Proceedings of 2016 IEEE
International Conference on Mechatronics and Automation
August 7 - 10, Harbin, China

Pedestrian Localization in Distributed Vision System for Mobile Robot Global Path Planning

Yue Guo, Yijia He, Feng Wen and Kui Yuan
Institute of Automation
Chinese Academy of Sciences
95 Zhongguancun East Road, Beijing, China
guoyue2013@ia.ac.cn

***Abstract*—In this paper, we propose a general framework of a distributed vision system combining pedestrian features with a static map of a mobile robot. In indoor environment with complex architectural structures, mobile robots cannot find the optimal global path only with the static map except moving objects such as pedestrians outside the view of sensors. Therefore, we propose a new calibration approach about the transformation from the pedestrian features in distributed webcams to those in the static map. Robot Operating System (ROS) is used for the mobile robot static map building, global path planning, localization and navigation with the laser scanner. And webcams scattered in indoor environment are available for the distributed vision calibration and pedestrian detection. We assume that light conditions of all the webcams remain relatively constant in indoor environment, then multi-scale Histograms of Oriented Gradients (HOG) slid on normalized sub images accelerates the pedestrian detection, and we also test the detection using convolutional neural networks (CNN). Next, detected features are calculated with calibration parameters and added on the static map. Finally, a real-time global path is planned for our mobile robot given the dynamic map.**

***Index Terms*—Pedestrian Localization, Distributed Vision Calibration, Mobile Robot, Global Path Planning.**

I. Introduction

With the increasing applications of the mobile robot technologies in recent years, the development of autonomous localization and navigation makes mobile robots more capable of working with humans in many occasions. Most of mobile robots carries sensors such as laser scanners, odometers and cameras for delivery tasks. But the measurement ranges of onboard sensors limit the mobile robot perception, so the mobile robot cannot acquire the global information from the indoor environment.

Webcams for intelligent monitoring exist in people's lives for years with the Internet and the cloud storage technology. The popularity of the Internet greatly reduces the costs of the high bandwidth network, and the topology transformation of the intelligent monitoring system from star structure to net one contributed to the cloud storage technology allows users to access video streams from any cameras shared on the cloud in network nodes covered anywhere. And webcams in public are usually placed in high-volume areas. Therefore, mobile robots can get most of dynamic global information in indoor environment to obtain temporal-spatial global path optimally.

II. Related Work

Researches about pedestrians and the mobile robots are studied in many interesting perspectives for decades, such as the pedestrian features, extraction from backgrounds, tracking and interaction with mobile robots.

Features are studied with diverse devices. The pairs of legs are tracked with the laser range finder on the mobile robot [1]. Then faces are also identified and forereached by the mobile robots with LIDAR [2]. Finally, whole bodies can be detected through Histograms of Oriented Gradients and linear Support Vector Machines (SVM) [3], [4].

Extraction from backgrounds, in other words, finding the pedestrian locations, also depends on what kind of devices a mobile robot carries. For example, a dense disparity map is generated with the stereo vision to crop the regions of interest (ROI) according to their depth and texture density [3], or pedestrian-similar areas are detected with a laser range finder before pedestrian detection [4]. Some methods even depend on the devices that pedestrians carry, for instance, a wireless localization network to localize the pedestrians who must attach small-size wireless localization tags [5].

Track of pedestrians also attracts many researchers in the mobile robot field with the methods including pedestrian ego-graph [6], global-nearest-neighbor [7], Kalman filter [3], [7], particle filter [4] and multiple hypothesis [8].

Interaction between humans and machines aims to find models or patterns of pedestrians in public. Hiroyuki Kidokoro et al. propose models of pedestrian flow, pedestrian interaction and walking comfort and their combination, studying the balance between the walking comfort and the task of the mobile robot with many three dimensional range sensors on ceilings everywhere in a shopping mall [9]. Sachit Butail proposes an agent-based social force model and concludes that rate-of-interaction increases with flow density, average-interaction-time is independent and interaction-time depend on interaction-speed [10]. Move and work patterns are classified through pedestrians' trajectories [11].

The following part of this paper is organized as follows:

978-1-5090-2396-7/16/$31.00 ©2016 IEEE

The architecture of the distributed vision system is described in section III. Then we introduce the calibration method and the pedestrian localization method respectively in section IV and V. Next, global path planning concerned with pedestrians based on ROS and experiment results are introduced in section VI and VII. Finally, some conclusions are given in section VIII.

III. System Architecture

The typical architecture of the distributed vision system is shown in Fig. 1. The mobile robot cannot detect the pedestrian due to the limit measurement scopes of on-board sensors, thus if its destination locates on the position of the green circle, and a global path (green dashed line) to the destination is generated according to the static map and the current information from the on-board sensors. However, the mobile robot does not know that the destination is unreachable until the on-board sensors detect the pedestrian. Instead, the mobile robot will know the pedestrian locations and plan a spatiotemporal optimal global path (the red solid line) if the information of the webcams are utilized.

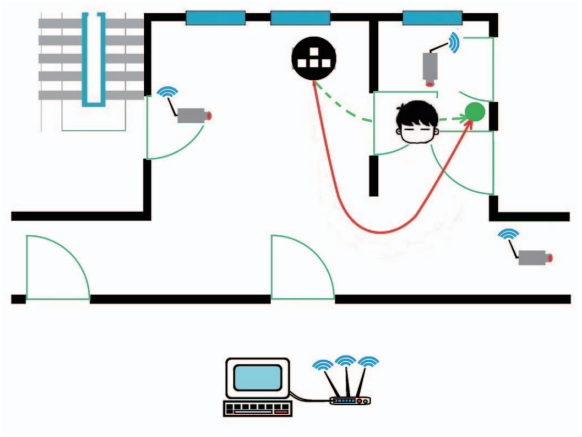

Fig. 1. Architecture of distributed vision system.

A. Hardware Architecture

The hardware of a distributed vision system consists of webcams, a wireless router, a server and a mobile robot.

1) Webcams: Every D-Link wireless monitoring module we use for experiment consists of a domestic infrared wireless webcam DCS-932L, a wireless broadband router DIR-605LW and two 5V/1.2A power adapters, so a unique local IP address will be allocated to each webcam, which image resolution is 640*480 pixels and the corresponding transmission speed can be manually set to 20 fps.

2) Wireless Router: A high bandwidth wireless central router acts more like a wireless hub collecting all the video streams from webcams.

3) Server: The server connects the central router with cables. But several WIFI signal amplifiers would be necessary if used in large scale indoor environment.

4) Mobile Robot: A mobile robot includes a mobile underpan module, a motor control module, a laser scanner URG-04LX-UG01, a blue-tooth module and a host computer. Motion information measured from the laser scanner and the odometry on the mobile robot are sent to the host computer through the blue-tooth module. Our host computer is responsible for building the global static map, receiving the global information about pedestrians, fusing them to the static map and periodically updating the global path.

B. Software Architecture

The software of a distributed vision system includes algorithms about static map construction, distributed vision calibration, pedestrian detection and localization, global map spatial fusion, mobile robot global path planning, mobile robot localization and navigation.

1) Static Map Construction: Remote control the mobile robot to move around the entire indoor environment and use gmapping [12] package in ROS [13] to build static maps.

2) Distributed Vision Calibration: After building the static map in indoor environment with low person flow, mobile robot can localize itself in indoor environment. Record 2D central positions of the mobile robot in the map coordinate system and the corresponding centers of the artificial marker in the active webcam coordinate system. Here the active webcam is the only camera that can see the mobile robot. Then find the correlations between these two centers.

3) Pedestrian Detection and Localization: In indoor environment, we assume webcams are fixed on the ceilings and the indoor light conditions are nearly constant, so background subtraction and morphological open operation can robustly crop the regions of dynamic objects at first. Next, pedestrians are classified discriminately by linear Support Vector Machine (SVM) with HOG descriptors [14] in all sub images including dynamic objects on normalized scale. Then bounding boxes of the detected pedestrians are recovered in different active image coordinate system. Here the active images correspond to the webcams that detect pedestrians. Eventually, pedestrian locations on the 2D map coordinate system are transformed from the bounding boxes in all webcams with calibration parameters.

4) Global Map Spatial Fusion: A self-defined pedestrian layer periodically updating pedestrian locations are plugged into the global costmap [15].

5) Mobile Robot Global Path Planning, Localization and Navigation: Dijkstra's algorithm [16] is used to calculate the global path planning, and Adaptive Monte Carlo Localization (ACML) [17] in ROS is used for mobile robot localization and navigation with the prior knowledge of static and fused global costmap.

IV. Calibration Method

The implicit spatial constraints among these webcams increase the difficulty of calculating the explicit relationships of coordinate system transformation with each other. In order to solve this, a new 2D map coordinate system is defined to unify the relationships among the coordinate systems of all the webcams, pedestrians and the mobile robot.

The variety of pedestrian heights makes the pedestrian localization hard, so a most reasonable height of the map coordinate system we think is the same as that of the ground plane where our system works in indoor environment. Because all the pedestrians and the mobile robot stand on the same ground plane, and the positions where pedestrians standing on the ground plane in the images are easier to obtain. But the precise localization of the mobile robot on the ground plane still depends on the on-board laser scanner instead of webcams.

The artificial marker placed on the mobile robot is detected in the active image, which corresponds to the only one webcam that can detect the artificial marker at most during the calibration. Then the corresponding marker center are calculated simultaneously. However, the direct calibration for the spatial relationship between the ground plane and the active image plane becomes infeasible, because the artificial marker and the ground plane are not on the same height. Thus the 3D central positions on the artificial marker plane have to be transformed to those on the ground plane.

A. Artificial Marker Pattern

In Fig. 2 (a), the artificial marker is a "T" symbol made up of four small square blocks, and the length of the side of each block is 10 cm. Three blocks compose the horizontal part of "T", while the middle block of the three above and another block form the vertical part. The central position of the artificial marker is the overlap block of the horizontal and vertical parts.

B. Artificial Marker Detection

1) Color Threshold: Calibrate the original image with intrinsic parameters, and then split the original image into red, green and blue channels. Next, set minimum and maximum threshes to remove most part of the background except the artificial marker in each channel and get a color threshold mask.

2) Color Contour Extraction: Contours which fill colors similar to the artificial marker are extracted by the color threshold mask.

3) Contour Noise Rejection: Perimeters of contours are used to delete contour noises in improper size, and minimum enclosing rectangles are employed to fit the contours left. Finally, remove the contours noises in wrong aspect ratio.

4) Block Center Calculation: The parameters in step 1) to 3) are manually adjusted until only four correct square blocks are left behind.

5) Artificial Marker Center Calculation: The geometry constraint among the square blocks fastens the extraction of the artificial marker center. Firstly, denote the centers of square blocks as points A, B, C and D. Secondly, calculate the slopes and corresponding angles of lines AB, AC, AD, BC and CD (Angles are calculated with the atan function). Thirdly, search all the angles until two almost equal ones are found, which means two lines are parallel. Fourthly, record the similar angles and the pairs of centers as the candidates. Here, the correct candidates should be (B, C) and (C, D). Sort the candidate points B, C, C, D in X or Y axis in the image coordinate system. Finally, select the median point, in other words, point C as the center of the artificial marker, as shown in Fig. 2 (b).

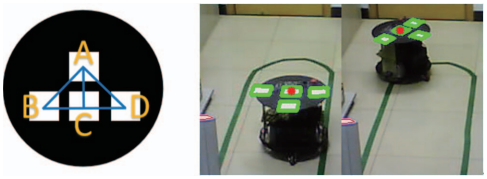

Fig. 2. (a) Pattern of the artificial marker.
(b) Central position estimations of the artificial marker.

C. Distributed Vision Calibration

1) Distortion Elimination: Distortion are eliminated with known intrinsic parameters of the webcam, because excessive distortion sometimes affects the pedestrian shapes.

2) Artificial Marker Plane Transformation: In order to recover a 2D point on the image plane to that on the ground plane, the transformation between two planes has to be known. Assume that the central position of the mobile robot in the map coordinate system $(x, y, 0)$ is already obtained with onboard sensors, so the central position of the artificial marker in the map coordinate system is $(x + a, y + b, c)$. Besides, the central position of the artificial marker captured by the active webcam is (w, h). According to the theory of pinhole imaging, the relationship between (w, h) and $(x + a, y + b, c)$ without distortion is:

$$s \begin{bmatrix} w \\ h \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x+a \\ y+b \\ c \\ 1 \end{bmatrix}. \quad (1)$$

Where s represents the scale factor, f_x, f_y, c_x, c_y are the intrinsic parameters, and r_{11}-r_{33}, t_x, t_y, t_z are the extrinsic parameters of the transformation between the image coordinate system and the map coordinate system.

Four non-collinear central positions of the artificial marker at least can be easily detected in the image coordinate system when the mobile robot is moving around. So we get extrinsic parameters of the webcam by solving the PnP problem.

3) Ground Plane Transformation: The central position of the mobile robot (u, v) on the ground plane in the image coordinate system is generated with that $(x, y, 0)$ in the map coordinate system and the extrinsic parameters. The process can be expressed as:

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ 1 \end{bmatrix}. \quad (2)$$

Then the perspective transformation between the image plane and the map plane on the ground plane is:

$$s'\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}. \quad (3)$$

Where the scale factor s' is used for homogeneous coordinate normalization, and h_{00}-h_{22} compose the homogenous matrix.

Similar to the mobile robot, the pedestrian foothold can also be a pixel point (u, v) in the image coordinate system, therefore, the detected pedestrian foothold $(x, y, 0)$ in the map coordinate system can be calculated.

Localization failure of the mobile robot may cause the samples mixed with the outliers. So RANSAC and least square method are used to fit the samples while removing the outliers.

V. Pedestrian Localization Method

The localization method we introduce is based on the classical pedestrian detection framework, in which pedestrians are featured by HOG and classified by SVM. It also takes advantage of the indoor environment characteristics for faster pedestrian detection, as shown in Fig. 3.

A. Foreground Region Extraction

1) The color image sampled from a webcam is converted into the original gray image.

2) Subtract the current gray image with the background gray image to obtain a foreground gray image.

3) The foreground gray image is handled through the image threshold.

4) Morphological open operation with a $5 * 5$ square structuring element and close operation with a $50 * 50$ square structuring element are used.

5) Rectangles are used to fit the white block, and their features are preserved as the local mask features.

6) Fill the rectangles with pixel value 255 to obtain new local masks. Then save all the rectangle features as valid area features.

B. Foreground Region Scale Normalization

1) Search and crop the original gray image in the valid areas, and then save the cropped sub images as the local foreground images that may contain pedestrians.

2) Along the horizontal and vertical directions, scale the local foreground image simultaneously with the constant scale factor until the height and the width of the scaled local image are larger than the total step size and half of that of the sliding window respectively. The results are preserved as the scale-normalized local foreground gray image, and the total scale factor is also saved. Similarly, process other local foreground images.

C. Pedestrian Detection

1) Calculate the amplitudes and phase angles in the scale-normalized local foreground gray image, and save them all temporarily as the gradient buffer.

2) Scan the scale-normalized local foreground gray image and crop the amplitudes and phase angles corresponding to the location of the sliding window in the gradient buffer. Then calculate and normalize the histograms of oriented gradients.

3) The normalized histograms of oriented gradients are sent into the SVM classifier. If the output is positive, then the sliding window will be localized in the original color image, otherwise process in another window scale and return to step 1) until all scales are scanned, then select the next sliding window and return to step 2).

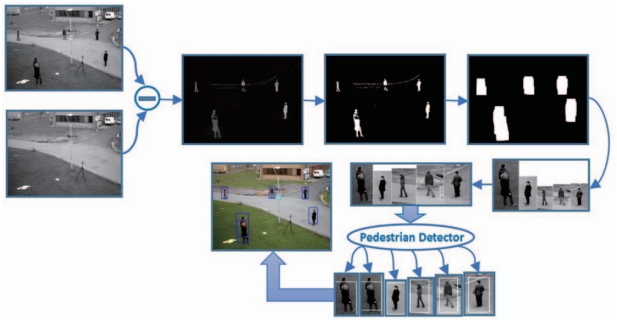

Fig. 3. Pedestrian detection procedure.

D. Pedestrian Localization

1) The positive sliding windows are recovered from the normalized scale to the original scale through dividing the location and size of the sliding windows by the total scale factor s_t.

2) The positive sliding window are recovered from the local foreground image to the original gray image.

3) Specifically, given the location and size of the sliding window in the local foreground image $(u_s, v_s, w_s, h_s)^T$, the transformation from the coordinates in the original gray image (u, v) to those at the normalized scale (u_n, v_n) is shown as below:

$$\begin{bmatrix} u_n \\ v_n \\ 1 \end{bmatrix} = \begin{bmatrix} s_t & 0 & s_t(w_s/2 - u_s) \\ 0 & s_t & s_t(h_s/2 - v_s) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}. \quad (4)$$

Given that the coordinates at the left bottom and those at the right bottom of the sliding window, then we recover the corresponding coordinates in the original gray image.

4) The calibrated parameters between the ground plane and the webcam associated with the IP address are chosen, then the lower left and right coordinates of the sliding window in the image coordinate system are transformed to coordinates in the map coordinate system.

5) The pedestrian in the map coordinate system is modelled as a circle which central position is the location and radius is the influence range.

VI. Global Path Planning

Pedestrian features in the distributed vision system are used to optimize the global path planned for the mobile robot in large scale indoor environment.

A. Publisher Node

The publisher node is created to publish the pedestrian features. If one of the webcams detects pedestrians, its corresponding calibration parameters will be found with its IP address. Then the pedestrians will be localized in the map coordinate system, and their locations and the influence ranges will be published in the pedestrian topic.

B. Pedestrian Layer

The ranges of pedestrian influences are always predicted larger than the actual ranges. Therefore, a proper range is set manually and applied as the radius of the inflation layer. Specifically, a pedestrian layer class is defined and instantiated, then the radius of the inflation layer, observation source of the pedestrian layer, marker, clean-up and update frequency are configured in the common parameter file. Next, plug the pedestrian layer into the global costmap, where the layer type and update frequency should be set.

VII. Experiment Results

Experiments are taken in indoor environment, where a mobile robot runs on the ground plane and a webcam is installed on the ceiling. During the calibration, the mobile robot positions are sent to the server at constant time, and once the server receives, it saves them and records the artificial marker central positions at the same time. Then pedestrian detection and localization are also experimented in indoor environment. Finally, global paths are planned with the mobile robot model in ROS.

A. Calibration

After calibration, the 3D predicted centers of the mobile robot on the ground plane are checked and compared with the ground truth centers using square errors, as shown in Fig. 4 and Fig. 5.

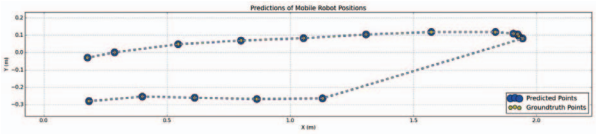

Fig. 4. Mobile robot position predictions.

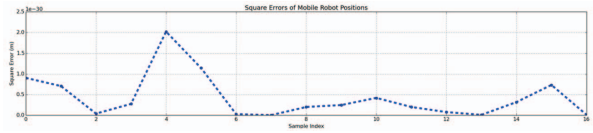

Fig. 5. Square errors of mobile robot position predictions.

B. Pedestrian Detection

A pedestrian in indoor environment is experimented, and the experiment results are shown in Fig. 6. Bounding boxes are sliding windows that are represented as the positive outputs of the SVM classifier with HOG features which correspond to the local foreground images, the sliding windows at original scale and the sliding window after Kalman filtering in green, blue and yellow.

Fig. 6. Detections of pedestrian sequences.

We also fine-tune a unified CNN object detection framework [18] to extract the positions and sizes of the pedestrian in images. Experiments show that results with features extracted by CNN are more accurate, compact and robust to the variations of camera angles and scales, as shown in Fig. 7. However, its detection rate relies on the performance of GPU temporarily, which we use is simply GTX860M and its detection rate is about 10 fps.

Fig. 7. Detections of pedestrians using CNN.

The detection using HOG descriptors greatly depends on the trained template, implicitly assuming that scales of pedestrians in the same environment are similar, so test accuracies on INRIA Person dataset and our samples can achieve 98% and 100% with each of them retrained alone. But both HOG template and CNN cannot generalize well in all situations. For CNN, its accuracy depends on the depth of the easily extensible network architecture and data, and its performance increases faster with suitable weights than classical machine learning algorithms when adding more data.

C. Pedestrian Localization

The pedestrians can be localized from the detected results with the homography matrix. In Fig. 8, the pedestrian footholds are transformed from the image coordinate system to the map coordinate system on the ground plane.

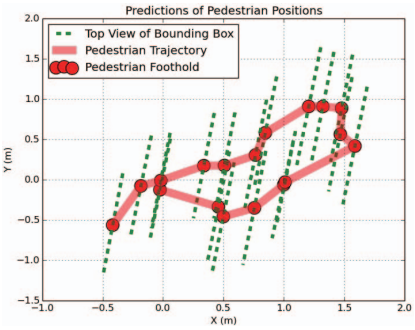

Fig. 8. Position predictions of pedestrian sequences.

D. Global Path Planning

The global path of the mobile robot is planned in the global costmap, as shown in Fig. 9. The red and white circles represent pedestrian models and mobile robots in the map coordinate system respectively. After the pedestrian model is concerned, the original global path in green changes.

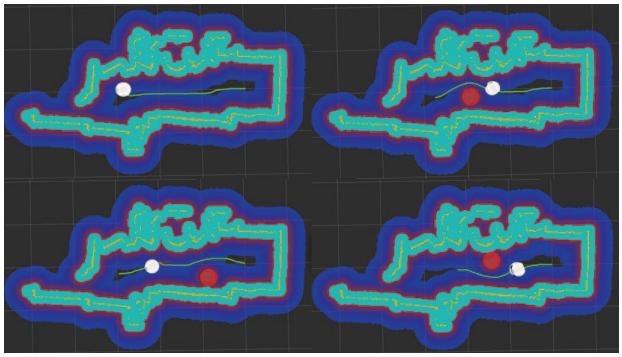

Fig. 9. Global paths planned without and with pedestrians.

VIII. Conclusion

In this paper, we present a method to localize pedestrians from a distributed vision system to a unified static map built with a mobile robot. A reliable calibration method we present is convenient in the distribution vision system, because the only physical work is controlling the mobile robot to move around the indoor environment, and the artificial marker center extraction is simple and robust after adjusting given parameters. The pedestrians are localized on the static map with positive sliding windows, and their localization accuracy depends on the detection method we use, therefore we compare detection performances between SVM classifier with features extracted by HOG and a unified CNN framework. Global path planning is operated on ROS, which easily demonstrates the effectiveness of the static map fusion with the pedestrian layer.

Acknowledgment

This work is supported by a grant from National Natural Science Foundation of China (No. 61203328).

References

[1] Horiuchi T, Thompson S, Kagami S, et al, Pedestrian tracking from a mobile robot using a laser range finder, IEEE International Conference on Man and Cybernetics, pp. 931-936, 2007.
[2] K, Takeuchi E, Ohno K, et al, Forereaching motion generation of mobile robots for pedestrian face identification, Proceedings of SICE Annual Conference, 2010.
[3] Nam B, Kang S, Hong H, Pedestrian detection system based on stereo vision for mobile robot, 17th Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV), pp. 1-7, 2011.
[4] Saito M, Yamazaki K, Hatao N, et al, Pedestrian detection using a LRF and a small omni-view camera for outdoor personal mobility robot, IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 155-160, 2010.
[5] Ahn H S, Yu W, Indoor mobile robot and pedestrian localization techniques, IEEE International Conference on Control, Automation and Systems (ICCAS), pp. 2350-2354, 2007.
[6] Chung S Y, Huang H P, A mobile robot that understands pedestrian spatial behaviors, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5861-5866, 2010.
[7] Ozaki M, Kakimuma K, Hashimoto M, et al, Laser-based pedestrian tracking in outdoor environments by multiple mobile robots, Journal of Sensors, vol. 12, no. 11, pp. 14489-14507, 2012.
[8] Chang F M, Lian F L, Polar grid based robust pedestrian tracking with indoor mobile robot using multiple hypothesis tracking algorithm, Proceedings of SICE Annual Conference (SICE), pp. 1558-1563, 2012.
[9] Kidokoro H, Kanda T, Brscic D, et al, Will I bother here?-a robot anticipating its influence on pedestrian walking comfort, ACM/IEEE International Conference on Human-Robot Interaction (HRI), pp.259-266, 2013.
[10] Butail S, Simulating the effect of a social robot on moving pedestrian crowds, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2413-2418, 2015.
[11] Nitta J, Sasaki Y, Mizoguchi H, Path Planning Using Pedestrian Information Map for Mobile Robots in a Human Environment, IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 216-221, 2015.
[12] Grisetti G, Stachniss C, Burgard W, Improved techniques for grid mapping with rao-blackwellized particle filters, IEEE Transactions on Robotics, vol. 23, no. 1, pp. 34-46, 2007.
[13] Quigley M, Conley K, Gerkey B, et al, ROS: an open-source Robot Operating System, IEEE International Conference on Robotics and Automation (ICRA), pp. 5, 2009.
[14] Dalal N, Triggs B, Histograms of oriented gradients for human detection, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), no.1, pp. 886-893, 2005.
[15] Lu D V, Hershberger D, Smart W D, Layered costmaps for context-sensitive navigation, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 709-715, 2014.
[16] Skiena S, Dijkstra's algorithm, Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica: Addison-Wesley, pp. 225-227, 1990.
[17] Fox D, Burgard W, Dellaert F, et al, Monte carlo localization: Efficient position estimation for mobile robots, Innovative Applications of Artificial Intelligence (IAAI), pp. 343-349, 1999.
[18] Redmon J, Divvala S, Girshick R, et al, You only look once: Unified, real-time object detection, arXiv: 1506.02640, 2015.