Robotica (2022), **40**, pp. 464–474
doi:10.1017/S026357472100059X

CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

A lead through approach for programming a welding arm robot using machine vision

Mohamed Hosni Mohamed Ali* and Mostafa Rostom Atia

Arab Academy for Science, Technology and Maritime Transport, Sheraton, Cairo, Egypt
*Corresponding author. Email: m-hosni@hotmail.com

Received: 26 November 2019; **Revised:** 24 April 2021; **Accepted:** 26 April 2021; **First published online:** 4 June 2021

Keywords: lead through programming; welding robot; machine vision

Abstract

Welding is a complex manufacturing process. Its quality depends on the welder skills, especially in welding complex paths. For consistency in modern industries, the arm robot is used to accomplish this task. However, its programming and reprogramming are time consuming and costly and need an expert programmer. These limit the use of robots in medium and small industries. This paper introduces a new supervised learning technique for programming a 4-degree of freedom (DOF) welding arm robot with an automatic feeding electrode. This technique is based on grasping the welding path control points and motion behavior of an expert welder. This is achieved by letting the welder move the robot end effector, which represents the welding torch, through the welding path. At the path control points, the position and speed are recorded using a vision system. Later, these data are retrieved by the robot to replicate the welding path. Several 2D paths are tested to assess the proposed approach accuracy and programming time and easiness in comparison with the common one. The results prove that the proposed approach includes fewer steps and consumes less programming time. Moreover, programming can be accomplished by the welder and no need for an expert programmer. These enhancements will improve the share of robots in welding and similar industries.

1. Introduction

Welding is one of the important and widely used manufacturing processes. It is a process to join metals by melting and fusing them, which adds value to the product and economy. In the manual welding process, the weld quality depends on the skillfulness of the welder. However, it is a physically demanding process and leads to repetitive injuries with bad effects on the welder's health. This is due to the heat, radiation, and toxic gases. Due to the disadvantages over manual operation, robotic welding plays an important role in industrial manufacturing in recent years [1]. This is because of its flexibility and intelligence with the potential of performing tasks as well as humans, or even better at acceptable cost and quality levels. The welding robot needs a program to set the trajectory profile and follow the predefined planned path. Robot programming approaches are classified into; off-line method, teach pendant method, and lead-through approach [2].

The off-line programming method includes text-based and graphical programming environments. Text-based programming is based on conventional programming languages. It is time consuming, costly, and needs a well-trained expert programmer. The graphical programming environment provides an alternative to their text-based counterparts as it enables the possibility of using simulation technology tools and robot kinematics [3, 4]. Therefore, the robot is programmed and simulated before the move to the real world. However, this method requires extra time and an expert programmer [5].

In the teach-pendant programming method, the operator uses the teach pendant buttons to move the robot arm from point-to-point and save every point position. The robot can redo the process once

© The Author(s), 2021. Published by Cambridge University Press.

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

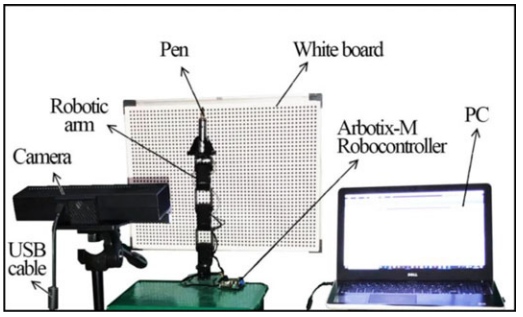


Figure 1. *Experimental setup.*

the whole program has been generated using these points. To define the desired trajectory in welding and painting, there are difficulties due to the distance between the programmer and the workplace [6]. Therefore, the teach-pendant programming method not intuitive especially in workpieces with complex geometry, where it needs rounds of refs. [7, 8]. More challenges are raised when the application includes a common force control tool between the robotic arm and humans such as peg-hole intersection and surface grinding [9].

When compared to other approaches, the lead through method is useful in teaching the robot arm the control points and in between speed along the welding path [10, 11]. It is based on the physical movement of the manipulator by the skilful operator [12], while its motion is recorded then replicated later by the robot. However, one of the existing challenges for generating a robotic arm trajectory is the observation of a human demonstration. To replicate this motion, the programmer requires deep knowledge of robot motion modes and programming steps [13]. Therefore, it is not guaranteed that the tip of the robotic arm reaches the desired point.

The literature review addresses several difficulties in using robots in some processes such as welding. They are the long time of programming and reprogramming, need of an expert in process, and programming and consequently high cost. This limits the propagation of using robots, especially in medium and small industries. Therefore, developing a fast and simple programming method is one of the main goals for the researchers.

This paper aims to introduce a new programming approach based on lead-through programming and a vision system. Its contribution is intended to simplify the programming process to the level of the technicians and reduce the programming and reprogramming time. It is expected that this will reduce the cost and complexity and hence increase the share of the robot in the welding industry.

The experimental setup is explained in detail in Section 2. The method details are described in Section 3. The experimental work including a discussion of the results and a comparison between the proposed approach and common practice is in Section 4. Section 5 is the conclusions and contribution.

2. Experimental setup

This section describes the experimental setup used in this research. The setup is used to test experimentally the proposed programming approach. Figure 1 shows the main components of the setup. It consists of a 4-degree of freedom (DOF) robotic arm with a pen, with 120 mm length, as its end effector. The pen mimics the welding torch for safety and simplifies the test procedure. Besides, the setup includes a personal computer (PC) and a machine vision. The test paths are drawn on a whiteboard with 590 mm width and 440 mm height. It contains a matrix of black dots with 5mm space in X and Y directions. Figure 2 shows an example of a test path.

The robot type is phantom-x pincher robotic arm, which has four revolute joints. Every joint has a smart servo motor running from 0○ to 300○. Its reach (350 mm) vertical and (310 mm) horizontal. It has smart actuators with a daisy chain connected through USB2Dynamixels and controlled by Arbotix-m

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

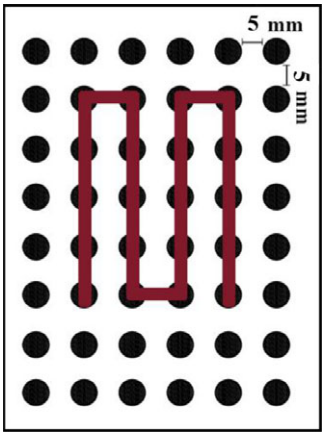


Figure 2. *Test path.*

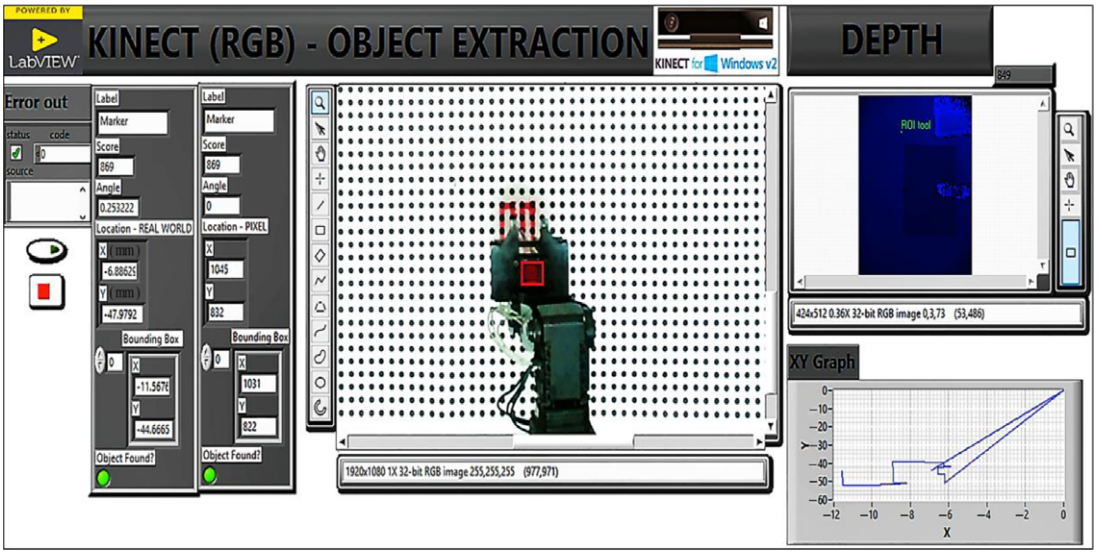


Figure 3. *User interface.*

robocontroller. The smart actuators can record, save, and playback their motions. The robot controller is connected directly to the PC by USB port 2.0. The PC has a processor Intel core i7-7500U Central Processing Unit (CPU) @ 2.70 Gigahertz (GHz) and 32.0 Gigabyte (GB) random access memory (RAM).

The machine vision is Microsoft Kinect one for windows. It is connected to PC through USB 3.0 port in eye-to-hand concept (fixed) and developed using the Software Development Kit (SDK) 2.0 and graphical programming G language, which perceives the depth based on its time-of-flight (ToF) technology [14], as well as a red, green, and blue (RGB) camera to capture videos and images. The captured images are processed using LABVIEW robotics software to identify the pen back location as an indication to the pen tip as shown in Fig. 3. The machine vision is calibrated using National Instruments (NI) vision calibration tool. The aim of this calibration is giving the measurements in pixels a value in mm unit. The camera's axis should be perpendicular as possible to the planned path for less distortion error and the background is dark to enhance light contrast.

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

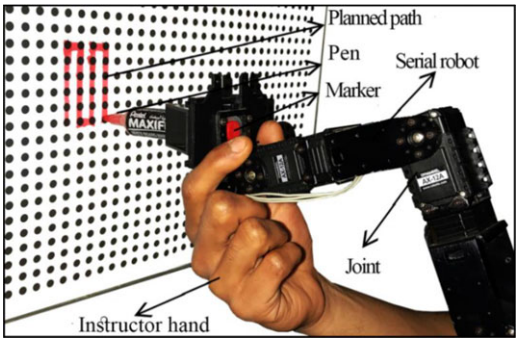


Figure 4. *Proposed approach.*

3. Method description

In this section, the proposed lead-through programming approach is described in steps. The programming is accomplished in two main steps; record and playback. In record step, the operator grasps the stylus pen, as the robot end effector, and follows the desired path on the board as if he welds this path as shown in Fig. 4. The stylus pen back is used as a pointer to the welding tip as the pen length is known and the two ends are coaxial. During this motion, arms angle and angular velocity are recorded by the robot controller. These data are transferred and recorded on the PC. Moreover, the machine vision system records the path control points as the desired reference.

In the playback step, these data are used to generate the robot move-program code using the developed software. This program intends to replicate the original move. The machine vision system observes and records the control points of the actual path of the end effector. These data are used later to assess the system's accuracy.

4. Results and discussions

The experiments have two objectives. The first one is assessing the validity of the proposed approach through measuring programming time and position accuracy. From these data, the sources of errors can be defined. The second objective is comparing the proposed approach with the common one.

4.1. Proposed approach assessment

In this group of experiments, the robot is programmed using the proposed lead-through approach. Six paths are used for testing; three are simple and the other three are complex as shown in Figs. 5 and 6, respectively. For every path, absolute error and executed time are measured. The absolute error can be distinguishable in mm unit by using the machine vision. The summary of the results is shown in Table I.

The first set of paths is distinguished by the simple move in one direction as shown in Fig. 5. Therefore, the mechanical backlash error does not affect the robot hardware. The paths are vertical line shape #1, horizontal line shape #2, and stairs-like line shape #3. In the vertical line, the absolute error range is from 0.0395 to 0.0439 mm and point number 4 is the focal point with 0.0398 mm absolute error in the positive Y direction. While in horizontal shape, the absolute error range is from 0.0698 to 0.0775 mm and point number 4 is the focal point with 0.0699 mm absolute error in the positive X direction. However, in stairs-like line shape, the absolute error is from 0.0807 to 0.0863 mm and point number 7 is the focal point with 0.0807 mm absolute error, which is the intersection point between X, Y positive directions.

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

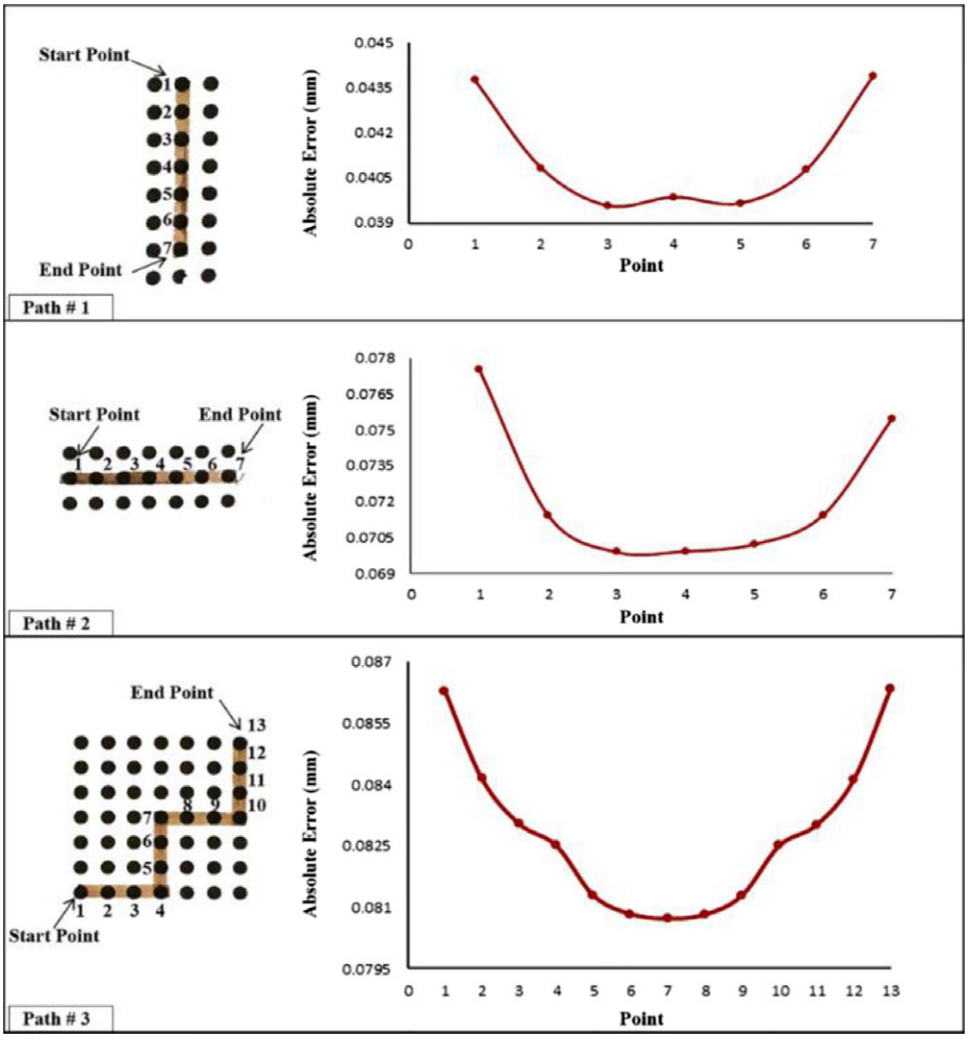


Figure 5. *Test results of simple paths.*

In Fig. 6, all of the experiments applied on the second set are complex moves in two directions such as square shape. In this case, the robot's mechanical backlash is observed in turn points. In path #4, the absolute error range is from 0.0809 to 0.0866 mm and point's numbers 4, 9, 15, 21 are the turn points with 0.0810, 0.0812, 0.0809, and 0.08143 mm absolute errors in *X*, *Y* positive and negative directions. Hence, in path #5, the absolute error range is from 0.1007 to 0.1082 mm and point's numbers 7, 13, 19, 22 are the turn points with 0.0103, 0.0102, 0.1030, and 0.1007 mm absolute errors in *X*, *Y* positive, and negative directions. Finally, in path #6, the absolute error range is from 0.1717 to 0.1908 mm and point's numbers 3, 9, 16, 21 are the turn points with 0.1770, 0.1718, 0.1717, and 0.1769 mm absolute errors in *X*, *Y* positive, and negative directions.

For welding applications, the results show small and acceptable absolute errors in retrieving the original control points on the path. These errors have several sources. One source is the mechanical inaccuracies such as backlash and inertia. This is clear in error in the first control point where the robot is at an arbitrary point before reaching the desired point. In addition, it appears in every sharp turn and when the movement direction is reversed. Another source is the camera axis perpendicular to the path

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

Table I. *Proposed approach results.*

	Path # 1	**Path # 2**	**Path # 3**	**Path # 4**	**Path # 5**	**Path # 6**
Number of points	7	7	13	22	24	24
Total path length (mm)	30	30	60	105	115	115
Average of absolute error (mm)	0.041	0.063	0.083	0.082	0.103	0.178
Standard deviation of absolute error (mm)	0.001	0.009	0.002	0.002	0.002	0.006
Time of programming (min)	0.1	0.1	0.183	0.283	0.333	0.383
Time of execution (min)	0.083	0.083	0.2	0.3	0.35	0.4

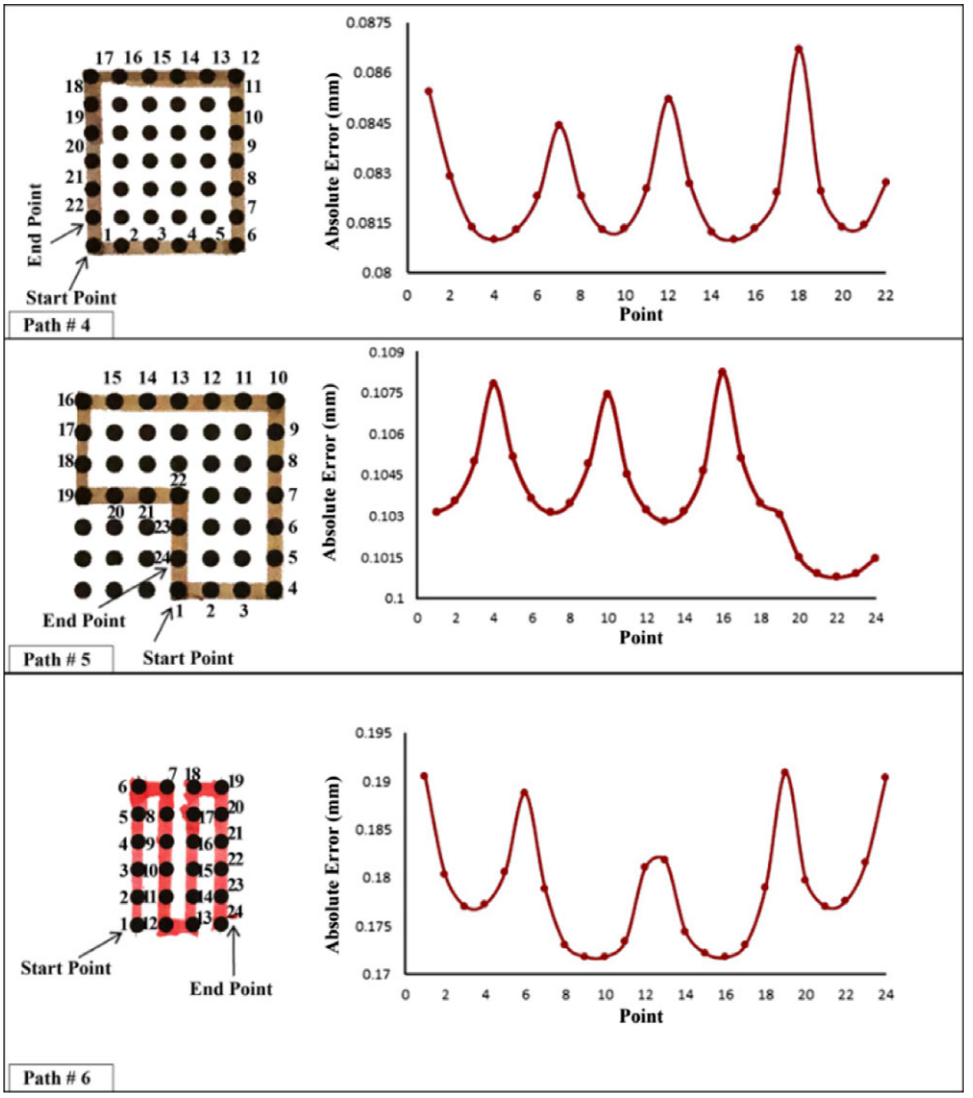


Figure 6. *Test results of complex paths.*

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

Table II. *DH parameters.*

Joint	(Joint angle) *Theta θ*	(Link offset) *d*	(Link length) *A*	(Revolute angle) *a*
1	θ_1	45 mm	0	0
2	θ_2	0	150 mm	0
3	θ_3	0	150 mm	90 deg
4	θ_4	0	90 mm	0

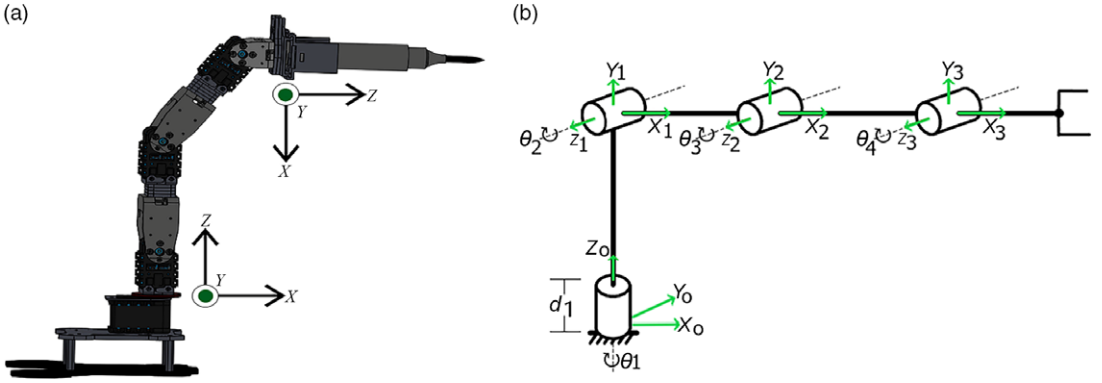


Figure 7. *(a) PhantomX-Pincher—CAD model and (b) coordinate frame assignment. Where: θ_1 = Base, θ_2 = Shoulder, θ_3 = Elbow and θ_4 = Wrist.*

plan. The accuracy of the points degrades with the distance from the focus point. This is clear in straight-line paths. Human error is one of the major sources of errors as the operator follows the path using his vision and move ability during lead-through programming.

4.2. Common practice

In robot programming, the common practice has several main steps. It starts with driving the kinematic model of the robot and calculating Denavit–Hartenberg (DH) parameters. Using the inverse kinematic model, the arms angles are calculated from the desired point coordinates of the end effector. These points are selected to be on the desired welding path with a suitable pitch. A special difficulty in welding operation is that the end effector path has an offset from the welding path and in-between speeds change according to the welding direction. The point coordinates and in-between speeds are the input to the robot program, which is developed by a highly expert programmer. To check the program's accuracy and collision-free moves, an off-line model is used to simulate the program. If the simulation is acceptable, the program is fed to the robot controller for execution.

For comparison with the proposed approach, the common practice approach is applied on four paths in this research. This includes developing the robot's mathematical kinematic model, and visual model using V-Rep simulation software. Next, the text-based robot program is developed by an expert. Then, the program is fed to the robot for testing.

4.2.1. Kinematics modeling

Figure 7(a) shows the used 4-DOFs arm robot with its main axes. Figure 7(b) shows coordinate frame assignment. This model is commonly used in the kinematics chain analysis [15]. It is based on attaching a reference system for every joint and specifying four parameters known as DH parameters for every link. Table II shows the robot parameters used to construct the DH matrix. The MATLAB robotics system

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

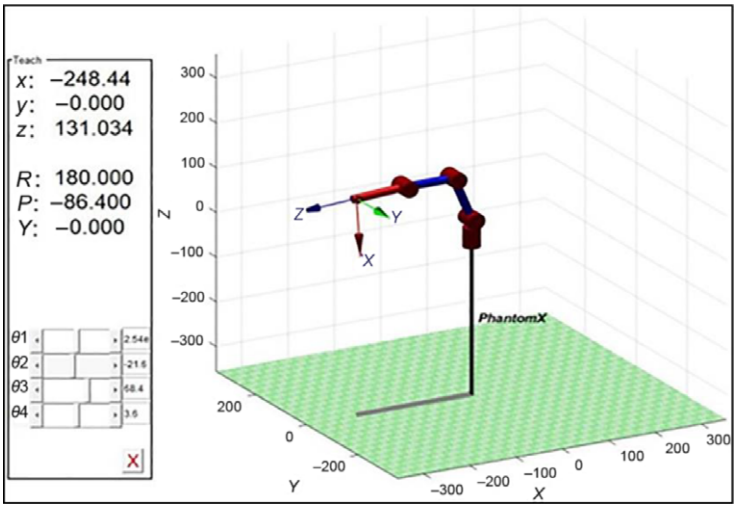


Figure 8. *Numerical kinematics solution at point coordinates (−248.44, 0, 131,034).*

toolbox [16] is used for solving the kinematics problem of the robot. For example, Fig. 8 shows the user interface while displaying the numerical solution of the inverse kinematics problem for the end effector coordinates at a certain point. Based on DH convention, the transformation matrix from joint i to joint $i+1$ is given by:

$$Ti^{i-1} = \begin{pmatrix} \cos\theta i & -\sin\theta i\cos\alpha i & \sin\theta i\sin\alpha i & \alpha i\cos\theta i \\ \sin\theta i & \cos\alpha i\cos\theta i & -\sin\alpha i\cos\theta i & \alpha i\sin\theta i \\ 0 & \sin\theta i & \cos\alpha i & di \\ 0 & 0 & 0 & 1 \end{pmatrix} \ldots \quad (1)$$

4.2.2. Modeling and simulation

In this section, a visual model is developed using V-Rep simulation software. Four paths are programmed using the text-based method by an expert programmer based on the robot kinematic solution. Every path program is tested until satisfactory. The consumed time in programming, testing, and simulation is recorded for every path. Then, the written program is fed to the robot controller for execution. Figure 9 shows the four paths during the simulation and the executed paths in the real world. The detailed results are shown in Table III.

4.3. Comparison between the proposed approach and common practice

This section discusses the comparison between the proposed and common practice approaches. There are several comparison points of view. They are work complexity, number of execution steps, sources of errors, and programming time.

From the work complexity point of view, the proposed approach has advantages over the common one. This is because the path points and in-between speeds are grasped directly from the real world with all real manufacturing deviations. However, in the common approach, the path points are calculated using complex mathematics from a theoretical path, which can be introduced as a Computer Aided Design (CAD) file. In addition, the welder can avoid obstacles during the learning step. These advantages decrease the programming complexity, time and effort, and increase the safety of the operation.

Figure 10 shows a process sequences between the steps of the proposed and the common practice approaches. It shows a less step number for the proposed approach. Table IV shows a summary of

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

Table III. *Common practice results.*

	Path # 1	Path # 2	Path # 6	Path # 7
Number of points	6	6	24	8
Total path length (mm)	25	25	115	35
Time of programming (min)	30	30	120	45
Time of execution (min)	0.083	0.083	0.4	0.166

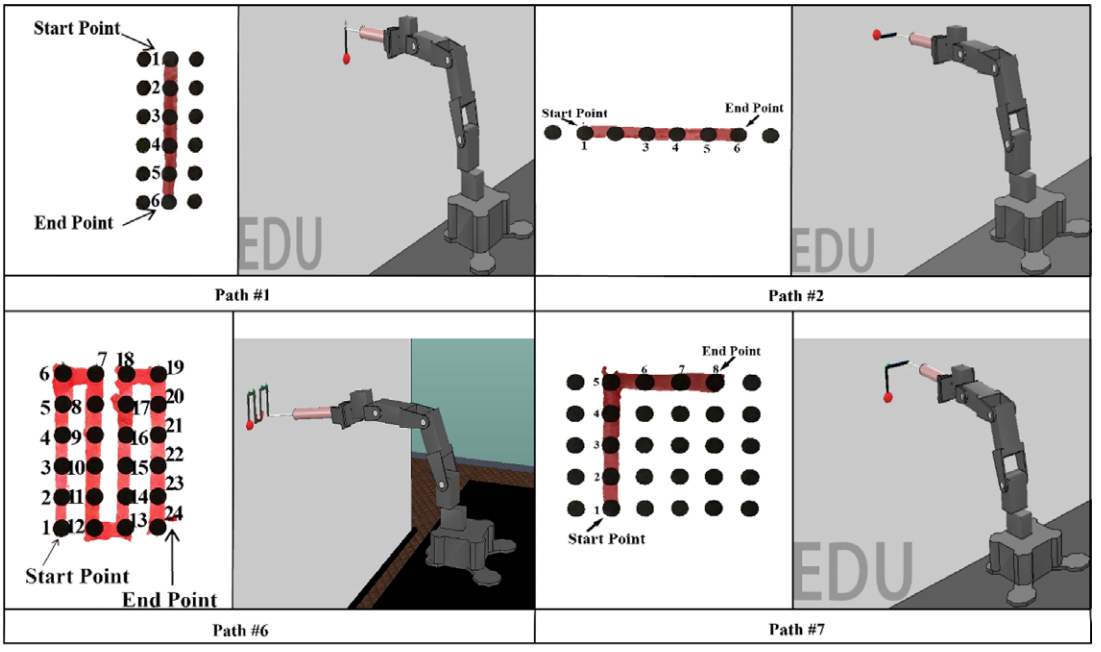


Figure 9. *Robot simulation and the executed paths in the real world.*

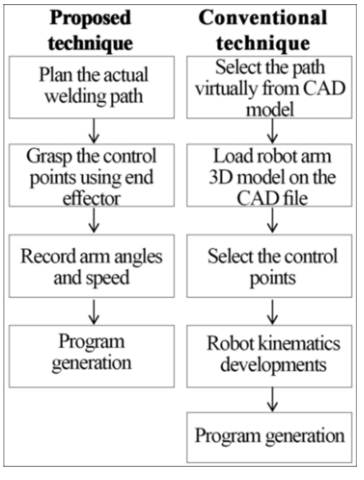


Figure 10. *Process sequences of the proposed approach and the common practice.*

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

Table IV. *Comparison between the proposed approach and the common practice.*

	Proposed approach	**Common practice**
Time of programming	Shorter time	Longer time
Expert programmer	Not required	Required
CAD simulation	Not required	Required
Kinematics solutions	Not required	Required
Product	Small piece	Large piece

Table V. *Source of error.*

Source of error	
Proposed approach	Common practice
	Line text-based error
Human hand motion error	CAD model
	Length of welding torch
	Reference point identification

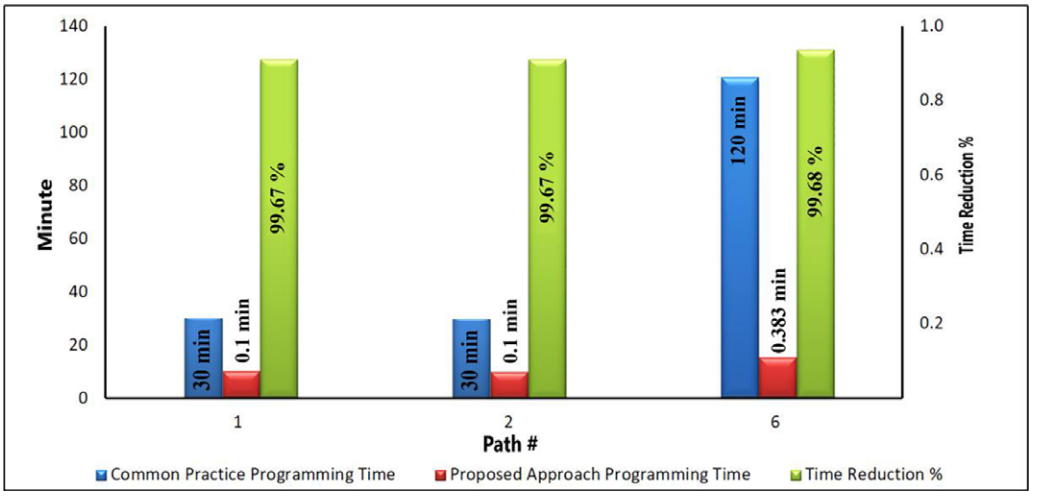


Figure 11. *Programming time reduction %.*

the comparison between the two approaches. Table V shows the source of error between the proposed approach and common practice. Figure 11 shows a comparison between the programming times of the two approaches for three paths. The programming time reduction (*TR*%) in the proposed approach is significant.

5. Conclusion

In conclusion, this paper introduces a new lead-through programming approach with the assistant of a vision system. The proposed approach is applied on a 4-DOF welding arm robot with an automatic feeding electrode. The proposed approach is tested experimentally. A comparison between the proposed and common approaches is carried out. The proposed approach has fewer steps and less complexity in programming and reprogramming. It has a shorter programming time with a high percentage of more than 99%. As the programming can be done by the skilled welder, there is no need for the rare and

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press

expensive expert programmer. Path following accuracy is expected to increase as the welder follows the actual path with the desired offset and with all manufacturing deviations. However, since the inputs are based on the human senses, the proposed approach is limited to this range of accuracy. Therefore, the proposed approach is suitable for programming and reprogramming of robots used in small and medium enterprises with acceptable accuracy. This will increase the robot share in global industries.

References

[1] Z. Wenming, D. Zhihai and L. Zhanqi, "Present Situation and Development Trend of Welding Robot," *2nd International Conference on Materials Science, Machinery and Energy Engineering (MSMEE)* (2017).
[2] M. H. A. L. Wei and L. S. Yong, "An Industrial Application of Control of Dynamic Behavior of Robots - A Walk Through Programmed Welding Robot," *Proceedings of the IEEE, International Conference on Robotics and Automation*, San Francisco (2000).
[3] H.-C. Lin, Y. Fan, T. Tang and M. Tomizuka, "Human Guidance Programming on a 6-DoF Robot with Collision Avoidance," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea (2016).
[4] M. Karlsson, A. Robertsson and R. Johansson, "Autonomous Interpretation of Demonstrations for Modification of Dynamical Movement Primitives," *IEEE International Conference on Robotics and Automation (ICRA)*, Singapore (2017).
[5] L. Bascetta, G. Ferretti, G. Magnani and P. Rocco, "Walk-through programming for robotic manipulators based on admittance control," *Robotica* **31**(7), 1143–1153 (2013).
[6] L. Qi, D. Zhang and J. Li, "A Lead-Through Robot Programming Approach Using A 6-DOF Wire-based Motion Tracking Device," *Proceedings of the 2009 IEEE, International Conference on Robotics and Biomimetics*, Guilin, China (2009).
[7] T.-W. Kim, K.-Y. Lee, J. Kim, M.-J. Oh and J. H. Lee, "Wireless Teaching Pendant for Mobile Welding Robot in Shipyard," *Proceedings of the 17th World Congress, The International Federation of Automatic Control*, Seoul, Korea (2008).
[8] I. W. Muzan, T. Faisal, H. Al-Assadi and M. Iwan, "Implementation of Industrial Robot for Painting Applications," *International Symposium on Robotics and Intelligent Sensors (IRIS)* (2012).
[9] H.-C. Lin, T. Tang, Y. Fan, Y. Zhao, M. Tomizuka and W. Chen, "Robot Learning from Human Demonstration with Remote Lead Through Teaching," *European Control Conference (ECC)* (2016) pp. 388–394.
[10] A. A. Mohammed and M. Sunar, "Kinematics Modeling of a 4-DOF Robotic Arm," *International Conference on Control, Automation and Robotics*, Singapore (2015).
[11] A. Somasundar and G. Yedukondalu, "Robotic Path Planning and Simulation by Jacobian Inverse for Indstrial Applications," *International Conference on Robotics and Smart Manufacturing (RoSMa)* (2018).
[12] L. A. Ferreira, Y. L. Figueira, I. F. Iglesias and M. Á. Souto, "Offline CAD-based Robot Programming and Welding Parametrization of a Flexible and Adaptive Robotic Cell Using Enriched CAD/CAM System for Shipbuilding," *27th International Conference on Flexible Automation and Intelligent Manufacturing*, Modena, Italy (2017).
[13] H.-I. Lin, Y.-C. Liu and Y.-H. Lin, "Intuitive kinematic control of a robot arm via human motion," *Procedia Eng.* **79**, 411–416 (2014). https://www.sciencedirect.com/science/article/pii/S1877705814009412.
[14] E. Lachat, H. Macher, M. A. Mittet, T. Landes and P. Grussenmeyer, "First Experiences with Kinect v2 Sensor for Close Range 3D Modeling," *6th International Workshop 3D-ARCH*, Avila, Spain (2015).
[15] B. Prashanth and A. R. S, "Design, Fabrication and Control of Four Degrees of Freedom Serial Manipulator," *International Conference on Smart Systems and Inventive Technology (ICSSIT)* (2018).
[16] P. Corke, "Using the Robotics Toolbox with a real robot," **In:** *Robotics, Vision & Control* (Springer, 2013).

Cite this article: M. H. M. Ali and M. R. Atia (2022). "A lead through approach for programming a welding arm robot using machine vision", Robotica **40**, 464–474. https://doi.org/10.1017/S026357472100059X

https://doi.org/10.1017/S026357472100059X Published online by Cambridge University Press