Intuitive Robot Programming by Demonstration

This document is a short summary of [Bagge-Carlson(2013)]

Fredrik Bagge Carlson*
Lund University
Faculty of Engineering
ael07fba@student.lu.se

Background

Recent developments in industrial robotics has led to cheaper robots, which have become more accessible to small and medium-sized enterprises (SMEs). The traditional way of programming a robot however has not seen any such large scale advancements and still require a significant amount of man-hours, thus being a major part of the costs related to operating an industrial robot. Traditionally, robots are programmed using an *on-line* programming procedure. This requires a robot operator to manually control the robot to a number of poses along the actual trajectory, which will be stored in a robot program. This must be done for all individual robots on the floor, requiring a significant amount of work load. Another way of programming industrial robots made popular by increased absolute accuracy and a demand for more complex trajectories, is *off-line* programming. Off-line programming involves the planning of a robot trajectory using a CAD model of the workpiece.

Both programming methods described above require a significant time investment to be successful. For a large enterprise operating at high volumes, this may be of low concern. For SMEs however, the situation might be different. SMEs often face High-Mix, Low-Volume (HMLV) orders. To program an industrial robot using on-line programming might be economically unfeasible if the task is to be repeated a low number of times. Furthermore, the CAD model of the work piece may not be available. This might be the case if none ever existed or if the work piece has suffered wear and tear since it was fabricated and the CAD model no longer represents the actual work piece. Without a CAD model, the robot programmer must resort to reverse engineering of the work piece in order to allow off-line programming.

To summarize, many SMEs would benefit greatly from a faster, more cost efficient way of programming industrial robots, which do not require a CAD model of the work piece. A company looking for a way to automate HMLV orders, such as maintenance and finishing tasks, require an economically feasible programming method where a desired solution meets the following requirements

*Work supported by the project C12-R-006 at Singapore Institute of Manufacturing Technology

- Low cost implementation.
- No CAD model requirements.
- Sufficient accuracy for contact finishing tasks using force control operation.
- Reduced programming time compared to on-line programming.
- Low operator skill level required.

This work will asses the feasibility of constructing a robot trajectory by capturing the users hand movements as he demonstrates the trajectory. The ultimate goal is to develop a method which will reduce the costs related to the programming and operation of an industrial robot, while maintaining an acceptable level of accuracy for tasks such as spray painting, polishing or deburring.

Recording a path

Using an optical tracking system, consisting of a number of cameras mounted above the workspace (figure 1), the location of a small handheld device can be captured. This allows the user to indicate a path for the robot to execute.

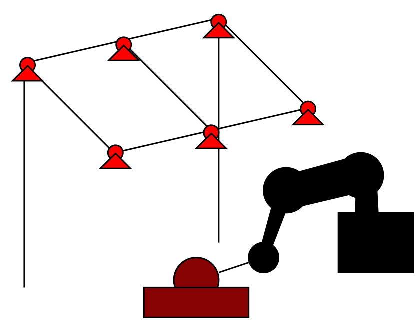

Fig. 1. Workspace setup

Indicating a feature on a workpiece without error has proven difficult. Often times, errors caused by the user can reach several millimeters, raising the need for path reconstruction. Analysis of the recorded data determines the properties of the indicated path, ensuring proper reconstruction for each segment along the path. Figure 2 illustrates a typical recorded path along with its reconstruction.

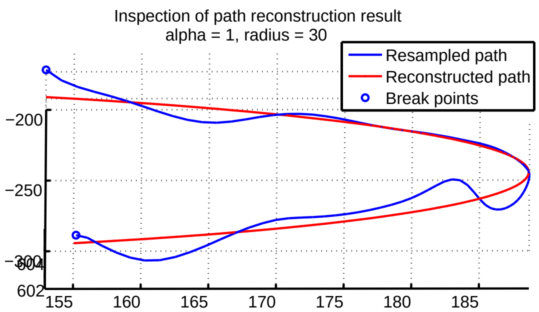

Fig. 2. Reconstructed path

Scanning the workpiece

When a path is indicated, the surface of the workpiece will be scanned using a 3D scanner mounted on the robot (see figure 3). After passing through filtering, normal estimation and curvature estimation, the scan can be used to gain additional knowledge of the workpiece, needed to automatically construct a robot program. The estimated normal direction of the workpiece surface, together with the tangent of the path, determines a target frame. A target frame specifies both the desired location and orientation of the robot tool as it passes through a point on the path. Figure 4 illustrates a scanned surface, on top of which a path is indicated. Estimated normal directions are shown with yellow lines.

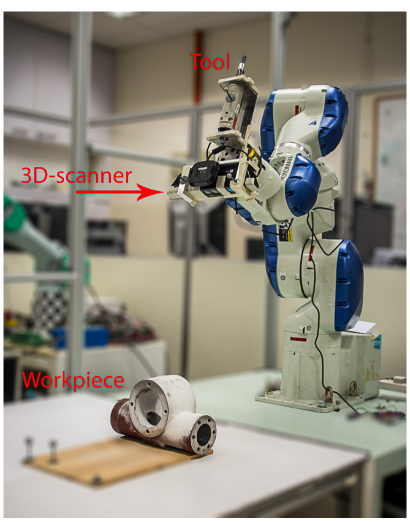

Fig. 3. End effector assembly

Executing the path

Due to errors in both the robot kinematic model and the calibration between the robot base coordinate system and the coordinate system of the tracker, an executed path will not coincide with the desired path. By recording the executed motion and comparing it to the desired trajectory, a transformation can be found which maps them together. By applying this transformation to the desired path before execution, the trajectory following error can be reduced significantly. Figure 5 illustrates an example where the first execution of the program resulted in a large error. After applying the calculated transformation, the error is reduced.

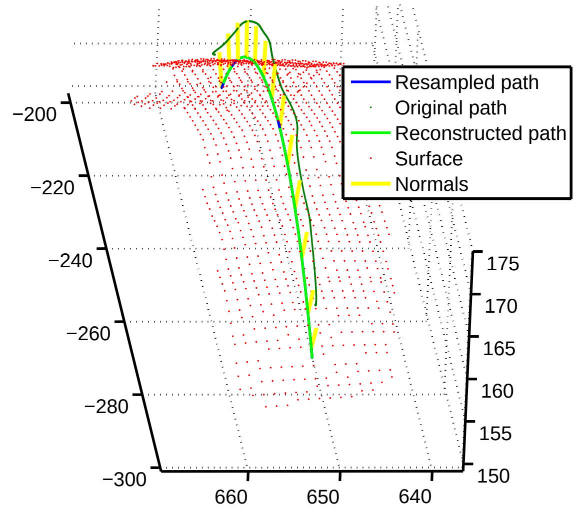

Fig. 4. Scanned surface with estimated normals

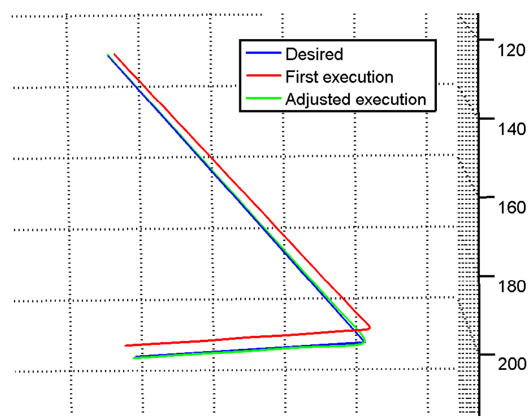

Fig. 5. Local calibration improvement reduces trajectory following errors.

Results

By being fast and using no expensive equipment, the developed programming method provides a cost efficient way of programming industrial robots. A path can typically be indicated and executed with as little as 1mm trajectory following error, making it suitable for execution using compliant tools or force control.

References

[Bagge-Carlson(2013)] F. Bagge-Carlson. Intuitive robot programming by demonstration. Master's thesis, TFRT-5918, Dept. Automatic Control, Lund University, 2013.