

Исследование позиционирования цели для всенаправленного мобильного манипулятора с 3 степенями свободы на основе машинного зрения

Яо Ду

Пекинский транспортный университет, Пекин, Китай
Петербургский государственный университет путей сообщения Императора Александра I,
Санкт-Петербург, Россия
15121222@bjtu.edu.cn

Цзиву Ван, Вэньшэн Сюй

Механический, электронный и контрольный факультет,
Пекинский транспортный университет, Пекин, Китай

Сугишака Масанори

Alife Robotics Corporation Ltd, Япония
и Открытый университет, Великобритания
15121244@bjtu.edu.cn*

Аннотация. Рассматривается всенаправленный мобильный робот с тремя степенями свободы, для которого обеспечивается улучшение контроля операционной позиции и точности работы манипулятора в узком или переполненном рабочем месте по сравнению с общим манипулятором. Решаемая задача заключается в улучшении гибкости робота при работе с несколькими разными объектами. При этом эффективным решением является распознавание цели с обработкой изображений. На основе обработки изображения можно определить положение и позу цели. Затем сигнал отправляется в систему управления манипулятором. В статье изучаются условия освещения, искажения и т. д. При распознавании цели с помощью обработки изображений ее позиция сверяется с реальными координатами. Эксперименты показывают, что распознавание целей с помощью обработки изображений позволяет эффективно улучшить гибкость работы робота.

Ключевые слова: мобильный робот, машинное зрение, манипулятор с тремя степенями свободы.

ВВЕДЕНИЕ

Сегодня степень автоматизации заводов и складов становится все выше [1]. Однако трудно расширить функции манипулятора, поскольку его база фиксированная. В этой статье описывается всенаправленный мобильный манипулятор с тремя степенями свободы, который может повысить гибкость манипулятора и значительно увеличить его функциональность.

Сначала через анализ манипулятора с тремя степенями свободы устанавливается модель кинематики манипулятора методом Д-Н (Denavit-Hartenberg), на основе этой модели выводится уравнение кинематики манипулятора. Изучен метод решения обратной кинематической задачи манипулятора. Получено аналитическое решение обратной кинематической задачи манипулятора.

Затем проверяются и анализируются аналитические решения уравнений прямой кинематики и обратных решений. Экспериментальные результаты доказывают эффективность

метода получения обратного решения манипулятора и метода целевого местоположения.

СИСТЕМНЫЙ ОБЗОР

Система управления роботом очень важна при реализации автоматического управления им. Рассматриваемый в статье робот использует четыре колеса Mecanum для обеспечения всенаправленного движения платформы робота. В составе системы управления роботом есть четыре части: ПС (персональный компьютер), главная панель управления, драйверы шасси и манипулятор (рис. 1).

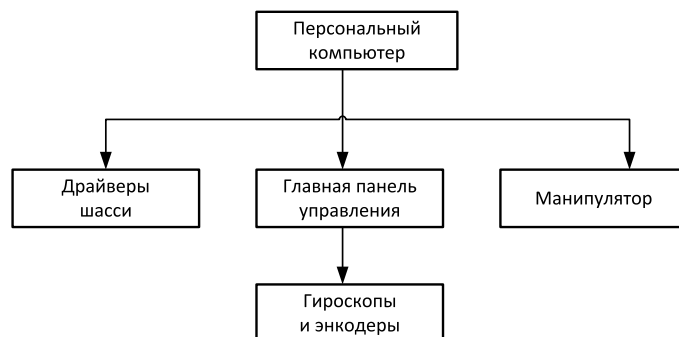


Рис. 1. Схема управления роботом

ПС запускает операционную систему Windows через последовательный порт для установления связи с основной панелью управления. Основная плата управления связывается с двумя управляющими платами привода через шину CAN.

ДИЗАЙН ВСЕНАПРАВЛЕННОГО МОБИЛЬНОГО ШАССИ НА ОСНОВЕ КОЛЕСА MECANUM

Четыре колеса Mecanum установлены для всенаправленного движения платформы (рис. 2) [2].

* Опубликовано: ISBN 978-4-9908350-2-6: Jiwu Wang, Yao Du, Wensheng Xu, Sugisaka Masanori. Study On The Target Positioning For An Omni-Directional 3DOF Mobile Manipulator Based On Machine Vision. The 2017 International Conference on Artificial Life and Robotics (ICAROB2017), Seagaia Convention Center, Miyazaki, Japan, January 19-22, 2017. (на английском).



Рис. 2. Колесо Mecanum

На рис. 3 представлены четыре варианта установки Mecanum. Каждое колесо приводится в движение только двигателем щетки постоянного тока с левым и правым вращением.

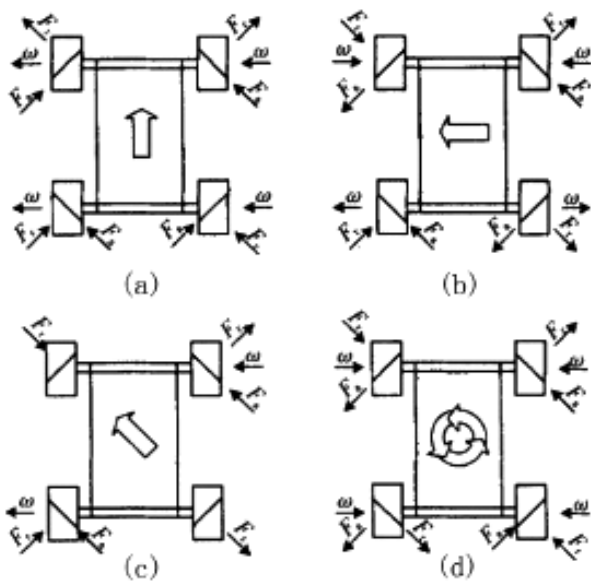


Рис. 3. Варианты установки шасси

На рис. 4 показана система координат модели шасси с четырьмя колесами Mecanum. Система координат $O_i X_i Y_i$ ($i = 1, 2, 3, 4$) является центром каждого колеса, а исходная система координат – центром положения робота.

Как показано на рис. 4, длина тела робота составляет $2L$, ширина тела $2l$, скорость шасси робота $\vec{V} = (V_x, V_y)$, угловая скорость робота вокруг оси Z равна ω_z , линейная скорость колеса $\vec{V}_{iw} = \omega_i R$, где ω_i – угловая скорость каждого колеса; R – радиус колеса. Ролик и контакт заземления, когда скорость линии \vec{V}_{ir} . Предположим, что скорость каждого колеса $\vec{V}_i = (V_{ix}, V_{iy})$. Под $O_i X_i Y_i$ ($i = 1, 2, 3, 4$) можно разложить

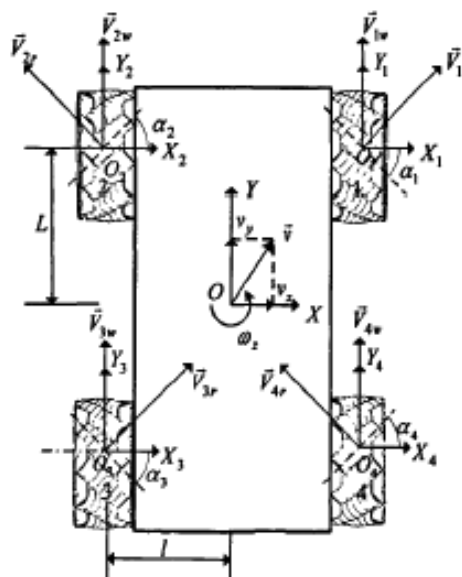


Рис. 4. Моделирование координат всенаправленного мобильного робота

вдоль X_i и Y_i , так как \vec{V}_{iw} и \vec{V}_{ir} , а V_{ix} можно выразить как $-V_{ir} \sin \alpha_i$, V_{iy} – сумма \vec{V}_{iw} и $\vec{V}_{ir} \cos \alpha_i$, которые удовлетворяют соотношениям

$$\begin{aligned} V_{1x} &= -V_{1r} \sin \alpha_1, V_{1y} = -V_{1w} + V_{1r} \cos \alpha_1; \\ V_{2x} &= -V_{2r} \sin \alpha_2, V_{2y} = -V_{2w} + V_{2r} \cos \alpha_2; \\ V_{3x} &= -V_{3r} \sin \alpha_3, V_{3y} = -V_{3w} + V_{3r} \cos \alpha_3; \\ V_{4x} &= -V_{4r} \sin \alpha_4, V_{4y} = -V_{4w} + V_{4r} \cos \alpha_4. \end{aligned}$$

В формуле α_i – круглая ось ролика и угол оси ступицы ($i = 1, 2, 3, 4$).

В целом скорость каждого колеса \vec{V}_i – это сумма скорости робота и относительного центра колеса. То есть $V = \vec{v}_i + \vec{\omega}_z \cdot OO_1$ рассматривают как векторную направленность вдоль координат XY и $X_i Y_i$. Можно получить формулу декомпозиции координат:

$$\begin{aligned} V_{1x} &= V_x - L\omega_z, V_{1y} = V_y + l\omega_z; \\ V_{2x} &= V_x - L\omega_z, V_{2y} = V_y - l\omega_z; \\ V_{3x} &= -V_x + L\omega_z, V_{3y} = V_y - l\omega_z; \\ V_{4x} &= V_x + L\omega_z, V_{4y} = V_y + l\omega_z. \end{aligned}$$

Принимая во внимание то, что скорость ролика не контролируется, и исключив её, получим уравнение всенаправленной обратной кинематики робота:

$$\begin{aligned} W_1 &= \frac{1}{R} [-V_x \cot \alpha + V_y + (l + L \cot \alpha) \omega_z]; \\ W_2 &= \frac{1}{R} [V_x \cot \alpha + V_y - (l + L \cot \alpha) \omega_z]; \\ W_3 &= \frac{1}{R} [-V_x \cot \alpha + V_y - (l + L \cot \alpha) \omega_z]; \\ W_4 &= \frac{1}{R} [-V_x \cot \alpha + V_y + (l + L \cot \alpha) \omega_z]. \end{aligned}$$

Это уравнение отображает способ управления роботом в целом.

ПРОЕКТИРОВАНИЕ МНОГОСТУПЕНЧАТОЙ СВОБОДЫ ЗАХВАТА МАНИПУЛЯТОРА

Конструкция рычага робота в сумме трех степеней свободы: направление по оси X, два шатуна – это оси Y и Z.

Управление движением ручки манипулятора линейной интерполяции

Расчет показывает, что есть взаимосвязь между позицией сустава робота и позицией руки:

$$\theta_2 = \arccos \frac{p_y^2 + p_z^2 - l_1^2 - l_2^2}{2l_1 l_2};$$

$$\theta_1 = \arccos \frac{p_y(l_1 + l_2 \cos \theta_2) - p_z l_2 \sin \theta_2}{(l_1 + l_2 \cos \theta_2)^2 + l_2^2 \sin^2 \theta_2}.$$

Для положения пространства руки робота, когда она перемещается от (y_1, z_1) до (y_2, z_2) по прямой линии, линия интерполируется. Из-за простой интерполяции по оси X или оси Y много неравномерности.

Предполагая, что наклон линии k , когда $y_1 \neq y_2$, $k = (z_2 - z_1)/(y_2 - y_1)$, угол наклона прямой:

$$\alpha = \arctan k.$$

Предполагая, что d – расстояние от любой точки (y, z) до (y_1, z_1) на этом отрезке, точка, соответствующая $d(y, z)$, равна

$$\begin{cases} y = y_1 + d \cos \alpha \\ z = z_1 + d \sin \alpha \end{cases},$$

где d находится в диапазоне $[0, \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}]$. Когда d равномерно изменяется в этом интервале, движение руки изменяется равномерно. Полученное описанным уравнением (y, z) подставляется в выражение позиции сустава, чтобы реализовать прямолинейное движение руки.

Управление скоростью руки манипулятора робота

Соотношением между пространственным положением руки робота и переменной сустава является соотношение

$$\begin{cases} p_y = l_2 c_1 c_2 - l_2 s_1 s_2 + l_1 c_1 \\ p_z = l_2 c_2 s_1 - l_2 c_1 s_2 + l_1 s_1 \end{cases}.$$

Дифференцированные производные из предыдущей формулы могут быть представлены в виде

$$\begin{bmatrix} dp_y \\ dp_z \end{bmatrix} = \begin{bmatrix} -l_2 s_{12} - l_1 s_1 & l_2 c_1 c_2 \\ l_2 c_{12} + l_1 c_1 & l_2 c_{12} \end{bmatrix}.$$

В матричной форме

$$\left\{ \begin{aligned} dp_y &= \frac{\partial y}{\partial \theta_1} d\theta_1 + \frac{\partial y}{\partial \theta_2} d\theta_2 = (-l_2 s_{12} - l_1 s_1) d\theta_1 + (-l_2 s_{12}) d\theta_2 \\ dp_z &= \frac{\partial z}{\partial \theta_1} d\theta_1 + \frac{\partial z}{\partial \theta_2} d\theta_2 = (l_2 s_{12} - l_1 c_1) d\theta_1 + (-l_2 c_{12}) d\theta_2 \end{aligned} \right\}.$$

Как видно из определения матрицы Якоби,

$$\dot{p} = J \dot{q}.$$

Матрица Якоби для манипулятора может быть получена по формуле

$$J = \begin{bmatrix} -l_2 s_{12} - l_1 s_1 & l_2 c_1 c_2 \\ l_2 c_{12} + l_1 c_1 & l_2 c_{12} \end{bmatrix}.$$

Якобианская матрица в основном используется для описания дифференциального отношения смещения сустава и пространственного положения, являющееся соотношением между пространственной скоростью руки робота и угловой скоростью сустава:

$$v = J \dot{\theta}.$$

И наоборот: скорость движения каждого сустава может быть определена с помощью желаемой скорости руки:

$$\dot{\theta} = J^{-1} v.$$

Согласно расчетам,

$$J^{-1} = \begin{bmatrix} -\frac{c_{12}}{l_1 c_{12} s_1} - l_1 s_{12} c_1 & -\frac{s_{12}}{l_1 c_{12} s_{12}} - l_1 s_{12} c_1 \\ \frac{l_2 c_{12} l_1 c_1}{l_1 l_2 c_{12} s_1} - l_1 l_2 s_{12} c_1 & \frac{l_2 s_{12} + l_1 s_1}{l_1 l_2 c_{12} s_1} - l_1 l_2 s_{12} c_1 \end{bmatrix}.$$

ДИЗАЙН ОБРАБОТКИ ИЗОБРАЖЕНИЙ

Согласование шаблонов – это поиск целевого изображения [3]. Для этого необходимо измерить сходство между двумя изображениями по некоторым критериям подобия.

Согласование шаблонов на основе серого значения

Когда шаблон скользит по изображению, сопоставим подобия. Простейшей мерой сходства является вычисление суммы абсолютных значений различий между шаблоном и изображением или суммой квадратов всех различий (SAD и SSD).

$$sad(r, c) = \frac{1}{n} \sum_{(u,v) \in T} |t(u, v) - f(r + u, c + v)|;$$

$$sad(r, c) = \frac{1}{n} \sum_{(u,v) \in T} x |t(u, v) - f(r + u, c + v)|^2,$$

где $f(r + u, c + v)$ – серое значение строки $r + u$ столбца $c + v$ изображения, которое должно быть измерено; $t(r, c)$ – серое значение шаблона изображения T строки r столбца c

[4]. Когда свет меняется, значение серого становится больше или меньше. Независимо от этого разница между значениями после изменения и их средним значением мала, затем они станут нулевыми средними и единичными дисперсиями, разделив дисперсию так, чтобы изображение было менее чувствительным к изменению яркости уровня и обладало сильной устойчивостью.

$$ncc(r, c) = \frac{1}{n} \sum_{(u, v) \in T} \frac{t(u, v) - m_c}{\sqrt{s_t^2}} \cdot \frac{f(r + u, c + v) - m_f(r, c)}{\sqrt{s_f^2(r, c)}}$$

где m_c – среднее значение серого шаблона; s_t^2 – дисперсия всех значений серого измеряемого изображения, (r, c) – среднее значение серого измеряемого изображения.

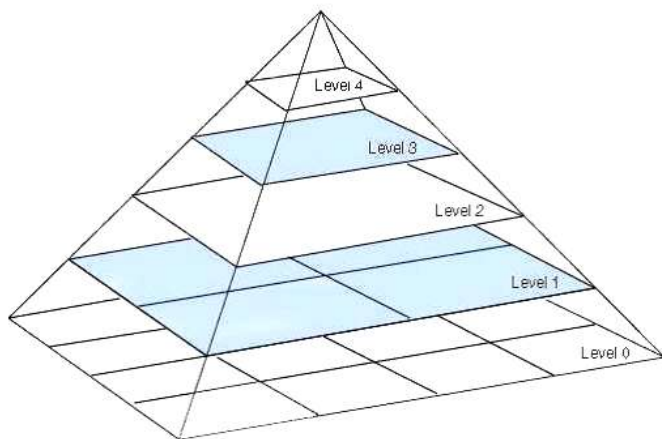


Рис. 5. Схематическая пирамида изображения

Как показано на рис. 5, стратегия поиска на основе пирамиды такова: от верхней к нижней части поиска в поиске высокого уровня изображения для отслеживания экземпляра шаблона в нижней части пирамиды изображения, затем в небольшой области досопоставить в следующей области поиска, чтобы найти результаты сопоставления небольшой области, на которой рассчитываются степень подобия, которая сегментируется порогом, и локальные экстремумы.

Реализация обхода

Чтобы обеспечить автоматический обход роботов, мы должны получить фактическое местоположение предметов в мировой системе координат.

Как показано на рис. 6, сначала должны быть исходная коррекция искажения изображения и сглаживание шума; затем извлеките изображение компонента S насыщения из пространства изображения HSV, мы можем ясно видеть, что насыщающая компонента объекта, который мы хотим обойти, относительно высока; третий шаг – получить область изображения с высокой насыщенностью с помощью метода динамического порога. Последний шаг – отобразить полученную область и получить информацию об ориентации области и информацию об ориентации бутылки.

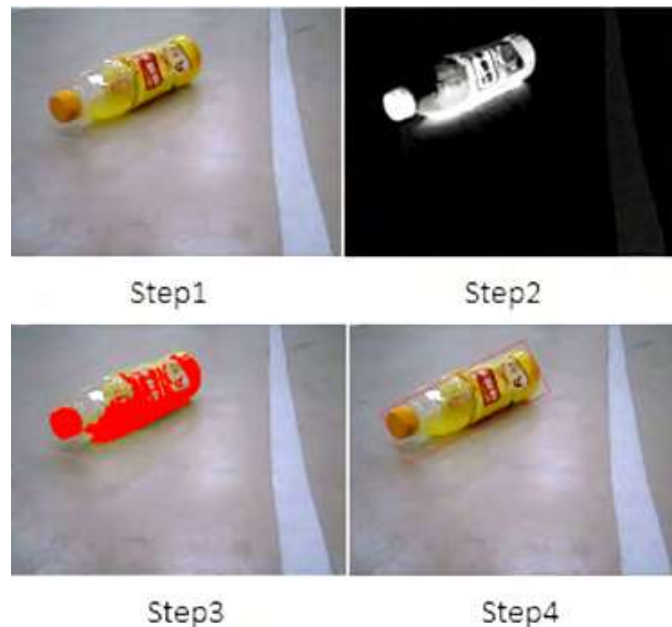


Рис. 6. Шаги по поиску предметов

ЗАКЛЮЧЕНИЕ

В статье представлен всенаправленный мобильный манипулятор с тремя степенями свободы, основанный на машинном зрении. Однако в реальных экспериментах робот контролировался, чтобы схватить бутылку. Наши эксперименты показали, что относительное желаемое положение и ориентация между камерой и объектом поддерживаются в пределах нескольких сантиметров. Следующий шаг заключается в дальнейшем улучшении способа обработки изображений, чтобы ускорить обработку изображения, чтобы робот более точно понимал окружающую среду. Добавление некоторых датчиков, таких как лидар, дает роботу возможность планировать путь и предотвращать столкновения с препятствиями. Надо увеличить подвесную систему шасси, чтобы адаптироваться к более неровной местности.

ЛИТЕРАТУРА

1. Birmal Singh N. Vision Based Autonomous Navigation of Mobile Robots. – India, Kolkata, Jadavpur Univ., 2010.
2. Shi J., Tomasi C. Good features to track. Computer Vision and Pattern Recognition.
3. Tomasi C., Kanade T. Detection and tracking of point features // Tech. Rept. CMU-CS-91132.
4. Davison A. J. Real-time simultaneous localisation and mapping with a single camera // Computer Vision, IEEE. 2003. P. 1403-1410.
5. Civera J., Davison A. J., Montiel J. Inverse depth parametrization for monocular SLAM. IEEE Trans. Rob. 2008. № 24 (5). P. 932-945.
6. Civera J., Grasa O. G., Davison A. J., et al. L-Point RANSAC for extended Kalman filtering: Application to real-time structure from motion and visual odometry // J. Field Rob. 2010. № 7 (5). P. 609-631.
7. Yilmaz O. Javed, Shah M. Object tracking // A survey ACM Comput. Surv. 2006. Vol. 38, № 4. P. 13.

8. Edgar R. Arce-Santana, Jose M. Luna-Rivera, Daniel U. Campos-Delgado, and Ulises Pineda-Rico. Real-Time Vision Tracking Algorithm // IJCV, 2006.

9. Neira J., Beiro M.1., Tardos J.D. Mobile Robot Localisation and MaP Building Using Monoocular Vision. In Proc. Int. 1 Symp. on Intell. Rob. Systems, 1997. P. 1-10.

10. Friedman J.K., Bentley J.L., Finkel R.A. An algorithm for finding best matches in logarithmic expected time // ACM Trans. on Math. Software. 1977. № 3. P. 209-226.

11. Arya S., Mount D.M., Netanyahu N.S., Silverman R., Wu A. An optimal algorithm for approximate nearest neighbor searching // ACM. 1998. № 45. P. 891-923.

12. Zhang Z. A flexible new technique for camera calibration. IEEE Trans. Pattern Anal. Mach. Intell. 2000. № 22 (11). P. 1330-1334.

13. Bouguet J.-Y. Camera Calibration Toolbox for Matlab. URL: http://www.vision.caltech.edu/bouguetj/calib_doc.

14. Intel OpenCV Computer Vision Library (C++). URL: <http://www.intel.com/research/mrl/research/opencv>.

15. Fischler M.A., Bolles R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography // Comm. of the ACM. 1981. Vol. 24. P. 381-395.

16. Rafael C. Gonzalez, Richard E. Woods. Digital image processing. Addison Wesley Publ. Company, 1992.

17. Sim R. Mobile Robot Localization Using Learned Landmarks // J. NY Entomol. Soc. 1998. № 4. P. 166-169.

18. Nayar S. K., Murase H., Nene S. A. Learning, positioning, and tracking visual appearance // IEEE Int. Conf. Rob. Autom. Proc. IEEE. 1994. Vol. 4. P. 3237-3244.

19. Nikolić N. Mobile Robot Localization Using Computer Vision and Artificial Landmarks. 2008.

20. Murphy R. R. Sensor and information fusion for improved vision-based vehicle guidance // IEEE Expert. 1998. № 13 (6). P. 49-56.

Study On The Target Positioning For An Omni-Directional 3 DOF Mobile Manipulator Based On Machine Vision

Yao Du

School of Mechanical, Electronic and Control Engineering,
Beijing Jiaotong University
Beijing, China,
Emperor I Petersburg State Transport University,
St. Petersburg, Russia
15121222@bjtu.edu.cn

Jiwu Wang, WenSheng Xu

School of Mechanical, Electronic and Control Engineering,
Beijing Jiaotong University
Beijing, China

Sugisaka Masanori

Alife Robotics Corporation Ltd, Japan
Open University, United Kingdom
15121244@bjtu.edu.cn

Abstract. The omni-directional mobile robot with multi DOF, because the operation posture and operation accuracy of the manipulator can be better controlled in a narrow or crowded workplace compared with the general manipulator, is getting more interested in practical applications. The present problem is to improve its flexibility for operating multiple different targets. Target recognition with image processing is an effective solution. Based on the image processing, the position and posture of the target can be determined. Then the signal will be sent to the arm control system. In this paper, the illumination conditions, distortion, etc. are studied in the target recognition. The target position with image processing, is verified with real coordinates. The experiments show target recognition with image processing can effectively improve the flexibility of our robot.

Keywords: mobile robot, machine vision, 3 DOF manipulator, omni-direction.

REFERENCES

1. Singh B. N. Vision Based Autonomous Navigation of Mobile Robots. Jadavpur Univ., Kolkata, India, 2010.
2. Shi J., Tomasi C. Good features to track, *Computer Vision and Pattern Rec.*
3. Tomasi C., Kanade T. Detection and tracking of point features, *Tech. Rept. CMU-CS-91132*.
4. Davison A.J. Real-time simultaneous localisation and mapping with a single camera. *Comp. Vision, IEEE*, 2003, pp. 1403-1410.
5. Civera J., Davison A. J., Montiel J. Inverse depth parametrization for monocular SLAM. *IEEE Trans. Rob.*, 2008, no. 24 (5), pp. 932-945.
6. Civera J., Grasa O. G., Davison A. J., et al. 1-Point RANSAC for extended Kalman filtering. Application to real-time structure from motion and visual odometry. *J. Field Rob.*, 2010, no. 27 (5), pp. 609-631.
7. Javed Yi. O., Shah M. Object tracking. *A survey, ACM Comput. Surv.*, 2006, vol. 38, no. 4, p. 13.
8. Arce-Santana E. R., Luna-Rivera J. M., Campos-Delgado D. U., Pineda-Rico U. Real-Time Vision Tracking Algorithm. *IJCV*, 2006.
9. Neira J., Beiro M.1., Tardos J.D. Mobile Robot Localisation and MaP Building Using Monoocular Vision. In Proc. Int. 1 Symp. on Intell. Rob. Systems, 1997, pp. 1-10.
10. Friedman J.K., Bentley J.L., Finkel R.A. An algorithm for finding best matches in logarithmic expected time, *ACM Trans. on Math. Software*, 1977, no. 3, pp. 209-226.
11. Arya S., Mount D.M., Netanyahu N.S., Silverman R., Wu A. An optimal algorithm for approximate nearest neighbor searching, *ACM*, 1998, no. 45, pp. 891-923.
12. Zhang Z. A flexible new technique for camera calibration, *IEEE Trans. Pattern Anal. Mach. Intell.*, 2000, no. 22 (11), pp. 1330-1334.
13. Bouguet J.-Y. Camera Calibration Toolbox for Matlab. Available at: http://www.vision.caltech.edu/bouguetj/calib_doc.
14. Intel OpenCV Computer Vision Library (C++). Available at: <http://www.intel.com/research/mrl/research/opencv>.
15. Fischler M.A., Bolles R. C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography, *Comm. of the ACM*, 1981, Vol. 24, pp. 381-395.
16. Gonzalez R. C., Woods R. E. Digital image processing. Addison Wesley Publishing Company, 1992.
17. Sim R. Mobile Robot Localization Using Learned Landmarks, *J. NY Entomol. Soc.*, 1998, vol. 4, pp. 166-169.
18. Nayar S. K., Murase H., Nene S. A. Learning, positioning, and tracking visual appearance, *IEEE Int. Conf. Rob. Automat.*, 1994. Proc. IEEE, 1994, vol. 4, pp. 3237-3244.
19. Nikolić N. Mobile Robot Localization Using Computer Vision and Artificial Landmarks. 2008.
20. Murphy R. R. Sensor and information fusion for improved vision-based vehicle guidance, *IEEE Expert*, 1998, vol. 13 (6), pp. 49-56.