Computer Communications 160 (2020) 730–737

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Design of the navigation system through the fusion of IMU and wheeled encoders

Shibing Yu, Zhen Jiang *

School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200072, PR China

ARTICLE INFO

Keywords:
Navigation system
Local obstacle avoidance
Pose estimation
Multisensor fusion
Mobile robot

ABSTRACT

Mobile robots must autonomously navigate in structured environments. In this paper, we proposed a new navigation system for mobile robots, where lost-cost components were used to complete accurate navigation tasks. To accomplish the navigation task, we designed and validated a set of algorithms for obstacle avoidance and estimation of the robot pose. The proposed navigation system consists of three main modules: local occupancy map generation, extended Kalman filter (EKF)-based localization and modified obstacle avoidance algorithm. The system initially performed visual feature point tracking followed by feature point classification using an inverse perspective transformation for obstacle detection. When the estimated obstacle position is provided, a local obstacle avoidance algorithm is activated to generate motion commands for safe navigation. When the local occupancy map is generated, a modified artificial potential field based on boundary detection is developed for navigation through an indoor environment. Moreover, we proposed a strategy to localize a mobile robot using an IMU, a camera and wheel encoders. The EKF is adopted to reduce the sensor noise and bias. Experiments demonstrate that our navigation system is useful and can significantly benefit other reactive navigations.

1. Introduction

For an autonomous mobile robot, obstacle avoidance can discriminate obstacles from transmittable terrains to make decisions about its navigation. Such strategies must be simple and have high efficiency, since onboard sensors are scarce, and efficient path planning is required. Laser range finder and ultrasonic are basically used for obstacle avoidance and local navigation in the environment. However, these range sensing systems only provide distance information on a single plane, which makes it difficult to perceive the world scene with abundant 3D space. When a camera is employed, points with coordinates $\mathbf{X} = (x, y, z)^T$ are mapped in a 3D space. This projection relationship can provide abundant obstacle information about the surrounding.

Another issue is the accurate pose estimation in mobile robotics, which involves data fusion acquired from equipped sensors such as the onboard encoders, IMUs, and cameras. Due to slippage and disturbance, the pose estimation obtained by only two wheel encoders is not reliable. Moreover, the pose estimation using an IMU is only reliable in a short time due to systematic errors. Vision-based odometry builds a check correspondence from two consecutive image frames, which are not susceptible to slippage of the mobile robot. If a consumer-grade camera is equipped on mobile robots to avoid obstacle, it is advisable to use the minor disturbance of visual odometry to estimate the robot pose. In addition, the onboard sensorial equipment can be reduced to decrease energy consumption. Meanwhile, vision information can be used to solve the obstacle detection and localization problem.

Bonin-Font, et al. [1] developed methods to detect obstacles only with monocular vision using feature points. A similar strategy is used in [2]. Feature points are extracted to build a point2point correspondence for ground detection using the inverse perspective transform [3,4]. Detection and tracking feature points have been extensively investigated in robotic vision. Feature points are typically labelled with a feature descriptor. Harris corner [5] was used for feature detection and tracking and has repeatable results . However, it fails when there are abrupt scale changes or considerable rotation between two image frames. Lowe [6] proposed the SIFT feature detector, which was invariant to image scale, rotation, illumination, and change in the 3D viewpoint. SURF [7], FAST [8], CenSurE [9], BRIEF [10], and ORB [11] also show excellent performance to detect feature points. SURF speeds up using an integral image but generates fewer interest points. Machine-learning-based FAST significantly improves the computation time but is sensitive to noise. CenSurE outperforms the other detectors with stability and accuracy for image registration and visual odometry. BRIEF is highly discriminative but very sensitive to scale and rotation changes. However, these feature matching algorithms may produce biased results. A random sample consensus (RANSAC) is commonly required to remove data outliers.

* Corresponding author.
E-mail addresses: yushibing@baosight.com (S. Yu), zhjiang@shu.edu.cn (Z. Jiang).

https://doi.org/10.1016/j.comcom.2020.07.009
Received 12 April 2020; Received in revised form 4 July 2020; Accepted 6 July 2020
Available online 9 July 2020

Visual odometry (VO) [12,13] has accurate pose estimation and is not affected by wheel slip in an uneven terrain. Motion estimation is described by the generally called essential matrix E, which depicts the geometric relations between two images of a calibrated camera. Meanwhile, VO can be used to estimate the robot localization through a combination with an IMU and/or wheel encoders [14–17]. EKF has been extensively used in autonomous robots, such as mapping and localization. These solutions involve integration of the observed environmental information using onboard sensors with control information such as the pose or velocity of a robot. Mourikis, et al. [18] proposed a visual navigation method for planetary landing, which was tested for a sounding-rocket flight, where Harris corners were detected from consecutive image frames that were captured during the descent process, and an EKF was adopted to estimate using an IMU and visual information to infer the relative pose and velocity of the planetary lander. A vision-aided Inertial Navigation System (V-INS) was proposed and verified in autonomous flight for an Unmanned Aerial Vehicle (UAV) [19], where a Harris corner detector was used to estimate the feature localization. These systems usually utilize vision cue to pose estimation. Obstacle detection is implemented using additional ranging sensors. Cherubini, et al. [20] proposed a key-frame visual navigation strategy, where an onboard LIDAR is used to detect obstacles. Jung, et al. [21] presented an algorithm with the integration of the 2D motion estimation and range measurement from a LIDAR, which retrieved partial 3D information of moving objects and built a local occupancy grid map for navigation. Liu, et al. [22] presented an autonomous localization method using an EKF-based novel feature-level data fusion for indoor environments.

In an unknown environment, different algorithms are developed for obstacle avoidance. Artificial Potential Field (APF) is the typical path-planning strategy for mobile robots [23]. Because of its specific physical implication and simple mathematical description, APF is very appropriate to control mobile robots in real time [24–26]. Therefore, APF has been extensively investigated for real-time path planning of mobile robots. However, there are often so-called local minimums in the entire potential field. Therefore, modified methods have been investigated. Bacterial Potential Field (BPF) was proposed to improve the APF with the Bacterial Evolutionary Algorithm (BEA) [27,28]. Modifications of Newton's method were investigated to solve the inherent oscillation problems in potential field methods (PFMs) for nonholonomic mobile robots [29]. Furthermore, local minimum avoidance is implemented by reshaping the potential field to offer better performance with the local minima [30]. These proposed methods are developed to enable a feasible, optimal and safe path.

In this paper, to solve the aforementioned problem, we design and validate a set of algorithms to avoid obstacles and estimate the robot pose. The proposed navigation system consists of three main modules: local map generation, EKF-based localization and obstacle avoidance algorithm. The obstacles are detected with the image-processing strategy to build a local polar grid map to perceive the partial 3D position, and a modified artificial potential field is activated to go through the explored environment. In addition, homography-based and EKF-based pose estimation is performed, which is fused with the IMU and wheel encoders. The proposed EKF is adopted to fuse the sensing information and incorporate the known kinematics of mobile robots to reduce the effects of sensor noise and bias.

This paper is organized as follows. Section 2 presents related background techniques and the entire implementation framework of our navigation system. Section 3 presents the homography-based robot pose estimator using an EKF by fusion the pose estimation from an IMU, a camera and wheel encoders. The boundary-detection-based APF algorithm is investigated to generate optimal paths and eliminate the local minimum in Section 4. An experimental analysis is provided in Section 5. Finally, we discuss our findings and conclude this paper in Section 6.

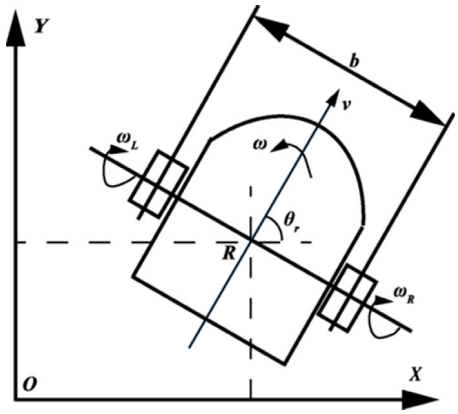

Fig. 1. Kinematics of wheeled robots.

2. Background

2.1. Nonholonomic mobile robot system

The wheeled robot is modelled as a kinematic unicycle and has three degrees of freedom: x_r, y_r, and θ_r. It moves on the xy-plane and vertically rotates with angular velocity ω, as illustrated in Fig. 1. The kinematics of the wheeled robot are depicted in Eq. (1).

The encoders and IMU are commonly equipped to obtain the angular velocities of mobile robots. Here, the angular velocities are represented by ω_L and ω_R for the left and right wheels, as shown in Fig. 1.

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta_r) & 0 \\ \sin(\theta_r) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{1}$$

The relationship of the linear and angular velocities is described by Eq. (2).

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{b} & -\frac{r}{b} \end{bmatrix} \begin{bmatrix} \omega_L \\ \omega_R \end{bmatrix} \tag{2}$$

where r is the radius of the wheels, and b is the distance between two wheels. The IMU also measures the linear acceleration using accelerometers.

2.2. Feature-point-based obstacle detection

The 3D object space is mapped to a 2D image via a camera. Suppose that a camera is modelled using the pinhole model; a point with coordinates $\mathbf{X} = (x, y, z)^T$ in 3D space is linearly mapped to the point on the image plane, both of which are depicted in homogeneous coordinates. The linear mapping can be written as [31]

$$\begin{pmatrix} x_p \\ y_p \\ f \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} T_w^c \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \quad T_w^c = \begin{bmatrix} R & \mathbf{t} \\ 0 & 1 \end{bmatrix} \tag{3}$$

where (x_p, y_p) are the coordinates of the image points; f is the focal length of the cameras; T_w^c is the transform matrix, including a rotational matrix and a translational vector, as shown in Fig. 2).

The scene point in world coordinates can be computed from the inverse perspective transform as follows:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} + \lambda \begin{bmatrix} x_p \cos\theta - f \cos\varphi \sin\theta + y_p \sin\varphi \sin\theta \\ x_p \sin\theta + f \cos\varphi \cos\theta - y_p \sin\varphi \cos\theta \\ f \sin\varphi + y_p \cos\varphi \end{bmatrix} \tag{4}$$

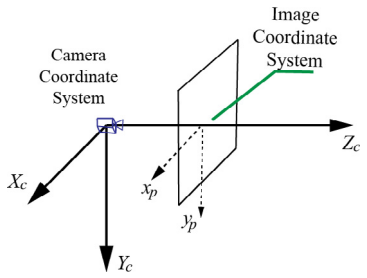

Fig. 2. Image coordinate system.

where (X_c, Y_c, Z_c) are the camera world coordinates; θ and φ are the yaw angle and pitch angle of the camera, respectively; λ is a nonzero value, which denotes the exact world point position. In this paper, we implement the ground detection using IPT if all points on the floor have $z = 0$. We obtain

$$\lambda = -\frac{Z_c}{f \sin\varphi + y_p \cos\varphi} \tag{5}$$

The remaining floor coordinates (x, y) that correspond to $z = 0$ are

$$\begin{cases} x = X_c - \dfrac{Z_0\left(x_p \cos\theta - f\cos\varphi\sin\theta + y_p \sin\varphi \sin\theta\right)}{f\sin\varphi + y_p\cos\varphi} \\ y = Y_c - \dfrac{Z_0\left(x_p \sin\theta - f\cos\varphi\cos\theta + y_p \sin\varphi \cos\theta\right)}{f\sin\varphi + y_p\cos\varphi} \end{cases} \tag{6}$$

A sufficiently large and relevant set of feature points is extracted and continuously tracked during the robot motion in our algorithm. When a camera is used, some significant points, which are called corner or feature points, in the image are tracked frame to frame. All feature points are projected using IPT between successive frames assuming that $z = 0$. Hence, our implementation begins by detecting feature points using CenSurE, since it is more suitable in long track lengths for visual odometry than other detectors. All image points are classified as ground points or obstacles using the following criteria

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}, \quad \begin{cases} d > \zeta \geq obstacle \\ d \leq \zeta \geq ground \end{cases} \tag{7}$$

where d denotes the relationship between image points in two consecutive frames captured at different time instants; ζ is a threshold value to classify the feature points as flat ground or an obstacle, which is equal to zero.

2.3. Plane-based homography

The transnational motion of a mobile robot between two continuous time instances can be estimated by constructing a homographic relationship in two successive images captured by an onboard camera. At least four coplanar and noncollinear points are required to establish a homographic relationship. These points must have one-to-one correspondence in both images.

Fig. 3 demonstrates the relationship between two images taken from different poses, including translation and rotation. F_k is considered a current frame. The relationship between two different poses are expressed via (T_k, R_k). The coordinates of the detected feature point in frame F_k are depicted by vector $\overline{m}_j = [x_j, y_j, z_j]^T \in R^3$.

The feature points in these 2D planes are encapsulated as a normalized set of 3-D coordinates. The image plane coordinates as measured from F_k are defined as vector $m_j = [u_j, v_j, 1]^T \in R^3$. Formally, world point $\overline{m}_j$ is mapped to point m_j on the image plane. These vectors are depicted by

$$m_j = [u_j, v_j, 1]^T = \left[\frac{x_k}{z_k}, \frac{y_k}{z_k}, 1\right] \tag{8}$$

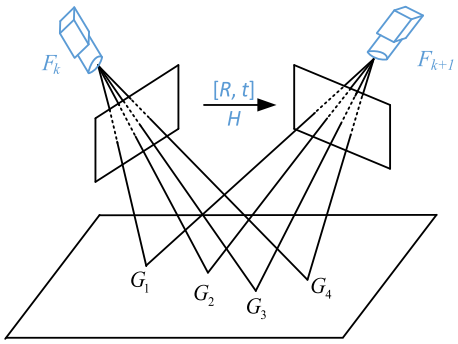

Fig. 3. Plane-based translation and rotation.

For the navigation of mobile robots, a visual odometry strategy is utilized using plane-based homography. The proposed vision-based method is delivered to estimate the position and orientation of mobile robots. Euclidean homography is induced assuming that all feature points lie on a plane. Suppose that two images of points $\overline{m}_j$ lay on plane π; the transformation between the two sets of points is depicted by Eq. (9).

$$\overline{m}_j^{k+1} = R^k \overline{m}_j^k + T^k \tag{9}$$

The relationship of two feature points, which are mapped by the same world point, can be described in term of image points as shown in Eq. (10).

$$m_j^{k+1} = H_d^k m_j^k \tag{10}$$

where matrix H_d^k is Euclidean homography matrix, which refers to the relationship of plane-based homography. It is given by

$$H_d^k = R^k + \frac{T^k}{d^k}\mathbf{n}^k \tag{11}$$

where $\mathbf{n}^k \in R^3$ is a unit normal vector of plane π in F_k, and d^k is the scalar distance from the optical centre of the camera to plane π. In this paper, the configuration of our mobile robot is known for the development. Thus, we assume that d^k and $\mathbf{n}^k$ are known constants.

3. Multisensor-based robot localization

Due to slippage and disturbance, the pose estimation that is only obtained by two wheel encoders is not reliable. The pose estimation from an IMU is only reliable during a short time due to systematic and random errors. However, vision-based odometry is not susceptible to the slippage of a mobile robot. In this paper, a disturbance compensator is presented to compute the current position of the robot. We propose a strategy to localize a mobile robot using an IMU, a camera and wheel encoders. The EKF is adopted to reduce sensor noise and bias. Experiments have demonstrated that our navigation system is useful and can significantly benefit other reactive navigations. The state space equation of the system is represented by Eq. (12).

$$\begin{cases} s_k = A_k s_{k-1} + q_k \\ z_k = B_k s_k + r_k \end{cases} \tag{12}$$

where s_k is the state vector, A_k is the state transition matrix, z_k is the measurement vector, and B_k is the measurement matrix. q_k and r_k are normally distributed and randomly processed with zero mean and covariance matrices Q_k and R_k.

An accurate localization of a nonholonomic mobile robot implies that we must know the position (x_r^k, y_r^k) and orientation θ_r^k in world coordinate. The state vector of the system is:

$$s_k = \left[x_r, y_r, \theta_r, v, \omega\right]^T \in R^5 \tag{13}$$

where velocity $v(k)$ and angular velocity $\omega(k)$ are measured in the robot frame. The state transition matrix $A_k \in R^{5\times5}$ is depicted as follows

$$A_k = \begin{bmatrix} 1 & 0 & 0 & \cos\theta \cdot \Delta t & 0 \\ 0 & 1 & 0 & \sin\theta \cdot \Delta t & 0 \\ 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{14}$$

The state noise covariance matrix $Q_k \in R^{5\times5}$ is described as

$$Q_k = \begin{bmatrix} \sigma_1 & 0 & 0 & \cos\theta \cdot \sigma_4 & 0 \\ 0 & \sigma_2 & 0 & \sin\theta \cdot \sigma_4 & 0 \\ 0 & 0 & \sigma_3 & 0 & \sigma_5 \\ \cos\theta \cdot \sigma_4 & \sin\theta \cdot \sigma_4 & 0 & \sigma_4 & 0 \\ 0 & 0 & \sigma_5 & 0 & \sigma_5 \end{bmatrix} \tag{15}$$

The measurements are collected by wheel encoders, inertial sensors, and vision-based estimation. As depicted in Section 2.3, Euclidean homography matrix H_d^k is calculated and decomposed at each frame to estimate the translation and rotation of the robot. Measurements from the vision-based estimation provide the pose of mobile robots using the constant transformation between the camera frame and the robot frame. Vision-based pose estimates can determine the position and orientation of the robot in the robot frame, which are denoted by $x_h(t)$, $y_h(t)$, and $\theta_h(t)$. The measurements from the wheel encoders are comprised of linear velocity $v_w(k)$ and angular velocity $\omega_w(t)$ of mobile robots as described in Eq. (2). The final measurements from the IMU are the linear and angular velocities $\omega_g(t)$. All measurements are combined to form measurement vector z_k, which is defined as

$$z_k = \left[\omega_w, \omega_g, v_w, \theta_h, x_h, y_h\right] \in R^6 \tag{16}$$

The measurement matrix $B_k \in R^{6\times5}$ can be written by

$$B_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{17}$$

Finally, the measurement noise covariance matrix R_k is depicted in Eq. (18).

$$diag\left(R_k\right) = \left[\sigma_w^{\omega}, \sigma_g^{\omega}, \sigma_w^{v}, \sigma_h^{\theta}, \sigma_h^{x}, \sigma_h^{y}\right] \tag{18}$$

4. Boundary-detection-based artificial potential field

In this paper, we present a boundary-detection-based APF to complete the local obstacle avoidance task. To solve the problem of goal nonreachable with obstacle nearby (GNRON), a component of the repulsion field is added in the direction of attraction. To avoid the local minimum of obstacle avoidance, the task of boundary detection is to determine the next waypoint when the robot moves in the region of crowded obstacles; then, a virtual force is added to escape from the crowded region. In this case, the obstacle is considered a simplified particle, and the obstacle boundary is extended in the space to ensure absolute security. Additionally, the constraints of the robot size and a safe distance should be carefully considered to avoid obstacles when the robot is going through the obstacle region. When the robot is approaching the target, the repulsion component decreases, and the resultant force in the direction of attraction increases to eliminate GNRON, as shown in Fig. 4.

In 2D space, the modified attraction field is

$$U_{att} = \frac{1}{2}k\rho^2\left(\mathbf{X}, \mathbf{X}_t\right) \tag{19}$$

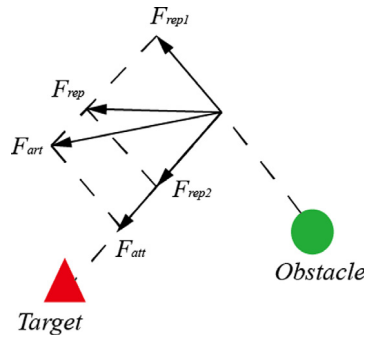

Fig. 4. Diagram of the artificial potential force.

In addition, the modified repulsion field is

$$U_{rep}(\mathbf{X}) = \begin{cases} 0.5\eta\left(\frac{1}{\rho(\mathbf{X},\mathbf{X}_0)} - \frac{1}{\rho_0}\right)^2\rho^n(\mathbf{X},\mathbf{X}_t), & \rho(\mathbf{X},\mathbf{X}_0) \le \rho_0, \\ 0, & \rho(\mathbf{X},\mathbf{X}_0) > \rho_0. \end{cases} \tag{20}$$

where $\mathbf{X} = (x, y)$ is the coordinates of the waypoints; $\mathbf{X}_0 = (x_0, y_0)$ is the coordinates of the obstacles; $\mathbf{X}_t = (x_t, y_t)$ is the coordinates of the target; ρ_0 is the safe distance to avoid obstacle; $\rho(\mathbf{X}, \mathbf{X}_t)$ is the shortest path in 2D space; $\rho(\mathbf{X}, \mathbf{X}_0)$ is the distance from the robot to the obstacle; $\rho^n(\mathbf{X}, \mathbf{X}_t)$ is an adjustment factor. Then, the negative gradient of attraction field F_{att} is:

$$F_{att} = -\nabla\left[\frac{1}{2}k_p\rho^2(\mathbf{X}, \mathbf{X}_t)\right] = k\rho(\mathbf{X}, \mathbf{X}_t) \tag{21}$$

Meanwhile, the negative gradient of repulsion field F_{rep} is expressed by Eq. (22).

$$F_{rep} = \begin{cases} \eta F_{rep1}(\mathbf{X}) + \frac{n}{2}(1-\eta)F_{rep2}(\mathbf{X}), & \rho(\mathbf{X},\mathbf{X}_0) \le \rho_0, \\ 0, & \rho(\mathbf{X},\mathbf{X}_0) > \rho_0. \end{cases} \tag{22}$$

In our modified model, F_{rep} consists of F_{rep1} and F_{rep2}, where F_{rep1} is related to an obstacle, and F_{rep2} is produced by the target, as shown in Fig. 4.

$$\begin{cases} F_{rep1} = \left(\frac{1}{\rho(\mathbf{X},\mathbf{X}_o)} - \frac{1}{\rho_o}\right)\frac{\rho^n(\mathbf{X},\mathbf{X}_t)}{\rho^2(X,X_o)} \\ F_{rep2} = \left(\frac{1}{\rho(\mathbf{X},\mathbf{X}_o)} - \frac{1}{\rho_o}\right)^2\rho^{n-1}(\mathbf{X},\mathbf{X}_t) \end{cases} \tag{23}$$

The two components of repulsion force F_{rep} and attraction force F_{att} can be obtained by Eq. (24).

$$\begin{bmatrix} F_{repx}(\mathbf{X},\mathbf{X}_o) \\ F_{repy}(\mathbf{X},\mathbf{X}_o) \\ F_{attx}(\mathbf{X},\mathbf{X}_t) \\ F_{atty}(\mathbf{X},\mathbf{X}_t) \end{bmatrix} = \begin{bmatrix} F_{rep}(\mathbf{X},\mathbf{X}_o)\cos\beta \\ F_{rep}(\mathbf{X},\mathbf{X}_o)\sin\beta \\ F_{attx}(\mathbf{X},\mathbf{X}_t)\cos\alpha \\ F_{atty}(\mathbf{X},\mathbf{X}_t)\sin\alpha \end{bmatrix} \tag{24}$$

The steering angle θ' of the robot between the resultant force and the x-axis is obtained by Eq. (25). The next waypoint of the robot is computed until the convergence of the path in Eq. (26).

$$\theta' = \arctan\frac{F_{atty}(\mathbf{X},\mathbf{X}_t) + F_{rep1y}(\mathbf{X},\mathbf{X}_o) + F_{rep2y}(\mathbf{X},\mathbf{X}_o)}{F_{attx}(\mathbf{X},\mathbf{X}_t) + F_{rep1x}(\mathbf{X},\mathbf{X}_o) + F_{rep2x}(\mathbf{X},\mathbf{X}_o)} \tag{25}$$

$$\begin{cases} x' = x + v^*\cos\theta \\ y' = y + v^*\cos\theta \end{cases} \tag{26}$$

where v^* is the velocity of the robot, and (x', y') is the coordinates of the next waypoint. The modified APF ensures that the robot safely arrives at the target when obstacles are simplified as a series of particles. Assume that $n = \rho^n(\mathbf{X}, \mathbf{X}_t)$. If $0 < n < 1$, F_{rep1} tends to the infinity when the robot is approaching the target. If $n = 1$, F_{rep2} tends to be a constant, and F_{rep1} will tend to zero. If $n = 0$, F_{rep1} and F_{rep2} will tend to zero. Thus, GNRON can be avoided if n is set as a positive real number.

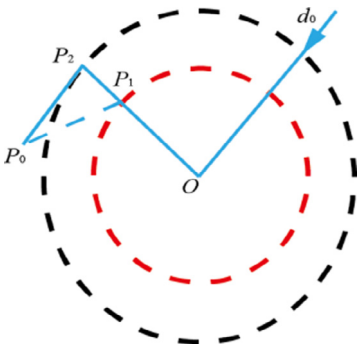

Fig. 5. Diagram of boundary detection for a safety zone.

The local minimum makes the robot stay in a region so that the robot cannot directly arrive at the target. If the robot falls into the region of a local minimum, it should satisfy three situations, as shown in Eq. (27).

$$\begin{cases} A. \quad \left|F_{att}(x) + F_{rep}(x)\right| \le \varepsilon \\ B. \quad |x(t+T) - x(t)| \le s_1 \\ C. \quad |x_A - x_B| \le \alpha \cdot s_{AB} \end{cases} \tag{27}$$

where ε is a minimum positive value. The three situations are as follows. In condition *A*, the resultant force of F_{att} and F_{rep} is close to ε, which implies the robot falls into the most common local minimum. In Condition *B*, s_1 is a threshold, and the displacement of the robot is small during interval *T*. In Condition *C*, α is a decimal percentage, s_{AB} is the distance that the robot has travelled between waypoint *A* and waypoint *B*, and $|x_A - x_B|$ is the distance between *A* and *B*. When the robot falls into the region of local minimum in most instances, it can be denoted using Condition *A*, *B* or *C*. If the next waypoint of the robot is within the safety zone of the obstacle, the boundary of the safety zone should be adjusted, as shown in Fig. 5.

As shown in Fig. 5, $O(x, y)$ denotes an obstacle, and d_0 is its safety distance. $P_0(x, y)$ and $P_1(x, y)$ are the waypoints to which the robot originally plans to travel. To maintain safety, $P_2(x, y)$ is the new waypoint after the boundary of the obstacle is revised. The boundary detection is executed as shown in Eq. (28).

$$\begin{cases} \left|P_1(x, y) - O(x, y)\right| < d_0 \\ \theta = \arctan\left(\dfrac{O(x, y)_y - P_1(x, y)_y}{O(x, y)_x - P_1(x, y)_x}\right) \\ P_2(x, y)_x = O(x, y)_x - \cos\theta \cdot d_0 \\ P_2(x, y)_y = O(x, y)_y - \sin\theta \cdot d_0 \end{cases} \tag{28}$$

where θ is the heading angle. If $OP_1 \le d_0$, the boundary of the safe zone is revised, and P_2 is recalculated in Eq. (28).

Otherwise, a secondary temporary target is determined to solve the local minimum, as shown in Fig. 6. R_0, T, and T_2 are the current position of the robot, target and secondary target, respectively. After the robot arrives at the secondary temporary target, it can continuously move towards the goal. If a robot cannot escape from the region of the local minimum, the secondary temporary target helps the robot to get away from its dilemma. Thus, the first, second and third secondary targets should be continuously updated until the robot is absolutely out of difficulties.

To help the robot escape from the local minimum, the boundary detection method is proposed to adjust the attitude of the robot and find an optimal feasible path. Next, three steps are taken to specifically set the secondary target.

Step 1: The straight-line K_0 is formed by the next waypoint R_0 and target *T*.

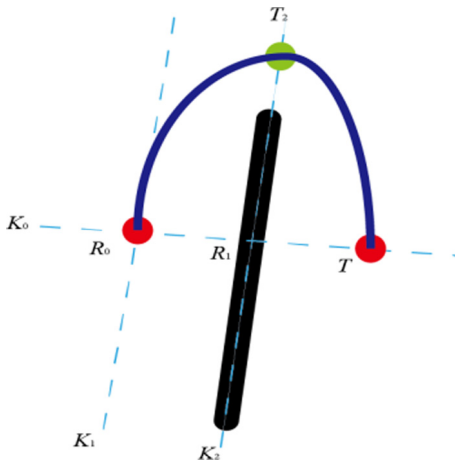

Fig. 6. Secondary target selection strategy.

Step 2: The robot moves along K_0, and the first encountered obstacle is stamped as R_1. The second straight-line K_2, which is perpendicular to K_0, is formed. To avoid obstacles, the second target T_2 is set to ensure the shortest path according to the geometric constraints.

Step 3: A straight-line K_1 parallel to K_0 and passing through R_0 is determined. Then, the robot moves along K_1. The secondary target is determined on the side without obstacles. Furthermore, if obstacles are distributed on two sides, the secondary target is selected with a shorter distance between the current waypoint and one side.

5. Experimental analysis

Extensive experiments were investigated to determine the effectiveness and robustness of the proposed localization and obstacle avoidance strategy. An xPartner IN-R is used in this paper. In our implementation, a consumer-graded camera and a MicroStrain 3DM-GX-25TM are used. The camera is supplied by Edmund. The Intel OpenCV library was adopted for image processing [32]. A Notebook Thinkpad T430 was affixed to the robot to perform image processing, robot control, and path planning, as shown in Fig. 7. Here, Thinkpad T430 is a powerful device to implement the related algorithms. The motors are driven by a DSP-based motion control module. The image processing is sped up by a CUDA-enabled GPU installed in our notebook.

The three modules are implemented using different techniques. First, we perform visual feature point tracking followed by feature point classification using the inverse perspective transformation for obstacle detection. When the estimated obstacle position is given, a local obstacle avoidance algorithm is activated to generate motor commands to guide the robot for the point2point navigation. Here, a modified APF based on boundary detection is developed to navigate through an indoor environment. Here, an accurate localization of mobile robots is obtained via the fusion of pose estimation from an IMU, a camera and wheel encoders.

5.1. Simulation of obstacle avoidance

As shown in Fig. 8, two different trajectories are generated using two different algorithms. The trajectory marked as the red line is generated by the proposed modified APF algorithm, and the other marked as the green line is generated using the original APF algorithm. The safe zone of an obstacle is marked as the blue dash line. In Fig. 8(a), the planning path using the proposed modified APF algorithm becomes smoother than the path generated by the original APF algorithm. To avoid obstacles, the robot moves along the revised obstacle boundary generated by the proposed modified APF algorithm. Here, a secondary target is determined to solve the local minimum. Otherwise, as shown

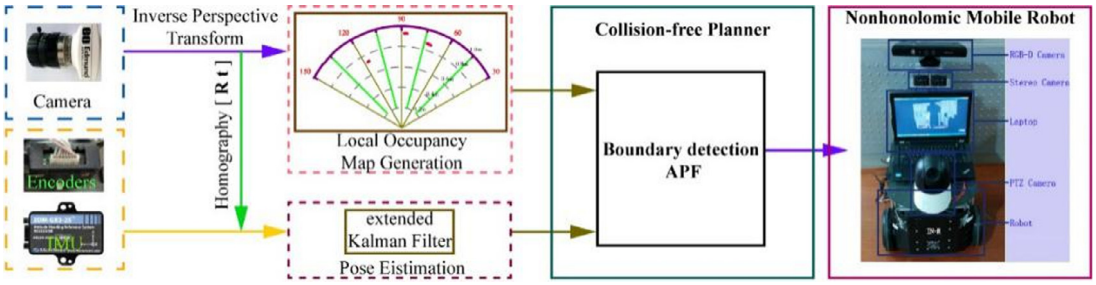

Fig. 7. Navigation system architecture.

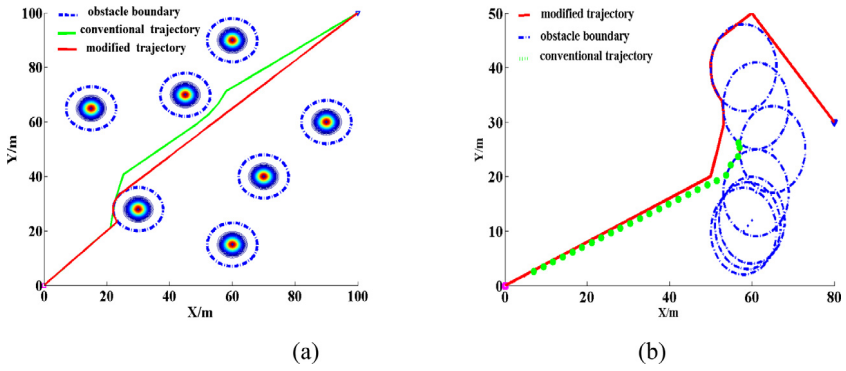

Fig. 8. Simulation results of boundary detection APF. (a) Simulation results in the general case; (b) simulation results in a local minimum. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

in Fig. 8(b), the robot moves in a crowded region. The robot struggles in front of the obstacles and falls into the region of the local minimum using the original APF algorithm. However, the robot can escape from the region of the local minimum and move towards the goal continuously using the proposed modified APF algorithm. The simulation results demonstrate that our proposed modified APF is feasible.

5.2. Pose estimation of motion scenarios

The measurements from wheel encoders provide accurate poses and can be simply considered zero when the robot is stationary. However, they are not accurate when the robot moves. Furthermore, the measurements from the cameras and IMUs are not reliable because of sensor noise under any condition. However, vision-based measurements can be adopted to eliminate ambiguous solutions. Suppose that the planar feature points have a face normal $\mathbf{n}^*$, and the distance from the camera to the ground is d^*; then, the camera mounted at the relative position ensures that $\mathbf{n}^*$ and d^* are constants. We validate our proposed localization algorithm in a square zone, as shown in Fig. 9. The experimental result shows that the vision-based localization fused with IMU and encoders is more robust than that of IMU and encoders. In the initial stage, two methods have similar performances. However, with time accumulation over a period and slippage occurrence, the localization with IMU and encoders severely deviates from the real trajectory. The proposed localization method using the IMU, encoders and a camera performs better over a long period. Points discriminated from the criteria in Eq. (7) on the ground are used to find the homography matrix.

5.3. Local occupancy map generation

In our implementation, a consumer-graded camera is used to obtain the pose measurements. Let (X_c, Y_c, Z_c) and $(x, y, 0)$ denote the camera position and ground point; then, the relative distance and orientation between the camera and the ground can be calculated. In this implementation, we assume that the world coordinate is z = 0 on

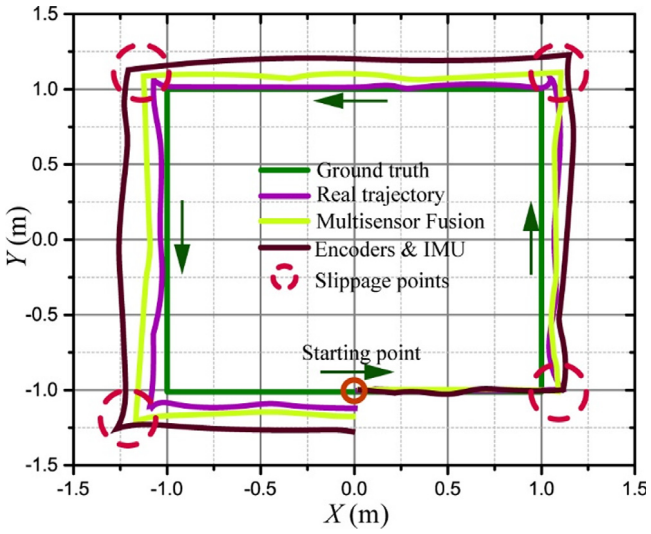

Fig. 9. Experimental result with the square motion scenario.

the ground plane. The obstacle profiles are described in [1], and we adopt an edge detector to classify obstacles, as shown in Fig. 10. The camera world coordinates, and robot global coordinate were obtained from the pose estimation using the proposed EKF in this paper.

5.4. Obstacle avoidance of the mobile robot

An autonomous navigation task is performed to test our proposed system. The localization of the robot is realized using the proposed odometry in this paper. The navigation task involves goal seeking accompanied by obstacle avoidance. Here, the navigation system is to guide the mobile robot to reach a destination away from any obstacle. Meanwhile, the goal-seeking task ensures that the robot moves towards

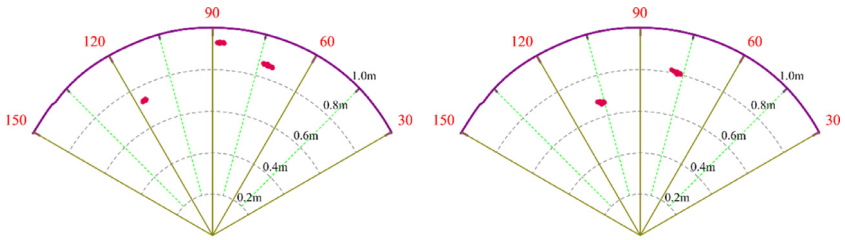

Fig. 10. Scenarios of robot navigation. (a) Scenario 1: local occupancy map with three obstacles; (b) Scenario 2: local occupancy map with two obstacles.

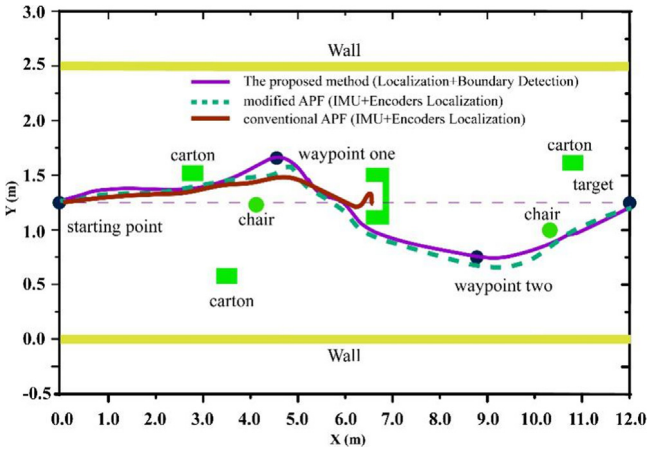

Fig. 11. Trajectory recorded of navigation experiments using different methods.

the target. We established a 2.5-m × 12-m rectangle area, where cartons and chairs are placed as obstacles. To validate our proposed algorithms, two contrast experiments were implemented using different techniques, as shown in Fig. 11. In the contrast experiments, the robot was localized using dead reckoning data fused with IMU and wheel encoders. In this work, the exposed experiments demonstrate that the feasibility is proven using vision information to improve the robot pose estimation and obstacle avoidance of the navigation system.

6. Conclusion

In this paper, a set of algorithms was designed and validated for obstacle avoidance and estimation of the robot pose. In our vision-guided navigation system, we focus on tackling the slippage and accumulating sensor error during robot motion. Meanwhile, a collision-free planner is employed to reach the target when a local map is built. First, we perform visual feature point tracking followed by feature point classification using the inverse perspective transformation for obstacle detection. The obstacle detection is fused with the image-processing strategy to build a local polar grid map to perceive partial 3D position using dead reckoning. When the estimated obstacle position is given, a local obstacle avoidance algorithm is activated to generate motor commands to guide the robot for point2point navigation. On this basis, the proposed modified APF based on boundary detection is developed for navigation through an indoor environment. Finally, a method to localize the robot is proposed via fusion of the pose estimation from an IMU, a camera and wheel encoders using an EKF. The proposed EKF is adopted to fuse sensor measurements with the known system kinematics to reduce the effect of noise and bias from the sensors. The three modules were implemented using different techniques, and the simulation and experiment were set to verify our method. Through a combination of several techniques, extensive experiments verified the effectiveness of the proposed navigation system.

CRediT authorship contribution statement

Shibing Yu: Conceptualization. **Zhen Jiang:** Writing - original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 61433016). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] F. Bonin-Font, A. Burguera, A. Ortiz, G. Oliver, A monocular mobile robot reactive navigation approach based on the inverse perspective transformation, Robotica 31 (02) (2013) 225–249.
[2] C.-H. Lin, K.-T. Song, Robust ground plane region detection using multiple visual cues for obstacle avoidance of a mobile robot, Robotica 33 (02) (2015) 436–450.
[3] E. Fazl-Ersi, J.K. Tsotsos, Region classification for robust floor detection in indoor environments, in: Image Analysis and Recognition, Springer, 2009, pp. 717–726.
[4] Z. Ullah, et al., Applications of artificial intelligence and machine learning in smart cities, Elsevier Comput. Commun. J. 154 (2020) 313–323.
[5] C. Harris, M. Stephens, A combined corner and edge detector, in: Alvey Vision Conference, Vol. 15, Citeseer, 1988, p. 50.
[6] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[7] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (SURF), Comput. Vis. Image Underst. 110 (3) (2008) 346–359.
[8] E. Rosten, R. Porter, T. Drummond, Faster and better: A machine learning approach to corner detection, IEEE Trans. Pattern Anal. Mach. Intell. 32 (1) (2010) 105–119.
[9] M. Agrawal, K. Konolige, M.R. Blas, Censure: Center surround extremas for realtime feature detection and matching, in: Computer Vision–ECCV 2008, Springer, 2008, pp. 102–115.
[10] M. Calonder, V. Lepetit, C. Strecha, P. Fua, Brief: Binary robust independent elementary features, in: Computer Vision–ECCV 2010, 2010, pp. 778–792.
[11] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: an efficient alternative to SIFT or SURF, in: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011, pp. 2564–2571.
[12] D. Scaramuzza, F. Fraundorfer, Visual odometry [tutorial], Robot. Autom. Mag. IEEE 18 (4) (2011) 80–92.
[13] M.Z. Hasan, et al., Analysis of cross-layer design of quality-of-service forward geographic wireless sensor network routing strategies in green internet of things, IEEE Access J. 6 (1) (2018) 20371–20389.
[14] F. Al-Turjman, A novel approach for drones positioning in mission critical applications, Wiley Trans. Emerg. Telecommun. Technol. (2019) http://dx.doi.org/10.1002/ett.3603.
[15] D. Tick, A.C. Satici, J. Shen, N. Gans, Tracking control of mobile robots localized via chained fusion of discrete and continuous epipolar geometry, IMU and odometry, IEEE Trans. Cybern. 43 (4) (2013) 1237–1250.
[16] I.E. Livieris, An advanced active set L-BFGS algorithm for training weight-constrained neural networks, Neural Comput. Appl. 32 (2020) 6669–6684.

[17] J. Otegui, A. Bahillo, I. Lopetegi, L.E. Diez, Simulation framework for testing train navigation algorithms based on 9-DOF-IMU and tachometers, IEEE Trans. Instrum. Meas. PP (99) (2019) 1.
[18] A. Mourikis, N. Trawny, S. Roumeliotis, A.E. Johnson, A. Ansar, L. Matthies, Vision-aided inertial navigation for spacecraft entry, descent, and landing, IEEE Trans. Robot. 25 (2) (2009) 264–280.
[19] G. Chowdhary, E.N. Johnson, D. Magree, A. Wu, A. Shein, GPS-denied indoor and outdoor monocular vision aided navigation and control of unmanned aircraft, J. Field Robotics 30 (3) (2013) 415–438.
[20] B. Jung, G.S. Sukhatme, Real-time motion tracking from a mobile robot, Int. J. Soc. Robot. 2 (1) (2010) 63–78.
[21] M. Hwangbo, J.-S. Kim, T. Kanade, Gyro-aided feature tracking for a moving camera: fusion, auto-calibration and GPU implementation, Int. J. Robot. Res. (2011) 0278364911416391.
[22] M. Liu, Y. Wang, H. Leung, J. Yu, A novel feature-level data fusion method for indoor autonomous localization, Math. Probl. Eng. 2013 (2013).
[23] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, Int. J. Robot. Res. 5 (1) (1986) 90–98.
[24] Z. Jia, A. Balasuriya, S. Challa, Sensor fusion-based visual target tracking for autonomous vehicles with the out-of-sequence measurements solution, Robot. Auton. Syst. 56 (2) (2008) 157–176.
[25] Deepak Kumar Jain, Akshi Kumar, Geetanjali Garg, Sarcasm detection in mash-up language using soft-attention based bi-directional LSTM and feature-rich CNN, Appl. Soft Comput. 91 (2020) 106198.
[26] L. Xie, S. Xue, J. Zhang, M. Zhang, W. Tian, S. Haugen, A path planning approach based on multi-direction A algorithm for ships navigating within wind farm waters, Ocean Eng. 184 (JUL.15) (2019) 311–322.
[27] J.D. Hol, T.B. Schön, H. Luinge, P.J. Slycke, F. Gustafsson, Robust real-time tracking by fusing measurements from inertial and vision sensors, J. Real-Time Image Process. 2 (2–3) (2007) 149–160.
[28] A. Cherubini, F. Chaumette, Visual navigation of a mobile robot with laser-based collision avoidance, Int. J. Robot. Res. 32 (2) (2013) 189–205.
[29] J. Ren, K.A. McIsaac, R.V. Patel, Modified Newton's method applied to potential field-based navigation for nonholonomic robots in dynamic environments, Robotica 26 (01) (2008) 117–127.
[30] A.A. Masoud, A harmonic potential approach for simultaneous planning and control of a generic UAV platform, J. Intell. Robot. Syst. 65 (1–4) (2012) 153–173.
[31] Z. Zhang, A flexible new technique for camera calibration, IEEE Trans. Pattern Anal. Mach. Intell. 22 (11) (2000) 1330–1334.
[32] G. Bradski, A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly Media, Inc., 2008.

Yu Shibing, a master's degree from the School of Software of Beijing University of Aeronautics and Astronautics, has been engaged in the software development of automation control in enterprises, and entered the Institute of Mechanical Engineering and Automation of Shanghai University in 2012 to study for doctorate, engaged in machine vision, image processing and tracking research.

Jiang Zheng, Professor of Shanghai University, doctoral supervisor. Graduated from Nanjing Aviation Institute. Mainly engaged in, robot related technology research, computer aided design, and image processing research and so on.