MATLAB Toolboxes: Robotics and Vision for Students and Teachers

By Peter Corke

In this column, Dr. Peter Corke of CSIRO, Australia, gives us a description of MATLAB Toolboxes he has developed. He has been passionately developing tools to enable students and teachers to better understand the theoretical concepts behind classical robotics and computer vision through easy and intuitive simulation and visualization. The results of this labor of love have been packaged as MATLAB Toolboxes: the Robotics Toolbox and the Vision Toolbox.

–Daniela Rus, RAS Education Cochair

MATLAB (http://www.mathworks.com) is a computational environment for linear algebra, graphics, and dynamic simulation. It is available on a very wide range of computer platforms and is extensively used in universities for teaching and research. The core functionality of MATLAB can be extended by application-specific toolboxes and many are available commercially or under various open-source licenses. The fundamental data types of MATLAB, the vector and matrix, are highly applicable to problems in both robotics and computer vision. This column briefly discusses two open-source extensions for MATLAB, the Robotics Toolbox (RT) and the Machine Vision Toolbox (MVT).

RT [1] provides a collection of functions that support fundamental algorithms in robotics such as representations of orientation in SO(3), kinematics, dynamics, and trajectory generation. Most robotics textbooks present examples with simple two-link robots for which the analysis is tractable. For a realistic six-axis robot, the kinematics and dynamic computations are complex and it can be difficult to obtain insight into a problem. The toolbox contains functions to make it as easy to operate on a two-link robot as for a six-link (or more) robot. For instance, the effect of payload mass on the inertia matrix, or the variation in link inertia seen by the motor, can be very easily explored, which is difficult, if not impossible, to do analytically. The toolbox is based on a very general method of representing the kinematics and dynamics of serial-link manipulators using objects. Each link is represented by a link object whose attributes are the standard or modified Denavit and Hartenberg parameters as well as link and motor inertial properties, friction, and gear ratio.

The link objects comprise a robot object on which operations such as forward and inverse kinematics and forward and inverse dynamics can be computed. Example objects are provided for classical robots such as the Puma 560 and the Stanford arm. The toolbox also provides functions for converting between homogeneous transformations and various angle and axis/angle representations as well as providing a class to support unit quaternions.

Simulink, a companion product for MATLAB, provides for simulation of dynamic systems based on a block-diagram modeling language. Wrappers for the toolbox functions allow the nonlinear robot system to be described in block diagram form, enabling the closed-loop performance of a user-provided axis control system to be studied.

The MVT [2] provides a collection of functions for machine vision and vision-based control. It is a somewhat eclectic collection reflecting the author's personal interest in areas of photometry, photogrammetry, and colorimetry. An image is

Digital Object Identifier 10.1109/M-RA.2007.912004

usually treated as a rectangular array of scalar values (representing intensity or perhaps range), which is a natural data type for MATLAB, and many image operations such as thresholding, filtering, and statistics can be achieved with existing MATLAB functions. The toolbox extends this core functionality with more than 60 functions spanning operations such as image file reading and writing, acquisition, display, filtering, blob, point and line feature extraction, mathematical morphology, homographies, visual Jacobians, camera calibration, and color space conversion. For modest image sizes, the processing rate can be sufficiently real time to allow for closed-loop control.

The functions in both toolboxes are written in a straightforward, or textbook, manner for pedagogical reasons rather than for maximum computational efficiency. MATLAB vectorization has been used as much as possible to improve efficiency, and some particularly compute-intensive functions are provided in C-source as MEX files that execute in the native architecture. It is also possible to use MEX files to interface with image acquisition hardware and robots, moving from being a pure simulation tool to a control tool.

These toolboxes have been progressively developed since the early nineties and have tracked considerable change in the MATLAB language, the most interesting and important of which was the introduction of objects in MATLAB5 that greatly simplified the representation of robot links, robots, and quaternions. Operator overloading allows for natural representation of operations such as composing robots or quaternions. The MVT is not a clone of the Mathwork's own Image Processing Toolbox (IPT), although there are many functions in common. MVT predates IPT by many years, is open-source, and is slanted toward image feature extraction and control.

Clearly, there is something odd about open-source software, such as these toolboxes, requiring a proprietary software environment in which to run, even if it is reasonably priced for universities. A number of open-source packages with some of MATLAB's functionality do exist, for instance Octave (http://www.octave.org) and SciLab (http://www.scilab.org). Both environments share many aspects of MATLAB syntax but are each different enough to confound efforts to support all three. Particular points of difference are with respect to graphics and block-diagram modeling of dynamic systems. A port of the RT to SciLab has recently been completed and a port to Python is also being developed by the author. Python, with the numpy and matplotlib (http://matplotlib.sourceforge.net/) classes, brings scientific capability to an efficient modern portable and truly object-oriented language that also has rich support for networking. Python is widely used in scientific communities such as astronomy.

The toolboxes are available from www.ict.csiro.au/downloads.php. Each toolbox is accompanied by a comprehensive manual in the style of a MATLAB toolbox manual and some tutorial examples.

References

[1] P. Corke, "Machine vision toolbox," *IEEE Robot. Autom. Mag.*, vol. 12, pp. 16–25, Nov. 2005.

[2] P. Corke, "A robotics toolbox for MATLAB," *IEEE Robot. Autom. Mag.*, vol. 3, pp. 24–32, Sept. 1996.

Peter Corke is a chief research scientist with the CSIRO ICT Centre. His research activities include span machine vision, vision-based robot control, field robotics (with a focus on mining applications), real-time systems, and sensor networks. He holds B.Eng. and M.Eng.Sc. degrees, both in electrical engineering, and a Ph.D., all from the University of Melbourne. He is an adjunct professor in ITEE at the University of Queensland, a founding editor of the *Journal of Field Robotics,* and a member of the editorial board of the *International Journal of Robotics Research*. He has held visiting positions at the CMU Robotics Institute, the Coordinated Science Laboratory at UIUC, and the GRASP Laboratory at the University of Pennsylvania.

Address for Correspondence: Peter Corke, P.O. Box 883, Kenmore, Australia, 4069. E-mail: peter.corke@csiro.au.