Jarkko Kotaniemi* and Janne Saukkoriipi and Tuomas Seppälä

Robot Deburring Path Generation Based on a 3D Point Cloud

Abstract: Modern manufacturing demands flexibility, customization, and efficiency in traditionally rigid processes. Digital automation has been increasingly more prevalent. With small batch sizes, increased automation means that offline programming and planning of tool paths takes a larger portion of work for each part. We present a method of creating a robot path for deburring an oxy-fuel cut part using a 3D point cloud. This method allows for faster automation of a common finishing step in parts manufacturing without the use of a CAD model or offline programming. Experiments show that our method can be used to easily generate a deburring path on a part that is randomly placed in the work area.

Keywords: point cloud, robot path generation, deburring

***Corresponding Author: Jarkko Kotaniemi:** VTT Technical Research Centre of Finland, E-mail: jarkko.kotaniemi@vtt.fi

Janne Saukkoriipi: VTT Technical Research Centre of Finland, E-mail: janne.saukkoriipi@vtt.fi

Tuomas Seppälä: VTT Technical Research Centre of Finland, E-mail: tuomas.seppala@vtt.fi

1 Introduction

With ever-increasing demands for agility and flexibility, robotized manufacturing is facing challenges and barriers to the use of robots because; the high cost of engineering the work cells (i.e., the design, fabrication, and installation of jigs, fixtures, conveyors, and third-party sensors and software) [1]. Robot skills are high-level abstractions of tasks, that enable robots to perform complex operations like pick-and-place, grinding, or assembly with minimal low-level coding [2]. By leveraging skill programming, developers can streamline robot programming, enhance adaptability across various tasks, and accelerate deployment in dynamic environments.

3D cameras integrated with robotic systems are predominantly utilized for object localization and quality control. Conventionally, object detection and localization are achieved using 3D point clouds, while the generation of robot paths continues to depend on CAD models and offline programming tools. In the method we present, CAD objects are not utilized, and robot paths are generated directly from the captured point cloud data.

2 3D Point Cloud Processing

Our point cloud processing is performed using in-house developed software, which requires a structured point cloud, the default output for 3D depth cameras. Initially, the software extracts surface segments by comparing the distances between neighboring points and the directions of their surface normals. Optionally some filtering can also be applied: removing the background (like the table) or decimation. Once the segments are extracted, inner and outer edge points are identified for each segment, separated into individual edges, and incorporated into the segment data. In this context, the outer edge defines the boundary between the plane and the burr. This edge is then analyzed to provide vertices. Due to the randomness of the burr lines, the shapes remain overly detailed. For robot deburring, it is only necessary to identify the main corners and disregard the irregularities of the burr line and thus require more filtering.

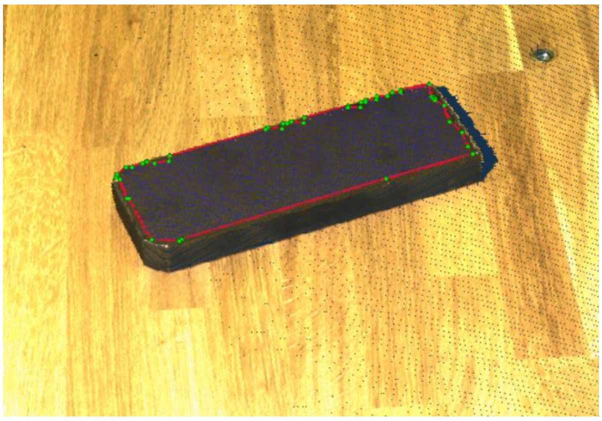

Fig. 1. Segmented edge and extracted vertices of the part. The edge is marked in red, and the vertices are marked in green.

Besides vertex extraction (Figure 1), more detailed analysis of the burr can also be performed. Information such as the height of the burr can help determine which tool should be used for deburring. The burr in the object is calculated by analyzing the segment’s neighborhood, identifying which unsegmented points are located above the surface, and calculating their distance from the segment plane. A minimum distance limit is recommended to eliminate noise in the segmentation and accurately recognize only the burrs (Figure 2).

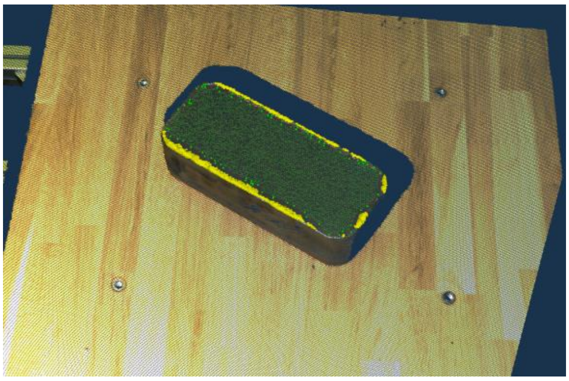

Fig. 2. Detected burr is marked in yellow.

3 Robot Tool Path Generation

The extracted vertices of the part are used to generate the tool path for the deburring operation. Because of the uneven nature of the burr, there are quite many extra vertices even on a straight edge of the part which need to be filtered out. Using the Ramer–Douglas–Peucker (RDP) algorithm, the excess detail given by the vertices can be reduced to only the essential portions, which are the actual corner positions of the part. For effective deburring, it is crucial for the tool to follow the edge's direction. By aligning the tool's orientation to point toward the subsequent edge pose, the deburring process can be executed properly.

The path generation generates a path in JSON format that the robot can execute easily utilizing our robot skill-based programming system. The path was programmed so that it starts from the closest corner to the robot and operates in clockwise direction around the object.

4 Experimental Setup and Testing

The generated paths were implemented in our laboratory test facilities. The test objects were metal objects that had been cut and had burrs within them. As 3D camera we used Zivid2+ M60 model which represents the high-end of 3D cameras. 3D camera was placed in the environment and calibrated to the robot with methods presented in here [3]. The robot used in the tests was Universal robot’s UR5 with a simple pointer tool (Figure 3). The aim of this experiment was not to do actual deburring but to demonstrate the accuracy of the generated path.

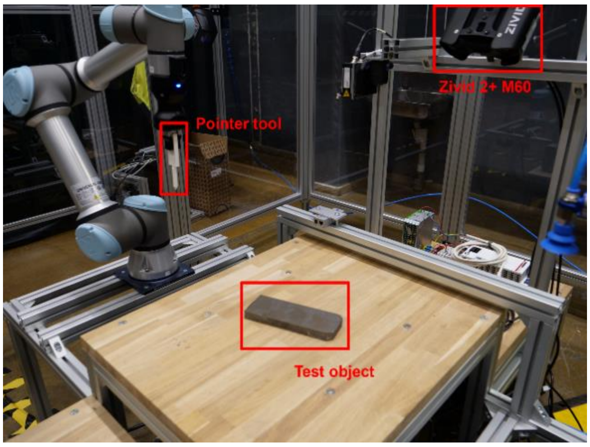

Fig. 3. Experimental setup with the tool, test object and camera marked.

5 References

[1] J. Saukkoriipi, T. Heikkilä, J. Ahola, T. Seppälä and P. Isto, "Programming and control for skill-based robots" *Open Engineering*, vol. 10, no. 1, 2020, pp. 368-376. https://doi.org/10.1515/eng-2020-0037

[2] T. Lohi, S. Soutukorva and T. Heikkilä, "Programming of Skill-Based Robots", *2024 IEEE 19th Conference on Industrial Electronics and Applications (ICIEA)*, Kristiansand, Norway, 2024, pp. 1-7, https://doi.org/10.1109/ICIEA61579.2024.10664981

[3] T. Heikkilä, J. M. Ahola, J. Koskinen and T. Seppälä, "Calibration procedures for object locating sensors in flexible robotized machining", *2014 IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, Senigallia, Italy, 2014, pp. 1-6, https://doi.org/10.1109/MESA.2014.6935567