

DOI:10.3969/j.issn.1001-4551.2015.11.026

基于 SFM 的移动机器人运动 轨迹的计算方法研究^{*}

陈 健, 李晓明^{*}, 苏必达, 吴 昊
(浙江理工大学 机械与自动控制学院, 浙江 杭州 310018)

摘要:针对移动机器人在摄像机标定过程中所行驶的运动轨迹计算问题,对摄像机标定的世界坐标系、摄像机坐标系与图像物理坐标系、经典针孔成像模型、标定基本原理、三维重建的 SFM 思想,以及运动轨迹计算方法进行了研究。首先分析了在摄像机标定过程中的 4 个常用坐标系之间的转换关系和摄像机模型;其次在摄像机标定中,介绍了基本投影几何、透镜畸变以及基本标定方法;最后在移动机器人运动轨迹的坐标计算原理的基础上,在 VC++ 环境下运用 OpenCV 函数库进行了具体的实验与详细地分析,并使用 Matlab 计算矩阵公式并得出了相应运动轨迹的三维图。研究结果表明,该方法充分发挥了 OpenCV 函数库功能,提高了实验结果的准确性,完成了摄像机的标定工作,获得了机器人的运动轨迹坐标。

关键词:摄像机标定;移动机器人;运动轨迹;OpenCV

中图分类号:TP39;TP24

文献标志码:A

文章编号:1001-4551(2015)11-1524-05

Calculation method of motion trajectory for mobile robot based on SFM

CHEN Jian, LI Xiao-ming, SU Bi-da, WU Hao

(School of Mechanical Engineering and Automation, Zhejiang Sci-tech University, Hangzhou 310018, China)

Abstract: Aiming at the problem that the calculation method of motion trajectory for mobile robot based on camera calibration, numerical simulation was carried out to study the coordinate system of the world, the camera, and the physics image, the SFM idea of 3D reconstruction, as well as the classic pinhole camera model, and the calculation method of motion trajectory of the camera calibration. Firstly, the work was conducted on the four common coordinates transformation in camera calibration and camera model. Then, the basic projective geometry, the lens distortions and the basic camera calibration were introduced. Lastly, the coordinate calculation method of motion trajectory for Mobile Robot was summarized, and tested and analyzed on OpenCV in Visual C++ environment. The calibration result was worked out by Matlab on the formula and the 3D figure of motion trajectory for Mobile Robot was presented. The results indicate that the proposed method makes use of the functions of the library effectively, improves precision of experimental result, finishes the work of the camera calibration, and obtains the coordinates of the motion trajectory for mobile robot.

Key words: camera calibration; mobile robot; motion trajectory; OpenCV

0 引 言

随着计算机技术的快速发展,以及多种新型传感器的广泛使用,移动机器人的应用在人们生活中起到了重要作用。通过移动机器人拍摄的多组图片,还原

模型的基本三维结构也成为了计算机视觉技术领域的
一个重要分支。从二维图像中获得三维信息,必须利用光学上的数学矩阵变换,以及实验与计算得到,这一过程称为摄像机标定,即确定摄像机的几何模型和光学参数。如何通过移动机器人所提供的摄像机标定来

收稿日期:2015-07-03

基金项目:国家自然科学基金资助项目(61271110);国家光学辐射重点实验室基金资助项目(9140C610505120C61232)

作者简介:陈 健(1990-),男,湖北十堰人,主要从事机器人视觉技术方面的研究. E-mail:chenjian158978@gmail.com

通信联系人:李晓明,男,副教授,硕士生导师. E-mail:lxmzist@zstu.edu.cn

计算机人的运动轨迹是本研究的研究方向。

本研究的研究思路主要源于 SFM (structure from motion) 思想,根据运动相机拍摄序列图像,来反求移动机器人轨迹的坐标,是 SFM 技术的用途之一。目前该研究国外已有较成熟的成果,并有完整的理论与应用,例如 Shervin Emami 等人^[1]所编写《Mastering OpenCV with Practical Computer Vision Projects》的第四章“Exploring Structure from Motion Using OpenCV”,以及谷歌软件工程师 Changchang Wu^[2]所研发的基于 SFM 思想的 GUI 三维重建系统 (VisualSFM) 等等,而国内虽有部分领域的研究,但仍不够深入化与系统化。该研究并非旨在图像的三维重建,而是通过拍摄的图像返求运动的轨迹,有助于机器人的导航、漫游以及定位,对便捷生活与军事活动都有许多帮助。

本研究针对移动机器人在摄像机标定过程中所行驶的运动轨道计算问题,对摄像机标定的世界坐标系、摄像机坐标系与图像物理坐标系。经典针孔成像模型、标定基本原理、三维重建的 SFM 思想以及运动轨迹计算方法进行研究。

1 摄像机标定原理

1.1 世界、摄像机与图像物理坐标系

本研究中的摄像机安放在移动机器人上,摄像机模型是以最简单的针孔成像模型为基础。摄像机成像模型及四个常用坐标系如图 1 所示,光心为 O_c ,视野品面为像面 π ,其到 O_c 的距离为镜头焦距 f 。研究过程中常用到的 4 个坐标系^[3]分别为:世界坐标系(O, x_w, y_w, z_w)、摄像机坐标系(O_c, x_c, y_c, z_c)、图像物理坐标系(O_1, X, Y) (单位 mm),以及图像像素坐标系(O, u, v) (位于视野平面左上角,单位 pix)。

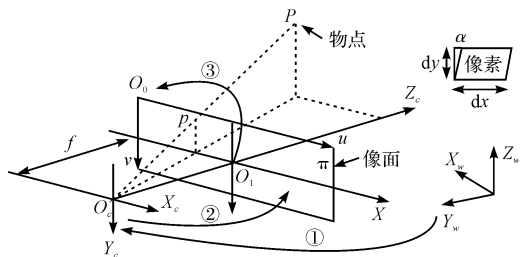


图 1 摄像机成像模型及 4 个常用坐标系

将空间中某点 $P(X_w, Y_w, Z_w)$ 转换到像点 $p(u, v)$ 主要通过这 4 个坐标系之间的 3 次转换:

(1) 刚体转换。将世界坐标系进行平移和转换 (即 $R = \vec{t}$) 得到摄像机坐标系,即 $(x_w, y_w, z_w) \rightarrow (x_c, y_c, z_c)$,转换公式如下:

$$\lambda A[R, \vec{t}] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \frac{1}{Z_c} \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} \quad (1)$$

(2) 透视投影变换。将摄像机坐标系 (O_c, x_c, y_c, z_c) 通过三角几何变换得图像物理坐标系,即 $(x_c, y_c, z_c) \rightarrow (x, y)$,转换公式如下:

$$\frac{1}{Z_c} \begin{bmatrix} f_x & y & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} 1/d_x & tga/d_x & u_0 \\ 0 & 1/d_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (2)$$

(3) 线性变换。将图像物理坐标系通过像素和公制单位的比率得到图像像素坐标系,即 $(x, y) \rightarrow (u, v)$,转换公式如下:

$$\begin{bmatrix} 1/d_x & tga/d_x & u_0 \\ 0 & 1/d_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (3)$$

1.2 针孔成像模型

摄像机模型采用经典的针孔成像模型 (pin-hole model),同样 OpenCV 亦是以其为基础。将世界坐标系中的点投影到摄像机上,可用齐次坐标和矩阵表示:

$$\mathbf{q} = \mathbf{K} \mathbf{D} \quad (4)$$

$$\text{式中: } \mathbf{q} = \begin{bmatrix} x \\ y \\ w \end{bmatrix}, \mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{Q} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}.$$

展开该式,可发现 $w = Z$,并且 $\mathbf{q}$ 是齐次坐标形式。其中 $\mathbf{K}$ 矩阵为摄像机内参数矩阵,是 3×3 矩阵。 $\mathbf{K}$ 矩阵中的 4 个内参数 (f_x, f_y, c_x, c_y) 可以通过标定计算得到。

在理论上可定义一种完美的摄像机模型,但现实世界中没有完美的透镜,这主要是透镜在制造,以及整个摄像机组装过程中的偏差,从而引进了透镜畸变^[4],前者产生了径向畸变,后者产生了切向畸变。两个畸变的公式如下:

$$\begin{aligned} x_{\text{corrected}} &= x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y_{\text{corrected}} &= y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{aligned} \quad (5)$$

$$\begin{aligned} x_{\text{corrected}} &= x + [2p_1 y + p_2(r^2 + 2x^2)] \\ y_{\text{corrected}} &= y + [2p_1 x + p_2(r^2 + 2y^2)] \end{aligned} \quad (6)$$

式中: $r^2 = x^2 + y^2$;这两个畸变放置到一个畸变向量中,则是一个 5×1 的矩阵,按顺序依次包含 k_1, k_2, p_1, p_2 和 k_3 (针对鱼眼)。

2 基于 OpenCV 的摄像机标定

摄像机采用 Canon PowerShot SX130 IS 摄像机分辨率:180 × 180 dpi;输出画面分辨率:640 × 480 像素。

采用黑白棋盘作为标定模板,正方形边长为 190 mm,模板长宽个数为 9×9,角点数为 81 个。黑白棋盘模板如图 2 所示。

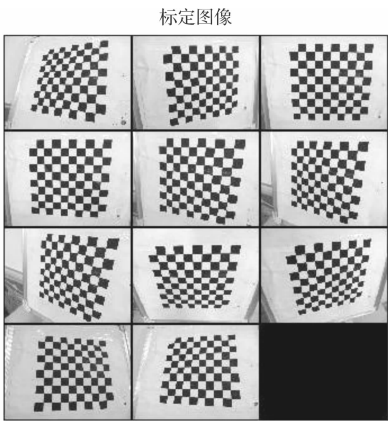


图 2 十一张黑白棋盘模板

本研究采用 Windows 7 操作系统和 Visual Studio 2010 软件平台,开发基于 OpenCV 的标定程序。

基于 OpenCV 的摄像机标定^[5-6]流程如下:
(1)读取设置。读取文件中的 XML 文件,该文件中设置了标定模板的各项参数,以及输出文件等;在读入文件中判断模板类型。

(2)通过寻找一些特征点来判断读入文件中的模板类型是棋盘、圆筒还是非对称圆筒;

(3)用 cvFindChessboardCorners() 函数在读入文件中寻找棋盘角点。先通过 cvtColor() 函数将图像转换为灰度图像,然后可使用提高精确度的 cornerSubPix() 函数,进一步准确的坐标的位置。寻找角点结果如图 3 所示;

(4)标定与保存。在 runCalibrationAndSave() 函数中包含摄像机标定 cvCalibrationCamera2() 函数和相关数据存储 saveCameraParams() 函数;

(5)输出结果 XML 文件。结果包含摄像机内外参数、每个图像的旋转向量和移动向量,以及每个角点的坐标。实验结果如表 1 所示。

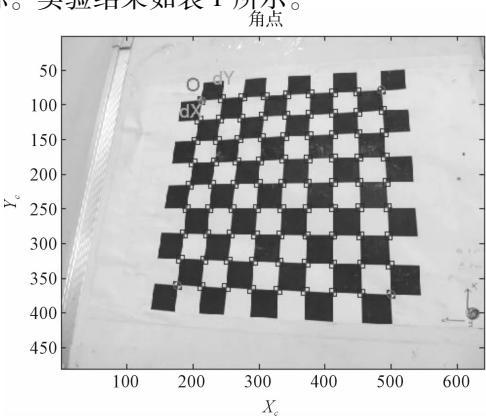


图 3 寻找角点结果

表 1 标定结果

序列	旋转向量 r	移动向量 T	相机坐标 P_c
1	-1.987 590	-63.379 001	154.024 2
	-2.145 422	-83.044 942	195.002 4
	0.803 607	310.138 445	212.955 0
2	2.092 874	-68.280 300	168.577 5
	2.140 006	-83.408 899	118.825 2
	-0.529 362	298.497 498	241.209 8
3	1.997 091	-60.397 320	146.959 5
	1.991 493	-77.248 578	14.515 8
	-0.094 943	264.543 792	240.397 2
4	2.213 140	-55.721 972	19.138 1
	2.062 617	-59.493 049	-13.109 5
	0.506 985	220.828 609	234.246 7
5	-2.117 658	-65.304 358	24.752 1
	-2.140 098	-68.252 447	64.376 5
	-0.200 468	230.442 562	239.311 1
6	-2.066 293	-76.502 968	26.734 6
	-2.116 207	-71.692 376	96.393 4
	-0.105 984	237.496 207	239.562 6
7	-1.905 486	-74.717 146	20.868 0
	-2.077 423	-75.044 711	138.870 9
	0.065 659	237.654 063	219.027 9
8	-1.792 009	-76.868 706	13.065 7
	-2.020 380	-78.549 984	183.718 5
	0.202 972	264.357 130	219.181 2
9	-1.609 285	-58.689 271	-2.095 5
	-1.970 805	-87.604 419	201.089 6
	0.273 454	261.948 557	198.228 6
10	-1.887 935	-86.1927 03	-78.354 1
	-1.882 903	-43.106 371	91.631 7
	-0.533 480	211.926 527	199.155 5
11	-2.117 303	-61.331 032	-63.319 4
	-1.923 208	-39.452 621	6.606 7
	-0.810 662	200.176 391	203.311 6

3 运动轨迹计算

3.1 计算方法综述

本研究对摄像机所得到的标定模板的每一副图像,均可在摄像机坐标系上用旋转与平移来描述物体运动的相对位置。其中,旋转可表示为坐标向量与适当尺寸方阵的乘积,从一个旋转等价于另一个不同坐标系下的对点位置的重新描述。在空间中,旋转可分解为绕各自坐标轴的二维旋转,而轴线度量保持不变。平移可表示将一个坐标系下的点移动到另一个坐标系的点的过程,即两个坐标系下的点的偏移量。因此,从世界坐标系的原点移动到以摄像机坐标的原点,相应的平移向量为 $T = \text{世界坐标系原点} - \text{摄像机坐标系原点}$ 。从而点在世界坐标系中的点 P 到摄像机坐标系

中的点 P_c 的公式^[7]如下:

$$P_c = R(P - T) \quad (7)$$

移动机器人运动轨迹的算法流程^[8-10]如图 4 所示。

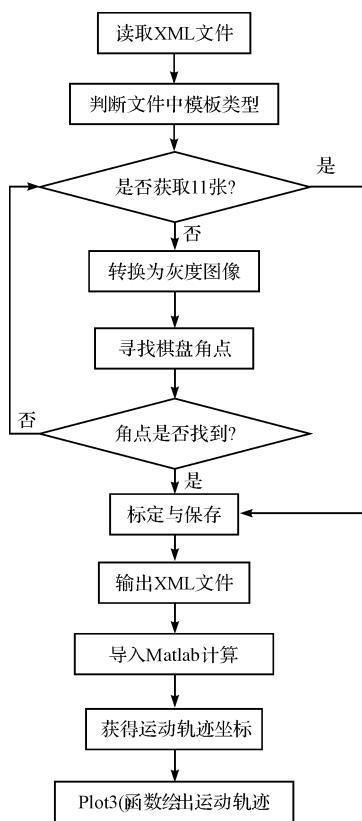


图 4 移动机器人运动轨迹的算法流程

实验结果得到的是旋转向量 $\mathbf{r}$ 和移动向量 $\mathbf{T}$, 其中需通过罗德里格斯变换, 在 OpenCV 中的 `cvRodrigues2()` 函数, 将 3×1 的 $\mathbf{r}$ 变换为 3×3 的旋转矩阵 $\mathbf{R}_1$, 并对 $\mathbf{R}_1$ 进行转置运算获得旋转矩阵 $\mathbf{R}$ 。在世界坐标系中进行公式(7)的计算, 则相对应得点 P 为 3×1 的零向量, 并代入旋转矩阵 $\mathbf{R}$ 与移动向量 $\mathbf{T}$, 通过 Matlab 的矩阵计算^[11]的到点 P_c 的坐标值, 即运动轨迹的坐标值。

以移动机器人运动在位置 6 作为示例, 经过基于 OpenCV 的摄像机标定后, 得到点 6 的旋转向量 $\mathbf{r}$ 和移动向量 $\mathbf{T}$ 。

$$\mathbf{r} = [-2.066\ 293, -2.116\ 207, -0.105\ 984]^T$$

$$\mathbf{T} = [-76.502\ 968, -71.692\ 376, 237.496\ 207]^T$$

通过罗德里格斯变换, 将 $\mathbf{r}$ 转换为 $\mathbf{R}_1$, 并对 $\mathbf{R}_1$ 求转置, 获得旋转矩阵 $\mathbf{R}$ 。

$$\mathbf{R} = \mathbf{R}_1^T = (\text{cvRodrigues2}(\mathbf{r}))^T = \begin{bmatrix} 0.016\ 6 & 0.983\ 7 & 0.179\ 0 \\ 0.996\ 7 & 0.030\ 6 & -0.075\ 6 \\ -0.079\ 8 & 0.177\ 2 & -0.980\ 9 \end{bmatrix}$$

将获得的旋转矩阵 $\mathbf{R}$ 与移动向量 $\mathbf{T}$, 以及 P 值

(3×1 的零向量) 代入公式(7), 得到机器人运动到点 6 的坐标 $P_c(6)$:

$$P_c(6) = [26.734\ 6 \quad 96.393\ 4 \quad 239.562\ 6]^T$$

移动机器人每拍摄一张图像, 图像同时也记录了机器人所运动轨迹的一个坐标点。示例中点 6 的坐标值便是如此, 同理所示, 依次计算出另外 10 个图像点, 从而获得移动机器人运动轨迹的所有坐标值。

3.2 实验及结果分析

实验使用标定结果中的旋转向量 $\mathbf{r}$ 和移动向量 $\mathbf{T}$, 计算获得运动轨迹的所有坐标值, 再运用 Matlab 的 `plot3()` 函数, 准确形象地将移动机器人的运动轨迹表示出来, 移动机器人运动轨迹如图 5 所示。

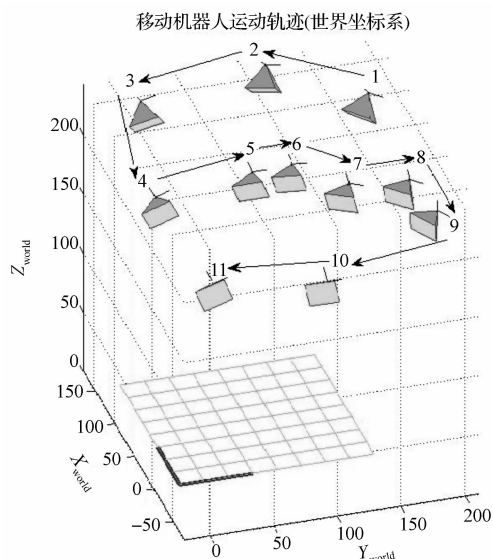


图 5 移动机器人运动轨迹

其中, 相机运动坐标如表 1 所示, 获得数据过程严格按照计算方法综述中的各项公式, 并在 Matlab 中实现。整个空间坐标系是以世界坐标系为基础, 蓝色栅格表示标定棋盘模块, 红色直角线表示世界坐标系的原点, 从 x 轴和 y 轴的范围 ($-50 \sim 0$) 可以得知, 有部分坐标超出了标定棋盘所在表面面向范围, 从表 1 的 P_c 列可找到点 4、9、10 和 11 便是超出范围的坐标点。从黑色箭头, 可以看出移动机器人的运动轨迹与点 1 依次到点 11 的顺序。每个点上青色锥形的黄色朝向表示机器人在该点所拍摄的镜头方向。从移动机器人运动轨迹图来看, 机器人从点 1 运动到点 11, 转向两次, 分 3 层拍摄标定棋盘模块, 最上层拍摄 3 张, 中间 6 张, 最低层拍摄 2 张。

4 结束语

本研究就移动机器人在摄像机标定过程中所行驶

的运动轨迹计算问题进行了研究,利用基于 SFM 思想,在 VC++ 环境下通过 OpenCV 的摄像机标定技术,以及 Matlab 软件,从标定中获到各项数据,从数据中计算出移动机器人运动的坐标,并在三维坐标系中同表现结果。本研究的实验中利用 11 张图像便能得到机器人运动轨迹。当机器人拍摄图像数目越多,以及质量越高时,更能得到详细完整的运动轨迹,同时标明机器人拍摄时间,可以得到一系列机器人的时间-位移图等等。这些设想与延展可以在进一步研究中实验与分析。

本研究通过整体叙述,较为完整与准确地得到了机器人的运动轨迹,并已以实验加以验证与解析,可以为一般类似问题研究及其算法提供启发与参考。

参考文献 (References):

- [1] EMAMI S, MAHMOOD N, et al. Mastering OpenCV with Practical Computer Vision Projects[M]. UK: packt Publishing Ltd., 2012.
- [2] WU Chang-chang. VisualSFM: A Visual Structure from Motion System [EB/OL]. [2011-01-01]. <http://www.cs.washington.edu/homes/ccwu/vsfm/index.html>, 2011.

- [3] HARTLEY R, ZISSERMAN A. Multiple View Geometry in computer vision [M]. Cambridge: Cambridge University Press, 2003.
- [4] 梅迎春,王朝立,廖勤武. 基于 OpenCV 的摄像机标定的应用研究[J]. 计算机工程与设计, 2009, 30(16): 3856-3862.
- [5] OPENCV DEV TEAM. Camera calibration With OpenCV [EB/OL]. [2015-02-15]. http://docs.opencv.org/doc/tutorials/calib3d/camera_calibration/camera_calibration.html.
- [6] 中科院自动化所自由软件协会. Cv 照相机定标和三维重建[EB/OL]. [2011-11-30]. http://wiki.opencv.org.cn/index.php/Cv_照相机定标和三维重建#FindFundamentalMat.
- [7] BRADSI G, KAEHLER A. Learning OpenCV[M]. USA: O'Reilly Media, Inc., 2008.
- [8] 岳龙旺,朱敬花. 基于轮式移动平台的新型湿式清洁机器人系统[J]. 轻工机械, 2013, 31(6): 66-68.
- [9] 毛剑飞,邹细勇,诸 静. 改进的平面模板两步标定摄像机[J]. 中国图像图形学报, 2004, 9(7): 846-852.
- [10] 景彦哲,刘 越,田 鸿,等. 适用于 SFM 点云的未标定摄像机注册方法[J]. 信号处理, 2013, 29(2): 274-278.
- [11] 董 霖. Matlab 使用详解[M]. 北京: 科学出版社, 2008.

[编辑:张 豪]

本文引用格式:

陈 健,李晓明. 基于 SFM 的移动机器人运动轨迹的计算方法研究[J]. 机电工程, 2015, 32(11): 1524-1528.

CHEN Jian, LI Xiao-ming, SU Bi-da, et al. Calculation method of motion trajectory for mobile robot based on SFM[J]. Journal of Mechanical & Electrical Engineering, 2015, 32(11): 1524-1528.

《机电工程》杂志: <http://www.meem.com.cn>

(上接第 1523 页)

参考文献 (References):

- [1] HORI Y. Future vehicle driven by electricity and control—research on four-wheel-motored “UOT electric march II” [J]. TRANSACTIONS ON INDUSTRIAL ELECTRONICS, 2004, 5(51): 954-962.
- [2] NAM K, FUJIMOTO H, HORI Y. Lateral stability control of in-wheel-motor-driven electric vehicles based [J]. TRANSACTIONS ON VEHICULAR TECHNOLOGY, 2012, 5(61): 1972-1985.
- [3] 丁荣军. 快速控制原型技术的发展现状[J]. 机电传动, 2009(4): 1-3, 15.
- [4] 贺 萍,董铸荣,韩承伟,等. 纯电动汽车综合性能试验台的国内外现状与改进[J]. 机电工程, 2014(9): 1231-1234, 1238.
- [5] 于金泳,刘志远. 基于快速控制原型的 ABS 控制器开发

[J]. 汽车技术, 2006(12): 20-24.

- [6] 沈悦明,陈启军. dSPACE 快速控制原型在机器人控制中的应用[J]. 机器人, 2002(6): 545-549.
- [7] 邹 渊,孙逢春,张承宁. 电传动履带车辆双侧驱动快速控制原型开发[J]. 北京理工大学学报, 2007(1): 29-33.
- [8] 杨向忠,安锦文,崔文革. 快速控制原型仿真技术应用[J]. 航天控制, 2009(2): 72-75, 80.
- [9] 吴志军. 多轮独立驱动电动汽车驱动控制系统设计[D]. 杭州: 浙江大学机电工程学院, 2014.
- [10] 邱斌斌,朱绍鹏,马浩军,等. 电动车辆驱动控制系统仿真测试平台设计[J]. 浙江大学学报: 工学版, 2015, 49(2): 1-6.
- [11] 朱绍鹏,邱斌斌,吴志军,等. 两轮独立驱动电动汽车的驱动力控制策略研究[C]. 2013 中国汽车工程学会年会论文集, 2013: 1-11.

[编辑:张 豪]