



Research paper

Acquiring the Coordinates for the Welding Seam through the Utilization of Point Cloud and Welding Map

Shiva Zeymaran¹, Vali Derhami^{1*}, and Mehran Mehrandehz²

1. Computer Engineering Department, Yazd University, Yazd, Iran.

2. Faculty of Engineering & Applied Science, University of Regina, Saskatchewan, Canada.

Article Info

Article History:

Received 13 October 2024

Revised 20 December 2024

Accepted 12 January 2024

DOI:10.22044/jadm.2025.15180.2624

Keywords:

Point Cloud, Weld Seam Trajectory, Welding Map, Welding Robot

*Corresponding author:
vderhami@yazd.ac.ir (V. Derhami)

Abstract

This paper presents an accurate and efficient method for determining the coordinates of welding seams, addressing a significant challenge in the deployment of welding robots for complex tasks. Despite welding robots' precision in following predetermined paths, they struggle with seam identification due to noisy industrial environments, stringent accuracy requirements, and computational complexity. Unlike existing approaches, which either rely on random sampling or are limited to simple geometries, our method combines splicing techniques with welding map alignment to handle complex shapes with multiple seams. This research employs a weighed method to integrate point clouds captured by RGB-D cameras, producing a low-noise point cloud. By leveraging the welding map of parts drawn, the method identifies probable regions for weld seams within the point cloud, substantially reducing the search space. This enables the system to find the weld seam in a timely manner. Knowing the approximate shape of the weld based on the available weld map, an innovative technique is then used to accurately locate the weld seam within these regions. Experimental results on fence-shaped structures in a simulated environment show a mean average error of 1.30 mm, achieving a 30% improvement in precision and a 77% reduction in computation time compared to the state-of-the-art methods. The approach's ability to accurately identify weld seams in complex shapes, coupled with its computational efficiency, suggests strong potential for real-world application. By leveraging welding maps and robust point cloud processing techniques, the method effectively addresses noise and variability, key challenges in industrial environments.

1. Introduction

Welding is a manufacturing technique employed across various industries to connect metal components and parts. However, the industry faces significant challenges when relying on human welders. High injury rates due to exposure to extreme heat, fumes, and heavy machinery remain a persistent concern, with welding-related injuries accounting for a considerable proportion of workplace accidents in manufacturing

environments. Turnover rates are also high, as the physically demanding nature of welding contributes to job dissatisfaction and workforce shortages. Furthermore, quality control issues arise from human error, leading to inconsistencies in weld quality and increased production costs. These challenges underscore the growing need for robotic welding technology, which offers improved safety, precision, and efficiency, while mitigating the

reliance on human labor in harsh working conditions. Robot welding technology is extensively used in various domains, including ship and bridge construction, automobile manufacturing, aircraft component production, railway carriage fabrication, and numerous other fields. Welding robots are the preferred choice for most welding tasks due to their versatility, efficiency, and precision in operation [1-4].

Welding robots have been utilized in various research projects, generally falling into two categories: those that do not use a camera and those that do. Wang and their collaborators [5] introduced a digital twin system for welding path planning in ship sub-assembly welding. Lei and their collaborators [6] utilized the arc voltage tracking method, incorporating a self-developed arc voltage module for orbital robotic welding. Rokossa [7] scanned components using a laser scanner, modeled the geometric contours in a simulator, and generated the synchronous movements of two UR5e robots. These studies are examples of the first category and assume that the weld seam trajectory is predetermined.

The second category of research studies employs depth cameras in welding tasks to find weld seam trajectories. Takubo and their collaborators [8] used 3D point clouds obtained from a depth camera to weld two flat plates at right angles to each other. They used the RANSAC algorithm to segment planes, extracting the weld line as the intersection of the two planes. Similarly, Yang and their collaborators [9] segmented planes and identified feature points on one plane, using a spline function for path fitting. In these studies, the RANSAC algorithm works randomly and struggles to find planes in complex shapes. These two latter methods could only determine the intersection of two angled intersecting planes, lacking solutions for more complex geometries involving more than two planes or parallel planes.

Kusumoto and their collaborators [10] captured each plane vertically to minimize the noise in point clouds. However, this approach is impractical for complex shapes with more than two planes since it is time-consuming to locate and capture each plane individually. Wang and their collaborators [11] proposed a multi-layer positioning strategy based on point clouds, which includes both a coarse positioning process and a fine positioning process. In the coarse positioning stage, the maximum likelihood method is employed, followed by fine positioning to improve accuracy.

Gao and their collaborators [12] focused on D-type welds by identifying edge points and fitting a weld curve to these points, though their method only

works for superficial weld seams. Yang and their collaborators [13] extracted similar weld seams that are the intersection of two pipes. They try to extract feature points from a preprocessed point cloud. Then key points are generated from the extracted feature points using a new algorithm called the bubble method. Finally, they find a course weld seam using key points and refine this weld seam.

While several methods have been developed for weld seam trajectory identification, they are often constrained by their reliance on idealized geometries or simplified welding environments. For instance, methods using RANSAC-based segmentation [8,9] are effective for planar intersections but struggle with complex or irregular weld seam geometries due to random sampling inefficiencies. Similarly, approaches requiring multiple scans or vertical captures [10] are impractical in industrial settings, where time and resource constraints are critical. These limitations hinder their applicability to real-world welding tasks involving intricate shapes and multiple seams. This paper addresses these challenges by leveraging welding maps to reduce search spaces and employing innovative splicing techniques for more accurate seam detection.

The designed approach is specifically tailored to identify weld seam trajectories in complex welding tasks, such as those involving parts with multiple seams oriented in various directions. While the method demonstrates significant improvements in precision and efficiency under these scenarios, its effectiveness may vary in environments with entirely different characteristics or constraints and may not be universally applicable without adjustments.

In summary, this paper makes the following contributions:

1. Combination and prioritization of information from cameras placed at different angles to reduce errors.
2. Matching the welding map with the point cloud to identify the initial seam search area.
3. Introduction of a new method for determining the position of the welding seam trajectory in a specified search area. The combination of steps 1 to 3 leads to an effective and efficient approach.
4. Development of the approach in a simulation environment.

The remainder of this paper is organized as follows: Section 2 describes the primary concepts. Section 3 provides a comprehensive explanation of our method for weld seam extraction. Section 4 presents the experimental results to demonstrate

the reliability of the proposed method. Finally, section 5 summarizes the study and discusses future work.

2. Primary Concepts

This section outlines the definitions of key concepts used in this research. Additionally, it presents descriptions of the algorithms and methods employed throughout the paper.

The main input of the system which come from RGB-D camera are point cloud. Point cloud is a set of data points in a three-dimensional space, each representing a specific location, often obtained using RGB-D camera, 3D scanners, or photogrammetry. These points collectively form a "cloud" that approximates the shape and structure of an object or environment. Each point is defined by coordinates (X, Y, Z) relative to an origin. Point clouds are widely used in applications such as 3D modeling, computer graphics, geographic information systems (GIS), and robotics. They are crucial for creating accurate digital models of physical objects and environments. Visualization of point clouds allows users to view and interact with the data in a 3D space, aiding in understanding the scanned object's structure and details [14].

Here, we use some depth camera to capture depth information as well as along with standard color images. Unlike traditional cameras that capture only two-dimensional color information, depth cameras provide three-dimensional data by measuring the distance of objects from the camera. This additional dimension allows for the creation of 3D models and spatial understanding of the scene. The output of a depth camera is a depth image, which contains information about the distance of each pixel in the image from the camera, enabling applications such as 3D modeling, robotics, and spatial analysis. Depth images are often showed as grayscale, where lighter shades represent closer objects and darker shades represent farther objects.

The Iterative Closest Point (ICP) algorithm [15] is the most widely used point cloud registration algorithm at present which aligns two point clouds to minimize error. If one point cloud is designated as the source and the other as the target, the algorithm computes an optimal transformation matrix that minimizes the error between the transformed source and target point clouds.

The point-to-point method iteratively finds the nearest point in the target cloud for each point in the source cloud. It then calculates a transformation matrix to map the source points to the corresponding target points [16]. The source point cloud is transformed using this matrix, and the

error is calculated using (1). This process is repeated iteratively: the source point cloud is transformed, and the error is recalculated until the alignment is optimized.

$$E_d(R, T) = \frac{1}{n} \sum_{i=1}^n (R \cdot B_i + T - A_i)^2 \quad (1)$$

In the above equation, R and T represent the rotation and translation matrices, n is the number of points in point cloud, and A_i and B_i are the points of the target and source point clouds, respectively. To reduce computational complexity, the Voxel Grid Filter [17] is employed to reduce the number of points in a point cloud. It subdivides the original point cloud into grids with N voxels [18]. Then, the center of gravity of each voxel is calculated by averaging all the points within that voxel and is used to replace all voxel's points [19]. If the barycenter point does not exist, the data point closest to the barycenter in the voxel is used to replace all the points. As a result, the number of points in the point cloud is reduced by a factor of N .

3. Proposed Approach

In this section, the proposed approach is explained. The following framework is considered. The welding parts, which features multiple weld seams, is positioned on a work table. To capture comprehensive depth information, several fixed RGB-D cameras are strategically placed around the welding parts, with three cameras utilized in this study. The depth images obtained from these cameras, along with their corresponding point clouds, are used for calculating the coordinates of visible weld seams. For weld seams that are not visible and are on the underside of the welding parts, calculations can be performed by flipping the welding parts on the table.

As a new idea, our approach incorporates a 3D welding map, as designing such a map is a crucial primary step in all industrial welding projects. Since this study was initiated to address the needs of the fence production industry, our proposed method has been developed and tested specifically on fence shapes. However, the method can be used for other forms of welding.

Figure 1 shows the flowchart of the proposed weld seam extraction method, which outlines the general steps of our approach.

3.1. Capture Depth Images

In this research, we used the PyBullet library in Python [20] to simulate the desired environment. The environment includes a ground plane, a desk, a fence, and a Kuka arm robot. The fence has two

horizontal and two vertical parts, featuring 16 weld seams. The configuration of the designed environment is shown in Figure 2. Additionally, there is a 1 mm gap between each welding part to allow the cameras to capture each part more effectively.

We use three cameras around the welding parts to obtain depth images from different perspectives, providing a comprehensive view of the fence. Adding more cameras could increase costs for an industry. Two cameras are positioned on the left and right sides of the fence, each at a 45-degree angle from the table and at equal distances from the fence.

The fixed cameras 1 and 2 are located at (1, 0, 2) and (-0.7, 0, 2) respectively, in the world coordinate system. The third camera is an eye-on-hand camera mounted on the robot arm, located at (0.15, -1, 2.5) in the world coordinate system. The positions of the cameras are shown in Figure 3. Depth images contain information about the relative distances of objects in the picture from the camera and are used to determine the coordinates of each object. The captured images are shown in Figure 4.

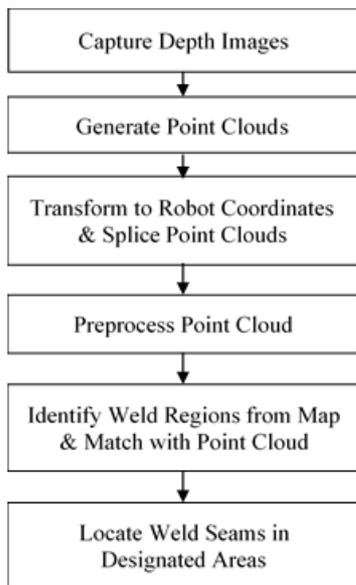


Figure 1. Flowchart for the proposed method of weld seam extraction approach.

3.2. Generate Point Clouds

Using depth images along with the camera's view matrix and projection matrix, we can calculate a point in a point cloud corresponding to each pixel in the depth image. The final point cloud is in the world coordinate system and includes every object in the environment. Therefore, we remove all undesired points from the final point cloud (background points) and retain only the points corresponding to the welding parts. To achieve

this, we set the filter range of the passthrough filter based on the coordinates of the target workpiece, ensuring that only the points within the required coordinates are saved, as shown in (2).

$$\left\{ \begin{array}{l} X_l \leq x_i \leq X_h \\ Y_l \leq y_i \leq Y_h \\ Z_l \leq z_i \leq Z_h \end{array} \right\}, \quad (2)$$

where (x_i, y_i, z_i) are the coordinates of each point in the point cloud X_l, X_h, Y_l, Y_h, Z_l and Z_h are the lower and upper bounds along the x, y, and z-axis respectively, which filter out the undesired points. This equation selects the points with coordinates in the desired range among all the points in the point cloud.

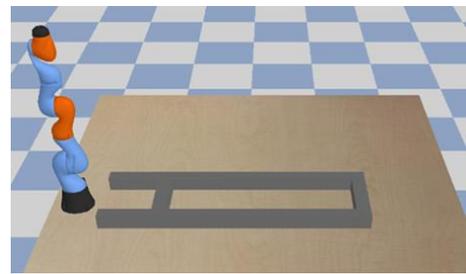


Figure 2. Simulated welding environment.

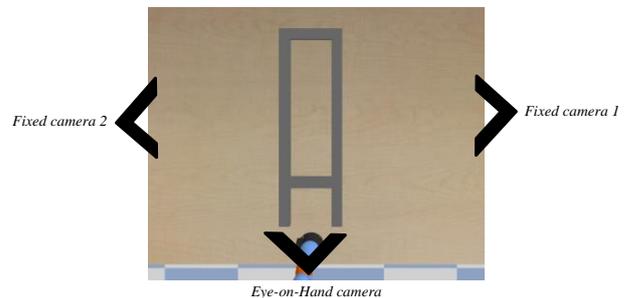


Figure 3. The placement of three depth cameras in the simulation environment relative to the welding part.

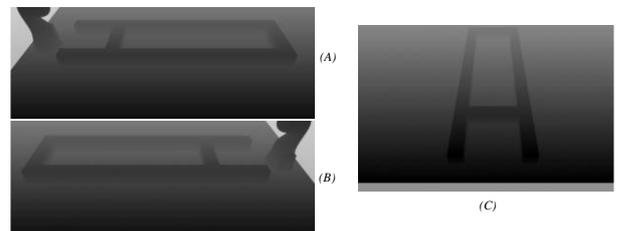


Figure 4. Generated depth images (A. camera 1, B. camera 2, C. eye-on-hand camera).

3.3. Transform to Robot Coordinates & Splice Point Clouds

The final calculated coordinates of the weld seams should be in robot coordinates so that they can be directly used by the robot arm. Therefore, we need to convert the point clouds from the world coordinate system to the robot coordinate system.

For this purpose, a transformation matrix is defined, with its rotation set to 0 degrees and its translation part calculated based on the location of the arm in the world coordinate system, which is (0.15, -1.1, 1.9). Equation (3) shows the calculated transformation matrix used for this conversion. Figure 5 shows the point clouds calculated from depth images in the robot coordinate system.

$$TransformationMatrix = \begin{bmatrix} 1 & 0 & 0 & -X_r \\ 0 & 1 & 0 & -Y_r \\ 0 & 0 & 1 & -Z_r \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

where (X_r, Y_r, Z_r) is the position of the robot arm in the world coordinate system.

These point clouds show different views of the fence. Some points may overlap in each point cloud, but generally, they complement each other. It is necessary to merge them to create a single, complete point cloud. To merge two point clouds, we use the ICP algorithm in point-to-point mode with a maximum correspondence distance of 0.001 m and 50 iterations. Setting the maximum correspondence distance to lower values will cause more points to be removed during down-sampling. If the value is set lower than 0.001, important points around the welding seams will be lost. The three point clouds are merged into one using the following process:

1. Use the ICP algorithm on the point clouds from camera 1 and camera 2 to find the optimal transformation matrix for the camera 1 point cloud.
2. Merge the transformed point cloud from camera 1 with the point cloud from camera 2 using a specific algorithm with prioritization.
3. Use the ICP algorithm on the result of step 2 and the point cloud from the eye-on-hand camera.
4. Merge the transformed point cloud from step 3 with the point cloud from the eye-on-hand camera.

The point clouds in step 2 are merged using an approach designed to minimize noise in the final point cloud. Since camera 1 is on the right side of the welding parts, it is too far from the shape's left side, resulting in 4-5 mm of noise on the left side of the point cloud from camera 1. Similarly, camera 2 experiences this noise on the right side of its point cloud.

To avoid transferring these noise values to the spliced point cloud, we merge the point clouds selectively, as shown in Figure 6. This way, the left and right sides of the fence are formed only by the camera that has the best view of that side.

Additionally, any errors in the measurements can lead to a general bias, but the ICP algorithm can handle this bias effectively by matching the point clouds. Figure 7 shows the final spliced point cloud that will be used in further steps.

3.4. Preprocess Point Cloud

After splicing the point clouds, some noise will appear on the z-axis due to the splicing process. In this step, we first remove this noise by noting that the Z direction points are distributed within a limited range. Thus, passthrough filtering is adopted to quickly eliminate outliers in the Z direction, as shown in (4). This equation selects the points with Z values within the desired range from all the points in the spliced point cloud. There is no such noise in the X direction because we use the prioritized approach for splicing. There might be some error in the Y direction after splicing, but we don't have enough information about its value to remove the noise using limited ranges. However, the approach introduced in the following steps could help overcome this error.

$$Z_{min} \leq z_i \leq Z_{max} \quad (4)$$

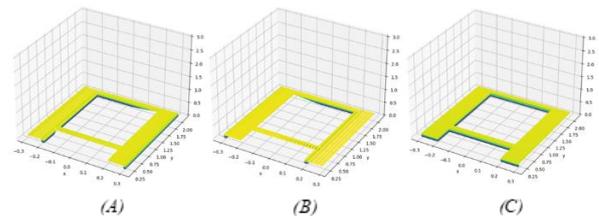


Figure 5. Point clouds in robot coordinates (A. camera 1, B. camera 2, C. eye-on-hand camera).

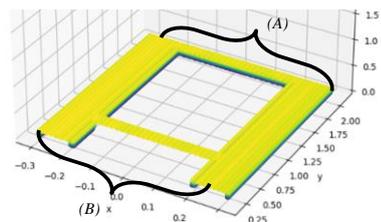


Figure 5. Selected parts for merging (A. selected part of camera 1 point cloud, B. selected part of camera 2 point cloud).

Secondly, since there are many points in the final point cloud that negatively affect the speed of point cloud computing in the later stages, we perform down-sampling to reduce the number of points while preserving important information around the weld seams. Subsampling of the point cloud using the Voxel Grid filter is applied for this purpose. A Voxel size of 0.01 m has been chosen as appropriate for this fence shape. The down-sampled point cloud is shown in Figure 8.

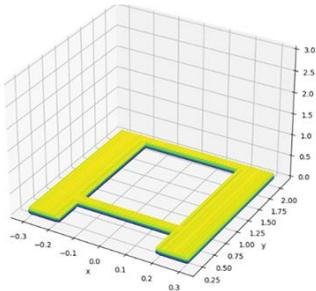


Figure 6. Final spliced point cloud.

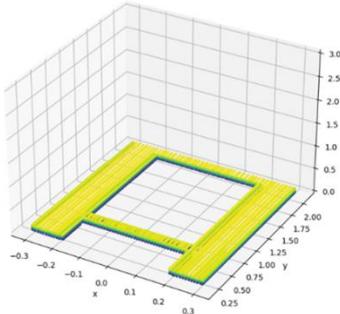


Figure 7. Downsampled point cloud.

3.5. Identify Weld Regions from Map & Match with Point Cloud

In this step, we aim to use a 3D welding map of the welding parts. Designing a 3D welding map is essential for almost all industrial welding projects. Therefore, a welding map should always be available before starting a welding project. This map is a DXF file designed in AutoCAD. Figure 9 shows the map in AutoCAD software and its corresponding point cloud. Note that the weld seams are shown in red on the welding map. The ezdxf library [21] in Python has been used to read the DXF file, extracting the start and end point coordinates of each weld seam in world coordinates. Using these coordinates from the welding map, we can estimate the weld line in the environment. Therefore, by considering an area around the start and end coordinates of each weld line on the map, we can confine our search area to this region in the real environment.

Hegedus-Kuti and their collaborators [22] used a CAD model for welding defect recognition with 3D scanners, employing the ICP algorithm to match the CAD model with the welding part. Similarly, Bjorndal [23] used the ICP algorithm to match CAD models with point clouds. In our study, the ICP algorithm is also used for matching the point cloud with the welding map.

If there is no difference between the welding parts and the welding map, aligning the map with the point cloud using the ICP algorithm is sufficient to determine the coordinates of the weld seam. In this case, the coordinates of the weld seams in the map

will be identical to their locations in the simulation environment.

Most of the time, in the real world, errors in weld part sizes occur due to errors in building the parts or errors in their locations in the environment, making them different from the welding map. According to an industrial expert, the error for the width and height of a weld part in real cases could be up to 1% of their values, 5% for the length, and 1% for their locations. In such cases, matching the point cloud and welding map is not sufficient to find the exact coordinates of weld seams. In the next section, we introduce our approach to find weld seam coordinates when there are errors in the welding parts.

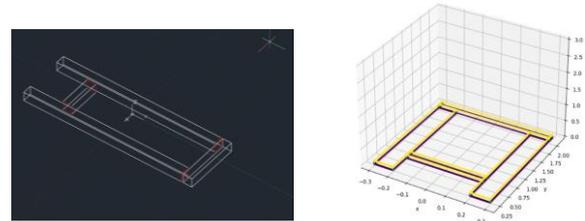


Figure 8. Welding map and its point cloud.

3.6. Locate Weld Seams in Designated Areas

These are the steps of our proposed method after globally matching the map point cloud with the preprocessed point cloud. This approach is utilized when matching the point cloud and welding map is insufficient for finding weld seam coordinates due to errors in the welding parts.

1. Set a larger mask around a specific weld seam: The welding map helps us search locally around any weld seam. Therefore, we consider one of the weld seams. Using a mask with a margin as shown in (5), we extract a cube around this weld seam.

$$\left\{ \begin{array}{l} \text{margin} = 3\text{cm} \times 5\text{cm} \times l_z \\ l_z = Z_{\max} - Z_{\min} \end{array} \right\}, \quad (5)$$

Where Z_{\max} and Z_{\min} represent the minimum and maximum coordinates of points in the Z direction.

2. Extract boundary points: From the masked point cloud, extract boundary points to remove extra points and focus on the points on the weld seam.
3. Set a smaller mask: Similar to step 1, use a mask with a margin as shown in (6) around the weld seam to further filter out extra points.

$$\left\{ \begin{array}{l} \text{margin} = 1.5\text{cm} \times 3\text{cm} \times l_z \\ l_z = Z_{\max} - Z_{\min} \end{array} \right\} \quad (6)$$

4. Calculate the start point of the weld line:
 - a. Weld lines along the y-axis:

- i. X: Find the maximum and minimum X values in the point cloud from step 3. The value closer to the map's start point X value is the desired X.
 - ii. Y: The second most frequent Y value in the point cloud from step 3. The most frequent value usually comes from noise.
 - iii. Z: The same as the map's start point Z value.
- b. Weld lines along the x-axis: Follow the same steps as for lines along the y-axis, but reverse the algorithm to calculate the X and Y values.
 - c. Weld lines along the z-axis: If this line is near the weld line along the y-axis, use the calculations from step (a). If it is near the weld line along the x-axis, use the calculations from step (b).
5. With the start point and the fixed size of the weld line, the target weld line can be calculated.

4. Experiments

The proposed method is verified using two fence shapes with different sizes, structures, and errors. It is assumed that the fences consist of boxes with a maximum size of 200×20×20 cm. Figure 10 shows the result of matching the map with the point cloud when there is an error-free fence in the simulated environment for the shape described earlier. To verify the algorithm for fence shapes with errors (meaning that fence parts differ from their original sizes in the welding map), we first added some errors to our fence shape, as shown in Figure 11. Then, we ran the algorithm on the point cloud shown in Figure 12.

To further improve the effectiveness of this algorithm, we considered a new fence with different structures and part sizes. The new fence, shown in Figure 13, includes some applied errors. Note that this shape has a new welding map with each weld part's proper size and location. The experimental results are listed in Table 1. The number of weld seams in this Table indicates the number of seams calculated among all existing weld seams in the welding parts. For each weld seam, ME is the summation of ME values along the x and y-axes. Our method achieved an average ME of 1.30 in three different scenarios.

We also implemented three other existing methods described in other studies [8, 9, 12]. The best result on our target error-free fence was achieved using the method by Takubo and their collaborators [8] with a slight enhancement. We executed their method three times for each weld seam due to the

randomness of RANSAC algorithm. The maximum ME calculated by our method shows a 30.51% improvement compared to the minimum ME calculated by the existing method.

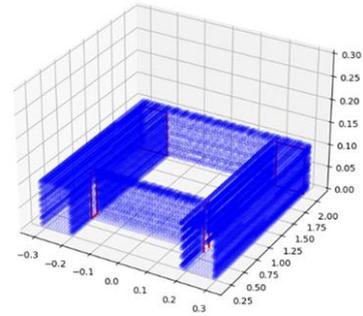


Figure 9. Extracted weld lines in the error-free fence.

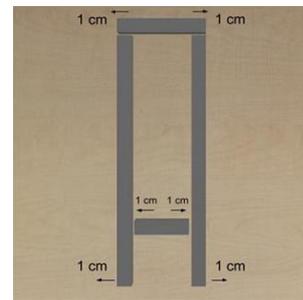


Figure 10. Errors applied to welding parts.

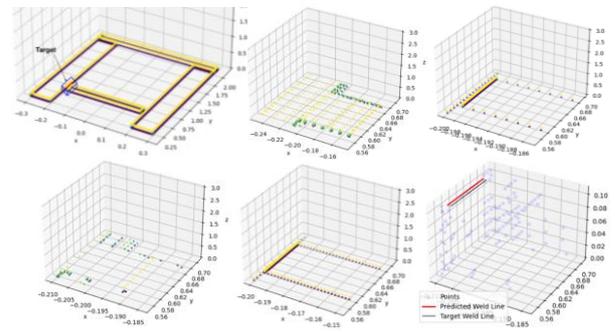


Figure 12. Execution of the method's steps on a shape with errors to calculate one target weld seam.

For our approach, the running time has been calculated using the CPU hardware accelerator in Google Colab. By comparing the execution time of calculating one weld seam in an error-free fence using our method with the approach by Takubo and their collaborators [8], our method reduces the running time by 77.43%. However, when there are errors in the fence shape, the execution time increases with the proposed method. Despite this, the execution time remains acceptable and is still shorter than the reported running times of other existing methods.

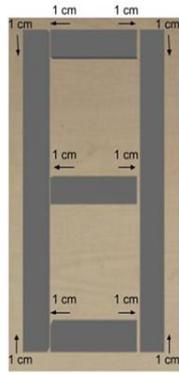


Figure 113. New fence with error.

Table 1. Weld seam detection results.

Method	Scenario	Number of Weld Seams	Average MAE (mm)	Average Running Time (ms)
Proposed method	Error-free fence	8	0.18	44
	Fence no.1 different from the map	8	1.02	565
	Fence no.2 different from the map	8	2.71	630
Best existing method [8]	Error-free fence	8	3.90	195
The improvement ratio			30.51%	77.43%

5. Conclusion

This paper proposed an efficient and applicable method for offline weld seam extraction in complex welding shapes using point cloud data and welding maps. The study addressed three main challenges: low precision, high execution time, and inapplicability to complicated welding shapes. Experiments and results in a simulated environment demonstrated that using welding maps could significantly reduce the initial search area and consequently decrease execution time. The introduced point cloud splicing technique produced a point cloud with minimal error, successfully increasing precision according to the experiments and results. Additionally, the innovative method using maps confined the search area around the weld seam, reducing costs and time, and effectively handled complex welding shapes compared to the RANSAC algorithm and other previous methods. Experiments in the simulated environment achieved high precision for offline weld seam extraction, with an average mean absolute error of 1.3 mm across different scenarios. While this study focused on developing and validating the proposed method in a simulated environment, future research will explore its application in real-world scenarios. These real-world tests will provide valuable insights into the method's performance under practical conditions,

including handling noise, irregularities, and varying environmental factors. This step will also help refine the approach and validate its feasibility for industrial applications. Future work will also focus on guiding a robot arm toward the estimated weld seam. An eye-on-hand camera will then capture new images for further calculations to refine precision even more.

References

[1] Y. Zou and R. Lan, "An end-to-end calibration method for welding robot laser vision systems with deep reinforcement learning," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 7, pp. 4270-4280, 2019.

[2] B. Zhou, Y. Liu, Y. Xiao, R. Zhou, Y. Gan and F. Fang, "Intelligent guidance programming of welding robot for 3D curved welding seam," *IEEE Access*, vol. 9, pp. 42345-42357, 2021.

[3] T. Lei, Y. Rong, H. Wang, Y. Huang and M. Li, "A review of vision-aided robotic welding," *Computers in Industry*, vol. 123, p. 103326, 2020.

[4] X. Wang, X. Zhou, Z. Xia and X. Gu, "A survey of welding robot intelligent path optimization," *Journal of Manufacturing Processes*, vol. 63, pp. 14-23, 2021.

[5] X. Wang, Y. Hua, J. Gao, Z. Lin and R. Yu, "Digital Twin Implementation of Autonomous Planning Arc Welding Robot System," *Complex System Modeling and Simulation*, vol. 3, no. 3, pp. 236-251, 2023.

[6] T. Lei, C. Wu and H. Yu, "Electric arc length control of circular seam in welding robot based on arc voltage sensing," *IEEE Sensors Journal*, vol. 22, no. 4, pp. 3326-3333, 2022.

[7] D. Rokossa, "Automatic Path and Program Generation for Fixtureless Welding with Two Synchronized Robots," in *2022 IEEE 1st Industrial Electronics Society Annual On-Line Conference*, 2022.

[8] T. Takubo, E. Miyake, A. Ueno and M. Kubo, "Welding Line Detection Using Point Clouds from Optimal Shooting Position," *Journal of Robotics and Mechatronics*, vol. 35, no. 2, pp. 492-500, 2023.

[9] L. Yang, Y. Liu, J. Peng and Z. Liang, "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.

[10] R. Kusumoto, T. Takubo and T. Tsujioka, "Welding seam detection between cylinder and plane using point cloud data," in *2022 International Symposium on Micro-NanoMechatronics and Human Science*, 2022.

[11] H. Wang, J. Xu, Y. Huang, G. Zhang, Y. Rong and W. Yu, "Multilayer positioning strategy for tubesheet welding robot based on point cloud model," *IEEE Sensors Journal*, vol. 23, no. 12, 2023.

- [12] J. Gao, F. Li, C. Zhang, W. He, J. He and X. Chen, "A method of d-type weld seam extraction based on point clouds," *IEEE Access*, vol. 9, pp. 65401-65410, 2021.
- [13] S. Yang, X. Shi, X. Tian and Y. Liu, "An Approach to the Extraction of Intersecting Pipes Weld Seam Based on 3D Point Cloud," in *2022 IEEE 11th Data Driven Control and Learning Systems Conference, 2022*.
- [14] Y. Xu, X. Tong, and U. Stilla, "Voxel-based representation of 3D point clouds: Methods, applications, and its potential use in the construction industry," *Automation in Construction*, vol. 126, p.103675, 2021.
- [15] J. L. Dai and Z. Y. Chen, "Application of ICP algorithm in point cloud registration," *Chin. J. Image Graph.*, vol. 12, no. 3, pp. 517–521, 2007.
- [16] Y. Zhang, G. Geng, X. Wei, S. Zhang, and S. Li, "A statistical approach for extraction of feature lines from point clouds," *Comput. Graph.*, vol. 56, pp. 31–45, 2016.
- [17] Seyedeh R. Mahmudi Nezhad Dezfouli, Y. Kyani, and Seyed A. Mahmoudinejad Dezfouli, "A Novel Classification and Diagnosis of Multiple Sclerosis Method using Artificial Neural Networks and Improved Multi-Level Adaptive Conditional Random Fields," *Journal of AI and Data Mining* 10.3, pp. 361-372, 2022.
- [18] 3D Geodata Academy, "Point-cloud-data-sampling-with-Python," colab.research.google.com. [Online]. Available: https://colab.research.google.com/drive/1addhGqN3ZE1mIn4L6jQnnkVs7_y__qSE?usp=sharing#scrollTo=sPPe6Hhq-_BN. [Accessed: Jan. 15, 2024].
- [19] J. Lu, W. Wang, H. Shao, and L. Su, "Point cloud registration algorithm fusing of super 4PCS and ICP based on the key points," in *Proc. Chin. Control Conf. (CCC), Guangzhou, China, 2019*, pp. 27–30.
- [20] "Bullet Real-Time Physics Simulation," pybullet.org. [Online]. Available: <https://pybullet.org/wordpress/>. [Accessed: Dec. 1, 2023].
- [21] Aka863, "ezdxf," github.com. [Online]. Available: <https://github.com/aka863/ezdxf>. [Accessed: Mar. 15, 2024].
- [22] J. Hegedűs-Kuti, J. Szőlősi, D. Varga, J. Abonyi, M. Andó and T. Ruppert, "3D Scanner-Based Identification of Welding Defects—Clustering the Results of Point Cloud Alignment," *Sensors*, vol. 23, p. 5, 2023.
- [23] E.B. Njåstad, "Robotsveising med korreksjon fra 3D-kamer," M.S. thesis, NTNU, 2015.

پیدا کردن موقعیت درز جوش با توجه به ابر نقاط و اطلاعات نقشه‌ی جوشکاری

شیوا ضیمران^۱، ولی درهمی^{۱*} و مهران مهران‌دژ^۲

^۱ دانشکده مهندسی کامپیوتر، دانشگاه یزد، یزد، ایران.

^۲ دانشکده مهندسی و علوم کاربردی، دانشگاه رجا، ساسکاچوان، کانادا.

ارسال ۲۰۲۴/۱۰/۱۳؛ بازنگری ۲۰۲۴/۱۲/۲۰؛ پذیرش ۲۰۲۵/۰۱/۱۲

چکیده:

این مقاله یک روش دقیق و کارآمد برای تعیین مختصات درزهای جوشکاری ارائه می‌دهد. با وجود دقت بالای ربات‌های جوشکاری در دنبال کردن مسیرهای از پیش تعیین‌شده، این ربات‌ها در شناسایی درزهای جوش به دلیل محیط‌های صنعتی پر نویز، الزامات دقیق برای جوشکاری مناسب، و پیچیدگی محاسباتی با مشکل مواجه هستند. روش‌های موجود برای پیدا کردن ناحیه جوش یا به نمونه‌گیری تصادفی متکی و یا محدود به هندسه‌های ساده درز جوش هستند. در اینجا، با ترکیب تکنیک‌های ادغام و هم‌ترازی نقشه‌های جوش رسم شده توسط طراح، روشی موثر برای پیدا کردن مختصات درز جوش با شکل‌های پیچیده با چندین درز ارائه می‌شود. ابتدا، ابر نقاط بدست آمده توسط دوربین‌های RGB-D موجود در محیط با یک روش وزن‌دار با هم ترکیب شده تا ابر نقاطی با نویز کم را تولید کند. سپس با بهره‌گیری از نقشه‌های جوش موجود، مناطق محتمل برای درزهای جوش در ابر نقاط بدست می‌آید. در ادامه جستجو برای تعیین مختصات درزهای جوش تنها در این مناطق محتمل انجام می‌شود، لذا به طور قابل‌توجهی فضای جستجو کاهش یافته و منجر به افزایش سرعت روش در پیدا کردن درز جوش می‌شود. با دانستن شکل تقریبی درز جوش بر اساس نقشه جوش موجود، یک تکنیک نوآورانه برای شناسایی دقیق درز جوش در مناطق محتمل به کار گرفته می‌شود. در نتایج تجربی بر روی ساختارهایی به شکل نرده در یک محیط شبیه‌سازی‌شده، میانگین خطای ۱/۳۰ میلی‌متر برای مختصات درز جوش‌ها حاصل شده است که حاکی از بهبود ۳۰ درصدی در دقت و کاهش ۷۷ درصدی در زمان محاسبات در مقایسه با آخرین روش‌های موجود است.

کلمات کلیدی: ابر نقاط، استخراج درز جوش، تشخیص سطح، ربات جوشکاری، نقشه جوشکاری.