Vision-guided-robotics retrofit reinvigorates a 10-year-old robot

Dennis Murphey and Remus Boca

Upgrading older autonomous workstations with imaging technologies for finding and identifying parts is gratifying, but not easy.

Braintech has been deploying vision-guidance software for over 10 years. We were recently asked to support one of the top automakers with a robotic workstation upgrade project: a sheet-metal welding line that consisted of hundreds of robotic cells, many with aging robots and old controllers. Most stations relied on hard tooling and limited sensor guidance. The chief problem for the site was the frequent downtime of the existing system. The line would go out of commission because of inconsistent part placement, and the laser system (used to help position the robot) was slow and inaccurate. The automaker asked Braintech to update four robotic workstations as a retrofit exercise. The problem to be solved was twofold: alleviating downtime due to sensor inaccuracy, and reducing the length of training and runtime cycles (i.e., to identify a part, calculate its position, and send the 3D position to the robot).

The first order of business was to carry out a feasibility study using Sony HR70 cameras equipped with 8mm lenses. The welding line comprised several part models. We mounted a camera at each end of the robot effector, or gripper. The robot would move to an image-capture position (i.e., camera lens to part surface) at ~950mm from the upper section of the part (i.e., in the rack) to enable the upper camera a view of that area (see Figure 1). The robot then moved the lower camera to the lower section of the part (see Figure 2). Depending on the accuracy required, one or two images per camera were obtained. A 3D frame was calculated for each section, followed by a pose average, which was sent to the robot. The robot would pick up the part inside the newly created 3D frame. The robot then moved to an indexed position for the second part (again with camera lens at a distance of ~950mm), and the process was repeated.

The problem with integrating retrofit systems is the age of robot controller technology and the lack of open standards and training time. This is especially true for retrofit cases as described above, where machinery can be more than three to five years old. It can also be the case that a three-year-old robot is operating with a five-year-old controller. Since most of the robot controllers use proprietary solutions whose architecture is closed, solutions are not readily available.

The vision-guidance application we devised provided a rapid solution by using a number of enhanced tools that require minimal human intervention. The ultimate goal is to use robot movement to simulate part movement. For example, programs that automatically move the robot in a variety of ways make it possible to calibrate the camera’s internal and external parameters, to

Continued on next page
calculate (train) the 3D model of the part, and finally to validate the vision solution (‘Accutest’).

Figure 2. Second camera viewing position for the lower area.

Figure 3 presents sample results from this tool test. The report includes the percentage of images for which the 3D pose of the part was calculated, the percentage of part features detected, and a breakdown of the processing time for the pose calculator. The report also shows statistics for each translation and rotation component of individual poses. For instance, a pose average is calculated from all the poses and then the differences between each pose and the average are determined. All of these calculations go into constructing statistical data sets for each component.

From the point of view of vision guidance, integration with a robot controller is not difficult because the required programming is greatly simplified (the calculation is done on the vision guidance side) and because a generic interface for developing robot-specific modules is available. In this specific retrofit example, the robot programs and specific modules were a rate-limiting factor due to the old controller. When variables were sent to the robot, the program had to be stopped. Moreover, creating 3D frames on the robot side was challenging. Finally, the communication software development kit (SDK) was limited and not very well documented. All of this required learning unfamiliar robots and programming software before the vision solution could begin integrating.

The challenges that we faced were mostly application-specific to the old robot and not related to vision-guidance-application programming. In particular, proper 3D data and coordinate reference-frame information had to be sent to the controller, which entailed experimenting with the robot position. 3D frames must be created on the vision-guidance side because creating them on robot controllers requires frames that include points that the robot can reach. This process becomes iterative, as it is presently difficult to adequately represent the robot model in the vision-guidance program.1–4

The technologies available today in our vision-guided robotics software include single 2D, 2.5D, and 3D cameras, a 3D multicamera (for focusing on different sections of a part), stereo, random bin picking, and structured 3D light. While these capabilities enable ambitious vision-guidance solutions, the integration process remains time-consuming.

The existing vision tool and existing pose calculators helped us to devise a relatively fast vision-guidance solution compared with conventional alternatives. But there are areas that can be improved, such as using simulation and better integration with robot controllers. For the work reported here, the vision solution provided a 500% improvement in accuracy, 50% improvement in image-capture cycle time, and reliability increased from 30 to 99%. The 10-year-old robot now outperforms more recent installations.

In summary, this case study illustrates the current status of vision-guided robotic technology. The most time-consuming tasks in engineering an automation workstation retrofit continue to be the setup, integration, and debugging processes. Making adaptive vision technology available through an open and portable architecture would enable objects to be located instantly at any time. Object recognition would also help to identify objects and capture data for reporting status and so on. These enhancements would expedite development of vision-guidance solutions for specific applications. Dynamic simulation of workstations would aid preinstallation engineering, while enhanced

---

Continued on next page
adaptive vision would help to solve deployment issues. Retrofitting older robots to improve operation and productivity is feasible. But now we need to expand the application interface and tools within the SDK to make dealing with new or unfamiliar controllers faster and easier.

Author Information

Dennis Murphey
BrainTech Inc.
McLean, VA
http://www.braintech.com

Dennis Murphey joined Braintech in 2009. In the late 1970s, he developed and patented a robotic assembly system using machine vision for random bin picking. His has also worked on six- and nine-degrees-of-freedom and gantry robotics for welding and assembly, an 18-robot electronics assembly line, and an open-architecture robotic controller. He received a master’s in administrative science with an emphasis in computer science and optics from the Johns Hopkins University.

Remus Boca
BrainTech
Pontiac, MI
http://www.braintech.com

Remus Boca has applied experience in advanced computer programming, robotics, vision, and mechatronics. In 2002 he joined BrainTech, where he directs the modeling, development, and implementation of robotic-vision technologies such as single- and multicamera 3D imaging, random bin picking, automatic calibration, 3D model training, and automatic testing of vision solutions. In 1994, he graduated from the Polytechnic University of Bucharest, Romania, with a BS and MS, and in 2001 he completed his PhD on automated computation of the characteristics of industrial robots.

References