2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)
August 28 - September 1, 2024. Bari, Italy

2024 IEEE 20th International Conference on Automation Science and Engineering (CASE) | 979-8-3503-5851-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/CASE59546.2024.10711784

Clothoid-based CAD Model Compensation for Precise Welding in Manufacturing Processes

Edoardo Fiorini, Riccardo Muradore, and Francesco Visentin

***Abstract*— In an industry production line, a crucial task has always been welding, which requires millimeter precision to meet the high-quality standards of the products. In recent years, collaborative robots (cobots) have been integrated into robotic welding cells to guarantee production quality. A human operator can program cobots to follow any desired welding trajectory by moving the manipulator around the workpiece. However, this approach can be physically demanding and lead to poor accuracy. It is also time-consuming since it has to be repeated for each workpiece. This paper presents a novel motion planning approach for welding robots that combines welding workpiece CAD data with optical-tracked *learning-by-demonstration*. With the proposed approach, it is possible to automatically extract the motion primitives from the CAD model and map them into the workpiece by identifying a set of waypoints using a pointer and a motion capture system. As a result, the process becomes less time-consuming and more straightforward for the operator. We validated the approach in a laboratory setting, achieving an average error of 1 mm in the positioning and a time reduction of 75%.**

***Index Terms*— Welding Robots, Learning by Demonstration, Motion Capture System, Motion Planning.**

I. Introduction

Welding metal parts is commonly carried out manually, which results in a lack of precision and consistency. For companies, certifying and training individuals for this task means high costs and safety concerns for operators due to exposure to smoke, toxic gases, and intense light. Integrating robots in factories has improved safety and precision and provided a new approach to automatizing the process. These solutions require a complete offline procedure where all the information for generating the welding trajectory is taken a priori from the workpiece CAD model [1]. Such information allows the planner to generate robust trajectories without input from the human operator. However, relying solely on the CAD model can pose a problem regarding precision when the workpiece does not conform to the ideal model due to manufacturing defects.

To better adapt to the changes in the production line, companies have started using collaborative welding cells [2], [3], [4], in which the kinesthetic Learning by Demonstration (LbD) paradigm is exploited [5]. In such a configuration, the operator teaches the manipulator the desired welding trajectory, moving it throughout the desired waypoints. Once all the waypoints have been acquired, the optimal trajectory is generated considering the interpolation rules (i.e., circular path, linear interpolation) and the constraints (i.e., execution time or smoothness) defined by the operator.

Despite this procedure guaranteeing high adaptability to the workpiece, the method can be cumbersome for the operator, even if the robot is in gravity compensation mode. Thus, it is possible to introduce errors in the training phase that can lead to lower accuracy in the recorded waypoints and, consequently, poor quality in the welding process result. In addition, manually setting all the motion primitives is time-consuming and can introduce mistakes and slow down the production line.

This paper presents a novel method to improve the planning algorithms for welding robots, making the LbD paradigm easier and reducing the operator's uncertainty and training time. The approach combines offline and online strategies, considering the information available in the CAD model and the measurements taken by the operator. From the CAD, we extract the relevant waypoints and the motion primitives, and then we map those on the workpiece using the same set of waypoints captured using a lightweight, hand-tool device. Thus, the whole process becomes more streamlined and faster, reducing the timing of the teaching phase while maintaining high accuracy.

In the following Subsection, we will introduce the current state of the art in welding robots and the two main approaches (i.e., offline and online) used in the field. We then present in Section II the workflow used in this work to extract relevant information from CAD models and their mapping in the real world. We evaluate the approach using a collaborative robot (UR5e by Universal Robots) and a motion capture system (Optitrack); this setup is described in Sections III. The experimental results are presented and discussed in Sections IV. Section V draws final remarks and conclusions.

A. Related Work

As introduced earlier, the two main approaches to tackling robot-based welding are based on the extraction of features either from the CAD model or the workpiece and the use of collaborative robots programmed by the operator. An example of the first approach is the work of Larkin et al. [6], which is focused on the CAD model to generate the trajectories. The trajectory optimization involves a graph cost function and a greedy search algorithm, complemented by a Gaussian probabilistic roadmap for efficient motion planning between welding regions, considering the irregular shape of the workpiece. Another approach is presented in [7], where a digital twin of the environment is created offline, starting

The authors are with the Department of Engineering for Innovation Medicine, University of Verona, Verona, Italy. edoardo.fiorini@univr.it, riccardo.muradore@univr.it, francesco.visentin@univr.it E.F. is now with the The German Aerospace Center (DLR). edoardo.fiorini@dlr.de

TABLE I

Comparison of features. Within the table, execution time comparisons are as follows: "fast" for times < 30 s, "medium" for times < 60 s, and "long" for times $\geqslant 60$ s. Trajectory generation methods as follows: MP for Motion Primitives, SP for Spline, DRL for Deep Reinforcement Learning, and C for Clothoids. When data is unavailable, it is indicated as N/A.

Method	Execution time	Precision	LbD	CAD-based	AI-based	AR-based	Trajectory Generation
standard method	medium	1 mm	x				MP
[6]	medium	N/A		x			
[7]	low	N/A		x	x		
[8]	N/A	N/A			x		SP
[9]	N/A	0.32 mm			x		
[10]	N/A	N/A			x		
[11]	medium	N/A		x	x		
[12]	medium	N/A	x				SP
[13]	N/A	N/A			x		DRL
[14]	N/A	N/A				x	P
our method	fast	1 mm	x	x			C

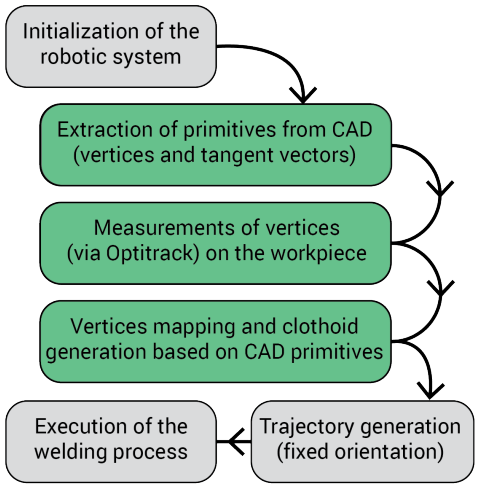

Fig. 1. A schematic representation of the workflow. The gray boxes represent the standard approach in a welding planner; green boxes are improvements introduced by the proposed method.

from the CAD models of the welding workpiece and the robot. If the generated trajectory bumps into an obstacle, an offset is introduced, starting from the online information obtained from a laser positioned in the cell.

A different approach is proposed by A. Gómez-Espinosa et al. [8], who leverage an RGBD camera, stereo vision algorithms, and structured light to extract the 3D structure of the workpiece and identify welding seams marked with distinct colors. A similar approach is proposed in [9] by Zou et al., where the workpiece's pose is determined by analyzing the angle between the incident line of a three-wire laser. The robot is controlled using a fuzzy algorithm that adjusts its parameters in real time, considering the seam points and the current robot state. Recent results published by Yang et al. [10] show using a digital light processing projector, combined with cloud feature-segmentation algorithms, to extract details with high precision, even in situations of weak light and low contrast. The accurate reconstruction enables the generation of precise position and orientation trajectories and the method's adaptability to various object materials, including scratches and rust. A complete approach is presented in [11] by Zheng et al., who developed an offline solution based on the combination of two sources of information: a CAD model to extract the ideal trajectory and a vision-based system to detect the workpiece's geometric features. The two are then combined to derive the final welding path.

A different approach not based on CAD is the one proposed by Antonelli et al. [12]. In their setup, the operator teaches the robot the trajectory using a probe detected by a motion capture system. The acquired data is then processed to fit the points into spheres approximated with B-splines. The system has been thoroughly tested under different configurations, proving its accuracy and reliability.

Another robot-less approach is presented in [13], where Deep Reinforcement Learning (DRL) techniques have been used to extract information from visual feedback. The robot's training is performed in a virtual environment where the agent can have enough exploration and observation variability to generate applicable policies to be transferred to a real-world manipulator. In addition, the case of partial-state observation is addressed by combining the DRL algorithms with Convolutional Neural Networks to predict motion for complex 3D trajectories. Differently from the previous work, the approach of Tavares et al. [14] leverages the Building Information Modeling standards to orchestrate the tasks between the robot and the operator automatically. The operator is equipped with augmented reality (AR) cues to position the welding workpiece without requiring time-consuming acquisition procedures. Laser sensors continuously update these AR-guided user inputs in real time, allowing the system to align the actual with the ideal cell configuration, generate collision-free trajectories for the robotic manipulator, and execute the welding task efficiently. The approach has proven to speed up the process and reduce errors in object placement, with the added benefit of allowing the user to easily select and visualize critical parameters such as tool orientation, trajectory velocity, and object position. Table I summarizes the features and disadvantages of the current methods used in robotic-based welding.

II. Materials and Methods

The proposed approach combines offline and online processing to overcome the operator's burden of moving heavy co-robots, mounting the welding tool, and reducing the execution time in the robot-assisted welding procedure while maintaining a high-quality result. Figure 1 shows an overview of the implemented workflow, while Figure 2 focuses only on the offline approach.

A. Extraction of Vertices from CAD

The initial step in the process consists of extracting the boundaries from the CAD file. This work focuses only on the two-dimensional case or three-dimensional plane extrusion (Figure 2A). In either case, the CAD can be reduced to points referenced to a fixed origin (Figure 2B). Ordering unarranged points is a crucial part of the process since it defines the path to identify the object in the scene and the actual workpiece's shape. The approach used is the one introduced in [15], [16].

To better understand the local behavior of the set of points, we convert them from Euclidean coordinates to the Frenet frame, allowing us to consider orientation and curvature properties along the object boundaries. In the case of a discrete set of points, it is possible to describe the Frenet frame as follows [17], [18]. For a given point $\mathbf{c}_i$ in the set of 3D points $\mathcal{C}$, we can define a sequence of the recursively discrete frame as $\mathcal{F}_i = (\boldsymbol{T}_i, \boldsymbol{N}_i, \boldsymbol{B}_i)$. The unit tangent vector for the i-th point is defined as

$$\boldsymbol{T}_i = \frac{\boldsymbol{c}_{i+1} - \boldsymbol{c}_i}{|\boldsymbol{c}_{i+1} - \boldsymbol{c}_i|}, \tag{1}$$

while the bi-normal and normal vectors are

$$\boldsymbol{N}_i = \frac{\boldsymbol{T}_i - \boldsymbol{T}_{i-1}}{|\boldsymbol{T}_i - \boldsymbol{T}_{i-1}|}, \tag{2}$$

$$\boldsymbol{B}_i = \boldsymbol{T}_i \times \boldsymbol{N}_i. \tag{3}$$

It should be noted that, for an open curve, there will be no discrete Frenet frame definition for the first and the last point due to the dependence of calculation requiring at least three neighboring points. From the above equations, the discrete curvature and torsion can be derived as

$$\kappa_i = \|\boldsymbol{T}_i - \boldsymbol{T}_{i-1}\| \tag{4}$$

$$\tau_i = \frac{\boldsymbol{B}_i - \boldsymbol{B}_{i-1}}{|\boldsymbol{N}_i|} \tag{5}$$

In a two-dimensional case, the curvature, κ, and its first derivative, κ', can be used to identify critical points such as sharp edges (i.e., abrupt changes in κ') or identify the initial point and the final point of a semi-circle (i.e., slow changes in κ'). Figure 2C shows an example of both cases.

B. Clothoids as Motion Primitives

A clothoid, also named as Cornu or Euler spiral, is a kind of curve with a linear relationship between curvature $\mathcal{K}$ and arc length s

$$\mathcal{K}(s) = k_1 s + k_2 \tag{6}$$

where k_1, k_2 are constant coefficients. According to the definition of Fresnel Integrals form, [19], a clothoid curve satisfies the following ordinary differential equations

$$\begin{aligned} x'(s) &= \cos(\alpha(s)), & x(0) &= x_0 \\ y'(s) &= \sin(\alpha(s)), & y(0) &= y_0 \\ \alpha'(s) &= \mathcal{K}(s), & \alpha(0) &= \alpha_0 \\ \alpha(s) &= \frac{k_1}{2}s^2 + k_2 s + \alpha_0 \end{aligned} \tag{7}$$

where $\alpha(s)$ is the direction of the tangent curve $(x'(s), y'(s))$ at point $(x(s), y(s))$ with respect to the arc parameter s. There are two special cases: when $k_1 = 0$, the curve is a circle, and when $k_1 = k_2 = 0$, the curve is a straight line. This property allows a smooth transition between straight segments and curves and, thus, tracks the changes in the shape of a workpiece, ensuring continuity in the generated curve.

A piece-wise clothoid curve can be defined by a set of points and the direction of the tangent vector of the curve that connects two vertices. To extract such information, we select a set of P points in the neighborhood of each vertex within a radial distance r (Figure 2D). To ensure continuity between the curves, we compute the tangent vector on both sides of the vertex (Figure 2E). Once the values are acquired, it is possible to directly map the information by identifying a new set of vertices on the workpiece (Figure 2F).

C. Learning-by-Demonstration via Optical Tracker

The mapping between the points extracted from the CAD model and the workpiece is obtained by a set of acquisitions performed by the operator on the real object. The operator uses a hand-tool device, and an optical tracking system extracts its pose. The registration between the two reference frames (i.e., robot and optical system) is performed as in [20].

The protocol requires that each real workpiece vertex is acquired in order and separately, according to the fixed Frenet reference frame. Therefore the i-th waypoint is represented by a set of measurements $\boldsymbol{H}^i = \{\boldsymbol{h}_1^i, \ldots, \boldsymbol{h}_t^i, \ldots, \boldsymbol{h}_T^i\}$ over time t, where $\boldsymbol{h}_t^i = \{h_x, h_y, h_z\} \in \mathbb{R}^3$ is a single acquisition in operational space at time t. To obtain the desired vertex data, the measurements are averaged over time

$$\boldsymbol{w}^i = \frac{\sum_{t=1}^{T} \boldsymbol{h}_t^i}{T}. \tag{8}$$

Therefore, the final set of waypoints can be defined as $\boldsymbol{W} = \{\boldsymbol{w}^1, \ldots, \boldsymbol{w}^M\}$ where M is the total number of welding workpiece waypoints, automatically extracted from CAD and refined by the operator during the LbD phase.

D. Optimal Desired Trajectory

The motion primitives and the waypoints are combined to obtain the final trajectory described as a parametric path of the end-effector pose

$$\boldsymbol{x}_e(u) = \begin{bmatrix} \boldsymbol{p}(u) \\ \boldsymbol{o}(u) \end{bmatrix}, \qquad u \in [u_{min}, u_{max}], \tag{9}$$

where u is the parameter with $\boldsymbol{p}(u_{min}) = \boldsymbol{w}^1$ and $\boldsymbol{p}(u_{max}) = \boldsymbol{w}^M$, $\boldsymbol{p} = \begin{bmatrix} x & y & z \end{bmatrix}^\top$ is the position, and

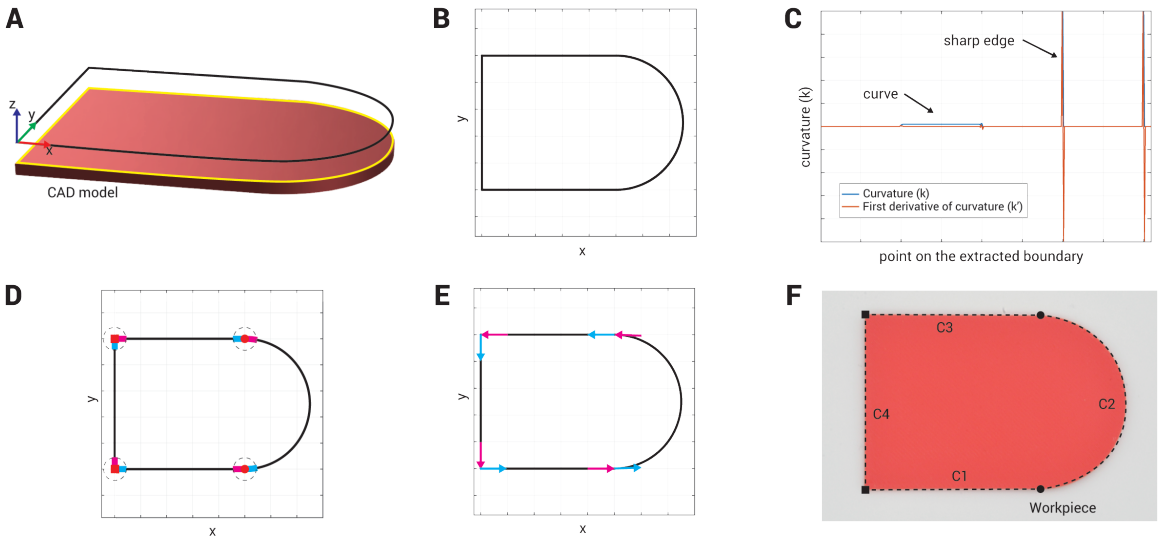

Fig. 2. Framework to extract vertices and motion primitive from CAD data. (A) CAD design can be either two– or three-dimensional; in either case, thresholding of the vertices allows extracting points representing the shape's boundary. (B) Ordered set of points. (C) Analysis of the curvature and its first derivative obtained by the conversion in the Frenet frame. (D) Vertices extraction and identification of neighboring points. (E) Tangent vector calculation. (F) Piece-wise clothoid representation of the CAD model applied to the workpiece. Each clothoid is defined by the curve C_i $(i = 1, \ldots, 4)$.

$\boldsymbol{o} = \begin{bmatrix} o_1 & o_2 & o_3 & o_4 \end{bmatrix}^\top$ the orientation described using quaternion; o_1 is the real part while (o_2, o_3, o_4) are the imaginary parts. By applying the timing law, $u = u(t)$, it is possible to define the trajectory of the robot, $\tilde{\boldsymbol{x}}_e(t)$, with respect to time, t, as the composition of $\boldsymbol{x}_e(u)$ and $u(t)$

$$\tilde{\boldsymbol{x}}_e(t) = (\boldsymbol{x}_e \circ u)(t). \tag{10}$$

While the position $\boldsymbol{p}$ is given by the clothoid, the orientation $\boldsymbol{q}$ is chosen by the operator before starting the teaching phase and it varies based on the desired inclination of the welding tool tip with respect to the workpiece plane. The interpolation between the quaternions describing such orientation for each segment is obtained by the SLERP (Spherical Linear Interpolation) algorithm [21].

E. Robot Control

An inverse dynamics control schema maps the generated reference trajectory into the necessary torques for each robot joint. The complete control law for the input command $\boldsymbol{\tau}$ is given by

$$\boldsymbol{\tau} = B(\boldsymbol{q})\boldsymbol{y} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + g(\boldsymbol{q}), \tag{11}$$

$$\boldsymbol{y} = J_A'(\boldsymbol{q})(\ddot{\boldsymbol{x}}_{\boldsymbol{d}} + K_D\dot{\tilde{\boldsymbol{x}}} + K_P\tilde{\boldsymbol{x}} - \dot{J}_A(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}), \tag{12}$$

where $\boldsymbol{q}$ is the generalized coordinate vector, B, C, and g are the inertia, Coriolis and centrifugal, and gravity matrices, respectively; J_A is the analytical Jacobian, K_P the proportional gain, K_D the derivative gain, $\ddot{\boldsymbol{x}}_{\boldsymbol{d}}$ is the desired trajectory acceleration in operational space, and $\tilde{\boldsymbol{x}}$ and $\dot{\tilde{\boldsymbol{x}}}$ are the position and velocity error computed as the difference between the desired and actual Cartesian pose of the robot.

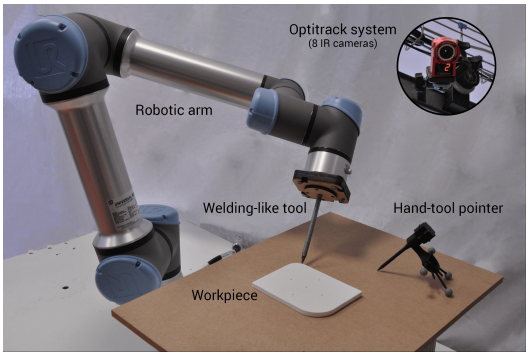

Fig. 3. The experimental setup consists of a 6-DoF, collaborative, robotic arm and a motion capture system based on eight IR cameras. The waypoints on the workpiece are acquired by a lightweight, hand-held device that integrates markers to be correctly identified by the optical system.

III. Experimental Setup

The proposed approach has been tested in a laboratory setting using a 6-DoF (Degree of Freedom), collaborative, robotic arm (UR5e, Universal Robots), mounting a metal pointer at its end-effector, which simulates the torch used for welding. The waypoints on the workpiece were captured using 3D-printed hand-tool mounting IR-reflecting markers combined with a commercially available motion capture system (Flex3 by OptiTrack), which offers sub-millimeter tracking accuracy. The optical system consists of eight infrared cameras positioned onto a stable mounting structure above the working area to provide a full view of the scene and to minimize the number of occlusions. Figure 3 shows the overall experimental setup.

The system is based on the Robot Operating System (ROS 2) [22] distributed over three different computers. The first

device is dedicated to acquiring motion data from the vision system; the second device runs the robot controller, and the third device imports the CAD data and performs the offline analysis to extract the waypoints and the parameters for the clothoid generation, processes the motion data and extracts the waypoints on the workpiece to generate the welding trajectory. The three devices communicate via ROS ensuring realtime performance if required by the application.

IV. Results and Discussion

We evaluated the approach on a 3D-printed workpiece with linear and circular primitives (Figure 3), which provides a realistic complexity for the validation of our method. Specifically, the workpiece has four straight lines with different lengths, two square corners (i.e., 90°), and two circular arcs (90° angle) with different radii. Compared to the CAD file, the printed object has a larger surface with a discrepancy of around 2% (Figure 4A). This error in manufacturing does not allow a direct application of classical offline-only approaches; thus, the user must perform an online compensation of the model directly on the workpiece.

To assess the correctness of the approach, we compared the results with two other methods. The first consists of direct learning-by-demonstration, where the robot's end-effector is moved along a set of waypoints. The motion primitives and the parameters are manually set using the robot teach pendant. The internal solver computes the final trajectory, which is then executed by the robot. The second approach exploits the motion tracking system and asks the user to acquire multiple points along each segment of the workpiece's perimeter. The complete trajectory is then computed as a combination of lines and arcs generated as results of the RANSAC model [23]. For each approach, we asked five subjects without prior knowledge of robotics to teach the robot, saving the generated trajectory for comparison.

To assess the precision in the trajectory generation, we computed the discrete Fréchet distance [24] between a reference curve (i.e., the perimeter of the workpiece) and any other. Using this distance, we are not constrained by the trajectory sampling structure but only by the location and order of the points along the curves. Figure 4B and C depict the results on highlighted workpiece sections. As shown in the pictures, the less-performing approach is the one that uses the information at the workpiece boundaries (with an error of 5 mm with respect to the object perimeter). The reason can be traced back to the numerical error introduced by the optical tracker during the motion of the pointer, occlusions during the data acquisition, or reduced precision due to movement. Those errors introduce outliers that can negatively influence the RANSAC algorithm, reducing accuracy. The other two methods have comparable results with an average error of around 1 mm, thanks to the fact that the operator directly points to either the end-effector or the hand-tool pointer on the edge of the workpiece.

In addition to measuring precision in generating and following the trajectory, we included additional criteria to assess the three methods. Values such as the overall process execution time and the number of points required are critical to evaluate improvements in the industrial production setting. Concerning the execution time, the proposed approach significantly requires less time (around 10s on average, i.e., 75% of the time) than the others. The main reason is the reduced amount of additional information needed for the method: the operator has only to identify the ordered vertices. The speed in the execution is also given by the lower number of points required by the method. In fact, by processing the CAD data offline and using the clothoid as a motion primitive, only two vertices are needed for each segment, either in the linear or circular path. In contrast, by manually selecting the motion primitives with the robot, a circular path requires at least three points, introducing further errors in the process. Table II summarizes the results for each approach.

TABLE II

Comparative results between three possible approaches: 1) using the robot to acquire the waypoints and the motion primitives, 2) using boundary extracted by the optical tracker as waypoints, and 3) using the proposed method and the data acquire from the optical tracker.

Method	Time (s)	Waypoints	Error (mm)
1) Robot, manual selection	40	8	1.2
2A) OptiTrack, perimeter	35	300	5.3
2B) OptiTrack, perimeter	35	100	8.1
2C) OptiTrack, perimeter	35	50	17.9
2D) OptiTrack, perimeter	35	20	25.6
3) OptiTrack, Clothoids	10	6	1.5

V. Conclusions

Robotic welding technologies have revolutionized manufacturing industries by offering efficient, precise, and consistent solutions. Including robots within the workspace requires defining new and intuitive workflows that non-expert users can adopt.

In this paper, we present a novel approach to collaborative welding robots, where the operator can quickly and safely teach the robot the desired trajectory without the burden of moving it along the workpiece or identifying the strategies to move between consecutive vertices. The approach extracts crucial information (such as vertices and motion primitives) from the CAD model and asks the operator to identify a set of ordered vertices onto the workpiece by a lightweight, hand-tool pointer whose position is tracked in real-time by a motion capture system. Then, the motion primitives are mapped on the new set of points using clothoids. Using such curves as motion primitives makes it robust to uncertainties in the CAD model and reduces the time needed to complete the task. We evaluated the approach in a laboratory scenario, achieving an average error of 1 mm in the positioning and a time reduction up to 75% compared to standard approaches.

As future work, the method can be extended to more complex two-dimensional geometries with more complex perimeters, such as sigmoid-like shapes, and on three-dimensional surfaces. Being able to handle such situations paves the way for more intuitive workflows and a considerable increase in production by reducing the time needed to perform the task.

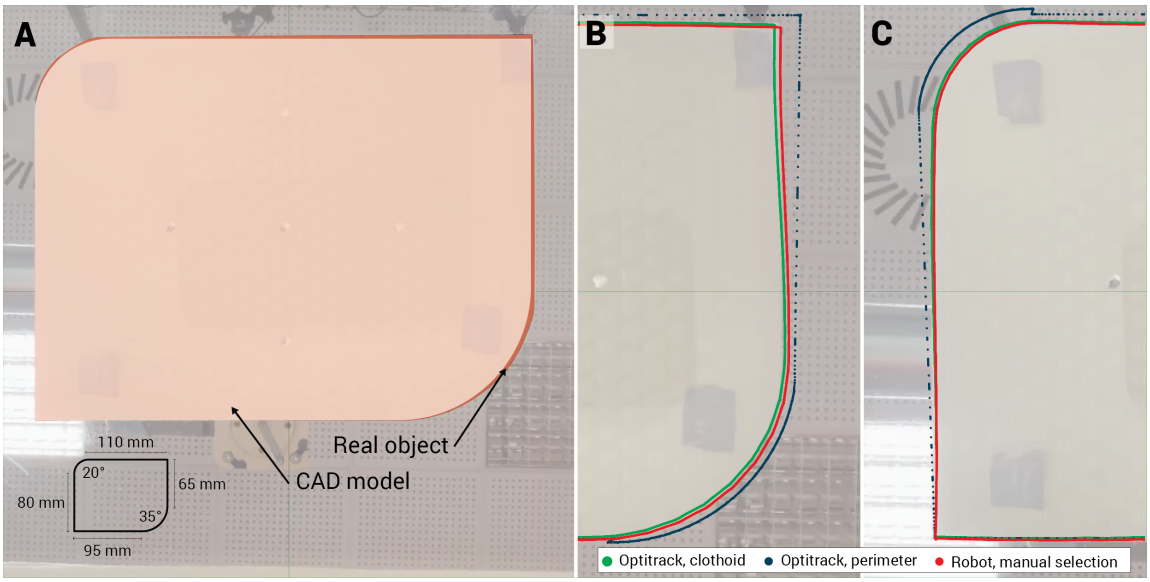

Fig. 4. Results on a selected workpiece. (A) The discrepancy between the 3D-printed workpiece and the CAD model is due to the manufacturing process. (B) The results of the proposed method (green line) were compared with two additional approaches: (red line) robot-based learning by demonstration and (blue line) workpiece parameter fitting.

REFERENCES

[1] Larkin, N., Short, A., Pan, Z., and van Duin, S., "Automatic program generation for welding robots from cad," in *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 2016, pp. 560–565.

[2] KUKA. Kuka arc-welding. URL: https://www.kuka.com/en-de/industries/metal-industry/arc-welding/kuka-arc-welding-robots (see on 02-2024).

[3] UR. Ur arc-welding. URL: https://www.universal-robots.com/plus/products/hirebotics/cobot-welder-powered-by-beacon/ (see on 02-2024).

[4] ABB. Abb arc-welding. URL: https://new.abb.com/products/robotics/application-equipment-and-accessories/arc-welding/cobot-arc-welding-package (see on 02-2024).

[5] Ravichandar, H., Polydoros, A. S., Chernova, S., and Billard, A., "Recent advances in robot learning from demonstration," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, pp. 297–330, 2020.

[6] Larkin, N., Short, A., Pan, Z., and van Duin, S., "Automatic program generation for welding robots from cad," in *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 2016, pp. 560–565.

[7] Zhou, X., Wang, X., Xie, Z., Li, F., and Gu, X., "Online obstacle avoidance path planning and application for arc welding robot," *Robotics and Computer-Integrated Manufacturing*, vol. 78, p. 102413, 2022.

[8] Gómez-Espinosa, A., Rodríguez-Suárez, J. B., Cuan-Urquizo, E., Cabello, J. A. E., and Swenson, R. L., "Colored 3d path extraction based on depth-rgb sensor for welding robot trajectory generation," *Automation*, vol. 2, no. 4, pp. 252–265, 2021.

[9] Zou, Y., Wang, Y., Zhou, W., and Chen, X., "Real-time seam tracking control system based on line laser visions," *Optics Laser Technology*, vol. 103, pp. 182–192, 2018.

[10] Yang, L., Liu, Y., Peng, J., and Liang, Z., "A novel system for off-line 3d seam extraction and path planning based on point cloud segmentation for arc welding robot," *Robotics and Computer-Integrated Manufacturing*, vol. 64, p. 101929, 2020.

[11] Zheng, C., An, Y., Wang, Z., Wu, H., Qin, X., Eynard, B., and Zhang, Y., "Hybrid offline programming method for robotic welding systems," *Robotics and Computer-Integrated Manufacturing*, vol. 73, p. 102238, 2022.

[12] Antonelli, D., Astanin, S., Galetto, M., and Mastrogiacomo, L., "Training by demonstration for welding robots by optical trajectory tracking," *Procedia CIRP*, vol. 12, pp. 145–150, 2013, eighth CIRP Conference on Intelligent Computation in Manufacturing Engineering.

[13] Maldonado-Ramirez, A., Rios-Cabrera, R., and Lopez-Juarez, I., "A visual path-following learning approach for industrial robots using drl," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102130, 2021.

[14] Tavares, P., Costa, C. M., Rocha, L., Malaca, P., Costa, P., Moreira, A. P., Sousa, A., and Veiga, G., "Collaborative welding system using bim for robotic reprogramming and spatial augmented reality," *Automation in Construction*, vol. 106, p. 102825, 2019.

[15] Fan, J., Visentin, F., Del Dottore, E., and Mazzolai, B., "An image-based method for the morphological analysis of tendrils with 2d piecewise clothoid approximation model," in *Biomimetic and Biohybrid Systems*, Vouloutsi, V., Mura, A., Tauber, F., Speck, T., Prescott, T. J., and Verschure, P. F. M. J., Eds. Cham: Springer International Publishing, 2020, pp. 80–91.

[16] Fan, J., Dottore, E. D., Visentin, F., and Mazzolai, B., "Image-based approach to reconstruct curling in continuum structures," in *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*, 2020, pp. 544–549.

[17] Hu, S., Lundgren, M., and Niemi, A. J., "Discrete frenet frame, inflection point solitons, and curve visualization with applications to folded proteins," *Phys. Rev. E*, vol. 83, p. 061908, Jun 2011.

[18] Lim, S. and Han, S., "Helical extension method for solving the natural equation of a space curve," vol. 5, no. 3, p. 035002, sep 2017.

[19] Bertolazzi, E. and Frego, M., "G1 fitting with clothoids," *Mathematical Methods in the Applied Sciences*, 3. 2014.

[20] Roberti, A., Piccinelli, N., Meli, D., Muradore, R., and Fiorini, P., "Improving rigid 3-d calibration for robotic surgery," *IEEE Transactions on Medical Robotics and Bionics*, vol. 2, no. 4, pp. 569–573, 2020.

[21] Shoemake, K., "Animating rotation with quaternion curves," *SIGGRAPH Comput. Graph.*, vol. 19, no. 3, pp. pp. 245–254, Jul 1985.

[22] Macenski, S., Foote, T., Gerkey, B., Lalancette, C., and Woodall, W., "Robot operating system 2: Design, architecture, and uses in the wild," *Science Robotics*, vol. 7, no. 66, p. eabm6074, 2022.

[23] Fischler, M. A. and Bolles, R. C., "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," in *Readings in Computer Vision*, Fischler, M. A. and Firschein, O., Eds. San Francisco (CA): Morgan Kaufmann, 1987, pp. 726–740.

[24] Aronov, B., Har-Peled, S., Knauer, C., Wang, Y., and Wenk, C., "Fréchet distance for curves, revisited," in *Algorithms – ESA 2006*, Azar, Y. and Erlebach, T., Eds. Berlin, Heidelberg: Springer Berlin Headelberg, 2006, pp. 52–63.