ICRA2026 Late Breaking Results Poster presented at
2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Real-Time Seam Tracking for Robotic Welding via Registration-Based Deformation Estimation

S. Balajepalli, A. Baskaran, S. Naik, C. Eubel, A. Meyarian, A. Dai, P. Rajendran,
K. Yamane*, and A. Trazkovich
Path Robotics, Inc., Columbus, Ohio, USA
*E-mail: kyamane@path-robotics.com

***Abstract*—Arc welding induces thermal deformation that continuously displaces the seam path during execution, causing the robot to miss the joint on long seams. We present a real-time seam tracking system with three principal contributions: (1) a constrained ICP registration of live leading-laser scans against a prescan point-cloud prior combined with exponentially decaying spatial propagation; (2) a laser-line detection network retrained on 1,000 arc-on images, raising F1 from 0.59 (prescan baseline) to 0.84 on a held-out arc-on test set; and (3) an asynchronous execution architecture for ensuring that smooth joint commands are sent at the robot's control cycle (40 Hz) even with perception delays or interruptions. Internal testing confirms the system remains in the joint on 97 cm-long seams with less than 2 sec cycle-time overhead. Field deployment improved weld quality acceptance rate from 81% to 95–98%.**

I. Introduction

Industrial robotic welding systems plan trajectories offline from CAD geometry. However, the heat generated by the arc causes the workpiece to deform dynamically during welding—altering gap dimensions, joint configuration, and the seam path by as much as 10 mm on long (>90 cm) straight seams. On thin materials (1/8–3/16 in), this deformation accumulates quickly enough to drive the weld bead out of the joint, necessitating costly rework or rejection.

In this abstract, we describe our approach to real-time seam tracking for addressing the thermal deformation issue during welding. The main challenges include (1) how to conduct robust registration with partial observations; (2) how to detect the laser lines robustly under intense arc glow and spatter; and (3) how to ensure that the robot receives smooth and up-to-date joint commands even with perception delays and interruptions.

II. System Overview

The system assumes the robot is equipped with a laser scanner configured such that there is always at least one laser intersecting with the seam ahead of the torch (*leading laser*). Before welding, a *pre-weld* scan is performed and observed point cloud is registered to the CAD model, establishing a geometric prior for each seam point. This step ensures that the robot always starts the weld (when thermal deformation is not present) at the correct location.

During welding, four asynchronous threads operate concurrently:

- The **scanning** thread detects the leading laser line in images and converts them to point clouds.
- The **registration** thread consumes leading-laser observations and computes correction vectors, i.e. tip location difference from the original uncorrected trajectory.
- The **correction** thread obtains the corrected robot trajectory by computing the inverse kinematics to offset the tip location by the correction vector.
- The **point streaming** thread publishes the joint commands with the latest correction at the robot's control cycle (40 Hz).

III. Methods

A. Local Point-Cloud Registration

Let i denote the current seam point index and let $\mathbf{p}_i^{\text{pre}}$, $\mathbf{p}_i^{\text{obs}}$ denote the seam point locations in the prescan and observation clouds, respectively. The observation patch is registered against the prescan patch using ICP [1] with *constrained degrees of freedom*: translation along the seam direction is suppressed, anchoring the solution at the still-rigid seam start. The resulting transform yields the correction vector

$$\Delta_i = \mathbf{p}_i^{\text{obs}} - \mathbf{p}_i^{\text{pre}}, \tag{1}$$

representing the lateral and vertical displacement due to thermal deformation.

To smooth corrections across upcoming seam points, we apply an exponentially decaying model:

$$\Delta_{i+k} = \Delta_i \cdot e^{-\alpha k}, \quad k = 1, 2, \ldots, n, \tag{2}$$

where α is a tunable decay rate. This pre-corrects future waypoints, ensuring smooth robot motion and fast convergence as deformation accumulates.

B. Arc-On Network Retraining

Standard laser-line detection networks trained on arc-off prescan images fail under arc-on conditions: the prescan network achieves recall of only 0.52 on arc-on images—meaning the laser line is missed nearly half the time, eliminating the correction signal.

1) Data collection and annotation: We collected 8,300 arc-on images from test welds with varying voltage, travel speed and surface conditions resulting in a range of spatter as well as fume and arc brightness intensity. Images were manually annotated for training and validation.

TABLE I
PERFORMANCE OF THE LASER LINE SEGMENTATION NETWORKS

Network	Precision	Recall	F1
Prescan baseline	0.77	0.52	0.59
Arc-on retrained	0.78	**0.86**	0.80
Arc-on + negatives $\star$	**0.88**	**0.88**	**0.87**

2) Training configurations: Two networks were trained on this dataset using a SegFormer [2]-based backbone (MiT-B0) with the U-Net decoder:

- **Arc-on only**: trained on arc-on images; recall improves from 0.52 to 0.86.
- **Arc-on + negatives** (deployed): augments with hard negative samples of arc-flash events; improves precision (0.78→0.88) with further improvement in recall.

Detailed results are shown in Table I.

C. Asynchronous Execution Architecture

The robot controller requires the commands to be sent at a rate of 40 Hz. In contrast, the registration step takes up to 100 ms depending on the size of point clouds and amount of correction. It is also possible that no useful observations come in for a while, for example at the end of a weld where the leading laser goes out of the seam.

To ensure uninterrupted, smooth robot motion, we employ an asynchronous execution architecture where multiple threads, each dedicated to a single task, communicate through shared buffers as follows:

- **Scanning → registration**: The scanning thread pushes the observed point clouds to **ObersvationBuffer**. The registration thread waits until sufficient new observations are accumulated, and uses them to compute the associated correction vector.
- **Registration → correction**: The correction vectors are pushed to **CorrectionBuffer**. The correction thread computes the inverse kinematics to offset the tip location of *all* points after the current observation. This ensures that the robot always receives smooth commands even with perception delays or interruptions.
- **Correction → point streaming**: The correction thread overwrites the joint states stored in **SharedTrajectory**. The point streaming thread publishes the latest joint states at a fixed (40 Hz) rate.

IV. RESULTS

A. Internal coupon tests

The system was validated on 97 cm-long seams on $3/16$ in and $1/8$ in stock coupons. All acceptance criteria were satisfied:

- The weld bead remained in the joint for the full 97 cm length on both stock thicknesses.
- Cycle time: 49 s without seam tracking vs. 50 s with seam tracking on an 8-inch reference weld (<2 sec overhead).

B. Field deployment

The system was deployed in the field and welded 3 parts each with 200+ seams totaling over 42 m of weld length, 83% of which was welded with seam tracking. These parts achieved **95–98%** weld quality acceptance rates compared to the **81%** average acceptance rate prior to the deployment. Field seam placement error was $\leq$1.0 mm.

V. DISCUSSION

The largest remaining failure mode is insufficient prescan cloud density due to occlusion by tack welds. The system correctly detects this condition and falls back to non-seam-tracking execution rather than applying unconstrained corrections.

The arc-on network retraining demonstrates that 8,300 carefully annotated images suffice to close the domain gap between arc-off and arc-on conditions. Hard negative samples (arc-flash events without a visible laser line) were critical for controlling false-positive rate, which would otherwise inject spurious corrections and destabilize the trajectory.

VI. CONCLUSIONS

We have presented a real-time seam tracking system for industrial robotic welding addressing the challenges including geometric correction of thermally induced deformation and robust laser-line perception under arc illumination. Key properties of the system are:

- **Accurate and smooth corrections** are achieved via the point-streaming control interface without interruptions.
- **Constrained ICP** against a pre-weld scan prior isolates lateral/vertical shift from along-seam motion.
- **Asynchronous architecture** with shared buffers guarantees continuous joint commands under perception delays and interruptions.
- **Arc-on network retraining** raises laser-line detection F1 from 0.59 to 0.84, with recall from 0.52 to 0.86.
- **Field validation** confirms 81%→97% weld quality improvement with <2 sec cycle-time overhead.

VII. FUTURE WORK

A data-driven deformation motion model (e.g., PointNet or Transformer trained on synchronized leading/trailing laser datasets) could replace the exponential decay prior for nonlinear deformation profiles near thin stock. Extending scope to lap joints and non-straight seam geometries requires relaxing the along-seam ICP constraint and developing real-time seam-length adaptation. Continued arc-on data collection across new cell configurations will maintain network generalization as the fleet expands.

REFERENCES

[1] P.J. Besl and N.D. McKay: “Method for registration of 3-D shapes,” Sensor fusion IV: control paradigms and data structures, vol. 1611, pp. 586–606, 1992.

[2] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J.M. Alvarez, and P. Luo: “SegFormer: Simple and efficient design for semantic segmentation with transformers,” Advances in neural information processing systems, vol. 34, pp. 12077–12090, 2021.