Vol 1 (Iss. 1), pp. 165-171, 2024 African Scientific Annual Review www.asarev.net ISSN: 3007-0902

Robot Manipulator Programming Via Demonstrative-Kinesthetic Teaching for Efficient Industrial Material Handling Applications

Griffin Pilot Mabong[1]
Emmanuel A. E. Osore[2]
Peter T. Cherop[3]

[1]seig01-701902022@student.mmust.ac.ke
[2]eosore@mmust.ac.ke
[3]pcherop@mmust.ac.ke

[1,2,3]Department of Mechanical and Industrial Engineering, Masinde Muliro University of Science and Technology, Kenya

ABSTRACT

The Fourth Industrial Revolution (4IR) is on course for onboarding by the various manufacturing industries. The fluctuating nature of consumer needs has made it necessary to renovate existing industry approaches to industrial activities. This is an attempt to hold a competitive edge over conventional manufacturing activities. The development of cyber-physical systems (CPS), technological advancements, and the integration of computers in manufacturing processes have seen the introduction of robotized systems to aid in the processes. This has led to the start of smart factories, hence smart manufacturing. Today, robots are deployed in social places for entertainment, hospitals for telemedicine, industries for various activities, and fields of exploration. The robot programming techniques have significantly advanced from traditional text-based language programming (TLP) and offline programming (OLP) to various paradigms of programming by demonstration (PbD), such as tele-kinesthetic teaching (TKT) and demonstrative-kinesthetic teaching (DKT). This paper aimed to program a robotic manipulator using DKT and contrasted it against programming using structured text for industrial material handling applications. The study was validated via palletizing and contour path welding experiments. The materials for the experiments were a robotic manipulator (Dobot Magician), a laptop, USB cables, and stopwatches. A control platform for the arm was created using Microsoft Visual Studio code. The platform allowed arm manipulation, demonstratively using the hand-held trigger button on the forearm to capture motions. The palletizing and contour path weld experiments used the DKT and structured texts as programming modes. The results showed a shorter DKT reaction time than structured texts when conducting the palletizing experiment. The palletizing experimental time was lesser than when performed over the structured texts. The contour path welding indicated an equal reaction time for both programming modes but comparable experimental times. The efficiency of the DKT was taken on account of comparison of the reaction and experimental time to the structured texts, which served as the control experiment. The conclusion was that DKT was more efficient as a manipulator programming method than structured texts. Future works aimed at actual contour path welding and improvement of the control platform to a more user-friendly interface.

Key words: Dobot Magician, Demonstrative-Kinesthetic Teaching, Robot Manipulator, Material Handling, Palletizing

I. INTRODUCTION

An industrial robot is a programmable automatic system, mechanically controlled, with multiple degrees of freedom, which can be stationary or mobile [1]. A robotic manipulator is an electro-mechanical device consisting of joints and links that are driven by motors or other actuators [2]. The manipulator should be considered more than just a set of mechanical components [3]. It comprises software programming, sensors and actuators, a computer interface, a source of power, and a gripper [4, 5]. The efficiency of the manipulator can be determined by taking note of the reaction time, time taken to complete a task, and margin of error [4]. Conventional or manual programming methods, such as text-based, graphical, offline programming (OLP), and teach-pendant programming [6], are tedious, non-intuitive, and laborious [7]. Automatic programming means are learning from demonstration (LfD) or programming by demonstration (PbD, reinforced learning (RL), augmented and virtual reality (AVR), machine learning (ML) technologies, speech-recognition-based, one-shot learning [8, 9], and kinesthetic teaching (KT).

Kinesthetic teaching or guidance is a programming approach where the programmer shows new behaviours via learner robot body manipulation as it records through its sensors (proprioception) [10-12]. The techniques employed are physical manipulation (DKT) and robot movement control through interfaces (tele-kinesthetic teaching (TKT). TKT technique offers an opportunity for remote programming. Still, it faces the limitations of additional lengthy user training on the interfaces, availability of the chosen input hardware, and additional effort required to develop the selected interface. The DKT technique naturally allows for programming; the onboard sensors record the state of the robot during interaction [13], provides an intuitive approach with minimal training requirement [14, 15] as it does not burden the programmer with the requirement of knowledge of programming languages such as Python [16]. It provides an avenue for exploring the physical human-robot interaction (pHRI) [17].

Kinesthetic teaching as a robot programming method has been applied in controlling industrial robots using accelerometers[18], pick and place operations[19], acquiring robust robot skills [20] learning grasping poses in assembly tasks[21] and programming a robot to provide real-time feedback using haptics[22]. This study aimed to program a robot manipulator using DKT for efficient industrial material handling applications. The study was validated by conducting palletizing and contour path welding experiments.

II. MATERIALS AND METHODS

The experiment materials were a robotic manipulator (Dobot Magician), laptop, wooden block, USB connection cables, power source, stopwatches, and a pneumatic gripper.

2.1 Description of Robotic Manipulator

The Dobot Magician is a multifunctional desktop manipulator that allows anyone from essential to expert programmers to carry out practical tasks. It is a 4-DOF articulated robotic arm with 4- axes with 1-DOF contributed by the gripper movement. It has a position repeatability of 0.2mm, a work envelope of 320mm maximum reach, and a payload of 0.5kg [23]. It consists of the forearm, rear arm, end effector, and base for support, as shown in Figure 1.

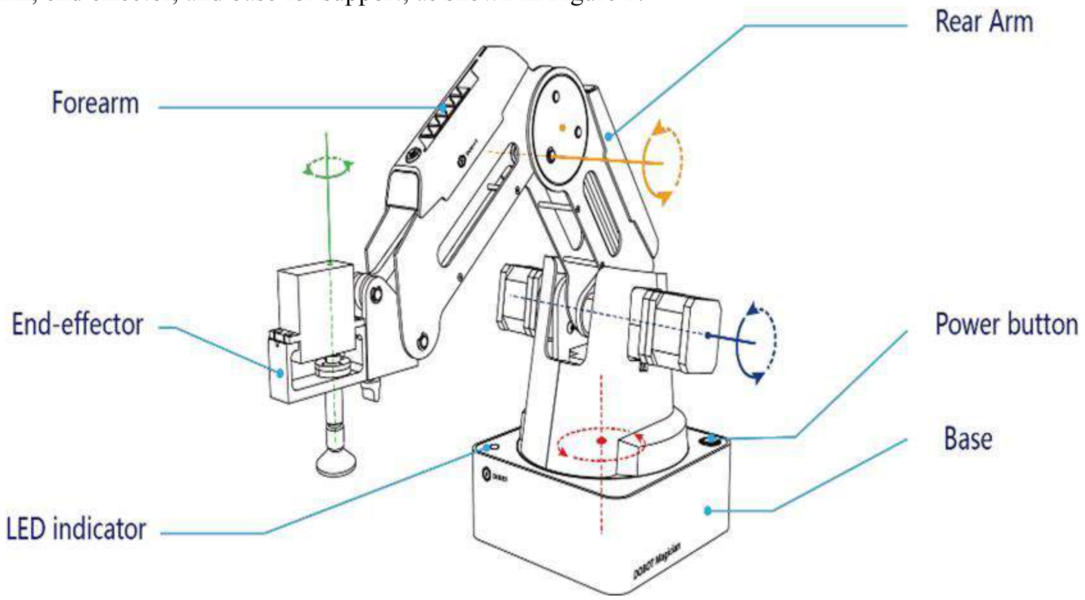

Figure 1
Dobot Magician (Adapted from Shenzhen Yuejiang Technology Co. [23])

2.2 Experimental Set-up and Procedure

The experimental setup was as shown in Figure 2.

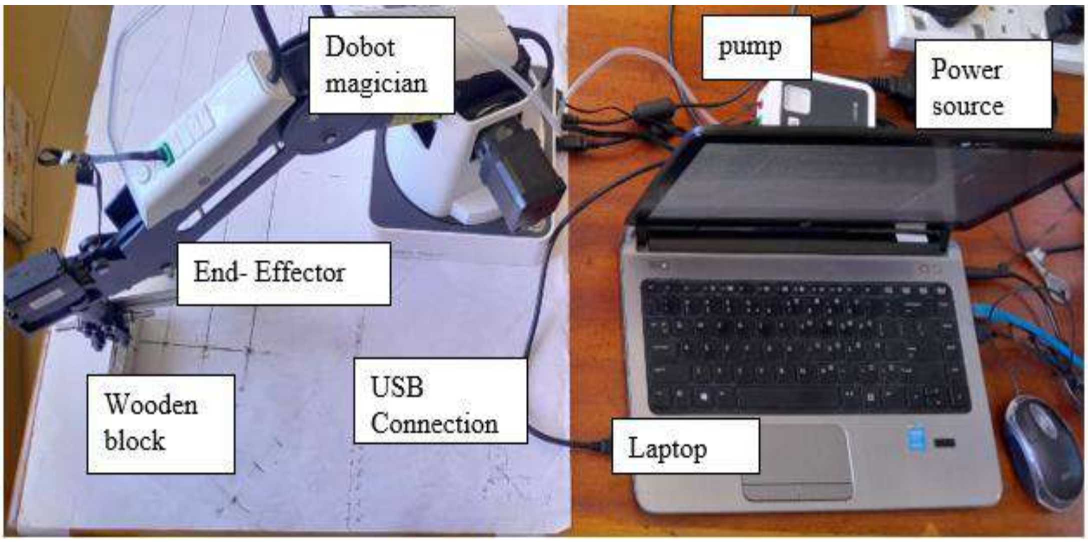

Figure 2
Experimental Set-up

The Dobot Magician was connected to the laptop via a USB connection. A control platform in Figure 3, based on Python language, was created in Microsoft Visual Studio Code integrated development environment (IDE) to map the arm movements to the specific control commands.

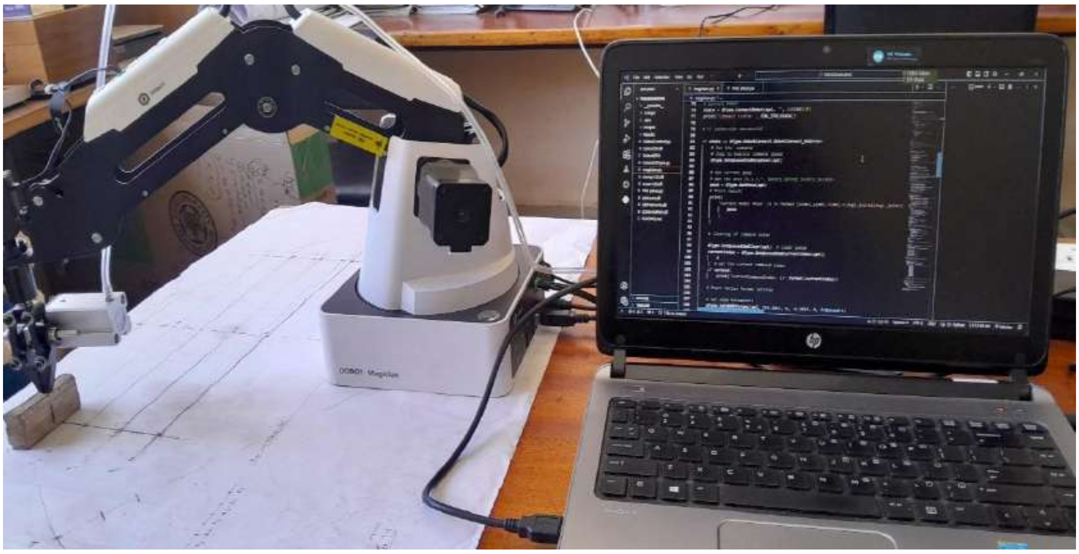

Figure 3
The Control platform in VS Studio Code and Dobot Magician.

The arm was programmed on the kinds of motion to undertake by pressing the lock button on the forearm. This allowed arm programming by demonstrations; thus, DKT. The arm was programmed to pick the wooden block from pick point A and place it at B. This was done over various intervals of A and B. The position of points A and B were determined for distances 25mm, 30mm, and 35mm apart from the initial position of A_1 (Start) and B_1 (Stop) on a straight line as the stop points of the experiment as shown in Figure 4 and Figure 5.

Figure 4
Dobot Magician at Pick Point A

Figure 5
Dobot Magician at Place Point B

The same palletizing (pick and place) experiment was done by programming the Dobot Magician using structured texts. The commands in the code to move the arm to various positions and the motion speed, velocity, and acceleration were used to program the arm to experiment. The stopwatches were used to measure the experimental and reaction times during the experiments. The times were recorded for the DKT and the structured texts.

The contour path welding experiment was done by programming the Dobot Magician demonstratively using the DKT technique with the help of the control platform in Figure 3, to collect the joint poses for three points, A, B, and E, along arcs of different radii to indicate the various replication points carried out at each arc level. Points A, B, and E all lay on the same arcs, but points A and E were not on straight lines at the different levels. Point A was the starting point, B between A and E, and E was the endpoint as shown from Figure 6 to Figure 8. For the contour path welding, the experiment primarily focused on programming the robot manipulator to follow the contour path without the actual welding process.

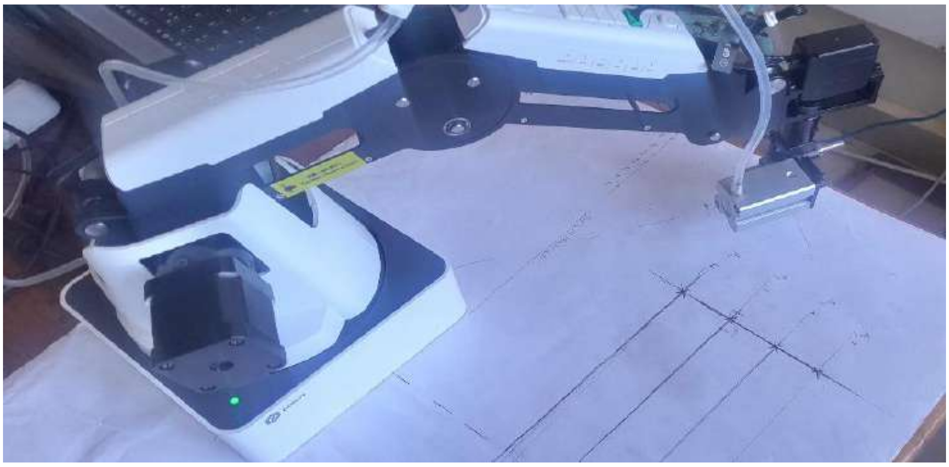

Figure 6
End-effector at Start Position A_1 on Contour1(arc1)

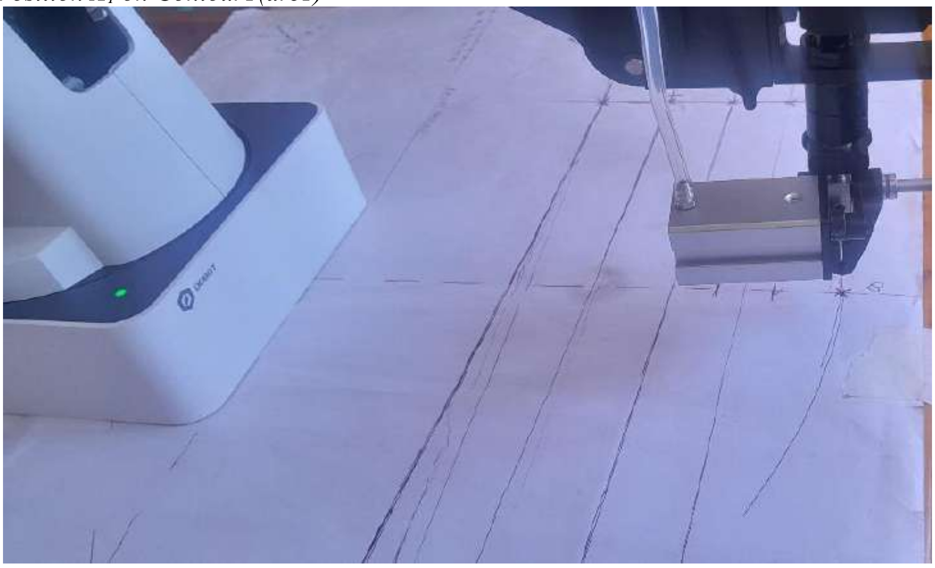

Figure 7
End-effector at Mid-Position B_1 on Contour1(arc1)

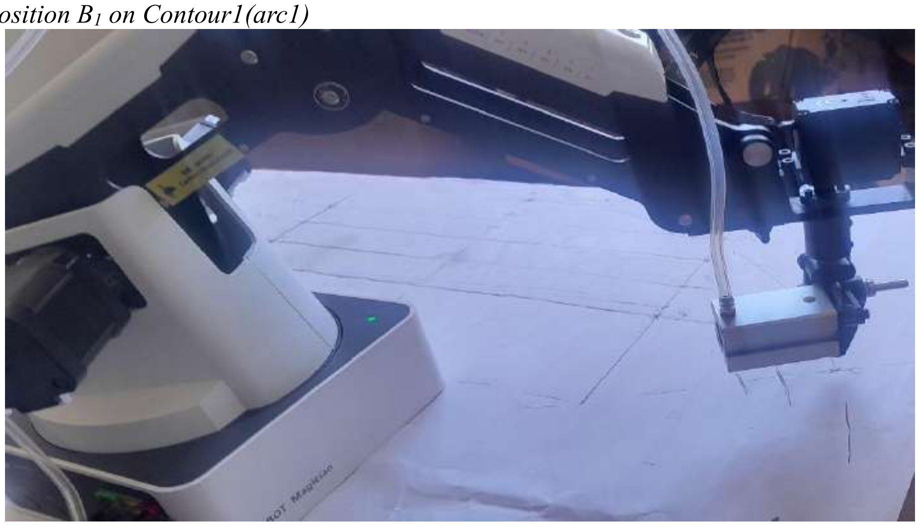

Figure 8
End-effector at End Position E_1 on Contour1(arc1)

III. RESULT & DISCUSSION

The time duration for tasks results for the palletizing experiment were as summarized

Table 1
Time Durations for Tasks (Palletizing)

Serial	Activity	DKT	Control
1	Robot Reaction Time (average)	3 seconds	3 seconds
2	Experimental Time (average)	3 mins 59s	5 mins 01s

For the palletising application, time required for experimenting was an average time of approximately 3 minutes and 59 seconds for DKT whereas approximately 5minutes and 01 second for the Control as shown in Table 1. The experiment was carried out for various experimental trials and a graph of the same was as shown in Figure 9.

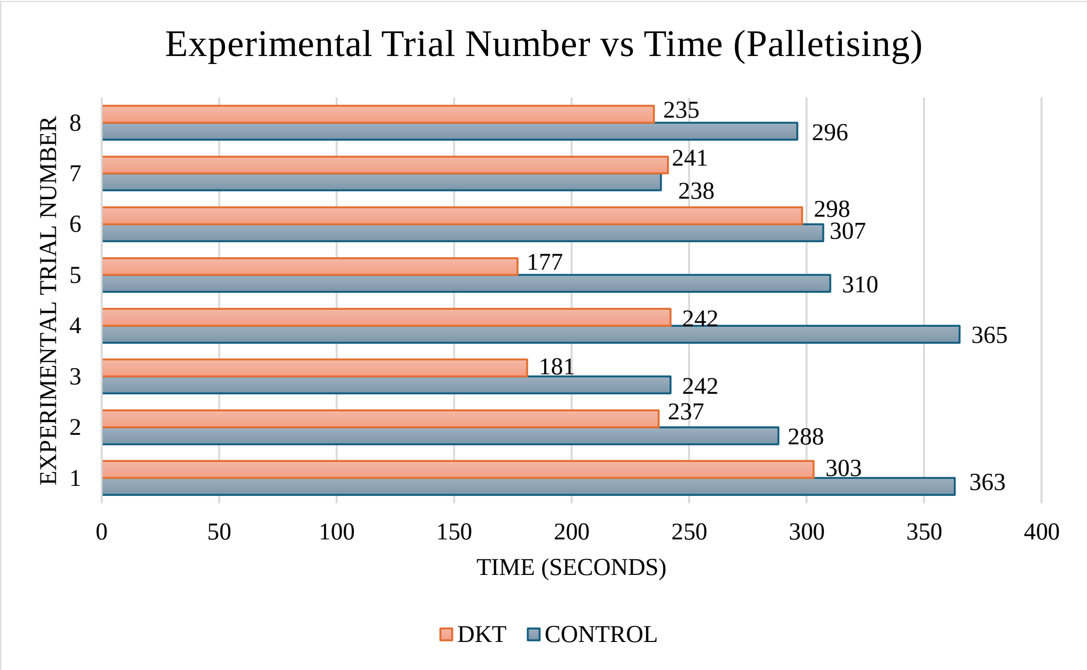

Figure 9
Palletizing Experimental Time

For the contour path welding application, the time required for carrying out the experiment was an average time of approximately 30s for DKT whereas approximately 45s for the Control as shown in Table 1, and Figure 10 for the two programming modes.

Table 2
Time Durations for Tasks (Contour Welding Path)

Serial	Activity	DKT	Control
1	Robot Reaction Time (average)	1s	2s
2	Experimental Time (average)	30s	45s

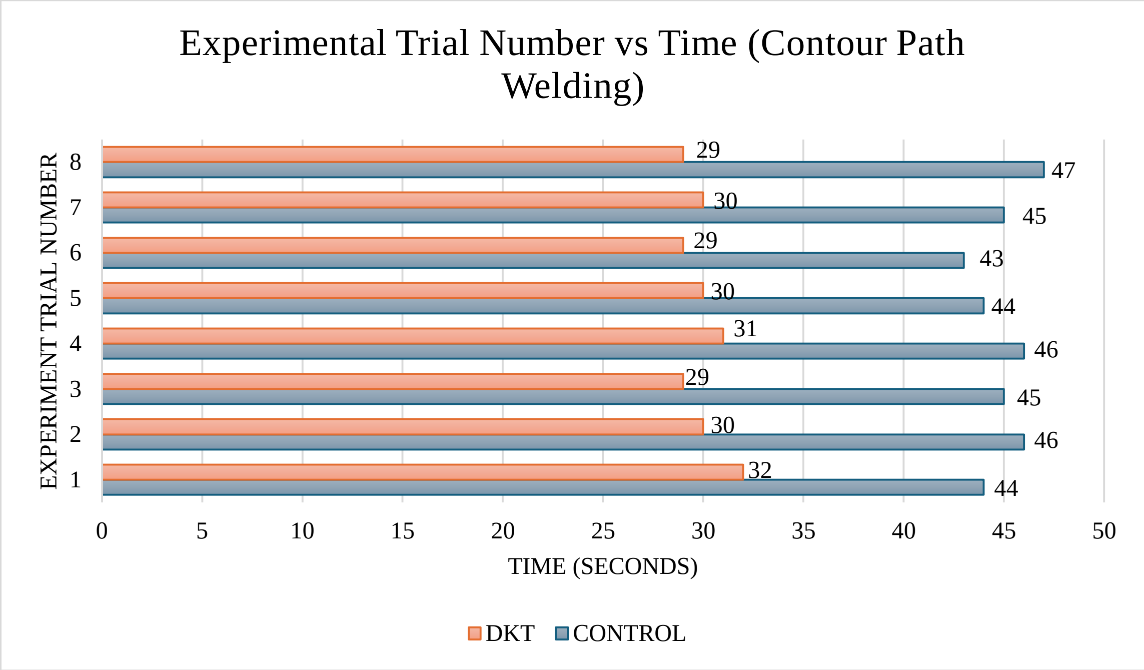

Figure 10
Contour Path Welding Experimental Time

From Table 2, the robot programming time of the contour path welding was lower than that of the palletising task, as the activity's complexity was lower. In this experiment, the robotic arm was only programmed to follow the contour path, reducing the time. The structured-texts approach required rechecking every code to determine whether the correct positions were entered for A, B, and E on all the arcs in the codes. There was no need for gripper function implementation in the code as the trajectory was only a matter of concern; hence, it took a shorter time than the palletising task.

The robot reaction time was also reduced during this task as the number of steps in the code were significantly reduced compared to the palletising code, which had functional code lines that required more processing time. However, the DKT method had the least time of 1 second compared to the 2 seconds of the text-based control, as the trajectory was stored in the robot's memory while being programmed. The text-based code required the robot's keen understanding of every line and the motion to be executed to reach the desired point, hence a longer reaction time.

The reduced complex nature of the contour path welding also meant reduced experimental times, hence the 30 seconds for the DKT technique and 45 seconds for the text-based control on average. The 15 seconds longer by the text-based control was attributed to the fact that the robot had to go through each of the lines of codes stepwise while executing the code as compared to the DKT technique, in which the robotic arm had already stored the motion and trajectory in its memory. From

Table **1**, it was seen that the reaction time of the robotic arm for both methods was the same, 3 seconds. This resulted from both methods relying on the execution of codes to connect to the robotic arm run by the same system. The DKT platform was executed on the Python code to control it, the same as the control code in Python for the text-based programming approach.

The programming time involved for both methods has differed sharply as it took a long period of time to code the Control code for the pick and place. This was for a more extended period compared to the DKT technique as an accurate code performance was required, hence writing and rechecking the code at every step of the code. The target points and the trajectory needed had to be considered to determine the kind of robotic motion required, whether MOVL, MOVJ, or JUMP, for the correct movement to the set coordinates.

While the DKT technique took short amount of time to program the robot, a huge chunk of the time was used to ensure the gripper performed as wanted, as it still required coding aspects for the opening and closing of the arms. Getting the right amount of time needed for the gripper hydraulics to pressurise for efficient functioning was highly considered. The actual demonstrative aspect only helped the robot to identify the positions of the pick points using the release and locking of the lock arm and the trajectory to be taken.

From

Table **1** and Table 2, it can be said that the DKT technique is more efficient than the text-based Control programming approach owing to the less time taken to execute the experiments. The text-based programming approach was quite cumbersome and tedious compared to the DKT, which, on the other hand, was more manageable, hence the less time. The experimental times also showed how, on average, it was more efficient to use the DKT technique than text-based programming, with 1 minute less for palletising and 15 seconds less for contour path welding experiments, respectively.

IV. CONCLUSION

The Dobot Magician robot manipulator was programmed using the DKT technique and structured texts, which acted as the control experiment for the study. Compared to the structured text of robot programming, DKT proved to be better for programming, especially for workforces that are not proficient in programming languages and semi or non-skilled. While the method is more straightforward and quite going for some tasks, it may prove futile, especially where the geometry of items is rather complex. Where the materials to be handled are hazardous, thus requiring distance between the material and the demonstrator, other methods, such as TKT, teach-pendant, and text-based, may be more helpful in handling such operations. Future works aim at improving on the user interface for the control platform, carrying out the actual welding and incorporating of sensors to detect obstacles.

REFERENCES

[1] Bartoš, M., Bulej, V., Bohušík, M., *et al.* (2021) *An overview of robot applications in automotive industry.* Transportation Research Procedia. **55**: p. 837-844.

[2] Jahnavi, K. and Sivraj, P.(2017) *Teaching and learning robotic arm model*. 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT).1570-1575. IEEE.

[3] Abdelaal, M.(2019) *A study of robot control programing for an industrial robotic arm*. 2019 6th International Conference on Advanced Control Circuits and Systems (ACCS) and 2019 5th International Conference on New Paradigms in Electronics and information Technology (PEIT).23-28. IEEE.

[4] Kohrt, C., Stamp, R., Pipe, A., *et al.* (2013) *An online robot trajectory planning and programming support system for industrial use.* Robotics and Computer-Integrated Manufacturing. **29**(1): p. 71-79.

[5] Kohrt, C., Pipe, A., Schiedermeier, G., *et al.*(2008) *A robot manipulator communications and control framework*. 2008 IEEE International Conference on Mechatronics and Automation.846-851. IEEE.

[6] Amar, R.H.E., Benchikh, L., Dermeche, H., *et al.* (2020) *Efficient trajectory reconstruction for robot programming by demonstration.* Rev. Roum. Sci. Techn.–Électrotechn. et Énerg. **65**: p. 1-2.

[7] Zhou, Z., Xiong, R., Wang, Y., and Zhang, J. (2020) *Advanced robot programming: A review.* Current Robotics Reports. **1**: p. 251-258.

[8] Mosavi, A. and Varkonyi, A. (2017) *Learning in robotics.* International Journal of Computer Applications. **157**(1): p. 8-11.

[9] Orendt, E.M., Fichtner, M., and Henrich, D.(2016) *Robot programming by non-experts: Intuitiveness and robustness of one-shot robot programming*. 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN).192-199. IEEE.

[10] Zieliński, C.(1995) *Robot programming methods*. Oficyna Wydawnicza Politechniki Warszawskiej.

[11] Villani, V., Pini, F., Leali, F., *et al.* (2018) *Survey on human-robot interaction for robot programming in industrial applications.* Ifac-PapersOnline. **51**(11): p. 66-71.

[12] Calinon, S. (2018) *Learning from demonstration (programming by demonstration).* Encyclopedia of robotics: p. 1-8.

[13] Ravichandar, H., Polydoros, A.S., Chernova, S., and Billard, A. (2020) *Recent advances in robot learning from demonstration.* Annual review of control, robotics, and autonomous systems. **3**: p. 297-330.

[14] Tykal, M., Montebelli, A., and Kyrki, V.(2016) *Incrementally assisted kinesthetic teaching for programming by demonstration*. 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI).205-212. IEEE.

[15] Eiband, T., Liebl, J., Willibald, C., and Lee, D. (2023) *Online task segmentation by merging symbolic and data-driven skill recognition during kinesthetic teaching.* Robotics and Autonomous Systems. **162**: p. 104367.

[16] Heimann, O. and Guhl, J.(2020) *Industrial robot programming methods: A scoping review*. 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA).696-703. IEEE.

[17] Landi, C.T., Ferraguti, F., Sabattini, L., *et al.*(2017) *Admittance control parameter adaptation for physical human-robot interaction*. 2017 IEEE international conference on robotics and automation (ICRA).2911-2916. IEEE.

[18] Neto, P., Pires, J.N., and Moreira, A.P.(2009) *Accelerometer-based control of an industrial robotic arm*. RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication.1192-1197.

[19] Tanwani, A.K. and Calinon, S. (2016) *Learning robot manipulation tasks with task-parameterized semitied hidden semi-Markov model.* IEEE Robotics and Automation Letters. **1**(1): p. 235-242.

[20] Hersch, M., Guenter, F., Calinon, S., and Billard, A. (2008) *Dynamical system modulation for robot learning via kinesthetic demonstrations.* IEEE Transactions on Robotics. **24**(6): p. 1463-1467.

[21] Zhu, Z., Hu, H., and Gu, D.(2018) *Robot performing peg-in-hole operations by learning from human demonstration*. 2018 10th Computer Science and Electronic Engineering (CEEC).30-35. IEEE.

[22] Valdivia, A.A., Habibian, S., Mendenhall, C.A., *et al.* (2023) *Wrapping Haptic Displays Around Robot Arms to Communicate Learning.* IEEE Transactions on Haptics. **16**(1): p. 57-72.

[23] Shenzhen Yuejiang Technology Co., L. Dobot Magician User Manual Available from: https://www.dobot-robots.com/service/download-center[cited 2023 18-07]