiments in order to reduce errors. With these results, it is necessary to make a detailed analysis of the efficiency by the use of other kind of methods reported in the literature to place SMD components in PCB prototypes, using automatic Pick & Place machines.

Finally, given that the main motivation of the system is to automatically pick up dispersed components in a container, through the image analysis system, a next step will also involve the development of the necessary software modules that allow to read CSV coordinate files to determine the position and rotation of each SMD components in the desired location within the working space of the Cartesian robot, without the need to use the vision system and guides predisposed on the PCB board.

8 Acknowledgments

We appreciate the support to develop this project by the Instituto Politécnico Nacional (IPN) and Secretaría de Investigación y Posgrado (SIP-IPN) under the projects SIP20201397, SIP20200885, SIP20211657, SIP20200569, SIP20210124, and 20210788, Comisión de Operación y Fomento de Actividades Académicas (COFAA-IPN), also to Consejo Nacional de Ciencia y Tecnología (CONACYT-México) under projects 65 (Frontiers of Science 2015) and 6005 (FORDECYT-PRONACES).

References

1. Huang R, Gu J, Sun X, Hou Y, Uddin S. A Rapid Recognition Method for Electronic Components Based on the Improved YOLO-V3 Network. Electronics. 2019; 8(8).
2. Ardhy F, Hariadi FI. Development of SBC based machine- vision system for PCB board assembly Automatic Optical Inspection. In: 2016 International Symposium on Electronics and Smart Devices (ISESD); 2016. p. 386–393.
3. Kim Y, Lim D, Ryu J, Park T. SMD Defect Classification by Convolution Neural Network and PCB Image Transform. Kathmandu Nep: Springer; 2018.
4. Andhare P, Rawat S. Pick and place industrial robot controller with computer vision. In: 2016 International Conference on Computing Communication Control and Automation (ICCUBEA); 2016. p. 1–4.
5. Novastarsite. Novastar (visited 2021 April 10). Available from: http://www.ddmnovastar.com/pick-and-place/manual-systems/MPP-21-pick-and-place-system.
6. Kaehler A, Bradski G. Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library. 1st ed. California (USA): Sebastopol, CA, USA: O'Reilly Media, Inc.; 2016.

7. Wolm P, Chen X, Chase J, Pettigrew W, Hann C. Analysis of a PM DC Motor Model for Application in Feedback Design for Electric Powered Mobility Vehicles. In: 15th International Conference on Mechatronics and Machine Vision in Practice. Auckland N.Z: IEEE-Xplore; 2-4 Dec, 2010. p. 640–645.
8. Ästrom K, Hagglund T. Advanced PID Control. ISA-The Instrumentation, Systems, and Automation Society; 2009.
9. Pichardo-Centeno J. Compensación de zona muerta y variación de carga en el control de velocidad de motores de corriente directa. CICATA, Instituto Politécnico Nacional, Qro, México. 2008.
10. Muhammad R. Power Electronics Handbook: Devices, Circuits and Applications. 2nd ed. Academic Press; 2010.
11. Wei L, Jiao Z. Visual Location System for Placement Machine Based on Machine Vision. In: Fifth IEEE International Symposium on Embedded Computing. Beijing Ch.; 6-8 October, 2008. 141–146.
12. Ngadimin J, Hariadi F, Arsyad M. Design and Implementation of 3D Motion Control of Small Scale Pick and Place Surface- Mount Technology Machine. In: International Symposium on Electronics and Smart Devices (ISESD). Yogyakarta Ind.; 17- 19 Oct, 2017. p. 95–100.
13. Pajares-Martinsanz G , Santos-Pajares G, De la Cruz-García J. Ejercicios Resueltos de Visión por Computador. 1st ed. Ra-Ma SA. Editorial y Publicaciones; 2007.
14. Sossa-Azuela J. Rasgos Descriptores para el Reconocimiento de Objetos. 1st ed. Ra-Ma SA. Editorial y Publicaciones; 2013.
15. Vega A, Felipe S, Déniz A. Control Software for Pick & Place Machines. In: in Proc Technologies Applied to Electronics Teaching, TAEE. Sevilla Spain; 22-24 Jun, 2016. p. 1–7.
16. Dömel A, Kriegel S, Brucker M, Suppa M. Autonomous pick and place operations in industrial production. In: 2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI); 2015. p. 356–356.
17. Kato G, Onchi D, Abarca M. Low cost flexible robot manipulator for pick and place tasks. In: 2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI); 2013. p. 677–680.
18. Li W, Xiao Y, Bi S, Du G. Automatic elliptical trajectory planning algorithm for pick and place operation. In: Proceedings of the 2013 International Conference on Advanced Mechatronic Systems; 2013. p. 36–39.
19. Kim J, Lee W, Kang S. Pick-and-place task with manipulator by modular approach. In: 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI); 2017. p. 202–203.
20. Tian S, Chen Y, Gu Q, Hu R, Li R, He D. Optimal Path Planning for a Robot Shelf Picking System. In: 2020 39th Chinese Control Conference (CCC); 2020. p. 3898–3903.
21. Jin J, Yuen SL, Lee YH, Jun C, Kim YB, Lee S, et.al. Minimal Grasper: A Practical Robotic Grasper With Robust Performance for Pick-and-Place Tasks. IEEE Transactions on Industrial Electronics. 2013;60(9):3796–3805.
22. CIC-IPN P&P robotic machine: https://www.youtube.com/watch?v=L2T19zglMCg.
23. Arévalo-Hernández J.L. Automatización de una herramienta manual para el montaje de componentes electrónicos. Master thesis. CIC-IPN; June 2012.
24. Jolion J.M, Montantvert A. The Adaptive Pyramid Framework for 2D Image Analysis, Comput. Vision Graphics Image Process.: Image Understanding 55 (3); 1992. p. 339-348.
25. Marfil R, Molina-Tanco L, Bandera A, Rodríguez J.A, Sandoval F, Pyramid Segmentation Algorithms Revisited. Pattern Recognition (39); 2016. p. 1430-1451.